

DRAFT FOR DEVELOPMENT

Structured vocabularies for information retrieval – Guide –

Part 5: Exchange formats and protocols for interoperability

ICS 35.040

This publication is not to be regarded as a British Standard.

It is being issued in the Draft for Development series of publications and is of a provisional nature because the model and schema in Clauses 5 and 6 are considered to need additional user testing with full-scale applications before they can be regarded as stable. No existing format capable of conveying thesauri that conform to BS 8723-2 was identified in widespread use, therefore original work was undertaken to develop the format recommended herein. DD 8723-5 should be applied on this provisional basis so that information and experience of its practical application can be obtained.

A review of this Draft for Development will be carried out not later than 2 years after its publication.

Notification of the start of the review period, with a request for the submission of comments from users of this Draft for Development, will be made in an announcement in the appropriate issue of Update Standards.

According to the replies received, the responsible BSI Committee will judge whether the Draft for Development can be converted into a British Standard or what other action should be taken.

Observations which it is felt should receive attention before the official call for comments will be welcomed.

These should be sent to the Secretary of BSI Subcommittee IDT/2/2 at British Standards House, 389 Chiswick High Road, London W4 4AL.

Publishing and copyright information

The BSI copyright notice displayed in this document indicates when the document was last issued.

© BSI 2008

ISBN 978 0 580 53864 3

The following BSI references relate to the work on this standard:
Committee reference IDT/2/2

Publication history

First published July 2008

Amendments issued since publication

Amd. no.	Date	Text affected
-----------------	-------------	----------------------

Contents

Foreword *iii*

Introduction *1*

1	Scope	<i>2</i>
2	Normative references	<i>2</i>
3	Definitions	<i>3</i>
4	Functions involving vocabularies that need interoperability	<i>4</i>
5	Data model	<i>7</i>
5.1	General	<i>7</i>
5.2	Notes on the model	<i>9</i>
5.3	Tabular presentation	<i>12</i>
6	Exchange formats	<i>17</i>
6.1	XML schema for thesaurus exchange	<i>17</i>
6.2	Exchanging a whole thesaurus	<i>18</i>
6.3	Updating a thesaurus	<i>18</i>
6.4	Exchanging thesaurus subsets	<i>18</i>
7	Alternative formats in widespread use	<i>28</i>
8	Protocols	<i>29</i>

Annexes

Annex A (informative) Examples of thesaurus data in DD 8723-5 format *30*

Bibliography *51*

Index (DD 8723-5) *54*

List of figures

Figure 1	– UML model	<i>8</i>
Figure 2	– XML schema	<i>19</i>
Figure A.1	– Example A: Hierarchical display	<i>31</i>
Figure A.2	– Example A: Alphabetical display	<i>31</i>
Figure A.3	– Example A: Data encoded in DD 8723-5 format	<i>32</i>
Figure A.4	– Example B: Alphabetical display in English	<i>35</i>
Figure A.5	– Example B: Alphabetical display in Spanish	<i>36</i>
Figure A.6	– Example B: Bilingual classified display, with node labels and notation	<i>37</i>
Figure A.7	– Example B: Data encoded in DD 8723-5 format	<i>38</i>
Figure A.8	– Example C: Alphabetical display	<i>42</i>
Figure A.9	– Example C: Hierarchical display	<i>42</i>
Figure A.10	– Example C: Data encoded in DD 8723-5 format	<i>43</i>
Figure A.11	– Example D: Alphabetical display	<i>45</i>
Figure A.12	– Example D: Classified display	<i>46</i>
Figure A.13	– Example D: Hierarchical display	<i>46</i>
Figure A.14	– Example D: Data encoded in DD 8723-5 format	<i>47</i>

List of tables

Table 1 – Attributes and associations of <i>Thesaurus</i>	13
Table 2 – Attributes and associations of <i>ThesaurusArray</i>	14
Table 3 – Attributes and associations of <i>ThesaurusConcept</i>	14
Table 4 – Attributes and associations of <i>ThesaurusTerm</i>	16
Table 5 – Attributes and associations of <i>Note</i>	17
Table 6 – Definitions of occurrence as used in Tables 1 to 5	17
Table A.1 – Tags used for Example B	37

Summary of pages

This document comprises a front cover, an inside front cover, pages i to iv, pages 1 to 56, an inside back cover and a back cover.

