

ComNorm – Beschreibung Mechanismus, Version 1.4

Einleitung

Das Dokument beschreibt einen Mechanismus, der es ermöglicht, auf online verfügbare Produkt- und Artikelinformationen zuzugreifen und mit Anbietern Business-Dokumente auszutauschen. Dabei wird die Koexistenz von unterschiedlichen Katalogsystemen, Portalen, Marktplätzen und Branchensoftwarepaketen zugelassen.

Die Grundidee besteht darin, dass das Onlinesystem auf Anfragen von Client-Anwendungen, maschinell weiterverarbeitbare Antworten liefert. Dabei genügen die Anfragen und erzeugten Antworten vorgegebenen Richtlinien. Diese werden in dem vorliegenden Dokument festgehalten.

Grossen Wert wurde darauf gelegt, dass sich die verwendeten Mechanismen an Standards halten und einfach zu realisieren sind. Solche Standards sind XML und HTTP.

Das Dokument richtet sich an Entwickler, die selbst Anbindungen auf Anwender- oder Anbieterseite realisieren wollen.

Wichtige Information:

Der in diesem und weiteren Dokumenten definierte Kommunikationsmechanismus ist Eigentum der ComNorm GmbH. Interessierte Kreise dürfen die Informationen erst nach Unterzeichnung des 'ComNorm Partner Agreement' verwenden, um entsprechende Anbindungen zu realisieren.

Kontaktadresse: ComNorm GmbH, Turmstrasse 30, 6312 Steinhausen, info@comnorm.ch

Voraussetzungen

Der Kommunikationsmechanismus beruht auf aktuellen Standards. Es wird daher davon ausgegangen, dass der Leser Kenntnisse über die Sprache XML, das Protokoll HTTP und die damit verbundenen Technologien hat.

Festlegungen

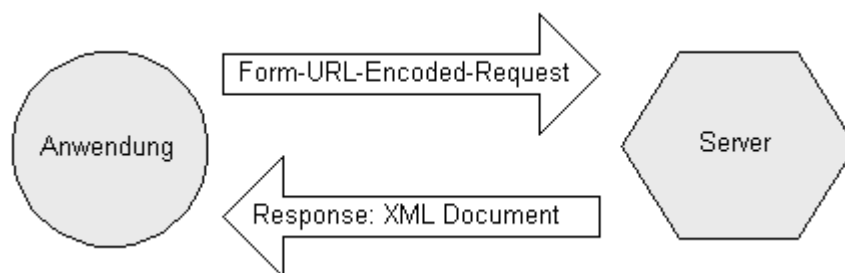
Soweit nicht anders erwähnt, gelten folgende Festlegungen:

- **Sprachcode** (LanguageCode): Es wird der 2-stellige Sprachcodes nach [ISO 639-2] (de=German, en=English, es=Spanish, fr=French, it=Italian,) verwendet.
- **Ländercode** (CountryCode): 3-stellige Ländercodes nach [ISO 3166-1] (CHE=Switzerland, DEU=Germany, FRA=France, ITA=Italy, ESP=Spain, BEL=Belgium, GBR=United Kingdom, ...)
- **Währungscod** (CurrencyCode): 3-stellige Währungen gemäss [ISO 4217] (CHF-Schweizer Franken, ATS=Oesterreichischer Schilling, DEM=Deutsche Mark, EUR=Euro, ...)
- **Einheiten** (UnitCode): Code gemäss separater Codeliste (CMT=Zentimeter, MTQ=Kubikmeter, GRM=Gramm, MMT=Millimeter, C62=Stück, WTT=Watt, ...)
- **Attribute** (AttributeCode): Code gemäss separater Codeliste (LENGTH=Länge, WIDTH=Breite, DEPTH=Tiefe/Dicke/Höhe, VOLUME=Volumen, ...)
- **Datums- und Zeitangaben**: ISO 8601 (Datum: 2021-02-15, Zeit: 15:05:27)

- Zeichen-Codierung: ISO-8859-1 (Es soll jedoch bei der Entwicklung beachtet werden, dass ein zukünftiger Wechsel auf ‚utf-8‘ leicht möglich ist).
- Zahlenformat: Als Dezimaltrennzeichen wird der Punkt verwendet. Ein Zifferngruppierungszeichen fällt weg. Bsp.: '<Value>1753.30</Value>'

Mechanismus

Das Anwendersystem fragt das Anbietersystem nach Informationen ab, indem eine HTTP- oder HTTPS-Anfrage mit dem POST-Mechanismus an das Anbietersystem gestellt wird. Im Meldungs-Body der Anfrage werden die für die jeweilige Anfrage relevanten Parameter mitgegeben. So kann das Anbietersystem diese bearbeiten. Es bereitet ein XML-Dokument mit den gewünschten Informationen auf und gibt es an das Anwendersystem zurück.



Unterschiedliche Methoden/Funktionen des Anbietersystems lassen sich mittels entsprechender Parameter ansprechen.

Die Parameter und deren Werte werden für die Übertragung Form-codiert (application/x-www-form-urlencoded). Die Reihenfolge der Parameter ist relevant. Gewisse Parameter sind zwingend, andere optional.

