



HSA WebService Användarhandledning

Version 2.35.0_1.1

EJ FÖR NYANSLEGNING



Innehåll

1. Dokumentinformation	9
1.1 Förkortningar	9
1.2 Referenser	9
1.3 Versioner	10
2. Användning	10
2.1 Inledning	10
2.2 URL	10
2.2.1 Miljöer	10
2.2.2 WSDL och XML-schema	11
2.3 SOAP-anrop	11
2.4 soap:header	11
2.4.1 {http://www.w3.org/2005/08/addressing}To	11
2.4.2 {http://www.w3.org/2005/08/addressing}MessageId	11
2.5 Felhantering	12
2.5.1 Detail	12
2.5.2 Felkoder	12
2.5.3 Exempel	13
2.6 Autentisering	13
2.6.1 CA	13
2.7 Behörighetshantering	14
3. Metoder	14
3.1 Generella metoder	14
3.2 Intressentspecifika metoder	15
3.3 Övriga metoder	15
3.4 Nya metoder	15
4. Metodbeskrivningar	16
4.1 GetCareUnit	16
4.1.1 Inparametrar	16
4.1.2 Returvärde	16
4.1.3 Exempel	17
4.2 GetHospPerson	19
4.2.1 Inparametrar	19



4.2.2	Returvärde	19
4.2.3	Exempel	21
4.3	GetHsaUnit	23
4.3.1	Inparametrar	24
4.3.2	Returvärde	24
4.3.3	Exempel	32
4.4	GetInformationList	38
4.4.1	Inparametrar	38
4.4.2	Returvärde	38
4.4.3	Informationslista	38
4.4.4	Exempel	40
4.4.5	Java	41
4.4.6	Microsoft .Net	47
4.5	GetMiuForPerson	52
4.5.1	Inparametrar	52
4.5.2	Returvärde	52
4.5.3	Exempel	56
4.6	GetPriceUnitsForAuth	59
4.6.1	Inparametrar	59
4.6.2	Returvärde	60
4.6.3	Exempel	61
4.7	GetCareUnitList	66
4.7.1	Inparametrar	66
4.7.2	Returvärde	67
4.7.3	Exempel	68
4.8	GetCareUnitMembers	70
4.8.1	Inparametrar	70
4.8.2	Returvärde	70
4.8.3	Exempel	72
4.9	HsaWsSimpleLookup	74
4.9.1	Inparametrar	75
4.9.2	Returvärde	76
4.9.3	Exempel	77
4.10	IsAuthorizedToSystem	79
4.10.1	Inparametrar	80



4.10.2	Returvärde	80
4.10.3	Exempel	83
4.11	Ping	87
4.11.1	Inparametrar	87
4.11.2	Returvärde	87
4.11.3	Exempel	87
4.12	VpwGetPublicUnits	89
4.12.1	Inparametrar	89
4.12.2	Returvärde	89
4.12.3	Exempel	91
4.13	GetHsaPerson	93
4.13.1	Inparametrar	93
4.13.2	Returvärde	93
4.13.3	Exempel	96
4.14	GetStatisticsCareGiver	99
4.14.1	Inparametrar	99
4.14.2	Returvärde	100
4.14.3	Exempel	100
4.15	GetStatisticsHsaUnit	102
4.15.1	Inparametrar	102
4.15.2	Returvärde	103
4.15.3	Exempel	106
4.16	GetStatisticsPerson	108
4.16.1	Inparametrar	109
4.16.2	Returvärde	109
4.16.3	Exempel	110
4.17	GetStatisticsNames	112
4.17.1	Inparametrar	112
4.17.2	Returvärde	113
4.17.3	Exempel	113
4.18	GetHospLastUpdate	115
4.18.1	Inparametrar	116
4.18.2	Returvärde	116
4.18.3	Exempel	116
4.19	HandleCertifier	117



4.19.1	Inparametrar	118
4.19.2	Returvärde	119
4.19.3	Exempel.....	119

Revisionshistorik

Version	Datum	Författare	Kommentar
1.0	2009-12-02	Henrik Båkman	Första version.
1.1	2009-12-02	Henrik Båkman	Kompletterat metodbeskrivningarna med javabaserade exempel.
1.2	2009-12-06	Henrik Båkman	Lagt till information om ny metod, GetHospPerson.
1.3	2009-12-17	Henrik Båkman	Rättat felaktig URL i avsnitt 2.2.
1.4	2010-01-15	Ulf Wahlqvist	Lagt till avsnitt 4.9 (hsawsSimpleLookup)
1.5	2010-01-25	Henrik Båkman	Lagt till information om metoderna <i>GetHsaUnit</i> , <i>VpwGetPublicUnits</i> och <i>VpwGetUnit</i> .
1.6	2010-01-29	Henrik Båkman	Uppdaterat referenslistan efter kommentarer från Johan Eltes.
1.7	2010-02-09	Henrik Båkman	Lagt till nya attribut till <i>GetHsaUnit</i> (4.2). Tagit bort metoden <i>VpwGetUnit</i> .
1.8	2010-03-22	Henrik Båkman	Uppdaterad för hsaws 2.9.1 (inga ändringar).
1.9	2010-03-31	Ronny Nilsson HSA Förvaltning	Uppdaterat enligt kommentarer från HSA Förvaltningsgrupp, främst SVR-mall och att benämningar stämmer med HSA-schemat.
1.10	2010-05-04	Henrik Båkman	Uppdaterad för hsaws 2.10.0. GetMiuForPerson (4.4) VpwGetPublicUnits (4.12)
1.11	2010-06-02	Henrik Båkman	Uppdaterad för hsaws 2.11.0. GetMiuForPerson (4.4)
1.11.1	2010-07-08	Ronny Nilsson	Bytt till Ineras dokumentmall.
1.12.0	2010-08-23	Henrik Båkman	Uppdaterad för hsaws 2.12.0. Ny metod IsAuthorizedToSystem (4.10).



			Lagt till information om behörighetshantering (2.7 och 4).
2.12.1_0.1	2010-08-24	Henrik Båkman	Lagt till nya attribut (surName) till UserInformation (0). Ändrat versionsnummer till att matcha version av hsaws. Lagt till versionsnummer för dokument <hsaws>_<dok>.
2.14.0_0.1	2010-09-27	Henrik Båkman	Uppdaterad för hsaws 2.14.0. Ny metod GetInformationList (4.4).
2.15.0_0.1	2010-12-15	Robert Lundmark	Uppdaterat UnitTemporaryInformation för GetHsaUnit metod.
2.16.0_0.1	2011-03-14	Robert Lundmark	Lagt till GetCareUnitList metod samt uppdaterat GetHsaUnit metod.
2.17.0_0.1	2011-04-20	Robert Lundmark	Lagt till GetPriceUnitsForAuth metoden samt lagt till utparameter till isAuthorizedToSystem samt lagt till ny informationslista för public persons till GetInformationList. Rättat beskrivningen av hsawsSimpleLookup metoden.
2.17.1_0.1	2011-05-07	Robert Lundmark	Lagt till DentalCareFunction som utresultat från GetPriceUnitsForAuth. Lagt till personnummer som inparameter och MiddleName som utresultat från IsAuthorizedToSystem.
2.18.0_0.1	2011-06-07	Robert Lundmark	Lagt till Pascal specifik metod GetCareUnitMembers. Lagt till personnummer som inparameter till GetMiuForPerson Last till enhetsnamn som utdata från GetCareUnitList.
2.19.0_1.0	2011-11-11	Michael Karén	Lagt till GetHsaPerson.
2.19.1_1.0	2011-11-18	Michael Karén	Rättat returtyp till enkel för hsaSwitchboardNumber och tagit bort mobileTelephoneNumbers då den är samma attribut i katalogen som mobiles. Även ändrat returtypen hsaSwitchboardNumber för isAuthorizedToSystem där det var samma fel. Uppdaterat från log4j 2.1.14 till 2.1.16.
2.19.2_1.0	2011-11-21	Michael Karén	Ändrat tillbaka returtypen hsaSwitchboardNumber för isAuthorizedToSystem för bakåtkompatibilitet.
2.19.3_1.0	2011-11-30	Michael Karén	Lagt till hsaVpwWebPage till GetHsaUnitResponseType. GetCareUnitList: Returnerar inte objekt som inte finns inom startDate och endDate om dessa attribut finns.
2.20.0_1.0	2012-01-12	Robert Lundmark	Lagt till modifyTimestamp som returnerat attribut av VpwGetPublicUnits metoden



2.20.0_2.0	2012-01-19	Robert Lundmark	Uppdaterat enligt Nilex 26705 (vårdgivare till vårdenhet)
2.21.0_1.0	2012-06-19	Robert Lundmark	Uppdaterat för ny release enligt Nilex ärenden: 23571, 27813 och 30764
2.21.1_1.0	2012-08-29	Robert Lundmark	Ändrat format för mellan och efternamn för getMiuForPerson metoden.
2.21.1_2.0	2012-09-10	Robert Lundmark	Rättat referenser som varit felaktiga eller gamla.
2.24.0_1.0	2013-04-03	Robert Lundmark	Uppdaterad för version 2.24.0
2.24.0_1.0	2013-05-23	Robert Lundmark	Uppdaterad för version 2.25.0 för nytt HSA schema 4.1
2.26.1_1.0	2013-11-06	Robert Lundmark	Uppdaterad för version 2.26.1 för metoder för Intygstjänster.
2.26.3_1.0	2013-12-12	Jon Sandelin	Uppdaterad för metoder för WebCert
2.26.3_1.1	2013-12-13	Robert Lundmark	Uppdaterat Statistikmetoder efter granskning av Inera.
2.27.0_1.0	2013-12-18	Jon Sandelin	Uppdaterat för version 2.27 – ny inparameter till GetMiuForPerson
2.28.0_1.0	2015-06-01	Robert Lundmark	Lagt till ny utparameter för alla metoder, isFeignedObject
2.28.1_1.0	2015-07-07	Robert Lundmark	Lagt till stöd för fingerad vårdgivare i getStatisticsHsaUnit.
2.28.1_1.1	2016-02-09	Henrika Littorin	Förtydligt att HSA Webservice kommer att avskaffas och ersättas av tjänstekontraktсанrop samt att inga nya tjänster ansluts via Webservice. Därtill vissa språkliga justeringar samt överfört till ny Inera-mall.
2.29.0_1.0	2016-05-26	Robert Lundmark	Reserverade funktionsnamn hämtas från kodverket cn=Reserverade funktionsnamn,dc=koder,dc=Services,c=SE. HsaWs ska nu kunna startas utan att kodverket är korrekt.
2.30.0_1.0	2016-08-25	Robert Lundmark	GetHsaUnit nytt returattribut, smsTelephoneNumber Ny hantering av rättigheter. Varje metod får sin egen ROLE_ Ny hantering av skyddade personer. Rätt att läsa skyddade personer styrs nu av klientanvändarens egna rättigheter.
2.30.1_1.0	2016-10-18	Robert Lundmark	Tilltalsnamn behöver inte längre vara obligatorisk uppgift för person. getStatisticsPerson går nu att anropa utan sökbas.
2.30.2_1.0	2016-11-18	Robert Lundmark	Ändrat läsning av personnummer för getStatisticsPerson så att klientanvändaren inte behöver ha rätt att läsa personnummer.



2.31.0_1.0	2017-01-19	Robert Lundmark	Lagt till validering av XML-svar och hantering av kodverk vid schemaändringar.
2.32.0_1.0	2017-03-10	Robert Lundmark	Cachning av kodverk. Rättning av validering av XML-svar. HTML-kodning försvann i förra versionen.
2.32.1_1.0	2017-03-29	Robert Lundmark	Validering av XML-svar flyttas till direkt när strängar läses från katalogobjekt.
2.33.0_1.0	2017-09-28	Robert Lundmark	Hantering av telefonnummer som innehåller 1 eller 2 siffror. GetHospLastUpdate med korrekt tidsangivelse.
2.33.1_1.0	2017-11-22	Robert Lundmark	Lagt till funktion för hantering av tecken ej tillåtna i XML.
2.34.0_1.0	2018-01-15	Robert Lundmark	Ändrat format för attributen surgeryHours, dropInHours, telephoneHours och visitingHours. Attributen languageKnowledgeCode och smsTelephoneNumber (från person) borttagna
2.35.0_1.0	2018-06-01	Robert Lundmark	Ny returparameter till GetHsaUnit, financingOrganization
2.35.0_1.1	2018-07-09	Ronny Nilsson	Några rättningar främst av språklig karaktär samt tillägg av från- och till-datum i TimeSpan.

EJ FÖR NYA
SÄTTNING



1. Dokumentinformation

Detta dokument innehåller användarhandledning för Inera:s funktion *HSA Webservice Riv 2.0* (hsaws2).

1.1 Förkortningar

Följande förkortningar används i detta dokument;

BIF	Bastjänster för Informationsförsörjning, tidigare benämning för Säkerhetstjänster
hsaws2	HSA Webservice 2
SKL	Sveriges kommuner och landsting
VpW	Vården på Webben, tidigare benämning för Hitta och jämför vård (1177.se)

1.2 Referenser

Följande externa referenser används i detta dokument,

[1]	HSA Innehåll Medarbetaruppdrag syfte (se www.inera.se/hsa)
[2]	HSA Innehåll Medarbetaruppdrag rättigheter (se www.inera.se/hsa)
[3]	HSA Innehåll Legitimerad Yrkesgrupp (se www.inera.se/hsa)
[4]	RIV Tekniska Anvisningar 2.0, (http://rivta.se/ , https://github.com/Aask/rivta/tree/master/wiki)
[5]	Formulär för anslutning av tjänst, http://www.inera.se/hsa/ , under Anslutning och Anslutning av tjänst
[6]	HSA-schema organisationsträdet (se www.inera.se/hsa)
[7]	HSA Innehåll Verksamhetskod (se www.inera.se/hsa)
[8]	HSA Innehåll Enhetstyp (se www.inera.se/hsa)
[9]	HSA Innehåll Vårdform (se www.inera.se/hsa)
[10]	HSA Innehåll Länskod (se www.inera.se/hsa)
[11]	HSA Innehåll Kommunkod (se www.inera.se/hsa)
[12]	HSA Innehåll Visas för (se www.inera.se/hsa)
[13]	HSA Innehåll Ägarform (se www.inera.se/hsa)
[14]	SOAP Message Transmission Optimization Mechanism (MTOM), http://www.w3.org/TR/soap12-mtom/
[15]	XML-binary Optimized Packaging, http://www.w3.org/TR/xop10/



[16]	DotNetZip, http://dotnetzip.codeplex.com/
[17]	Java Architecture for XML Binding (JAXB), http://www.oracle.com/technetwork/articles/javase/index-140168.html
[18]	ZIP, http://en.wikipedia.org/wiki/ZIP_(file_format)
[19]	Java API for XML WebServices (JAX-WS), http://docs.oracle.com/javase/7/docs/technotes/guides/xml/jax-ws/index.html
[20]	HSA Innehåll Befattning (se www.inera.se/hsa)

1.3 Versioner

Denna version av användarhandledningen är uppdaterad för hsaws2 till och med version 2.35.0. *Notera att ingen vidareutveckling sker av hsaws2 då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA FileService.*

2. Användning

I detta avsnitt ger vi en allmän beskrivning av hsaws2. *Notera att inga nya tjänster ansluts till hsaws2 då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA FileService.*

2.1 Inledning

Hsaws2 är en SOAP-baserad Webservice för hantering av information och övriga tjänster för HSA. Webservices metoder är implementerade enligt principen *wrapped document/literal*. Som transportprotokoll för SOAP-meddelanden stöds HTTPS.

Hsaws2 är anpassad mot RIV Tekniska Anvisningar 2.0 (se referens [4]) och då specifikt RIV TA Basic Profile 2.0.

2.2 URL

Hsaws2 anropas med följande URL

- <https://<miljö>/svr-hsaws2/hsaws>

2.2.1 Miljöer

Hsaws2 är implementerad i följande HSA-miljöer.

- **Produktion:** ws.hsa.sjunet.org
- **Integration:** wstest.hsa.sjunet.org



- **Test:** ws.test.hsa.sjunet.org

2.2.2 WSDL och XML-schema

Aktuell WSDL-fil och använda XML-scheman kan erhållas från Inera:s förvaltningorganisation för HSA. WSDL och schemafilerna kan även hämtas direkt från hsaws2 med följande URL.

- <https://<miljö>/svr-hsaws2/hsaws?wsdl>

2.3 SOAP-anrop

SOAP-anropen mot hsaws2 följer följande exempel.

