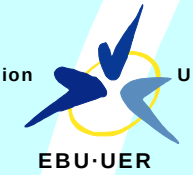
ETSI TS 102 822-6-3 V1.1.1 (2006-01)

Technical Specification

Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 6: Delivery of metadata over a bi-directional network; Sub-part 3: Phase 2 - Exchange of Personal Profile



Reference
DTS/JTC-TVA-PH1-21-06-03

Keywords
broadcasting, content, service, TV, video

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2006.
© European Broadcasting Union 2006.
All rights reserved.

DECT™, **PLUGTESTS**™ and **UMTS**™ are Trade Marks of ETSI registered for the benefit of its Members.
TIPHON™ and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.
3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights 5
Foreword 5
Introduction 6
1 Scope 7
2 References 8
3 Definitions and abbreviations 9
3.1 Definitions 9
3.2 Abbreviations 11
4 TV-Anytime User Profile Service Type 11
4.1 Service Type Definition 11
4.2 Discovery 12
4.3 Data Model 12
5 TV-Anytime User Profile Message Interface 12
5.1 Common Parts 12
5.1.1 Resources 12
5.1.2 <Select> element 13
5.1.3 Discovery Option Keywords 15
5.1.4 <Status> 15
5.1.5 Linking with ids 16
5.1.6 The timeStamp Attribute 17
5.2 Querying Data 17
5.2.1 <Query> 17
5.2.2 <QueryResponse> 18
5.2.3 Processing Rules for Query 19
5.3 Modifying Data 19
5.3.1 <Modify> 19
5.3.2 Modify Response 20
5.3.3 Processing Rules for Modify 20
6 Notes on Privacy and Security 21
6.1 Privacy Considerations 21
6.1.1 Introduction 21
6.1.2 Authorization of Watchers 21
6.1.3 Consent 21
6.1.4 Usage Directives 21
6.1.5 Options 21
6.2 Security Mechanisms 21
Annex A (normative): Schema and classification scheme 22
A.1 Schema 22
A.2 Status Classification Scheme 24
Annex B (informative): Sample instances 26
B.1 Sample SOAP Header 26
B.2 Example of Query 26
B.3 Example of QueryResponse 27
B.4 Example of Modify 27

B.5 Example of ModifyResponse 28
B.6 Sample of WDSL 28
Annex C (normative): Schema files 29
Annex D (informative): Bibliography 30
History 32

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

The present document is part 6, sub-part 3 of a multi-part deliverable covering Broadcast and On-line Services: Search, select and rightful use of content on personal storage systems ("*TV-Anytime*"), as identified below:

Part 1: "Benchmark Features";

Part 2: "System description";

Part 3: "Metadata";

Part 4: "Content referencing";

Part 5: "Rights Management and Protection (RMP)";

Part 6: "Delivery of metadata over a bi-directional network";

Sub-part 1: "Service and transport";

Sub-part 2: "Phase 1 - Service discovery";

Sub-part 3: "Phase 2 - Exchange of Personal Profile";

Part 7: "Bi-directional metadata delivery protection";

Part 8: "Phase 2 - Interchange Data Format";

Part 9: "Phase 2 - Remote Programming".

Introduction

"*TV-Anytime*" (TVA) is a full and synchronized set of specifications established by the *TV-Anytime* Forum. TVA features enable the search, selection, acquisition and rightful use of content on local and/or remote personal storage systems from both broadcast and online services.

TS 102 822-1 [1] and TS 102 822-2 [2] set the context and system architecture in which the standards for Metadata, Content referencing, Bi-directional metadata and Metadata protection are to be implemented in the *TV-Anytime* environment. TS 102 822-1 [1] provides benchmark business models against which the *TV-Anytime* system architecture is evaluated to ensure that the specification enable key business applications. TS 102 822-2 [2] presents the *TV-Anytime* System Architecture. These two documents are placed ahead of the others for their obvious introductory value. These first two documents are largely informative, while the remainder of the series is normative.

The features are supported and enabled by the specifications for Metadata (TS 102 822-3-1 [3], TS 102 822-3-2 [4], TS 102 822-3-3 [5] and TS 102 822-3-4 [6]), Content Referencing (TS 102 822-4 [7]), Rights Management (TS 102 822-5-1 [8] and TS 102 822-5-2 [9]), Bi-directional Metadata Delivery (TS 102 822-6-1 [10], TS 102 822-6-2 [11] and TS 102 822-6-3) and Protection (TS 102 822-7 [12]), Interchange Data Format (TS 102 822-8 [13]) and Remote Programming (TS 102 822-9 [14]).

The present document covers the exchange of personal metadata, such as user preferences, usage history, demographic and other information.

1 Scope

The present Phase 2 document extends the scope for data exchange between users and content/service providers. The exchange of such user profiles offers a number of possible benefits to both consumers and service providers. For example, a personalized web-service in which a service provider provides the response of user's request using his consumer profile data in a secure environment, can offer the consumer a personally relevant content.

TV-Anytime Phase 1 and Phase 2 offers an evolutionary range of features which describe PDR (Personal Digital Recorder) usage models that the *TV-Anytime* standards facilitate.

The features are supported and enabled by the specifications for Metadata (TS 102 822-3-1 [3], TS 102 822-3-2 [4], TS 102 822-3-3 [5] and TS 102 822-3-4 [6]), Content Referencing (TS 102 822-4 [7]), Rights Management (TS 102 822-5-1 [8] and TS 102 822-5-2 [9]), Bi-directional Metadata Delivery (TS 102 822-6-1 [10], TS 102 822-6-2 [11] and TS 102 822-6-3) and Protection (TS 102 822-7 [12]), Interchange Data Format (TS 102 822-8 [13]) and Remote Programming (TS 102 822-9 [14]).

These specifications enable search, select, acquire and rightful use of content on local and/or remote personal storage systems from both broadcast and online services.

Although *TV-Anytime* Phase 1 already provides a rich and varied set of metadata including user preferences and usage history to describe content and usage, the possibilities of exchanging this data were limited (TS 102 822-6-1 [10] and TS 102 822-6-2 [11]).

The exchange of personal data shall allow:

- Providers to receive data from a wide range of PDR devices from different users.
- Service providers to receive detailed and comprehensive user related data.
- Consumers to have confidence in the services (e.g. billing, tracking, monitoring).
- Consumers to search, select and acquire content of their direct interest such as programmes with additional audio description services for improved access, or local news, or specific travel data.
- Consumers to "carry" their profiles and other personal data.
- Content providers and creators to understand their audiences better.
- To develop a wider range of t-Commerce (particularly PPV) and other network service business models.
- To maximize the resources of a wide range of user appliances like a set-top-box or a PDA.

The present document describes mechanisms for requesting metadata from, and submitting user's personal profile data to, IP based web services. In the present document, these web services are termed "personalized services". The present document complements TS 102 822-6-1 [10], which specifies a client initiated means for requesting TV-Anytime metadata from IP-based metadata services and also provides detailed service scenarios.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

[1] ETSI TS 102 822-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 1: Benchmark Features".

[2] ETSI TS 102 822-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 2: System description".

[3] ETSI TS 102 822-3-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 1: Phase 1 -Metadata schemas".

[4] ETSI TS 102 822-3-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 2: System aspects in a uni-directional environment".

[5] ETSI TS 102 822-3-3: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 3: Phase 2 - Extended Metadata Schema".

[6] ETSI TS 102 822-3-4: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 4: Phase 2 - Interstitial metadata".

[7] ETSI TS 102 822-4: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 4: Content Referencing".

[8] ETSI TS 102 822-5-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 5: Rights management and protection information (RMPI); Sub-part 1: RMPI for broadcast application".

[9] ETSI TS 102 822-5-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 5: Rights management and protection information (RMPI); Sub-part 2: Binding of rights management and protection information".

[10] ETSI TS 102 822-6-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 6: Delivery of metadata over a bi-directional network; Sub-part 1: Service and transport".

[11] ETSI TS 102 822-6-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 6: Delivery of metadata over a bi-directional network; Sub-part 2: Phase 1 - Service discovery".

[12] ETSI TS 102 822-7: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 7: Phase 1 - Bi-directional metadata delivery protection".

