

## Rapport

# CPA Creatie Proces Ontwerp

Datum  
15-05-2007

Onderwerp

Auteur  
E.J. van Nigtevecht

E-mail  
EJvN@Sonnenglanz.net

Versie  
0.3

Projectmanager  
Brian Dommissie

Opdrachtgever  
Stuurgroep BBO

## Inhoudsopgave

<b>CPA Creatie Proces Ontwerp</b>	<b>1</b>
<b>Inhoudsopgave</b>	<b>2</b>
<b>1. Projectplan</b>	<b>4</b>
1.1 Doel van dit document	4
1.2 Plaats document	4
1.3 Versie geschiedenis	4
1.4 Goedkeuring	4
1.5 Verspreiding	4
1.6 Opbouw van het document	5
<b>2. Inleiding</b>	<b>6</b>
2.1 Aansluiten op de OSB	6
2.2 CPA Creatie Proces	6
2.3 OSB-Profiel	7
2.4 Uitgangspunten	7
<b>3. Specificatie</b>	<b>8</b>
3.1 Service Identificatie	8
3.2 Berichten specificatie	8
3.3 Parameter specificatie	8
3.4 Parameter Waardes	10
3.5 Gegeneerde bestanden	11
<b>4. Creatie Proces</b>	<b>12</b>
4.1 Rollen	12
4.2 Sequence Diagram	13
4.3 Connectivity CPA	16
<b>5. Specialisaties</b>	<b>17</b>
5.1 CPA Creatie op basis van een bericht specificatie	18
5.2 CPA Creatie op basis van een CPA template	20
5.3 CPA Creatie voor één Service Provider en meerdere Service Requesters	21
5.4 Python scripts gebruiken	22
<b>A. Referenties</b>	<b>23</b>



## 1. Projectplan

### 1.1 Doel van dit document

Dit document beschrijft het ontwerp van het CPA creatie proces in het kader van het projectvoorstel "Modelgestuurd Partner Management en Berichtenuitwisseling" [BBO-MPM]. Dit project heeft ten doel om *schaalbaarheid* van berichtenverkeer te verbeteren en ketenorganisaties te helpen zich toe te rusten voor situaties waarin niet slechts een handvol maar tientallen of honderden organisaties in de keten samenwerken, en waarin niet slechts enkele tientallen berichtenstromen moeten worden beheerd, maar honderden of zelfs duizenden, sommig in meerdere versies naast elkaar.

### 1.2 Plaats document

De bron van dit document is te vinden in het archief van het project Bouwstenen Berichtenuitwisseling, ondergebracht bij het CIP.

### 1.3 Versie geschiedenis

Versienummer	Versiedatum	Veranderingen
0.1	23-04-2007	Initieel rapport
0.2	29-04-2007	Commentaar Paul Schlotter (GBO.Overheid) verwerkt.
0.3	15-05-2007	

### 1.4 Goedkeuring

Dit document is geldig indien goedgekeurd en ondertekend door:

Naam	Handtekening	Functie	datum	versie
Stuurgroep				1.0

### 1.5 Verspreiding

Dit document wordt toegestuurd naar:

Naam	Functie	Datum	Vanaf versie
Stuurgroep Bouwstenen			
Pim van der Eijk			
Ernst Jan van Nigtevecht			
Albert Kappe			
Brian Dommissie			

## 1.6 Opbouw van het document

In hoofdstuk 2, *Inleiding*, wordt het gebruik van CPA's beschreven en het de aannames die gemaakt zijn voor het CPA creatie proces. Daarnaast wordt de relatie met de Directory aangestipt.

Hoofdstuk 3, *Specificatie*, beschrijft de verschillende gegevens waarmee een CPA uiteindelijk gemaakt kan worden.

Het 4<sup>e</sup> hoofdstuk, *Creatie Proces*, beschrijft het proces waarin een CPA tot stand komt. Hierbij wordt op basis van een viertal rollen en een aantal interacties een CPA gemaakt. Het proces is bruikbaar voor meer dan twee partijen die een samenwerking willen aangaan. Voor release 1 van de OSB zal het gebruikt worden voor 2 partijen.

In hoofdstuk 5, *Specialisaties*, wordt een specialitie beschreven van het in hoofdstuk 4 beschreven proces. De specialisatie richt zich op het op interactieve wijze aanvragen van een CPA.

## 2. Inleiding

Voor het uitwisselen van gegevens met behulp van ebMS zijn een aantal afspraken nodig. Deze afspraken zijn vastgelegd in een aantal documenten:

- Een procesbeschrijving van de samenwerking, in principe op basis van ebBP ([ebBP]). Voor de OSB wordt voor de eerste release een eenvoudigere versie gebruikt, het zg. "Simple Message Format". Dit wordt nader uitgewerkt in hoofdstuk 3.
- De ebXML Deployment Profile waarin vastgelegd wordt hoe bepaalde ebMS configuratie waardes gebruikt moeten worden ([ebMS-DPT]). De OSB benoemd twee typen voor ebMS, aangeduid met "Best Effort" en "Reliable Messaging" ([OSB1.0]). Voor elk van deze typen zijn de configuratie waardes gedefinieerd, en is terug te vinden in [OSB-Profiel].
- Een "Collaboration Protocol Profile" (zie [CPPA]), aangeduid met CPA, waarin voor een specifieke bilaterale samenwerking (tussen precies 2 partijen) verschillende transport parameters beschreven staan. De ebXML standaard specificeert de CPA tesamen met de Collaboration Profile, aangeduid met CPP, maar deze laatste wordt vooralsnog niet gebruikt als input voor het CPA creatie proces.
- De uiteindelijke structuur van de ebMS berichten is beschreven in [ebMS] en wordt in dit document verder niet toegelicht. Deze berichten zullen door o.a. de Gateway gemaakt moeten worden.

Om het CPA Creatie Proces begrijpelijk te houden is geprobeerd om de verschillende aspecten zoveel mogelijk zichzelf te laten verklaren, dus zonder specifieke kennis van ebXML, CPA, etc.