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    ...
  </soap:Body>
</soap:Envelope>
```

För detaljerad information om soap:Body se respektive metodbeskrivning längre fram i detta dokument.

2.4 soap:header

Följande header-parametrar används för alla metoder.

2.4.1 {<http://www.w3.org/2005/08/addressing>}To

Denna parameter är specificerad av RIV TA 2.0 (se referens [4]) och anger HSA-identiteten för mottagaren av SOAP-anropet.

För hsaws2 skall denna parameter alltid sättas till följande värde.

- SE165565594230-1000 (Inera AB)

2.4.2 {<http://www.w3.org/2005/08/addressing>}MessageId

Denna parameter är till för korrelering av loggposter mellan anropande system och hsaws2 för enklare felsökning. Meningen är att anropande system genererar en för aktuellt anrop unik identitet. Denna identitet används sedan av hsaws2 för att märka upp loggposter för aktuellt anrop.



2.5 Felhantering

I det fall ett fel har uppstått i ett metodanrop returneras en så kallad *WebService fault*. Felmeddelandet innehåller följande information.

Parameter	Datotyp	Beskrivning
faultcode	String	SOAP-baserad felkod. Är alltid satt till <i>soap:Server</i> .
faultstring	String	Generellt felmeddelande. Ger normalt information om vilken metod som anropats.
detail	Se 2.5.1	Ger information om det fel som har uppstått.

2.5.1 Detail

Detaljerad information om aktuellt fel ges i felparametern *detail*. Denna parameter består av en instans av XML-typen *HsaWsFault*. *HsaWsFault* innehåller följande parametrar.

Parameter	Datotyp	Beskrivning
code	Integer	Felkod. Se nedanstående tabell.
message	String	Felmeddelande kopplat till aktuellt fel.

2.5.2 Felkoder

Följande felkoder används för parametern *code* i *HsaWsFault*.

Felkod	Beskrivning
1	Okänt fel. Kontakta Inera:s förvaltningsorganisation för HSA om felet består [5].
2	Internt serverfel. Kontakta Inera:s förvaltningsorganisation för HSA om felet består [5].
3	Ogiltig inparameter till metodanropet. För mer information se felparametern <i>message</i> .
4	Internt fel i användning av HSA kodtabell. Kontakta Inera:s förvaltningsorganisation för HSA om felet består [5].
5	Metodanropet har givit upphov till ett ogiltigt domänobjekt. För mer information se felparametern <i>message</i> .
6	Internt fel vid sökning i HSA-katalogen. Kontakta Inera:s förvaltningsorganisation för HSA om felet består [5]. OBS! Exempelvis om en icke existerande sökbas anges returneras denna felkod.



7	LDAP-objekt kan inte hittas i HSA-katalogen. För mer information se felparametern <i>message</i> .
8	Fel i informationsinnehållet i HSA-katalogen. För mer information se felparametern <i>message</i> .

2.5.3 Exempel

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>
        Error executing getMiuForPerson()
      </faultstring>
      <detail>
        <HsaWsFault
          xmlns="urn:riv:hsa:HsaWsResponder:3"
          xmlns:ns2="http://www.w3.org/2005/08/addressing">
          <code>7</code>
          <message>
            LDAP object not found: SE1000-807/null
          </message>
        </HsaWsFault>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

2.6 Autentisering

Notera att inga nya tjänster ansluts till *hsaws2* då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA *FileService*. Kontakta HSA Förvaltning via formulär för anslutning av tjänst [5].

Hsaws2 använder HTTPS (SSL) med krav på klientcertifikat. För använt klientcertifikat skall följande gälla.

- Anropande systems HSA-id skall anges i attributet *serialNumber* i certifikatets attribut *subject*.
- Anropande system måste finnas upplagd i HSA-katalogen med aktuellt HSA-id.

2.6.1 CA

Klientcertifikat från följande certifikatsutgivare kan användas med *hsaws2*.



Produktion

- SITHS Type3 CA v1

Testmiljöer

- SITHS Type3 CA v1
- SITHS Type3 CA v1 PP

2.7 Behörighetshantering

Notera att inga nya tjänster ansluts till hsaws2 då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA FileService. Kontakta HSA Förvaltning via formulär för anslutning av tjänst [5].

Förutom att anropande system måste kunna autentisera sig mot hsaws2 enligt kraven i föregående avsnitt måste även anropande system ha behörighet att använda hsaws2 och anropad metod. Vissa metoder kräver ingen extra behörighet utöver behörighet att anropa hsaws2 medan andra metoder kräver utökad behörighet. För mer information se respektive metodbeskrivning i avsnitt 4.

3. Metoder

Följande metoder tillhandahålls av hsaws2 i nuvarande version. *Notera att inga nya tjänster ansluts till hsaws2 då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA FileService. Tjänster som redan använder hsaws2 kan i undantagsfall beviljas ytterligare metoder under en övergångsperiod. Kontakta HSA Förvaltning via formulär för anslutning av tjänst [5].*

3.1 Generella metoder

Följande metoder är generella och på så sätt användbara för alla externa intressenter.

- **GetCareUnit:** Information om vårdenhet och vårdgivare för specificerad enhet. Se avsnitt 4.1.
- **GetHsaUnit:** Generell information för en enhet. Se avsnitt 4.3.
- **GetHospPerson:** Information om en så kallad HOSP-person (Socialstyrelsens personregister) i HSA-katalogen. Se avsnitt 4.2.
- **GetInformationList:** Hämtning av informationslista. Se avsnitt 4.4.
- **IsAuthorizedToSystem:** Kontrollerar om en person har behörighet till ett system (enligt attributet *hsaSystemRole*). Om så är fallet returneras information om personen. Se avsnitt 4.10.



- **GetHsaPerson:** Hämtar information om en HSA-person(hsaPersonExtension). Se avsnitt 4.13
- **GetHospLastUpdate:** Hämtar tidpunkten då senaste HOSP-inläsning gjordes (dvs senaste tidpunkten då ny HOSP-data fanns tillgänglig).

3.2 Intressentspecifika metoder

Följande metoder är framtagna och anpassade för specifika intressenter.

- **GetMiuForPerson:** Information om behörighetsegenskaper för specificerad person. Se avsnitt 4.5. Denna metod är framtagen för BIF.
- **GetCareUnitList:** Hämtar en lista med alla HsaUnits som tillhör en CareUnit. Denna metod är framtagen för BIF.
- **VpwGetPublicUnits:** Hämta en förgenererad lista på alla publika enheter enligt specifikation för VpW. Se avsnitt 4.12. Denna metod är framtagen för VpW.
- **GetPriceUnitsForAuth:** Hämta en lista med publika tandvårdsenheter, funktioner och deras publika personer som har vissa yrkesroller. Denna metod är framtagen för VpW.
- **GetCareUnitMembers:** Hämta lista med alla kopplade enheter till en vårdenhet. Denna metod är framtagen för Pascal.
- **GetStatisticsCareGiver:** Hämta information om vårdgivare. Denna metod är framtagen för Intygstjänster.
- **GetStatisticsHsaUnit:** Hämta information om enhet. Denna metod är framtagen för Intygstjänster.
- **GetStatisticsPerson:** Hämta information om person. Denna metod är framtagen för Intygstjänster.
- **GetStatisticsNames:** Hämta information om namn för intygsutfärdare. Denna metod är framtagen för Intygstjänster.
- **HandleCertifier:** Registrera eller avregistrera privat intygsutfärdare. Denna metod är framtagen för WebCert.

3.3 Övriga metoder

- **hsaWsSimpleLookup:** Generell sökning efter LDAP-objekt i HSA-katalogen. Se avsnitt 4.9. *Ingen ny användning av denna metod beviljas.*
- **Ping:** Test av att hsaws2 är tillgänglig och att sökning kan utföras i den bakomliggande HSA-katalogen. Se avsnitt 4.11. Denna metod är framför allt avsedd för övervakning och felsökning.

3.4 Nya metoder



Externa intressenter som vill komma åt information i HSA-katalogen och utföra andra HSA-tjänster skall i första hand använda befintliga tjänstekontrakt eller FileService-filer. Vår målsättning är att dessa metoder i förlängningen skall uppfylla de flesta intressenters behov av HSA-information.

Det kan dock alltid finnas fall där en generell metod inte riktigt uppfyller en intressents behov. Det kan även finnas fall där en intressent av olika skäl har behov av en intressentspecifik metod. Grundprincipen här är att i första hand utöka funktionaliteten hos en befintlig generell metod eller implementera en ny generell metod. Behovet kan dock vara så pass specifikt att en ny intressentspecifik metod måste utvecklas.

Om det finns ett behov av att komplettera en befintlig generell metod eller en ny intressentspecifik metod önskas, kontakta HSA Förvaltning [5]. *Notera att inga nya metoder inom hsaws2 utvecklas då denna anropsmetod mot HSA ska ersättas av anrop via tjänstekontrakt eller HSA FileService.*

4. Metodbeskrivningar

I de följande avsnitten ger vi detaljerade beskrivningar av varje metod i hsaws2.

4.1 GetCareUnit

GetCareUnit används för att söka ut information om den vårdenhet som en specificerad enhet tillhör. Vidare returneras information om den vårdgivare utsökt vårdenhet tillhör.

4.1.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsalentity	String	1 - 1	HSA-identitet på den enhet vars vårdenhet skall sökas ut.
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används c=SE som sökbas.

4.1.2 Returvärde

- Om specificerad enhet tillhör en vårdenhet returneras information om vårdenheten i objekt av XML-typen *GetCareUnitResponse*.
- Om enheten inte tillhör vårdenhet returneras *null*.

GetCareUnitResponse

Parameter	Datotyp	Antal	Beskrivning
careUnitHsalentity	String	1 - 1	Vårdenhetens HSA-identitet
HSA: HSA-Id LDAP: hsalentity			



careUnitStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdenhetens verksamhet.
careUnitEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdenhetens verksamhet.
careGiver HSA: HSA-Id LDAP: hsaResponsibleHealth CareProvider	String	0 - 1	HSA-identitet på den vårdgivare som aktuell vårdenhet tillhör.
careGiverStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdgivarens verksamhet.
careGiverEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdgivarens verksamhet.
careUnitFeignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdenheten är ett fingerat objekt.
careGiverFeignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdegivaren är ett fingerat objekt.

4.1.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnit*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
```



```
</soap:Header>
<soap:Body>
  <urn:GetCareUnit>
    <urn:hsaIdentity>SE1000-003</urn:hsaIdentity>
  </urn:GetCareUnit>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetCareUnitResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <careUnitHsaIdentity>SE1000-002</careUnitHsaIdentity>
      <careGiver>SE1000-001</careGiver>
    </GetCareUnitResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnit*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
```



```
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1000-003");

// Call hsaws
GetCareUnitResponseType response =
    hsaws.getCareUnit(logicalAddress, id, parameters);

// Handle response
System.out.println("Care unit: " +
    response.getCareUnitHsaIdentity());
System.out.println("Care giver: " + response.getCareGiver());
```

4.2 GetHospPerson

GetHospPerson används för att söka efter så kallade HOSP-personer (Socialstyrelsens personregister) i HSA-katalogen.

För att kunna läsa ut HOSP personer krävs dessutom att anropande system har behörighet att läsa HOSP data. Detta ges genom att lägga till anropande system i behörighetsgruppen för HOSP användare.

I den centrala katalogen heter användargruppen för HOSP användare hosp_users men detta är helt beroende av hur behörigheter är definierade i den aktuella katalogen.

4.2.1 Inparametrar

Parameter	Datatyp	Antal	Beskrivning
personallIdentityNumber	String	1 – 1	Personnummer på den HOSP-person som skall sökas ut.

4.2.2 Returvärde

- Om eftersökt HOSP-person finns i HSA-katalogen returneras information om personen i element av XML-typen *GetHospPersonResponseType*.
- Om ingen HOSP-person hittas returneras *null*.



GetHospPersonResponseType

Parameter	Datotyp	Antal	Beskrivning
personallIdentityNumber HSA: Person-id LDAP: personallIdentityNumber	String	1 – 1	Personnummer
personalPrescriptionCode HSA: Förskrivarkod LDAP: personalPrescriptionCode	String	1 – 1	Förskrivarkod
hsaSosNursePrescriptionRight HSA: Förskrivningsrätt för barnmorskor/sjuksköterskor LDAP: hsaSosNursePrescriptionRight	String	0 – n	Förskrivningsrätt för barnmorskor/sjuksköterskor. Anges enligt följande syntax om personen har förskrivningsrätt som barnmorska/sjuksköterska: "<Kod för legitimerad yrkesgrupp>;<J>". Exempel: "SJ;J" för sjuksköterska med förskrivningsrätt.
educationCodes	Collection	1 – 1	
educationCode HSA: <i>namn saknas</i> LDAP: hsaSosEducation Code	String	0 – n	Utbildningskod
restrictionCodes	Collection	1 – 1	
restrictionCode HSA: <i>namn saknas</i> LDAP: hsaSosRestrictions Code	String	0 – n	Restriktionskoder
restrictions	Collection	1 – 1	
restriction HSA: <i>namn saknas</i> LDAP: hsaSosRestrictions	String	0 – n	Restriktioner i klartext
specialityCodes	Collection	1 – 1	
specialityCode HSA: Specialitetskod LDAP: specialityCode	String	0 – n	Specialitetskod
specialityNames	Collection	1 – 1	



Parameter	Datotyp	Antal	Beskrivning
specialityName HSA: Specialitet LDAP: specialityName	String	0 – n	Specialitet i klartext
titleCodes	Collection	1 – 1	
titleCode HSA: <i>namn saknas</i> LDAP: hsaSosTitleCode	String	0 – n	Yrkestitel. Aktuella legitimerade yrkestitlar finns beskrivna i HSA kodtabell (se referens [3]), i Hosp-informationen finns dock även äldre titlar.
hsaTitles	Collection	1 – 1	
hsaTitle HSA: Legitimerad yrkesgrupp LDAP: hsaTitle	String	0 – n	HSA-titel för specificerad person. Aktuella HSA-titlar specificeras i HSA kodtabell. För mer information se referens [3].
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om svaret innehåller fingerade objekt.

4.2.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHospPerson*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetHospPerson>
      <urn:personalIdentityNumber>
        1912121212
      </urn:personalIdentityNumber>
    </urn:GetHospPerson>
  </soap:Body>
</soap:Envelope>
```



```
</soap:Body>  
</soap:Envelope>
```

Resultat

```
<soap:Envelope  
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">  
  <soap:Body>  
    <GetHospPersonResponse  
      xmlns="urn:riv:hsa:HsaWsResponder:3"  
      xmlns:ns2="http://www.w3.org/2005/08/addressing">  
      <personalIdentityNumber>  
        191212121212  
      </personalIdentityNumber>  
      <personalPrescriptionCode>  
        1234567  
      </personalPrescriptionCode>  
      <educationCodes>  
        <educationCode>LEG</educationCode>  
        <educationCode>SPEC</educationCode>  
      </educationCodes>  
      <restrictions>  
        <restriction>  
          Prövotid/Inskr. Förskrivningsrätt  
        </restriction>  
      </restrictions>  
      <restrictionCodes>  
        <restrictionCode>003</restrictionCode>  
      </restrictionCodes>  
      <titleCodes>  
        <titleCode>LK</titleCode>  
      </titleCodes>  
      <specialityCodes>  
        <specialityCode>32</specialityCode>  
        <specialityCode>74</specialityCode>  
      </specialityCodes>  
      <specialityNames>  
        <specialityName>Klinisk fysiologi</specialityName>  
        <specialityName>Nukleärmedicin</specialityName>  
      </specialityNames>  
    </GetHospPersonResponse>  
  </soap:Body>  
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHospPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.



```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
GetHospPersonType parameters = new GetHospPersonType();
parameters.setPersonalIdentityNumber("191212121212");

// Call hsaws
GetHospPersonResponseType response =
    hsaws.getHospPerson(logicalAddress, id, parameters);

System.out.println("Personal prescription code: " +
    response.getPersonalPrescriptionCode());
for (String titleCode : response.getTitleCodes().getTitleCode())
{
    System.out.println("Titelkod: " + titleCode);
}
```

4.3 GetHsaUnit

GetHsaUnit används för att söka ut information om en specifik enhet eller funktion.



4.3.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsalentity	String	1 - 1	HSA-identitet på den enhet som skall sökas ut.
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <code>c=SE</code> som sökbas.

4.3.2 Returvärde

- Om specificerad enhet hittas returneras information om enheten i ett objekt av XML-typen *GetHsaUnitResponse*.
- Om enheten inte hittas returneras *null* (en tom SOAP-body).
- Arkiverade enheter returneras om klientanvändaren har rätt att läsa dem (objectClass=hsaArchivedObject)

GetHsaUnitResponse

För detaljerad information om respektive parameter se HSA-specifikationen [6].

Parameter	Datotyp	Antal	Beskrivning
alternateNames		1 – 1	
alternateName HSA: Alternativt namn LDAP: ouShort	String	0 – n	Alternativt namn på enheten som används vid sidan av det officiella namnet (<i>publicName</i>).
alternateText HSA: alt-text LDAP: hsaAltText	String	0 – 1	Beskrivande text som alternativ till bild på enhet.
businessClassifications		1 – 1	
businessClassification HSA: Verksamhet LDAP: businessClassification Name	String	0 – n	Typ av verksamhet (klartext). För mer information se HSA kodtabell [7].
businessClassificationCodes		1 – 1	
businessClassificationCode HSA: Verksamhetskod LDAP: businessClassification Code	String	0 – n	Typ av verksamhet (kod). För mer information se HSA kodtabell <i>verksamhetskod</i> [7].
businessTypes		1 – 1	



businessType HSA: Enhetstyp LDAP: hsaBusinessType	String	0 – n	Klassificering av enhet (t.ex. sjukhus). För mer information se HSA kodtabell <i>enhetstyp</i> [8]
careTypes		1 – 1	
careType HSA: Vårdform LDAP: careType	String	0 – n	Vårdform. För mer information se HSA kodtabell <i>vårdform</i> [9].
county HSA: Länsnamn LDAP: countyName	String	0 – 1	Namn på län. För mer information se HSA kodtabell <i>länskod</i> [10].
countyCode HSA: Länskod LDAP: countyCode	String	0 – 1	Kod för län. För mer information se HSA kodtabell <i>länskod</i> [10].
description HSA: Beskrivning LDAP: description	String	0 – 1	Allmän beskrivning för enheten.
directoryContact HSA: Innehållsansvarig LDAP: hsaDirectoryContact (hämtad från överliggande o eller ou-objekt)	String	0 – 1	Ansvarig för informationen om enheten (emailadress).
displayOption HSA: Alternativ namnvisning LDAP: displayOption	String	0 – 1	Används för att beräkna enhetens publika namn (<i>publicName</i>). För mer information se referens [6].
dropInHours		1 – 1	
dropInHour HSA: Tid för dropin-besök utan tidsbokning LDAP: dropInHours	Time Span	0 – n	Tider för så dropin-besök (utan tidbokning).
email HSA: e-postadress LDAP: mail	String	0 – 1	Officiell emailadress till enheten.
faxNumbers		1 – 1	



faxNumber HSA: Fax LDAP: facsimileTelephone Number	Telefon	0 – n	Faxnummer till enheten.
financingOrganizations		1 - 1	
financingOrganization HSA: finansierande landsting/kommun LDAP: financingOrganization	String	0 – n	Angivelse av organisationsnummer för de landsting eller kommuner som finansierar den vård som bedrivs av en enhet.
geographicalCoordinatesRt90 HSA: Geografiska koordinater LDAP: geographicalCoordinates	Geo Coord	0 – 1	Geografiska koordinater för enhetens huvudsakliga fysiska placering. Koordinaterna anges enligt RT90.
healthCareArea HSA: Områdesansvar LDAP: hsaHealthCareArea	String	0 – 1	Geografiskt område för någon typ av administrativt område.
hsaDestinationIndicators		1 – 1	
hsaDestinationIndicator HSA: Visas för LDAP: hsaDestinationIndicator	String	0 – n	Anger vilka parter som får ta del av enhetens information. För mer information se HSA kodtabell <i>visas för</i> [12].
hsaldentity HSA: HSA-id LDAP: hsaldentity	String	1 – 1	HSA-identitet
hsaJpegLogotype HSA: Logotyp LDAP: hsaJpegLogotype	String	0 – 1	Logotype för enheten. Logotypen är en binär bild i JPEG-format. Bildinformationen returneras Base64-kodat.
invoiceAddress HSA: Fakturaadress LDAP: hsaInvoiceAddress	Address	0 – 1	Fakturaadress.
jpegPhoto HSA: Bild LDAP: jpegPhoto	String	0 – 1	Allmän bild för enheten. Bilden är en binär bild i JPEG-format. Bildinformationen returneras Base64-kodat.



labeledUri HSA: Webbadress LDAP: labeledURI	String	0 – 1	Fullständig webbadress (inklusive <i>http://</i> eller <i>https://</i>).
location HSA: Geografisk plats LDAP: l	String	0 – 1	Namn på geografiskt område där enheten i huvudsak är placerad.
managements		1 – 1	
management HSA: Ägarform LDAP: management	String	0 – n	Ägarform i klartext. För mer information se HSA kodtabell <i>ägarform</i> [13]
municipality HSA: Kommunnamn LDAP: municipalityName	String	0 – 1	Namn på kommun. För mer information se HSA kodtabell <i>kommunkod</i> [11].
municipalityCode HSA: Kommunkod LDAP: municipalityCode	String	0 – 1	Kod för kommun. För mer information se HSA kodtabell <i>kommunkod</i> [11].
municipalitySection HSA: Kommunfelsnamn LDAP: municipalitySectionName	String	0 – 1	Namn på kommundel.
name HSA: Organisationsnamn, Enhetsnamn eller Objekttnamn, LDAP: o, ou eller cn (beroende på objekttyp)	String	1 – 1	Namn Namn i detta sammanhang definieras som värdet i LDAP-attributet <i>o</i> (organization), <i>ou</i> (organisationsenhet) eller <i>cn</i> (organisationsroll).
patientInformation HSA: Information till patient LDAP: hsaVpwlInformation4	String	0 – 1	Informationstext till patienter.
postalAddress HSA: Postadress LDAP: postalAddress	Address	0 – 1	Postadress.
postalCode HSA: Postnummer LDAP: postalCode	String	0 – 1	Postnummer.