Beispiel:

```
Method=GetChapterList&ChapterId=7
```

Einstiegspunkt

Um mit einem Anbietersystem in Kontakt zu treten, muss bekannt sein, wohin die Anfragen gestellt werden müssen. Dazu muss eine eindeutige Internetadresse (URL) des Services bekannt sein. Beispiele sind z.B.:

- <http://www.lieferant.ch/xmlcatalog.pl>
- <http://www.marktplatz.ch/servlet/catalog>
- <https://www.handel.com/service.asp>

Um herauszufinden, welche Anbieter den Zugriff als Dienst zur Verfügung stellen, und via welche Internetadressen diese zugreifbar sind, ist ein Mechanismus via eine zentrale Liste definiert. Siehe dazu 'Ermittlung von Anbieterdiensten'.

Arten der Kommunikation

ComNorm unterscheidet zwei Arten des Datenaustausches:

- **Katalogorientierte Kommunikation:** Bei dieser Art werden genau spezifizierte Abfragen an einen Server gemacht, die als Ergebnis katalogbasierte Detailinformationen zurückbringen (z.B. Ermittlung der Beschreibung eines Produktes, Ermittlung der Unterkapitel eines Kapitels).
- **Dokumentbasierte Kommunikation:** Hier werden gesamte XML-Dokumente an den Server übergeben oder abgeholt (z.B. Bestellung wird ausgelöst und als Ergebnis wird eine Bestellbestätigung bereitgestellt).

Sprachwahl

Das abfragende System kann das Anbietersystem darauf hinweisen, dass es die in der Antwort enthaltenen Texte gerne in einer bestimmten Sprache haben möchte. Dazu kann im HTTP-Header der Anfrage der Wert des Feldes 'Accept-Language' (bevorzugte Sprache) festgelegt werden. Als Sprachcode werden zwei Zeichen gemäss ISO 639 verwendet (en=English, de=German, fr=French).

```
POST /xmlservice.pl HTTP/1.1
Accept: text/xml
Accept-Language: de
Content-Type: application/x-www-form-urlencoded
Host: localhost
Content-Length: 68
...
```

Wird eine Anfrage für eine bestimmte Sprache gemacht, so versucht das Anbietersystem die Texte in der gewünschten Sprache zurückzugeben.

Das antwortende System bestimmt mit dem HTTP-Header-Feld 'Content-Language' die Sprache, in der es das Dokument dann aufbereitet hat.

```
HTTP/1.0 200
Date: Monday, 05-Sep-99 12:12:47 GMT
Content-Language: de
Content-Type: text/xml
Content-Length: 5319
Last-Modified: Thursday, 01-Jun-98 09:44:17 GMT
...
```

Meldungsumschlag

Eine Anfrage an das Anbietersystems liefert ein Antwort-Dokument in XML-Struktur zurück. Es hat sich nun gezeigt, dass neben dem Dokument aber auch noch Statusinformationen dem Anfragesystem mitgeteilt werden müssen. Deshalb werden alle Antworten in einen Art Umschlag verpackt.

Beispiel: Verpackungsstruktur um die Dokumentdaten

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ResponseEnvelope>
  <ResponseHeader>
    <ResultCode>200</ResultCode>
    <ResultMessage>OK</ResultMessage>
  </ResponseHeader>
  <ResponseBody>
    ... Hier kommen die eigentlichen Dokumentdaten ...
  </ResponseBody>
</ResponseEnvelope>
```

Das Element 'ResponseBody' enthält das eigentliche Antwort-Dokument. Im Header-Teil werden Angaben gemacht über den Erfolg der auszuführenden Aktion. Als 'ResultCode' finden ähnliche Codes Verwendung, wie im HTTP-Protokoll. Momentan gilt bei ComNorm aber eine Beschränkung auf einige wenige:

Code	Bedeutung
200	OK
400	Syntax-Fehler, der Server hat die Anfrage nicht verstanden. Parameter sind falsch oder fehlen.
401	Der Client muss sich erst authentifizieren, bevor ihm der Zugriff auf das gewünschte Dokument gestattet wird.
403	Der Server verweigert den Zugriff, auch eine Authentifizierung hilft da nicht.
404	Der Server hat das angegebene Dokument nicht gefunden (Es sind keine Daten vorhanden, die der Anfrage genügen oder die Methode wird nicht unterstützt).
500	Interner Fehler des Servers, aufgrund dessen die Anfrage nicht bearbeitet werden konnte.
501	Der Server unterstützt die geforderte Aktion nicht.

Das Element 'ResponseBody' erscheint nur, wenn der Wert von 'ResultCode' gleich 200 ist. Ansonsten wird es weggelassen. 'ResultMessage' wird verwendet, um den Anwender mit einer Meldung zu informieren, wenn das System nicht selber geeignet reagieren kann.

Beispiel: Anbietersystem verlangt Authentifizierung

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ResponseEnvelope>
  <ResponseHeader>
    <ResultCode>401</ResultCode>
    <ResultMessage>Anmeldung erwünscht</ResultMessage>
  </ResponseHeader>
</ResponseEnvelope>
```

Katalogorientierte Kommunikation

Der Bereich Katalogzugriff umfasst folgende Abfragen/Meldungen:

- **GetChapterList:** Haupt- und Unterkapitel eines Kataloges ermitteln
- **GetProductList:** Produkte eines Kapitels ermitteln
- **GetProductInfo:** Detailinformationen zu einem Produkt
- **SearchProductList:** Produktsuche

Die unten aufgeführten Abfragen haben einen optionalen Parameter 'SessionKey'. Dieser wird bei einer Anmeldung beim Anbietersystem zurückgegeben und identifiziert den Anwender und dessen Sitzung. Siehe dazu das Kapitel 'Authentifizierung'.

