

Direkt meddelandedeklarering
med Tullen
Teknisk guide

Uppdatering 13.12.2024

Innehåll

1	Syftet med denna guide.....	5
2	Web service-tjänsten för direkt meddelandedeklarering.....	5
2.1	Logiska uppbyggnaden av direkt meddelandedeklarering.....	6
2.2	Tjänstens datasäkerhet.....	7
2.3	Meddelandenotifikationstjänsten.....	8
2.4	Meddelandetjänsten för bifogade filer.....	9
3	Tullens SOAP-meddelanden.....	9
3.1	SOAP-meddelandenas struktur.....	9
3.1.1	SOAP-begärandemeddelande.....	10
3.1.2	SOAP-svarsmeddelande.....	11
3.2	Svar i normal- och felsituationer.....	11
3.2.1	ResponseHeader-returkoder.....	11
3.2.2	SOAP fault, dvs. felelement.....	11
3.2.3	Kundprogrammets beredskap för felsituationer.....	12
3.2.4	Sändning av meddelande på nytt.....	12
3.3	Faserna för byggande och förmedling av ett meddelande (UPLOAD).....	14
3.4	Faserna för att skapa och förmedla ett meddelande (UPLOAD-ATTACHMENT).....	15
4	Tullens WSDL- och XSD-filer.....	15
5	Meddelande om schemafel.....	17
5.1.1	DmeErrorMessage, struktur.....	17
5.1.2	DmeErrorMessage, beskrivning av elementen.....	18
6	Bifogade filer till svarsmeddelanden.....	19
6.2	LiituResponseMessage (konvertering av bifogade filer).....	19
6.3	Meddelande innehållande PDF-dokumentets metadata (DmeDocumentInfoMessage).....	22
6.3.1	DmeDocumentInfoMessage, beskrivning av elementen.....	22
6.3.2	Hämtning av PDF-dokument från gränssnittet för direkt meddelandedeklarering.....	23
7	Operationer som tjänsten omfattar.....	23
7.1	Tjänster som implementeras av Tullen.....	23
7.2	Tjänst som implementeras av kunden.....	23
7.3	Tjänsteoperationer i WSDL.....	24
7.3.1	Upload.....	24
7.3.2	UploadAttachment.....	27
7.3.3	DownloadList.....	30
7.3.4	Download.....	34
7.3.5	Notify.....	37
7.3.6	CheckConnectivity.....	40
8	Beskrivning av uppgifterna i XML-elementen.....	41
8.1	RequestHeader.....	41
8.2	ApplicationRequestMessage.....	41

8.3	ResponseHeader	41
8.4	MessageInformation.....	42
8.5	AttachmentRequestMessage	43
8.6	AttachmentResponseMessage	43
8.7	DownloadMessageListFilteringCriteria.....	44
8.8	DownloadMessageFilteringCriteria	45
8.9	ApplicationResponseMessage	45
8.10	EchoContent	45
8.11	ApplicationRequest.....	46
8.12	AttachmentRequest.....	47
8.13	ApplicationContent	48
8.14	AttachmentContent	48
8.15	ApplicationResponse	48
8.16	NotifyRequest.....	49
8.17	Notifyresponse	50
8.1	DocumentInformation	51
9	Autentisering och auktorisation av tekniska aktörer.....	51
9.1	Autentisering av meddelandedeklarant	52
9.2	Autentisering av byggare/förmedlare	52
9.3	Autentisering av meddelandenotifikationstjänstens leverantör	52
10	Servercertifikat	52
10.1	Anskaffning och ibrukttagande av servercertifikat	54
10.2	Förnyande av servercertifikat	54
11	Sändningsuppgifter för kundmeddelanden	54
11.1	Tjänsteleverantörs ID	54
11.2	Tjänstens URL-adress	54
11.3	Kontrollreferens för sändningstillfället	54
11.4	Sändningsuppgifter i kundmeddelanden.....	55
11.4.1	RequestHeader	55
11.4.2	ApplicationRequest.....	55
11.4.3	Applikationsmeddelande	56
12	XML-signatur	56
12.1	XML-signaturens struktur.....	56
12.2	Implementering av XML-signatur	56
12.3	Validering av XML-signatur samt eventuella problem	57
13	Begränsningar vid direkt meddelandedeklarering.....	57
13.1	Protokollversioner	57
13.2	Krypteringsalgoritmer.....	57
13.3	Begränsningar gällande meddelanden.....	57
13.4	Begränsningar gällande antalet begäranden	58

13.5 Tidsövervakning av begäranden.....	59
Bilaga 1: ResponseCode och ResponseText	60
Bilaga 2: Felkoder som används i NotifyResponse-meddelandet.....	64
Bilaga 3: Krav gällande Tullens XML-meddelanden	65
Bilaga 4: XML-scheman och användningen av namnrymder	67
Bilaga 5: Tekniska standarder för direkt meddelandedeklarering.....	70

Tabeller:

Tabell 1: DmeErrorMessage – rotelement.....	18
Tabell 2: DmeErrorMessage – MessageIdentifier-element	18
Tabell 3: DmeErrorMessage – ErrorInformation-element	19
Tabell 4: LiituResponseMessage – rotelement	20
Tabell 5: LiituResponseMessage – MessageHeader-element.....	21
Tabell 6: LiituResponseMessage - Acceptance-element	21
Tabell 7: LiituResponseMessage - Rejection-element	22
Tabell 8: LiituResponseMessage - RejectionReason-element.....	22
Tabell 9: DmeDocumentInfoMessage, beskrivning av elementen	23
Tabell 10: Upload.....	24
Tabell 11: UploadAttachment	27
Tabell 12: DownloadList	31
Tabell 13: Download	34
Tabell 14: Notify	37
Tabell 15: CheckConnectivity	40
Tabell 16: RequestHeader	41
Tabell 17: ApplicationRequestMessage	41
Tabell 18: ResponseHeader	42
Tabell 19: MessageInformation	43
Tabell 20: AttachmentRequestMessage.....	43
Tabell 21: AttachmentResponseMessage	44
Tabell 22: DownloadMessageListFilteringCriteria	45
Tabell 23: DownloadMessageFilteringCriteria.....	45
Tabell 24: ApplicationResponseMessage	45
Tabell 25: EchoContent	45
Tabell 26: ApplicationRequest	47
Tabell 27: AttachmentRequest	48
Tabell 28: ApplicationContent.....	48
Tabell 29: AttachmentContent	48
Tabell 30: ApplicationResponse	49
Tabell 31: NotifyRequest	50
Tabell 32: NotifyRespondsHeader.....	50

Tabell 33: RespondsCode	50
Tabell 34: DocumentInformation	51
Tabell 35: Tekniska aktörer	51
Tabell 36: ResponseCode och ResponseText.....	63
Tabell 37: Felkoder som används i NotifyResponse-meddelandet.....	64
Tabell 38: Tekniska standarder för direkt meddelandedeklarering	70

Bilder:

Bild 1: Exempel på web service-tjänster	6
Bild 2: Exempel på Tullens och kundens lösning	7
Bild 3: Inkommande meddelande från kunden.....	10
Bild 4: SOAP-begärandemeddelandets struktur på hög nivå	11
Bild 5: SOAP-svarsmeddelandets struktur på hög nivå	11
Bild 6: Sändning av meddelande på nytt och kontrollreferens för sändningstillfället.....	13
Bild 7: Hur ett meddelande byggs för Tullen	14
Bild 8: Skapa ett UploadAttachment-meddelande till Tullen	15
Bild 9: DmeErrorMessage:s struktur	17
Bild 10: DmeErrorMessage – MessageIdentifier:s struktur	17
Bild 11: DmeErrorMessage – ErrorIdentifier:s struktur	18
Bild 12: Beskrivning av LiituResponseMessage	20
Bild 13: Upload, begäran	25
Bild 14: Beskrivning av ApplicationRequest	25
Bild 15: Upload, svar	26
Bild 16: UploadAttachment, begäran	28
Bild 17: Beskrivning av AttachmentRequest.....	29
Bild 18: UploadAttachment, svar	30
Bild 19: DownloadList, begäran med elementen StartDate och EndDate	31
Bild 20: DownloadList, begäran med elementen StartTimestamp ja EndTimestamp.....	32
Bild 21: DownloadList, svar	33
Bild 22: Download, begäran	34
Bild 23: Download, svar	35
Bild 24: Beskrivning av ApplicationResponse.....	36
Bild 25: Beskrivning av NotifyRequest som skickas av Tullen.....	38
Bild 26: Beskrivning av NotifyResponse som skickas av kunden.....	39
Bild 27: Förtroendekedjan för certifikat.....	53
Bild 28: Upprättande av en HTTPS-förbindelse med ett certifikat (förenklat).....	53

1 Syftet med denna guide

Denna anvisning innehåller en redogörelse över de tekniska detaljer i Tullens web service-tjänst för direkt meddelandedeklarering som behövs för att koppla kundens system till Tullens tjänst, instruktioner om hur man skaffar ett servercertifikat samt en beskrivning av web service-tjänsten för direkt meddelandedeklarering. Bilagorna innehåller också mycket användbar information.

De senaste ändringarna/tilläggen är markerade med gult.

Följande nyttiga publikationer finns på Tullens webbplats:

1. Inledning till Tullens meddelandedeklarering
2. Användarvillkoren för direkt meddelandedeklarering
3. Checklista för direkt meddelandedeklarering – (på finska och engelska)

Inledning till Tullens meddelandedeklarering är nyttig då man funderar på vilket deklareringsätt man vill använda. Den innehåller också en mer detaljerad beskrivning över aktörerna och rollerna i direkt meddelandedeklarering. Direkt meddelandedeklarering, liksom övrig meddelandebaserad deklarerung, kräver tillstånd. Guiden innehåller också mer information om detta.

https://tulli.fi/documents/162752825/211745463/Johdanto+Tullin+sanoma-asiointiin+22.7.2024_SV.pdf/0876600a-4ca6-16b3-aea7-87fb54215dc8/Johdanto+Tullin+sanoma-asiointiin+22.7.2024_SV.pdf?t=1721650799498

Användarvillkoren för direkt meddelandedeklarering finns på Tullens webbplats. De som kommer att börja använda direkt meddelandedeklarering ska bekanta sig med användarvillkoren.

<https://tulli.fi/sv/om-oss/material/anvandarvillkor/anvandarvillkoren-for-direkt-meddelandedeklarering>

Checklistan för direkt meddelandedeklarering (FAQ) innehåller beskrivningar av problem som kunderna stött på i samband med ibruktagandet och användningen av direkt meddelandedeklarering samt lösningarna på dessa problem. Listan uppdateras regelbundet så det lönar sig att ta en titt på den nu och då.

http://tulli.fi/documents/2912305/3589990/sanoma_asiointiin_ongelmatilanteet.pdf/1fb48f7b-b099-41f8-b225-ba7dec459eb2

På Tullens webbplats finns ett omfattande urval av **exempelmeddelanden**. De finns vid versionsmeddelandena för direkt meddelandedeklarering.

<https://tulli.fi/sv/foretag/foretagsarenden/meddelandedeklarering/for-programutvecklare>

2 Web service-tjänsten för direkt meddelandedeklarering

Vid meddelandebaserad deklarerung kan företag både skicka meddelanden till Tullens system och hämta meddelanden från dem direkt över Internet. Det kallas för direkt meddelandedeklarering, är en web service-tjänst som Tullen erbjuder sina företagskunder.

Direkt meddelandedeklarering grundar sig på att kundens system skickar sina meddelanden till Tullen och hämtar de meddelanden som väntar på att hämtas från Tullens system. Tullen skickar aldrig sina svarsmeddelanden till kunden. Kunden får identifieringsuppgifterna för de meddelanden som väntar på att hämtas genom att göra en begäran om meddelandelista. Med hjälp av denna lista kan kunden sedan hämta meddelandena. Alternativt kan kunden ta i bruk meddelandenotifikationstjänsten som meddelar kundens system om de meddelanden som väntar på att hämtas.

Tullen erbjuder gränssnitt för direkt meddelandedeklarering mot följande system:
Tullklareringstjänsten (UTU), AREX och Intrastat.

Kunden implementerar i sin egen miljö de tjänster som behövs för att kunden ska kunna använda de tjänster som Tullen erbjuder.

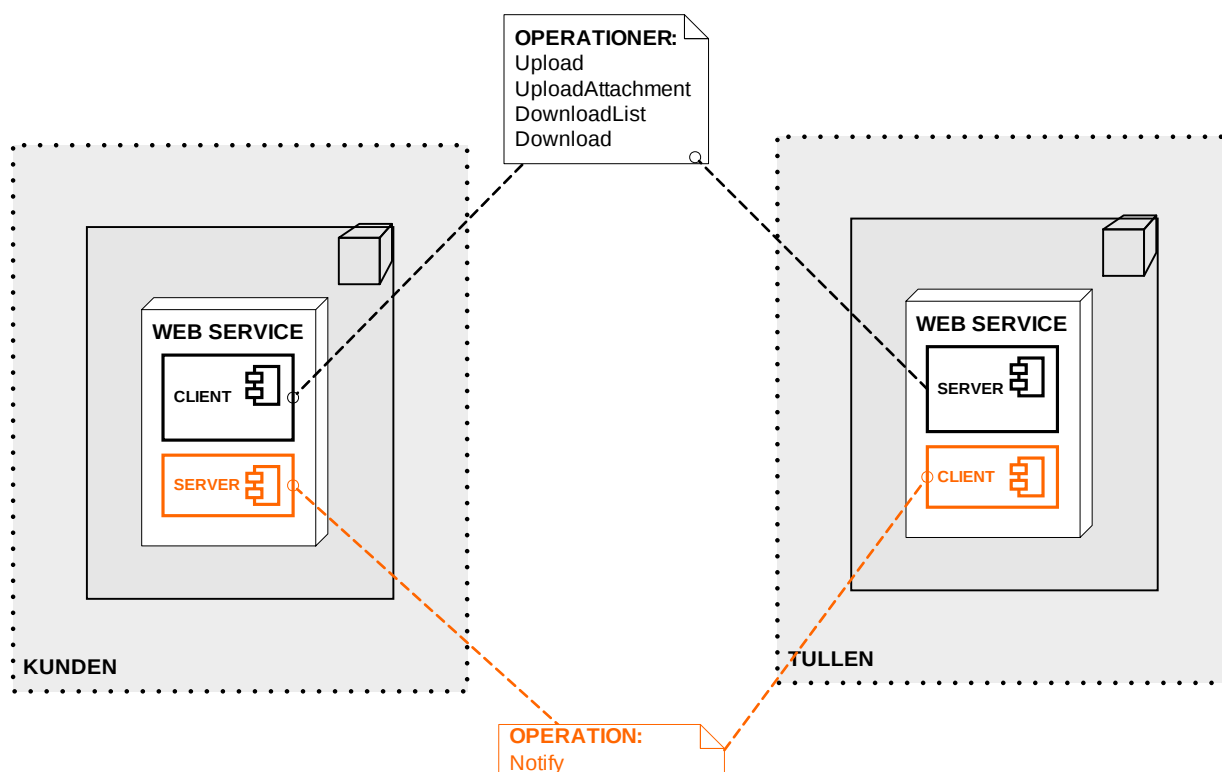


Bild 1: Exempel på web service-tjänster

2.1 Logiska uppbyggnaden av direkt meddelandedeklarering

Kunden kan använda ett eller flera av Tullens datasystem via gränssnittet för direkt meddelandedeklarering. Det enklaste och klaraste sättet för kunden att genomföra implementeringen är att bygga ett integrationssystem som bildar och skickar alla meddelanden som skickas till Tullens system. Då behövs endast ett servercertifikat och det är lättare för kunden att importera nya kundprogram till servicekanalen för direkt meddelandedeklarering.

Det är också lättare att implementera meddelandenotifikationstjänsten när kundens lösning endast omfattar en servicekanal genom vilken kunden mottar alla meddelandenotifikationer.

Om utmaningar som andra lösningar se närmare:

<https://tulli.fi/sv/foretag/foretagsarenden/meddelandedeklarering/for-programutvecklare>

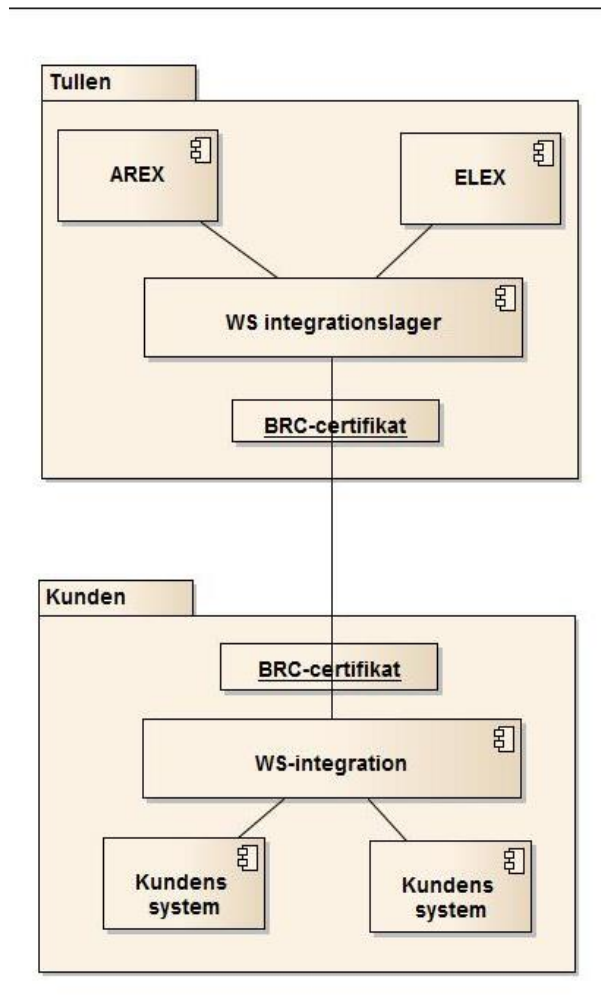


Bild 2: Exempel på Tullens och kundens lösning

2.2 Tjänstens datasäkerhet

Konfidentialiteten av meddelandeförmedlingen vid direkt meddelandedeklarering grundar sig på att en tillräckligt stark SSL/TSL-kryptering används vid samtliga meddelandeförmedlingstransaktioner som styrs via offentliga nätverk.

Vid direkt meddelandedeklarering sker autentiseringen av kunden genom dubbelriktad SSL/TLS-autentisering. Genom att i samband med SSL/TLS-handskakningen be kunden om det av Myndigheten för Digitalisering och Befolkningsdata signerade certifikatet för den publika nyckeln autentiserar Tullens gränssnitt för direkt meddelandedeklarering kundens identitet och kontrollerar vilka användarrättigheter till Tullens system som beviljats till kunden. Tullens gränssnitt utnyttjar också MDB:s spärrlista över servercertifikat och förkastar alla meddelandeförmedlingsförsök som görs med certifikat som finns på MDB:s spärrlista. Kunden kan också kontrollera identiteten på Tullens gränssnittssystem med hjälp av certifikatet för den publika nyckeln som uppvisas i samband med SSL/TLS-handskakningen.

Certifikatbaserad ömsesidig autentisering som används i gränssnittet för direkt meddelandedeklarering begränsar också antalet Denial of Service-attacker som riktas av externa aktörer mot gränssnittet, eftersom Tullens gränssnitt avvisar sådana SSL/TSL-förbindelser som upprättas där meddelandets avsändare inte använder ett av MDB beviljat giltigt certifikat.

Integriteten av de förtullningsmeddelanden som kunden skickar säkerställs genom en XML-signatur (XMLDSig). Med hjälp av XML-signaturen kontrollerar Tullens gränssnittssystem varje meddelandes integritet och signerarens identitet.

Tullens gränssnitt för direkt meddelandedeklarering sparar både de meddelanden som kunden skickat och de meddelanden som returneras till kunden under minst den lagstadgade tiden. Under denna tid kan meddelandenas innehåll och sändningsuppgifter vid behov kontrolleras.

Tullens gränssnitt för direkt meddelandedeklarering kontrollerar att de meddelanden som kunden skickar inte medför något XML-hot eller SQL-injektionsangrepp.

Tullens gränssnitt hindrar överbelastningsattacker, som kan orsakas av avsiktliga eller oavsiktliga överbelastningssituationer, från att lyckas genom att begränsa trafikvolymen från kunderna.

I samband med tillståndsansökan godkänner kunden Tullens användarvillkor för direkt meddelandedeklarering, i vilka kundens skyldigheter och ansvar angående datasäkerheten har fastställts ([Användarvillkor för direkt meddelandedeklarering](#)).

2.3 Meddelandenotifikationstjänsten

När kunden tar i bruk **meddelandenotifikationstjänsten** skickar Tullen omedelbart identifieringsuppgifterna för ett meddelande som väntar på att hämtas till den av kunden angivna URL-adressen. Meddelandedeklareringen sker då nästan i realtid och man undviker också onödig trafik mellan kundens och Tullens system. Tullen rekommenderar att meddelandenotifikationstjänsten tas i bruk.

Meddelandenotifikationstjänsten är FO-nummerspecifik.

För att kunna ta emot meddelandenotifikationen behöver kunden följande:

- Web service server-funktionalitet
- En egen intern processlogik för meddelandenotifikationer
- Servermiljö (t.ex. programserver + https-server)
- Brandväggsöppningar i kundens nät
- Certifikat som även omfattar ett servercertifikat (MDB)
- Företag kan (per ett FO-nummer) endast ha en mottagningstjänst för meddelandenotifikationer för alla system och program vid direkt meddelandedeklarering. Kundens tjänst ska vid behov kunna dirigera de inkomna meddelandenotifikationerna till rätt system
- När meddelandenotifikationstjänsten tas i bruk omfattar den kundens samtliga system. Kunden ska själv filtrera fram de notifikationer som denne önskar att ska handläggas, om man inte vill utnyttja notifikationerna i alla system.

Till dessa delar ligger ansvaret för implementeringen och upprätthållandet hos kunden.

När meddelandenotifikationstjänstens dataöverföringsförbindelser upprättas i ett offentligt nätverk används alltid SSL/TSL-kryptering. Ur datasäkerhetssynvinkel vore det önskvärt att kunden skulle konfigurera dataförbindelserna så att de endast tillåter trafik mellan kundens meddelandenotifikationsserver och Tullens adress varifrån meddelandenotifikationerna skickas. Därtill rekommenderas det att kunden med hjälp av certifikatet för den publika nyckeln (HTTPS client authentication) kontrollerar identiteten på programmet som kontaktar kundens meddelandenotifikationstjänst i det skede då förbindelsen upprättas. Kundprogrammet skulle alltså autentisera Tullens tjänst med hjälp av servercertifikatet som den skickat och kundens tjänst skulle på så sätt kunna försäkra sig om meddelandenotifikationens avsändare.

Även när kunden använder meddelandenotifikationstjänsten är det bra att nu och då skicka en begäran om meddelandelista, eftersom Tullen vid störningar inte försöker skicka en meddelandenotifikation på nytt. Kunden ska då själv ta reda på identifierarna för de meddelanden som väntar på att hämtas.

En begäran om meddelandelista får dock i dessa fall skickas mer sällan än normalt, högst en gång i timmen.

2.4 Meddelandetjänsten för bifogade filer

Med hjälp av meddelandetjänsten för bifogade filer kan kunden skicka bifogade filer som hänför sig till tulldeklarationer till Tullen. Sändningen av bifogade filer sker i form av en UploadAttachment-operation i gränssnittet för direkt meddelandedeklarering.

När kunden skickat sin tulldeklaration och behandlat meddelandet om godkännande som kunden hämtat från Tullen kan kunden skicka bifogade filer som gäller tulldeklarationen. Alternativt kan Tullen i samband med sakinnehållskontrollen be kunden komplettera deklarationen med bestämda bilagedokument.

