ONIX for Price Catalog
Format Overview
Version 1.2

Draft to support pilot exchanges, November 2015

EDItEUR invites comments on this specification and the associated XML schema. Please send comments or suggestions for improvement to info@editeur.org.

ONIX-PC Version 1.2, revision notes

Version 1.2 of ONIX-PC extends the format to make it possible to communicate details about open access products or product components, alongside the conventional paid products for which the standard was originally designed.

The addition of a new <UnpricedItemType> element allows the <CatalogPrice> composite to be suppressed if there is no applicable price, for example for a wholly-OA product.

A new section has also been added to communicate details of arrangements that may be in place with preservation agencies for particular journals or other product components.

Specifically:

- An optional <OpenAccessDetails> composite is included within the composite <OnlineComponentScope>.
- An <UnpricedItemType> element is now included within the <SubscriptionProductRecord> composite – either <UnpricedItem> or <CatalogPrice> must appear, but not both.
- A <PreservationAgency> composite has been added within the <ProductComponent> composite.

The positions of these new composites within the message hierarchy are described more fully later in this document.

A number of new codelists and codelist values have been included in Issue 8 of the ONIX Serials Codelists for use with various elements within the new composites.

Alongside these functional extensions, version 1.2 expresses more clearly the preferred "synonyms" for element names where a particular element has for historical reasons been associated with two descriptive names. In the following ten cases, the preferred element name

now appears first in this documentation. In each case, the original form, now deprecated, is shown second and italicized. This treatment applies to the following elements and synonyms:

Preferred and *deprecated* element synonyms	
preferred form	*original form, now deprecated*
<ProductComponent>	*<SerialVersion> -*
<ProductComponentIdentifer>	*<SerialVersionIdentifier>*
<ProductComponentIDType>	*<SerialVersionIDType>*
<ProductComponentName>	*<SerialVersionName>*
<Work>	*<SerialWork>*
<WorkIdentifier>	*<SerialWorkIdentifier>*
<WorkIDType>	*<SerialWorkIDType>*
<ProductComponentForm>	*<SerialVersionForm>*
<PhysicalComponentScope>	*<PhysicalVersionScope>*
<OnlineComponentScope>	*<OnlineVersionScope>*

EDItEUR is the international group that coordinates the development and promotion of standards for electronic commerce in the books and serials sectors.

1. Introduction

The ONIX for Price Catalog format (ONIX-PC) is used for transmitting a list of subscription products offered by a publisher, online publisher, content hosting service, or any other party that sells subscriptions, including price information. Recipients are most likely to be subscription agencies, fulfillment houses, and online publishers.

This message can convey price list information for subscription products consisting of a single component, for packages containing multiple components, or for combinations of print and online versions. It can transmit prices based on a variety of pricing models. The message can identify products that have been deleted from a publisher's price list because they have ceased publication or that have been transferred to another publisher. Products new to a publisher's list can be identified as transferred from another publisher (see Section 8).

Business cases might include the following:

* A publisher sends its complete price list to one or more subscription agents for loading into their price catalogs.
* A publisher sends *updates* to its price list as a broadcast message to one or more subscription agents.
* A fulfillment service sends its price list to a subscription agent who orders magazines through the fulfillment service.
* A publisher sends its price list to a fulfillment service for loading into its price catalog.

2. Related Documents

The formal definition of the ONIX-PC format is given in an XML schema and associated code lists, provided as a zipped file *here*. From version 1.1.1, the zipped package also includes the full XML schema for the ONIX Coverage Statement, rather than the subset of elements included in earlier versions. A package of human-readable *HTML documentation* is also available.

Permissible values for coded elements are found in *ONIX Serials Code Lists*.

The ONIX for Serials Coverage Statement is documented in *ONIX for Serials: Coverage Statement*.

Sample files illustrating a range of frequently encountered business cases are also available.

3. Structure of ONIX-PC: subscription products and product components

The basic structural unit in ONIX-PC is the *subscription product*; that is, a group of one or more serial versions, e-books and/or databases that are sold as a single subscription. For example, a subscription product might consist of any of the following components:

* A single serial version (e.g. the print version of a journal)
* A print + online combination (the print version plus the online version of a journal)
* A package consisting of print and/or online versions of multiple journals
* A hosted collection of a large number of online journals
* An e-book or database sold on a subscription basis (e.g., for an annual fee)

- A package or collection containing journals, e-books, online databases and/or print resources, sold on a subscription basis.

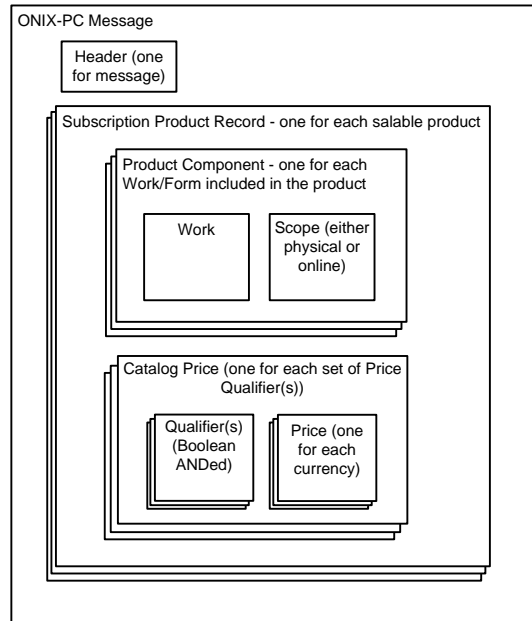The basic structure of ONIX-PC can be summarized as follows:

A **header**, including details of:
    The message sender
    The message recipient (if the message is not a broadcast message)
    Message number and date/time stamp for message tracking purposes
    Various defaults that are assumed unless explicitly overridden within the message

One or more **subscription products**, each of which contains:
    Product-level metadata
    Details of each **component** in the product (metadata and scope information)
    Price information for the whole subscription product

For those who prefer a more graphic representation, the following diagram of the data structure may be helpful:

4. About post-cancellation access

Some online resources are sold on a subscription basis, with access entitlements to the material maintained even after the subscription has expired or been cancelled. This is often referred to as "post-cancellation access". Post-cancellation access is indicated with a <PostCancellationAccess/> "empty element" found within <SubscriptionPeriodCoverage> and <PackageDetail>.

If post-cancellation access is granted for the material released during the subscription period, this empty element will appear within <SubscriptionPeriodCoverage>. If post-cancellation access is granted for all the material made accessible during the subscription period, this empty element will appear within <PackageDetail>.

If a subscription product is available for different prices depending on whether post-cancellation access is included, these are considered separate products. The <PostCancellationAccess/> empty element should be included where appropriate, and a free-text note should be included in <SubscriptionProductDescription> to distinguish the two products in a display to users.

5. About physical component scopes

Physical product components (e.g., print, CD-ROM) carry a <PhysicalComponentScope> composite, which may be repeated for different subscription periods, that describes the number of issues per year and the enumeration and chronology of the issues that are to be released in a particular subscription period. This is relevant only when the component is a serial version, as opposed to an e-book.

6. About online product components and online component scopes

Online product components (e.g. online journals available via a hosting service) carry an <OnlineComponentScope>, which includes

(a) if the component is an ongoing serial, the enumeration and chronology of the issues that are to be released in a particular subscription period, similar to the information found in <PhysicalComponentScope> for physical product components.

(b) additional information specific to the online content:
- The website(s) where the online content is accessible
- The complete online coverage included in a subscription, including years earlier than the subscription period.

Some subscription products consist only of backfiles; that is, content that was released during a specific period of time in the past. For these products, where no new content is released, <SubscriptionPeriodCoverage> is omitted from <OnlinePackage>. However, <Coverage> should be included in <PackageDetail>, to indicate what content is included in the subscription. <SubscriptionPeriodCoverage> is also omitted from <OnlinePackage>for e-books sold on a subscription basis.

7. About prices

The <CatalogPrice> composite lists a publisher's or agent's prices for a subscription product.

The <CatalogPrice> composite contains four child elements:
1. <PriceQualifier>
2. <PriceTier>
3. <TotalPrice>
4. A <PriceNote> for further free-text explanation.

7.1 Price qualifiers

Price qualifiers specify the criteria to which a price applies; for example,

- Type of Subscriber (such as person, institution, or member);
- Subscription Period;
- Countries or regions where the price applies.

If there are multiple qualifiers describing a price; for example, subscriber type and countries where applicable, then <PriceQualifier> is repeated. When there are different prices based on different sets of price qualifiers for a subscription product, then the entire <CatalogPrice> composite is repeated.

Within an instance of <CatalogPrice>, if multiple qualifiers are necessary to describe a specific price, these are Boolean ANDed (see examples below).

This also means that it is possible to have two Catalog Prices with the same price amount, if two sets of qualifiers describe the same price. For example, if the same price is valid for two separate subscriber types, this price will appear as two separate <CatalogPrice> composites, even if the actual price is the same, since a subscriber could not be both one subscriber type AND the other.

Each <PriceQualifier> consists of two mandatory child elements: <PriceQualifierType> and <PriceQualifierValue>. Some price qualifier types are associated with free-text or non-coded values (for example, 02, Moveable subscription term), and some are associated with coded lists of possible values found in code list 137: for example, price qualifier 04 (Subscriber type) must carry a price qualifier value found in code list 137A, which includes a comprehensive list of subscriber types, such as student, hospital, government agency or museum.

Senders who find that code list 137 does not adequately cover their price qualifier values should please contact EDItEUR so that the values may be added to the controlled list.

For example, consider a subscription product in a publisher's price catalog that has separate prices for the following:

1. Personal subscription anywhere
2. Non-profit institutional subscription in the US
3. Non-profit institutional subscription for all other countries

The <SubscriptionProduct> would carry three <CatalogPrice> composites, one for each of the above prices. The first would have one qualifier: Subscriber type (personal). The second price would have two qualifiers: "Subscriber type" (institutional, not-for-profit) and "Countries where applicable" (US). The third price would also have two qualifiers: "Subscriber type" (institutional, not-for-profit) and "Countries excluded" (US).

Price Regions. Some subscriptions are priced depending on the region where the product is purchased; e.g. North America, South America, Europe. Since such regions may be defined differently by different senders, and since no commonly accepted standard for regions exists, senders who wish to use regions rather than country lists must define their regions in the Header, in the RegionDefinition composite. Subsequently, these regions may be used with the price qualifier types "Region(s) where applicable" and "Region(s) excluded."

A subscription product may contain catalog prices qualified by price regions or country lists or neither of them, but both price regions and country lists should not appear in the prices for the same subscription product.