Wird der Parameter 'SessionKey' weggelassen oder hat keinen Wert, so wird davon ausgegangen, dass es sich um einem anonymen Zugriff handelt.

GetChapterList: Haupt- und Unterkapitel eines Kataloges

Mit der Methode 'GetChapterList' ermittelt die Clientanwendung, welches die Struktur in einem Katalog ist.

Parameter 1 (fix)

Method=GetChapterList

Parameter 2 (optional)

ChapterId=xxx

Parameter 3 (optional)

SessionKey=sss

Werte:

xxx=Identifikation des Kapitels (C16). Hat xxx den Wert ,0' oder ist leer oder wird nicht angegeben, so wird die Haupt-Kapitel-Struktur zurückgegeben. Ansonsten die Unterkapitel zum verlangten Kapitel.
sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'ChapterList' aufbereitet. Der Wert des Attributes 'ChapterId' wird für weitere Abfragen verwendet.

Enthält ein Kapitel keine Daten oder ist die Kapitelidentifikation ungültig, so kommt eine leere Liste zurück. Das Element 'ChapterType' dient dazu, anzugeben ob das Kapitel weitere Unterkapitel und oder Produkteinträge enthält (0 = Unbestimmt, 1 = Kapitel, 2 = Produkte, 3 = Kapitel und Produkte). Fehlt diese Angabe, so muss mit Folgeabfragen ermittelt werden, ob entsprechende Informationen vorliegen.

Beispiel: Method=GetChapterList&ChapterId=2

```

...
<ChapterList>
  <ChapterItem ChapterId="4">
    <ChapterNumber>100</ChapterNumber>
    <ChapterName>Griffe und Knoepfe fuer Moebel</ChapterName>
    <ChapterText></ChapterText>
    <ChapterURL>http://www.kat.ch/file/griff.html</ChapterURL>
    <ChapterType>1</ChapterType>
    <ChapterThumbNailURL>http://www.kat.ch/img/g77.gif</ChapterThumbNailURL>
  </ChapterItem>
  <ChapterItem ChapterId="24">
    <ChapterNumber>200</ChapterNumber>
    <ChapterName>Schlösser</ChapterName>
    <ChapterText>Schloesser zum Einbau in Moebel</ChapterText>
    <ChapterURL>http://www.kat.ch/file/schloss.html</ChapterURL>
    <ChapterType>2</ChapterType>
    <ChapterThumbNailURL>http://www.kat.ch/img/s94.gif</ChapterThumbNailURL>
  </ChapterItem>
  ...
</ChapterList>
...
```

Beispiel leere Liste

```

...
<ChapterList>
</ChapterList>
...
```

GetProductList: Produkte eines Kapitels

Die Methode 'GetProductList' ermittelt, welches die in einem Katalog in einem bestimmten Kapitel vorhandenen Produkte sind.

Parameter 1 (fix)
Method=GetProductList
Parameter 2
ChapterId=xxx
Parameter 3 (optional)
SessionKey=sss
Werte:

xxx=Identifikationsnummer des Kapitels (Ganzzahliger Wert) für das die Produkte ermittelt werden sollen.
 sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'ProductList' aufbereitet.

Beispiel: Method=GetProductList&ChapterId=4

```

...
<ProductList>
  <ProductItem>
    <ProductNumber>43.389.03</ProductNumber>
    <ProductName>Aufsatz-Muschelgriff</ProductName>
  </ProductItem>
  <ProductItem>
```

```

        <ProductNumber>43.389.03</ProductNumber>
        <ProductName>Möbelgriff Stahl weiss Typ 5</ProductName>
    </ProductItem>
    ....
</ProductList>
...

```

GetProductInfo: Produktinformationen

Die Methode 'GetProductInfo' ermittelt die Detailinformationen zu einem Produkt.

<u>Parameter 1 (fix)</u>	<u>Parameter 2</u>	<u>Parameter 3 (optional)</u>	<u>Parameter 4 (optional)</u>
Method=GetProductInfo	ProductNumber=yyy	ParamData=zzz	SessionKey=sss

Werte:

yyy=Eindeutige Produktnummer
 zzz=URL-codierte Parameterisierungs-Angaben
 sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'ProductInfo' aufbereitet.

Das Vorhandensein des Tags 'OrderNumber' bestimmt, ob es sich um ein bestellbares Produkt handelt. Für die Bestellung sind immer 'OrderNumber' und die in 'ParameterData' enthaltenen Informationen relevant. Siehe dazu das Kapitel 'Parameterisierung'.

Sollen für den Anwender gewisse Informationen (z.B. Preise) nicht einsehbar sein, so werden die entsprechenden Elemente weggelassen. Das gleiche gilt wenn die Standardwerte gelten.

Der Bereich 'AttributeData' dient dazu, zusätzliche Informationen über das Produkt beizufügen und ist beliebig erweiterbar. So kann ein Anbieter z.B. Eigenschaften eines Produktes festlegen die nur für spezielle Branchenlösungen relevant sind.

Preisangaben sind immer in Bestelleinheiten.

