

# **EVERY**

**Integrasjonsgrensesnitt basert på  
webservices mot EVERYs Innholdstjenester**

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

## Innholdsfortegnelse

<b>1</b>	<b><i>Innledning</i></b> .....	<b>3</b>
1.1	<b>Formål</b> .....	<b>3</b>
1.2	<b>Definisjoner og forkortelser</b> .....	<b>3</b>
1.3	<b>Relaterte dokumenter</b> .....	<b>3</b>
<b>2</b>	<b><i>Web-services mot Innholdstjenester</i></b> .....	<b>4</b>
2.1	<b>Teknologisk rammeverk</b> .....	<b>5</b>
2.1.1	XML .....	5
2.1.2	Web-services .....	6
2.1.3	Enkoding.....	6
2.2	<b>Sikkerhet og autorisasjon</b> .....	<b>7</b>
2.3	<b>Navnestandard</b> .....	<b>7</b>
2.3.1	Navnekonvensjon på tjenester og produkter .....	7
2.4	<b>Spesifikasjon av web-services</b> .....	<b>8</b>
2.4.1	Autentisering og sesjonshåndtering .....	8
2.4.2	Transaksjonsinformasjon og tracing .....	8
2.4.3	Feilhåndtering .....	10
2.4.4	Versjonshåndtering .....	11
2.4.5	Validering .....	13
2.4.6	Nyeste versjon av WSDL og XML Schema-dokumenter.....	13
2.5	<b>Retningslinjer for integrasjon</b> .....	<b>14</b>
2.5.1	Testklient .....	14
2.5.2	Auto-login.....	14
2.5.2.1	Utelatelse av sesjonsid.....	16
2.5.3	GUI-tilpasset autentisering .....	16
2.5.4	Kodeeksempler .....	17

EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

# 1 Innledning

EVERY tilbyr en ny teknologisk plattform for Innholdstjenester, betegnet ” EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester”, i dette dokumentet betegnet som Integrasjonsgrensesnittet.

Dette dokumentet spesifiserer integrasjonsgrensesnittet for web-servicene som tilbys til EVERYs kunder.

## 1.1 Formål

Formålet med dette dokumentet er å dokumentere web-services-grensesnittene for kunder/konsumenter som skal ta disse ibruk.

