

6. INTERACTIVE DATA

The following instructions provide guidance on the preparation, submission, and validation of EDGAR-acceptable electronic filings with attached Interactive Data documents in eXtensible Business Reporting Language (XBRL) format.

SEC XBRL Rules

In 2004, the Securities and Exchange Commission (SEC) began assessing the benefits of interactive data and its potential for improving the timeliness and accuracy of financial disclosure and analysis of Commission filings. As part of this evaluation, the Commission adopted rules in 2005 permitting filers, on a voluntary basis, to provide financial disclosure in interactive data format as an exhibit to certain filings to the EDGAR system.³ In 2007, the voluntary program was extended to enable mutual funds to submit risk/return summary information.⁴ In 2008, the Commission voted to adopt rules requiring issuers and mutual funds to provide financial disclosure and risk/return summary information in interactive data format, respectively, as described in more detail in the adopting releases.

Tagging Instructions

The rules the Commission voted to adopt in 2008 specify that filers are required to tag their financial statements and mutual fund risk/return summary information according to the tagging instructions presented in this chapter of the EDGAR Filer Manual. These tagging instructions require working knowledge of XML and XBRL (as the instructions directly reference elements and attributes from the XML 1.0 and XBRL 2.1 specifications). This approach, though admittedly technical, is intended to provide information that is independent of the various commercially available software applications that filers may use to create their XBRL documents. It is also intended to provide detailed and unambiguous instructions that enable an XBRL document to successfully pass through EDGAR validation and rendering for the public display of a human-readable document. Registrants filing under the voluntary program are encouraged to adhere to the tagging instructions described in this chapter. However, voluntary participants are only required to use the instructions that are described in Section 5.2.4, “Unofficial XBRL.”

Rendering

Tagging instructions as detailed in this document have a direct impact on the Commission’s ability to generate human-readable documents (from raw XBRL data) to compare against the official HTML/ASCII versions of the same documents.

³ XBRL Voluntary Financial Reporting Program on the EDGAR System (Release No. 33-8230).

⁴ Extension of Interactive Data Voluntary Reporting Program on the EDGAR System to Include Mutual Fund Risk/Return Summary Information (Release No. 33-8781).

The Commission processes raw XBRL data to produce and display human-readable documents, a process known as rendering that is detailed below in section 6.24. EDGAR disseminates both raw and rendered XBRL documents.

6.1 XBRL Overview

XBRL is an XML-based language that is used for the exchange and analysis of business and financial information. An XBRL document consists of the following:

- One or more instance documents (instances) that contain actual data and facts,
- One or more schema documents (schemas) that declare the elements that can be used in the instance, and identify other schemas and files where relationships among those elements are declared,
- One or more linkbase documents (linkbases) containing additional information about, or relationships among, the elements in a schema document. There are five types of linkbases: Label, Definition, Reference, Presentation, and Calculation.

Note: Although the Reference Linkbase file is a valid attachment type, at the moment it is not used.

Schema and linkbase documents contain references to each other in the form of Uniform Resource Identifiers (URIs).

Taxonomies are sets of schemas and linkbases that are designed to be loaded and used together; for example, a schema may contain a list of linkbases that have the URIs of other schemas to be loaded, and so on. Taxonomies generally fall into one of two categories: (1) standard base taxonomies and (2) company extension taxonomies. Filers use company extension taxonomies to supplement base taxonomies and, within limits, customize those base taxonomies to their reporting goals.

Instances also use URIs to reference schemas and linkbases. The Discoverable Taxonomy Set (DTS) of an instance document is the set of all schemas and linkbases that are found by following all URI links and references.

6.2 Supported Versions of XBRL Standard Taxonomies

Below is a list of XBRL schemas for the core document files that are supported by EDGAR (e.g. instance, linkbase, XLink documents). The namespace that represents each document must be used in the form as shown. A recommended prefix that represents each namespace is provided. The location of the actual schema file is also identified

1. Taxonomy Schema for XBRL
 - a. Namespace name: <http://www.xbrl.org/2003/instance>
 - b. Recommended prefix: **xbrli**
 - c. Location of file: <http://www.xbrl.org/2003/xbrl-instance-2003-12-31.xsd>
2. Schema for XML instance
 - a. Namespace name: <http://www.w3.org/2001/XMLSchema-instance>
 - b. Recommended prefix: **xsi**
 - c. Location of file: <http://www.w3.org/2001/XMLSchema-instance.xsd>
3. Taxonomy Schema for XML
 - a. Namespace name: <http://www.w3.org/2001/XMLSchema>
 - b. Recommended prefix: **xsd**⁵
 - c. Location of file: <http://www.w3.org/2001/XMLSchema.xsd>
4. XBRL linkbase schema constructs
 - a. Namespace name: <http://www.xbrl.org/2003/linkbase>
 - b. Recommended prefix: **link**
 - c. Location of file: <http://www.xbrl.org/2003/xbrl-linkbase-2003-12-31.xsd>
5. XBRL simple and extended link constructs
 - a. Namespace name: <http://www.xbrl.org/2003/XLink>
 - b. Recommended prefix: **xl**
 - c. Location of file: <http://www.xbrl.org/2003/xl-2003-12-31.xsd>

⁵ Some taxonomies use the 'xs' prefix instead of 'xsd:'

6. XLink attribute specification
 - a. Namespace name: <http://www.w3.org/1999/xlink>
 - b. Recommended prefix: **xlink**
 - c. Location of file: <http://www.xbrl.org/2003/xlink-2003-12-31.xsd>
7. Reference Parts schema (FRTA 1.0)
 - a. Namespace name: <http://www.xbrl.org/2004/ref>
 - b. Recommended prefix: **ref**
 - c. Location of file: <http://www.xbrl.org/2004/ref-2004-08-10.xsd>
8. Reference Parts schema (FRTA 1.0 revised)
 - a. Namespace name: <http://www.xbrl.org/2006/ref>
 - b. Recommended prefix: **ref**
 - c. Location of file: <http://www.xbrl.org/2006/ref-2006-02-27.xsd>
9. Dimensions taxonomy specification
 - a. Namespace name: <http://xbrl.org/2005/xbrldt>
 - b. Recommended prefix: **xbrldt**
 - c. Location of file: <http://www.xbrl.org/2005/xbrldt-2005.xsd>
10. Dimensions instance specification
 - a. Namespace name: <http://xbrl.org/2006/xbrldi>
 - b. Recommended prefix: **xbrldi**
 - c. Location of file: <http://www.xbrl.org/2006/xbrldi-2006.xsd>

In addition to the core XBRL schema files in the table above, EDGAR supports additional schema and linkbase files. The SEC “Information for EDGAR Filers” web site provides an up-to-date listing of these “recognized” files. Different types of filings at different times will be permitted (or required) to reference certain published schemas and linkbases. Filings must always refer to recognized files at the specified URI locations and use recommended namespace prefixes.

6.3 Submission Syntax

This section details rules of syntax that apply to an Interactive Data submission and all of its attachments.

6.3.1 Use EDGARLink Online Submission Templates to attach XBRL Documents.

Filers may submit submissions with attached XBRL documents to the EDGAR Filing Website. The EDGAR Online Forms Website does not support the attachment of XBRL documents.

EDGARLink Online attaches an XBRL document to a filing in a similar manner as they attach an ASCII/SGML or HTML document. See 5.2.2.10 and 5.2.3.1 for restrictions on the number, size, lengths of document names and other properties of attached documents.

6.3.2 XBRL attachments to an EDGAR submission.

XBRL instance, schema, and linkbase documents must be attached to an EDGAR submission using the EX-101.*, EX-99.SDR K.*, or EX-99.SDR L.* document types.

To minimize the need for preparing multiple data files, filers subject to 17 CFR 232.406 should satisfy its requirement for providing a Cover Page Interactive Data File (Exhibit 104) using an Inline XBRL Document Set (see 5.2.5 above) with EX-101.* attachments other than EX-101.INS.

A single submission must not contain more than one of EX-101 or EX-99.SDR K attachments. A submission may contain both EX-99.SDR K and EX-99.SDR L attachments.

All XBRL document types are listed in the table below.

Note: Although the Reference Linkbase file is a valid attachment type, at the moment it is not used.

XBRL Document	Interactive Data Document Types	Root Element	Required Element
Instance	EX-101.INS, EX-99.SDR K.INS, EX-99.SDR L.INS	xbrli:xbrl	
Schema	EX-101.SCH, EX-99.SDR K.SCH, EX-99.SDR L.SCH	xsd:schema	
Calculation Linkbase	EX-101.CAL, EX-99.SDR K.CAL, EX-99.SDR L.CAL	link:linkbase	link:calculationLink

XBRL Document	Interactive Data Document Types	Root Element	Required Element
Definition Linkbase	EX-101.DEF, EX-99.SDR K.DEF, EX-99.SDR L.DEF	link:linkbase	link:definitionLink
Label Linkbase	EX-101.LAB, EX-99.SDR K.LAB, EX-99.SDR L.LAB	link:linkbase	link:labelLink
Presentation Linkbase	EX-101.PRE, EX-99.SDR K.PRE, EX-99.SDR L.PRE	link:linkbase	link:presentationLink
Reference Linkbase	EX-101.REF	link:linkbase	link:referenceLink

6.3.3 XBRL document names.

Document names must match {base}-{date}[_{suffix}].{extension}.

XBRL Document	Documentname Format
Instance	{base}-{date}.xml
Schema	{base}-{date}.xsd
Calculation Linkbase	{base}-{date}_cal.xml
Definition Linkbase	{base}-{date}_def.xml
Label Linkbase	{base}-{date}_lab.xml
Presentation Linkbase	{base}-{date}_pre.xml
Reference Linkbase	{base}-{date}_ref.xml

Note: Although the Reference Linkbase file is a valid attachment type, at the moment it is not used.

The {base} on the instance filename should, but need not, begin with the registrant's ticker symbol or other mnemonic abbreviation.

The {date} must denote the ending date of the period. If the instance document is a prospectus or other report whose period is indefinite, {date} should match the filing date.

For example, a Form 10-Q covering the period ending September 30, 2009 has {date} = 20090930; and a 485BPOS filed on January 23, 2010 has {date} = 20100123.

The {base} and {date} on other files in the submission should, but need not, be the same as that used for the instance in the same submission.

6.3.4 An XBRL document must not contain HTML character name references.

Section 5.2.2.6 defines extended ASCII characters and how they may be referenced in HTML, but XBRL documents are not HTML documents, and recognize only XML predefined entities (" & ' < and >).

To include characters greater than 127, use XML numeric character references.

For example, ® and ® are ASCII sequences producing the ® symbol.

6.3.5 The ampersand character

The ampersand character must begin a valid XML predefined entity or numeric character reference. XML with invalid predefined entity or numeric character references are treated as if an invalid character had appeared. For example, &&; is invalid XML.

6.3.6 Restrictions on URI content

The URI content of the xlink:href attribute, the xsi:schemaLocation attribute and the schemaLocation attribute must be relative and contain no forward slashes, or a recognized external location of a standard taxonomy schema file, or a “#” followed by a shorthand xpointer. The xlink:href attribute must appear on the link:loc element; the schemaLocation attribute must appear on the xsd:import element, and the xsi:schemaLocation attribute must appear on the link:linkbase element and may appear on the xbrli:xbrl element.

Valid entries for the xlink:href attribute, the xsi:schemaLocation attribute and the schemaLocation attribute are document locations. If they are relative URIs, they are subject to EDGAR attachment naming conventions. Therefore, all documents attached to the submission will be at the same level of folder hierarchy. By restricting the content of these attributes, all documents in the DTS of an instance will be either a standard taxonomy or present in the submission.

Examples:

- xlink:href="abccorp-20100123.xsd"
- <xsd:import schemaLocation="http://xbrl.org/2006/xbrldi-2006.xsd" namespace= .../>

Counterexamples:

- xlink:href="http://www.example.com/2007/example.xsd"
- xlink:href="#element(1/4)" (Comment: this is a scheme-based xpointer, not shorthand)

6.3.7 XBRL document names are case sensitive.

XBRL validation requires instances, linkbases and schemas to refer to one another using URIs, which are case sensitive.

6.3.8 A submission must contain exactly one EX-101.INS or EX-99.SDR K.INS.

An Interactive Data instance in XBRL format must be an EX-101.INS, EX-99.SDR K.INS, or EX-99.SDR L.INS.

6.3.9 Filers must use one of the standard taxonomies.

The set of standard taxonomy files changes from time to time. A current listing of all taxonomy files can be found on the SEC website at <https://www.sec.gov/info/edgar/edgartaxonomies.shtml>.

6.3.10 A submission must contain at least one company extension schema.

A company extension taxonomy might be as simple as single schema (exhibit type EX-101.SCH, EX-99.SDR K.SCH or EX-99.SDR L.SCH) that defines the namespace to be used by the company, and contains xsd:import elements for the relevant base taxonomy schemas. In most cases, though, the company extension taxonomy will consist of a schema and several linkbases.

6.3.11 Attribute xml:base must not appear in any Interactive Data document.

XML processors interpret this attribute differently, so it must not be used.

6.4 Interactive Data Filing Syntax

This section describes the processing of Interactive Data filings.

6.4.1 Reserved.

6.4.2 EDGAR deletes all XBRL attachments from a submission that does not contain an Inline XBRL document if any of the XBRL attachments contains a syntax error.

If the submission does not contain an Inline XBRL document, EDGAR will not suspend the submission; it removes all XBRL documents from the submission and the filer receives an “EDGAR Warning” of an “XBRL Error” in their notification message.

This is performed without regard to whether the submission is live or test. A live filing that is truncated will be distributed without its XBRL attachments.

However, if the submission contains an Inline XBRL document, then an “XBRL Error” will cause the entire submission to be suspended and no part of the submission will be distributed. See 5.2.5 (“Inline XBRL Documents”) of this Volume for more information regarding Inline XBRL.

6.4.3 The XBRL instance documents in a submission must be XBRL 2.1 valid.

EDGAR will suspend a submission with EX-99.SDR K.* or EX-99.SDR L.* with syntax errors.

6.5 Syntax of Instances

This section defines rules governing syntax restrictions on instances. A valid Interactive Data instance is a valid XBRL 2.1 instance, but not all valid XBRL 2.1 instances are valid Interactive Data instances.

Restrictions on the xlink:href, xsi:schemaLocation and schemaLocation attributes, 6.3.6, above, ensure that the DTS of an instance will contain only documents in the same submission or in a standard taxonomy.

6.5.1 The scheme attribute of the xbrli:identifier element must be <http://www.sec.gov/CIK>.

6.5.2 An xbrli:identifier element contains the CIK of the registrant, with one exception

The CIK is an xsd:normalizedString of exactly ten digits from 0 to 9. An xbrli:identifier element must have the CIK of the registrant as its content except in attachment EX-99.SDR L.INS when it contains the financial report of an entity without a CIK.

The xbrli:identifier contains the CIK of the registrant (issuer), not the filer. Fund Series and Class identifiers, which are nine digits following an S or C, are not allowed in the xbrli:identifier element. Document type EX-99.SDR L.INS may contain the financial report of an entity that is not an SEC registrant; in that case use 0000000000 in xbrli:identifier.

If multiple subsidiaries are reported in the same instance, then the CIK of the holding company is the only CIK that will appear in the xbrli:identifier element of that one instance. Section 6.6 below contains rules that apply to the xbrli:context elements in an instance containing a combined submission.

6.5.3 All xbrli:identifier elements in an instance must have identical content.

6.5.4 The xbrli:scenario element must not appear in any xbrli:context.

If the xbrli:scenario element was used, then XBRL requires that it have child elements; however, all dimensions in Interactive Data filings appear in the xbrli:segment element. Therefore, the xbrli:scenario element cannot be used.

6.5.5 Reserved.

6.5.6 Reserved.

6.5.7 An instance must not contain duplicate xbrli:context elements.

An instance must not contain equivalent xbrli:context elements. Contexts are defined to be equivalent if they have S-equal identifiers, equal dateUnion values for startDate, endDate and instant (respectively), and equivalent segment element children. xbrli:segment elements are tested for equality of their children without regard to order (the scenario element is disallowed by another rule).

The id attribute of an xbrli:context element can be any xsd:NCName, but users will find it helpful when it is a mnemonic string that contains the xbrli:period (or fiscal period), and the local part of the QName contents of the dimension attribute and xbrldi:explicitMember. There is usually no need to repeat the registrant name in the id attribute. There is no limit on the length of the id attribute.

The table shows examples of this usage.

Period	xbrli:context id attribute
12 months of fiscal 2007	FY07d
End of fiscal 2007	FY07e
Start of fiscal 2007	FY06e
3 months of the 2nd Quarter of fiscal 2007	FY07Q2d
End of the 2nd Quarter of fiscal 2007	FY07Q2e
9 months year to date in fiscal 2007	FY07M9d
Fiscal 2006 previously reported amounts	FY06d_ScenarioAxis_PreviouslyReportedMember

6.5.8 Every xbrli:context element must appear in at least one contextRef attribute.

Unused xbrli:context elements have no benefit to users and are easily removed by the filer before submission.

6.5.9 Pairs of contexts must not overlap by 24 hours or less.

If the duration of a context is more than 24 hours, then its endDate datetime value must be greater than the startDate datetime of any other context by 24 hours or less.

XBRL 2.1 interprets a date used as a context start date as “midnight at the beginning of” that day. A date used as an instant or endDate in a context means “midnight at the end of” that day.

For example, a company reporting at a May 31st, 2009 fiscal year-end will have contexts whose end date-time is midnight at the end of 2008-05-31 (the prior fiscal year) and contexts whose start date-times are midnight at the beginning of 2008-06-01 (the current fiscal year). It must not

have any contexts with start date of midnight at the beginning of 2008-05-31, and no contexts with end date of midnight at the end of 2008-06-01.

6.5.10 Reserved.

6.5.11 Element `xbrli:xbrl` must not have duplicate child `xbrli:unit` elements.

Element `xbrli:xbrl` must not have equivalent child `xbrli:unit` elements. Units are equivalent if they have equivalent measures or equivalent numerator and denominator. Measures are equivalent if their contents are equivalent QNames. Numerators and Denominators are equivalent if they have a set of equivalent measures.

6.5.12 Fact duplication

An instance must not have more than one fact having S-Equal element names, equal `contextRef` attributes, and if they are present V-Equal `unitRef` attributes and `xml:lang` attributes effective values, respectively, unless their values are consistent, as described below, in which case the distinct facts are consolidated into a single fact for all other validations.

For numeric facts, all such values must be consistent with having been rounded from a single value. Where the `decimals` attribute of such of facts is the same, the values MUST be numerically equal. Where the `decimals` attribute is different, consistency is determined by considering a closed interval centered on each fact value, and of size $10^{(n)}$ where n is the value of the `decimals` attribute, and checking for overlap of all such intervals. For example, a value of 500 with a `decimals` attribute of -2 would result in an interval of 450 to 550, inclusive. A value of 550 with `decimals` attribute of -1 results in an interval of 545 to 555. These values are considered consistent as the intervals overlap; they are both consistent with having been rounded from a value in the range 545 to 550. The use of a closed interval means that intervals are inclusive of the end values.

Consistent numeric facts are consolidated into a single fact having the value of the fact with the maximum specified `decimals` value for purposes of validation and rendering.

For non-numeric facts, the values are consistent only if they are V-Equal.

The `xml:lang` attribute effective value is relevant only for types derived from `normalizedStringItemType` or `stringItemType`. A fact is an occurrence in an instance of an element with a `contextRef` attribute. The values of the `id` attributes are not relevant to detection of duplicate facts. Other rules forbidding equivalent `xbrli:context` and `xbrli:unit` elements ensure that duplicate values of the `contextRef` and `unitRef` attributes can be detected without dereferencing.

The predicate V-Equal is defined in XBRL 2.1 section 1.4, and specifies that non-numeric values are compared after whitespace normalization.

Calculation inconsistencies have no impact on the validity of EDGAR submissions, and therefore the effect of numeric fact duplication in XBRL 2.1 section 5.2.5.2 is moot.

6.5.13 Attribute `xml:lang` defaults to 'en-US'.

The default value of the `xml:lang` attribute on non-numeric facts and on `link:footnote` is 'en-US'. XBRL 2.1 does not specify a default value for the `xml:lang` attribute, but an Interactive Data instance assumes one.

6.5.14 Non-US English content may appear but it must be translated into US English.

An instance having a fact with non-nil content and the xml:lang attribute not equal to “en-US” must also contain a fact using the same element and all other attributes with an xml:lang attribute equal to “en-US”.

For example, the US English fact below can appear in an instance without the French fact, but the French fact cannot appear without the US English fact

```
<eg:q contextRef='x'>yes</eg:q>
```

```
<eg:q contextRef='x' xml:lang='fr'>oui</eg:q>
```

6.5.15 Block text content containing a left angle bracket must be well formed XML.

If the un-escaped content of a fact with base type textBlockItemType or a type equal to or derived by restriction of the type ‘escapedItemType’ in a standard taxonomy schema namespace contains the “<” character followed by a QName and whitespace, “/>” or “>”, then the un-escaped content must contain only a sequence of text and XML nodes. The “<” character may appear in the text content of an XML element as “<”, “<”, “<” or some other guise; when it appears, the content of the element will then be “un-escaped” for analysis. If “<” is followed by a QName and whitespace, then the content is tested for XML well-formedness.

The element name has no significance in this rule, only the base type. For example, although the name of an element in a standard taxonomy having base type textBlockItemType conventionally ends with the string “TextBlock”, the reverse is not always true.

Example XBRL	Text After “un-escaping”	Valid
<eg:AcidityTextBlock> ph<1.0 </eg:AcidityTextBlock>	pH < 1.0	Yes. Without a QName the un-escaped content is not interpreted as markup.
<us-gaap:InventoryTextBlock contextRef="x"> Inventory
 </us-gaap:InventoryTextBlock>	Inventory 	No
<us-gaap:InventoryTextBlock contextRef="x"> 3. Inventory </us-gaap:InventoryTextBlock>	3. Inventory	Yes
<eg:BenchmarkTextBlock> The <i>S&P 500&reg;</i> Index </eg:BenchmarkTextBlock>	The <i>S&P 500&reg;</i> Index	Yes

Some software applications may render the resulting content of the element as if it was embedded in an HTML 3.2 document, and well-formed XML is a prerequisite for well-formed HTML. Considering the status of the text as a kind of management assertion, well-formed XML is in the interest of the registrant because it decreases the likelihood of incorrect renderings.

HTML entity references such as ® must be doubly quoted as shown in the example above. Likewise, an XML entity reference such as & will appear in the un-escaped text only if doubly quoted.

6.5.16 Tag restrictions in 5.2.2 apply to block text XML content

Facts of type “text block” whose un-escaped content contains xml elements must satisfy a content model derived from the BODY tag as defined in 5.2.2.

This specifies the circumstances under which content containing escaped xml elements will be interpreted as HTML. HTML as accepted by EDGAR is detailed in Section 5.2.2 and is based on HTML 3.2 elements and entities, with extensions (such as allowing the “style” attribute) and restrictions (such as disallowing the “textarea” element). Text block content may additionally use HTML elements “span”, “tbody”, “thead”, “tfoot”, and the unprefixd “lang” attribute.

Other than exceptions specified in 6.10, there is no prohibition on escaped HTML elements appearing in other XBRL elements in schemas (such as link:definition) or linkbases (such as link:label), and the renderer (see 6.24) does not un-escape that content.

6.5.17 The xbrli:xbrl element must not have any facts with the precision attribute.

This applies to the submission only; the word “precision” may be used in other ways by XBRL preparation software.

6.5.18 Reserved.

6.5.19 Required contexts

An instance covering a reporting period must contain a Required Context that is an xbrli:context having xbrli:startDate equal to 00:00:00 on the first day of the reporting period and xbrli:endDate equal to 24:00:00 on its last day. This rule defines “Required Context”. For example, this rule applies to Form 10-Q filings for a company with a June 30, 2009 fiscal year end as follows:

Quarter	xbri:startDate	xbri:endDate
1	2009-07-01	2009-09-30
2	2009-07-01	2009-12-31
3	2009-07-01	2010-03-31

Many submissions require a second Required Context; see 6.5.21 below. Required contexts are distinguished by having no xbrli:segment elements.

