INTERNATIONAL STANDARD

**ISO/IEC 19988**

Second edition
2017-10

# Information technology — Core Business Vocabulary Standard

*Technologies de l'information — Vocabulaire normatif relatif aux activités de base*

Reference number
ISO/IEC 19988:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

# Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by the GS1 and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

This second edition cancels and replaces the first edition (ISO/IEC 19988:2015), which has been technically revised.

The main changes compared to the previous edition are as follows:

— A new standard vocabulary for EPCIS error declaration reason identifiers is added.

— The URI structure for EPCIS event identifiers is specified.

— New business step values dispensing and voidShipping added.

— New disposition values dispensed and partially_dispensed added.

— A new section for trade item master data attributes is added, and the section on location and party master data attributes is expanded.

## Document Summary

| Document Item | Current Value |
|---|---|
| Document Name | Core Business Vocabulary Standard |
| Document Date | Sep 2016 |
| Document Version | 1.2 |
| Document Issue | |
| Document Status | Ratified |
| Document Description | specifies the structure of vocabularies and specific values for the vocabulary elements to be utilised in conjunction with the GS1 EPCIS standard |

## Log of Changes

| Release | Date of Change | Changed By | Summary of Change |
|---|---|---|---|
| 1.0 | Oct 2010 | | Initial release |
| 1.1 | March 2014 | | A new standard vocabulary for EPCIS source/destination type is added.<br>Templates for new user vocabularies for EPCIS source/destination identifier, EPCIS transformation identifier, and object classes are added.<br>New business step, disposition, and business transaction type values are added. The definitions of existing values are also clarified.<br>Disposition values non_sellable_expired, non_sellable_damaged, non_sellable_disposed, non_sellable_no_pedigree_match, and non_sellable_recalled defined in CBV 1.0 are deprecated in favour of new disposition values expired, damaged, disposed, no_pedigree_match, and recalled introduced in CBV 1.1.<br>RFC5870-compliant geocoordinate URIs are now permitted as location identifiers.<br>The introductory material is revised to align with the GS1 System Architecture. |
| 1.2 | Sep 2016 | | CBV 1.2 is fully backward compatible with CBV 1.1 and 1.0.<br>CBV 1.2 includes these new or enhanced features:<br>A new standard vocabulary for EPCIS error declaration reason identifiers is added.<br>The URI structure for EPCIS event identifiers is specified.<br>New business step values dispensing and voidShipping added.<br>New disposition values dispensed and partially_dispensed added.<br>A new section for trade item master data attributes is added, and the section on location and party master data attributes is expanded. |

## Disclaimer

GS1®, under its IP Policy, seeks to avoid uncertainty regarding intellectual property claims by requiring the participants in the Work Group that developed this **Core Business Vocabulary Standard** to agree to grant to GS1 members a royalty-free licence or a RAND licence to Necessary Claims, as that term is defined in the GS1 IP Policy. Furthermore, attention is drawn to the possibility that an implementation of one or more features of this Specification may be the subject of a patent or other intellectual property right that does not involve a Necessary Claim. Any such patent or other intellectual property right is not subject to the licencing obligations of GS1. Moreover, the agreement to grant licences provided under the GS1 IP Policy does not include IP rights and any claims of third parties who were not participants in the Work Group.

Accordingly, GS1 recommends that any organisation developing an implementation designed to be in conformance with this Specification should determine whether there are any patents that may encompass a specific implementation that the organisation is developing in compliance with the Specification and whether a licence under a patent or other intellectual property right is needed. Such a determination of a need for licencing should be made in view of the details of the specific system designed by the organisation in consultation with their own patent counsel.

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGMENT, FITNESS FOR PARTICULAR PURPOSE, OR ANY WARRANTY OTHER WISE ARISING OUT OF THIS SPECIFICATION. GS1 disclaims all liability for any damages arising from use or misuse of this Standard, whether special, indirect, consequential, or compensatory damages, and including liability for infringement of any intellectual property rights, relating to use of information in or reliance upon this document.

GS1 retains the right to make changes to this document at any time, without notice. GS1 makes no warranty for the use of this document and assumes no responsibility for any errors which may appear in the document, nor does it make a commitment to update the information contained herein.

GS1 and the GS1 logo are registered trademarks of GS1 AISBL.

# Table of Contents

# 1 Introduction – Core Business Vocabulary

This GS1 standard defines the Core Business Vocabulary (CBV). The goal of this standard is to specify various vocabulary elements and their values for use in conjunction with the EPCIS standard [EPCIS1.2], which defines mechanisms to exchange information both within and across organisation boundaries. The vocabulary identifiers and definitions in this standard will ensure that all parties who exchange EPCIS data using the Core Business Vocabulary will have a common understanding of the semantic meaning of that data.

This standard is intended to provide a basic capability that meets the above goal. In particular, this standard is designed to define vocabularies that are *core* to the EPCIS abstract data model and are applicable to a broad set of business scenarios common to many industries that have a desire or requirement to share data. This standard intends to provide a useful set of values and definitions that can be consistently understood by each party in the supply chain.

Additional end user requirements may be addressed by augmenting the vocabulary elements herein with additional vocabulary elements defined for a particular industry or a set of users or a single user. Additional values for the standard vocabulary types defined in this standard may be included in follow-on versions of this standard.

This standard includes identifier syntax and specific vocabulary element values with their definitions for these *Standard Vocabularies*:

- Business step identifiers
- Disposition identifiers
- Business transaction types
- Source/Destination types
- Error reason identifiers

This standard provides identifier syntax options for these *User Vocabularies:*

- Objects
- Locations
- Business transactions
- Source/Destination identifiers
- Transformation identifiers
- Event identifiers

This standard provides *Master Data Attributes and Values* for describing Physical Locations including:

- Site Location
- Sub-Site Type
- Sub-Site Attributes
- Sub-Site Detail

Additional detailed master data regarding locations (addresses, etc.) are not defined in this standard.

# 2 Relationship to the GS1 System Architecture

The Core Business Vocabulary is a companion standard to the EPCIS standard. EPCIS is the standard that defines the technical interfaces for capturing and sharing event data. EPCIS defines a framework data model for event data. The Core Business Vocabulary is a GS1 *data standard* that supplements that framework by defining specific data values that may populate the EPCIS data model. As such, the CBV exists in the “Share” group of GS1 standards.

# 3 Relationship to EPCIS

This section specifies how the Core Business Vocabulary standard relates to the EPC Information Services (EPCIS) standard.

## 3.1 EPCIS event structure

The EPCIS 1.2 standard [EPCIS1.2] specifies the data elements in an EPCIS event. The following lists these data elements, and indicates where the Core Business Vocabulary provides identifiers that may be used as values for those data elements.

- **The "what" dimension**: The *what* dimension for most event types contains one or more unique identifiers for physical or digital objects or classes of physical or digital objects. Identifiers for physical or digital objects in the Core Business Vocabulary are specified in *Section 8.2 (instance-level) and Section 8.3 (class-level)*. In the case of an EPCIS TransformationEvent, an optional TransformationID may be used to link together multiple events that describe the same transformation. The Core Business Vocabulary includes TransformationIDs in *Section 8.7.*
- **The "when" dimension**: The moment in time at which an EPCIS event occurred. Event time is fully specified in the EPCIS standard.
- **The "where" dimension**: The "where" dimension consists of two identifiers that describe different aspects of where an event occurred:
  - **Read Point**: The location where the EPCIS event took place. In the case of an EPCIS event arising from reading a barcode or RFID tag, the Read Point is often the location where the barcode or RFID tag was read. Identifiers for read points in the Core Business Vocabulary are specified in *Section 8.3.*

    Example: A reader is placed at dock door #3 at the London Distribution Centre (DC). Product passed through the dock door. Read point = <The identifier that stands for London DC Dock Door #3>
  - **Business Location**: The location where the subject of the event is assumed to be following an EPCIS event, until a new event takes place that indicates otherwise. Identifiers for business locations in the Core Business Vocabulary are specified in *Section 8.3*.

    Example: A product is read through the sales floor transition door at store #123. The product is now sitting on the sales floor. Business location = <The identifier that stands for store #123 Sales Floor>
- **The "why" dimension**: The "why" dimension consists of two identifiers and a list of business transaction identifiers, which collectively provide the business context or "why" the event occurred:
  - **Business Step**: Denotes a specific activity within a business process. The business step field of an event specifies what business process step was taking place that caused the event to be captured. Identifiers for business steps in the Core Business Vocabulary are specified in *Section 7.1*.

    Example: an EPCIS event is generated as a product departs the location identified by the Read Point. Business Step = <The identifier that denotes "shipping">
  - **Disposition**: Denotes the business state of an object. The disposition field of an event specifies the business condition of the subject of the event (the things specified in the "what" dimension), subsequent to the event. The disposition is assumed to hold true until another event indicates a change of disposition. Identifiers for dispositions in the Core Business Vocabulary are specified in *Section 7.2.*

    Example: an EPCIS event is generated and afterward the products can be sold as-is and customers can access product for purchase. Disposition = <The identifier that denotes "sellable and accessible">
  - **Business Transaction References**: An EPCIS event may refer to one or more business transaction documents. Each such reference consists of two identifiers:

- **Business Transaction Type**: Denotes a particular kind of business transaction. *Example: the identifier that denotes "purchase order"*. Identifiers for business transaction types in the Core Business Vocabulary are specified in *Section 7.3.*
- **Business Transaction Identifier**: Denotes a specific business transaction document of the type indicated by the Business Transaction Type.

  Example: <The identifier that denotes Example Corp purchase order #123456> Identifiers for business transactions in the Core Business Vocabulary are specified in *Section 8.5.*

□ **Source and Destination References**: An EPCIS event may refer to one or more sources and/or destinations that describe the endpoints of a business transfer of which the event is a part. Each source or destination reference consists of two identifiers:

- **Source or Destination Type**: Denotes a particular kind of source or destination. *Example: the identifier that denotes "owning party"*. Identifiers for source and destination types in the Core Business Vocabulary are specified in *Section 7.4*.
- **Source or Destination Identifier**: Denotes a source or destination of the type indicated by the Business Transaction Type. *Example: <The identifier that denotes Example Corp as an owning party>* Identifiers for sources and destinations in the Core Business Vocabulary are specified in *Section 8.6.*

## 3.2 Vocabulary kinds

(The material in this section is adapted directly from [EPCIS1.2], *Section 6.2*.)

Vocabularies are used extensively within EPCIS to model conceptual, physical, and digital entities that exist in the real world.

Examples of vocabularies defined in the EPCIS standard are business steps, dispositions, location identifiers, physical or digital object identifiers, business transaction type names, and business transaction identifiers. In each case, a vocabulary represents a finite (though open-ended) set of alternatives that may appear in specific fields of events.

It is useful to distinguish two kinds of vocabularies, which follow different patterns in the way they are defined and extended over time:

- **Standard Vocabulary**: A Standard Vocabulary is a set of Vocabulary Elements whose definition and meaning must be agreed to in advance by trading partners who will exchange events using the vocabulary.
- **User Vocabulary**: A User Vocabulary is a set of Vocabulary Elements whose definition and meaning are under the control of a single organisation.

These concepts are explained in more detail below.

### 3.2.1 Standard Vocabulary

A Standard Vocabulary is a set of Vocabulary Elements whose definition and meaning must be agreed to in advance by trading partners who will exchange events using the vocabulary. For example, the EPCIS standard defines a vocabulary called "business step," whose elements are identifiers denoting such things as "shipping," "receiving," and so on. One trading partner may generate an event having a business step of "shipping," and another partner receiving that event through a query can interpret it because of a prior agreement as to what "shipping" means.

Standard Vocabulary elements tend to be defined by organisations of multiple end users, such as GS1, industry consortia outside GS1, private trading partner groups, and so on. The master data associated with Standard Vocabulary elements, if any master data is defined at all, are defined by those same organisations, and tend to be distributed to users as part of a standard or by some similar means. New vocabulary elements within a given Standard Vocabulary tend to be introduced through a very deliberate and occasional process, such as the ratification of a new version of a standard or through a vote of an industry group.

The Standard Vocabularies specified in the Core Business Vocabulary standard are: *business steps (Section 7.1), dispositions (Section 7.2), business transaction types (Section 7.3)*, and *source and* destination types (Section 7.4). The elements and definitions are agreed to by parties prior to exchanging data, and there is general agreement on their meaning.

Example: the following is a business step identifier defined in Section 7.1 herein:

```
urn:epcglobal:cbv:bizstep:receiving
```

This identifier is defined by the GS1 Core Business Vocabulary standard, and its meaning is known and accepted by those who implement the standard.

While an individual end user organisation acting alone may introduce a new Standard Vocabulary element, such an element would have limited use in a data exchange setting, and would probably only be used within an organisation's four walls. On the other hand, an industry consortium or other group of trading partners may define and agree on standard vocabulary elements beyond those defined by the Core Business Vocabulary, and these may be usefully used within that trading group.