Foreword

Publishing information

This part of BS 8723 is published by BSI and came into effect on 31 July 2008. It was prepared by Subcommittee IDT/2/2, *Information description, source identification and indexing*, under the authority of Technical Committee IDT/2, *Information and documentation*. A list of organizations represented on these committees can be obtained on request to their secretary.

Relationship with other publications

BS 8723 comprises five parts:

- Part 1: *Definitions, symbols and abbreviations*;
- Part 2: *Thesauri*;
- Part 3: *Vocabularies other than thesauri*;
- Part 4: *Interoperability between vocabularies*;
- Part 5: *Exchange formats and protocols for interoperability*.

Part 1 provides definitions, symbols, abbreviations and other conventions applying to all the parts.

Part 2 covers thesauri, designed for situations in which human indexers analyse documents and express their subjects using thesaurus terms, before searchers retrieve the documents with the same vocabulary.

Part 3 covers other types of structured vocabulary, especially those used in selecting terms or codes for use in subject metadata.

Part 4 applies to situations in which more than one language or vocabulary is in use, but access to all resources is needed through the one vocabulary chosen by the user.

Part 5 deals with the protocols and formats needed for the exchange of vocabulary data, and is being issued as a Draft for Development in the first instance.

Use of this document

As a guide, this part of BS 8723 takes the form of guidance and recommendations. It should not be quoted as if it were a specification and particular care should be taken to ensure that claims of compliance are not misleading.

Any user claiming compliance with this part of BS 8723 is expected to be able to justify any course of action that deviates from its recommendations.

Presentational conventions

The provisions in this standard are presented in roman (i.e. upright) type. Its recommendations are expressed in sentences in which the principal auxiliary verb is “should”.

Commentary, explanation and general informative material is presented in smaller italic type, and does not constitute a normative element.

Contractual and legal considerations

Liability. The British Standards Institution (“BSI”) shall not be liable, to the fullest extent permitted by law, for loss suffered by any party as a result of the use or application of the Draft for Development including loss of profits, loss of business, loss of contract, loss or corruption of data or information, any special, indirect, consequential or pure economic loss, costs, damages, charges or expenses. This exclusion shall not extend to death or personal injury caused by negligence.

Users of the Draft for Development should ensure that its application is entrusted to those who are appropriately qualified and experienced.

Intellectual property rights. The British Standards Institution (“BSI”) gives no warranty that the contents of this Draft for Development do not breach the intellectual property rights, including but not limited to the copyright, design rights and/or patent rights of any party and does not accept any liability, to the fullest extent permitted by law, for any claim that the content of the Draft for Development breaches such third party rights. **Users of the Draft for Development should satisfy themselves with regard to any such rights.**

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Compliance with a Draft for Development cannot confer immunity from legal obligations.

Introduction

Widespread electronic networks and the almost universal use of computers in information management present opportunities for sharing data and information on a scale not previously imaginable: from a single computer connected to the internet one can simultaneously search different collections of books, articles, photographs, data sets, or museum artefacts residing on remote computers all over the world. The application of standards and facilities for interoperability makes it easier to search any of those collections by subject, where the subject terms used have been drawn from a structured vocabulary.

In the round, interoperability is encouraged by the following components:

- network infrastructure;
- formats for data exchange;
- protocols, including basic computer communication protocols as well as those applying specifically to vocabularies;
- registries of vocabularies, metadata and other common tools;
- robust data architecture.

While all are needed in the context of information retrieval, this part of BS 8723 addresses only the formats and protocols specific to exchange of vocabulary information.

Sometimes the functions of managing a structured vocabulary and making information retrievable using that vocabulary are integrated into a single system. However, information services are often provided by different computer applications running on different computers that cooperate to meet the needs of a user. For example, in order to allow a user to perform a federated search of several different databases with a single search strategy, an information retrieval application might need to translate search terms from a thesaurus used in one database into the equivalent terms used in another. In order to perform this translation, the search application usually needs to consult a structured vocabulary application running on another computer at a different location on the network; the two applications have to communicate and cooperate in order to provide this federated search service to the user.

The goal of accessing information quickly, easily and effectively therefore depends not just on common approaches to defining thesauri and to managing them within a single computer system, as described in earlier parts of BS 8723; it also depends on common approaches that promote interoperability of different computer applications when sharing the responsibility for management and use of structured vocabularies. Included among these common approaches are the way in which the data can be requested by one computer application and delivered by another (i.e. service protocols), and the presentation of the data requested (i.e. exchange formats). This part of BS 8723 addresses these two aspects of interoperability.

