

# CPA Creation Toolkit

## From ebBP to CPA

Sonnenglanz Consulting BV  
June 2010, Version 1.0

**Business  
Process  
Modelling**

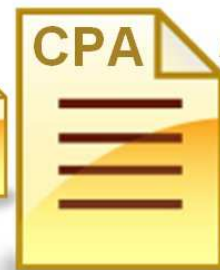


ebBP

**CPA Creation  
Toolkit**



**Collaboration  
Protocol  
Agreement**



*Automatic  
Configuration*



ebMS



## Table of Contents

|  |    |
|--|----|
| 1 Introduction .....                           | 3  |
| 1.1 ebBP .....                                 | 3  |
| 1.2 CPA .....                                  | 3  |
| 1.3 eBIZ-TCF .....                             | 4  |
| 2 Workflow Overview .....                      | 5  |
| 2.1 Specification .....                        | 5  |
| 2.2 Publication of the Specification .....     | 6  |
| 2.3 Publication of the Business Partners ..... | 6  |
| 2.4 Creation of the CPA .....                  | 6  |
| 3 Workflow Details .....                       | 9  |
| 3.1 Specification .....                        | 9  |
| 3.2 Publication of the Specification .....     | 12 |
| 3.3 Publication of the Business Partner .....  | 13 |
| 3.4 Creation of a CPA .....                    | 14 |

(This page intentionally left blank.)

# 1 Introduction

This section introduces the notions of ebBP and CPA as well as the project eBIZ-TCF where it has been applied.

## 1.1 ebBP

The **ebXML Business Process**<sup>1</sup> (ebBP) OASIS Standard provides a business process foundation that promotes the automation and predictable exchange of business collaboration definitions using XML. The specification is advanced by the OASIS ebXML Business Process Technical Committee<sup>2</sup>. This specification extends the original ebXML BPSS work performed jointly with UN/CEFACT and adds capabilities and functions to define the business process mechanisms (BPM) between external parties in e-Business collaboration.

## 1.2 CPA

The **ebXML Collaboration Protocol Profile and Agreement** (CPPA) OASIS Standard<sup>3</sup> (ISO 15000-1) defines how trading partners engage in electronic business collaborations through the exchange of electronic messages. Collaboration Profiles are one of the key pieces in ebXML - enabling users to quickly configure their partner interactions and relationships - detailing the communications methods, end-points, partner IDs, security and reliability options. Furthermore, the profiles add the ability to define message transactions and their send/receive couplings so that ebMS services can validate message exchanges with partners. Information from the profiles is also used in the ebXML messaging envelopes to validate and identify the exchange details.

Whenever two partners agree to the use of certain profiles, a **Collaboration Protocol Agreement** (CPA) is created. The CPA is used to configure the messaging software. A CPA contains information about the partners, amongst others, related to:

- The name of the partner;
- The identity (partyId) of the partner;
- The name and identifier of the UBL process of which the messages are part of;
- The identifier that uniquely identifies the CPA;
- The life-time of the CPA (this can depend on the life-time of the certificates in use);
- The roles of the partner within the business process;
- The messages that a partner can send;
- The messages that a partner can receive;
- The characteristics of the message exchange, for instance the reliability of the message exchange or the encryption of the payload;
- The transport security that has to be used by the server (SSL/TLS) as well as the client;
- The type of transport that has to be used, such as HTTP and the Internet address (URL) of the partner;
- The public keys of the certificates in use;

---

<sup>1</sup> ebXML Business Process [www.oasis-open.org/specs/index.php#ebxmlbp2.0.4](http://www.oasis-open.org/specs/index.php#ebxmlbp2.0.4)

<sup>2</sup> ebXML Committee [www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=ebxml-bp](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ebxml-bp)

<sup>3</sup> ebXML CPPA [www.oasis-open.org/specs/index.php#cppav2](http://www.oasis-open.org/specs/index.php#cppav2)

As a consequence, whenever some of the items are changed by a partner (for instance: a new certificate is obtained or the URL has changed) a new CPA has to be created and uploaded into the messaging software.

This technology can benefit end users now in reducing the costs of initial collaboration setup and in the management of the lifecycle of the configuration information. Monitored agreements may also be of value in reducing costs of operation and in resolving service problems.

