Appendix 2: Comparison of key CAQDAS software functionalities

			ATLAS.ti 5	Nvivo 2	Max QDA	QDA Miner	Qualrus	Trasana 2.12
1	**Data Source**		Primary Documents	Documents	Document System	Documents	Documents	Series
2	**Data Type / Format**		Plain Text Rich Text HTML Picture ACSII/ANSI Audio Video UNIX Portable Images	Plain Text Rich Text	Rich Text	Plain Text Rich Text Word Document HTML Word Perfect	Pain Text Rich Text Picture HTML Audio Video	Audio Video
3	**Concept Tag(s)**		Codes	Nodes	Codes	Codes	Codes	Keywords
4	**Code Structure**		Single level	Free, Tree & Case nodes	Hierarchical structure	Hierarchical structure with 1 level	Single level	Hierarchical structure with 1 level
5	**Links**	*between*	Quotations Codes Families Memos	Documents Nodes DataBites Memos	Codes Memos Texts	Codes Comments Variables	Codes Memos	Keywords Clips
6	**Notes or**		Memos	Documents	Codes	Codes	Documents	Comments

	Comments	Relationships Links Families Networks	Nodes Models Links Memos	Memos Documents	Variables	Codes Links Memos	
7	**Link Names**	is associated with : == is part of : [] is cause of : => is a : isa noname is property of : *}	is coded by equals to : = not equal to : not = is an extract Ancestor of Descendant of Children of Siblings of	None	equal to enclosing included in overlapping followed by preceeded by near	causes { default } excludes { default } isa { default } User defined relations functionality is available	None
8	**Multimedia Compatibility**	Yes *(see 2)*	No *(see 2)*	No *(see 2)*	No; but includes HTML *(see 2)*	Yes (see 2)	Yes (only)
9	**User Interface**	Complex	Cumbersome	Fairly Easy	Intuitive	Fairly Easy	Fairly Easy
10	**Coding Techniques**	In vivo Coding Open Coding List Coding SuperCode	In vivo Free Coding Automatic Coding Contextual Coding	In Vivo Coding Free Coding Automatic Coding Contextual Coding	Open Coding Automatic Coding Contextual Coding Variable	Open Coding Automatic Coding Contextual Coding List Coding	Open Keywording

Appendix 3: Comparison of existing and relevant metadata schema

	Data Document Initiative ***DDI***	**Text Encoding Initiative** ***TEI***	**Qualitative Data Interchange Format** ***QDIF***	**UKDA Quali XML Schema** ***QDML***	**Dublin Core** ***DC***	**Metadata Encoding and Transmission Standard** ***METS***
Main Purpose	To produce a meta-data specification for the description of social science data resource based on XML standards to provide effective, efficient use of data set.	The main purpose of TEI is to develop the guidelines for encoding machine-readable texts to the humanities & social sciences and to digitally describe the text using XML.	The main purpose is to exchange of annotated textual document collections, with special reference to qualitative social science data.	The main purpose of the schema is to expose the digital qualitative data to make them fully shareable and exploitable	The main purpose of this standard is to facilitate the finding, sharing and management of information.	The main purpose is to create a digital object standard for encoding structural, descriptive and administrative metadata along with primary content.
Key Functions	*a*. Interoperability *b.* Richer Content *c.* Single document with Multi purpose e.g. SAS/SPSS/Stata *d.* Precision in searching	*a.* Store, analyse and share information with others. *b.* Data Preservation. *c.* Increasing availability of a text. *d.* Data Portability. *e.* Enhancing with *descriptive* nature.	*a.* Interoperability. *b.* Data Sharing. *c.* Data Exchange.	*a.* Data searching *b*. Data sharing *c*. Anonymising Data d. Data Publishing *e*. Preservation	a. Simplicity of creation and maintenance b. Limpid semantics c. Extensibility d. International scope.	a. Creating dynamic interfaces using XSLT. b. Digital Preservation. c. Interoperability. d. Wrapping-up the digital content e. Richer description
Metadata Markup	A <*codebook*> defining various elements about the study.	The <*teiHeader*> defining various elements about the study.	The <*dataBundle*> holds all the elements that define the study.	The <*teiHeader*> and <*text*> defines all the elements that describe the study.	It uses 15 different metadata elements to describe the data source.	The <*mets*> element at the top constitutes the 7 different sections to define the data descriptively.
Stand-off Annotations	No.	Yes.	Yes.	Yes.	No.	Yes.
Metadata Specification	*1.* Document Description *2.* Study Description *3.* Files Description *4.* Variable Description *5.* Other study related	*1.* File Description. *2.* Encoding Description. *3.* Profile Description. *4.* Revision Description.	*1.* Metadata *2.* Primary Documents *3.* Selections *4.* Codes *5.* Relation names *6.* Relations	<u>TEI Header</u> : *1.* File Description *2.* Profile Description. <u>Text</u> :	*1.* Title *2.* Creator *3.* Subject *4.* Description *5.* Publisher *6.* Contributor	*1.* METS Header. *2.* Descriptive Metadata *3.* Administrative Metadata *4.* File Section

	material			1. Body	7. Date 8. Type 9. Format 10. Identifier 11. Source 12. Language 13. Relation 14. Coverage 15. Rights	5. Structural Map 6. Structural Links 7. Behavior
Limitations	1. Survey data bias 2. No referencing of identical variables represented in more than one study. 3. Modularity 4. Extensibility lacking 5. Metadata Modeling.	1. Complex 2. False proximity 3. Confusing to read 4. Allows annotate points, not ranges.	Input and output of QDIF itself is currently unsupported. However, the only way of currently reading a Data Bundle is to import an ATLAS.ti project.	1. Interoperability 2. Data exchange within various CAQDAS tools	1. Generic model. 2. Detail Description lacking 3. Not rich enough.	1. Tightly structured.