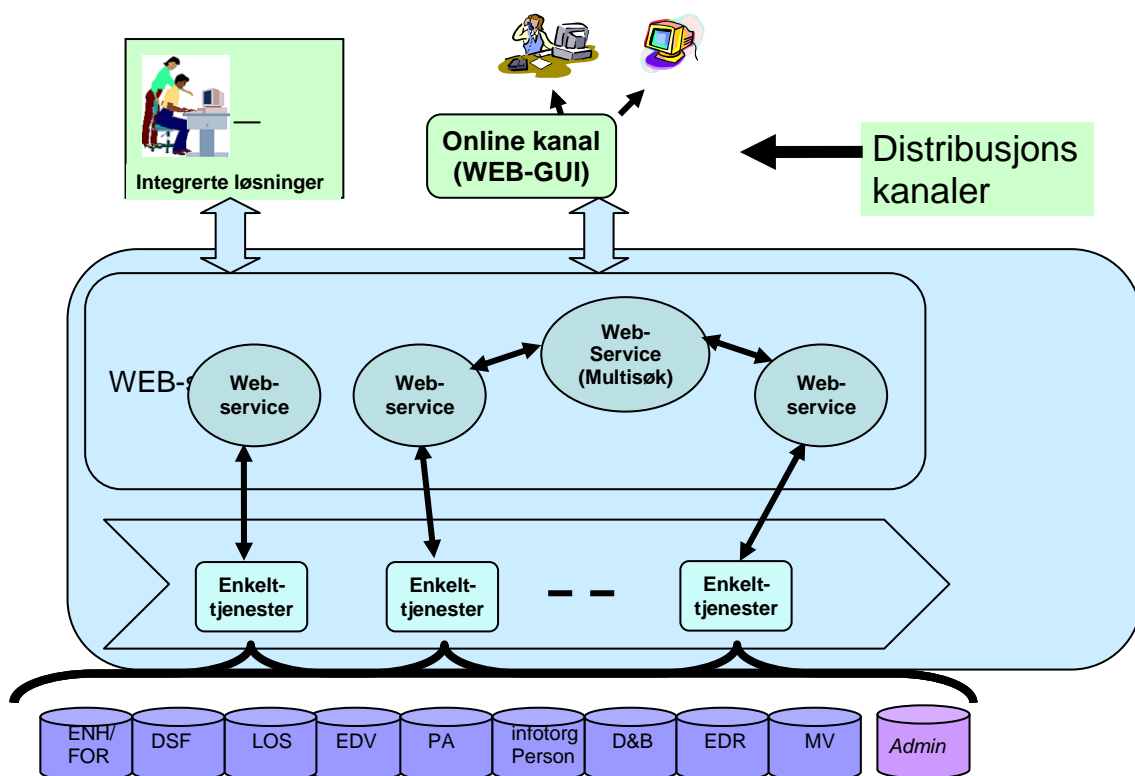
## 1.2 Definisjoner og forkortelser

Begrep	Beskrivelse
Innholdsleverandør	Juridisk enhet som leverer informasjon/innhold. Innholdet formidles gjennom integrasjonsgrensesnittet.
Datakilde	Register med innhold som formidles av innholdsleverandør.
Konsument	En ”klient” som kaller/bruker en web-service. Klienten kan også være en web-service.
Produkt	Produkt er en enhet som prissettes. Dette kan være en oppslagsfunksjon i et register, en rapport til kunde eller et element i et ajourholdsoppdrag.
SesjonsID	Unik ID som identifiserer en sesjon.
Tjeneste	En tjeneste består av ett eller flere produkter. Eksempler på tjenester er Enhetsregisteret og Postens Adresseregister (PA).
UTF	Unicode Transformation Format. Internasjonal standard for enkoding av Unicode-tegnsettet, et standard tegnsett for alle datamaskiner som støtter alle språk som er i praktisk bruk.
W3C	The World Wide Web Consortium, et konsortium som utvikler utvekslingsvennlige teknologier (spesifikasjoner, retningslinjer, programvare, og verktøy) for å påvirke utviklingen av Internett
WS-I	Web Services Interoperability Organization.
WSDL	Et XML-basert språk som forteller om web-services og hvordan de skal brukes. Et WSDL-dokument kan betraktes som en kontrakt mellom den som skal bruke en web-service (konsumenten/klienten) og tilbydereren av web-servicen (Innholdstjenester).
XML	XML (eXtensible Markup Language) er en W3C-anbefaling for å strukturere data eller beskrivelser av data (metadata) i elementer ved å bruke tekstkoding eller markeringskoder kalt tagger. Kilde: Wikipedia.
XML Schema	Et XML schema-språk. XML schema-dokumenter brukes til å validere XML-dokumenter med hensyn til struktur og innholdsformater.
XSD	Forkortelse for XML Schema Definition, som er et XML Schema-dokument. Beskriver og angir regler for innholdet i XML-dokumenter.

## 1.3 Relaterte dokumenter

[WS-I Basic Profile 1.1]: <http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html>

## 2 Web-services mot Innholdstjenester



**Figur 1 – Systemskisse**

Systemskissen over viser EVERYS tekniske plattform for levering av innholdstjenester som web-services.

Web-servicene som tilbys beskrives ved hjelp av WSDL.