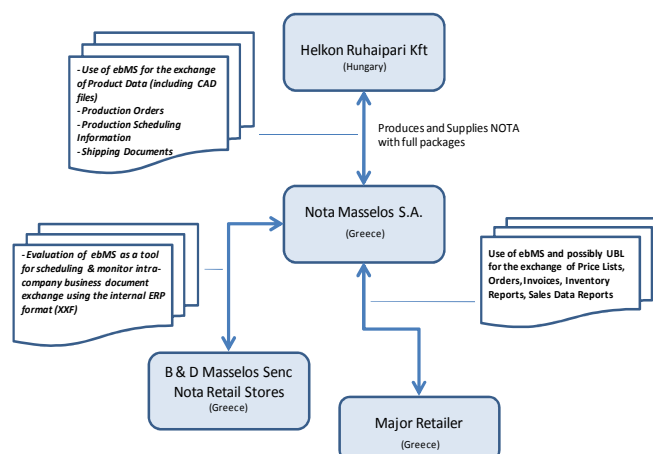
### 1.3 eBIZ-TCF

In an increasingly complex and fast-changing trade/economic environment, the fashion and footwear industries require fast time-to-market reaction. Standards and agreed reference architectures allow suppliers, manufacturers and retailers to connect their businesses within the value chain. Despite lot of initiatives in the fashion industry, standardisation and widespread adoption by users is still lacking.



The eBIZ-TCF project <sup>4</sup> is a 2-year cooperation project launched in January 2008 by the European Commission to boost e-business processes in the Textile/Clothing and Footwear (TCF) Industries. The eBIZ-TCF Project creates a single e-language for the supply chain of the European Textile, Clothing & Footwear industry. The eBIZ-TCF architecture is an interoperability framework. Its core uses established standards and reference architectures, such as Moda-ML/TexWeave, UBL profiles for TCF, ebXML CPPA and GS1 GLN.

A pilot project has implemented this architecture for the companies NOTA (Greece), Mavrommatis (Greece, Retailer) and Helkon Ruhaipari (Hungary).



<sup>4</sup> eBIZ-TCF Website <http://www.ebiz-tcf.eu>

## 2 Workflow Overview

This chapter presents the basic steps in the workflow of the CPA creation. It introduces the concepts and the essential input and output documents; a detailed description is presented in chapter three.

### 2.1 Specification

The **first step** in the workflow is the creation of a specification which defines the message exchange (or the business process, with an emphasis on message exchange). The specification activities will address for instance the following questions:

- Which partners, in terms of roles, are involved (like 'Buyer' or 'Seller')?
- How many partners are involved in the message exchange?
- What actions (transactions) have to be performed between the partners?
- What are the message exchange characteristics? (Do we need transport or payload encryption? How do we exchange messages in a reliable manner? How much time can be used for the message exchange?)

Besides these process related aspects, some specific information about business partners is required as well. Because:

- How do we identify and authenticate a business partner? (Note that multiple parties can send a message to a particular partner.)
- What is the transport mechanism used and what is the transport address (URL)?
- In case of payload encryption, what are the public certificates that are used by the partner?

A formal standard like ebBP can be used to describe the business process, including the roles, the business transactions and the messages (actions). Due to the ebBP complexity, it might not be an appropriate tool for everybody to specify a business process.

To allow a 'lean and mean' (or 'agile') method for specifying the message exchanges, an intermediate format is has been defined. This format is called 'Simple Message Format' (SMF). This format has been defined by Sonnenglanz Consulting BV as a result of several projects during the last years. The main goal of this intermediate format is to concentrate on the essentials of message exchanges. As a result of the eBIZ-TCF project, an ebBP can now be transformed into an SMF.

The result of the specification activities is therefore either an ebBP or an SMF. In case of an ebBP, a transformation will be performed to derive an SMF. The SMF is input for the next step.

## 2.2 Publication of the Specification

One has to realise that the SMF specification is part of an overall and much larger business process (including several manual activities and production steps). The SMF should be less dependent on the various time-dependent values such as the transport URL or authentication methods. The SMF specification of the message exchange should be an intrinsically reusable document.