För att skicka bilagor kontaktas UploadAttachment-operationen i Tullens gränssnitt för direkt meddelandedeklarering. Per varje UploadAttachment-meddelande kan en bifogad fil skickas till Tullen. Vill man skicka flera bifogade filer ska var och en av dem skickas i ett eget UploadAttachment-meddelande.

Tullens integrationslager konverterar den bifogade filen till ett format som förtullningsprogrammen kan hantera och skickar kunden antingen ett meddelande om avslag eller ett meddelande om godkännande (LiituResponseMessage). Den egentliga kontrollen av sakinnehållet i de bifogade filerna sker i det aktuella förtullningsprogrammet i samband med att tulldeklarationens innehåll kontrolleras.

Meddelandesändningssekvensen presenteras i kapitel 2.1.4 Funktionerna och processen när meddelandetjänsten för bifogade filer används i dokumentet "Introduktion för meddelandetrafik med Tullen".

3 Tullens SOAP-meddelanden

3.1 SOAP-meddelandenas struktur

I det direkta meddelandegränssnittet (web service) används SOAP-begäranden och -svar för överföring av applikationsmeddelanden. Strukturen för SOAP-begärandena och -svara beskrivs i WSDL- och XML-scheman.

Varje transaktion består av kundens SOAP-begärandemeddelande och Tullens SOAP-svarsmeddelande (request / response).

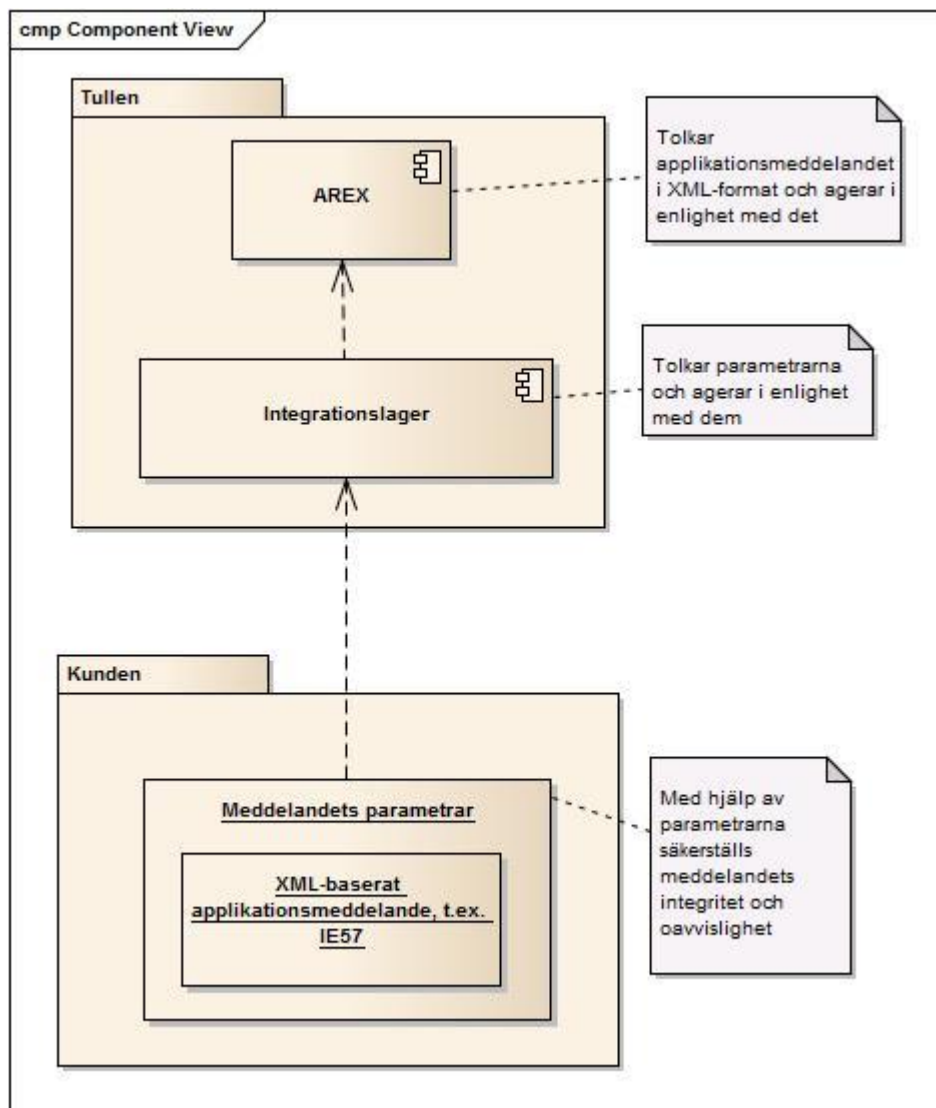


Bild 3: Inkommande meddelande från kunden

3.1.1 SOAP-begärandeddelande

SOAP-meddelandets yttersta lager är ett SOAP-kuvert (*SOAP Envelope*). En obligatorisk del av kuvertet är kroppen (*SOAP Body*). Kroppen innehåller som standard XML-elementet *RequestHeader*. Begäran innehåller också ett operationsspecifikt XML-element (operationen beskrivs senare i detta dokument).

XML-strukturen för Tullens SOAP-meddelanden (meddelandets parametrar) är av standardtyp och programoberoende medan Tullens applikationsmeddelanden som är i XML-format är programspecifika.

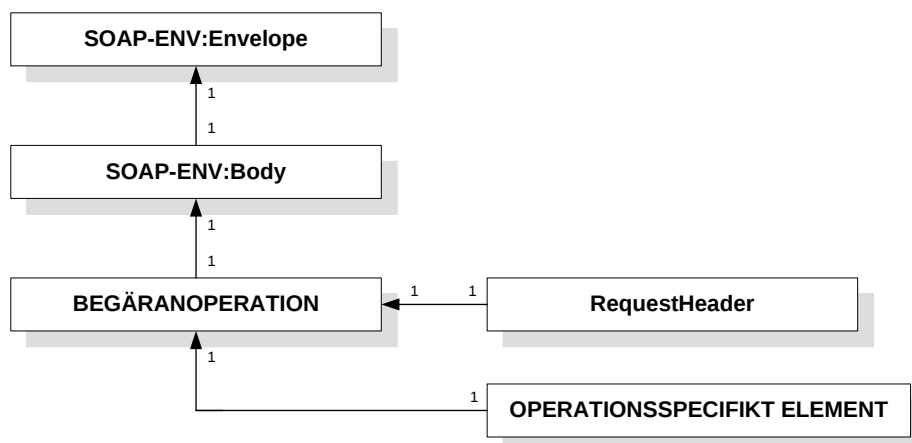


Bild 4: SOAP-begärandemeddelandets struktur på hög nivå

3.1.2 SOAP-svarsmeddelande

SOAP-svarsmeddelandets kropp innehåller som standard XML-elementet *ResponseHeader*. Svaret innehåller också ett eller flera operationsspecifika XML-element.

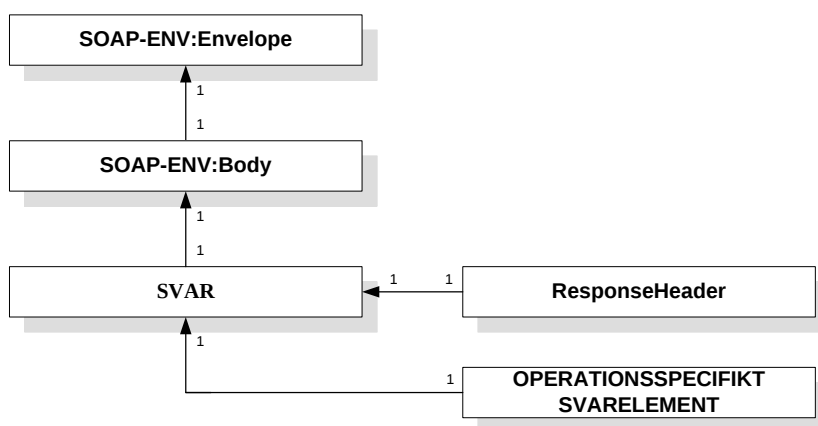


Bild 5: SOAP-svarsmeddelandets struktur på hög nivå

3.2 Svar i normal- och felsituationer

3.2.1 ResponseHeader-returkoder

I **normalsituationer** returnerar meddelandegränssnittet ett SOAP-svar vars HTTP-returvärde är '200 OK'. Värdet för XML-elementet ResponseCode, som ingår i ResponseHeader, är då '000' och innehållet i XML-elementet ResponseText är 'OK'.

I **felsituationer** försöker meddelandegränssnittet alltid returnera ett normalt SOAP-svar vars HTTP-returvärde är '200 OK'. Värdet för XML-elementet ResponseCode, som ingår i ResponseHeader, och innehållet i XML-elementet ResponseText är vid felsituationer i enlighet med det som anges i bilaga 1.

3.2.2 SOAP fault, dvs. felelement

Tullen strävar efter att i mån av möjlighet returnera en uppgift till kundens system om felsituationer i dataelementet ResponseHeader. I sällsynta och oväntade felsituationer kan det dock vara omöjligt

att processa det mottagna SOAP-meddelandet, vilket medför att ett tillbörligt svar som innehåller ResponseHeader inte kan returneras. För dylika situationer har ett s.k. *felement* (fault) definierats i WSDL-beskrivningen.

I dessa felsituationer returnerar meddelandegränssnittet ett SOAP-fault vars HTTP-returvärde är '500 Internal Error'.

Följande uppgifter anges för felementet:

- I XML-elementet code anges alltid 999 för oväntade fel
- I XML-elementet text anges för oväntade fel alltid texten *Unexpected error* eller *Backend technical error*.

Tullen förbehåller rätten att förse felement med innehåll som avviker från det ovan nämnda. Därtill finns det också felsituationer som man känner till. För dessa returneras endast de generiska SOAP-felementen faultcode och faultstring till kunden.

3.2.3 Kundprogrammets beredskap för felsituationer

Kundprogrammen ska konstrueras så att de är förberedda på de vanligaste felsituationerna. Hur man ska reagera vid en felsituation varierar beroende på vilken typ av fel det är fråga om. I bilaga 1 har felen indelats utifrån följande kriterier

- Lönar det sig för kundprogrammet att skicka meddelandet på nytt
- Måste man korrigera ett fel i meddelandet och först efter det skicka meddelandet på nytt
- Finns det fel i auktorisationsuppgifterna som ger orsak att kontakta Tullens kundstöd.

Kundprogrammen ska också kunna hantera sådana felsituationer där Tullens system inte kan returnera ens ett SOAP response utan endast ett SOAP fault.

3.2.4 Sändning av meddelande på nytt

När ett meddelande skickas på nytt ska kunden vara noggrann med när kontrollreferensen för sändningstillfället ändras så att ett dubbelmeddelande inte i misstag skickas till Tullens system. Situationen åskådliggörs i bilden nedan.

Med dubbelmeddelande avses ett meddelande som är identiskt med ett meddelande som Tullens system redan tidigare har mottagit. Dubbelmeddelanden ger upphov till korrigeringsbehov och orsakar därmed extra arbete både för kunden och Tullen.

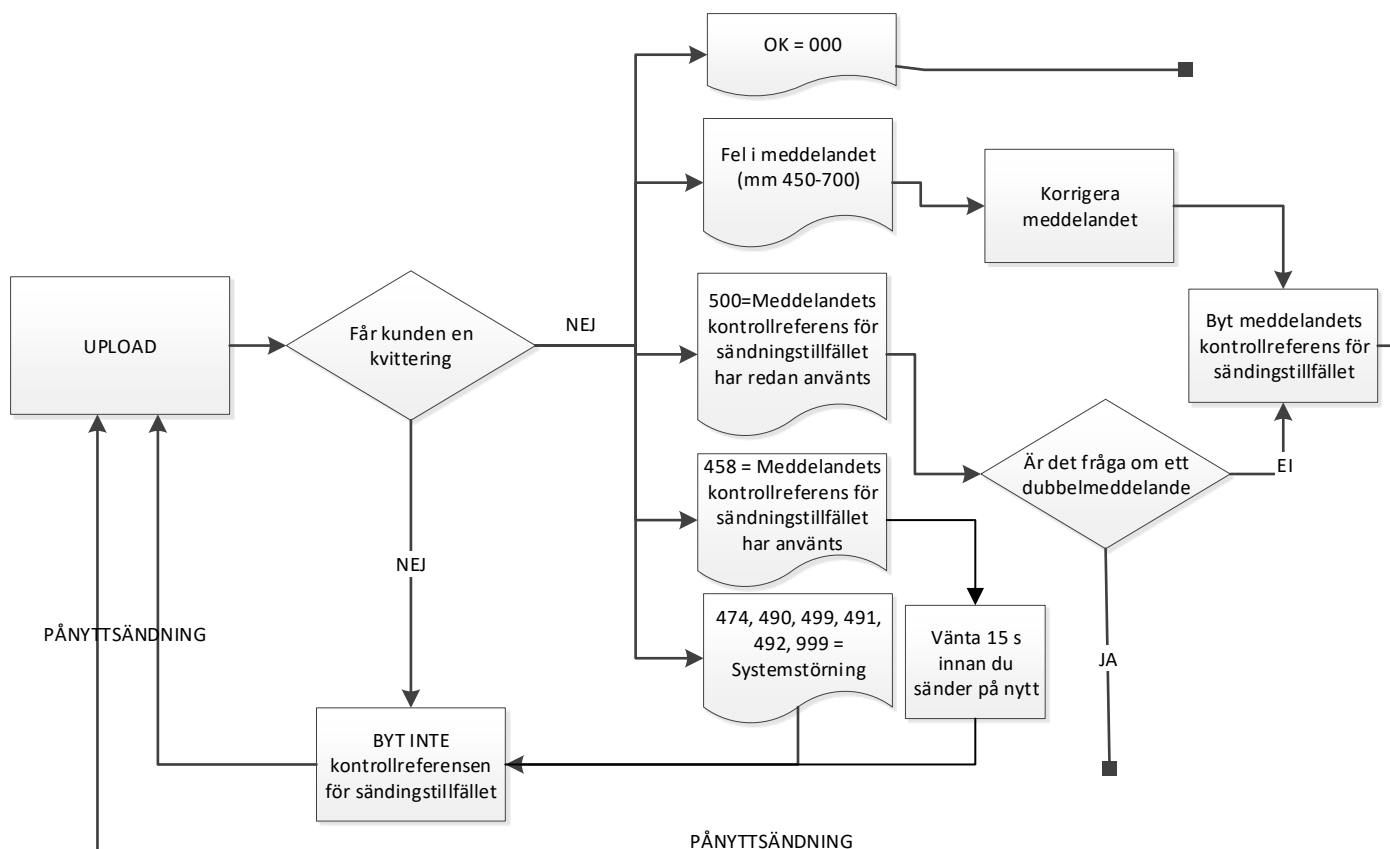


Bild 6: Sändning av meddelande på nytt och kontrollreferens för sändningstillfället

Timeouten för kundprogrammets https-kontakter till Tullens gränssnitt borde vara minst 120 s. Så säkerställer man att kundprogrammet inte avbryter sändningsförsöket eller försöker skicka samma meddelande på nytt medan meddelandet ännu är under handläggning i Tullens system.

3.3 Faserna för byggande och förmedling av ett meddelande (UPLOAD)

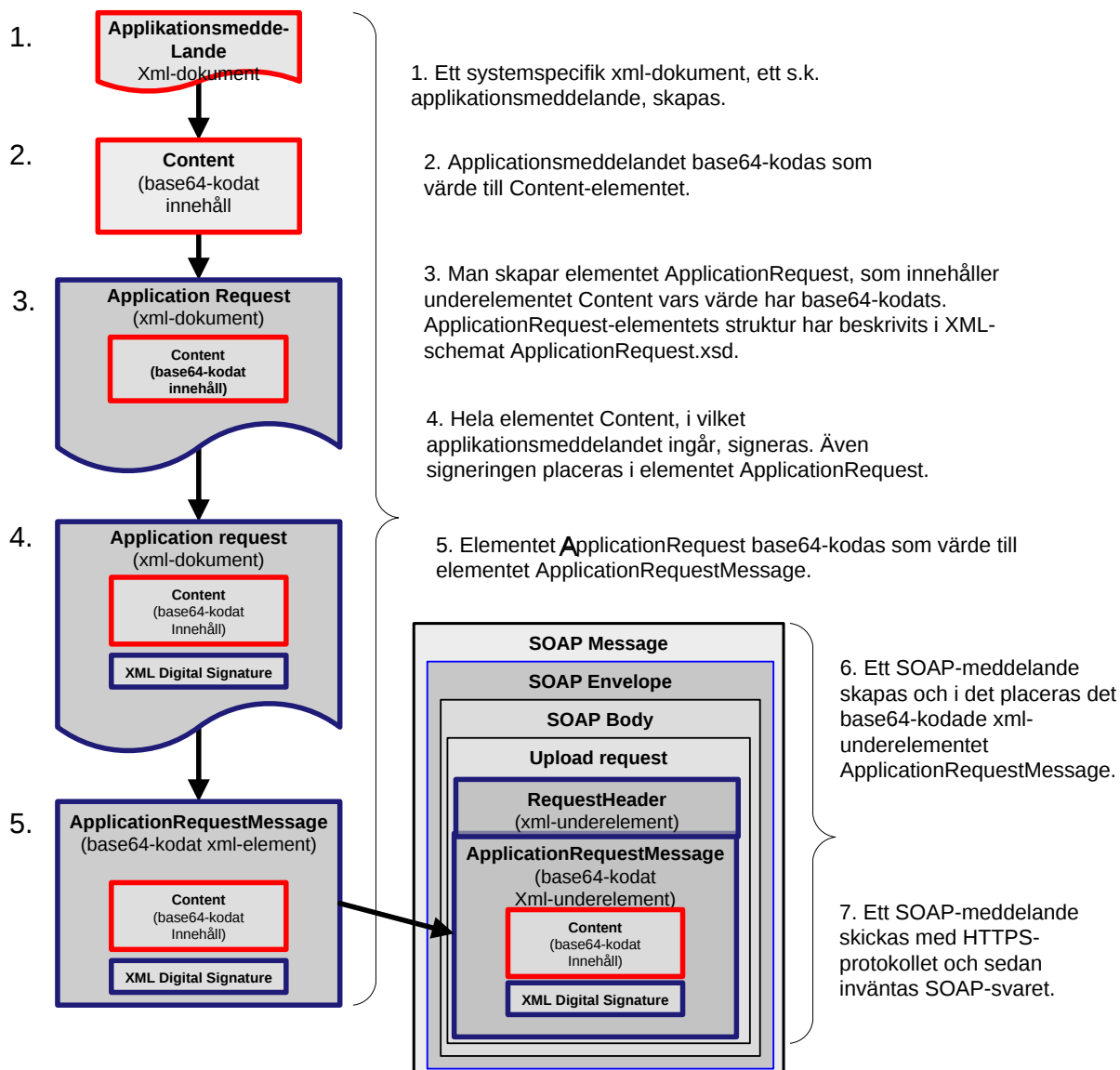


Bild 7: Hur ett meddelande byggs för Tullen

Runt det egentliga applikationsmeddelandet byggs ett `ApplicationRequest`-block. Dess innehåll består av ett `Content`-element som byggs utifrån applikationsmeddelandet samt byggarens signatur som omfattar `Content`-elementet.

Förmedlaren bäddar in det färdiga, signerade `ApplicationRequest`-blocket i base64-kodat form vidare som värde till `ApplicationRequestMessage`-elementet.

Både applikationsmeddelandet och `ApplicationRequest` base64-kodas.

3.4 Faserna för att skapa och förmedla ett meddelande (UPLOAD-ATTACHMENT)

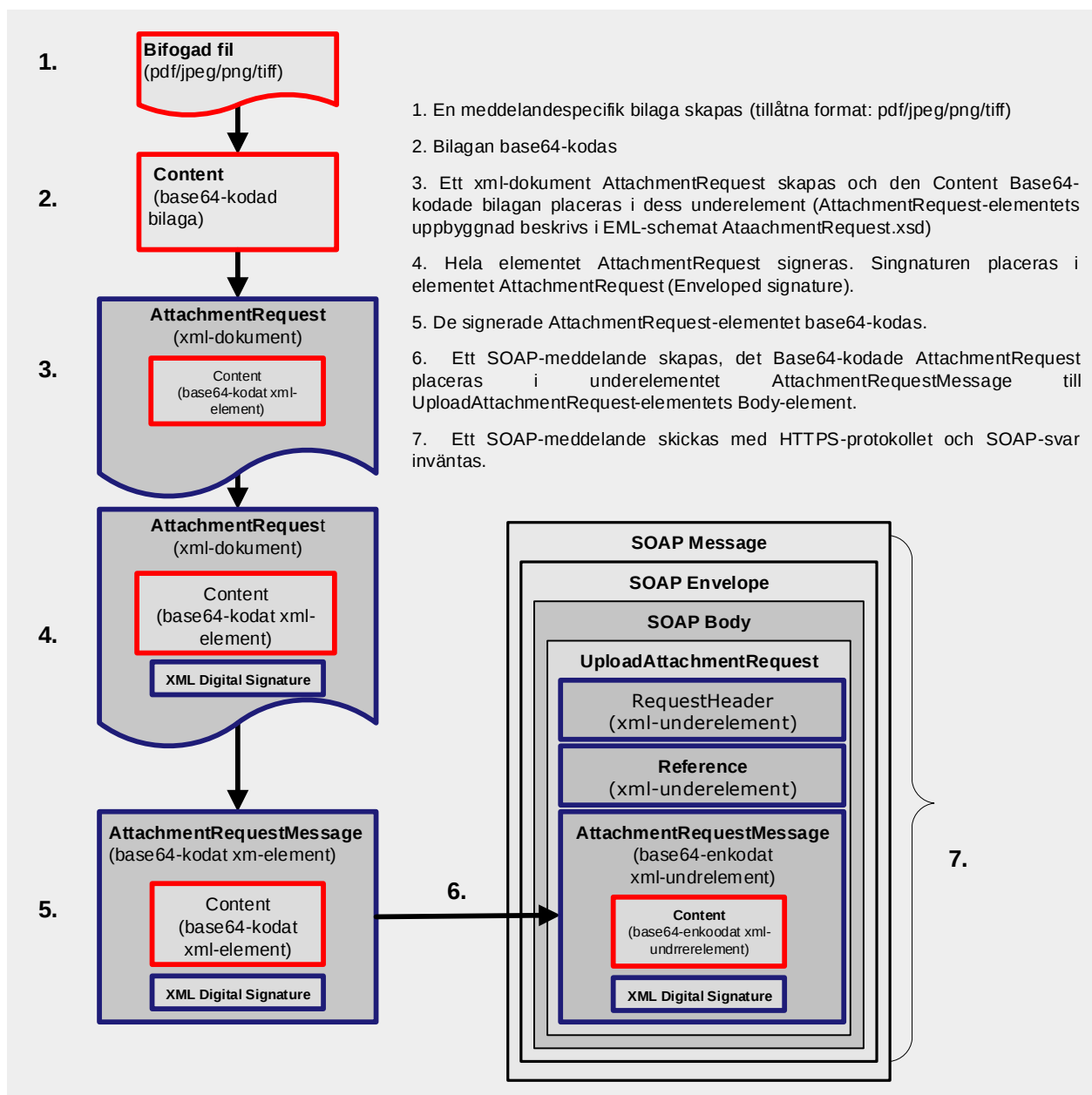


Bild 8: Skapa ett UploadAttachment-meddelande till Tullen

Runt en Base64-kodat bifogad fil bildas ett AttachmentRequest-block. Det innehåller utöver den bifogade filen även metadata om bilagan samt skaparens signatur som gäller hela AttachmentRequest-dokumentet.