At the time of publication of this Draft for Development, networking technology is advancing rapidly. Future editions are likely to provide for enhanced interoperability functions.

1 Scope

This part of BS 8723 gives guidance on the exchange of structured vocabulary data between different computer applications involved in information retrieval.

It recommends a standard format for exchange of whole thesauri or subsets thereof. The format is applicable to thesauri that conform to BS 8723-2, utilizing any or all of the optional features described therein. Equally it supports multilingual thesauri displaying structural unity as described in BS 8723-4:2007, **4.2**. It does not address the additional needs of the vocabulary types described in BS 8723-3, although in some cases it may be adapted to serve such vocabularies. It does not address the exchange of data providing mappings between vocabularies that do not share the same structure.

This part of BS 8723 also considers the interaction between a computer application acting as a server for a structured vocabulary (the server) and an application applying that structured vocabulary to a particular information retrieval problem (the client). It considers only the highest level of protocol, and assumes the existence of lower level network services and protocols which will be required to support computer-to-computer communication over a network.

This part of BS 8723 does not address the internal design of systems that provide multiple functions and require communication only between one part and another part of the same application (such as an indexing system that incorporates its own internal subject authority system with a structured vocabulary). Nor does it address the interaction of a computer application and a user, including user interface issues such as commands, menus, windows, buttons, and screen displays (although some guidance on these issues is provided in earlier parts of BS 8723).

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

BS 8723-1:2005, *Structured vocabularies for information retrieval – Guide – Part 1: General*

BS 8723-2:2005, *Structured vocabularies for information retrieval – Guide – Part 2: Thesauri*

BS 8723-3:2007, *Structured vocabularies for information retrieval – Guide – Part 3: Vocabularies other than thesauri*

BS 8723-4:2007, *Structured vocabularies for information retrieval – Guide – Part 4: Interoperability between thesauri*

BS ISO 639-1, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*

RFC 4646, *Tags for identifying languages*. The Internet Society. Best Current Practice recommendation, 2005.¹⁾

¹⁾ Available at <http://www.rfc-editor.org/rfc/rfc4646.txt>.

3 Definitions

For the purposes of this part of BS 8723, the definitions in BS 8723-1, BS 8723-3, BS 8723-4 and the following apply. Where a term used in a definition is defined here or in the other parts, that term is italicized.

3.1 client

initiating *computer application* in a network interaction

NOTE The client application is the application that usually interacts with the human user.

3.2 computer application

computer program or set of programs that provides high-level processing related to a specific user need

NOTE In DD 8723-5, a computer application is sometimes referred to as an application.

3.3 data model

abstract model that describes how data is represented and used

3.4 exchange format

convention concerning the structure and semantics of data to be transferred between one *computer application* and another

NOTE For example, an exchange format might specify that a broader term should be preceded by a label BT and a specified punctuation symbol, and followed by an end of line marker. The layout and sequence of elements is the structure, and the label BT indicates the semantics of the information being exchanged.

3.5 federated search

simultaneous search of multiple information resources using a single search strategy and a common user interface

NOTE Also known as metasearch or cross-database search.

3.6 markup

annotations or other type of encoding embedded in text, in conformity with a *markup language*

3.7 markup language

set of encoding conventions that can be used to provide instructions for the interpretation of a text, by the use of annotations embedded in the text itself

NOTE The interpretation may concern issues such as content, structure or rendering of the text. Widely used examples include HTML (Hypertext Markup Language) [1], which is largely concerned with presentation, and XML (eXtensible Markup Language) [2–4], which addresses the structure of text.

3.8 protocol

convention that defines the syntax, semantics and synchronization of the communication process between two computers in order to enable a particular service

3.9 server

responding *computer application* in a network interaction

NOTE The server of a structured vocabulary is a vocabulary server.