### 2.1 Aansluiten op de OSB

Voor het aansluiten van organisaties op de OSB via ebMS zijn CPA's nodig tussen de organisaties die gegevens willen uitwisselen. De rol die zo'n organisatie speelt kan een *Service Requester* (SR) zijn maar evengoed een *Service Provider* (SP). In dit document wordt vaak de algemene term 'Service' gehanteerd: welke rol (SP of SR) een Service heeft hangt af van de gewenste samenwerking.

In de eerste instantie beperken we ons tot de situatie met 1 SP (met voorafgedefinieerde services) en meerdere SR. De SR zal het initiatief nemen tot het creëren van een relatie met een SP. Tevens wordt aangenomen dat er vele SR's zullen zijn voor een specifieke SP, en dat er diverse SP's kunnen zijn.

Als er een nieuwe service beschikbaar gesteld wordt aan de OSB zullen een aantal zaken ingeregeld moeten worden. Dit zal allemaal "Design Time" plaatsvinden, dus nog voordat de service operationeel is. Eén van de aspecten is het beschrijven van de service. Voor ebMS gerelateerde services zal er specificatie gemaakt worden op basis van een OSB profiel en service definitie. Daarnaast zullen er een aantal parameters geïdentificeerd moeten worden die ingevuld moeten worden om tot een CPA te komen. De specifieke waardes van de parameters worden bepaald door de partijen die een samenwerking met elkaar willen aangaan (dus zowel van een SP als van een SR).

### 2.2 CPA Creatie Proces

Het handmatig construeren van een CPA is een lastig en bewerkelijk proces. Het proces waarin een CPA gemaakt wordt kan beter beheerbaar gemaakt worden door het te automatiseren.

Het CPA creatie proces heeft tot doel om op geautomatiseerde wijze een CPA te maken op basis van de OSB profielen als vastgelegd in de Koppelvlakstandaard, de service definitie en de specifieke transport gegevens van de samenwerkende partijen (zowel van de SP als de SR). De CPA zal voldoen aan de ebXML CPPA standaard ([CPPA]).

### 2.3 OSB-Profiel

De OSB ebMS profielen kennen 2 soorten. Voor ebMS zijn er de volgende profielen vastgelegd ([OSB-Profiel]):

- Best Effort: dit zijn asynchrone uitwisselingen die geen faciliteiten voor betrouwbaarheid (ontvangstbevestigingen, duplicaateliminatie etc.) vereisen.
- Reliable Messaging: asynchrone uitwisseling met ontvangstbevestigingen en duplicaateliminatie door de ontvangende message handler.

Op basis van deze profielen zijn verschillende ebXML parameters gedefinieerd. Deze ebXML parameters zullen gebruikt worden bij de constructie van een CPA template die aan de basis ligt van alle CPA's voor de OSB.

### 2.4 Uitgangspunten

1. Het proces is bedoeld om op geautomatiseerde wijze een CPA, conform [CPPA], voor diverse partijen te creëren.
2. Het proces moet uitgevoerd kunnen worden op basis van ebMS transport, maar strikt noodzakelijk is dit niet. Het moet ook mogelijk zijn om op interactieve wijze een CPA te verkrijgen.
3. Een CPA zal gemaakt moeten kunnen worden op basis van een ebBP, of vergelijkbare specificatie, in combinatie met specifieke informatie van de partijen die met elkaar willen samenwerken. (Een ebBP specificeert bedrijfsprocessen op basis van 'rollen', zie [ebBP], terwijl voor de daadwerkelijke samenwerking specifieke informatie nodig is van de partijen, zoals een ebMS adres, een transport url, certificaten, etc.)
4. Voordat een organisatie of partij gebruik gaat maken van ebMS zal er een ebMS Deployment Profile document geschreven worden [ebMS-DPT]. Hierin worden een aantal keuzes gemaakt voor de diverse ebMS parameters. Het uiteindelijke document wordt ook wel aangeduidt met een 'profiel' voor die organisatie. Zo is er een JAB profiel voor Justitie en een OSB profiel voor de overheid servicebus van de GBO.Overheid.
5. Voor het creëren van een CPA wordt een profiel-template gebruikt op basis van een bepaald profiel (zoals die in hoofdstuk 2.3 genoemd zijn).

### 3. Specificatie

Voordat een CPA gemaakt kan worden zal de specificatie van de samenwerking (“Business Collaboration”) gemaakt moeten worden. Het proces waarin een dergelijke specificatie tot stand komt wordt buiten beschouwing gelaten in dit document. We richten ons op het resultaat.

#### 3.1 Service Identificatie

Elke specificatie van een samenwerkingsverband tussen twee of meer partijen moet uniek identificeerbaar zijn binnen de OSB. Dit wordt vooralsnog in dit document aangeduid met “ID”, maar het zal duidelijk zijn dat de echte vorm complexer is.

#### 3.2 Berichten specificatie

Volgens de ebXML standaard is het de bedoeling om op basis van een ebBP een CPA template te maken. In de praktijk wordt er (nog) niet veel met ebBP’s gewerkt en is dat nog niet mogelijk; er is daarom voor een vereenvoudigd formaat gekozen waarin de essentie wordt vastgelegd. Het is een xml document met een of meerdere message elementen. Voor elk element worden de “van” (“from”) en “naar” (“to”) attributen opgegeven. Dit zijn de rollen die de partijen innemen in de samenwerking. Daarnaast moet voor elk bericht het gekozen profiel worden opgegeven. Voor de OSB zijn dit of “osb-rm” voor reliable messaging of “osb-be” voor best-effort.

Een voorbeeld xml ziet er als volgt uit:

```
<?xml version="1.0" encoding="UTF-8"?>
<messages>
  <message from="Gemeente" to="GBA" profile="osb-be">BSNInformatieRequest</message>
  <message from="GBA" to="Gemeente" profile="osb-be">BSNInformatieResponse</message>
  <message from="GBA" to="Gemeente" profile="osb-rm">BSNAdresWijzigingNotificatie</message>
</messages>
```

Het is de afspraak om de naam van het xml bestand als volgt op te bouwen: **ID.msg.xml**, waarbij ID een uniek identificerende naam is.

