



SPECIFIKATION AV BAS- OCH TILLÄGGSPRODUKTER

Innehåll

1	BASPRODUKTER	2
1.1	Produktområden	2
1.2	Specifikationer	2
1.2.1	Teknik och arkitektur	2
1.2.2	Curalia SHS-asynk	4
1.2.3	Curalia SHS-synk	4
1.2.4	Curalia SHS-DN	4
1.2.5	Curalia SHS-IN	4
1.2.6	Curalia SHS-IP	5
1.3	Leveranstider	5
2	TILLÄGGSPRODUKTER	6
2.1	Produktområden	6
2.2	Gränssnitt för behörighetshantering och drift	6
2.3	Gränssnitt och produkter för mottagning och utskick	6
2.4	Gränssnitt för betalningshantering	6
2.5	Gränssnitt för webbinlämning med svar	6
2.6	Externa gränssnitt	7
2.7	Leveranstider	7



1 BASPRODUKTER

1.1 PRODUKTOMRÅDEN

Produktområden	Beteckning	Ingår i ramavtalet	Antal bas-produkter
Delfunktion	DF	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	
DF A Asynkron överföring	B2A	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	Curalia SHS-Asynk
DF B Synkron överföring	B2B	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	Curalia SHS-Synk
DF C Dotternod	B2C	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	Curalia SHS-DN
DF D Inlämningsnod	B2C	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	Curalia SHS-IN
DF E Inlämningsprogram	B2E	Ja <input checked="" type="checkbox"/> Nej <input type="checkbox"/>	Curalia SHS-IP

Curalia:s produktsvit för SHS bygger på en gemensam kärna för att uppfylla själva kommunikationen med andra SHS noder. Kärnan är uppbyggd av öppen programvara från i första hand Apacheprojekt. Alla Apache projekt är förvaltda öppna programvara vilket ger trygghet i form av support och vidareutveckling. Med tanke på SHS vidareutveckling mot internationell standard finns stora fördelar med utveckla den med öppen programvara eftersom alla kan samlas kring ett gemensamt projekt.