7.2 Tiered prices

From version 1.1, the message offers enhanced mechanisms for structured communication of tiered pricing arrangements, which are now widely encountered in the subscriptions market. In these arrangements, prices are determined by some measure of the "size" of the customer organization and thus, into which publisher price tier the organization falls.

An optional <PriceTier> composite has been introduced into the <CatalogPrice> composite. In turn, <PriceTier> contains the following child elements: <PriceTierUnit> and associated code values to describe the unit of measurement; <PriceTierFromValue> and <PriceTierToValue> to specify the lower and upper bounds of the tier, respectively.

The PriceTier composite only appears if there is a Price qualifier with Price qualifer type of 16 (Price tier, structured description), and a Price qualifer value of "Structured description." This is also explained in *ONIX Serials Code Lists,* list 136.

The Price qualifier type 07 (Price tier description) may be used when the price tier is described in free text. The accompanying Price qualifier value is a free text string.

7.3 Total Price and Price components

The <TotalPrice> composite contains the price in a specific currency, repeated if a price is given in multiple currencies. The <TotalPrice> composite contains the currency code and various price components. The optional <Rate> element within <PriceComponent> allows a price component to be expressed as a percentage as well as, or instead of, an amount.

For each Catalog Price, the <TotalPrice> composite is repeated once for each currency in which it may be paid. <TotalPrice> consists of several Price Components. Each is identified using a code from code list 112:

- 01 is the total price to be remitted by the recipient, with all discounts and extra charges applied. In the case of a publisher's price list sent to a subscription agent, this would be the "agent's price" after any discount has been taken and any other charges have been added.
- 02 is the base or "list" price as paid by the end customer. This is the price that the recipient, usually an agent, displays in its price catalog.
- 03, service charge, is not expected to be used in the ONIX-PC message, but it is included for those cases where the sender expects the recipient to pay a service charge.
- 04, shipping charge, is added to the base price, when the shipping charge is not included in the base price.
- 05, tax, is added to the base price when the recipient is expected to include tax to the sender.
- 06, discount to message recipient, is subtracted from the base price
- 07, handling charge, is added to the base price, when it is an extra charge to the end customer.

All price components are expressed either as positive numbers (in the <PriceAmount> element) or as a percent (in the <Rate> element). Rates are always applied to the base price.

There is a certain amount of redundancy built into this scheme, to allow senders to express prices in a manner that is convenient for them. Price Component 02 should always be included, as should any extra charges that are billed to the end customer. In the end the total price (01) will equal 02 + 03 + 04 + 05 + 07 – 06.

8. About notification types, cessations and transfers – best practices

Each <SubscriptionProduct> includes a Notification Type with one of the following values:
- New: the product is new to the sender's price list
- Delete: the product was previously in the sender's price list and is to be deleted from it
- Update: the product record is to replace a previously sent product record in the sender's price list
- Unspecified: it is strongly recommended that this notification type *not* be used. If it is used the recipient will assume that the product record is either New or Updated.

In the case of Deletes and Updates, the match point is the <SubscriptionProductIdentifier>, most likely the sender's product identifier (often known as the product code). Therefore any <SubscriptionProductIdentifier><IDValue> must be unique.

The Notification Type is followed by an optional coded Reason for Notification Type, which further clarifies the intent of the notification. Even further clarification can be expressed in the optional free-text NotificationTypeNote; for example, if the <ReasonForNotificationType> is "Product new to this publisher; transferred from another publisher," this note could be used to identify the previous publisher.

The table on the following page gives guidelines for handling a variety of situations (the coded value is given for each ReasonForNotificationType).

Situation	Notification Type	ReasonForNotificationType (serials codelist 183)	Comments
Product is an entirely new publication or combination	New	01 New product	
Product has been transferred *to* the party sending the message	New	02 Product new to this publisher; transferred from another publisher	Presumably the previous publisher will send a "Delete" notification.
Product is available in a new format (e.g. print + online now sold separately)	New	03 Product available in a new format	Each new format is sent as a separate "New" transaction. If the previous format(s) are discontinued, a delete transaction is sent for each previous product.
Product has ceased publication	Delete	04 Publication ceased	
Product has been transferred *from* the party sending the message	Delete	05 Product no longer available from this publisher, transferred to another publisher	Presumably the new publisher will send a "new" notification.
A combination product is no longer available.	Delete	06 Combination no longer available	In this case a new product record might also be sent, if the discontinued combination has been replaced by another.
Product is no longer available in its previous format (e.g. print + online is no longer available)	Delete	07 Product no longer available in this format	
Publication is merged into another publication (e.g. a journal is merged into another)	Delete	08 Publication merged into another	The original product is deleted, and the merged one sent as either a new or updated product record.
Combination is no longer available, and the components have been merged into another product.	Delete	09 Product no longer available, components merged into another product	The original product is deleted, and the merged combination sent as either a new or updated product record.
Publication is split into two or more new publications.	Delete	10 Product no longer available; publication split into others	The previous publication is deleted, and the new ones sent as new products.
Content of a combination product has changed; e.g. package has new or different components (2 options)	Update	11 Components of product have changed	An update transaction is sent for the product, showing the new components.
Product descriptive metadata have not changed; message simply informs of new price(s).	Update	12 No change to descriptive metadata, only price and likely PhysicalComponent Scope or OnlineComponentScope	An update transaction is sent for the product, showing new price(s).