Förmedlaren bäddar in detta färdiga, signerade AttachmentRequest-dokument Base64-kodat vidare som värde för AttachmentRequestMessage-elementet.

4 Tullens WSDL- och XSD-filer

Gränssnittet för meddelandeförmedling beskrivs med hjälp av en WSDL-specifikation som grundar sig på beskrivningsspråket för XML-baserade webbtjänster. WSDL-specifikationen för Tullens direkta meddelandedeklaration genomförs i filen:

- CustomsCorporateService.wsdl

WSDL hänvisar till följande XML-schemafilerna där de tillämpade meddelandestrukturerna beskrivs:

- ApplicationRequest.xsd
- ApplicationResponse.xsd
- ApplicationMessageTypes.xsd
- AttachmentRequest.xsd
- EchoContent.xsd
- WsdITypes.xsd
- xmldsig-core-schema.xsd

Om kunden vill ta i bruk meddelandenotifikationstjänsten lönar det sig också att bekanta sig med:

Meddelandenotifikationstjänstens WSDL-specifikationer

- NotificationService.wsdl

Som hänvisar till följande XML-schemafil

- NotificationTypes.xsd

Filerna är tillgängliga i form av ett distributionspaket (ZIP-arkiv) som kan laddas ner från Tullens webbplats:

<https://tulli.fi/sv/foretag/foretagsarenden/meddelandedeklarering/for-programutvecklare>

WSDL kan endast hämtas från adressen ovan. WSDL kan inte hämtas genom att skicka en HTTP GET '?wsdl'-begäran till web service-tjänstens URL-adress.

5 Meddelande om schemafel

Tullen genomför kontroller på ett applikationsmeddelande när Tullen kvitterat ett Upload-meddelande som mottaget vid direkt meddelandedeklarering. En av kontrollerna är att Tullen validerar XML-applikationsmeddelandets riktighet mot meddelandets schema.

Tullen avvisar applikationsmeddelandet om det strider mot XML-schemat. Tullen skickar ett meddelande om schemafel till kunden om avslaget. Kunden hämtar meddelandet om schemafel via direkt meddelandedeklarering på samma sätt som alla andra svarsmeddelanden från Tullen. Avsändaren av meddelandet om schemafel är det av Tullens system till vilket kundens applikationsmeddelande som innehöll fel hade skickats.

Alla andra av Tullens system utom AREX-systemet skickar ett enhetligt meddelande om schemafel när Tullen avvisar kundens applikationsmeddelande som stridande mot XML-schemat. Strukturen av meddelandet om schemafel beskrivs nedan. (AREX skickar i denna felsituation ett svarsmeddelande, som beskrivs i systemens egna meddelandebeskrivningar.)

5.1.1 DmeErrorMessage, struktur

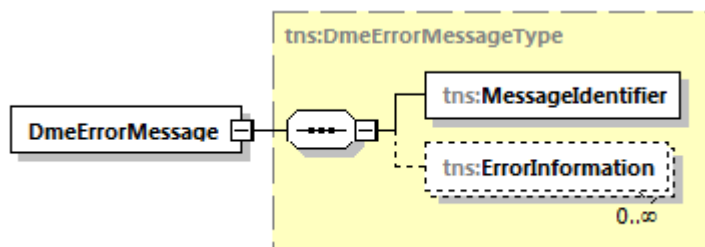


Bild 9: DmeErrorMessage:s struktur

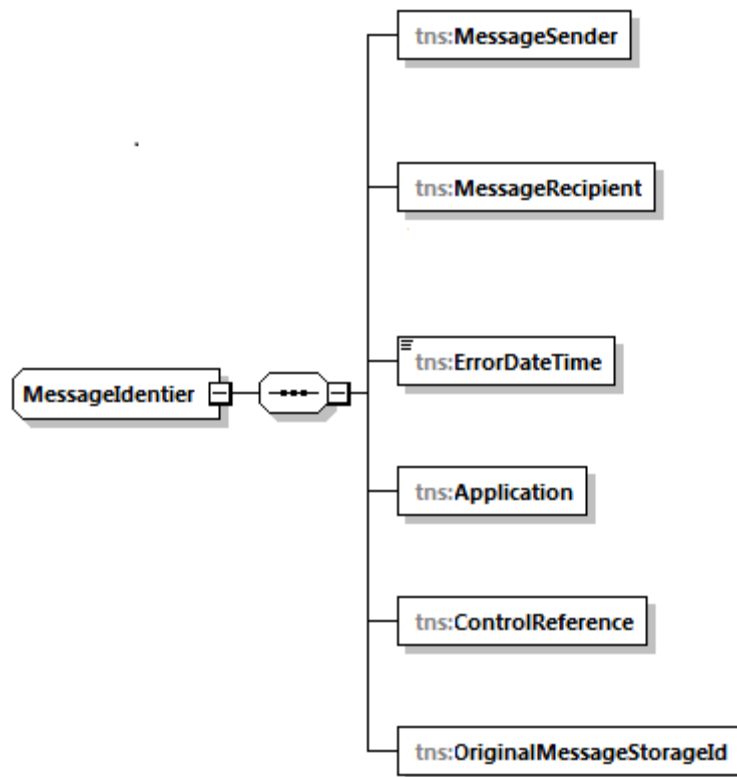


Bild 10: DmeErrorMessage – MessageIdentifier:s struktur

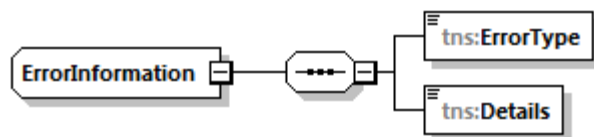


Bild 11: DmeErrorMessage – ErrorIdentifier:s struktur

5.1.2 DmeErrorMessage, beskrivning av elementen

Elementets namn	Beskrivning			
DmeErrorMessage	Dokumentets rotelement			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageIdentifier	Element med rubrikdata	Message Identifier	J	[1..1]
ErrorInformation	Element med rubrikdata	ErrorInformation	J	[0..*]

Tabell 1: DmeErrorMessage – rotelement

Elementets namn	Beskrivning			
MessageIdentifier	Element med rubrikdata			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageSender	Meddelandeavsändarens (Tullen) kod: landskod och FO-nummer	string	J	[1..1]
MessageRecipient	Meddelandemottagarens kod: landskod och FO-nummer	string	J	[1..1]
ErrorDateTime	Tidstämpel för upptäckt av fel	dateTime	J	[1..1]
Application	Kort namn på Tullens program dit deklarationen som meddelandet om schemafel hänförs sig till skickades.	string	J	[1..1]
ControlReference	En av kunden ursprungligen angiven kontrollreferens för sändningstillfället, med vilken kunden kan bunta ihop alla meddelanden som hör till transaktionen (tulldeklaration). Mera information i kapitel Virhe. Viitteen lähde ei löytynyt.. (Reference-element)	string	J	[1..1]
OriginalMessageStorageId	Meddelandets individuella kontrollreferens med vilken tulldeklarationen som bilagan hänförs sig till har sparats i Tullens system.	string	J	[1..1]

Tabell 2: DmeErrorMessage – MessageIdentifier-element

Elementets namn	Beskrivning			
ErrorInformation	Element med fel data			

Underelementets namn	Beskrivning	Typ	Obligato risk (J/N)	Förekomst
ErrorType	Typ av fel, alltid SCHEMA	Message Identifier	J	[1..1]
Details	Detaljerade uppgifter om felet, i allmänhet utskrivna av XML-parser eller schemavalidator	ErrorInformation	J	[1..1]

Tabell 3: DmeErrorMessage – ErrorInformation-element

6 Bifogade filer till svarsmeddelanden

Tullen återsänder förtullningsbesluten i XML-format i ett svarsmeddelande till kundens Download-begäran. I detta kapitel presenteras några specialfall av dokument som sänds tillbaka. Övriga förtullningsbeslut som återsänds har presenterats i samband med dokumentationen av det aktuella förtullningsförfarandet.

6.1 PDF-filer

Till vissa av Tullens svarsmeddelanden bifogas också ett elektroniskt dokument i PDF-format. Kunden kan skriva ut dokumentet och arkivera det samt använda det när han eller hon sköter tullärenden.

I export- och transiteringssystemen packas PDF-dokumenterna i ett ZIP-arkiv, eftersom PDF-filerna tar mindre utrymme då de komprimeras i ZIP-format. Vid import skickas svarsmeddelandena direkt i PDF-form till kundsystemen.

Kunden hämtar Tullens XML-svarsmeddelande och PDF-dokumentet eller ZIP-arkivet som innehåller PDF-dokumentet på en och samma gång genom att skicka en Download-begäran till web service-tjänsten för direkt meddelandedeklarering. Båda datainnehållen ingår i samma Download-svar från web service-tjänsten.

XML-svarsmeddelandet och PDF-dokumentet/ZIP-arkivet base64-kodas och tillfogas till XML-dokumentet ApplicationResponse. ApplicationResponse base64-kodas och inkluderas i Download-svaret som skickas från web service-tjänsten till kunden.

PDF-filen eller ZIP-arkivet kan vara högst 2 megabyte (2048 kilobyte) stort. Om PDF-filen eller ZIP-arkivet är större än 2 MB kan det inte skickas till kunden via web service-tjänsten för direkt meddelandedeklarering. Kunderna kan vid behov kontakta Tullens elektroniska servicecentral och komma överens om skickandet av filer som överskrider maximistorleken.

6.2 LiituResponseMessage (konvertering av bifogade filer)

Bifogade filer från kunden konverteras i Tullens integrationslager till ett format som det aktuella förtullningsprogrammet förstår. För varje bifogad fil som kunden skickar bildas vid konverteringen ett meddelande om avslag eller ett meddelande om godkännande, som återsänds till meddelandelagret i Tullens integrationslager för att hämtas av kunden. Applikationskoden (Application) för systemet som skickar meddelandet om avslag eller godkännande är alltid "LIITU" oberoende av till vilket förtullningsprogram kunden har skickat sin bilaga.

Godkännande eller avslag av en bifogad fil presenteras i ett LiituResponseMessage-meddelande som skickas till kunden. Den konverterade bifogade filen är alltid antingen godkänd eller avslagen. Konverteringen förutsätter inga extra åtgärder av kunden, utan den bifogade filen fortsätter till innehållskontrollen i det aktuella förtullningsprogrammet. Vid avslag avbryts handläggningen av den bifogade filen och orsakerna till avslaget specificeras i meddelandet som sänds till kunden. När det är fråga om en bilaga som Tullen bett om ska kunden skicka en korrigerad bilaga till Tullen.

6.2.1 LiituResponseMessage, struktur

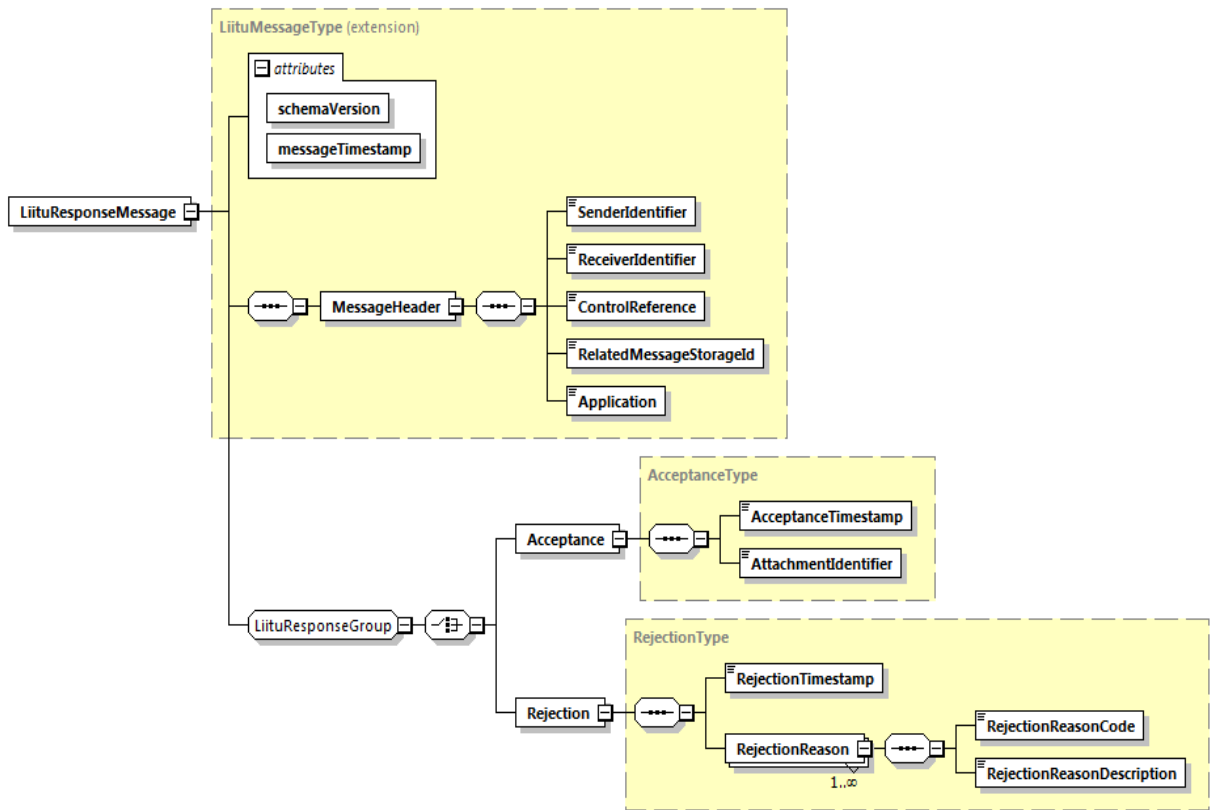


Bild 12: Beskrivning av LiituResponseMessage

6.2.2 LiituResponseMessage, beskrivning av elementen

Elementets namn	Beskrivning			
LiituResponseMessage	Dokumentets rotelement			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageHeader	Element som innehåller gemensamma data för meddelandet	Message Header	J	[1..1]
LiituResponseGroup	Består antingen av ett AcceptanceType-element eller ett RejectionType-element beroende på om bilagan godkänts eller avslagits.	AcceptanceType eller RejectionType	J	[1..1]

Tabell 4: LiituResponseMessage – rotelement

Elementets namn	Beskrivning			
MessageHeader	Element med rubrikdata			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
SenderIdentifier	Avsändarens (Tullen) kod: landkod och FO-nummer	string	J	[1..1]
ReceiverIdentifier	Mottagarens kod: landkod och FO-nummer	string	J	[1..1]
ControlReference	En av kunden ursprungligen angiven kontrollreferens för sändningstillfället, med vilken kunden kan bunta ihop alla meddelanden som hör till transaktionen (avsändning av bilaga).	string	J	[1..1]
RelatedMessageStorageId	Meddelandets individuella kontrollreferens med vilken tulldeklarationen som bilagan hänför sig till har sparats i Tullens system.	string	J	1.1
Application	Kortnamn på Tullens program dit deklarationen till vilken bilagan hänför sig har skickats.	string	J	[1..1]

Tabell 5: LiituResponseMessage – MessageHeader-element

Elementets namn	Beskrivning			
Acceptance	Element som anger den godkända konverteringen av bilagan			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
AcceptanceTimestamp	Tidstämpel för bilagekonvertering	dateTime	J	[1..1]
AttachmentIdentifier	Identifierare för konverterad bilaga i Liitu.	string	J	[1..1]

Tabell 6: LiituResponseMessage - Acceptance-element

Elementets namn	Beskrivning			
Rejection	Element som anger den avslagna konverteringen av bilagan			
Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
RejectionTimestamp	Tidstämpel för bilagekonvertering	string	J	[1..1]

RejectionReason	Element som innehåller de konverteringsfel som lett till avslaget	RejectionReason	J	[1..*]
-----------------	---	-----------------	---	--------

Tabell 7: LiituResponseMessage - Rejection-element

Elementets namn	Beskrivning			
RejectionReason	Element som beskriver ett enskilt konverteringsavslag			
Underelementets namn	Beskrivning	Typ	Obliga- torisk (J/N)	Förekomst
RejectionReasonCode	Kontrollreferens som identifierar konverteringsavslaget	string	J	[1..1]
RejectionReasonDescription	Beskrivning av konverteringsavslaget	string	J	[1..1]

Tabell 8: LiituResponseMessage - RejectionReason-element

6.3 Meddelande innehållande PDF-dokumentets metadata (DmeDocumentInfoMessage)

Tullens nya tullklareringssystem (UTU) bygger upp de beslutsdokument som skapas på basis av tulldeklarationerna på ett annat sätt än de gamla tullklareringssystemen. Till skillnad från de gamla tullklareringssystemen ska dokumenten i XML- och PDF-format hämtas i UTU från Tullens meddelandelager med hjälp av separata web service-operationer.

Ett nytt dokument i PDF-format är i fortsättningen alltid förknippat med ett meddelande som innehåller PDF-dokumentets metadata. Meddelandets struktur beskrivs nedan.

6.3.1 DmeDocumentInfoMessage, beskrivning av elementen

DmeDocumentInfoMessage Datastruktur som innehåller PDF-dokumentets metadata

Underelementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageRecipient	Dokumentmottagarens (kund) kod: landskod och FO-nummer	string	J	[1..1]
DocumentId	PDF-dokumentets entydiga identifikationsnummer	string	J	[1..1]
RelatedMessageStorageId	XML-beslutsdokumentets entydiga identifikationsnummer i Tullens gränssnitt för direkt meddelandedeklarering	string	J	[1..1]
ControlReference	Entydigt identifikationsnummer med anknytning till en tullklarering, kopplar tulldeklarationen och förtullningsbesluten (både i PDF- och XML-format) till varandra. Tilldelat av kunden	string	N	[0..1]

CustomsDocumentReference	Identifikationsnummer tilldelat av Tullen för en tullklarering, vanligen ett MRN-nummer	string	N	[0..1]
DocumentType	Dokumentets MIME-mediatyp, application/PDF	string	J	[1..1]
DocumentCreationTime	Tid då PDF-dokumentet skapats i Tullens system Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid som standard.	datetime	J	[1..1]

Tabell 9: DmeDocumentInfoMessage, beskrivning av elementen

6.3.2 Hämtning av PDF-dokument från gränssnittet för direkt meddelandedeklarering

Kunden får besked om PDF-dokument och meddelanden innehållande PDF-dokumentens metadata på samma sätt som om vanliga beslut i XML-format: antingen i DownloadList-svaret för direkt meddelandedeklarering eller i meddelandenotifikationstjänstens meddelande som skickas av Tullen.

Meddelanden som innehåller PDF-dokumentens metadata hämtas från gränssnittet för direkt meddelandedeklarering på samma sätt som vanliga beslut i XML-format: med Download-operation.

PDF-dokumentet hämtas från gränssnittet för direkt meddelandedeklarering också med Download-operationen men då måste det DocumentId-element som angetts i metadata-meddelandet och specificerar PDF-dokumentet läggas till i DownloadMessageFilteringCriteria-elementet i det DownloadRequest-meddelande som skickas.

7 Operationer som tjänsten omfattar

Web service-tjänsten för direkt meddelandedeklarering innehåller följande SOAP-operationer:

7.1 Tjänster som implementeras av Tullen

Upload = Används vid sändning av en dataentitet, t.ex. en tulldeklaration, till Tullens system.

UploadAttachment = Används för sändning av en bifogad fil som hänför sig till en tulldeklaration till Tullens system. Tulldeklarationen ska sändas före bilagan.

DownloadList = Genererar en lista över grunduppgifterna för svarsmeddelanden som väntar på att hämtas (download). Uppgifter som härigenom erhållits som svar kan användas som söknyckel vid download-funktionen.

Download = Används vid hämtning av en dataentitet från Tullens system. Med denna funktion kan man exempelvis hämta ett svar till en tulldeklaration som skickats tidigare.

CheckConnectivity = Används vid testning för att verifiera den tekniska kompatibiliteten mellan kundens och Tullens system.

7.2 Tjänst som implementeras av kunden

Om kunden använder meddelandenotifikationstjänsten så kontaktar Tullens Notify-tjänst kundens tjänst.

Notify = Kontaktar den av kunden implementerade tjänsten för att informera kunden om svarsmeddelanden som väntar på att hämtas i Tullens system.

7.3 Tjänsteoperationer i WSDL

I WSDL har tjänstens funktioner grupperats i logiska operationer. En typisk operation består av två sammanlänkade (huvud)dataelement: begäran och svar. Gränssnittet för meddelandeförmedling erbjuder operationerna som beskrivs i tabellerna nedan.