[13] ETSI TS 102 822-8: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 8: Phase 2 - Interchange data format".

[14] ETSI TS 102 822-9: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 9: Phase 2 - Remote Programming".

[15] W3C Recommendations (version 20010502) "XML Schema".

NOTE: Available at:
http://www.w3.org/TR/2001/REC-xmlschema-0-20010502.
http://www.w3.org/TR/2001/REC-xmlschema-1-20010502
http://www.w3.org/TR/2001/REC-xmlschema-2-20010502.

[16] W3C Recommendation, 14 January 1999: "Namespaces in XML".

NOTE: Available at: http://www.w3.org/TR/REC-xml-names/.

[17] IETF RFC 2396: "Uniform Resource Identifiers (URI): Generic Syntax".

NOTE: Available at: http://www.ietf.org/rfc/rfc2396.txt.

[18] W3C Note, 8 May 2000: "Simple Object Access Protocol (SOAP) 1.1".

NOTE: Available at: http://www.w3.org/TR/2000/NOTE-SOAP-20000508/.

[19] OASIS Standard 200401: "Web Services Security: SOAP Message Security 1.0".

NOTE: Available at: http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf.

[20] "Liberty ID-WSF Data Services Template Specification", Version 1.1.

NOTE: Available at: https://www.projectliberty.org/specs/liberty-idwsf-dst-v1.1.pdf.

[21] "Liberty ID-WSF Discovery Service Specification", Version 1.2.

NOTE: Available at: https://www.projectliberty.org/specs/liberty-idwsf-disco-svc-v1.2.pdf.

[22] "Liberty ID-WSF SOAP Binding Specification", Version 1.2.

NOTE: Available at: https://www.projectliberty.org/specs/liberty-idwsf-soap-binding-v1.2.pdf.

[23] "Liberty Metadata Description and Discovery Specification", Version: 1.1.

NOTE: Available at: https://www.projectliberty.org/specs/liberty-metadata-v1.1.pdf.

[24] "Liberty Reverse HTTP Binding for SOAP Specification".

NOTE: Available at: https://www.projectliberty.org/specs/liberty-paos-v1.1.pdf.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

access control: act of mediating requested access to a resource based on privilege attributes of the requestor and control attributes of the requested resource

account: formal business agreement for providing regular transactions and services between a principal and service providers

assertion: piece of data produced by a SAML authority regarding an act of authentication performed on a principal, attribute information about the principal, or authorization permissions applying to the principal with respect to a specified resource

attribute provider: entity that provides attributes related to principals

NOTE: For example a personalized profile service.

circle of trust: federation of service providers and identity providers that have business relationships based on Liberty architecture, and operational agreements, with whom users can transact business in a secure and seamless environment

consumer profile: consumer profile includes user description (user preferences, usage viewing history), user information (e.g. biographic information), and user environment descriptions (e.g. user device information, network environment)

content provider: entity that acts as the agent for and is the prime exploiter of the content

credentials: data attesting to the truth of certain stated facts

discovery service: web service for locating attribute providers

entry point: what a discovery service allows one to discover i.e. a SOAP (RPC) address and function name

federation: association comprising any number of service providers and identity providers

identity: essence of an entity and often described by its characteristics

identity federation: associating, connecting, or binding multiple accounts for a given principal to various entities within a circle of trust

identity provider: entity that creates, maintains, and manages identity information for principals and provides principal authentication to other service providers within a circle of trust

identity personal profile: identity information regarding the principal, be it in private or business capacity

metadata: generally, data about content, such as the title, genre, and summary of a television programme

NOTE: In the context of *TV-Anytime*, metadata also includes personal profile and history data.

nonce: number that is used once and then discarded in a cryptographic protocol

personally identifiable information: any data that identifies or locates a particular person, consisting primarily of name, address, telephone number, e-mail address, bank accounts, or other unique identifiers such as a social security number

policy: logically defined, executable and testable set of rules of behaviour

principal: entity that can acquire a federated identity, that is capable of making decisions, and to which authenticated actions are done on its behalf

NOTE: Examples of principals include an individual user, a group of individuals, as well as a corporation or other legal entities.

profile: data comprising the broad set of attributes that may be maintained for an identity, over and beyond its identifiers and the data required to authenticate under that identity

NOTE: Some of those attributes (e.g. address, preferences, card numbers) are provided by the principal.

pay per view: content for which the consumer had had to pay a one off fee

personal profile: data that represents the interests and preferences of the consumer

programme: editorially coherent piece of content

NOTE: Typically, a programme is acquired by the PDR as a whole.

provider: entity that delivers content or services to the PDR

return path: part of a bi-directional distribution system over which data flows from the consumer to the service provider

service provider: aggregator and supplier of content which may include gateway and management roles

NOTE: In the present document, a service provider provides services or goods to a principal.

resource: either data related to some identity or identities, or a service acting on behalf of some identity or group of identities

NOTE: An example of a resource is a calendar containing appointments for a particular identity.

service: collection of entry points designed to offer some service or to provide information

service instance: a running web service (physical instantiation of a particular type of service) at a distinct protocol endpoint

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

EPG	Electronic Programme Guide
HTTP	Hyper Text Transport Protocol
ID-WSF	IDentity - Web Service Framework
OASIS	Organization for the Advancement of Structured Information Standards
PDA	Personal Digital Assistance
PDR	Personal Digital Recorder
PPV	Pay Per View
SAML	Security Assertion Markup Language

NOTE: XML standard for exchanging authentication and authorization data between security systems.

SOAP	Simple Object Access Protocol
URI	Uniform Resource Identifier
URN	Uniform Resource Name
W3C	World Wide Web Consortium
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XPath	W3C XML Path language

4 TV-Anytime User Profile Service Type

4.1 Service Type Definition

The *TV-Anytime* user profile service is a web service that acts upon some resource to either retrieve and update information (such as preferences, demography, and location) about a user, or perform some action for the benefit of some user.

The types used in the *TV-Anytime* user profile service are defined in the `"urn:tva:metadata:profile:2005"` target namespace [16].

This allows schema aware tools to validate the various messages. The types defined in the TS 102 822-3-1 [3] and TS 102 822-3-3 [5] may be referenced in the transport namespace (using XML Schema's import mechanisms [15]). The schema fragments in the following clauses are all defined within this namespace. The ServiceType element is used to identify the *TV-Anytime* user profile service. This URI needs be constant across all implementations of the *TV-Anytime* user profile service to enable interoperability. Therefore, it is recommended that this URI be the same as the targetNamespace URI of the abstract WSDL description for the service.

A service instance is the physical instantiation of a particular type of the *TV-Anytime* user profile service. Each service instance is hosted by some provider, which is identified by a URI (RFC 2396 [17]). An example of a service instance is a SOAP-over-HTTP end-point offering a personalized EPG service. A service instance exposes a protocol interface to a set of resources. A resource in this specification is either data related to some user (or the attributes thereof) or a service acting for the benefit of some user. When a client sends a request message to a service instance, it includes the resource identifier (i.e. a URI) for the resource it wishes the service instance to act upon. A resource commonly has access to control policies associated with it. These access control policies are typically under the purview of the entity or entities associated with the resource (in common language, the entity or entities could be said to "own" the resource). The access control policies on a resource must be enforced by the service instance.

4.2 Discovery

A service provider can use the discovery service to ascertain the location of a specific identity service for a user. The discovery service enables various entities to dynamically and securely discover a user's identity services, and it responds, on a permission-basis, with a service description of the desired identity service. This way, a service provider can gain access to portions of the user's identity information that may be in the possession of multiple providers and offer a user profile based service to a user.

The discovery service itself is not defined in the present document. Examples of discovery services can be found in the Liberty ID-WSF Discovery Service specification [21] and in the Liberty Metadata Description and Discovery Specification [23].

4.3 Data Model

The data model of the *TV-Anytime* user profile is defined in the *TV-Anytime* metadata specification TS 102 822-3-3 [5]. The XML schema defined thereby specifies the data the service can host. The data model defines the data and the data structure. Typically this structure is hierarchical and has the root node `TVAMain`. The part of the *TV-Anytime* metadata mainly relevant to the *TV-Anytime* user profile exchange service is the `ExtendedUserDescription.`

5 TV-Anytime User Profile Message Interface

This clause defines two protocols, one for querying data and another for modifying data. These protocols are based on those specified in Liberty ID-WSF Data Services Template Specification [20].