### 3.2.2 User Vocabulary

A User Vocabulary is a set of Vocabulary Elements whose definition and meaning are under the control of a single organisation. For example, the EPCIS standard defines a vocabulary called "business location," whose elements are identifiers denoting such things as "Acme Corp. Distribution Centre #3." The location identifier and any associated master data is assigned by the user. Acme Corp may generate an event whose business location field contains the identifier that denotes "Acme Corp. Distribution Centre #3," and another partner receiving that event through a query can interpret it either because the partner recognises the identifier as being identical to the identifier received in other events that took place in the same location, or because the partner consults master data attributes associated with the location identifier, or both.

Example:

```
urn:epc:id:sgln:0614141.12345.400
```

This identifier is assigned by the End User who owns the GS1 Company Prefix 0614141, and the meaning of the identifier (that is, what location it denotes) is determined exclusively by that end user. Another End User can understand the meaning of this identifier by consulting associated master data.

User Vocabulary elements are primarily defined by individual end user organisations acting independently. The master data associated with User Vocabulary elements are typically defined by those same organisations, and are usually distributed to trading partners through the EPCIS Query Interface or other data exchange / data synchronisation mechanisms. New vocabulary elements within a given User Vocabulary are introduced at the sole discretion of an end user, and trading partners must be prepared to respond accordingly.

While the Core Business Vocabulary standard does not (and as the discussion above makes clear, cannot) specify particular user vocabulary elements, the Core Business Vocabulary does provide syntax templates that are recommended for use by End Users in constructing their own user vocabulary elements. See Section 8.1. The user vocabularies for which templates are specified in this standard are: physical or digital objects (Sections 8.2 and 8.3), locations which include both read points and business locations (Section 8.4), business transaction identifiers (Section 8.5), source/destination identifiers (Section 8.6), and transformation identifiers (Section 8.7).

# 4 Terminology and typographical conventions

Within this standard, the terms SHALL, SHALL NOT, SHOULD, SHOULD NOT, MAY, NEED NOT, CAN, and CANNOT are to be interpreted as specified in Annex G of the ISO/IEC Directives, Part 2, 2001, 4th edition [ISODir2]. When used in this way, these terms will always be shown in ALL CAPS; when these words appear in ordinary typeface they are intended to have their ordinary English meaning.

All sections of this document, with the exception of Sections 2, 3 and 3 are normative, except where explicitly noted as non-normative.

The following typographical conventions are used throughout the document:

- ALL CAPS type is used for the special terms from [ISODir2] enumerated above.

- Monospace type is used to denote programming language, UML, and XML identifiers, as well as for the text of XML documents.

➢ Placeholders for changes that need to be made to this document prior to its reaching the final stage of approved GS1 standard are prefixed by a rightward-facing arrowhead, as this paragraph is.

# 5 Compliance and compatibility

The GS1 Core Business Vocabulary is designed to facilitate interoperability in EPCIS data exchange by providing standard values for vocabulary elements to be included in EPCIS data. The standard recognises that the greatest interoperability is achieved when all data conforms to the standard, and also recognises that individual End Users or groups of trading partners may need to extend the standard in certain situations.

To that end, this standard defines two levels of conformance for EPCIS documents:

- **CBV-Compliant**: An EPCIS document that only uses vocabulary identifiers specified in the Core Business Vocabulary standard in the standard fields of EPCIS events.
- **CBV-Compatible**: An EPCIS document that uses a combination of vocabulary identifiers specified in the Core Business Vocabulary standard and other identifiers that are outside the standard.

An EPCIS document is neither CBV-Compliant nor CBV-Compatible if it wrongly uses identifiers defined in the Core Business Vocabulary standard or if it violates any other rules specified herein.

The formal definition of these terms is specified below.

## 5.1 CBV Compliant

A “CBV-Compliant Document” is a document that conforms to the schema and other constraints specified in [EPCIS1.2], and which furthermore conforms to all the normative language in this standard that pertains to a “CBV-Compliant Document.”

A “CBV-Compliant Application” is any application for which both of the following are true:

- If it operates in a mode where it claims to accept a CBV-Compliant Document as an input, the application SHALL accept any document that is a CBV-Compliant Document according to this standard, and furthermore in processing that input SHALL interpret each CBV identifier according to the meaning specified herein.
- If it operates in a mode where it claims to produce a CBV-Compliant Document as an output, the application SHALL only produce a document that is a CBV-Compliant Document according to this standard, and furthermore in generating that output SHALL only use CBV identifiers to denote their meaning as specified herein.

The following list summarises the requirements for an EPCIS document to be a “CBV-Compliant Document,” as specified elsewhere in this standard:

- A CBV-Compliant Document SHALL conform to the schema and other constraints specified in [EPCIS1.2].
- A CBV-Compliant Document SHALL NOT use any URI beginning with `urn:epcglobal:cbv:` except as specified in this standard.
- Each EPCIS event in a CBV-Compliant Document SHALL include a `bizStep` field, and the value of the `bizStep` field SHALL be a URI consisting of the prefix `urn:epcglobal:cbv:bizstep:` followed by the string specified in the first column of some row of the table in *Section 7.1.3.*
- A CBV-Compliant Document MAY include a `disposition` field. If the `disposition` field is present, the value of the `disposition` field SHALL be a URI consisting of the prefix `urn:epcglobal:cbv:disp:` followed by the string specified in the first column of some row of the table in *Section 7.2.3*.

- Each EPCIS event in a CBV-Compliant Document MAY include one or more bizTransaction elements. If bizTransaction elements are present, each such element MAY include a type attribute. If a given bizTransaction element includes a type attribute, the value of the type attribute SHALL be a URI consisting of the prefix urn:epcglobal:cbv:btt: followed by the string specified in the first column of some row of the table in Section 7.3.3.
- Each EPCIS event in a CBV-Compliant Document MAY include one or more source or destination elements. The value of the type attribute of each such element SHALL be a URI consisting of the prefix urn:epcglobal:cbv:sdt: followed by the string specified in the first column of some row of the table in Section 7.4.3.
- Each EPCIS event in a CBV-Compliant Document MAY include an ErrorDeclaration element, and when present, the ErrorDeclaration element MAY include a reason field. When present in a CBV-Compliant Document, the value of the reason field of the ErrorDeclaration element SHALL be a URI consisting of the prefix urn:epcglobal:cbv:er: followed by the string specified in the first column of some row of the table in Section 7.5.3.
- URIs defined in the EPC Tag Data standard SHALL only be used in a CBV-Compliant Document as specified in Section 8.1.1.
- A CBV-Compliant document SHALL use one of the three URI forms specified in Section 8.2 to populate instance-level identifiers in the "what" dimension of EPCIS events (that is, the epcList, parentID, childEPCs, inputEPCList, and outputEPCList fields in EPCIS ObjectEvents, AggregationEvents, TransacationEvents, and TransformationEvents), for every such field that is not null. A CBV-Compliant document SHOULD use the EPC URI form as specified in Section 8.2.1 unless there is a strong reason to do otherwise.
- A CBV-Compliant document SHALL NOT use an SGLN EPC (urn:epc:id:sgln:…) as an object identifier.
- A CBV-Compliant document SHALL use one of the three URI forms specified in Section 8.3 to populate class-level identifiers in the "what" dimension of EPCIS events (that is, the epcClass fields in all EPCIS event types), for every such field that is not null. A CBV-Compliant document SHOULD use the EPC URI form as specified in Section 8.3.1 unless there is a strong reason to do otherwise.
- A CBV-Compliant document SHALL use one of the four URI forms specified in Section 8.4 to populate the "where" dimension of EPCIS events (that is, the readPoint and businessLocation fields in all EPCIS event types), for every such field that is not null. A CBV-Compliant document SHOULD use the EPC URI form as specified in Section 8.4.1 unless there is a strong reason to do otherwise.
- When using an EPC URI as a location identifier (Section 8.4.1), a CBV-Compliant document SHOULD NOT use EPC schemes other than SGLN (urn:epc:id:sgln:…), unless there is a strong reason to do so.
- A CBV-Compliant document SHALL use one of the four URI forms specified in Section 8.5 to populate the business transaction identifier field (that is, the text content of the bizTransaction element) of EPCIS events, for every such field that is not null.
- When using an EPC URI as a business transaction identifier, a CBV-Compliant Documents SHOULD NOT use EPC schemes other than GDTI EPCs (urn:epc:id:gdti:…) or GSRN EPCs (urn:epc:id:gsrn:…), unless there is a strong reason to do so. GDTI EPCs SHOULD only be used as business transaction identifiers when they have been assigned to denote a business transaction, rather than a physical document not connected with any business transaction.
- A CBV-Compliant document SHALL use one of the three URI forms specified in Section 8.6 to populate a source or destination identifier field (that is, the text content of a source or destination element), for every such field that is not null. A CBV-Compliant document SHOULD use the EPC URI form as specified in Section 8.6.1 unless there is a strong reason to do otherwise.

- When using an EPC URI as a source or destination identifier (Section 8.6.1), a CBV-Compliant document SHOULD NOT use EPC schemes other than SGLN (`urn:epc:id:sgln:…`), unless there is a strong reason to do so.
- A CBV-Compliant document SHALL use one of the four URI forms specified in Section 8.7 to populate the transaction identifier field (that is, the text content of the `transformationID` element) of EPCIS `TransformationEvents`, for every such field that is not null.
- When using an EPC URI as a transformation identifier, a CBV-Compliant Document SHOULD NOT use EPC schemes other than GDTI EPCs (`urn:epc:id:gdti:…`) unless there is a strong reason to do so. GDTI EPCs SHOULD only be used as transformation identifiers when they have been assigned to denote a transformation, rather than a physical document not connected with any transformation.
- A CBV-Compliant document SHALL use one of the URI forms specified in Section 8.8.1 to populate the event identifier field (that is, the text content of the `eventID` element) of an EPCIS event, whenever that field is not null.

## 5.2 CBV compatible

A “CBV-Compatible Document” is a document that conforms to the schema and other constraints specified in [EPCIS1.2], and which furthermore conforms to all the normative language in this standard that pertains to a “CBV-Compatible Document.”

A “CBV-Compatible Application” is any application for which both of the following are true:

- If it operates in a mode where it claims to accept a CBV-Compatible Document as an input, the application SHALL accept any document that is a CBV-Compatible Document according to this standard, and furthermore in processing that input SHALL interpret each CBV identifier according to the meaning specified herein.
- If it operates in a mode where it claims to produce a CBV-Compatible Document as an output, the application SHALL only produce a document that is a CBV-Compatible Document according to this standard, and furthermore in generating that output SHALL only use CBV identifiers to denote their meaning as specified herein.

The following list summarises the requirements for an EPCIS document to be a “CBV-Compatible Document,” as specified elsewhere in this standard.

- A CBV-Compatible Document SHALL conform to the schema and other constraints specified in [EPCIS1.2].
- A CBV-Compatible Document SHALL NOT use any URI beginning with `urn:epcglobal:cbv:` except as specified in this standard.
- URIs defined in the EPC Tag Data standard SHALL only be used in a CBV-Compatible Document as specified in Section 8.1.1.
- A CBV-Compatible Document SHOULD use the EPC URI form as specified in Section 8.2.1 for each instance-level object identifier unless there is a strong reason to do otherwise.
- A CBV-Compatible Document SHOULD use the EPC URI form as specified in Section 8.3.1 for each class-level object identifier unless there is a strong reason to do otherwise.
- A CBV-Compatible Document SHALL NOT use an SGLN EPC (`urn:epc:id:sgln:…`) as an object identifier.
- A CBV-Compatible Document SHOULD use the EPC URI form as specified in Section 8.4.1 for each location identifier unless there is a strong reason to do otherwise.
- When using an EPC URI as a location identifier (Section 8.4.1), a CBV-Compatible Document SHOULD NOT use EPC schemes other than SGLN (`urn:epc:id:sgln:…`), unless there is a strong reason to do so.
- When using an EPC URI as a business transaction identifier, a CBV-Compatible Document SHOULD NOT use EPC schemes other than GDTI EPCs (`urn:epc:id:gdti:…`) or GSRN EPCs (`urn:epc:id:gsrn:…`), unless there is a strong reason to do so. GDTI EPCs SHOULD only be

used as business transaction identifiers when they have been assigned to denote a business transaction, rather than a physical document not connected with any business transaction.

- When using an EPC URI as a source or destination identifier (*Section 8.6.1)*, a CBV-Compatible document SHOULD NOT use EPC schemes other than SGLN (`urn:epc:id:sgln:…`), unless there is a strong reason to do so.
- When using an EPC URI as a transformation identifier, a CBV-Compatible Document SHOULD NOT use EPC schemes other than GDTI EPCs (`urn:epc:id:gdti:…`) unless there is a strong reason to do so. GDTI EPCs SHOULD only be used as transformation identifiers when they have been assigned to denote a transformation, rather than a physical document not connected with any transformation.

In general, every CBV-Compliant Document is also a CBV-Compatible Document, though not every CBV-Compatible Document is a CBV-Compliant Document. A CBV-Compatible Document may include an identifier that is compliant with [EPCIS1.2] but which is not permitted for CBV-Compliant Documents, provided that it meets the requirements above. A CBV-Compatible Document may also include an event in which the `bizStep` field is omitted, whereas that field is always required for CBV-Compliant Documents.

# 6 Use of Uniform Resource Identifiers (URIs)

This section specifies general rules that apply to all uses of URIs in this standard.