Als uitbreiding wordt voorgesteld om een attribuut “xsdref” op te nemen voor de verwijzing naar de payload specificatie. Voor release 1 wordt deze functionaliteit nog niet geboden. (Voor de kenners: de verwijzing kan dan worden opgenomen in de “SimplePart” van de CPA, zie [CPPA].)

Er zou een mogelijkheid geboden kunnen worden om op basis van een WSDL een vertaling te maken naar het message formaat. Omdat er echter verschillende manieren zijn waarop een WSDL vorm gegeven kan worden, wordt gezien de eenvoud van het message formaat er voor gekozen geen geautomatiseerde ondersteuning te bieden voor het omzetten van een WSDL naar het message formaat.

#### 3.3 Parameter specificatie

De bericht specificatie wordt in principe afgeleid van de ebBP. Deze specificatie bevat echter geen informatie over de technische voorzieningen die nodig zijn voor de daadwerkelijke berichtenuitwisselen. Ook wordt in een ebBP niet specifiek aangegeven welke echte partijen, zoals de “Gemeente Soest” of “Gemeente Den Haag” meespelen. Deze informatie wordt gespecificeerd in de parameter specificatie.

De volgende parameters zijn mogelijk:

- CPAID. Een unieke identificatie van de samenwerking tussen twee partijen.
- StartDate. De start datum/tijdstip van de overeenkomst.
- EndDate. De eind datum/tijdstip van de overeenkomst.
- PartyName. De naam de partij die een bepaalde rol inneemt.
- PartyId. Het (logische) adres van de partij die een bepaalde rol inneemt. Binnen de OSB is dit het FI-nummer, eventueel gevolgd door het vestigingsnummer.
- PartyIdType. De typering van de PartyId. Voor de OSB is dit osb:fin:1:0
- ProcessSpecification. De naam van de proces specificatie waar de samenwerking op gebaseerd is.
- Service. De service aanduiding van de samenwerking.
- PartyRef. Een referentie naar informatie over de organisatie en de betreffende service.
- ClientCert. Het publieke deel van het client certificaat van de partij.
- ServerCert. Het publieke deel van het server certificaat van de partij.
- RootCert. De publieke delen van vertrouwde certificaten. Dit kan een keten van (elkaar) vertrouwde certificaten zijn.
- EndpointUri(HTTPS). De url van het messaging endpoint van de betrokken partij.

Voor elke parameter kan aangegeven worden of deze verplicht is of niet. Indien de parameter niet verplicht is (aangegeven met het attribuut "required=false") moet er een default waarde gegeven worden bij de specificatie.

Een voorbeeld van een parameter specificatie is hieronder gegeven.

```
<?xml version="1.0" encoding="UTF-8"?>
<parameters>
  <partner name="GBA">
    <parameter name="CPAID" required="false">03213</parameter>
    <parameter name="StartDate" required="false">2007-01-01T00:00:00Z</parameter>
    <parameter name="EndDate" required="false">2017-01-01T00:00:00Z</parameter>
    <parameter name="ProcessSpecification" required="false">BSN</parameter>
    <parameter name="Service" required="false">BSNServices</parameter>
    <parameter name="PartyName" required="false">Gemeente Basisadministratie</parameter>
    <parameter name="PartyId" required="false">FIN06525242</parameter>
    <parameter name="PartyIdType" required="false">osb:fin:1:0</parameter>
    <parameter name="PartyRef" required="false">http://www.gba.nl</parameter>
    <parameter name="EndpointUriHTTPS" required="false">https://gw.gba.nl</parameter>
  </partner>
  <partner name="Gemeente">
    <parameter name="PartyName" required="true"/>
    <parameter name="PartyId" required="true"/>
    <parameter name="PartyRef" required="false"/>
    <parameter name="PartyIdType" required="false">osb:fin:1:0</parameter>
    <parameter name="ClientCert" required="true"/>
    <parameter name="ServerCert" required="true"/>
    <parameter name="EndpointUriHTTPS" required="true"/>
  </partner>
</parameters>
```

Het is de afspraak om de naam van het xml bestand als volgt op te bouwen: **ID.params.xml**, waarbij ID een uniek identificerende naam is.

### 3.4 Parameter Waardes

Na de specificatie van de parameters zullen er waardes toegekend moeten worden aan deze parameters. Zoals eerder al eerder aangegeven kan er een default waarde gegeven worden tijdens de specificatie. Echter, in het algemeen kunnen meerdere partijen een bepaalde partij rol innemen, waardoor de mogelijkheid geboden moet worden om op een bepaald tijdstip een nieuwe CPA te maken op basis van de gegevens die de betrokken partij aanlevert.

Het aanleveren van de waardes voor de parameters gebeurt ook via een xml bestand. Om de invulling te vereenvoudigen wordt een template voorgesteld waarin alleen nog de waardes van de elementen ingevuld hoeven te worden.

Een voorbeeld van een parameter template bestand.

```
<?xml version="1.0" encoding="UTF-8"?>
<CPANegotiationRequest>
  <Parameters>
    <Parameter name="PartyId"></Parameter>
    <Parameter name="PartyName"></Parameter>
    <Parameter name="EndpointUriHTTPS"></Parameter>
    <Parameter name="ClientCert">
</Parameter>
    <Parameter name="ServerCert">
</Parameter>
  </Parameters>
</CPANegotiationRequest>
```

Een voorbeeld van een ingevulde parameter waardes bestand (de certificaten zijn weggelaten voor de leesbaarheid).

```
<?xml version="1.0" encoding="UTF-8"?>
<CPANegotiationRequest>
  <Parameters>
    <Parameter name="PartyId">FIN9562725</Parameter>
    <Parameter name="PartyName">Gemeente Den Haag</Parameter>
    <Parameter name="EndpointUriHTTPS">https://gw.denhaag.osb.nl</Parameter>
  </Parameters>
</CPANegotiationRequest>
```

Het is de afspraak om de naam van het xml bestand als volgt op te bouwen:

**osb.ndd.ID.cpa.ROL.params.xml**, waarbij ID een uniek identificerende naam is en ROL een rol is die de betreffende partij vervult. In het voorbeeld zou de naam kunnen worden:

**osb.ndd.ID.cpa.Gemeente.params.xml**.

### 3.5 Gegeneerde bestanden

Op basis van de hierboven beschreven gegevens worden twee soorten bestanden geeneerd. Dit zijn:

- Op basis van de service definitie ("ID.msg.xml") wordt het CPA template geeneerd.
- Op basis van de parameter defintie ("ID.params.xml") wordt een Negotiation Descriptor Document (NDD), zie [NDD], geeneerd.
- Tesamen met de parameter waardes wordt een CPA geeneerd, conform de [CPPA] standaard.