Det er definert én WSDL per tjeneste. Dette for å unngå at klientene må oppgradere med ny WSDL ved endringer i tjenester som klienten ikke bruker. Med tjenester menes her tjenester som tilbyr informasjon fra ulike innholdsleverandører, samt en intern "Admin"-tjeneste som benyttes for av- og pålogging, bytte av passord, og henting av debiteringsgrunnlag.

Web-servicenes grensesnitt baseres på *WS-I Basic Profile 1.1*, som er en samling retningslinjer for å sikre at standarder for web-services virker sammen.

Kapittel 2.1 med underseksjoner beskriver mer konkret hvordan Basic Profile 1.1 benyttes i denne løsningen.

Det benyttes egendefinerte SOAP-headere til håndtering av kontekst-informasjon (autentisering / sesjonshåndtering og transaksjonsinformasjon), slik at body-delen av SOAP-meldingene kun inneholder data fra de respektive innholdsleverandørene (rene "nyttedata").

Håndtering av sikkerhet og autorisasjon er beskrevet i kap. 2.2.

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

Navnestandard er beskrevet i kap. 2.3.

Generelle aspekter rundt web-services – autentisering/sesjonshåndtering, tracing, feilhåndtering og versjonshåndtering – er spesifisert i kapittel 2.4

De konkrete web-servicene som tilbys er spesifisert med WSDL og tilhørende XML Schema-dokumenter under <http://ws-test.infotorg.no/> (testmiljø) og <http://ws.infotorg.no/> (produksjonsmiljø).

## 2.1 Teknologisk rammeverk

Web-servicene som tilbys i integrasjonsgrensesnittet baseres som nevnt på WS-I Basic Profile 1.1. Denne profilen består av et sett spesifikasjoner for web-services med tilhørende avklaringer, presiseringer og utvidelser rundt bruken av disse spesifikasjonene.

De viktigste spesifikasjonene i WS-I Basic Profile 1.1 er disse:

- WSDL 1.1
- XML Schema 1.0
- SOAP 1.1
- XML 1.0
- HTTP 1.1

I de følgende underkapitler utdypes bruken av Basic Profile 1.1 i forbindelse med integrasjonsgrensesnittet.

### 2.1.1 XML

Både SOAP-meldingene, som utgjør de faktiske dataene som utveksles, og WSDL-dokumentene, som beskriver web-servicene, er basert på XML.

Det er lagt vekt på at XML med tilhørende teknologier brukes på en standardisert måte. Videre er det lagt opp til en fleksibel dokument-struktur i XML-sammenheng for å sikre en smidig håndtering av fremtidige endringer.