These protocols both rely on a request/response message-exchange pattern. The messages specified in the present document for those protocols are carried in the SOAP body [18]. More information on the general aspects of SOAP can be found in TS 102 822-6-1 [10] and Liberty ID-WSF SOAP Binding Specification [22]. The messages for querying and modifying data have common attributes and elements. These common parts are discussed prior to specifying the actual messages.

5.1 Common Parts

5.1.1 Resources

Both protocols, for querying and modifying data, have a defined hierarchy for accessing data.

In the first level the desired resources are selected. For example, a resource might be the personal profile of a certain person. Multiple resources can be accessed in a single request, but querying and modifying cannot be mixed in one request message.

For each resource there is one `<Query>` or `<Modify>` element in the request message. Inside this element there is another element identifying the resource. This identifying element is either the `<ResourceID>` element or the `<EncryptedResourceID>` element. The `ResourceIDGroup` declaration is shown below.

```
<complexType name="ResourceIDType">
    <simpleContent>
        <extension base="anyURI">
            <attribute name="id" type="tva:TVAIDType"/>
        </extension>
    </simpleContent>
</complexType>

<complexType name="EncryptedResourceIDType">
    <sequence>
        <element ref="xenc:EncryptedData"/>
        <element ref="xenc:EncryptedKey"/>
    </sequence>
</complexType>

<group name="ResourceIDGroup">
        <choice maxOccurs="unbounded">
            <element name="ResourceID" type="tvap:ResourceIDType"/>
            <element name="EncryptedResourceID" type="tvap:EncryptedResourceIDType"/>
        </choice>
</group>
```

Name	Definition
ResourceIDType	A complex type to define a resource identifier
Id	An attribute of TVAIDType type
EncryptedResourceIDType	A complex type for the transport of obfuscated resource identifiers
ResourceIDGroup	Defines a group of ResourceIDs and/or EncryptedResourceIDs
ResourceID	The identifier of a particular resource
EncryptedResourceID	Contains a ResourceID that has been encrypted using encryption, and an encrypted key that was used to encrypt the ResourceID. See clause 5.1.2. and Liberty ID-WSF Discovery Service Specification [22] for more explanations

When the <ResourceID> element would have the value "urn:liberty:isf:implied-resource" [21], the element may be left out of the containing <Query> or <Modify> element. In all other cases either the <ResourceID> element or the <EncryptedResourceID> element must be present. More examples of when the value urn:liberty:isf:implied-resource can be used can be found in Liberty Reverse HTTP Binding for SOAP Specification [24].

The schema also defines an element EncryptedResourceID for the transport of obfuscated resource identifiers. The EncryptedResourceID contains a ResourceID that has been encrypted using encryption, and an encrypted key that was used to encrypt the ResourceID. Use of EncryptedResourceID is often necessary for privacy reasons. If a non-predictable nonce is used for the EncryptedKey, each discovery service client will get a different identifier. This will prevent discovery service clients from colluding with other discovery service clients about the identity of the Principal on the basis of the ResourceID for one of the Principal's services.

NOTE: Strictly speaking, the key does not need to be a nonce, rather just unique for each client.

5.1.2 <Select> element

The second level of resource selection is inside the <Query> and <Modify> elements. The request message must describe in more detail what it wants to access inside the specified resource. This is specified in <Select> elements. As an example, when the resource is a personal profile, the <Select> can point to UserPreferences. In the case of a <Query>, this means that the whole UserPreferences is requested, or for a <Modify>, the whole UserPreferences is being modified. When only a part of UserPreferences is being queried or modified, the <Select> element must point only to that part, or the parts not to be modified must be rewritten using their existing values, when whole UserPreferences is given. Different parts of the resource can be accessed using the same <Query> or <Modify> element as both of those elements can contain multiple <Select> elements in their own substructure.

The <Select> points to some place(s) in the *TV-Anytime*'s user description metadata and it is recommended that a string containing an XPath expression is used for <Select> element. It is not always necessary to support full XPath. Each *TV-Anytime* user profile exchange service should limit the required set of XPath expressions when full XPath is not required. The predefined set of abbreviated Select Path expressions, to be used in place of full XPath expressions, is listed in table 1. When a service supports full XPath even if it is not required, the service may register the urn:liberty:dst:fullXPath discovery option keyword. If the required set of XPath expressions does not include the path to each element, a service may still support all paths without supporting full XPath. In that case the service may register the urn:liberty:dst:allPaths discovery option keyword.

Table 1: Abbreviation for XPath Expressions

Abbreviated Selected Field	Full XPath Expression
tva:profile:UserSearchPreferences	/TVAMain/ExtendedUserDescription/UserPreferences/FilteringAndSearchPreferences
tva:profile:UserBrowsingPreferences	/TVAMain/ExtendedUserDescriptions/UserPreferences/BrowsingPreferences
tva:profile:UserActionHistory	/TVAMain/ExtendedUserDescriptions/UsageHistory/UserActionHistory
tva:profile:UserName	/TVAMain/ExtendedUserDescription/UserInformationTable/BioGraphicInformation/Name
tva:profile:UserAge	/TVAMain/ExtendedUserDescription/UserInformationTable/BioGraphicInformation/Age
tva:profile:UserGender	/TVAMain/ExtendedUserDescription/UserInformationTable/BioGraphicInformation/Gender
tva:profile:UserLanguage	/TVAMain/ExtendedUserDescription/UserInformationTable/BioGraphicInformation/Language
tva:profile:UserLocation	/TVAMain/ExtendedUserDescription/UserInformationTable/UsageEnvironment/NaturalEnvironment/Location

Successful queries always have the TVAMain element returned, and it must contain the data defined by the parameters inside the Select element.

The schema for SelectType is as follows.

```
<simpleType name="SelectTypeType">
    <restriction base="string">
        <enumeration value="xpath"/>
        <enumeration value="abbreviated"/>
    </restriction>
</simpleType>

<complexType name="SelectType">
    <simpleContent>
        <extension base="string">
            <attribute name="type" type="tvap:SelectTypeType" use="optional"
            default="abbreviated"/>
        </extension>
    </simpleContent>
</complexType>
```

The semantics is as follows.

Name	Definition
SelectTypeType	Simple type for the type of query format. The following are the possible values: - *xpath* – the query format is XPath - *abbreviated* -- the query format is in the Abbreviated format
SelectType	Complex type defining the "select" element
type	specifies the type of query format; either Xpath or the abbreviated format

5.1.3 Discovery Option Keywords

The following discovery option keywords are used to indicate services options of the *TV-Anytime* user profile services.

Table 2: Description of Discovery Option Keywords

Discovery Option Keywords	Definition
`Urn:liberty:dst:allPaths`	Optional. This indicates that the service provider supports all (required) paths without supporting full XPath.
`Urn:liberty:dst:fullXPath`	Optional. This indicates that the service provider supports full XPath.
`urn:liberty:dst:multipleResources`	Optional. This indicates that the service provider supports accessing multiple resources.
`urn:liberty:dst:changeHistorySupported`	Optional. This indicates that the service provider processes the changedSince attribute.
`Urn:liberty:dst:noModify`	Optional. This indicates that the service provider does not support modifications at all.
`urn:liberty:dst:multipleQueryItems`	Optional. This indicates that the service provider supports multiple <QueryItem> elements inside a <Query>.
`urn:liberty:dst:multipleModification`	Optional. This indicates that the service provider supports multiple <Modification> element inside a <Modify>.

5.1.4 <Status>

A response message contains one or more `<Status>` elements to indicate whether or not the processing of the request succeeded. A `<Status>` element has a code attribute, which contains the return status as a string. Table 3 defines the status codes to be used as values for the code attribute.