7.3.1 Upload

Operation	Beskrivning
Upload	Används vid sändning av en dataentitet, t.ex. ett meddelande som innehåller en tulldeklaration, till Tullens system. När funktionen är genomförd fås en teknisk kvittering på att meddelandet har tagits emot (= tagits upp till handläggning). Efter den tekniska kvitteringen handläggs dataentiteten i Tullens system. Den fortsatta handläggningen leder (vanligtvis) till att ett svarsmeddelande genereras. Svarsmeddelandet måste hämtas med en separat download-operation. Listan över meddelandena som kan hämtas fås med DownloadList-operationen (se operationerna nedan) eller, om meddelandenotifikationstjänsten är i bruk, så skickar Tullens system en meddelandenotifikation om att ett meddelande väntar på att hämtas.
	Begäran
	UploadRequest
	RequestHeader
	ApplicationRequestMessage
	Svar
	UploadResponse
	ResponseHeader
	MessageInformation

Tabell 10: Upload

7.3.1.1 Upload, begäran

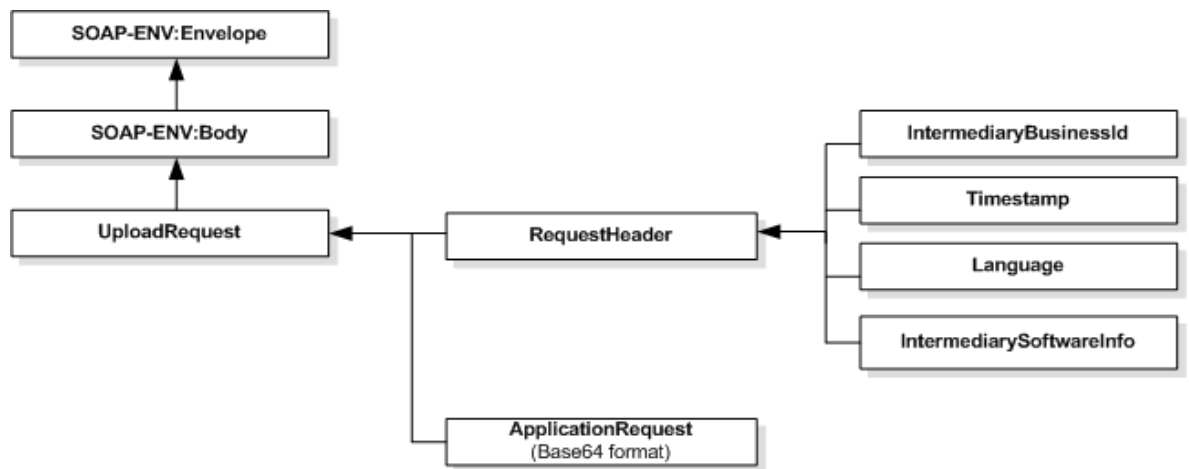


Bild 13: Upload, begäran

7.3.1.2 ApplicationRequest:

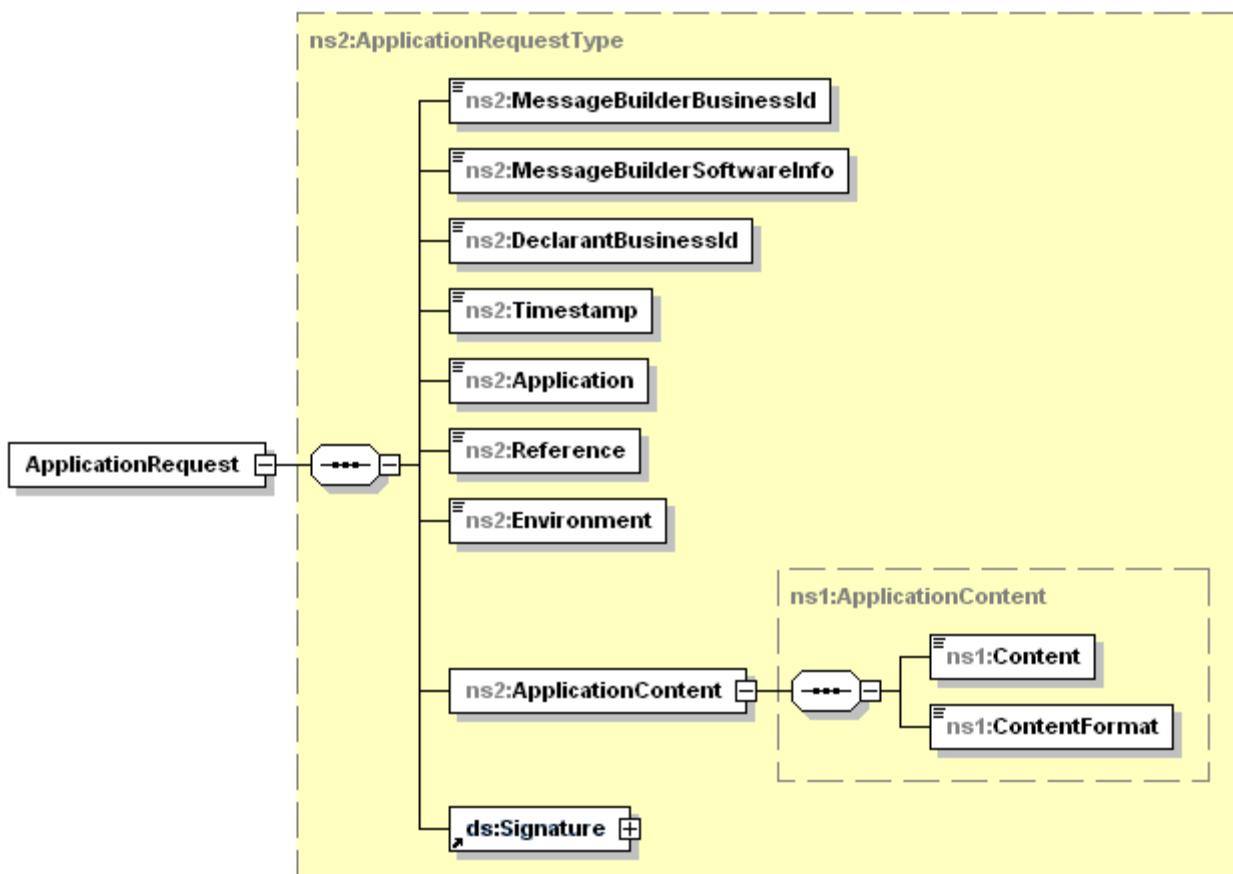


Bild 14: Beskrivning av ApplicationRequest

7.3.1.3 Upload, svar

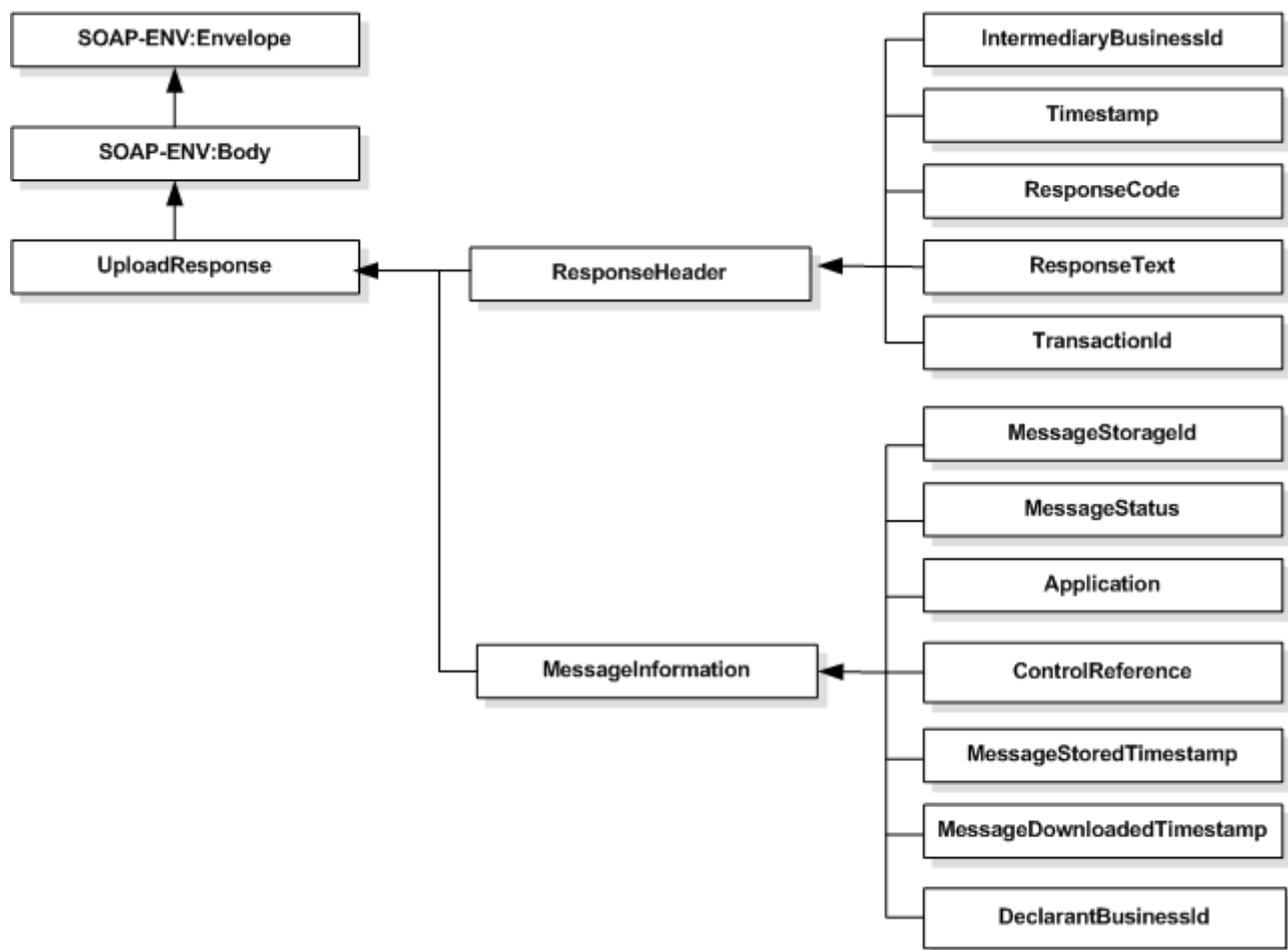


Bild 15: Upload, svar

7.3.2 UploadAttachment

Operation	Beskrivning
UploadAttachment	Används för att sända ett meddelande med en bifogad fil till Tullens system. När funktionen är genomförd fås en teknisk kvittering på att meddelandet har tagits emot (= tagits upp till handläggning). Efter den tekniska kvitteringen handläggs den bifogade filen som ingår i meddelandet i Tullens system. Handläggningen genererar till slut ett svarsmeddelande vars innehåll anger om handläggningen av bilagan lyckats. Svarsmeddelandet ska hämtas separat med hjälp av en separat Download-operation. En lista över meddelanden som kan hämtas fås med DownloadList-operationen (se operationerna nedan) eller, om notifikationstjänsten är i bruk, skickar Tullens system en notifikation om att ett meddelande väntar på att hämtas. Ett svarsmeddelande om att bilagan är under handläggning garanterar inte att bilagan har godkänts. Bilagan godkänns bara om den hänför sig till ett godkänt deklarationsmeddelande. Mera information finns i kapitel 2.1.4 Funktionerna och processen när meddelandetjänsten för bifogade filer används i dokumentet "Introduktion för meddelandetrafik med Tullen".
Begäran	
UploadAttachmentRequest	
RequestHeader	
Reference	
AttachmentRequestMessage	
Svar	
UploadAttachmentResponse	
ResponseHeader	
AttachmentRequestMessageInformation	

Tabell 11: UploadAttachment

7.3.2.1 UploadAttachment, begäran

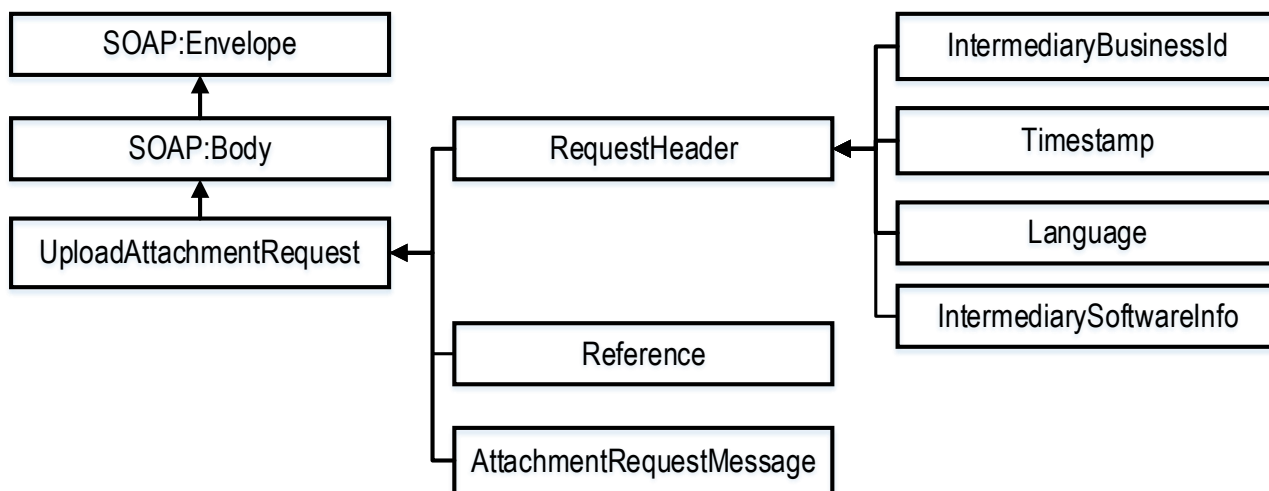


Bild 16: UploadAttachment, begäran

7.3.2.2 AttachmentRequest

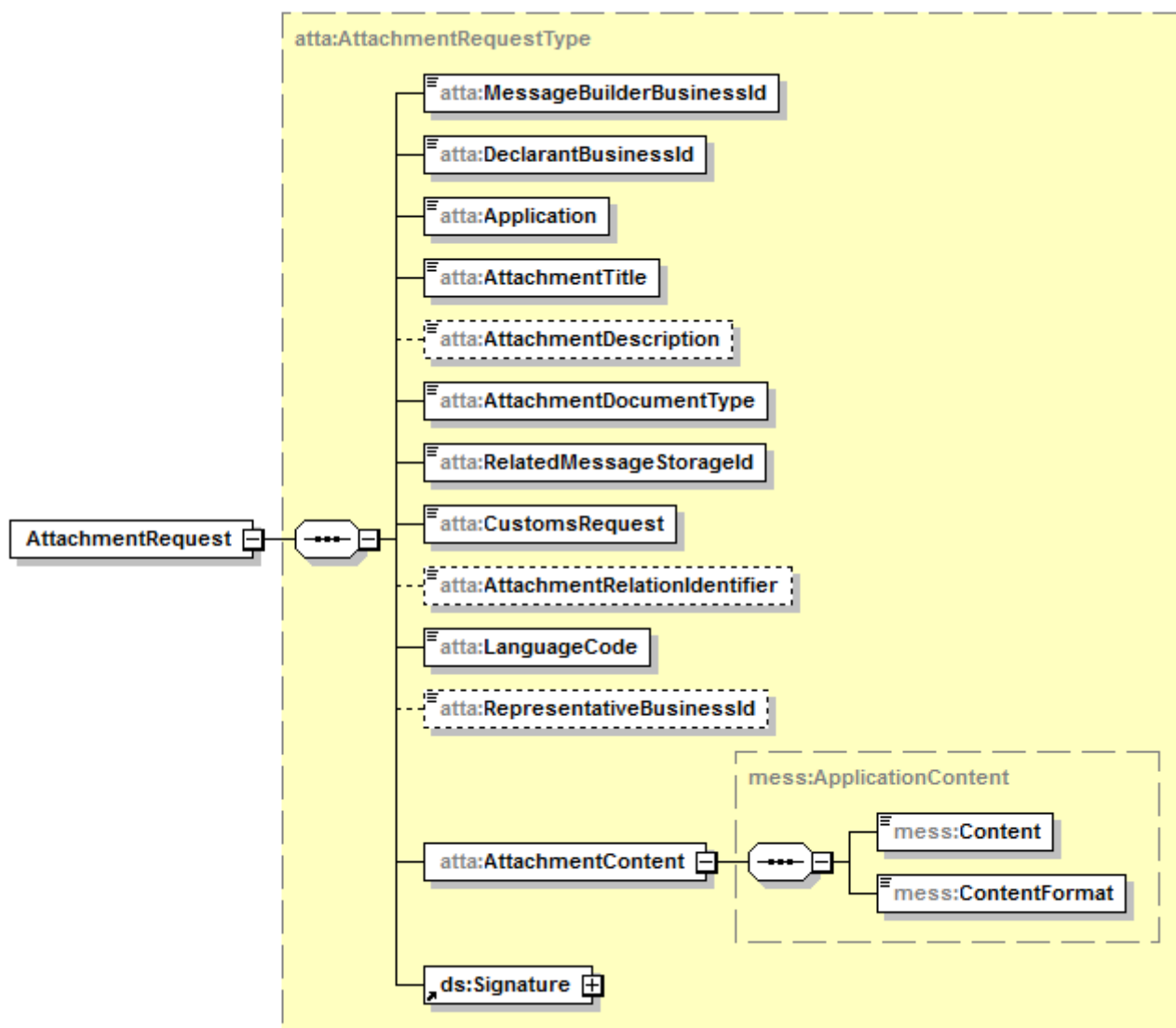


Bild 17: Beskrivning av AttachmentRequest

7.3.2.3 UploadAttachment, svar

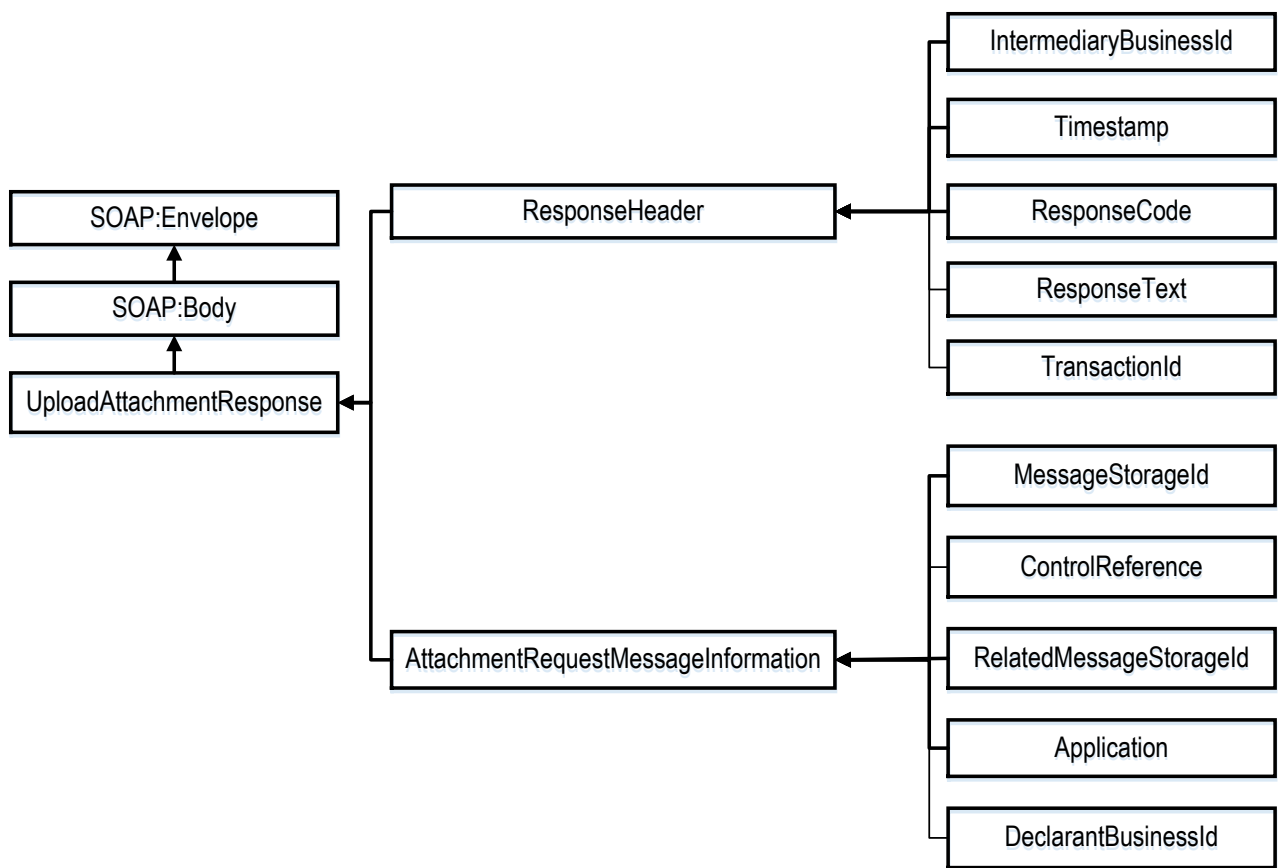


Bild 18: UploadAttachment, svar

7.3.3 DownloadList

Operation	Beskrivning
DownloadList	Används vid hämtning av datalista från Tullens system. Innehåller en lista över grunduppgifterna för svarsmeddelanden som väntar på att hämtas (download). Uppgifter som härigenom erhållits som svar kan användas som söknnyckel vid download-funktionen.
	Begäran
	DownloadListRequest
	RequestHeader
	DownloadMessageListFilteringCriteria
	Svar
	DownloadListResponse

ResponseHeader
DownloadMessageListFilteringCriteria
MessageInformation

Tabell 12: DownloadList

7.3.3.1 DownloadList, begäran

När en begäran om DownloadList görs kan söktiden fastställas med elementen Timestamp eller Date.

Det rekommenderas att Timestamps används eftersom man med elementet Timestamp kan göra en mer exakt avgränsning av sökningen, vilket i sin tur medför att sökfunktionen blir snabbare och effektivare. Särskilt i fall där begäran om DownloadList görs t.ex. med fem minuters mellanrum är det ändamålsenligt att avgränsa begäran så att den omfattar meddelanden från den senaste halvtimmen eller timmen och inte från hela dygnet.

