INTERNATIONAL STANDARD

ISO/IEC 24800-3

First edition
2010-05-01

Information technology — JPSearch —

Part 3: Query format

Technologies de l'information — JPSearch —

Partie 3: Format d'interrogation

Reference number
ISO/IEC 24800-3:2010(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

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.

ISO/IEC 24800-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 24800 consists of the following parts, under the general title *Information technology — JPSearch*:

— *Part 1: System framework and components*

— *Part 2: Registration, identification and management of schema and ontology*

— *Part 3: Query format*

— *Part 4: File format for metadata embedded in image data (JPEG and JPEG 2000)*

The following part is under preparation:

— *Part 5: Data interchange format between image repositories*

Reference software will form the subject of a future Part 6.

Introduction

ISO/IEC 24800 aims to provide a standard for interoperability for still image search and retrieval systems. There are many systems which provide image search and retrieval functionality on computer desktops, on the World Wide Web (i.e., websearch), on imaging devices, and in other consumer and professional applications. Existing systems are implemented in a way that tightly couples many components of the search process. ISO/IEC 24800 provides an abstract framework search architecture that decouples the components of image search and provides a standard interface between these components.

Aligning image search system design to this standard framework facilitates the use and reuse of metadata; the use and reuse of profiles and ontologies to provide a common context for searching; and the provision of a common query language to easily search across multiple repositories with the same search semantics. It allows image repositories to be independent of particular system implementations; and allows users to easily move or upgrade their image management applications or to move to a different device or upgrade to a new computer.

This part of ISO/IEC 24800 contains the tools of the JPEG Query Format (JPQF) as an adaptation for the still images domain of ISO/IEC 15938-12:2008. It addresses the normative aspects of the Query Format and also illustrates some non-normative examples.

Information technology — JPSearch —

Part 3:
Query format

1 Scope

This part of ISO/IEC 24800, also known as "JPSearch Query Format (JPQF)", provides a standardized interface for image search and retrieval systems in three aspects: Input Query Format, Output Result Format, and Query Management. The Input Query Format provides users/systems with a set of precise input parameters to describe their search criteria in addition to a set of preferred output parameters to depict the return result sets. The Output Result Format provides users/systems with a set of output parameters to describe the aggregated return result sets for user presentation or machine consumption. The Query Management provides a means for selecting services (e.g., MPEG-7 database) or aggregated services (e.g., service provider that administers a set of different services) based on service properties (e.g., supported query format). The goal is to define a query language that provides the industry with a standardized format to accept and respond to user/system specification for image searches.

2 Normative references

The following referenced documents are indispensable 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.

ISO/IEC 15938-12, *Information technology — Multimedia content description interface — Part 12: Query format*

XQuery 1.0 and XPath 2.0 Data Model (XDM). W3C Recommendation, 23 January 2007. http://www.w3.org/TR/xpath-datamodel/

XML Path Language (XPath) 2.0. W3C Recommendation, 23 January 2007. http://www.w3.org/TR/xpath20/

3 Terms, definitions, abbreviated terms and conventions

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1
input query format
interface format going from a requester to one or more responders with two functionalities:

1) provision of a combination of syntax and semantics of the interface, through which the requester assigns search criteria and associated data;

2) provision of syntax and semantics of the interface, through which the requester specifies the format of the result data

NOTE The second functionality of the input query format provides tools by which the requester can express desired output format that should be conformant to the specification of the output result format.

3.1.2
output result format
interface format going from the responder to the requester as a response to the request specified by the input query format

NOTE Output Result Format defines all possible structures of return from responder to the requester. The structure of an actual return is decided by OutputDescription element in Input Query Format.

3.1.3
query management tools
tools to support the functionality required to manage the query transaction between the requesters and the responders

NOTE The query management tools do not include tools that are supported by network protocols. The query management tools are intended to be network agnostic and media agnostic.

3.1.4
multimedia content
coded representation of the information contained in or related to a multimedia resource in a formalized manner suitable for interpretation by human means

3.1.5
content
data and the associated metadata

3.1.6
multimedia resource
URI identifiable portion of raw data of an image in JPEG format, which is associated with a MIME Content-Type

3.1.7
metadata
data expressed as a schema valid XML instance to carry additional information describing a multimedia resource, where the schema defines the information model of the data

3.1.8
evaluation item
EI
unit against which the query condition is tested

NOTE By default, an EI is a multimedia content of the multimedia repository, but other types of EI are also possible: a multimedia content; a segment of a multimedia resource; and an XPath-item related to the multimedia content's metadata XML tree.