De formaten en betekenis van deze documenten wordt hier verder niet beschreven en is te vinden via de aangegeven referenties.

## 4. Creatie Proces

Het CPA creatie proces is een samenspel van interacties tussen een specifieke set van rollen. Elke rol heeft een bepaalde verantwoordelijkheid. Dit hoofdstuk specificeert de rollen en hun interacties. Merk op dat een specifieke partij één of meerdere rollen kan vervullen.

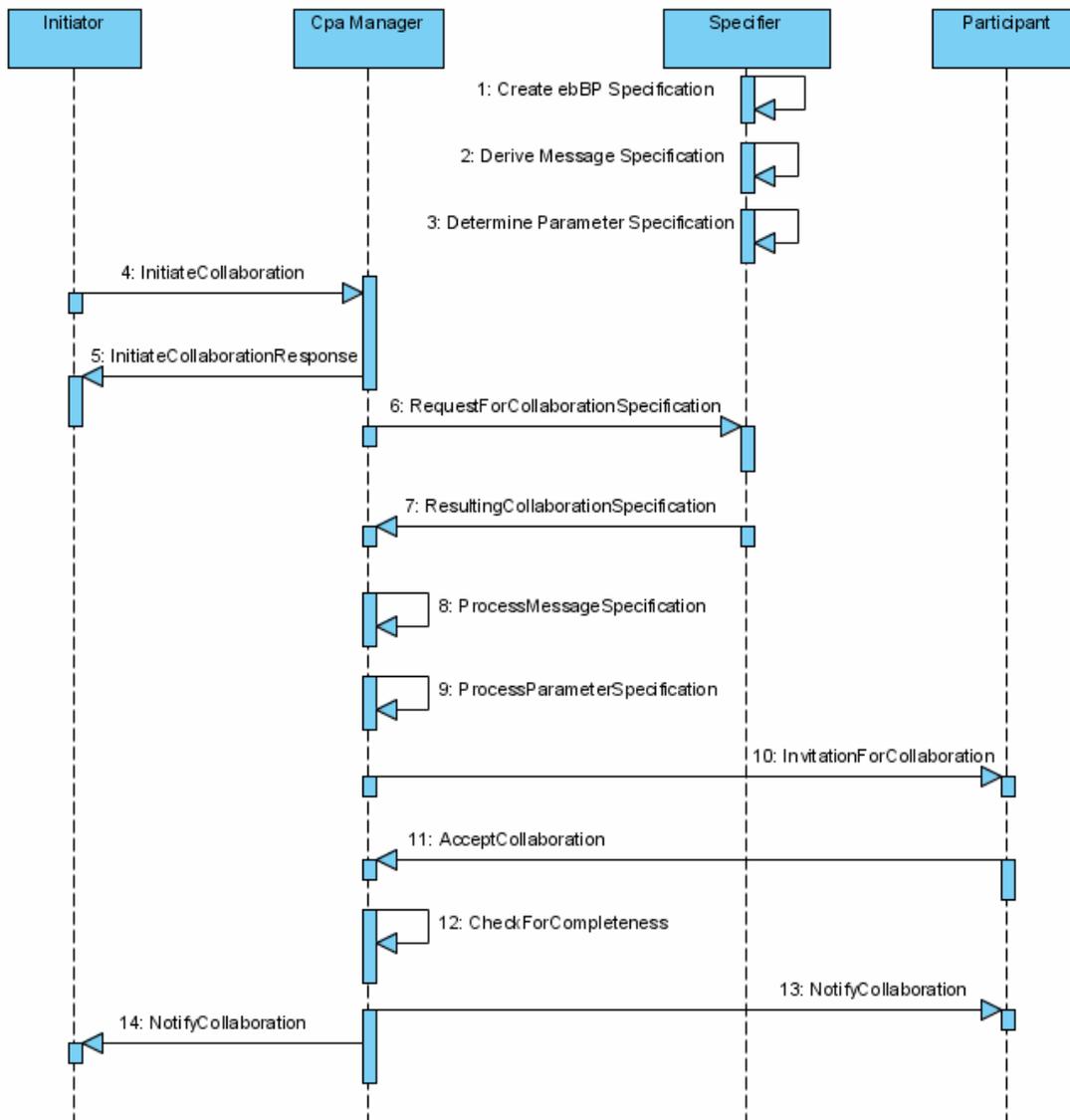
### 4.1 Rollen

Het proces waarin de CPA tot stand komt kent vier rollen:

- *Initiator*  
De initiator is degene die het proces laat beginnen. Er wordt vanuit gegaan dat de specificatie van de samenwerking reeds eerder tot stand is gekomen en er alleen maar naar verwezen hoeft te worden. De initiator is niet noodzakelijk een van de partijen die met elkaar willen samenwerken; de initiator kan ook een soort 'opdrachtgever' zijn.
- *Specificier*  
De specificier kan de specificaties leveren van de door de initiator aangewezen samenwerking. Op basis van de specificatie kan een CPA template en NDD worden afgeleid. Merk op dat er vanuit gegaan wordt dat de specificaties reeds beschikbaar zijn. Hoe deze specificaties tot stand komen wordt niet aangegeven in dit document.
- *Participant*  
Een participant is een deelnemer in de samenwerking. Omdat een samenwerking uit minimaal 2 rollen bestaat zijn er dus minimaal 2 partijen die de rol van participant zullen vervullen.
- *CpaManager*  
De CpaManager coördineert het proces en levert uiteindelijk de gevraagde CPA's op. Er kunnen meerdere CPA's opgeleverd worden, afhankelijk van de gewenste samenwerking en de partijen.