## 6.1 URI prefix for Standard Vocabularies in the CBV

All URIs for standard vocabulary elements specified in the Core Business Vocabulary standard have the following syntax:

`urn:epcglobal:cbv:`*`qualifier`*`:`*`payload`*

where the *`qualifier`* denotes the type of the vocabulary the vocabulary element belongs to and *`payload`* the vocabulary element unambiguously identifies an element of the vocabulary.

## 6.2 Limitation on Use of the URI prefix

The Core Business Vocabulary standard is the only GS1 standard in which URIs beginning with `urn:epcglobal:cbv:` are defined.

A CBV-Compliant or CBV-Compatible document SHALL NOT use any URI beginning with `urn:epcglobal:cbv:` or `urn:epc:` except as specified in this standard.

Both CBV-Compliant and CBV-Compatible documents MAY contain URIs that do not begin with `urn:epcglobal:cbv:`, provided that the requirements specified elsewhere in this standard are met. These SHALL be used to identify vocabulary elements not defined by the CBV standard. URIs beginning with `urn:epcglobal:` SHALL NOT be used except as specified herein or in another GS1 standard.

### 6.2.1 Example of limitation of use of URI prefix (non-normative)

Suppose a user needs a new disposition value to stand for “quarantined.” The user may NOT use the following URI:

`urn:epcglobal:cbv:disp:quarantined`

In this case the particular URI above is NOT part of this standard and therefore may not be used. Instead a URI like the following could be used and considered CBV-Compatible. However, it must be noted that this vocabulary would have limited meaning to supply chain participants receiving this unless a prior understanding had been established.

`http://epcis.example.com/disp/quarantined`

# 7 Standard Vocabularies

This section specifies standard vocabulary elements for four EPCIS standard vocabularies: business steps, dispositions, business transaction types, and source/destination types.

## 7.1 Business steps

This section specifies standard identifiers for the EPCIS `BusinessStepID` vocabulary. These identifiers populate the `bizStep` field in an EPCIS event, as specified below.

### 7.1.1 URI structure

All business step values specified in this section have the following form:

urn:epcglobal:cbv:bizstep:*payload*

where the *payload* part is a string as specified in the next section. Every payload string defined herein contains only lower case letters and the underscore character.

### 7.1.2 Compliant usage

Each EPCIS event in a CBV-Compliant Document SHALL include a `bizStep` field, and the value of the `bizStep` field SHALL be a URI consisting of the prefix `urn:epcglobal:cbv:bizstep:` followed by the string specified in the first column of some row of the table in Section 7.1.3 below. The portion following the prefix SHALL be written exactly as specified in the table below, in all lowercase letters (possibly including underscores, as indicated).

Each EPCIS event in a CBV-Compatible Document MAY include a `bizStep` field, and the value of the `bizStep` field MAY be a URI as specified above for a CBV-Compliant document, and MAY be any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

#### 7.1.2.1 Example of correct and incorrect usage (non-normative)

The following shows an excerpt of a CBV-Compliant EPCIS document in XML format containing a single event, where the business step of that event is the Core Business Vocabulary “shipping” value:

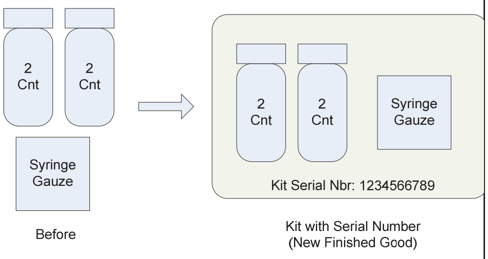
```
<epcis:EPCISDocument xmlns:epcis="urn:epcglobal:epcis:xsd:1" ...>
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        ...
        <bizStep>urn:epcglobal:cbv:bizstep:shipping</bizStep>
        ...
      </ObjectEvent>
    </EventList>
  </EPCISBody>
</epcis:EPCISDocument>
```

The following example is NOT CBV-Compliant, because it does not use the full URI string in the business step field. It is also not CBV-Compatible, because the value of the business step field is not a URI with an owning authority, as required by Section 6.4 of [EPCIS1.2].

```
<epcis:EPCISDocument xmlns:epcis="urn:epcglobal:epcis:xsd:1" ...>
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        ...
        <bizStep>shipping</bizStep>           WRONG
        ...
      </ObjectEvent>
    </EventList>
  </EPCISBody>
</epcis:EPCISDocument>
```

Additional samples may be found Section 11.1.

### 7.1.3 Element values and definitions – Business step

| Business steps | | |
|---|---|---|
| Value | Definition | Examples |
| `accepting` | Denotes a specific activity within a business process where an object changes possession and/or ownership. | Retailer X unloads a pallet on to the receiving dock. The numbers of cases on the pallet are counted. The pallets are disaggregated from the shipping conveyance. The quantity is verified against the delivery document (Freight Bill or Bill of Lading), notating any over, short or damaged product at the time of delivery. Typically this process releases freight payment and completes the contractual agreement with the carrier of delivering the product/assets to a specified location.<br>A parcel carrier drops off five boxes at Distributor Y's DC. A person on the Receiving Dock signs that they accept the five boxes from the parcel carrier.<br>A wholesaler is assigned a lot of fish at a fish auction, verifies the quantity and acknowledges receipt.<br>A manufacturer's fork lift driver scans the IDs of components which have been removed from a consignment warehouse. In doing so, the components are added to the manufacturer's inventory |
| `arriving` | Denotes a specific activity within a business process where an object arrives at a location. | Truckload of a shipment arrives into a yard. Shipment has not yet been received or accepted. |
| `assembling` | Denotes an activity within a business process whereby one or more objects are combined to create a new finished product.<br>In contrast to transformation, in the output of `assembling` the original objects are still recognisable and/or the process is reversible; hence, `assembling` would be used in an Aggregation Event, not a Transformation Event. | Computer parts (hard drive, battery, RAM) assembled into a consumer ready computer<br>Healthcare kitting: a surgical kit including drug, syringe, and gauze are combined to create a new 'product': a *kit*<br> 2 Cnt<br>2 Cnt<br>Syringe Gauze<br>Before<br>2 Cnt<br>2 Cnt<br>Syringe Gauze<br>Kit Serial Nbr: 1234566789<br>Kit with Serial Number (New Finished Good) |
| `collecting` | Denotes a specific activity within a business process where an object is picked up and collected for future disposal, recycling or re-used. | An organisation picks up disposed consumer electronics in an end of life state from various different organisations. After the goods are picked up, they typically are brought back and received into a Collection Centre<br>Rented or leased pallets are picked up and brought to a collection centre. |

| Business steps | | |
|---|---|---|
| commissioning | Process of associating an instance-level identifier (such as an EPC) with a specific object, or the process of associating a class-level identifier, not previously used, with one or more objects. A tag may have been encoded and applied in this step, or may have been previously encoded.<br>In the case of a class-level identifier, commissioning differs from creating_class_instance in that commissioning always indicates that this is the first use of the class-level identifier, whereas creating_class_instance does not specify whether the class-level identifier has been used before. | On a packaging line, an encoded EPC is applied to a case and associated to the product.<br>An individual virtual document (e.g. digital coupon, digital voucher, etc.) is assigned an EPC<br>One hundred bottles of a particular batch of pharmaceutical product are produced, those being the first bottles of that batch to be produced.<br>Sides of beef are transformed into individual packaged steaks. This may be an EPCIS 1.2 TransformationEvent if the input sides of beef are also tracked. |
| consigning | Indicates the overall process of staging_outbound, loading, departing, and accepting. It may be used when more granular process step information is unknown or inaccessible.<br>The use of consigning is mutually exclusive from the use of staging_outbound, loading, departing, and accepting.<br>Note: This business step is similar to shipping, but includes a change of possession and/or ownership at the outbound side. | A wholesaler comes aboard a fishing vessel, selects and buys boxes of fish, and brings them to his premises.<br>A manufacturer retrieves components from a consignment warehouse for use in its assembly line. In the logical second of leaving the consignment warehouse, the components pass into the ownership of the manufacturer.<br>A manufacturer stages products for loading, loads them into a container, the container is sealed, and the container departs. Ownership transfers to the receiver sometime during this overall process. If this is done in a single step, then business step consigning is used. |
| creating_class_instance | Denotes a step in a business process where an instance or increased quantity of a class-level identifier is produced. Unlike commissioning, this business step may be repeated for the same class-level identifier. | Water, sugar, and other ingredients are combined to produce a single batch of soda over a single shift on a single production line. This may be an EPCIS 1.2 TransformationEvent if the input ingredients are tracked.<br>Potatoes are sorted by size and quality, washed, and packed into cases of a single lot in a single packaging facility on a single date. |
| cycle_counting | Process of counting objects within a location in order to obtain an accurate inventory for business needs other than accounting purposes (e.g., replenishment and allocation). | A preselected subset of objects (for instance, all products belonging to a certain brand owner or a specific object class) within a retail store, are counted by a handheld reader.<br>All objects of a specific sub-location (sales floor or a shelf on the sales floor, e.g.) are counted by a handheld reader. |
| decommissioning | Process of disassociating an instance-level identifier (such as an EPC) with an object. The object may be re-commissioned at some point in the future – however only with a new instance-level identifier. | An eSeal on a reusable container is broken when the container is opened, so that the container is no longer identified by the instance-level identifier that was in the eSeal.<br>A digital coupon or an empties refund voucher is redeemed at retail point-of-sale |
| departing | Denotes a specific activity within a business process where an object leaves a location on its way to a destination. | Truckload of a shipment departs a yard, typically through a gate and begins transit to another location |

| Business steps | | |
|---|---|---|
| destroying | Process of terminating an object. For an instance-level identifier, the object should not be the subject of subsequent events; subsequent events are likely indicative of error (such as a stray read of a tag inside an incinerator). For a class level identifier, quantities are reduced; however, the class-level identifier may still be used in subsequent events (referring to different instances that were not destroyed). | Distributor or Retailer puts empty case in the incinerator or box crusher. |
| disassembling | Denotes a specific activity within a business process where an object is broken down into separate, uniquely identified component parts. | Before feeding a consumer electronics end of life item (a computer) into recycling operation line, it is necessary to disassemble the parts for the purpose of being recycled or disposed of in an environmentally sound manner.<br>A surgical kit (e.g. 2- 50 count bottles of medication and 1 syringe gauze) is broken down into its separate component parts |
| dispensing | Denotes a specific activity within a business process where a product is made available in full or part to a consumer. | A pharmacist dispenses a pharmaceutical to fill a specific prescription written by a physician, to a consumer or patient.<br>A deli manager slices a 5 pound package of turkey for sale. |
| encoding | Process of writing an instance-level identifier (typically an EPC) to a barcode or RFID tag, where the identifier is not yet associated with an object at this step in the process. | 3rd Party writes tags and returns spool of case tags to Manufacturer |
| entering_exiting | Denotes a specific activity at the Entrance/Exit door of a facility where customers are either leaving with purchased product or entering with product to be returned to the facility. | Customer leaves the facility of Retailer X with their purchased items through a customer entrance/exit door. |
| holding | Denotes a specific activity within a business process where an object is segregated for further review. | Retailer X unloads a second pallet on to their receiving dock, and, finding no purchase order for the pallet, moves the pallet to a holding area on the dock<br>Distributor Y obtains a shipment of pharmaceutical product. Distributor Y finds that their supplier cannot provide a complete pedigree. Distributor Y moves the shipment to a quarantine area on their dock.<br>Shipper Z is told by Customs to move a container to a special area until Customs can inspect and clear the container. |
| inspecting | Process of reviewing objects to address potential physical or documentation defects. | Manufacturer A pulls 10 bottles from every batch to ensure that the product and pill count in the bottles match expectations<br>Distributor Y checks all returned products to designate them either as saleable or as damaged<br>Regulator R pulls 3 bottles from a shelf to determine if the bottles have a correct pedigree<br>Customs Agent C uses a machine to scan the contents of a shipping container<br>Pallet pool operator Z checks if certain pallets comply with quality standards. |