3.1.9
segment
spatial, temporal, or spatio-temporal unit of multimedia

EXAMPLE A spatial segment of an image.

3.1.10
XPath-item
node from the multimedia content's metadata XML tree or an atomic value

NOTE Details about the different types of nodes and atomic values can be found in the W3C Recommendation on XQuery 1.0 and XPath 2.0 Data Model. An XPath-item of a multimedia content's metadata may or may not be related to a multimedia content's segment. Also, a multimedia content or a multimedia content's segment may or may not be related to XML metadata. Within JPQF queries, XPath can be used to select a sequence of multimedia content's segments and/or metadata XPath-items. According to the W3C Recommendation on XQuery 1.0 and XPath 2.0 Data Model, a sequence is an ordered collection of zero or more XPath-items.

3.2 Abbreviated terms

MPEG-7 ISO/IEC 15938

JPQF JPEG Query Format (ISO/IEC 24800-3)

URI Uniform Resource Identifier (IETF Standard is RFC 2396)

URL Uniform Resource Locator (IETF Standard is RFC 2396)

XML Extensible Markup Language (W3C, http://www.w3.org/XML/)

3.3 Conventions

The Syntax defined in this part of ISO/IEC 24800 assumes the following Schema Wrapper:

```
<schema xmlns:jpqf="urn:jpeg:jpqf:schema:2008"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:mpqf="urn:mpeg:mpqf:schema:2008"
targetNamespace="urn:jpeg:jpqf:schema:2008" elementFormDefault="qualified"
attributeFormDefault="unqualified">
<import namespace="urn:mpeg:mpqf:schema:2008" schemaLocation="mpqf.xsd"/>
```

4 Structure and data model

4.1 Structure

As shown in Figure 1, JPSearch architecture is conceptually divided into four layers, which are User Layer, Query Layer, Management Layer and Content Layer. The interfaces marked with rounded pink boxes are within the scope of this part of ISO/IEC 24800. Data flow between each layer is defined as Input Query and Output Result. Where the nature of the data flow is requesting information, it is defined as Input Query and when the nature of the data flow is returning data upon a request, it is defined as Output Query. This part of ISO/IEC 24800 defines the format for input queries and output results which are used not only between the user layer and the query layer, but also between all other layers, as defined in the architecture, so that each layer of the JPSearch system can be implemented and maintained in a distributed way. This part of ISO/IEC 24800 also defines the messages for querying and providing information regarding the service description, service discovery and selection. The tools specifically serving these purposes are called Management tools.

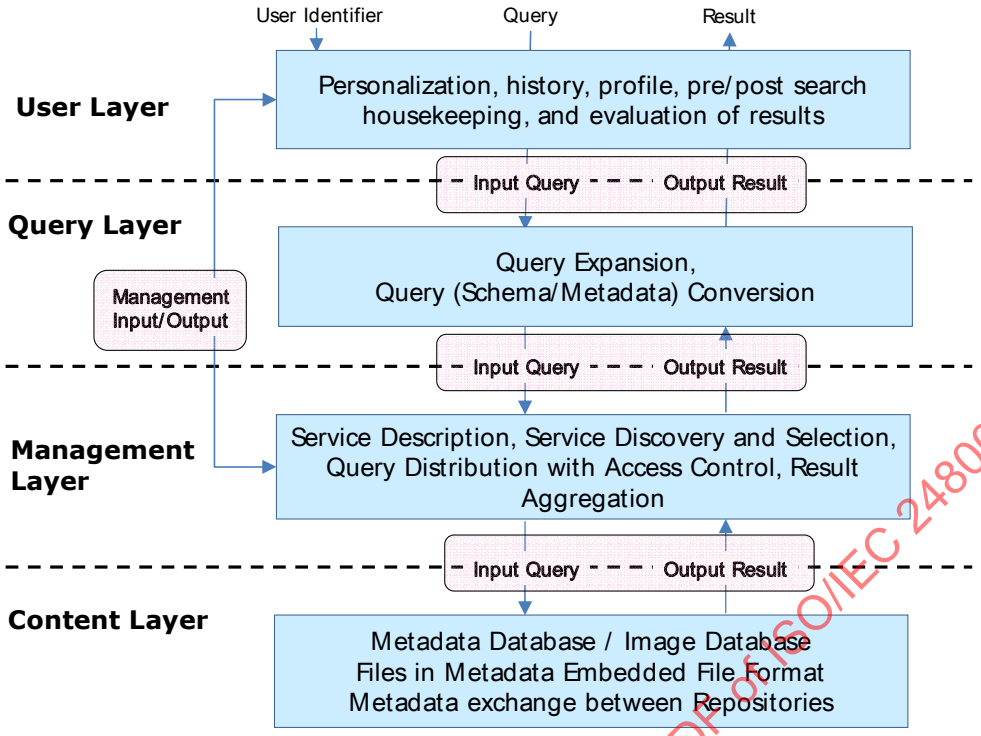

Figure 1 — JPSearch architecture and scope of this part of ISO/IEC 24800

The JPEG Query Fomat is defined as the complex type of element named JPEGQuery. The JPEGQuery contains a choice of InputQuery, FetchResult, OutputResult, and Management elements. This simplified structure of the JPEGQuery element is depicted in Figure 2.