9. About open access

From version 1.2 onward, ONIX-PC includes extended structures to describe open access features and policies at the journal level. These structures are grouped together in a new composite element entitled <OpenAccessDetails>. We distinguish three types of information: status updates, indicating for example that a particular journal is for transitioning from a paid-for model to OA; coverage-dependent details, where the OA features concerned apply only to a part of the journal's publication range; and general details, which are relevant for the whole journal, regardless of publication range.

For the purposes of ONIX-PC, we have endeavored to select clear but neutral terminology for a number of key features. Examples include the <OAModel> element to describe whether a title is freely available from the publisher's website ("gold") or from an alternative repository (which might be institution- or subject-based) chosen by the author ("green"). Similarly, we have used the element <OAExtent> to signal whether a particular journal is fully OA or a hybrid containing both OA and paid-for articles. This is still a developing area and we will continue to monitor evolving models and terminologies as appropriate.

Other features of OA publishing can also be communicated using ONIX-PC, including:
- Embargoes, where relevant
- Qualifiers on the availability of OA content
- Percentages of OA articles in some previous period
- License types available for a particular journal
- Registration details, if such are required for an otherwise-OA resource
- Article processing charges – whether or not these are levied and how they are priced for particular circumstances
- Links to (particularly) green repositories
- OA resource directories which cite the journal in question.

Finally, an <UnpricedItemType> has been added to the message structure to signal where necessary that a particular product has no associated <CatalogPrice>.

Most of the topics newly covered in version 1.2 have been requested by members of the ICEDIS Open Access working group, which has acted as a business-focussed forum within which these changes can be discussed and tested. If further requirements surface, particularly during pilot exchanges, these will be considered for inclusion in future versions of the standard and/or in the controlled vocabulary codelists that are associated with many of the elements.

Overview of the structure of ONIX-PC

The tables on the following pages give an overview of the ONIX-PC format, and show how elements are nested. For elements that contain coded values, please refer to *ONIX Serials Code Lists*.

For historical reasons several data elements have synonyms or alternate names. In each such case the preferred name now appears first, with the earlier (and now deprecated) equivalent underneath *in italics*. Use of the preferred names is recommended, but either is acceptable. They should not, however, be mixed.

Cardinality is indicated in the right-hand column for each element: 1 = Mandatory, not repeatable within its parent. 0-1 = Optional, not repeatable within its parent. 1-n = Mandatory, repeatable within its parent. 0-n = Optional, repeatable within its parent.

1	<ONIXPriceCatalog version="1.2" xmlns="http://www.editeur.org/onix/serials/SPS">			A list of serial products with prices	1

2	**<Header>**			Message header	1
3	<Sender>			The sender of the message (either identifier or name or both must be present)	1
4		<SenderIdentifier>		A coded identifier of the message sender, eg a SAN or GLN. Repeatable if multiple identifiers are sent.	0-n
5			<SenderIDType>	A code indicating the scheme from which the identifier is taken. See code list 44A for permissible values.	1
6			<IDTypeName>	The name of a proprietary scheme, if applicable	0-1
7			<IDValue>	The identifier value	1
8		<SenderName>		The name of the sender organisation	0-1
9		<SenderContact>		The name of a contact person in the sender organisation	0-1
10		<SenderEmail>		An email address for the sender	0-1
11	<Addressee>			The addressee of the message (omitted in "broadcast" messages). Repeatable for multiple addressees (either identifier or name or both must be present).	0-n
12		<AddresseeIdentifier>		A coded identifier of the message addressee. Repeatable if multiple identifiers are sent.	0-n
13			<AddresseeIDType>	A code indicating the scheme from which the identifier is taken. See code list 44A for permissible values.	1
14			<IDTypeName>	The name of a proprietary scheme, if applicable	0-1
15			<IDValue>	The identifier value	1
16		<AddresseeName>		The name of the addressee organisation	0-1
17		<AddresseeContact>		The name of a contact person in the addressee organisation	0-1
18		<AddresseeEmail>		An email address for the addressee	0-1
19	<MessageNumber>			Message sequence number	0-1

20	<MessageRepeat>			A number which distinguishes any repeat transmissions of a message	0-1
21	<SentDateTime>			The date and time, when a message was sent, in one of the following forms: YYYYMMDDTHHMMSSZ (universal time) YYYYMMDDTHHMMSS±HHMM (time zone) where "T" and "Z" are constants; that is, the letters T and Z. The recommended form is YYYYMMDDTHHMMZ, using universal time (UTC/GMT).	1
22	<MessageNote>			A free-text note about the contents of the message.	0-1
23	<DefaultCurrency Code>			The currency of prices listed in the message, unless otherwise specified. (use ISO 4217 currency codes, see code list 96 for permissible values)	0-1
24	<DefaultPublisher>			The publisher whose subscription products are listed in the message (when the message applies wholly or mainly to a single publisher). Either <PublisherIdentifier> or <PublisherName> must be present.	0-1
25		<PublishingRole>		A code indicating a role played in the publishing process, in this case, always carries value "01" (Publisher) (see code list 45B)	1
26		<PublisherIdentifier>		A coded identifier of the publisher. Repeatable for multiple identifiers.	0-n
27			<PublisherIDType>	A code indicating the scheme from which the identifier is taken. See code list 44D for permissible values.	1
28			<IDTypeName>	The name of a proprietary scheme, if applicable	0-1
29			<IDValue>	The identifier value	1
30		<PublisherName>		The name of the publisher	0-1
31	<Subscription PeriodDefinition>			A period for which prices and/or coverage details are given in the message: repeatable if a message includes prices or coverage for multiple periods. All subscription periods used in the message must be defined in the <SubscriptionPeriodDefinition>. Must contain <SubscriptionPeriodLabel>. Must also contain *either* <StartDate> and <EndDate> *or* <AnytimeStart/>, but not both.	1-n
32		<SubscriptionPeriod Label>		An arbitrary label assigned to a subscription period by the sender and used to refer to the subscription period in the body of the message	1
33		<StartDate>		The date on which the subscription period starts: YYYYMMDD	0-1
34		<EndDate>		The date on which the subscription period ends: YYYYMMDD	0-1
35		<AnytimeStart/>		An "empty element" indicating a subscription period with no predefined start or end date. If present, neither <StartDate> nor <EndDate> is present.	0-1
36	<Region Definition>			A region defined by the sender. Repeatable if a message includes multiple regions. Must include both <RegionLabel> and <CountriesInRegion>.	0-n
37		<RegionLabel>		An arbitrary label assigned to a region by the sender and used to refer to regions in the body of the message. <RegionLabel> may not contain spaces	1
38		<CountriesInRegion>		A list of countries that belong to the region. Use 2-letter ISO 3166-1 codes (see code list 91) separated by a space.	1