## 4.2 Sequence Diagram

Het onderstaande sequence diagram geeft de verschillende interacties weer tussen de eerder genoemde rollen.



Voor elk van de getekende interacties zal uitgelegd worden wat ze betekenen.

*1: Create ebBP Specification*

*2: Derive Message Specification*

*3: Determine Parameter Specification*

Zoals eerder aangegeven gaat het CPA creatie proces er vanuit dat er reeds een ebBP is. Deze interactie geeft uitdrukking aan het feit dat een deel van de informatie uit een ebBP beschikbaar gemaakt moet worden, voorlopig uitgedrukt in een (eenvoudig) *message specification* bestand (ID.msg.xml), en de specificatie van de verschillende parameters (ID.params.xml). De specifier is in het algemeen de partij die eind- of hoofdverantwoordelijk is voor de gewenste samenwerking.

#### 4: *InitiateCollaboration*

De initiator legt vast welke partijen gebruik willen maken van de rollen, zoals die in de bericht specificatie (of ebBP) zijn vastgelegd. Omdat partijen binnen de OSB geïdentificeerd worden met hun FI-nummer, zal er dus een relatie gelegd worden tussen rol en FI-nummer. De initiator zal aangeven voor welke samenwerking er een CPA gemaakt moet worden. Dit is de unieke ID zoals die eerder genoemd is (en in dit document alleen met 'ID' wordt weergegeven). Daarnaast geeft de initiator aan welke partij de specificatie moet leveren (verkregen door in de Directory te zoeken naar de gewenste service; deze stap is niet getoond in het sequence diagram).

#### 5: *InitiateCollaborationResponse*

Als de CPA Manager in staat is om het proces te starten zal er een positieve response gegeven worden aan de initiator. Hierbij wordt er een referentie mee teruggegeven waarmee het proces gevolgd kan worden.

#### 6: *RequestForCollaborationSpecification*

Omdat de CPA Manager zelf de specificatie niet kent, zal deze gevraagd worden aan de specificier (die door de initiator aangegeven is).

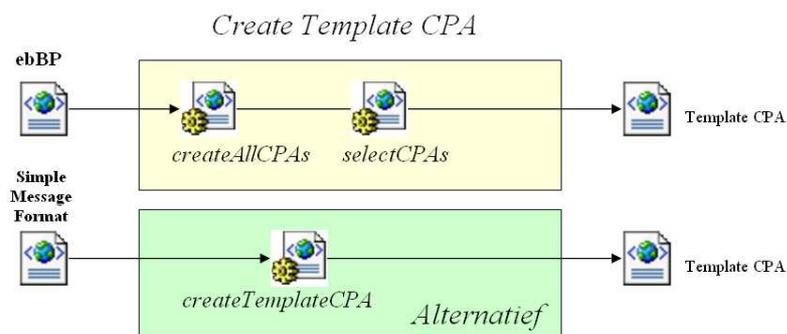
#### 7: *ResultingCollaborationSpecification*

De specificier zal de volgende specificatie leveren:

- De bericht specificatie (ebBP of de ID.msg.xml).
- De parameter specificatie, inclusief template bestand voor de participanten.

#### 8: *ProcessMessageSpecification*

Op basis van de bericht specificatie zal de CPA template afgeleid worden. Zoals eerder aangegeven wordt deze opgebouwd op basis van een profiel. Het onderstaande diagram geeft de (XSLT) transformatie weer. In principe is het de bedoeling om op basis van een ebBP de CPA template te maken, maar voor release 1 is dit alleen nog mogelijk op basis van het "Simple Message Format" (de ID.msg.xml).



#### 9: *ProcessParameterSpecification*

Op basis van de parameter specificatie worden een Negotiation Descriptor Document (NDD) gemaakt en een transformatie script (XSLT) waarmee op een later tijdstip de parameter waardes ingevuld kunnen worden in het CPA template.

#### 10: InvitationForCollaboration

De participanten (zoals die zijn aangegeven door de initiator) worden gevraagd mee te doen in de collaboration. Hierbij worden de specificaties meegestuurd.

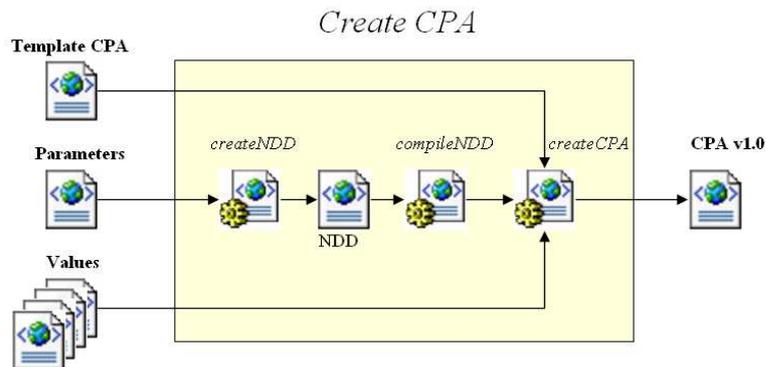
#### 11: AcceptCollaboration

Als een participant meedoet in de collaboration worden de parameter waardes gegeven voor de betreffende parameter. Als een participant niet mee wil doen in een collaboration, zal deze een *DeclineCollaboration* bericht sturen (niet getoond in het sequence diagram) waarna het gehele proces stopgezet zal worden door de CPA Manager.

#### 12: CheckForCompleteness

In een collaboration kunnen twee of meer partijen deelnemen. De CPA Manager zal pas de CPA's kunnen maken als alle partijen gereageerd hebben en hun gegevens aangeleverd hebben.

