

# Digikoppeling Best Practices ebMS2 3.2.1

Logius Best practice

Vastgestelde versie 11 april 2022

**Deze versie:**

<https://publicatie.centrumvoorstandaarden.nl/dk/bpebms/3.2.1>

**Laatst gepubliceerde versie:**

<https://publicatie.centrumvoorstandaarden.nl/dk/bpebms/>

**Laatste werkversie:**

<https://logius-standaarden.github.io/Digikoppeling-Best-Practices-ebMS2/>

**Vorige versie**

<https://publicatie.centrumvoorstandaarden.nl/dk/bpebms/3.2/>

**Redacteurs:**

[Peter Haasnoot](#) (Logius)

[Pieter Hering](#) (Logius)

**Auteur:**

[Logius](#)

**Doe mee:**

[GitHub Logius-standaarden/Digikoppeling-Best-Practices-ebMS2](#)

[Dien een melding in](#)

[Revisiehistorie](#)

[Pull requests](#)

This document is also available in this non-normative format: [pdf](#)

This document is licensed under a [Creative Commons Attribution 4.0 License](#).

---

## Samenvatting

Alle Digikoppeling profielen die op ebMS2 gebaseerd zijn, moeten zich conformeren aan de Koppelvlakstandaard ebMS2. Dit document is een aanvulling hierop. Het heeft als doel ontwikkelaars te adviseren en te informeren over de huidige werkwijze bij het toepassen van Digikoppeling Koppelvlakstandaard ebMS2 – deze informatie geldt dus alleen voor de ebMS2-variant.

## Status van dit document

Dit is de definitieve versie van de best practice. Wijzigingen naar aanleiding van consultaties zijn doorgevoerd.

## Inhoudsopgave

### Samenvatting

### Status van dit document

#### 1. Inleiding

- 1.1 Doel en doelgroep
- 1.2 Opbouw Digikoppeling documentatie
- 1.3 Doel en scope van Digikoppeling
  - 1.3.1 Uitwisseling binnen Digikoppeling

1.4 Opbouw van dit document

## **2. Werkwijze/Aanbevelingen/Best Practices**

2.1 EB001 ebMS2 Producten

2.2 EB002 CPA Gebruik

2.3 EB003 Productie- en ontwikkelomgevingen

2.4 EB004 PartyId postfix

2.5 EB005 Certificaten

2.6 EB006 Service & Action naamgeving

2.7 EB007 Rollen

2.8 EB008 Overdrachtskarakteristieken

2.9 EB009 Vaststelling CPAId

2.10 EB010 Geldigheidsperiode van een CPA

2.11 EB011 MessageOrder en ConversationId

2.12 EB012 MessageOrder en ReliableMessaging

2.13 EB013 MessageId

2.14 EB014 Meerdere PartyId's

2.15 EB015 SyncReplyMode

2.15.1 Voorwaarden voor toepassen van het synchrone bevestiging

2.16 EB016 Correlatie van berichten

2.16.1 Response op een specifiek request

2.16.2 Response binnen een conversatie

## **3. CPA Gebruik en Kenmerken**

3.1 Inleiding

3.2 Wat is een CPA?

3.3 Waarom wordt er een CPA gebruikt?

3.4 Wat zijn de uitgangspunten voor de CPA?

3.5 Hoe wordt een CPA gemaakt?

## **4. Het gebruik van berichtvolgordelijkheid**

4.1 Granulariteit

4.2 Verwerking in de organisatie

4.3 Alternatieven voor berichtvolgorde

## **5. Message Ordering in ebXML**

5.1 MessageOrder Module\*\*

5.2 MessageOrder Element\*\*

5.3 Productondersteuning

5.4 Zelfbouwoverwegingen

5.5 Ontwerp Pattern

5.5.1 Specificatie (Design Time)

5.5.2 Verwerking (Run Time)

## **6. Conformiteit**

## **7. Lijst met figuren**

## **A. Referenties**

A.1 Normatieve referenties

A.2 Informatieve referenties

# Documentbeheer

Datum	Versie	Auteur	Opmerkingen
-------	--------	--------	-------------

22-11-2011	1.5	Logius	-
------------	-----	--------	---

Datum	Versie	Auteur	Opmerkingen
09-06-2014	1.6	Logius	Redactioneel bijwerken
13-01-2015	3.0	Logius	Redactioneel
01-10-2017	3.1	Logius	Herzien nav: - Compliance Voorziening - Begrippen ebMS2 en CPA Register
16-05-2019	3.2	Logius	EB015 SyncReply toegevoegd
11/04/2022	3.2.1	Logius	Vermelding REST-API koppelvlak

## Colofon

Postbus 96810 2509  
 JE Den Haag  
 Logius Servicecentrum: t. 0900 555 4555 (10 ct  
 p/m)  
 e.[servicecentrum@logius.nl](mailto:servicecentrum@logius.nl)

## 1. Inleiding

### 1.1 Doel en doelgroep

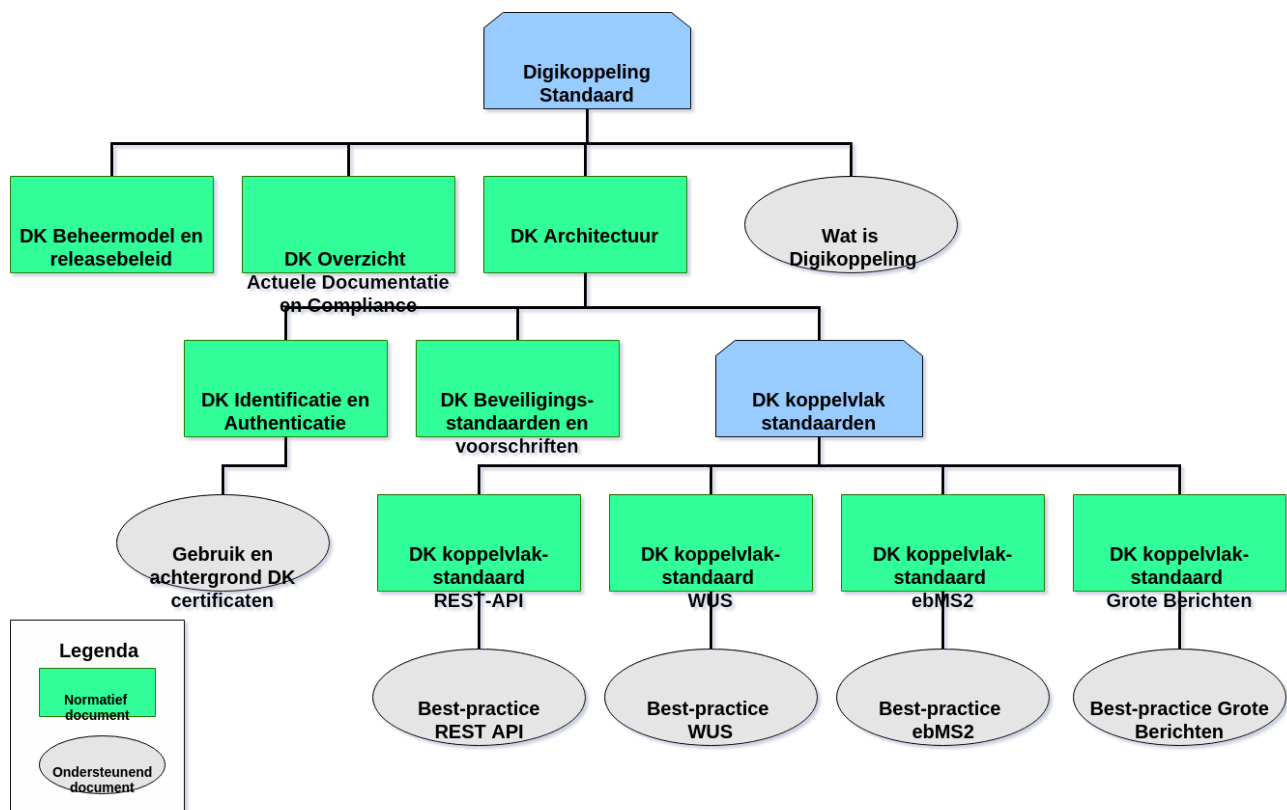
Alle Digikoppeling profielen die op ebMS2 gebaseerd zijn, moeten zich conformeren aan de Koppelvlakstandaard ebMS2. Dit document is een aanvulling hierop. Het heeft als doel ontwikkelaars te adviseren en te informeren over de huidige werkwijze bij het toepassen van Digikoppeling Koppelvlakstandaard ebMS2 – deze informatie geldt dus alleen voor de ebMS2-variant.