Appendix 4: WP3 Glossary

ASCII	American Standard Code for Information Interchange
CAQDAS	Computer-Assisted Qualitative Data Analysis Software
DDI	Data Documentation Initiative
DTD	Document Type Definition
ESDS	Economic and Social Data Service
ESRC	Economic and Social Research Council
HTML	HyperText Markup Language
JISC	Joint Information Systems Committee
METS	Metadata Encoding & Transmission Standard
QuDEx	Qualitative data exchange schema
ODaF	Open Data Foundation
RTF	Rich Text Format
TEI	Text Encoding Initiative
TIFF	Tagged Image File Format
XML	eXtensible Markup Language
XSD	XML schema Definition
XSL	Extensible Stylesheet Language
XSLT	XSL Transformations

Appendix 5: WP3 Stakeholders and evaluators

CAQDAS vendors

Thomas Muhr, ATLAS.ti, Berlin, Germany

Lynne Richards, QRS (Nvivo, NU*DIST), Australia

Udo and Anne Kukartz, Max QDA, Marburg, Germany

Normand Pelandeau, Provalis Research (QDA Miner), Montreal, Canada

Ed Brent, Qualrus, USA

Sharlene Hess-biber and Richard Gaskin, Researchware Inc. (HyperResearch), Boston, USA

Mark Bernstein, Eastgate Systems Inc (Tinderbox), Watertown, MA, USA

David Woods, Transana, Wisconsin, USA

Alex Fenton, WeftQDA, Cambridge, UK

Open Data Foundation

Arofan Gregory, Arizona

Chris Nelson, UK

Jack Gager, New York

Academic projects

Prof. Graham Gibbs, Victoria Climbie project, Huddersfield, UK

Andy Crabtree and Professor Chris Greenhalgh, ESRC NCESS Digital Records Research Node, Nottingham, UK

Patrick Carmichael, CAET, Cambridge, UK

Rene Van Horik, Data Archiving and networked Services (DANS), MIXED project, Netherlands

Andrew Smith, Baden Hughes, Desmond Schmidt, Language Technology Group, Melbourne ands Queensland, Australia

Mary Vardigan, ICPSR, Michigan, USA

Wendy Thomas, Minnesota Population Centre, Minnesota, USA

APPENDIX 6: QuDEx v03.00 Reference

Contents

QuDEx v03.00 Reference **1**
Contents **1**
Document Control **2**
Key: 2
Introduction **3**
Attributes **4**
Standard Attributes Group 4
Resource Attributes Group 5
1. <qudex> **6**
2. <resourceCollection> **6**
2.1 <sources> 6
2.1.1 <source> 6
2.2 <memoSources> 7
2.2.1 <memoSource> 7
2.3 <documents> 7
2.3.1 <document> 7
3. <segmentCollection> **7**
3.1 8
3.1.1 <text> 8
3.1.2 <audio> 8
3.1.3 <video> 10
3.1.4 <image> 11
3.1.5 <xml> 11
4. <codeCollection> **12**
4.1 <code> 12
5. <memoCollection> **12**
5.1 <memo> 12
5.1.1 <memoDocumentRef> 12
5.1.2 <memoText> 12
6. <categoryCollection> **12**
6.1 <category> 13
7. <relationCollection> **14**
7.1 <objectRelation> 14
8. Relevant Acronyms **16**

Document Control

01.00	Approved for release by Louise CortI	2008-02-29
00.03	HL'H: Merged in the content from the QuDEx glossary document	2008-02-19
00.02	HL'H: Tracked change version based on internal change documentation created for approval of LC and AB	2008-02-15
00.01	Based on version 1.0 of the QuDEx 2.0 reference. Amendments by Hervé L'Hours (HL'H) based on revisions to QuDEx by Louise Corti (LC) and Angad Bhat (AB) of UK Data Archive (UKDA) and Arofan Gregory of the Open Data Foundation (ODaF). Amended with changes to the standard described in the QuDEx v3.00 release notes	2008-02-14

Key:

- Element names are *italicised*
- Attribute names are preceded by the @ symbol.
- **Imp':** indicates implementation advice beyond the scope of schema validation
- **Definition:** A number of definitions are duplicated as xs:documentation in the QuDEx schema

Introduction

The purpose of this document is to support the development of the QuDEx xml schema by explaining the elements of the schema and related concepts in more detail.

QuDEx

QuDex is the qualitative data exchange model for the archiving and interchange of data and metadata between CAQDAS packages. The UK Data Archive has been working on this standard together with the social science data archiving community and international XML schema experts. We are particularly grateful to the JISC, ESRC and the Open Data Foundation (ODaF) for all their support.

DExT: Data Exchange Tools

Versions 1.0 to 1.3of the draft QuDEx standard was developed under the auspices of the DExT project. The DExT project undertaken in 2007 was funded under the JISC Repositories Programme which aimed to develop, refine and test models for data exchange for both survey and qualitative research data based on XML schema. The 12 month long R&D grant aimed to develop a draft model and schema that might enable the exchange and archiving of a software-neutral format for qualitative data. At present there is no interchange common format and no internationally agreed metadata schema for describing collections of raw and annotated qualitative data collections. The primary goal for the initial versions of QuDEx has been to develop a standard suitable for interchange and long term archiving of qualitative data.

All documentation referencing 'QuDEx' refers to the qualitative side of the DExT Project, to keep distinct from the quantitative DEXT project work done on survey micro data. The test data selected for this project are from the social sciences, but these formats are typically found across all domains of primary research.

CAQDAS: Computer Assisted Qualitative Data Analysis Software

CAQDAS packages were developed in the late 80s typically by keen qualitative researchers and the resulting software thus embodies different methodological and analytical approaches. The past decade has seen a huge take up of the use of these packages in research and in teaching and in the UK the CAQDAS Networking group has provided an invaluable information portal, forum and outreach program for helping get users started. While a basic common denominator set of functions can be seen across the software, various new functions have been added to some and not others. Thus each has its own flavour, and also terminology.

The core functions in CAQDAS packages are: coding, classifying, memoing, search and retrieval. Vendors have been consulted throughout the QuDex development process.

METS: Metadata Encoding and Transmission Standard

METS is a standard for encoding descriptive, administrative, and structural metadata regarding digital objects, expressed using the XML schema language. The standard is maintained in the Network Development and MARC Standards Office of the Library of Congress, and is being developed as an initiative of the Digital Library Federation. Rather than attempting to perform every function required by researchers QuDEx aims to deliver core functionality while other metadata standards are used as appropriate. It is recommended that all materials and metadata relating to a file are packaged as 'complex objects' using a standard such as METS.

Future

Following a final round of comments to this latest release, the schema work will be transferred under a working group of the DDI committee(http://www.icpsr.umich.edu/DDI/). The schema will also be supported by the Open Data Foundation (ODaF) and any further versions and tools will be hosted on their web site (http://www.opendatafoundation.org/). We welcome anyone interested in participating in further developments to contact us and get involved (via corti@essex.ac.uk) .

Attributes

A number of attributes are commonly used within the QuDEx standard. Attribute groups in use are defined below. Other attributes are defined at their first appearance in the documentation.