6.5.20 The required contexts must contain all required document information facts.

For each required Document Information element, an instance should (in some cases, must) contain a fact with that element and a contextRef attribute referring to its Required Context. Required Document Information facts may specify attribute xml:lang, but facts with xml:lang attribute absent or xml:lang equal to “en-US” take precedence over facts having other values for xml:lang.

The Required Document Information elements are:

Element	Base Type	Example values
dei:DocumentType	xsd:string	8-K, 485BPOS, POS AM, SDR K
dei:DocumentPeriodEndDate	YYYY-MM-DD	2009-09-30 ***
dei:AmendmentFlag	xsd:boolean	true
dei:AmendmentDescription	xsd:normalizedString	Correct the company HQ address.
dei:DocumentFiscalYearFocus	YYYY	2021 *
dei:DocumentFiscalPeriodFocus	Q1, Q2, Q3, or FY	FY **

***Note:** The value space of xsd:gYearItemType is a year in the form YYYY.

****Note:** Reports for the 1st, 2nd and 3rd quarters must have a fiscal period focus of Q1, Q2, and Q3 respectively, and annual reports must have fiscal period focus of FY.

*****Note:** See entries for 'period' in Table 7-1 for restrictions on future dates.

The document information facts are required or permitted where indicated for the following submission types and their "/A" variants, and should not appear in other document types:

Element	8-K, 8-K12B, 8-K12G3, 8-K15D5	10-K, 10- KT, 10-Q, 10-QT, 20- F, 40-F	S-1, S-3, S-4, S-11, F-1, F-3, F-4, 10 12B, 10 12G, 20FR12B, 20FR12G, 40FR12B, 40FR12G	POS AM, POS EX	6-K, SP 15D2, K SDR, L SDR	485APOS, 485BPOS, 485BXT, 497, N-1A,	All Other ⁶
DocumentType	X	X	X*	X*	X*	X*	X*
AmendmentDescription	A	A	A	A	A	A	A
DocumentPeriodEndDate	R (use the date of report, date of earliest event reported)	R (use the end of the reporting or transition period)		O* (use the end of the period reported)	R* (use the end of the period reported)	R* (use the filing date)	
DocumentFiscalYearFocus	Op*	R*	Op*	Op*			
DocumentFiscalPeriodFocus	Op*	R*	Op*	Op*			
AmendmentFlag	R*	R*	R*	R*	R*	R*	R*

In the required context:

X – A non-null fact must exist (and in an inline doc, be visible – see 6.5.45).

X* – A non-null fact must exist.

R – A non-null fact should exist (and in an inline doc, be visible).

R* – A non-null fact should exist.

A – A non-null fact should exist if and only if AmendmentFlag is true.

O* – A non-null fact may or may not exist.

Op* – A non-null fact may exist for each of the fiscal year and period focus, or neither.

Blank – A non-null fact should not exist.

6.5.21 The required contexts must contain all required entity information facts.

An instance should (in some cases, must) contain one non-empty fact for each required Entity Information element, each with a contextRef attribute referring to a Required Context. The value of an EntityPublicFloat fact in an instance will be 0 for an entity that has only public debt. Required Entity Information facts may specify attribute xml:lang, but facts with xml:lang attribute absent or xml:lang equal to “en-US” take precedence over facts having other values for xml:lang.

Required Entity Information facts having period type “instant” use a context containing an xbrli:instant value that is the measurement date (for public float and common shares or other units of ownership outstanding), otherwise identical to the Required Context.

Additional facts using the Required Entity Information elements are allowed in other contexts.

⁶ POSASR, S-1MEF, S-3D, S-3DPOS, S-3MEF, S-4EF, S-4MEF, S-4 POS, S-11MEF, F-1MEF, F-3MEF, F-3ASR, F-3D, F-3DPOS, F-3MEF, F-4EF, F-4MEF, F-4POS, F-10EF, F-10POS, SP15D2, and their /A variants

The table below contains a description of the elements along with their base type, as well as providing an example for each.

Element	Base Type	(Example values)
EntityRegistrantName	xsd:normalizedString	General Example Company
EntityCentralIndexKey	dei:centralIndexKey	0005551212
EntityCurrentReportingStatus	dei:yesNo	Yes
EntityVoluntaryFilers	dei:yesNo	No
CurrentFiscalYearEndDate	xbri:gMonthDayItemType	--12-31*
EntityFilerCategory	dei:filerCategory	Large Accelerated Filer
EntityWellKnownSeasonedIssuer	dei:yesNo	Yes
EntityPublicFloat	xsd:decimal	987654321

***Note:** The value space of xbrli:gMonthDayItemType is in the format - - MM-DD.

The following table shows which elements are required or permitted for each document type.

Element	8-K, 8-K12B, 8-K12G3, 8-K15D5	10-K, 10- KT	10-Q, 10- QT	20- F	40- F	10-12B, 10-12G, S-1, S-3, S-4, S-11	POS AM, POS EX	20FR 12B, 20FR 12G	40FR 12B, 40FR 12G	6-K, K SDR, L SDR	All Other ⁷
EntityRegistrantName	X	X	X	X	X	X*	X*	X*	X*	X*	X*
EntityCentralIndexKey	X*	X*	X*	X*	X*	X*	X*	X*	X*	X*	X*
EntityCurrentReportingStatus		R	R	R	R			O*	O*		
EntityVoluntaryFilers (i.e. registrants "not required to file")		R WV		R WV							
EntityInteractiveDataCurrent		R	R	R	R			O*	O*		
CurrentFiscalYearEndDate	E503	R	R*	R	R			O*	O*	R*	
EntityFilerCategory		R	R	R		R*	O*				
EntityWellKnownSeasonedIssuer		R WS WV		R WS WV							
EntityPublicFloat		R									

In the required context:

X – A non-null fact must exist (and in an inline doc, should be visible).

X* – A non-null fact must exist.

R – A non-null fact should exist (and in an inline doc, be visible).

R* – A non-null fact should exist.

⁷ POSASR, S-1MEF, S-3D, S-3DPOS, S-3MEF, S-4EF, S-4MEF, S-4 POS, S-11MEF, F-1, F-1MEF, F-3, F-3MEF, F3ASR, F-3D, F-3DPOS, F-3MEF, F-4, F-4EF, F-4MEF, F-4POS, F-10, F-10EF, F-10POS, SP 15D2, 485APOS, 485BPOS, 485BXT, N-1A, and their /A variants

WS – Must not have EntityWellKnownSeasonedIssuer=Yes and EntityShellCompany=true.
WV – Must not have EntityWellKnownSeasonedIssuer=Yes and EntityVoluntaryFilers=Yes.
E503 – In an inline doc, if Item 5.03 is present, then a non-null fact must exist and be visible, else a non-null fact should not be present.
O* – A non-null fact may or may not exist.
Blank – a non-null fact should not exist.

6.5.22 Reserved.

6.5.23 The Entity CIK in the required context.

The contents of the dei:EntityCentralIndexKey fact in the Required Context must equal the content of the xbrli:identifier element in that context.

6.5.24 The Registrant Name in the required context.

The official Registrant Name in EDGAR (a “conformed name”) that corresponds to the CIK of the xbrli:identifier must match the dei:EntityRegistrantName fact in the Required Context, unless the xbrli:identifier value is 0000000000.

For non-Inline XBRL instances, the conformed name must be a case-insensitive prefix of the fact. For example, conformed name "SMITH" would match fact text "Smith, John". For Inline XBRL files, other differences are ignored:

1. Differences in whitespace or punctuation; hyphen characters (-,-) may either match whitespace or nothing. For example, "ABC Real-time" will match both "A.B.C. RealTime" and "ABC, Real Time".
2. Single-character words are ignored; "A. B. SMITH" will match "SMITH A B".
3. Text entered between "/"", such as "ABC CO /DE/", matches "ABC CO".
4. Matching abbreviations that EDGAR imposes, such as truncating CORPORATION to CORP, shortening "and" to ampersand "&", and dropping the word "The" at the beginning. For example, "The Smith and Jones Corporation" matches "SMITH & JONES CORP".

The cover pages of forms 20-F and 40-F distinguish the "Exact name of Registrant as specified in its charter" from the "Translation of Registrant’s name into English". If either differs from the official Registrant Name, then the filer should provide an additional dei:EntityRegistrantName fact with an xml:lang attribute value other than "en-US".

6.5.25 Domain item type elements are not facts.

Elements with a type attribute equal to or a restriction of ‘domainItemType’ in a standard taxonomy schema target namespace must not appear as facts in an instance. Elements of this type are for use only in the xbrldi:explicitMember element.

6.5.26 Common Stock Shares Outstanding facts.

An instance with dei:DocumentType in the table below must have at least one non-empty dei:EntityCommonStockSharesOutstanding fact for each class of stock or other units of ownership outstanding.

	8-K	10-K, 10-KT, 10-Q, 10-QT	20-F	40-F	20FR12B, 20FR12G	40FR1G, 40FR12G	All Other
EntityCommonStockSharesOutstanding		R	R	O	O*		

R – A non-null fact should exist (and in an inline doc, be visible) in the required context.

O – A non-null fact may exist (if inline, visible) or not exist.

O* - A non-null fact may or may not exist in the required context.

Blank – a non-null fact should not exist.

If the entity represented in the Required Context does not have common shares or other ownership units outstanding, the `dei:EntityCommonStockSharesOutstanding` fact value must be zero. A fact having `xsi:nil='true'` cannot be used to satisfy this criterion.

If an entity represented in the Required Context has only one class of common stock or other units of ownership outstanding, then no matter what the class is named, the instance requires only one `dei:EntityCommonStockSharesOutstanding` fact, and the context of that fact will have an `xbrli:instant` equal to the measurement date, and that context will have no `xbrli:segment` element.

If the entity represented in the Required Context has multiple classes of common shares or other units of ownership outstanding, then the instance must not have any fact with element `EntityCommonStockSharesOutstanding` in a standard namespace in any context without an `xbrli:segment`. Instead, the instance must have a distinct context for each class of common stock or other units of ownership outstanding, and each context will have `xbrli:instant` equal to the measurement date, an `xbrli:segment` with an explicit member of the `StatementClassOfStockAxis` or `ClassesOfShareCapitalAxis` in a standard namespace for each class of stock or other units of ownership, and a fact with element `EntityCommonStockSharesOutstanding` in a standard namespace in each such context. This is a specific case of the more general requirement in 6.6.10 below.

6.5.27 A footnoteLink element must have no children other than loc, footnote, and footnoteArc.

6.5.28 The role attribute of a footnote element must be defined in XBRL 2.1.

6.5.29 The role attribute of a loc element must be empty, or defined in XBRL 2.1.

6.5.30 The arcrole of a footnoteArc element must be defined in XBRL 2.1 or a standard taxonomy.

6.5.31 Reserved.

6.5.32 A link:footnoteLink link:loc xlink:href attribute must start with the sharp sign “#”.

Occurrences of `link:footnote` in an instance can only refer to facts in that instance.

The target of a `link:footnote` locator `xlink:href` attribute may be an element with `xsi:nil='true'`, so registrants should not assume that a “nil” fact is completely equivalent to a “missing” fact.

6.5.33 Every nonempty link:footnote element must be linked to at least one fact.

A nonempty `link:footnote` element must have an `xlink:label` attribute equal to an `xlink:to` attribute of a sibling `link:footnoteArc`.

Footnotes with text must not “dangle”. By contrast, a link:loc element that is not connected by a link:footnoteArc is legal syntax.

6.5.34 Tag restrictions in 5.2.2 apply to footnote content.

The content of a link:footnote element must satisfy a content model derived from the BODY tag as defined in 5.2.2. XBRL 2.1 allows link:footnote elements to contain XHTML. HTML as accepted by EDGAR is detailed in Section 5.2.2 and is based on HTML 3.2 elements and entities, with extensions (such as allowing the “style” attribute) and restrictions (such as disallowing the “textarea” element). Footnote content may additionally use HTML elements “span”, “tbody”, “thead” and “tfoot”.

6.5.35 Unit type registry restrictions apply.

The value of each ‘unitRef’ attribute on each fact of a type in the unit type registry must refer to a unit declaration consistent with the data type of that fact, where consistency is defined by that registry.

XBRL 2.1 already enforces the requirement that a fact of xbrli:monetaryItemType must have a unitRef whose xbrli:measure is an ISO standard currency. A standard numeric data type registry is similar but broader: it has a schema with numeric type declarations, and each numeric data type is associated with consistent unit declaration measures, numerators and denominators.

A portion of a registry is shown below; full details are maintained at <https://www.sec.gov/info/edgar/edgartaxonomies.shtml>.

In this table, for example, facts whose type is based on areaItemType must have a unit whose measure is one of sqft, sqyd, sqmi, etc. The notation “(any)” means that the element namespace is ignored for the purposes of checking for consistency

Type name space	Type element name	Unit measure name space	Unit measure element name	Meaning	Unit numerator namespace	Unit numerator element name	Unit denominator namespace	Unit denominator element name
(any)	percentItem-Type	xbrli	Pure	Rate or Fraction	n/a	disallowed	n/a	disallowed
(any)	perShareItem-Type	n/a	disallowed	Monetary unit per Share	iso4217	3-letter ISO 4217 codes	xbrli	shares
(any)	arealtem Type	(any)	sqft	Square Foot	n/a	disallowed	n/a	disallowed
(any)	arealtem Type	(any)	sqyd	Square Yard	n/a	disallowed	n/a	disallowed
(any)	arealtem Type	(any)	sqmi	Square Mile	n/a	disallowed	n/a	disallowed
(any)	arealtem Type	(any)	A	Acre	n/a	disallowed	n/a	disallowed

6.5.36 The local name in xbrli:measure in UTF-8 must not exceed 200 bytes.

This length restriction applies to bytes in UTF-8 encoding, not characters. See 6.7.29 below.

6.5.37 The decimals attribute value must not cause non-zero digits to be interpreted as zero.

If the decimals attribute of a numeric fact is not equal to 'INF', then the value is interpreted as if certain digits were zero. Interactive Data documents must not contain usage that results in such truncations. The examples below illustrate correct and incorrect use:

Fact text	Decimals value	Interpreted value	Result
-2345.67	INF	-2,345.67	
-2345.67	2	-2,345.67	
-2345.67	0	-2,345.00	Error
-2345.67	-2	-2,300.00	Error
-2345.67	-3	-2000.00	Error
-2345.67	-6	0000.00	Error

For further information and guidance using the decimals attribute, see 6.6.32 below.

6.5.38 Do not use element xbrli:forever in contexts.

6.5.39 The dimension of xbrli:typedMember must be defined in a standard taxonomy.

The value of the dimension attribute must resolve to an element whose namespace-uri denotes a standard taxonomy schema as defined in 6.3.9. In a standard taxonomy, the xbrldt:typedDomainRef attribute will only refer to an element having 'simple' content as defined in XML Schema.

6.5.40 Submission header element values and entity fact values should be consistent.

Submission header elements and entity data values for reporting period, company type, company status, and other data should be consistent with a fact of a corresponding standard element in the required context and in the absence of a submission header element, there should be no such matching fact. As described in the EDGARLink Online Technical Guide, many submission types require certain XML tags and values to appear in the submission header while disallowing others. For example, submission type 10-Q requires element emergingGrowthCompany, but not acceleratedFilerStatus. Note that older versions of the dei namespace do not have all the relevant elements, while 6.5.20 and 6.5.21 in some cases require a fact in the required context that will not appear in the submission header, so the conditions for each such element may differ slightly. The following cases apply:

1. If an instance has dei:CurrentFiscalYearEndDate in the required context, then its month and day should match the fiscal year end month and day, if known, for the filer with CIK matching the required context xbrli:identifier.
2. If a submission header has a periodOfReport element then its date should agree with the value of dei:DocumentPeriodEndDate in the required context. This does not apply when 6.5.20 requires dei:DocumentPeriodEndDate for a submission type that does not have periodOfReport in the header.

3. If a submission has a voluntaryFilerFlag element then its value should agree with the value of dei:EntityVoluntaryFiler in the required context. This does not apply where 6.5.21 requires dei:EntityVoluntaryFiler for a submission type that does not have voluntaryFilerFlag in the header.
4. If a submission has a wellKnownSeasonedIssuerFlag then its value should agree with the value of dei:EntityWellKnownSeasonedIssuer in the required context. This does not apply where 6.5.20 requires dei:EntityWellKnownSeasonedIssuer for a submission type that does not have wellKnownSeasonedIssuer in the header.
5. If a submission has a shellCompanyFlag and there is a EntityShellCompany element in the dei namespace in the DTS of the instance, then the required context should have a dei:EntityShellCompany fact with a value that agrees. If the header does not have a shellCompanyFlag then the required context should not contain such a fact.
6. If a submission has a emergingGrowthCompanyFlag and there is an EntityEmergingGrowthCompany element in the dei namespace in the DTS of the instance, then the required context should have a dei:entityEmergingGrowthCompany fact with a value that agrees. If the header does not have an emergingGrowthCompanyFlag then the required context should not contain such a fact.
7. If a submission has a exTransitionPeriodFlag and there is an EntityExTransitionPeriodFlag element in the dei namespace in the DTS of the instance, then the required context should have a dei:EntityTransitionPeriodFlag fact with a value that agrees. If the header does not have a dei:EntityTransitionPeriodFlag then the required context should not contain such a fact.
8. If the submission type is 485APOS, 485BPOS, 485BXT, 497, or N-1A it has an invCompanyType element in its header. If there is an EntityInvCompanyType element in the dei namespace of the DTS of the instance, then the required context should have a dei:EntityInvCompanyType fact with a value that agrees.
9. If the submission has a smallBusinessFlag and the dei namespace in the DTS of the instance has element EntitySmallBusiness, then the required context should have a dei:AcceleratedFilerStatus fact value *not* containing the words "Smaller Reporting Company" and a dei:EntitySmallBusiness fact with a value that agrees. If the dei namespace in the DTS of the instance does not have element EntitySmallBusiness then the required context should have an dei:AcceleratedFilerStatus fact value containing the words "Smaller Reporting Company".
10. If the submission has an acceleratedFilerStatus element with value "Large Accelerated Filer" then the required context should have a dei:FilerCategory fact value starting with "Large Accelerated Filer".
11. If the submission has an acceleratedFilerStatus element with value "Accelerated Filer" then the required context should have a dei:FilerCategory fact value starting with "Accelerated Filer".
12. If the submission has an acceleratedFilerStatus element with value other than "Large Accelerated Filer" or "Accelerated Filer" then the required context should have a dei:FilerCategory fact value not starting with "Large Accelerated Filer" or "Accelerated Filer".

13. If the submission has no acceleratedFilerStatus element and dei:EntityFilerCategory is not required by 6.5.21 then dei:EntityFilerCategory should not appear in the required context.

The table below, in conjunction with the table in 6.5.21, shows the submission types affected:

Non-null fact in a context	8-K, 8-K12B, 8-K12G3, 8-K15D5	10-K, 10-KT, 10-Q, 10-QT	20-F	40-F	10-12B, 10-12G, S-1, S-3, S-4, S-11	20FR12B , 20FR12G	40FR12B, 40FR12G, F-1, F-3, F-4	POS EX, POS AM	485APOS, 485BPOS, 485BXT, 497, N-1A	All others ⁸
EntityShellCompany		R WS	R WS							
EntitySmallBusiness		R SB			R* SB			O* SB		
EntityEmergingGrowth Company	R	R	R	R	R*	R*	R*	O*		
EntityExTransitionPeriod	ET	ET	ET	ET	ET*	ET*	ET*	ET*		
EntityInvCompanyType									R*	

In the required context:

R – A non-null fact should exist (and in an inline doc, be visible).

R* – A non-null fact should exist.

O* – A non-null fact may or may not exist.

SB – If the registrant file number begins with 811- or 814- then EntitySmallBusiness should not be true.

ET – If EntityEmergingGrowthCompany is "true", then a non-null EntityExTransitionPeriod fact must exist (and in an inline doc, be visible), else EntityExTransitionPeriod should not exist.

ET* – If EntityEmergingGrowthCompany is "true", then a non-null EntityExTransitionPeriod fact must exist, else EntityExTransitionPeriod should not exist.

WS – Must not have EntityWellKnownSeasonedIssuer=Yes and EntityShellCompany=true.

Blank – A non-null fact should not exist.

6.5.41 Each series ID in the header should be a member of LegalEntityAxis in a context.

The series identifiers in the scope of a submission can be determined in one of two ways. If the submission header element rptIncludeAllSeriesFlag is true, then the scope is all active series IDs for all CIKs in the header. Otherwise, the scope contains the set of all values of the seriesId element children of rptSeriesClassInfo. Each series ID in scope should appear as a member of the dei:LegalEntityAxis with element name matching the series ID followed by the word "Member". For example, if the submission header contains:

```
<seriesClasses xmlns="http://www.sec.gov/edgar/coregsc">
  <reportSeriesClass>
```

⁸ 6-K, SP 15D2, POSASR, S-1MEF, S-3ASR, S-3D, S-3DPOS, S-3MEF, S-4EF, S-4MEF, S-4 POS, S-11MEF, F-1MEF, F-3MEF, F-3ASR, F-3D, F-3DPOS, F-3MEF, F-4EF, F-4MEF, F-4POS, F-10EF, F-10POS, and their /A variants

```

<rptSeriesClassInfo xmlns="http://www.sec.gov/edgar/sccommon">
  <seriesId>S000123456</seriesId>
  <includeAllClassesFlag>true</includeAllClassesFlag>
</rptSeriesClassInfo>
</reportSeriesClass>
</seriesClasses>

```

Then there should be at least one context with an explicit member of the dei:LegalEntityAxis whose local name is S000123456Member.

6.5.42 Instances should not use elements declared "deprecated" in any standard namespace.

Each time the edgartaxonomies.xml file is updated with allowed versions of standard namespaces (see 6.22) newer namespaces are normally accompanied with a list of elements from earlier versions that have been deprecated. It is also possible for an obsolete standard namespace to be deprecated as of some date, with no substitute. Subsequent submissions should no longer use the deprecated elements.

For example, suppose element Xyz exists in us-gaap 2018 and is used in a submission that is accepted in February 2019. The new us-gaap 2019 taxonomy then deprecates element Xyz. Subsequently, on March 1st, 2019, us-gaap 2019 is added to the standard taxonomies. Subsequently, the submission previously using Xyz will still be accepted but with an XBRL warning.

6.5.43 Numeric elements that should not have negative values.

Facts having an element name in a standard namespace that is declared "not negative" and not having a contextRef attribute to a context with a consolidation or adjustment axis and an elimination or adjustment member should not have negative values. Sections 6.6.8, 6.6.10, 6.6.11, and 6.6.30 discuss appropriate tagging of numeric values and the distinction between a fact value and its presentation with or without brackets. Standard elements such as OtherExpenses appearing on a consolidated income statement should never be negative; a negative value suggests that element OtherIncome is likely to have been more appropriate. In a consolidating income statement, however, a negative value of OtherExpenses is appropriate when it is an intercompany elimination.

The declaration of numeric elements, combinations of axis and member that should not appear with negative (or positive) values are published at <https://www.sec.gov/info/edgar/signwarnings.json> and updated at the same time that edgartaxonomies.xml is updated (see 6.22).

6.5.44 Custom axis elements should not duplicate standard taxonomy axis elements.

The table below shows a few widely used standard taxonomy axis elements, each playing an important disaggregating or other role in financial reporting. Instances should use these axes instead of custom axes whose names are minor variations. For example, instead of a custom element such as Product "Line", Product "Category" or Product "Type" axis, filers should use the "ProductsOrServices" axis instead.

Standard Taxonomy Axis Elements (sample)
CurrencyAxis

Standard Taxonomy Axis Elements (sample)
ConsolidatedEntitiesAxis
MajorCustomersAxis
ConsolidationItemsAxis
OwnershipAxis
RestatementAxis
StatementGeographicalAxis
ProductsOrServicesAxis
CounterpartyNameAxis

Each time the edgartaxonomies.xml file is updated with allowed versions of standard namespaces (see 6.22) the file <https://www.sec.gov/info/edgar/axiswarnings.json> may be updated with patterns of custom axis names that will result in XBRL warnings.