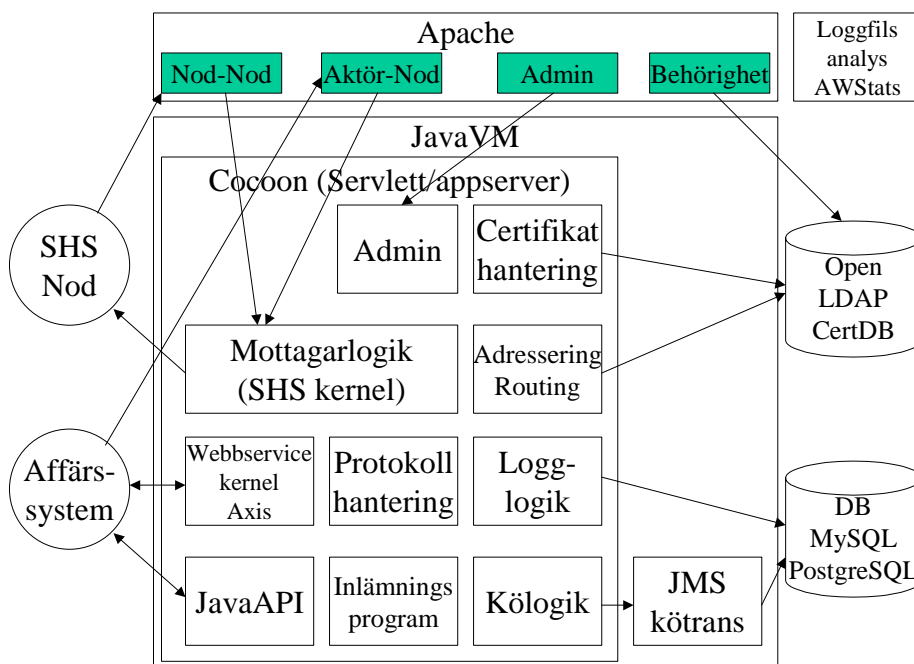
1.2 SPECIFIKATIONER

1.2.1 TEKNIK OCH ARKITEKTUR

Curalia SHS är modulärt uppbyggt med en flexibel arkitektur. Produkten är genomgående baserad på Java standardklasser och öppna API:er så långt det är möjligt. Systemdesignen följer Apache Avalon komponentramverk, som utökar den objektorienterade designen (OOP) i Java till tjänsteorienterad design (COP - Component Oriented Programming). Apache Avalon ramverk tillser att interna programkomponenter är modulära och återanvändningsbara. Apache Avalon s.k. livscykelkontrakt för programkomponenterna innebär också att designmönstret IoC (Inversion of Control) uppfylls, vilket gör det lätt att centralt konfigurera, övervaka och profilera enskilda systemkomponenter.

Curalia SHS kan köras på samtliga plattformar som tillhandahåller en virtuell Java-maskin, JVM. (Windows, Unix och Linux m.fl.) Baskravet är en Java servlet-container för drift av Apache Cocoon och Apache Axis servlets, vilket kan uppfyllas antingen i J2SE-miljö med (t.ex. Apache Tomcat eller Resin som applikationsserver) eller i en befintlig J2EE-miljö med centralt konfigurerade datakällor (t.ex. JBoss, BEA WebLogic eller IBM WebSphere).

Distribuering och lastbalansering kan ske med ett antal olika tekniker beroende på kundens driftmiljö. Ett exempel på en standardkonfiguration är en central Apache webbproxy som lastbalanserar (och utför fail-over vid driftstörningar i enskilda noder) tjänsteanropen mellan ett antal Java applikationsservrar med Curalia SHS noder



Alla händelser i systemet (överföringar, utväxlingar, hantering) loggas i en logg med tider mm. Loggen och övrig information lagras i databas som kan ersättas med vilken databas som helst som har JDBC-drivrutiner. Rapporter, hantering av loggar och övrig administration sker med ett webbgränssnitt.

För systemadministrationens webbgränssnitt kan autentisering utföras med godtycklig plugin till Apache HTTP-server (dvs RADIUS för VPN, Microsoft NTLM för internt nätverk eller LDAP för central rättighetshantering i katalog). Behörighetshandlingen i själva SHS-nodens kärna baseras på LDAP och kan kopplas till den interna OpenLDAP-katalogen eller en befintlig rättighetstjänst som stödjer LDAPv3.



SSL-standarden används för säker åtkomst och kryptering. Apache HTTP-server har en flexibel konfiguration för vilka SSL-versioner och/eller kryptoalternativ som måste vara uppfyllda för att få åtkomst till systemet.

All åtkomstkontroll som misslyckas loggas med tidstämpel, programfunktion, användare och händelse/orsak i SHS-nodens systemlogg. Normal inloggning och åtkomstkontroll loggas i Apache serverlogg. Vi använder tredjepartsprodukten AWstats (baserad helt på öppen källkod) för avancerad realtidsstatistik och analys av loggfilerna baserat på valbara filtrerings- och sorteringsvillkor.

Den interna katalogen är baserad på OpenLDAP och LDAPv3-standard. I den interna katalogen lagras SHS-avtal, SHS-produktyper samt certifikathantering (godkända server-certifikat, återkallningslistor m.m.). Rättighetshanteringen administreras genom OpenLDAPs behörighetssystem.

Den lokala LDAP v3-katalogen i Curalias DF-nod kan synkroniseras med den nationella LDAP-katalog (både läsning och skrivning) via webbgränssnittet för systemadministration. LDAP-replikering i intern och publik katalog uppfylls av OpenLDAP. shs.agreement kan registreras via interna webbjärnen eller Java API.

1.2.2 CURALIA SHS-ASYNK

Asynkron SHS nod uppfyller kraven enligt SHS 1.2 för asynkron kommunikation med Java API och gränssnitt för Internal Web Services Interface.