@ id: the unique identifier attribute is used globally for all QuDEx elements.

Standard Attributes Group

The "standard attribute" group is designed to support the management of complex layers of analysis by multiple authors within a single QuDEx instance.

@ cdate:	the date and time the instance of the element was created
@ mdate:	the last date and time the instance of the element was modified
@ creator:	the original creator of the instance of the element or the author of the relevant resource
@ label:	a human readable string for the element in general or its specific contents
@ displayLabel:	a version of the label text appropriate for display, for example in a user interface
@ language:	this caters for describing the overall language of the study while permitting element level variations such as defining a segment, memo or code as being in a different language

Imp': If users or developers choose to implement @label and @display label, their local usage should be documented as comments within QuDEx output

e.g. "@labels is more detailed and is used for output to the administrative back end of our application; @displayLabel is used for all customer-facing applications"

The standard attribute group is used in the following elements

- qudex
- source
- memoSource
- document
- segment
- code
- memo
- category
- objectRelation

Resource Attributes Group

The following attribute group is common to all resourceCollection child elements

@ sequence: an optional integer indicating the sequence of this file relative to the others in its file group

@ size: an optional long integer specifying the size of the file in bytes.

@ location: the location of the source material

@ locType: the type of locator used to reference a file. It must have one of the following values:

- "urn" Uniform Resource Name
- "url" Uniform Resource Locator
- "purl" persistent URL
- "handle" CNRI Handle
- "doi" Digital Object Identifier
- "other" a form of locator not specified above

@ otherLocType: If @locType = "other" then @otherLocType is recommended

@ checksumType: an optional attribute specifying the checksum algorithm used to produce the value contained in @checksum. @checksumType must contain one of the following values:

- "haval"
- "md-5"
- "sha-1"
- "sha-256"
- "sha-384"
- "sha-512"
- "tiger"
- "whirlpool"
- "crc32"
- "adler3"
- "other"

@ otherChecksumType: if @checkSumType" = "other" then @otherChecksumType is recommended

@ checksumValue: an optional string attribute providing a checksum value for the included file

@ mimeType: an optional string to describe the source mimeType

@ ResourceType: in addition to @mimeType both *sources* and *memoSources* have the simple source definition @ResourceType. This is an enumerated list of …

- "text"
- "audio"
- "video"
- "xml"
- "other"

@otherResourceType: if @resourceType = "other" then @otherResourceType is recommended

<qudex>

Definition: The root element; a 'wrapper' for all other elements of the QuDEx Schema. Each top level element in QuDEx is defined as a ‘collection’ and must appear in the order outlined below.

Sub-elements: resourceCollection, segmentCollection, codeCollection, memoCollection, categoryCollection, relationCollection
Attributes: id, ‘standard attribute group’, status

@ status: an optional string indicating the status of the QuDEx file

Imp’: Status is mainly for internal processing purposes e.g. draft, complete, superseded

<resourceCollection>

Definition: The *resourceCollection* section lists and locates all content available to the QuDEx file. A *source* points to the original location of the resource while each author working on the QuDEx file is assigned a surrogate *document* which points to the relevant *source*. The child elements *sources* and *memoSources* contain direct references to the files under analysis; the *documents* section contains their surrogates.

Sub-elements: sources, memoSources, documents
Attributes: id

<sources>

Definition: Container for all *source* elements

Sub-elements: source
Attributes: id

<source>

Definition: Each *source* represents a single resource available to the QuDEx file. A *source* may be anything identifiable by a URI which is to be analysed using CAQDAS methodology. Each time a new @creator uses a *source*, a *document* surrogate is created. Other sections of the QuDEx instance reference the *document* rather than the *source.*

Sub-elements: none
Attributes: id, ‘standard attribute group’, ‘resource attribute group’

<memoSources>

Definition: Container for all memoSource elements.

Sub-elements: memoSource
Attributes: id

<memoSource>

Definition: In addition to creating a pure text *memo* and embedding it in the QuDEx file (inline memo) it is possible to reference an external *memo* using a *memoSource*. In QuDEx an external *memo* (held outside the body of the QuDEx file) is managed in the same way as an external *source*. A memoSource may be anything identifiable by a URI. Each time a new @creator uses a *memoSource*, a *document* surrogate is created. Other sections of the QuDEx instance reference the *document* rather than the *source*.

Sub-elements: none
Attributes: id, 'standard attribute group', 'resource attribute group',

<documents>

Definition: Container for all *document* elements

Sub-elements: document
Attributes: id

<document>

Definition: A *document* is a surrogate version of a *memo* or *memoSource* specific to an individual author. The first time each author (@creator) uses a *source* or *memoSource* in QuDEx a surrogate *document* is created. This *document* is referenced by the various collection sections. The *document* concept permits the controlled re-use and analysis of each *source* and *memoSource* within a single QuDEx instance. A *document* may refer to any underlying resource (*source* or *memoSource*) that can be referenced by a URI including text, audio, video, still images and XML.

Sub-elements: none
Attributes: id, 'standard attribute group', resourceRef, documentType

@ resourceRef: an **@**id reference denoting which resource (either *source* or *memoSource)* the document is based on

@ documentType: defines whether the *document* is the surrogate of a *source* or a *memoSource.* One of…

- "source"
- "memoSource"

<segmentCollection>

Definition: The parent element for all *segment*s.

Sub-elements: segment,
Attributes: id

**

Definition: A subset of a *document* (text, audio, video or image) under analysis defined in a manner appropriate to the format (text, audio, video, image or xml). *Segment*s may overlap and multiple *memo*s and *code*s may be assigned to a *segment*.

Sub-elements: text, audio, video, xml, image
Attributes: id, 'standard attribute group'

<text>

Definition: Container for text document segment information.

Sub-elements: lineParam, characterParam
Attributes: id, src

@ src: an **@** id reference to the relevant *document*

<lineParam>

Definition: Text document *segment* parameters based on lines and characters.

Sub-elements: none
Attributes: id, startOffset, startLineOffset, endOffset, endLineOffset

@ startLine: number of lines from beginning of the document
@ startOffset: number of characters from start of the appropriate line
@ endLine: number of lines from beginning of the document
@ endOffset: number of characters from start of the appropriate line

Imp': a *LineParam* may be appropriate if carriage returns/line feeds (CR/LF) exist and are retained by the CAQDAS analysis package.

<characterParam>

Definition: Text document *segment* parameters based on characters.