Het document is bestemd voor Architecten, integratie specialisten en ontwikkelaars van webservices die zijn aangesloten op Digikoppeling. Het gaat hierbij om zowel (service) aanbieders als (service) afnemers. Zie onderstaande tabel bij welke taken dit document ondersteunt.

Afkorting	Rol	Taak	Doelgroep?
[MT]	Management	Bevoegdheid om namens organisatie (strategische) besluiten te nemen.	<b>Nee</b>
[PL]	Projectleiding	Verzorgen van de aansturing van projecten.	<b>Nee</b>
[A&D]	Analyseren & ontwerpen (design)	Analyseren en ontwerpen van oplossings-richtingen. Het verbinden van Business aan de IT.	<b>Nee</b>
[OT&B]	Ontwikkelen, testen en beheer	Ontwikkelt, bouwt en configureert de techniek conform specificaties. Zorgen voor beheer na ingebruikname.	<b>Ja</b>

### 1.2 Opbouw Digikoppeling documentatie

Digikoppeling is beschreven in een set van documenten. Deze set is als volgt opgebouwd:



Figuur 1 Opbouw documentatie Digikoppeling

## ► Tekstalternatief

Alle met de kleur groen aangegeven documenten vallen onder het beheer zoals geformaliseerd in het Beheermodel en releasebeleid.

### 1.3 Doel en scope van Digikoppeling

Voor de Overheid als geheel is interoperabiliteit tussen een groot aantal serviceaanbieders en serviceafnemers van essentieel belang. Die grootschalige interoperabiliteit wordt bereikt door sterke standaardisatie van het koppelvlak tussen de communicatiepartners.

Deze communicatie vindt plaats in het domein van Digikoppeling, en daarbij worden Digikoppeling Koppelvlakstandaarden toegepast. Dat is een beperkte set van standaarden waaruit onder gedefinieerde omstandigheden gekozen kan worden.

Digikoppeling biedt de mogelijkheid om op deze gestandaardiseerde wijze berichten uit te wisselen tussen service aanbieders en service afnemers. Digikoppeling richt zich voornamelijk op uitwisselingen tussen overheidsorganisaties maar er zijn ook implementaties bekend van de toepassing van Digikoppeling bij uitwisseling tussen bedrijven en overheden.

#### 1.3.1 Uitwisseling binnen Digikoppeling

De uitwisseling tussen partijen is in drie lagen opgedeeld:

- Inhoud (gegevens): deze laag bevat afspraken over de inhoud van het uit te wisselen bericht, dus de structuur, semantiek en waardebereiken  
Digikoppeling houdt zich niet met de inhoud bezig, 'heeft geen boodschap aan de boodschap'.

- Logistiek (processen): op deze laag bevinden zich de afspraken betreffende transportprotocollen (HTTP/TLS), messaging (SOAP), adressering, beveiliging (authenticatie en encryptie) en betrouwbaarheid. Dit is laag van Digikoppeling.
- Transport (techniek): deze laag verzorgt het daadwerkelijke transport van het bericht (TCP/IP).

Digikoppeling richt zich uitsluitend op de logistieke laag.

## 1.4 Opbouw van dit document

-

Hoofdstuk 1 bevat een aantal algemene inleidende onderwerpen.

Hoofdstuk 2 bevat de aanbevelingen, werkwijze en best practices.

Hoofdstuk 3 gaat in op de kenmerken van een CPA.

Hoofdstuk 4 gaat over bericht volgordelijkheid.

Begrippen en afkortingen worden toegelicht in het document "DigikoppelingArchitectuur". Deze zit in de Digikoppeling standaard.

Dit document en andere documentatie is beschikbaar op [www.logius.nl/digikoppeling](http://www.logius.nl/digikoppeling).

## 2. Werkwijze/Aanbevelingen/Best Practices

### 2.1 EB001 ebMS2 Producten

-

Gebruik een ebMS2 product met aantoonbare ervaring binnen het Digikoppeling domein. Op de website van Logius is een overzicht van Digikoppeling leveranciers te vinden en ook in de marktscan van KING en Logius is een overzicht te vinden van het aanbod van Digikoppeling adapters dat voldoet aan de eisen die KING stelt aan een Digikoppeling Adapter voor de gemeentelijke markt.

### 2.2 EB002 CPA Gebruik

-

Voor het definiëren van een CPA geldt het volgende advies:

1. Raadpleeg het hoofdstuk 3 'CPA Gebruik en Kenmerken'.
2. Maak gebruik van het online CPA Register<sup>1</sup>, zie document 'Digikoppeling CPA Creatie Handleiding'.

<sup>1</sup>: Het online CPA Register wordt gehost door Justid en is beschikbaar op <https://cparegister.minvenj.nl/logius>

### 2.3 EB003 Productie- en ontwikkelomgevingen

-

Het is raadzaam om test- en productieservices ('OTAP') op aparte machines onder te brengen om het

onderscheid tussen beide helder te houden. Geef ze een eigen DNS naam (en dus verschillende PKI overheid certificaten), bijvoorbeeld door het gebruik van verschillende subdomeinnamen.

## 2.4 EB004 PartyId postfix

Voorzie de PartyId van een postfix voor het onderscheid tussen test- en productieservices ('OTAP'). De naamgevingsconventie is hierbij:

- Ontwikkelomgeving met de postfix '\_O'
- Testomgeving met de postfix '\_T'
- Acceptatieomgeving met de postfix '\_A'
- Productieomgeving zonder postfix (het oorspronkelijke nummer).

Samenstellingen zijn ook mogelijk, bijvoorbeeld de postfix '\_OTA' als er één specifiek adres gebruikt wordt voor de ontwikkel-, test-, en acceptatieomgeving. Aangezien Digikoppeling een strikte scheiding tussen test en productie nastreeft zou een combinatie van productie met andere omgevingen nooit moeten voorkomen<sup>2</sup>.

<sup>2</sup>: De scheiding komt ook tot uitdrukking in het gebruik van een andere certificaat-root voor productie en andere omgevingen. Zie hiervoor het document "Gebruik en achtergrond Digikoppeling-certificaten".

## 2.5 EB005 Certificaten

Vanuit het oogpunt van beveiliging is het dringende advies om aparte certificaten te gebruiken voor de 'OTA' omgevingen aan de ene kant en de productie ('P') omgeving aan de andere kant.