6.5.45 Cover page facts

For most submission types, an Inline XBRL document set will have "cover page" facts that should be not appear exclusively in an ix:hidden section if they are not null. This defines "visible" facts. Most, but not all dei facts (defined in 5.2.5.14) in a Required Context (defined in 6.5.19) or otherwise required are cover page facts:

Non-null fact in a required context	as required by	dei/cover
AmendmentDescription	6.5.20	dei
DocumentPeriodEndDate	6.5.20	cover
DocumentType	6.5.20	dei
AmendmentFlag	6.5.20	dei
DocumentFiscalPeriodFocus	6.5.20	dei
DocumentFiscalYearFocus	6.5.20	dei
EntityCurrentReportingStatus	6.5.21	cover
EntityFilerCategory	6.5.21	cover
EntityPublicFloat	6.5.21	cover
EntityRegistrantName	6.5.21	cover
EntityVoluntaryFilers	6.5.21	cover
EntityWellKnownSeasonedIssuer	6.5.21	cover
CurrentFiscalYearEndDate	6.5.21	dei
EntityCentralIndexKey	6.5.21	dei
EntityCommonStockSharesOutstanding	6.5.26	cover

Submissions not using Inline XBRL are subject to 6.5.20, 6.5.21 and 6.5.26, but have no "ix:hidden" section. Therefore, whether such facts are cover page facts is irrelevant.

Although locating cover page facts within an attachment so that they are shown on the first page displayed in a browser is preferred, cover page facts need not appear in any specific file attachment within an Inline XBRL document set.

6.5.46 Registered securities cover page information

Filers having only a single registered security should have facts in the Required Context for the following elements:

Element	cover/dei	Type	Sample values (remarks)
Security12bTitle	cover	normalizedString up to 150 characters	(Equity): "Common Shares", "Class A Common Stock", "2.2% Exchangeable Subordinated Debentures due 2031", "American Depositary Shares, each representing one half of an Ordinary Share" (Debt): "Series A, Fixed Rate Step-Up Notes Due 2029"
NoTradingSymbolFlag	dei	Fixed value "true"	"true"
TradingSymbol	cover	normalizedString up to 25 characters	"XYZ", "XYZ.A";
SecurityExchangeName	cover	EDGAR national exchange codes	"NYSE" (Other than National Exchanges are not tagged).
Security12gTitle	cover	normalizedString up to 150 characters	"Preferred Shares", "Common shares"
SecurityReportingObligation	cover	Fixed value "15(d)"	(indicates whether security has 15(d) reporting obligation)

For different submission types, registered securities facts are required, optional, or disallowed. The dei fact DocumentType matches the submission type (see 6.5.20). In the table below and in the remainder of section 6.5, the columns represent all variants of different forms, both the variants of 8-K including 8-K12B, 8-K12G3, and 8-K15D5, and amending forms (10-K is amended by a 10-K/A, 8-K12B is amended by 8-K12B/A).

Non-null fact in a context	8-K, 10-K, 10-KT, 10-Q, 10-QT	20-F, 40-F	10-12B, 10-12G	20FR12G, 40FR12G, 20FR12B, 40FR12B	485APOS, 485BPOS, 485BXT, N-1A, N-1A/A	All others
Security12bTitle	Ot1	Ot1	Ot1*	Ot1*		
Security12gTitle	Ot1	Ot1	Ot1*	Ot1*		
SecurityExchangeName	Te	Te	Te*	Te*		
NoTradingSymbolFlag	T1	T1	T1*	T1*		
TradingSymbol	T1	T1	T1*	T1*	O*	
SecurityReportingObligation		O		O*		

Ot1 – If inline, then one, but not both non-null facts may exist and be visible. If an ADR, see below.

Ot1* – One, but not both non-null facts may exist. If an ADR, see below.

Te – If inline and Security12bTitle exists, then a non-null fact should exist and be visible.

Te* – If Security12bTitle exists, then a non-null fact should exist.

T1 – If inline and Security12bTitle exists, then one non-null fact, but not both, should exist and be visible.

T1* – If Security12bTitle exists, then one null-null fact, but not both, may exist.

O – If inline, a non-null fact may exist and be visible.

O* - A non-null fact may exist.

Blank – a non-null fact should not exist.

An "ADR Context" is defined as a context with dei:AdrMember as a member of StatementClassOfStockAxis or ClassesOfShareCapitalAxis. If Security12bTitle exists in an ADR Context, then conditions Te and Te* are treated as O and O*, respectively, in all contexts other than ADR Contexts.

Transformation ixt-sec:exchnameen in section 5.2.5.12 transforms commonly used names of exchanges to their internal EDGAR acronyms; for example, the following Inline XBRL element will display in a browser as "The New York Stock Exchange, Inc." but produce the cover page fact with value "NYSE":

```
<ix:nonNumeric contextRef="x" name="dei:SecurityExchangeName"
                format="ixt-sec:exchnameen" >The New York Stock Exchange,
                Inc.</ix:nonNumeric>
```

Filers having multiple registered securities each on a different exchange will have cover page facts using these same elements. For each distinct security, the conditions apply to facts within a separate context having a distinct (possibly custom) members of dimensions StatementClassOfStockAxis or ClassesOfShareCapitalAxis in a standard namespace (use the same axis whether a member represents a debt or equity security). Just as with the cover page fact EntityCommonStockSharesOutstanding (see 6.5.26), the presence of more than one security means that there should be no SecurityTitle fact in a context having no dimensions.

For securities trading on multiple exchanges, disaggregate the context where the security title is provided by using dimension EntityListingsExchangeAxis in a standard namespace with members drawn from a standard namespace containing "/exch/". (Keep in mind that Inline XBRL allows a single text fragment to be tagged as more than one fact by nesting ix:nonNumeric tags).

For example, assume there are contexts c, ca, cb, cany, and cach, having the same period and the following axes and members. Contexts "cany" and "cach" disaggregate context "ca":

example context	StatementClassOfStockAxis members	EntityListingsExchangeAxis members
c		
ca	us-gaap:ClassASharesMember	
cb	us-gaap:ClassBSharesMember	
cany	us-gaap:ClassASharesMember	exch:XYYS
cach	us-gaap:ClassASharesMember	exch:XCHI
cfn	xyz:FixedNotes2022Member	
c1	dei:AdrMember	
c2	us-gaap:CommonStockMember	

A US company reporting one class of 12(b) securities trading on a national exchange needs only the required context and three facts:

example context	element	example value
c	Security12bTitle	Common Shares
c	TradingSymbol	XYZ
c	SecurityExchangeName	NYSE

A US company with only an over-the-counter 12(g) security need only provide its title:

example context	element	example value
c	Security12gTitle	Common Shares

A company with multiple classes of securities uses additional contexts to distinguish the data about them:

example context	element	example value	remarks
cany	Security12bTitle	Common Class A	Common Class A, a 12(b) security with the same symbol XYZ traded on New York and Chicago Exchanges.
cach	Security12bTitle	Common Class A	
cany	TradingSymbol	XYZ	
cach	TradingSymbol	XYZ	
cany	SecurityExchangeName	NYSE	
cach	SecurityExchangeName	XCHI	
cb	Security12bTitle	Common Class B	Common Class B, a 12(b) security traded only in New York. So, context cb has no exchange axis member.
cb	TradingSymbol	XYZ.B	
cb	SecurityExchangeName	NYSE	
cfn	Security12bTitle	2.375% Notes due 2022	Registered debt traded in New York without a trading symbol.
cfn	NoTradingSymbolFlag	true	
cfn	SecurityExchangeName	NYSE	

Non-US companies with American Depositary Receipts (ADR) on a US national exchange should distinguish them from the underlying share class. The trading symbol represents the ADR, not the underlying shares, in the following way:

example context	element	example value	remarks
c1	Security12bTitle	ADR representing one half of an Ordinary share	It is the ADR that is traded on NYSE, not the Ordinary shares.
c1	TradingSymbol	ZYX	

example context	element	example value	remarks
c1	SecurityExchangeName	NYSE	
c2	Security12bTitle	Ordinary shares	These ordinary shares are only traded on non-US national exchanges.
c2	NoTradingSymbolFlag	true	

Note that the examples above only illustrate tagged facts and their contexts; it has no bearing on whether any particular registration would be permitted by SEC regulations.

6.5.47 Company identification cover page elements

Different submission types require different types of company identifying information:

Cover page fact element	cover/dei	Type	Example values
EntityFileNumber	cover	fileNumber	"001-12345"
EntityIncorporationStateCountryCode	cover	2-character string	"KY"; "E9"
EntityPrimarySicNumber	cover	sicNumber	"1234"
EntityTaxIdentificationNumber	cover	employerId	"12-3456789"

Transformation `ixt-sec:edgarprovcountry` (see 5.2.5.12) transforms the full names of countries and Canadian provinces to their 2-character EDGAR codes; for example, the following Inline XBRL element will display in a browser as "China" but produce a cover page fact value "F4":

```
<ix:nonNumeric contextRef="x" name="dei:EntityIncorporationStateCountryCode" format="ixt-sec:edgarprovcountry" >China</ix:nonNumeric>
```

The table below shows the conditions for the facts to appear in different document types:

Non-null fact in the required context	8-K, 10-K, 10-KT, 10-Q, 10-QT	20-F	40-F	10-12B, 10-12G	S-1, S-3, S-4, S-11, POS EX, POS AM, F-1, F-3, F-4	20FR12B, 20FR12G	40FR12B, 40FR12G	6-K	All others
EntityFileNumber	Ri	Ri	Ri			O*	O*	O*	
EntityIncorporationStateCountryCode	Ri	Ri	Ri	O*	O*	O*	O*		
EntityTaxIdentificationNumber (shown on forms as "IRS Employer ID")	Ri		O	O*	O*		O*		
EntityPrimarySicNumber			O		O*		O*		

Ri – If an inline document, then a non-null fact should exist in the required context and be visible; else may or may not exist.

O – A non-null fact may exist (and if inline, be visible) or may not exist.

O* – A non-null fact may exist or may not exist.

Blank – a non-null fact should not exist.

6.5.48 Cover page principal office address elements

Many submission types require a company principal office address on the cover page.

Cover page fact element	dei/cover	Type	Example values
EntityAddressAddressLine1	cover	normalizedString	1 Lakeside Drive
EntityAddressAddressLine2	cover	normalizedString	Research Campus
EntityAddressAddressLine3	cover	normalizedString	Park Royal
EntityAddressCityOrTown	cover	normalizedString	Edmonton
EntityAddressStateOrProvince	cover	postalStateOrProvinceCode	AB
EntityAddressCountry	cover	ianaCountryCode	CA
EntityAddressPostalZipCode	cover	normalizedString	NW10 7HQ
CityAreaCode	cover	normalizedString	+44 (0)20
LocalPhoneNumber	cover	normalizedString	8978 6000

The principal address does not use the EDGAR state/country codes. Transformations that convert state and province names to postal codes, and that convert country names to IANA (ISO 3166-1 alpha 2) codes are shown in section 5.2.5.12. For example, the Inline XBRL elements below represent two facts with values "AB" and "CA", respectively:

```
<ix:nonNumeric contextRef="x" name="dei:EntityAddressStateOrProvince"
format="ixt-sec:stateprovnameen" >Alberta</ix:nonNumeric>
<ix:nonNumeric contextRef="x" name="dei:EntityAddressCountry"
format="ixt-sec:countrysnameen" >Canada</ix:nonNumeric>
```

Each fact in the principal address should appear in the required context.

Non-null fact in the required context	8-K, 10-K, 10-KT, 10-Q, 10-QT	20-F	40-F	10-12B, 10-12G	20FR12B, 20FR12G, 6-K	40FR12B, 40FR12G	S-1, S-3, S-4, S-11, POS EX, POS AM, F-1, F-3, F-4	All others
EntityAddressAddressLine1	Ri	Ri	Ri	O*	O*	O*	O*	
EntityAddressAddressLine2	O	O	O	OL1*	OL1*	OL1*	OL1*	
EntityAddressAddressLine3	OL2	OL2	OL2	OL2*	OL2*	OL2*	OL2*	
EntityAddressCityOrTown	Ri	Ri	Ri	O*	O*	O*	O*	
EntityAddressStateOrProvince	O2	O	O2	O*	O*	O*	O*	
EntityAddressCountry	O2	Ri	O2	O*	O*		O*	
EntityAddressPostalZipCode	Ri	Ri	Ri	O*	O*	O*	O*	
CityAreaCode	Ri		Ri	Oph*		Oph*	Oph*	
LocalPhoneNumber	Ri		Ri	O*		O*	O*	

Ri – If an inline document, then a non-null fact should exist in the required context and be visible; else may or may not exist.

O – A non-null fact may exist (and if inline, be visible) or may not exist in the required context.

O* – A non-null fact may or may not exist in the required context.

O2 – One or both non-null facts should exist (and if inline, be visible), because forms may be filed by non-US companies.

OL1* – A non-null fact may exist only if EntityAddressAddressLine1 exists.

OL2- A non-null fact may exist (and if inline, be visible) only if EntityAddressAddressLine2 exists.

OL2*- A non-null fact may exist only if EntityAddressAddressLine2 exists.

Oph* – A non-null fact should exist if and only if a non-null LocalPhoneNumber is present.

Blank – a non-null fact should not exist.

6.5.49 Cover page elements and submission types

Some forms serve multiple purposes. There are check boxes on their cover page to distinguish them. Form 10-K, for example, can be submitted to EDGAR either as submission type "10-K" when it covers a 12-month period or as a "10-KT" when it is a transition report for a company changing its fiscal year end date. Forms 20-F and 40-F are used for both initial registration and for annual reports.

Cover page fact element	dei/cover	Type	Example values
DocumentAnnualReport	cover	boolean	true
AnnualInformationForm	cover	boolean	true
AuditedAnnualFinancialStatements	cover	boolean	true
DocumentQuarterlyReport	cover	boolean	true
DocumentTransitionReport	cover	boolean	true
DocumentPeriodStartDate	cover	yyyy-mm-dd	2019-04-01
DocumentShellCompanyReport	cover	boolean	true
DocumentShellCompanyEventDate	cover	yyyy-mm-dd	2019-07-11
DocumentRegistrationStatement	cover	boolean	true
EntityBankruptcyProceedingsReportingCurrent	cover	boolean	true
DocumentsIncorporatedByReferenceText-Block	cover	text block	(see 6.5.16)

Each fact should appear in the required context for some submission types:

Non-null fact in the required context	10-K	10-KT	10-Q	10-QT	20-F	40-F	20FR12B, 20FR12G	40FR12B, 40FR12G	All others
DocumentAnnualReport	Y	N			TF3	Y	N*	N*	
AnnualInformationForm						Ri			
AuditedAnnualFinancialStatements						Ri			
DocumentQuarterlyReport			Y	N					
DocumentTransitionReport	N	Y	N	Y	TF3		N*		
DocumentPeriodStartDate		Ri		Ri	TR				
DocumentShellCompanyReport					TF3		N*		
DocumentShellCompanyEventDate					SR				
DocumentRegistrationStatement					N	N	Y*	Y*	

Non-null fact in the required context	10-K	10-KT	10-Q	10-QT	20-F	40-F	20FR12B, 20FR12G	40FR12B, 40FR12G	All others
EntityBankruptcy- Proceedings- ReportingCurrent	O	O	O	O	O		O*		
DocumentsIncorporat edByReferenceText- Block	O	O							

Ri – If an inline document, then a non-null fact should exist and visible, else it may or may not exist.

Y – If an inline document, then a fact should exist with value "true", else it may or may not exist.

Y* - if the fact exists, then it should have the value "true", else it may or may not exist.

N – If an inline document, then a fact should exist with value "false", else it may or may not exist.

N* - If the fact exists, then it should have the value "false", else it may or may not exist.

TF3 – If an inline document, then all three facts marked TF3 should exist, with one being "true" and the other two "false"; else they may or may not exist, with one being "true" and the other two "false".

TR – If an inline document and DocumentTransitionReport is true, then a non-null fact should exist and be visible; else a fact may or may not exist.

SR – If an inline document and DocumentShellCompanyReport is true, then a non-null fact should exist and be visible; else a fact may or may not exist.

O – A non-null fact may exist (and if inline, be visible) or may not exist in the required context.

O* – A non-null fact may or may not exist in the required context.

Blank – a non-null fact should not appear.

6.5.50 Form 20-F cover page elements

Form 20-F has two unique cover page facts, both of which have only a small range of case-sensitive values that match the text appearing on Form 20-F:

Cover page fact element	dei/cover	Type	Valid case-sensitive values
DocumentAccountingStandard	cover	string	U.S. GAAP International Financial Reporting Standards Other
OtherReportingStandardItemNumber	cover	string	Item 17 Item 18

This table shows the submission types permitting these elements:

Non-null fact in the required context	20-F	20FR12B, 20FR12G	All Other
DocumentAccountingStandard	Ri	O*	
OtherReportingStandardItemNumber	Oth	Oth*	

Ri – If an inline document, then a non-null fact should be present and visible, else it may or may not exist.

O* – A non-null fact may or may not exist.

Oth – A non-null fact should exist (and if inline, be visible) if and only if DocumentAccountingStandard value is "Other".

Oth* - A non-null fact should exist if and only if DocumentAccountingStandard value is "Other".

Blank – a non-null fact should not appear.

6.5.51 Business contact addresses

Forms 20-F and 40-F require a contact person and address. The elements are a superset of those required for the principal address:

Cover page fact element	dei/cover	Type	Example values
ContactPersonnelName	cover	normalizedString	Raymond Lomas
EntityAddressAddressLine1	cover	normalizedString	Iron Tower
EntityAddressAddressLine2	cover	normalizedString	120 Trunk Road
EntityAddressAddressLine3	cover	normalizedString	Clyde Campus
EntityAddressCityOrTown	cover	normalizedString	Blackpool
EntityAddressStateOrProvince	cover	normalizedString	QC
EntityAddressCountry	cover	normalizedString	CA
EntityAddressPostalZipCode	cover	normalizedString	J0J 0B8
CityAreaCode	cover	normalizedString	450
LocalPhoneNumber	cover	normalizedString	499-4999
ContactPersonnelFax	cover	normalizedString	(450) 499-5000
ContactPersonnelEmailAddress	cover	normalizedString	rl@example.com

Unlike the principal address in the required context, the contact facts should appear in a context that has the same period as the required context, but with dimension EntityAddressesAddressTypeAxis and member BusinessContactMember, both in a standard namespace.

Non-null fact in the Business Contact Member context:	20-F	40-F	20FR12B, 20FR12G	40FR12B, 40FR12G	F-1, F-3, F-4	S-1, S-3, S-4, S-11, POS AM, POS EX	All others
ContactPersonnelName	Ri	Ri	O*	O*	O*	O*	
EntityAddressAddressLine1	Ri	Ri	O*	O*	O*	O*	
EntityAddressAddressLine2	O	O	OL1*	OL1*	OL1*	OL1*	
EntityAddressAddressLine3	OL2	OL2	OL2*	OL2*	OL2*	OL2*	
EntityAddressCityOrTown	Ri	Ri	O*	O*	O*	O*	
EntityAddressStateOrProvince	O2	Ru	O*	Ou*	O*	O*	
EntityAddressCountry	O2		O*		O*		
EntityAddressPostalZipCode	Ri	Ri	O*	O*	O*	O*	
CityAreaCode	Oph	Ri	Oph*	Oph*	Oph*	Oph*	
LocalPhoneNumber	O3	Ri	O*	O*	O*	O*	
ContactPersonnelFax	O3		O*	O*			
ContactPersonnelEmailAddress	O3		O*	O*			

Ri – If an inline document, then a non-null fact should be present and visible, else it may or may not exist.

Ru – If an inline document, then a non-null fact should exist and be visible with a US state as a value.

- Ou* – A non-null fact may exist with a US state as a value.
- O – A non-null fact may exist (and if inline, be visible) or may not exist.
- O* - A non-null fact may or may not exist.
- O2 – If an inline document, then one or both non-null facts should exist and be visible; else may or may not exist.
- O3 – If an inline document, then one, two, or all three non-null facts should exist and be visible.
- Oph – A non-null fact should exist (and if inline, be visible) if and only if a non-null LocalPhoneNumber is present.
- Oph* – A non-null fact should exist if and only if a non-null LocalPhoneNumber is present.
- OL1* – A non-null fact may be present only if EntityAddressAddressLine1 is present.
- OL2* – A non-null fact may be present only if EntityAddressAddressLine2 is present.
- OL2 – A non-null fact may exist (and if inline, be visible) only if EntityAddressAddressLine2 is present.
- Blank – a non-null fact should not appear.

6.5.52 Form 8-K cover page and name changes

Although not every 8-K is required to contain Interactive Data, an 8-K submitted as Inline XBRL has four cover page facts required to represent the checkboxes on the cover. Also, the 8-K and 10-Q variants allow filers to report a name change:

Cover page fact element	dei/cover	Type	Sample values
WrittenCommunications	cover	boolean	false
SolicitingMaterial	cover	boolean	false
PreCommencementTenderOffer	cover	boolean	false
PreCommencementIssuerTenderOffer	cover	boolean	false
EntityInformationFormerLegalOrRegisteredName	cover	normalizedString	OLDCO INC.

Each fact should appear in the required context, whether true or false:

Non-null fact in the required context	8-K	10-Q, 10-QT	All Others
WrittenCommunications	Ri		
SolicitingMaterial	Ri		
PreCommencementTenderOffer	Ri		
PreCommencementIssuerTenderOffer	Ri		
EntityInformationFormerLegalOrRegisteredName	O	O	

- Ri – If an inline document, then a non-null fact should be present and visible, else it may or may not exist.
- O – A non-null fact may exist (and if inline, should be visible) or may not exist.
- Blank – a non-null fact should not be present.

6.5.53 Former address cover page elements

Although not every 8-K is required to contain Interactive Data, an 8-K submitted as Inline XBRL may include a former address. Unlike the principal address in the required context, these facts should appear in a context that has the same period as the required context, but with dimension EntityAddressesAddressTypeAxis and member FormerAddressMember, both in a standard namespace.

Non-null fact in the former address context	8-K, 10-Q, 10-QT	All others
EntityAddressAddressLine1	O	
EntityAddressAddressLine2	OL1	
EntityAddressAddressLine3	OL2	
EntityAddressCityOrTown	OL1	
EntityAddressStateOrProvince	F2	
EntityAddressCountry	F2	
EntityAddressPostalZipCode	OL1	

O – A non-null fact may exist (and if inline, be visible) or may not exist.

OL1 – A non-null fact may exist (and if inline, be visible) only if EntityAddressAddressLine1 is present.

OL2 – A non-null fact may exist (and if inline, be visible) only if EntityAddressAddressLine2 is present.

F2 – One or both non-null facts should exist (and if inline, be visible) if EntityAddressAddressLine1 is present.

Blank – a non-null fact should not be present.

6.6 Contents of Instance Data

This section describes requirements on the content of instance data.

6.6.1 Fiscal year contexts for non-numeric facts.

In an instance reporting a fiscal year, non-numeric facts containing text about any portion of that or a prior year must have a contextRef attribute to an xbrli:context for the reporting period year. For example, in a fiscal year 2009 report a company describes litigation settled in fiscal 2007. Nevertheless, the disclosure text should be in a context for fiscal 2009. A reporting period begins at 00:00:00 of its first day and ends at 24:00:00 of its last day, which is the XBRL 2.1 default for periods. Only the date, not the time part of the ISO 8601 date-times, should be used in contexts.

6.6.2 Fiscal year-to-date contexts for non-numeric facts.

In an instance reporting a fiscal year-to-date, the non-numeric facts containing text about any portion of the year-to-date or prior year must have a contextRef attribute to an xbrli:context representing the year-to-date. For example, a company completes an acquisition in its second fiscal quarter. In its 3rd quarter fiscal report, the Acquisitions note contains text describing that same acquisition. The 3rd quarter text should be in the context for the first nine months (that is, the year-to-date).

