

the attribute. Tricky situation where one party wants to express that it only accepts to have an element if the value will be in a certain range for example.

Section 11.5 of the ANCPA specification does not cover in detail how to deal with elements referenced via their ID's by other elements. This particular issue is left open in the specification. This research investigated this XML referencing problem in more detail (see section 6.4.2 of the results chapter).

The ANCPA specification also addresses security considerations in section 11.7 which are similar to those in the CPA composition.

The specification declares in section 11.8 that some CPP elements such as **ActionContext**, **CollaborationActivity** are not negotiable. Regardless of what is negotiable and what is not, the specification further suggests that if an element has a certain degree of complexity (such as element references), then instead of modifying that element, a new one shall be created. It can be said that it is better to create a new instance of an element than to modify a problematic one. Specifically it states that it is preferred to create a new **DeliveryChannel** element rather than modifying an existing one. This specific problem is also analysed in the XML referencing problem section in 6.4.2. The list of elements and attributes in the ANCPA specification highlight that it is not clear when an element and attribute is negotiable and when not. The usage of the word "can", "should", and "might" in the ANCPA specification indicate no clear rules. The ANCPA specification needs to better illustrate which elements and attributes of a CPP or CPA template are negotiable.

5.4.3 Negotiation Descriptor Document

The ANCPA specification introduces the Negotiation Descriptor Document (NDD) in section 12. An NDD is an XML document, which lists negotiable information items (via the **NegotiableInformationItem** XML element) of a CPP or CPA template. The negotiable information items reference the negotiable elements and attributes of a CPP or CPA template. These items also provide negotiation information such as preferences for values, ranges of values, if an optional element or attribute is preferred to be present or not, as well as if the element or attribute is present which preferred value it has. An NDD associated with a CPP is important for the CPA composition. An NDD associated with a CPA template is important for the CPA negotiation. In the CPA formation scenario in Figure 27 on page 77, there are three different NDD's. One NDD is for party A's CPP, one NDD is for party B's CPP and one NDD is for the CPA template of party A and party B. Once a final CPA is reached no further NDD's are needed.

CPP's already have basic negotiation capabilities, such as providing a list of options. But these negotiation capability settings of a CPP are limited to very few elements. The NDD enhances these basic negotiation capabilities dramatically.

This section will use the following notation for the differentiation of the two NDD's: NDD_{CPP} and $NDD_{CPA-template}$. NDD alone is used for both types of NDD's. This special notation is only used in this section as it does not represent the official notation of the ANCPA specification.

As the two processes, the CPA composition and the CPA negotiation, get closer, their transition becomes a critical point in the CPA formation process. The CPA template and the $NDD_{CPA-template}$ are the output of the CPA composition and are at the same time the input for the CPA negotiation at the same time. This seems like a smooth and possibly an electronic transition. But it has been analysed that whenever there is a issues list like the gap list or conflict file then there is a high probability that only human to human negotiation will be necessary. Simply because the CPA composition had already tried to match that element or attribute. Otherwise

the conflict would not have been listed in the gap list in the first place.

Section 12.1 of the ANCPA specification sets some very important rules. One such rule is that only the elements and attributes listed in the NDD are negotiable. Another rule is, that the $NDD_{CPA-template}$ cannot be changed during a CPA negotiation. Even though not explicitly stated, the ANCPA specification does not preclude the parallel negotiation of all negotiation items of the $NDD_{CPA-template}$.

Section 12.2 of the ANCPA specification introduces the principle of the content of the NDD. Each negotiable item listed in the NDD includes an XPath expression which points to the negotiable element or attribute within the CPP or CPA template. This research analysed that for an NDD_{CPP} , it is sufficient to have one XPath expression, but that an $NDD_{CPA-template}$ better uses two XPath expressions (each of both XPath expressions pointing to one of the complimentary element or attribute).

If the XPath expression points to an attribute, the attribute is negotiable. If the XPath expression points to a leaf element, the leaf element is negotiable. If the XPath expression points to a non-leaf-element, the non-leaf-element and the whole subtree is negotiable. Section 12.3 of the ANCPA specification talks about $NDD_{CPA-template}$ composition. More comments on this later.