Ook wordt geadviseerd om autorisaties te baseren op het OIN dat uit het certificaat verkregen wordt. Indien autorisaties plaatsvinden met het PartyId (dat ook het OIN bevat) dient zeker gesteld te worden dat de beide voorkomens van het OIN (certificaat en PartyId) identiek zijn of dat de organisatie in het certificaat mag handelen namens de organisatie in het PartyId. Dit kan met geautomatiseerde controle (bijvoorbeeld bij ontvangst/verzending van een bericht op de TLS-offloader) of door handmatige controle (bijvoorbeeld bij het aanmaken van het CPA dat in de Digikoppeling-adapter ingelezen wordt).

## 2.6 EB006 Service & Action naamgeving

Advies 1: gebruik een functionele naam voor de naamgeving van de Service. Verwerk de versie in de service naam. De servicenaam mag de URN zijn.

### VOORBEELD 1

Voor Digimelding wordt een service gedefinieerd voor de verwerking van de berichten tussen de afnemer en Digimelding en tussen de registratiehouders en de Digimelding. De service krijgt de naam:

Digimelding:1:0

Advies 2: als er gebruik gemaakt wordt van het CPA Register, gebruik dan als Identificerende naam de naam van de service (het laatste onderdeel van het pad in de URN.)

### VOORBEELD 2

Voor Digimelding wordt een service gedefinieerd voor de verwerking van de berichten tussen de afnemer en Digimelding en tussen de registratiehouders en Digimelding. De service wordt opgeslagen in het CPA Register met de Identificerende naam:

Digimelding.1.0

Met deze Identificerende naam kan een afnemer een CPA laten maken op basis van de naam (zie 'Digikoppeling CPA Creatievoorziening Handleiding' op Digikoppeling website).

Advies 3: gebruik een functionele naam (liefst een werkwoord) voor de naamgeving van de Actions. Denk eraan dat een Service meerdere Actions mag bevatten (meldingen).

### VOORBEELD 3

Voor de Digimelding wordt in een service de Action gedefinieerd voor het terugmelden van een geconstateerde foutieve registratie. De naam voor de service is: terugmelden

In Digikoppeling Koppelvlakstandaard WUS staat bij voorschrift WW003 hoe de payload in de SOAP body opgenomen moet worden: op basis van de 'document-literal style'. Bij document-literal mag de payload slechts 1 XML element bevatten; hierbinnen kunnen wel meerdere elementen opgenomen worden. Het is ook bijvoorbeeld mogelijk om meerdere elementen van het type `{http://www.w3.org/2001/XMLSchema}` base64Binary op te nemen binnen dit eerste element. Daarmee ondersteunt deze koppelvlakstandaard het versturen van attachments met binaire data impliciet.

Het wordt sterk aangeraden om voor ebMS2 deze werkwijze over te nemen. De naam van het payload element zal dan gebruikt worden als naam voor de Action.

## 2.7 EB007 Rollen

Als een overheidsorganisatie in een bepaalde service zowel berichten kan versturen als wel berichten kan ontvangen, ga dan na wat de functionele rol is. In welke hoedanigheid wordt de functie uitgevoerd? Deze functionele rol zal een bepaalde naam hebben. Gebruik dan die naam voor de rol in de CPA.

#### VOORBEELD 4

Een abonnements service biedt de mogelijkheid om organisaties zich te laten inschrijven op een topic, of om zich te laten uitschrijven op een topic. De abonnements service zal op gezette tijden een nieuw item van een topic naar een afnemer sturen. Merk op dat de berichten in alle gevallen meldingen zijn.

Vanuit een eenvoudig oogpunt zou je kunnen zeggen dat de organisatie die de abonnements service implementeert, zowel berichten verstuurt als ontvangt:

- ontvangt, voor het verwerken van de aanvragen van de afnemer, en
- verstuurt, voor het verzenden van nieuwe topics.

Vanuit de optiek dat de organisatie een samenhangende verzameling van berichten gedefinieerd heeft voor de implementatie de abonnements service, is het zinvol om de organisatie die de service aanbiedt een en dezelfde rol te geven: bijvoorbeeld 'TopicHouder'.

De service krijgt de naam "AbonnementenService". Afnemers krijgen de rol 'Abonnee'. De organisatie die de service implementeert krijgt de rol 'TopicHouder'. De volgende meldingen zijn mogelijk:

- Van 'Abonnee' rol naar 'TopicHouder' rol: melding `InschrijvenOpTopic(topic)`
- Van 'Abonnee' rol naar 'TopicHouder' rol: melding `UitschrijvenOpTopic(topic)`
- Van 'Abonnee' rol naar 'TopicHouder' rol: bevraging `RaadpleegTopics()`
- Van 'TopicHouder' rol naar 'Abonnee' rol: melding `NieuwTopicItem()`

(Voor de volledigheid: het opvragen van de beschikbare topics is een 'bevraging' op Digikoppeling en zal met WUS gedaan moeten worden.)

De Abonnee kan dus berichten versturen (om zich in- of uit te schrijven), maar ook ontvangen (een nieuw item van een topic). De topic houder kan berichten ontvangen (de in of uitschrijvingen van afnemers), maar ook berichten versturen (de nieuwe items van een topic).

## 2.8 EB008 Overdrachtskarakteristieken

De karakteristieken voor de betrouwbare overdracht worden uitgedrukt in 'RetryInterval' en 'RetryCount'. Digikoppeling Koppelvlakstandaard ebMS2 heeft hiervoor een aanname gemaakt. Evalueer met de betrokken partijen of deze waarden van toepassing zijn. Wijzig de waarden zo nodig. Houd hierbij rekening met zowel de bedrijfsmatige verwerking van de meldingen als met de netwerk karakteristieken (beschikbare bandbreedte, omvang van de payload, quality of service en dergelijke) tussen de twee overheidsinstanties. Raadpleeg voor meer informatie hoofdstuk 3 'CPA Gebruik en Kenmerken'.

## 2.9 EB009 Vaststelling CPAId

Geef zowel een CPAId als een start- en einddatum op bij het maken van een overeenkomst tussen een service requester en een service provider. Deze invulling moet in overleg met de twee partijen bepaald worden. Raadpleeg voor meer informatie hoofdstuk 3 'CPA Gebruik en kenmerken'.

## 2.10 EB010 Geldigheidsperiode van een CPA



-

Laat de start- en einddatum van de CPA mede afhangen van de geldigheid van de gebruikte client- & servercertificaten.

Een CPA voor testdoeleinden kan een korte geldigheidsperiode krijgen, afgestemd op de test periode.

## 2.11 EB011 MessageOrder en ConversationId

-

Als de MessageOrder functionaliteit gebruikt wordt, moeten alle betreffende (samenhangende) berichten dezelfde ConversationId krijgen.

Er zijn meerdere, onafhankelijke, berichtstromen mogelijk waar MessageOrder op van toepassing is als elke berichtstroom zijn eigen ConversationId krijgt.

## 2.12 EB012 MessageOrder en ReliableMessaging

-

Als MessageOrder gebruikt wordt is dit van invloed op de overdrachtskarakteristieken (EB008). Als een enkel bericht in de overdracht faalt, heeft dit tot gevolg dat alle opvolgende berichten niet verzonden kunnen worden, of afgeleverd kunnen worden. De waardes voor de RetryCount en RetryInterval zullen daarom met zorg gekozen moeten worden.

## 2.13 EB013 Messageld

-

De Koppelvlakstandaard ebMS2 schrijft het gebruik van een Messageld voor conform RFC2822, in de vorm van "[UUID@URI](#)".

De URI in de Messageld kan ook het domein kan zijn van Digikoppeling messagehandler.