Table 3: Description of Values for Status Element

Status	Definition
`ActionNotAuthorized`	Indicates that the requested action is not authorized.
`ActionNotSupported`	Indicates that the requested action is not supported by this service.
`AllReturned`	Indicates that when an attribute provider may not necessarily be giving only the newest information since the time specified in the `changedSince` parameter. This is not considered a failure. It is also used to show that all the data currently allowed to be returned (i.e. except for those revoked, etc.).
`ChangedSinceReturnsAll`	Indicates that when an attribute provider does not support the `changedSince` parameter and returns all the data addressed by `Select` regardless of the latest modification times. Some or all data returned might actually be older than the time specified in the `changedSince` parameter.
`DataTooLong`	Indicates that the size of the requested data exceeds the preset threshold of the service provider. This is only for modifications.
`ExistsAlready`	Indicates that the request is trying to modify to a pre-existing value.
`Failed`	Indicates that the request has failed.
`InvalidData`	Indicates that the request contains invalid data.
`InvalidResourceID`	Indicates that the `ResourceID` of the request is invalid.
`InvalidSelect`	Indicates that the `Select` element is invalid in the request.
`MissingNewDataElement`	Indicates that the request specifies no new data to be updated.
`MissingResourceIDElement`	Indicates that the request fails to specify the `ResourceID`.
`MissingSelect`	Indicates that the request fails to specify the `Select` element.
`NoMoreElements`	Indicates that the returned data ends prematurely for lack of elements.
`NoMultipleAllowed`	Indicates that multiple requests are not allowed.
`NoMultipleResources`	Indicates that the use of multiple resources is not allowed.
`OK`	Indicates that the action has been successfully acknowledged.
`TimeOut`	Indicates that the request has failed because of timeout.
`UnexpectedError`	Indicates that an unexpected error has occurred.

The code attribute of the top level within the <Status> element must contain either the value OK or Failed. The <Status> element may contain <StatusDescription> element supplying more detailed return status information. The values in table 3 are used to indicate more detailed return status. If the request fails for some reason, the requestIDRef attribute of the <StatusDescription> element should contain the value of the itemID attribute of the offending element in the request message.

The following is the syntax for Status and StatusDescription.

```
<complexType name="StatusDescriptionType">
    <simpleContent>
        <extension base="mpeg7:TextualType">
            <attribute name="href" type="mpeg7:termReferenceType" use="optional"/>
        </extension>
    </simpleContent>
</complexType>

<complexType name="StatusType">
    <sequence>
        <element name="StatusDescription" type="tvap:StatusDescriptionType" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
    <attribute name="code" use="required">
        <simpleType>
            <restriction base="string">
                <enumeration value="OK"/>
                <enumeration value="Failed"/>
            </restriction>
        </simpleType>
    </attribute>
    <attribute name="requestIDRef" type="tva:TVAIDRefType" use="optional"/>
</complexType>
```

The following is the semantics for Status and StatusDescription.

Name	Definition
StatusType	Complex type that specifies the status codes to be used as values for the code attribute of Status.
code	Attribute that indicates the returned status value. It must contain either the value OK or Failed.
requestIDRef	Attribute that contains the value of the itemID attribute of the element in the request message that this status is about.
StatusDescriptionType	Complex type that specifies the status codes to be used as values for StatusDescription.
href	An optional attribute that specifies the URN used to point to the code within the status code classification scheme.

5.1.5 Linking with ids

Different types of id attributes are used to link queries and responses together. Response messages are correlated with requests using messageId and inResponseToMessageId attributes that are present in the SOAP Header [18]. Services must include messageId and inResponseToMessageId attributes in all request and response messages defined here. Use of these must follow the processing rules specified in Liberty ID-WSF Data Service Template Specification [20]. Inside messages, itemID and itemIDRef attributes are used for linking information inside response messages to the details of request messages.

NOTE: Response messages do not contain the <ResourceID> or the <EncryptedResourceID> element, so they cannot be used for this. See the definitions and the processing rules of <Query> and <Modify> elements for more detailed information.

5.1.6 The timeStamp Attribute

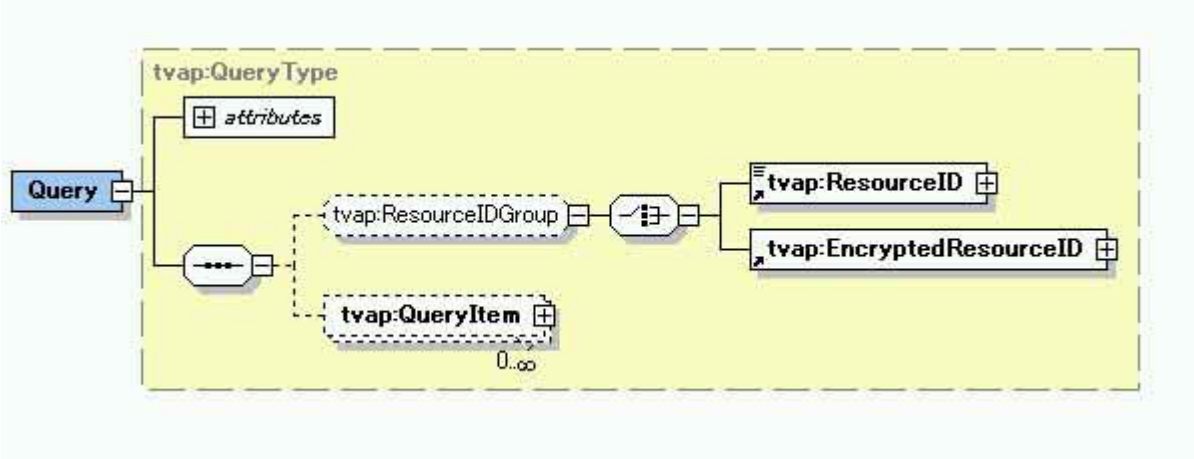
A response message can also have a time stamp. This time stamp is provided so that the requesting party can later check whether there have been any changes since a response was received, or make modifications, which will only succeed if there have been no other modifications made after the time stamp was received.

5.2 Querying Data

Two different kinds of queries are supported, one for retrieving current data, and another for requesting only changed data. These two different kinds of queries can be present together in the same message. The response can contain the data with or without the common technical attributes, depending on the request. Some common attributes are always returned for some elements.

5.2.1 <Query>

The schema for the Query operation is based on the Query operation specified in Liberty ID-WSF Data Service Template Specification [20].

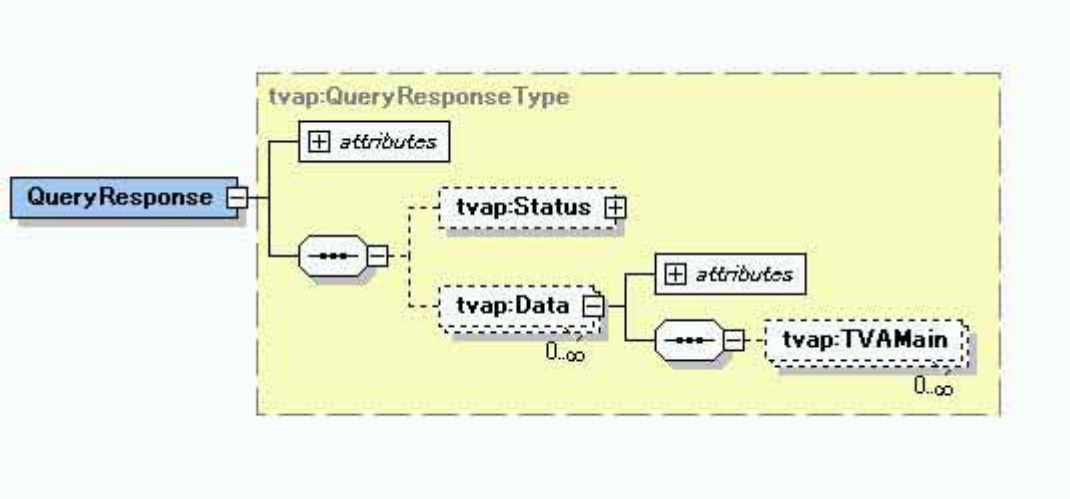



```
<complexType name="QueryType">
    <sequence>
        <group name="ResourceIDGroup" type="tvap:ResourceIDGroup" minOccurs="0"/>
        <element name="QueryItem" minOccurs="0" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="Select" type="tvap:SelectType" minOccurs="0"/>
                </sequence>
                <attribute name="itemID" type="tva:TVAIDType"/>
                <attribute name="changedSince" type="dateTime"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="queryID" type="tva: TVAIDType"/>
</complexType>
<element name="Query" type="tvap:QueryType"/>
```