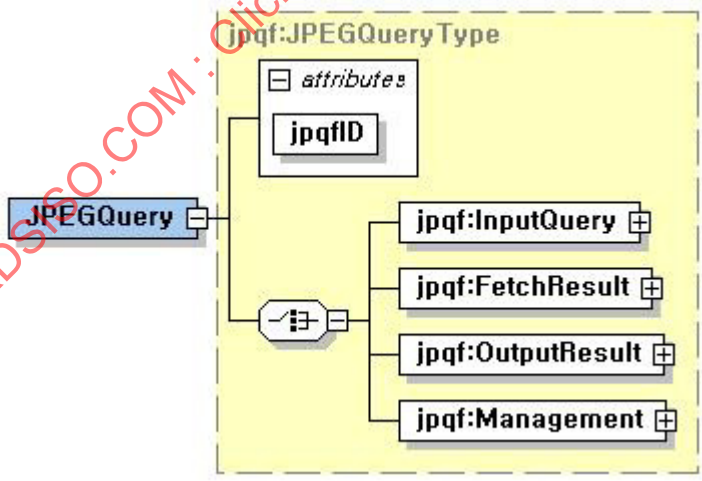

Figure 2 — Schema overview of the uppermost elements of the JPEG Query Format

Table 1 explains these four elements in more detail:

Table 1 — Top-Level JPEGQuery elements

`InputQuery`	The `InputQuery` element provides a container for describing a query request. Such a request can consist of a query condition and/or the output description which specifies the structure and content of the output result format and/or some declaration parts.
`FetchResult`	The `FetchResult` element is also part of the Input Query Format. It allows the user to request the results of a previous query issued using e.g., the asynchronous mode.
`OutputResult`	The `OutputResult` element describes the Output Result Format. It provides a container for output result returned from servers or returned from lower layers of the JPSearch architecture. It may contain not only query results but also any messages such as errors, exceptions or comments.
`Management`	A set of tools for the query management defined for the JPEG Query Format including service discovery, querying service capability, and service capability description. The structure defined for the `Management` element provides a container for Management Input Tools or the Management Output Tools.

Note that the `Input` and the `FetchResult` elements belong to the Input Query Format, which provides a container for describing a query request (e.g., the query condition and/or the output description which specifies the structure and content of the output result format).

The `Management` element of the JPEG Query Format represents the Query Management Tools of the Query Format as defined in ISO/IEC 15938-12:2008. It describes a set of tools for the query management including service discovery, querying service capability, and service capability description. Similar to the query tools (Input Query Format and Output Result Format), the management part distinguishes the tools for the request (Input Management Type) and the response (Output Management Type). Table 2 depicts the containing elements in more detail.

Table 2 — Elements in Query Management Tools

`Input`	The `Input` element, which is defined for the management tools, is intended to be sent from a requester to one or more responders.
`Output`	The `Output` element, which is defined for the management tools, is intended to be sent from a responder to one requester.

4.2 Data model

Processing and evaluation of a JPFQ query is executed against one or more multimedia repositories. Note, all introduced terms used within this Subclause are explained in 3.1.

JPQF assumes the same data model defined in 1.3 of ISO/IEC 15938-12:2008. For details of the data model, refer to 1.3 of ISO/IEC 15938-12:2008.

5 Root Element

5.1 Introduction

The `JPEGQuery` element serves as the root element of the JPEG Query Format. The root element shall be used as the topmost element in all messages transmitted. This applies on the one side to the input query format and the output result format of a query request/response as well as on the other side to the query management of the input/output.