39	<CompleteFile/>			One or other of these empty elements is mandatory, to indicate whether the	0-1
40	<DeltaFile/>			message is a complete replacement file or an incremental update	0-1

1	<SubscriptionProduct Record>		**Details of a priced or unpriced product, which may be a single version of a single work, or a combination of multiple works and/or versions: repeatable.** Either <SubscriptionProductIdentifier> or <SubscriptionProductName> must be present	1-n
2	<NotificationType>		A code indicating whether the subscription product record is new or updated or a deletion. See code list 1A for permissible values (default 00).	1
3	<ReasonFor NotificationType>		Further clarification of the Notification Type. See Introduction, section 8 and code list 183S for permissible values.	0-1
4	<NotificationType Note>		A free text note further explaining the reason for the notification type. For example, if the <ReasonForNotificationType> is "Product new to this publisher; transferred from another publisher," this note could be used to identify the previous publisher.	0-1
5	<Subscription ProductIdentifier>		A coded identifier of a subscription product, eg a publisher's product code. Repeatable if multiple codes are sent. These identifiers are used as match points, when a record is updated or deleted; therefore <IDValue> must be unique.	0-n
6		<SubscriptionProduct IDType>	A code indicating the scheme from which the identifier is taken. See code list 4S for permissible values.	1
7		<IDTypeName>	The name of a proprietary scheme, if applicable	0-1
8		<IDValue>	The identifier value	1
9	<SubscriptionProduct Name>		The name given by the sender to a serial product. This should be a human readable name, such as "Journal of ABC, print," since it will be loaded up into a seller's price catalog, to be read by end customers. Coded names such as "JOABC /P" should be included as Subscription Product Identifiers, with <SubscriptionProductIDType> 01 (Proprietary).	0-1
10	<Subscription ProductDescription>		Additional free text description of a serial product	0-1
11	<ContentHosting System>		Content hosting system through which all online components in this product are available. Used only when all online product components are available through a single content hosting system. If present, there will be no <ContentHostingSystem> elements in the product components in this <SubscriptionProductRecord>.	0-1
12	<OnlinePublisher>		Organization that operates the content hosting system through which all online components in this product are available. Used only when all online product components are available through a single content hosting system. If used, there will be no <OnlinePublisher> composites in the product components in this <SubscriptionProductRecord>. Used only for products that contain one or more online components.	0-1
13		<PublishingRole>	A code indicating the role of the publisher. See code list 45A for permissible values. In this case the PublishingRole will always be "05," (Host/distributor of electronic content)	1

14		<PublisherIdentifier>		A coded identifier of the publisher	0-n
15			<PublisherIDType>	A code indicating the scheme from which the identifier is taken. See code list 44D for permissible values.	1
16			<IDTypeName>	The name of a proprietary scheme, if applicable	0-1
17			<IDValue>	The identifier value	1
18		<PublisherName>		The name of the publisher.	0-1

30			<Language>	Language of full-text of the work. Repeatable for multi-lingual publications. Permissible values: ISO 639-2/B (see code list 74)	0-n
31			<CountryOfPublication>	Country where the editorial offices reside, and where editorial work is done. See code list 91 (ISO 3166-1 2-letter codes) for permissible values.	0-1
32			<CountryOfDispatch>	Country from which physical items are shipped. Repeatable if dispatched from multiple countries. See code list 91 (ISO 3166-1 2-letter codes) for permissible values.	0-n
33			<Website>	Composite: details of a website for the serial work as a whole. The expected value of <WebsiteRole> in this context is "04" (Journal home page) (see code list 73C for permissible values of <WebsiteRole>)	0-n
34		<ProductComponent Form> *<SerialVersionForm>*		A code indicating the form in which the product component is published. See code list 7A for permissible values.	1
35		<PreservationAgency>		Organization(s) committed to preserving content of a product component. Repeatable if content of this component is preserved by multiple preservation agencies. Must contain either <PreservationAgencyIdentifier> or <PreservationAgencyName> or both.	0-n
36			<PreservationAgencyIdentifier>	Composite: a coded identifier of a preservation agency. Contains <PreservationAgencyIDType>, <IDTypeName>, and <IDValue>. Repeatable if multiple identifiers are sent. See code list 44A for permissible values for <PreservationAgencyIDType>.	0-n
37			<PreservationAgencyName>	The name of a preservation agency, for example: • British Library • Portico • CLOCKSS Archive • HathiTrust	0-1
38			<PreservationAgencyLink>	URL of website where preserved content may be accessed. Repeatable for mirror sites.	0-n
39		<PhysicalComponent Scope> *<PhysicalVersion Scope>*		Composite: Details of the range of issues included in a subscription to a printed or other physical component (eg CD-ROM) – see expansion in green later in this document. Repeatable for multiple subscription periods. Either <PhysicalComponentScope> or <OnlineComponentScope>, but not both, may appear in a <ProductComponent>	0-n
40		<OnlineComponent Scope> *<OnlineVersionScope>*		Composite: Details of online content included in a subscription – see expansion in green later in this document. Repeatable for multiple subscription periods.	0-n