To support the reusability, publication of the specification is the **second step**. Practically, this means that the specification is stored into a repository. A unique name will be assigned to the specification, so it can be easily found for reused.

Once a specification is published, one can select, at any time, a specification by means of a simple method (such as a drop-down menu), thereby initiating the CPA creation for different partners. This assumes that information about business partners is available as well. Consequently, the (time-dependent) information of a business partner has to be published beforehand.

## 2.3 Publication of the Business Partners

As indicated in the previous sections, each business partner has to provide some information. This type of information is more technical in nature and reflects the actual situation of a business partner in the messaging infrastructure. Examples of this type of information are the partner identification, the identification type, the transport URL or the certificate for the partner authentication.

Because each partner can be part of several business processes, reuse of this information is preferred.

As a **third step**, to enable reuse, each business partner will collect and publish its information. Practically, this means that each business partner will create a file in which all the relevant values are put together and stored into the repository. A unique name will be assigned to the information of the business partner, so it can be easily reused.

## 2.4 Creation of the CPA

The result of the previous steps is a repository of various business processes and (technical) information about different business partners. But not all business partners will collaborate in all business processes.

Therefore, the **fourth step** consists of:

1. Selection of a specification of a business process. As a result, a set of roles are presented.
2. Selection of the appropriate business partners for the required roles.

Once a specification of a business process is selected, a set of roles is presented that are defined within that business process. For each role a business partner has to be assigned. It is possible to

select a particular sub-set of roles, in case there are more than two roles and one has to create a specific CPA for a specific partner.

As a result of the selection of a specification and the assignment of business partners to the appropriate roles, one or more CPA's are created. The CPA's are downloaded and used for the automatic configuration of the ebMS messaging software.



(This page intentionally left blank.)

### 3 Workflow Details

This chapter will describe in more detail the workflow of the creation of a CPA. An overview is presented in chapter two.

It assumes that the reader is familiar with XML, URL, SSL, CPA terminology (such as partyId or partyName) and Certificates. Besides this, the reader must have a good understanding of the SMF specification, as described in the document:

“CPA Creation Toolkit - Simple Message Format Specification 2.0.pdf”.

#### 3.1 Specification

The specification is the result of an information and process analysis phase in close cooperation with the business partners. These particular activities are not elaborated here. This Section concentrates on the SMF.

The main ingredients for the specification are:

- The **roles** of the business partners;
- The **services** that are used by the business partners;
- The **interactions** (actions or operations) between the business partners;
- The **parameters** that have to be provided for each business partner;

As an example, three processes described in the eBIZ-TCF architecture document<sup>5</sup> are used to explain the relation to and use of the SMF. The three processes are:

- Section 4.1.1 Process "cyclic replenishment program - CRP"
- Section 4.1.2 Process "classical preorder"
- Section 4.1.3 Process "vendor managed inventory - VMI"

The ebBP processes are transformed into an SMF. The figure below shows the main XML structure of the final SMF.

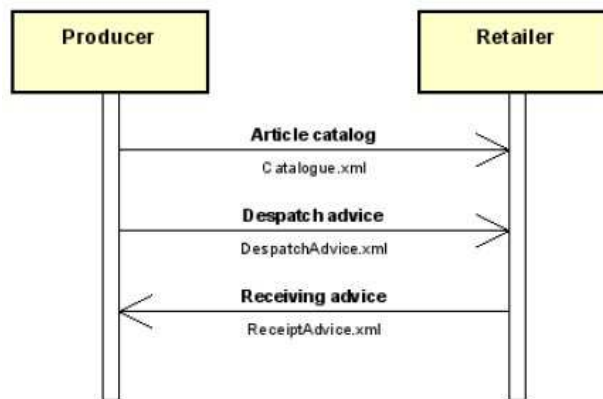
```
<?xml version="1.0" encoding="UTF-8"?>
<specification xmlns="http://sonnenglanz.net/2009/07/01/smf.services.2.0.xsd">
  <services>
    <service process="cyclic replenishment program CRP Business Collaboration" [22 lines]
    <service process="classical preorder Business Collaboration" [27 lines]
    <service process="vendor managed inventory VMI Business Collaboration" [27 lines]
  </services>
  <rolebindings>
    <binding name="Producer" [5 lines]
    <binding name="Retailer" [5 lines]
  </rolebindings>
  <parameters> [11 lines]
</specification>
```

---

<sup>5</sup> Architecture Report for eBusiness harmonisation in Textile/Clothing and Footwear sectors, D3.5 - Final Report, Version 1.0, December, 2009. ([Download from the web](#))

It shows that each process is modelled as a separate **service**. (The details of the service are hidden for the moment.) Within these processes there are two **roles**: Producer and Retailer (the element ‘binding’ within the element ‘rolebindings’). Not shown in detail is the element ‘parameters’: it specifies the elements that have to be provided by each business partner, such as the URL of their system or their partyId.