Beispiel: Method=GetProductInfo&ProductNumber=K732

```

...
<ProductInfo>
  <ProductNumber>43.139.44</ProductNumber>
  <ProductName>Tisch Typ Rondello</ProductName>
  <ProductText>Tisch Buche natur mit Holzfuesen</ProductText>
  <ProductURL>http://www.katalog.ch/produkt/p5678.html</ProductURL>
  <BaseUnitCode>C62</BaseUnitCode>
  <OrderNumber>43.139.44</OrderNumber>
  <ConfigurationData>
    <ConfigurationURL>http://www.katalog.ch/config.exe?P=5678</ConfigurationURL>
    <ParamData>L=1600/B=800</ParamData>
  </ConfigurationData>
  <PackageUnitCode>C62</PackageUnitCode>
  <BaseQuantityPerPackageUnit>1</BaseQuantityPerPackageUnit>
  <OrderUnitCode>C62</OrderUnitCode>
  <PackageQuantityPerOrderUnit>1</PackageQuantityPerOrderUnit>
  <OrderMinQuantity>1</OrderMinQuantity>
  <OrderIntervalQuantity>1</OrderIntervalQuantity>
  <ProductGroupCode></ProductGroupCode>
  <AttributeData>
    <AttributeItem>
      <AttributeCode>LENGTH</AttributeCode>
      <AttributeName>Länge</AttributeName>
      <UnitCode>MMT</UnitCode>
      <AttributeValue>1600</AttributeValue>
    </AttributeItem>
    <AttributeItem>

```

```

        <AttributeCode>WIDTH</AttributeCode>
        <AttributeName>Breite</AttributeName>
        <UnitCode>MMT</UnitCode>
        <AttributeValue>800</AttributeValue>
    </AttributeItem>
</AttributeData>
<PriceData>
    <PriceItem>
        <CurrencyCode>CHF</CurrencyCode>
        <PriceType>0</PriceType>
        <FromOrderQuantity>1</FromOrderQuantity>
        <PriceValue>543.00</PriceValue>
        <TaxRate>0.075</TaxRate>
        <ExpiresDate>2001-3-3</ExpiresDate>
    </PriceItem>
    <PriceItem>
        <CurrencyCode>CHF</CurrencyCode>
        <PriceType>0</PriceType>
        <FromOrderQuantity>10</FromOrderQuantity>
        <PriceValue>498.00</PriceValue>
        <TaxRate>0.075</TaxRate>
        <ExpiresDate>2001-3-3</ExpiresDate>
    </PriceItem>
    <PriceItem>
        <CurrencyCode>CHF</CurrencyCode>
        <PriceType>1</PriceType>
        <FromOrderQuantity>1</FromOrderQuantity>
        <PriceValue>600.00</PriceValue>
        <TaxRate>0.075</TaxRate>
        <ExpiresDate>2001-3-3</ExpiresDate>
    </PriceItem>
</PriceData>
</ProductInfo>
...

```

SearchProductList: Produktsuche

Die Methode 'SearchProductList' ermittelt die Produkte, die vorgegebenen Suchkriterien genügen.

<u>Parameter 1 (fix)</u>	<u>Parameter 2</u>	<u>Parameter 3</u>	<u>Parameter 4 (optional)</u>
Method=SearchProductList	Field=fff	Pattern=ppp	SessionKey=sss

Werte:

fff=Feld, Attribut, Tag in dem gesucht werden soll (z.B. ProductNumber, ProductName, ProductText).
 ppp=Suchmuster (z.B. Produktnummer oder ein Teil davon)
 sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'ProductList' aufbereitet.
 Bei der Suche wird, wenn möglich keine Unterscheidung zwischen Gross- und Kleinbuchstaben gemacht.
 Ebenso sollte jegliches Vorkommen des Suchbegriffes ermittelt werden und nicht nur wenn ein exaktes Uebereinstimmen vorliegt.
 Suchen nach mehreren Begriffen und Kombinationen muss nicht möglich sein.

Unterstützt ein Anbietersystem die Suche nach einem Suchkriterium nicht, so gibt es den HTTP-Code 501 zurück. Wird nichts gefunden, so kommt als Ergebnis eine leere Liste.

Das Anbietersystem soll als Suchergebnis höchstens 200 Einträge zurückgeben. Dies, um die Zeit für die Aufbereitung und Datenübertragung in einem sinnvollen Rahmen zu halten.

Minimal die Suche nach 'ProductNumber', 'ProductName' und 'ProductText' sollte von einem Anbietersystem unterstützt werden.

Enthält der Parameter 'Field' keinen Wert, so wird für eine Volltextsuche angefragt. Es ist Sache des Anbietersystems, wie auf diese Anforderung eingegangen wird. Gewisse können diese Operation sicher nicht durchführen, und werden z.B. das gleiche Ergebnis wie bei einer Suche nach Produkttext und Produktnummer zurückgeben.

Beispiele:

Method=SearchProductList&Field=ProductName&Pattern=Glas

Method=SearchProductList&Field=ProductNumber&Pattern=4711

Method=SearchProductList&Field=ProductText&Pattern=Unterbau

Dokumentorientierte Kommunikation

Mit der katalogorientierten Kommunikation kann das Anwendersystem beim Anbietersystem Informationen abholen. Oft ist es aber auch nötig, dass bei Bedarf Informationen, z.B. eine Bestellung wieder zurückfliessen kann.

Der Bereich Dokumentaustausch umfasst dazu folgende Anfragen/Meldungen:

- **GetServiceInfo:** Informationen über die Transfereigenschaften des Anbietersystems ermitteln
- **PostDocument:** Dokument, z.B. Bestellung oder Preisanfrage an das Anbietersystem übermitteln
- **GetDocumentList:** Vom Anbietersystem bereitgestellte Dokumente auflisten
- **GetDocumentInfo:** Bestimmtes Dokument vom Anbietersystem abholen

Beispiel: Eine Branchenlösung stellt eine Bestellung zusammen. Mit 'PostDocument' wird diese an das Anbietersystem übermittelt. Das System verarbeitet diese und erstellt darauf hin entsprechende Folgedokumente, z.B. Auftragsbestätigung, Rechnung. Diese Dokumente könnten nur per Post oder als Mailanhang dem Besteller zugestellt werden. Eine andere oder zusätzliche Möglichkeit ist, dass das Anbietersystem diese Dokumente für den Abruf bereithält. Eine entsprechend programmierte Branchenlösung kann solche Dokumente direkt übernehmen und weiterverarbeiten.