Sub-elements: none
Attributes: id, startCharOffset, endCharOffset

@ startCharOffset: number of characters from the start of the document
@ endCharOffset: number of characters from the start of the document

Imp': *characterParam* may be appropriate independent of the presence of CR/LF information

<audio>

Definition: Container for audio segment information

Sub-elements: param
Attributes: id, src

<param>

Definition: audio *segment* parameters defined by start and end points for a clip. Clip annotation may be taken from the enumerated list or user-defined

Sub-elements: none
Attributes: id, clipType, otherClipType, clipBegin, clipEnd

@ clipType: the annotation method used to describe the video clip, one of…

- "byte"
- "smil"
- "midi"
- "time"
- "tcf"
- "other"

@ otherClipType: recommended if @clipType = "other"
@ clipBegin: a string, appropriate to the @clipType, describing the start of the clip
@ clipEnd: a string, appropriate to the @clipType, describing the end of the clip

<video>

Definition: Container for video segment information.

Sub-elements: param
Attributes: id, src

<param>

Definition: video *segment* parameters defined by start and end points for a clip. Clip annotation may be taken from the enumerated list or user-defined.

Sub-elements: none
Attributes: id, clipType, otherClipType, clipBegin, clipEnd

@ clipType: the annotation method used to define the video clip, one of

- "byte"
- "smil"
- "midi"
- "smpte-25"
- "smpte-24"
- "smpte-df30"
- "smpte-ndf30"
- "smpte-df29.97"
- "smpte- ndf29.97"
- "time"
- "tcf"
- "other"

<image>

Definition: Container for image segment information.

Sub-elements: area
Attributes: id, src

<area>

Definition: Image *segment* parameters are defined by the shape and coordinates of an area within the image.

Sub-elements: none
Attributes: id, shape, cords

@ shape: the shape of the segment

- `"Rect"` a rectangular area
- `"Circ"` a circular area
- `"poly"` an irregular polygon area

@ coords: a comma-delimited string of coordinates

The @shape should be used as in HTML 4.
Reference: http://www.w3.org/TR/html401/struct/objects.html

- for `rect:` left-x, top-y, right-x, bottom-y.
- for `circ:` center-x, center-y, radius. **Note.** When the radius value is a percentage value, user agents should calculate the final radius value based on the associated object's width and height. The radius should be the smaller value of the two.
- for `poly:` x1, y1, x2, y2, ..., xN, yN. The first x and y coordinate pair and the last should be the same to close the polygon. When these coordinate values are not the same, user agents should infer an additional coordinate pair to close the polygon.

Coordinates are relative to the top, left corner of the object. All values are lengths. All values are separated by commas.

<xml>

Definition: Container for XML segment information.

Sub-elements: range.
Attributes: id

<range>

Definition: XML segment parameters are defined by an Xpointer range

Sub-elements: none.
Attributes: id, src, xPtrExp

@ xPtrExp the Xpointer expression describing the xml range

<codeCollection>

Definition: The parent element for all *code*s.

Sub-elements: code
Attributes: id

<code>

Definition: A short alphanumeric string, usually a single word; may be assigned to a segment or document though assignment is not required. A code may optionally be taken from a controlled vocabulary defined under @ authority.

Sub-elements: none
Attributes: id, 'standard attribute group', authority

@ authority: provides for the definition of a controlled vocabulary from which codes are to be selected

<memoCollection>

Definition: The parent element for all *memo*s; these may be pure text and embedded in the QuDEx file (inline memo) or may refer to external files.

Sub-elements: memo
Attributes: id

<memo>

Definition: A text string internal to the document (inline memo) or an externally held document (external memo) which may be assigned to a segment, code, document, category or to another memo.

Sub-elements: memoDocumentRef, memoText
Attributes: id, 'standard attribute group'

<memoDocumentRef>

Definition: A reference to an external file to be used as a *memo*; the reference must be to a *memoSource* via the id of a *document.*

Sub-elements: none
Attributes: id, src

<memoText>

Definition: An inline memo held as a single variable length alphanumeric string.

Sub-elements: none
Attributes: id

<categoryCollection>

Definition: The parent element for all categories.

Sub-elements: category
Attributes: id

<category>

Definition: An alphanumeric string (stored in @label) assigned to one or more documents. Categories may be hierarchically nested. Documents contained within a category are referenced using @documentRefs. Nested categories are referenced using @categoryRefs.

Sub-elements: none
Attributes: id, 'standard attribute group', documentRefs, categoryRefs

@ documentRefs: The list of document **@** IDs grouped within this category
@ categoryRefs: The list of category **@** IDs nested under this category

<relationCollection>

Definition: The parent element for all relationships between objects.

Sub-elements: objectRelation
Attributes: id

<objectRelation>

Definition: For the purposes of a *relation* all of the following are considered to be 'objects'

- A *document*: surrogate of a *source* or *memoSource*
- A *segment* within a *document*
- An assigned value: *code*, *memo*, *category*, *relation*

A relation is a link between two objects in a QuDEx file. Each object is either the start or end point of a relation (source vs target). Every relation may, optionally, have a name.

Sub-elements: none
Attributes: id, 'standard attribute group', objectType, relationName, otherRelationName objectSource, objectTarget

@objectType: The type of objects involved in the relationship. The enumerated list is a concatenation of the object names

- `segmentCode`
- `segmentDocument`
- `segmentCategory`
- `segmentMemo`
- `codeSegment`
- `codeCode`
- `codeDocument`
- `codeCategory`
- `codeMemo`
- `documentSegment`
- `documentCode`
- `documentDocument`
- `documentCateogry`
- `docuementMemo`
- `memoSegment`
- `memoCode`
- `memoDocument`
- `memoCategory`
- `memoMemo`
- `categorySegment`
- `categoryCode`
- `categoryDocument`
- `categoryCategory`

@relationName: an enumerated list of relationship types. Other types of relationship may be added by the user and described in otherRelationName

- isA
- isChildOf
- isParentOf
- isMemberOf
- isDerivedFrom
- isRevisionHistoryFor
- isCriticalReviewOf
- isOverviewOf
- isVersionOf
- isFormatOf
- isReferencedBy
- isBasedOn
- isPartOf
- isAssociateOf
- isInstanceOf
- isLinkedTo
- isRelatedTo
- isAssignedTo
- isEqualTo
- isNotEqualTo
- hasFormat
- hasVersion
- hasPart
- hasBibliographicInfoIn
- references
- requires
- other

@otherRelationName: the use of this attribute is required if @relationName = “other”