| Business steps | | |
|---|---|---|
| installing | Denotes a specific activity within a business process where an object is put into a composite object (not merely a container).<br>In installing the composite object exists prior to this step, whereas in assembling the composite object is created during the step. | Additional memory chips and a rechargeable battery are installed within a computer<br>A duplexing unit is installed on a laser printer<br>Additional safety equipment is installed within the cabin of an aircraft or vehicle (e.g. fire extinguishers) |
| killing | Process of terminating an RFID tag previously associated with an object. The object and its instance-level identifier may continue to exist and be the subject of subsequent events (via a barcode, manual data entry, replacement tag, etc.). | Kill Command is issued to the tag to prevent any further reading of the tag or the information on the tag. |
| loading | Denotes a specific activity within a business process where an object is loaded into shipping conveyance. | Manufacturer A loads pallets into a container. The pallets are aggregated to the container.<br>Distributor Y loads racks full of totes on to a truck |
| other | A business step not identified by any of the values listed in the core business vocabulary. | "Other" may be used for terms that have yet to be added to the core business vocabulary from an industry or a user |
| packing | Denotes a specific activity within a business process that includes putting objects into a larger container – usually for shipping. Aggregation of one unit to another typically occurs at this point. | 12 packs of soda are placed into a case<br>Loose potatoes are placed into a tote. |
| picking | Denotes a specific activity within a business process that includes the selecting of objects to fill an order. | Distributor Y places three units into a tote to meet the requirements of a purchase order<br>Manufacturer A pulls three pallets from its racks to fulfil a purchase order |
| receiving | Denotes a specific activity within a business process that indicates that an object is being received at a location and is added to the receiver's inventory.<br>The use of receiving is mutually exclusive from the use of arriving and accepting. | Retailer X confirms that the count of cases on the pallet equals the expected count in a purchase order. Retailer X takes the cases into inventory. Typically, this process matches the product to the purchase order for payment to the supplier.<br>A shipment from a manufacturer factory site to manufacturer distribution centre, is matched against the transaction record then added to local inventory. |
| removing | Denotes a specific activity within a business process where an object is taken out of a composite object. | A defective airplane part is taken out of the engine |
| repackaging | Denotes a specific activity within a business process where an object's packaging configuration is changed. | Distributor Y receives one box full of batteries and another box full of laptops without batteries. Distributor Y ships out new boxes containing one laptop and one battery. |
| repairing | Denotes a specific activity within a business process where a malfunctioning product is repaired (typically by a post-sales service), without replacing it by a new one. | A computer is brought to a repair centre to fix a problem<br>An airplane part is in maintenance centre to diagnose an issue |
| replacing | Denotes a specific activity within a business process where an object is substituted or exchanged for another object. | A defective airplane part is replaced by a new part. |

| Business steps | | |
|---|---|---|
| reserving | Process in which a set of instance-level identifiers, not yet commissioned, are provided for use by another party. | Manufacturer provides set of case EPC numbers to a 3rd Party labeller |
| retail_selling | Denotes a specific activity within a business process at a point-of-sale for the purpose of transferring ownership to a customer in exchange for something of value (currency, credit, etc.). | Retailer X sells a screwdriver to a customer by checking it out through a point-of-sale system. |
| shipping | Indicates the overall process of staging_outbound, loading and departing. It may be used when more granular process step information is unknown or inaccessible. It may indicate a final event from a shipping point.<br>The use of shipping is mutually exclusive from the use of staging_outbound, departing, or loading. | Manufacturer A loads and reads product into the shipping container and closes the door. The product has been read out of the shipping facility. The shipment is immediately picked up and a BOL is associated at this point. (The shipment has left the yard)<br>At Distributor Y, the truck containing racks full of totes pulls away from the shipping dock or staging area.<br>Manufacturer A completes loading product into trailer and seals door. The trailer is ready for pickup. The generation of a Despatch Advice / ASN triggers a "shipping" event.<br>A 3PL picks and tags the product. The product is loaded into a trailer and signed over to a transportation carrier. The 3PL notifies the manufacturer who generates a "shipping" event. NOTE: This would be the case if there were NO departing step at a read point at the gate.<br>Typical Process flow:<br>staging_outbound<br>loading<br>departing<br>The above steps assume an organisation's ability and desire to share all steps in the process. If those process steps are not captured, the single business step of shipping would be used. |
| staging_outbound | Denotes a specific activity within a business process in which an object moves from a facility to an area where it will await transport pick-up. | Container is being closed and will be subsequently loaded onto a vehicle in the yard.<br>Container is being closed and seal is applied, and will be subsequently loaded onto a vehicle in the yard<br>Product has been picked and is now in a staging lane waiting for loading into a container |
| stock_taking | Process of counting objects within a location following established rules and/or standards to serve as a basis for accounting purposes. | All EPCs in a retail store are read by a handheld reader following a procedure accepted by the organisation's accounting firm. |
| stocking | Denotes a specific activity within a business process within a location to make an object available to the customer or for order fulfilment within a DC. | Retailer X places cans from a case on to a shelf on the sales floor<br>Dist X moves goods from a storage area to a picking area |
| storing | Denotes a specific activity within a business process where an object is moved into and out of storage within a location. | Manufacturer A moves a pallet from the receiving area to a rack<br>Retailer X moves a case from the receiving dock to a shelf in the backroom |

| Business steps | | |
|---|---|---|
| transforming<br>(**Deprecated**) | Denotes a specific activity within a business process where one or more objects are an input into a process that irreversibly changes that object / those objects into a new object or objects; the output has a new identity and characteristics.<br>This business step is **deprecated** for use with EPCIS 1.2. The EPCIS 1.2 standard has an event type, `TransformationEvent`, dedicated to transformations. The business steps `commissioning`, `creating_class_instance`, or other business steps may be used with `TransformationEvent`. | Meat packer X cuts a whole cow into two sides of beef (1 to many)<br>Food processor Y combines water, vegetables, and meat to create a unit of soup (many to one)<br>Butcher Z combines meat from multiple carcasses, grinds them together, and creates individual packages of ground beef (many to many) |
| transporting | Process of moving an object from one location to another using a vehicle (e.g., a ship, a train, a lorry, an aircraft). | Carrier X conveys 150 sea containers from Hong Kong seaport to Hamburg seaport with a container vessel.<br>A train with 20 goods wagons goes from one train station to another.<br>A lorry moves a swap trailer from a depot to a distribution centre. |
| unloading | Denotes a specific activity within a business process where an object is unloaded from a shipping conveyance. | Manufacturer A unloads pallets from a shipping conveyance. The pallets are disaggregated from the shipping conveyance.<br>Distributor Y unloads racks full of totes from a truck |
| unpacking | Denotes a specific activity within a business process that includes removing products (individuals, inners, cases, pallets) from a larger container – usually after receiving or accepting. Disaggregation of one unit from another typically occurs at this point. | 12 packs of soda are removed from a case<br>Loose potatoes are taken off from a tote. |
| void_shipping | Denotes a process of declaring that one or more objects in a prior outbound process (captured in an EPCIS event having business step `shipping`, `departing`, or `consigning`) were not shipped (or departed or consigned) as previously indicated. | A sender cancels a shipment after a prior shipping event.<br>A sender discovers, either by notification from a recipient or on their own, that a shipment they believed occurred and created a shipping event for, did not actually occur. The record is updated to reflect this.<br>A sender discovers that three out of ten items previously shipped were not included in the shipment. The `voidShipping` event indicates that those three items were not shipped. |

## 7.2 Dispositions

This section specifies standard identifier values for the EPCIS `DispositionID` vocabulary. These identifiers populate the `disposition` field in an EPCIS event, as specified below.

### 7.2.1 URI structure

All disposition values specified in this section have the following form:

urn:epcglobal:cbv:disp:*payload*

where the *payload* part is a string as specified in the next section. Every payload string defined herein contains only lower case letters and the underscore character.

### 7.2.2 Compliant usage

Each EPCIS event in a CBV-Compliant Document MAY include a `disposition` field. If the `disposition` field is present, the value of the `disposition` field SHALL be a URI consisting of the prefix `urn:epcglobal:cbv:disp:` followed by the string specified in the first column of some row of the table below. The portion following the prefix SHALL be written exactly as specified in the table below, in all lowercase letters (possibly including underscores, as indicated).

Each EPCIS event in a CBV-Compatible Document MAY include a `disposition` field, and the value of the `disposition` field MAY be a URI as specified above for a CBV-Compliant document, and MAY be any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4,* except for those URIs which in this standard are forbidden or designated for a different purpose.

#### 7.2.2.1 Example of correct and incorrect usage (non-normative)

The following shows an excerpt of a CBV-Compliant EPCIS document in XML format containing a single event, where the disposition of that event is the Core Business Vocabulary “in progress” value:

```
<epcis:EPCISDocument xmlns:epcis="urn:epcglobal:epcis:xsd:1" ...>
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        ...
        <disposition>urn:epcglobal:cbv:disp:in_progress</disposition>
        ...
      </ObjectEvent>
    </EventList>
  </EPCISBody>
</epcis:EPCISDocument>
```

The following example is NOT CBV-Compliant, because it does not use the full URI string in the disposition field. It is also not CBV-Compatible, because the value of the disposition field is not a URI with an owning authority, as required by *Section 6.4* of [EPCIS1.2].

```
<epcis:EPCISDocument xmlns:epcis="urn:epcglobal:epcis:xsd:1" ...>
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        ...
        <disposition>in_progress</disposition>         WRONG
        ...
      </ObjectEvent>
    </EventList>
  </EPCISBody>
</epcis:EPCISDocument>
```

Additional examples may found in *Section 11.1.*

### 7.2.3 Element Values and definitions – Dispositions

| Dispositions | | |
|---|---|---|
| Value | Definition | Examples |

| Dispositions | | |
|---|---|---|
| active | A commissioned object has just been introduced into the supply chain. | Manufacturer A commissions tags for 10 cases of product.<br>A virtual document has been assigned an EPC<br>Business step:<br>commissioning |
| container_closed | Object has been loaded onto a container, the doors have been closed and the shipment sealed. | Container is being closed and will be awaiting pickup in the yard.<br>Container is being closed and electronic seal is applied.<br>Business step:<br>staging_outbound |
| damaged | Object is impaired in its usefulness and/or reduced in value due to a defect. | Pallet pool operator P notices that a plank of a pallet is broken and records this incident by scanning the EPC of the pallet.<br>Retailer R receives a shipment where the product packages on the pallet have been dented<br>Business step:<br>accepting<br>inspecting<br>receiving<br>removing<br>repairing<br>replacing |
| destroyed | Object has been fully rendered non-usable. | Incinerator Operator B indicates that product and packaging have been incinerated<br>Business step:<br>destroying |
| dispensed | A full quantity of product is distributed to a consumer. | A pharmacist dispenses a pharmaceutical in a container's totality to fill a specific prescription written by a physician, to a consumer or patient.<br>A deli manager slices the complete contents of a 5 pound package of turkey for sale.<br>NOTE: this disposition reflects the disposition of the original object, not what was dispensed. |
| disposed | Object has been returned for disposal. | A package of pharmaceuticals has been picked up by a distributor and will be subsequently destroyed |
| encoded | An instance-level identifier has been written to a barcode or RFID tag, but not yet commissioned. | 3rd Party has written EPCs to tags and returns spool of case tags to Manufacturer<br>Business step:<br>encoding |
| expired | Object is past expiration date. | Distributor Y indicates that a product is past its expiration date<br>Business step:<br>holding<br>staging_outbound<br>storing |

| Dispositions | | |
|---|---|---|
| in_progress | Default disposition for object proceeding through points in the supply chain. | Product arrives at a location and is being accepted and verified.<br>Product is being prepared for shipment.<br>Business step:<br>receiving<br>picking<br>loading<br>accepting<br>staging_outbound<br>arriving<br>void_shipping |
| in_transit | Object being shipped between two trading partners. | Shipper Z pulled a container/product out of a manufacturer's yard on to a road<br>Business step:<br>shipping<br>departing |
| inactive | Decommissioned object that may be reintroduced to the supply chain. | A reusable tag is removed from a reusable transport item.<br>A digital coupon or an empties refund voucher has been redeemed at retail point-of-sale<br>Business step:<br>decommissioning |
| no_pedigree_match | In validating the pedigree for the object, no match was found, causing the product to be quarantined for further investigation and disposition. | Distributor Y could not obtain a valid pedigree for a product from its Manufacturer A<br>Business step:<br>holding<br>staging_outbound<br>storing |
| non_sellable_other | Object cannot be sold to a customer. | A product is not sellable pending further evaluation.<br>A product is not sellable, and one of the other dispositions (expired, recalled, damaged, no_pedigree_match) does not apply.<br>Product has been sold and is awaiting customer pick-up.<br>Business step:<br>holding<br>inspecting<br>staging_outbound<br>storing |
| partially_dispensed | A portion of a product is distributed to a customer, while additional product is retained for subsequent distribution. | A pharmacist dispenses 10 pills from a 100-count bottle to fill a customer prescription.<br>A deli manager slides and packages one pound of a 10 pound ham for customer purchase.<br>NOTE: This disposition reflects the disposition of the original object, not what was dispensed. |
| recalled | Object is non-sellable because of public safety reasons. | Manufacturer A requested that all Retailers and Distributors return its batteries that could overheat and explode<br>Business step:<br>holding<br>staging_outbound<br>storing |

| Dispositions | | |
|---|---|---|
| reserved | Instance-level identifier has been allocated for a third party. | Distributor receives EPC numbers and can encode tag with the numbers.<br>Business step:<br>reserving |
| retail_sold | Product has been purchased by a customer. | A customer at Retailer X purchased a screwdriver by checking it out through the point of sale system<br>Business step:<br>retail_selling |
| returned | Object has been sent back for various reasons. It may or may not be sellable. | Product is received at a returns centre from a customer because of an over-shipment, recall, expired product, etc.<br>Business step:<br>receiving<br>holding<br>shipping |
| sellable_accessible | Product can be sold as is and customer can access product for purchase. | Retailer X puts a case of screwdrivers on to a shelf or display within customer reach<br>Business step:<br>stocking<br>receiving |
| sellable_not_accessible | Product can be sold as is, but customer cannot access product for purchase. | Retailer X puts a case of screwdrivers on to a shelf in a store backroom<br>Business step:<br>receiving<br>storing<br>loading<br>holding<br>inspecting |
| stolen | An object has been taken without permission or right. | A pharmaceutical manufacturer completes an investigation of serial numbers that are missing from inventory, and concludes that they have been stolen |
| unknown | An object's condition is not known. | |

#### 7.2.3.1 CBV 1.0 Disposition Values deprecated in CBV 1.1

CBV 1.0 defined several disposition values that are deprecated in CBV 1.1. The following table lists the deprecated dispositions and the values which replace them in CBV 1.1. Each CBV 1.1 value applies to all the situations that the corresponding CBV 1.0 value did, but may also be applied to similar situations where the concept of "sellable" is not relevant. For example, in CBV 1.1 the disposition damaged may be applied to a returnable asset, which was never considered "sellable" even when it was undamaged.

| CBV 1.0 Disposition (deprecated) | CBV 1.1 Disposition |
|---|---|
| non_sellable_expired | expired |
| non_sellable_damaged | damaged |
| non_sellable_disposed | disposed |
| non_sellable_no_pedigree_match | no_pedigree_match |
| non_sellable_recalled | recalled |

## 7.3 Business Transaction Types

This section specifies standard identifier values for the EPCIS `BusinessTransactionTypeID` vocabulary. These identifiers may be used to populate the `type` attribute of a `bizTransaction` element in an EPCIS event. See *Section 8.5* for details of when these identifiers should be used.