1.2.3 CURALIA SHS-SYNK

Synkron SHS nod uppfyller kraven enligt SHS 1.2 för synkron kommunikation med Java API, gränssnitt för Internal Web Services Interface och anropen shssend/shsfetch.

1.2.4 CURALIA SHS-DN

Dotternod uppfyller kraven enligt SHS 1.2 för både asynkron och synkron kommunikation med Java API, gränssnitt för Internal Web Services Interface och anropen shssend/shsfetch. Dock saknas gränssnitt för administration och logganalys.

1.2.5 CURALIA SHS-IN

Inlämningsnod uppfyller kraven enligt SHS 1.2 för både asynkron och synkron kommunikation med Java API, gränssnitt för Internal Web Services Interface och anropen shssend/shsfetch. Dock saknas gränssnitt för administration och logganalys.



SHS 2004
Curalia AB

Dnr 2004/243-3
Ramavtal 6769/05
Bilaga 3: Specifikation av bas- och
tilläggsprodukter
2005-06-03 Version 1.0

1.2.6 CURALIA SHS-IP

Inlämningsprogram uppfyller kraven enligt SHS 1.2 för synkron kommunikation med Java API och gränssnitt för Internal Web Service Interface. Det saknas gränssnitt för administration och logghantering.

1.3 LEVERANSTIDER

Samtliga produkter i Curalia:s SHS produktsvit kan börja levereras under hösten 2005 efter godkända funktions- och interoperabilitetstester. I de flesta fall skall SHS-noderna integreras med befintliga system eller ingå i olika verksamhetslösningar vilket kräver planering från fall till fall. Igångsättande av införandeprojekt kan ske inom 1 månad från beställning.



2 TILLÄGGSPRODUKTER

Syftet med tilläggsprodukter är att komplettera SHS-installationen med funktioner som stöder myndighetens behov i samband med mottagning och utskick av handlingar.

2.1 PRODUKTOMRÅDEN

- Gränssnitt för behörighetshantering och drift
- Gränssnitt för integration med verksamhetssystem
- Gränssnitt och produkter för mottagning och utskick
- Gränssnitt för betalningsförmedling
- Gränssnitt för webbinlämning med svar
- Externa gränssnitt

2.2 GRÄNSSNITT FÖR BEHÖRIGHETSHANTERING OCH DRIFT

All behörighetshantering och drift sker via ett webbaserat gränssnitt för administration. Därutöver kan vi erbjuda möjlighet till webbservices gränssnitt. För övervakning används Nagios som har ett eget webbaserat gränssnitt förutom att det kan samverka med andra övervakningssystem via SNMP.

2.3 GRÄNSSNITT OCH PRODUKTER FÖR MOTTAGNING OCH UTSKICK

Vi tillhandahåller generella Java API och webbservices. Som komplement till dessa finns funktioner för konvertering och formattering med hjälp av Apache Xalan. Genom att produkten också innehåller Apache Cocoon finns färdiga komponenter för webbapplikationer och direkt koppling till databaser.

2.4 GRÄNSSNITT FÖR BETALNINGSHANTERING

Erbjuds inte för närvarande.

2.5 GRÄNSSNITT FÖR WEBBINLÄMNING MED SVAR

Med hjälp av de standardkomponenter vi har för Cocoon kan det enkelt skapas anpassade webbtillämpningar för formulär som är ren HTML och inte ställer några krav på JavaScript eller liknande på klientsidan.



2.6 EXTERNA GRÄNSSNITT

Alla verksamhetssystem som stödjer nedanstående standardtekniker kan integreras med befintliga komponenter:

FTP, HTTP, HTTPS, SMTP, JMS, SOAP/WebServices

Java standardklasser (FTP, HTTP, HTTPS)

JavaMail API (SMTP)

ActiveMQ (JMS)

Apache Axis (SOAP)

2.7 LEVERANSTIDER

När det gäller samtliga tillägsfunktioner gäller det integration med befintliga system eller produkter vilket kräver från fall till fall. Igångsättande av införandeprojekt kan ske inom 1 månad från beställning.