4 Functions involving vocabularies that need interoperability

All of the functions described in earlier parts of BS 8723 as applying within a single system should be supported in a networked environment. (These are described variously in BS 8723-2:2005, Clause **11**, **13.5** and Clause **14**; BS 8723-3:2007, **6.8** and **6.9**; and BS 8723-4:2007, Clause **11**.) More specifically, there can be a need to:

- publish the whole vocabulary or part of a vocabulary;
- publish vocabulary updates;
- search a collection with direct human interaction;
- search a collection without direct human interaction;
- browse a collection with direct human intervention;
- cross-search multiple collections using different vocabularies;
- search a collection of metadata harvested from different sources each using a different vocabulary;
- manually index or classify documents in a collection;
- automatically index or classify documents in a collection.

In a networked environment, a vocabulary server commonly has to send data to a variety of clients, such as a search engine, an indexing system or a vocabulary management system. To support the range of functions enumerated in this clause, the design of an exchange format and protocol should take account of the fact that different combinations of client and vocabulary server applications are liable to apportion the processing burden differently between them. At one extreme, all processing functions are provided by the server prior to transmission of the data, and the client merely displays the result. At the other, the server merely serves the raw data and all further processing is undertaken by the client.

For example, when a thesaurus is published on the web for display via a standard web browser, the server commonly sends the data with presentational markup in Hypertext Markup Language (HTML) [1] and the client passively displays the result. In this case, what the user sees (including any hyperlinks) is dependent entirely upon the HTML generated by the server; the client browser needs only to render the HTML it receives and often performs no additional processing. The server-generated HTML is needed simply to ensure that the thesaurus data are formatted for visual presentation in accordance with the guidelines provided elsewhere in BS 8723, particularly BS 8723-2:2005, Clause **10** and BS 8723-4:2007, Clause **10**, and structural markup is largely redundant. Optionally, some client applications add value by utilizing both presentational and structural markup.

In other cases, the client application (whether browser-based or otherwise) needs to add considerable functionality of its own to the data received from the server. For instance, a search application might offer query expansion facilities which require it to interrogate either the cached data on the client or the original data on the server, or both, in order to determine and display broader, narrower and related terms or any preferred term corresponding to the user's entry term. Client applications such as automatic indexing systems need not only to interrogate the server data, but also to undertake the text processing needed to select index terms for particular documents. A description of the latter type of processing is outside the scope of this part of BS 8723, but any presentational markup in the server data is largely irrelevant in this context.

In addition, some of the interactions in a network can be highly interactive (e.g. switching terms at the time of a search) and some are primarily a batch transaction (e.g. sending a complete thesaurus in electronic form to another institution for loading into their system). Where batch functionality is required, the exchange format is important but protocols are likely to be very simple and based on existing communication and application protocols. For example, a network request is sometimes made via File Transfer Protocol [5] for a file that contains a complete copy of the thesaurus. If exchange does not go through an electronic network, a protocol might not be required at all: the request could be made by telephone and a copy of a thesaurus sent on CD-ROM through the postal system, then loaded on another computer system. A standard format should be used for the data.

In order to support these uses and achieve interoperability with a wide range of thesauri and classification schemes, the lower level functions listed in items a) to c) below are required. Not all of these functions apply to all contexts. For example:

- the ability to specify a language is not required to provide access to a monolingual thesaurus;
- in the absence of mappings between one structured vocabulary and another, a request to switch a term is irrelevant.

However, an exchange format or protocol which supports only a subset of these functions is less likely to be widely implemented and hence is less likely to encourage interoperability.

In summary, as many as possible of the following functions should be supported.

a) Publish or exchange a vocabulary

- Export/import a whole vocabulary, or a subset which could include some selection of concepts, classes or categories and could include some or all of the information associated with each selected concept, class or category.

b) Update a vocabulary publication

- Export all changes made since the last edition of the vocabulary for (ideally) automatic import into another vocabulary system.

c) Access a vocabulary for indexing or searching

- Search for a particular concept in a structured vocabulary based on:
 - the full preferred term or non-preferred term in a specified language;
 - words from the preferred term or non-preferred term in a specified language;
 - character strings from the preferred term or non-preferred term in a specified language;
 - the identifier of the term, concept, class or category;
 - the classification code or notation.
- Retrieve for a given term, concept, class, etc., information directly related to that term, concept or class:
 - all information concerning the term, concept or class;
 - all information concerning the term, concept or class except internal housekeeping information;
 - all information related to a particular language (for a multilingual vocabulary);
 - equivalent terms in other languages;
 - other combinations of data elements, when needed to support anticipated use patterns;
 - mappings into other structured vocabularies;
 - alternative forms of a term (to support augmentation or expansion of a search term, possibly to take in hierarchically and/or associatively related terms as well as equivalents).
- Retrieve for a given term or class the context of that term or class:
 - the alphabetical context of a term in a particular language, including a specifiable number of terms that follow that term in the alphabetical sequence of terms and preferably a specifiable number of terms that precede that term in the alphabetical sequence;
 - the hierarchical (and possibly associative) context of a term or class, including terms or classes higher and lower in the hierarchy by a specifiable number of levels;
 - the classified context of a term, including a specified number of terms before or after the term in the classified sequence and, where applicable, any node labels pertaining to the terms displayed.
- Validate a particular term in a particular language.