GetServiceInfo: Informationen über Anbietersystem ermitteln

Mit der Methode 'GetServiceInfo' kann u.a. ermittelt werden, was für Möglichkeiten der Dokumentübermittlung das Onlinesystem zulässt.

Parameter 1 (fix)

Method=GetServiceInfo

Parameter 2 (optional)

SessionKey=sss

Werte:

sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'ServiceInfo' aufbereitet.

Im Transferteil wird spezifiziert, auf welche Art Preisanfragen und Bestellungen übermittelt werden können. Eine Branchenlösung kann so die entsprechenden Optionen anbieten. Normalerweise können Dokumente via die Internetadresse des Einstiegspunktes transferiert werden. Mit dem Element 'OnlineTo' kann dies bei Bedarf geändert werden.

Bietet ein Anbieter zusätzlich einen individuellen Katalogen an (Eintrag 'Catalog' in ServiceList), so kann im Element 'DefaultType' optional angegeben werden, welches der bevorzugt zu verwendende Katalog ist.

Beispiel: Method=GetServiceInfo

```
...
<ServiceInfo>
  <Supplier>
    ...
  </Supplier>
  <Transfer>
    <OnlineTo>http://www.handel.ch/access.cf</OnlineTo>
    <MailTo>order@handel.ch</MailTo>
    <FaxTo>+41 32 123 232</FaxTo>
    <TelTo>+41 32 123 111</FaxTo>
    <LetterTo>Handel AG, Hofgasse 7, 3000 Bern</FaxTo>
  </Transfer>
  <Catalog>
    <DefaultType>1</DefaultType>
  </Catalog>
</ServiceInfo>
...
```

PostDocument: Dokument übermitteln

Mit der Methode 'PostDocument' kann an das Onlinesystem ein aufbereitetes Dokument (z.B. Bestellung oder Preisanfrage) übergeben werden.

Parameter 1 (fix)**Method=PostDocument****Parameter 2****DocumentData=xxx****Parameter 3 (optional)****SessionKey=sss****Werte:**

xxx=XML-Dokument (form-encoded) gemäss DTD 'Order' (Bestellung) oder 'Inquiry' (Preisanfrage).
sss=Identifikation des Benutzers und Session.

Antwort:

Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'PostAcknowledgement' aufbereitet.

Das Zielsystem kann bei Bedarf im Element 'AcknowledgementURL' eine Internetadresse zurückgeben, die vom Clientsystem angesprungen werden soll, um z.B. auf einer speziellen Seite die Bestellung nochmals zu bestätigen.

Will dies das Clientsystem nicht, so kann es dies als Wunsch mitteilen, indem sie im Kopf der Bestellung oder Anfrage im Element 'ProcessingHint' den Wert 'A' einträgt. Die Verarbeitung sollte dann automatisch und ohne Rückgabe einer AcknowledgementURL erfolgen. Weitere Optionen werden evt. noch folgen.

Beispiel: Method=PostDocument&DocumentData=...

```
...
<PostAcknowledgement>
  <PostCode>1</PostCode>
  <PostMessage>Das Bestellung ist eingegangen.</PostMessage>
  <DocumentInfo>
    <DocumentType>ORDER</DocumentType>
    <DocumentNumber>R4.389</DocumentNumber>
    <DocumentName>Bestellung Müller</DocumentName>
    <DocumentDate>2000-03-20</DocumentDate>
    <DocumentURL>http://www.katalog.ch/file/R4389.xml</DocumentURL>
  </DocumentInfo>
  <ProcessingInfo>
    <DocumentItem DocumentId="667">
      <DocumentType>CONFIRMATION</DocumentType>
      <DocumentNumber>A4.389-4</DocumentNumber>
      <DocumentName>Auftragsbestätigung Müller</DocumentName>
      <DocumentDate>2000-03-15</DocumentDate>
      <DocumentURL>http://www.katalog.ch/file/A43894.html</DocumentURL>
    </DocumentItem>
  </ProcessingInfo>
  <AcknowledgementURL>http://www.kat.ch/con.pl?id=xder7re</AcknowledgementURL>
</PostAcknowledgement>
...
```

GetDocumentList: Dokumentliste ermitteln

Mit der Methode 'GetDocumentList' kann ermittelt werden, was für Dokumente der Anbieter auf seinem Onlinesystem zum Abruf zur Verfügung hält.