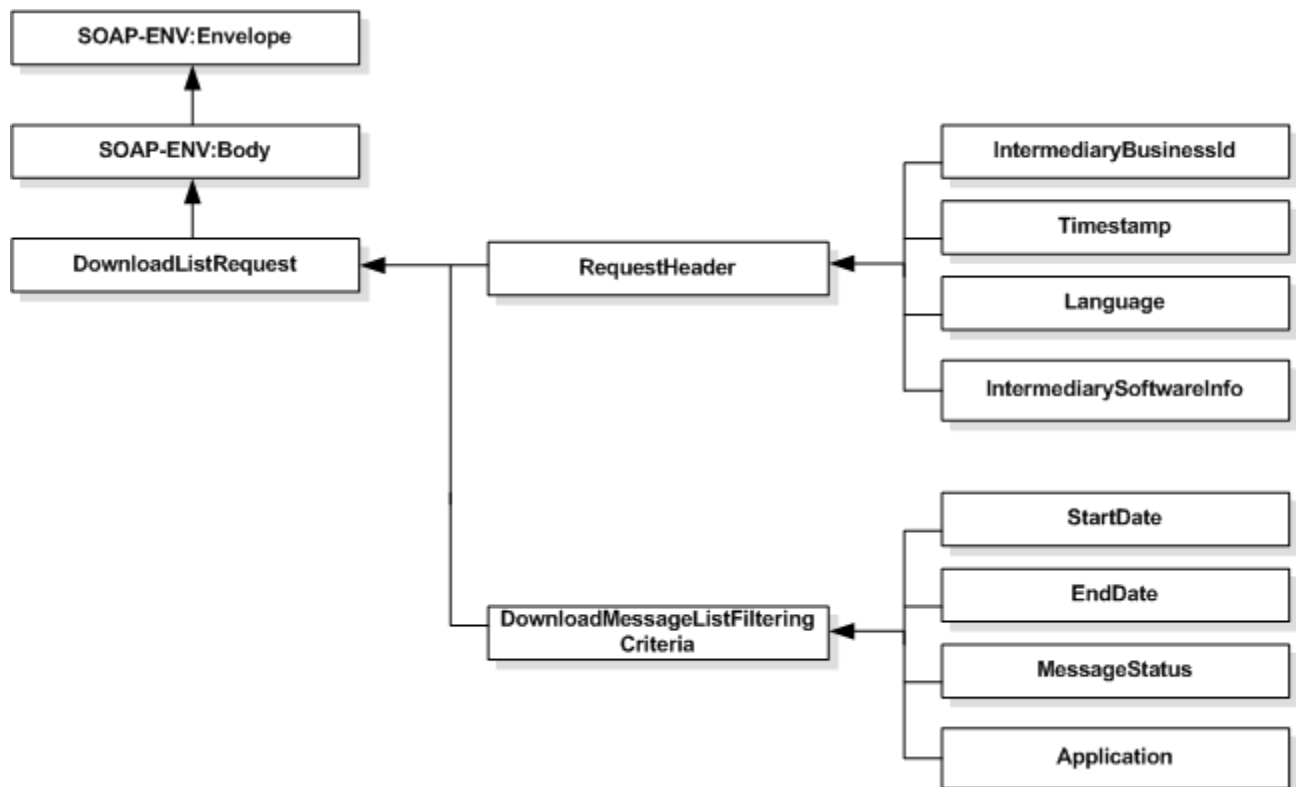


Bild 19: DownloadList, begäran med elementen StartDate och EndDate

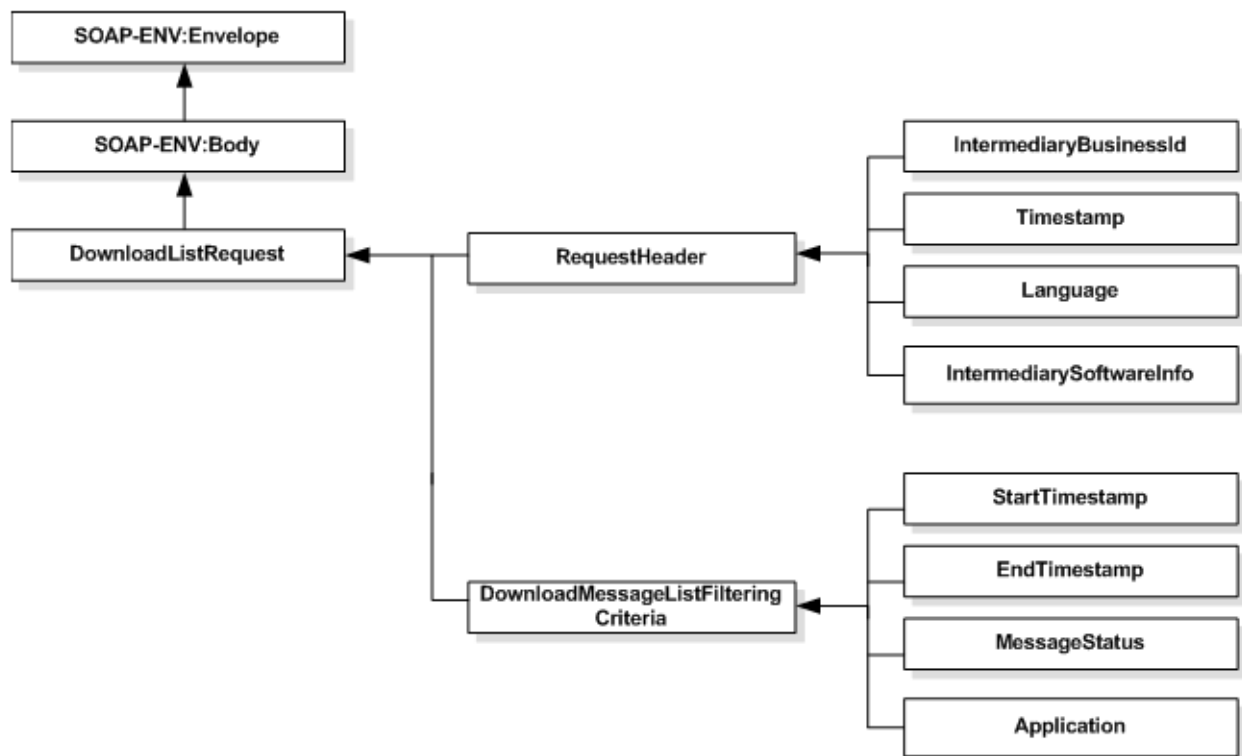


Bild 20: DownloadList, begäran med elementen StartTimestamp ja EndTimestamp

DownloadList, svar

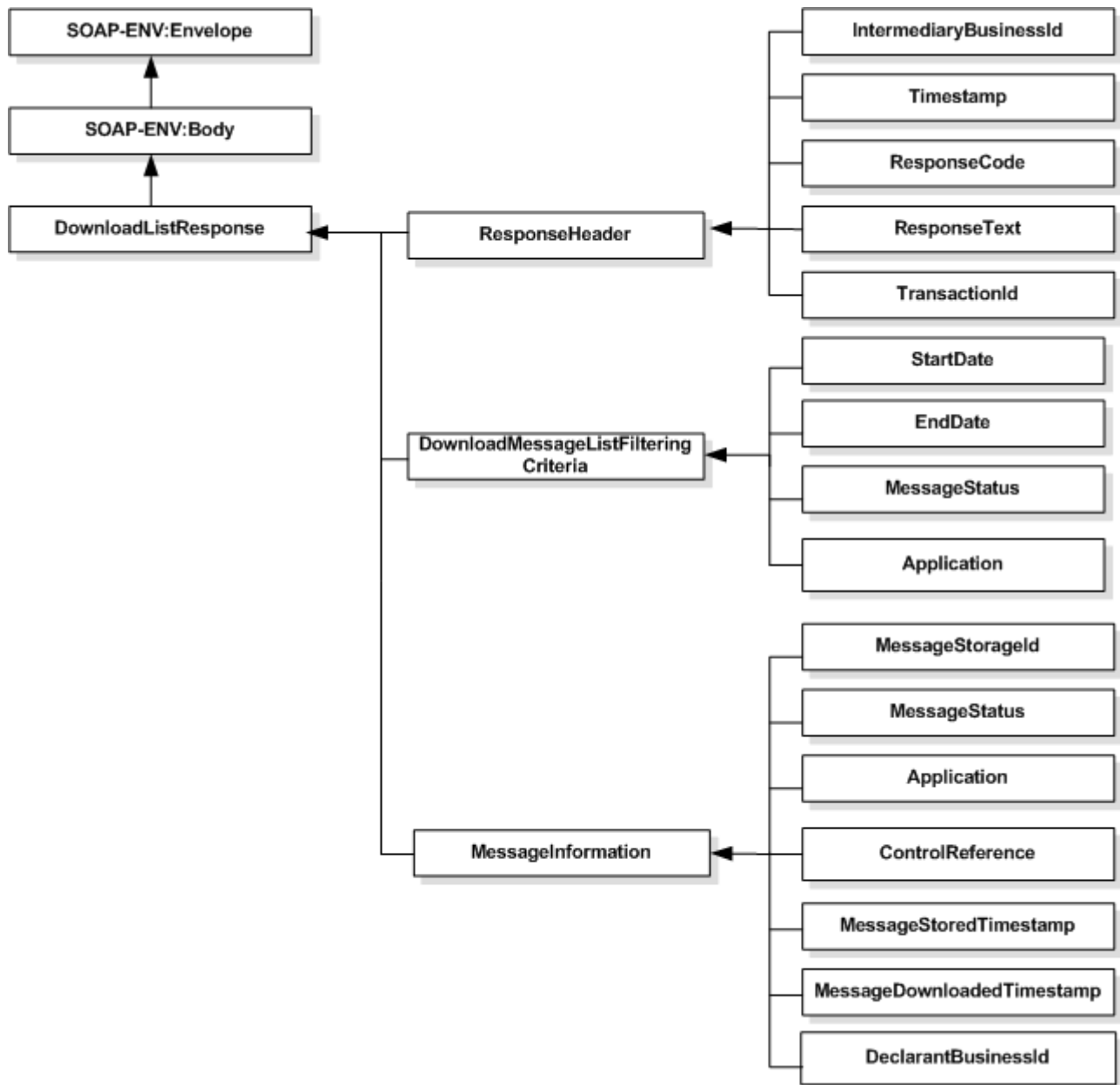


Bild 21: DownloadList, svar

7.3.4 Download

Operation	Beskrivning
Download	Används vid hämtning av en dataentitet från Tullens system. Med denna funktion kan man exempelvis hämta ett svar till en tulldeklaration som skickats tidigare.
Begäran	
DownloadRequest	
RequestHeader	
DownloadMessageFilteringCriteria	
Svar	
DownloadResponse	
ResponseHeader	
ApplicationResponseMessageInformation	
ApplicationResponseMessage	

Tabell 13: Download

7.3.4.1 Download, begäran

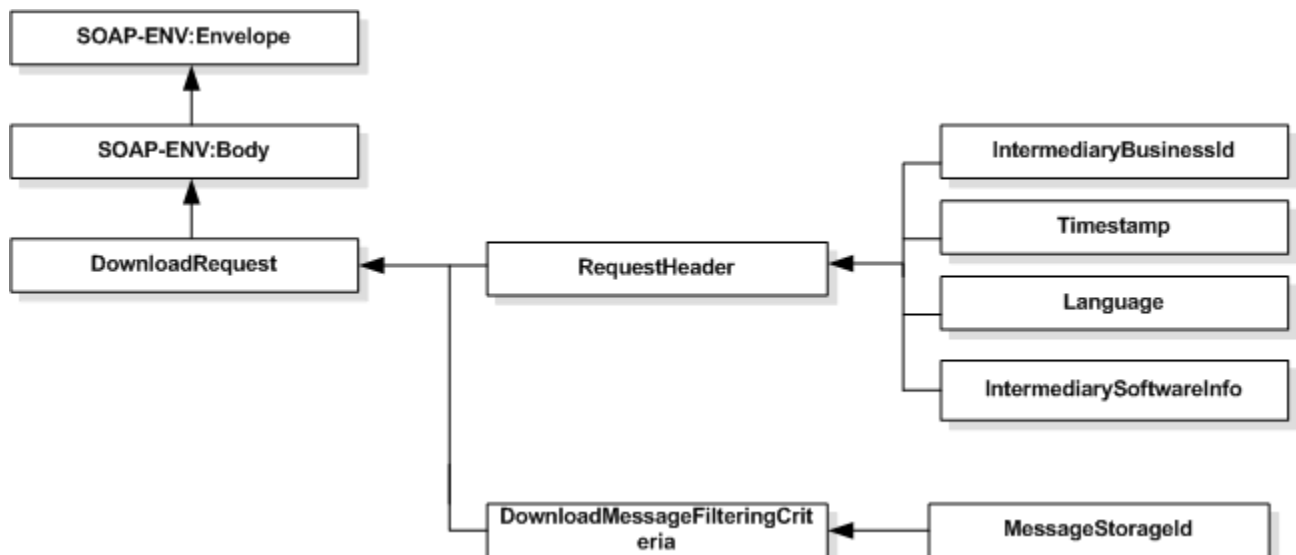


Bild 22: Download, begäran

7.3.4.2 Download, svar

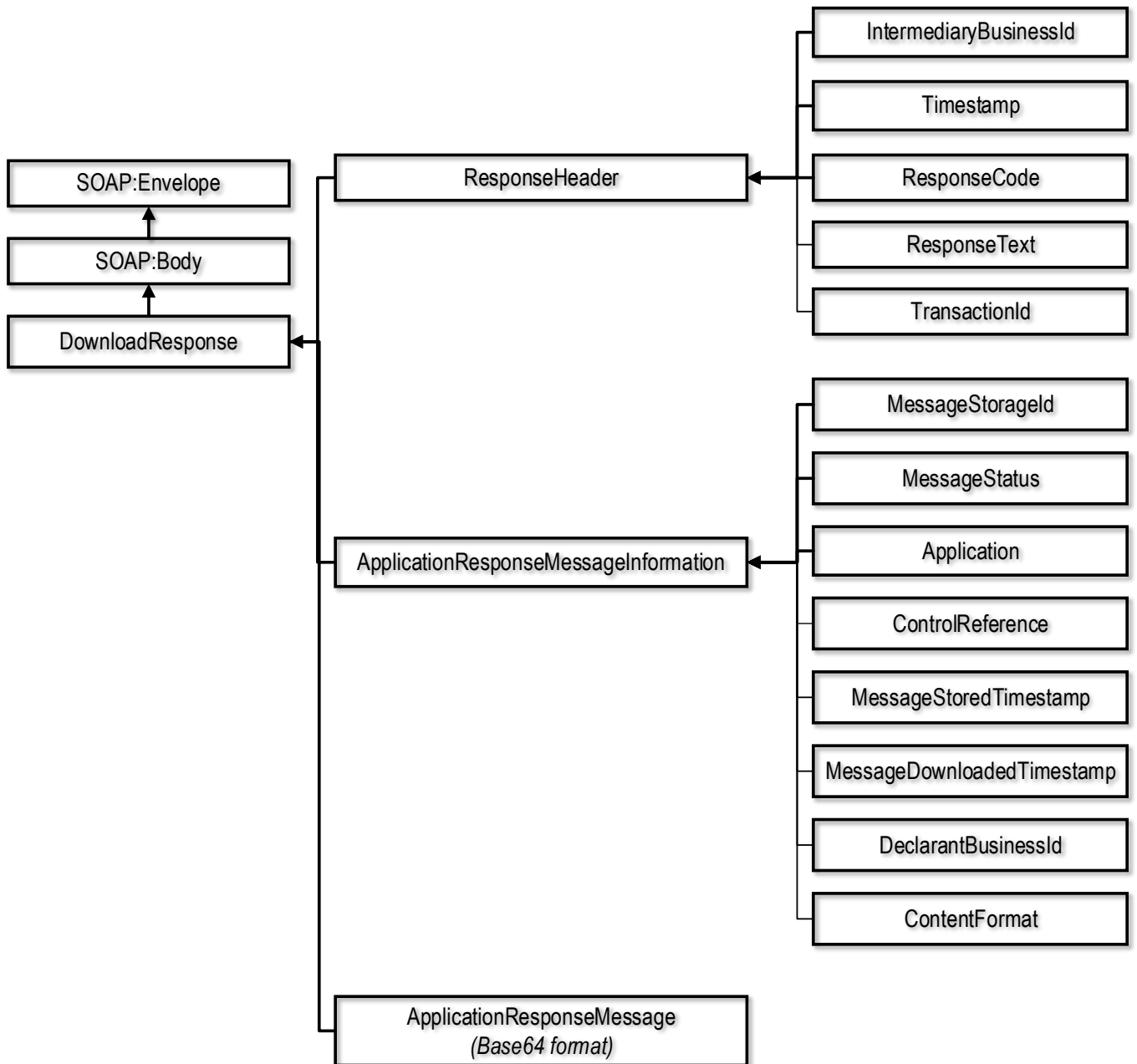


Bild 23: Download, svar

7.3.4.3 ApplicationResponse

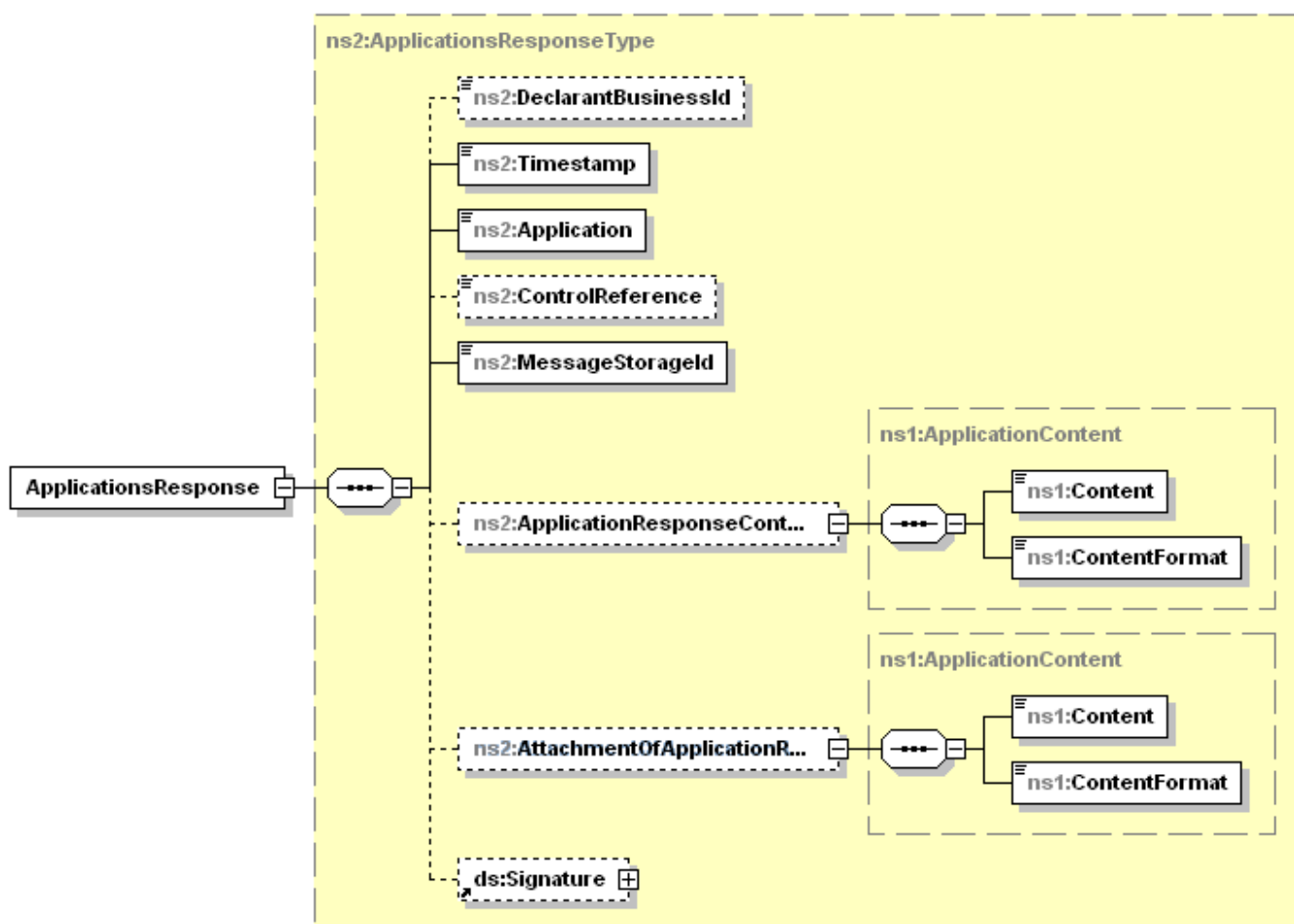


Bild 24: Beskrivning av ApplicationResponse

7.3.5 Notify

Operation	Beskrivning
Notify	Används när Tullen meddelar till kunden att ett meddelande väntar på att hämtas. Förutsätter att kunden har byggt en tjänst som Tullen kan kontakta. En notifikation skickas för varje meddelande som väntar på att hämtas. Begäran NotifyRequest RequestHeader MessageInformation Svar NotifyResponse ResponseHeader ResponseCode ResponseText

Tabell 14: Notify

7.3.5.1 NotifyRequest (skickas av Tullen)

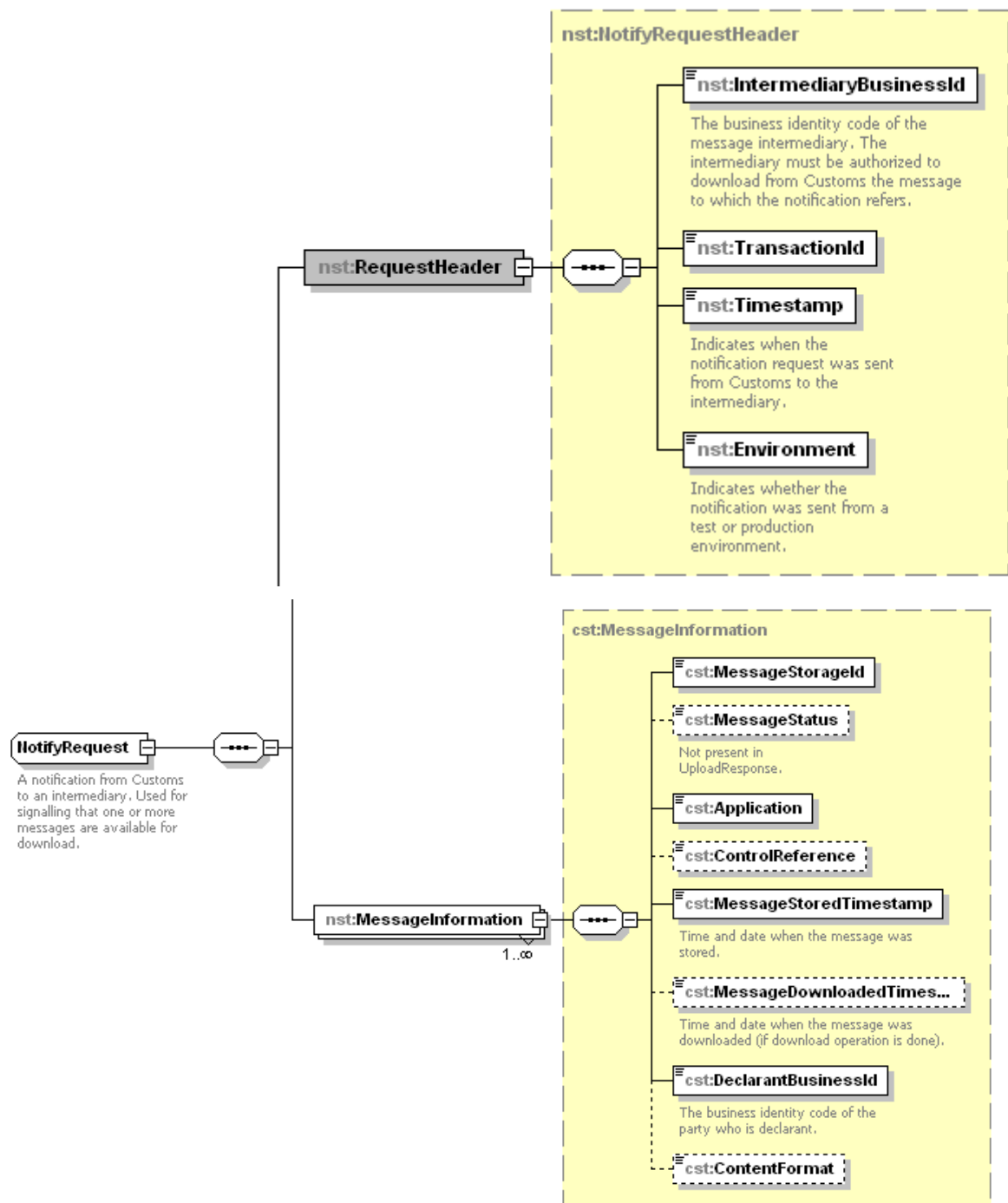


Bild 25: Beskrivning av NotifyRequest som skickas av Tullen

NotifyResponse (skickas av kunden)

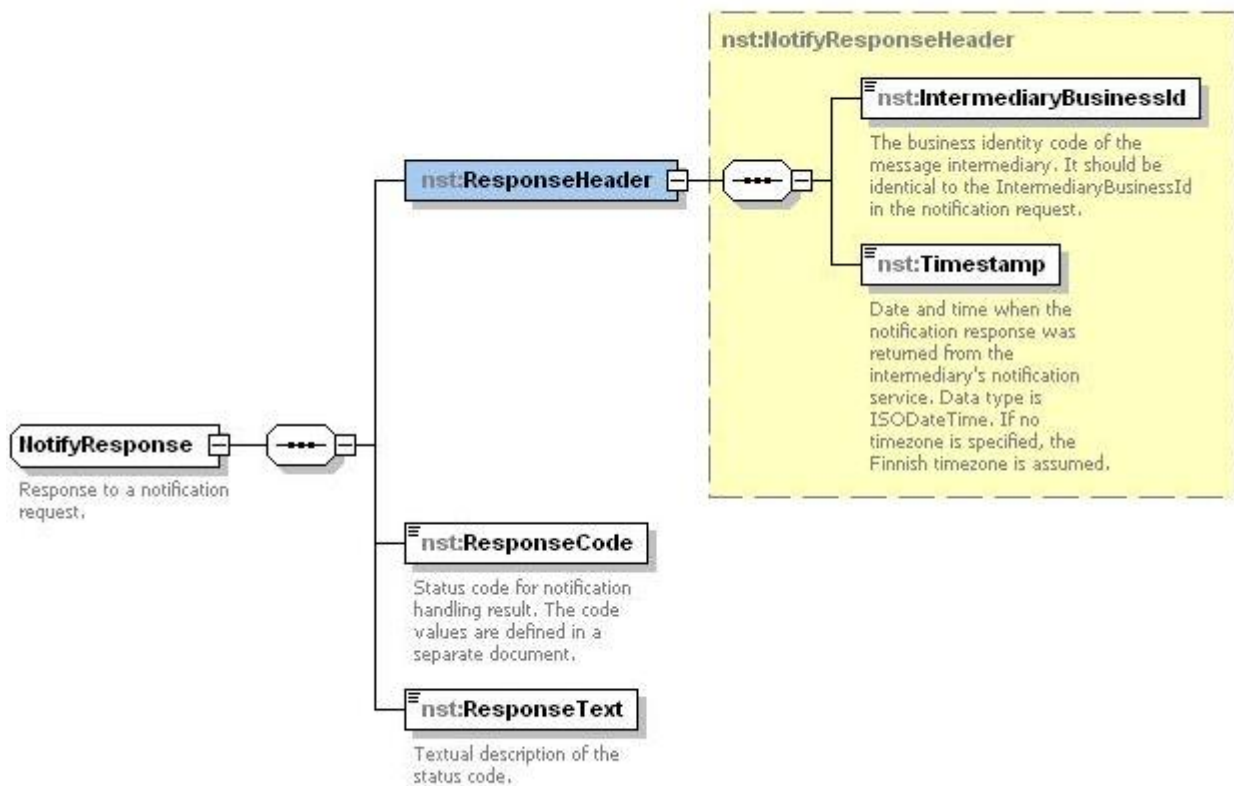


Bild 26: Beskrivning av NotifyResponse som skickas av kunden

7.3.6 CheckConnectivity

Operation	Beskrivning
CheckConnectivity	Används vid testning för att verifiera den tekniska kompatibiliteten mellan kundens och Tullens system. I praktiken kontrollerar man att den meddelandeförmedling mellan parterna som sker via HTTPS-/SOAP-protokollet fungerar. Certifikaten hos XML-meddelandets förmedlare och byggare (autentisering och auktorisation av parterna) kontrolleras också.
	Begäran
	CheckRequest
	RequestHeader
	EchoRequest
	Svar
	CheckResponse
	ResponseHeader
	EchoResponse