@objectSource: the source object in the relationship

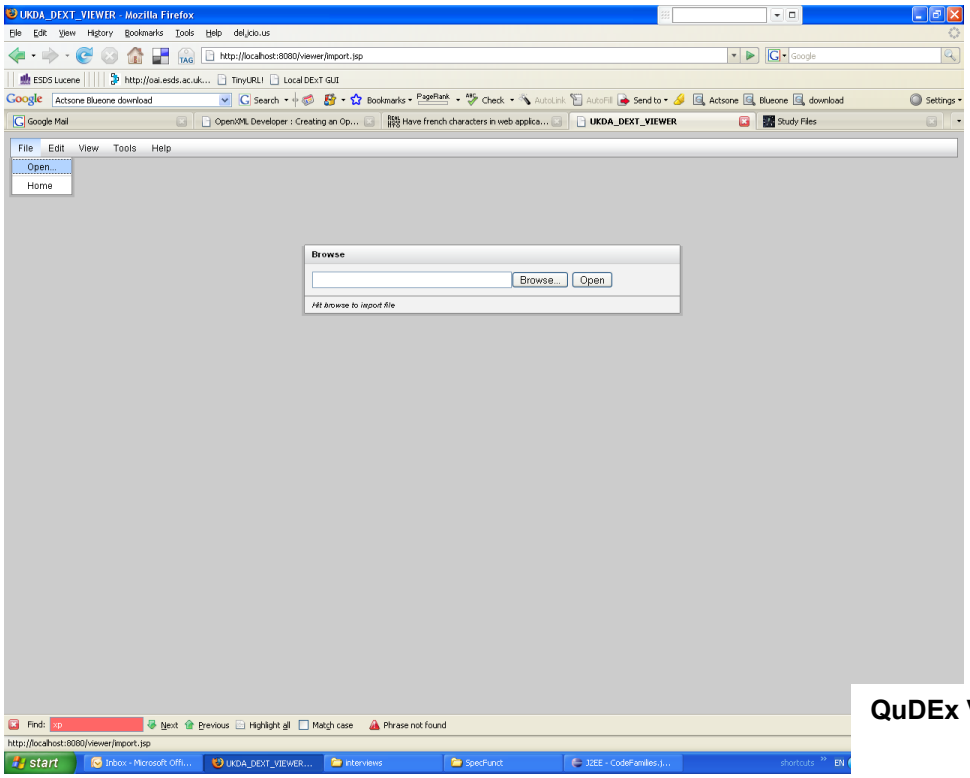
@objectTarget: the target object in the relationship

A. Appendix: Relevant Acronyms

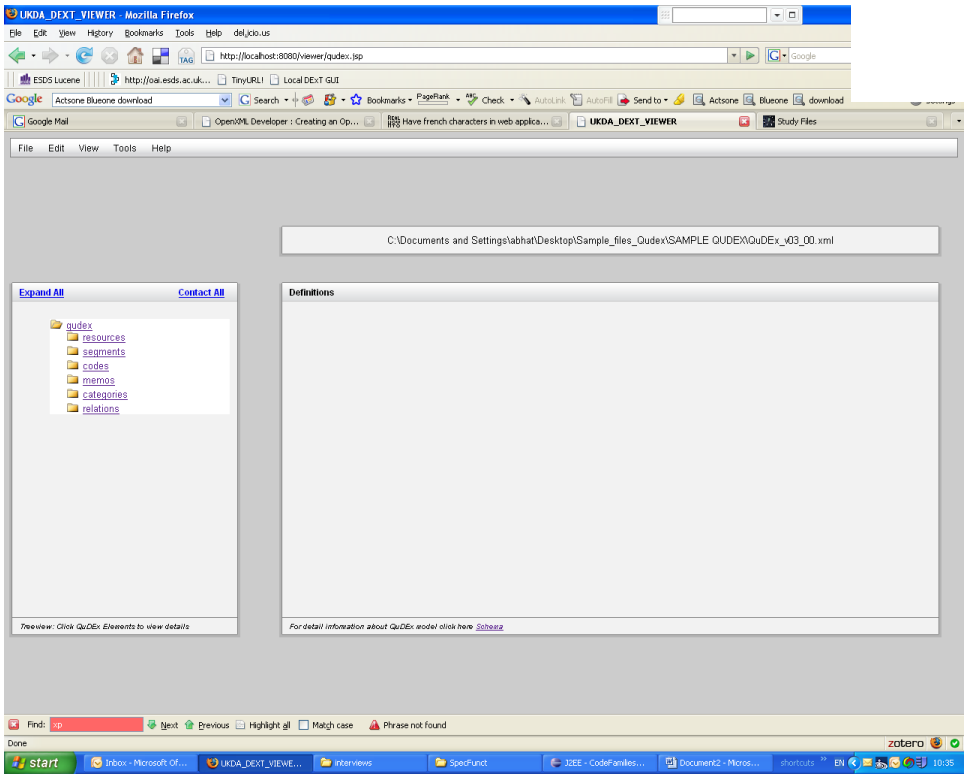
The acronyms below may be found in various documents supporting the DExT project and QuDEx standard.