5 Data model

5.1 General

This clause models the underlying structure of the data which may be conveyed in a thesaurus conforming to BS 8723-2. The XML schema in **6.2** is derived from this model and should be used when sending or receiving thesaurus data in electronic form. It is not intended for use internally within a computer system, although it might be found useful, in whole or part, in that situation too.

The basic recommendations of BS 8723-2 are quite simple, needing only a simple model. But BS 8723-2 also allows and sometimes encourages a number of optional features that require much greater complexity in the model. For thesaurus exchange to be successful in a wide range of situations, it is important to have a format hospitable to sophisticated as well as simple thesauri.

The model presented in the present part of BS 8723 accommodates the full range of options described in BS 8723-2. Some elements of this model are shown as optional, and need not be used when exchanging thesauri that lack the optional features. Software systems importing data may ignore optional elements that they do not support, and assume appropriate default values (which may be null) for optional elements that they do support but which are not present in the imported data, while interpreting the remaining elements consistently and correctly.

The model is presented first as a UML (Unified Modelling Language [6–8]) class diagram (Figure 1). UML is highly expressive and discourages ambiguity, but as some users might be unfamiliar with its conventions, an alternative tabular presentation is added at **5.3**. The data elements and attributes in the UML diagram and the data tables are identical in substance though some details of relationships are not made explicit in the tabular presentation. In case of doubt, the diagram should be regarded as the authority.

NOTE UML defines its own special conventions and terminology for modelling purposes. In the present clause, the term “class” is used in the UML sense, as a construct that is used to group related instance variables. Similarly, UML gives special meanings to the expressions “String”, “Boolean” and “aggregation”. More information about UML can be found from references in the Bibliography.

Diagram illustrating the structure of a Thesaurus system, showing relationships between various entities and their attributes.

Entities and Attributes:

- Thesaurus**
 - +identifier: String[1..*]
 - +contributor: String[0..*]
 - +coverage: String[0..*]
 - +creator: String[0..*]
 - +date: date[0..*]
 - +created: date[0..1]
 - +modified: date[0..*]
 - +description: String[0..*]
 - +format: String[0..*]
 - +language: language[1..*]
 - +publisher: String[0..*]
 - +relation: String[0..*]
 - +rights: String[0..*]
 - +source: String[0..*]
 - +subject: String[0..*]
 - +title: String[0..*]
 - +type: String[0..*]
- NodeLabel**
 - +Notation: String[0..1]
 - +LexicalValue: String[1]
 - +created: date[0..1]
 - +modified: date[0..1]
 - +lang: language[0..1]
- ThesaurusArray**
 - +identifier: String[1]
 - +Ordered: Boolean = false[1]
- ThesaurusConcept**
 - +identifier: String[1]
 - +created: date[0..1]
 - +modified: date[0..1]
 - +Status: String[0..1]
 - +Notation: String[0..*]
 - +TopConcept: Boolean[0..1]
- TopLevelRelationship**
 - +HasTopConcept
 - +IsTopConceptOf
- AssociativeRelationship**
 - +Role: String[0..1]
 - +HasRelatedConcept
 - +IsRelatedConcept
- HierarchicalRelationship**
 - +Role: String[1]
 - +HasHierRelConcept
 - +IsHierRelConcept
- CustomAttribute**
 - +LexicalValue: String[1]
 - +CustomAttributeType: String[1]
- Note**
 - +LexicalValue: String[1]
 - +created: date[0..1]
 - +modified: date[0..1]
 - +lang: language[0..1]
- ScopeNote**
- HistoryNote**
- Definition**
 - +source: String[0..1]
- EditorialNote**
- Equivalence**
 - +Role: String[0..1]
- CompoundEquivalence**
- SimpleNonPreferredTerm**
 - +HasNonPreferredTerm
- PreferredTerm**
 - +HasPreferredTerm
- CompoundNonPreferredTerm**
 - +HasNonPreferredTerm
- ThesaurusTerm**
 - +LexicalValue: String[1]
 - +identifier: String[1]
 - +created: date[0..1]
 - +modified: date[0..1]
 - +source: String[0..1]
 - +Status: String[0..1]
 - +lang: language[0..1]