Her følger noen retningslinjer for bruk av XML-teknologier som er benyttet i forbindelse med integrasjonsgrensesnittet:

- Det benyttes XML Schema for beskrivelse av dokumentstruktur og syntaksregler til alle XML-dokumenter som formidles via integrasjonsgrensesnittet.
- Schemaene vil i størst mulig grad være modulære, det vil si at der hvor to eller flere schemaer har overlappende beskrivelser, skilles overlappende beskrivelse (understrukturer) ut i egne XSD-filer for inkludering/import<sup>1</sup>.
- Det benyttes *child*-elementer fremfor attributter i dokumentstrukturen, med mindre man er sikker på at det ikke vil komme undernivåer av elementet, og hvor bruk av *child*-elementer ikke får implikasjoner for ytelsen.
- Det unngås lange elementnavn i tilfeller hvor det kan forekomme store dokumenter / mange gjentakelser. Størrelsen på XML-dokumentene påvirker ytelsen i forbindelse med parsing og transformering. Eventuelle forkortelser dokumenteres i kommentar.
- Rekursiv bruk av elementer unngås i størst mulig grad.

<sup>1</sup> `include` brukes når det refereres til schema med samme namespace, mens `import` benyttes når det refererte schema bruker et annet namespace.

EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

- Schema-elementet *any* brukes i XML Schema-dokumentene for å sikre enklere fremtidige utvidelser (nye elementer kan da legges til i en datastruktur uten at XML-schemaet gir feil under validering). Se for øvrig seksjon 2.4.4 Versjonshåndtering.
- I forbindelse med bruk av arrays i XML-dokumentene så skilles det mellom null-arrays og tomme arrays ved å legge et element utenpå arrayet, hvor dette elementet er nillable dersom det er aktuelt med null-arrays.  
Eksempel:

```
<complexType name="info">
  <sequence>
    <element name="detaljer" nillable="true"/>
    <complexType>
      <sequence>
        <element name="detalj" minOccurs="0" maxOccurs="unbounded">
          ...
        </sequence>
      </complexType>
    </element>
  </sequence>
</complexType>
```

Ved null-array blir da XML-dokumentet slik:

```
<info>
  <detaljer xsi:nil="true"/>
</info>
```

Mens ved tomt array blir XML-dokumentet slik:

```
<info>
  <detaljer/>
</info>
```

- I tilfelle dato-elementer med verdi 'null', brukes `xsi:nil="true"` for å angi verdi.

## 2.1.2 Web-services

Det benyttes *document/literal*-stil for SOAP-meldingene, det vil si at body-delen angis som XML (i motsetning til kodet tekststreng), og beskrives ved hjelp av XML Schema. Schemaene legges i egne .xsd-filer (ikke som *embedded* informasjon i WSDL-filene).

Web-servicene som tilbys er synkrone, det vil si at spørringer besvares direkte av web-servicene.

## 2.1.3 Enkoding

Det benyttes UTF-8 enkoding, både for SOAP og WSDL.

```
<?xml version="1.0" encoding="UTF-8"?>
```

UTF-8 er for øvrig default, og kan derfor utelates (`<?xml version="1.0"?>`).

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

## 2.2 Sikkerhet og autorisasjon

Autentisering ivaretas ved bruk av brukernavn og passord.

Autorisering ivaretas gjennom autorisasjonsinformasjon. Denne informasjonen hentes fra back-office-systemet, og formidles av integrasjonsgrensesnittet som svar på pålogging (slik at klienten selv vet hvilke datakilder den aktuelle brukeren har tilgang til).

I tillegg sjekkes hver forespørsel ved hjelp av autorisasjonsinformasjon fra back-office, om forespørselen er tillatt utført.

Konfidensialitet ivaretas ved å bruke SSL (HTTPS) med minimum 128-bits kryptering på meldinger som går utenfor brannmur (dvs. meldinger som går gjennom distribusjonskanalene og mot innholdsleverandørene).

## 2.3 Navnestandard

Generelt benyttes begrepene som innholdsleverandørene selv bruker, uforandret i alle svar som formidles gjennom web-servicene. Dette for å unngå vedlikehold av "mapping" mellom begreper (våre vs. leverandørenes), med tilhørende vedlikehold av dokumentasjon.

For spørringer benyttes felles begreper på tvers av tjenestene.

Følgende navnestandard benyttes i XML-dokumenter for spørringer (web-service-kall) fra brukere og kunder, samt i XML-dokumenter mot back-office:

- Norsk benyttes som språk for navngiving av elementer i XML-dokumentene.
- Norske tegn erstattes av 'æ' for æ, 'øe' for ø og 'aa' for 'å'. Tilsvarende for store bokstaver.
- Elementnavn starter med liten bokstav. Når et navn er sammensatt av flere enkeltord, markeres dette med stor forbokstav på hvert enkeltord (unntatt det første). Eksempel: `dagligLeder`.
- Datakilder, tjenester og produkter navngis med stor forbokstav, f.eks. `Brreg`, `EnhetForetak`, `Basisdata`.

Innholdsleverandørens egen navnestandard formidles i utgangspunktet transparent gjennom integrasjonsgrensesnittet/web-servicene.