6.6.3 Contexts for the primary registrant.

Facts about a registrant must have an xbrli:context element in the default legal entity, except for facts that apply only to a reportable segment or a subsidiary with separate reporting obligations to the Commission. A context is defined to be “in the default legal entity” if and only if it has no xbrldi:explicitMember with a dimension attribute equal to ‘LegalEntityAxis’ in a standard namespace and ‘ConsolidatedEntitiesAxis’ in a standard namespace.

The entity that it refers to is a Consolidated Entity, which is also the entity in the xbrli:identifier element of the Required Context.

For example, ABC company is a consolidated entity if it has subsidiaries DEF and GHI that each have separate reporting obligations to the Commission. ABC is a consolidated entity from the point of view of an instance that contains data about both DEF and GHI.

For example, suppose fund family MNO has two series MNOX and MNOY. MNO is considered a ‘consolidated entity’. An instance containing the annual statements (or schedule of investments, or risk/return summaries) for both MNOX and MNOY must use the CIK of the fund family MNO as the xbrli:identifier. In that instance, MNO is the “default legal entity”.

6.6.4 Consolidating entities and subsidiaries.

If facts about a Consolidated Entity and one or more of its subsidiaries, each with separate reporting obligations to the Commission, appear in a single instance, it is a consolidating instance. This is common when disclosures about the parent company and its separately reporting subsidiaries are included in a single, combined filing, such as Form 10-K.

For example, parent company ABC is a consolidated entity from the point of view of an instance that contains the reconciliation of ABC’s consolidated statements with its subsidiaries DEF and GHI.

For example, an instance containing the annual statements (or schedule of investments, or risk/return summaries) for two series MNOX and MNOY of fund family MNO is considered a “consolidating instance” for the “consolidated entity” MNO. Even if the report does not, strictly speaking, contain a “consolidation”, it will contain facts such as narratives that apply to all the series.

6.6.5 Consolidated entities.

The contexts for facts about entities other than the Consolidated Entity must have xbrli:explicitMember with a dimension attribute value in a standard namespace and equal to LegalEntityAxis or ConsolidatedEntitiesAxis in a standard namespace, and distinct values. The dimension member must be declared in the company extension schema and linkbases as detailed in Sections 6.7 and 6.16 below.

The LegalEntityAxis axis may be used in any instance; a consolidating instance that contains facts about multiple subsidiaries of the consolidated entity should use ConsolidatedEntitiesAxis.

For example, a holding company (ABC) files a Form 10-K that contains:

1. The face financial statements of the holding company on a consolidated basis;
2. Selected financial data for each of its separately reporting subsidiaries (DEF and GHI)

If facts about all entities ABC, DEF and GHI are included in a single instance document, at least three contexts are required, as shown below.

```
<xbrli:context id="FY09"><xbrli:entity>  
<xbrli:identifier scheme="http://www.sec.gov/CIK">9999999999</xbrli:identifier>  
</xbrli:entity>...</xbrli:context>
```

```
<xbrli:context id="FY09_DEF"><xbrli:entity>  
<xbrli:identifier scheme="http://www.sec.gov/CIK">9999999999</xbrli:identifier>  
<xbrli:segment>
```

```

<xbrldi:explicitMember dimension="xyz:ConsolidatedEntitiesAxis
">abc:DEF_Member</xbrldi:explicitMember>
</xbrli:segment>
</xbrli:entity>...</xbrli:context>

```

```

<xbrli:context id="FY09_GHI"><xbrli:entity>
<xbrli:identifier scheme="http://www.sec.gov/CIK">9999999999</xbrli:identifier>
<xbrli:segment>
<xbrldi:explicitMember dimension=" xyz:ConsolidatedEntitiesAxis
">abc:GHI_Member</xbrldi:explicitMember>
</xbrli:segment>
</xbrli:entity
>...</xbrli:context>

```

In the above example, assume namespace prefix “abc” is bound to the company’s extension schema namespace and "xyz" is bound to some standard namespace.

If facts about two investment fund series MNOX and MNOY are included in a single instance, then at least three contexts are required, as shown below. The MNO context is the context for facts such as dei:EntityCentralIndexKey or accounting policies text blocks. A Fund Series identifier must not appear as an xbrli:identifier; it must appear only as the element name of the domain member representing the series (“S777777777”, “S666666666”)

```

<xbrli:context id="FY09_MNO"><xbrli:entity>
<xbrli:identifier scheme="http://www.sec.gov/CIK" >8888888888</xbrli:identifier>
</xbrli:entity>...</xbrli:context>
<xbrli:context id="FY09_MNOX"><xbrli:entity>
<xbrli:identifier scheme="http://www.sec.gov/CIK" >8888888888</xbrli:identifier>
<xbrli:segment>
<xbrldi:explicitMember dimension="dei:LegalEntityAxis"
>mno:S777777777Member</xbrldi:explicitMember>
</xbrli:segment>
</xbrli:entity>...</xbrli:context>
<xbrli:context id="FY09_MNOY" ><xbrli:entity>
<xbrli:identifier scheme="http://www.sec.gov/CIK" >8888888888</xbrli:identifier>
<xbrli:segment>
<xbrldi:explicitMember dimension="dei:LegalEntityAxis"
>mno:S666666666Member</xbrldi:explicitMember>
</xbrli:segment></xbrli:entity>...</xbrli:context>

```

In the above example, assume namespace prefix “mno” is bound to the fund family’s extension schema namespace.

6.6.6 Contexts for subsidiaries in a consolidating instance.

Facts in a consolidating instance with a context that names an entity with subsidiaries, applies collectively to subsidiaries within that subset. For example, ABC is a public holding company whose submission has an instance document containing the consolidated statements of ABC, selected financial data of each of its separately reporting subsidiaries DEF, GHI and JKL, a (non-numeric) Note to the Financial Statements applies only to subsidiaries GHI and JKL (but not to

DEF), and in that note appears a material (numeric) figure “USD 30m” for (say) combined expenses of GHI and JKL. If the numeric fact is reported then a new domain member (such as “G_J”) could be used to denote the combination of those two subsidiaries, and be used in the `xbrldi:explicitMember` element. In this example the namespace prefix “abc” is bound to the company’s extension taxonomy namespace and “xyz” is bound to some standard namespace:

```
<xbrli:context id="FY09_GHIJKL"><xbrli:entity>  
<xbrli:identifier scheme="http://www.sec.gov/CIK">9999999999</xbrli:identifier>  
<xbrli:segment>  
<xbrldi:explicitMember dimension="xyz:ConsolidatedEntitiesAxis  
">abc:G_J_Member</xbrldi:explicitMember>  
</xbrli:segment>  
</xbrli:entity>...</xbrli:context>
```

However, creating such “synthetic” entities should be avoided if the only facts that will be disclosed are non-numeric; in that case it is much better to simply duplicate the non-numeric facts. For example, consider a fund prospectus that contains a narrative fee disclosure that applies to share classes A, B and T, but a different narrative fee disclosure that applies to share class I. Tag the disclosure separately, once for each class. Do not create a new domain member named ABT to denote the combination of classes for the context of the narrative fee disclosure.

Section 6.16 below requires the members of each domain to form a tree, so that if GHI and JKL are combined in a context as G_J_Member, then DEF and GHI cannot also be combined for a different context in any instance using the same company extension schema and linkbases.

6.6.7 Contexts for a consolidating instance.

In a consolidating instance, facts that apply only to the parent company and not to any specific subsidiaries must have contexts whose `xbrldi:explicitMember` elements have a dimension attribute of `ConsolidatedEntitiesAxis` in a standard namespace and value `ParentCompanyMember` in a standard namespace.

6.6.8 Contexts for consolidating eliminations.

In a consolidating instance, facts that apply only to eliminations between subsidiaries must have contexts whose `xbrldi:explicitMember` elements have a dimension attribute of `ConsolidationItemsAxis` in a standard namespace and value `ConsolidationEliminationsMember` in a standard namespace. Rule 6.6.4 above gives the registrant freedom to define domain members to identify subsidiaries, but the registrant cannot choose the domain member to use for eliminations.

For example, a consolidation of ABC with its subsidiaries contains corporate headquarters expenses that must be expressed as facts about `ParentCompanyMember`, and eliminations between DEF and GHI that are expressed as (negative) figures that are facts about `ConsolidationEliminationsMember`.

6.6.9 Contexts for all classes of stock.

Facts that apply to all classes of stock in a submission must have an `xbrli:context` element without a dimension attribute equal to `StatementClassOfStockAxis` or `ClassesOfShareCapitalAxis` in a standard namespace.

6.6.10 Contexts for a single class of stock.

An instance containing facts that are only specific to distinct stock classes in a statement must distinguish those facts using `xbrli:context` elements whose `xbrldi:explicitMember` elements have a dimension attribute of `StatementClassOfStockAxis` or `ClassesOfShareCapitalAxis` in a standard namespace. Many “generic” stock class domain members appear in standard namespaces (for example, `CommonClassAMember`), but filers may also create new domain members to refer to specific classes.

Investment company submissions cannot use the generic members. Each class in each series must be defined as a domain member with element name equal to the Fund Class identifier (for example, if MNOX has A, B and I classes, they might be domain members `C555555555`, `C555555556`, `C555555557`).

6.6.11 Contexts for different reporting assumptions.

An instance containing multiple reports about the same entity for the same periods under different reporting assumptions must distinguish the facts in different reports using `xbrli:context` elements whose `xbrldi:explicitMember` elements have a dimension attribute of `StatementScenarioAxis` in a standard namespace. Although the default scenario domain member is normally used to the same effect as “actual”, when there is more than one entirely distinct report the `Scenario` axis is used to distinguish the reports.

6.6.12 Facts on the face of financial statements.

An instance must contain a fact for each combination of line item and period that appears on the face of the financial statements of the corresponding official HTML/ASCII document. For example, a small fragment of a balance sheet:

Line Item	Amount in thousands (2007)	Amount in thousands (2006)
Land	\$31,659	\$31,601

This example corresponds to these two facts for the combination of line item and period:

```
<us-gaap:Land unitRef="usd" decimals="-3" contextRef="FY07e">31659000</us-gaap:Land>  
<us-gaap:Land unitRef="usd" decimals="-3" contextRef="FY06e">31601000</us-gaap:Land>
```

6.6.13 Reserved

6.6.14 Facts for parenthetical disclosures on the face of financial statements.

An instance must contain a fact for each amount disclosed parenthetically in line items that appear on the face of the financial statements of the corresponding official HTML/ASCII document. Example:

Line Item	Amount in thousands (2007)
Receivables (net of allowance for bad debts of \$200 in 2007)	\$700

The instance contains two facts:

```
<us-gaap:AccountsReceivableNetCurrent unitRef="usd" decimals="-3" contextRef="FY07e">700000</us-gaap:AccountsReceivableNetCurrent>
```

<us-gaap:AllowanceForDoubtfulAccountsReceivableCurrent unitRef="usd" decimals="-3" contextRef="FY07e">200000</us-gaap:AllowanceForDoubtfulAccountsReceivableCurrent>

For example, note that even if Receivables had been \$1,000 at the end of 2006 with no allowance for doubtful accounts material enough to be parenthetically disclosed, that would not make it a Gross Receivables figure, and the net value would nevertheless be reported as

<us-gaap:AccountsReceivableNetCurrent unitRef="usd" decimals="-3" contextRef="FY06e">1000</us-gaap:AccountsReceivableNetCurrent>

6.6.15 Nil facts.

The xsi:nil="true" attribute must be used only to convey a value that is different from both “zero” and different from not reporting the fact at all, or to identify a fact detailed only by a link:footnote. For example, a small fragment of an original HTML/ASCII document balance sheet:

Line Item	Amount in thousands (2007)	Amount in thousands (2006)
Commitments and Contingencies	-	-
Preferred Shares (x,xxx shares authorized, none issued)	-	-

This corresponds to these facts:

<us-gaap:CommitmentsAndContingencies unitRef="USD" contextRef="FY07e" xsi:nil="true"/>
 <us-gaap:CommitmentsAndContingencies unitRef="USD" contextRef="FY06e" xsi:nil="true"/>
 <us-gaap:PreferredStockValue unitRef="USD" contextRef="FY07e" xsi:nil="true"/>
 <us-gaap:PreferredStockValue unitRef="USD" contextRef="FY06e" xsi:nil="true"/>

6.6.16 Financial statement notes and schedules.

An instance must contain facts containing each complete footnote and each required schedule (as set forth in Article 12 of Regulation S-X) of the corresponding official HTML/ASCII document, as a single block of text. This “block tagging” requirement that applies to most submission and document types; some document types also require tagging as described below in 6.6.19, 6.6.20, and 6.6.22.

Submission Types	Document Types	Block Tagging Required	Level 2, 3, and 4 Tagging Required
485APOS, 485BPOS, 485BXT, 497, N-1A, N-1A/A	EX-101.INS	No	No
SDR	EX-99.SDR K.INS, EX-99.SDR L.INS	Yes	No
Any other	EX-101.INS	Yes	Yes

Each non-numeric fact must reflect the same information in the corresponding text in the official HTML/ASCII document. Formatting and layout is relevant for elements of a type attribute textBlockItemType or a type equal to or derived by restriction from the type ‘escapedItemType’ in a standard taxonomy schema namespace, but not for any other types. A fact of either type

whose content represents a disclosure, footnote or schedule in the HTML/ASCII documents must contain escaped HTML to preserve that layout or formatting.

For example, the original text as displayed in a browser:

Dividends

Our Board of Directors declared the following dividends:

Declaration Date	Per Share Dividend
(Fiscal year 2008)	\$.09
September 17, 2008	

The original HTML 3.2 format text:

```
<B><I>Dividends</I></B><P>Our Board of Directors declared the following dividends:</P><TABLE>
<TR><TH align=left valign=bottom>Declaration Date</TH><TH/><TH colspan=2 width=70>Per Share
Dividend</TH></TR>
<TR><TD align=left><I>Fiscal year 2008</I></TD></TR>
<TR><TD align=left>September 17, 2008</TD><TD/><TD>$</TD><TD align=right>0.09</TD></TR>
</TABLE>
```

This text must appear in a text block. But if all the layout and formatting are removed and whitespace is normalized, the result is much harder to understand:

Dividends Our Board of Directors declared the following dividends: Declaration Date Per Share Dividend
(Fiscal year 2008) September 17, 2008 \$ 0.09

Therefore the entire text must appear as text containing only well-formed XHTML, in which the tags are balanced and the attributes quoted:

```
<us-gaap:CashFlowSupplementalDisclosuresTextBlock contextRef="FY08Q1">
&lt;b> &lt;i>Dividends &lt;/i>
&lt;/b> &lt;p>Our Board of Directors declared the following dividends: &lt;/p> &lt;table> &lt;tr> &lt;th
align="left" valign="bottom" style="border-bottom: 1pt solid black">Declaration Date&lt;/th> &lt;th/> &lt;th
colspan="2" style="width: 70pt; border-bottom: 1pt solid black">Per Share Dividend &lt;/th> &lt;/tr> &lt;tr>
&lt;td align="left"> &lt;i>(Fiscal year 2008) &lt;/i> &lt;/td> &lt;/tr> &lt;tr> &lt;td align="left">September 17,
2008 &lt;/td> &lt;/td> &lt;td>$ &lt;/td> &lt;td align="right">0.09 &lt;/td> &lt;/tr> &lt;/table>
</us-gaap:CashFlowSupplementalDisclosuresTextBlock>
```

6.6.17 No extra facts.

An instance must not contain facts that do not appear in the corresponding official HTML/ASCII document. The information in interactive data format should not be more or less than the information in the related registration statement or report, with the exception of "dei" facts required by 6.5.20, 6.5.21, 6.5.26 and 6.5.40.

6.6.18 No page headers and footer facts.

Page headers and footers appearing in an official HTML/ASCII document must not appear in any of the facts or link:footnote elements of an instance.

6.6.19 Level 2 (Policy) Tagging

For each significant accounting policy within the accounting policies footnote of the corresponding official HTML/ASCII document, an instance must contain a fact containing the policy as a block of text, if the document type requires “level 2” tagging. Footnotes (equivalently, “Notes to the Financial Statements”) are represented in instances at four levels of detail. Level 1 is always required (Rule 6.6.16 above), and levels 2, 3 and 4 are required based on the phase-in schedule of the registrant.

6.6.20 Level 3 (Table) Tagging

An instance must contain each table within each footnote in the corresponding official HTML/ASCII document as a separate fact block of text, if the document type requires “level 3” tagging. In the example given in Rule 6.6.16 above, the table would be tagged separately:

```
<us-gaap:ScheduleOfDividendsPayableTextBlock contextRef="FY08Q1">
  &lt;table&gt;
    &lt;tr&gt;&lt;th align="left" valign="bottom"&gt;Declaration Date&lt;/th&gt;&lt;th&gt;&lt;th colspan="2"
width="70"&gt;Per Share Dividend&lt;/th&gt;&lt;/tr&gt;
    &lt;tr&gt;&lt;td align="left"&gt;&lt;l&gt;Fiscal year 2008&lt;/l&gt;&lt;/td&gt;&lt;/tr&gt;
    &lt;tr&gt;&lt;td align="left"&gt;September 17, 2008&lt;/td&gt;&lt;td&gt;&lt;td&gt;$&lt;/td&gt;&lt;td
align="right"&gt;0.09&lt;/td&gt;&lt;/tr&gt;
  &lt;/table&gt;
</us-gaap:ScheduleOfDividendsPayableTextBlock >
```

6.6.21 Reserved

6.6.22 Level 4 (Detail) Tagging

An instance must contain separately each monetary value, percentage, and number in each footnote in the corresponding official HTML/ASCII document, as a fact, if the document type requires “level 4” tagging. An instance may also contain dates and narrative disclosures as level 4 facts.

Level 4 facts appear separately from the text blocks of levels 1 to 3; the fact may have non-material changes to the formatting of dates and possibly other facts, for example:

Declaration Date	Per Share Dividend
(Fiscal year 2008)	\$.09
September 17, 2008	

This table contains only two facts, in which “September 17, 2008” becomes “2008-09-17” and “.09” becomes “.09”:

```
<us-gaap:DividendsPayableDeclarationDate contextRef="..."
>2008-09-17</us-gaap:DividendsPayableDeclarationDate>
<us-gaap:DividendsPayablePerShare contextRef="..." unitRef="USD_per_share" decimals="INF"
>.09</us-gaap:DividendsPayablePerShare >
```

For another example, the sentence “Accretion expense declined from 30 thousand to six thousand in 2009” contains two facts

```
<us gaap:AccretionExpense unitRef="USD" decimals="-3" contextRef="FY08" >30000</us
gaap:AccretionExpense>
```



```
<us gaap:AccretionExpense unitRef="USD" decimals="-3" contextRef="FY09" >6000</us
gaap:AccretionExpense>
```

6.6.23 Period types

An element used in numeric facts representing amounts must have an `xbrli:periodType` attribute that is the same as the amounts reported. An element with an `xbrli:periodType` attribute of “instant” has numeric values that are only measurable at a point in time; the value “duration” is used for all other elements, including textual information.

Elements in standard taxonomies are never identified as being a beginning or ending period amount. The same element can represent both a beginning and ending balance, because the underlying financial concept is the same. The differentiating factor is the point in time, which is identified in the instance document and not the taxonomy.

6.6.24 Alignment of numeric facts and element definitions

If an element used in numeric facts representing amounts in one or more periods has a definition, then the scope of that definition must include the material amounts reported for that line item in the corresponding official HTML/ASCII document. An element “has a definition” if there is text in the `link:label` element located as follows: label linkbases listed at <https://www.sec.gov/info/edgar/edgartaxonomies.shtml> contain `link:label` elements with an `xml:lang="en-US"` attribute and an `xlink:role` attribute equal to

‘<http://www.xbrl.org/2003/role/documentation>’. If a `link:labelArc` with an `xlink:to` attribute matching the `xlink:label` attribute of such a `link:label` element, and an `xlink:from` attribute that matches the `xlink:label` attribute of a `link:loc` whose `xlink:href` attribute is an element, then the text of `link:label` is the definition of that element.

The definition is an element’s most important attribute and must be consistent with the financial concept reported. An element should be interpreted by the substantive meaning provided in its definition. Definitions cannot be changed. As important as they are, all definitions have limitations, so preparers should not base their choice of an element simply on minor, immaterial differentials in definitions. Determining whether a definition is consistent with the financial concept requires judgment, and other attributes of the element must be considered.

6.6.25 Numeric element definitions.

An element must not be used in numeric facts representing amounts of a line item in different periods if it has a definition that explicitly excludes one or more of the material amounts in the corresponding official HTML/ASCII document. For example, the definition for element “Other Restructuring Costs” states that it “excludes costs associated with the retirement of a long-lived asset and severance costs associated with established compensation plans.”

6.6.26 Selecting elements based on definitions.

When there is a choice among different elements that have definitions consistent with a set of facts in one or more periods, use the element with the narrowest definition. For example, while in principle, eleven possible word combinations may be derived from “depreciation”, “amortization”, “impairment”, and “depletion”, all eleven might not be included as distinct elements in a standard taxonomy namespace. If the line item being reported consists only of depreciation, then use an element such as `DepreciationAndAmortization`; do not use any of the alternative elements whose definition also includes impairment or depletion.

6.6.27 Selecting elements based on data types.

If there is a choice among different elements whose type attribute is consistent with a set of facts in one or more periods, use the element with the most specific type attribute. For example, a footnote contains the sentence “The assumed discount rate is 2%” or, equivalently, “The assumed discount rate is two percent”. There is a numeric element declared in a standard taxonomy for the value of assumed discount rates, and another element for “assumptions”. Use the numeric assumed discount rate.

Another example is if a fact is a dollar amount and there are some potential elements that are monetary and others that are strings or text blocks, the monetary elements must be used. Similarly, “per share” dollar amounts must be tagged with “per share” elements.

6.6.28 Selecting elements based on authoritative references.

When there is a choice among different elements having distinct link:reference elements in a standard taxonomy, use the element with the most specific reference. Reference linkbases containing link:reference elements do not have to be in the DTS of the instance as submitted but they should be used during the preparation of the instance.

For example, an element with a link:reference to a specific paragraph in a FAS is likely to be a better choice than an element that simply refers to the entire FAS. Determining whether references supporting the definition are consistent with the financial fact requires judgment, and other attributes of the element must be considered.

6.6.29 Precedence when selecting elements.

When choosing the most appropriate element for facts in one or more periods, the element’s link:reference elements take precedence over the xbrli:periodType attribute, which takes precedence over the type attribute, which takes precedence over the element’s documentation string if present, which in turn takes precedence over the standard label string. The calculation, definition, and presentation linkbases published along with standard taxonomies schemas communicate how elements are related to each other. Preparers use linkbases that organize common sets of tags as a starting point. However, these linkbases are to be used as templates by preparers to build their own linkbases to communicate their own intended relationships. Any element in a standard taxonomy schema may be used in an instance that has the schema in its DTS independently of the calculation, definition and presentation linkbases that it might have appeared in. It is the elements standing alone with their authoritative references, period and data type attributes, and standard documentation labels that are definitive.

The name attribute of an element is a mnemonic, not a definition; do not use the name attribute of an element as a definitive indicator of its meaning.

6.6.30 Selecting signs based on the balance attribute.

Invert the sign of a numeric fact whose element has an xbrli:balance value that is inconsistent with the reporting concept being reported. Often, this means entering a negative value into the instance, irrespective of whether that negative value will subsequently be rendered without brackets as a result of applying a negating label.

The value of xbrli:balance (debit or credit) is assigned to monetary elements in a standard taxonomy namespace from the perspective of the income statement and balance sheet. This perspective may be inconsistent with the presentation of the element in the financial statements.

For example, a financial concept in the cash flow statement may be represented by an element that was assigned an xbrli:balance based on the income statement. As a result, the xbrli:balance may be different from preparers' expectations. A different xbrli:balance value for an element must not influence the registrant in deciding whether the element is appropriate for the fact representing a financial concept, and registrants should not create a new element if an element in a standard taxonomy namespace is consistent with the financial concept reported in all respects except xbrli:balance.