<p>priceInformation</p> <p>HSA: Prisinformation</p> <p>LDAP: hsaVpwInformation3</p>	String	0 – 1	Prisinformation.
<p>publicName</p> <p>(Namn som sätts samman utifrån flera olika HSA-attribut, se beskrivningen)</p>	String	1 – 1	<p>Publikt officiellt namn.</p> <p>Det publika namnet beräknas i första hand utifrån enhetens <i>DN</i> tillsammans med värdet i attributet <i>displayOption</i> i enlighet med beskrivningen i HSA-specifikationen [6].</p> <p>Om enheten inte har något värde för <i>displayOption</i> beräknas det publika namnet utifrån följande attributkombinationer, i prioritetsordning;</p> <ul style="list-style-type: none"> • ou<blanktecken> • o<blanktecken> • cn<blanktecken>
relatedUnits		1 – 1	
<p>relatedUnit</p> <p>HSA: Relaterad enhet</p> <p>LDAP: hsaVpwNeighbouring Object</p>	String	0 – n	<p>HSA-identitet på en enhet som på något sätt hör ihop med aktuell enhet.</p> <p>Informationen hämtas från LDAP-attributet <i>hsaVpwNeighbouringObject</i>.</p>
<p>route</p> <p>HSA: Vägbeskrivning</p> <p>LDAP: route</p>	String	0 – 1	Vägbeskrivning.
<p>street</p> <p>HSA: Besöksadress</p> <p>LDAP: street</p>	String	0 – 1	Besöksadress (gatuadress).
<p>smsTelephoneNumber</p> <p>HSA: sms-telefon</p> <p>LDAP: smsTelephoneNumber</p>	Telefon	0 – 1	Telefonnummer som används för SMS meddelanden
surgeryHours		1 – 1	
<p>surgeryHour</p> <p>HSA: Öppettider</p> <p>LDAP: surgeryHours</p>	Time Span	0 – n	Öppettider.



switchboardNumber HSA: Växeltелефон LDAP: hsaSwitchboardNumber	Telefon	0 – 1	Telefonnummer till växel
telephoneHours		1 – 1	
telephoneHour HSA: Telefontid LDAP: telephoneHours	Time Span	0 – n	Telefontider för telefonnummer i parametern <i>telephoneNumbers</i> .
telephoneNumbers		1 – 1	
telephoneNumber HSA: Telefonnummer LDAP: telephoneNumber	Telefon	0 – n	Publikt direkttelefonnummer.
textTelephoneNumbers		1 – 1	
textTelephoneNumber HSA: Texttelefon LDAP: hsaTextTelephone Number	String	0 – n	Texttelefonnummer för personer med tal- eller hörselhandikapp.
unitExtraInformation HSA: Mer om LDAP: hsaVpwInformation1	String	0 – 1	Kompletterande information om enheten
unitFunctions		1 – 1	
unitFunction (se nedan för HSA och Ldapnamn)	UnitFunc	0 – n	Reserverade funktionsnamn kopplade till aktuell enhet.
unitTemporaryInformation HSA: Tillfällig information LDAP: hsaVpwInformation2	DateSpan	0 – 1	Temporär information om enheten.
visitingHours		1 – 1	
visitingHour HSA: Besökstider för anhöriga LDAP: visitingHours	TimeSpan	0 – n	Besökstider för anhöriga.
visitingRuleAge HSA: Åldersregler för vårdkontakt LDAP: hsaVisitingRuleAge	AgeSpan	0 – 1	Åldersintervall på patienter som tas emot.



visitingRuleReferral HSA: Remisskrav LDAP: hsaVisitingRuleReferral	String	0 – 1	Beskrivning av remisskrav.
visitingRules HSA: Besöksregler LDAP: hsaVisitingRules	String	0 – 1	Besöksregler
startDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdenhetens verksamhet.
endDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdenhetens verksamhet.
hsaVpwWebPage HSA: 1177-kontaktkortslänk LDAP: hsaVpwWebPage	String	0 – 1	Länk till enhetens kontaktkort på 1177.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om svaret innehåller fingerade objekt.

TimeSpan

Tidsintervall.

Parameter	Datatyp	Antal	Beskrivning
fromDay	String	1 – 1	Från dag. Måndag (1) – Söndag (7).
fromTime	Time (XML)	1 – 1	Från tid.
toDay	String	1 – 1	Till dag. Måndag (1) – Söndag (7).
toTime	Time (XML)	1 – 1	Till tid.
comment	String	0 – 1	Information om aktuellt tidsintervall.
fromDate	String	0 – 1	Datum då tidsintervallet börjar gälla
toDate	String	0 – 1	Sista datum för tidsintervallet



GeoCoord

Geografiska koordinater.

Parameter	Datotyp	Antal	Beskrivning
Type	String	1 – 1	Typ av geografisk koordinat. Idag används enbart <i>RT90</i> .
X	String	1 – 1	X-koordinat.
Y	String	1 – 1	Y-koordinat.

Address

Adressinformation

Parameter	Datotyp	Antal	Beskrivning
addressLine	String	1 – n	Adressrad.

UnitFunc

Information om så kallad enhetsfunktion. Idag finns följande enhetsfunktioner;

- Avbokning
- Bokning
- Sjukresebeställning
- Receptförnyelse
- Rådgivning

Parameter	Datotyp	Antal	Beskrivning
namn HSA: Objekt LDAP: cn	String	1 – 1	Enhetsfunktionens namn (se ovanstående lista).
telephoneHours		1 – 1	
telephoneHour HSA: Telefontid LDAP: telephoneHours	TimeSpan	0 – n	Telefontider för telefonnummer i parametern <i>telephoneNumbers</i> .
telephoneNumbers		1 – 1	
telephoneNumber HSA: Telefonnummer LDAP: telephoneNumber	Telefon	0 – n	Publikt direkttelefonnummer.



AgeSpan

Åldersintervall

Parameter	Datotyp	Antal	Beskrivning
From	String	1 – 1	Från ålder. 00 för nyfödd.
To	String	1 – 1	Till ålder. 99 för ingen övre åldersgräns.
Comment	String	0 – 1	Kommentar i fri text

Telefon

Telefonnummer anges med XML-typen *string*. Telefonnummer kodas i nationellt format utan bindestreck och mellanslag (t.ex. 08123456).

DateSpan

Datumintervall.

Parameter	Datotyp	Antal	Beskrivning
fromDate	String	1 – 1	Från datum. Exempel: 20101123
toDate	String	1 – 1	Till datum. Exempel: 20101131
comment	String	1 – 1	Information om aktuellt datumintervall.

4.3.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHsaUnit*. Exemplet ges i form av SOAP-meddelanden.



Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetHsaUnit>
      <urn:hsaIdentity>SE1777-001</urn:hsaIdentity>
    </urn:GetHsaUnit>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetHsaUnitResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <alternateNames>
        <alternateName>CS</alternateName>
      </alternateNames>
      <alternateText>
        Centralsjukhusets huvudentre
      </alternateText>
      <businessClassifications>
        <businessClassification>
          Akutsjukvård
        </businessClassification>
        <businessClassification>
          Allmänsjukvård
        </businessClassification>
      </businessClassifications>
      <businessClassificationCodes>
        <businessClassificationCode>
          2
        </businessClassificationCode>
        <businessClassificationCode>
          3
        </businessClassificationCode>
      </businessClassificationCodes>
      <businessTypes>
        <businessType>4</businessType>
      </businessTypes>
    </GetHsaUnitResponse>
  </soap:Body>
</soap:Envelope>
```



```
<careTypes>
  <careType>01</careType>
</careTypes>
<county>Centrala länet</county>
<countyCode>103</countyCode>
<directoryContact>
  katalogansvarig@landstinget.se
</directoryContact>
<displayOption>2#1</displayOption>
<dropInHours>
  <dropInHour>
    <fromDay>1</fromDay>
    <fromTime>10:00:00.487+01:00</fromTime>
    <toDay>4</toDay>
    <toTime>12:00:00.487+01:00</toTime>
    <comment>Anmälan i receptionen</comment>
  </dropInHour>
</dropInHours>
<email>info@sjukhuset.se</email>
<faxNumbers>
  <faxNumber>08555555</faxNumber>
  <faxNumber>08666666</faxNumber>
</faxNumbers>
<geographicalCoordinatesRt90>
  <type>RT90</type>
  <x>1234567</x>
  <y>7654321</y>
</geographicalCoordinatesRt90>
<healthCareArea>Centralhöjden</healthCareArea>
<hsaDestinationIndicators>
  <hsaDestinationIndicator>
    03
  </hsaDestinationIndicator>
</hsaDestinationIndicators>
<hsaIdentity>SE1777-001</hsaIdentity>
<hsaJpegLogotype>
  /9j/HG5D+y82eNM2KvGd54DsJR...
</hsaJpegLogotype>
<invoiceAddress>
  <addressLine>Fakturaavdelningen</addressLine>
  <addressLine>Centralsjukhuset</addressLine>
  <addressLine>123 45 Huvudstaden</addressLine>
</invoiceAddress>
<jpegPhoto>
  /9j/YqUy82eNM2Kv08Dsw5Gf+D...
</jpegPhoto>
<labeledUri>
  http://www.centralsjukhuset.se/
</labeledUri>
<location>Huvudstaden</location>
<managements>
  <management>6</management>
</managements>
```



```
<municipality>Centrala staden</municipality>
<municipalityCode>903</municipalityCode>
<municipalitySection>
  Centralhöjden
</municipalitySection>
<name>Centralsjukhuset</name>
<patientInformation>
  Vid sjukdom kontakta mottagningen innan besök
</patientInformation>
<postalAddress>
  <addressLine>Box 123</addressLine>
  <addressLine>123 45 Huvudstaden</addressLine>
</postalAddress>
<postalCode>12345</postalCode>
<priceInformation>
  Vanlig patienttaxa gäller
</priceInformation>
<publicName>Centralsjukhuset, Landstinget</publicName>
<relatedUnits>
  <relatedUnit>SE1777-901</relatedUnit>
  <relatedUnit>SE1777-902</relatedUnit>
</relatedUnits>
<route>Tag buss 3 från centralstationen</route>
<street>Centralvägen 123</street>
<surgeryHours>
  <surgeryHour>
    <fromDay>1</fromDay>
    <fromTime>10:00:00.487+01:00</fromTime>
    <toDay>7</toDay>
    <toTime>16:00:00.487+01:00</toTime>
    <comment>Receptionen öppen</comment>
  </surgeryHour>
</surgeryHours>
<switchboardNumber>08123456</switchboardNumber>
<telephoneHours>
  <telephoneHour>
    <fromDay>1</fromDay>
    <fromTime>16:00:00.487+01:00</fromTime>
    <toDay>7</toDay>
    <toTime>22:00:00.487+01:00</toTime>
    <comment>Telefonsvarare</comment>
  </telephoneHour>
</telephoneHours>
<telephoneNumbers>
  <telephoneNumber>08111111</telephoneNumber>
  <telephoneNumber>08222222</telephoneNumber>
</telephoneNumbers>
<textTelephoneNumbers>
  <textTelephoneNumber>08333333</textTelephoneNumber>
  <textTelephoneNumber>08444444</textTelephoneNumber>
</textTelephoneNumbers>
<unitExtraInformation>
  Ingång från entre 2
```



```
</unitExtraInformation>
<unitFunctions>
  <unitFunction>
    <name>Bokning</name>
    <telephoneHours>
      <telephoneHour>
        <fromDay>1</fromDay>
        <fromTime>10:00:00.503+01:00</fromTime>
        <toDay>5</toDay>
        <toTime>11:00:00.503+01:00</toTime>
        <comment/>
      </telephoneHour>
    </telephoneHours>
    <telephoneNumbers>
      <telephoneNumber>08999999</telephoneNumber>
    </telephoneNumbers>
  </unitFunction>
</unitFunctions>
<unitTemporaryInformation>
  <fromDate>20101115</fromDate>
  <toDate>20101125</toDate>
  <comment>Influensavaccinering erbjuds</comment>
</unitTemporaryInformation>
<visitingHours>
  <visitingHour>
    <fromDay>1</fromDay>
    <fromTime>08:00:00.503+01:00</fromTime>
    <toDay>5</toDay>
    <toTime>20:00:00.503+01:00</toTime>
    <comment/>
  </visitingHour>
  <visitingHour>
    <fromDay>6</fromDay>
    <fromTime>10:00:00.503+01:00</fromTime>
    <toDay>7</toDay>
    <toTime>18:00:00.503+01:00</toTime>
    <comment>Gå direkt till avdelningen</comment>
  </visitingHour>
</visitingHours>
<visitingRuleAge>
  <from>18</from>
  <to>99</to>
</visitingRuleAge>
<visitingRuleReferral>
  Remiss krävs
</visitingRuleReferral>
<visitingRules>
  Inga blommor får medtagas till avdelningen
</visitingRules>
</GetHsaUnitResponse>
</soap:Body>
</soap:Envelope>
```



JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnit*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService = new
HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1777-001");

// Call hsaws
GetHsaUnitResponseType response =
    hsaws.getHsaUnit(logicalAddress, id, parameters);

// Handle response
System.out.println("HSA-identity: " +
    response.getHsaIdentity());
System.out.println("Name: " + response.getPublicName());
```



4.4 GetInformationList

GetInformationList används för att hämta en specificerad informationslista. En informationslista är en XML-baserad lista med utvald information ur HSA-katalogen.

4.4.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
listIdentity	String	1 – 1	Identitet (namn) på den informationslista som skall hämtas.

4.4.2 Returvärde

- Metoden returnerar ett objekt av XML-typen *GetInformationListResponseType*.

GetInformationListResponseType

Parameter	Datotyp	Antal	Beskrivning
containsList	Boolean	1 – 1	True om svaret innehåller eftersökt informationslista. False om svaret inte innehåller en informationslista.
informationList	(xop:Include)	0 – 1	Informationslista. För mer information se avsnitt 4.4.3.

4.4.3 Informationslista

Informationslistan är en XML-baserad lista med utvald information ur HSA-katalogen. Då generering av informationslistorna är en relativt tung operation förgenereras de varje natt.

XML-format

En informationslista består av en instans av XML-datatypen *HsaInformationListType* (element *HsaInformationList*). Formatet på listorna definieras i följande XML-scheman. Dessa scheman kan erhållas från Ineras HSA-förvaltning (referens [5]).

- HsaInformationList_1.4.xsd
- HsaWsResponder_3.11.xsd

InformationListType

Parameter	Datotyp	Antal	Beskrivning
StartDate	dateTime	1 – 1	Tid då genereringen av aktuell lista startade.
HsaObjects		1 – 1	De HSA-objekt som ingår i aktuell lista.



Parameter	Datotyp	Antal	Beskrivning
HsaObject	HsaObjectType	0 – n	Information om ett ingående HSA-objekt.
EndDate	dateTime	1 – 1	Tid då genereringen av aktuell lista avslutades.
TotalHsa	long	1 – 1	Totalt antal HSA-objekt i listan.

HsaObjectType

Parameter	Datotyp	Antal	Beskrivning
DN	string	1 – 1	DN (identitet) för aktuellt HSA-objekt.
Attribut	HsaAttributeType	0 – n	Informationsattribut för aktuellt objekt.

HsaAttributeType

Parameter	Datotyp	Antal	Beskrivning
name (attribute)	string	1 – 1	Namn på attributet. Namnet är identiskt med LDAP-namnet på aktuellt attribut.
(värde)	(se 0)	1 – 1	Attributets värde.

Attributvärden

Ett attributvärde för ett HSA-objekt består av någon av följande typer.

Datotyp	Format
Sträng	<S>strängvärde</S>
Adress	Se avsnitt 0
Geografisk koordinat	Se avsnitt 0
Åldersspann	Se avsnitt 0
Tidsspann	Se avsnitt 0
Enhetsfunktion	Se avsnitt 0

Komprimering

En informationslista kan bli mycket stor. För att kunna överföra en sådan lista på ett effektivt sätt med hsaws är listan komprimerad. För komprimeringen används vanligt zip-format (se referens [18]). Informationslistan i webservicessvaret (0) består av innehållet i en zip-fil. Denna zip-fil består sedan av en zip-post. Denna zippost är vår informationslista och har namnet <listIdentity>.xml.



För att ytterligare optimera överföringen av informationslistan används webservicestandarden MTOM (se referens [14]).

4.4.4 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetInformationList*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soapenv:Body>
    <urn:GetInformationList>
      <urn:listIdentity>demolista</urn:listIdentity>
    </urn:GetInformationList>
  </soapenv:Body>
</soapenv:Envelope>
```

Resultat

Som vi har sagt tidigare returneras informationslistan som en binär "attachment" enligt standarden MTOM (se referens [14]) och XOP (se referens [15]).

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetInformationListResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <containsList>true</containsList>
      <informationList>
        <xop:Include href="cid:6b3bf09e-6ccf-4db3-ae83-8306a1e381e5-2@urn%3Ariv%3Ahsa%3AHsaWsResponder%3A3"
          xmlns:xop="http://www.w3.org/2004/08/xop/include"/>
      </informationList>
    </GetInformationListResponse>
  </soap:Body>
</soap:Envelope>
```



Informationslista

I detta avsnitt ger vi ett exempel på hur en informationslista kan se ut.

```
<?xml version="1.0" encoding="UTF-8"?>
<HsaInformationList
  xmlns="urn:riv:hsa:HsaInformationList:1"
  xmlns:hwr="urn:riv:hsa:HsaWsResponder:3">
  <StartDate>2010-09-24T16:24:33.853+02:00</StartDate>
  <HsaObjects>
    <HsaObject>
      <DN>
        ou=Husläkarmottagningen,o=Landstinget,
        l=Länet, c=SE
      </DN>
      <Attribute name="objectClass">
        <S>HSAOrganizationExtension</S>
      </Attribute>
      <Attribute name="telephoneNumber">
        <S>+468123456</S>
      </Attribute>
      <Attribute name="postalAddress">
        <Address>
          <hwr:addressLine>
            Husläkarmottagningen
          </hwr:addressLine>
          <hwr:addressLine>Box 123</hwr:addressLine>
          <hwr:addressLine>123 45 Staden</hwr:addressLine>
        </Address>
      </Attribute>
      <Attribute name="surgeryHours">
        <TimeSpan>
          <hwr:fromDay>1</hwr:fromDay>
          <hwr:fromTime2>07:45</hwr:fromTime2>
          <hwr:toDay>4</hwr:toDay>
          <hwr:toTime2>16:30</hwr:toTime2>
          <hwr:comment/>
        </TimeSpan>
      </Attribute>
      ...
    </HsaObject>
    ...
  </HsaObjects>
  <EndDate>2010-09-24T16:24:51.290+02:00</EndDate>
  <TotalHsaObjects>45</TotalHsaObjects>
</HsaInformationList>
```

4.4.5 Java

I detta avsnitt ger vi ett längre exempel på hur metoden `GetInformationList` används i en javaapplikation. Vi använder javastandarden JAX-WS 2.1 (referens [19]) som ingår i Java 6.



Kodgenerering

Vi startar med att generera webservicekod för hsaws. Denna kod genereras med hjälp av javakommandot `wsimport` utifrån information i hsaws WSDL-fil (*HsaWsInteraction_3.7.wsdl*) och schemafiler (*HsaWsResponder_3.7.xsd* och *ws-addressing-1.0.xsd*). Dessa filer kan erhållas från Ineras HSA-förvaltning (referens [5]).

```
C:\Temp2\hsaws-wsi\wsdl>dir
Volume in drive C has no label.
Volume Serial Number is 2CC5-2E97

Directory of C:\Temp2\hsaws-wsi\wsdl

2010-09-26  10:24    <DIR>          .
2010-09-26  10:24    <DIR>          ..
2010-09-24  09:33                14 034 HsaWsInteraction_3.7.wsdl
2010-09-26  10:25    <DIR>          src
                1 File(s)                14 034 bytes
                3 Dir(s)  14 459 469 824 bytes free

C:\Temp2\hsaws-wsi\wsdl>dir ..\schemas
Volume in drive C has no label.
Volume Serial Number is 2CC5-2E97

Directory of C:\Temp2\hsaws-wsi\schemas

2010-09-26  10:20    <DIR>          .
2010-09-26  10:20    <DIR>          ..
2010-09-21  12:48                27 997 HsaWsResponder_3.7.xsd
2010-02-13  20:04                5 849 ws-addressing-1.0.xsd
                2 File(s)                33 846 bytes
                2 Dir(s)  14 459 469 824 bytes free

C:\Temp2\hsaws-wsi\wsdl>wsimport -d src -p se.inera.hsaws.ws -
extension -XadditionalHeaders -Xno-addressing-databinding
-Xnocompile HsaWsInteraction_3.7.wsdl
parsing WSDL...

generating code...

C:\Temp2\hsaws-wsi\wsdl>
```

I vårt exempel har vi valt att enbart generera källkodsfiler (*-Xnocompile*).