5.2 Syntax

```
<element name="JPEGQuery" type="jpqf:JPEGQueryType"/>

<complexType name="JPEGQueryType">
  <choice>
     <element name="InputQuery" type="jpqf:InputQueryType"/>
     <element name="FetchResult">
        <complexType>
           <attribute name="queryID" type="anyURI"/>
           <attribute name="retrievePageNum" type="positiveInteger"/>
        </complexType>
     </element>
     <element name="OutputResult" type="mpqf:OutputQueryType"/>
     <element name="Management">
        <complexType>
           <choice>
              <element name="Input" type="jpqf:InputManagementType"/>
              <element name="Output" type="jpqf:OutputManagementType"/>
           </choice>
        </complexType>
     </element>
  </choice>
  <attribute name="jpqfID" type="anyURI" use="required"/>
</complexType>
```

5.3 Semantics

Semantics of the `JpegQueryType` type:

Name	*Definition*
`JPEGQuery`	Serves as the root element of the JPEG Query Format. The `JPEGQuery` element shall be used as the topmost element in any instance of JPEG Query Format.
`JPEGQueryType`	Specifies the syntax of the root element. Within this element, either one of the `InputQuery`, `FetchResult` or `OutputResult` element to specify the query and search result, or one of the `Input` or `Output` element of the `Management` element shall be instantiated, respectively.
`InputQuery`	Wraps the user query specified using the Input Query Format, which is to be sent from the requester to the responder.
`FetchResult`	Allows the user to request the result of a previous query. In case of a query request issued using the asynchronous mode (with the `immediateResponse` attribute of the `InputQueryType` type set to false), the `FetchResult` element shall be used to indicate the preparedness of the user to receive results. In case of a query request issued using the synchronous mode, the `FetchResult` element shall be used to receive individual pages of the result set (if paging is activated).
`queryID`	The requester is looking for a the result of a certain identifier in asynchronous mode. The attribute queryID specifies this identifier.
`retrivePageNum`	Specifies the number of page the requester wants the responder to return (optional). The default value is '1.'
`OutputResult`	Wraps the responder response to a user query as described using the Output Query Result Format. The OutputResult is sent from the responder to the requester.
`Management`	Describes the Query Management tools which may be exchanged between responders and requesters. Within this element only one of `Input` or `Output` element shall be instantiated.
`Input`	Wraps a user request for service capabilities.
`Output`	Wraps the response to a user request for service capabilities.
`jpqfID`	Specifies a unique identifier which is assigned to every message sent between a requester and a responder.

5.4 Example

The following examples show the use of the root element for the JPEG Query Format. The first three examples demonstrate the use of the `Query` element. First, a user query request is simulated by using the `Input` element. Note, that no details have been specified. Second, the result of a previous asynchronous request is going to be fetched by the user. Finally, the result set of a query request is demonstrated by using the `Output` element.

The last example shows the use of a service discovery request by using the `Management` element followed by an `Input` element. In this case the response would contain a list of all available services which are registered.

```
<JPEGQuery jpqfID="idForMyRequest1">
  <Input>
      . . .
  </Input>
</JPEGQuery>
```

```
<JPEGQuery jpqfID="idForMyRequest2">
  <FetchResult queryID="idFromAsyncRequest1"/>
  </JPEGQuery>
```

```
<JPEGQuery jpqfID="idForMyResponse1">
  <OutputResult>
      . . .
  </OutputResult>
</JPEGQuery>
```

```
<JPEGQuery jpqfID="idForMyRequest3">
  <Management>
    <Input>
      . . .
    </Input>
  </Management>
</JPEGQuery>
```

6 Datatypes

6.1 mimeType

6.1.1 Introduction

The `mimeType` type is defined to specify the syntax for a `mimeType` element which shall be used to identify the media type of a multimedia resource. This `mimeType` allows defining media types of image or text.

6.1.2 Syntax

```
<simpleType name="mimeType">
  <restriction base="string">
    <whiteSpace value="collapse"/>
    <pattern value='(image|text)?/([Xx]\-
                )?[!#$%&#x27;*+.0-9A-Z\^-~&#x7f;\-]+'/>
  </restriction>
</simpleType>
```

6.1.3 Semantics

Semantics of the `mimeType` type:

Name	*Definition*
`mimeType`	Specifies the syntax of the `mimeType` type, which is used to identify the type of multimedia resource.
`pattern value`	Specifies the allowed pattern for mimetypes and corresponds to RFC 2045 MIME (Part 1).

6.2 MediaLocatorType

6.2.1 Introduction

The `MediaLocatorType` type serves as a basic datatype for locating a multimedia resource, specifically image resource in the `MediaResource` element. This `MediaLocatorType` is identical to that of ISO/IEC 15938-12:2008, except that the `mimeTyppe` used in the `InlineMeidaType` type is replaced by the `mimeType` defined in this standard to limit the access to the media resources to image types.