Use an element even if its xbrli:balance is not the balance type expected.

6.6.31 The content of a numeric fact never has a scale factor.

Examples:

- The value “twenty thousand” may appear in a numeric fact as any legal decimal representation of 20,000, such as 20000, 20000.0, or 020000. It must not appear as “20”.
- The value “20%” may appear in a numeric fact as any legal decimal representation of .2, such as 0.2, 0.20, 000.2000.
- The value “20%” must not appear in a numeric fact as “20”, “20/100”, “20%” or any variation of the integer “20”.

6.6.32 Selecting the decimals attribute.

The value of the decimals attribute of a fact must correspond to the accuracy of the corresponding amount as reported in the official HTML/ASCII document. The decimals attribute influences how numbers are interpreted in XBRL and any value for the decimals attribute other than the value INF implies rounding or truncation. Use the following table to select the correct value of the decimals attribute for a fact so that it corresponds to the value as presented (most often rounded) on the corresponding official HTML/ASCII document.

Accuracy of the amount as shown in official HTML/ASCII document	Value of decimals attribute
Exact monetary, percentage, basis point or any other amount	INF
Rounded to billions	-9
Rounded to millions	-6
Rounded to thousands	-3
Rounded to units	0
Rounded to cents	2
Rounded to a whole percentage	2
Rounded to basis points	4

Examples:

Fact	Value	Value of decimals attribute
A federal tax rate of (exactly) 46%	.46	INF
An management fee of (exactly) 10 basis points	.001	INF
A (rounded) profit margin of 9.3%	.093	3
A (rounded) change in NAV of 12 basis points	.0012	4

Fact	Value	Value of decimals attribute
A (rounded) inventory "in thousands" of 100	100000	-3
A (rounded) inventory "in thousands" of 100.2	100200	-2

The decimals attribute is not a scale factor.

The decimals attribute is not a formatting code; it does not indicate that the digits in the instance must subsequently be presented to a user in any particular way.

6.6.33 Do not insert digits that do not appear in the official HTML/ASCII document.

For example, rounding can result in calculation inconsistencies. In this example, XBRL validation software will identify a calculation inconsistency:

Line Item	Amount
Earnings per share, Basic	1.30
Income (Loss) from Discontinued Operations, Net of Tax, Per Basic Share	.01
Income (Loss) from Continuing Operations, Per Basic Share	1.28

The decimals attribute must be equal to "2" for all three amounts, because these digits as reported in the financial statement have been rounded. Adjusting the decimals attribute of "2" on the facts to "1" or "3" will not resolve the inconsistency. Adding hidden digits, such as changing .01 to .014 and 1.28 to 1.284, and setting their decimals attributes to "3" may resolve the calculation inconsistency, but the extra digits were not reported in the official HTML/ASCII document.

6.6.34 Do not define or use units that imply a scale factor on a currency.

To express amounts in US Dollars, use only `xbrli:unit` with one `xbrli:measure` element whose content is the QName `iso4217:USD`.

Do not define units such as "thousands of USD", "millions of GBP", or "pence".

6.6.35 Each unit must appear with only one implicit power-of-ten scale factor per instance.

Filers must use units and implicit scale factors consistently throughout submitted instances.

element type attribute	Abbreviation	Meaning
us-types:boeltemType	Boe MBoe MMBoe	Barrel Oil Equivalent (Energy) Thousand Barrels of Oil Equivalent Million Barrels of Oil Equivalent
us-types:volumeltemType	cf Mcf MMcf Fbbls MBbls MMbbls	Cubic Feet of Natural Gas Thousand Cubic Feet of Natural Gas Million Cubic Feet of Natural Gas Barrels of Oil at 60 degrees F Thousands of Barrels of Oil Millions of Barrels of Oil

For example, these facts have a unitRef attribute of different scales in a single instance and only one such scale may be used:

```
<us-gaap:ProvedDevelopedReservesVolume ... unitRef="MMbbl">3880</...>
<us-gaap:ProvedDevelopedReservesVolume ... unitRef="Mbbbl">2200000</...>
<us-gaap:ProvedUndevelopedReservesVolume ... unitRef="Fbbbl">42000000</...>
```

By contrast, units such as “lb” and “oz” are distinct, not related by a power of ten, and may be used, for example, as “hundreds of pounds” and “ounces” in a single instance.

Define and use other xbrli:unit elements (usually needed for performance metrics such as “stores” or “customers”) consistently within an instance and in subsequent submissions.

To define a custom unit, the xbrli:measure element requires the use of a namespace prefix. For example, a company with schema namespace prefix “abc” would declare Thousands of Barrels of Oil as follows:

```
<xbrli:unit id="MBbbl">
<xbrli:measure>abc:MBbbl</xbrli:measure>
</xbrli:unit>
```

6.6.36 The Date String item type.

If a fact whose element type attribute equals ‘dateStringItemType’ in a standard namespace or a restriction of it refers to a specific date, then its value must be a valid XML Schema date. The type dateStringItemType is used for elements for which the fact values are usually a specific date, but may be accompanied by text that modifies or qualifies the date.

If the date is known and specific then it must appear in the form “YYYY-MM-DD”. Otherwise, string values such as “After 2007” or “before March 31, 2009” are acceptable.

For example, “Tuesday, October 19, 2010” violates this rule; it must be 2010-10-19.

6.6.37 The Duration String item type.

If a fact whose element type attribute is ‘durationStringItemType’ in a standard namespace or a restriction of it refers to a specific length of time, then its value must be a valid XML Schema duration. The type durationStringItemType is used for elements for which the fact values are most often a specific duration, but are sometimes accompanied by text that modifies or qualifies that duration.

If the duration is known and specific then it must appear in forms such as “P1D” “P6M” or “P5Y”. Otherwise, a string value such as “From 6 months to 5 years” is valid.

For example, the text “One year and eight months” violates this rule; it must be “P1Y8M”.

6.6.38 Reserved.

6.6.39 Page level footnotes of the face financials.

Text that is shown in the official HTML/ASCII document at the bottom of a page on the face of the financials preceded by a superscript must appear in the corresponding instance as the text of a link:footnote element. The content of link:footnote should not contain the superscript symbol or number originally appearing in the official HTML/ASCII document.

Financial statement “footnotes” (Notes to the Financial Statements) do not appear in a link:footnote.

6.6.40 Distinct page level footnote facts.

Distinct texts that are shown in the official HTML/ASCII document at the bottom of a page on the face of the financials preceded by distinct superscripts must appear in the corresponding instance as the text of distinct link:footnote elements.

6.7 Syntax of Company Extension Schemas

This section defines rules governing the syntax restrictions on attached schemas. A valid Interactive Data schema is a valid XBRL 2.1 schema, but not all valid XBRL 2.1 schemas are valid Interactive Data schemas.

6.7.1 The `xsd:schema` must not have an `xsd:include` element.

This rule does not apply to schemas in a standard taxonomy.

There is one namespace per `xsd:schema` element and therefore no “chameleon schemas”, and additional XBRL 2.1 syntax restrictions apply.

The `elementFormDefault` attribute is usually “qualified” and the `attributeFormDefault` attribute usually “unqualified”, but there are no formal restrictions on the values of these attributes and no formal restrictions on the `formDefault` attribute.

6.7.2 Standard taxonomy schema locations.

If an `xsd:import` element has a namespace attribute equal to a standard taxonomy schema, then its `schemaLocation` attribute must be the standard taxonomy assigned to that namespace.

6.7.3 Custom taxonomy namespace authority parts.

The authority part of an `xsd:schema targetNamespace` attribute must not equal the authority part of a `targetNamespace` attribute of any standard taxonomy schema. The authority part of a valid URI is the part after ‘//’ and before the subsequent ‘/’ character.

6.7.4 The `targetNamespace` attribute must match `http://{authority}/{versionDate}`.

The company-specific schema has a unique `targetNamespace` attribute name for each schema. Namespaces are not to be confused with external references even though they may appear to have very similar formats. An external reference describes the location of a particular file with the intent of accessing the contents of that file. A namespace, on the other hand, is a name that identifies elements that belong to a particular markup vocabulary. However, since they function very differently, restrictions that are placed on external references do not apply to namespaces. Since a particular instance document is expected to reference multiple vocabularies, namespaces provide a convention by which each vocabulary is uniquely identified. This avoids problems of recognition and collision of similarly named elements from different vocabularies appearing in XBRL documents.

6.7.5 Authority parts of custom taxonomy namespaces.

The `targetNamespace` attribute must be a valid URI with an {authority} that is either a domain name controlled by the publisher of the schema, a domain name controlled by the registrant, or if neither exists, then a mnemonic name for the registrant such as its ticker symbol. From time to time, regulatory, accounting or other authorities may publish schemas to support new reporting rules. Until such schemas are added to the standard taxonomy lists, registrants may provide a

copy of such a schema in their submission. In such a case, the targetNamespace attribute will contain an {authority} different from the registrant.

The registrant must own or control the authority name; for example, “example.com” could only be used by Example Inc. itself.

The {authority} used in the targetNamespace attribute must match the {authority} in the URI of any role or arc role declarations.

6.7.6 Restrictions on custom taxonomy namespaces.

The targetNamespace attribute must be a valid URI with a {versionDate} that identifies the release date of the schema. Examples:

- <http://abcinc.com/20080331>, <http://www.definc.us/20081231>

Counterexamples:

- <http://sec.gov/abc/20080331>, <http://abcinc.com/2009>

The targetNamespace attribute of the schema is different than the scheme attribute in the xbrli:identifier element; the scheme attribute refers to the SEC and the targetNamespace attribute does not.

6.7.7 Recommended custom schema namespace prefix.

Element xsd:schema must bind a Recommended Namespace Prefix for the targetNamespace attribute that does not contain the underscore character. A mnemonic such as the ticker symbol of the company in lowercase is suitable.

For example, `<xsd:schema xmlns:abc='http://abcinc.com/20080331' ...>`

6.7.8 Element xsd:schema must not contain any occurrences of “embedded” linkbases.

Elements in the “link” namespace having a type attribute equal to ‘extended’, ‘arc’, ‘resource’ or ‘locator’ must not occur anywhere in an xsd:schema.

6.7.9 Restrictions on the roleURI attribute.

The roleURI attribute of a link:roleType element must begin with the same {scheme} and {authority} as the targetNamespace attribute. For example, in an xsd:schema with a targetNamespace attribute equal to

‘http://abcinc.com/20090229’, the string ‘http://abcinc.com/’ must start the roleURI attribute value of any link:roleType.

The roleURI attribute should be considered permanent, to be used in future submissions.

In a link:roleType declaration the roleURI attribute should end with “/role/” and a mnemonic name.

For example

```
<link:roleType @roleURI="http://abcinc.com/role/StatementOfIncome"> ...</link:roleType>
```

6.7.10 Unique roleURI attribute values.

A DTS must not contain more than one link:roleType element with equal values of the roleURI attribute. Duplicate role declarations in a single schema file are forbidden by XBRL 2.1; this forbids duplicate role declarations in the instance DTS as a whole.

6.7.11 Restrictions on the roleType usedOn element.

A link:roleType declaration with link:usedOn containing link:presentationLink, link:definitionLink or link:calculationLink must also have a link:usedOn for the other two. This rule is relevant to three of the linkbase elements whose type attribute is fixed at 'extended' as shown in the table below.

type attribute	QName in link:usedOn	Declared by
extended	link:calculationLink	link:roleType
extended	link:definitionLink	link:roleType
extended	link:presentationLink	link:roleType

6.7.12 Content of the roleType definition element.

A link:roleType element must contain a link:definition child element whose content will communicate the title of the section, the level of facts in the instance that a presentation relationship in the base set of that role would display, and sort alphanumerically into the order that sections appear in the official HTML/ASCII document. The link:roleType link:definition text must match the following pattern:

{SortCode} - {Type} - {Title}

The {Type} must be one of the words 'Disclosure', 'Document', 'Schedule' or 'Statement'.

The meaning of the base set appears in {Title}.

{SortCode} is used only to sort base sets for display. The sort code is sorted alphanumerically, not numerically, so "10" would appear before "2". Filers must choose a scheme for their sort code and declare separate role types so as to achieve the following:

1. Each Statement must appear in at least one base set, in the order the statement appeared in the official HTML/ASCII document.
2. If the presentation relationships of more than one base set contains the facts of a Statement (to achieve a layout effect, such as a set of rows, followed by a table with a dimension axis on the vertical, followed by another set of rows) then the {SortCode} of that base set must sort in the order that the rows of the Statement will be displayed.
3. A Statement that contains parenthetical disclosures on one or more rows must have a base set immediately following that of the Statement, where all facts in its parenthetical disclosures appear in presentation relationships.
4. All base sets containing the contents of Footnotes (level 1) should appear after base sets containing the contents of Statements (level 1).
5. A Text Block for each Footnote must appear in at least one presentation relationship in a base set.
6. Each base set for a "Footnote as a Text Block" presentation link must contain one presentation relationship whose target is a Text Block.

7. Base sets with presentation relationships for a Footnote tagged at level 2 should appear after all base sets tagged at level 1.
8. A base set with presentation relationships for a Footnote tagged at level 3 should appear after all base sets tagged at level 2.
9. A base set with presentation relationships for a Footnote tagged at level 4 should appear after all base sets tagged at level 3.
10. Base sets with {Type} 'Document' should appear before all other base sets as "Cover" pages with level "0".

The {Title} is the text that follows “ - ” in the link:definition. The text should distinguish to a human reader what each separate relationship group contains. The table below shows an example in which the filer has simply used a two-digit sequence number.

The {Title} must not contain scale or units (such as “in millions of US dollars except per share data”) text.

Note: The link:definition MUST NOT have leading or trailing XML whitespace or newlines.

Type of Facts in Presentation Links:

Example link:definition Text	Each Footnote as a Text Block	Each Accounting Policy as a Text Block	Each Table in a Footnote as a Text Block	Individual Values or Narratives
01 – Statement – Statement of Income				Yes
02 – Statement – Balance Sheet				Yes
03 – Statement – Balance Sheet (Parenthetical)				Yes
04 – Statement – Cash Flows				Yes
05 – Statement – Changes in Equity				Yes
06 – Statement – Comprehensive Income				Yes
07 – Disclosure – Accounting Policies	Yes			
08 – Disclosure – Inventories	Yes			
09 – Disclosure – Earnings per Share	Yes			
10 – Disclosure – Unearned Revenue	Yes			
11 – Disclosure – Equity	Yes			
12 – Disclosure – Accounting Policies, by Policy		Yes		
13 – Disclosure – Inventories (Tables)			Yes	
14 – Disclosure – Unearned Revenue (Tables)			Yes	
15 – Disclosure – Equity, Share Repurchases (Table)			Yes	
16 – Disclosure – Equity, Dividends (Table)			Yes	
17 – Disclosure – Inventories (Detail)				Yes
18 – Disclosure – Unearned, by Component (Detail)				Yes
19 – Disclosure – Unearned, by Segment (Detail)				Yes
20 – Disclosure – Equity, Share Repurchases (Detail)				Yes
21 – Disclosure – Equity, Dividends (Detail)				Yes

Defining roles is important, because the renderer displays *all* the facts in an instance if they appear in an effective presentation relationship, and displays facts *only* when they appear in an effective presentation relationship in a base set of the role. Note that if the link:definition text matches “{SortCode} – {Type} – {Title}” then the renderer shows only the {Title} part, having removed any layout qualifiers (see 6.24.16) from the text.

6.7.13 Restrictions on the arcroleURI attribute.

The arcroleURI attribute of a link:arcroleType element must begin with the same {scheme} and {authority} parts as the targetNamespace attribute. For example, in a schema with a targetNamespace attribute http://abcinc.com/20090229, the string http://abcinc.com/ must start the arcroleURI attribute of any link:arcroleType.

The arcroleURI attribute should be considered permanent, to be used in future submissions.

In a link:arcroleType declaration the arcroleURI attribute should end with “/arcrole/” followed by a mnemonic name.

For example,

```
<link:arcroleType @arcroleURI="http://abcinc.com/arcrole/SpecialRelationship"> ...</link:arcroleType>
```

6.7.14 Unique arcroleURI attribute values.

A DTS must not contain more than one link:arcroleType element with equal values of the arcroleURI attribute.

6.7.15 A link:arcroleType element must have a nonempty link:definition.

The link:arcroleType link:definition text should explain the purpose of the arc role.

The content of link:usedOn is the QName of an arc-type element; however, note that there are additional rules that restrict what may be used as the value of the xlink:arcrole attribute in instances, schemas and linkbases.

6.7.16 Unique element names.

The name attribute of an xsd:element must not equal any xsd:element name attribute in a standard taxonomy schema that appears in the same instance DTS. It is not necessary to compare the @name attribute to all element declarations in all standard taxonomy schemas. Only those schemas that are present in the DTS of a specific instance being validated are relevant.

6.7.17 Restrictions on the element id attribute.

The id attribute of an xsd:element must consist of the Recommended Namespace Prefix of the element namespace, followed by one underscore, followed only by its name attribute. When constructing a name attribute, a common convention is to capitalize the first letter and use mainly lowercase characters, using capitalization only to indicate a natural word break, for example “CurrentPortionOfLongTermDebt”. Note the first character of a name attribute must not be underscore, and if the name attribute is originally based on a label and in a subsequent version of the schema, the label changes, the name attribute must not be changed merely to maintain agreement.

6.7.18 The nillable attribute value of an xsd:element must equal ‘true’.

6.7.19 No tuple type elements.

The xsd:element substitutionGroup attribute must not be a member of a substitution group with head ‘xbrli:tuple’.

6.7.20 An xsd:element must not have an xbrldt:typedDomainRef attribute.

6.7.21 Restrictions on abstract elements.

If the abstract attribute of xsd:element is 'true', then the xbrli:periodType attribute must be 'duration'.

6.7.22 Reserved.

6.7.23 Restriction on dimension item element names.

The xsd:element substitutionGroup attribute must equal 'xbrldt:dimensionItem' if and only if the name attribute ends with 'Axis'. An element is defined to be an "Axis" if and only if its substitutionGroup attribute equals 'xbrldt:dimensionItem'.

6.7.24 Restriction on hypercube item element names.

The xsd:element name attribute must end with 'Table' if and only if substitutionGroup attribute equals 'xbrldt:hypercubeItem'. An element is defined to be a "Table" if and only if its substitutionGroup attribute equals 'xbrldt:hypercubeItem'.

6.7.25 Restriction on substitutionGroup attribute.

If the xsd:element substitutionGroup attribute is not equal to 'xbrldt:dimensionItem' or equal to 'xbrldt:hypercubeItem' then it must equal 'xbrli:item'.

6.7.26 If an element name ends with 'LineItems' then the abstract attribute must be 'true'.

6.7.27 Restriction on domain item type element names.

The xsd:element name attribute must end with 'Domain' or 'Member' if and only if the type attribute equals or is derived from 'domainItemType' in a standard taxonomy schema target namespace. A "domain member" is defined as an element meeting this condition.

6.7.28 Restriction on domain item type periodType attribute.

If xsd:element type attribute equals or is derived from 'domainItemType' in a standard taxonomy schema target namespace then the xbrli:periodType attribute must equal 'duration'.

6.7.29 Restriction on element name length.

The content of an xsd:element, xsd:complexType, or xsd:simpleType name attribute in UTF-8 must not exceed 200 bytes in length. This length restriction applies to bytes in UTF-8 encoding, not characters. For example, an element name of 101 repetitions of character "Ā" (Unicode x0100) violates this criterion because its UTF-8 encoding has 202 bytes.

By contrast, the six-byte EDGAR accepted encoding of that same character, "Ā" when read and stored as an XML text node in UTF-8 encoding, would be only two bytes. Therefore up to 100 repetitions of that sequence would be allowed.

Names of reasonable length limited to US-ASCII characters avoid such complexities entirely and are advisable.

6.7.30 Restriction on URI length.

The content of a targetnamespace, roleURI or arcroleURI attribute in UTF-8 must not exceed 255 bytes in length. This length restriction applies to bytes in UTF-8 encoding, not characters. See 6.7.29 above.

6.7.31 The xsd:element type must not be equal to or derived from xbrli:fractionItemType.

6.7.32 Declarations requiring period type 'duration'.

An element declaration having a non-numeric base type, abstract not 'true', and not derived from domainItemType must have the value 'duration' for xbrli:periodType. In combination with 6.7.21 and 6.7.28 this means that only element declarations having a numeric base type may have period type 'instant'.

6.8 Content of Schemas

This section describes the content required in schemas.

6.8.1 Changes in declarations from period to period.

A schema that changes any xsd:element or type declarations or changes any relationships in its DTS from an earlier version of itself in such a way as to invalidate earlier instances must use only the {versionDate} portion of its targetNamespace attribute to identify the new version. From submission to submission, a company extension's schema and linkbases almost always change, for example, to accommodate changes in the parenthetical portions of element labels. Change the {versionDate} portion of the schema's targetNamespace with each submission to ensure that submissions validate independently and maintain their correspondence to Original HTML/ASCII Documents.

For example, a 2nd quarter Form 10-Q submission contains a schema with a targetNamespace attribute <http://abcinc.com/20091231>. The schema for the 3rd quarter Form 10-Q changes one element from the xbrli:balance attribute of 'credit' to 'debit'. Its targetNamespace attribute changes to <http://abcinc.com/20100331>.

6.8.2 Reserved.

6.8.3 No duplicative custom role and arc role type declarations.

Do not define link:arcroleType (or link:roleType for a resource-type element) that means the same as arc roles or resource roles that are already defined in the XBRL 2.1 specification or in a standard taxonomy. The table below shows declarations that are technically possible, although any use of the defined role or arc role would be subject to all other rule restrictions.

type attribute	QName in link:usedOn	Declared by
extended	link:labelLink	link:roleType
extended	link:referenceLink	link:roleType
resource	link:label	link:roleType
resource	link:footnote	link:roleType
resource	link:reference	link:roleType
arc	link:calculationArc	link:arcroleType
arc	link:definitionArc	link:arcroleType
arc	link:labelArc	link:arcroleType
arc	link:presentationArc	link:arcroleType
arc	link:referenceArc	link:arcroleType

6.8.4 No duplicative custom element declarations.

Wherever possible, registrants should assign a standard and other labels for an element defined in a standard taxonomy schema in preference to declaring a new element in a company schema. A standard label is a link:label element with an xlink:role attribute equal to

‘http://www.xbrl.org/2003/role/label’; other labels use other xlink:role attribute values.

Defining a company-specific element should be done only under specific circumstances discussed in items 6.8.6 – 6.8.23 with respect to each type of element.

When the disclosure intended by the preparer is effectively a synonym for a disclosure represented by the element’s authoritative references and documentation label, registrants should assign a label to the standard element, in preference to defining a custom element.

For example, there may be an element “GrossProfit.” In a standard namespace. The standard namespace does not include “Gross margin,” because this is defined the same as “Gross Profit”: both are used to mean “excess of revenues over the cost of revenues.” A registrant presenting “Gross margin” in a statement should not define a custom “GrossMargin” element; instead, use the “GrossProfit” element and define the label for this element as “Gross margin”.

6.8.5 Reserved.

6.8.6 Use company- and period-specific names only for domain members.

Do not include company-specific or period-specific information in an xsd:element name attribute for elements having other than type ‘domainItem Type’. Elements with other item types, including (but not limited to) monetary, percent, integer, shares, per share, string, or text block item types, should not include company-specific or period-specific information in the element name. Domain members may include company-specific or period-specific information in the element name.

For example, filers should not create a monetary element with the name “AcquisitionOfDefCo” or “FourthQuarterAdjustment”. However, they may create a domain member with the name “AbcSegmentMember”.

6.8.7 Reserved.

6.8.8 Custom abstract elements.

Declare an xsd:element with an abstract attribute equal to ‘true’ if an appropriate abstract element does not exist, and use the presentation linkbase to have facts rendered sequentially. An abstract element cannot appear as a fact in an instance. Therefore, unless the element appears as a domain member in an instance context, none of the element selection considerations described above in the “Instance Semantics” section 6.6 (documentation labels, references, similarity to other elements, and so on) apply to the determination of whether an existing abstract element is “appropriate”. Filers may create abstract elements at their discretion to arrange presentation relationships.

