

# CPA Creation Toolkit – 4.0

## Simple Message Format Specification 2.0

June 2010  
Version: 1.0

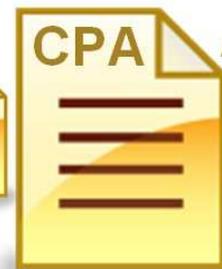
**Business  
Process  
Modelling**



**CPA Creation  
Toolkit**



**Collaboration  
Protocol  
Agreement**



*Automatic  
Configuration*



**ebMS**

## Inhoudsopgave

1 Inleiding.....	3
1.1 XML Validatie	4
2 Profiel en Service specificatie.....	5
2.1 Inleiding	5
2.2 Het <profiles> element	5
2.3 Het <services> element (meervoud)	6
2.4 Acknowledgements en de defaultDeliveryChannel	6
2.5 Het <service> element (enkelvoud)	7
2.6 Het <message> element	8
2.7 Regels voor het afleiden van de attribuut waarde in het <message> element	10
2.8 Persist Duration	10
3 Parameters specificatie .....	11
3.1 Het <parameters> element	11
3.2 RetryCount en RetryInterval	12
4 Role binding specificatie.....	13
4.1 Het <rolebindings> element	13
5 Bijlage – Lijst van beschikbare parameters.....	15
5.1 Verplichte parameters	15
5.2 Optionele parameters	16
5.3 Gebruik HTTP Endpoint URL	18
6 Bijlage – Parameter waardes opgeven.....	19

(Deze bladzijde is bewust leeg.)

# 1 Inleiding

Ten behoeve van het maken van een CPA (Collaboration Protocol Agreement) wordt een service specificatie opgesteld. Met behulp van het CPA Creatie programma en deze service specificatie worden dan de CPA's gemaakt.

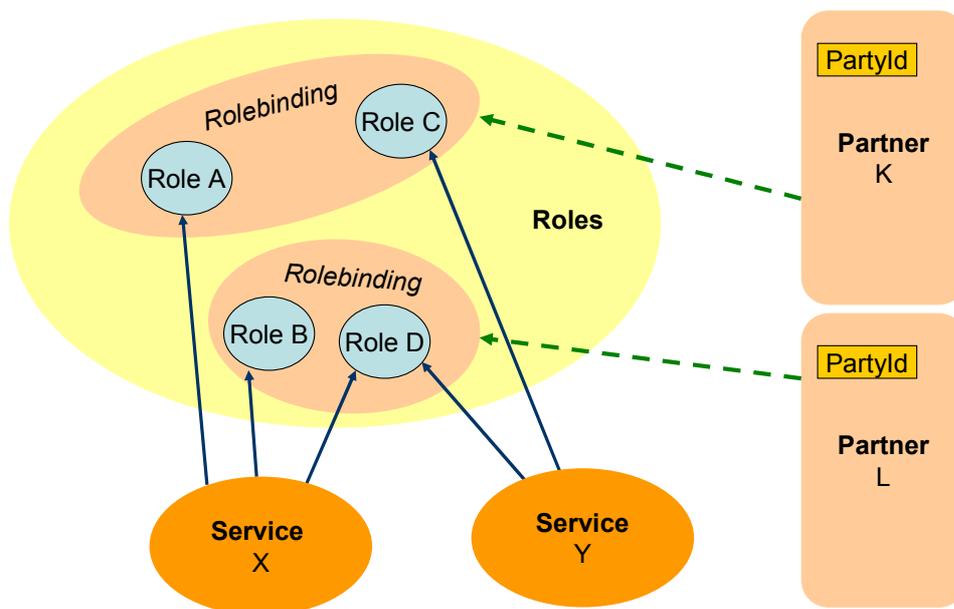
Dit document beschrijft de verschillende elementen van zo'n service specificatie. Een service specificatie beschrijft de berichten en de services in combinatie met de rollen die twee of meerdere partners kunnen innemen in een service. De gegevens worden vastgelegd in een xml formaat, aangeduid met SMF: **Simple Message Format 2.0**. Het bestaat uit vier elementen:

- [profiles](#)  
De **overdrachtskarakteristieken** uitgedrukt in een **profielnaam** (zoals 'jab-basis' of 'osb-rm').
- [services](#)  
De eigenschappen van de **services** en de berichten die met de services uitgewisseld kunnen worden. De berichten worden uitgewisseld tussen rollen. Er kunnen meerdere services in één specificatie beschreven worden.
- [rolebindings](#)  
De **groepering van rollen** die door een partner (participant) kunnen worden ingenomen. Een partner kan in verschillende services verschillende rollen innemen.
- [parameters](#)  
De **parameters** die door de participant moeten worden opgegeven om te kunnen komen tot een CPA. Er zijn verschillende groepen van parameters mogelijk.