For innholdsleverandører som leverer svar på XML-format, formidles dette mest mulig uforandret.

Ved spørringer på XML-format mot innholdsleverandøren, omformes disse - på samme måte som for andre tjenester - fra felles-begrepene (se over) til innholdsleverandørens format og navnestandard.

For innholdsleverandører som leverer svar på andre formater enn XML, omformes disse formatene til XML hvor leverandørens navnestandard brukes på elementnivå.

### 2.3.1 Navnekonvensjon på tjenester og produkter

Produktforespørsler navngis med "hent" etterfulgt av navn på produkt. Eksempel: `hentRoller`.

Med produktforespørsel menes her en forespørsel mot et gitt produkt hos en innholdsleverandør, med andre ord: en oppslagsfunksjon i et register. Hver forespørsel defineres som en "SOAP-message" i WSDL-filen for den aktuelle tjenesten.

Eksempler på navngiving av forespørsler mot (EnhetForetak):

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

hentBasisdata, hentSaerligeOpplysninger, hentOpplTilknRegister.

Hver tjeneste beskrives gjennom et WSDL-dokument, hvor namespace reflekterer hvem som er innholdsleverandør (kilde). Som namespace benyttes URL. Eksempel:

```
http://ws.infotorg.no/xml/Brreg/EnhetForetak/2006-05-09/EnhetForetak.wsdl
```

"Brreg" er her datakilden, mens tjenesten er "EnhetForetak". Produktene er definert i filen EnhetForetak.wsdl. For øvrig blir namespace for XSD og ikke WSDL-namespacet brukt i transaksjonene (eks: <http://ws.infotorg.no/xml/Brreg/EnhetForetak/2006-05-29/Basisdata.xsd>)

Tjenestenavn er unike, også på tvers av datakilder.

## 2.4 Spesifikasjon av web-services

### 2.4.1 Autentisering og sesjonshåndtering

Autentisering overfor web-servicene gjøres ved hjelp av brukernavn og passord. Brukernavn og passord kan sendes med som parametere i loggInn-funksjonen, sammen med distribusjonskanal (PTP) og systemnavn (navn på integrerende system). Disse parametere kan også formidles i elementet "Brukersesjon" i SOAP-header i forbindelse med automatisk pålogging ved såkalt auto-login. Auto-login er foretrukket metode ved integrasjon fra fagsystemer, da dette gir automatisk gjenoppretting av sesjon i tilfelle sesjonsbrudd.

Sesjoner håndteres ved hjelp av sesjonsID. SesjonsID formidles, på samme måte som distribusjonskanal, systemnavn, brukernavn og passord, i SOAP-header-elementet "Brukersesjon".

Schema Brukersesjon.xsd er dokumentert under <http://ws-test.infotorg.no/> (testmiljø) og <http://ws.infotorg.no/> (produksjonsmiljø).

### 2.4.2 Transaksjonsinformasjon og tracing

Informasjon om transaksjonene (informasjon som følger spørringer/svar mot innholdsleverandører) legges i SOAP-header.

For spørringene gjelder denne informasjonen referansenummer fra sluttbruker, samt parametere som angir hva slags transaksjonsinformasjon som ønskes returnert. Tracing (transaksjonsinformasjon) i forbindelse med web-services foregår på to nivåer:

Transaksjonsinformasjonen som returneres består av to deler:

- Sammendrag
  - Her vises referansenummer, antall transaksjoner (mer enn 1 hvis multisøk), samlet responstid og hvor mange av transaksjonene som er debitert.
- Detaljer
  - Visning av detaljer styres av `visDetaljer=true`.
  - Det er kontroll (via back-office) på hvilke brukere som har lov til å forespørre slik tracing.
  - I detaljer vises informasjon om hver enkelt transaksjon:
    - datakilde



EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

- tjeneste
- produkt
- rådata  
Selve dataene (spøringer / svar) som utveksles mot Innholdsleverandør.  
Styres av `visRaadata=true`. Det er kontroll (via back-office) på hvilke brukere som har lov til å forespørre tracing av rådata.
- Responstid (total, intern, database og datakilde).  
Styres av `visResponstid=true`. Det er kontroll (via back-office) på hvilke brukere som har lov til å forespørre om responstid.
- debitering (antall debiteringer)  
Styres av `visDebitering=true`

Schema Transaksjon.xsd er dokumentert under <http://ws-test.infotorg.no/> (testmiljø) og <http://ws.infotorg.no/> (produksjonsmiljø).

Når det gjelder tracing av SOAP mellom klient og integrasjonsgrensesnittet, kan slik tracing gjøres ved hjelp av en standard SOAP-monitor (som f.eks. Axis SOAP Monitor eller PocketSOAP).

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

### 2.4.3 Feilhåndtering

Feilhåndtering på web-service-nivå håndteres ved hjelp av `soap:Fault` (`soap:Fault` sendes i body). Det skilles mellom feil på systemnivå ("Server") og feil i forespørselen ("Client").