## 2.14 EB014 Meerdere PartyId's

-

Als een grote organisatie, bestaande uit meerdere deel-organisaties, via een gateway (in de algemene zin, niet te verwarren met Digikoppeling Gateway) aangesloten wordt op Digikoppeling, kan de situatie ontstaan dat de deel-organisaties een ander OIN hebben dan het OIN van de gehele organisatie. Ook kan de situatie ontstaan dat voor de cliënt authenticatie van de gateway er maar 1 certificaat gebruikt kan worden (namelijk die van de organisatie als geheel).

Het is toegestaan om op transport niveau het cliënt certificaat te gebruiken voor de authenticatie van de organisatie als geheel. Als waarde voor de PartyId mogen OIN's gebruikt worden die afwijken van het OIN in het authenticatie certificaat. Het is aan de samenwerkende partijen om er op toe te zien dat de juiste PartyId's gebruikt worden. In dergelijke situaties is het tevens toegestaan om PartyId's te gebruiken die afwijken van de gangbare OIN's, mits er een ander PartyId type aangegeven wordt. (Voor de OIN geldt urn:osb:oin.) Vanuit Digikoppeling wordt in dergelijke gevallen een dringend advies gegeven om de deel-organisaties een eigen OIN te laten aanvragen. (Het toestaan van een eigen PartyId waarde in de communicatie op Digikoppeling tussen de Nederlandse overheden wordt gezien als een tijdelijke situatie).

## 2.15 EB015 SyncReplyMode

Het Digikoppeling ebMS2 Reliable Messaging (RM) profiel vereist dat berichten bevestigd worden met een *acknowledgement* bericht. Binnen het Digikoppeling ebMS2 RM-profiel moet deze *acknowledgement* *asynchroon* worden verzonden. Een Digikoppeling oplossing houdt hiervoor een administratie bij zodat de *acknowledgement* aan het initiële *MessageId* van het bericht kan worden gerefereerd

Bij het gebruik van het uitwisselen van zeer hoge volumes van berichten in beperkte tijd via het Digikoppeling ebMS RM-profiel *kan* de overhead van het asynchroon bevestigen van het bericht via asynchrone *acknowledgement* (te) groot worden. Vandaar dat sinds versie 3.3 van dit profiel het gebruik van SyncReply Profile in de Digikoppeling ebMS2 Specificatie wordt toegestaan voor een aantal uitzonderingsgevallen. Vanaf deze versie is het in bepaalde gevallen mogelijk om een bericht synchroon – dus in dezelfde http-sessie- te bevestigen met een *acknowledgement* (of een foutsituatie synchroon te beantwoorden met een *error message*)

### Toepassen van Synchrone *acknowledgement* is uitzondering

Het gebruik van Syncreply wordt in de regel afgeraden. Tot aan versie 3.3 van Digikoppeling ebMS was een invulling van het attribuut SyncReplyMode in het CPA anders dan “none” niet toegestaan, met als onderbouwing: *Asynchronous messaging does not preclude fast response times, as is required to support interactive applications. Asynchronous messaging supports higher levels of scalability and supports scenarios where a response message may be sent minutes, hours or days after the initial request message. Asynchronous messaging may be combined transparently with store-and-forward intermediaries.*

In de praktijk blijkt dat er toch situaties bestaan waarin het synchroon bevestigen van een bericht voordeel biedt:

#### Casusbeschrijving

Bij uitwisseling tussen een GDI-voorziening en een grote afnemer worden door de verzendende partij aan de ontvangende partij in korte tijd zeer hoge volumes berichten aangeboden. Vanwege de eis op betrouwbaarheid is voor de uitwisseling gekozen voor het ebMS2 RM-profiel. Deze ebMS berichten bevatten per zending 1 functioneel bericht. Hierdoor is de overhead van het ebMS protocol voor de ontvangende partij relatief hoog.

De berichtuitwisseling in de keten was oorspronkelijk opgezet volgens asynchrone bevestiging. Bij de ketenintegratietest tussen beide partijen kwamen verwerkingsproblemen aan de ontvangende kant aan het licht. In onderling overleg tussen de ketenpartners is afgesproken om de ebMS2 uitwisseling zo in te stellen dat *acknowledgements* synchroon verstuurd worden. Hierna verliep de uitwisseling volgens de afgesproken eisen aan capaciteit, performance en betrouwbaarheid.

### 2.15.1 Voorwaarden voor toepassen van het synchrone bevestiging

- Scope: Dit profiel is alleen geldig voor de Digikoppeling ebMS2 RM-profielen
- Aanleiding: Door omvang van het volume van uitwisseling van berichten in beperkte tijd bestaan verwerkingsproblemen bij (een van beide) providers. Asynchrone uitwisseling van berichten binnen het ebMS profiel blijft de defaultmodus. Dus als de verwerking probleemloos verloopt is er geen reden om over te gaan op synchrone uitwisseling.
- Voorwaarde: De Digikoppeling oplossing van beide partijen ondersteunen het instellen van SyncReplymode op mshSignalsOnly. Het instellen van deze mode kan dus niet eenzijdig worden opgelegd.

Indien de berichtuitwisseling via een intermediary verloopt dient deze ook de SyncReplymode te ondersteunen om de synchrone communicatie tussen partijen mogelijk te maken.

## 2.16 EB016 Correlatie van berichten

Voor het correleren van berichten in ebMS2 spelen de elementen *MessageId*, *ConversationId* en *RefToMessageId* een rol. Het gebruik van de combinatie *MessageId* en *RefToMessageId* en het *ConversationId* dienen verschillende doelen.

### 2.16.1 Response op een specifiek request

*MessageId* en *RefToMessageId* worden gebruikt om heel specifiek aan te geven op welke request een response wordt gegeven. *RefToMessageId* wordt dus gebruikt om de relatie tussen een ebMS bericht en het daarop volgende ACK of NACK bericht aan te geven, een protocol gerelateerde relatie.

#### VOORBEELD 5

Voor een terugmelding op de GBA met *MessageId* 1234 moet de GBA in het antwoord op deze terugmelding het *RefToMessageId* 1234 gebruiken. Zo zijn request en response onlosmakelijk met elkaar verbonden. Normaal gesproken worden geen verdere antwoorden op een dergelijke melding terugverwacht waardoor het bericht met het *RefToMessageId* ook vaak de transactie beëindigt.

De invulling van het *RefToMessageId* element voor "normale" messages (dwz met business documenten) is afhankelijk van de processpecificaties die in een domein worden afgesproken. Als daarin namelijk wordt besloten om de relatie tussen vraag en antwoord alléén tot uitdrukking te brengen in de business documenten zelf, bijvoorbeeld in een SBDH, dan is het mogelijk om het *RefToMessageId* in het ebMS bericht waarin het antwoord is opgenomen weg te laten.

### 2.16.2 Response binnen een conversatie

Een *conversatie* gaat verder dan een simpele request/response. Een *conversation* is een gedefinieerd proces waarin meerdere berichten tussen twee partijen worden uitgewisseld. In alle ebMS berichten in een *instantie* van dat proces wordt dan hetzelfde *ConversationId* te gebruiken. De samenhang van de *individuele* transacties wordt bewaakt door het *MessageId* en *RefToMessageId* en de samenhang van het *proces* door het *ConversationId*. De waarde van 'ConversationId' wordt in bepaald door de Partij die het eerste bericht verstuurd, de initiator.