De xml structuur van de SMF is vrij eenvoudig en wordt in de vervolg hoofdstukken beschreven. De basis opzet is als volgt:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <specification xmlns="http://sonnenglanz.net/2009/07/01/smf.services.2.0.xsd">
3 <!-- *****
4     PROFILES
5     *****-->
6 <profiles> [44 lines]
51 <!-- *****
52     SERVICES
53     *****-->
54 <services> [6 lines]
61 <!-- *****
62     ROLEBINDINGS
63     *****-->
64 <rolebindings> [7 lines]
72 <!-- *****
73     PARAMETERS
74     *****-->
75 <parameters> [11 lines]
87 </specification>
```

Het onderstaande voorbeeld is een visuele weergave van een specificatie met twee services, (service X en service Y) waarbij partner K in elke service een andere rol vervult: role A in service X en role C in service Y.



De Rolebinding groepeert de rollen die een partner kan vervullen in verschillende services: elke groep moet uiteindelijk aan een partner worden gekoppeld.

## 1.1 XML Validatie

Misschien ten overvloedde: valideer uw specificaties altijd met een goede XML editor! Gebruik hiervoor de bijgeleverde XSD's. Het betreft de XSD's:

- smf.services.2.0.xsd voor het valideren van service specificaties;
- smf.parameters.2.0.xsd voor het valideren van partner specifieke parameter bestanden;

## 2 Profiel en Service specificatie

### 2.1 Inleiding

Elke service definitie krijgt een unieke naam. Deze naam wordt aangeduidt met 'ID'. De naam ('ID') wordt ook gebruikt voor het opslaan en opzoeken van services en voor het maken van de CPA's met behulp van het CPA Creatie programma.

Het is de verantwoordelijkheid van de *Specifier* (een persoon!) om een unieke en betekenisvolle naam ('ID') te geven aan de service.

#### *Naamgevingsconventie*

De opbouw van de naam is gebaseerd op een hiërarchische structuur van sub-namen, gescheiden door een punt ('.'). Laat de naam beginnen met een unieke naam van de organisatie. De naam mag geen spaties of speciale tekens bevatten.

#### *Voorbeeld*

De service voor de EBV, CCI, 'Justitie Acceptatie Test' krijgt bijvoorbeeld de ID:  
EBV.CCI.JAT.1.0

### 2.2 Het <profiles> element

Het <profiles> element beschrijft een aantal profielen op basis waarvan berichten kunnen worden verzonden zoals aangegeven in het <message> element (zie het betreffende hoofdstuk).

Het <profiles> element bestaat uit één of meerdere elementaire <profile> elementen. Elk <profile> element heeft een aantal attributen. Naast een verplicht attribuut voor het opgeven van de naam van het profiel worden één of meerdere eigenschappen van de verzending opgegeven. Afgezien van het **name** attribuut worden de attributen uitvoerig beschreven in hoofdstuk "Het <message> element"!

- **name** (verplicht): de naam waarmee dit profiel wordt aangeduid in het <message> element (met behulp van het attribuut **profile**);
- **transport** (optioneel): de methode van de overdracht;
- **security** (optioneel): de methode van beveiliging;
- **retryCount** (optioneel) in geval van **transport="reliable"**: hoe vaak een poging wordt ondernomen voor het verzenden van een bericht;
- **retryInterval** (optioneel) in geval van **transport="reliable"**: hoe lang de periode duurt voordat een nieuwe poging wordt ondernomen voor het verzenden van een bericht;

Het moge duidelijk zijn dat als een profiel wordt gedefinieerd, er naast het **name** attribuut er natuurlijk minstens één ander attribuut moet worden opgegeven.

## 2.3 Het <services> element (meervoud)

Het hoofdelement <services> is een opsomming van één of meerdere elementaire <service> elementen. Voor het <services> element (meervoud) kan worden aangegeven welke default waardes er moeten worden gebruikt voor de **transport** en **security** attributen. De betekenis van de attributen wordt uitvoerig beschreven in hoofdstuk “Het <message> element”. De vier attributen zijn voorzien van een prefix ‘default’:

- **defaultTransport** (optioneel) met een default waarde **defaultTransport="besteffort"**;
- **defaultSecurity** (optioneel) met een default waarde **defaultSecurity="none"**;
- **defaultRetryCount** (optioneel) in geval van **transport="reliable"**, met een default waarde **defaultTransport="8"**;
- **defaultRetryInterval** (optioneel) in geval van **transport="reliable"**, met een default waarde **defaultSecurity="PT3H"**;

Deze default waardes worden gebruikt voor:

- het specificeren van de transport eigenschappen en de beveiliging, in het geval dat deze NIET zijn opgegeven in het <message> element of in het gerefereerde <profile> element (met het attribuut **profile** in het <message> element).

## 2.4 Acknowledgements en de defaultDeliveryChannel

Met betrekking tot het gebruik van de **acknowledgments** (en het bijbehorende defaultDeliveryChannel in de CPA) moeten er keuzes worden gemaakt: deze acknowledgements kunnen namelijk binnen één CPA maar op basis van één default waarde voor het **security** attribuut worden gedefinieerd.

Acknowledgments worden dus op basis van de waarde van het **defaultSecurity** attribuut afgeleverd.

Indien er een aantal berichten moeten worden gedefinieerd met **verschillende waardes voor het default security attribuut** dan zullen er meerdere service specificaties (bestanden) moeten worden gemaakt en daarmee dus verschillende CPA's.

Versie 4.0 van het CPA Creatie programma ondersteunt nog **niet** dat automatisch op basis van één service specificatie er meerdere CPA's worden gemaakt met verschillende default waardes voor het default security attribuut, en dien ten gevolge acknowledgements. Hierdoor zullen er handmatig meerdere service specificatie bestanden moeten worden gemaakt.

**NB.** De ebMS 2.0 standaard staat niet toe dat acknowledgments worden overgedragen met ‘acknowledgements’ (logisch!), zie bladzijde 42 van de [ebMS 2.0] specificatie:

“Note: *Acknowledgment Messages* are never acknowledged.”

Want dan zouden er een oneindig aantal acknowledgments moeten worden verstuurd voordat een bericht ‘reliable’ kan worden overgedragen...

## 2.5 Het <service> element (enkelvoud)

Het <service> element specificeert alle berichten die tussen twee partners, op basis van een bepaalde rol, kunnen worden uitgewisseld. Voor elke service zijn er drie verplichte attributen en drie optionele attributen:

- **service** (**verplicht**): de unieke naam van de service binnen het proces waarvoor de berichtenuitwisseling is gedefinieerd.
- **serviceId** (**verplicht**): de unieke identificatie van de service binnen deze specificatie (daarbuiten heeft het geen betekenis).
- **serviceType** (optioneel): het type van de service.
- **process** (**verplicht**): de naam van de procesbeschrijving waarvoor de berichtenuitwisseling is gedefinieerd.
- **processHref** (optioneel): een verwijzing (HREF) naar van de procesbeschrijving waarvoor de berichtenuitwisseling is gedefinieerd.
- **processUuid** (optioneel): de UUID van de procesbeschrijving waarvoor de berichtenuitwisseling is gedefinieerd.
- **processVersion** (optioneel): de versie van de procesbeschrijving waarvoor de berichtenuitwisseling is gedefinieerd.

*Voorbeeld*