Section 12.4 of the ANCPA specification continues with the content of the NDD. Each XML element and attribute of a CPP or CPA template are an instance of a type, specified in the CPPA XML schema, such as integer, strings, or enumerations. A numeric element or attribute can have extra negotiation information in the NDD such as the minimum and maximum value or the step size of value changes during the negotiation. The different types are reflected in the negotiation item list of the NDD where each item has a type associated with it, such as

- **Value,**
- **UnOrderedValue,**
- **OrderedValue,**
- **IntegerValue,**
- **ValueWithPreferenceMeasure,**
- **PresentOrNot,**
- **Preference,**
- **Cardinality,**
- **BooleanValue,**
- **DurationWithPreference.**

This research sees Section 12.4 of the ANCPA specification as one of its most important sections because the content of the $NDD_{CPA-template}$ is one core component of the CPA negotiation, apart from any negotiation algorithms and negotiation strategies. Section 12.4 of the ANCPA specification is unfortunately very immature. It is important to provide more explanations of the negotiable information types. Only the sample NDD in Appendix F of the ANCPA specification provide some more information.

Further, the specification has to make it clear that these negotiable information types are all that is needed for an NDD. This research sees that the $NDD_{CPA-template}$ has a central part in the CPA negotiation. The success of the NDD concept lies in its capability to precisely specify what is negotiable in the CPP or CPA template.

One reason to introduce NDD's was to allow a negotiation to converge faster. This research acknowledges that to get the negotiable information types right, is one of the most difficult tasks of the ANCPA specification. Further, this research believes that this area has to be further developed in the next revisions of the ANCPA specifications. This analysis believes that AI negotiation knowledge should be considered to be applied to this problem.

The ANCPA specification makes it clear that it is the CPA negotiation which handles preferences. The CPA negotiation must choose one element from the preference lists of the two original CPP's. From Appendix E of the CPPA specification indicates that the CPA composition tool leaves the preferences as they are, hence putting the preference lists into the CPA template. In relation to the CPA composition tool, the **Preference** element of the **NegotiableInformationItem** element indicates that the ANCPA specification wants the CPA composition tool to add all preference occurrences of the CPP's into the newly created *NDD_{CPA-template}*. This analysis shows, that the two specifications talking about the CPA formation process are not aligned enough that a CPA composition tool adds enumerations, preferences information into the *NDD_{CPA-template}*.

This analysis proposes to put examples into the Appendix of the specification how preferences are handled. Examples should include raw CPP's with and without *NDD_{CPP}*'s, CPA templates including an *NDD_{CPA-template}*, and a final CPA.

The question why there is a CPA negotiation at all arose. This research found three reasons to have a CPA negotiation:

1. The determination if an optional element or attribute must be present or not.
The CPPA XML schema allows optional elements and attributes. The problem if an element or attribute is present in one **PartyInfo** but not in the second **PartyInfo** has to be addressed in the CPA negotiation. Of course the CPA composition must have found this issue already. The CPA composition should then check the *NDD_{CPP}* if they list possible solutions to the given problem. First the two negotiation parties have to negotiate whether the element or attribute must be present or not. Second, if they agree to have the element or attribute present, the parties have to negotiate over the value of the element or attribute.
2. The determination of cardinality.
The CPPA specification and CPPA XML schema allows parties to list options or alternatives in the CPP. The **ChannelId** element is such an element, where one party can have more than one **ChannelId** element in their CPP but in the final CPA, there will be only one **ChannelId** (Schlegel S. 2004, '[ebxml-cppa-negot] ANCPA Version 0.10 review', ebxml-cppa-negot discussion list, 6 January. Retrieved: March 2, 2004, from <http://lists.oasis-open.org/archives/ebxml-cppa-negot/200401/msg00001.html>). In this example, the CPA negotiation has to determine (negotiate) which compatible **ChannelId** of both parties is selected. The ANCPA specification does not provide an algorithm how this selection is achieved; this is left out for to the CPA negotiation implementors or CPA negotiation actors.
3. To simply negotiate values of elements and attributes.
This introduces a totally new possibility in the ebXML framework. The ANCPA specification introduces negotiation capabilities to actively negotiate elements and attributes and their values of the CPA. The ANCPA handles negotiation on a technical level (CPA provides the technical capabilities of two party's). Potentially this negotiation can be reused for negotiation on a business level.

The first two tasks are necessary because of the nature of the CPPA XML schema. The ANCPA specification clearly states, that the cardinality and preference of elements and attributes are left out from the CPA composition for the CPA negotiation. This means, that whenever there is a cardinality or preference case, there must be a CPA negotiation. Further, the ANCPA specification is not clear, if the cardinality's and preferences must be listed in the *NDD_{CPP}* to be negotiable, or if, per default, certain elements and attributes are negotiable no matter if they are listed as a **NegotiableInformationItem** in the corresponding NDD. The ANCPA would have to clearly list those elements and attributes.