Als alle gegevens ontvangen zijn, kunnen de CPA's gemaakt worden. Het onderstaande diagram geeft de transformatie weer nadat alle parameter bekend zijn van de verschillende participanten:



#### 13: NotifyCollaboration

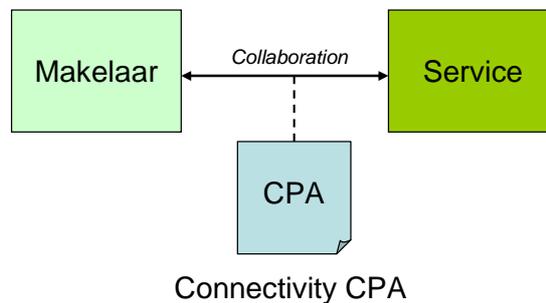
De resulterende CPA's worden naar de partijen gestuurd. Partijen zullen alleen die CPA's ontvangen die voor hun van belang zijn. De initiator wordt ook genotificeerd en ontvangt daarbij alle parameters en CPA's.

Als het gehele proces is afgerond kan de CPA Manager alle informatie verwijderen. Het is in principe de verantwoordelijkheid van de verschillende partijen om de relevante informatie te bewaren.

### 4.3 Connectivity CPA

Omdat veel services een standaard interface aan zullen bieden, en het aantal afnemers groot zal zijn, is een geautomatiseerde ondersteuning voor het creëren (en beheren) van een CPA wenselijk. Tenzij elke Service Provider (SP) een standaard service voorziet voor elke willekeurige partner, kan een Service Requester (SR) geen geautomatiseerde koppeling maken met een SP. Het gevolg is dat zo'n "afstemmings CPA" alle SP en SR adressen moeten bevatten. Het beheren van deze CPA's is complex omdat zo'n CPA bij elke Service geïnstalleerd moet worden. Voor elke nieuw aan te sluiten partner op de OSB zullen alle reeds aangesloten partners een update moeten krijgen van de "afstemmings CPA".

Om dit te voorkomen wordt er een standaard "**Connectivity CPA**" voorgesteld met een nader uit te werken "*OSB CPA Manager*" waarmee services aangesloten worden op de OSB. Het gebruik van deze CPA is daarmee onderdeel geworden van de "Technische Aansluit Voorwaarden" op de OSB.



## 5. Specialisaties

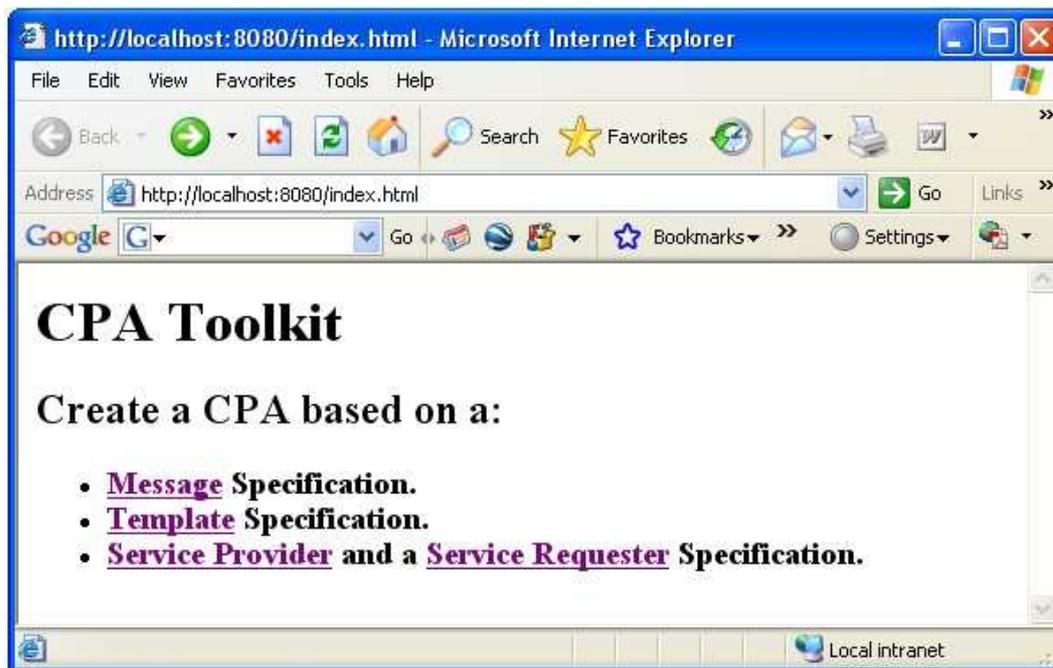
Voor een aantal gevallen kan er een vereenvoudiging aangebracht worden in het cpa creatie proces. De vereenvoudiging is gebaseerd op de specialisatie van de rollen:

- Er is sprake van twee partijen waarbij alle rollen door één van de partijen uitgevoerd wordt.
- Er is slechts sprake van twee partijen waarbij de initiator en de specifier verschillende partijen zijn en waarbij de specifier maar éénmalig zijn gegevens wil verstrekken.

Voor deze specialisaties zijn er twee web interfaces gemaakt:

- Eén webinterface waarbij er voor twee willekeurige partijen een cpa gemaakt wordt.
- Twee webinterfaces, één voor de specifier (aangeduid met Service Provider) en één voor de initiator (aangeduid met Service Requester).