```
<service    serviceId="s1"  
           process="Bijwerken"  
           service=" Bijwerken:1:2"  
           serviceType="urn:sonnenglanz:services">
```

*Einde voorbeeld*

## 2.6 Het <message> element

Elk bericht in een service wordt beschreven met een eigen <message> element. Elk <message> element heeft een afzender attribuut (“From”) en een ontvanger attribuut (“To”). De waarde van deze attributen komt overeen met **de rol** die de partner inneemt *in deze service specificatie*. Optioneel worden de eigenschappen van de verzending opgegeven. Dit wordt gedaan met behulp van de (optionele) attributen **security**, **transport**, **retryCount** & **retryInterval** en **profile**.

- **security** (optioneel): de aanduiding voor het beveiligings nivo. De default waarde is **none** (geen transport of payload beveiliging) tenzij anders aangegeven met het **defaultSecurity** attribuut in het <services> element.  
De mogelijke waardes zijn:
  - **transport** voor *dubbelzijdig* ssl (default geen transport beveiliging);
  - **sign** voor ondertekening van de payload (default geen signing);
  - **payload** voor ondertekening en encryptie van de payload (default geen payload beveiliging);
  - **transport-and-sign** voor een combinatie van transport en payload beveiliging;
  - **transport-and-payload** voor een combinatie van transport en payload beveiliging;
- **transport** (optioneel): de aanduiding voor de methode van de overdracht. De default waarde is **besteffort**, tenzij anders aangegeven met het **defaultTransport** attribuut in het <services> element.  
De mogelijke waardes zijn:
  - **besteffort** ten behoeve van interactieve bevragingen zonder ontvangstbevestiging;
  - **reliable** ten behoeve van berichtenuitwisseling met ontvangstbevestiging;
- **retryCount** (optioneel): het aantal pogingen om een bericht opnieuw te verzenden nadat voorafgaande verzending is mislukt. Dit attribuut is alleen zinvol als **transport="reliable"** is opgegeven! Het moet een integer waarde zijn. De default waarde is ‘8’.
- **retryInterval** (optioneel): het tijdsinterval tussen twee pogingen om een bericht opnieuw te verzenden nadat een voorafgaande verzending is mislukt. Dit attribuut is alleen zinvol als **transport="reliable"** is opgegeven! Mogelijke waardes moeten worden gespecificeerd conform ‘duration’ in “XML Schema[XMLSCHEMA-2]”. De default waarde is 3 uur: ‘PT3H’. Default is de totale periode waarin her-transmissies kunnen plaatsvinden  $(8+1) \times 3 \text{ uur} = 27 \text{ uur}$ .
- **profile** (optioneel): de naam van het profiel zoals deze is gespecificeerd in het <profiles> element.

Een service specificatie kan verschillende <message> elementen definiëren met verschillende waardes voor transport **transport**, **security**, **retryCount** of **retryInterval**.

### Voorbeeld

Stel het `<services>` element definieert de waarde `defaultSecurity="payload"`. Dan is:

```
<message from="A" to="B">A</message>
```

Hetzelfde als:

```
<message from="A" to="B" transport="besteffort" security="payload">A</message>
```

Een ander, uitgebreider, voorbeeld:

```
<services defaultTransport ="besteffort">
  <service serviceId="s1" service="VerwijsIndex:Actualiseren:0:8"
    servicetype="urn:ebv:services" process="Verwijsindex Actualiseren">
    <message from="Ketenpartner" to="VerwijsIndex"
      transport ="reliable" security="payload"
      retryCount="12" retryInterval="PT3H"
    >InschrijvenPersoonsverwijzing</message>
    <message from="Ketenpartner" to="VerwijsIndex"
      transport ="reliable" security="payload"
      retryCount="5" retryInterval="PT8H"
    >UitschrijvenPersoonsverwijzing</message>
  </service>
  <service serviceId="s2" service="VerwijsIndex:ZoekenPersoon:0:8"
    servicetype="urn:ebv:services" process="Verwijsindex ZoekenPersoon">
    <message from="Ketenpartner" to="VerwijsIndex"
    >ZoekenPersoon</message>
    <message from="VerwijsIndex" to="Ketenpartner"
      security="payload"
    >PersoonsgegevensGevonden</message>
    <message from="VerwijsIndex" to="Ketenpartner"
    >NietsGevonden</message>
  </service>
</services>
```

*Einde voorbeeld*

## 2.7 Regels voor het afleiden van de attribuut waarde in het <message> element

Omdat de attributen een waarde kunnen krijgen op basis van :

- (i) de impliciete default waarde,
- (ii) de gespecificeerde default waarde,
- (iii) de waarde in een <profile> element of
- (iv) de waarde zoals aangegeven in het <message> element,

kan de vraag worden gesteld wat de uiteindelijke waarde is van zo'n attribuut!

De wijze waarop de uiteindelijke waarde van een attribuut wordt afgeleid is hieronder beschreven. Het gaat hierbij (dus) om de waarde voor de attributen **transport**, **security**, **retryCount** of **retryInterval**.

1. Wordt het attribuut gespecificeerd in het <message> element?  
Zo ja, gebruik dan die waarde.
2. Is het bovenstaande niet van toepassing en wordt het attribuut gespecificeerd in het gerefereerde profiel in het <message> element?  
Zo ja, gebruik dan die waarde van uit het opgegeven profiel.
3. Als het bovenstaande niet van toepassing is, wordt het attribuut dan gespecificeerd door een default waarde in het <services> element?  
Zo ja, gebruik dan die waarde.
4. Als er niets is opgegeven, worden de default waardes gebruikt zoals die zijn voorgedefinieerd in de toolkit:  
**transport** = "best-effort"  
**security** = "none"  
**retryCount** = "8"  
**retryInterval** = "PT3H"

Samenvattend: de waarde wordt bepaald door de rang-orde waarin een attribuut voorkomt:

- 1<sup>e</sup> : het Message element;
- 2<sup>e</sup> : het gerefereerde Profile element;
- 3<sup>e</sup> : de gespecificeerde default waarde;
- 4<sup>e</sup> : de impliciete waarde (de 'default' als er geen default waarde is opgegeven).

## 2.8 Persist Duration

Op basis van de **retryCount** en **retryInterval** wordt tevens de **Persist Duration** afgeleid. Een minimum termijn van 7 dagen is ingesteld. Daar bovenop komt dan de periode die volgt uit de **retryCount** en **retryInterval**.

**Let op:** het feit dat de 'persist duration' wordt opgegeven in de CPA wil nog niet zeggen dat elke ebMS adapter deze waarde respecteert!

## 3 Parameters specificatie

Elke partner die gebruik wil gaan maken van een bepaalde service zal een aantal technische gegevens moeten aanleveren, zoals de URL en de PartyId. Welk soort gegevens dit zijn wordt vastgelegd in het `<parameters>` element.

### 3.1 Het `<parameters>` element

Het `<parameters>` element bestaat uit één of meerdere `<group>` elementen waarbij elk `<group>` element een attribuut `parameterId` heeft met een unieke identificerende waarde.

Voor elk soort gegeven in `<group>` kan aangegeven worden of deze verplicht is of niet. Dit wordt aangegeven met het attribuut `required` en kan de waarde “true” of “false” hebben.

Indien een soort gegeven niet verplicht is (`required="false"`) zal er een default waarde moeten worden gegeven.

**NB.** Gegevens die niet worden gespecificeerd met het `<parameters>` element kunnen naderhand niet meer worden opgegeven in het input bestand `parameters.xml` (*waarmee elke partner de specifieke waardes opgeeft*). Met andere woorden, de parameters die moeten of kunnen worden opgegeven in het bestand `parameters.xml` volgen uit de gegevens die in deze specificatie zijn beschreven.

**Let op:** het `parameters.xml` is een separaat bestand en dus niet het service specificatie bestand. Hierdoor kan hergebruik plaatsvinden van de gegevens ten behoeve van de deelname van de partner in andere services!

*Voorbeeld*