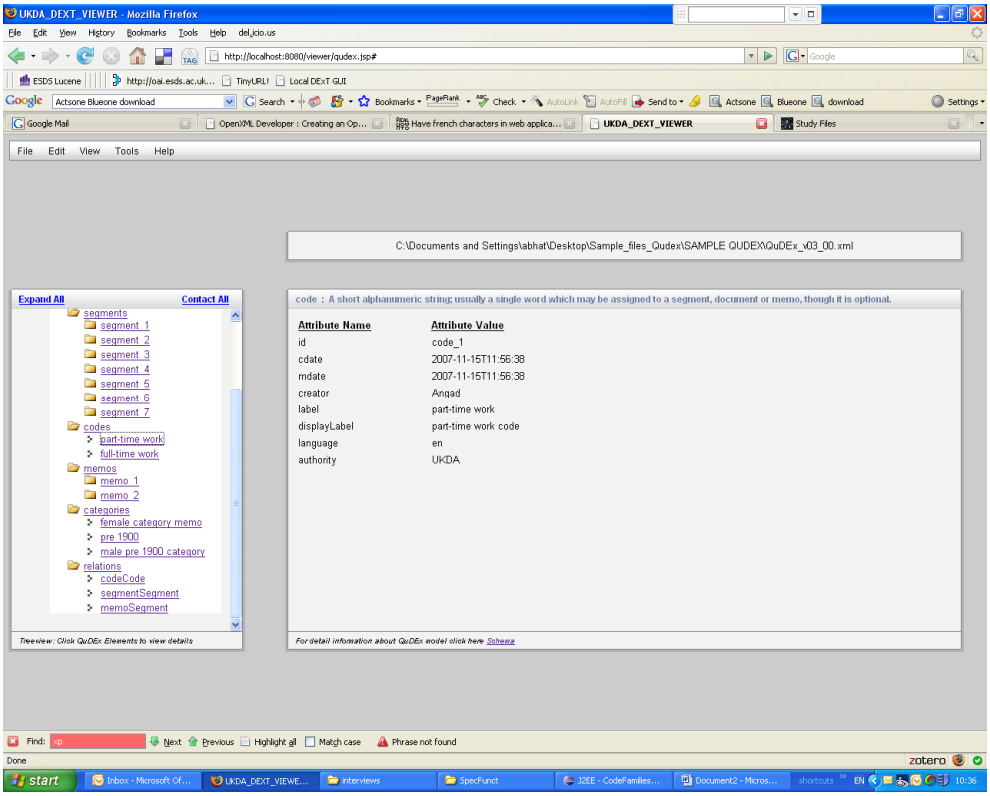
- **ASCII** American Standard Code for Information Interchange
- **CAQDAS** Computer-Assisted Qualitative Data Analysis Software
- **DDI** Data Documentation Initiative
- **DTD** Document Type Definition
- **ESDS** Economic and Social Data Service
- **ESRC** Economic and Social Research Council
- **HTML** HyperText Markup Language
- **JISC** Joint Information Systems Committee
- **METS** Metadata Encoding & Transmission Standard
- **PDF** Portable Document Format
- **RTF** Rich Text Format
- **TEI** Text Encoding Initiative
- **TIFF** Tagged Image File Format
- **XML** eXtensible Markup Language
- **XSD** XML schema Definition
- **XSL** Extensible Stylesheet Language
- **XSLT** XSL Transformations

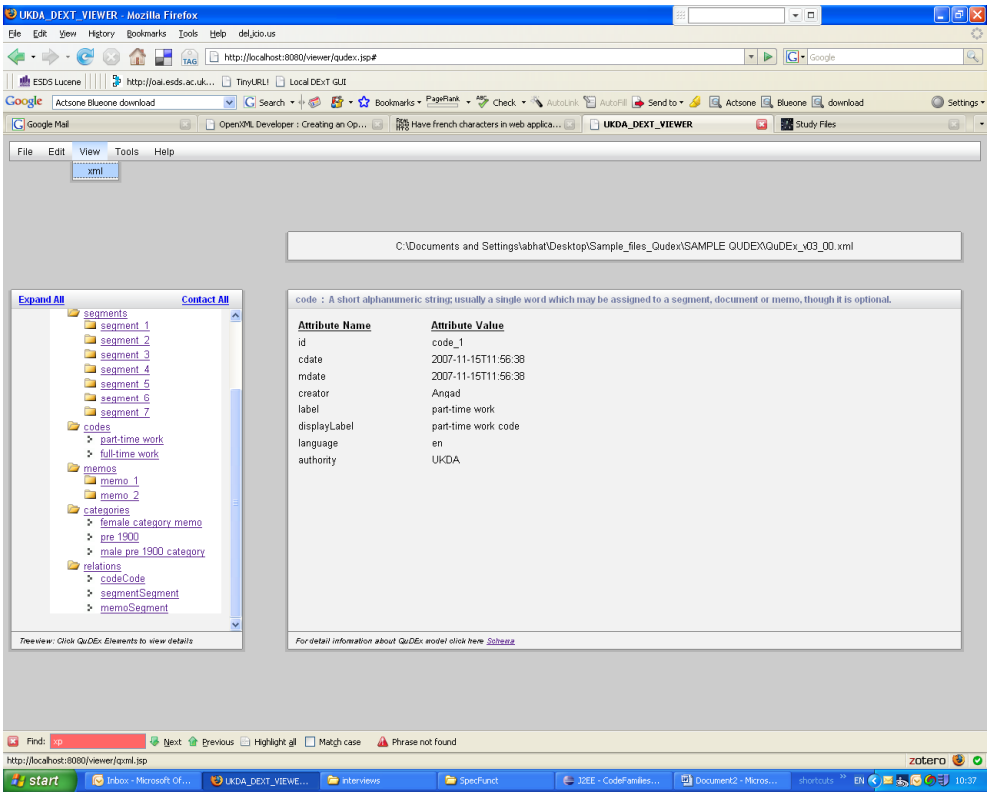
APPENDIX 7: Screen shots of QuDex Viewer and Transformation