Het gebruik van het *ConversationId* is dus op *business-niveau*, met verschillende implementaties:

- Simpel vraagbericht met *ConversationId* waarop het antwoordbericht hetzelfde *ConversationId* bevat
- Transactieachtig gebruik, waarbij meerdere vraag- en antwoordberichten, hetzelfde *ConversationId* bevatten
- Uitwisselingen waarin het case nummer in *ConversationId* wordt gezet, dat kunnen meerdere opeenvolgende transacties zijn waarbij er ook enige tijd kan verstrijken tussen de transacties
- Random, geen relatie tussen vraag en antwoordbericht op basis van *ConversationId*

#### VOORBEELD 6

Binnen het Justitie-domein is een goed voorbeeld hiervan het proces van de identiteitsvaststelling, waarbij één verzoek resulteert in meldingen naar meerdere registers waarna uiteindelijk een gecombineerd antwoord wordt teruggestuurd naar de Politie. Bij alle meldingen in één identiteitsvaststelling wordt hetzelfde *ConversationId* gebruikt, maar uiteraard hebben ze allemaal wel unieke *MessageId*'s (met bijbehorende *RefToMessageId*'s voor de antwoorden).

In een losstaande transactie heeft het *ConversationId* dus niet zoveel toegevoegde waarde. Die toegevoegde

waarde onstaat pas op het moment dat meerdere transacties door middel van het overkoepelende ConversationId met elkaar verbonden worden.

## 3. CPA Gebruik en Kenmerken

### 3.1 Inleiding

Digikoppeling bevat de ebMS2 koppelvlakstandaard voor de overdracht van gegevens. Dit document gaat in op het gebruik van een Collaboration Protocol Agreement (CPA) in het geval dat ebMS2 gebruikt wordt voor de gegevensoverdracht.

### 3.2 Wat is een CPA?

Een CPA is een formeel xml document om de gebruikte functionele en technische eigenschappen van de ebMS2 protocol-karakteristieken vast te leggen. Het is dus een formele beschrijving voor het vastleggen van de gegevensuitwisseling.

De CPA is gestandaardiseerd in [ISO 15000-1: ebXML Collaborative Partner Profile Agreement (afgekort tot ebCPP<sup>3</sup>). Het ebMS2 protocol is gestandaardiseerd in ebXML Messaging Service Specification (afgekort tot ebMS<sup>4</sup>).

<sup>3</sup>: [\[ebCPP\]](http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0c.pdf) Collaboration-Protocol Profile and Agreement Specification Version 2.0, September 23, 2002. Url: <http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0c.pdf>

<sup>4</sup>: Message Service Specification, Version 2.0, 1 April 2002. Url: [http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS\\_v2\\_0.pdf](http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS_v2_0.pdf) [EBXML-MSG]

De eigenschappen van de gegevensoverdracht geven onder andere aan:

- tussen welke partijen er informatie uitgewisseld wordt;
- welke services en actions ('functies') er zijn waar de berichtuitwisseling op wordt gebaseerd;
- hoe certificaten gebruikt worden voor bijvoorbeeld transportbeveiliging, payload encryptie en/of ondertekening van berichten;
- wat de overdrachts karakteristieken zijn, zoals de intervallen voor hertransmissie als betrouwbare overdracht gewenst is;
- hoe om te gaan met acknowledements;
- wat de eigenschappen zijn van de transportkanalen;

### 3.3 Waarom wordt er een CPA gebruikt?

Redenen voor het toepassen van een CPA:

- het is een formeel contract tussen twee partijen, die op basis van ebMS2 gegevens willen uitwisselen;
- het automatiseert de e-configuratie van de ebMS-adapter (het inlezen van de CPA volstaat);
- het biedt de zekerheid dat beide partijen dezelfde instellingen gebruiken;

Daarom is het hebben van een CPA het uitgangspunt voor de specificatie en configuratie van de gegevensuitwisseling tussen twee partijen op Digikoppeling.

### 3.4 Wat zijn de uitgangspunten voor de CPA?

De kenmerken van het ebMS2 verkeer op Digikoppeling zijn beschreven in het document:

'Digikoppeling Koppelvlakstandaard ebMS2'. Dit document is op de website van Digikoppeling te vinden [[Digikoppeling Logius website](#)].

De kenmerken zijn vertaald naar relevante onderdelen van een CPA. Deze CPA onderdelen worden hieronder beschreven in termen zoals benoemd in [\[ebCPP\]](#).

- Service

Een Service is een unieke aanduiding voor een set van samenhangende operaties.

De service moet uniek zijn. Advies voor de naamgeving is als volgt:

`[organisatie]:[service]:[versie]`

De service heeft als type "urn:osb:services" (zonder quotes).

Een service wordt als volgt in het ebMS2 contract opgenomen (met een voorbeeld service "OSB:Service:1:0"):

```
<tns:Service tns:type="urn:osb:services">OSB:Service:1:0</tns:Service>
```

- CPAId

Een CPAId is een unieke aanduiding voor een overeenkomst voor een bepaalde service. De CPAId moet uniek zijn voor die bepaalde service. Denk hierbij aan het feit dat er één service wordt gespecificeerd door een service provider en dat meerdere service requesters een samenwerking aangaan. Elke samenwerking tussen twee partijen moet een ander CPAId krijgen. Een CPAId moet uniek zijn. Advies voor de naamgeving is als volgt:

`[ServiceID]_[PartyId-A]_[PartyId-B]_[versie]`

NB. Gebruik geen punten of andere vreemde tekens, want sommige ebMS-adapters zouden de CPAId wel eens als filenaam kunnen gebruiken...

Hierbij zijn:

- `[ServiceID]` de unieke Service ID
  - `[PartyId-A]` de PartyId van partij A
  - `[PartyId-B]` de PartyId van partij B
  - `[versie]` een UUID (of een versie nummer mits de uitgever van de CPA kan garanderen dat de CPAId uniek blijft)
- Start- en einddatum

Elke CPA heeft een start en einddatum. Dit moet afgestemd worden tussen de twee partijen die een samenwerking aangaan. Merk op dat er een afhankelijkheid is met de geldigheidsperiode van de gebruikte client- en servercertificaten.

- Default Message Channel

Een CPA bevat twee PartyInfo elementen: voor elke deelnemer in de samenwerking één. Elk PartyInfo element kent precies één 'default channel' dat gebruikt wordt voor de verzending van onderliggende protocol berichten (zoals de acknowledgments). In de CPA wordt deze 'default channel' aangegeven met het defaultMshChannelId attribuut. De eigenschappen van dit channel worden bepaald op basis van het Digikoppeling ebMS2 profiel met de hoogste beveiliging. Als een CPA verschillende Actions bevat waarvoor de acknowledgements verschillende profiel eigenschappen hebben, zullen de Actions verdeeld moeten worden over meerdere CPA's: in elke CPA komen die Actions die dezelfde profiel eigenschappen hebben. Als er gebruik gemaakt wordt van de CPA Creatievoorziening zullen er verschillende Digikoppeling-ebMS2 Servicespecificaties gemaakt moeten worden.

#### VOORBEELD 7

Er zijn twee Actions: Action1 : profiel osb-rm-e Action2 : profiel osb-be De default channel zal de eigenschappen overnemen van het profiel osb-rm-e. Als dit NIET wenselijk is, zullen de twee actions in twee verschillende CPA's geplaatst moeten worden.

- PartyName

De naam van de partij zoals die opgegeven moet worden in de CPA. Dit zal voor elke organisatie anders zijn, maar blijft wel hetzelfde voor alle CPA's die gemaakt zullen worden.