```
<parameters>
  <group parameterId="p1">
    <parameter name="PartyName" required="true"/>
    <parameter name="PartyRef" required="true"/>
    <parameter name="PartyIdList" required="true"/>
    <parameter name="PartyIdType" required="false">urn:xyz:private</parameter>
    <parameter name="ConditionalEndpointUri" required="true"/>
  </group>
</parameters>
```

*Einde voorbeeld*

Het hoofdstuk ‘Lijst van beschikbare parameters’ geeft een compleet overzicht van alle parameters.

## 3.2 RetryCount en RetryInterval

Ten aanzien van de specificatie voor Reliable Messaging kan per partner een RetryCount en RetryInterval worden opgegeven (de parameters hebben de overeenkomstige naam). De default waarden zijn RetryCount=8 (maximaal 8 extra pogingen als de eerste verzending mislukt) en RetryInterval=PT3H (3 uur).

Als er meerdere partners een rol spelen in een service specificatie, kan het zijn dat de RetryCount en RetryInterval verschillend moeten zijn tussen de partners. Om dit *in enige mate* (en dus beperkt!) te kunnen ondersteunen is het attribuut ‘**bothsides**’ geïntroduceerd. Als één van de partners een waarde heeft aangegeven met de aanduiding **bothsides="true"** dan zal deze waarde ook worden gebruikt voor de andere partner in dezelfde CPA.

### Voorbeeld

In dit voorbeeld zijn er drie partners die berichten met elkaar uitwisselen. Eén van de partners fungeert als centrale ‘hub’ (“P1”): de andere partners (“P2” en “P3”) communiceren alleen met deze centrale partner. De RetryCount tussen partner P1 en P2 moet worden ingesteld op 10 en de RetryCount tussen partner P1 en P3 moet worden ingesteld op 20. Het onderstaande voorbeeld geeft aan hoe dit wordt gespecificeerd. Merk op dat P1 **geén** waarde opgeeft voor de RetryCount: bij het maken van de CPA wordt de waarde *van de andere partner* gebruikt (waardoor de waarde dus afhankelijk is van wat *de andere partner* opgeeft). Merk op dat als P2 en P3 met elkaar communiceren, het niet vaststaat welke waarde er uiteindelijk gebruikt wordt. *Het gebruik van het attribuut “bothsides” heeft dus beperkingen.*