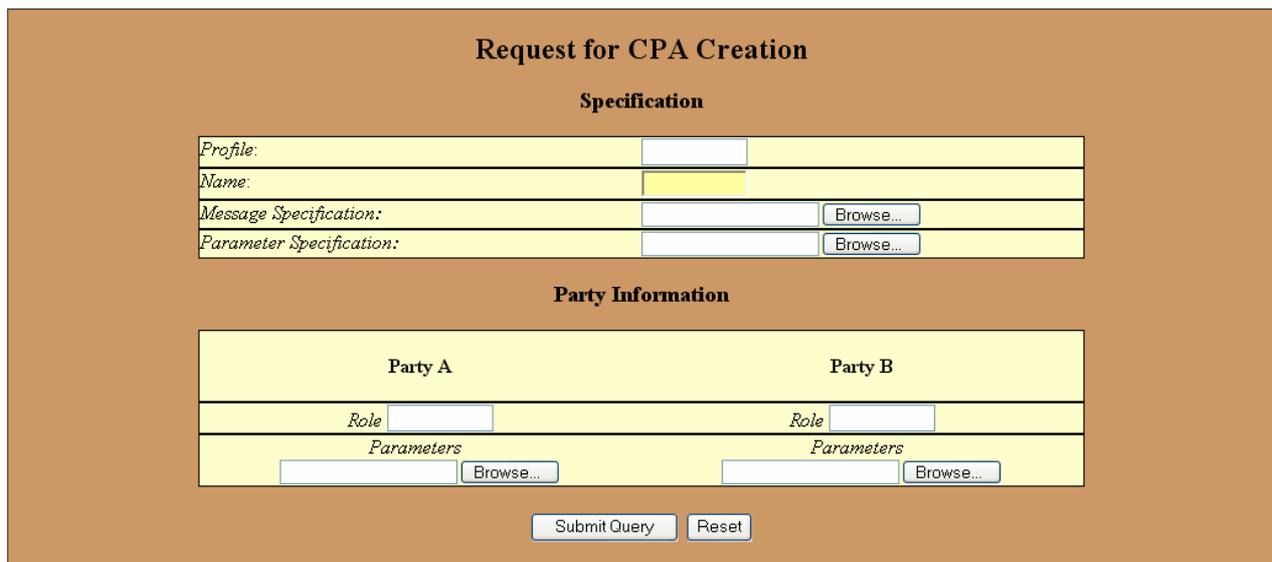
Hieronder worden voorbeelden gegeven van de genoemde webinterfaces. De implementatie maakt gebruik van Apache. Als de url localhost is met poort 8080, dan zijn deze scripts aan te roepen met **`http://localhost:8080/index.html`**



## 5.1 CPA Creatie op basis van een bericht specificatie

Met behulp van één webinterface wordt een CPA voor twee willekeurige partijen, op basis van de specificatie en parameter waardes. Hieronder wordt de interface getoond.

Voorbeeld url: <http://localhost:8080/singleCpa.html>



Request for CPA Creation	
Specification	
Profile:	<input type="text"/>
Name:	<input type="text"/>
Message Specification:	<input type="text"/> <input type="button" value="Browse..."/>
Parameter Specification:	<input type="text"/> <input type="button" value="Browse..."/>
Party Information	
Party A	Party B
Role <input type="text"/>	Role <input type="text"/>
Parameters <input type="text"/> <input type="button" value="Browse..."/>	Parameters <input type="text"/> <input type="button" value="Browse..."/>
<input type="button" value="Submit Query"/> <input type="button" value="Reset"/>	

Het veld "Profile" geeft het profiel aan dat gekozen is (voor de OSB is dat 'osb'). Het "Name" veld moet de unieke identificatie bevatten van de service waarvoor een CPA gemaakt moet worden (in de voorbeelden aangeduid met 'ID'). De "Message Specification" veld moet het bestand zijn waar de berichten in gespecificeerd staan ('ID.msg.xml'). Het "Parameter Specification" veld moet het bestand zijn waar de parameters in gespecificeerd staan ('ID.params.xml').

Omdat elke participant in de samenwerking ("Partij" in de interface) een specifieke rol vervult zal dit moeten worden opgegeven. Elke partij zal ook op basis van de eerder genoemde parameter specificatie een aantal waardes moeten leveren. De waardes worden vastgelegd in een bestand die via de interface opgegeven worden. Als de aanvraag verstuurd is zal de CPA Manager de CPA creëren en beschikbaar stellen als 'download' in het resultaat scherm.

### Voorbeeld

Een voorbeeld is hieronder gegeven. Het betreft de specificatie van berichten voor het opvragen van BSN informatie bij het GBA. Elke gemeente kan dit doen, waardoor er dus voor elke samenwerking een CPA nodig is (tussen het GBA en de betreffende gemeente).

Het profiel is: osb.

De 'unieke' naam is: BSN.

De berichten staan gespecificeerd in: BSN.msg.xml.

De parameters staan gespecificeerd in BSN.params.xml.

De parameter waardes voor de rol GBA staan in: osb.ndd.BSN.cpa.GBA.parameters.xml.

De parameter waardes voor de rol Gemeente staan in:  
 osb.ndd.BSN.cpa.Gemeente.parameters.xml.

Request:

### Request for CPA Creation

#### Specification

Profile:	<input type="text" value="osb"/>
Name:	<input type="text" value="BSN"/>
Message Specification:	<input type="text" value="input\BSN\BSN.msg.xml"/> <input type="button" value="Browse..."/>
Parameter Specification:	<input type="text" value="it\BSN\BSN.params.xml"/> <input type="button" value="Browse..."/>

#### Party Information

Party A	Party B
Role <input type="text" value="GBA"/>	Role <input type="text" value="Gemeente"/>
Parameters <input type="text" value="pa.GBA.parameters.xml"/> <input type="button" value="Browse..."/>	Parameters <input type="text" value="meente.parameters.xml"/> <input type="button" value="Browse..."/>

Response:

### Response of CPA Toolkit

#### Specification Summary

Profile:	osb
Name:	BSN

#### Party Information Summary

Party A	Party B
Role GBA	Role Gemeente

**Result: [CPA](#)**



## 5.2 CPA Creatie op basis van een CPA template

In een aantal gevallen kan het nodig zijn om zelf een CPA template te maken, in plaats van deze te laten genereren op basis van bericht specificatie (msg.xml bestand).

De stappen zijn gelijk aan de stappen zoals die beschreven zijn in het vorige hoofdstuk, met één uitzondering: in plaats van een msg.xml bestand moet het CPA template bestand worden opgegeven. Dit blijkt ook uit de interface.

### Request for CPA Creation

**Specification**

Profile:	<input type="text"/>	
Name:	<input type="text"/>	
Template CPA Specification:	<input type="text"/>	<input type="button" value="Browse..."/>
Parameter Specification:	<input type="text"/>	<input type="button" value="Browse..."/>

**Party Information**

Party A		Party B	
Role	<input type="text"/>	Role	<input type="text"/>
Parameters	<input type="text"/> <input type="button" value="Browse..."/>	Parameters	<input type="text"/> <input type="button" value="Browse..."/>

Voor het kunnen gebruiken van een template moet er een correcte relatie gelegd worden tussen de **placeholders** in het template en de **parameters** die opgegeven worden.