Program

```
public static void main(String[] args) {
    try {
```



```
System.out.println("Get information list");

DataHandler infoListDh =
    getInformationList(
        "https://testhotell2.carelink.sjunet.org/" +
        "svr-hsaws2-internet/hsaws",
        "demoList");

if (infoListDh == null) {
    System.err.println("No information list data handler");
    return;
}

HsaInformationListType infoList =
    unzipAndDeserialize(infoListDh);

if (infoList == null) {
    System.err.println("No information list");
    return;
}

handleInformationList(infoList);
}
catch (Throwable e) {
    System.err.println("Error in hsaws demo: " +
        e.getMessage());
}
}
```

Vårt javaprogram består av tre delar.

- Anrop av hsaws för inhämtning av informationslista
- Dekomprimering och deserialisering av informationslista
- Hantering av informationslista

Anrop av hsaws

```
public static DataHandler getInformationList(String url,
    String listIdentity) throws HsaWsFault {
    System.out.println("Get information list from hsaws: " +
        url + " " + listIdentity);

    // Create hsaws client stub
    HsaWsResponderService hsawsService =
        new HsaWsResponderService();
    HsaWsResponderInterface hsaws =
        hsawsService.getHsaWsResponderPort();

    // We must set the hsaws endpoint url explicitly
    BindingProvider bp = (BindingProvider)hsaws;
    bp.getRequestContext().put(
```



```
BindingProvider.ENDPOINT_ADDRESS_PROPERTY, url);

// Enable MTOM
SOAPBinding binding = (SOAPBinding)bp.getBinding();
binding.setMTOMEnabled(true);

// Create general request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");

// Create method specific parameters
GetInformationListType parameters =
    new GetInformationListType();
parameters.setListIdentity(listIdentity);

// Call hsaws
GetInformationListResponseType response =
    hsaws.getInformationList(logicalAddress, id, parameters);

if (!response.isContainsList()) return null;
return response.getInformationList();
}
```

Här använder vi den webservicekod vi genererade i avsnitt 0. Vi börjar med att instansiera en så kallad klientstubbe. Med hjälp av stubben anger vi URL:en på den instans av hsaws som skall anropas. Vidare aktiveras stöd för MTOM.

Innan vi kan anropa hsaws måste vi skapa våra anropsparametrar. Hsaws har två generella parametrar; hsaws HSA-identitet (logicalAddress) och vår unika loggidentitet (id). För mer information se avsnitt 2.4. Vidare skapar vi en metodspezifisk anropsparameter (GetInformationListResponseType) som vi instansierar med identiteten på den informationslista vi vill hämta.

Vi är nu redo att anropa GetInformationList i hsaws via vår klientstubbe. Som returvärde får vi en instans av den genererade klassen GetInformationListResponseType (se avsnitt 0). Vi börjar med att kontrollera att svaret innehåller en informationslista. Om så är fallet returneras denna till vårt huvudprogram.

Som vi sett är informationslistan komprimerad och överföringen hanteras med hjälp av MTOM. I den genererade javakoden får vi då informationslistan som en instans av javatypen *javax.activation.DataHandler*.

Dekomprimering

```
@SuppressWarnings("unchecked")
public static HsaInformationListType unzipAndDeserialize(
    DataHandler infoListDh) throws IOException, JAXBException
{
```



```
System.out.println("Unzip information list zip");
ZipInputStream infoListZip = null;
try {
    infoListZip =
        new ZipInputStream(infoListDh.getInputStream());
    // There should be only one entry in the zip
    ZipEntry ze = infoListZip.getNextEntry();

    if (ze == null) throw
        new RuntimeException("No information list xml in zip");

    System.out.println("Deserialize information list xml");
    JAXBContext jc =
        JAXBContext.newInstance("se.inera.hsaws.jaxb");
    Unmarshaller um = jc.createUnmarshaller();

    JAXBElement<HsaInformationListType> infoListElement =
        (JAXBElement<HsaInformationListType>)
            um.unmarshal(infoListZip);

    return infoListElement.getValue();
}
finally {
    if (infoListZip != null) infoListZip.close();
}
}
```

Informationslistan är komprimerad med vanlig zip. Vi måste således börja med att dekomprimera denna. Java 6 innehåller inbyggd funktionalitet för hantering av zip (*java.util.zip*). Vi skapar en instans av *ZipInputStream* utifrån inströmmen från vår *DataHandler* (informationslistan från webservicessvaret). Zip är en standard som både komprimerar och packar ihop en hel katalogstruktur. Vi måste således börja med att ta fram vår informationslista i zip-strömmen. Detta är mycket enkelt då vår zip-ström enbart består av en zippost (vår lista). Informationslistan är nu redo att läsas från vår *ZipInputStream*.

Deserialisering

Informationslistan är som vi tidigare sagt XML-baserad. Det finns flera sätt att hantera XML i java. Vi har valt att deserialisera xml-strömmen till en objektstruktur som sedan kan användas för enkel åtkomst till listinformationen. För deserialiseringen använder vi javastandaren JAXB 2.1 ([17]) som ingår som standard i Java 6.

Innan vi kan deserialisera vår XML-ström måste vi generera en klasstruktur utifrån informationen i de XML-scheman som definierar vår informationslista (se avsnitt 0). Detta utförs med javakommandot *xjc*.

```
C:\Temp2\hsaws-jaxb>dir
Volume in drive C has no label.
Volume Serial Number is 2CC5-2E97
```



```
Directory of C:\Temp2\hsaws-jaxb

2010-09-27  08:23    <DIR>          .
2010-09-27  08:23    <DIR>          ..
2010-09-21  16:28                1 753 HsaInformationList_1.0.xsd
2010-09-21  12:48                27 997 HsaWsResponder_3.7.xsd
2010-09-27  08:24    <DIR>          src
                2 File(s)          29 750 bytes
                3 Dir(s)    14 457 364 480 bytes free
C:\Temp2\hsaws-jaxb>xjc HsaInformationList_1.0.xsd -p
se.inera.hsaws.jaxb -d src

parsing a schema...
compiling a schema...
se\inera\hsaws\jaxb\AddressType.java
se\inera\hsaws\jaxb\AgeSpanType.java
...

C:\Temp2\hsaws-jaxb>
```

Vi har nu det vi behöver för att deserialisera vår XML-ström. Källkoden för detta visades i föregående avsnitt (dekomprimering).

Vi startar med att skapa en JAXB-kontext (*javax.xml.bind.JAXBContext*) som initialiseras med vår genererade klasstruktur. Med hjälp av JAXB-kontexten kan vi sedan instansiera en deserialiserare (*javax.xml.bind.Unmarshaller*).

Nu kan vi deserialisera vår XML-ström via vår *ZipInputStream* för dekomprimeringssteget. Resultatet är ett speciellt JAXB-objekt där vi sedan kan plocka ut vår informationslista som en instans av *se.inera.hsaws.jaxb.HsaInformationListType* som ingår i vår genererade klasstruktur.

Användning

Vi kan nu hämtat vår informationslista och skapat ett objekt som vi kan använda för enkel åtkomst till informationen i informationslistan. Med hjälp av deserialiseringen i föregående avsnitt slipper vi hantera XML-strömmen manuellt vilken är ganska omständigt i java. Istället har vi enkelt åtkomst till informationen via vanliga så kallade getters.

```
public static void handleInformationList(
    HsaInformationListType infoList) {
    System.out.println("Handle information list");

    System.out.println("Start date: " + infoList.getStartDate());
    System.out.println("End date: " + infoList.getEndDate());
    System.out.println("Total objects: " +
        infoList.getTotalHsaObjects());

    for (HsaObjectType hsaObject :
        infoList.getHsaObjects().getHsaObject()) {
        System.out.println("DN: " + hsaObject.getDN());
        // ...
    }
}
```



```
}  
}
```

4.4.6 Microsoft .Net

I detta avsnitt ger vi ett längre exempel på hur metoden `GetInformationList` används i en C#-baserad Microsoft .Net-applikation. Vi använder .Net-standarden Windows Communication Foundation (WCF) för webservice-hanteringen.

Kodgenerering

Vi startar med att generera webservicekod för hsaws. Denna kod genereras med hjälp av kommandot `svcutil` utifrån information i hsaws WSDL-fil (*HsaWsInteraction_3.7.wsdl*) och schemafiler (*HsaWsResponder_3.7.xsd* och *ws-addressing-1.0.xsd*). Dessa filer kan erhållas från Ineras HSA-förvaltning (referens [5]).

```
C:\Temp2\hsaws-net>dir  
Volume in drive C has no label.  
Volume Serial Number is 2CC5-2E97  
  
Directory of C:\Temp2\hsaws-net  
  
2010-09-27  11:37    <DIR>          .  
2010-09-27  11:37    <DIR>          ..  
2010-09-23  11:37                14 102 HsaWsInteraction_3.7.wsdl  
2010-09-21  12:48                27 997 HsaWsResponder_3.7.xsd  
2010-02-13  20:04                 5 849 ws-addressing-1.0.xsd  
                3 File(s)                47 948 bytes  
                2 Dir(s)   14 456 389 632 bytes free  
  
C:\Temp2\hsaws-net>svcutil HsaWsInteraction_3.7.wsdl  
HsaWsResponder_3.7.xsd ws-addressing-1.0.xsd /language:c#  
/namespace:*,Inera.Hsaws.Ws  
Microsoft (R) Service Model Metadata Tool  
[Microsoft (R) Windows (R) Communication Foundation, Version  
3.0.4506.648]  
Copyright (c) Microsoft Corporation. All rights reserved.  
  
Generating files...  
C:\Temp2\hsaws-net\HsaWsInteraction.cs  
C:\Temp2\hsaws-net\output.config  
  
C:\Temp2\hsaws-net>
```

Program

```
public static void Main(string[] args)  
{  
    string listIdentity = "xmlExport";
```



```
byte[] infoListZip = GetInformationList(listIdentity);  
if (infoListZip == null)  
{  
    Console.WriteLine("Information list not found: " +  
        listIdentity);  
    return;  
}  
  
Inera.Hsaws.Schemas.HsaInformationListType infoList =  
    UnzipAndDeserialize(infoListZip);  
if (infoList == null)  
{  
    Console.WriteLine("No information list in zip: " +  
        listIdentity);  
    return;  
}  
  
HandleInformationList(infoList);  
}
```

Vårt c#-program består av tre delar.

- Anrop av hsaws för inhämtning av informationslista
- Dekomprimering och deserialisering av informationslista
- Hantering av informationslista

Anrop av hsaws

```
private static byte[] GetInformationList(string listIdentity)  
{  
    Console.WriteLine("Fetch information list from hsaws: " +  
        listIdentity);  
  
    // Make hsaws call to get information list in zip format  
    try  
    {  
        // Create hsaws client stub  
        HsaWsResponderInterface hsaws = new  
            HsaWsResponderInterfaceClient("HsaWsResponderPort");  
  
        // Create request with common parameters  
        GetInformationListRequest req = new  
            GetInformationListRequest();  
        AttributedURIType to = new AttributedURIType();  
        to.Value = "SE165565594230-1000";  
        req.To = to;  
        AttributedURIType id = new AttributedURIType();  
        id.Value = System.Guid.NewGuid().ToString();  
        req.MessageID = id;  
  
        // Add method specific parameters
```



```
GetInformationListType gilt = new
    GetInformationListType();
gilt.listIdentity = listIdentity;
req.GetInformationList = gilt;

GetInformationListResponse resp =
    hsaws.GetInformationList(req);

if (resp == null || !resp.containsList) return null;
return resp.informationList;
}
catch (TimeoutException timeProblem)
{
    Console.WriteLine("The service operation timed out. " +
        timeProblem.Message);
    throw timeProblem;
}
catch (FaultException<HsaWsFaultType> hsawsFault)
{
    Console.WriteLine(hsawsFault.Code);
    Console.WriteLine(hsawsFault.Message);
    Console.WriteLine(hsawsFault.Detail.Nodes[0].Name);
    Console.WriteLine(hsawsFault.Detail.Nodes[0].InnerText);
    Console.WriteLine(hsawsFault.Detail.Nodes[1].Name);
    Console.WriteLine(hsawsFault.Detail.Nodes[1].InnerText);
    throw hsawsFault;
}
catch (FaultException unknownFault)
{
    Console.WriteLine("An unknown exception was received. " +
        unknownFault.Message);
    throw unknownFault;
}
catch (CommunicationException commProblem)
{
    Console.WriteLine("There was a communication problem. " +
        commProblem.Message + commProblem.StackTrace);
    throw commProblem;
}
}
```

Här använder vi den webservicekod vi genererade i avsnitt 0. Vi börjar med att instansiera en så kallad klientstubbe. Innan vi kan anropa hsaws måste vi skapa våra anropsparametrar. Hsaws har två generella parametrar; hsaws HSA-identitet (To) och vår unika loggidentitet (MessageID). För mer information se avsnitt 2.4. Vidare skapar vi en methodspecifik anropsparameter (GetInformationListType) som vi instansierar med identiteten på den informationslista vi vill hämta.

Vi är nu redo att anropa GetInformationList i hsaws via vår klientstubbe. Som returvärde får vi en instans av den genererade klassen GetInformationListResponse (se avsnitt 0). Vi börjar med att kontrollera att svaret innehåller en informationslista. Om så är fallet returneras denna till vårt



huvudprogram. Som vi sett är informationslistan komprimerad och överföringen hanteras med hjälp av MTOM. I den genererade javakoden får vi då informationslistan som en bytearray (*byte[]*).

Resten av exempelkoden visar en strukturerad hantering av de olika exceptions som kan kastas av anropet till *hsaws*.

Dekomprimering

```
private static Inera.Hsaws.Schemas.HsaInformationListType
    UnzipAndDeserialize(byte[] infoListZip)
{
    Console.WriteLine("Unzip and deserialize information list");

    // Create a zip input stream using DotNetZip library
    // (http://dotnetzip.codeplex.com/)
    using (ZipInputStream zip = new ZipInputStream(new
        MemoryStream(infoListZip))
    {
        // The information list xml file is the first, and only,
        // entry in the zip stream
        ZipEntry infoListEntry = zip.GetNextEntry();
        if (infoListEntry == null) throw new
            InvalidDataException("No information list file" +
                "in zip");

        XmlSerializer xs = new
            XmlSerializer(typeof(Inera.Hsaws.Schemas.
                HsaInformationListType));
        return (Inera.Hsaws.Schemas.HsaInformationListType)
            xs.Deserialize(zip);
    }
}
```

Informationslistan är komprimerad med vanlig zip. Vi måste således börja med att dekomprimera denna. Till skillnad mot java innehåller tyvärr inte .Net inbyggt stöd för hantering av zip-filer. Vi använder istället ett fritt tredjepartsbibliotek (DotNetZip) med vars hjälp zip-filer mycket enkelt kan hanteras (se referens [16]). Vi skapar en instans av *ZipInputStream* utifrån vår bytearray (informationslistan från webservicessvaret). Zip är en standard som både komprimerar och packar ihop en hel katalogstruktur. Vi måste således börja med att ta fram vår informationslista i zip-strömmen. Detta är mycket enkelt då vår zip-ström enbart består av en zippost (vår lista). Informationslistan är nu redo att läsas från vår *ZipInputStream*.

Deserialisering

Informationslistan är som vi tidigare sagt XML-baserad. Det finns flera sätt att hantera XML i Microsoft .Net. Vi har valt att deserialisera xml-strömmen till en objektstruktur som sedan kan



användas för enkel åtkomst till listinformationen. För deserialiseringen använder vi det inbyggda stödet för XML-serialisering i Microsoft .Net.

Innan vi kan deserialisera vår XML-ström måste vi generera en klasstruktur utifrån informationen i de XML-scheman som definierar vår informationlista (se avsnitt 0). Detta utförs med kommandot *xsd*.

```
C:\Temp2\hsaws-il>xsd HsaInformationList_1.0.xsd
HsaWsResponder_3.7.xsd /language:c#
/namespace:Inera.Hsaws.Schemas /classes
Microsoft (R) Xml Schemas/DataTypes support utility
[Microsoft (R) .NET Framework, Version 2.0.50727.1432]
Copyright (C) Microsoft Corporation. All rights reserved.
Writing file 'C:\Temp2\hsaws-
il\HsaInformationList_1_0_HsaWsResponder_3_7.cs'.

C:\Temp2\hsaws-il>
```

Vi har nu det vi behöver för att deserialisera vår XML-ström. Källkoden för detta visades i föregående avsnitt (dekomprimering).

Vi börjar med att skapa en deserialiserade (`System.Xml.Serialization.XmlSerializer`) som kan deserialisera vår informationslista (`Inera.Hsaws.Schemas.HsaInformationListType`). Nu kan vi deserialisera vår XML-ström via vår `ZipInputStream` för dekomprimeringssteget. Resultatet är en instans av `Inera.Hsaws.Schemas.HsaInformationListType` som ingår i vår genererade klasstruktur.

Användning

Vi kan nu hämtat vår informationslista och skapat ett objekt som vi kan använda för enkel åtkomst till informationen i informationslistan. Med hjälp av deserialiseringen i föregående avsnitt slipper vi hantera XML-strömmen manuellt vilken är ganska omständigt i Microsoft .Net. Istället har vi enkelt åtkomst till informationen via vanliga properties.

```
private static void HandleInformationList(
    Inera.Hsaws.Schemas.HsaInformationListType infoList)
{
    Console.WriteLine("Handle information list");

    Console.WriteLine("Start date: " + infoList.StartDate);
    Console.WriteLine("End date: " + infoList.EndDate);
    Console.WriteLine("Total HSA objects: " +
        infoList.TotalHsaObjects);

    foreach (Inera.Hsaws.Schemas.HsaObjectType hsaObject
        in infoList.HsaObjects)
    {
        Console.WriteLine("DN: " + hsaObject.DN);
        // ...
    }
}
```



}

4.5 GetMiuForPerson

GetMiuForPerson används för att söka ut behörighetsegenskaper (så kallade *medarbetare i uppdrag*) för en specificerad HSA-person.

Observera att denna metod primärt är till för användning i BIF.

Endast en av parametrarna `hsaIdentity` och `personallIdentityNumber` skall och måste användas.

4.5.1 Inparametrar

Parameter	Datatyp	Antal	Beskrivning
<code>hsaIdentity</code>	String	0 – 1	HSA-identitet på den person vars behörighetsegenskaper (MiU) skall utsökas..
<code>personallIdentityNumber</code>	String	0 – 1	Personnummer på den person vars behörighetsegenskaper (MiU) skall utsökas.
<code>searchBase</code>	DN	0 – 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <code>c=SE</code> som sökbas.
<code>includeDirectlyBelowProvider</code>	Boolean	0-1	Anger om medarbetaruppdrag som ligger direkt under Vårdgivare ska returneras eller ej. Värdet "true" tolkas som att medarbetaruppdrag direkt under vårdgivare ska inkluderas i svaret. "False" tolkas som att medarbetaruppdrag direkt under Vårdgivare inte ska inkluderas i svaret. Om parametern utelämnas tolkas det på samma sätt som "false".

4.5.2 Returvärde

- Om specificerad person har behörighetsstyrande egenskaper (personen är kopplad till en eller flera giltiga MiU) returneras dessa i objekt av XML-typen *MiuInformation*. Ett objekt per kopplad MiU returneras. Listan med funna *MiuInformation* packas in i ett objekt av XML-typen *GetMiuForPersonResponse*.
- Observera att enbart MiU:s med aktuella kopplingar (aktuell tidpunkt ligger inom kopplingens start och slutdatum) returneras.
- Om personen inte har några behörighetsegenskaper (personen är inte kopplad till någon giltig MiU) returneras *null*.
- Om personen inte hittas i HSA-katalogen returneras ett *fault*.



GetMiuForPersonResponse

Parameter	Datotyp	Antal	Beskrivning
MiuInformation	Se nedan	0 – n	Funna behörighetsegenskaper (MiU:s) för specificerad person.