```
<parameters>
  <group parameterId="P1">
    <parameter name="PartyName" required="true"/>
    <parameter name="PartyId" required="true"/>
  </group>
  <group parameterId="P2">
    <parameter name="PartyName" required="true"/>
    <parameter name="PartyId" required="true"/>
    <parameter name="RetryCount" required="false"
      bothsides="true">10</parameter>
  </group>
  <group parameterId="P3">
    <parameter name="PartyName" required="true"/>
    <parameter name="PartyId" required="true"/>
    <parameter name="RetryCount" required="false"
      bothsides="true">20</parameter>
  </group>
</parameters>
Einde voorbeeld
```

## 4 Role binding specificatie

Een partner kan *verschillende rollen* vervullen in *één en dezelfde service*. Een partner kan echter ook een rol vervullen in *verschillende services*. Welke rollen er worden vervuld is aangegeven in het `<binding>` element binnen de `<rolebindings>` structuur in de service specificatie. Omdat er meerdere services kunnen zijn, moet worden aangegeven binnen welke service de rol wordt gebruikt.

### 4.1 Het `<rolebindings>` element

Het `<rolebindings>` element bevat twee of meer `<binding>` elementen. Elke `<binding>` element heeft twee verplichte attributen:

- **name** (**verplicht**): de naam van de binding. Deze naam zal ook gepresenteerd worden aan de gebruiker bij het maken van een CPA.
- **parameterId** (**verplicht**): een unieke identificatie van de specificatie van de op te geven parameters (zie vorig hoofdstuk).

Een `<binding>` kan één of meerdere rolnamen bevatten. Deze rolnamen moeten natuurlijk wel terug te vinden zijn in de `<message>` elementen (de “to” of “from” attribuut waarde). Een `<role>` heeft twee attributen:

- **serviceId** (**verplicht**): de unieke identificatie van de service waartoe de rol behoort.
- **name** (**verplicht**): de naam van de rol binnen de aangegeven service.

### *Voorbeeld*

Er worden twee bindings opgegeven. Welke parameters moeten worden opgegeven is aangegeven met de identifier parameterId: deze wijst naar het <group> element waarin de soorten gegevens worden opgesomd.

Voor elke rol is aangegeven binnen welke service deze voorkomt: de rol 'Beantwoorden' wordt zowel binnen service s1 als wel binnen s2 gebruikt.

```
<rolebindings>
  <binding name="JustID" parameterId="p1">
    <role serviceId="s1" name="Beantwoorden"/>
    <role serviceId="s2" name="Beantwoorden"/>
  </binding>
  <binding name="Ketenpartner" parameterId="p1">
    <role serviceId="s1" name="Aanleveren"/>
    <role serviceId="s2" name="Vragen"/>
  </binding>
</rolebindings>
```

### *Einde voorbeeld*

Hoewel het bovenstaande voorbeeld het niet weergeeft is het mogelijk dat er meerdere *verschillende rollen* in *eenzelfde* service worden opgegeven in een binding. Ter illustratie het onderstaande voorbeeld.

### *Voorbeeld*