6.8.9 Custom monetary elements.

Declare an xsd:element with a type attribute equal to ‘xbrli:monetaryItemType’ if the standard taxonomy schema contains only monetary type elements that, in the judgment of the registrant,

are too broadly defined for a given line item. For example, an original HTML/ASCII document may contain a financial statement line item that encompasses a significant fraction of a nearby line item. For example, an original HTML/ASCII document contains these lines:

Line Item	Amount
Accounts payable	7,324
Securities lending payable	1,274
Other liabilities, current	2,362

The us-gaap namespace does not have an element whose definition is sufficiently narrow enough to encompass “Securities Lending Payable” alone while leaving the other line items shown indistinguishable. Meeting the accounting disclosure requirements justifies defining an xsd:element with a name attribute SecuritiesLendingPayable.

For example, an original HTML/ASCII document may contain a financial statement line item that combines two elements defined in a standard taxonomy schema. For example, an original HTML/ASCII document contains these lines:

Line Item	Amount
Goodwill	9,845
Prepaid pension and postretirement benefits	8,731
Other assets, noncurrent	872

The us-gaap namespace has separate elements PrepaidExpenseOtherNoncurrent and PrepaidPensionCosts, but the registrant judges the former to be too narrow and the latter to be too broad for the line item. Meeting the accounting disclosure requirements justifies defining the xsd:element with a name attribute PrepaidPensionAndPostretirementBenefits.

6.8.10 Do not declare different elements for different values of the same underlying line item.

For example, a line item that appears with different values (even in the same period) is still only a single xsd:element. For example, a cash flow statement may have a line item “Reclassification of proceeds from Operations to Investments” that appears in “Cash flow from Operations” as (10) and under “Cash flow from Investments” as 10; it is still the same fact and the same xsd:element. More generally, a line item such as “Net Income” is the same as the line item for “Net Loss”, since such an xsd:element may appear in facts with different values, positive, negative, or zero in different periods.

6.8.11 Custom elements requiring a balance attribute.

An xsd:element with a type attribute equal to ‘xbrli:monetaryItemType’ must have an xbrli:balance attribute if it appears on a statement of income or balance sheet. An xsd:element “appears on a statement of income or balance sheet” if the xsd:element is the target of a presentation relationship in a link:presentationLink with an xlink:role attribute with a link:definition that contains phrases such as “income,” “balance,” “financial position.”

The xbrli:balance attribute values ‘credit’ and ‘debit’ serve to disambiguate the meaning of a positive (or negative) fact value in an instance. In combination with the xbrli:periodType

attribute, a user can identify all monetary facts on the income statement and balance sheet as assets, liabilities, income or expenses no matter what their name attribute.

6.8.12 Custom monetary elements requiring period type 'instant'.

An `xsd:element` with a `type` attribute equal to `'xbrli:monetaryItemType'` has an `xbrli:periodType` attribute equal to `'instant'` if and only if it represents beginning and end of period balances, as distinct to balances defined over a period of time. Otherwise, use an `xbrli:periodType` attribute equal to `'duration'`. Do not define separate elements to represent the beginning and ending balances.

Facts in the instance have a `contextRef` attribute that contain the calendar date for the value. Therefore, the same fact represents the ending balance of a prior period and beginning balance of the next period.

6.8.13 Custom monetary elements requiring period type 'duration'.

An `xsd:element` with a `type` attribute equal to `'xbrli:monetaryItemType'` that represents an adjustment must have a `xbrli:periodType` attribute equal to `'duration'`. By convention, a balance adjustment fact has a `contextRef` attribute that refers to the period prior to the end-of-period balance that it applies to.

6.8.14 Custom percent item type elements.

A ratio of values that would have the same `unitRef` attribute must be declared as an `xsd:element` with a `type` attribute equal to or derived from `'percentItemType'` in a standard taxonomy schema target namespace, even though its value is not scaled by 100. Only if both the numerator and denominator would have an `xbrli:periodType` attribute of `'instant'` could this element's `xbrli:periodType` attribute also be `'instant'`.

Although a concept such as "Change in Revenues" is conventionally rendered scaled by 100 and followed by the "%" symbol, a fact of this `xsd:element` in an instance will have a `unitRef` attribute referencing a unit having no division element and exactly one measure `xbrli:pure` and a value such as .20 or -.03. To reduce ambiguity in the meaning of such a concept, use a name attribute that expresses the true ratio, using the word 'Over'.

For example, an element "Change in Revenues from the Period One Year Earlier, Over Revenues from the Period One Year Earlier" is awkwardly named, but explicit and applicable to both quarterly and annual reports. As explained below, the label linkbase can contain a more compact "terse" label and the presentation linkbase can indicate where the terse label should be used.

6.8.15 Custom pure item type elements.

A ratio of values for which its facts would have an ISO 4217 currency in the numerator and a different ISO 4217 currency in the denominator must be declared as an `xsd:element` with a `type` attribute equal to or derived from `xbrli:pureItemType`. For example, an "Exchange Rate" concept is a "pure" number, being a ratio of two monetary values and the `unitRef` attribute of its underlying values having different currencies.

6.8.16 Custom date item type elements.

If facts in an instance are dates, but no `xsd:element` in a standard taxonomy with a `type` attribute equal to `'xbrli:dateItemType'` is appropriate, declare an `xsd:element` with a `type` attribute equal to

‘xbrli:dateItemType’. For example, an original HTML/ASCII document shows a table of contracts with values and maturity dates. There is no appropriate element in any standard taxonomy schema for the date, so declare an element ‘ContractMaturityDate’.

6.8.17 Custom string item type elements.

If facts in an instance are a mixture of text, date, numbers or other values, and no xsd:element in a standard taxonomy with a type attribute equal to ‘xbrli:stringItemType’ is appropriate, then declare an xsd:element with a type attribute equal to ‘xbrli:stringItemType’. For example, an original HTML/ASCII document shows a table of contracts with a short text description of the project and its terms. Declare an element ‘ProjectTermsText’.

6.8.18 Custom domain item type elements.

If no standard taxonomy schema contains domain member elements specific enough to distinguish between facts needing distinct values of ‘xbrldi:explicitMember’, then declare an xsd:element with a type attribute equal to or derived from ‘domainItemType’ in a standard taxonomy schema target namespace. Standard taxonomy schemas define domain members for every axis, one of those elements being the default for that axis. Registrants define elements with mnemonic name attribute values to organize facts into contexts.

Examples:

Standard Domain Element	Examples of Custom Domain Members
EntityDomain (default of LegalEntityAxis) (see also 6.6.5)	Separately reporting subsidiaries (DefCoMember, GhiCoMember) EDGAR Series Identifier (S000099999, S000999999)
ConsolidatedEntitiesDomain (default of ConsolidationEntitiesAxis)	Subsidiary1Member, Subsidiary2Member
ClassOfStockDomain (default of StatementClassOfStockAxis) (see also 6.6.9)	EDGAR Class Identifier (C000099999, C000999999) Share Class (AMember, BMember)
ProductOrServiceDomain (default of ProductOrServiceAxis)	Product Name (Drug1Member, Drug2Member)
SegmentGeographicalDomain (default of SegmentGeographicalAxis)	Regions (NewEnglandMember, CaribbeanMember)

Because a default domain member element never appears in a valid instance, its name is of little interest other than to indicate which axis it belongs to.

6.8.19 Do not declare 'total' domain member elements.

Do not declare an xsd:element with a type attribute equal to or derived from ‘domainItemType’ in a standard taxonomy schema target namespace as an explicit “total” domain member. Elements with a name attribute ending in Domain, as the default member of an Axis, already serve as a total.

6.8.20 Custom axis elements.

For value and narrative facts, declare an xsd:element with a substitutionGroup attribute equal to ‘xbrldt:dimensionItem’ if a standard taxonomy schema contains no Axis element for a reporting

axis appearing in the original HTML/ASCII document. A narrative fact is a non-numeric fact with a base type derived from 'xsd:string'.

If the axes of an existing table are insufficient to capture a complex disclosure, a preparer may need to add an axis element. For example, suppose there is a table for "Long-lived Assets Held-for-sale by Asset Type" that has an axis for the asset type. If a registrant needs to disaggregate an asset type (property, for example) according to its degree of distress, declare an `xsd:element` name attribute such as 'DistressDegreeAxis'.

6.8.21 Custom table elements.

Declare an `xsd:element` with a `substitutionGroup` attribute equal to 'xbrl:hypercubeItem' if a standard taxonomy schema contains no Table elements appropriate to the reporting axis needed. Only create a new table to meet a reporting goal that cannot be met by modifying an existing table's relationships.

6.8.22 Reserved.

6.8.23 Custom text block item type elements.

Define an `xsd:element` with a `type` attribute equal to 'textBlockItemType' or a type equal to or derived by restriction from the type 'escapedItemType' in a standard taxonomy schema namespace if the standard taxonomy schema contains only text block type elements that, in the judgment of the registrant, are too broadly defined for a footnote or table. Creating a new non-numeric, non-abstract element should always be a last resort. A non-numeric element in a company extension has the same reusability drawbacks of a numeric element, without the benefit of relationships such as calculation relationships to indicate how it relates to other existing elements.

For example, a text block can be created to tag text that covers the subject matter of two distinct, existing text blocks. For example, if a company discusses the sale of shares in consolidated subsidiaries and the sale of shares in equity method investees in the same footnote, then neither `us-gaap:EquityMethodInvestmentsTextBlock` nor `us-gaap:StockholdersEquityNoteDisclosureTextBlock` would be appropriate to cover all of the disclosures in that footnote. Accordingly, the registrant should declare an `xsd:element` with a `name` attribute such as 'IssuanceOfStockByEquityMethodInvesteesTextBlock' for the footnote.

6.9 Syntax of all Linkbases

This section defines rules governing syntax of linkbases. A valid Interactive Data linkbase is a valid XBRL 2.1 linkbase, but not all valid XBRL 2.1 linkbases are valid Interactive Data linkbases.

6.9.1 A link:linkbase must be XML Linking Language (XLink) 1.0 valid.

Two syntax rules about elements with a `type` attribute equal to 'extended' ("extended-type links") require emphasis:

1. The scope of the `xlink:label` attribute is only its enclosing extended-type link. The `xlink:label` attribute does not function like the XML `id` attribute, which must be unique in an entire document. Therefore, the same value of the `xlink:label` attribute may appear on any number of elements, so long as those elements appear in different extended-type links.

2. Two elements with a type attribute equal to 'arc' ("arcs") in the same extended-type link must not have the same values for the xlink:from attribute and the xlink:to attribute, even if their xlink:arcrole attribute is different. However, this prohibition against duplicate arcs is unrelated to any XBRL or EDGAR level syntax rule that forbids equivalent arcs (XBRL 2.1 section 3.5.3.9.7.4)

Reliance on these two rules of XLink syntax yields the meaning of the term 'base set' of relationships as defined in XBRL 2.1 and used elsewhere in this manual.

6.9.2 Effective relationships.

An effective relationship exists between a target and source element when there is an element with an xlink:type attribute of 'arc' and a use attribute of "optional" that has a higher value of the priority attribute than any equivalent relationship in its base set. This is a definition of the term 'effective relationship' for the scope of this manual. Equivalent relationships are defined by XBRL 2.1.

6.9.3 A link:linkbase in a submission must have no ineffectual relationships.

Elements of the xlink:type='arc' attribute are ineffectual when there is an equivalent relationship with the same or higher priority or when it overrides an unprohibited arc. An arc-type element with use="prohibited" always takes precedence over arcs with use="optional" when their priorities are the same.

Examples:

- An arc with use="prohibited" in a submission may prohibit an arc in a standard taxonomy.
- An arc with use="prohibited" with no equivalent arc is ineffectual.
- An arc with use="prohibited" with the priority attribute less than the priority attribute of an effective arc is ineffectual.
- An arc with use="optional" in a submission that is prohibited is ineffectual.
- An arc with use="optional" in a submission may not be equivalent to an unprohibited arc in a standard taxonomy.

6.9.4 Extended link role attributes.

The xlink:role attribute of an element with a type='extended' attribute or a type='resource' attribute must be present and must not be empty.

6.9.5 Restrictions on resource role attributes.

The xlink:role attribute of an element with an xlink:type attribute of 'resource' must be present and must be defined in XBRL 2.1 or a standard taxonomy. Custom roles are acceptable on extended links but not on resources.

6.9.6 Restrictions on arcroleRef attributes.

The text preceding a sharp sign '#' in an xlink:href attribute of link:arcroleRef must be a standard taxonomy. No custom arc roles.

6.9.7 All extended link elements in a linkbase must have the same namespace and local name.

An element is an extended link if its type attribute is equal to ‘extended’.

A single linkbase cannot mix different kinds of link elements.

6.9.8 Duplicate relationships, defined.

Arcs that are defined as equivalent in XBRL 2.1 and having the same value for the use attribute are duplicate relationships. This is a definition of “duplicate relationship” for the scope of this manual.

6.9.9 The value of the priority attribute must be strictly less than 10.

Standard taxonomy linkbases may prevent specific relationships from being prohibited.

6.10 Syntax of Label Linkbases

This section defines rules governing the syntax restrictions on label linkbases. A valid Interactive Data label linkbase is a valid XBRL 2.1 label linkbase, but not all valid XBRL 2.1 label linkbases are valid Interactive Data label linkbases.

6.10.1 Elements requiring English standard labels.

An element used in a fact or `xbrldi:explicitMember` in an instance must have an English standard label in the DTS of that instance. An element “has an English standard label” in a DTS if there is an effective relationship with the defining `xsd:element` source, an `xlink:arcrole` attribute equal to “`http://www.xbrl.org/2003/role/concept-label`” and target `link:label` with an `xml:lang` attribute equal to “en-US” (this comparison is case sensitive) and an `xlink:role` attribute equal to “`http://www.xbrl.org/2003/role/label`”.

This rule is particularly relevant to elements declared in the company schema, because a document that could contain a `link:label` for a company-specific element would never appear in a standard taxonomy, and therefore has to be in a label linkbase in the same submission. It is not necessary for the DTS to have a standard label for all elements declared in the DTS.

6.10.2 Role and language attributes of a label define a label not to be duplicated.

An element used in a fact or `xbrldi:explicitMember` in an instance must have at most one label for any combination of the `xlink:role` attribute and the `xml:lang` attribute in the DTS of that instance. An element “has a label” in a DTS if there is an effective relationship with the defining `xsd:element` source, an `xlink:arcrole` attribute equal to ‘`http://www.xbrl.org/2003/role/label`’ and target `link:label` with an `xml:lang` attribute equal to ‘en-US’.

This rule is particularly relevant to elements declared in the company schema, because a document that could contain a `link:label` for a company-specific element would never appear in a standard taxonomy, and therefore has to be in a label linkbase in the same submission.

6.10.3 US English labels are required.

If an element used in an instance is assigned a label in the DTS whose `xml:lang` attribute is not “en-US”, then the DTS must also contain a `link:label` for the same element and all other attributes with an `xml:lang` attribute equal to “en-US”. Non-US English labels may appear for elements in the schema, but they must be translated into US English if they are used in an instance.

6.10.4 Standard labels are unique.

The DTS of an instance must have no distinct elements having the same English standard label (`xml:lang` attribute equal to “en-US”). Users will most often see the English standard label of a concept, and it decreases clarity if distinct elements have the same label. Note that there is no restriction on elements having duplicated labels for other values of an `xlink:role` attribute.

6.10.5 A label linkbase must not have a definition for an element defined in a standard taxonomy.

The rule prevents an extension linkbase from removing from, or adding to, the documentation of an element in a standard taxonomy schema. Even a standard element having no published definition must not have a definition added by an extension. This is important for comparability and to prevent contradictory definitions for elements.

6.10.6 Restrictions on label lengths.

The ASCII text of `link:label` must be a string of fewer than 511 characters with no consecutive XML whitespace characters and no occurrences of ‘<’ unless its `xlink:role` attribute is ‘<http://www.xbrl.org/2003/label/documentation>’. ASCII character 60 (decimal) and the following ASCII sequences are prohibited in the content of labels other than documentation labels: `<`; `<`; `<`.

6.10.7 Reserved.

6.10.8 The text of `link:label` must not have leading or trailing XML whitespace.

All labels may contain the XML whitespace characters ASCII 9, 10, 13 and 32 anywhere except at their start or end. Documentation labels may contain sequences of extra whitespace; all other labels must be normalized; see 6.10.6.

6.10.9 Do not use numeric label roles on non-numeric elements.

Non-numeric elements must not have labels whose xlink:role value implies they apply to numeric values.

Value of the xlink:role attribute on link:label	Non-numeric
http://www.xbrl.org/2003/role/positiveLabel http://www.xbrl.org/2003/role/positiveTerseLabel http://www.xbrl.org/2003/role/positiveVerboseLabel http://www.xbrl.org/2003/role/negativeLabel http://www.xbrl.org/2003/role/negativeTerseLabel http://www.xbrl.org/2003/role/negativeVerboseLabel http://www.xbrl.org/2003/role/zeroLabel http://www.xbrl.org/2003/role/zeroTerseLabel http://www.xbrl.org/2003/role/zeroVerboseLabel http://www.xbrl.org/2003/role/totalLabel http://www.xbrl.org/2009/role/negatedLabel http://www.xbrl.org/2009/role/negatedPeriodEndLabel http://www.xbrl.org/2009/role/negatedPeriodStartLabel http://www.xbrl.org/2009/role/negatedTotalLabel http://www.xbrl.org/2009/role/negatedNetLabel http://www.xbrl.org/2009/role/negatedTerseLabel http://xbrl.us/us-gaap/role/label/negated http://xbrl.us/us-gaap/role/label/negatedTotal http://xbrl.us/us-gaap/role/label/negatedPeriodStart http://xbrl.us/us-gaap/role/label/negatedPeriodEnd	Disallowed
All Others	Allowed

6.11 Content of Label Linkbases

This section describes requirements on the content of label linkbases.

6.11.1 Line item element labels.

Assign a label of an element used in an instance the same text as the corresponding line item in the original HTML/ASCII document. An element is defined to be “used in an instance” if the element is the element of any fact, or whose QName is a dimension attribute or content of xbrldi:explicitMember.

A label is “assigned to an element” by an effective relationship with an xlink:arcrole attribute equal to ‘<http://www.xbrl.org/2003/role/concept-label>’ from the element declaration to the link:label.

Examples:

- The following line item appears in an income statement:

Line Item	Amount in thousands (2007)	Amount in thousands (2006)
Gross margin	\$15,212	\$10,195

- The appropriate element is us-gAAP:GrossProfit and it has the label “Gross Profit”. Both “gross profit” and “gross margin” mean “excess of revenues over the cost of revenues.” The registrant assigns the label “Gross margin,” and may choose to either exclude from the DTS the linkbase containing all standard labels, or to override the existing label using an equivalent element-label arc with a use attribute equal to ‘prohibit’, and an element-label arc with a use attribute equal to ‘optional’ and a value of the priority attribute greater than the existing element-label arc priority attribute.

Usually, the standard label of the element will be displayed to the user. If the element appears in different places of the Original HTML/ASCII Document with different labels, then use the preferredLabel attribute on a presentation relationship to distinguish different labels. For example, an element may appear as “Gross Profit” in a footnote but “Gross Profit (Loss)” on the face of the financials.

- The following line item appears in a balance sheet and the appropriate element is us-gAAP:ReceivablesNetCurrent.

Line Item	Amount in thousands (2007)	Amount in thousands (2006)
Receivables, less allowances of \$1,260 and \$1,150	\$31,659	\$31,601

The standard label of the element assigned in a standard taxonomy linkbase is “Receivables, Net, Current.” If the linkbase is included in the instance DTS, that label must be changed. The registrant should assign a label of “Receivables, less allowances of \$1,260 and \$1,150”.

6.11.2 Parenthetical item element labels.

Assign a label of an element in a parenthetical disclosure the same text as the corresponding text in the original HTML/ASCII document. For example, continuing the example from above, the treatment of the parenthetical will be as follows:

Line Item	Amount in thousands (2007)	Amount in thousands (2006)
Allowances	1,260	1,150

The registrant assigns element us-gAAP:AllowanceForDoubtfulAccountsReceivable a label “Allowances”.

6.11.3 When a numeric element label requires a unit designation.

If a numeric element is in presentation relationships in a relationship group with other numeric elements, and its units differ from the other elements and in the minority, then the suffix of its label must specify those units. Add a suffix such as “(in shares)” or “(in dollars per share)” to a label even though this would otherwise result in an exception to Rules 6.11.1 or 6.11.2 because the text differs.

Line Item	Amount (2009)	Amount (2008)
Adjustments (in millions of dollars)	12	-
Net Earnings (in millions of dollars)	331	174
...		
Earnings per Share	.03	.02

The element `us-gAAP:EarningsPerShare` has a data type of `xbrli:perShareItemType`, which is different from (and in the minority of) the other `xbrli:monetaryItemType` elements in this example. Therefore, in this case, assign the element `us-gAAP:EarningsPerShare` the label “Earnings per Share (in dollars per share)”.

6.11.4 Labels for numeric totals.

An `xsd:element` should be assigned a total label if the element will be presented with different labels depending on whether it is shown as a line item or as a summation of other line items. An element is assigned a total label if and only if it is assigned a label with an `xlink:role` attribute equal to `‘http://www.xbrl.org/2003/label/totalLabel’`.

For example, a fact may be shown with the label “Marketable Securities” on a balance sheet, but in a note detailing its composition, the same fact may be shown as “Marketable Securities, Total”.

The renderer will display a fact having a total label with an additional horizontal line below it. Moreover, the only way that the filer can arrange for such a horizontal line to be displayed in the renderer is by assigning a total label to the element that represents the total. As Rule 6.11.1 above points out, the `preferredLabel` of the presentation relationship must also be set to the total label.

6.11.5 Monetary elements without a balance attribute.

An `xsd:element` with a type attribute equal to `‘xbrli:monetaryItemType’` that does not have an `xbrli:balance` attribute must have a definition that disambiguates its sign. In most cases the registrant does not need to provide a definition label (that is, one with an `xlink:role` attribute equal to `‘http://www.xbrl.org/2003/role/documentation’`). However, a special case arises when a monetary item could take on a positive or negative value in different periods, and there is potential for confusion because it is not required to have an `xbrli:balanceType` attribute. The registrant must therefore either assign an appropriate `xbrli:balanceType` attribute or assign a label with an `xlink:role` attribute equal to `‘http://www.xbrl.org/2003/role/documentation’` containing text that makes the meaning of a positive (or negative) value explicit.

For example, a registrant defines an element called “Other Loss Adjustments, Net” with a type attribute equal to `‘xbrli:monetaryItemType’` but does not provide an `xbrli:balanceType` attribute. The text of the documentation is “A positive adjustment value indicates a net increase in cumulative losses.”

6.11.6 Monetary element negating labels.

Assign a “negating” label to an `xsd:element` with an `xbrli:balanceType` attribute that is inconsistent with the presentation in the official HTML/ASCII document. A numeric fact value will occasionally have the reverse sign of the figure as presented in the official HTML/ASCII document. A `link:label` has several possible `xlink:role` attributes that will ensure the number will be displayed with a reversed sign, called “negating” labels.

For example, one of the most common cases for a negating label is the monetary element `us-gAAP:TreasuryStockValue`, which has `xbrli:balanceType` of `‘debit’`. Its negating labels could be assigned as shown in this table:

xlink:role attribute	link:label
http://www.xbrl.org/2003/role/label	Treasury Stock, Value
http://www.xbrl.org/2009/role/negatedLabel	(Less) Treasury Stock, Value
http://www.xbrl.org/2009/role/negatedPeriodEndLabel	(Less) Treasury Stock, Value, Ending Balance
http://www.xbrl.org/2009/role/negatedPeriodStartLabel	(Less) Treasury Stock, Value, Beginning Balance
http://www.xbrl.org/2009/role/negatedTotalLabel	(Less) Treasury Stock, Value, Total
http://www.xbrl.org/2009/role/negatedNetLabel	(Less) Treasury Stock, Value, Net
http://www.xbrl.org/2009/role/negatedTerseLabel	(Less) Treasury Stock

6.11.7 Numeric element period start and end labels.

If an `xsd:element` with an `xbrli:periodType` attribute equal to ‘instant’ could be presented as either a beginning or end of period value in a roll forward, assign period start labels or period end labels. An element is assigned a period start label if and only if it is assigned a label with an `xlink:role` attribute equal to ‘http://www.xbrl.org/2003/label/periodStart’.