MiuInformation

Parameter	Datotyp	Antal	Beskrivning
miuName HSA: Objektamn LDAP: cn	String	1 – 1	Namn på aktuell MiU (commission).
hsalentity HSA: HSA-id LDAP: hsalentity	String	1 – 1	HSA-identitet för aktuell MiU (commission).
miuPurpose HSA: medarbetaruppdragets syfte LDAP: hsaCommissionPurpose	String	1 – 1	Syfte med aktuell MiU. Aktuella syften specificeras i HSA kodtabell. För mer information se referens [1].
careUnitHsalentity HSA: HSA-id LDAP: hsalentity	String	0 – 1	HSA-identitet på den vårdenhet aktuell MiU (commission) tillhör. Lämnas tom om MiU inte tillhör en vårdenhet.
careUnitName HSA: Enhetsnamn eller Organisationsnamn LDAP: ou eller o	String	0 – 1	Namn på den vårdenhet aktuell MiU (commission) tillhör. Lämnas tom om MiU inte tillhör en vårdenhet.
careUnitStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdenhetens verksamhet. Lämnas tom om startdatum saknas eller om MiU inte tillhör en vårdenhet.
careUnitEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdenhetens verksamhet. Lämnas tom om slutdatum saknas eller MiU inte tillhör någon vårdenhet.
careGiver HSA: HSA-id LDAP: hsalentity	String	0 – 1	HSA-identitet på den vårdgivare aktuell MiU (commission) tillhör. Observera att en MiU inte är direkt kopplad till en vårdgivare utan tillhör den vårdgivare som MiU:ns vårdenhet tillhör.
careGiverName HSA: Enhetsnamn eller Organisationsnamn	String	0 – 1	Namn på den vårdgivare aktuell MiU tillhör.



Parameter	Datotyp	Antal	Beskrivning
LDAP: ou eller o			
careGiverOrgNo HSA: Organisationsnummer LDAP: orgNo	String	0 – 1	Organisationsnummer för den vårdgivare aktuell MiU (commission) tillhör.
careGiverStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdgivarens verksamhet.
careGiverEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdgivarens verksamhet.
personalPrescriptionCode HSA: Förskrivarkod LDAP: personalPrescriptionCode	String	0 – 1	Förskrivarkod för specificerad person. Om personlig förskrivarkod saknas läggs eventuell gruppförskrivarkod i detta fält.
hsaTitles	Collection	1 – 1	
hsaTitle HSA: Legitimerad yrkesgrupp LDAP: hsaTitle	String	0 – n	HSA-titel för specificerad person. Aktuella HSA-titlar specificeras i HSA kodtabell. För mer information se referens [3].
miuRights	Collection	1 – 1	
miuRight HSA: Medarbetaruppdragets rättigheter LDAP: hsaCommissionRight	String	0 – n	Rättighet för aktuell MiU (commission), För mer information se avsnitt 0.
hsaSystemRoles	Collection	1 – 1	
hsaSystemRole HSA: individuell behörighets-egenskap för it-tjänster LDAP: hsaSystemRole	String	0 – n	Systemroller kopplade till specificerad person. För mer information se avsnitt 0.
hsalentityPerson HSA: HSA-id LDAP: hsalentity	String	1 – 1	HSA-identitet för aktuell Person
hsaProtectedPerson HSA: skyddad person LDAP: hsaProtectedPerson	Boolean	0 – 1	true: om person är skyddad person false eller uteblivet attribut om personen inte är skyddad.



Parameter	Datotyp	Antal	Beskrivning
paTitleCode HSA: befattningskod LDAP: paTitleCode	String	0 – n	Befattningskoder kopplade till specificerad person.
givenName HSA: förnamn LDAP: givenName	String	0 – 1	Förnamn för person
middleAndSurName HSA: mellannamn + efternamn LDAP: middleName + surName (sn)	String	1 – 1	Mellannamn + efternamn för person. Namnen är separerade med mellanslag.
feignedMiuObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om medarbetaruppdraget är ett fingerat objekt.
feignedCareGiverObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdgivaren är ett fingerat objekt.
feignedCareUnitObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdenheten är ett fingerat objekt.
feignedPersonObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om personen är ett fingerat objekt.

MiuRights

Rättigheterna för en MiU specificeras i en sträng enligt följande syntax,
<aktivitet>;<informationstyp>;<organisationsomfång>.

- **Aktivitet:** Vad som får göras med aktuell information. Aktuella aktiviteter specificeras i HSA kodtabell. För mer information se referens [2].
- **Informationstyp:** Den typ av information aktiviteten får utföras på. Aktuella informationstyper specificeras i HSA kodtabell. För mer information se referens [2].
- **Organisationsomfång:** Inom vilken sfär får specificerad person komma åt informationen. Aktuella informationsomfång specificeras i HSA kodtabell. För mer information se referens [2].



HsaSystemRoles

De så kallade systemroller (*hsaSystemRole*) som är kopplade till aktuell person. En systemroll specificeras i en sträng med följande syntax, *<systemidentitet>;<systemroll>*.

4.5.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetMiuForPerson*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetMiuForPerson>
      <urn:hsaIdentity>SE1000-007</urn:hsaIdentity>
    </urn:GetMiuForPerson>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetMiuForPersonResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <MiuInformation>
        <miuName>Läkare</miuName>
        <hsaIdentity>SE1000-010</hsaIdentity>
        <miuPurpose>Vård och behandling</miuPurpose>
        <careUnitHsaIdentity>
          SE1000-002
        </careUnitHsaIdentity>
        <careUnitName>Kirurgavdelningen</careUnitName>
        <careGiver>SE1000-001</careGiver>
        <careGiverOrgNo>1001</careGiverOrgNo>
        <personalPrescriptionCode>
          1007
        </personalPrescriptionCode>
      </MiuInformation>
    </GetMiuForPersonResponse>
  </soap:Body>
</soap:Envelope>
```



```
<hsaTitles>
  <hsaTitle>Chef</hsaTitle>
  <hsaTitle>Läkare</hsaTitle>
</hsaTitles>
<miuRights>
  <miuRight>Läsa;dia;SJF</miuRight>
  <miuRight>Läsa;fun;SJF</miuRight>
  <miuRight>Läsa;lkm;SJF</miuRight>
  <miuRight>Läsa;lko;SJF</miuRight>
</miuRights>
<hsaSystemRoles>
  <hsaSystemRole>VpW;Redaktör</hsaSystemRole>
</hsaSystemRoles>
</MiuInformation>
</GetMiuForPersonResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetMiuForPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1000-007");
```



```
// Call hsaws
GetMiuForPersonResponseType response =
    hsaws.getMiuForPerson(logicalAddress, id, parameters);

// Handle response
for (MiuInformationType miu : response.getMiuInformation()) {
    System.out.println("Miu name: " + miu.getMiuName());
    System.out.println("Miu purpose: " + miu.getMiuPurpose());
}
```

EJ FÖR NYANS SLUTNING



4.6 GetPriceUnitsForAuth

GetPriceUnitsForAuth används för att söka ut vårdenheter, funktioner och publika personer inom tandvården för sökbaser, hsaIdentiteter eller organisationsnummer.

Observera att denna metod primärt är till för användning i VpW.

4.6.1 Inparametrar

Minst en och endast en av parametrarna searchBases, hsaIdentitySearchBases och orgIds är obligatorisk inparameter.

Inparametrar som genererar sökbaser som börjar med attribut c eller l, exempelvis c=se, är inte tillåtna och kommer inte att returnera några enheter.

Parameter	Datotyp	Antal	Beskrivning
searchBases		0 - 1	
searchBase	DN	1 - n	Sökbaser för sökning av publika tandvårdsenheter i HSA-katalogen.
hsaIdentitySearchBases		0 - 1	
hsaIdentitySearchBase	HsaIdentitySearchBase (se 0)	1 - n	HSA-identiteter för sökbaser för sökning av publika tandvårdsenheter i HSA-katalogen.
orgIds		0 - 1	
orgId	String	1 - n	Organisationsnummer för enhet som ska användas som sökbaser vid sökning av publika tandvårdsenheter i HSA-katalogen.
getPersons	Boolean	1	true: om publika personer ska hämtas
hsaTitles		0 - 1	
hsaTitle	String	1 - n	Yrkesroller för personer som ska hämtas

HsaIdentitySearchBase

Parameter	Datotyp	Antal	Beskrivning
searchBase	DN	1	Sökbaser för sökning av HSA-identitet
hsaIdentity	String	1	HSA-identitet för sökbaser för sökning av publika tandvårdsenheter i HSA-katalogen



4.6.2 Returvärde

- För varje sökbas returneras en DentalCareUnit med underliggande träd av DentalCareUnits, DentalCareFunctions och dentalCarePersons. Alla enheter under sökbas kommer att returneras oavsett om de är publika eller tandvårdsenheter.
- Arkiverade enheter, personer eller roller returneras ej (objectClass=hsaArchivedObject)
- Enhet, person eller roll som ligger under arkiverad enhet returneras ej.

GetPriceUnitsForAuthResponse

Parameter	Datatyp	Antal	Beskrivning
DentalCareUnit	Se 0	0 – n	Funna enheter under sökbas.

DentalCareUnit

Parameter	Datatyp	Antal	Beskrivning
hsalidentity	String	1	HSA-identitet för enhet
dentalCareUnitBaseName	String	1	Namn på enhet
hsaDestinationIndicator	String	1	"03" om enhet är publik, "null" för övriga
dentalUnit	Boolean	1	True om enhet är tandvårdsenhet
feignedObject	Boolean	0 - 1	Anger om enheten är ett fingerat objekt.
DentalCareUnit	DentalCareUnit	0 – n	Enheter som ligger under denna enhet
dentalCareFunction	dentalCareFunction	0 - n	Funktioner som ligger under enhet
dentalCarePerson	dentalCarePerson	0 – n	Publika personer som ligger under enhet

dentalCareFunction

Parameter	Datatyp	Antal	Beskrivning
hsald	String	1	HSA-identitet för funktion
functionName	String	1	Namn på funktion
hsaDestinationIndicator	String	1	"03" om funktion är publik, "null" för övriga
dentalUnitFunction	Boolean	1	True om funktion är tandvårdsfunktion
feignedObject	Boolean	0 - 1	Anger om funktionen är ett fingerat objekt.



dentalCarePerson

Parameter	Datotyp	Antal	Beskrivning
hsald	String	1	HSA-identitet för person
givenName	String	0 - 1	Förnamn för person.
middleName	String	0 – 1	Mellannamn för person
surName	String	1	Efternamn för person
title	String	0 – 1	Titel för person
feignedObject	Boolean	0 - 1	Anger om personen är ett fingerat objekt.

4.6.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetPriceUnitsForAuth*. Exemplet ges i form av SOAP-meddelanden.

Anrop med DN som indata

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetPriceUnitsForAuth>
      <urn:searchBases>
        <urn:searchBase>
          ou=Externa,o=Landstinget,l=län,c=SE
        </urn:searchBase>
        <urn:searchBase>
          ...
        </urn:searchBase>
      </urn:searchBases>
      <urn:getPersons>>false</urn:getPersons>
    </urn:GetPriceUnitsForAuth>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
```



```
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
  <GetPriceUnitsForAuthResponse
    xmlns="urn:riv:hsa:HsaWsResponder:3"
    xmlns:ns2="http://www.w3.org/2005/08/addressing">
    <DentalCareUnit>
      <hsaIdentity>SE1000-005</hsaId>
      <dentalCareUnitBaseName>Externa</name>
      <dentalUnit>>false</dentalUnit>
      <dentalCareUnit>
        <hsaId>SE1000-045</hsaId>
        <name>Specialisttandvården barntandvård</name>
        <hsaDestinationIndicator>
          03
        </hsaDestinationIndicator>
        <dentalUnit>>true</dentalUnit>
        <dentalFunction>
          <hsaId>SE1000-049</hsaId>
          <name>Funktion1</name>
          <hsaDestinationIndicator>
            03
          </hsaDestinationIndicator>
          <dentalUnitFunction>
            True
          </dentalUnitFunction>
        </dentalFunction>
      </dentalCareUnit>
      <dentalCareUnit>
        ...
      </DentalCareUnit>
      <DentalCareUnit>
        ...
    </GetPriceUnitsForAuthResponse >
  </soap:Body>
</soap:Envelope>
```

Anrop med HSA-id som indata

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetPriceUnitsForAuth>
      <urn:hsaIdSearchBases>
        <urn:hsaIdSearchBase>
```



```
<urn:searchBase>
  o=Landstinget,l=länet,c=SE
</urn:searchBase>
<urn:hsaId>SE1000-85</urn:hsaId>
<urn:hsaIdSearchBase>
<urn:hsaIdSearchBase>
  <urn:searchBase>
    o=Landstinget2,l=länet2,c=SE
  </urn:searchBase>
  <urn:hsaId>SE1000-96</urn:hsaId>
  <urn:hsaIdSearchBase>
</urn:hsaIdSearchBases>
<urn:getPersons>>false</urn:getPersons>
</urn:GetPriceUnitsForAuth>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
<soap:Body>
  <GetPriceUnitsForAuthResponse
    xmlns="urn:riv:hsa:HsaWsResponder:3"
    xmlns:ns2="http://www.w3.org/2005/08/addressing">
    <DentalCareUnit>
      <hsaIdentity>SE1000-005</hsaId>
      <dentalCareUnitBaseName>Externa</name>
      <dentalUnit>>false</dentalUnit>
    </DentalCareUnit>
    <DentalCareUnit>
      <hsaId>SE1000-045</hsaId>
      <name>Specialisttandvården barntandvård</name>
      <hsaDestinationIndicator>
        03
      </hsaDestinationIndicator>
      <dentalUnit>>true</dentalUnit>
      <dentalFunction>
        <hsaId>SE1000-049</hsaId>
        <name>Funktion1</name>
        <hsaDestinationIndicator>
          03
        </hsaDestinationIndicator>
        <dentalUnitFunction>
          True
        </dentalUnitFunction>
      </dentalFunction>
    </DentalCareUnit>
    <DentalCareUnit>
      ...
    </DentalCareUnit>
  </DentalCareUnit>
```



```
...
  </GetPriceUnitsForAuthResponse >
</soap:Body>
</soap:Envelope>
```

Anrop med organisationsnummer som indata samt personobjektshämtning

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetPriceUnitsForAuth>
      <urn:orgIds>
        <urn:orgId>559542-1840</urn:orgId>
        <urn:orgId>538542-9236</urn:orgId>
      </orgIds>
      <urn:hsaTitles>
        <urn:hsaTitle>Tandläkare</urn:hsaTitle>
        <urn:hsaTitle>Tandhygienist</urn:hsaTitle>
      </urn:hsaTitles>
      <urn:getPersons>true</urn:getPersons>
    </urn:GetPriceUnitsForAuth>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetPriceUnitsForAuthResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">

      <DentalCareUnit>
        <hsaIdentity>SE1000-005</hsaId>
        <dentalCareUnitBaseName>Externa</name>
        <dentalUnit>>false</dentalUnit>
      </DentalCareUnit>
      <dentalCareUnit>
        <hsaId>SE1000-045</hsaId>
        <name>Specialisttandvården barntandvård</name>
        <hsaDestinationIndicator>
          03
        </hsaDestinationIndicator>
      </dentalCareUnit>
    </GetPriceUnitsForAuthResponse>
  </soap:Body>
</soap:Envelope>
```



```
</hsaDestinationIndicator>
<dentalUnit>>true</dentalUnit>
<dentalFunction>
  <hsaId>SE1000-049</hsaId>
  <name>Funktion1</name>
  <hsaDestinationIndicator>
    03
  </hsaDestinationIndicator>
  <dentalUnitFunction>
    True
  </dentalUnitFunction>
</dentalFunction>
<DentalCarePerson>
  <hsaId>SE1000-37</hsaId>
  <givenName>Lars</givenName>
  <middleName>Olof</middleName>
  <surName>Larsson</surName>
  <title>Dentist</title>
</DentalCarePerson>
<DentalCarePerson>
  ...
</dentalCareUnit>
<dentalCareUnit>
  ...
  <DentalCarePerson>
    <hsaId>SE1000-34</hsaId>
    <givenName>Anna</givenName>
    <middleName>Karin</middleName>
    <surName>Andersson</surName>
    <title>Dentist</title>
  </DentalCarePerson>
  <DentalCarePerson>
    ...
  </DentalCareUnit>
<DentalCareUnit>
  ...
</GetPriceUnitsForAuthResponse >
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetMiuForPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
```



```
hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1000-007");

// Call hsaws
GetPriceUnitsForAuthResponseType response =
    hsaws.getPriceUnitsForAuth(logicalAddress, id, parameters);

// Handle response
for (DentalCareUnitType dcu : response.getDentalCareUnit()) {
    System.out.println("Dcu name: " + dcu.get...());
    System.out.println("Dcu purpose: " + dcu.get...());
}
```

4.7 GetCareUnitList

Denna metod hämtar en lista på alla ingående vårdenheter inom en vårdgivare. Alla vårdenheter som har vårdgivarens HsaId i attributet hsaResponsibleHealthCareProvider kommer att returneras.

Observera att denna metod primärt är till för användning i BIF.

4.7.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsaIdentity	String	1 - 1	HSA-identitet på vårdgivare.



searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <i>c=SE</i> som sökbas.
------------	----	-------	--

4.7.2 Returvärde

- Om specificerad enhet är en vårdgivare och finns inom *startDate* och *endDate* (om dessa attribut finns) returneras information om dess vårdenheter i objekt av XML-typen *GetCareUnitListResponse*.
- Om enheten inte är en vårdgivare returneras *null*.

GetCareUnitListResponse

Parameter	Datotyp	Antal	Beskrivning
careUnitGiverHsaldentity HSA: HSA-Id LDAP: hsaldentity	String	1 - 1	Vårdgivarens HSA-identitet
careUnitGiverName HSA: enhetsnamn LDAP: ou	String	1 - 1	Vårdgivarens namn
careUnitGiverStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdgivarens verksamhet.
careUnitGiverEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdgivarens verksamhet.
careUnits		1 - 1	
careUnit	CareUnit	0 - n	Ingående vårdenhet
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdgivaren är ett fingerat objekt.

CareUnit

Parameter	Datotyp	Antal	Beskrivning
hsaldentity HSA: HSA-Id LDAP: hsaldentity	String	1 - 1	HSA-identitet ingående enhet



careUnitName HSA: enhetsnamn LDAP: ou	String	1 - 1	Namn ingående enhet
careUnitStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdenhetens verksamhet.
careUnitEndDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdenhetens verksamhet.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdenheten är ett fingerat objekt.

4.7.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnitList*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetCareUnitList>
      <urn:hsaIdentity>SE1000-003</urn:hsaIdentity>
      <urn:searchBase>c=SE</urn:searchBase>
    </urn:GetCareUnitList>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```



```
<GetCareUnitListResponse
  xmlns="urn:riv:hsa:HsaWsResponder:3"
  xmlns:ns2="http://www.w3.org/2005/08/addressing">
  <careUnitGiverHsaIdentity>
    SE1000-003
  </careUnitGiverHsaIdentity>
  <careUnits>
    <careUnit>SE1000-007</careUnit>
    <careUnit>SE1000-009</careUnit>
    <careUnit>SE1000-012</careUnit>
    <careUnit>SE1000-078</careUnit>
    <careUnit>SE1000-064</careUnit>
  </careUnits>
</GetCareUnitListResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnitList*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1000-003");
```



```
// Call hsaws
GetCareUnitListResponseType response =
    hsaws.getCareUnitList(logicalAddress, id, parameters);

// Handle response
System.out.println("Care unit giver: " +
    response.getCareUnitGiverHsaIdentity());
System.out.println("Care giver: " +
    response.getCareUnitGiverHsaIdentity());
```

4.8 GetCareUnitMembers

Denna metod hämtar en lista på alla enheter kopplade till en specifik vårdenhet dvs alla enheter som pekats ut av vårdenhetens attribut `hsaHealthCareUnitMember`.

Observera att denna metod primärt är till för användning i Pascal.

4.8.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsaidentity	String	1 - 1	HSA-identitet på vårdenhet.
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <code>c=SE</code> som sökbas.

4.8.2 Returvärde

- Om specificerad enhet är en vårdenhet returneras information om dess vårdenheter i objekt av XML-typen *GetCareUnitMembersResponse*.