Parameter 1 (fix)**Method=GetDocumentList****Parameter 2 (optional)****DocumentNumber=nnn****Parameter 3 (optional)****DocumentType=rrr**

<u>Parameter 4 (optional)</u> FromDate=fff	<u>Parameter 5 (optional)</u> ToDate=ttt	<u>Parameter 6 (optional)</u> SessionKey=sss
<p>Werte:</p> <p>nnn = Eindeutige Nummer des Dokumentes (String) von dem abhängige Dokumente ermittelt werden sollen.</p> <p>rrr = Einschränkung der in der vorkommenden Dokumenttypen (DELIVERY, INVOICE, ...)</p> <p>fff = Von-Datumseinschränkung (Es werden nur Dokumente ermittelt, die grösser gleich fff sind. Ohne Wert erfolgt keine Von-Einschränkung)</p> <p>ttt = Bis-Datumseinschränkung (Es werden nur Dokumente ermittelt, die kleiner gleich ttt sind. Ohne Wert erfolgt keine Bis-Einschränkung)</p> <p>sss = Identifikation des Benutzers und Session.</p>		
<p>Antwort:</p> <p>Als Antwort wird ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'DocumentList' aufbereitet.</p> <p>Dabei erscheinen Dokumente, die mit dem als Parameter angegebenen Dokument eine Beziehung haben. Wenn z.B. die Dokumentnummer zu einer Bestellung gehört, so werden Auftragsbestätigung, Rechnung und evt. Lieferschein aufgelistet.</p> <p>Wird als 'DocumentNumber' kein Wert übergeben oder der Parameter weggelassen, so erfolgt eine Auflistung der aktuellsten Dokumente gemäss den anderen Parametern, maximal aber etwa 200.</p>		
<p>Beispiel: <u>Method=GetDocumentList&DocumentNumber=812&FromDate=2012-03-01</u></p> <pre> ... <DocumentList> <DocumentItem DocumentId="812-343A95"> <DocumentType>ORDER</DocumentType> <DocumentNumber>R4.389</DocumentNumber> <DocumentName>Bestellung Müller</DocumentName> <DocumentDate>2012-03-20</DocumentDate> <DocumentURL>http://www.katalog.ch/file/R4389.xml</DocumentURL> </DocumentItem> <DocumentItem DocumentId="667-249689"> <DocumentType>CONFIRMATION</DocumentType> <DocumentNumber>A4.389-4</DocumentNumber> <DocumentName>Auftragsbestätigung Müller</DocumentName> <DocumentDate>2012-03-15</DocumentDate> <DocumentURL>http://www.katalog.ch/file/A43894.xml</DocumentURL> <AlternateDocumentURL>http://www.katalog.ch/file/A43894.pdf </AlternateDocumentURL> </DocumentItem> ... </DocumentList> ... </pre>		

GetDocumentInfo: Dokument abholen

Die Methode 'GetDocumentInfo' ermittelt die Detailinformationen eines Standard-Dokumentes.

<u>Parameter 1 (fix)</u> Method=GetDocumentInfo	<u>Parameter 2</u> DocumentNumber=yyy	<u>Parameter 3 (optional)</u> SessionKey=sss
<p>Werte:</p> <p>yyy=Eindeutige Nummer des Dokumentes (String)</p> <p>sss=Identifikation des Benutzers und Session.</p>		

Antwort:

Als Antwort wird je nach Dokumenttyp ein in einen Umschlag eingebettetes XML-Dokument gemäss DTD 'Order' (Bestellung), 'Confirmation' (Auftragsbestätigung), 'Inquiry' (Preisanfrage), 'Offer' (Offerte), 'Invoice' (Rechnung) oder 'Delivery' (Lieferung) aufbereitet.

Beispiel: Method=GetDocumentInfo&DocumentNumber=R748

```
...
<Offer>
  ...
</Offer>
...
```

Authentifizierung

Gewisse Anbieter verlangen, dass von Anwenderseite her eine Anmeldung erfolgt. Dies, um sicherzustellen, dass nur berechtigte Anwender Informationen einsehen können, oder um individuell gestaltete Antworten (z.B. Preise, Produktpalette) aufzubereiten.

Möchte ein Anbietersystem erzwingen, dass eine erste oder erneute Authentifizierung erfolgt, so gibt er auf die Anfrage den Statuscode '401 Authorisation Required' zurück.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ResponseEnvelope>
  <ResponseHeader>
    <ResultCode>401</ResultCode>
    <ResultMessage>Anmeldung erforderlich</ResultMessage>
  </ResponseHeader>
</ResponseEnvelope>
```

Das Anwendersystem ermittelt beim Anwender die benötigten Angaben und sendet diese mit der Methode 'GetLoginInfo' an das Anbietersystem. Dieses gibt als Antwort einen SessionKey zurück. Dieser muss in Folgeoperationen als Zusatzparameter Verwendung angeben werden.

GetLoginInfo: Benutzerauthentifizierung

Mit der Methode 'GetLoginInfo' kann beim Onlinesystem eine Benutzeranmeldung vorgenommen werden.

Parameter 1 (fix)**Method=GetLoginInfo****Parameter 2****CN=ccc****Parameter 3****UI=uuu****Parameter 3****PW=ppp****Werte:**

ccc=Kundennummer (ClientNumber)
 uuu=Benutzer (UserIdentification).
 ppp=Passwort (PassWord).

Antwort:

Als Antwort wird ein in einen Umschlag verpacktes XML-Dokument gemäss DTD 'LoginInfo' aufbereitet.

Der Wert von 'SessionKey' muss in Folgeabfragen als Zusatzparameter mitgegeben werden und hat Gültigkeit, bis das Anbietersystem eine erneute Anmeldung verlangt.

Beispiel: Method=GetLoginInfo&CN=MH77&UI=Muster.Hans&PW=12345