As an example of an **interaction**, the activity "Permanent replenishment" is shown below. It is part of the process "vendor managed inventory - VMI". (For more information, see Section 4.1.3.3 of the eBIZ-TCF architecture document.)



The above diagram shows that there are three message exchanges. In the SMF these appear as follows (the corresponding names are indicated in grey):

```
<?xml version="1.0" encoding="UTF-8"?>
<specification xmlns="http://sonnenglanz.net/2009/07/01/smf.services.2.0.xsd">
  <services>
    <service process="cyclic replenishment program CRP Business Collaboration" [22 lines]
    <service process="classical preorder Business Collaboration" [27 lines]
    <service process="vendor managed inventory VMI Business Collaboration" serviceId="s3" processUuid="vendormanagedinventoryVMI-1_2008-1" service="vendor manager
      <message from="Retailer" to="Producer" [2 lines]
      <message from="Retailer" to="Producer" [2 lines]
      <message from="Retailer" to="Producer" [2 lines]
      <message from="Producer" to="Retailer" [2 lines]
      <message from="Producer" to="Retailer" id="ReqBtArticlecatalogue830__bdArticlecatalogue83" transport="reliable" security="transport">ArticleCatalogue</message>
      <message from="Producer" to="Retailer" id="ReqBtDespatchadvice8990__bdDespatchadvice89" transport="reliable" security="transport">DespatchAdvice</message>
      <message from="Retailer" to="Producer" id="ResBtReceivingadvice9089__bdReceivingadvice90" transport="reliable" security="transport">ReceivingAdvice</message>
      <message from="Producer" to="Retailer" [2 lines]
      <message from="Producer" to="Retailer" [2 lines]
      <message from="Producer" to="Retailer" [2 lines]
    </service>
  </services>
  <rolebindings>
    <binding name="Producer" [5 lines]
    <binding name="Retailer" [5 lines]
  </rolebindings>
  <parameters> [11 lines]
</specification>
```

(Note that the service contains more message exchanges, but for convenience they are ‘hidden’ in the diagram.)

Because there are three separate services, it has to be specified how the “from” and “to” roles are mapped to a business partner that will fulfil these roles. The main reason for this mapping is

the possibility that a business partner can have different roles between or within service specifications.

Our example uses only two roles, consistently, so this will make it easier to map them to a generic role for a business partner. This mapping is defined by means of the 'rolebindings' element, as shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<specification xmlns="http://sonnenglanz.net/2009/07/01/smf.services.2.0.xsd">
  <services> [78 lines]
  <rolebindings>
    <binding name="Producer" parameterId="p1">
      <role servid="s1" name="Producer"/>
      <role servid="s2" name="Producer"/>
      <role servid="s3" name="Producer"/>
    </binding>
    <binding name="Retailer" parameterId="p1">
      <role servid="s1" name="Retailer"/>
      <role servid="s2" name="Retailer"/>
      <role servid="s3" name="Retailer"/>
    </binding>
  </rolebindings>
  <parameters> [11 lines]
</specification>
```

The example shows that *generic* role 'Producer', as specified by the 'binding' element, is bound to the name 'Producer' *within the referenced service* (each service declares an identifier by which it is referenced). Note that it does not specify how the name is used (the "from" and "to"): that is determined by the service specification itself.

Once the services have been defined and the roles within the services have been mapped to generic roles for the business partners, one has to specify the elements that each collaborating business partner must supply.