41	<UnpricedItemType>			If a message is sent with no price at all, this element carries the reason there is no price; for example, product is fully open access, price to be announced. Either <UnpricedItemType> or <CatalogPrice> must occur, but not both. See ONIX for Books code list 57 for permissible values.	0-1
42	<CatalogPrice>			Composite: publisher's or agent's publicly listed prices for the subscription product – see expansion in yellow at the end of this document.	0-n

Expansion of <PhysicalComponentScope>

1	**<PhysicalComponentScope>** *<PhysicalVersionScope>*			**Details of the range of issues included in a subscription to a printed or other physical product component (eg CD-ROM) for a specified subscription period. Repeatable for multiple subscription periods.** Must include <SubscriptionPeriodLabel> and one or more of <IssuesPerYear> or <Coverage>, or <CoverageNote>. If neither <IssuesPerYear> nor <Coverage> nor <CoverageNote> are sent, then <PhysicalComponentScope> is omitted.	0-n
2	<SubscriptionPeriodLabel>			The subscription period to which a coverage statement applies, identified by the label assigned in the header	1
3	<IssuesPerYear>			The number of issues expected to be published in a year. Only include those that are part of the regular enumeration.	0-1
4	<Coverage>			Composite: the range of issues included in the subscription period, using the ONIX for Serials Coverage statement, documented separately *here*. If the subscription period is defined as an "anytime start", the <Coverage> composite is omitted.	0-1
5	<CoverageNote>			A free text note explaining coverage.	0-1

Expansion of <OnlineComponentScope>

1	**<OnlineComponentScope>** ***<OnlineVersionScope>***			**Details of the online content included in a subscription to an online product component for a specified subscription period.** Repeatable for multiple subscription periods. Must include <SubscriptionPeriodLabel> and <OnlinePackage>	0-n
2	<SubscriptionPeriod Label>			The subscription period to which the <OnlineComponentScope> statement applies, identified by a label assigned in the header. This should be included even for fully Open Access components, to indicate which year the open access information applies to.	1
3	<IssuesPerYear>			The number of issues expected to be published each year	0-1
4	<OnlinePackage>			Details of online content for the specified product component from a specified content hosting system. Repeatable if the product component is available through multiple content hosting systems.	1-n
5		<ContentHosting System>		The name of the content hosting system. Will not appear if <ContentHostingSystem> has been specified at the product level	0-1
6		<OnlinePublisher >		The organization that operates the content hosting system. Will not appear if <OnlinePublisher> has been specified at the product level.	0-1
7			<PublishingRole>	A code indicating the role of the publisher. See code list 45A for permissible values.	1
8			<PublisherIdentifier>	Composite: a coded identifier of an online publisher. See code list 44D for permissible values for <PublisherIDType>.	0-n
9			<PublisherName>	The name of the online publisher.	0-1
10		<Website>		A website through which the online content of a product component is accessed	0-n
11			<WebsiteRole>	A code indicating the role of the website: the expected value in this context is "05" (Online journal "available contents" page) (see code list 73D for permissible values)	1
12			<WebsiteDescription>	Free text describing the website.	0-1
13			<WebsiteLink>	The URL for the website	1
14			<MirrorSite>	Composite: Details of a mirror site, if any: repeatable if there are several. Includes <WebsiteDescription> and <WebsiteLink>	0-n
15		<SubscriptionPeriod Coverage>		Details of the online content that is expected to be released during the subscription period in a product component offered on the specified content hosting system. Inclusion of this information is strongly recommended if available. Either <Coverage> or <CoverageNote> or both, must be present. For "anytime starts," <SubscriptionPeriodCoverage> is omitted. For backfiles and other non-serial resources, where no new content is released, <SubscriptionPeriodCoverage> is omitted.	0-1

16			<PostCancellationAccess/>	Flag indicating that access rights to the material released during the subscription period are maintained even after the subscription has been cancelled.	0-1
17			<Coverage>	Composite: the range of issues included in the subscription period, using the ONIX for Serials Coverage statement, documented separately *here.*	0-1
18			<CoverageNote>	A free text note explaining the subscription period coverage.	0-1
19		<PackageDetail>		Details of the online coverage accessible with a subscription to a product component through the specified content hosting system. Inclusion of this information is strongly recommended if available. In the case of backfile subscriptions, the <Coverage> statement should indicate what content is included in the subscription. If a component is fully open access, then <PackageDetail> is omitted. Coverage information is found in <OpenAccessDetails>	0-1
20			<PostCancellationAccess/>	Flag indicating that access rights to the online coverage described in the <Coverage> statement within <PackageDetail> are maintained even after the subscription has been cancelled.	0-1
21			<Coverage>	An ONIX for Serials coverage statement, documented separately *here.*	0-1
22			<CoverageNote>	A free text note explaining coverage.	0-1
23	<OpenAccessDetails>			Details of Open Access offerings for an online component. See expansion in light purple below.	0-1