```
<?xml version="1.0" encoding="ISO-8859-1"?>
...
<LoginInfo>
  <SessionKey>Eser3se4332XSER</SessionKey>
</LoginInfo>
```

Zusätzliche Sicherheit wird erreicht, wenn SSL (Secure Socket Layers) verwendet wird. Dann wird die Kommunikation über verschlüsselte Socket-Verbindungen abgewickelt. Als Protokollidentifikator gilt dann 'https'.

Ermittlung von Anbieterdiensten

Verschiedene Anbieter stellen den Katalogzugriff und den Dokumentaustausch als Internetdienste zur Verfügung. Monatlich kommen neue Dienste hinzu, bestehende sind via neue Adressen zugreifbar oder verändern sich in ihrem Funktionsumfang. Vom Anwender kann nicht verlangt werden, dass er die Lokationen der Dienste manuell in seine Branchenlösung einträgt. Daher muss ein Mechanismus zur Verfügung stehen, mit dem sich bestehende und neu hinzukommende Dienste ermitteln lassen.

Dieser Mechanismus besteht darin, dass eine Liste von offiziell verfügbaren Diensten von der Firma ComNorm unterhalten wird. Wenn ein Anbieter eine korrekt funktionierende Anbindung realisiert hat und die vereinbarten Abmachungen erfüllt, so wird er in die Liste aufgenommen.

Die Firma ComNorm stellt die Liste auf ihrer Internet-Site unter '<http://www.comnorm.ch/ServiceList.xml>' als XML-Dokument zur Verfügung und aktualisiert sie periodisch. Dabei wird gewährleistet, dass jeder Service eine eindeutige Identifikationsnummer (ServiceId) erhält. Auch wenn der Service z.B. seine Internetadresse wechselt, kann so der Bezug jederzeit wieder hergestellt werden.

In der Branchenlösung muss jetzt nur eine Programmooption zur Verfügung stehen, mit der die aktuellste Liste gelesen und verarbeitet werden kann. So ist der Anwender immer über die verfügbaren Dienste informiert.

Das XML-Dokument 'ServiceList.xml' hat folgende Struktur:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ServiceList ListDate="2015-06-22">
  <ServiceItem ServiceId="342">
    <ServiceName>Markt AG, Zürich</ServiceName>
    <ServiceHolder>Markt AG, Tannweg 99, 8000 Zürich</ServiceHolder>
    <ServiceDesc>Beschlüsse für den Holzverarbeiter</ServiceDesc>
    <ServiceVersion>1.0</ServiceVersion>
    <ServiceEntryPoint>http://www.marktplatz.ch/xy/service.asp</ServiceEntryPoint>
    <ServiceURL>http://www.marktplatz.ch/info.html</ServiceURL>
  </ServiceItem>
  <ServiceItem ServiceId="765">
    <ServiceName>Bau und Kunst</ServiceName>
    <ServiceName LanguageCode="en">Construction and Art</ServiceName>
    <ServiceHolder>BauKunst GmbH, Im Tal, 3000 Bern</ServiceHolder>
    <ServiceDesc>Kunstobjekte auf Holzbasis</ServiceDesc>
    <ServiceDesc LanguageCode="en">Wood and Art</ServiceDesc>
    <ServiceVersion>1.0</ServiceVersion>
    <ServiceEntryPoint>https://www.baukunst.ch/catalog/cat.cf</ServiceEntryPoint>
    <ServiceURL>http://www.baukunst.ch</ServiceURL>
    <CatalogURL>http://catalog.baukunst.ch</CatalogURL>
  </ServiceItem>
  ...
  <ServiceArea AreaCode="DE">
    <ServiceEntry ServiceId="100133">
      <ServiceName>Einkauf AG</ServiceName>
      ...
    </ServiceEntry>
    <ServiceEntry ServiceId="100785">
      <ServiceName>Müller Holz</ServiceName>
      ...
    </ServiceEntry>
    ...
  </ServiceArea>
  ...
</ServiceList>
```

Es ist zu beachten, dass das Element 'ServiceEntryPoint' auf die Internetadresse verweist, via die der Dienst angesprochen wird.

Bietet ein Anbieter einen individuellen Katalog an, so hat ein ServiceEintrag ein zusätzliches Element 'CatalogURL'.

Um erweiterte Anforderungen (z.B. länderspezifische Anbieterlisten) abzudecken können ab Version 1.4 zusätzliche Gruppierungen 'ServiceArea' enthalten sein. Die darin vorkommenden Elemente 'ServiceEntry' weisen die gleichen Unterelemente auf wie 'ServiceItem'

Die Elemente 'GroupName', 'ServiceName' und 'ServiceDesc' könnten mehrfach vorkommen, so dass, mit dem Attribut 'LanguageCode' versehen, die Abdeckung mehrerer Sprachen möglich wäre.

Die Serviceliste ist ein kritisches Element für das Funktionieren des Datenaustausches. Ist aus irgendwelchen Gründen die ComNorm-Site (,www.comnorm.ch') einmal nicht verfügbar, so können Anwendungen evt. nicht mehr funktionieren. Aus diesem Grund wird eine Kopie der Liste noch auf einem anderen Host unter ,<http://www.mronmoc.ch/ServiceList.xml>' redundant bereitgestellt.