Relationships:

- Thesaurus** (1) **ThesaurusArray** (0..*) via **IsLabelledBy** (0..*)
- Thesaurus** (1) **ThesaurusConcept** (0..1) via **Contains** (1..*)
- Thesaurus** (1) **ThesaurusConcept** (0..1) via **isPartOf** (1)
- ThesaurusConcept** (0..1) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (1..*) **ThesaurusArray** (0..*) via **IsMemberOf** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..1) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSubordinate** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasMember** (0..*)
- ThesaurusConcept** (0..*) **ThesaurusArray** (0..*) via **HasSuperOrdinate** (0..*)

5.2 Notes on the model

COMMENTARY ON 5.2

*Conventions for naming and symbols are described first. Since the classes **Thesaurus**, **ThesaurusConcept** and **ThesaurusTerm** are fundamental to every thesaurus, their attributes and other features are described next. Some of the same attributes are found also in the classes **Note** and **NodeLabel**. (For definitions of “thesaurus”, “concept”, “term”, “node label”, etc., see BS 8723-1. For a fuller discussion of all of them, see BS 8723-2.) After these come brief notes on **ThesaurusArray** and **NodeLabel**. Attributes and associated features of the relationship classes are described last.*

These notes apply to the tabular presentation in 5.3, as well as to the diagram in Figure 1.

5.2.1 Naming conventions

Some Dublin Core metadata elements [9] are shown in the model and in the derived XML schema. The names of these elements follow the Dublin Core lowerCamelCase convention. Other elements use UpperCamelCase in line with the standard naming convention for UK public sector schema development [10].

5.2.2 Symbols for associations

The diamond symbol in the diagram represents an aggregation. This symbol is used to indicate that in the derived XML schema the definition of an object will be nested in its parent object.

Normal associations (without a diamond) indicate a relationship of the type key/keyref in XML as described in XML documentation at http://www.w3.org/TR/xmlschema-1/#cIdentity-constraint_Definitions and <http://www.w3.org/TR/xmlschema-0/#specifyingKeysAndtheirRefs> (see [2–4] for the complete reference). Note that the XML schema shown in Clause 6, Figure 2 does not actually enforce the key/keyref constraints. The reason for this relaxation of the constraints is that enforcement would have the effect of invalidating XML output files that contain only an extracted portion of the thesaurus rather than a whole thesaurus.

5.2.3 Identifier

This is an attribute of **Thesaurus**, **ThesaurusConcept**, **ThesaurusTerm** and **ThesaurusArray**. Identifiers for terms and concepts are described in BS 8723-2:2005, 12.3.1.2 and 12.3.1.3. For trouble-free use of the format described in Clause 6, the application of both types of identifier is strongly recommended. A separate identifier for each array (see 5.2.5) may also be applied.

5.2.4 Dates

The date created and date modified are optional attributes of **Thesaurus**, **ThesaurusConcept**, **ThesaurusTerm**, **Note** and **NodeLabel**. They should be given in YYYY-MM-DD format, in line with ISO 8601. In the case of **Thesaurus**, another option is to use the simple attribute “date” without specifying whether it applies to the first edition or a subsequent version. The same format should be used.

5.2.5 Status

This is an optional attribute of *ThesaurusConcept* and *ThesaurusTerm*. See BS 8723-2:2005, 12.3.1.2b).

5.2.6 Notation

This is an optional attribute of *ThesaurusConcept* and *NodeLabel*. See BS 8723-2:2005, 10.1.2 and 10.5.2.

5.2.7 Source

This is an optional attribute of *Thesaurus* and *ThesaurusTerm*. See BS 8723-2:2005, 12.3.1.2a).

5.2.8 Language codes

Language is an optional attribute of *ThesaurusTerm*, *Note* and *NodeLabel* and a mandatory attribute of *Thesaurus*. It should be given as a code selected from BS ISO 639-1, extended where necessary with the additional codes in RFC 4646.