### 7.3.1 URI structure

All business transaction type values specified in this section have the following form:

`urn:epcglobal:cbv:btt:`*`payload`*

where the *`payload`* part is a string as specified in the next section. Every payload string defined herein contains only lower case letters and the underscore character.

### 7.3.2 Compliant usage

Each EPCIS event in a CBV-Compliant Document MAY include one or more `bizTransaction` elements. If `bizTransaction` elements are present, each such element MAY include a `type` attribute. If a given `bizTransaction` element includes a `type` attribute, the value of the `type` attribute SHALL be a URI consisting of the prefix `urn:epcglobal:cbv:btt:` followed by the string specified in the first column of some row of the table below. The portion following the prefix SHALL be written exactly as specified in the table below, in all lowercase letters (possibly including underscores, as indicated). See *Section 8.5* for more compliance requirements concerning business transaction types.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in *Section 11.1*.

Each EPCIS event in a CBV-Compatible Document MAY include one or more `bizTransaction` elements. If `bizTransaction` elements are present, each such element MAY include a `type` attribute. If a given `bizTransaction` element includes a `type` attribute, the value of the `type` attribute MAY be a URI as specified above for a CBV-Compliant document, and MAY be any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4,* except for those URIs which in this standard are forbidden or designated for a different purpose.

### 7.3.3 Element Values and Definitions – Business Transaction Types

| Business Transaction Types | |
|---|---|
| **Value** | **Definition** |
| `bol` | **Bill of Lading**. A document issued by a carrier to a shipper, listing and acknowledging receipt of goods for transport and specifying terms of delivery |
| `desadv` | **Despatch Advice**. A document/message by means of which the seller or consignor informs the consignee about the despatch of goods. Also called an “Advanced Shipment Notice,” but the value `desadv` is always used regardless of local nomenclature. |
| `inv` | **Invoice**. A document/message claiming payment for goods or services supplied under conditions agreed by the seller and buyer. |
| `pedigree` | **Pedigree**. A record that traces the ownership or custody and transactions of a product as it moves among various trading partners. |
| `po` | **Purchase Order**. A document/message that specifies details for goods and services ordered under conditions agreed by the seller and buyer. |
| `poc` | **Purchase Order Confirmation**. A document that provides confirmation from an external supplier to the request of a purchaser to deliver a specified quantity of material, or perform a specified service, at a specified price within a specified time. |
| `prodorder` | **Production Order**. An organisation-internal document or message issued by a producer that initiates a manufacturing process of goods. |

| Business Transaction Types | |
|---|---|
| recadv | **Receiving Advice**. A document/message that provides the receiver of the shipment the capability to inform the shipper of actual goods received, compared to what was advised as being sent. |
| rma | **Return Merchandise Authorisation**. A document issued by the seller that authorises a buyer to return merchandise for credit determination. |

## 7.4 Source / Destination types

This section specifies standard identifier values for the EPCIS SourceDestTypeID vocabulary. These identifiers may be used to populate the type attribute of a source or destination element in an EPCIS event. See *Section 8.6* for details of when these identifiers should be used.

### 7.4.1 URI structure

All source/destination type values specified in this section have the following form:

urn:epcglobal:cbv:sdt:*payload*

where the *payload* part is a string as specified in the next section. Every payload string defined herein contains only lower case letters and the underscore character.

### 7.4.2 Compliant usage

Each EPCIS event in a CBV-Compliant Document MAY include one or more source and/or destination elements. The value of the type attribute of the source or destination element SHALL be a URI consisting of the prefix urn:epcglobal:cbv:sdt: followed by the string specified in the first column of some row of the table below. The portion following the prefix SHALL be written exactly as specified in the table in Section 7.4.3, in all lowercase letters (possibly including underscores, as indicated). See *Section 8.6* for more compliance requirements concerning source and destination types.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in *Section 11.1*.

Each EPCIS event in a CBV-Compatible Document MAY include one or more source and/or destination elements. The value of the type attribute of the source or destination element MAY be a URI as specified above for a CBV-Compliant document, and MAY be any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4*, except for those URIs which in this standard are forbidden or designated for a different purpose.

### 7.4.3 Element Values and Definitions – Source / Destination Types

| Source/Destination Types | |
|---|---|
| Value | Definition |
| owning_party | The source or destination identifier denotes the party who owns (or is intended to own) the objects at the originating endpoint or terminating endpoint (respectively) of the business transfer of which this EPCIS event is a part. |
| possessing_party | The source or destination identifier denotes the party who has (or is intended to have) physical possession of the objects at the originating endpoint or terminating endpoint (respectively) of the business transfer of which this EPCIS event is a part. |
| location | The source or destination identifier denotes the physical location of the originating endpoint or terminating endpoint (respectively) of the business transfer of which this EPCIS event is a part. When a source of this type is specified on an EPCIS event at the originating endpoint of a business transfer, the source identifier SHOULD be consistent with the Read Point specified in that event. When a destination of this type is specified on an EPCIS event at the terminating endpoint of a business transfer, the destination identifier SHOULD be consistent with the Read Point specified in that event. |

## 7.5 Error reason identifiers

This section specifies standard identifier values for the EPCIS ErrorReasonID vocabulary. These identifiers may be used to populate the reason attribute of an errorDeclaration element in an EPCIS event.

### 7.5.1 URI structure

All error reason identifier values specified in this section have the following form:

urn:epcglobal:cbv:er:*payload*

where the *payload* part is a string as specified in the next section. Every payload string defined herein contains only lower case letters and the underscore character.

### 7.5.2 Compliant usage

Each EPCIS event in a CBV-Compliant Document MAY include an ErrorDeclaration element, and when present, the ErrorDeclaration element MAY include a reason field. When present in a CBV-Compliant Document, the value of the reason field of the ErrorDeclaration element SHALL be a URI consisting of the prefix urn:epcglobal:cbv:er: followed by the string specified in the first column of some row of the table in Section 7.5.3. The portion following the prefix SHALL be written exactly as specified in the table below, in all lowercase letters (possibly including underscores, as indicated).

Each EPCIS event in a CBV-Compatible Document MAY include an ErrorDeclaration element, and when present, the ErrorDeclaration element MAY include a reason field. When present in a CBV-Compatible Document, the value of the reason attribute of the ErrorDeclaration element MAY be a URI as specified above for a CBV-Compliant document, and MAY be any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4*, except for those URIs which in this standard are forbidden or designated for a different purpose.

### 7.5.3 Element Values and Definitions – Error reason identifiers

| Error reason identifiers | |
|---|---|
| Value | Definition |
| did_not_occur | The prior event is considered erroneous because it did not actually occur. There are no corrective events. (In a CBV-Compliant Document, this error reason SHALL NOT be used in an error declaration that contains one or more corrective event IDs.) |
| incorrect_data | The prior event is considered erroneous because some or all of the data in the event are incorrect. Subsequent events may provide a correct indication of what actually occurred when the prior event was captured. These events may be linked using the corrective event IDs in the error declaration. |

# 8 User vocabularies

This section specifies syntax templates that end users may use to define vocabulary elements for three EPCIS user vocabularies: physical or digital objects, locations (both read points and business locations), and business transactions.

## 8.1 General considerations

Unlike the standard vocabularies discussed in *Section 7*, a vocabulary element in a User Vocabulary is created by an End User. For example, an End User who creates a new business location such as a new warehouse may create a business location identifier to refer to that location in EPCIS events. The specific identifier string is defined by the End User, and its meaning may be described to trading partners via master data exchange, or via some other mechanism outside of the EPCIS Query Interface.

The EPCIS standard (*Section 6.4)* places general constraints on the identifiers that End Users may create for use as User Vocabulary elements. Specifically, an identifier must conform to URI syntax, and must either conform to syntax specified in GS1 standards or must belong to a subspace of URI identifiers that is under the control of the end user who assigns them.

The Core Business Vocabulary provides additional constraints on the syntax of identifiers for user vocabularies, so that CBV-Compliant documents will use identifiers that have a predictable structure. This in turn makes it easier for trading partners to understand the meaning of such identifiers.

For each user vocabulary considered here, several different syntax templates are provided for constructing vocabulary elements:

- **EPC URI**: An Electronic Product Code "pure identity" URI may be used as a user vocabulary element. EPCs have a structure and meaning that is widely understood. EPCs may also be encoded into data carriers such as RFID tags and barcodes according to GS1 standards. For this reason, EPCs are often the best choice for creating user vocabulary elements when it is possible to do so.
- **Private or Industry-wide URN**: A Uniform Resource Name (URN) of the form `urn:`*`URNNamespace`*`:…` may be used as a user vocabulary element. Doing so requires that the user who creates the vocabulary element be authorised to use the URN namespace that appears following the `urn:` prefix. For example, the End User may register its own URN namespace with the Internet Assigned Numbers Authority (IANA). Alternatively, an industry consortium or other trading group could register a URN namespace, and define a syntax template beginning with this namespace for use by its members in creating vocabulary elements. Because of the difficulty of registering a URN namespace, this method is typically used by trading groups, not individual end users.
- **HTTP URL**: A Uniform Resource Locator (URL) of the form

  `http://`*`Domain`*`/…` may be used as a user vocabulary element. Doing so requires that the user who creates the vocabulary element be authorised to use the Internet domain name that appears following the `http:` prefix. Often a subdomain of the End User's organisation domain is used; for example, the Example Corporation may choose to use `epcis.example.com` as a domain name for constructing user vocabulary identifiers. Because registering an Internet domain name is relatively easy, this method is quite appropriate for use by individual end users as well as by industry groups.

  Note that HTTP URLs used as EPCIS user vocabulary elements do not necessarily refer to a web page. They are just identifiers (names) that happen to use the HTTP URI scheme for the sake of convenience.

  Further details about each of these three forms are specified below.

**Non-Normative**: Explanation: The reason that several different syntax templates are provided for each user vocabulary is to provide flexibility for end users to meet their business requirements. Use of an EPC is preferred for most end user vocabularies; however, EPC codes are somewhat constrained in syntax (e.g., limitations on character set and number of characters allowed), and may not easily accommodate the construction of identifiers based on codes already in use within legacy business systems. The other forms provide an alternative.

### 8.1.1 General Considerations for EPC URIs as User Vocabulary Elements

Where an EPC URI is used as a User Vocabulary Element, both CBV-Compliant and CBV-Compatible documents SHALL use an EPC Pure Identity URI, except as noted below. An EPC Pure Identity URI is a URI as specified in [TDS1.9], *Section 6* (specifically, a URI matching the grammar production EPC-URI in [TDS1.9], *Section 6.3*). EPC "pure identity" URIs begin with `urn:epc:id:…`.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]. In particular, documents SHALL NOT use EPC Tag URIs (`urn:epc:tag:…`) , EPC Pure Identity Pattern URIs (`urn:epc:idpat:…`), or EPC Pattern URIs (`urn:epc:pat:…`), except that both CBV-Compliant and CBV-Compatible documents MAY use EPC Pattern URIs for class-level identification of objects as specified in *Section 8.3.1*. Both CBV-

Compliant and CBV-Compatible documents MAY use EPC Raw URIs (`urn:epc:raw:…`) as defined in [TDS1.9], *Section 12*, provided that the raw value cannot be decoded as an EPC. Both CBV-Compliant and CBV-Compatible documents SHALL NOT use an EPC Raw URI representing EPC memory bank contents that could be successfully decoded into an EPC Pure Identity URI according to [TDS1.9].

**Non-Normative**: Explanation: [EPCIS1.2] specifies that "When the unique identity [for an instance-level identifier in the "what" dimension] is an Electronic Product Code, the [identifier] SHALL be the "pure identity" URI for the EPC as specified in [TDS1.9], *Section 6*. Implementations MAY accept URI-formatted identifiers other than EPCs." The above language clarifies this requirement, and provides more specific references to [TDS1.9]. The above language also extends these restrictions to the use of EPC URIs in other dimensions of EPCIS events beyond the "what" dimension.

### 8.1.2 General Considerations for Private or Industry-wide URN as User Vocabulary elements

Where specified in *Sections 8.2* through *8.5*, a CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below.