An element is assigned a period end label if and only if it is assigned a label with an `xlink:role` attribute equal to ‘http://www.xbrl.org/2003/label/periodEnd’.

This rule often applies to cash balances as presented in a statement of cash flows and other balances shown in roll forwards.

xlink:role attribute	link:label
http://www.xbrl.org/2003/role/label	Cash and Cash Equivalents
http://www.xbrl.org/2003/role/periodEnd	Cash and Cash Equivalents, Beginning Balance
http://www.xbrl.org/2003/role/periodStart	Cash and Cash Equivalents, Ending Balance

6.11.8 Terse and verbose labels.

Assign an `xsd:element` different terse or verbose labels if the same element will appear with different labels depending on the presentation relationships that have it as a target. An element is assigned a terse label if and only if it is assigned a label with an `xlink:role` attribute equal to ‘http://www.xbrl.org/2003/label/terse’.

An element is assigned a verbose label if and only if it is assigned a label with an `xlink:role` attribute equal to ‘http://www.xbrl.org/2003/label/verbose’.

The terms ‘terse’ and ‘verbose’ are only suggestive. There is no specific label role that is supposed to “match” the original HTML/ASCII documents; the presentation, with `preferredLabel` taken into account, is what matters.

For example, a fact whose element has the standard label ‘Earnings Per Share, Diluted’ in a standard taxonomy might appear in an income statement with the label “Diluted, per share” and in an Equity note with the label “Per share, diluted”; as long as the two labels have different values of the `xlink:role` attribute, either could be considered terse, verbose, or even as the standard label.

6.11.9 A label linkbase of a standard taxonomy should not be included in the DTS of an instance.

An exception to this general rule is when the label linkbase is required by a schema that defines elements or types of a standard taxonomy.

6.12 Syntax of Presentation Linkbases

This section defines rules governing the syntax restrictions on presentation linkbases. A valid Interactive Data presentation linkbase is a valid XBRL 2.1 presentation linkbase, but not all valid XBRL 2.1 presentation linkbases are valid Interactive Data presentation linkbases.

6.12.1 The link:presentationArc element requires an order attribute.

6.12.2 Presentation child order restrictions.

All effective presentation relationships in the same base set with the same source element must have distinct values of the order attribute. This rule ensures an intentional ordering of facts when displayed.

6.12.3 Every element in an instance must be presented.

An element used in an instance must participate in at least one effective presentation relationship in the DTS of that instance. An element “participates in” a presentation relationship in a DTS if it is a source or target of a presentation relationship in that DTS.

An element is “a source of a presentation relationship” in a DTS if there is an effective relationship with the defining xsd:element source and an xlink:arcrole attribute equal to ‘http://www.xbrl.org/2003/role/parent-child’ in a document of that DTS.

An element is “a target of a presentation relationship” in a DTS if there is an effective relationship with the defining xsd:element target and an xlink:arcrole attribute equal to ‘http://www.xbrl.org/2003/role/parent-child’ in a document of that DTS.

Every fact must be displayable in some way using presentation relationships. This rule is relevant to all elements but particularly so to elements declared in the company schema, because a linkbase that could contain a link:loc element for a company-specific element would never appear in a standard taxonomy, and therefore has to be in a linkbase in the same submission.

Elements used in an instance only as QNames in xbrldi:explicitMember must nevertheless have a presentation relationship in the DTS of that instance.

It is not necessary for the DTS to have a presentation relationship for all elements declared in the DTS.

6.12.4 Reserved.

6.12.5 When distinct preferredLabel attribute values are required.

If an element used in an instance is the target in the instance DTS of more than one effective presentation relationship in a base set with the same source element, then the relationships must have distinct values of the preferredLabel attribute. This rule prevents the same fact from appearing twice in a set of line items, except when it is, for example, shown as both the beginning and ending value of a roll forward.

6.12.6 Each effective presentation relationship base set should have only one root element.

An element that appears in a presentation base set only as a source and never as a target is a root element of that base set. The root element is normally abstract, but need not be (6.13.3).

6.12.7 Do not use preferredLabel roles for instant elements on duration elements.

An effective presentation relationship whose target is an `xsd:element` with an `xbrli:periodType` attribute equal to 'duration' should not have a `preferredLabel` attribute value that is a role for elements with `xbrli:periodType` attribute equal to 'instant'. A "duration-type element" is an element with `xbrli:periodType` equal to "duration". An "instant-type element" is an element with `xbrli:periodType` equal to "instant".

Roles that contain the text "periodStart", "periodEnd", "PeriodStart", or "PeriodEnd" anywhere in the role URI should not be used as the preferred label of a duration-type element.

6.12.8 Each axis element should be the parent of some domain element.

Each explicit axis (explicit dimension) element in an effective presentation relationship base set should be the source of at least one effective presentation relationship in the same base set whose target is a `domainItemType` element. As described in section 6.24, an axis element (defined in 6.8.20) appearing in an effective presentation relationship base set without any child elements will filter out all facts and prevent the entire base set from being used for presentation.

6.12.9 Presentation relationships for units.

A base set having one effective presentation relationship whose target has the same local name as the `unitRef` attribute value of a fact of a source or target element in the same base set should provide an ordering for all such `unitRef` attribute values. As described in 6.24, the renderer displays sets of facts having multiple units of measure that cannot be merged into columns (or rows) in one of two ways. Either the units are ordered by the order they are declared in the instance, or by using the built-in unit axis ordered by relationships in a presentation base set. The presentation base set should either order all the units used, or none of them, so that the ordering is not mixed.

6.13 Content of Presentation Linkbases

This section describes requirements on the content of presentation linkbases.

6.13.1 Reserved.

6.13.2 Present facts according to their locations in the original HTML/ASCII document.

The element of every fact in an instance must participate in an effective presentation relationship in a base set whose `xlink:role` attribute corresponds to the locations where the fact appears in the original HTML/ASCII document.

An Inline XBRL formatted Mutual Fund Risk/Return Summary (RR) instance (as defined in 6.25) is exempt from this requirement.

This rule requires that each `link:presentationArc` be assigned an `xlink:role` attribute value placing facts into the appropriate part of the financial statement at the appropriate level of detail.

6.13.3 Order and indent facts according to their positions in the original HTML/ASCII document.

Organize the effective presentation relationships in a base set using the ordering and indentation of the facts in the original HTML/ASCII document.

An Inline XBRL formatted Mutual Fund Risk/Return Summary (RR) instance (as defined in 6.25) is exempt from this requirement.

Typically, each base set in Rule 6.7.12 above will have a “root” element that is an abstract element, and that abstract element will be used by the renderer to display a heading that precedes all the facts in the base set.

Use presentation relationships to achieve ordering effects. Order a set of line items to appear as they do in the original HTML/ASCII document by using an element as their heading that will be the source of presentation relationships that have the line items as their target.

Occasionally, a single line in the original HTML/ASCII such as “Proceeds from/Payments to ...” with both positive and negative values in different periods will correspond to two elements, one representing the proceeds, and one the payments. In such cases, the order of the two adjacent elements relative to each other is arbitrary.

Use abstract elements to insert headings. A line item with no facts associated with it, serving as a subheading, is represented with an element having an abstract attribute equal to ‘true’. A total element often appears at the end of the list under such a heading.

For additional detail see section 6.24.6.

6.13.4 Presentation groups for parenthetical facts.

All elements of facts corresponding to parentheticals in the original HTML/ASCII document must be the targets only of effective presentation relationships in one base set and all having the same source abstract element. Filers must declare a link:roleType to contain the parentheticals in each statement, as shown in the example of Rule 6.7.12, and use an abstract element to serve as a heading for every parenthetical in that statement. Normally, this would be the same abstract element at the root of the base set representing the corresponding statement. In example 6.6.14, the element

us-gaap:AccountsReceivableNetCurrent appears in the balance sheet statement, so Rule 6.11.2 would be satisfied with a presentation relationship from the element us-gaap:BalanceSheetAbstract to the target us-gaap:AllowanceForDoubtfulAccountsReceivableCurrent in a base set separate from the main balance sheet statement itself.

6.14 Syntax of Calculation Linkbases

This section defines rules governing the syntax restrictions on calculation linkbases. A valid Interactive Data calculation linkbase is a valid XBRL 2.1 calculation linkbase, but not all valid XBRL 2.1 calculation linkbases are valid Interactive Data calculation linkbases.

- 6.14.1 Element link:calculationArc requires an order attribute.**
- 6.14.2 Element link:calculationArc requires a weight attribute value equal to 1 or -1.**
- 6.14.3 The source and target of a calculation relationship must have the same period type.**

Facts of elements with different values of the xbrli:periodType attribute must have different values of the contextRef attribute and therefore the calculation relationship between them has no meaning.

6.14.4 No directed cycles in calculation relationships.

There must be no directed cycles in effective relationships having arc role <http://www.xbrl.org/2003/role/summation-item>. This rule prevents a fact from participating in a summation that includes itself.

6.14.5 Facts in a calculation must appear together in at least one presentation group.

If an instance contains non-empty facts for the source and target of an effective calculation relationship, then at least one effective presentation relationship that the source and target appear in (because of 6.12.3) must be either (a) a relationship with each other or (b) two relationships with any other elements that share a single extended link role. When facts participate in a calculation together, they must be shown with presentation relationships in the same relationship group, although not necessarily adjacent to each other.

6.15 Content of Calculation Linkbases

This section describes requirements on the content of calculation linkbases.

6.15.1 Reserved.

6.15.2 When calculation relationships are required.

If the original HTML/ASCII document shows two or more line items along with their net or total during or at the end of the Required Context period, and the instance contains corresponding numeric facts, then the DTS of the instance must have an effective calculation relationship from the total element to each of the contributing line items. A calculation relationship is a link:calculationArc with an xlink:arcrole attribute equal to '<http://www.xbrl.org/2003/role/summation-item>'. The Required Context is defined in Rule 6.5.19, above.

Examples:

- A company's Cash flow from investments for the most recent quarter is shown as the sum of two lines: Payments for plant and equipment, plus Payments for marketable securities. Two calculation relationships are required.
- An income statement shows the line items "Revenues", "Cost of Goods Sold" and "Gross margin" as the net of the two values during the current quarter. Two calculation relationships are required. In this case, the relationship subtracting Cost of Goods Sold will have a weight attribute of -1.
- A balance sheet shows assets as the sum of current and non-current assets, as of the date falling at the end of the period of the Required Context. Two relationships are required.

- An income statement shows only earnings per share and diluted earnings per share, but no reconciling per-share amount. Calculation relationships are not required.
- An income statement shows earnings per share before and after an adjustment for change in accounting principles, along with the adjusting amount. Two calculation relationships are required, from the net earnings per share, to its two contributing amounts.
- A balance sheet shows Net Current Receivables with a parenthetical value for Allowances. Only two values are shown, so no calculation relationship is required. In general, parentheticals do not, by themselves, require calculation relationships.
- A footnote for ABC contains a table in which the Revenue of its separately reporting subsidiaries DEF, GHI and JKL are totaled. But, each of the four facts has a different contextRef attribute. Therefore, this does not require any calculation relationships.

There is no separate, independent requirement that every company-specific element be included in calculations. It is, however, one of the consequences of this rule that a company-specific monetary or other numeric item is often defined in such a way that it must participate in calculation relationships anyway.

6.15.3 Alternate calculations for a single fact.

If a footnote in the original HTML/ASCII document contains alternate line items that sum to the same total amount, then the XBRL instance document for that footnote must have calculation relationships for the original and alternate line items in distinct base sets. Calculation inconsistencies are tested separately in each base set.

For example, a tax liability is shown in a tax footnote as the sum of current and deferred tax liabilities, and elsewhere in the same footnote as the sum of domestic and foreign tax liabilities. There are two base sets, each containing two calculation relationships.

6.15.4 Each element source and target pair should only appear in one calculation relationship.

A fact in an instance whose element is the source of an effective calculation relationship in the instance DTS should not have the same calculation relationship target in more than one base set. An xsd:element should be the source of only one calculation relationship for any one target, without regard for base set.

Note that this rule refers to the calculation relationship, not the element; an element can occur in any number of face financial statements or footnotes. Legitimate exceptions to this rule occur when an element is shown in different parts of the financial statement as a sum of different, but overlapping, sets of other elements.

Examples:

- The income statement contains amounts pre-tax income, tax, and post-tax income. There are two line items and their net; therefore two calculation relationships are required in the base set for the balance sheet. In the tax footnote there is another occurrence of pre-tax income, tax, and post-tax income. The tax footnote does not need two calculation relationships, because the same relationship already exists on the income statement.

- A balance sheet shows Net Current Receivables with a parenthetical value for Allowances. Only two values are shown, so no calculation relationship is required. A footnote also includes an analysis of (the same) Net Current Receivables including, among other details, amounts for Gross and Allowances. The footnote has those two line items and their net and therefore a need for two calculation relationships. Whether any of these facts also appear elsewhere is relevant only if it would result in duplicated relationships.

6.16 Syntax of Definition Linkbases

This section defines rules governing the syntax restrictions on definition linkbases. A valid Interactive Data definition linkbase is a valid XBRL 2.1 definition linkbase, but not all valid XBRL 2.1 definition linkbases are valid Interactive Data definition linkbases.

6.16.1 Element link:definitionArc requires an order attribute.

This ensures an intentional displayed order of definition relationships.

6.16.2 Reserved.

6.16.3 Definition relationships requiring a domainItemType target.

The target of an effective relationship with an xlink:arcrole attribute equal to 'http://xbrl.org/int/dim/arcrole/dimension-domain' or 'http://xbrl.org/int/dim/arcrole/dimension-default' must be a domain member. In this rule both the dimension-domain and the dimension-default arc roles must have a source that is an Axis (xbrl:dimensionItem); these two rules work together to ensure that each Axis has a meaningful set of domain members.

6.16.4 Restrictions on cycles in definition relationships.

The xlink:arcrole attributes 'http://xbrl.org/int/dim/arcrole/domain-member' and 'http://xbrl.org/int/dim/arcrole/dimension-domain' must have no undirected cycles in any Directed Relationship Set as defined in XBRL Dimensions 1.0. For example, company ABC defines, in us-gaap:SegmentGeographicalDomain, the regions abc:MidwestMember and abc:SoutheastMember, but stpr:KY (Kentucky) cannot be in both regions.

This rule also impacts line items, so that the balance at the start and end of a roll forward cannot appear twice under a single axis. The same rendering effect is achieved by including only the ending balance in the domain-member relationships, so that the beginning balance will appear simply as the ending balance of the previous period.

Tables define the rows and columns (the axes) that cells (the facts) may have. The domain-member arc role defines relationships within each row or column, such as those between a parent entity and its reportable segments, among sets of classes of equity, and or among geographical regions. Tables become difficult to consistently populate with facts and ambiguous to display when elements can appear in more than one domain member. This rule ensures that any given element does not appear in more than one place along an Axis, and will not have any overlapping domain subsets or members. In general, almost every situation that at first appears to call for an Axis with tangled and overlapping subsets of Member elements actually turns out to be a case more clearly modeled using two distinct axes.

6.16.5 Restriction on dimension element relationships.

The DTS of an instance must contain in each base set, for each source element, at most one effective relationship with an xlink:arcrole attribute equal to ‘http://xbrl.org/int/dim/arcrole/all’. A fact can always appear in more than one Table (hypercube), but this rule prevents a fact from having contradictory meanings in different Tables.

6.16.6 Restriction on the "closed" attribute.

An effective relationship with an xlink:arcrole attribute equal to ‘http://xbrl.org/int/dim/arcrole/notAll’ must have an xbrldt:closed attribute equal to ‘false’.

6.16.7 Restriction on negative tables and axes.

An axis of a negative table must appear in a positive table in a definitionLink having an equal value of xlink:role. An Axis cannot appear as an Axis of a negative hypercube (that is, axis excluded from a table) unless there are members of that Axis in a positive table.

Formally, an axis element that is the target of an effective relationship with arcrole equal to ‘http://xbrl.org/int/dim/arcrole/hypercube-dimension’ that is consecutive from a relationship with arcrole ‘http://xbrl.org/int/dim/arcrole/notAll’ must also be the target of an effective relationship in a link:definitionLink having the same value of xlink:role and which itself is consecutive from an effective relationship with xlink:arcrole attribute equal to ‘http://xbrl.org/int/dim/arcrole/all’.

An instance DTS in which the arc role ‘http://xbrl.org/int/dim/arcrole/notAll’ does not appear will not be affected by this rule.

6.16.8 The target of an effective relationship with an xlink:arcrole attribute equal to

‘http://xbrl.org/int/dim/arcrole/notAll’ must not be the target of an effective arc with an xlink:arcrole attribute equal to ‘http://xbrl.org/int/dim/arcrole/all’ in link:definitionLink elements having equal values of xlink:role.

This rule ensures that a Table (hypercube) is not both positive and negative.

6.16.9 Restriction on target roles.

If the value of attribute xbrldt:targetRole on an effective definition relationship is not empty, then that relationship must have at least one effective consecutive relationship (as defined by the XBRL Dimensions specification). The xbrldt:targetRole attribute is used to connect certain definition arcs that appear in different extended link roles. A targetRole attribute that points to no arcs indicates that a table is malformed.

6.17 Content of Definition Linkbases

This section describes requirements on the content of definition linkbases.

6.17.1 Reserved.

6.18 Syntax of Reference Linkbases

This section defines rules governing the syntax restrictions on reference linkbases. A valid Interactive Data reference linkbase is a valid XBRL 2.1 reference linkbase, but not all valid XBRL 2.1 reference linkbases are valid Interactive Data reference linkbases.

Note: Although the Reference Linkbase file is a valid attachment type, at the moment it is not used.

6.18.1 An element that has a company specific namespace must not have a reference.

The elements defined in a company extension schema must not have any authoritative references.

An `xsd:element` “has a reference” if the DTS of the instance contains an effective reference relationship whose source is that `xsd:element`. A reference linkbase should only be attached to a submission when it is a copy of a linkbase published along with a schema for early adopters. In that situation the schema would have a `targetNamespace` attribute of some authority other than the registrant itself.

6.18.2 Standard elements must have no additional reference elements.

A company extension reference linkbase must not add, remove, or change references for any element declared in a standard taxonomy schema. A company extension also cannot remove or change references in standard taxonomies via prohibition of concept-reference relationships (this is a technical consequence of the rule prohibiting URI fragments other than shorthand xpointers).

6.19 Content of Reference Linkbases

This section describes requirements on the content of reference linkbases.

Note: Although the Reference Linkbase file is a valid attachment type, at the moment it is not used.

6.19.1 A reference linkbase of a standard taxonomy should not be in the DTS of an instance.

An exception to this general rule is when the linkbase is required by a schema that defines elements or types of a standard taxonomy. Note that in such a case some of the relationships may have priority attribute values that do not permit overriding.

6.20 EDGAR Module Processing with XBRL Taxonomy Extensions

EDGAR provides limited support for XBRL taxonomy extension documents as part of EDGAR Module processing. EDGAR Type 1 Modules (partial documents) are not allowed in XBRL format. Only EDGAR Type 2 Modules (complete documents) can be submitted in XBRL format.

EDGAR currently supports up to 10 EDGAR Module files per CIK. These 10 Modules may be used to store any combination of XBRL extension taxonomy files (schema and/or linkbase) and may be managed by the filer using the EDGAR Filing Website. These taxonomy extension files may be submitted before the official filing. Through the use of EDGAR Type 2 Module references to these XBRL documents, EDGAR can assemble these large documents into the filing without delaying the receipt of the entire filing.

As with any other kind of EDGAR Type 2 Module submission filed with EDGAR, filers may include an XBRL document, or XBRL documents, as attachments to an EDGAR Module submission, Template #5. A master submission may reference the XBRL EDGAR Module in a normal Type 2 fashion by using the Documents page.

6.21 Segment Functionality Not Supported for XBRL Documents

At this time, EDGAR does not support EDGAR segment processing of XBRL documents as discussed in Section 5.3.

XBRL segments can be used as described in the XBRL Specification. However, segments as described in Section 5.3 of the EDGAR Filer Manual are not supported. In EDGAR, “segment” refers to parts of a filing that can be submitted ahead of time and later assembled in a submission. It is this functionality that is not supported for XBRL documents. In the XBRL Specification 2.1, “segment” also refers to an XBRL tag that is used to provide additional information in cases where the entity identifier is insufficient. This use of segment is supported.

6.22 Supported Versions of XBRL Standard Taxonomies

Refer to the SEC’s public website (<http://www.sec.gov/info/edgar/edgartaxonomies.shtml>) for an up-to-date listing of standard taxonomy files that are supported by EDGAR, their locations, required namespaces, and where appropriate, the relevant EDGAR form types.

6.23 Reserved.

6.24 Rendering Interactive Data

The renderer transforms XBRL data into human readable output in the form of individual HTML files called “reports” or “R-files” and other supporting output. Each R-file has a table layout with rows, row headings, columns, column headings and footnotes.

XBRL parent-child relationships in the presentation base sets of a filing are generally the most important component for determining the overall layout of each R-file. Contexts, facts and footnotes in the instance, roles and data types declared in schemas, text in label linkbases, and the axis default members defined in definition linkbases, all further contribute to producing the output.

6.24.1 Presentation Groups

A “presentation group” is an effective presentation relationship base set. Section 6.13.3 described the content of presentation groups.

The relationships in a presentation group form a directed graph with one root element (see 6.12.7) having child elements in a fixed order, child elements of those elements each in a fixed order, and so on. Each presentation group has a single role URI and definition text as described in sections 6.7.9 through 6.7.12.

The overall order of presentation groups in the set of reports for a single instance is determined by the “SortCode” token described in 6.7.12.

The simplest report layout is one in which the presentation group contains no Axis or Domain Member elements. All of the facts in the instance whose elements appear as sources or targets of relationships in the presentation group in contexts having no `xbrli:segment` elements are displayed in the R-file. The facts are arranged into columns, one for each context. The presentation group, when traversed top down and in order, determines the order that the elements appear in the rows.

6.24.2 Context Selection

Contexts are selected for display by a presentation group by comparing the axis and their descendant members appearing in the group to the axes and members appearing in the context `xbrli:segment`. All contexts are selected in every presentation group except:

1. If a context segment has an axis with a member for which the axis does not have that member as a descendant in the presentation group does not appear in the base set, then the context will not be selected (it will be filtered out).
2. If the presentation contains an axis that does not have its default member as a descendant, and the context segment has no member for that axis, then the context will not be selected.

A warning is issued if the presentation group has axes but lacks descendants that would allow any context to be selected (6.26.1).

Contexts cannot be filtered based on anything other than the contents of `xbrli:segment`.

The set of periods in a set of contexts are on its Built-in Period Axis, or simply “period axis”.

6.24.3 Unit Selection

Units are always selected for display in a presentation group. The set of units is on the Built-in Unit Axis, or simply “unit axis”.

6.24.4 The Primary Axis

Elements in a presentation group that are not axis elements, not member elements, and do not have same local-name as a `unitRef` attribute value, are collectively called members on the Built-in Primary Axis, or simply “primary axis”.

6.24.5 Fact Selection

Non-numeric facts are selected for display when their context is selected and their element is on the primary axis.

Numeric facts are selected for display when their context is selected and their element is on the primary axis.

The values of `@xml:lang`, `@xsi:nil` and `@decimals` have no effect on selection whether present or not.

When there are duplicate facts as defined by 6.5.12, only one fact is selected.

6.24.6 Layout as Columns and Rows

A set of facts to be displayed has, at a minimum, a primary axis and a period axis. If there are numeric facts, it also has a unit axis. If the contexts have `xbrli:segment` elements then it has a set of additional axes. To lay out a report, every axis with more than one member must be assigned a layout direction (column or row) and if there is more than one axis per direction, they must be nested.