5.2.9 Dublin Core elements

The *Thesaurus* class has a number of attributes intended for assigning Dublin Core metadata [9] to a file using the BS 8723 format. As well as identifier, date, source and language already described, these include contributor, coverage, creator, description, format, publisher, relation, rights, subject, title and type. All except language and identifier are optional.

5.2.10 Multiplicity of *PreferredTerm*

The model allows more than one preferred term per concept, to provide for multilingual thesauri, where one is permitted per language. For monolingual thesauri the multiplicity is reduced to one.

5.2.11 Top level concepts

BS 8723-2:2005, 10.2d) makes brief mention of “Top Term” as an optional feature of alphabetical displays and single term displays, linking a concept to the top concept of the hierarchy in which it occurs. The class *TopLevelRelationship* enables this.

BS 8723-2:2005, 14.8b) recommends a browse capability that starts by presenting editors with a list of top terms. The attribute *TopConcept* addresses this, specifying that a concept is at the top of a hierarchy, i.e. it has no broader concepts. This makes it easier, on importing a thesaurus, to pick out all the top level concepts from which to build a navigational tree.

5.2.12 Note references

The association between *Note* and *ThesaurusConcept* enables any note for one concept to refer (using marked up text) to any other concept in the thesaurus. This capability is particularly useful for scope notes. See BS 8723-2:2005, 6.1.2.4.

5.2.13 Custom attributes

The model includes a class *CustomAttribute* for custom attributes of the concept. This class enables recording of custom information about the concept such as the inclusion of the concept within a subset or micro-thesaurus, or a mapping to a related concept in another vocabulary, etc.

It is included as a separate class rather than as a normal attribute so that the administrator of the thesaurus management system can specify the values of custom attributes that can be assigned to a concept. The class has an attribute *CustomAttributeType*, allowing the administrator to specify which type of attribute is being used.

5.2.14 Arrays and node labels

Although all thesauri have arrays of sibling terms, it is not always necessary to model them explicitly and so this class is optional. The *ThesaurusArray* and *NodeLabel* classes of the model are needed particularly to support the generation of displays incorporating node labels and/or non-alphabetic sequences of sibling terms. (See examples in BS 8723-2:2005, Figures 4, 6, 7 and 8; also BS 8723-4:2007, Figure 7 and 10.5.3.) The attribute *Ordered* enables non-alphabetic sequences when necessary.

5.2.15 Relationship roles

The classes *Equivalence*, *HierarchicalRelationship* and *AssociativeRelationship* each have a “role” attribute. This can be used to convey the common relationships USE/UF, BT/NT and RT/RT. It can also be used for the optional relationship types described in BS 8723-2:2005, 8.3, as well as custom relationship types as suggested in BS 8723-2:2005, 8.5. Each occurrence of a relationship is expressed in one direction only, between a “source” class and a “target” class. For example, “Concept A has a broader concept, namely Concept B”, where the role is “broader”. If the complementary relationship “Concept B has a narrower concept, namely Concept A” needs to be expressed, then this is shown as a separate occurrence with the role “narrower”.

For custom relationship types, the text given in the “role” attribute should be composed of (a) the name of the parent relationship type followed by (b) the symbol forward slash “/” and finally (c) the name of the custom relationship type.

For example the text in the “Role” attribute of *HierarchicalRelationship* may be one of the following:

NT
NT/NTP
NT/NTI
NT/NTG
NT/NTI/NTX
BT
BT/BTP
BT/BTI
BT/BTG
BT/BTI/BTX
etc.

5.2.16 Compound non-preferred terms

The classes *CompoundNonPreferredTerm* and *CompoundEquivalence* enable representation of complex concepts by a combination of terms, as described in BS 8723-2:2005, 8.2.5.

5.2.17 Role of relationship tags

Figure 1 shows a relationship between *SimpleNonPreferredTerm* and *PreferredTerm*, with the tags USE and UF, which are described in BS 8723-2:2005, Clause 4 and 8.2 in the context of conventions and displays for human readers. In the context of data exchange, it could be argued that this relationship is redundant, since all the preferred and non-preferred terms for a given concept are linked to that concept. However, some redundancy is retained, to facilitate exchange of data for simpler thesauri and to make it easier to exchange update information and other specialized subsets of the data.