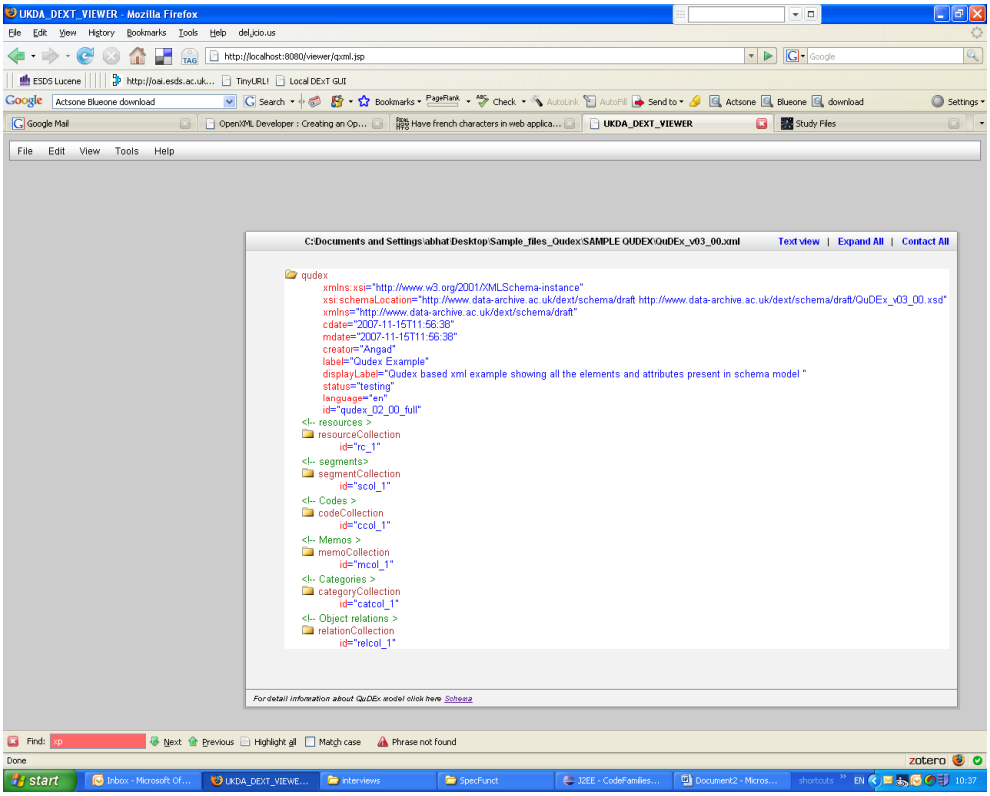


QuDEx Viewer




UKDA_DEXT_VIEWER - Mozilla Firefox
File Edit View Tools Help
C:\Documents and Settings\abhat\Desktop\Sample_files_Qudex\SAMPLE QUDEX\QuDEx_v03_00.xml
Expand All
Contact All
segments
segment_1
segment_2
segment_3
segment_4
segment_5
segment_6
segment_7
codes
part-time work
full-time work
memos
memo_1
memo_2
categories
female category memo
pre 1900
male pre 1900 category
relations
codeCode
segmentSegment
memoSegment
code : A short alphanumeric string; usually a single word which may be assigned to a segment, document or memo, though it is optional.
Attribute Name
Attribute Value
id code_1
cdate 2007-11-15T11:56:38
mdate 2007-11-15T11:56:38
creator Angad
label part-time work
displayLabel part-time work code
language en
authority UKDA
Treeview: Click QuDEx Elements to view details
For detail information about QuDEx model click here Schema


UKDA_DEXT_VIEWER - Mozilla Firefox
File Edit View Tools Help
xml
C:\Documents and Settings\abhat\Desktop\Sample_files_Qudex\SAMPLE QUDEX\QuDEx_v03_00.xml
Expand All
Contact All
segments
segment_1
segment_2
segment_3
segment_4
segment_5
segment_6
segment_7
codes
part-time work
full-time work
memos
memo_1
memo_2
categories
female category memo
pre 1900
male pre 1900 category
relations
codeCode
segmentSegment
memoSegment
code : A short alphanumeric string; usually a single word which may be assigned to a segment, document or memo, though it is optional.
Attribute Name
Attribute Value
id code_1
cdate 2007-11-15T11:56:38
mdate 2007-11-15T11:56:38
creator Angad
label part-time work
displayLabel part-time work code
language en
authority UKDA
Treeview: Click QuDEx Elements to view details
For detail information about QuDEx model click here Schema



Installation instructions for QuDex Viewer

In order to run QuDEx viewer application you need to have tomcat 5.5 or higher running on the system. The latest version of tomcat could be downloaded from this site http://tomcat.apache.org/download-60.cgi. Once the tomcat is installed you can test it by pointing your web browser to this location http://localhost:8080/.

Now, when the tomcat is up and running perform the following steps to run the application.

1. Stop the Tomcat server first.
   Start > Control Panel > Administrative Tools > Services > Apache Tomcat > Right click > stop.

2. Copy the viewer.war file in webapps dir.
   Tomcat installation directory ** > webapps

2. Start the Tomcat server to extract the war file.
   Start > Control Panel > Administrative Tools > Services > Apache Tomcat > Right click > start.

3. A new folder "viewer" should now be under webapps directory.
   Tomcat installation directory ** > webapps

4. Copy the following jar files
   a. jdom.jar
   b. ukda_dext_transformation.jar
   c. xml-apis.jar
   d. xerces-2.6.0.jar

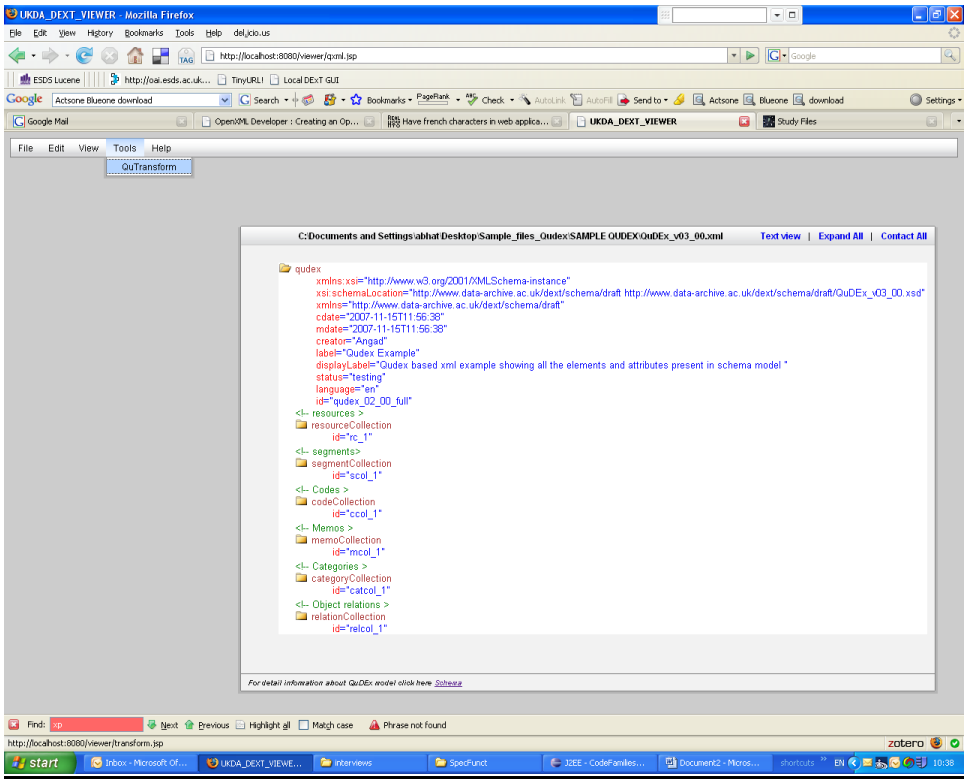
Tomcat installation directory ** > webapps > viewer > WEB-INF > lib