GetCareUnitMembersResponse

Parameter	Datotyp	Antal	Beskrivning
careUnitName HSA: Enhetsnamn LDAP: ou	String	1 – 1	Namn på vårdenheten.
careUnitStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdenhetens verksamhet.
careUnitEndDate HSA: Slutdatum	dateTime	0 - 1	Slutdatum för vårdenhetens verksamhet.



LDAP: endDate			
unitPrescriptionCode HSA: Arbetsplatskod LDAP: unitPrescriptionCode	String	1 – n	Arbetsplatskod för vårdenhet
telephoneNumber HSA: direkttelefon LDAP: telephoneNumber	String	0 – n	Vårdenhetens telefonnummer.
postalAddress HSA: Postadress LDAP: postalAddress	Address	0 – 1	Vårdenhetens postadress.
postalCode HSA: Postnummer LDAP: postalCode	String	0 – 1	Vårdenheten postnummer.
careUnitMembers	Collection	1 - 1	Kopplade enheter
careUnitMember	CareUnitMember	0 – n	Kopplad enhet
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdenheten är ett fingerat objekt.

Address

Adressinformation

Parameter	Datotyp	Antal	Beskrivning
addressLine	String	1 – n	Adressrad.

CareUnitMember

Parameter	Datotyp	Antal	Beskrivning
careUnitName HSA: Enhetsnamn LDAP: ou	String	1 – 1	Namn på kopplad enhet.
careUnitMemberStartDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för kopplade enhetens verksamhet.
careUnitMemberEndDate HSA: Slutdatum	dateTime	0 - 1	Slutdatum för kopplade enhetens verksamhet.



LDAP: endDate			
unitPrescriptionCode HSA: Arbetsplatskod LDAP: unitPrescriptionCode	String	1 – n	Arbetsplatskod för kopplad enhet
telephoneNumber HSA: direkttelefon LDAP: telephoneNumber	String	0 – n	Telefonnummer för kopplad enhet.
postalAddress HSA: Postadress LDAP: postalAddress	Address	0 – 1	Postadress för kopplad enhet.
postalCode HSA: Postnummer LDAP: postalCode	String	0 – 1	Postnummer för kopplad enhet.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om den kopplade enheten är ett fingerat objekt.

4.8.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnitMembers*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetCareUnitMembers>
      <urn:hsaIdentity>SE1000-003</urn:hsaIdentity>
      <urn:searchBase>c=SE</urn:searchBase>
    </urn:GetCareUnitMembers>
  </soap:Body>
</soap:Envelope>
```



Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetCareUnitMembersResponse
      <careUnitName>Kliniken</careUnitName>
      <unitPrescriptionCode>783450050</unitPrescriptionCode>
      <telephoneNumber>03487628
      <postalAddress>
        <addressLine>A-stads lasarett</addressLine>
        <addressLine>Akutvårdsenheten</addressLine>
        <addressLine>Södra Gatan 30</addressLine>
        <addressLine></addressLine>
        <addressLine>111 11</addressLine>
        <addressLine>A-stad</addressLine>
      </postalAddress>
      <postalCode>111 11</postalCode>
      <careUnitMembers>
        <careUnitMember>
          <careUnitName>Operationsavdelning</careUnitName>
          <unitPrescriptionCode>144615</unitPrescriptionCode>
          <telephoneNumber>0932226000</telephoneNumber>
          <postalAddress>
            <addressLine>A-stad Lasarett</addressLine>
            <addressLine>Operationsavdelning</addressLine>
            <addressLine>Södra Gatan 30</addressLine>
            <addressLine></addressLine>
            <addressLine>111 11</addressLine>
            <addressLine>A-stad</addressLine>
          </postalAddress>
          <postalCode>111 11</postalCode>
        </careUnitMember>
      </careUnitMembers>
    </GetCareUnitMembersResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnitMembers*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
```



```
hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
LookupHsaObjectType parameters = new LookupHsaObjectType();
parameters.setHsaIdentity("SE1000-003");

// Call hsaws
GetCareUnitMembersResponseType response =
    hsaws.getCareUnitMembers(logicalAddress, id, parameters);

// Handle response
System.out.println("Care unit name: " +
    response.getCareUnitName());
for(String phonenumber : response.getTelephoneNumber()) {
    System.out.println("Telephone number: " + phonenumber);
}
...
```

4.9 HsaWsSimpleLookup

HsaWsSimpleLookup kan användas för sökning (*Search*) eller en uppslagning (*Lookup*).

En *search* tillåter olika sökoperatörer, men är låst till attributet *simpleSearch*.

En *lookup* tillåter olika sökattribut, men är låst till sökoperatören *EXACT*.

OBS! Ingen ny behörighet beviljas till denna metod, som på sikt helt kommer att ersättas med specifika metoder.



4.9.1 Inparametrar

HsawsSimpleLookupType

Parameter	Datatyp	Antal	Beskrivning
Search	search	0 – 1	Specificerar en sökning.
Lookup	lookup	0 – 1	Specificerar en uppslagning.
Attributes		0 – 1	
attribute	String	0 – n	<p>Attribut som skall ingå i svaret. Om inga attribut anges returneras enbart DN för utsökta objekt.</p> <p>Specialfall:</p> <ul style="list-style-type: none"> • Katalogansvarig: Om attributet <i>hsaDirectoryContact</i> anges så returneras katalogansvarige för varje objekt i svaret. • Reserverade funktionsnamn: För att söka efter något av de reserverade funktionsnamnen (Avbokning, Bokning, Rådgivning, Receptförnyelse, Sjukresebeställning) anges <i>funktionsnamn:attributnamn</i>, exempelvis <i>Avbokning:hsaTelephoneNumber</i>.
searchBase	String	0 – 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <i>c=SE</i> som sökbas.

Observera att enbart en av Search eller Lookup skall användas per anrop. Se exempel i avsnitt 4.9.3.

search

Parameter	Datatyp	Antal	Beskrivning
searchAttribute (attribut)	String	0 – 1	LDAP-attribut som skall användas vid sökningen . Låst till attributet <i>simpleSearch</i> .
searchOperator (attribut)	String	1 – 1	<p>Sökoperatör som skall användas i sökningen. Sökoperatörerna motsvarar motsvarande sökoperatörer i LDAP.</p> <ul style="list-style-type: none"> • STARTS_WITH • ENDS_WITH • CONTAINS • EXACT



Parameter	Datotyp	Antal	Beskrivning
(värde till Search)	String	1 – 1	Söksträng.

lookup

Parameter	Datotyp	Antal	Beskrivning
searchAttribute (attribut)	String	0 – 1	LDAP-attribut som skall användas vid sökningen. Om inget värde anges används <i>hsalidentity</i> .
searchOperator (attribut)	String	1 – 1	Sökoperator som skall användas i sökningen. Läst till sökoperatorm <i>EXACT</i> .
(värde till Search)	String	1 – 1	Söksträng.

4.9.2 Returvärde

- Sökresultatet returneras i ett objekt av XML-typen *HsawsSimpleLookupResponse*.
- Om sökningen inte ger några träffar så returneras *null*.
- Ett *fault* returneras om:
 - Anropande system inte finns i HSA-katalogen
 - Sökattributet saknas i HSA-katalogens schema
- Arkiverade enheter returneras om klientanvändaren har rätt att läsa dem (objectClass=hsaArchivedObject)

HsawsSimpleLookupResponse

Parameter	Datotyp	Antal	Beskrivning
responseValues		0 – n	En instans per utsökt LDAP-objekt.
DN	String	1 – 1	Dn för utsökt LDAP-objekt.
response		0 – n	Värden på efterfrågade attribut för utsökt LDAP-objekt.
attribute	String	1 – 1	Attributnamn
value	String	1 – 1	Attributvärde
feignedObject	Boolean	0 - 1	Anger om objektet är ett fingerat objekt.



4.9.3 Exempel

Search

I detta avsnitt ger vi exempel på ett typiskt anrop av *HsaWsSimpleLookup* för att göra en sökning på attributet *simpleSearch*, sökoperatorn *STARTS_WITH* och sökvärdet *Perlemo*, med sökbasen *l=BIF,c=SE*.

Svaret ska innehålla attributen *cn*, *hsaIdentity* och *hsaDirectoryContact*.

Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:HsawsSimpleLookup>
      <urn:Search
        searchAttribute="SimpleSearch"
        searchOperator="STARTS_WITH">
        Perlemo
      </urn:Search>
      <urn:searchBase>l=BIF,c=SE</urn:searchBase>
      <urn:attributes>
        <urn:attribute>cn</urn:attribute>
        <urn:attribute>hsaIdentity</urn:attribute>
        <urn:attribute>hsaDirectoryContact</urn:attribute>
      </urn:attributes>
    </urn:HsawsSimpleLookup>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <HsawsSimpleLookupResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <responseValues>
        <DN>
          cn=Nils Pärlemo,ou=BVC,o=Kvartersakuten,l=BIF,c=SE
        </DN>
      </responseValues>
    </HsawsSimpleLookupResponse>
  </soap:Body>
</soap:Envelope>
```



```
<response>
  <attribute>cn</attribute>
  <value>Nils Pärlemo</value>
</response>
<response>
  <attribute>hsaIdentity</attribute>
  <value>SE2000-003</value>
</response>
<response>
  <attribute>hsaDirectoryContact</attribute>
  <value>SE2000-005</value>
</response>
</responseValues>
</HsawsSimpleLookupResponse>
</soap:Body>
</soap:Envelope>
```

Lookup

I detta avsnitt ger vi exempel på ett typiskt anrop av *HsawsSimpleLookup* för att göra en uppslagning med attributet *hsaIdentity*, sökoperator *EXACT* och sökvärde *SE2000-004*, med sökbasen *l=BIF,c=SE*.

Svaret ska innehålla attributen *AVBOKNING:hsaTelephoneNumber*, *hsaIdentity* och *ou*. Dvs vi söker objektet med HSA-id=SE2000-004, och vill få tillbaka objektets

- HSA-id,
- Enhetsnamn och
- Telefonnummer från en underliggande funktion med det reserverade funktionsnamnet ”Avbokning” (om det finns en sådan, annars returneras null för det här attributet).

Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsawsResponder:3">
  <soap:Header>
    <add:MessageID>23</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:HsawsSimpleLookup>
      <urn:Lookup
        searchAttribute="hsaIdentity"
        searchOperator="EXACT">
        SE2000-004
      </urn:Lookup>
    </urn:searchBase></urn:searchBase>
```



```
<urn:attributes>
  <urn:attribute>hsaIdentity</urn:attribute>
  <urn:attribute>
    AVBOKNING:hsaTelephoneNumber
  </urn:attribute>
  <urn:attribute>ou</urn:attribute>
</urn:attributes>
</urn:HsawsSimpleLookup>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <HsawsSimpleLookupResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <responseValues>
        <DN>ou=Husläkare,o=Kvartersakuten,l=BIF,c=SE</DN>
        <response>
          <attribute>ou</attribute>
          <value>Husläkare</value>
        </response>
        <response>
          <attribute>hsaIdentity</attribute>
          <value>SE2000-004</value>
        </response>
        <response>
          <attribute>
            AVBOKNING:hsaTelephoneNumber
          </attribute>
          <value>1234567890</value>
        </response>
      </responseValues>
    </HsawsSimpleLookupResponse>
  </soap:Body>
</soap:Envelope>
```

4.10 IsAuthorizedToSystem

IsAuthorizedToSystem används för att kontrollera om en person har behörighet till ett system (enligt attributet hsaSystemRole). Om personen har behörighet returneras information om personen.



4.10.1 Inparametrar

Endast en av parametrarna `hsaIdentity` och `personalIdentityNumber` skall och måste användas.

Parameter	Datatyp	Antal	Beskrivning
<code>hsaIdentity</code>	String	0 - 1	HSA-identitet för den användare vars behörighet skall kontrolleras.
<code>personalIdentityNumber</code>	String	0 - 1	Personnummer för den användare vars behörighet skall kontrolleras .
<code>System</code>	String	1 - 1	Namn på system behörighet skall kontrolleras för. Versaler och gemener hanteras som samma bokstav.
<code>searchBase</code>	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används <code>c=SE</code> som sökbas.

4.10.2 Returvärde

- En instans av XML-typen *IsAuthorizedToSystem* returneras. Detta objekt anger om personen har behörighet till specificerat system eller inte.
- Behörig i detta sammanhang specificeras som så att användaren HSA-objekt innehåller ett värde i attributet *hsaSystemRole* med specificerat system som systemdel (t.ex. värdet *vpw;admin* i *hsaSystemRole* ger behörighet till systemet *vpw*).
- Om användaren har behörighet returneras användarinformation i en instans av XML-typen *userInformation*.
- Observera att en användare kan ha flera HSA-objekt i katalogen där olika objekt har olika behörigheter till olika system. Om flera av användarens HSA-objekt har behörighet till specificerat system returneras en instans av XML-typen *userInformation* för varje behörigt HSA-objekt.

IsAuthorizedToSystemResponse

Parameter	Datatyp	Antal	Beskrivning
<code>isAuthorized</code>	Boolean	1 - 1	<i>true</i> om personen har behörighet till specificerat system. Endast system-id godkända av HSA Förvaltningsgrupp ger behörighet i denna sökning. <i>false</i> om personen inte är behörig.
<code>userInformations</code>	Collection	1 - 1	



userInfo	User Information	0 – n	Information om personen. Om personen har flera HSA-objekt som är behörig till specificerat system returneras en instans per HSA-objekt.
----------	------------------	-------	--

UserInformation

Information om den behöriga personen. För detaljerad information om respektive parameter se HSA-specifikationen [6].

Parameter	Datatyp	Antal	Beskrivning
description HSA: beskrivning LDAP: description	String	0 – 1	Generell beskrivning.
email HSA: e-postadress LDAP: mail	String	0 – 1	Epostadress.
givenName HSA: tilltalsnamn LDAP: givenName	String	0 – 1	Tilltalsnamn.
hsalentity HSA: HSA-id LDAP: hsalentity	String	1 – 1	HSA-identitet.
hsaSwitchboardNumbers	Collection	1 – 1	
hsaSwitchboardNumber HSA: Växeltefon LDAP: hsaSwitchboardNumber	Telefon	0 – n	Telefonnummer till växel
hsaSystemRoles	Collection	1 – 1	
hsaSystemRole HSA: Individuell behörighets-egenskap för IT-tjänster LDAP: hsaSystemRole	Hsa System Role	0 – n	Systemroller (se avsnitt 0). Observera att enbart systemroller för i anropet specificerat system levereras.
hsaTelephoneNumbers	Collection	1 – 1	
hsaTelephoneNumber HSA: tjänstetelefon LDAP: hsaTelephoneNumber	String	0 – 1	Tjänstetelefonnummer. Skall inte visas för allmänheten.



labeledUri HSA: webbadress LDAP: labeledURI	String	0 – 1	Webbadress.
mobileNumbers	Collection	1 – 1	
mobileNumber HSA: mobiltelefon LDAP: mobile	String	0 – n	Mobiltelefonnummer.
nickname HSA: smeknamn LDAP: nickname	String	0 – 1	Smeknamn för användning istället för tilltalsnamn.
specialityCodes	Collection	1 – 1	
specialityCode HSA: specialitetskod LDAP: specialityCode	String	0 – n	Klassificeringskod för specialistutbildning utöver grundutbildning.
specialityNames	Collection	1 – 1	
specialityName HSA: specialitet LDAP: specialityName	String	0 – n	Specialistutbildning utöver grundutbildning.
surName HSA: efternamn LDAP: sn	String	1 – 1	Efternamn.
telephoneHours	Collection	1 – 1	
telephoneHour HSA: telefontid LDAP: telephoneHour	TimeSpan	0 – n	Telefontider för publik telefon (telephoneNumbers). För mer information se avsnitt 0.
telephoneNumbers	Collection	1 – 1	
telephoneNumber HSA: direkttelefon LDAP: telephoneNumber	String	0 – n	Publikt direkttelefonnummer.
title HSA: titel LDAP: title	String	0 – 1	Yrkestitel i fritext
DN	DN	1	DN för person



middleName	String	0 – 1	Mellannamn för person
hsaProtectedPerson HSA: skyddad person LDAP: hsaProtectedPerson	Boolean	0 – 1	true: om person är skyddad person false eller uteblivet attribut om personen inte är skyddad.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om personen är ett fingerat objekt.

HsaSystemRole

För detaljerad information om respektive parameter se HSA-specifikationen [6].

Parameter	Datotyp	Antal	Beskrivning
System	String	1 – 1	System/tjänst som rollen gäller för.
Role	String	1 – 1	Roll i specificerat system/tjänst.

4.10.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *IsAuthorizedToSystem*. Exemplet ges i form av SOAP-meddelanden.

Anrop – Behörig

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soapenv:Body>
    <urn:IsAuthorizedToSystem>
      <urn:hsaIdentity>SE1000-004</urn:hsaIdentity>
      <urn:system>vpw</urn:system>
    </urn:IsAuthorizedToSystem>
  </soapenv:Body>
</soap:Envelope>
```



Resultat – Behörig

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <IsAuthorizedToSystemResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <isAuthorized>true</isAuthorized>
      <userInformations>
        <userInformation>
          <description>Kalles beskrivning</description>
          <email>kalle@sos.se</email>
          <givenName>Kalle</givenName>
          <hsaIdentity>SE1000-004</hsaIdentity>
          <hsaSwitchboardNumbers>
            <hsaSwitchboardNumber>
              08111111
            </hsaSwitchboardNumber>
          </hsaSwitchboardNumbers>
          <hsaSystemRoles>
            <hsaSystemRole>
              <system>vpw</system>
              <role>admin</role>
            </hsaSystemRole>
          </hsaSystemRoles>
          <hsaTelephoneNumbers>
            <hsaTelephoneNumber>
              08333333
            </hsaTelephoneNumber>
            <hsaTelephoneNumber>
              08444444
            </hsaTelephoneNumber>
          </hsaTelephoneNumbers>
          <labeledUri>http://kalle.sos.se/</labeledUri>
          <mobileNumbers>
            <mobileNumber>08888888</mobileNumber>
            <mobileNumber>08999999</mobileNumber>
          </mobileNumbers>
          <nickname>Kåce</nickname>
          <specialityCodes>
            <specialityCode>111</specialityCode>
            <specialityCode>222</specialityCode>
          </specialityCodes>
          <specialityNames>
            <specialityName>kirurgi</specialityName>
            <specialityName>medicin</specialityName>
          </specialityNames>
          <surName>Karlsson</surName>
          <telephoneHours>
            <telephoneHour>
              <fromDay>1</fromDay>
            </telephoneHour>
          </telephoneHours>
        </userInformation>
      </userInformations>
    </IsAuthorizedToSystemResponse>
  </soap:Body>
</soap:Envelope>
```



```
<fromTime>10:00:00.895+01:00</fromTime>
<toDay>4</toDay>
<toTime>17:00:00.895+01:00</toTime>
<comment>Öppet</comment>
</telephoneHour>
</telephoneHours>
<telephoneNumbers>
  <telephoneNumber>08555555</telephoneNumber>
  <telephoneNumber>08666666</telephoneNumber>
</telephoneNumbers>
<title>Chifen</title>
<DN>cn=Kalle Karlsson,ou=enheten,o=ozon,c=se</DN>
<middleName>Bson</middleName>
</userInformation>
</userInformations>
</IsAuthorizedToSystemResponse>
</soap:Body>
</soap:Envelope>
```

Anrop – Ej behörig

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soapenv:Body>
    <urn:IsAuthorizedToSystem>
      <urn:hsaIdentity>SE1000-004</urn:hsaIdentity>
      <urn:system>unknown</urn:system>
    </urn:IsAuthorizedToSystem>
  </soapenv:Body>
</soap:Envelope>
```

Resultat – Ej behörig

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <IsAuthorizedToSystemResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <isAuthorized>>false</isAuthorized>
      <userInformations/>
    </IsAuthorizedToSystemResponse>
  </soap:Body>
```



```
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *IsAuthorizedToSystem*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
IsAuthorizedToSystemType parameters = new
    IsAuthorizedToSystemType();
parameters.setHsaIdentity("SE1000-004");
parameters.setSystem("vpw");

// Call hsaws
IsAuthorizedToSystemResponseType response =
    hsaws.isAuthorizedToSystem(logicalAddress, id, parameters);

// Handle response
if (response.isIsAuthorized()) {
    for (HsaUserType person : response.getUserInformations())
        .getUserInformation() {
        System.out.println("HSA-identity: " +
            person.getHsaIdentity());
        System.out.println("Email: " + person.getEmail());
    }
}
```



4.11 Ping

Ping används för att testa åtkomsten till hsaws2 och den bakomliggande HSA-katalogen. Metoden används primärt för övervakning och felsökning.