6.2.2 Syntax

```
<complexType name="MediaLocatorType">
  <sequence>
     <choice minOccurs="0">
        <element name="MediaUri" type="anyURI"/>
        <element name="InlineMedia" type="jpqf:InlineMediaType"/>
     </choice>
  </sequence>
</complexType>
<complexType name="InlineMediaType">
  <choice>
     <element name="MediaData16" type="hexBinary"/>
     <element name="MediaData64" type="base64Binary"/>
  </choice>
  <attribute name="type" type="jpqf:mimeType" use="required"/>
</complexType>
```

6.2.3 Semantics

For the semantics, refer to 6.1.3 of ISO/IEC 15938-12:2008.

7 Tools of Input Query Format

7.1 Introduction

The functionality of specifying the query is provided by the two elements of InputQuery and FetchResult. This Clause only specifies the `InputQuery` element defined for the Input Query Format, as the FetchResult is defined inline within the JPEGQueryType and explained in the previous Clause. The `InputQuery` element is defined by the `InputQueryType` and is identical to the `InputQueryType` of ISO/IEC 15938-12:2008 except that the QueryCondition element of the InputQueryType is redefined for this standard. However, please note that there are new QueryByMedia and QueryByROI types defined as extensions of QueryType, which is again an extension of BooleanExpressionType. Note that Temporal type of queries are not allowed in this standard.

7.2 Syntax

```
<complexType name="InputQueryType">
  <sequence>
    <element name="QFDeclaration" type="mpqf:QFDeclarationType"
minOccurs="0"/>
    <element name="OutputDescription" type="mpqf:OutputDescriptionType"
minOccurs="0"/>
    <element name="QueryCondition" type="jpqf:QueryConditionType"
minOccurs="0"/>
    <element name="ServiceSelection" type="mpqf:ServiceSelectionType"
minOccurs="0"/>
  </sequence>
  <attribute name="previousAnswerID" type="anyURI" use="optional"/>
  <attribute name="immediateResponse" type="boolean" use="optional"
default="true"/>
  <attribute name="timeout" type="duration" use="optional"/>
</complexType>

<complexType name="QueryConditionType">
  <sequence>
    <element name="EvaluationPath" type="mpqf:xPathType" minOccurs="0"/>
    <element name="TargetMediaType" type="jpqf:mimeType" minOccurs="0"
maxOccurs="unbounded"/>
    <choice>
        <element name="Join" type="jpqf:JoinType"/>
        <element name="Condition" type="mpqf:BooleanExpressionType"/>
    </choice>
  </sequence>
</complexType>
```

For the syntax of `QFDeclarationType`, `OutputDescriptionType` and `ServiceSelectionType` types, refer to Clauses 8, 9, and 7.2 of ISO/IEC 15938:2008, respectively.

Note that the syntax of `QueryConditionType` and `JoinType` types are identical to those of ISO/IEC 15938-12:2008, except that the contained elements using `mimeType` are using the new `mimeType` type of this standard.

7.3 Semantics

The semantics of the elements and attributes defined for the `InputQueryType` type are identical to those of `InputQueryType` defined in 7.3 of ISO/IEC 15938-12:2008.

The semantics of the `QFDeclarationType`, `OutputDescriptionType` and `ServiceSelectionType` types are given in 8.3, 9.3, and 7.3 of ISO/IEC 15938-12:2008, respectively.

The semantics of the `QueryConditionType` and JoinType types are given in 10.3, and 12.13.3 of ISO/IEC 15938-12:2008, respectively.

8 Expression Types

8.1 Introduction

Expression types are designed to allow complex expressions in the query conditions. There are three basic types for contructing Query Condition, which are `BooleanExpressionType`, `ArithmeticExpressionType`, and StringExpressionType types. ISO/IEC 15938-12:2008 defines various

contructors of expression based on these three types. This standard reuses those contructors defined in ISO/IEC 15938-12:2008 with a few limitations. This Clause provides list of the expression contructors which are allowed in this standard and the contructors which are not allowed in this standard.