A Private or Industry-wide URN SHALL have the following form:

`urn:`*`URNNamespace`*`:**:`*`qual`*`:`*`Remainder`*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `urn:` | The characters u, r, n, and : (colon). |
| *URNNamespace* | A URN Namespace registered with the Internet Assigned Numbers Authority according to [RFC2141]. |
| `:**:` | Denotes either a single colon character or any string that conforms to the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace, and which begins and ends with a colon character. In other words, any number of additional subfields may be included between the URN Namespace and the *qual* component, in order to provide flexibility for URN Namespace owners to administer their namespace. |
| *qual*: | A qualifier as specified in *Sections 8.2* through *8.5*, depending on the type of identifier. |
| *Remainder* | The remainder of the identifier as specified in *Sections 8.2* through *8.5.* |

In addition, an identifier of this form SHALL be 128 characters or fewer, and SHOULD be 60 characters or fewer.

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

### 8.1.3 General Considerations for HTTP URLs as User Vocabulary elements

Where specified in *Sections 8.2* through *8.5*, a CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL.

An HTTP URL SHALL have the following form:

`http://[`*`Subdomain.`*`]`*`Domain`*`/**/`*`qual`*`/`*`Remainder`*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `http://` | The seven characters h, t, t, p, : (colon), / (slash), and / (slash). |

| Template Component | Description |
|---|---|
| *[Subdomain.]Domain* | An Internet Domain name that has been registered with an Internet Domain Name Registrar, optionally preceded by one or more subdomain names.<br>For example, if example.com is a registred Internet Domain Name, then the following are acceptable values for this component:<br>example.com<br>epcis.example.com<br>a.rather.verbose.example.com<br>Unless there is a reason to do otherwise, epcis.example.com is recommended for most End Users (where the End User substitutes its own company or organisational Domain Name for example.com).<br>Explanation (non-normative): Use of a subdomain dedicated to EPCIS, such as epcis.example.com, helps to avoid the possibility of conflict with other uses of the company or organisational domain name, such as URLs of web pages on the company web site. While HTTP URLs used as identifiers in EPCIS events are not usually intended to be dereferenced via a web browser, it is usually helpful to emphasise this fact by making the URL distinct from the URLs used by the company web site. |
| */**/* | Denotes either a single slash character, or any string that matches the grammar rule path-absolute defined in [RFC3986], *Section 3.3*. In other words, any number of additional path components may be included between the authority component and the obj component, in order to provide flexibility for domain owners to administer their namespace. |
| *qual/* | A qualifier as specified in *Sections 8.2* through *8.5*, depending on the type of identifier. |
| *Remainder* | The remainder of the identifier as specified in *Sections 8.2* through *8.5*. |

In addition, an identifier of this form SHALL be 128 characters or fewer, and SHOULD be 60 characters or fewer.

Identifiers of this form must be assigned by the owner of the Internet domain *Domain*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

## 8.2 Physical or digital objects (Instance-Level Identification)

Instance-level identifiers for physical or digital objects populate the “what” dimension of EPCIS events. This includes the epcList, parentID, childEPCs, inputEPCs, and outputEPCs fields in EPCIS ObjectEvents, AggregationEvents, TransacationEvents, and TransformationEvents. See *Section 1* of [EPCIS1.2] for a further definition of “object” in this sense, also reproduced below.

A CBV-Compliant document SHALL use one of the three URI forms specified in this section to populate the above fields of EPCIS events, for every such field that is not null. A CBV-Compatible document MAY use one of the three URI forms specified in this section, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4*, except for those URIs which in this standard are forbidden or designated for a different purpose.

Both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form as specified in *Section 8.2.1* unless there is a strong reason to do otherwise.

**Non-Normative**: Explanation, quoted from [EPCIS1.2]: “Objects” in the context of EPCIS typically refers to physical objects that are identified either at a class or instance level and which are handled in physical handling steps of an overall business process involving one or more organisations. Examples of such physical objects include trade items (products), logistic units, returnable assets, fixed assets, physical documents, etc. “Objects” may also refer to digital objects, also identified at either a class or instance level, which participate in comparable business process steps. Examples of such digital objects include digital trade items (music downloads, electronic books, etc.), digital documents (electronic coupons, etc.), and so forth. Throughout this document the word “object” is used to denote a physical or digital object, identified at a class or instance level, that is the subject of a business process step. *Section 8.2* of this CBV standard defines identifier structures for instance-level identification of Objects; *Section 8.3* defines identifier structures for class-level identification of Objects.

### 8.2.1 EPC URI for Instance-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use an EPC Pure Identity URI as specified in *Section 8.1.1* to populate the epcList, parentID, and childEPCs fields in EPCIS ObjectEvents, AggregationEvents, and TransacationEvents. Both CBV-Compliant and CBV-Compatible documents SHOULD use this form unless there is a strong reason to do otherwise.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use an SGLN EPC (urn:epc:id:sgln:…) as an Object identifier.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see *Section 8.1.1 for details*.

### 8.2.2 Private or Industry-wide URN for Instance-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below to populate the epcList, parentID, and childEPCs fields in EPCIS ObjectEvents, AggregationEvents, and TransacationEvents. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (*Section 8.2.1)* unless there is a strong reason to do otherwise. See *Section 8.1* for general considerations regarding the use of Private or Industry-wide URI identifiers.

A Private or Industry-wide URI suitable for populating the epcList, parentID, and childEPCs fields of EPCIS events SHALL have the following form:

urn:*URNNamespace*:**:obj:*Objid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| urn:*URNNamespace*:**: | As specified in *Section 8.1.2.* |
| obj: | The characters o, b, j, and : (colon). |
| *Objid* | An identifier for the object that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace *URNNamespace*, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

ⓘ **Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in *Section 11.2.*

### 8.2.3 HTTP URLs for Instance-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below to populate the epcList, parentID, and childEPCs fields in EPCIS ObjectEvents, AggregationEvents, and TransacationEvents. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (*Section 8.2.1)* unless there is a strong reason to do otherwise. See *Section 8.1* for general considerations regarding the use of HTTP URL identifiers.

An HTTP URL suitable for populating the epcList, parentID, and childEPCs fields of EPCIS events SHALL have the following form:

http://[*Subdomain*.]*Domain*/**/obj/*Objid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `http://[Subdomain.]Domain/**/` | As specified in *Section 8.1.3*. |
| `obj/` | The characters `o`, `b`, `j`, and `/` (slash). |
| *Objid* | An identifier for the object that matches the grammar rule `segment-nz` defined in [RFC3986], *Section 3.3* (among other things, this means `Objid` may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain *Domain*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in *Section 11.2.*

## 8.3 Physical or digital objects (Class-level identification)

Class-level identifiers for physical or digital objects populate the "what" dimension of EPCIS events. This includes the `epcClass` field within the EPCIS `QuantityEvent` (deprecated in EPCIS 1.1) and within the `quantityElement` structures of EPCIS `ObjectEvents`, `AggregationEvents`, `TransacationEvents`, and `TransformationEvents`. See *Section 1* of [EPCIS1.2] for a further definition of "object" in this sense, also reproduced below.

A CBV-Compliant document SHALL use one of the three URI forms specified in this section to populate the above fields of EPCIS events, for every such field that is not null. A CBV-Compatible document MAY use one of the three URI forms specified in this section, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], *Section 6.4,* except for those URIs which in this standard are forbidden or designated for a different purpose.

Both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form as specified in *Section 8.3.1* unless there is a strong reason to do otherwise.

**Non-Normative**: Explanation (non-normative), quoted from [EPCIS1.2]: "Objects" in the context of EPCIS typically refers to physical objects that are identified either at a class or instance level and which are handled in physical handling steps of an overall business process involving one or more organisations. Examples of such physical objects include trade items (products), logistic units, returnable assets, fixed assets, physical documents, etc. "Objects" may also refer to digital objects, also identified at either a class or instance level, which participate in comparable business process steps. Examples of such digital objects include digital trade items (music downloads, electronic books, etc.), digital documents (electronic coupons, etc.), and so forth. Throughout this document the word "object" is used to denote a physical or digital object, identified at a class or instance level, that is the subject of a business process step. *Section 8.2* of this CBV standard defines identifier structures for instance-level identification of Objects; *Section 8.3* defines identifier structures for class-level identification of Objects.

### 8.3.1 EPC URI for Class-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use one of the following URI forms specified in the EPC Tag Data standard to populate the `epcClass` field within the EPCIS `QuantityEvent` (deprecated in EPCIS 1.1) and within the `quantityElement` structures of EPCIS `ObjectEvents`, `AggregationEvents`, `TransacationEvents`, and `TransformationEvents`:

| Identifier Type | URI Form | Normative Reference |
|---|---|---|
| GTIN | `urn:epc:idpat:sgtin:`*`CCC.III`*`.*` | [*TDS1.9, Section 8*] |

| Identifier Type | URI Form | Normative Reference |
|---|---|---|
| GTIN+batch/lot | urn:epc:class:lgtin:*CCC.III.LLL* | [*TDS1.9, Section 6*] |
| GRAI (no serial) | urn:epc:idpat:grai:*CCC.TTT*.* | [*TDS1.9, Section 8*] |
| GDTI (no serial) | urn:epc:idpat:gdti:*CCC.TTT*.* | [*TDS1.9, Section 8*] |
| GCN (no serial) | urn:epc:idpat:sgcn:*CCC.TTT*.* | [*TDS1.9, Section 8*] |
| CPI (no serial) | urn:epc:idpat:cpi:*CCC.TTT*.* | [*TDS1.9, Section 8*] |

where:

- *CCC* is the GS1 Company Prefix portion of an EPC Pure Identity Pattern URI
- *III* is the Indicator + Item Reference portion of an SGTIN EPC Pure Identity Pattern URI or the Indicator + Item Reference portion of an LGTIN EPC Class URI
- *TTT* is the Returnable Asset Type, Document Type, Coupon Reference, or Component/Part Type portion of an EPC Pure Identity Pattern for GRAI, GDTI, SGCN, or CPI, respectively.

A CBV-Compliant document or CBV-Compatible document SHALL NOT use any other Pure Identity Pattern URI form specified in [TDS1.9, *Section 8*]. This includes, for example, an SSCC Pure Identity Pattern URI, or an SGTIN Pure Identity Pattern URI with two "*" wildcards.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see *Section 8.1.1* for details.

#### 8.3.1.1 Explanation (non-normative)

The EPC Tag Data standard defines EPC Pure Identity Pattern URIs as a way to specify a pattern that matches many instance-level EPCs. For example, the EPC Pure Identity Pattern URI urn:epc:idpat:sgtin:0614141.112345.* matches any SGTIN URI that begins with urn:epc:idpat:sgtin:0614141.112345, for example the specific SGTIN URI urn:epc:idpat:sgtin:0614141.112345.400. In the EPCIS Simple Event Query, such a pattern may be used to match EPCIS events whose "what" dimension contains instance-level identifiers that have a specified GTIN and any serial number.

The table above specifies the use of EPC Pure Identity Pattern URIs to achieve a second purpose, namely as class-level identifiers for use in the Quantity Element fields of EPCIS events. In this usage, the URI urn:epc:idpat:sgtin:0614141.112345.* refers to the object class identified by GTIN 10614141123459.

Not all EPC Pure Identity Pattern URIs make sense as class-level identifiers. For example, when urn:epc:idpat:sgtin:0614141.*.* is used in an EPCIS query to match instance-level identifiers, it matches all SGTIN identifiers that include GS1 Company Prefix 0614141. This is valid as a matching condition for a query, but there is no corresponding object class and so this is not a valid class-level identifier. A similar argument applies to a URI such as urn:epc:idpat:sscc:0614141.*, and the other EPC Pattern URIs not included in the table above.

### 8.3.2 Private or Industry-wide URN for Class-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below to populate the epcClass field within the EPCIS QuantityEvent (deprecated in EPCIS 1.1) and within the quantityElement structures of EPCIS ObjectEvents, AggregationEvents, TransacationEvents, and TransformationEvents. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (*Section 8.3.1*) unless there is a strong reason to do otherwise. See *Section 8.1* for general considerations regarding the use of Private or Industry-wide URI identifiers.

A Private or Industry-wide URI suitable for populating the epcClass field of EPCIS events SHALL have the following form:

urn:*URNNamespace*:**:class:*ObjClassid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| urn:*URNNamespace*:**: | As specified in Section 8.1.2. |
| class: | The characters c, l, a, s, s, and : (colon). |
| *ObjClassid* | An identifier for the object class that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace *URNNamespace*, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

### 8.3.3 HTTP URLs for Class-level identification of objects

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below to populate the epcClass field within the EPCIS QuantityEvent (deprecated in EPCIS 1.1) and within the quantityElement structures of EPCIS ObjectEvents, AggregationEvents, TransacationEvents, and TransformationEvents. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (Section 0) unless there is a strong reason to do otherwise. See Section 8.1 for general considerations regarding the use of HTTP URL identifiers.