```
<binding name="JustID" parameterId="p1">
  <role serviceId="s1" name="Leverancier"/>
  <role serviceId="s1" name="Aanvrager"/>
</binding>
```

### *Einde voorbeeld*

## 5 Bijlage – Lijst van beschikbare parameters

De parameters zijn verdeeld naar verplichte parameters en optionele parameters.

### 5.1 Verplichte parameters

#### **Verplichte parameters**

<i>Parameter</i>	<b>PartyName</b>
<i>Betekenis</i>	De naam van de partij die een bepaalde rol inneemt.
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Verplicht.
<i>Voorbeeld</i>	<code>&lt;parameter name="PartyName" required="true"/&gt;</code>
<i>Parameter</i>	<b>PartyId</b>
<i>Betekenis</i>	De logische identificatie van de partij die een bepaalde rol inneemt.
<i>Default</i>	De SYSDA code van de organisatie. (Voor de OSB wordt een OIN gebruikt.) Specificeer de juiste PartyId type! (Zie volgende bladzijde.)
<i>Voorkomen</i>	Er moet OF een PartyId OF een PartyIdList gedefinieerd zijn (niet beide).
<i>Voorbeeld</i>	<code>&lt;parameter name="PartyId" required="true"/&gt;</code>
<i>Parameter</i>	<b>PartyIdList</b>
<i>Betekenis</i>	Een lijst met logische identificaties van de partijen die een bepaalde rol innemen. Deze PartyId's zijn verbonden aan één en dezelfde PartyName (formeel zijn het aliassen voor de eerste in de lijst).
<i>Default</i>	Zie <i>PartyId</i> .
<i>Voorkomen</i>	Er moet OF een PartyId OF een PartyIdList gedefinieerd zijn (niet beide).
<i>Voorbeeld</i>	<code>&lt;parameter name="PartyIdList" required="true"/&gt;</code> of <code>&lt;parameter name="PartyIdList" required="false"&gt;</code> <code>&lt;value&gt;01&lt;/value&gt;</code> <code>&lt;value&gt;02&lt;/value&gt;</code> <code>&lt;value&gt;03&lt;/value&gt;</code> <code>&lt;/parameter&gt;</code>
<i>Parameter</i>	<b>EndpointUriConditional</b>
<i>Betekenis</i>	De HTTP(S) transport url van de ebMS adapter van de organisatie en de url van de ebMS adapter zelf.
	Noot: lees ook hoofdstuk 5.3 “ <b>Gebruik HTTP Endpoint URL</b> ”.
<i>Default</i>	Er kunnen (nog) geen default waardes opgegeven worden!
<i>Voorkomen</i>	Verplicht.
<i>Voorbeeld</i>	<code>&lt;parameter name="ConditionalEndpointUri" required="true"/&gt;</code>

## 5.2 Optionele parameters

<i>Parameter</i>	<b>PartyIdType</b>
<i>Betekenis</i>	De typering van de PartyId (of PartyIdList elementen) van de logische identificatie.
<i>Default</i>	De default waarde is <b>urn:epv:sysda</b> en typeert het gebruik van een identificatie schema van de organisaties (voor EBV de SYSDA tabel) als PartyId. Indien een andere identificatie schema gehanteerd wordt voor de PartyId zal de typering gebruikt moeten worden die daarbij past. (Voor de OSB wordt een OIN gebruikt welke getypeerd wordt door <b>urn:osb:oin</b> )
<i>Voorkomen</i>	Optioneel indien gebruik gemaakt wordt van SYSDA codes; verplicht in de overige gevallen.
<i>Voorbeeld</i>	<code>&lt;parameter name="PartyIdType" required="false"&gt;urn:epv:sysda&lt;/parameter&gt;</code>
<i>Parameter</i>	<b>PartyRef</b>
<i>Betekenis</i>	Een referentie naar een webpagina of website met informatie over de betreffende service van de organisatie. De referentie is bedoeld om gebruikt te worden door personen.
<i>Default</i>	Leeg.
<i>Voorkomen</i>	Optioneel.
<i>Voorbeeld</i>	<code>&lt;parameter name="PartyRef" required="true"/&gt;</code>
<i>Parameter</i>	<b>EndpointUri</b>
<i>Betekenis</i>	De HTTP(S) transport url van de ebMS adapter van de organisatie. <div style="border: 1px solid black; padding: 2px; margin-top: 5px;">Noot: lees ook hoofdstuk 5.3 “<b>Gebruik HTTP Endpoint URL</b>”.</div>
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Optioneel als er gebruik gemaakt wordt van <b>EndpointUriConditional</b> ; anders Verplicht.
<i>Voorbeeld</i>	<code>&lt;parameter name="EndpointUri" required="true"/&gt;</code>
<i>Parameter</i>	<b>RetryCount</b>
<i>Betekenis</i>	De RetryCount geeft aan hoe vaak een poging gedaan moet worden om een bericht te versturen.
<i>Default</i>	De default waarde voor reliable messaging is acht (8) retries (dus totaal negen momenten waarop de verzending plaatsvindt).
<i>Voorkomen</i>	Optioneel. Mogelijke waarden: integer groter dan nul.
<i>Optie</i>	Als extra optie kan aangegeven worden of de waarde van toepassing is voor beide partijen in de CPA. Hiervoor moet een extra attribuut opgegeven worden: <b>bothsides="true"</b> (default is deze “false”).
<i>Voorbeeld</i>	<code>&lt;parameter name="Retries" required="false"&gt;8&lt;/parameter&gt;</code>

<i>Parameter</i>	<b>RetryInterval</b>
<i>Betekenis</i>	Het RetryInterval geeft aan hoe lang er gewacht wordt voordat er een nieuwe poging gedaan wordt om een bericht opnieuw te versturen.
<i>Default</i>	De default waarde voor reliable messaging is een interval van drie (3) uur.
<i>Voorkomen</i>	Optioneel. Mogelijke waarde: zie ‘duration’ in “XML Schema[XMLSCHEMA-2]”.
<i>Optie</i>	Als extra optie kan aangegeven worden of de waarde van toepassing is voor beide partijen in de CPA. Hiervoor moet een extra attribuut opgegeven worden: <code>bothsides="true"</code> (default is deze “false”).