Tabell 15: CheckConnectivity

8 Beskrivning av uppgifterna i XML-elementen

8.1 RequestHeader

Elementets namn	Beskrivning			
RequestHeader	Detta element utgör rubriken för varje begäran som skickas till WS-gränssnittet. Denna rubrik byggs av förmedlaren.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
IntermediaryBusinessId	Förmedlarens landskod och FO-nummer.	string	J	[1..1]
Timestamp	Tid och datum då begäran skickats. Datatypen är ISODateTime. Om tidszon inte har fastställts använder man finsk lokaltid som standard.	dateTime	J	[1..1]
Language	Språkkod med vilken anges med vilket språk texterna i informativa dataelement visas. För närvarande bara EN (engelska) stöds.	string	E	[0..1]
IntermediarySoftwareInfo	Namn och versionsnummer på programmet som förmedlaren använder.	string	J	[1..1]

Tabell 16: RequestHeader

8.2 ApplicationRequestMessage

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
ApplicationRequestMessage	Innehåller ett XML-meddelande som upprättats av byggaren. XML är base64-kodat.	base64Binary	J	[1..1]

Tabell 17: ApplicationRequestMessage

8.3 ResponseHeader

Elementets namn	Beskrivning			
ResponseHeader	Detta element utgör rubriken för varje svar som WS-gränssnittet returnerar.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
IntermediaryBusinessId	Landskod och FO-nummer för begärandemeddelandets förmedlare	string	J	[1..1]
Timestamp	Tid och datum då svaret skickats.	dateTime	J	[1..1]

	Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.			
ResponseCode	Returkod som anger om åtgärderna som begäran lett till har lyckats eller misslyckats (se bilaga 1).	string	J	[1..1]
ResponseText	Verbal förklaring av returkoden (på engelska). (Se bilaga 1)	string	J	[1..1]
TransactionId	En av Tullens integrationslager genererad unik transaktionskod, med vilket begäran och svar kan buntas ihop.	string	J	1.1

Tabell 18: ResponseHeader

8.4 MessageInformation

Elementets namn	Beskrivning			
MessageInformation	Detta element innehåller meddelandets grunduppgifter.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageStorageId	Meddelandets unika ID med vilket ett svarsmeddelande kan hämtas från Tullens system.	string	J	[1..1]
MessageStatus	Meddelandets status: - NEW = Meddelandet har ännu inte hämtats. - DLD = Meddelandet har redan hämtats. - ALL – Alla meddelanden.	string	E	[0..1]
Application	Kortnamn på Tullens program som meddelandet hänför sig till.	string	J	[1..1]
ControlReference	En av kunden (XML-meddelandets byggare) ursprungligen angiven kontrollreferens för sändningstillfället, med vilken kunden kan bunta ihop alla meddelanden som hör till transaktionen. Mera information ges i beskrivningen av <i>ApplicationRequest</i> -elementets underelement <i>Reference</i> .	string	E	[0..1]
MessageStoredTimestamp	Datum och tid (tidstämpel) då meddelandet har sparats i Tullens system. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.	dateTime	J	[1..1]
MessageDownloadedTimestamp	Datum och tid (tidstämpel) då meddelandet har hämtats från Tullens system (om download-operationen har genomförts). Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.	dateTime	E	[0..1]
DeclarantBusinessId	Meddelandedeclarantens landskod och FO-nummer.	string	J	[1..1]

ContentFormat	Content-elementets dataformat. Eventuella värden är MIME-mediatyper. Eftersom kunderna endast kan förmedla XML-meddelanden till Tullen ska man som värde ställa 'application/xml'. Tidigare var det enda tillåtna värdet för elementet 'XML'. Det är fortfarande tillåtet att använda det. I DownloadResponse-svarsmeddelanden som innehåller en PDF-bilaga är ContentFormat inte ett obligatoriskt fält på SOAP-nivå.	string	J	[0..1]
---------------	---	--------	---	--------

Tabell 19: MessageInformation

8.5 AttachmentRequestMessage

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
AttachmentRequestMessage	Innehåller ett base64-kodat AttachmentRequest-element	base64Binary	J	[1..1]

Tabell 20: AttachmentRequestMessage

8.6 AttachmentResponseMessage

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
AttachmentResponseMessage	Meddelande som bildas av Tullens integrationslager och innehåller referenserna för den behandlade bilagan. Ingår som underelement i ett lyckat UploadAttachmentResponse-meddelande.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageStorageId	Meddelandets individuella referens med vilken bilagan har sparats i Tullens system.	string	J	[1..1]
ControlReference	En av kunden ursprungligen angiven kontrollreferens för sändningstillfället, med vilken kunden kan bunta ihop alla meddelanden som hör till transaktionen (avsändning av bilaga). Mera information ges i beskrivningen av ApplicationRequest-elementets underelement Reference.	string	J	1.1
RelatedMessageStorageId	Kontrollreferens som Tullen gett deklarationsfilen som bilagan hänför sig till, vilken referens finns på UploadResponse-meddelandet som hänför sig till det aktuella deklarationsmeddelandet.	string	J	[1..1]
Application	Kortnamn på Tullens program dit deklarationen till vilken bilagan hänför sig	string	J	[1..1]

	har skickats inbäddad i AttachmentContent-elementet.			
DeclarantBusinessId	Meddelandedeklarantens landskod och FO-nummer.	string	J	[1..1]

Tabell 21: AttachmentResponseMessage

8.7 DownloadMessageListFilteringCriteria

Elementets namn	Beskrivning			
DownloadMessageListFilteringCriteria	Uppgifterna i detta element används som filteringskriterier när man gör en begäran (downloadList) för att få veta vilka svarsmeddelanden som kan hämtas enskilt (download).			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
StartDate	Startdatum under eller efter vilket meddelandet har sparats. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.	date	J	[1..1]
EndDate	Slutdatum under eller före vilket meddelandet har sparats. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.	date	J	[1..1]
StartTimestamp	Tidstämpel för starttiden under eller efter vilken meddelandet har sparats. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.	dateTime	J	[1..1]
EndTimestamp	Tidstämpel för sluttiden under eller före vilken meddelandet har sparats. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid. Det är endast möjligt att använda antingen kombinationen StartDate och EndDate eller kombinationen StartTimestamp och EndTimestamp.	dateTime	J	[1..1]
MessageStatus	Begäran kan avgränsas med hjälp av meddelandets status. Följande statusar kan användas: - NEW – En lista över de svarsmeddelanden som ännu inte har hämtats (downloaded) returneras. - DLD – En lista över de svarsmeddelanden som redan har hämtats returneras. - ALL - En lista över alla svarsmeddelanden returneras	string	J	[1..1]

(båda sådana som hämtats och inte hämtats).

Tabell 22: DownloadMessageListFilteringCriteria

8.8 DownloadMessageFilteringCriteria

Elementets namn	Beskrivning			
DownloadMessageFilteringCriteria	Detta element behövs som nyckeldata när ett (1) (svars)meddelande hämtas från Tullens system.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageStorageId	Meddelandets unika ID med vilket ett svarsmeddelande kan hämtas från Tullens meddelandelager.	string	J	[1..1]
DocumentId	PDF-dokumentets unika identifikationsnummer	string	N	[0..1]

Tabell 23: DownloadMessageFilteringCriteria

8.9 ApplicationResponseMessage

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
ApplicationResponseMessage	Innehåller ett svar t.ex. till ett meddelande som har skickats med upload-operationen. Svaret är i XML-format. XML är base64-kodat.	base64Binary	J	[1..1]

Tabell 24: ApplicationResponseMessage

8.10 EchoContent

Elementets namn	Beskrivning			
EchoContent	Används för teständamål: inmatnings- och resultatuppgift för echo-operationen.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
Text	Fritt formulerad text.	string	J	[1..1]
Signature	XML-signatur. EchoContent har signerats.	string	E	[0..1]

Tabell 25: EchoContent

8.11 ApplicationRequest

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
ApplicationRequest	XML konstruerat av meddelandets byggare. Innehåller systemspecifika uppgifter (det egentliga applikationsmeddelandet och därtill hörande styruppgifter). XML-signerad med signaturen av meddelandets byggare.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageBuilderBusinessId	Landskod och FO-nummer för XML-meddelandets byggare.	string	J	[1..1]
MessageBuilderSoftwareInfo	Namn och versionsnummer på programmet som byggaren använder.	string	J	[1..1]
DeclarantBusinessId	Meddelandedeclarantens landskod och FO-nummer.	string	J	[1..1]
Timestamp	Tidstämpel för tidpunkten då meddelandet skapats. Datatypen är ISODateTime. Om tidszon inte har fastställts använder man inte som standard finsk lokaltid utan UTC-tiden som sådan.	dateTime	J	[1..1]
Application	Kort namn på Tullens system till vilket det egentliga applikationsmeddelandet som placerats i Content-elementet styrs.	string	J	[1..1]
Reference	En av kunden angiven kontrollreferens för sändningstillfället, med vilken kunden kan bunta ihop meddelandena som denne skickat samt Tullens svarsmeddelanden på dessa meddelanden. Kontrollreferensen för sändningstillfället är unik för varje system och meddelandedeclarant. Den består av följande delar: En av Tullen given fem tecken lång namnförkortning på företaget fogat till ett löpande nummer. Elementets minimilängd är 6 tecken och maximilängden är 14 tecken. Namnförkortningen bildas utifrån namnet på företaget som fungerar som meddelandedeclarant. Vanligtvis ges kontrollreferensen för sändningstillfället av XML-meddelandets byggare. Exempel: För ett företag som Tullen gett namnförkortningen FIRMA är den första kontrollreferensen för sändningstillfället FIRMA000000001	string	J	[1..1]
Environment	Tullens miljö där verksamheten äger rum. Möjliga värden: • PRODUCTION	string	J	[1..1]

ApplicationContent	• TEST Innehåller det egentliga applikationsmeddelandet som styrs till Tullens system. Se beskrivningen av elementet ApplicationContent.	Element	J	[1..1]
Signature	XML-signatur. Hela ApplicationRequest signeras. Tullen använder signaturen för att identifiera meddelandets byggare och säkerställa integriteten av data.	Signature	J	[0-1]

Tabell 26: ApplicationRequest

8.12 AttachmentRequest

Elementets namn	Beskrivning			
AttachmentRequest	XML konstruerat av meddelandets byggare. Innehåller data om den bifogade filen (den egentliga bilagan och därtill anknytande metadata). Elementet är XML-signerat (enveloped signature) och signeras av den som skapar meddelandet. I signaturen används ett av MDB beviljat servercertifikat.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
MessageBuilderBusinessId	Landskod och FO-nummer för XML-meddelandets byggare.	string	J	[1..1]
DeclarantBusinessId	Meddelandedeclarantens landskod och FO-nummer.	string	J	[1..1]
Application	Kortnamn på Tullens program som meddelandet hänför sig till.	string	J	[1..1]
AttachmentTitle	Namnet på den bifogade filen utan registersträngen.	string	J	[1..1]
AttachmentDescription	Fritext som anknyter till den bifogade filen	string	E	1.0
AttachmentDocumentType	Kod för bifogad handling, från Tullens dokumentkoder, till exempel N380 (faktura)	string	J	[1..1]
RelatedMessageStorageId	Kontrollreferens som Tullen gett deklarationsfilen som bilagan hänför sig till, vilken referens finns på UploadResponse-meddelandet som hänför sig till det aktuella deklarationsmeddelandet.	string	J	[1..1]
CustomsRequest	Har Tullen bett att få den bifogade filen?	boolean	J	[1..1]
AttachmentRelationIdentifier	Transaktionskod eller MRN-nummer.	string	J	[1..1]
LanguageCode	Språkkod för den bifogade filen. Tillåtna värden: fi, sv, en	string	J	[1..1]
RepresentativeBusinessId	Varuhavarens eller ombudets FO-nummer.	string	E	[0..1]
AttachmentContent	Innehåller den egentliga bilagan. Se beskrivning av elementets AttachmentContent.	Element	J	[1..1]
Signature	XML-signatur (Enveloped signature). Signerar hela AttachmentRequestMessage. Tullen	Signature	J	[1..1]

använder signaturen för att identifiera meddelandets byggare och säkerställa integriteten av data.

Tabell 27: AttachmentRequest

8.13 ApplicationContent

Elementets namn	Beskrivning			
ApplicationContent	Applikationsmeddelande med tilläggsuppgifter.			
Elementets namn	Beskrivning	Typ	Obliga- torisk (J/N)	Före- komst
Content	Innehåller ett base64-kodat applikationsmeddelande (t.ex. ett AREX- eller ELEX-meddelande).	string	J	[1..1]
ContentFormat	Content-elementets dataformat. Eventuella värden är MIME-mediatyper. Eftersom kunderna endast kan förmedla XML-meddelanden till Tullen ska man som värde ställa 'application/xml'. Tidigare var det enda tillåtna värdet för elementet 'XML'. Det är fortfarande tillåtet att använda det. I svarsmeddelanden som innehåller en ZIP-bilaga är fältets värde "application/zip".	string	J	[1..1]

Tabell 28: ApplicationContent

8.14 AttachmentContent

Elementets namn	Beskrivning			
AttachmentContent	Applikationsmeddelande med tilläggsuppgifter.			
Elementets namn	Beskrivning	Typ	Obliga- torisk (J/N)	Före- komst
Content	Innehåller den bifogade filen base64-kodad	string	J	[1..1]
ContentFormat	Content-elementets dataformat. Eventuella värden är MIME-mediatyper. Tillåtna värden " application/pdf", " image/jpeg", " image/png" och image/tiff".	string	J	[1..1]

Tabell 29: AttachmentContent

8.15 ApplicationResponse

Elementets namn	Beskrivning
ApplicationResponse	Tullens svarsmeddelande. Detta är vanligtvis svaret på ett meddelande som Upload-operationen förmedlar. Innehåller systemspecifika uppgifter (det

egentliga applikationsmeddelandet och därtill hörande styruppgifter). XML-signerad med signaturen av meddelandets byggare.

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
DeclarantBusinessId	Meddelandedeklarantens landskod och FO-nummer.	string	J	[1..1]
Timestamp	Tidstämpel för tidpunkten då meddelandet skapats. Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid som standard.	dateTime	J	[1..1]
Application	Kort namn på Tullens system i vilket det egentliga applikationsmeddelandet som placerats i elementet ApplicationResponseContent har sitt ursprung.	string	J	[1..1]
ControlReference	Ursprungligen kundens kontrollreferens för sändningstillfället. Kontrollreferensen är unik för varje system och meddelandedeklarant. Med hjälp av kontrollreferensen kan kunden bunta ihop meddelandena som denne skickat samt Tullens svarsmeddelanden på dessa meddelanden. Mera information ges i beskrivningen av ApplicationRequest-elementets underelement Reference.	string	E	[0..1]
MessageStorageId	Meddelandets unika ID.	string	J	[1..1]
ApplicationResponseContent	Innehåller det egentliga svarsmeddelandet som skapats av Tullens system. Se beskrivningen av elementet ApplicationContent.	Element	E	[0-1]
AttachOfApplicationResponseContent	Bilaga. Ett meddelande som Tullen skickar till kunden kan innehålla ett ZIP-arkiv som bilaga. Tullen anger då 'application/zip' som värde för elementet ContentFormat. Se beskrivningen av elementet ApplicationResponseContent.	Element	E	[0..1]
Signature	XML-signatur. Används endast vid separat överenskomna fall.	Signature	E	[0..1]

Tabell 30: ApplicationResponse

8.16 NotifyRequest

Elementets namn	Beskrivning	Typ	Obliga- torisk (J/N)	Före- komst
NotifyRequestHeader	Detta element utgör rubriken för varje meddelandenotifikationstjänstbegäran som skickas från WS-gränssnittet. Denna rubrik byggs av Tullen.			
Elementets namn	Beskrivning	Typ	Obliga- torisk (J/N)	Före- komst
IntermediaryBusinessId	Förmedlarens landskod och FO-nummer.	string	J	[1..1]
Timestamp	Tid och datum då begäran skickats.	dateTime	J	[1..1]

	Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid.		
TransactionId	Unikt ID för begäran.	J	[1..1]
Environment	Har notifikationen skickats från test- eller produktionsmiljön.	J	[1..1]
	Möjliga värden: • PRODUCTION • TEST		
MessageInformation	(se 7.4)		

Tabell 31: NotifyRequest

8.17 Notifyresponse

Elementets namn	Beskrivning			
NotifyResponseHeader	Detta element utgör rubriken för varje meddelandenotifikationstjänstbegäran som skickas från WS-gränssnittet. Denna rubrik byggs av kunden.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
IntermediaryBusinessId	Förmedlarens landskod och FO-nummer.	string	J	[1..1]
Timestamp	Tid och datum då begäran skickats.	dateTime	J	[1..1]
	Datatypen är ISODateTime. Om tidszon inte har fastställts används finsk lokaltid som standard.			

Tabell 32: NotifyRespondsHeader

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
ResponseCode	Returkod som anger om åtgärderna som begäran lett till har lyckats eller misslyckats. Kunden ska använda följande värden: 000 = ok 899 = felsituation Kunden kan också beskriva felsituationen mer detaljerat med de värden som beskrivs i bilaga 2.	string	J	[1..1]
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
ResponseText	Verbal beskrivning av returkoden (på engelska).	string	J	[1..1]

Tabell 33: RespondsCode

8.1 DocumentInformation

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
DocumentInformation	Detta valfria element finns i MessageInformation-blocket som skickas av Tullen, om ett PDF-dokument som hänför sig till ett XML-beslut beskrivs i blocket. Denna rubrik byggs av Tullen.			
Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
DocumentId	PDF-dokumentets unika identifikationsnummer	string	J	[1..1]
RelatedMessageStorageId	Unikt identifikationsnummer i Tullens system för det XML-dokument som PDF-dokumentet grundar sig på	string	J	[1..1]

Tabell 34: DocumentInformation

9 Autentisering och auktorisation av tekniska aktörer

I guiden *Introduktion till meddelandetrafik med Tullen* beskrivs de roller som är centrala med tanke på direkt meddelandedeklarering: meddelandedeklarant, byggare och förmedlare. Rollerna hänvisar till aktörerna i rent teknisk bemärkelse.

SOAP-meddelandet innehåller uppgifter om tekniska aktörer och deras roller i följande XML-element:

Elementets namn	Beskrivning	Typ	Obligatorisk (J/N)	Förekomst
DeclarantBusinessId	Meddelandedeklarantens landskod och FO-nummer.	string	J	[1..1]
MessageBuilderBusinessId	Landskod och FO-nummer för XML-meddelandets byggare.	string	J	[1..1]
IntermediaryBusinessId	Förmedlarens landskod och FO-nummer.	string	J	1.1

Tabell 35: Tekniska aktörer

När meddelandet handläggs vid Tullen utsätts det för datasäkerhetskontroller. Datasäkerhetskontrollerna som hänför sig till dataöverföring och datasäkerhetskontrollerna på affärssystemnivå är separerade från varandra. Som en följd av detta är det möjligt för företagskunder att använda en tjänsteleverantör som meddelandets byggare och förmedlare. Meddelandedeklaranten ska då auktorisera tjänsteleverantören att fungera i dessa roller. Auktorisationsuppgifterna insamlas från tillståndsansökningarna för kunder som använder direkt meddelandedeklarering. Tjänsteleverantören kan skicka och hämta meddelandedeklarantens meddelanden endast om meddelandedeklaranten har auktoriserat denne.

9.1 Autentisering av meddelandedeklarant

Meddelandedeklaranten autentiseras med hjälp av landskoden och FO-numret i XML-elementet DeclarantBusinessId. Tullens förfarande kan förutsätta att EORI-numret används som värde för XML-elementet DeclarantBusinessId.

Meddelandedeklaranten kan inte autentiseras med servercertifikatet som ingår i XML-signaturen, eftersom meddelandedeklaranten inte nödvändigtvis bygger meddelandena själv.

9.2 Autentisering av byggare/förmedlare

Meddelandets byggare / förmedlare behöver ett servercertifikat både för HTTPS-förbindelsen och XML-signaturen. Tullens meddelandegränssnitt autentiserar aktören genom att jämföra servercertifikatets och meddelandets uppgifter. I samband med upprättandet av förbindelsen granskar Tullen också att certifikatet inte finns på spärrlistan över servercertifikat.

9.3 Autentisering av meddelandenotifikationstjänstens leverantör

I sin tillståndsansökan anger meddelandedeklaranten en URL-adress dit Tullen skickar meddelandenotifikationerna. Tullen varken kontrollerar eller tar ställning till vem som innehar URL-adressen, förutsatt att tjänsten som där erbjuds uppvisar ett av Tullen godkänt servercertifikat i det skede då förbindelsen upprättas. Det ligger fullkomligt på meddelandedeklarantens ansvar att rätt URL-adress har angetts i tillståndsansökan. Om meddelandenotifikationerna mottas av samma aktör som bygger och förmedlar meddelandena så räcker det med ett certifikat.

10 Servercertifikat

Kundföretag behöver ett certifikat för direkt meddelandedeklarering.

För att få använda direkt meddelandedeklarering med Tullen måste kundföretaget anskaffa ett servercertifikat från en certifikatleverantör som godkänts av Tullen. Tullen godtar endast servercertifikat utfärdade av Myndigheten för Digitalisering och Befolkningsdata (MDB, på finska DVV). Attributet "SerialNumber" under fältet "Subject" i servercertifikatet måste fyllas i med kundföretagets EU MOMS-nummer.

Servercertifikatet är avgiftsbelagt. Tullen återförsäljer inte certifikat som utfärdats av MDB. Kunden köper certifikatet direkt från Myndigheten för Digitalisering och Befolkningsdata.

Den aktör som bygger och förmedlar meddelandena till Tullens meddelandegränssnitt ansöker om servercertifikatet. Om meddelandedeklaranten gör båda två själv, ansöker man om servercertifikatet till meddelandedeklaranten. Om man anlitar en tjänsteleverantör som bygger och förmedlar meddelandena på meddelandedeklarantens vägnar så ansöker tjänsteleverantören om servercertifikatet.

Samma certifikat kan användas i test- och produktionsmiljön.

Olika undantagsfall. Se FAQ.

Ett kundföretag behöver MDB:s "certifikatutfärdarens certifikat" när kundens program kontrollerar riktigheten hos Tullens servercertifikat.