In the default layout, the period axis is on the columns and the primary axis is on the rows. For example:

Period Axis →	12 mo. ended Dec 31, 2013	Dec 31, 2012
Primary Axis ↓		
Line Items:		
Revenue:		
Assets:		

If there is a Unit axis, the units are on the Columns. If there are any other axes, they are on the rows. The nesting of the columns in the default layout is:

- The period axis is outermost;
- The unit axis is innermost.

For example:

Period Axis →	12 mo. ended Dec 31, 2013	12 mo. ended Dec 31, 2013	Dec 31, 2012	Dec 31, 2012
Unit Axis →	USD	CNY	USD	CNY
Primary Axis ↓				
Line Items:				
Revenue:				
Assets:				

The nesting of rows in the default layout is:

- The primary axis is innermost;
- Other axes are nested from outermost in, according to the order in which their axis element appears in the presentation base set.

For example, if the presentation base set had ordered the “Entity” Axis before the primary axis with a default member and one member “ABS Subsidiary”, then the nesting would appear as follows:

Entity Axis ↓	Period Axis →	12 mo. ended Dec 31, 2013	Dec 31, 2012
	Primary Axis ↓		
	Line Items:		
	Revenue:		
	Assets:		
ABC Subsidiary			
ABC Subsidiary	Line Items:		
ABC Subsidiary	Revenue:		
ABC Subsidiary	Assets:		

A warning is issued when the nesting of axes cannot be determined (see 6.26.6).

6.24.7 Member Ordering

Each axis has a characteristic ordering of its members either implicitly for all built-in axes (primary, period and unit), implicitly by a specified ordering of typed dimension members, or explicitly by the order of axis member elements in each specific presentation group.

When the period axis is on the columns, the periods are ordered first by durations before instants, then by increasing duration, and then by descending end date. For example:

Period Axis →	3 mo. ended Dec 31, 2013	3 mo. ended Dec 31, 2012	9 mo. ended Dec 31, 2013	9 mo. ended Dec 31, 2012
Primary Axis ↓				
Line Items:				
Revenue:				
Income:				

The order of elements along the primary axis is determined by traversing the presentation relationships top-down, with parents before their immediate children, and the immediate children ordered by the value of the order attribute ascending.

If there is no Unit Axis in the presentation base set, units are ordered by their order of appearance in the instance (6.12.9). The example above using USD and CNY would appear if the USD unit appeared in the instance before the CNY unit.

If there is more than one value of attribute `xml:lang` appearing on facts in the instance, then there is an implicit Language Axis in which facts with language “en-US” appear before all others, and the other language codes appear in case-insensitive lexical order.

Members along a typed dimension axis whose `@xbrldt:typedDomainRef` references an element whose XML Schema data type defines an order relation, are ordered by that relation. If there is no defined ordering, but the data type declaration includes `xs:enumeration` elements, the value ordering is determined by the sequence of `xs:enumeration` elements. If there is no ordering and no enumeration, then the values are sorted space-normalized, case sensitive, lexically.

The order of members along any other axis is determined in the same way as for the primary axis, by traversing the presentation base relationships. Conventionally, the default member of the axis appears first without any label, as illustrated in the “ABC Subsidiary” example above.

6.24.8 Merged Columns

If every column in the default layout represented a different combination of period and unit, then the display would often have significant empty space. In the examples below, the Revenue element is a duration-type element and the Assets element is an instant-type element, so that the figures could only appear this way:

Period Axis →	3 mo. ended Dec 31, 2013	9 mo. ended Dec 31, 2013	Dec 31, 2013
Primary Axis ↓			
Line Items:			
Revenue:	xxx,xxx	yyy,yyy	
Assets:			zzz,zzz

Columns representing an instant period are instead shown merged with the columns representing durations that end on that date:

Period Axis →	3 mo. ended Dec 31, 2013	9 mo. ended Dec 31, 2013
Primary Axis ↓		
Line Items:		
Revenue:	xxx,xxx	yyy,yyy
Assets:	zzz,zzz	zzz,zzz

A set of facts with different units all on distinct elements may be merged into a single column as well. In particular, when facts whose unit is a currency, facts with units “shares”, and facts whose unit is that same currency “per share” have compatible periods, they are merged into the same columns.

6.24.9 Period Start labels

An instant-type element of the primary axis that has a period start preferred label (see 6.12.7) results in a special kind of layout and column merge called a movement analysis. In a movement analysis, duration-type facts are matched to instant-type facts whose date-times are the same as its start and end dates. The same fact may thus appear as the ending value of one column, and the starting value of another. In the example below, the fact shown as YYY,YYY at Dec 31, 2012 is at the end of one period and the start of the next:

USD	12 mo. ended Dec 31, 2013	12 mo. ended Dec 31, 2012
Line Items:		
Balance, start of period:	YYY,YYY	xxx,xxx
Changes:	---	---
Balance, end of period:	zzz,zzz	YYY,YYY

In general, the starting and ending balance facts should be matched with at least one fact representing the duration between them, otherwise a warning may be issued (6.26.2).

Note that the order of elements along the primary axis is preserved and does not affect whether the layout is considered a movement analysis; the following example is equally valid, resembling a statement of cash flows:

USD	12 mo. ended Dec 31, 2013	12 mo. ended Dec 31, 2012
Line Items:		
Increases:	---	---
Decreases:	---	---
Other Changes:	---	---
Balance, start of period:	YYY,YYY	xxx,xxx
Balance, end of period:	zzz,zzz	YYY,YYY

6.24.10 Column Headings and Promotion

If all column headings share the same units or period, then those axes are “promoted” into the upper left corner of the layout to avoid redundancy.

Columns representing an instant in time are shown as (for example) "Oct. 19, 2008" or "May 07, 2009". Columns representing durations are shown as (for example) "6 months ended Oct 19, 2008" rounded to the nearest number of months. For example, the duration from Jul. 1 to Sep. 19 rounds to 3 months, but the period from Jul. 10 to Sep. 19 rounds to 2 months. Periods of more than one year are nevertheless shown in months, for example, two years starting Jan. 1 2008 would be shown as "24 months ended Dec. 31, 2009".

6.24.11 Row Headings and Promotion

The text shown in each row heading on the primary axis is the text of preferred label on that element position on the primary axis (see 6.11.4).

The text shown for non-primary axis members is a concatenation of the outermost member name’s preferred label (defaulting to the standard label), followed by a vertical bar “|”, followed by the next outermost member name label, and so on.

If the default member of an axis is the first number shown in order on that axis, its label is considered blank.

Any member that is not the only member shown for a given axis in that report is promoted into the upper left corner of the layout.

6.24.12 Footnotes and Merging

Footnotes of a report are displayed at the bottom of a report, sequentially numbered, and the footnote marks are numbers that appear as a superscript to the right of each fact to which they refer.

Every report's footnote numbering is local to that report, so that if a fact with a footnote is displayed in two different reports, it will be displayed with the same footnote text at the bottom of the report, but a different superscript number.

If all of the facts on a row have the same footnote mark, then that footnote mark is removed from the facts and “merged” to a position just to the right of the row labels and removed from the cells.

If all of the facts on a column have the same footnote mark, then that footnote mark is “merged” into the heading of the column and removed from the cells.

6.24.13 Flow Through Suppression on Statements

“Flow through” occurs when a fact is selected for display in more than one presentation group. One of the most common examples is that “Net Income” appears on both an income statement and a statement of cash flows.

Only presentation groups for which the “Type” token as described in 6.7.12 has the value “Statement” are subject to “flow through” suppression.

When a statement contains a column of facts that are all rendered in a column of some other presentation group, the column that is a subset of the other is removed (suppressed).

If columns in two statements have identical facts, then the column in the statement that appears first according to “SortCode” (6.7.12) is retained.

6.24.14 Cash Flow Statements

Presentation base sets for which the “Type” token as described in 6.7.12 has the value “Statement” and the “Title” text contains the word “cash” followed by “flow”, then the only columns shown are those with the longest duration and those which have at least one-fourth the number of rows containing values as the columns with longest duration.

This is a special case of flow through suppression; without it, a typical cash flow statement for a 3rd quarter financial statement might display unwanted columns.

6.24.15 Statements of Changes in Shareholder Equity

Presentation base sets for which the “Type” token has the value “Statement”, that present facts with period start and period end labels, and the “Title” text does not contain “parenthetical” but does contain the case insensitive pattern:

- a. “stockholder”, “shareholder”, or “changes” in addition to either “equity” or “deficit”, or
- b. “partners” or “accounts” in addition to “capital”, or
- c. “statement” and “equity”

are considered “Statement of Changes in Equity” statements. These have a special movement analysis layout. These axes, if present, are shown on the rows, in this nesting order:

- a. Period
- b. Scenario (StatementScenarioAxis)
- c. Primary
- d. Creation Date (CreationDateAxis)
- e. New Accounting Pronouncements
(AdjustmentsForNewAccountingPronouncementsAxis)
- f. Adjustments for Change in Accounting Principle
(AdjustmentsForChangeInAccountingPrincipleAxis)

- g. Error corrections and Prior Period Adjustment
(ErrorCorrectionsAndPriorPeriodAdjustments-
RestatementByRestatementPeriodAndAmountAxis)

These axes, if present, are shown on the columns, in this nesting order:

- a. Legal Entity (LegalEntityAxis)
- b. Equity Components (StatementEquityComponentsAxis)
- c. Partner Capital Components (PartnerCapitalComponentsAxis)
- d. Class of Stock (StatementClassOfStockAxis)
- e. (All other axes present)
- f. Unit

Also if the Scenario Axis is present then its member ordering is fixed in this order:

- a. Previously Reported (ScenarioPreviouslyReportedMember)
- b. Restatement Adjustment (RestatementAdjustmentMember)
- c. Change In Accounting Principle (ChangeInAccountingPrincipleMember)
- d. Unspecified (the default)

If the set of facts presented does not contain instant-type facts and duration-type facts in the expected alternating arrangement, a warning may be issued (see 6.26.3).

6.24.16 Layout Qualifiers

In addition to general layout and formatting rules there are additional ways of altering a layout by modifying the role definition text. When the tokens “Unlabeled”, “Elements”, or “Transposed” (case sensitive) appear in the definition text between curly braces “{...}”, then one of three transformations apply:

1. When the definition text contains the token “{Unlabeled}”, then the elements’ labels do not appear in the leftmost column.
2. When the definition text contains the token “{Elements}”, then the display has an additional column to the right of the labels, showing the element id. The {Elements} display can be useful for reviewing all the detailed content of an Interactive Data instance. {Elements} cannot be used with the period axis on the rows.
3. When the text contains the token “{Transposed}”, then the entire display, after any other layout has been applied, is transposed (rows on the columns and vice versa).

If more than one layout qualifier is present in a definition text, all are ignored.

6.24.17 Uncategorized Facts

Facts (and their duplicates) that are not selected for display by at least one presentation base set are shown in an “Uncategorized Facts” report. The report number for the Uncategorized facts is 9999 and it is shown as if it had the “Elements” layout qualifier.

6.24.18 Numeric Formatting

Numeric facts are scaled for presentation to reduce the appearance of redundant sequences of “000” groups. Each set of distinct units of facts selected for a report is scaled independently to preserve their significant digits. Note that this is performed only with respect to the fact values; the value of the decimals attribute of a fact does not directly affect this determination. For example, if there are three facts with unit USD:

- Revenues 1,000,000 USD, decimals=-6,
- Assets 100,000 USD, decimals=-3,
- Income 1,000 USD, decimals=0,

then all three facts would be shown scaled by thousands. Other facts in the same report with units such as “Ratio”, “Shares”, or “USD per Share” would not be affected by the scaling of USD.

Cells in the layout have unit symbols assigned as follows:

1. Each numeric cell has a unit (USD, GBP, Shares, etc.). A nonzero number will be displayed with a prefix consisting of the unit.
2. If all of the cells in a column have the same unit, then the symbol is copied to the column heading and removed from the cells.
3. For the non-zero cells in each column, remove a symbol from the cell if it has the same unit as the immediately prior nonzero cell, unless it is in the last row.
4. For the non-zero cells, in each row, remove a symbol from a cell if it has the same unit as the immediately prior nonzero cell, unless it is in the last column.
5. If there is a unique symbol for the unit (\$, £, ¥, €, etc.) use that symbol as a prefix instead of the unit name as a suffix.

6.24.19 Non-numeric Formatting

Fact values may be non-numeric; the following rules apply when displaying non-numeric values of an element:

1. A date item type is displayed with format “mmm d, yyyy”;
2. A time item type is displayed with format “hh:mm:ss”;
3. A datetime item type where the time part is not 00:00:00 is displayed with format “mmm d, yyyy hh:mm:ss”;
4. A duration item type (the value looks like “P..Y..M..D..H..M..S”) is formatted as “.. years, .. months, .. days, .. hours, .. minutes and .. seconds”, omitting any zero values.
5. A fact value that matches the regex for an XML QName `[_A-Za-z][A-Za-z]*:[_A-Za-z][A-Za-z0-9]*` is displayed as follows:
 - a. If the QName is being displayed with preferred label p and it has preferred label p in this DTS, then display that label value;
 - b. If the QName as a standard label in this DTS, then display that standard label value;
 - c. Otherwise display the QName unchanged.

6. A string that matches the regex “[^~]*~\s+{URI}\s+[^~]*~.*” is interpreted as an embedding command;
7. A string that contains the “<” character followed by a QName and whitespace, “/>” or “>” is interpreted as escaped XHTML and is rendered as XHTML.
8. Otherwise the string is copied to the output unchanged.

6.24.20 The Filing Summary

In addition to individual R-files, the renderer produces a Filing Summary file that contains data about the input files, each R-file output, warnings and error messages logged during processing, and other information.

In particular, the Filing Summary contains for each report a computed “menu category” to assist in organizing a hierarchical menu for all reports. The renderer computes the menu category by analyzing the report's role definition text (6.7.12) as follows:

SortCode determines the order of appearance (6.24.1).

Type distinguishes between statements and non-statements.

Title is text to be displayed in the menu augmented with the following patterns: “(Policies”, “(Tables”, “(Details”, layout qualifier tokens (6.24.16), and text that indicates cash flows (6.24.15) and statements of changes in equity (6.24.16).

The order of appearance and patterns in the Title for a single Interactive Data instance are processed as follows:

Category	Condition
Cover	Appears in first position and Type is not Statement
Statements	Type is Statement
Notes	Appears after Statements and is not Policies, Tables or Details
Policies	None of the above, and Title contains “(Polic”
Tables	None of the above, and Title contains “(Table”
Details	None of the above, and Title contains “(Details”
Other	None of the above
Uncategorized	Always appears as the last report

Note that the reports are categorized in order of appearance, and their category may depend on what report categories have already appeared.

An Uncategorized Facts report, if present, would always be in the last position by virtue of being report number 9999.

6.24.21 Multiple Instances

EDGAR submissions may have multiple Interactive Data instance document attachments. The renderer processes them all together in the same order that the instances appeared as EDGAR attachments. There is a single Filing Summary file output, and all the reports are numbered sequentially starting at 1. Each instance has a separate set of menu categories. When assigning a

Menu Category as described in 6.24.19, any report whose SortCode is less than the SortCode of the previous report marks the beginning of a new set of menu categories.

If there are multiple instances each with uncategorized facts, they are numbered in descending order 9999, 9998, etc.

6.24.22 Workbook Output

The filing summary and reports are processed to produce a spreadsheet workbook. Each report is shown as a separate sheet, with a name derived from the first 32 characters of the presentation base set title. Each fact is presented in a spreadsheet cell. Formatting of individual numeric and non-numeric facts is simplified as compared to the report HTML format.

6.25 Rendering of Mutual Fund Risk/Return Summary Interactive Data

An Interactive Data instance containing any fact whose element namespace contains the subtext “xbrl.sec.gov/rr/” is a Mutual Fund Risk/Return Summary (RR) instance. These filings are rendered with some differences as compared to other instances:

- The numeric value 0 is formatted as “none” (6.24.18).
- If any instance in a filing is an RR instance then the menu category of each report is empty, and all reports are shown together in a single menu called “Risk/Return Reports” (6.25.21).
- There is no workbook output (6.25.22).

RR instances are also allowed to use the additional features below.

6.25.1 Embedding Commands

Conventional rendering of Interactive Data consists only of one report per presentation base set. These are called the “top level” reports. Embedding commands allow additional reports that are not at top level. A text block fact may contain an embedding command, which is a section of XHTML that looks like the following, anywhere in the text block:

```
~ http://xbrl.sec.gov/rr/role/ShareholderFeesData
column period compact *
row dei_LegalEntityAxis compact eg_S000005977Member
row dei_ProspectusShareClassAxis compact *
~ Comments can go here
```

The purpose of an embedding command is to identify a presentation base set, select a subset of the contexts to display, and display the resulting report in place of where the content of the text block would have been shown. An embedding command consists of:

- One Role URI, which identifies the presentation base set to use for the layout;
- A list of “iterators”, each of which:
 - Places one of the axes (primary, period, unit, as described in 6.24.4 and 6.24.5) on either the rows or columns.
 - identify the style of that axis display (compact or nodisplay);
 - indicates which members of the axis may appear in the selected contexts (6.24.2).

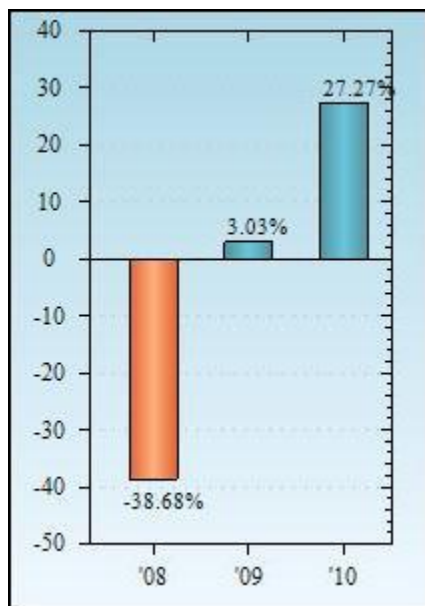
A “compact” display means that each axis member is shown on a separate row (or column) with its label; a “nodisplay” means that the label is not shown. Embedding layouts do not support the use of period start and period end preferred label roles.

Embedding command syntax, and the possible warning messages that may result from processing the command, are detailed in 6.26.4 through 6.24.6.

6.25.2 Bar Charts

When an embedded command has a role URI that contains the case-insensitive text “barchart” and the primary elements are drawn from the set rr_AnnualReturn2008, rr_AnnualReturn2009, and so forth, then the renderer produces a graphic instead of a conventional table. Up to 20 facts may be shown. The bar chart has a fixed height and auto-scales vertically; it has fixed width horizontal bars. For example, three facts are displayed in the graphic shown below:

Element	Value
rr_AnnualReturn2008	-0.3868
rr_AnnualReturn2009	0.0303
rr_AnnualReturn2010	0.2727



Although it is not necessary, the horizontal width of the graphic can be adjusted by adding ‘nil’ valued facts.

Footnotes on the fact values are ignored.

6.26 Syntax related to rendering

The renderer processes Interactive Data instances, linkbases and schemas that are valid with respect to the preceding sections 6.1 through 6.23. Even valid instances may contain syntax that cannot be interpreted by the renderer. This section defines the syntax that “must” or “must not” appear, with a violation error causing EDGAR to remove the Interactive Data exhibit.

A valid instance may also contain syntax that indicates that the data has been incorrectly tagged or that it cannot be displayed in a compliant fashion; this section defines syntax that “should” or “should not” occur and is communicated to the user via a warning.

6.26.1 Each axis that is presented requires at least one child element.

An effective presentation relationship base set that has an Axis element without its default member as a descendant should display at least one fact having a non-defaulted member on that axis. If the default member of an Axis does not appear in an effective presentation relationship base set, then the only facts that can be displayed by that presentation base set are facts in contexts having a non-defaulted member on that Axis. If the instance contains no such facts, then the base set will never display anything.

6.26.2 Period start and end preferred labels should be separated by duration elements.

Facts presented in an effective presentation relationship base set using period start or period end preferred labels should contain alternating instant and duration facts. The renderer lays out a set of facts with at least one column for each period, ordering the columns by increasing duration and descending date.

An effective presentation relationship base set in which facts are presented with the period start and period end roles as described in 6.12.6 is considered a movement analysis. When the renderer presents a movement analysis of a set of facts consisting of beginning and ending values and changes that occurred during the period, it orders the periods by alternating the instants and durations, in decreasing duration and increasing end date order.

The date-time of an instant-type fact shown with a period start (or end) label should appear as the start (or end) date-time of at least one duration-type fact selected (in the sense of 6.24.5) in the same movement analysis presentation group.

6.26.3 Changes in Equity presentation requires both instant and duration type facts.

A statement of changes in equity should contain at least one duration-type fact with starting and ending date-times matching instant facts at those date times. An effective presentation relationship base set that is recognized as representing a statement of changes is a table with sets of rows having alternating instant and duration-type elements (6.24.10). The special layout is not needed unless there is at least one duration-type fact that represents the change in value from one instant to another, even if it is a nil-valued fact used as a placeholder.

6.26.4 Text blocks containing embedding commands.

Facts of type “text block” having text content containing an embedding command must have valid embedding command syntax. A text block fact may contain an Embedding Command as detailed in Section 6.25. The following text is an example of an embedding command:

```
~ http://xbrl.sec.gov/rr/role/ShareholderFeesData
column period compact *
row dei_LegalEntityAxis compact (eg_AaaMember, eg_BbbMember)
row rr_ProspectusShareClassAxis compact *
~
```

The beginning and ending “~” (tilde, ASCII hex 7E) indicate that the content of the enclosing element is an embedding command. Any text appearing before the first or after the second tilde is ignored.

An embedding command consists of one role URI followed by zero or more iterators. Each iterator consists of case-sensitive tokens separated by whitespace.

The role URI must be the role URI of presentation relationships used to display the facts or contain the substring “BarChart”.

Any token with “_” (underscore, ASCII hex 5F) must denote an element. The text before the first underscore is the element preferred namespace prefix as defined in paragraph 6.7.7. The string after the first underscore must be nonempty and must be the element local name.

The first token of an iterator must be “row” or “column”.

The second token of an iterator must be “period”, ‘unit’, “primary”, or a token with an underscore that denotes an Axis as defined in 6.7.23.

The third token must be the word “compact” or “nodisplay”.

The fourth token is either “*” meaning “all”, or a token with an underscore that denotes a Domain Member as defined in 6.7.27, or a parenthesized list of Domain Members.

6.26.5 Restrictions on rows and columns of embedding commands.

An embedding command must have at least one row iterator and one column iterator, either explicit or defaulted. Embedding results in a different layout than the default layout of non-embedded reports (6.24.6). In an embedding command, the primary axis is on the rows by default and all other axes are the default columns (prior to transposition (6.24.16), which reverses rows and columns).

6.26.6 Completeness of axes in an embedding command.

An embedding command and the effective presentation relationship base set designated by its role URI should together provide an ordering of row axes and column axes for all facts it displays. The order of axes displayed on rows and the axes displayed on columns first takes into account the order of iterators in the embedding command. Then, for axes not named in the embedding command but used in the contexts of facts to be displayed, axis ordering is determined by the effective presentation relationship base set whose role URI is that of the embedding command.

6.26.7 A Bar Chart should select at least one Annual Return fact.

When the role URI of an embedding command as described in section 6.24 contains the substring “BarChart” then its row and column iterators define a set of Annual Return facts, and the set should not be empty.

6.26.8 A Bar Chart must select no more than twenty Annual Return facts.

When the role URI of an embedding command as described in section 6.24 contains the substring “BarChart” then its row and column iterators define a set of Annual Return facts, and the set must not contain more than twenty facts that are not duplicates (as defined by 6.5.12).

6.26.9 The {Elements} token implies "column primary" embedding.

An embedding command should not have both “{Elements}” in the definition text of its URI, and an iterator with direction token “column” and axis token “primary”. The layout qualifier “{Elements}” forces the primary axis to be displayed as rows.