- PartyId

Het Organisatie Identificatie nummer (OIN) van de organisatie. De PartyId is de (logische) aanduiding waarmee de organisatie geïdentificeerd wordt. Als de organisatie nog geen OIN heeft moet een nummer aangevraagd worden bij Logius. De COR (De Centrale OIN Raadpleegvoorziening) wordt gebruikt om informatie vast te leggen over de organisatie en het OIN. Het nummer zal ook gebruikt worden in het cliënt certificaat voor het subject.Serialnumber veld. Zie ook EB014 in hoofdstuk 2. Organisaties die op basis van standaarden met andere overheden communiceren wordt sterk aangeraden om een OIN aan te vragen.

- PartyId Type

Deze heeft de waarde urn:osb:oin voor PartyId's met een OIN.(Dit is ook de default waarde voor de CPA's zoals die door het CPA register wordt gehanteerd.)

De PartyId type wordt als volgt opgenomen in het ebMS2 contract (met een voorbeeld van de PartyId waarde "0123456789"):

```
<tns:PartyId tns:type="urn:osb:oin">123456789</tns:PartyId>
```

Het is toegestaan om een andere PartyId type te hanteren als de organisatie reeds andersoortige (geen OIN's) PartyId's heeft voor de organisatie identificatie. Het moge duidelijk zijn dat het in overleg met de samenwerkende organisaties vastgesteld moet worden. Zie ook EB014 in hoofdstuk 2.

- BusinessCharacteristics

Deze heeft de volgende verplichte waarde, waarbij alleen de timeToPerform een andere waarde kan krijgen (afhankelijk van de timing karakteristieken van de RequestingBusinessActivity en de RespondingBusinessActivity):

```
<tns:BusinessTransactionCharacteristics
  tns:isAuthenticated="transient"
  tns:isAuthorizationRequired="true"
  tns:isConfidential="transient"
  tns:isIntelligibleCheckRequired="false"
  tns:isNonRepudiationReceiptRequired="false"
  tns:isNonRepudiationRequired="false"
  tns:isTamperProof="transient"
  tns:timeToPerform="P2D"/>
```

TransportProtocol over HTTP met TLS met server certificaat.

Deze hebben de verplichte waardes:

```
<tns:TransportProtocol tns:version="1.1">HTTP</tns:TransportProtocol>
```

en voor bijvoorbeeld versie 1 van TLS

```
<tns:TransportSecurityProtocol tns:version="1.0">TLS</tns:TransportSecurityProtocol>
```

voor de actuele versies van het te gebruiken protocol bij de uitwisseling zie [[Digikoppeling Beveiligingsdocument](#)]

- Client Authentication over HTTP met client certificaat. Dit is verplicht. In het client certificaat staat in het subject.Serialnumber de PartyId van de 'client' organisatie.
- Endpoint  
Deze heeft de waarde van de URL (FQDN, met pad namen) van de ebMS-adapter waarmee over Digikoppeling gegevens uitgewisseld worden. De FQDN van de URL moet overeenkomen met de FQDN die in het server certificaat vermeld staat.

- MessageOrder

De MessageOrder geeft aan of er wel of geen gebruik gemaakt wordt van ordening van berichten. De default waarde voor MessageOrder is "NotGuaranteed" en wordt als volgt opgenomen in het ebMS2 contract:

```
<tns:MessageOrderSemantics>NotGuaranteed</tns:MessageOrderSemantics>
```

Indien er wel gebruik gemaakt wordt van MessageOrder is de waarde:

```
<tns:MessageOrderSemantics>Guaranteed</tns:MessageOrderSemantics>
```

MessageOrder wordt niet door alle ebMS-adapters implementaties ondersteund. Als het wel het geval is zal de interoperabiliteit goed getest moeten worden. Zie hoofdstuk "Het gebruik van bericht volgordelijkheid".

- ReliableMessaging

Deze heeft default een retryCount van 8 en een retryInterval van 3 uur, zonder MessageOrder:

```
<tns:MessagingCharacteristics tns:syncReplyMode="none"
  tns:ackRequested="always" tns:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
  tns:ackSignatureRequested="never" tns:duplicateElimination="always"/>
```

De waardes kunnen per CPA bepaald worden, en liggen dus niet bij voorbaat vast.

In het geval dat MessageOrder wel gebruikt wordt, komt in de CPA:

## [tns:MessageOrderSemanticsGuaranteed](#)

Conform de ebMS2 specificatie zal de applicatie dezelfde ConversationId moeten gebruiken voor de opeenvolgende berichten<sup>5</sup>.

<sup>5</sup>: [EBXML-MSG] H9.1.1 "The REQUIRED SequenceNumber element indicates the sequence a Receiving MSH MUST process messages. The SequenceNumber **is unique within** the ConversationId and MSH."

- PersistDuration

Deze heeft default de waarde van 1 dag, maar zal anders zijn als er andere waardes voor ReliableMessaging gebruikt worden:

```
<tns:PersistDuration>P1D</tns:PersistDuration>
```

- MessagingCharacteristic

Deze heeft de waarde:

```
<tns:MessagingCharacteristics
tns:syncReplyMode="none"
tns:ackRequested="always"
tns:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
tns:ackSignatureRequested="never"
tns:duplicateElimination="always"/>
```

Geen gebruik van Signing of (payload) Encryption. (Alleen op HTTP nivo wordt informatie beveiligd)

Er moet gebruik gemaakt worden van certificaten die voldoen aan de eisen van de PKI Overheid.

### 3.5 Hoe wordt een CPA gemaakt?

Op basis van de hierboven genoemde CPA onderdelen kan alleen een 'CPA template' gemaakt worden. Wat ontbreekt zijn de specifieke zaken rondom:

- De services en functies (Actions in ebMS2 terminologie) die aangesproken kunnen worden, inclusief procesnaam waar ze deel van uitmaken.
- De technische gegevens van een ebMS-gateway van een organisatie, zoals de te hanteren transport URL, de publieke sleutels van de client en server certificaten ende PartyId van de organisatie

In het document 'Digikoppeling CPA Creatiehandleiding' is te lezen met welke gegevens een CPA gemaakt wordt, in combinatie met Digikoppeling CPA Creatievoorziening.

Meer informatie	Zie document in de aansluitkit	Doelgroep
Handleiding CPA register	<a href="https://www.logius.nl/diensten/digikoppeling/aanvragen/voorbereiden">https://www.logius.nl/diensten/digikoppeling/aanvragen/voorbereiden</a>	[A&D] [OT&B]

## 4. Het gebruik van berichtvolgordelijkheid



-

Digikoppeling Koppelvlakstandaard ebMS2 raadt het gebruik van volgordelijkheid van berichten sterk af. Reden hiervoor is dat niet elk product dat ebMS2 implementeert de volgordelijkheid ondersteunt (in ebMS2 wordt dit MessageOrder genoemd). Voor situaties waar beide partijen bi-lateraal over een product beschikken dat dit ondersteund, wordt het gebruik ervan voor het Digikoppeling ebMS2 Koppelvlak wel toegestaan. Voorwaarde daarbij is dat vooraf bekend is dat alle (ook toekomstige) communicatiepartners dit kunnen ondersteunen. Gebruik lijkt daarom alleen realistisch voor bilaterale situaties of een zeer beperkt aantal vooraf bekende communicatiepartners. Gebruik door landelijke voorzieningen zoals basisregistraties is onwenselijk.

Partijen die gebruik willen maken van de volgordelijkheid zullen daarom vooraf onderling moeten verifiëren of de functionaliteit door de andere partij ondersteund wordt en of de ebMS-adapters op dit onderdeel ook interoperabel zijn. (Het moge duidelijk zijn dat als beide partijen van hetzelfde product gebruik maken, de interoperabiliteit gegarandeerd zou moeten zijn.)