Bild 27: Förtroendekedjan för certifikat

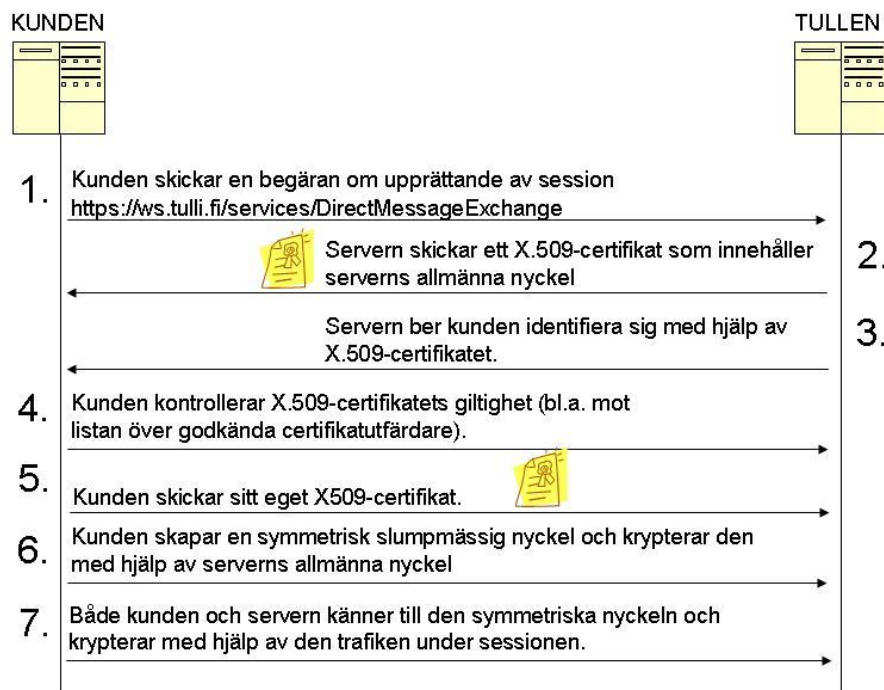


Bild 28: Upprättande av en HTTPS-förbindelse med ett certifikat (förenklat).

Förtroendekedjan i föregående bild gäller faserna 2 och 4.

10.1 Anskaffning och ibruktagande av servercertifikat

Anskaffningen och ibruktagandet av MDB:s servercertifikat består av tre faser:

1. Företaget ska beställa certifikatet via MDB:s e-tjänst (<https://asiointi.dvv.fi>). På MDB:s webbplats finns närmare information om hur man ska gå tillväga.
2. MDB skickar företaget servercertifikatet.
3. Företaget ska installera servercertifikatet på sin server så att det är tillgängligt för företagets egna program.

10.2 Förnyande av servercertifikat

Det servercertifikat som behövs vid direkt meddelandedeklarering och som beviljas av Myndigheten för Digitalisering och Befolkningsdata är i kraft i ett år i sänder. Man kan inte skicka meddelanden till Tullen med ett certifikat som inte är i kraft.

MDB skickar en påminnelse till Tullens kunder när certifikatet håller på att gå ut, men kunden är själv ansvarig för förnyandet av certifikatet.

Företaget behöver inte meddela Tullen om att certifikatet förnyats.

Tullen måste också förnya sitt certifikat med två års mellanrum. Kunderna meddelas inte separat om detta, eftersom utgångspunkten är att kunden kontrollerar Tullens certifikat mot listan över godkända certifikatutfärdare. Om kunden kontrollerar certifikatet statiskt mot Tullens certifikat som sparats i kundens system, kan meddelandedeklareringen avbrytas när Tullen byter sitt certifikat, om inte ändringen görs samtidigt i kundsystemen. Ansvaret för att iakttä när tidpunkten för ändringen inträffar ligger i detta fall hos kunden. Giltighetstiden för Tullens certifikat kan vid behov kontrolleras i MDB:s certifikatkatalog.

11 Sändningsuppgifter för kundmeddelanden

11.1 Tjänsteleverantörs ID

Vid direkt meddelandedeklarering används landskoden FI och FO-numret (har exempelvis formen FI1234567-8) som ID för meddelandedeklaranten och en eventuell tjänsteleverantör. För utländska kunder används MOMS-numret som ID.

11.2 Tjänstens URL-adress

SOAP-meddelandet skickas till följande URL-adresser:

Testmiljö:

<https://ws-customertest.tulli.fi/services/DirectMessageExchange>

Produktion:

<https://ws.tulli.fi/services/DirectMessageExchange>

Kundens meddelandenotifikationstjänst kontaktas från Tullens adress 157.129.127.247 eller **157.129.61.9**. Tullen rekommenderar att kunderna i sin egen meddelandenotifikationstjänst använder en standard https-port, men det är också möjligt att använda andra portar.

11.3 Kontrollreferens för sändningstillfället

Kontrollreferensen för sändningstillfället börjar med en fem tecken lång namnförkortning på företaget. Namnförkortningen, som tilldelas av Tullen, bildas utifrån namnet på företaget som fungerar som meddelandedeklarant. Slutdelen av kontrollreferensen för sändningstillfället väljs av kunden.

Elementets maximilängd är 14 tecken och i slutdelen kan man därmed använda ett upp till 9 tecken långt löpande nummer.

Kontrollreferensen för sändningstillfället ska vara unikt för varje kombination av en meddelandedeclarant och ett specifikt system hos Tullen. Detta betyder att företaget kan använda en (1) kontrollreferens för sändningstillfället per ett av Tullens system (t.ex. AREX).

Meddelanden som kunden skickar till Tullen avvisas om:

- Värdet för elementet Reference i XML-dokumentet ApplicationRequest och applikationsmeddelandets kontrollreferens för sändningstillfället inte är de samma.
- Samma kontrollreferens för sändningstillfället har redan tidigare mottagits för samma kombination av meddelandedeclarant och specifikt system hos Tullen.
- Tullen sparar kontrollreferensen för sändningstillfället omedelbart efter att Upload-begäran mottagits. Kontrollreferensen för sändningstillfället "frigörs" inte om ett fel upptäcks och Tullen avvisar begäran. Detta gäller också för situationer där avvisandet av Upload-begäran beror på ett tillfälligt tekniskt problem i Tullens system.

Tullens produktionsmiljö och testmiljö är helt åtskilda från varandra. Kundtestningen "förbrukar" inte produktionsmiljöns kontrollreferenser för sändningstillfället.

Det är kundens skyldighet att se till att kontrollreferenserna för sändningstillfället förblir unika.

11.4 Sändningsuppgifter i kundmeddelanden

11.4.1 RequestHeader

Innehållet i XML-elementet 'IntermediaryBusinessID' som ingår i SOAP-meddelandets RequestHeader består av meddelandedeclarantens eller tjänsteleverantörens ID.

Exempel på XML-elementet RequestHeader som ingår i SOAP-meddelandet:

```
<cst:RequestHeader
xmlns:cst="http://tulli.fi/ws/corporateservicetypes/v1">
  <cst:IntermediaryBusinessId>FI2340001-5</cst:IntermediaryBusinessId>
  <cst:Timestamp>2010-03-19T09:15:02.765Z</cst:Timestamp>
  <cst:Language>EN</cst:Language>
  <cst:IntermediarySoftwareInfo>Examples 1.5</cst:IntermediarySoftwareInfo>
</cst:RequestHeader>
```

11.4.2 ApplicationRequest

SOAP-meddelandet innehåller XML-dokumentet ApplicationRequest. Dess XML-element har följande betydelser:

- '**DeclarantBusinessID**' innehåller meddelandedeclarantens ID.
- '**MessageBuilderBusinessID**' innehåller antingen meddelandedeclarantens ID eller tjänsteleverantörens ID, om deklaranten använder en tjänsteleverantör.
- '**Application**' innehåller det korta namnet på Tullens system. Möjliga namn är 'AREX', 'CWAR', 'IMP', 'INSTAT', 'GUARANTEE', 'TRA', 'TSD', och 'EXP'.
- '**Environment**', som värde för elementet ställs
 - i testsituationer 'TEST'
 - vid produktion 'PRODUCTION'.
- '**Reference**' innehåller kontrollreferensen för sändningstillfället.

Exempel på parametrar som ingår i XML-dokumentet ApplicationRequest:

```
<req:ApplicationRequest
```

```
xmlns:req="http://tulli.fi/schema/corporateservice/appl/v1">
<req:MessageBuilderBusinessId>FI2340001-5</req:MessageBuilderBusinessId>
<req:MessageBuilderSoftwareInfo>Examples 1.5</req:MessageBuilderSoftwareInfo>
<req:DeclarantBusinessId>FI2340001-5</req:DeclarantBusinessId>
<req:Timestamp>2010-03-19T11:15:05.234+02:00</req:Timestamp>
<req:Application>AREX</req:Application>
<req:Reference>FIRMA000004671</req:Reference>
<req:Environment>TEST</req:Environment>
```

Ett applikationsmeddelande som överförs som XML-datainnehåll kan också innehålla kontrollreferensen för sändningstillfället. I dylika fall ska man i XML-elementet 'Reference' som ingår i XML-dokumentet ApplicationRequest ange samma kontrollreferens för sändningstillfället som i applikationsmeddelandet.

Exempelvis innehåller AREX-meddelandets Message-block följande XML-element:
<wco:reference>FIRMA000004671</wco:reference>

XML-elementet 'Reference' som ingår i ApplicationRequest innehåller
<req:Reference>FIRMA000004671</req:Reference>

11.4.3 Applikationsmeddelande

XML-elementet Content som ingår i XML-dokumentet ApplicationRequest innehåller ett XML-applikationsmeddelande. I XML-applikationsmeddelandets element 'recipient' används som Tullens ID landskoden FI och Tullens FO-nummer FI0245442-8.

Det lönar sig att kontrollera de systemspecifika sändningsuppgifterna i meddelandebeskrivningarna för ifrågavarande system. Meddelandebeskrivningarna finns på Tullens webbplats under:

<http://www.tulli.fi/fi/yrityksille/sahkoinenasiointi/edi/Sanomakuvaukset/index.jsp>

12 XML-signatur

Vid den direkta meddelandedeklareringen mellan Tullen och kundföretagen används XML-signaturer för att autentisera applikationsmeddelanden och för att säkerställa deras integritet och oavvislighet. XML-signaturerna innehåller ett av kundföretaget anskaffat certifikat.

Systemet som skapar applikationsmeddelandet signerar meddelandet med en XML-signatur.

Alla kundmeddelanden som vid direkt meddelandedeklarering skickas till Tullen med ett Upload-begärandemeddelande signeras med en XML-signatur.

Begärandemeddelandena DownloadList och Download innehåller varken ett kundmeddelande eller en XML-signatur. CheckConnectivity-begärandemeddelandet, som används för att testa förbindelsen, kan signeras med en XML-signatur.

12.1 XML-signaturens struktur

Enklast får man en överblick av XML-signaturens struktur genom att bekanta sig med de exempelmeddelanden som finns i distributionspaketet "Direkt meddelandedeklarering, exempel på meddelanden" på Tullens webbplats:

http://www.tulli.fi/fi/yrityksille/sahkoinenasiointi/edi/suora_sanoma_asiointi/index.jsp

Allmän information om rekommendationer kring XML-signatur finns på
<http://www.w3.org/2002/02/xmlsignature-pressrelease>

12.2 Implementering av XML-signatur

För XML-signaturen ska kunden ha en keystore som innehåller certifikatet som MDB beviljat kunden.

XML-signaturen skapas alltid med programverktyg. Det finns gott om dylika tredjepartsimplementeringar, t.ex. av Java och Microsoft.NET.

12.3 Validering av XML-signatur samt eventuella problem

Kunden ska validera XML-signaturen.

Kundföretaget kan också i samband med den tekniska ws-testningen (innan den egentliga kundtestningen) kontrollera att XML-signaturen fungerar genom att skicka ett signerat CheckConnectivity-begärandedmeddelande till Tullen.

Om signaturen inte valideras är det möjligt att den går sönder vid XML-handläggningen som sker i kundens system efter signerandet. Därför lönar det sig att base64-koda ApplicationRequest genast efter signerandet.

13 Begränsningar vid direkt meddelandedeklarering

Direkt meddelandedeklarering ställer vissa begränsningar på kundsystemen. Syftet med dessa begränsningar är att garantera att tjänsten är tillgänglig. Begränsningarna beskrivs i detta avsnitt.

13.1 Protokollversioner

Gränssnittet stöder följande protokollversioner:

- SOAP 1.1
- SOAP 1.2 (krävs om företaget vill använda meddelandenotifikationstjänsten)
- HTTP 1.1
- TLS version 1.2

13.2 Krypteringsalgoritmer

Tillåtna cipher suites i HTTPS-kontakten:

- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA

I XML-signaturens SignatureMethod är den tillåtna algoritmen:

- RSAwithSHA256

I XML-signaturens DigestMethod är den tillåtna algoritmen:

- SHA256

13.3 Begränsningar gällande meddelanden

För inkommande meddelanden har följande begränsningar gällande HTTP-trafiken fastställts:

- Endast begäranden som skickas med http POST-metoden är tillåtna
- Längden på URL:en får i sin helhet vara högst 4 kilobyte

- Den sammanlagda längden på meddelandets header-uppgifter är max 64 kilobyte
- Det maximala antalet header-element är 128 st.
- Längden på header-elementets namn får vara högst 256 bytes
- Värdet på header-elementet får vara högst 32 kilobytes
- Längden på URL:ens parameterdel får vara högst 3 kilobytes

För XML-meddelandena har följande gränsvärden gällande deras struktur ställts:

- Maximistorleken för XML-applikationsmeddelandet (XML-elementet Content i ApplicationRequest) är 512 kilobyte. Storleksbegränsningen gäller applikationsmeddelandet före det base64-kodats.
- Det maximala antalet nästade XML-elementnivåer är 128.
- Det maximala antalet attribut för ett XML-element är 64.
- Maximistorleken på bilagan i ett UploadAttachment-meddelande är 5 megabyte. Storleksbegränsningen gäller den bifogade filen före base64-kodning.
- Meddelandet får inte innehålla förbjudna teckensträngar som kan tolkas som ett eventuellt SQL-injektionsangrepp.

13.4 Begränsningar gällande antalet begäranden

Handläggningen av servicebegäranden har begränsats enligt följande per kund kopplad till tjänsten (IntermediaryBusinessID):

- **Upload:** Tjänsten handlägger med en sekunds mellanrum högst tre Upload-begäranden.
- **UploadAttachment:** Tjänsten handlägger med en sekunds mellanrum högst tre UploadAttachment-begäranden.
- **DownloadList:** Tjänsten handlägger med fem minuters mellanrum högst en DownloadList-begäran.
- **Download:** Tjänsten handlägger med en sekunds mellanrum högst fem Download-begäranden.
- **CheckConnectivity:** Tjänsten handlägger med en sekunds mellanrum högst en CheckConnectivity-begäran.

När kunden har tagit i bruk meddelandenotifikationstjänsten kan denne inte göra DownloadList-begäranden lika frekvent, utan högst med en timmes mellanrum.

Därtill har samtliga kunders gemensamma meddelandetrafikmängd begränsats enligt följande:

- **Upload:** Tjänsten handlägger högst 20 tulldeklARATIONER per sekund.
- **UploadAttachment:** Tjänsten handlägger högst 20 sändningar av bifogade filer per sekund.
- **DownloadList:** Tjänsten handlägger högst 10 begäranden om förteckningar över beslut som är redo att hämtas per sekund.
- **Download:** Tjänsten handlägger högst 100 hämtningar av förtullningsbeslut per sekund.

Tjänsten avvisar DownloadList- och CheckConnectivity-begäranden som överskrider de ovan beskrivna gränserna. Upload-, UploadAttachment- och Download-begäranden avvisas däremot inte omedelbart, utan de "buffras" när kunden har skickat mer begäranden än tillåtet. SOAP-begäran, som Tullen mottagit genom TCP-sessionen, ställs då i kö för att vänta på handläggning. En i kö ställd Upload- eller Download-begäran avvisas endast om man inte hinner handlägga den innan https-förbindelsen bryts på grund av överskriden tidsgräns.

Handläggningen av begäranden har begränsats enligt följande:

- **Upload:** Tjänsten handlägger med en sekunds mellanrum högst en (1) Upload-begäran per kund (IntermediaryBusinessID) kopplad till tjänsten.
- **UploadAttachment:** Tjänsten handlägger med en sekunds mellanrum högst en (1) UploadAttachment-begäran per kund (IntermediaryBusinessID) som är kopplad till tjänsten.
- **DownloadList:** Tjänsten handlägger med fem minuters mellanrum högst en (1) DownloadList-begäran per kund (IntermediaryBusinessID) kopplad till tjänsten.

- Download: Tjänsten handlägger med en sekunds mellanrum högst fem (5) Download-begäranden per kund (IntermediaryBusinessID) kopplad till tjänsten.
- CheckConnectivity: Tjänsten handlägger med en sekunds mellanrum högst en (1) CheckConnectivity-begäran per kund (IntermediaryBusinessID) kopplad till tjänsten.

Tjänsten avvisar DownloadList- och CheckConnectivity-begäranden som överskrider de ovan beskrivna gränserna.

Upload- och Download-begäranden behandlas något mildare: När kunden når gränserna, "buffras" nya begäranden från kunden. Detta innebär att TCP-sessionen genom vilken Tullen mottagit den nya SOAP-begäran ställs i kö för att vänta. Begäran plockas ur kön genom att fortsätta handläggningen. En Upload- eller Download-begäran som ställts i kö avslutas endast om det får vänta så länge där att https-kontaktens timeout-värde nås.

När kunden har tagit i bruk meddelandenotifikationstjänsten förutsätts det att DownloadList-begäranden inte görs lika frekvent, dvs. högst med en timmes mellanrum eller mer sällan.

13.5 Tidsövervakning av begäranden

För SOAP-begärandena från kundens program som kontakter Tullens gränssnitt för direkt meddelandedeklarering ska tidsövervakningen vara ställd på minst 120 sekunder. Normalt returnerar Tullens gränssnitt ett svar inom några sekunder, men när tjänsten är hårt belastad kan svaret dröja betydligt längre. Om kundprogrammet bryter SOAP-förbindelsen för fort kan Tullens gränssnitt inte returnera ett svarsmeddelande till kunden, vilket leder till att returkoden inte returneras till kundens system. Då undviker man också att s.k. dubbelmeddelanden uppstår i Tullens system.

För programmet som skickar Tullens meddelandenotifikationer till kunden är tidsövervakningen också ställd på 120 sekunder. Om kundens meddelandenotifikationstjänst inte svarar inom den för tidsövervakningen ställda tiden så försöker Tullens system inte skicka meddelandenotifikationen på nytt, utan sändningsförsöket statistikförs i Tullens system som ett fel.

13.6 Tillgång på Tullens beslut och övriga dokument

Tullen förvarar sina beslut angående tulldeklarationer som kunderna skickat och övriga därtill anknytande dokument (svarsmeddelanden), så att kundens program kan hämta dem i ett års tid från att tulldeklarationen tagits emot. När det gäller långvariga förtullningar kan tiden för att hämta vissa svarsmeddelanden vara kortare än ett år men dock minst tre månader från att svarsmeddelandet skapats.

Bilaga 1: ResponseCode och ResponseText

Kod	Text	Förklaring
Meddelandeskickandet lyckas		
000	OK	Operationen har genomförts på ett lyckat sätt.
Meddelandeskickandet misslyckas: Auktorisationsfel, kontakta Tullens EDI-stöd		
460	Intermediary id not valid	Längden på SOAP-begärandemeddelandets XML-element IntermediaryBusinessId är inte 9-17 tecken, eller så innehåller IntermediaryBusinessId en annan aktörsuppgift än den som ingår i servercertifikatet med vilken HTTPS-klienten autentiserar sig i tjänsten.
461	Intermediary authorization failed	Meddelandets förmedlare har inte auktoriserats av någon deklarant.
465	Declarant authorization failed	Deklaranten har inte användarrättigheter till ifrågavarande system (innehållet i ApplicationRequest-dokumentets XML-element Application).
466	Builder authorization failed	Meddelandets byggare har inte användarrättigheter till ifrågavarande system (innehållet i ApplicationRequest-dokumentets XML-element Application), eller så är byggaren inte också meddelandets deklarant eller förmedlare.
467	Intermediary authorization failed	Meddelandets förmedlare har inte auktoriserats av meddelandedeklaranten.
Meddelandeskickandet misslyckades: Fel i meddelandet, korrigera meddelandet och skicka det på nytt		
450	Invalid HTTP connection parameters	De gränsvärden som definierats för http-förbindelsen har överskridits.
451	Schema validation error in SOAP request	Ett fel har inträffat vid WSDL- eller schemavalideringen av SOAP-begärandemeddelandet.
452	Schema validation error in ApplicationRequest	Ett fel har inträffat vid schemavalideringen av ApplicationRequest-dokumentet.
453	Wrong target environment for DownloadRequest	Ett applikationsmeddelande avsett för testmiljön har försökt hämtas från produktionsmiljön
455	Rejected by policy	Generisk felkod med vilken det hänvisas till ett fel som inträffat vid kontrollen av en inkommande servicebegäran. Returneras när en mer specifik felkod inte har definierats.

Kod	Text	Förklaring
456	Rejected by filter	Generisk felkod som returneras när filterfunktionen som ingår i kontrollerna avvisar servicebegäran.
458	ApplicationRequest with duplicate reference received	ApplicationRequest-dokumentets kontrollreferens för sändningstillfället (XML-elementet Reference) har redan använts tidigare.
459	Encountered character not allowed by XML encoding	Servicebegäran innehåller en teckenkod som inte är i enlighet med den kodningspraxis som tillämpas för teckenkodningen.
463	Builder id not valid	Längden på XML-elementet MessageBuilderBusinessId är inte 9-17 tecken.
464	Declarant id not valid	Längden på XML-elementet DeclarantBusinessId är inte 9-17 tecken.
468	Application request environment not valid	Ett ApplicationRequest-dokument som är avsett för produktionsmiljön har skickats till testmiljön, eller ett ApplicationRequest-dokument som är avsett för testmiljön har skickats till produktionsmiljön.
469	Content format not XML	ApplicationRequest-dokumentets XML-element ContentFormat har inte 'application/xml' eller 'XML' som sitt värde.
470	ApplicationRequestMessage validation failed	XML-valideringen av ApplicationRequest-dokumentet misslyckades.
471	Content validation failed	XML-valideringen av applikationsmeddelandet (innehållet i ApplicationRequest-dokumentets XML-element Content) misslyckades.
472	Invalid application specified	DownloadList-begärandeddelandets XML-element Application innehåller en applikationsidentifierare som inte hör till gruppen tillåtna applikationsidentifierare.
473	Content exceeds size limit for application	Det förmedlade applikationsmeddelandet (ApplicationRequest-dokumentets XML-element Content) överskrider den systemspecifika storleksbegränsningen.
476	XML signature not valid	XML-signaturen är inte giltig. Orsaken till felet kan t.ex. vara att det sammandrag som sparats i XML-signaturen avviker från det sammandrag som beräknats i samband med kontrollen av signaturen.
477	SignatureMethod algorithm in signature not allowed	I XML-signaturen har använts en signaturalgoritm som inte hör till gruppen tillåtna algoritmer.

Kod	Text	Förklaring
478	DigestMethod algorithm in signature not allowed	I XML-signaturen har använts en sammandragsalgoritm som inte hör till gruppen tillåtna sammandragsalgoritmer.
479	Reference URI in signature invalid	I XML-signaturen som ingår i ApplicationRequest-dokumentet har använts ett Reference-element vars värde inte är tomt (""). För inbäddade XML-signaturer ska Reference-elementets värde vara tomt.
480	SOAP request exceeds size limit	SOAP-begärandemeddelandets storlek överskrider storleksbegränsningen.
482	Invalid RelatedMessageStorageId in AttachmentRequest	Den angivna referensen för meddelandet har fel format.
500	ApplicationRequest with duplicate reference received	Meddelandets byggare får endast avge ett ApplicationRequest-dokument med en viss kontrollreferens för sändningstillfället (värdet för XML-elementet Reference). Alla efterföljande ApplicationRequest-dokument som innehåller samma värde för XML-elementet Reference avvisas som duplikat.
501	Reference values in ApplicationRequest and Content do not match.	Värdet för XML-elementet Reference i ApplicationRequest-dokumentet motsvarar inte kontrollreferensen för sändningstillfället i applikationsmeddelandet (Content).
502	DeclarantBusinessId in ApplicationRequest and sender in content do not match.	Värdet för XML-elementet DeclarantBusinessId i ApplicationRequest-dokumentet motsvarar inte avsändarens ID i applikationsmeddelandet (Content).
503	Referenced declaration not found	Någon deklaration med det värde för RelatedMessageStorageId-elementet som anges i meddelandet hittas inte.
504	Identical attachment for the referenced declaration already exists	Samma bilaga har redan tidigare skickats till deklarationen som anges i meddelandet (RelatedMessageStorageId).
505	Referenced declaration not accepted	Handläggningen av den ursprungliga deklarationen som anges i meddelandet (RelatedMessageStorageId) har avslutats med avslag i Tullens system.
506	Referenced declaration and function do not match	Den ursprungliga deklarationen enligt meddelandet (RelatedMessageStorageId) och den angivna applikationskoden (Application) överensstämmer inte.
600	Start time too far away in the past.	Begäran om meddelandelista omfattar för gamla meddelanden.