Server-feil indikerer at man kan forsøke å sende samme forespørsel på nytt, mens Client-feil betyr at det er feil i forespørselen og at denne derfor må korrigeres før et eventuelt nytt forsøk.

NB! Alle svar fra innholdsleverandør, inkludert feilsvar som skyldes innhold i spørringer, formidles i SOAP:Body, og er derfor ikke en del av denne feilhåndteringen.

Spesifikk informasjon om feilene spesifiseres i `detail`-elementet i `soap:Fault`, på følgende format:

```
<element name="Feil">
  <complexType>
    <sequence>
      <element name="feilgruppekode" type="string"/>
      <element name="feilgruppetekst" type="string"/>
      <element name="feilkode" type="string"/>
      <element name="feiltekst" type="string"/>
      <element name="feilmelding" type="string"/>
      <any minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

Figur 2 – XML Schema for "soap:Fault / detail".

*Feilgruppekode* og *feilgruppetekst* angir feilkode fra integrasjonsgrensesnittet, mens *feilkode* og *feiltekst* angir nærmere spesifisering innenfor hver feilgruppe.

*Feilmelding* inneholder spesifikk informasjon om den konkrete feilen.

Schema Feil.xsd er dokumentert under <http://ws-test.infotorg.no/> (testmiljø) og <http://ws.infotorg.no/> (produksjonsmiljø).

Følgende verdier er definert for *feilgruppekode*, *feilgruppetekst*, *feilkode* og *feiltekst*.

Feilgruppekode	Nivå	Feilgruppetekst																						
autentisering	Client	Feil ved autentisering																						
		<table border="1"> <thead> <tr> <th>Feilkode</th> <th>Feiltekst</th> </tr> </thead> <tbody> <tr> <td>sesjonsid</td> <td>Sesjonsid er ugyldig eller har løpt ut på tid. Logg inn på nytt</td> </tr> <tr> <td>brukerid</td> <td>Feil brukernavn / passord</td> </tr> <tr> <td>passordsperr</td> <td>Feil passord er oppgitt for mange ganger, brukeren er sperret</td> </tr> <tr> <td>passordutloep</td> <td>Passordet må byttes</td> </tr> <tr> <td>brukerutloep</td> <td>Brukeren er utløpt</td> </tr> <tr> <td>brukersperre</td> <td>Brukeridenten er sperret</td> </tr> <tr> <td>kundesperre</td> <td>Kunden er sperret</td> </tr> <tr> <td>nyttPassord</td> <td>Nytt passord ikke godkjent</td> </tr> <tr> <td>endretPassord</td> <td>Passord kan ikke skiftes mer enn en gang per dag</td> </tr> <tr> <td>enbruker</td> <td>Kun en enkelt innlogging er tillatt for denne brukeren</td> </tr> </tbody> </table>	Feilkode	Feiltekst	sesjonsid	Sesjonsid er ugyldig eller har løpt ut på tid. Logg inn på nytt	brukerid	Feil brukernavn / passord	passordsperr	Feil passord er oppgitt for mange ganger, brukeren er sperret	passordutloep	Passordet må byttes	brukerutloep	Brukeren er utløpt	brukersperre	Brukeridenten er sperret	kundesperre	Kunden er sperret	nyttPassord	Nytt passord ikke godkjent	endretPassord	Passord kan ikke skiftes mer enn en gang per dag	enbruker	Kun en enkelt innlogging er tillatt for denne brukeren
Feilkode	Feiltekst																							
sesjonsid	Sesjonsid er ugyldig eller har løpt ut på tid. Logg inn på nytt																							
brukerid	Feil brukernavn / passord																							
passordsperr	Feil passord er oppgitt for mange ganger, brukeren er sperret																							
passordutloep	Passordet må byttes																							
brukerutloep	Brukeren er utløpt																							
brukersperre	Brukeridenten er sperret																							
kundesperre	Kunden er sperret																							
nyttPassord	Nytt passord ikke godkjent																							
endretPassord	Passord kan ikke skiftes mer enn en gang per dag																							
enbruker	Kun en enkelt innlogging er tillatt for denne brukeren																							
Autorisasjon	Client	Ikke tilgang																						

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

Feilgruppekode	Nivå	Feilgruppetekst	
		Feilkode	Feiltekst
		produkt	Ikke autorisert for forespurt tjeneste / produkt
		kreditt	Mangler kreditt
klient	Client	Feil i klientforespørsel	
		Feilkode	Feiltekst
		raaddata	Feil i rådata
		xml	Feil i XML-format
		soapversjon	Feil SOAP-versjon
		envelope	Feil i SOAP Envelope
		ukjentHeader	mustUnderstand="1" i ukjent header
		ukjentAutentisering	Finner ikke kjent autentiseringselement. Kontroller namespace og element i SOAP Header
		ugyldigAutentisering	Ugyldig autentiseringsforespørsel. Kontroller elementinnhold i SOAP Header
		ukjentSpoerring	Ukjent forespørsel. Kontroller namespace og element i SOAP Body
		ugyldigSpoerring	Ugyldig forespørsel. Kontroller elementinnhold i SOAP Body
tjener	Server	Prosesseringsfeil	
		Feilkode	Feiltekst
		fil	Feil ved filbehandling
		database	Feil ved kommunikasjon mot database
		validering	Feil ved datavalidering
		konvertering	Feil ved datakonvertering
		ukjent	Ukjent feil
backOffice	Server	Feil i Back Office	
		Feilkode	Feiltekst
		kommunikasjon	Feil i kommunikasjon mot Back Office
		timeout	Timeout fra Back Office
		holdoff	Back Office er ikke tilgjengelig, prøv igjen senere
datakilde	Server	Transaksjonsfeil	
		Feilkode	Feiltekst
		brukerid	Brukerid finnes ikke
		kommunikasjon	Feil i kommunikasjon mot dataleverandør
		timeout	Timeout fra dataleverandør
		holdoff	Tjenesten er ikke tilgjengelig, prøv igjen senere
		throttle	For mange samtidige forespørsler mot tjenesten, prøv igjen senere
		autorisasjon	Ikke autorisert hos dataleverandør
		svarfeil	Svarfeil fra dataleverandør
		ugyldigSvar	Feil i svar fra dataleverandør

#### 2.4.4 Versjonshåndtering

Det finnes ingen standard som spesifiserer hvordan nye versjoner av web-services skal håndteres. En ny versjon av en web-service er enten tilbakekompatibel med den forrige, og kan dermed fortsette å utveksle data med en gammel klient, eller så er den inkompatibel med den gamle versjonen og må betraktes som en ny web-service.

Hver enkelt web-service er spesifisert gjennom et WSDL-dokument. Siden det brukes *document/literal*-koding, det vil si ett innparameter og ett returparameter ("dokument inn" og "dokument ut"), begrenses versjonsproblematikken på WSDL-nivå til operasjoner som fjernes eller gis nytt navn, i tillegg til at *XML namespace* for schemaer byttes når endringer i en web-service krever endring i klientene – basert på følgende *namespace policy*:

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

- Det brukes *XML namespaces* i schemaene. Namespace legges på SOAP-message-nivå, hvilket gjør at alle produktnavn tilhører et namespace.
- Ved endringer i schema som ikke krever endringer i klientene, endres ikke namespace. En slik endring kan for eksempel være nye elementer som legges til i en struktur, og som klientene kan velge å forholde seg til.
- Ved endringer i schema som krever endringer i web-service-klientene, endres *namespace*. Dette gjøres ved å endre datodelen av *namespace*'ns URI. Eksempel:

fra