8.2 Tools for Expression

Name	*Definition and Examples*
`BooleanExpressionType`	Refer to Clause 10 of ISO/IEC 15938-12:2008.
`ArithmeticExpressionType`	Refer to Clause 10 of ISO/IEC 15938-12:2008.
`StringExpressionType`	Refer to Clause 10 of ISO/IEC 15938-12:2008.
`ArithmeticOperands`	Refer to 10.4.1 of ISO/IEC 15938-12:2008.
`BooleanOperands`	Refer to 10.4.2 of ISO/IEC 15938-12:2008.
`StringOperands`	Refer to 10.4.3 of ISO/IEC 15938-12:2008.
`DateTimeOperands`	Refer to 10.4.4 of ISO/IEC 15938-12:2008.
`DurationOperands`	Refer to 10.4.5 of ISO/IEC 15938-12:2008.
Extensions of `BooleanExpressionType` types: `AND`, `OR`, `XOR`, `NOT`	Refer to 10.5 of ISO/IEC 15938-12:2008.
Extensions of `ComparisonExpressionType` types: `GreaterThan`, `GreaterThanEqual`, `LessThan`, `LessThanEqual`, Equal, `NotEqual`, `Contains`	Refer to 10.6 of ISO/IEC 15938-12:2008.
Extensions of `ArithmeticExpressionType` types: `Add`, `Subtract`, `Multiply`, `Divide`, `Modulus`, `Abs`, `Ceiling`, `Round`, `Floor`	Refer to 10.7 of ISO/IEC 15938-12:2008.
`AggregateExpressionType`	Refer to 10.8 of ISO/IEC 15938-12:2008.
Extensions of `AggregateExpressionType` types: `AVG`, `StdDev`, `Variance`, `SUM`, `Count`, `MAX`, `MIN`	Refer to 10.8 of ISO/IEC 15938-12:2008.
`StringExpressionTypes`	Refer to 10.9 of ISO/IEC 15938-12:2008.
Extensions of `StringExpressionType` types: `UpperCase`, `LowerCase`	Refer to 10.9 of ISO/IEC 15938-12:2008.

9 Query Types

9.1 Introduction

Various types of queries including the query-by-example and query-by-free text are supported in this standard. Each of the specialized type of queries is defined as an extension of an abstract `QueryType` type. There are eight specific types of queries defined in ISO/IEC 15938-12:2008, among which only 6 types are supported exactly as defined in the ISO/IEC 15938-12:2008, in this standard. Furthermore, The `QueryByMedia` type is redefined in this standard to prevent the use of temporal media in the media example. Also, the `QueryByROI`

type is newly added in this standard. In this Clause, the list of supported query types is provided. The `QueryByMedia` and the `QueryByROI` types are defined in the following Clauses.

9.2 Supported Query Types

Name	*Definition and Examples*
`QueryTypes`	Refer to Clause 11 of ISO/IEC 15938-12:2008.
`QueryByMedia`	Refer to Clause 10. The `QueryByMedia` type defined in 11.4 of ISO/IEC 15938-12:2008 is not allowed to be used with this standard.
`QueryByDescription`	Refer to 11.5 of ISO/IEC 15938-12:2008.
`QueryByFreeText`	Refer to 11.6 of ISO/IEC 15938-12:2008.
`QueryByFeatureRange`	Refer to 11.7 of ISO/IEC 15938-12:2008.
`SpatialQuery`	Refer to 11.8 of ISO/IEC 15938-12:2008.
`TemporalQuery`	Not allowed to be used with this standard.
`QueryByXQuery`	Refer to 11.10 of ISO/IEC 15938-12:2008.
`QueryByRelevanceFeedback`	Refer to 11.11 of ISO/IEC 15938-12:2008.
`QueryByROI`	Refer to Clause 11

10 Query By Media

10.1 Introduction

The `QueryByMedia` type extends the abstract `QueryType` type. The `QueryByMedia` type enables a requester to realize a search based on a given example media resource. It provides also an attribute to indicate the search criteria regarding similar-match or exact-match. Note that difference to the `QueryByMedia` type of ISO/IEC 15938-12:2008 is that, in this type, the example media resource is restricted to the image domain only.

10.2 Syntax

```
<complexType name="QueryByMedia">
  <complexContent>
     <extension base="mpqf:QueryType">
        <choice>
           <element name="MediaResource" type="jpqf:MediaResourceType"/>
           <element name="MediaResourceREF" type="IDREF"/>
        </choice>
        <attribute name="matchType" use="optional" default="similar">
           <simpleType>
              <restriction base="string">
                 <enumeration value="similar"/>
                 <enumeration value="exact"/>
              </restriction>
           </simpleType>
        </attribute>
```

```
      </extension>
  </complexContent>
</complexType>

<complexType name="MediaResourceType">
  <complexContent>
     <extension base="mpqf:ResourceType">
        <sequence>
           <element name="MediaResource" type="jpqf:MediaLocatorType"/>
        </sequence>
     </extension>
  </complexContent>
</complexType>
```

10.3 Semantics

For the semantics of the `QueryByMedia` type, refer to 12.4.3 of ISO/IEC 15938-12:2008.

For the semantics of the `MediaResourceType`, refer to 8.3 of ISO/IEC 15938-12:2008.