Schemen, Datenstrukturen

Die als XML-Schemen festgelegten Strukturen sind separat dokumentiert. Folgende finden Verwendung:

Systeminformationen

- **ServiceList:** Liste von Anbieterdiensten die eine Schnittstelle gemäss Norm realisiert haben
- **ServiceInfo:** Informationen über ein Anbietersystem

Online-Katalogzugriff

- **ChapterList:** Liste von Kapiteln
- **ProductList:** Liste von Produkten
- **ProductInfo:** Detail-Informationen zu einem Produkt

Dokumenttransfer und Dokumentzugriff

- **PostAcknowledgement:** Bestätigung Dokumentübertragung
- **DocumentList:** Liste von Dokumenten
- **Order:** Bestellung
- **Confirmation:** Auftragsbestätigung
- **Inquiry:** Preisanfrage
- **Offer:** Angebot, Offerte
- **Invoice:** Rechnung
- **Delivery:** Lieferung

Codierung von Attributen

Verschiedene Dokumente/Struktur (z.B. ProductInfo) enthalten die Unterstruktur ,AttributItem' um abhängige Informationen festzuhalten.

Beispiel:

```
<AttributeData>
  <AttributeItem>
    <AttributeCode>LENGTH</AttributeCode>
    <AttributeName>Länge</AttributeName>
    <UnitCode>MMT</UnitCode>
    <AttributeValue>1600</AttributeValue>
  </AttributeItem>
  ...
</AttributeData>
```

Um die Daten austauschbar zu machen, gibt es eine Reihe von festgelegten Codes (siehe separates Dokument ‚Codelisten‘).

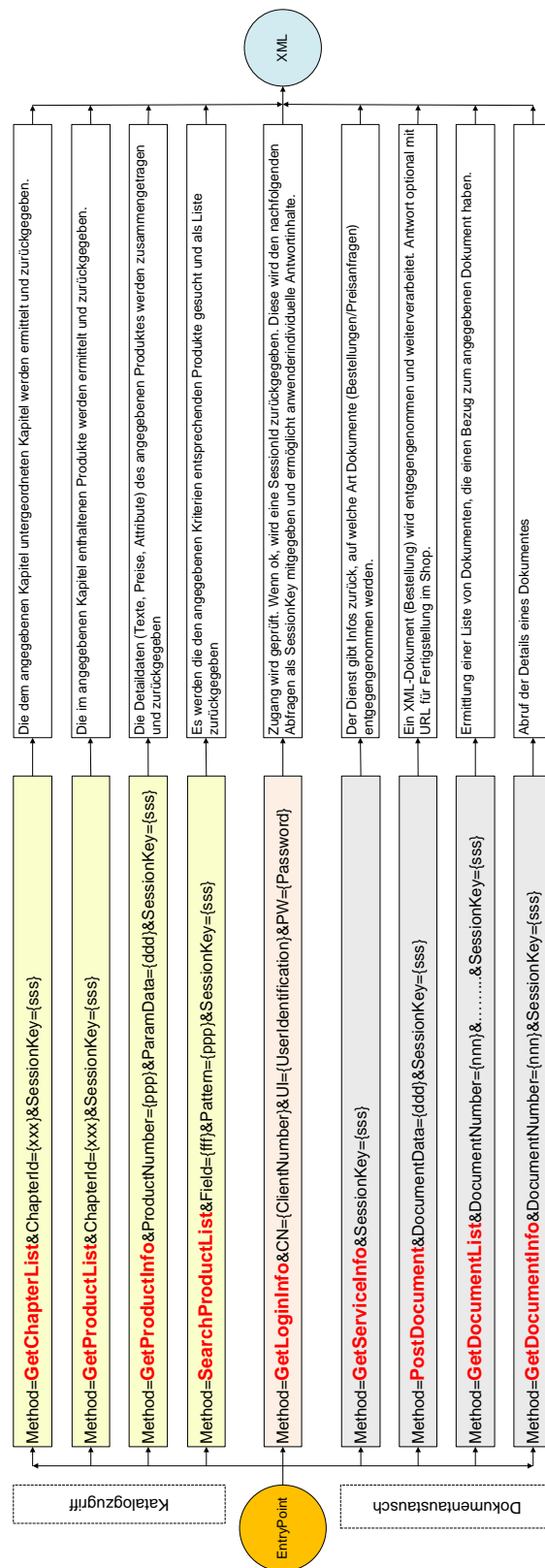
Codierung von Einheiten

Nachfolgend ein kleiner Ausschnitt aus der Einheitentabelle, wie sie das 'United Nations Centre for Trade Facilitation and Electronic Business' empfiehlt.

Code (UnitCode)	Name	Symbol
AMP	ampere	A
BO	bottle	
BX	box	
CT	carton	
CMT	centimetre	cm
MTQ	cubic metre	m ³
DD	degree	°
DZN	dozen	DOZ
HUR	hour	h
KGM	kilogram	kg
MTR	metre	m
MLT	millilitre	ml
MIN	minute	min
MON	month	
PK	pack	

PR	pair	
P1	percent	
C62	piece	
PG	plate	
RG	ring	
RO	roll	
SA	sack	
SEC	second	s
SET	set	
TN	tin	
TNE	tonne (metric ton)	t
TU	tube	
C62	unit	
VLT	volt	V
WTT	watt	W
ANN	year	a

Die weiteren Codierungen sind in einem separaten Dokument ‚Codelisten‘ aufgeführt oder unter http://www.unece.org/cefact/codesfortrade/codes_index.htm abrufbar.

Kommunikationschema

Änderungen, History

Änderungen gegenüber Vorgängerversionen	Version
Dieses Dokument als Diskussionsgrundlage	1.2
Anpassungen	1.2.1
Anpassungen	1.3
Anpassungen (Serviceliste: CatalogURL und Gruppierungen, Element 'Catalog/ServiceType' in GetServiceInfo-Anfrage, Dezimaltrennzeichen Punkt)	1.4
GetChapterList: Parameter 2 als optional deklariert	1.4