The usage of NDD's introduces new challenges. This research did not find solutions for the following challenges but considers it worthwhile to mention them here:

- The first challenge is to create valid NDD's. If a company provides a CPP (based on a BPSS), they potentially also provide an *NDD_{CPP}*. One of the issues here, as discussed in section 6.4.1, is the correctness, validity of the NDD.
- The *NDD_{CPP}*'s potentially provide solutions for conflicts during the CPA composition process. For each conflict, the CPA composition algorithm first has to find the matching **NegotiableInformationItem** of the *NDD_{CPP}*'s. Second, if and only if both *NDD_{CPP}*'s have the conflicting element or attribute as a **NegotiableInformationItem**, the algorithm has to determine, if the information in the **NegotiableInformationItem** element provides a solution for the conflict. If only one *NDD_{CPP}* has the conflicting element or attribute as negotiable means that the second *NDD_{CPP}* does not allow the conflicting element or attribute to be negotiated.
- The ANCPA specification outlines the requirement, that the CPA composition tool must use the available *NDD_{CPP}*'s for the CPA template composition and must merge the two *NDD_{CPP}*'s into one *NDD_{CPA-template}*. The challenge will be the NDD merging process. One question is, if the **NegotiableInformationItem**'s are always sufficient to find the right pair of the two *NDD_{CPP}*. Another question is, how the CPA composition tool has to report NDD specific conflicts. Most likely, such conflicts also have to be listed in the gap list, or a *NDD_{CPA-template}* gap list.

A short analysis reveals merging problems such as the following. The choice child element of a **NegotiableInformationItem** element must be of the same type, such as merging **BooleanValue** elements only with **BooleanValue** elements. If these are the fix values without the parties are willing to negotiate it is clear that the CPA negotiation will fail for this **NegotiableInformationItem**. Another problem with a range type is that one party has a range of "100-150" whereas the second range is of "200-250". For example a **BooleanValue** choice of the **NegotiableInformationItem** element cannot be used for an underlying "Date" element.

- Question whether the gap list and the *NDD_{CPA-template}* gap list are generally dealt with offline, by a telephone call for example. The alternative option would be to include the gap lists in the *NDD_{CPA-template}*. Previously, the gap list only listed conflicts of the two CPP's but this would allow to try to negotiate over serious problems of the CPA composition.

This question was communicated to the Negotiation of CPA Subcommittee (Schlegel S. 2004, '[ebxml-cppa-negot] CPA composition creates an NDD for the CPA negotiation (revised sample)', ebxml-cppa-negot discussion list, 19 January. Retrieved:

March 2, 2004, from <http://lists.oasis-open.org/archives/ebxml-cppa-negot/200401/msg00013.html>). If an element or attribute which causes a conflict is not listed in both NDD_{CPP} 's, then that element or attribute is not open for negotiation during the CPA negotiation. The ANCPA specification specifies, that only those element and attributes are negotiable, which are listed in the $NDD_{CPA-template}$. If they are not in both NDD_{CPP} 's, the element or attribute is simply not negotiable. This means, that it will not be possible to create a final CPA. This research understands, that the current ANCPA specification does not let the CPA composition algorithm add the conflicting elements or attributes to the $NDD_{CPA-template}$ which were not part of both NDD_{CPP} 's. If that is the case, and the gap list is not addressed after the CPA composition nor before the CPA negotiation, then there will always be conflicts and in such a case, no automatic (without human intervention) successful CPA formation is possible. The ANCPA specification does not provide information, how the gap list is handled nor provides its structure. It is assumed, that human intervention, after the CPA composition process, will be necessary to, first fix the gap list, and second, continue with the CPA negotiation.

- The CPA composition tool will create a CPA template and an $NDD_{CPA-template}$. The ANCPA specification requires the CPA composition to use NDD_{CPP} 's if they are available. Depending on the availability of the NDD_{CPP} 's, the CPA composition process can be started with 0, 1 or 2 NDD_{CPP} 's. The different cases are listed here:
 1. 0 NDD_{CPP} 's
If no NDD_{CPP} 's are available, this research concludes, that the CPA composition will not be able to create a useful $NDD_{CPA-template}$. Simply based on the rule, that a party only wants to negotiate what they have listed in their NDD_{CPP} . If there are no NDD_{CPP} 's then there is no $NDD_{CPA-template}$. And if there is no $NDD_{CPA-template}$ then there is no CPA negotiation.
 2. 1 NDD_{CPP} '
This research assumes, that the CPA composition will not be able to provide a useful $NDD_{CPA-template}$ because the second party does not have an NDD_{CPP} , hence the second party does not want to negotiate at all.
 3. 2 NDD_{CPP} 's
This is expected to be the normal case for a CPA formation process based on the ANCPA specification.