```
<?xml version="1.0" encoding="UTF-8"?>
<specification xmlns="http://sonnenglanz.net/2009/07/01/smf.services.2.0.xsd">
  <services> [78 lines]
  <rolebindings> [11 lines]
  <parameters>
    <group parameterId="p1">
      <parameter name="ProcessHref" required="false">http://www.moda-ml.net/ebiz-retail/repository/ebbp/2008-1/en/</parameter>
      <parameter name="PartyName" required="true"/>
      <parameter name="PartyId" required="true"/>
      <parameter name="PartyIdType" required="false">http://www.iso.int/schemas/eanucc/gln</parameter>
      <parameter name="PartyRef" required="false"/>
      <parameter name="EndpointUri" required="true"/>
      <parameter name="ClientCert" required="true"/>
      <parameter name="ServerCert" required="true"/>
    </group>
  </parameters>
</specification>
```

Each element has a particular name. If a value has to be provided by a business partner, it is indicated by means of the statement *required="true"*. For instance, each business partner has to provide the following information:

- **PartyName:** the name of the business partner.
- **PartyId:** the identifier of the business partner. In the eBIZ-TCF project a GLN is used.
- **PartyIdType:** the type of the identifier (GLN in this case).
- **EndpointUri:** a URL of the messaging service of the business partner.
- **ClientCert:** the (public) client certificate for the authentication of the business partner.
- **ServerCert:** the (public) certificate of the server to enable other business partners to trust the SSL certificate.

After all the various aspects have been described, and checked, the specification is finished and ready for publication.

### 3.2 Publication of the Specification

To publish the specification one has to provide the following details:

- The unique identifier of the specification. This is used to select the specification at a later stage. It is advised to use a clear name and version.
- The specification itself. (The XML file will be uploaded.)
- The name and e-mail address of the contact person. This contact person is informed by e-mail each time a CPA is created.

Below an impression is given of the form to publish the specification:

The screenshot shows a web form titled "Store Service Specification" under the "CPA Creation" header. The form is divided into two main sections: "Service Specification" and "Contact Information".

**Service Specification Section:**

- Identifying Name:** A text input field containing "eBIZ-TCF.1.0" with a "(ID)" label to its right.
- Service File:** A text input field containing "C:\Example.eBIZ-TCF.msg.xml" with a "Browse..." button to its right and a "(ID.msg.xml)" label. Below this, there is a note: "Make use of the provided [XSD](#) when you create the SMF file!".
- Format Selection:** Two radio buttons are present: "Simple Message Format (default)" (which is selected) and "Template".

**Contact Information Section:**

- Name:** A text input field with a "(Default)" label to its right.
- E-mail Address:** A text input field.

At the bottom of the form, there are two buttons: "Store Service" and "Reset".

After the "Store Service" button is pressed, the specification is stored in the repository.

### 3.3 Publication of the Business Partner

As indicated at the end of Section 3.1, each business partner has to provide a number of elements by which the actual message exchange can be realised. The information is stored in an XML document, called a 'participant file' or 'participant specification'.

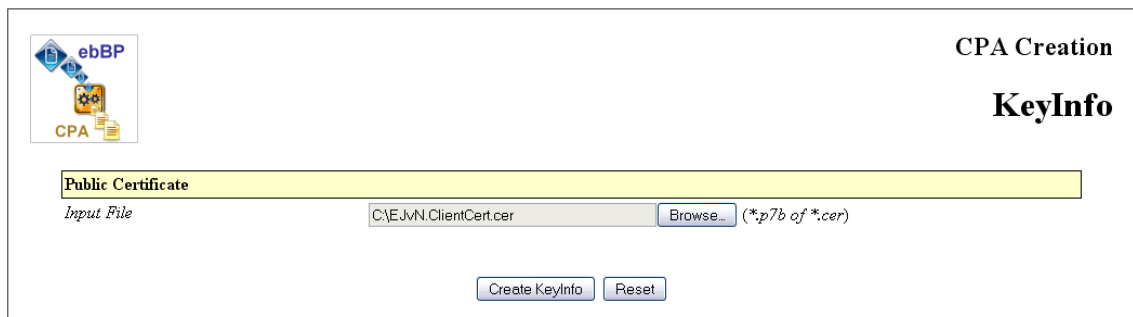
The format is straightforward, as the following example shows:

```
<?xml version="1.0" encoding="UTF-8"?>
<parameters xmlns="http://sonnenglanz.net/2009/07/01/smf.parameters.2.0.xsd">
  <parameter name="PartyName">Ernst Jan van Nigtevecht</parameter>
  <parameter name="PartyId">ejvn</parameter>
  <parameter name="PartyRef">http://www.vannigtevecht.info/</parameter>
  <parameter name="EndpointUri">https://rigel.vannigtevecht.info:443/exchange/ejvn</parameter>
  <parameter name="ClientCert"> [36 lines]
  <parameter name="ServerCert"> [36 lines]
</parameters>
```

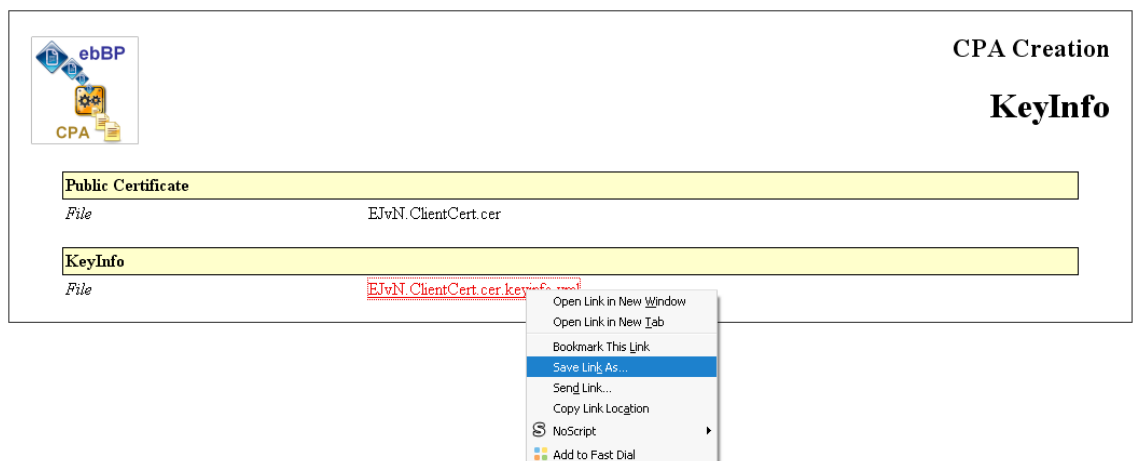
The names of the elements correspond to the names in the service specification!

The elements for public certificates (such as 'ClientCert' and 'ServerCert') contain the KeyInfo structures in xml of the certificates. The data is obtained by means of the "KeyInfo" utility, as shown below. To derive the KeyInfo xml from the certificates (\*.p7b or \*.cer), perform the following steps. First, provide the public certificate. Second, download the resulting KeyInfo xml structure using the 'Save As...' option (if you just click on the link, you will see the contents).

(1)



(2)



The *content* of the KeyInfo xml file has to be copied into the right place for the corresponding parameter in the participants file. Use an XML editor for that; never copy the data ‘grabbed’ from the browser screen!

Once the participant xml file is ready and checked (use the provided XSD), it can be published. One has to provide the following details:

- The unique identifier of the business partner. This is used to select the partner at a later stage. It is advised to use a clear name and version.
- The XML file of the business partner itself. (The XML file will be uploaded.)
- The name and e-mail address of the contact person.

Below an impression is given of the form to publish the specification:

The screenshot shows a web application titled "CPA Creation" with a sub-header "Store Participant Specification". On the left is a logo with "ebBP" and "CPA" icons. The form is divided into two main sections: "Participant Specification" and "Contact Information".

**Participant Specification**

*Identifying Name*: A text input field containing "EJVNLaptop" with a "(ID)" label to its right.

*Participant File*: A text input field containing "C:\EJVNLaptop.parameters.xml" with a "Browse..." button to its right. Below this field is a note: "Make use of the provided [XSD](#) when you create the file!".