<i>Voorbeeld</i>	<code>&lt;parameter name="RetryInterval" required="false"&gt;PT3H&lt;/parameter&gt;</code>
<i>Parameter</i>	<b>PersistDuration</b>
<i>Betekenis</i>	De PersistDuration geeft aan hoe lang berichten bewaard moeten worden door de ebMS adapter. Het is een minimum periode. Door deze periode langer te maken dan de periode van <i>RetryInterval</i> x <i>Retries</i> is de beheerder van de ebMS adapter in staat om berichten die uiteindelijk niet overgedragen konden worden binnen de gestelde termijn, toch opnieuw te versturen.
<i>Default</i>	De default waarde is 7 dagen verhoogd met de periode van de RetryCount en RetryInterval.
<i>Voorkomen</i>	Optioneel. Mogelijke waarde: zie ‘duration’ in “XML Schema[XMLSCHEMA-2]”.
<i>Voorbeeld</i>	<code>&lt;parameter name="PersistDuration" required="false"&gt;PT7D&lt;/parameter&gt;</code>
<i>Parameter</i>	<b>ClientCert</b>
<i>Betekenis</i>	Het publieke deel van het client certificaat van de organisatie.
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Optioneel. Het gegeven wordt opgegeven in de vorm van een KeyInfo structuur.
<i>Voorbeeld</i>	<code>&lt;parameter name="ClientCert" required="true"/&gt;</code>
<i>Parameter</i>	<b>ServerCert</b>
<i>Betekenis</i>	Het publieke deel van het server certificaat van de server van de ebMS adapter van de organisatie.
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Optioneel. Het gegeven wordt opgegeven in de vorm van een KeyInfo structuur.
<i>Voorbeeld</i>	<code>&lt;parameter name="ServerCert" required="true"/&gt;</code>
<i>Parameter</i>	<b>SigningCert</b>
<i>Betekenis</i>	Het publieke deel van het signing certificaat van de server van de ebMS adapter van de organisatie.
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Optioneel. Het gegeven wordt opgegeven in de vorm van een KeyInfo structuur.
<i>Voorbeeld</i>	<code>&lt;parameter name="SigningCert" required="true"/&gt;</code>
<i>Parameter</i>	<b>EncryptionCert</b>
<i>Betekenis</i>	Het publieke deel van het encryption certificaat van de server van de ebMS adapter van de organisatie.
<i>Default</i>	Niet van toepassing.
<i>Voorkomen</i>	Optioneel. Het gegeven wordt opgegeven in de vorm van een KeyInfo structuur.
<i>Voorbeeld</i>	<code>&lt;parameter name="EncryptionCert" required="true"/&gt;</code>

<i>Parameter</i>	<b>RootCert</b>
<i>Betekenis</i>	Het publieke deel van de root certificaten die vertrouwd worden door de server van de ebMS adapter van de organisatie. Er kunnen meerdere certificaten in opgenomen worden.
<i>Default</i>	Wordt niet gebruikt.
<i>Voorkomen</i>	<i>Onbepaald.</i> Het gegeven wordt opgegeven in de vorm van een KeyInfo structuur.
<i>Voorbeeld</i>	<code>&lt;parameter name="RootCert" required="true"/&gt;</code>

### 5.3 Gebruik HTTP Endpoint URL

Als er gebruik moet worden gemaakt van zowel HTTP als HTTPS en er wordt maar één URL voor het Endpoint opgegeven, dan geldt het volgende:

- Indien de standaard poorten worden gebruikt (poort 80 voor HTTP en poort 443 voor HTTPS) dan kan met één URL worden volstaan. Er moet dan géén poort nummer worden opgegeven in de URL van het Endpoint! Tijdens de CPA creatie zal op de juiste plek de ‘http’ indien nodig vervangen worden door ‘https’.
- Indien NIET de standaard poorten worden gebruikt (zie voorgaande ‘bullet’) dan wordt de volgende regel gehanteerd:
  - o Afhankelijk van de opgegeven url (met of zonder ‘s’ bij ‘http’), en het gebruik in de CPA, wordt het poort nummer verhoogd of verlaagd met de waarde 363 (= 443-80).
    - Als de opgegeven URL alleen HTTP bevat, wordt het poort nummer verhoogd.
    - Als de opgegeven URL alleen HTTPS bevat, wordt het poort nummer verlaagd.

**Voorbeeld:**

In de URL staat HTTP en poort 4080. Dan zal voor HTTPS poort 4443 worden berekend (= 4080 + (443-80) = 4443).

Indien *een andere waarde dan 363* voor deze increment nodig is, wijzig dan de overeenkomstige default waarde in het property bestand (onderdeel 10, zie de CPA Creatie Toolkit installatie handleiding of de bijlage in de CPA Creatie Toolkit gebruikers handleiding).

## 6 Bijlage – Parameter waardes opgeven

De service definitie beschrijft welke parameters opgegeven moeten worden door een partner die een rol wil innemen in service. Dit hoofdstuk geeft aan hoe de partner de feitelijke waardes moet opgeven voor deze parameters.

Het document is een xml structuur met één element `<parameters>` bestaande uit één of meerdere `<parameter>` elementen. Elk element `<parameter>` heeft één attribuut `name` waarmee aangegeven wordt voor welke parameter een waarde wordt opgegeven.

*Voorbeeld*

```
<?xml version="1.0" encoding="UTF-8"?>
<parameters>
  <parameter name="PartyName">Nederlandse Bedrijven</parameter>
  <parameter name="PartyRef">http://www.bedrijf.nl</parameter>
  <parameter name="PartyIdList">
    <value>IA0112</value>
  </parameter>
  <parameter name="ConditionalEndpointUri">
    <condition name="local"
      switch="false">http://ebms.bedrijf.nl:4080/exchange/IA0112</condition>
    <condition name="default"
      switch="true">http://10.19.3.21:4080/exchange/intermediary</condition>
  </parameter>
</parameters>
```

*Einde voorbeeld*