Name	Definition
QueryType	a type that defines a query for retrieving current data
ResourceIDGroup	Contains a groups of ResourceIDs and/or EncryptedResourceIDs
QueryItem	specifies what data the requester wants from the resource
Select	describes in detail what the request wants to access inside the specified resource
itemID	a unique identifier used to match response items to query items within the response
changedSince	used when the requester wants to get only the data which has changed since the time specified by this attribute
queryID	a unique identifier to match queries with responses

5.2.2 <QueryResponse>

The schema for QueryResponse is as follows.




```
<complexType name="QueryResponseType">
    <sequence>
        <element name="Status" type="tvap:StatusType" minOccurs="1"/>
        <element name="Data" minOccurs="0" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="TVAMain" type="tva:TVAMainType" minOccurs="0"/>
                </sequence>
                <attribute name="itemIDRef" type="tva:TVAIDRefType"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="queryIDRef" type="tva:TVAIDRefType"/>
    <attribute name="timeStamp" type="dateTime"/>
</complexType>
<element name="QueryResponse" type="tvap:QueryResponseType"/>
```

Name	Definition
QueryResponseType	A complex type to contain the response to the query
Status	indicates whether or not the processing of the request succeeded. See clause 5.1.5 for explanations and values
Data	Contains *TV-Anytime* profile data requested by one <QueryItem> element
itemIDRef	used to link elements in a response to the corresponding elements in the request
queryIDRef	used to link elements in a response to the corresponding elements in the request
timestamp	indicates the time, when the request was handled. To be used later, when querying for changes since that time

The instance document that is returned must be XML Schema [15] valid with respect to the appropriate metadata schema (TS 102 822-3-3 [5]). Furthermore, each instance document must contain the appropriate `TVAIDType` to allow complete dereferencing of all `TVAIDRefType` nodes within the instance document. When successful, the Data element contains the information specified by the `Select` element.

5.2.3 Processing Rules for Query

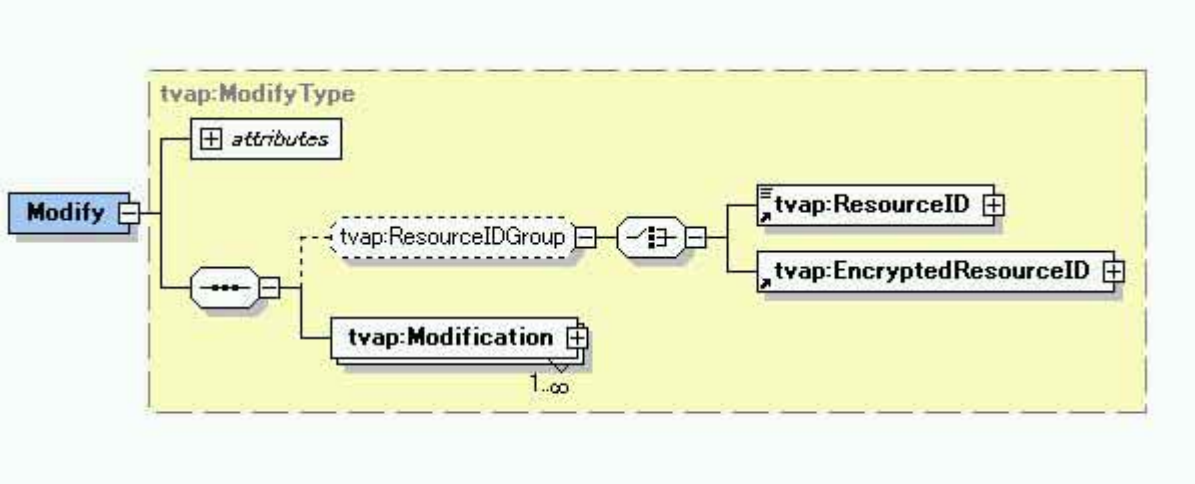
- A request message can contain multiple `<Query>` elements. The rules for how to implement multiple queries and responses are given in the Liberty ID-WSF Data Service Template Specification [20].

5.3 Modifying Data

The data stored by a data service can be given initial values, existing values can be replaced with new values and the data can also be removed. Usually the principal can make these modifications directly at the data service using the provided user interface, but these modifications may also be made by other service providers. The `<Modify>` element supports all these operations for service providers which want to modify the data stored in data services.

5.3.1 <Modify>

The schema for Modify is based on the Modify operation specified in Liberty ID-WSF Data Service Template Specification [20].

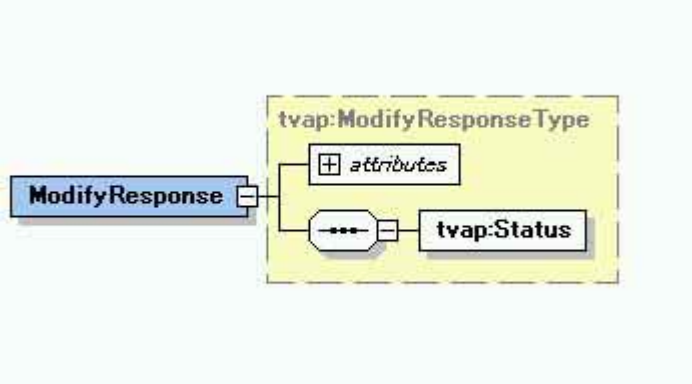



```
<complexType name="ModifyType">
    <sequence>
        <group name="ResourceIDGroup" type="tvap:ResourceIDGroup" minOccurs="0"/>
        <element name="Modification" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="Select" type="tvap:SelectType" minOccurs="0"/>
                    <element name="NewData" minOccurs="0">
                        <complexType>
                            <sequence>
                                <element name="TVAMain" type="TVAMainType" minOccurs="0"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
                <attribute name="itemID" type="tvap:IDType"/>
                <attribute name="notChangedSince" type="dateTime"/>
                <attribute name="overrideAllowed" type="boolean" default="0"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="modifyID" type="tvap:IDType"/>
</complexType>
<element name="Modify" type="tvap:ModifyType"/>
```

Name	Definition
ModifyType	modify the data stored in data services
ResourceIDGroup	contains a group of ResourceIDs and/or EncryptedResourceIDs
Modification	Specifies which data elements of the specified resource should be modified and how
itemID	used to separate elements in a request from each other
Select	Specifies the data this modification should affect
NewData	Specifies what data the requester wants from the resource
notChangedSince	a modification is not allowed, if the data has changed since the time specified by this attribute. Used to avoid concurrency problems
overrideAllowed	used to protect current values, when new data is added
modifyID	used to separate elements in a request from each other

5.3.2 Modify Response

The <ModifyResponse> element contains the <Status> element, which describes whether or not the requested modification succeeded. There is also a possible time stamp attribute, which provides a time value that can be used later to check whether there have been any changes since this modification, and an itemIDRef attribute to map the <ModifyResponse> elements to the <Modify> elements in the request.




```
<complexType name="ModifyResponseType">
    <sequence>
        <element name="Status" type="tvap:StatusType"/>
    </sequence>
    <attribute name="modifyIDRef" type="tvap:IDReferenceType"/>
    <attribute name="timeStamp" type="dateTime"/>
</complexType>
<element name="ModifyResponse" type="tvap:ModifyResponseType"/>
```

Name	Definition
ModifyResponseType	a complex type to contain the response to the modify
Status	indicates whether or not the processing of the request succeeded. See clause 5.1.5. for the explanation and the values
modifyIDRef	used to link elements in a response to the corresponding elements in the request
timestamp	indicates the time, when the request was handled. To be used later, when querying for changes since that time.

5.3.3 Processing Rules for Modify

A request message can contain multiple <Modify> elements. The rules for how to implement multiple modify requests and responses are given in Liberty ID-WSF Data Service Template Specification [20].