4.11.1 Inparametrar

Metoden har inga inparametrar.

4.11.2 Returvärde

- Om hsaws2 är nåbar och en sökning kan utföras i HSA-katalogen returneras information om version av hsaws2 och HSA-miljö i ett objekt av XML-typen *PingResponse*.

PingResponse

Parameter	Datotyp	Antal	Beskrivning
message	String	1-1	Information om version av hsaws2 och HSA-miljö.

4.11.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *Ping*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:Ping/>
  </soap:Body>
</soap:Envelope>
```



Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <PingResponse>
      <message>HSA Webservises 2.5.1 - Test 2</message>
    </PingResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *Ping*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
PingType parameters = new PingType();

// Call hsaws
PingResponseType response = hsaws.ping(logicalAddress, id,
    parameters);

// Handle response
```



```
System.out.println("Ping message: " + response.getMessage());
```

4.12 VpwGetPublicUnits

VpwGetPublicUnit returnerar en förgenererad lista med publika enheter för visning i VpW. Enhetslistan beräknas med jämna mellanrum av en fristående batchrutin. Ett anrop till VpwGetPublicUnits returnerar den senaste genererade enhetslistan.

En publik enhet är en enhet eller funktion som uppfyller följande krav;

- Skall vara av typen *hsaOrganizationExtension* eller *hsaRoleExtension*.
- Skall ha attributet *hsaDestinationIndicator* satt till värdet *03* (allmänhet) eller *04* (öppen).

4.12.1 Inparametrar

Metoden har inga inparametrar.

4.12.2 Returvärde

- Information om publika enheter i ett objekt av XML-typen *VpwGetPublicUnitsResponse*.

VpwGetPublicUnitsResponse

Parameter	Datatyp	Antal	Beskrivning
startDate	dateTime (XML)	1 – 1	Tid när batchrutinen för generering av aktuell enhetslista startade.
endDate	dateTime (XML)	1 – 1	Tid när batchrutinen för generering av aktuell enhetslista avslutades.
publicUnits			
publicUnit	PublicUnit	0 – n	Information om en publik enhet.
unavailableLocations			
unavailableLocation	UnavailLoc	0 – n	Information om ett otillgängligt delträd.
Statistics	Statistics	1 – 1	Statistikinformation.

PublicUnit

Information om en publik enhet.



Parameter	Datotyp	Antal	Beskrivning
hsalentity HSA: HSA-id LDAP: hsalentity	String	1 – 1	HSA-identitet.
Name HSA: Enhetsnamn eller Objektnamn LDAP: o, ou eller cn	String	1 – 1	Publikt officiellt namn. För mer information se avsnitt 4.3.2.
Location	String	1 – 1	DN för det katalogdelträd som enheten tillhör.
searchBase	String	1 – 1	Sökbas för användning vid utsökning av information för aktuell enhet med hjälp av metoden GetHsaUnit (avsnitt 4.2).
modifyTimestamp	dateTime (XML)	1 – 1	Tidpunkt för senaste uppdatering av enhetsobjekt.

UnavailLoc

Information om ett katalogträd som inte var tillgänglig när aktuell enhetslista genererades.

Ej längre aktuell. Strukturen finns kvar, men detta är en rest från tidigare var HSA-katalogen en distribuerad katalog som byggdes upp av flera länkade katalogservrar. En länkad regional underkatalog kunde då av tekniska skäl vara otillgänglig när genereringen av aktuell enhetslista utfördes.

Parameter	Datotyp	Antal	Beskrivning
Location	String	1 – 1	DN för katalogträdet.

Statistics

Statistikinformation.

Parameter	Datotyp	Antal	Beskrivning
totalPublicUnits	String	1 – 1	Totalt antal publika enheter i aktuell enhetslista.
Locations		1 – 1	
location	StatLoc	0 – n	Statistikinformation om ett katalogträd.

StatLoc

Statistikinformation om ett delträd.

Parameter	Datotyp	Antal	Beskrivning
Location	String	1 – 1	DN för katalogträdet.



Parameter	Datotyp	Antal	Beskrivning
publicUnits	String	1 – 1	Antal publika enheter i delträdet.

4.12.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnit*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:VpwGetPublicUnits/>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <VpwGetPublicUnitsResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <startDate>2010-01-25T10:20:26.155+01:00</startDate>
      <endDate>2010-01-25T10:20:54.963+01:00</endDate>
      <publicUnits>
        <publicUnit>
          <hsaIdentity>SE1000-100</hsaIdentity>
          <name>Folktandvården</name>
          <location>
            o=Centrala landstinget,c=SE
          </location>
          <modifyTimestamp>
            2011-10-20T15:08:44.992+02:00
          </modifyTimestamp>
        </publicUnit>
        ...
      </publicUnits>
```



```
<unavailableLocations>
  <unavailableLocation>
    <location>
      l=Norra länet,c=SE
    </location>
  </unavailableLocation>
</unavailableLocations>
<statistics>
  <totalPublicUnits>234</totalPublicUnits>
  <locations>
    <location>
      <location>
        o=Centrala landstinget,c=SE
      </location>
      <publicUnits>128</publicUnits>
    </location>
    ...
  </locations>
</statistics>
</VpwGetPublicUnitsResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetCareUnit*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService = new
HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
```



```
id.setValue("123");
VpwGetPublicUnitsType parameters = new VpwGetPublicUnitsType();

// Call hsaws
VpwGetPublicUnitsResponseType response =
    hsaws.vpwGetPublicUnits(logicalAddress, id, parameters);

// Handle response
System.out.println("Start date: " + response.getStartDate());
System.out.println("Total public units: " +
    response.getStatistics().getTotalPublicUnits());

for (PublicUnitType publicUnit :
    response.getPublicUnits().getPublicUnit()) {
    System.out.println("Public unit: " +
        publicUnit.getHsaIdentity());
}
```

4.13 GetHsaPerson

GetHsaPerson hämtar information om en HSA-person (hsaPersonExtension).

4.13.1 Inparametrar

Endast en av parametrarna hsaIdentity och personalIdentityNumber skall och måste användas.

Parameter	Datotyp	Antal	Beskrivning
hsaIdentity	String	0 - 1	HSA-identitet för användaren vars information skall hämtas.
personalIdentityNumber	String	0 - 1	Personnummer för användaren vars information skall hämtas.
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används c=SE som sökbas.

4.13.2 Returvärde

- En lista med de personobjekt som matchar sökkriterierna av XML-typen *GetHsaPersonResponseType* returneras. Detta objekt innehåller information om personen som eftersöks.



- Om inga personobjekt hittas returneras en tom lista.

GetHsaPersonResponseType

Parameter	Datotyp	Antal	Beskrivning
userInformations	Collection	1 – 1	
userInformation	User Information	0 – n	Information om personen. Om personen har flera HSA-objekt returneras en instans per HSA-objekt.

UserInformation

Information om HSA-personen i form av GetHsaPersonHsaUserType. För detaljerad information om respektive parameter se HSA-specifikationen [6].

Parameter	Datotyp	Antal	Beskrivning
hsalentity HSA: HSA-id LDAP: hsalentity	String	1 – 1	HSA-identitet.
givenName HSA: tilltalsnamn LDAP: givenName	String	0 – 1	Tilltalsnamn.
sn HSA: efternamn LDAP: sn	String	1 – 1	Efternamn.
middleName HSA: mellannamn LDAP: middleName	String	0 – 1	Mellannamn för person
nickName HSA: smeknamn LDAP: nickName	String	0 – 1	Smeknamn för användning istället för tilltalsnamn.
mail HSA: e-postadress LDAP: mail	String	0 – 1	Epostadress.
telephoneNumbers	Collection	1 – 1	



telephoneNumber HSA: direkttelefon LDAP: telephoneNumber	String	0 – n	Publikt direkttelefonnummer.
hsaSwitchboardNumber	String	0 – 1	Telefonnummer till växel.
hsaTelephoneNumbers	Collection	1 – 1	
hsaTelephoneNumber HSA: tjänstetelefon LDAP: hsaTelephoneNumber	String	0 – 1	Tjänstetelefonnummer. Skall inte visas för allmänheten.
mobiles	Collection	1 – 1	
mobile HSA: mobiltelefon LDAP: mobile	String	0 – n	Mobiltelefonnummer.
facsimilleTelephoneNumbers	Collection	1 – 1	
facsimilleTelephoneNumber HSA: faxnummer LDAP: facsimilleTelephoneNumber	String	0 – n	Faxnummer.
telephoneHours	Collection	1 – 1	
telephoneHour HSA: telefontid LDAP: telephoneHour	TimeSpan	0 – n	Telefontider för publik telefon (telephoneNumbers). För mer information se avsnitt TimeSpan.
postalAddress HSA: postadress LDAP: postalAddress	AddressType		Postadress.
labeledUri HSA: webbadress LDAP: labeledURI	String	0 – 1	Webbadress.
description HSA: beskrivning LDAP: description	String	0 – 1	Generell beskrivning.
title HSA: titel LDAP: title	String	0 – 1	Yrkestitel i fritext



hsaTitles	Collection	1 – 1	
hsaTitle HSA: Legitimerad yrkesgrupp LDAP: hsaTitle	String	0 – n	HSA-titel för specificerad person. Aktuella HSA-titlar specificeras i HSA kodtabell. För mer information se referens [3].
specialityNames	Collection	1 – 1	
specialityName HSA: specialitet LDAP: specialityName	String	0 – n	Specialistutbildning utöver grundutbildning.
specialityCodes	Collection	1 – 1	
specialityCode HSA: specialitetskod LDAP: specialityCode	String	0 – n	Klassificeringskod för specialistutbildning utöver grundutbildning.
DN	DN	1	DN för person
hsaProtectedPerson HSA: skyddad person LDAP: hsaProtectedPerson	Boolean	0 – 1	true: om person är skyddad person false eller uteblivet attribut om personen inte är skyddad.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om personen är ett fingerat objekt.

4.13.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHsaPerson*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetHsaPerson>
```



```
<urn:personalIdentityNumber>
  191212121212
</urn:personalIdentityNumber>
</urn:GetHsaPerson>
<urn:searchBase>c=SE</urn:searchBase>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
<soap:Body>
  <GetHsaPersonResponse
    xmlns="urn:riv:hsa:HsaWsResponder:3"
    xmlns:ns2="http://www.w3.org/2005/08/addressing">
    <userInformations>
      <userInformation>
        <description>Kalles beskrivning</description>
        <mail>kalle@sos.se</mail>
        <givenName>Kalle</givenName>
        <hsaIdentity>SE1000-004</hsaIdentity>>
        <hsaSwitchboardNumber>
          08111111
        </hsaSwitchboardNumber>
        <hsaTelephoneNumbers>
          <hsaTelephoneNumber>
            08333333
          </hsaTelephoneNumber>
          <hsaTelephoneNumber>
            08444444
          </hsaTelephoneNumber>
        </hsaTelephoneNumbers>
        <labeledUri>http://kalle.sos.se/</labeledUri>
        <mobiles>
          <mobile>08888888</mobile>
          <mobile>08999999</mobile>
        </mobile>
        <facsimileTelephoneNumbers>
          <facsimileTelephoneNumber>
            08888888
          </facsimileTelephoneNumber>
        </facsimileTelephoneNumbers>
        <nickname>Kåce</nickname>
        <specialityCodes>
          <specialityCode>111</specialityCode>
          <specialityCode>222</specialityCode>
        </specialityCodes>
        <specialityNames>
```



```
<specialityName>kirurgi</specialityName>
  <specialityName>medicin</specialityName>
</specialityNames>
<sn>Karlsson</sn>
<telephoneHours>
  <telephoneHour>
    <fromDay>1</fromDay>
    <fromTime>10:00:00.895+01:00</fromTime>
    <toDay>4</toDay>
    <toTime>17:00:00.895+01:00</toTime>
    <comment>Öppet</comment>
  </telephoneHour>
</telephoneHours>
<telephoneNumbers>
  <telephoneNumber>08555555</telephoneNumber>
  <telephoneNumber>08666666</telephoneNumber>
</telephoneNumbers>
<title>Chifen</title>
<DN>cn=Kalle Karlsson,ou=enheten,o=ozon,c=se</DN>
<middleName>Bson</middleName>
<postalAddress>
  <addressLine>Gästvägen 33</addressLine>
  <addressLine>12321 Stockholm</addressLine>
</postalAddress>
<hsaTitles>
  <hsaTitle>Konsult</hsaTitle>
</hsaTitles>
</userInformation>
</userInformations>
</GetHsaPersonResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHsaPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
```



```
BindingProvider.ENDPOINT_ADDRESS_PROPERTY,  
"https://10.253.18.149/svr-hsaws2/hsaws");  
  
// Create request parameters  
AttributedURIType logicalAddress = new AttributedURIType();  
logicalAddress.setValue("SE165565594230-1000");  
AttributedURIType id = new AttributedURIType();  
id.setValue("123");  
GetHsaPersonType parameters = new GetHsaPersonType();  
parameters.setPersonalIdentityNumber("191212121212");  
  
// Call hsaws  
GetHsaPersonResponseType response =  
    hsaws.getHsaPerson(logicalAddress, id, parameters);  
  
System.out.println("HsaIdentity: " +  
    response.getHsaIdentity());  
for (String title : response.getHsaTitle().getTitle())  
{  
    System.out.println("Hsa-title: " + title);  
}
```

4.14 GetStatisticsCareGiver

GetStatisticsCareGiver används för att söka ut information när ett enskilt intyg lagras i Intygstjänsters statistikdatabas. Det som hämtas är information om den vårdgivare som finns angiven på intyget.

4.14.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsaidentity	String	1 - 1	HSA-identitet för vårdgivare från intyget
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används c=SE som sökbas.



4.14.2 Returvärde

- Om specificerat hsaIdentity hör till en vårdgivare returneras information om vårdgivaren i objekt av XML-typen *GetStatisticsCareGiverResponse*.
- Om vårdgivare inte kan hittas eller om objektet inte är utpekad som vårdgivare returneras null.

GetStatisticsCareGiverResponse

Parameter	Datotyp	Antal	Beskrivning
hsaIdentity HSA: HSA-Id LDAP: hsaIdentity	String	1 - 1	HSA-identitet för vårdgivaren.
careGiverOrgNo HSA: Organisationsnummer LDAP: orgNo	String	0 - 1	Vårdgivarens organisationsnummer
startDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för vårdgivarens verksamhet
endDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för vårdgivarens verksamhet.
isArchived HSA: arkiverad LDAP: hsaArchivedObject	Boolean	0 - 1	Sätts till true om vårdgivaren är arkiverad. Sätts till false eller uteblivet attribut om enhet inte är arkiverad.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om vårdgivaren är fingerade objekt.

4.14.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsCareGiver*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
```



```
xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
<soap:Header>
  <add:MessageID>123</add:MessageID>
  <add:To>SE165565594230-1000</add:To>
</soap:Header>
<soap:Body>
  <urn:GetStatisticsCareGiver>
    <urn:hsaIdentity>SE1000-001</urn:hsaIdentity>
    <urn:searchBase>c=SE</urn:searchBase>
  </urn:GetStatisticsCareGiver>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetStatisticsCareGiverResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <hsaIdentity>SE1000-001</hsaIdentity>
      <careGiverOrgNo>5553-238947</careGiverOrgNo>
      <startDate>2013-08-07T10:54:37.000+02:00</startDate>
      <endDate>2014-01-17T09:56:00.000+01:00</endDate>
    </GetStatisticsCareGiverResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsCareGiver*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
```



```
"https://10.253.18.149/svr-hsaws2/hsaws");  
  
// Create request parameters  
AttributedURIType logicalAddress = new AttributedURIType();  
logicalAddress.setValue("SE165565594230-1000");  
AttributedURIType id = new AttributedURIType();  
id.setValue("123");  
GetStatisticsCareGiverType parameters = new  
GetStatisticsCareGiverType();  
parameters.setHsaIdentity("SE1000-003");  
  
// Call hsaws  
GetStatisticsCareGiverResponseType response =  
    hsaws.getStatisticsCareGiver(logicalAddress, id, parameters);  
  