```
xmlns:tm="http://ws.infotorg.no/xml/Brreg/EnhetForetak/2006-05-29/EnhetForetak.xsd"
```

til

```
xmlns:tm="http://ws.infotorg.no/xml/Brreg/EnhetForetak/2006-06-10/EnhetForetak.xsd"
```

Integrasjonsgrensesnittet håndterer klienter som bruker både gammelt og nytt schema. Dette gjøres ved at det sjekkes hvilket namespace som er brukt når den mottar et web-service-kall, og ruter forespørselen til riktig versjon (ny eller gammel web-service).

Når alle klienter er over på ny versjon, fjernes håndtering av gammel versjon i integrasjonsgrensesnittet.

Her følger eksempler på de to endringsscenariene omtalt ovenfor:

## 1. Endringer som ikke krever endring i klientene.

```
<element name="info">
  <complexType>
    <sequence>
      <element name="element1" type="string"/>
      <element name="element2" type="string"/>
      <any minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

**Figur 3 – Eksempel: XML Schema før endring**

Dersom det legges til et element i ovenstående schema som er valgfritt for klientene å bruke, kan dette elementet bare legges til uten å endre namespace.

```
<element name="info">
  <complexType>
    <sequence>
      <element name="element1" type="string"/>
      <element name="element2" type="string"/>
      <element name="nyttElement" type="string"/>
      <any minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

**Figur 4 – Eksempel: XML Schema etter endring**

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

Integrasjonsgrensesnittet kan her håndtere forespørslene fra klientene i en og samme versjon av web-servicen, uavhengig om det nye elementet er med eller ikke.

Tilsvarende kan elementer legges til i svaret fra web-servicen til klienten, dersom det ikke er påkrevd at klienten tar hensyn til de nye elementene.

## 2. Endringer som krever endring i klientene.

Dersom en web-service endres til for eksempel å kreve et nytt element fra klientene (eller tilsvarende at et nytt element i svaret kreves håndtert av klienten), må integrasjonsgrensesnittet i en overgangsperiode håndtere både klienter som støtter ny versjon (dvs. med nytt element) så vel som klienter som fortsatt forholder seg til gammel versjon (uten nytt element).

Vi kan bruke samme eksempel som i punkt 2 ("nyttElement" i "Autentisering"), men denne gang kreves det at klienten sender med det nye feltet.

Det opprettes da en ny xsd-fil med nytt namespace, fra <http://ws.infotorg.no/xml/Admin/Brukersesjon/2006-05-29/Brukersesjon.xsd>

til f.eks.

<http://ws.infotorg.no/xml/Admin/Brukersesjon/2006-06-10/Brukersesjon.xsd>

Videre opprettes det ny WSDL, f.eks.

<http://ws.infotorg.no/xml/Admin/Brukersesjon/2006-05-29/Brukersesjon.wsdl>

Denne WSDL'en bruker det nye namespace for Brukersesjon.xsd:

<http://ws.infotorg.no/xml/Admin/Brukersesjon/2006-06-10/Brukersesjon.xsd>

Klienter som baserer seg på den nye WSDL'en vil da bruke det nye namespace i sine web-service-kall, slik at basert på namespace rutes kallet til ny eller gammel versjon av web-servicen. Når alle klienter er over på ny versjon, kan gammel versjon (og rutingen) fjernes i integrasjonsgrensesnittet.

### 2.4.5 Validering

Formatet på forespørsler (web-service-kall) fra klientene vil alltid valideres. Det er opp til klienten å validere på innhold i spørringene (som f.eks. lengde på tekstfelder, maks/min på numeriske verdier eller modulus-kontroll på fødsels- og organisasjonsnummer).

Det gjøres for de fleste tjenester også schema-validering av svar fra innholdsleverandør på vegne av klienten.

### 2.4.6 Nyeste versjon av WSDL og XML Schema-dokumenter

Lenker til siste versjon av alle WSDL'er med tilhørende XML Schemaer-dokumenter for løsningen ligger under <http://ws-test.infotorg.no/> (testmiljø) og <http://ws.infotorg.no/> (produksjonsmiljø).

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

## 2.5 Retningslinjer for integrasjon

### 2.5.1 Testklient

Under dokumentasjon av Web Services finnes også en egen testklient: <http://ws-test.infotorg.no/ws/test.pl> (testmiljø) og <http://ws.infotorg.no/ws/test.pl> (produksjonsmiljø).

Testklienten inneholder eksempler på integrasjon mot alle tjenester, og gir sammen med ovennevnte dokumentasjon et godt utgangspunkt for egen integrasjon.

### 2.5.2 Auto-login

De fleste integrasjonsløsninger er fagsystemer der integrasjonen mot infotorg.no skjer uten at sluttbrukeren har noe forhold til autentisering, autorisering, eller sesjoner mot infotorg.no. I slike løsninger anbefales bruk av såkalt auto-login.

Ved auto-login sendes autorisasjonsdata og sesjonsid med i hver melding. Ved første spørring skal sesjonsid være tom, mens senere spørringer alltid skal benytte sesjonsid returnert i siste svar (merk at ny sesjonsid kan bli returnert etter sesjonsbrudd). Dette kan enkelt implementeres ved å benytte en sesjonsvariabel som initielt er tom. Et eksempel er vist under.

Spørring 1:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:brukersesjon="http://ws-
test.infotorg.no/xml/Admin/Brukersesjon/2006-07-07/Brukersesjon.xsd"
xmlns:transaksjon="http://ws-
test.infotorg.no/xml/Admin/Transaksjon/2006-07-07/Transaksjon.xsd"
xmlns:edr="http://ws-test.infotorg.no/xml/NE/EDROnline/2008-10-
01/EDROnline.xsd">
  <soap:Header>
    <brukersesjon:Brukersesjon>
      <distribusjonskanal>PTP</distribusjonskanal>
      <systemnavn>ditt_system</systemnavn>
      <brukernavn>XXXXXXXX</brukernavn>
      <passord>YYYYYYYY</passord>
      <!--
        <delegertBrukernavn></delegertBrukernavn>
      -->
      <sesjonsid></sesjonsid>
    </brukersesjon:Brukersesjon>
    <transaksjon:Transaksjon>
      <referanse>test</referanse>
    </transaksjon:Transaksjon>
  </soap:Header>
  <soap:Body>
    <edr:hentAdresse>
      <kommunenr>0301</kommunenr>
      <gaardsnr>219</gaardsnr>
```

EVRY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

```

    <bruksnr>118</bruksnr>
    <festenr>0</festenr>
    <seksjonsnr>0</seksjonsnr>
  </edr:hentAdresse>
</soap:Body>
</soap:Envelope>