Kod	Text	Förklaring
601	Start time greater than end time	Sökkriteriernas starttid är efter sluttiden.
700	Invalid request	Meddelandet hittas inte eller så har kunden inte rätt att hämta meddelandet.
Meddelandeskickandet misslyckades: Störning i datakommunikationen eller systemstörning, skicka meddelandet på nytt om en liten stund		
457	Allowed message frequency exceeded	Gränsvärdet som definierats för frekvensen för servicebegäranden har överskridits.
474	Uploads to application temporarily disabled	För närvarande förmedlas inga nya applikationsmeddelanden till ifrågavarande system (ApplicationRequest-dokumentets XML-element Application). Skicka senare ett nytt Upload-begärandemeddelande.
490	Backend connection error	Förbindelseproblem i Tullens integrationslager för direkt meddelandedeklarering.
491	Backend connection error	Förbindelseproblem i Tullens integrationslager för direkt meddelandedeklarering.
492	Backend connection error	Förbindelseproblem i Tullens integrationslager för direkt meddelandedeklarering.
499	Unknown Error	Intern felsituation i Tullens integrationslager för direkt meddelandedeklarering för vilken ingen mer specifik felkod har definierats.
999	Unexpected Error	Tullens integrationslager för direkt meddelandedeklarering har stött på en felsituation för vilken ingen mer specifik felkod har definierats.
999	Backend technical error	Tullens integrationslager för direkt meddelandedeklarering har stött på en felsituation för vilken ingen mer specifik felkod har definierats.

Tabell 36: ResponseCode och ResponseText

Bilaga 2: Felkoder som används i NotifyResponse-meddelandet

Response Code	ResponseText	Beskrivning
Meddelandeförmedlingen lyckades		
000	OK	Lyckat notifikationsmeddelande.
Meddelandeförmedlingen resulterade i ett fel		
899	Unexpected Error	Felsituation för vilken ingen mer specifik felkod har definierats.
Optionella meddelandefel		
851	Schema validation error in SOAP request	Schemafel i notifikationsmeddelandet.
860	Intermediary id not valid	Förmedlarens ID är inte giltigt.
864	Declarant id not valid	Deklaranten hör inte till förmedlarens auktoriserare.
868	Environment not valid	Deklarationsmeddelandet hänvisar till fel miljö.
872	Invalid application specified	Systemet är oidentifierbart eller ur bruk.
841	Control reference not valid	Oidentifierad kontrollreferens.
842	MessageStorageId not valid	Oidentifierat meddelande-ID.

Tabell 37: Felkoder som används i NotifyResponse-meddelandet

Bilaga 3: Krav gällande Tullens XML-meddelanden

Allmänna krav på XML-meddelanden

De krav gällande XML-meddelanden som ställs här gäller både för applikationsmeddelanden och SOAP-begärandemeddelanden och –svarsmeddelanden som används för överföring av applikationsmeddelanden.

I Tullens XML-meddelanden används version 1.0 av XML-standarden. I meddelandena används teckenkoden Unicode och för den kodningspraxis UTF-8. Det är skäl att ange dessa egenskaper i XML-prologen före meddelandets rotelement. Nedan ges ett exempel på en prolog där XML-versionen som använts i XML-processinstruktionen anges med version-attributet och meddelandets teckenkodning med encoding-attributet.

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

Om prologen innehåller ett standalone-attribut som markerar oavhängighet från en extern strukturbeskrivning så ska dess värde vara "no". Attributet är inte nödvändigt eftersom standardstatusen för denna egenskap (standalone="no") motsvarar Tullens praxis att granska meddelandenas riktighet vad gäller den externa strukturbeskrivningen. I prologen ska inte ingå andra processinstruktioner än ovan nämnda XML-version och XML-processinstruktionen som anger teckenkodningen (eventuellt inklusive attributet standalone="no").

Om inledningen av XML-meddelandets prolog innehåller markören BOM (byte order mark) så ska den vara förenlig med ovannämnda encoding-attribut. Teckenkodningen UTF-8 anges med en BOM-markör vars trebytes hexadecimalvärde är EF BB BF. När testmeddelanden upprättas som textfiler bör man beakta att flera XML- och textbehandlingsverktyg lägger till BOM-markören till början av filerna, trots att verktygen inte visar markören. Verktygets inställningar borde kontrolleras, så att BOM-markören som inkluderas i filen blir UTF-8.

Element som inte innehåller data borde utelämnas från meddelandena. Till exempel i exportsystemet medför tomma element att meddelandet förkastas.

XML CDATA-block ska inte användas som elementvärden i meddelandena. Det finns f.ö. inget behov för dylikt i meddelanden i Tullens system.

Strukturen hos Tullens XML-meddelanden beskrivs med XML-scheman. XML-scheman publiceras för att underlätta handläggningen av XML-meddelanden. Kunden ska försäkra sig om att datainnehållet i meddelandena är korrekt. XML-meddelandets riktighet kan kontrolleras genom att validera det mot meddelandets schema.

I de XML-meddelandescheman som specificerats av Tullen har man implementerat namnrymder och krav på entydiga namn för elementen (elementFormDefault="qualified"). I de meddelanden som upprättas enligt dessa scheman används endast element som hör till namnrymder, och namnen på taggar med vilka elementen markeras ska tillfogas till elementets namnrymd. Att elementen hör till namnrymder garanterar att varje element kan entydigt identifieras och inte ger upphov till namnkonflikter.

Bilaga 4 innehåller mera information om användningen av namnrymder i XML-meddelanden.

Krav som anknyter till applikationsmeddelanden

Begränsningar i vissa fält gällande visningssätt samt godtagbara värden för applikationsmeddelandena finns beskrivna i systemspecifika applikationsbeskrivningar. Här presenteras allmänna krav som ställs på datainnehållet i meddelandena samt några allmänna systemspecifika ändringar som fastställts.

Teckenuppsättningen **Unicode** kan inte användas i sin helhet för att presentera data som ska behandlas med Tullens system. Unicode-standard definierar delmängder för teckenkodningen som baserar sig på språkområden. I alla system kan man använda de vanliga tecknen som finns i delmängderna 'BasicLatin' (hexadecimala kodvärden mellan x00 – x7F) samt 'Latin-1Supplement' (hexadecimala kodvärden mellan x00 – x7F), förutom i följande special- och undantagsfall som beskrivs nedan. I praktiken är delmängden BasicLatin den gamla teckenuppsättningen ASCII och innehåller som sådan flera kontrolltecken som inte bör användas i XML-meddelanden. Det exakta innehåll i Unicodes delmängder kan kontrolleras på <http://www.unicode.org/charts/index.html>.

I enlighet med delmängderna för Unicode har motsvarande delmängder definierats i W3C:s XML Schema-standardens bilaga som finns på adressen <http://www.w3.org/TR/xmlschema-2/#character-classes>. Under rubrikem 'F.1 Character Classes' i punkt 'Block Escape' finns en tabell med teckenkodningsdelmängder enligt språkområde. Även kontrolltecknen finns definierade i den ovannämnda Länken i tabellen under punkten 'Category Escape'. Den XML teckenkodskategori 'Cc' för kontrolltecken som där nämns innehåller kontrolltecknen som omfattar unicond-blocken 'C0' och 'C1', men tecken som lämnats utanför dem är: SP (blanksteg), NBSP (hårt blanksteg) och SHY (mjukt bindestreck).

Av Unicode kontrolltecknen i delmängden 'BasicLatin', dvs. tecken i 'C0'-området (hexadecimala kodvärden mellan x00 – x1F) kan man endast använda formateringstecknen HT (vågrät tabulator), LF (radmatning), VT (vertikal tabulator) FF (sidmatning) och CR (vagnretur). Av dessa tecken bör man undvika VT och FF, eftersom de i praktiken inte har någon betydelse i XML. Till delmängden hör dessutom specialtecknet DEL (gömmer det tidigare tecknet), som är av samma karaktär som kontrolltecknen. Fastän det inte hör till den egentliga gruppen av kontrolltecken 'C0', har det inte heller något egentligt syfte och ska inte användas.

Av de kontrolltecken som hör till Unicode delmängden 'Latin-1Supplement', dvs. tecknen i 'C1'-området (hexadecimala kodvärden mellan x80 – x9F) kan man endast använda formateringstecknet NEL, men tecknet har ingen betydelse i XML och därför ska även det tecknet undvikas. Till delmängden hör dessutom de kontrollteckenliknande specialtecknen NBSP och SHY. Fastän de inte hör till den egentliga gruppen kontrolltecken i 'C1', ska inte heller dessa specialtecken utan egentligt syfte i XML användas.

De ovannämnda tillåtna kontrolltecknen, specialtecknen av liknande karaktär samt blanksteget SP (x20) bildar en grupp formateringstecken som används allmänt i texten. Dessa tecken har i XML textinnehåll ett begränsat användningsområde som beskrivs nedan.

HT	'Horizontal Tabulation' (hexadecimalt kodvärde x09) är en vågrät tabulator som ofta också används i XML-meddelanden för indentering.
LF	'Line Feed' (hexadecimalt kodvärde x 0A) är radmatning som används i XML-meddelanden som huvudsakligt tecken för radbyte.
VT	'Vertical Tabulation' (hexadecimalt kodvärde x0B) är en vertikal tabulator som inte ska användas i XML-meddelanden.
FF	'Form Feed' (hexadecimalt kodvärde 0C) är en sidmatning som inte ska användas i XML-meddelanden.
CR	'Carriage Return' (hexadecimalt kodvärde x0D) är vagnretur, ett tecken som används i XML-meddelanden som ett radbytetecken.
SP	'Space' (hexadecimalt kodvärde x20) är ett blanksteg som används i XML-meddelanden för indenteringar.
DEL	'Delete' (hexadecimalt kodvärde x 7F) har fungerat som ett tecken som gömmer ett tidigare tecken. Tecknet ska inte användas i XML-meddelanden.
NEL	'Next Line' (hexadecimalt kodvärde x85) är ett radbyte som hör hemma i EBCDIC-teckenuppsättningen. Tecknet ska inte användas i XML-meddelanden.
NBSP	'Non-Breaking Space' (hexadecimalt kodvärde xA0) är ett tecken som används som avgränsningstecken i HTML-innehåll och förhindrar radbyte. Tecknet ska inte användas i XML-meddelanden.
SHY	'Soft Hyphen' (hexadecimalt kodvärde xAD) är ett radbrytande tecken eller ett mjukt bindestreck. Tecknet ska inte användas i XML-meddelanden.

En del av formateringstecknen, dvs. tecknen HT, LF, CR och SP bildar en 'White Space'- grupp, dvs. en teckenkategori 's' till vilken XML definierar specifika handläggningsregler (<http://www.w3.org/TR/2008/REC-xml-20081126/#sec-white-space>). Utöver användningen i XML-strukturens formatering kan dessa även användas i textinnehåll i XML, men av de värden som ska normaliseras ersätts de med ett blanksteg mitt i texten och försvinner från början och slutet av värdena. Som radbyte i XML duger tecknen LF samt CR och kombinationer av dessa, men i XML behandlingen förvandlas alla radbyten till LF-tecken.

XML-scheman för Applikationsmeddelanden

Datainnehåll i XML-meddelanden visas alltid som text i teckenformat och kallas också för meddelandets textinnehåll. Tolkningen av textinnehållet av värdet på strukturdelen - element och attribut, som förekommer i XML-meddelandet, styrs med datatyper som används i XML-scheman. Värdena tolkas numeriskt eller på ett motsvarande sätt när strukturdelen baserar sig på primitiva datatyper som beskriver ett numeriskt värde, ett sanningsvärde (booleskt), en tidpunkt eller ett tidsintervall. För dessa värden kan man naturligtvis inte tillåta nollvärden eftersom det inte finns någon betydelsefull tolkning för dem. Nollvärden kan till sin karaktär vara betydelsefulla för datatyper som ska tolkas som texter, men de ska användas sparsamt. När textvärdet i strukturdelen är noll, ska det inte tolkas som om det fattas utan som noll. Det enda som tolkas som om data fattas är om strukturdelen fattas ur XML-strukturen.

I systemens XML-scheman gällande direkt meddelandedeklarering finns ofta primitiva datatyper som används som bas för **datatyper som anknyter till tid**: date, dateTime och time. Formaterna har dock begränsats så att millisekunder eller delar som uttrycker tidszonen inte tillåts.

I systemens XML-scheman gällande direkt meddelandedeklarering accepteras endast ett begränsat antal tecken ur den omfattande teckenuppsättningen Unicode, när det är frågan om **datatyper av textnatur**. Med begränsningarna strävar man efter att styra användningen av kontroll- och specialtecken i meddelanden enligt de ovannämnda begränsningarna. I scheman finns några systemspecifika skillnader som inte finns med i följande systemspecifika beskrivningar.

Krav gällande applikationsmeddelanden i Intrastat-systemet

I gränssnittets scheman har man för all data i textformat skapat en datatypsbas kallad **BaseTextType**. Datatypen godkänner endast de tecken ur Unicode som hör till delmängderna **'BasicLatin'** eller **'Latin-1Supplement'** samt dessutom **euro-symbolen**. Denna BaseTextType-datatyp är endast en bas ur vilken man härlett de egentliga datatyperna för textdata, som förekommer i meddelandena, genom att begränsa teckenurvalet för att få det ännu mindre. Detta har gjorts genom att härleda fyra datatyper för data av olika karaktär: TextType, LineType, CodeType och IdentifierType. Dessa finns noggrannare beskrivna nedan. Gemensamt för alla är att datatypen baserar sig på BaseTextType, men att den förbjuder användningen av kontrolltecken som inte passar för ändamålet.

TextType är ämnad för data som innehåller fritext. Undantagna är XML:s kontrolltecken som begränsats i kategorin **'Cc'**, här tillåts dock särskilt XML:s teckenkategori **'s'**, dvs. **'White Space'**-tecken.

LineType är ämnad för textdata på en rad, **CodeType** för kodförkortningar och **IdentifierType** för identifikationsdata. Undantagna är XML:s kontrolltecken som begränsats i kategorin **'Cc'**, här tillåts dock särskilt XML:s teckenkategori **'t'**, som innehåller tecknet **HT** (vågrät tabulator).

Bilaga 4: XML-scheman och användningen av namnrymder

För enkelhetens skull beskrivs här endast de mest typiska sätten på vilka namnrymder används i XML-meddelanden. Avvikande och mer komplexa sätt att använda namnrymder har utelämnats från beskrivningen.

XML-scheman

XML-meddelandets struktur består av element och av attribut som eventuellt lagts till dem. Meddelandets schema fastslår vilka element som används i strukturen, var i meddelandet de är placerade och hurdan elementinnehåll som är godtagbart. Vanligtvis har strukturerna i specifika meddelanden definierats i schemat för det ifrågakvarande meddelandets struktur, men specifikationerna för ofta förekommande strukturer är färdiga komponenter som inte härstammar från dessa scheman utan från s.k. komponentscheman.

Då namnrymder används ska man i schemat deklarera alla de namnrymder som används däri. Till meddelandets målnamnsrymd för schemat (targetNamespace) hör strukturer som definierats i själva schemat, även de som baserar sig på datatyper som tagits från utomstående källor. Strukturernas underelement som härstammar från utomstående källor hör i sin tur till namnrymden för komponentschemat.

XML-meddelanden

Allmän praxis är att använda namnrymder för meddelandeelement, men inte för attribut. När meddelandets egentliga schema förutsätter att namnrymder används, ska åtminstone den målnamnrymd som definieras i schemat deklarerars i meddelandet. Komponentschemas namnrymder behöver deklarerars endast om meddelandet innehåller strukturer som tillhör dem.

Därutöver kan det hända att namnrymden "http://www.w3.org/2001/XMLSchema-instance", som innehåller specifikationer för allmänna XML-strukturer, behöver deklarerars i ett XML-meddelande. Namnrymden deklarerars vanligtvis med förkortningen xsi enligt följande:

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

Denna namnrymd behöver deklarerars i meddelandet när attribut som hör till namnrymden används i meddelandet (http://www.w3.org/TR/xmlschema-1/#Instance_Document_Constructions).

I XML-meddelanden kan strukturer ur namnrymden "http://www.w3.org/XML/1998/namespace" förekomma och till den hänvisar man alltid med förkortningarna xml. I undantagsfall bör man ändå inte uppvisa denna namnrymd, eftersom den alltid används med XML-tolkning.

Deklarering av namnrymder i XML-meddelanden

Namnrymderna deklarerars vanligtvis i starttaggen i meddelandets rotelement. Deklarationerna kan dock förekomma i element belägna var som helst i meddelandet, och i så fall verkar de bara inom ifrågavarande element. Det får inte förekomma element som saknar namnrymd när schemat förutsätter att namnrymder används. I samband med att den deklarerars anger man också en förkortning för namnrymden, utom om det är fråga om en förvald (default) namnrymd. Nedan finns ett exempel på hur namnrymderna deklarerars i början av ett FISummaryDeclaration-meddelande i AREX-systemet.

```
<?xml version="1.0" encoding="UTF-8"?>
<decl:FISummaryDeclaration xmlns:decl="http://tulli.fi/schema/arex/declaration/v3"
xmlns:wco="http://tulli.fi/schema/common/wco/v12_0">
```

På motsvarande sätt visas nedan ett exempel på hur namnrymderna för element som förekommer i AREX-systemets FIEntryExitResponse-meddelande deklarerars.

```
<?xml version="1.0" encoding="UTF-8"?>
<resp:FIEntryExitResponse xmlns:arex="resp://tulli.fi/schema/arex/response/v3"
xmlns:wco="http://tulli.fi/schema/common/wco/v12_0">
```

Beaktas bör att den förkortning som anges för en namnrymd då den deklarerars kan väljas fritt. Den valda förkortningen syns enbart inom denna enskilda förekomst av strukturen, vanligtvis inom ett enskilt meddelande. I stället för förkortningarna "decl" och "resp" som ingår i exemplen kunde även andra förkortningar användas. Ofta bildas förkortningarna för namnrymderna som används i meddelandet automatiskt t.ex. med löpande numrering, så som i exemplet nedan.

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:FIEntryExitResponse
xmlns:ns1="http://tulli.fi/schema/arex/response/v3"
xmlns:ns2="http://tulli.fi/schema/common/wco/v12_0">
```

Hur elementet binds till namnrymden i ett XML-meddelande

När schemat förutsätter att namnrymder används för elementen hör varje element som förekommer i meddelandena till någon namnrymd, eftersom element utan namnrymd inte tillåts. Namnrymden för ett element i ett XML-meddelande uttrycks så att förkortningen för den namnrymd som elementet hör till visas som prefix framför elementets namn. Varje namnrymd har namngetts med en egen förkortning vid deklareringen av de namnrymder som används i meddelandet. Prefixet och elementet åtskiljs med ett kolon. I namnet förekommer prefixet både i elementets start- och sluttagg. Nedan finns ett exempel på ett antal element i ett FISummaryDeclaration-meddelande där förkortningarna i det ovan nämnda exemplet används.

```
<decl:Agent>
  <wco:Party>
    <wco:identity>FI5342687-3</wco:identity>
    <wco:identityExtension>T0001</wco:identityExtension>
```

```

<wco:name1>Huolintatesti Oy</wco:name1>
<wco:Address>
  <wco:line>Tullihallitus 2</wco:line>
  <wco:city>Helsinki</wco:city>
  <wco:postCode>00110</wco:postCode>
  <wco:country>FI</wco:country>
</wco:Address>
</wco:Party>
</decl:Agent>

```

Förkortningen som valts för namnrymden är alltid specifik för meddelandet och gäller endast inom det enskilda meddelandet. Det kan alltså finnas olikheter mellan de förkortningar och prefix som används i separata meddelanden som följer en struktur som definieras i ett visst schema även om förkortningarna hänvisar till samma namnrymd. Det väsentliga med tanke på meddelandets riktighet är att elementen har kopplats till de rätta namnrymderna, inte vilka förkortningar som använts för att hänvisa till dem.

Förvald (default) namnrymd som ett specialfall av det ovan beskrivna

När namnrymder används behöver en förvald namnrymd (default namespace) inte ingå i ett meddelande, men om man vill kan man utse någon av de namnrymder som förekommer i meddelandet som förvald namnrymd. En namnrymd som då den deklarerats inte får någon förkortning blir därmed en förvald namnrymd.

Om en namnrymd inte har fått någon förkortning används inte heller någon förkortning som prefix till de element som hör till namnrymden. När en förvald namnrymd har deklarerats tolkas alla element som inte har något prefix framför sitt namn tillhöra den förvalda namnrymden.

Endast en namnrymd åt gången kan vara förvald namnrymd, vilket innebär att om flera namnrymder används så ska de andra då de deklarerats ges en förkortning som ska användas som prefix. Nedan finns ett exempel på hur en förvald namnrymd används i ett FISummaryDeclaration-meddelande.

```

<?xml version="1.0" encoding="UTF-8"?>
<FISummaryDeclaration xmlns="http://tulli.fi/schema/arex/declaration/v3"
  xmlns:wco="http://tulli.fi/schema/common/wco/v12_0">
  <Message>
    <wco:function>FI547A</wco:function>
    <wco:sender>003753426873</wco:sender>
    <wco:recipient>003702454428</wco:recipient>
    <wco:issue>2010-04-13T09:37:16</wco:issue>
    <wco:reference>TESTI1000004-01</wco:reference>
    <wco:test>1</wco:test>
  </Message>
  <Declaration>
    <Document>
      <wco:reference>Testi 3 meri</wco:reference>
      <wco:issue>2010-04-13T09:37:16</wco:issue>
    </Document>
    <Agent>
      <wco:Party>
        <wco:identity>FI5342687-3</wco:identity>
        <wco:identityExtension>T0001</wco:identityExtension>
        <wco:name1>Huolintatesti Oy</wco:name1>
        <wco:Address>
          <wco:line>Tullihallitus 2</wco:line>
          <wco:city>Helsinki</wco:city>
          <wco:postCode>00110</wco:postCode>
          <wco:country>FI</wco:country>
        </wco:Address>
      </wco:Party>
    </Agent>
  </Declaration>
</FISummaryDeclaration>

```

Den förvalda namnrymden kan också deklarerats i ett element beläget var som helst i meddelandet, och i så fall verkar den endast inom detta element.

Bilaga 5: Tekniska standarder för direkt meddelandedeklarering

Standard	Version
WS-I Basic Profile	1.1
SOAP	1.1 och 1.2
WSDL	1.1
HTTP	1.1
TLS	1.0 , 1.1 och 1.2
XMLDSIG	1.0 http://www.w3.org/TR/2002/REC-xmlsig-core-20020212/
SHA256 DigestMethod	http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/
RSA-SHA256 SignatureMethod	http://www.ietf.org/rfc/rfc4051.txt

Tabell 38: Tekniska standarder för direkt meddelandedeklarering