Another detail concerns USE...+. In displays for human readers, there is always a gap between “USE” and “+”, to accommodate the term in between (e.g. Coal mining USE coal + mining). But if the tag is to be used behind the scenes by computers, it needs to be compressed, in the style coal mining USE+ coal; coal mining USE+ mining. So in the diagram, the gap is omitted.

5.3 Tabular presentation

The elements of the model shown in Figure 1 are listed in Tables 1 to 5, based on the five classes *Thesaurus*, *ThesaurusArray*, *ThesaurusConcept*, *ThesaurusTerm* and *Note*. Other subsidiary classes have been shown within these tables. The brief text in the description columns is complemented by the more extensive notes in 5.2. See also definitions for many of the terms in BS 8723-1.

The Type column indicates the data type, using the UML data types “String” and “Boolean” where applicable, the XML data types “date” and “language” where no UML type is applicable, and “class” in the case of an initial or associated class in the model.

The occurrence columns describe whether the elements are mandatory and/or repeatable as shown in Table 6.

Table 1 Attributes and associations of *Thesaurus*

Element	Type	Occurrence	Description
Thesaurus	class	1	The thesaurus as a whole
<i>Attributes of Thesaurus</i>			
identifier	String	1..*	Identifier for the thesaurus as a whole
contributor	String	0..*	Person or organization who contributed to the thesaurus
coverage	String	0..*	Spatial or temporal coverage of the thesaurus
creator	String	0..*	Person or organization primarily responsible for making the resource
date	date	0..*	Any date associated with the thesaurus
created	date	0..1	The date the thesaurus was created
modified	date	0..*	A date when this version was modified
description	String	0..*	An account of the thesaurus
format	String	0..*	The file format or physical medium of the thesaurus
language	language	1..*	Codes showing languages supported by the thesaurus
publisher	String	0..*	Entity responsible for publication
relation	String	0..*	A related publication
rights	String	0..*	Copyright or other rights information
source	String	0..*	Resource from which the thesaurus was derived
subject	String	0..*	Index terms indicating the subject content
title	String	0..*	Title of the thesaurus
type	String	0..*	The genre of the vocabulary, e.g. "thesaurus"
<i>Associated classes of Thesaurus</i>			
ThesaurusArray	class	0..*	An array of sibling concepts forming part of the thesaurus
ThesaurusConcept	class	1..*	A concept forming part of the thesaurus

Table 2 Attributes and associations of *ThesaurusArray*

Element	Type	Occurrence	Description
ThesaurusArray	class	0..*	An array of sibling concepts forming part of the thesaurus
Attributes of ThesaurusArray			
identifier	String	1	A unique identifier for the array
Ordered	Boolean	1	A true/false label indicating whether the order of concepts in the array has to be maintained. Default is “false”
Associated classes of ThesaurusArray			
HasSuperOrdinate array	class	0..1	A higher-level array of which this array is a member
HasMember array	class	0..*	A lower-level array which is a member of this array <i>An array needs to have at least one member, but this can be a concept or another array.</i>
HasSuperOrdinate concept	class	0..1	A higher-level concept to which this array is subordinated
HasMember concept	class	0..*	A concept which is a member of this array <i>An array needs to have at least one member, but this can be a concept or another array.</i>
NodeLabel	class	0..*	A label showing the basis on which the concepts in the array have been chosen and ordered
◆ Attributes of NodeLabel			
◆ Notation	String	0..1	A symbolic notation attached to the node label, which may be used for sorting and display
◆ LexicalValue	String	1	The wording of the node label
◆ created	date	0..1	The date when the node label was created
◆ modified	date	0..1	The date when the node label was last modified
◆ lang	language	0..1	A code showing the language of the node label. This should be included if the thesaurus supports more than one language

Table 3 Attributes and associations of *ThesaurusConcept*

Element	Type	Occurrence	Description
ThesaurusConcept	class	1..*	A concept which is a member of the thesaurus
Attributes of ThesaurusConcept			
identifier	String	1	A unique identifier for the concept
created	date	0..1	The date when the concept was created
modified	date	0..1	The date when the concept was last modified
Status	String	0..1	Indication of whether the concept is candidate, approved, superseded, etc.
Notation	String	0..*	Symbolic notation attached to the concept, which may be used for sorting and display
TopConcept	Boolean	0..1	A true/false label indicating whether the concept is at the top of a hierarchy, i.e. has no broader concepts