Hieronder worden de placeholders gegeven die gebruikt moeten worden voor de verschillende parameters. Hierbij moeten <Role> vervangen worden door de rolnaam zoals die gebruikt worden in de invoer velden voor partij A resp. Partij B:

- CPAID: CPAID\_<RoleA>\_<RoleB>\_PH
- PartyName: <Role>\_PartyName\_PH
- PartyId: <Role>\_PartyName\_PH
- PartyIdType: <Role>\_PartyIdType
- ProcessSpecification: geen placeholder nodig.
- Service: geen placeholder nodig.
- PartyRef: <Role>\_PartyRef
- SigningCert: <Role>\_SigningCert
- EncryptionCert: <Role>\_EncryptionCert
- ClientCert: <Role>\_ClientCert
- ServerCert: <Role>\_ServerCert
- RootCert: <Role>\_RootCert
- EndpointUri: <Role>\_TransportUrl\_PH

### 5.3 CPA Creatie voor één Service Provider en meerdere Service Requesters

In dit speciale geval zal een specifieke Service Provider eenmalig de specificatie en zijn parameter waarden opgeven. De Service Requester zal steeds een andere partij zijn en zal zijn gegevens in een ander scherm opgeven.

Voorbeeld url voor de provider: <http://localhost:8080/singleCpaSP.html>

**Request for CPA Creation**

**Specification**

<i>Profile:</i>	<input type="text"/>
<i>Name:</i>	<input type="text"/>
<i>Message Specification:</i>	<input type="text"/> <input type="button" value="Browse..."/>
<i>Parameter Specification:</i>	<input type="text"/> <input type="button" value="Browse..."/>

**Party Information**

Service Provider	Service Requester
<i>Role</i>	<i>Role</i>
<input type="text"/>	<input type="text"/>
<i>Parameters</i>	
<input type="text"/> <input type="button" value="Browse..."/>	

Voorbeeld url voor de Service Requester: <http://localhost:8080/singleCpaSR.html>

**Request for CPA Creation**

**Specification**

<i>Profile:</i>	<input type="text"/>
<i>Name:</i>	<input type="text"/>

**Party Information**

Service Provider	Service Requester
<i>Role</i>	<i>Role</i>
<input type="text"/>	<input type="text"/>
<i>Parameters</i>	
<input type="text"/> <input type="button" value="Browse..."/>	

Als de Service Requester zijn gegevens gegeven heeft zal de CPA Manager de CPA maken en beschikbaar stellen als 'download'.

#### 5.4 Python scripts gebruiken

Na installatie kan de gebruiker de Python scripts gebruiken vanaf de command-line. De “current directory” (CD) moet dan de installatie folder zijn uitgebreid met:

```
cpatoolkit\software\python
```

Als de stappen in hoofdstuk **Error! Reference source not found.** uitgevoerd zijn is niet nodig, maar zal er in de eigen scripts een import gedaan moeten worden met de juiste folder.

***Voorbeeld.***

De installatie is gedaan in:

```
D:\gbo.overheid\osb\ssm\
```

Dan moet de PYTHONPATH uitgebreid worden met:

```
D:\gbo.overheid\osb\ssm\software\python\
```

## A. Referenties

- [ebBP] OASIS ebXML BPSS.  
URL <http://docs.oasis-open.org/ebxml-bp/2.0.3/ebxmlbp-v2.0.3-Spec-cs-en-pdf.zip>.
- [ebBP2CPA] ebXML Business Process (ebBP) to ebXML Collaboration Protocol Profile and Agreement (ebCPPA) transformation project.  
URL <http://dev.havanawave.com/repos/ebBP2ebCPPA/>.
- [ebCPPA] OASIS ebXML Collaboration Protocols and Agreements. OASIS Standard.  
URL <http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0.pdf>
- [ebCPAinstant] NDD, negotiation/instantiation, CPA template update. Posting to ebXML-CPPA mailing list.  
URL <http://lists.oasis-open.org/archives/ebxml-cppa/200609/msg00004.html>
- [ebCPAnegot] Automated Negotiation of Collaboration Protocol Agreements. Version 0.10. OASIS ebXML Collaboration Protocols and Agreements TC.
- [ebMS] OASIS ebXML Message Service Specification v2.0  
URL [http://www.oasis-open.org/committees/download.php/272/ebMS\\_v2\\_0.pdf](http://www.oasis-open.org/committees/download.php/272/ebMS_v2_0.pdf)
- [ebMS3] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features.  
URL [http://www.oasis-open.org/apps/org/workgroup/ebxml-msg/download.php/20556/ebms\\_core-3.0-spec-wd-15.pdf](http://www.oasis-open.org/apps/org/workgroup/ebxml-msg/download.php/20556/ebms_core-3.0-spec-wd-15.pdf)
- [ePV-ebBP] ePV Feedback. An example ePV process using the v2.0.1 ebBP draft specification.  
URL [http://www.oasis-open.org/committees/document.php?document\\_id=16436&wg\\_abbrev=ebxml-bp](http://www.oasis-open.org/committees/document.php?document_id=16436&wg_abbrev=ebxml-bp).
- [JAB] Justitiestandaard Asynchrone Berichtenuitwisseling.
- [ebBP] ebXML Business Process Specification Schema, Technical Specification v2.0.3, Committee Specification, 28 April 2006.
- [CPPA] Collaboration-Protocol Profile and Agreement Specification, Version 2.0, September 23, 2002.
- [ebMS-DPT] Deployment Profile Template, For OASIS ebXML Message Service 2.0, Committee Draft 1.1, 20 June 2005.
- [NDD] Automated Negotiation of Collaboration Protocol Agreements Specification, Version 0.04, 27 January 2003.
- [BBO-MPM] Projectvoorstel Modelgestuurd Partner Management en Berichtenuitwisseling, 22-11-2006, versie 0.2.