An HTTP URL suitable for populating the epcClass fields of EPCIS events SHALL have the following form:

http://[*Subdomain*.]*Domain*/**/class/*ObjClassid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| http://[*Subdomain*.]*Domain*/**/ | As specified in Section 8.1.3. |
| class/ | The characters c, l, a, s, s, and / (slash). |
| *ObjClassid* | An identifier for the object class that matches the grammar rule segment-nz defined in [RFC3986], Section 3.3 (among other things, this means ObjClassid may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain *Domain*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example: An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

## 8.4 Locations

Identifiers for locations populate the “where” dimension of EPCIS events. This includes the readPoint and businessLocation fields in all EPCIS event types.

A CBV-Compliant document SHALL use one of the four URI forms specified in this section to populate the above fields of EPCIS events, for every such field that is not null. A CBV-Compatible document MAY use one of the four URI forms specified in this section, or MAY any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

Both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form as specified in Section 8.4.1 unless there is a strong reason to do otherwise.

### 8.4.1 EPC URI for Location identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an EPC Pure Identity URI as specified in Section 8.1.1 to populate the readPoint and businessLocation fields in all EPCIS event types. Both CBV-Compliant and CBV-Compatible documents SHOULD use this form unless there is a strong reason to do otherwise.

Both CBV-Compliant and CBV-Compatible documents SHOULD NOT use EPC schemes other than SGLN EPCs (urn:epc:id:sgln:…) for location identifiers, unless there is a strong reason to do so.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see Section 8.1.1 for details.

### 8.4.2 Private or Industry-wide URN for Location identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below to populate the readPoint and businessLocation fields in all EPCIS event types. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (Section 8.4.1) unless there is a strong reason to do otherwise. See Section 8.1 for general considerations regarding the use of Private or Industry-wide URI identifiers.

A Private or Industry-wide URI suitable for populating the readPoint and businessLocation fields in all EPCIS event types SHALL have the following form:

urn:*URNNamespace*:**:loc:*Locid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| urn:*URNNamespace*:**: | As specified in Section 8.1.2. |
| loc: | The characters l, o, c, and: (colon). |
| *Locid* | An identifier for the location that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace *URNNamespace*, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

### 8.4.3 HTTP URLs for Location identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below to populate the readPoint and businessLocation fields in all EPCIS event types. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (Section 8.4.1) unless there is a strong reason to do otherwise. See Section 8.1 for general considerations regarding the use of HTTP URL identifiers.

An HTTP URL suitable for populating the readPoint and businessLocation fields in all EPCIS event types SHALL have the following form:

http://[*Subdomain*.]*Domain*/**/loc/*Locid*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `http://[Subdomain.]Domain/**/` | As specified in Section 8.1.3. |
| `loc/` | The characters `l`, `o`, `c`, and `/` (slash). |
| *`Locid`* | An identifier for the location that matches the grammar rule segment-nz defined in [RFC3986], Section 3.3 (among other things, this means Locid may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain *`Domain`*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

### 8.4.4 Geographic Location URIs for Location identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a geographic location URI as specified in [RFC5870] to populate the `readPoint` and `businessLocation` fields in all EPCIS event types. Such identifiers may be used in situations where it is not feasible to assign a unique location identifier; for example, to indicate the location of a ship on the open ocean. Both CBV-Compliant and CBV-Compatible documents SHOULD use a location identifier as specified in Sections 8.4.1 through 8.4.3 (with preference given to the EPC URI form as specified in Section 8.4.1) unless a geographic location URI is the only feasible alternative.

The syntax and meaning of geographic location URIs is specified in [RFC5870].

**Non-Normative**: Explanation (non-normative): The simplest form of RFC5870-compliant geographic location URI looks like this: `geo:22.300,-118.44`
This example denotes the geographic location with latitude 22.300 degrees (north) and longitude 118.44 degrees (west). Other forms of the geo URI allow for the inclusion of altitude, uncertainty radius, and reference coordinate system. Please consult [RFC5870] for details of these and other considerations that apply to the use of the geographic location URI.

## 8.5 Business transactions

Identifiers for business transactions populate the “why” dimension of EPCIS events. This includes the `bizTrasactionList` field in all EPCIS event types.

The EPCIS standard provides for a business transaction to be identified by a pair of identifiers, the “business transaction identifier” (hereinafter “BTI”) that names a particular business transaction, and an optional “business transaction type” (hereinafter “BTT”) that says what kind of business transaction the identifier denotes (purchase order, invoice, etc.). Section 7.3 of this standard provides standardised values for BTTs.

URI forms for BTIs are specified below. A CBV-Compliant document SHALL use one of the four URI forms specified in this section to populate the BTI field (text content of the `bizTransaction` element) of EPCIS events, for every such field that is not null. A CBV-Compatible document MAY use one of the four URI forms specified in this section, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

A `bizTransaction` element in an EPCIS event includes a BTI and an optional BTT in any of the following three combinations:

- If the goal is to communicate a business transaction identifier without indicating its type, a BTI is included and the BTT omitted.

- If the goal is to communicate a business transaction identifier and to indicate its type, and furthermore the type is one of the CBV standard types specified in Section 7.3, a BTI is included, and one of the URIs specified in Section 7.3 is included as the BTT.
- If the goal is to communicate a business transaction identifier and to indicate its type, and furthermore the type is not one of the CBV standard types specified in Section 7.3, the BTI is included, and some URI that does not begin with `urn:epcglobal:cbv:…` is included as the BTT. (This is CBV-Compatible but not CBV-Compliant.)

### 8.5.1 EPC URI for Business transaction identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an EPC Pure Identity URI as specified in Section 8.1.1 as a business transaction identifier in all EPCIS event types.

Both CBV-Compliant and CBV-Compatible documents SHOULD NOT use EPC schemes other than GDTI EPCs (`urn:epc:id:gdti:…`) or GSRN EPCs (`urn:epc:id:gsrn:…`) for business transaction identifiers, unless there is a strong reason to do so. GDTI EPCs SHOULD only be used as business transaction identifiers when they have been assigned to denote a business transaction, rather than a physical document not connected with any business transaction.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see Section 8.1.1 for details.

**Non-Normative**: Explanation (non-normative): One of the intended uses of the Global Document Type Identifier (GDTI) is to identify business transactions such as invoices, purchase orders, and so on. When a GDTI is used in this way, it is suitable for use as a business transaction identifier in EPCIS. However, many business information systems use other types of identifiers for business transactions, and so the use of GDTI is not as strongly recommended as SGLNs are for locations or other types of EPCs are for physical or digital objects. It is also for this reason that the form in Section 8.5.2 is provided.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.1.

### 8.5.2 GLN-based identifier for legacy system business transaction identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a GLN-based identifier as specified below as a business transaction identifier in all EPCIS event types.

A GLN-based URI suitable for use as a business transaction identifier in all EPCIS event types SHALL have the following form: `urn:epcglobal:cbv:bt:`*`gln`*`:`*`transID`* where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `urn:epcglobal:cbv:bt:` | The 21 characters `u`, `r`, `n`, …, `b`, `t`, and `:` (colon). |
| *`gln`*`:` | A 13-digit Global Location Number (GLN) that identifies the business system within which *`transID`* is defined, followed by a colon. This is typically a “party GLN” that identifies the organisation responsible for the business transaction identifier, or a division of an organisation that maintains a separate divisional business information system. |
| *`transID`* | An identifier for the business transaction that complies with the requirements of [RFC2141] and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the GLN that is embedded in the identifier. The owner of the GLN may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

### 8.5.3 Private or Industry-wide URN for business transaction identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below as a business transaction identifier in all EPCIS event types.

A private or industry-wide URN suitable for use as a business transaction identifier in all EPCIS event types SHALL have the following form: `urn:`*`URNNamespace`*`:**:bt:`*`transID`* where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `urn:`*`URNNamespace`*`:**:` | As specified in Section 8.1.2. |
| `bt:` | The characters `b`, `t`, and `:` (colon). |
| *`transID`* | An identifier for the business transaction that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace *URNNamespace*, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2

### 8.5.4 HTTP URLs for business transaction identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below as a business transaction identifier in all EPCIS event types.

An HTTP URL suitable for use as a business transaction identifier in all EPCIS event types SHALL have the following form: *http://[Subdomain.]Domain/**/bt/transID* where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `http://[`*`Subdomain.`*`]`*`Domain`*`/**/` | As specified in Section 8.1.3. |
| `bt/` | The characters `b`, `t`, and `/` (slash). |
| *`transID`* | An identifier for the business transaction that matches the grammar rule `segment-nz` defined in [RFC3986], Section 3.3 (among other things, this means `transID` may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain *`Domain`*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example (non-normative): An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

## 8.6 Source / Destination identifiers

Identifiers for sources and destinations populate the `source` and `destination` elements (respectively) in the "why" dimension of EPCIS events.

A CBV-Compliant document SHALL use one of the three URI forms specified in this section to populate the above fields of EPCIS events. A CBV-Compatible document MAY use one of the three URI forms specified in this section, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

Both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form as specified in Section 8.6.1 unless there is a strong reason to do otherwise.

### 8.6.1 EPC URI for Source / Destination identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an EPC Pure Identity URI as specified in Section 8.1.1 to populate the source and destination elements in all EPCIS event types. Both CBV-Compliant and CBV-Compatible documents SHOULD use this form unless there is a strong reason to do otherwise.

Both CBV-Compliant and CBV-Compatible documents SHOULD NOT use EPC schemes other than SGLN EPCs (`urn:epc:id:sgln:…`) for source and destination identifiers, unless there is a strong reason to do so.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see Section 8.1.1 for details.

### 8.6.2 Private or Industry-wide URN for Source / Destination identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below, or a private or industry-wide URN as specified in Section 8.4.2, to populate the `source` and `destination` fields in all EPCIS event types. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (Section 8.6.1) unless there is a strong reason to do otherwise. See Section 8.1 for general considerations regarding the use of Private or Industry-wide URI identifiers.

In addition to the private or industry-wide URN form as specified in Section 8.4.2, a Private or Industry-wide URI suitable for populating the `source` and `destination` fields in all EPCIS event types SHALL have the following form: `urn:URNNamespace:**:sd:Locid` where the components of this template are as follows:

| Template Component | Description |
|---|---|
| urn:*URNNamespace*:**: | As specified in Section 8.1.2. |
| sd: | The characters s, d, and : (colon). |
| *Locid* | An identifier for the location that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace *URNNamespace*, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

### 8.6.3 HTTP URLs for Source / Destination identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below, or an HTTP URL as specified in Section 8.4.3, to populate the `source` and `destination` fields in all EPCIS event types. However, both CBV-Compliant and CBV-Compatible documents SHOULD use the EPC URI form (Section 8.6.1) unless there is a strong reason to do otherwise. See Section 8.1 for general considerations regarding the use of HTTP URL identifiers.

In addition to the HTTP URL form as specified in Section 8.4.3, an HTTP URL suitable for populating the `source` and `destination` fields in all EPCIS event types SHALL have the following form:

`http://[Subdomain.]Domain/**/sd/SourceOrDestId`

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `http://[Subdomain.]Domain/**/` | As specified in Section 8.1.3. |
| `sd/` | The characters s, d, and / (slash). |
| *`SourceOrDestId`* | An identifier for the location that matches the grammar rule segment-nz defined in [RFC3986], Section 3.3 (among other things, this means Locid may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain *Domain*. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

## 8.7 Transformation identifiers

Identifiers for transformations populate the `transformationID` field of EPCIS `TransformationEvents`.

URI forms for transformation identifiers are specified below. A CBV-Compliant document SHALL use one of the four URI forms specified in this section to populate the `transformationID` field of EPCIS `TransformationEvents`, for every such field that is not null. A CBV-Compatible document MAY use one of the four URI forms specified in this section, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

### 8.7.1 EPC URI for Transformation identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an EPC Pure Identity URI as specified in *Section 8.1.1* to populate the `transformationID` field of EPCIS `TransformationEvents`.

Both CBV-Compliant and CBV-Compatible documents SHOULD NOT use EPC schemes other than GDTI EPCs (`urn:epc:id:gdti:…`) for transformation identifiers unless there is a strong reason to do so. GDTI EPCs SHOULD only be used as transformation identifiers when they have been assigned to denote a transformation, rather than a physical document not connected with any transformation.

Both CBV-Compliant and CBV-Compatible documents SHALL NOT use any of the other URI forms for EPCs defined in [TDS1.9]; see *Section 8.1.1* for details.

**Non-Normative**: Explanation: One of the intended uses of the Global Document Type Identifier (GDTI) is to identify business transactions such as production orders which may be in one-to-one correspondence with transformations. When a GDTI is used in this way, it is suitable for use as a transformation identifier in EPCIS. However, many business information systems use other types of identifiers for transformations, and so the use of GDTI is not as strongly recommended as SGLNs are for locations or other types of EPCs are for physical or digital objects. It is also for this reason that the form in *Section 8.7.2* is provided.

### 8.7.2 GLN-based Identifier for Legacy System Transformation identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a GLN-based identifier as specified below *8.1.1* to populate the `transformationID` field of EPCIS `TransformationEvents`.