**Contact Information**

*Name*: A text input field with a "(Default)" label to its right.

*E-mail Address*: A text input field.

At the bottom of the form are two buttons: "Store Participant" and "Reset".

After the “Store Participant” button is pressed, the information of the business partner is stored in the repository.

### 3.4 Creation of a CPA

The creation of a CPA becomes a simple select-and-click activity because all the relevant information is already published.

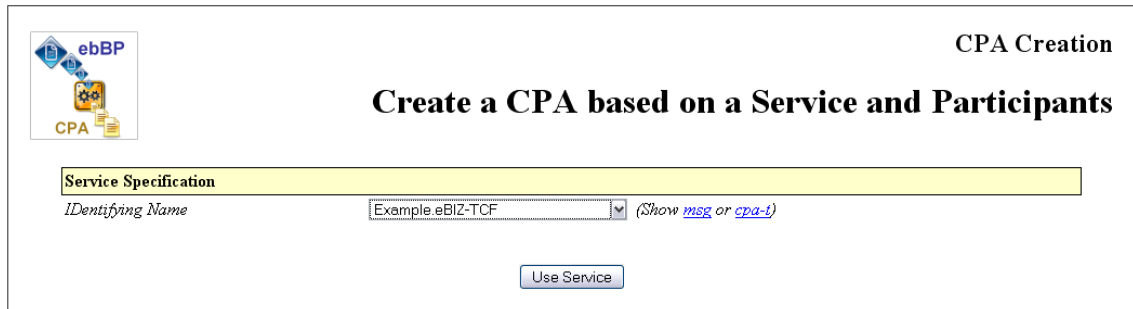
To start, select the service for which a CPA is needed.

The screenshot shows a web application titled "CPA Creation" with a sub-header "Create a CPA based on a Service and Participants". On the left is a logo with "ebBP" and "CPA" icons.

**Service Specification**

*Identifying Name*: A dropdown menu with the text "Choose a Service ID" and a list of service options: "Example.Hub", "Example.MultiplePartners", "Example.MultipleServicesAndRoles", "Example.Profiles", "Example.ProfilesXInclude", "Example.TransportTestCases", "Example.eBIZ-TCF", and "Example.fromAtoB". To the right of the dropdown is a link: "(Show [msg](#) or [cpa-t](#))".



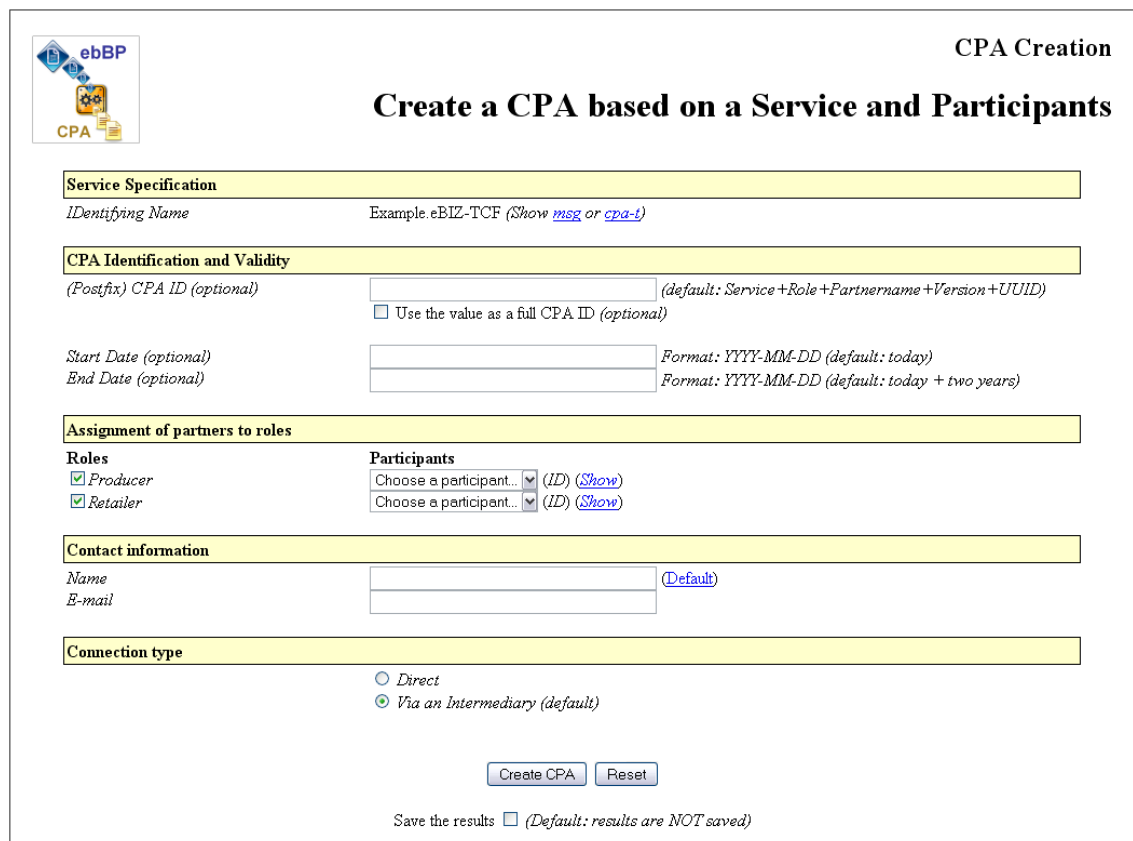


The screenshot shows the top part of a web form titled "CPA Creation" and "Create a CPA based on a Service and Participants". On the left is a logo with "ebBP" and "CPA" icons. The "Service Specification" section has a yellow header. Below it, the "Identifying Name" is set to "Example.eBIZ-TCF" in a dropdown menu, with a "(Show msg or cpa-t)" link. A "Use Service" button is at the bottom right of this section.

Press the “Use Service” button.

Based on the selected specification the system derives a new page containing the roles, as found in the selected service specification.

Each role must be mapped to a business partner that will fulfil that role. (If there are more than two roles, one can select the roles for which a CPA should be created.)



This screenshot shows the continuation of the form. The "CPA Identification and Validity" section has a yellow header and contains a text field for "(Postfix) CPA ID (optional)" with a "(default: Service+Role+Partnername+Version+UUID)" hint and a checkbox "Use the value as a full CPA ID (optional)". Below are "Start Date (optional)" and "End Date (optional)" fields with format hints: "Format: YYYY-MM-DD (default: today)" and "Format: YYYY-MM-DD (default: today + two years)". The "Assignment of partners to roles" section also has a