6 Notes on Privacy and Security

6.1 Privacy Considerations

6.1.1 Introduction

Information concerning demography, personal preferences, and viewing history, by its very nature, is fairly sensitive. Even a very simple indication of personal taste allows a watcher to have an idea of the user's personal traits. When detailed status and location information are added to the personal profile, control of such personal information becomes even more of a concern.

6.1.2 Authorization of Watchers

At a minimum a user must be able to specify who is able to request his personal profile for him/her. The consequence of this to the implementer is that incoming profile requests must be checked to ensure that their requesting Principal has been authorized by the target Principal to receive his/her personal profile information. While the implementation and maintenance of any authorization lists is specific to the individual protocols, the following supplies some mechanisms to indicate the requestor of the presence data and how the data will be used.

6.1.3 Consent

The Liberty ID-WSF SOAP Binding Specification [22] defines a `<Consent>` header block that allows profile exchange service clients to indicate to the profile exchange service provider that they have obtained consent to receive the presence data for a Principal.

6.1.4 Usage Directives

The Liberty ID-WSF SOAP Binding Specification [22] also provides a `<UsageDirective>` header block that allows a principal to specify limits on how his/her presence data may be used. The semantics for this block have not yet been fully specified, but one possible usage would involve a URL being specified in the UsageDirective block in the presence response, and this URL would reference a policy that describes a set of obligations that the requesting presence service client must fulfil or be in breach.

6.1.5 Options

As has been discussed, it is important that the user profile service provider be capable of specifying what information contained in user metadata can be exchanged and, in particular, whether the user ID is to be shared. This is handled by defining options in the offered resources that are publicized during the exchange of SOAP messages, as specified in TS 102 822-3-3 [5]. In this sense, an "options" element can be taken as expressing the "options" available for the resource offering, which provides hints to a potential requester whether certain data or operations may be available with a particular resource offering: e.g., an option may be provided stating whether information concerning user ID is available.

6.2 Security Mechanisms

The security mechanisms described in TS 102 822-7 [12] and Web Services Security:SOAP Message Security [19] should be supported.

Annex A (normative): Schema and classification scheme

A.1 Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="urn:tva:profile:2005" xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
xmlns:mpeg7="urn:tva:mpeg7:2005" xmlns:tvap="urn:tva:profile:2005" xmlns:tva="urn:tva:metadata:2005"
xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified" attributeFormDefault="unqualified">
    <annotation>
        <documentation xml:lang="en">This schema consists of datatypes that are normatively
        defined in TS 102 822-6-3 v1.1.2</documentation>
    </annotation>
    <import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="xml.xsd"/>
    <import namespace="urn:tva:mpeg7:2005" schemaLocation="tva_mpeg7.xsd"/>
    <import namespace="urn:tva:metadata:2005" schemaLocation="tva_metadata_3-1_v131.xsd"/>
    <import namespace="http://www.w3.org/2001/04/xmlenc#" schemaLocation="xenc-
    schema.xsd"/>
    <annotation>
        <documentation xml:lang="en">Base Types</documentation>
    </annotation>
    <complexType name="ResourceIDType">
        <simpleContent>
            <extension base="anyURI">
                <attribute name="id" type="tva:TVAIDType"/>
            </extension>
        </simpleContent>
    </complexType>
    <complexType name="EncryptedResourceIDType">
        <sequence>
            <element ref="xenc:EncryptedData"/>
            <element ref="xenc:EncryptedKey"/>
        </sequence>
    </complexType>
    <group name="ResourceIDGroup">
        <choice>
            <element name="ResourceID" type="tvap:ResourceIDType"/>
            <element name="EncryptedResourceID" type="tvap:EncryptedResourceIDType"/>
        </choice>
    </group>
    <simpleType name="SelectTypeType">
        <restriction base="string">
            <enumeration value="xpath"/>
            <enumeration value="abbreviated"/>
        </restriction>
    </simpleType>
    <complexType name="SelectType">
        <simpleContent>
            <extension base="string">
                <attribute name="type" type="tvap:SelectTypeType" use="optional"
                default="abbreviated"/>
            </extension>
        </simpleContent>
    </complexType>
    <complexType name="StatusDescriptionType">
        <complexContent>
            <extension base="mpeg7:TextualType">
                <attribute name="href" type="mpeg7:termReferenceType" use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="StatusType">
        <sequence>
            <element name="StatusDescription" type="tvap:StatusDescriptionType" minOccurs="0"
        maxOccurs="unbounded"/>
        </sequence>
        <attribute name="code" use="required">
            <simpleType>
                <restriction base="string">
                    <enumeration value="OK"/>
```

```
                <enumeration value="Failed"/>
            </restriction>
        </simpleType>
    </attribute>
    <attribute name="requestIDRef" type="tva:TVAIDRefType" use="optional"/>
</complexType>
<annotation>
    <documentation xml:lang="en">Clause 5.2.1 Query</documentation>
</annotation>
<complexType name="QueryType">
    <sequence>
        <group ref="tvap:ResourceIDGroup"/>
        <element name="QueryItem" minOccurs="0" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="Select" type="tvap:SelectType" minOccurs="0"/>
                </sequence>
                <attribute name="itemID" type="tva:TVAIDType"/>
                <attribute name="changedSince" type="dateTime"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="queryID" type="tva:TVAIDType"/>
</complexType>
<element name="Query" type="tvap:QueryType"/>
<annotation>
    <documentation xml:lang="en">Clause 5.2.2 Query Response</documentation>
</annotation>
<complexType name="QueryResponseType">
    <sequence>
        <element name="Status" type="tvap:StatusType" maxOccurs="unbounded"/>
        <element name="Data" minOccurs="0" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="TVAMain" type="tva:TVAMainType" minOccurs="0"/>
                </sequence>
                <attribute name="itemIDRef" type="tva:TVAIDRefType"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="queryIDRef" type="tva:TVAIDRefType"/>
    <attribute name="timeStamp" type="dateTime"/>
</complexType>
<element name="QueryResponse" type="tvap:QueryResponseType"/>
<annotation>
    <documentation xml:lang="en">Clause 5.3.1 Modify</documentation>
</annotation>
<complexType name="ModifyType">
    <sequence>
        <group ref="tvap:ResourceIDGroup"/>
        <element name="Modification" maxOccurs="unbounded">
            <complexType>
                <sequence>
                    <element name="Select" type="tvap:SelectType" minOccurs="0"/>
                    <element name="NewData" minOccurs="0">
                        <complexType>
                            <sequence>
                                <element name="TVAMain" type="tva:TVAMainType" minOccurs="0"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
                <attribute name="itemID" type="tva:TVAIDType"/>
                <attribute name="notChangedSince" type="dateTime"/>
                <attribute name="overrideAllowed" type="boolean" default="0"/>
            </complexType>
        </element>
    </sequence>
    <attribute name="modifyID" type="tva:TVAIDType"/>
</complexType>
<element name="Modify" type="tvap:ModifyType"/>
<annotation>
    <documentation xml:lang="en">Clause 5.3.2 Modify Response</documentation>
</annotation>
<complexType name="ModifyResponseType">
    <sequence>
        <element name="Status" type="tvap:StatusType" maxOccurs="unbounded"/>
    </sequence>
```

```
        <attribute name="modifyIDRef" type="tva:TVAIDRefType"/>
        <attribute name="timeStamp" type="dateTime"/>
    </complexType>
    <element name="ModifyResponse" type="tvap:ModifyResponseType"/>
</schema>
```

A.2 Status Classification Scheme