Het heeft tot gevolg dat veel partijen deze functionaliteit niet kunnen gebruiken. In deze situaties is volgordelijkheid op applicatieniveau een beter alternatief. Dit heeft als extra voordeel dat niet alleen zekerheid bestaat over het in volgorde ontvangen, maar ook over het in volgorde verwerken van berichten.

De granulariteit (zie hieronder) van de berichten waarop volgordelijkheid toegepast moet worden en de verwerking in de organisatie zal nader bekeken moeten worden. De volgende twee hoofdstukken geven hierover enkele overwegingen.

## 4.1 Granulariteit

-

Granulariteit betekent letterlijk: (fijn)korreligheid, ofwel de mate van detaillering. De granulariteit van de berichten waarvoor volgordelijkheid van belang is zal verstandig moeten worden gekozen.

De uitersten worden hieronder beschreven:

- Alle berichten die op basis van een ebMS2 contract (CPA) verzonden worden zijn van elkaar afhankelijk ten aanzien van de volgorde.
  - Als één enkel bericht faalt in de overdracht, heeft dit tot gevolg dat de gehele berichtenstroom stopt. Deze berichtenstroom kan dan pas weer op gang gebracht worden als het falende bericht opnieuw gesynchroniseerd is.
  - Bij grote hoeveelheden berichten die in een kort tijdsbestek verzonden worden zullen aan de ontvangende kant tijdelijk bewaard moeten worden: dit legt een claim op de resources van de ebMS-adapter.
  - Berichten die mogelijk een hogere prioriteit hebben kunnen niet eerder verwerkt worden dan passend is ten aanzien van de volgorde van reeds verzonden berichten.
- Voor elk afzonderlijk bericht wordt gekeken of het onderdeel uitmaakt van een nieuwe berichtenstroom waarvoor volgordelijkheid van belang is.
- Er kunnen meerdere berichtstromen actief zijn die onafhankelijk van elkaar de volgordelijkheid ondersteunen.
- Per object moet aangegeven worden (of bekend zijn) of de message order van belang is.

De granulariteit wordt in essentie bepaald door het object waarover berichten uitgewisseld worden en waarvan de volgordelijkheid van belang is: het gaat dan om transactionele berichten over hetzelfde object.

## 4.2 Verwerking in de organisatie

Als de berichten op Digikoppeling in volgorde moeten worden afgeleverd, zal dit tot gevolg hebben dat diezelfde berichten ook in de juiste volgorde aangeboden worden aan de verwerkende applicatie in de organisatie.

## 4.3 Alternatieven voor berichtvolgorde

Het garanderen van de volgordelijkheid binnen een keten kan in zijn algemeenheid op meerdere manieren worden geregeld:

1. Vermijden van de behoefte

Een zuivere gebeurtenis-gedreven architectuur kan de behoefte aan volgordelijkheid vaak vermijden. In deze architectuur worden gegevens niet meegeleverd met gebeurtenissen, maar naar aanleiding van gebeurtenissen bij de bron geraadpleegd. Als in een dergelijke situatie bijvoorbeeld overlijden eerder dan de geboorte doorgegeven wordt zal de actie die op overlijden (of geboorte) volgt altijd de meest actuele gegevens opleveren.

2. Risico-reductie

Bewuste vertragingen aan de bron tussen twee opeenvolgende (gerelateerde) gebeurtenissen, kan het risico op het uit volgorde raken van berichten beperken.

3. Applicatief: mitigatie

In deze situatie verwerkt de afnemer gebeurtenissen zodanig dat eventuele volgordeproblemen worden gemitigeerd (simpelste algoritme: als het BSN binnenkomt voordat de persoon bekend is, wordt het BSN terzijde gelegd totdat de persoon wél bekend is).

4. Applicatief: unipotente operaties

In deze situatie wordt er voor gezorgd dat de operatie naar aanleiding van een bericht slechts op één manier interpreteerbaar is. Bijvoorbeeld door bij verandering van een subsidie of burgerlijke staat zowel de oude als de nieuwe situatie mee te geven. Of bij 'saldo-informatie' een verhoging of verlaging te sturen in plaats van de nieuwe waarde.

5. Applicatieve volgorde door ontvanger

In deze situatie geeft de applicatie aan het bericht informatie mee waarmee de ontvanger de volgorde kan bepalen. Dit kan bijvoorbeeld met tellers, zodat de applicatie de berichten voor verwerking in volgorde kan plaatsen, maar ook met timesteps zodat de applicatie na verwerking eventuele correcties kan uitvoeren (zie ook bijlage 1 onder "zelfbouwoverwegingen"). Dit zal liefst selectief alleen voor kritische berichten gebeuren zodat andere berichten ongestoord doorgang vinden.

6. Applicatieve volgorde door verzender

In deze situatie wacht de verzendende applicatie op een ontvangstbevestiging of verwerkingsbevestiging van de ontvanger voordat een nieuw bericht gestuurd wordt. Dit zal liefst selectief alleen voor kritische berichten gebeuren zodat andere berichten ongestoord doorgang vinden.

7. Logistieke volgorde

Digikoppeling biedt als optie om berichten in volgorde af te leveren aan de ontvangende applicatie. Deze functie van de Digikoppeling-adapter software wordt echter door enkele belangrijke leveranciers niet ondersteund. Digikoppeling stelt daarom dat deze optie alleen gebruikt kan worden als vóóraf bilateraal overeenstemming bereikt is over ondersteuning.

## 5. Message Ordering in ebXML

Een onderdeel van de ebMS 2.0 specificatie is de volgordelijkheid van berichten, aangeduid met MessageOrder. overgenomen uit hoofdstuk 9 van [\[EBXML-MSG\]](#)

## 5.1 MessageOrder Module\*\*

The MessageOrder module allows messages to be presented to the To Party in a particular order. This is accomplished through the use of the MessageOrder element. Reliable Messaging MUST be used when a MessageOrder element is present.

MessageOrder module MUST only be used in conjunction with the ebXML Reliable Messaging Module (section 6) with a scheme of Once-And-Only-Once (sections 6.6). If a sequence is sent and one message fails to arrive at the To Party MSH, all subsequent messages will also fail to be presented to the To Party Application (see status attribute section 9.1.1).

## 5.2 MessageOrder Element\*\*

The MessageOrder element is an OPTIONAL extension to the SOAP Header requesting the preservation of message order in this conversation.

De ebMS standaard biedt daarmee de mogelijkheid om de volgordelijkheid van berichten te garanderen.

Maar het is wel een OPTIONAL<sup>7</sup> element, dus bekijk per product of het ook daadwerkelijk ondersteund wordt.

<sup>7</sup>: OPTIONAL, uit [\[EBXML-MSG\]](#): "This word means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item."

## 5.3 Productondersteuning

De ondersteuning voor de MessageOrder verschilt per product. Hermes 2.0 en OrionMsg ondersteunen het wel, AxWay ondersteunt het niet, en de recente IBM release 'WebSphere Partner Gateway V6.1' ondersteunt het wel.

De Drummond Group voert jaarlijks ebXML interoperabiliteitstesten uit, waarmee leveranciers hun ebMS producten kunnen laten certificeren. Er wordt echter niet getest op MessageOrder.

Uit het test rapport van de Drummond Group, blz 18, hoofdstuk "Differing interpretations on the use of ConversationId":

(..) The ebMS v2.0 specification requires that ConversationId be present in all messages, and requires that if you implement the optional MessageOrdering feature (not tested by DGI) that ConversationId must stay the same over all ordered messages. (..)