Expansion of <OpenAccessDetails>

1	<OpenAccessDetails>			Details of Open Access offerings for a component. Used only with online components.	0-1
2	<OAStatusUpdate>			A composite that signals a change in the OA status of a component, and the point at which the status change takes place. Repeatable if multiple status changes are announced in a message, or to show a history of changes.	0-n
3		<OAStatusUpdateType>		Type of update being announced. For example: • Paid to OA (component or product previously requiring payment becomes OA) • Coverage of OA extended backwards • Coverage of OA extended forwards (embargo shortened or removed) • Green to gold (OA material previously available in separate repository now available from publisher's website) • Article processing charges now apply • Article processing charges no longer apply (see code list 233 for permissible values):	1
4		<DateEffective>		Date the status change took place or will take place. YYYYMMDD	1
5	<OADetailsByCoverage>			OA features specific to a particular date and/or enumeration range. Repeatable for different coverage ranges or models or qualifiers. For example, Vols 11-20 (2001-2010) are hybrid gold open access. Vols 21 onward (2011-present) are fully gold open access.	0-n
6		<OACoverage>		Range of dates and/or enumeration to which a set of OA features applies. Either <FixedCoverage> or <MovingCoverage> or <EbookEmbargo> must appear, but not more than one. <OACoverage> may be omitted if this set of OA features applies to the entire component.	0-1
7			<FixedCoverage>	Dates and/or enumeration, if start and end point are fixed. This composite is taken from the Coverage Statement, which is documented separately here.	0-1
8			<MovingCoverage>	Dates and/or enumeration, if end point is moving. (Start point may be fixed or moving). This composite is taken from the Coverage Statement, which is documented separately here.	0-1
9			<EbookEmbargo>	Dates or period of time after which OpenAccess to an e-book component is available. Must contain either <SpecifiedDate> or <SpecifiedPeriod>, but not both.	0-1
10			<SpecifiedDate>	Open access becomes available on this date (YYYYMMDD).	0-1
11			<SpecifiedPeriod>	Open access becomes available after this period of time after e-book publication date.	0-1
12			<CountUnit>	Units may be days, weeks, months, years (see code list 108 for permissible values).	1
13			<Count>	Number of units to count.	1

14		<OAModel>		OA model for content in the date and/or enumeration range. For example: • gold (freely available on product website) • green (freely available in separate repository) See code list 231 for permissible values.	1
15		<OAExtent>		Extent of open access for releases in the date and/or enumeration range. For example: • full (entire component is Open Access) • hybrid (not all articles or book portions are Open Access) See code list 232 for permissible values.	1
16		<OAQualifier>		Criteria that qualify a "subscriber" to get Open Access to a product or component for the Coverage, Model, and Extent specified. Used only when Open Access is available in certain regions and/or to certain subscriber types. If multiple qualifiers apply (e.g.not-for-profit organizations in US), then <OAQualifier> is repeated. Multiple qualifiers are ANDed when applied to a particular <OADetailsByCoverage>. If different qualifiers apply independently (e.g. not-for-profit organizations anywhere, and also certain developing countries) then <OADetailsByCoverage> must be repeated.	0-n
17			<OAQualifierType>	The following codes from code list 136 are valid: • 04 Subscriber type • 05 Countries where applicable • 15 Regions where applicable	1
18			<OAQualifierValue>	Subscriber type has a list of permissible values found in code list 137A. Countries where applicable is a space-separated list of ISO codes, found in code list 91. Regions where applicable is a space-separated list or regions that have been defined in the message header.	1
19	<OADetailsGeneral>			OA features applicable to an entire component at the time the message is sent, not specific to a date or enumeration range	0-1
20		<PercentOfOAArticles>		Percent of articles released during a specified time period that were OA. Used only in the case of hybrid OA. Repeatable if multiple time periods are reported. Not applicable for E-books or databases.	0-n
21			<TimePeriod>	YYYYMMDD-YYYYMMDD	1
22			<OAPercent>	Percent of articles that were OA during the time period.	1
23		<OARepository>		Name and location of repository where author-archived OA content is found. Repeatable if content is found in multiple repositories. Used only if <OAModel> = "green". Must contain either <RepositoryName> or <RepositoryLink> or both.	0-n
24			<RepositoryName>	Name of repository	0-1
25			<RepositoryLink>	URL of repository. Repeatable if there are multiple locations, such as mirror sites.	0-1
26		<OAResourceDirectory>		Name and location of directory where a product or component is listed as OA. Repeatable if listed in multiple directories. Must contain either <ResourceDirectoryName> or <ResourceDirectoryLink> or both.	0-n
27			<ResourceDirectoryName>	Name of resource directory; for example, DOAJ, DOAB, ROAD.	0-1