```
<ClassificationScheme uri="urn:tva:profile:cs:StatusCS:2005">
    <!-- ########################################################################## -->
    <!-- Status Code returned to requests                                        -->
    <!--Definition: -->
    <!-- ########################################################################## -->
    <Term termID="1">
        <Name xml:lang="en">ActionNotAuthorized</Name>
        <Definition xml:lang="en">
    Indicates that the requested action is not authorized.
</Definition>
    </Term>
    <Term termID="2">
        <Name xml:lang="en">
ActionNotSupported</Name>
        <Definition xml:lang="en">
    Indicates that the requested action is not supported by this service.
    </Definition>
    </Term>
    <Term termID="3">
        <Name xml:lang="en">
AllReturned </Name>
        <Definition xml:lang="en">  definition</Definition>
    </Term>
    <Term termID="4">
        <Name xml:lang="en">
ChangedSinceReturnsAll</Name>
        <Definition xml:lang="en">Indicates that when an attribute provider does not support this
parameter and returns all the data addressed by Select regardless of the latest modification times.
Some or all data returned might actually be older than the time specified in the ChangedSince
parameter.
</Definition>
    </Term>
    <Term termID="5">
        <Name xml:lang="en">DataTooLong</Name>
        <Definition xml:lang="en">Indicates that the size of the requested data exceeds the preset
threshold of the service provider. This is only for modifications.
</Definition>
    </Term>
    <Term termID="6">
        <Name xml:lang="en">ExistsAlready</Name>
        <Definition xml:lang="en">Indicates that the request is trying to modify to a preexisting
value.
</Definition>
    </Term>
    <Term termID="7">
        <Name xml:lang="en">InvalidData </Name>
        <Definition xml:lang="en">Indicates that the request contains invalid data
</Definition>
    </Term>
    <Term termID="8">
        <Name xml:lang="en">InvalidResourceID</Name>
        <Definition xml:lang="en">Indicates that the ResourceID of the request is invalid.
</Definition>
    </Term>
    <Term termID="9">
        <Name xml:lang="en">InvalidSelect</Name>
        <Definition xml:lang="en">Indicates that the Select element is invalid in the request.
</Definition>
    </Term>
    <Term termID="10">
        <Name xml:lang="en">MissingNewDataElement</Name>
        <Definition xml:lang="en">Indicates that the request specifies no new data to be updated.
</Definition>
    </Term>
```

```
    <Term termID="11">
        <Name xml:lang="en">MissingResourceIDElement</Name>
        <Definition xml:lang="en">Indicates that the request fails to specify the ResourceID
</Definition>
    </Term>
    <Term termID="12">
        <Name xml:lang="en">MissingSelect</Name>
        <Definition xml:lang="en">Indicates that the request fails to specify the Select element
</Definition>
    </Term>
    <Term termID="13">
        <Name xml:lang="en">NoMoreElements</Name>
        <Definition xml:lang="en">Indicates that the returned data ends prematurely for lack of
elements.
</Definition>
    </Term>
    <Term termID="14">
        <Name xml:lang="en">NoMultipleAllowed</Name>
        <Definition xml:lang="en">Indicates that multiple requests are not allowed
</Definition>
    </Term>
    <Term termID="15">
        <Name xml:lang="en">NoMultipleResources</Name>
        <Definition xml:lang="en">Indicates that the use of multiple resources is not allowed
</Definition>
    </Term>
    <Term termID="16">
        <Name xml:lang="en">TimeOut</Name>
        <Definition xml:lang="en">Indicates that the request has failed because of timeout.
</Definition>
    </Term>
    <Term termID="17">
        <Name xml:lang="en">UnexpectedError</Name>
        <Definition xml:lang="en">Indicates that an unexpected error has occurred</Definition>
    </Term>
</ClassificationScheme>
```

Annex B (informative): Sample instances

B.1 Sample SOAP Header

```
<soapenv:Header>
    <sb:Correlation id="NTTAEB9DE0EB0B1A89B00797A14C6EE85F6" messageID="uuid:debbffd3-4ea8-973e-
5463-e5ecc2d95dde" timestamp="2004-03-10T05:59:01Z"
soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next" soapenv:mustUnderstand="1"
xmlns:sb="urn:liberty:sb:2003-08"/>
    <sb:Provider providerID="https://UKCP.com:8443/" soapenv:mustUnderstand="0"
xmlns:sb="urn:liberty:sb:2003-08"/>
    <wsse:Security soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="1" xmlns:wsse="http://schemas.xmlsoap.org/ws/2003/06/secext">
        <wsse:BinarySecurityToken EncodingType="wsse:Base64Binary" ValueType="wsse:X509v3"
wsu:Id="X509Token" xmlns:wsu="http://schemas.xmlsoap.org/ws/2003/06/utility">
MIICBDCCAW2gAwIBAgIBUTANBgkqhkiG9w0BAQQFADA1MQswCQYDVQQGEwJVUzEUMBIGA1UEChMLTGliZXJ0eSBJT1AxEDAOBgNV
BAMTB1Rlc3QgQ0EwHhcNMDMxMjA1MTQ0MjI1WhcNMDQxMjA0MTQ0MjI1WjBMMQswCQYDVQQGEwJVUzEYMBYGA1UEChMPTGliZXJ0
eSBJT1AgamtzMSMwIQYDVQQDExpudHQtYS1zaWduLmxpYmVydHktaW9wLm9yZzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
q9Sl+JvcHKNjjtE/v70TKMMXo+Ft05RBy/XUruHZsuH0b206MYG+CCBoPyJ6Jv+6+ZsCgIEjfJ6lqRZRZmPdGv92zcBHHO1/k1/G
G7FPNFH+mrUm/66nRDysv6JFMzW7+Ct7132IUUiTEvj45CXHEcyy8JUffd1J1+FOrVweAEUCAwEAAaMNMAswCQYDVR0TBAIwADAN
BgkqhkiG9w0BAQQFAAOBgQBwqsW22HMTLTcxn3jiifP+yBjKRaYpikrRzffeJ8XtLUrHCkm7ZOX/OeqidHAARB4lTxmITCB3LbHm
ViAk4G66K4Yb9Y0FFVJCFyaYHnY6W6oLDkTv5IMqDL//vV6QF9boO2gvTpap4WL5+6meNmCyWKoeO4CuwX3qys5yrA8opg==</ws
se:BinarySecurityToken>
        <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:SignedInfo>
                <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
                20010315"/>
                <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                <ds:Reference URI="#NTTAEB9DE0EB0B1A89B00797A14C6EE85F6">
                    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                    <ds:DigestValue>/vqZkvIo2MkbAntQ3j0+I0QsZ4k=</ds:DigestValue>
                </ds:Reference>
                <ds:Reference URI="#NTT43EBDA48A7965082DA284C13DE33EFDE">
                    <ds:Transforms>
                        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                    </ds:Transforms>
                    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                    <ds:DigestValue>js4Cmrbteuy9Epti94O9+xFj7yk=</ds:DigestValue>
                </ds:Reference>
            </ds:SignedInfo>
            <ds:SignatureValue>
Rh9MenehPh/9zIB/8wNg4tCKaLIs5ayiRbfKrepXpD9qbsIOVjZ0/2R1ChiX/WaDANtVtdfj/sD3utjTLLRNiXKF45RWKQtzZT3e
RG2elAfm7a9ZnWgFBm0Q+/kSPmPHzo3aCx9K8yVUPmdg/S8BWjh5VLvz9U99JDJKF4FEx3o=
</ds:SignatureValue>
            <ds:KeyInfo>
                <wsse:SecurityTokenReference Usage="sec:MessageAuthentication"
                xmlns:sec="urn:liberty:sec:2003-
                08">
                    <wsse:Reference URI="#X509Token"/>
                </wsse:SecurityTokenReference>
            </ds:KeyInfo>
        </ds:Signature>
    </wsse:Security>
</soapenv:Header>
```

B.2 Example of Query

```
<Query queryID="QR001">
    <ResourceID>http://profile-provider.com/d8ddw6dd7m28v628</ResourceID>
    <QueryItem itemID="preference" changedSince="2001-12-17T09:30:47-05:00">
        <Select type="abbreviated">tva:profile:UserSearchPreferences</Select>
    </QueryItem>
    <QueryItem itemID="where">
        <Select type="abbreviated">tva:profile:UserLocation</Select>
    </QueryItem>
</Query>
```

B.3 Example of QueryResponse