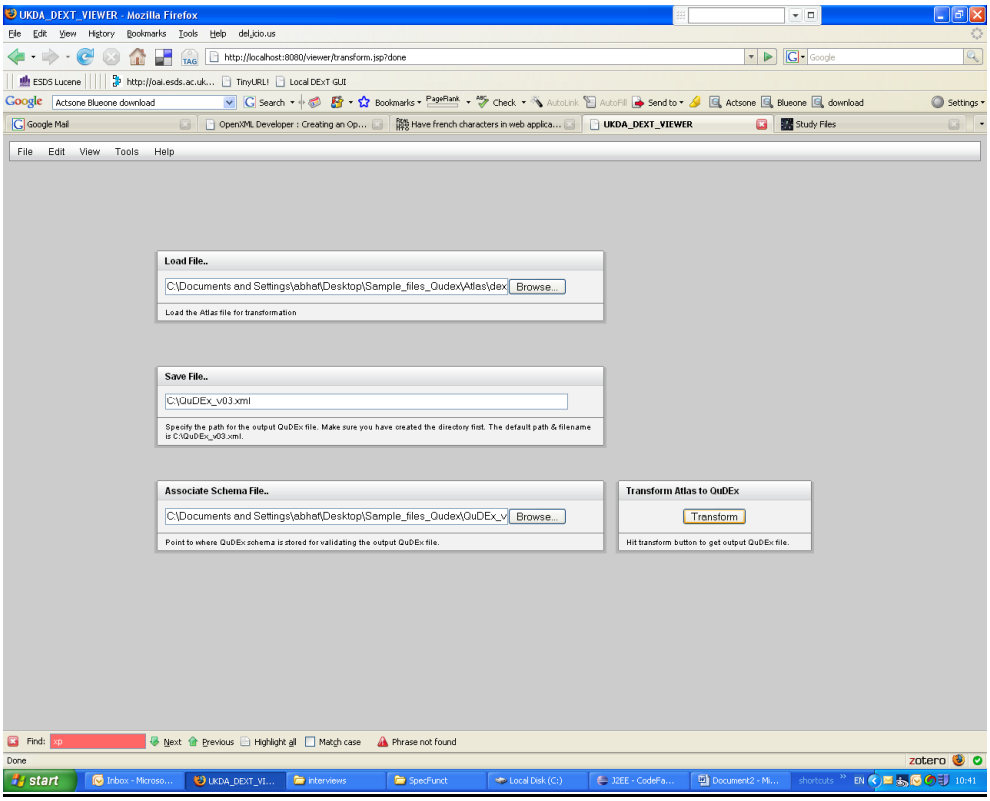
5. Restart the sever.
Start > Control Panel > Administrative Tools > Services > Apache Tomcat > Right click > stop.
Start > Control Panel > Administrative Tools > Services > Apache Tomcat > Right click > start.

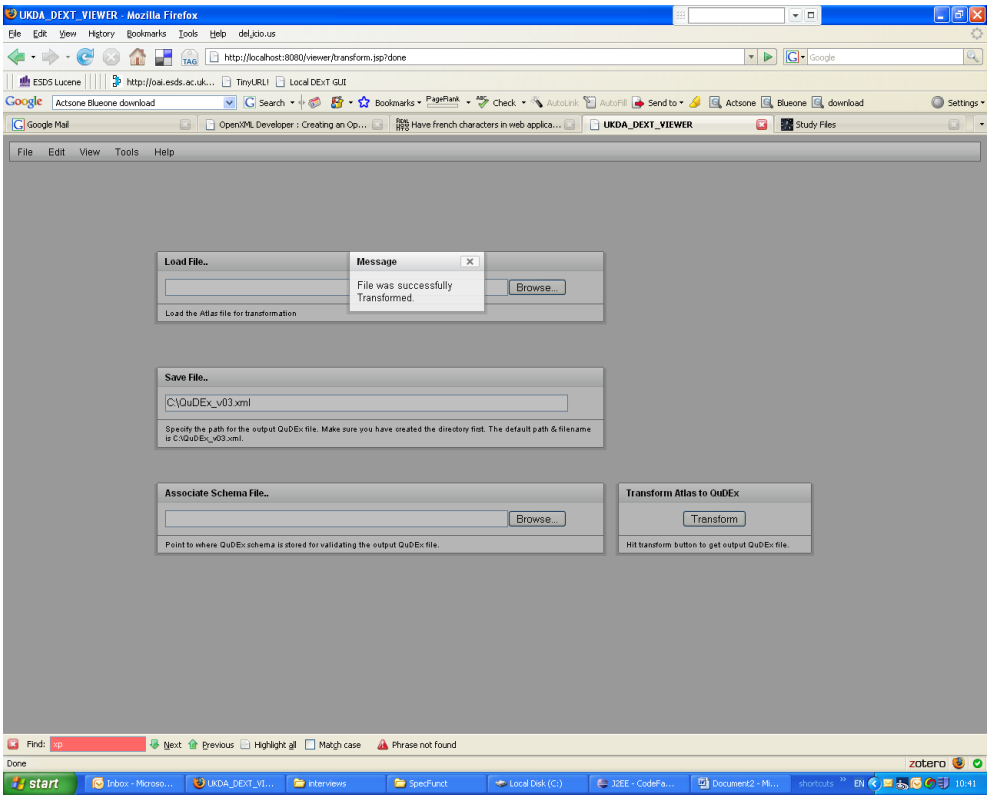
6. Run the Application by pointing the web browser to this address
http://localhost:8080/viewer/main.jsp

** By default it's installed in C:\Program Files\Apache Software Foundation\Tomcat

QuDEx Transformation




UKDA_DEXT_VIEWER - Mozilla Firefox
File Edit View Tools Help
Load File..
C:\Documents and Settings\abhat\Desktop\Sample_files_Qudex\Atlas\dex
Browse...
Load the Atlas file for transformation
Save File..
C:\QuDEx_v03.xml
Specify the path for the output QuDEx file. Make sure you have created the directory first. The default path & filename is C:\QuDEx_v03.xml.
Associate Schema File..
C:\Documents and Settings\abhat\Desktop\Sample_files_Qudex\QuDEx_v
Browse...
Point to where QuDEx schema is stored for validating the output QuDEx file.
Transform Atlas to QuDEx
Transform
Hit transform button to get output QuDEx file.


UKDA_DEXT_VIEWER - Mozilla Firefox
File Edit View Tools Help
Load File..
Message
File was successfully Transformed.
Browse...
Load the Atlas file for transformation
Save File..
C:\QuDEx_v03.xml
Specify the path for the output QuDEx file. Make sure you have created the directory first. The default path & filename is C:\QuDEx_v03.xml.
Associate Schema File..
Browse...
Point to where QuDEx schema is stored for validating the output QuDEx file.
Transform Atlas to QuDEx
Transform
Hit transform button to get output QuDEx file.