11 Query By ROI

11.1 Introduction

The `QueryByROI` type extends the `QueryByMedia` type and describes a query operation that takes a media resource as input and allows the specification of a region of interest. During the evaluation of this query type the region of interest is required to be considered for search (exact or similar).

For the `SpatialRegionOfInterest` element, a region is defined by the `IntegerMatrixType` type which allows the specification of a list of positive integer values describing individual points. The amount of necessary integer values per point is defined by the `dim` (dimension) attribute of the `IntegerMatrixType` type. If the `dim` attribute is set to two then two successive integer values specify one point in 2 D space. If the dim attribute is set to three then three successive integer values specify one point in 3 D space. The individual points define the region where for instance for 2 D, three points identify a triangle, four points a rectangular, and so on. The order of the individual points is demonstrated by the examples below and goes contraclockwise.

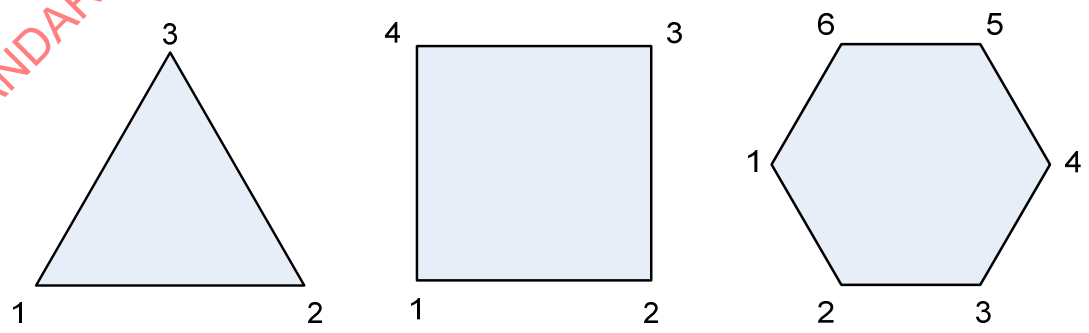

Figure 3 — Example point ordering for 2D objects

For 3D objects the individual 2D shapes need to be specified. The order is demonstrated by the example below and goes again contraclockwise. The letters mark the individual planes which need to be specified as 2D objects like above.

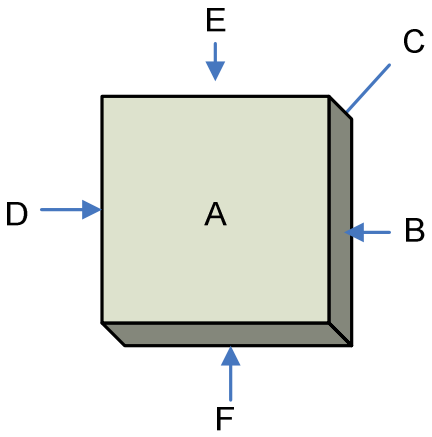

Figure 4 — Example plane ordering for 3D objects

11.2 Syntax

```
<complexType name="QueryByROI">
  <complexContent>
     <extension base="jpqf:QueryByMedia">
        <sequence>
           <element name="SpatialRegionOfInterest"
type="jpqf:IntegerMatrixType" maxOccurs="unbounded"/>
        </sequence>
     </extension>
  </complexContent>
</complexType>


<complexType name="IntegerMatrixType">
  <simpleContent>
    <extension base="mpqf:listOfPositiveInteger">
       <attribute name="dim" type="positiveInteger" use="required"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="listOfPositiveInteger">
  <list itemType="positiveInteger"/>
</simpleType>
```

11.3 Semantics

Semantics of the `QueryByROI` type:

Name	*Definition*
`QueryByROI`	The `QueryByROI` type extends the `QueryByMedia` type and describes a query operation that takes a media resource as input and allows the specification of a region of interest. During the evaluation of this query type the region of interest is required to be considered for search (exact or similar).
`SpatialRegionOfInterest`	Specifies a spatial region within a media resource where it is required that the media resource belongs to the spatial domain (image). The spatial region is described by the `IntegerMatrixType` type.
`IntegerMatrixType`	Specifies a spatial region. A spatial region is defined by the `IntegerMatrixType` type which allows the specification of a list of positive integer values describing individual points. The amount of necessary integer values per point is defined by the `dim` attribute of `IntegerMatrixType` type. The individual points define the region where two points mean a rectangular, three points a triangle and so on.
`Dim`	Attribute that defines the dimension of the individual points (e.g., 2 means that 2 integer values are required to define one point)
`listOfPositiveInteger`	Simple type that specifies a list of positive integer values which describes individual points. The amount of necessary integer values per point is defined by the `dim` attribute of the `IntegerMatrixType` type.