This also rises the issue of public verses non public NDD_{CPP} 's. According to the ANCPA specification the CPA composition must use any available NDD_{CPP} 's. This implies, that the CPA composition tool has to search for the NDD_{CPP} 's.

An attribute in the CPP, which indicates if there is an NDD_{CPP} for this CPP, whether public or not, would help the CPA composition. If the CPA composition tool knows, that there is an NDD_{CPP} , it can go and search it, otherwise it does not have to care to find it. This alone would be helpful. If there is an NDD_{CPP} the CPA composition can get the public NDD_{CPP} or if no public one is found, request an NDD_{CPP} directly from the other party. The situation where more than one public NDD_{CPP} is available for a CPP has to be addressed.

This "requesting an NDD_{CPP} " could be implemented with a collaborative ebXML business process, expressed in a CPA. Alternatively, it could also be implemented with very simple Web Service or simply with Remote Procedure Calls (RPC) or XML RPC.

Another question is, whether the CPA composition algorithm can produce an $NDD_{CPA-template}$ along the CPA template, which is good enough for a CPA negotiation, or if manual editing of the $NDD_{CPA-template}$ is necessary before the CPA negotiation can start.

- This analysis suggests to better separate between the NDD_{CPP} and the $NDD_{CPA-template}$. A different notation will help to clearly indicate of which NDD the specification is talking. This research, based on the results of the CPA composition and in particular of the construction of the conflict file (considered as the gap list), thinks, that the NDD_{CPP} **NegotiableInformationItem** child element needs only one XPath expression, whereas the $NDD_{CPA-template}$ **NegotiableInformationItem** child element might use two XPath expressions (Schlegel S. 2004, '[ebxml-cppa-negot] 1 or 2 XPath expression per negotiable information item', ebxml-cppa-negot discussion list, 3 February. Retrieved: March 2, 2004, from <http://lists.oasis-open.org/archives/ebxml-cppa-negot/200402/msg00000.html>). The reason for this suggestion is, that if a CPA composition tool creates the $NDD_{CPA-template}$, then the CPA composition tool already found and matched (see section 6.2.2 for the CPA composition algorithm) the corresponding elements of the two parity's (for example **CanSend** of one **PartyInfo** with a **CanReceive** of the second **PartyInfo**). To have two XPath expressions in an $NDD_{CPA-template}$ would free the CPA negotiation algorithms to first find the corresponding elements (it might be very likely, that the CPA negotiation algorithms nevertheless have to match the corresponding elements).
- This research brings up the question, whether optional elements and attributes and cardinality problematic elements and attributes are implicitly negotiable. Implicitly negotiable means that the party do not have to list those elements and attributes in a NDD_{CPP} and that those elements and attributes are automatically negotiable.

In general, the analysis of the NDD could not come to a conclusion, whether the current choices of child elements of the **NegotiableInformationItem** element are sufficient to what is required for a CPA negotiation. The $NDD_{CPA-template}$ equivalent taken in this research (the conflict file of the CPA composition) is only a subset of the **NegotiableInformationItem**.

The next section looks at the negotiation messages which are exchanged during a CPA negotiation.

5.4.4 Negotiation Messages

Section 13 of the ANCPA specification introduces the Negotiation Messages which are the business documents of the CPA negotiation ebBP. These messages are used to send negotiation information back and forth during a CPA negotiation. The negotiation messages are XML documents and are the payload of an ebXML message. The **NegotiationContent** element of the negotiation message holds what a party is currently negotiating.

During the negotiation, the negotiable information items of the NDD are the content of the **NegotiationContent** of the "offer" and "counter offer" negotiation messages. There can be more than one negotiable information item negotiated at one time. Having parallel negotiation of negotiable information item requires both negotiation backend applications for even greater sophistication. A negotiation item can either be: "accepted", "updated", "added" or "removed", according to ANCPA specification section 13.1.9. This information is located in the **NegotiationContent** element .