```

### Svar 1:

```

<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Header>
    <brukeresesjon:Brukeresesjon xmlns:brukeresesjon="http://ws-
test.infotorg.no/xml/Admin/Brukeresesjon/2006-07-
07/Brukeresesjon.xsd">
      <sesjonsid>1259943848.68497#1+47SXq/3nb8RUYezh0CvCAmgX0</sesjonsid>
    </brukeresesjon:Brukeresesjon>
    <transaksjon:Transaksjonsinfo xmlns:transaksjon="http://ws-
test.infotorg.no/xml/Admin/Transaksjon/2006-07-07/Transaksjon.xsd">
      <sammendrag>
        <datakilde>NE</datakilde>
        <tjeneste>EDROnline</tjeneste>
        <produkt>Adresse</produkt>
        <referanse>test</referanse>
        <antallTransaksjoner>1</antallTransaksjoner>
        <debitering>
          <debitert>0</debitert>
        </debitering>
      </sammendrag>
    </transaksjon:Transaksjonsinfo>
  </soap:Header>
  <soap:Body>
    <edr:Adresse xmlns:edr="http://ws-
test.infotorg.no/xml/NE/EDROnline/2008-10-01/EDROnline.xsd">
      <resultat fakturert="false" id="101005437540">
        <produkt kode="ED21110001">
          <info tekst="Eiendommen er seksjonert og dermed ugyldig
parameter til gjeldende produkt" kode="403"/>
          <adresser knr="0301" gnr="219" bnr="118" fnr="0" snr="0">
            <gateadresse gatenavn="ELIAS BLIX' GATE" nr="3"
gatenr="20081" tettbebygd="true" knr="0301" kommune="OSLO"
fylkenr="300" fylke="OSLO" oppdatertdato="2005-09-05">
              <bygg>
                <bygningstatus kode="TB" beskrivelse="Tatt i bruk"/>
              </bygg>
              <grunnkrets nr="1402" navn="ILA RODE 2"/>
              <postnummer nr="171" navn="OSLO"/>
              <valgkrets nr="404" navn="ILA SKOLE"/>
              <kirkesogn nr="1010701" navn="Gamle Aker"/>
              <bydel nr="4" navn="ST.HANSHAUGEN"/>
            </gateadresse>
          </adresser>
        </produkt>
      </resultat>
    </edr:Adresse>
  </soap:Body>
</soap:Envelope>

```

EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

```

    <gateadresse gatenavn="HERMAN FOSS' GATE" nr="19"
gatenr="12922" tettbebygd="true" knr="0301" kommune="OSLO"
fylkenr="300" fylke="OSLO" oppdatertdato="2005-09-05">
    <bygg>
        <bygningstatus kode="TB" beskrivelse="Tatt i bruk"/>
    </bygg>
    <grunnkrets nr="1402" navn="ILA RODE 2"/>
    <postnummer nr="171" navn="OSLO"/>
    <valgkrets nr="404" navn="ILA SKOLE"/>
    <kirkesogn nr="1010701" navn="Gamle Aker"/>
    <bydel nr="4" navn="ST.HANSHAUGEN"/>
    </gateadresse>
</adresser>
</produkt>
</resultat>
</edr:Adresse>
</soap:Body>
</soap:Envelope>

```

I neste spørring fylles returnert sesjonsid inn i elementet sesjonsid, dvs. i dette tilfellet:

```
<sesjonsid>1259943848.68497#l+47SXq/3nb8RUYezh0CvCAmgX0</sesjonsid>
```

Spørringen skal for øvrig være identisk. Sesjonsid returnert i siste svar skal alltid benyttes i neste spørring.

NB! Sesjonsid må svare til angitt brukernavn/delegertBrukernavn; sesjoner kan ikke deles mellom forskjellige brukeridenter.

### 2.5.2.1 Utelatelse av sesjonsid

Det er også mulig å utelate sesjonsid-elementet helt ved auto-login. Brukernavn og passord (bl.a.) vil da bli benyttet til å finne gjeldende intern sesjonsid.

Denne funksjonaliteten vil kunne gi noe høyere responstid, da det må gjøres et ekstra internt oppslag for å finne gjeldende sesjonsid. Dersom flere instanser av et fagsystem benytter samme brukerident vil disse instansene dele samme sesjon.

Funksjonaliteten kan forenkle integrasjonen i fagsystemer som bruker en felles innlogging mot forskjellige brukeridenter vha. såkalt delegert innlogging (benytter elementet delegertBrukernavn). Man slipper da å vedlikeholde en mapping mellom delegertBrukerid og sesjonsid.

### 2.5.3 GUI-tilpasset autentisering

Som et alternativ til auto-logon kan metoden loggInn i tjenesten Brukersesjon benyttes til å opprette en sesjon. loggInn returnerer autorisasjonsdata for brukeren, disse kan f.eks. benyttes til å konfigurere gyldige menyvalg i et GUI.



EVERY integrasjonsgrensesnitt basert på web-services for Innholdstjenester	Versjon: 1.2
	Dato: 9.12.2011

Ved å utelate overføring av brukernavn og passord i elementet Brukersesjon i SOAP Header (dvs. kun overføring av sesjonsid) sikrer man at brukerens sesjon automatisk blir avsluttet etter en periode med inaktivitet (p.t. 2 timer, eller mindre dersom brukeren har tilgang til spesielle tjenester). Funksjonaliteten kan dermed forenkle integrasjon i fagsystemer som har krav til automatisk utlogging.

NB! Merk at det blir opprettet en ny sesjon for hvert kall til loggInn; loggInn skal derfor kun kalles én gang ved oppstart av sesjonen og ikke før hver enkelt transaksjon. For å avslutte sesjonen kan loggUt kalles.

#### **2.5.4 Kodeeksempler**

Java-eksempler på integrasjon foreligger for flere tjenester og produkter: DsfDetaljer.java, EdrHjemmel.java, EdvSeksjonsliste.java, ItpFoedselsnrSoek.java

Merk at de implementerte Web Services er plattformuavhengige slik at man fritt kan velge programmeringsspråk for integrasjon.