```
<QueryResponse xmlns="urn:tva:profile:2005" xmlns:mpeg7="urn:tva:mpeg7:2005"
xmlns:tva="urn:tva:metadata:2005" queryIDRef="QR001" timeStamp="2001-12-17T09:30:47-05:00">
    <Status code="OK" requestIDRef="preference"/>
    <Data itemIDRef="preference">
        <TVAMain xml:lang="en" publisher="TV-Anytime.org" publicationTime="2006-12-
        17T09:30:47-05:00" rightsOwner="TV-Anytime.org">
                    <tva:UserDescription>
                <tva:UserPreferences>
                    <tva:UserPreference>
                        <mpeg7:FilteringAndSearchPreferences preferenceValue="90"
                        xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001">
                            <mpeg7:ClassificationPreferences>
                                <mpeg7:Genre href="urn:mpeg:GenreCS">
                                    <mpeg7:Name>Soccer</mpeg7:Name>
                                </mpeg7:Genre>
                            </mpeg7:ClassificationPreferences>
                            <mpeg7:CreationPreferences>
                                <mpeg7:Creator preferenceValue="70">
                                    <mpeg7:Role href="urn:tva:metadata:cs:TVARoleCS:2002:V43">
                                        <mpeg7:Name>Participant</mpeg7:Name>
                                    </mpeg7:Role>
                                    <mpeg7:Agent xsi:type="mpeg7:OrganizationType">
                                        <mpeg7:Name>
                                            <mpeg7:GivenName>Manchester United</mpeg7:GivenName>
                                        </mpeg7:Name>
                                    </mpeg7:Agent>
                                </mpeg7:Creator>
                            </mpeg7:CreationPreferences>
                        </mpeg7:FilteringAndSearchPreferences>
                    </tva:UserPreference>
                </tva:UserPreferences>
            </tva:UserDescription>
        </TVAMain>
    </Data>
</QueryResponse>
```

B.4 Example of Modify

```
<Modify xmlns="urn:tva:profile:2005" xmlns:mpeg7="urn:tva_mpeg7:2005"
xmlns:tva="urn:tva:metadata:2005" modifyID="MR001">
    <ResourceID>http://profile-provider.com/d8ddw6dd7m28v628</ResourceID>
    <Modification itemID="M001">
        <Select type="abbreviated">Genre</Select>
        <NewData>
            <TVAMain xml:lang="en" publicationTime="2006-12-17T09:30:47-05:00">
                <tva:UserDescription>
                    <tva:UserPreferences>
                        <mpeg7:FilteringAndSearchPreferences>
                            <mpeg7:ClassificationPreferences>
                                <mpeg7:Genre href="urn:tva:metadata:cs:ContentCS:2005:3.1.1.9"
                                preferenceValue="10">
                                    <mpeg7:Name xml:lang="en">Sports</mpeg7:Name>
                                </mpeg7:Genre>
                            </mpeg7:ClassificationPreferences>
                        </mpeg7:FilteringAndSearchPreferences>
                    </tva:UserPreferences>
                </tva:UserDescription>
            </TVAMain>
        </NewData>
    </Modification>
</Modify>
```

B.5 Example of ModifyResponse

```
<ModifyResponse xmlns="urn:tva:profile:2005" modifyIDRef="MR001">
    <Status code="Failed" requestIDRef="M001">
        <StatusDescription xml:lang="en"
        href="urn:tva:profile:cs:StatusCS:2005:7">InvalidData</StatusDescription>
    </Status>
</ModifyResponse>
```

B.6 Sample of WDSL

```
<wsdl:definitions xmlns:typens="urn:tva:metadata:user:wsdl:2004-07"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:pp="urn:tva:metadata:user:2004-07" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="urn:tva:metadata:user:wsdl:2004-07"
name="pp">
    <types>
        <xsd:schema>
            <xsd:import namespace="urn:tva:metadata:user:2004-07"
            schemaLocation="tva_metadata_v13.xsd"/>
            <xsd:annotation>
                <xsd:documentation>(An attempt at) TV Anytime User Profile Services
            WSDL</xsd:documentation>
            </xsd:annotation>
        </xsd:schema>
    </types>
    <message name="Query">
        <part name="body" element="pp:Query"/>
    </message>
    <message name="QueryResponse">
        <part name="body" element="pp:QueryResponse"/>
    </message>
    <message name="Modify">
        <part name="body" element="pp:Modify"/>
    </message>
    <message name="ModifyResponse">
        <part name="body" element="pp:ModifyResponse"/>
    </message>
    <portType name="DataServicePort">
        <operation name="QueryOperation">
            <input message="pp:Query"/>
            <output message="pp:QueryResponse"/>
        </operation>
        <operation name="ModifyOperation">
            <input message="pp:Modify"/>
            <output message="pp:ModifyResponse"/>
        </operation>
    </portType>
</wsdl:definitions>
```

Annex C (normative): Schema files

The *TV-Anytime* profile exchange schema listed in the present document has been aggregated into a series of xsd file, forming the reference documentation, contained in archive ts_1028220603v010101p0.zip, which accompanies the present document.

- tva_profile_exchange_6-3_v111.xsd
- xenc-schema.xsd
- xmldsig-core-schema.xsd

The profile exchange metadata scheme imports other files that need to be present in order to be valid:

- xml.xsd
- tva_mpeg7.xsd
- tva_metadata_3-1_v131.xsd

These files are available in archive ts_1028220301v010301p0.zip accompanying TS 102 822-3-1 [3].

Annex D (informative): Bibliography

- Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML): http://www.oasis-open.org/committees/security.
- IETF RFC 1945: "Hypertext Transfer Protocol - HTTP/1.0": http://www.ietf.org/rfc/rfc1945.txt.
- IETF RFC 2119: "Key words for use in RFCs to Indicate Requirement Levels" http://www.ietf.org/rfc/rfc2119.txt.
- IETF RFC 2616: "Hypertext Transfer Protocol - HTTP/1.1 http://www.ietf.org/rfc/rfc2616.txt.
- Liberty Reverse HTTP Binding for SOAP Specification.
- Liberty ID-WSF Security & Privacy Overview, Version 1.0.
- Liberty ID-WSF 1.0 Static Conformance Requirements, Version 1.0.
- Liberty ID-WSF Draft Guidelines.
- Liberty ID-WSF 1.0 Errata, Version 1.0.
- Liberty Alliance Developer Tutorial: http://www.projectliberty.org/resources/tutorial_draft.pdf.
- Liberty Alliance Technical Glossary Version 1.3: http://www.projectliberty.org/specs/draft-liberty-glossary-1.3-errata-v1.0.pdf.
- Liberty Alliance Specifications: http://www.projectliberty.org/resources/specifications.php.
- Liberty ID-WSF Discovery Service Specification, Version 1.1: http://www.projectliberty.org/specs/liberty-idwsf-disco-svc-v1.1.pdf.
- Liberty ID-WSF Data Services Template Specification, Version 1.0: http://www.projectliberty.org/specs/draft-liberty-idwsf-dst-1.0-errata-v1.0.pdf.
- Liberty ID-SIS Personal Profile Service Specification, Version 1.0: http://www.projectliberty.org/specs/liberty-idsis-pp-v1.0.pdf.
- Liberty ID-WSF SOAP Binding Specification, Version 1.1: http://www.projectliberty.org/specs/liberty-idwsf-soap-binding-v1.1.pdf.
- Liberty ID-WSF Architecture Overview: https://www.projectliberty.org/specs/liberty-idwsf-overview-v1.1.pdf.
- Liberty ID-WSF Implementation Guidelines: https://www.projectliberty.org/specs/draft-liberty-idwsf-guidelines-v1.0-12.pdf.
- Liberty ID-WSF Interaction Service Specification, Version 1.1: https://www.projectliberty.org/specs/liberty-idwsf-interaction-svc-v1.1.pdf.
- Liberty ID-WSF Security Mechanisms Specification, Version 1.2: https://www.projectliberty.org/specs/liberty -idwsf-security-mechanisms-v1.2.pdf.
- Liberty ID-WSF Client Profiles Specification, Version 1.1: https://www.projectliberty.org/specs/liberty-idwsf-client-profiles-v1.1.pdf.
- Liberty ID-WSF Authentication Service Specification, Version 1.1: https://www.projectliberty.org/specs/liberty-idwsf-authn-svc-v1.1.pdf.

List of tables

Table 1: Abbreviation for XPath Expressions 14

Table 2: Description of Discovery Option Keywords 15

Table 3: Description of Values for Status Element 15

History

Document history		
V1.1.1	January 2006	Publication