28			<ResourceDirectoryLink	URL of directory. Repeatable if there are multiple locations, such as mirror sites.	0-1
29		<RegistrationRequired/>		Empty element indicating that users must register in order to obtain free access.	0-1
30		<RegistrationLink>		URL of a website through which user can register for Open Access.	0-1
31		<EpubLicense>		Name and location of default license governing the use of OA content in a product or component. Repeatable if multiple license types are offered.	0-n
32			<EpubLicenseType>	Coded value of the EpubLicense. See code list 236 for permissible values.	0-1
33			<EpubLicenseName>	Name of license. For example, • CC0 (Creative Commons "No Rights Reserved") • CC BY 4.0 (Creative Commons Attribution v4.0) • CC BY-NC 4.0 (Creative Commons Attribution-NonCommercial v4.0) • CC BY-NC-ND 4.0 (Creative Commons Attribution-NonCommercial-NoDerivs v4.0) • CC BY-NC-SA 4.0 (Creative Commons Attribution-NonCommercial-ShareAlike v4.0)	0-1
34			<EpubLicenseExpression>	Details of the license expression	0-n
35			<EpubLicenseExpressionType>	Format in which the license is expressed. See code list 218 for permissible values: • Human readable • Professional readable (for legal specialists) • ONIX-PL	1
36			<EpubLicenseExpressionLink>	URL for the license expression. For example: http://creativecommons.org/licenses/by/4.0/legalcode	1
37		<APCDetails>		Details of article processing charges, if any, assessed for OA articles or book portions published in a product or component. Repeatable if multiple APC Qualifier sets apply	0-n
38			<APCQualifier>	Repeatable if multiple qualifiers apply (e.g. main articles with government funding). Qualifiers are AND'ed. If same APC applies to all article types, then APC Qualifier is omitted.	0-n
39			<APCQualifierType>	For example: • Article type • License type • Funding organization type • Page count • Word count See code list 234 for permissible values.	1

40			\<APCQualifierValue\>	Linked to APC QualifierType - coded values found in several code lists. Article type examples: • Main article • Editorial • Review article • Book portion • Etc Funding organization type examples: • Government agency • Non-profit agency See code list 235 for permissible values to be used with Article Type. See code list 236 for permissible values to be used with License Type. See code list 237 for permissible values to be used with Funding Organization Type. Page count and Word count values are integers.	1
41			\<APCAmount\>	Repeatable for different currencies, if payable in multiple currencies.	1-n
42			\<CurrencyCode\>	The currency in which APC amount is expressed, if different from the default specified in the header (ISO 4217 currency codes, see code list 96). If APC can be submitted in multiple currencies, \<APCAmount\> is repeated.	0-1
43			\<Amount\>	Amount charged, always sent as a positive number.	1

Expansion of <CatalogPrice>

1	<CatalogPrice>			Price for a single subscription to a subscription product. In the case of a product containing physical components, this implies a quantity of one for those components. Repeatable for different sets of qualifiers. See section 7.2 of the introductory text for further explanation. <CatalogPrice> must include <TotalPrice>. If <PriceTier> is present, then <PriceQualifier> must also be present. <CatalogPrice> is omitted if the product is unpriced (see <UnpricedItemType>)	0-n
2	<PriceQualifier>			A criterion that applies to a price. If multiple qualifiers apply to a price, then <PriceQualifier> is repeated. Note that multiple qualifiers are ANDed when applied to a specific catalog price (see Section 6 of the Introduction). If there is only one price, and it is not qualified in any way, then <PriceQualifier> is omitted.	0-n
3		<PriceQualifierType>		A code specifying the type of qualifier. See code list 136 for permissible values.	1
4		<PriceQualifierValue>		A coded value for the specified price qualifier type. Each price qualifier type has its own list of permissible values; some, such as price tier, allow proprietary values. Permissible values: see code list 137.	1
5	<PriceTier>			A price criterion expressed as a range of values. May only appear in combination with a <PriceQualifier> that has <PriceQualifierType> 16 and <PriceQualifierValue> "Structured description".	0-1
6		< PriceTierUnit>		The unit of measurement used to define a tier; e.g. total FTE, number of beds, number of simultaneous users. See code list 185S for permissible values.	1
7		< PriceTierFromValue>		The lower bound of the tier; usually but not necessarily an integer.	1
8		< PriceTierToValue>		The upper bound of the tier; usually but not necessarily an integer.	1
9	<TotalPrice>			A price that may be defined by one or more price qualifiers: repeatable if price is expressed in multiple currencies.	1-n
10		<CurrencyCode>		The currency of the price, if different from the default specified in the header (ISO 4217 currency codes, see code list 96S). All components of a price are expressed in the same currency. If price is to be given in multiple currencies, <TotalPrice> is repeated.	0-1
11		<PriceComponent>		A component of the total price, repeatable if there are multiple components	1-n
12			<PriceComponentType>	A code specifying the type of the price component. See code list 112S for permissible values.	1
13			<PriceComponent Description>	A free text description qualifying the price component.	0-1
14			<ShippingMethod>	A code specifying the shipping method if the price component includes shipping. See code list 114S for permissible values.	0-1

15			<PriceAmount>	The amount of the price component. Always sent as a positive number or zero. The PriceComponentType code indicates whether the amount is a charge or a discount. Either <PriceAmount> or <Rate> must be present within <PriceComponent>.	0-1
16			<Rate>	A percent of the base price. Used only for price components that may be expressed as a rate. Percent sign is not included.	0-1
17	<PriceNote>			A free text note clarifying the price.	0-1