yellow header and is divided into "Roles" and "Participants". Under "Roles", "Producer" and "Retailer" are checked. Under "Participants", there are two dropdown menus for selecting participants, each with an "(ID) (Show)" link. The "Contact information" section has a yellow header and fields for "Name" and "E-mail", with a "(Default)" link. The "Connection type" section has a yellow header and two radio buttons: "Direct" and "Via an Intermediary (default)". At the bottom are "Create CPA" and "Reset" buttons, and a "Save the results" checkbox with a "(Default: results are NOT saved)" hint.


The example form shows two roles: the Producer and the Retailer. For each role there is a drop-down box by which the appropriate business partner, the ‘participant’, can be selected.



For example:

| Participants            |                           |
|-------------------------|---------------------------|
| Choose a participant... | (ID) <a href="#">Show</a> |
| Choose a participant... | (ID) <a href="#">Show</a> |
| EJVN.laptop             |                           |
| Example.Partner1        |                           |
| Example.Partner2        |                           |
| Example.Partner3        |                           |
| Example.Partner4        |                           |

After the “Create CPA” button is pressed, one or more CPA’s are created and provided as a download. For example:

| CPA Creation  |  |
|---|--|
|  |  |
| <b>Create a CPA based on a Service and Participants</b>                           |  |
| CPA's created:  |  |
| Participants:   | CPA's for the roles:                           |
| <i>EJVN.laptop - Example.Partner1</i>   | <a href="#">CPA</a> <i>Producer - Retailer</i> |
| (There is no e-mail server configured. You will not be notified via e-mail.)      |  |

Use the ‘Save As...’ option (if you just click on the link, you will see the contents).

CPA's for the roles:

|                     |                            |
|---------------------|----------------------------|
| <a href="#">CPA</a> | <i>Producer - Retailer</i> |
|---------------------|----------------------------|

(There is no e-mail server configured. You will not be notified via e-mail.)

Right-click context menu options:

- Open Link in New Window
- Open Link in New Tab
- Bookmark This Link
- Save Link As...**
- Send Link...
- Copy Link Location
- NoScript
- Add to Fast Dial

You can now use the CPA for the configuration of the ebMS adapter.