System.out.println("HsaIdentity: " + response.getHsaIdentity());  
System.out.println("OrgNo: " + response.getCareGiverOrgNo());  
System.out.println("StartDate: " +  
    response.getStartDate().toString());  
System.out.println("EndDate: " +  
    response.getEndDate().toString());
```

4.15 GetStatisticsHsaUnit

GetStatisticsHsaUnit används för att söka ut information när ett enskilt intyg lagras i Intygstjänsters statistikdatabas. Det som hämtas är information om enhet på vilkens uppdrag intyget är skapat och den vårdenhet som enheten är kopplad till.

4.15.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsaidentity	String	1 - 1	HSA-identitet för enhet från intyget
searchBase	DN	0 - 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används c=SE som sökbas.
includeOrgNo	Boolean	0 - 1	Om True returneras vårdgivarens organisationsnummer.



			Om False eller utelämnat returneras INTE vårdgivarens organisationsnummer.
--	--	--	--

4.15.2 Returvärde

- Om specificerat hsaIdentity hör till en enhet returneras information om enheten, eventuell vårdenhet som enheten är kopplad till samt i förekommande fall även information om den vårdgivare som vårdenheten tillhör i objekt av XML-typen GetStatisticsHsaUnitResponse.
- Om enhet inte kan hittas returneras null.

GetStatisticsHsaUnitResponse

Parameter	Datotyp	Antal	Beskrivning
statisticsUnit	StatisticsHsaUnit	1 - 1	Information om intygets enhet.
statisticsCareUnit	StatisticsHsaUnit	0 - 1	Information om vårdenhet som intygets enhet tillhör. Utelämnas om intygets enhet är en vårdenhet eller om enheten inte tillhör någon vårdenhet.

StatisticsHsaUnit

Parameter	Datotyp	Antal	Beskrivning
hsaIdentity HSA: HSA-Id LDAP: hsaIdentity	String	1 - 1	HSA-identitet för enheten.
isHsaHealthCareUnit HSA: vårdenhet LDAP: hsaHealthCareUnit	Boolean	0 - 1	True om enhet är en vårdenhet (objectClass=hsaHealthCareUnit) False eller uteblivet attribut om enhet inte är en vårdenhet.
careGiverOrgNo HSA: organisationsnummer LDAP: orgNo	String	0 - 1	Organisationsnummer för den vårdgivare som enheten tillhör. Uteblir om enhet inte är en vårdenhet.
careGiverHsaIdentity HSA: vårdgivartillhörighet LDAP: hsaResponsibleHealthCareProvider	String	0 - 1	HSA-Id för den vårdgivare som enheten tillhör. Uteblir om enhet inte är en vårdenhet.



location HSA: Geografisk plats LDAP: l	String	0 – 1	Namn på geografiskt område där enheten i huvudsak är placerad.
county HSA: Länsnamn LDAP: countyName	String	0 – 1	Namn på län (se ref. 10)
countyCode HSA: Länskod LDAP: countyCode	String	0 – 1	Kod för län (se ref. 10)
municipality HSA: Kommunnamn LDAP: municipalityName	String	0 – 1	Namn på kommun (se ref 11)
municipalityCode HSA: Kommunkod LDAP: municipalityCode	String	0 – 1	Kod för kommun (se ref 11).
municipalitySectionCode HSA: Kommunelskod LDAP: municipalitySectionCode	String	0 – 1	Kommundelskod, information om vilken del av kommunen en enhet verkar inom.
municipalitySectionName HSA: Kommunelsnamn LDAP: municipalitySectionName	String	0 – 1	Kommundelsnamn, information om vilken del av kommunen en enhet verkar inom.
geographicalCoordinatesRt90 HSA: Geografiska koordinater LDAP: geographicalCoordinates	GeoCoord	0 – 1	Geografiska koordinater för enhetens huvudsakliga fysiska placering. Koordinaterna anges enligt RT90.
geographicalCoordinatesSWERE F99 HSA: Geografiska koordinater SWEREF 99 TM LDAP: geographicalCoordinatesSweref9 9TM	GeoCoord	0 – 1	Geografiska koordinater för enhetens huvudsakliga fysiska placering. Koordinaterna anges enligt SWEREF 99 TM.
businessTypes		0 – 1	
businessType HSA: Enhetstyp LDAP: hsaBusinessType	String	1 – n	Klassificering av enhet (t.ex. sjukhus) (se ref. 8)



managements		0 - 1	
management HSA: Ägarform LDAP: management	String	1 - n	Ägarform i klartext (se ref. 13)
businessClassificationCodes	Collection	0 - 1	
businessClassificationCode HSA: Verksamhetskod LDAP: businessClassificationCode		1 - n	Typ av verksamhet (kod) (se ref. 7)
careTypes	Collection	0 - 1	
careType HSA: Vårdform LDAP: careType	String	1 - n	Vårdform (se ref. 9)
startDate HSA: Startdatum LDAP: startDate	dateTime	0 - 1	Startdatum för enhetens verksamhet
endDate HSA: Slutdatum LDAP: endDate	dateTime	0 - 1	Slutdatum för enhetens verksamhet.
isArchived HSA: arkiverad LDAP: hsaArchivedObject	Boolean	0 - 1	Sätts till true om enheten är arkiverad. Sätts till false eller uteblivet attribut om enhet inte är arkiverad.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om enheten är ett fingerat objekt.
feignedCareGiverObject HSA: Fingerat objekt LDAP: hsaFeignedCareGiverObject	Boolean	0 - 1	Anger om enhetens vårdgivare är ett fingerat objekt.

GeoCoord

Geografiska koordinater.

Parameter	Datatyp	Antal	Beskrivning
Type	String	1 - 1	Typ av geografisk koordinat.



Parameter	Datotyp	Antal	Beskrivning
			RT90 eller SWEREF 99 TM.
X	String	1 – 1	X-koordinat om RT90
Y	String	1 – 1	Y-koordinat om RT90.
N	String	1 – 1	N-koordinat om SWEREF 99 TM
E	String	1 – 1	E-koordinat om SWEREF 99 TM.

4.15.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsHsaUnit*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetStatisticsHsaUnit>
      <urn:hsaIdentity>SE1000-001</urn:hsaIdentity>
      <urn:searchBase>c=SE</urn:searchBase>
      <urn:includeOrgNo>>false</urn:includeOrgNo>
    </urn:GetStatisticsHsaUnit>
  </soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetStatisticsHsaUnitResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <statisticsUnit>
        <hsaIdentity>SE1000-001</hsaIdentity>
        <location>Platsen</location>
      </statisticsUnit>
    </GetStatisticsHsaUnitResponse>
  </soap:Body>
</soap:Envelope>
```



```
<county>Länsnamnet</county>
<countyCode>23</countyCode>
<municipality>Kommunen</municipality>
<municipalityCode>95</municipalityCode>
<municipalitySectionCode>3</municipalitySectionCode>
<municipalitySectionName>
  Kommundelen
</municipalitySectionName>
<geographicalCoordinatesRt90>
  <Type>RT90</Type>
  <X>1234567</X>
  <Y>7654321</Y>
</geographicalCoordinatesRt90>
<geographicalCoordinatesSWEREF99>
  <Type>SWEREF99TM</Type>
  <N>6477375</N>
  <E>356352</E>
</geographicalCoordinatesSWEREF99>
<businessTypes>
  <businessType>6</businessType>
  <businessType>9</businessType>
</businessTypes>
<managements>
  <management>Privat</management>
  <management>Kommun</management>
</managements>
<businessClassificationCodes>
  <businessClassificationCode>
    1333
  </businessClassificationCode>
  <businessClassificationCode>
    2378
  </businessClassificationCode>
</businessClassificationCodes>
<careTypes>
  <careType>01</careType>
  <careType>02</careType>
</careTypes>
<startDate>2013-02-06T09:54:37.000+01:00</startDate>
<endDate>2014-01-17T09:56:00.000+01:00</endDate>
</statisticsUnit>
</GetStatisticsHsaUnitResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsCareGiver*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
```



```
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
GetStatisticsHsaUnitType parameters = new
GetStatisticsHsaUnitType();
parameters.setHsaIdentity("SE1000-003");
parameters.setIncludeOrgNo(false);

// Call hsaws
GetStatisticsHsaUnitResponseType response =
    hsaws.getStatisticsHsaUnit(logicalAddress, id, parameters);

System.out.println("HsaIdentity: " + response.getHsaIdentity());
System.out.println("Location: " + response.getLocation());
System.out.println("StartDate: " +
response.getStartDate().toString());
System.out.println("EndDate: " +
response.getEndDate().toString());
```

4.16 GetStatisticsPerson

GetStatisticsPerson används för att söka ut information när ett enskilt intyg lagras i Intygstjänsternas statistikdatabas. Det som hämtas är information om den som utfärdade intyget (intygsutfärdaren).



4.16.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsalentity	String	1 – 1	HSA-identitet för den som utfärdade intyget
searchBase	DN	0 – 1	Sökbas för sökning i HSA-katalogen. Om ingen sökbas anges används c=SE som sökbas.

4.16.2 Returvärde

- Om specificerad enhet tillhör en vårdenhet returneras information om vårdenheten i objekt av XML-typen GetStatisticsPersonResponse.
- Om person inte kan hittas returneras null.

GetStatisticsPersonResponse

Parameter	Datotyp	Antal	Beskrivning
hsalentity HSA: HSA-Id LDAP: hsalentity	String	1 - 1	Personens HSA-Id.
age	String	0 - 1	Personens ålder, anges i jämna år.
gender	String	0 - 1	Personens kön. 0, okänt (person saknar personnummer) 1, man 2, kvinna
paTitleNames	Collection	0 - 1	
paTitleName HSA: befattning LDAP: paTitleName	String	1 - n	Personens befattningar (se ref. 20)
paTitleCodes	Collection	0 - 1	
paTitleCode HSA: befattningskod LDAP: paTitleCode	String	1 - n	Personens befattningskoder (se ref. 20)
hsaTitles	Collection	0 - 1	
hsaTitle HSA: Legitimerad yrkesgrupp LDAP: hsaTitle	String	1 - n	Legitimerad yrkesgrupp (se ref. 3)
specialityNames	Collection	0 - 1	



specialityName HSA: specialitet LDAP: specialityName	String	1 - n	Specialistutbildning utöver grundutbildning
specialityCodes	Collection	0 - 1	
specialityCode HSA: specialitetskod LDAP: specialityCode	String	1 - n	Klassificeringskod för specialistutbildning utöver grundutbildning.
isProtectedPerson HSA: skyddad person LDAP: hsaProtectedPerson	Boolean	0 - 1	True om personen är skyddad. False eller uteblivet attribut om personen inte är skyddad.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om svaret innehåller fingerade objekt.

4.16.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsPerson*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing"
  xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soap:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soap:Header>
  <soap:Body>
    <urn:GetStatisticsPerson>
      <urn:hsaIdentity>SE1000-001</urn:hsaIdentity>
      <urn:searchBase> c=SE</urn:searchBase>
    </urn:GetStatisticsPerson>
  </soap:Body>
</soap:Envelope>
```



Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetStatisticsPersonResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <hsaIdentity>SE1000-001</hsaIdentity>
      <age>30</age>
      <gender>1</gender>
      <paTitleNames>
        <paTitleName>Dietist</paTitleName>
        <paTitleName>Audionom</paTitleName>
      </paTitleNames>
      <paTitleCodes>
        <paTitleCode>301018</paTitleCode>
        <paTitleCode>301015</paTitleCode>
      </paTitleCodes>
      <hsaTitles>
        <hsaTitle>Läkare</hsaTitle>
        <hsaTitle>Barnmorska</hsaTitle>
      </hsaTitles>
      <specialityNames>
        <specialityName>Allmänmedicin</specialityName>
        <specialityName>Geriatrisk</specialityName>
      </specialityNames>
      <specialityCodes>
        <specialityCode>0099</specialityCode>
        <specialityCode>18</specialityCode>
      </specialityCodes>
    </GetStatisticsPersonResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
```



```
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");

GetStatisticsPersonType parameters = new
GetStatisticsPersonType();
parameters.setHsaIdentity("SE1000-003");

// Call hsaws
GetStatisticsHsaUnitResponseType response =
    hsaws.getStatisticsHsaUnit(logicalAddress, id, parameters);

System.out.println("HsaIdentity: " + response.getHsaIdentity());
System.out.println("Age: " + response.getAge());
```

4.17 GetStatisticsNames

GetStatisticsNames används för att söka ut information om namn för intygsutfärdare när statistik ska presenteras. OBS! För personer med skyddade personuppgifter returneras inga namnuppgifter.

4.17.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
hsalidentities	Collection	1 – 1	
hsalidentity HSA: HSA-id LDAP: hsalidentity	String	0 – n	HSA-identiteter för intygsutfärdare (personer)
searchBase	DN	0 – 1	Sökbas för sökning i HSA-katalogen.



			Om ingen sökbas anges används c=SE som sökbas.
--	--	--	--

4.17.2 Returvärde

- Information om namn returneras i XML-typen GetStatisticsNamesResponse.

GetStatisticsNamesResponse

Parameter	Datotyp	Antal	Beskrivning
statisticsNameInfos	Collection	1 - 1	
statisticsNameInfo	StatisticsNameInfo	0 - n	Personens namnuppgifter.

StatisticsNameInfo

Parameter	Datotyp	Antal	Beskrivning
hsalidentity HSA: HSA-Id LDAP: hsalidentity	String	1 - 1	Personens HSA-identitet
personMiddleAndSurName HSA: efternamn LDAP: sn, surname	String	0 - 1	Eventuellt mellannamn samt efternamn för person.
personGivenName HSA: tilltalsnamn LDAP: gn, givenName	String	0 - 1	Tilltalsnamn för person.
feignedObject HSA: Fingerat objekt LDAP: hsaFeignedObject	Boolean	0 - 1	Anger om personen är ett fingerat objekt.

4.17.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsNames*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:add="http://www.w3.org/2005/08/addressing">
```



```
xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
<soap:Header>
  <add:MessageID>123</add:MessageID>
  <add:To>SE165565594230-1000</add:To>
</soap:Header>
<soap:Body>
  <urn:GetStatisticsNames>
    <urn:hsaIdentities>
      <urn:hsaIdentity>SE1000-001</urn:hsaIdentity>
    </urn:hsaIdentities>
    <urn:searchBase>c=SE</urn:searchBase>
  </urn:GetStatisticsNames>
</soap:Body>
</soap:Envelope>
```

Resultat

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetStatisticsNamesResponse
      xmlns="urn:riv:hsa:HsaWsResponder:3"
      xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <statisticsNameInfos>
        <statisticsNameInfo>
          <hsaIdentity>SE1000-001</hsaIdentity>
          <personMiddleAndSurName>
            Andersson
          </personMiddleAndSurName>
          <personGivenName>Arne</personGivenName>
        </statisticsNameInfo>
      </statisticsNameInfos>
    </GetStatisticsNamesResponse>
  </soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetStatisticsPerson*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();
```



```
// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create request parameters
AttributedURIType logicalAddress = new AttributedURIType();
logicalAddress.setValue("SE165565594230-1000");
AttributedURIType id = new AttributedURIType();
id.setValue("123");
GetStatisticsNamesType parameters = new
GetStatisticsNamesType();

HsaIdentities hsaIds = new HsaIdentities();
hsaIds.getHsaIdentity().add("SE1000-001");
parameters.setHsaIdentities(hsaIds);

// Call hsaws
GetStatisticsHsaUnitResponseType response =
    hsaws.getStatisticsHsaUnit(logicalAddress, id, parameters);

System.out.println("HsaIdentity: " + response.getHsaIdentity());
System.out.println("First name: " +
response.getPersonGivernName());
System.out.println("Middle and surname: " +
response.getPersonMiddleAndSurName());
```

4.18 GetHospLastUpdate

getHospLastUpdate returnerar tidpunkten då senaste HOSP-inläsning skedde.

För att ha behörighet att anropa denna metod måste det anropande systemet ligga med i en konfigurerad lista som anger vilka som har rätt att använda metoden.



4.18.1 Inparametrar

Inga inparametrar.

4.18.2 Returvärde

- Om allt gick bra returneras resultatet i element av XML-typen *GetHospLastUpdateResponseType*.

GetHospLastUpdateResponseType

Parameter	Datatyp	Antal	Beskrivning
lastUpdate	dateTime	1 – 1	Tidpunkten då HOSP-grenen senast uppdaterades.

4.18.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHospLastUpdate*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:add="http://www.w3.org/2005/08/addressing"
xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soapenv:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soapenv:Header>
  <soapenv:Body>
    <urn:GetHospLastUpdate/>
  </soapenv:Body>
</soapenv:Envelope>
```

Resultat

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetHospLastUpdateResponse
xmlns="urn:riv:hsa:HsaWsResponder:3"
xmlns:ns2="http://www.w3.org/2005/08/addressing">
      <lastUpdate>2013-11-11T18:32:19.000+01:00</lastUpdate>
    </GetHospLastUpdateResponse>
```



```
</soap:Body>  
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *GetHospLastUpdate*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub  
HsaWsResponderService hsawsService =  
    new HsaWsResponderService();  
HsaWsResponderInterface hsaws =  
    hsawsService.getHsaWsResponderPort();  
  
// We must set the hsaws endpoint url explicitly  
BindingProvider bp = (BindingProvider)hsaws;  
bp.getRequestContext().put(  
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,  
    "https://10.253.18.149/svr-hsaws2/hsaws");  
  
// Create our Webservice method parameters.  
AttributedURIType autLA = new AttributedURIType();  
autLA.setValue(getLogicalAddress());  
AttributedURIType autId = new AttributedURIType();  
autId.setValue(logId);  
  
// Call Webservice  
GetHospLastUpdateResponseType result =  
    getServer().getHospLastUpdate(autLA, autId, new  
    GetHospLastUpdateType());  
System.out.println("Last hosp update: " +  
    result.getLastUpdate().toString());
```

4.19 HandleCertifier

handleCertifier används för att lägga till eller ta bort personobjekt från Certifier-grenen med syfte att få in personernas HOSP-information i HOSP-grenen via HOSP-filinläsningen i HSA.

Tillägg, läsning och borttag mot katalogen sker med HSAWS systemanvändare.



4.19.1 Inparametrar

Parameter	Datotyp	Antal	Beskrivning
personallIdentityNumber	String	1 – 1	Personnummer för den person som skall läggas till eller tas bort från Certifier-grenen. Personnummer ska anges på formatet "AAAAMMDDXXXX".
addToCertifiers	Boolean	1 – 1	True om personnumret skall registreras i Certifiers-grenen False om personnumret ska avregistreras från Certifier-grenen
certifierId	String	0 – 1	Alternativt id kopplat till personnumret.
reason	String	0 – n	Lämnas tom vid registrering. Vid avregistrering anges orsaken till avregistreringen, t.ex. "Aviden" eller "Begärd avregistrering".

Skulle begäran om registrering ske för ett objekt som redan är registrerat av det anropande systemet kommer inget särskilt meddelande om det returneras utan registreringen kommer betraktas som framgångsrik. Det bifogade certifierId kommer läggas till objektet oavsett (om det inte redan finns på objektet).

Vid avregistrering hämtas först objektet med matchande personnummer från Certifier-grenen. Hittas ej ett matchande objekt betraktas avregistreringen som framgångsrik. På det matchande objektet tas "ursprung" för anropande system bort. Om det därefter inte finns några fler ursprung tas hela objektet bort, annars så får det ligga kvar. (Teoretiskt kan samma personnummer ha registrerats av flera olika system, dvs ha flera ursprung). Avregistreringen betraktas oavsett som framgångsrik.

Om det ej finns något ursprung på objektet som matchar det anropande systemet lämnas objektet orört och avregistreringen betraktas som framgångsrik. Ett exempel på när detta teoretiskt skulle kunna ske är om anropande system av någon anledning försöker avregistrera ett personnummer som de ej tidigare registrerat. Oavsett kan ju personnumret betraktas som avregistrerat efter avregistreringen.

De certifier-id som ligger på objektet tas inte bort utan de försvinner först när hela objektet försvinner. Observera att certifier-id inte fyller något syfte i HSA utan enbart sparas på objektet för att underlätta felsökning etc, särskilt med avseende på att man inte behöver hantera personnummer i mailkommunikation osv i onödan.

Vid anrop till handleCertificate ska HSA Webservice logga anropet inklusive alla inparametrarnas värde.

Inparameter "reason" sparas ingenstans i HSA utan kommer bara loggas i HSA WebServices ordinarie logg.



4.19.2 Returvärde

- Om registreringen/avregistreringen lyckades returneras resultatet i element av XML-typen *HandleCertifierResponseType*.
- Vid eventuellt fel så returneras ett Webservice fault.

HandleCertifierResponseType

Parameter	Datatyp	Antal	Beskrivning
Result	String	1 – 1	OK om registreringen alternativt avregistreringen lyckades.

4.19.3 Exempel

SOAP

I detta avsnitt ger vi exempel på ett typiskt anrop av *HandleCertifier*. Exemplet ges i form av SOAP-meddelanden.

Anrop

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:add="http://www.w3.org/2005/08/addressing"
xmlns:urn="urn:riv:hsa:HsaWsResponder:3">
  <soapenv:Header>
    <add:MessageID>123</add:MessageID>
    <add:To>SE165565594230-1000</add:To>
  </soapenv:Header>
  <soapenv:Body>
    <urn:HandleCertifier>

    <urn:personalIdentityNumber>191212121212</urn:personalIdentityNumber>
      <urn:addToCertifiers>>false</urn:addToCertifiers>
      <!--Optional:-->
      <urn:certifierId>WebCert-123</urn:certifierId>
      <!--Optional:-->
      <urn:reason>Opt out</urn:reason>
    </urn:HandleCertifier>
  </soapenv:Body>
</soapenv:Envelope>
```

Resultat

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```



```
<soap:Body>
  <HandleCertifierResponse
xmlns="urn:riv:hsa:HsaWsResponder:3"
xmlns:ns2="http://www.w3.org/2005/08/addressing">
  <result>OK</result>
  </HandleCertifierResponse>
</soap:Body>
</soap:Envelope>
```

JAX-WS

I detta avsnitt ger vi exempel på ett typiskt anrop av *HandleCertifier*. Exemplet ges i form av javakod kompatibel med JAX-WS.

```
// Create hsaws client stub
HsaWsResponderService hsawsService =
    new HsaWsResponderService();
HsaWsResponderInterface hsaws =
    hsawsService.getHsaWsResponderPort();

// We must set the hsaws endpoint url explicitly
BindingProvider bp = (BindingProvider)hsaws;
bp.getRequestContext().put(
    BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
    "https://10.253.18.149/svr-hsaws2/hsaws");

// Create our Webservice method parameters.
AttributedURIType autLA = new AttributedURIType();
autLA.setValue(getLogicalAddress());
AttributedURIType autId = new AttributedURIType();
autId.setValue(logId);

HandleCertifierType parameters = new HandleCertifierType();
    parameters.setPersonalIdentityNumber("191212121212");
    parameters.setAddToCertifiers(false);
    parameters.setCertifierId("WebCert-123");
    parameters.setReason("Opt out");

// Call Webservice
HandleCertifierResponseType result =
getServer().handleCertifier(autLA, autId, parameters);

System.out.println("HandleCertifier result: " + result.getResult());
```