## 5.4 Zelfbouwoverwegingen

Als het niet mogelijk is om de MessageOrder functionaliteit te gebruiken, kan zelfbouw overwogen worden. Het is wel raadzaam om een aantal aspecten in overweging te nemen voordat de implementatie van de volgordelijkheid in een applicatie opgepakt wordt.

- Worden berichten die niet in volgorde verwerkt hoeven te worden onderscheiden van berichten die wel in volgorde verwerkt moeten worden? Door verschillende berichttypes te gebruiken kan er op eenvoudige wijze onderscheid gemaakt worden tussen berichtstromen waarin volgordelijkheid al dan niet van belang is. De achterliggende gedachte is, dat het niet noodzakelijk is om alle berichten in volgorde te verwerken.
- Is er een functionele behoefte aan bevestigingen? Zo ja, dan is volgordelijkheid niet van belang.
- Hoe vaak komt het voor dat de volgorde wel van belang is? Als dat incidenteel voorkomt, zou een ontvangstbevestiging retour kunnen gaan, waarna een volgend bericht verzonden mag worden.
- Er zullen afspraken gemaakt moeten worden om situaties te kunnen identificeren (en bijbehorende acties uit te voeren) als bijvoorbeeld één specifiek bericht niet aangekomen is, ook niet met behulp van de betrouwbare overdracht. De stroom van de te verwerken berichten “stokt” dan.
- Welke acties onderneemt de ontvangende applicatie om de verzendende applicatie hierover te informeren?
- Welke consequenties heeft dit voor verzendende partij?
- In hoeverre moet dit proces geautomatiseerd worden?
- Het inregelen van dit proces is lastig en het is dan de vraag of een andere oplossing (zoals met bevestigingsberichten) een goed alternatief is.

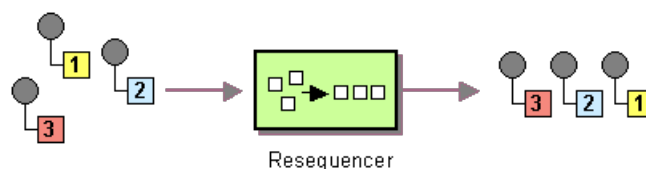
## 5.5 Ontwerp Pattern

Als uitgangspunt voor de realisatie van de volgordelijkheid kan het Resequencer patroon gebruikt worden: <http://www.enterpriseintegrationpatterns.com/Resequencer.html>



A [Message Router](#) can route messages from one channel to different channels based on message content or other criteria. Because individual messages may follow different routes, some messages are likely to pass through the processing steps sooner than others, resulting in the messages getting out of order. However, some subsequent processing steps do require in-sequence processing of messages, for example to maintain referential integrity.

*How can we get a stream of related but out-of-sequence messages back into the correct order?*



Figuur 2 Resequencer Pattern

*Use a stateful filter, a Resequencer, to collect and re-order messages so that they can be published to the output channel in a specified order.*

The Resequencer can receive a stream of messages that may not arrive in order. The Resequencer contains an internal buffer to store out-of-sequence messages until a complete sequence is obtained. The in-sequence

messages are then published to the output channel. It is important that the output channel is order-preserving so messages are guaranteed to arrive in order at the next component. Like most other routers, a Resequencer usually does not modify the message contents.



De oplossingsrichtingen voor berichten waarvoor een volgnummer van belang is wordt hieronder globaal beschreven. Er wordt uitgegaan van een 'push' mechanisme: de ontvangende applicatie wordt dus actief doordat de ebMS adapter een functie van de applicatie aanroept voor het afleveren van een bericht (bijvoorbeeld met behulp van een web service of JMS queue). Dit in tegenstelling tot een 'pull' mechanisme waarbij het initiatief bij de applicatie ligt om te bepalen of er een nieuw bericht is ontvangen.

### 5.5.1 Specificatie (Design Time)

- Voeg aan de specificatie van het bericht een element 'Volgnummer' toe.
- Definieer een 'Aanvang' en een 'Afsluit'-bericht waarmee de ontvangende partij geïnformeerd wordt over de te verwerken berichten. Dit kan met name van belang zijn als er meerdere parallelle stromen van berichten zijn die ieder afzonderlijk gebruik maken van volgordelijkheid. Het is dan wel van belang de ConversationId te gebruiken.
- Indien gewenst kan er een bericht gedefinieerd worden waarmee de ontvangende partij de verzendende partij kan informeren over de verwerkings toestand.
- De applicatie moet bijhouden wat het volgnummer is van het laatst verwerkte bericht.
- Er is een 'berichtenpool' beschikbaar waar berichten met een volgnummer in bewaard worden. Hierbij is het volgnummer een sleutel om berichten uit de 'berichtenpool' te halen.

### 5.5.2 Verwerking (Run Time)

- Bij ontvangst van een 'Aanvang'-bericht wordt de toestand geïnitieerd voor de volgordelijke verwerking van de berichten.
- Handel bij ontvangst van een bericht als volgt:
- Plaats het bericht in een 'berichtenpool'.
- Als bericht nummer N verwerkt is, moet de applicatie bericht nummer N+1 ophalen uit de 'berichtenpool'.
  - Als deze er niet is, zal de applicatie geen bericht verwerken.
  - Als deze er wel is, zal de applicatie het bericht verwerken EN daarna stap 2 opnieuw uitvoeren om een volgend bericht uit de 'berichtenpool' te verwerken.

Om te voorkomen dat een applicatie in een 'wait lock' terecht komt (één van de berichten in de sequentie komt niet aan, ook niet binnen de gestelde termijn van de betrouwbare overdracht), zal bekeken moeten worden wat de timingkarakteristieken zijn voor de verwerking van een volgend bericht.

Bij gebruik van het 'pull' mechanisme kan de berichtenpool gebruikt worden zoals in stap 2 beschreven: de applicatie zal dan op gezette tijden (op eigen initiatief) een bericht halen uit de berichtenpool. De ebMS adapter zal de berichten dan wel moeten afleveren aan de berichtenpool (zoals in stap 1).

## 6. Conformiteit

-

Naast onderdelen die als niet normatief gemarkeerd zijn, zijn ook alle diagrammen, voorbeelden, en noten in dit document niet normatief. Verder is alles in dit document normatief.

## 7. Lijst met figuren

-

[Figuur 1 Opbouw documentatie Digikoppeling](#)

[Figuur 2 Resequencer Pattern](#)

## A. Referenties

### A.1 Normatieve referenties

#### **[ebCPP]**

[Collaboration-Protocol Profile and Agreement Specification Version 2.0](#). Oasis. Oasis. september 2002.

URL: <http://www.ebxml.org/specs/ebcpp-2.0.pdf>

#### **[EBXML-MSG]**

[OASIS ebXML Message Service Specification](#). Ian Jones; Brian Gibb; David Fischer. 1 April 2002. URL:

[https://www.oasis-open.org/committees/download.php/272/ebMS\\_v2\\_0.pdf](https://www.oasis-open.org/committees/download.php/272/ebMS_v2_0.pdf)

### A.2 Informatieve referenties

#### **[Digikoppeling Beveiligingsdocument]**

[Digikoppeling Beveiligingsstandaarden en voorschriften](#). Logius. URL:

<https://publicatie.centrumvoorstandaarden.nl/dk/beveilig/>

#### **[Digikoppeling Logius website]**

[Logius Digikoppeling](#). Logius. URL: <https://logius.nl/diensten/digikoppeling/documentatie>