12 JoinType

12.1 Introduction

The Join operation is designed to define the filtering conditions which act over multiple sets of images. The JoinType defined in this Clause is identical to the Join Query defined in 12.13 of ISO/IEC 15938-12:2008, except that there are modifications to limit the search only over images.

12.2 Syntax

```
<complexType name="JoinType">
  <sequence>
     <element name="From" maxOccurs="2">
        <complexType>
           <sequence>
              <element name="EvaluationPath" type="mpqf:xPathType"
minOccurs="0"/>
              <element name="TargetMediaType" type="jpqf:mimeType"
minOccurs="0" maxOccurs="unbounded"/>
              <element name="Condition" type="mpqf:BooleanExpressionType"/>
           </sequence>
           <attribute name="id" type="ID" use="required"/>
        </complexType>
     </element>
     <element name="JoinCondition" type="mpqf:BooleanExpressionType"/>
  </sequence>
</complexType>
```

12.3 Semantics

For the semantics related to the JoinType, refer to 12.13.3 of ISO/IEC 15938-12:2008.

13 Tools of Output Result Format

`OutputResult` element provides a tool through which the query results are embedded and returned. The `OutputResult` element is of `OutputQueryType` defined in Clause 13 of ISO/IEC 15938-12:2008. For the syntax and semantics of the `OutputQueryType`, refer to Clause 13 of ISO/IEC 15938-12:2008.

14 Query Management Tools

14.1 Introduction

For the management of query, `Input` and `Output` elements are defined. The `Input` element is defined to provide service discovery capability and is of `InputManagmentType` type and the `Output` element is defined to provide response to the service discovery request through `Input` and is of `OutputManagementType` type. The `InputManagementType` and `OutputManagementType` types are identical to those of ISO/IEC 15938-12:2008, except that any element or type using `mimeType` element within the `InputManagementType` and `OutputManagementType` are redefined to use the `mimeType` defined in this standard.

14.2 Syntax

```
<complexType name="InputManagementType">
  <sequence>
     <element name="DesiredCapability" type="jpqf:CapabilityType"
minOccurs="0"/>
     <element name="ServiceID" type="anyURI" minOccurs="0"
maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="CapabilityType">
  <sequence>
     <element name="SupportedQFProfile" type="mpqf:CapabilityTermType"
minOccurs="0"/>
     <element name="SupportedMetadata" type="anyURI" minOccurs="0"
maxOccurs="unbounded"/>
     <element name="SupportedExampleMediaTypes" minOccurs="0">
        <simpleType>
           <list itemType="jpqf:mimeType"/>
        </simpleType>
     </element>
     <element name="SupportedResultMediaTypes" minOccurs="0">
        <simpleType>
           <list itemType="jpqf:mimeType"/>
        </simpleType>
     </element>
     <element name="SupportedQueryTypes" type="mpqf:CapabilityTermType"
minOccurs="0" maxOccurs="unbounded"/>
     <element name="SupportedExpressions" type="mpqf:CapabilityTermType"
minOccurs="0" maxOccurs="unbounded"/>
     <element name="UsageConditions" minOccurs="0" maxOccurs="unbounded">
        <complexType>
```

```
            <complexContent>
               <extension base="mpqf:TermType">
                  <attribute name="usageID" type="ID" use="optional"/>
               </extension>
            </complexContent>
         </complexType>
      </element>
   </sequence>
</complexType>

<complexType name="OutputManagementType">
   <sequence>
      <choice>
         <element name="AvailableCapability"
type="jpqf:AvailableCapabilityType" minOccurs="0" maxOccurs="unbounded"/>
         <element name="SystemMessage" type="mpqf:SystemMessageType"
minOccurs="0"/>
      </choice>
   </sequence>
</complexType>

<complexType name="AvailableCapabilityType">
   <complexContent>
      <extension base="jpqf:CapabilityType">
         <attribute name="serviceID" type="anyURI" use="required"/>
      </extension>
   </complexContent>
</complexType>
```

14.3 Semantics

The semantics of the types and elements of Query Management Tools are defined in Clause 14 of ISO/IEC 15938-12:2008.

15 Conformance

Any XML instance which is well-formed and valid against the schema and semantics provided by this part of ISO/IEC 24800 is considered to be conformant to this part of ISO/IEC 24800.

Annex A
(normative)

System Messages

A.1 Status messages

Refer to A.1 of ISO/IEC 15938-12:2008.

A.2 Warning messages

Refer to A.2 of ISO/IEC 15938-12:2008.

A.3 Exception messages

Refer to A.3 of ISO/IEC 15938-12:2008.