A GLN-based URI SHALL have the following form: `urn:epcglobal:cbv:xform:`*`gln`*`:`*`xformID`* where the components of this template are as follows:

| Template Component | Description |
|---|---|
| `urn:epcglobal:cbv:xform:` | The 24 characters u, r, n, …, r, m, and : (colon). |
| *`gln`*`:` | A 13-digit Global Location Number (GLN) that identifies the business system within which *`xformID`* is defined, followed by a colon. This is typically a “party GLN” that identifies the organisation responsible for the transformation identifier, or a division of an organisation that maintains a separate divisional business information system. |
| *`xformID`* | An identifier for the transformation that complies with the requirements of [RFC2141] and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the GLN that is embedded in the identifier. The owner of the GLN may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

### 8.7.3 Private or Industry-wide URN for Transformation identifiers

A CBV-Compliant document or CBV-Compatible document MAY use a private or industry-wide URN as specified below to populate the transformationID field of EPCIS TransformationEvents.

A private or industry-wide URN SHALL have the following form:
urn:URNNamespace:**:xform:transID where the components of this template are as follows:

| Template Component | Description |
|---|---|
| urn:*URNNamespace*:**: | As specified in Section 8.1.2. |
| xform: | The characters x, f, o, r, m, and : (colon). |
| xformID | An identifier for the transformation that complies with the requirements of [RFC2141] and any syntax rules defined for the registered URN namespace URNNamespace, and which does not contain a colon character. This identifier must be unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the URN Namespace. The owner of the URN Namespace may delegate the authority to assign new identifiers to End Users or other parties, provided that appropriate rules are employed to ensure global uniqueness.

### 8.7.4 HTTP URLs for Transformation identifiers

A CBV-Compliant document or CBV-Compatible document MAY use an HTTP URL as specified below to populate the transformationID field of EPCIS TransformationEvents.

An HTTP URL SHALL have the following form:

http://[*Subdomain*.]*Domain*/**/xform/*xformID*

where the components of this template are as follows:

| Template Component | Description |
|---|---|
| http://[*Subdomain*.]*Domain*/**/ | As specified in Section 8.1.3. |
| xform/ | The characters x, f, o, r, m, and / (slash). |
| *xformID* | An identifier for the transformation that matches the grammar rule segment-nz defined in [RFC3986], Section 3.3 (among other things, this means xformID may not contain a slash character), and which is unique relative to all other identifiers that begin with the same prefix. |

Identifiers of this form must be assigned by the owner of the Internet domain Domain. The owner of the domain may delegate the authority to assign new identifiers to other parties, provided that appropriate rules are employed to ensure global uniqueness.

**Non-Normative**: Example: An EPCIS document in XML format containing a usage sample may be found in Section 11.2.

## 8.8 Event identifiers

An event identifier may populate the eventID field of an EPCIS event. When an EPCIS event includes an eventID field, the identifier in that field must be globally unique (different from the event identifier in any other EPCIS event created by any party). Note that an EPCIS event is not required to include an event identifier.

A CBV-Compliant document SHALL use the URI form specified in Section 8.8.1 to populate the eventID field of EPCIS events, for every such field that is not null. A CBV-Compatible document MAY use the URI form specified in Section 8.8.1, or MAY use any other URI that meets the general requirements specified in [EPCIS1.2], Section 6.4, except for those URIs which in this standard are forbidden or designated for a different purpose.

### 8.8.1 Universally Unique Identifier (UUID) URIs for Event identifiers

A CBV-Compliant document SHALL and a CBV-Compatible document MAY use a UUID Version 1 or Version 4 URI as specified in [RFC4122] to populate the eventID fields in any EPCIS event where that field is not omitted.

**Non-Normative**: Example:
<eventID>urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6</eventID>

## 9 Trade item master data

This section specifies master data attributes that may be used to describe a trade item identifier that appears in the "what" dimension of an EPCIS event, including the EPC, Parent ID, and EPC Class fields.

Different trade item identifiers are used at different levels of trade item identification. Each master data attribute defined in the CBV for trade item identifiers specifies one or more of the following three levels of identification to which it is applicable:

| Identification Level | Description | Typical Identifier | Identifier use in EPCIS Event |
|---|---|---|---|
| Trade item-level | A master data attribute that applies to all instances of a given trade item. As trade items are usually identified by a GTIN, this is often called "GTIN-level". | urn:epc:idpat:sgtin:0614141.112345.* | EPC Class |
| Lot-level | A master data attribute that applies to all instances of a given trade item within a specified batch or lot. | urn:epc:class:lgtin:0614141.112345.L123 | EPC Class |
| Instance-level | A master data attribute that applies to a specific instance of a trade item. | urn:epc:id:sgtin:0614141.112345.400 | EPC<br>Parent ID |

A CBV-Compliant or CBV-Compatible document MAY include any of the master data attributes specified in this section within the master data section of the EPCIS Header, subject to the constraints specified elsewhere in this section. The master data attributes specified in this section may also be used in an EPCIS Master Data Document or in the response to an EPCIS Master Data Query. A CBV-Compliant or CBV-Compatible document MAY include any of the lot-level or instance-level master data attributes specified in this section in the ILMD section of an EPCIS event, but SHOULD NOT include trade item-level attributes in the ILMD section.

When a master data attribute specified in this section is used in the master data section of the EPCIS Header, in an EPCIS Master Data Document, or in the response to an EPCIS Master Data Query, each such attribute applies to the specific identifier cited and also all matching identifiers at a lower level. For example, a master data attribute specified for the trade item-level identifier urn:epc:idpat:sgtin:0614141.112345.* would also apply to lot-level and instance-level identifiers that share the same GTIN. A master data attribute specified for the lot-level identifier urn:epc:class:lgtin:0614141.112345.L123 would also apply to instance-level identifiers that share the same GTIN and lot.

When a master data attribute specified in this section is used in the ILMD section of an EPCIS event, it applies to all identifiers appearing in any EPC or QuantityElement field within that event.

## 9.1 Trade item master data attribute names

In the master data section of an EPCIS header, in an EPCIS Master Data Document, and in the response to an EPCIS Master Data Query, a master data attribute is specified as a name/value pair. The name of every trade item master data attribute defined in this section consists of the following namespace identifier:

`urn:epcglobal:cbv:mda`

followed by a pound sign (#) character, followed by a local name as specified in Section 9.2.

In the ILMD section of an EPCIS event, a master data attribute is specified as an XML element. The element name is an XML QName whose namespace is the same namespace identifier specified above and whose local name is the local name as specified in Section 9.2.

**Non-Normative:** Example: Here is how the attribute `sellByDate` would appear in the EPCIS header, Master Data Document or Master Data Query response:

```
<VocabularyElement id="urn:epc:class:lgtin:0614141.012345.L123">
  <attribute id="urn:epcglobal:cbv:mda#sellByDate">2016-03-15</attribute>
</VocabularyElement>
```

Here is how the same attribute would appear in the ILMD section of an event:

```
<epcis:EPCISDocument xmlns:cbvmd="urn:epcglobal:cbv:mda" …>
  ...
  <ObjectEvent>
    ...
    <QuantityElement>
       <epcClass>urn:epc:class:lgtin:0614141.012345.L123</epcClass>
    </QuantityElement>
    ...
    <ilmd>
      <cbvmd:sellByDate>2016-03-15</cbvmd:sellByDate>
    </ilmd>
    ...
  </ObjectEvent>
  ...
</epcis:EPCISDocument>
```

## 9.2 Trade item master data attributes

The tables below specify master data attributes that may be used to describe a trade item identifier.

The meaning of the "Level" column is as follows:

- **Trade Item**: the master data attribute is a trade item-level attribute as specified in Section 9.
- **Lot**: the master data attribute is a lot-level attribute as specified in Section 9.
- **Instance**: the master data attribute is an instance-level attribute as specified in Section 9.
- **Trade Item or Instance**: the master data attribute is either a trade item-level attribute or an instance-level attribute as specified in Section 9, depending on the trade item. For example, `netWeight` is a trade item-level attribute for a fixed weight product but an instance-level attribute for a variable weight product.
- **Trade Item or Lot or Instance**: the master data attribute is either a trade item-level attribute or a lot-level attribute or an instance-level attribute as specified in Section 9, depending on the trade item. For example, `countryOfOrigin` may be consistent across all instances of a trade item for a manufactured product, or consistent across all instances in a lot but varying across lots for fish species harvested in lots in varying territorial waters, or varying across all instances for fish species harvested individually in varying territorial waters.

Master data attributes for each level are shown below in separate tables. Master data attributes that may be used at multiple levels are repeated in more than one table as appropriate. Within each table, attributes are listed alphabetically.

### 9.2.1 Trade item master data attributes – trade item level

The following attributes may be used to describe a trade item identifier at the trade item (GTIN) level.

| Local Name | Type | Description | Level |
|---|---|---|---|
| additionalTradeItemIdentification | String (1–80 characters) | A trade item identifier that is in addition to the GTIN.<br>Example: 12345111111 | Trade Item |
| additionalTradeItemIdentificationTypeCode | Code | A code that indicates what type of identifier is used for additionalTradeItemIdentification<br>The code list for this attribute is defined in GDSN; see *http://apps.gs1.org/GDD/Pages/clDetails.aspx?semanticURN=urn:gs1:gdd:cl:AdditionalTradeItemIdentificationTypeCode&release=1*<br>Example: FDA_NDC_11 | Trade Item |
| countryOfOrigin | Code | Country from which the goods are supplied.<br>The code list for this attribute is the ISO 3166-1 Alpha-2 list of 2-letter country codes; see *http://www.iso.org/iso/country_codes*<br>Example: UK | Trade Item or Lot or Instance |
| descriptionShort | String (1–35 characters) | A free form short length description of the trade item that can be used to identify the trade item at point of sale.<br>Example: Acme Red Widgets | Trade Item |
| dosageFormType | String (1–35 characters) | A dosage form is the physical form of a medication that identifies the form of the pharmaceutical item.<br>Example: PILL | Trade Item |
| drainedWeight | Measurement (see Section *9.2.4*) | The weight of the trade item when drained of its liquid. For example 225 "grm", Jar of pickles in vinegar. Applies to defined bricks of GCI Global trade item Classification - Mainly food trade item. Has to be associated with a valid UoM.<br>Example: [see Section *9.2.4*] | Trade Item or Instance |
| functionalName | String (1–35 characters) | Describes use of the product or service by the consumer. Should help clarify the product classification associated with the GTIN.<br>Example: Widget | Trade Item |
| grossWeight | Measurement (see Section *9.2.4*) | Used to identify the gross weight of the trade item. The gross weight includes all packaging materials of the trade item. At pallet level the trade item-GrossWeight includes the weight of the pallet itself. For example, "200 grm", value - total pounds, total grams, etc. Has to be associated with a valid UOM.<br>Example: [see Section *9.2.4*] | Trade Item or Instance |
| manufacturerOfTradeItemPartyName | String (1–200 characters) | Party name information for the manufacturer of the trade item.<br>Example: Acme Corporation | Trade Item |
| netContentDescription | String (1–500 characters) | Free text describing the amount of the trade item contained by a package, usually as claimed on the label.<br>Example: 253 grams | Trade Item |

| Local Name | Type | Description | Level |
|---|---|---|---|
| netWeight | Measurement (see Section 9.2.4) | Used to identify the net weight of the trade item. Net weight excludes any packaging materials and applies to all levels but consumer unit level. For consumer unit, Net Content replaces Net Weight (can then be weight, size, volume). Has to be associated with a valid UoM.<br>Example: [see Section 9.2.4] | Trade Item or Instance |
| labelDescription | String (1–500 characters) | A literal reproduction of the text featured on a product's label in the same word-by-word order in which it appears on the front of the product's packaging. This may not necessarily match the GTIN description as loaded by the supplier into the GTIN description field in GDSN.<br>Example: Acme Corporation Tiny Red Widgets | Trade Item |
| regulatedProductName | String (1–500 characters) | The prescribed, regulated or generic product name or denomination that describes the true nature of the product and is sufficiently precise to distinguish it from other products according to country specific regulation.<br>Example: Epcistra | Trade Item |
| strengthDescription | String (1–500 characters) | Free text describing the strength of the active ingredient(s) of the product<br>Example: 200mg/100mg | Trade Item |
| tradeItemDescription | String (1–200 characters) | An understandable and useable description of a trade item using brand and other descriptors. This attribute is filled with as little abbreviation as possible while keeping to a reasonable length. Free form text field, this data element is repeatable for each language used and must be associated with a valid ISO language code. Field length is 178 characters. This should be a meaningful description of the trade item with full spelling to facilitate message processing. Retailers can use this description as the base to fully understand the brand, flavour, scent etc. of the specific GTIN in order to accurately create a product description as needed for their internal systems.<br>Example: GS1 Brand Base Invisible Solid Deodorant AP Stick Spring Breeze | Trade Item |

### 9.2.2 Trade item master data attributes – lot level

The following attributes may be used to describe a trade item identifier at the lot level.

| Local Name | Type | Description | Level |
|---|---|---|---|
| bestBeforeDate | Date | The date before which the product is best used or consumed. It is a statement about quality.<br>Example: 2017-03-15 | Lot |
| countryOfOrigin | Code | [see description in Section 9.2.1] | Trade Item or Lot or Instance |
| farmList | List of Farm | List of structures describing farm information; see below | Lot |
| firstFreezeDate | Date | The date of initial freezing, if different from the date of production.<br>Example: 2016-03-15 | Lot |