



VAMDC-TAP Data Access Protocol Specification

Document Information

Editors:	L. Nenadovic, M. Doronin
Authors:	G. Rixon
Contributors:	VAMDC WP6 working group
Type of document:	standards documentation
Status:	draft
Distribution:	public
Work package:	WP6
Version:	12.07
Date:	03/08/2012
Document code:	
Document URL:	http://www.vamdc.org/documents/vamdc-tap_v12.07.pdf

Abstract: VAMDC-TAP is a protocol for a class of web services giving access to databases in VAMDC. The protocol is a specialization of IVOA's TAP protocol. The required features of TAP and the specializations are defined here.

Version History

Version	Date	Modified By	Description of Change
V0.1	30/09/2010	G. Rixon	first draft
V11.05	27/05/2011	L. Nenadovic	editing for first release
v11.12	21/12/2011	G. Rixon	Changelog updates, VOSI endpoints clarification
v12.07	03/08/2012	M.Doronin	Update to mention last-modified header, VSS2 query

Disclaimer

The information in this document is subject to change without notice. Company or product names mentioned in this document may be trademarks or registered trademarks of their respective companies.

License

This document is protected by the Creative Commons license CC BY-ND 4.0.

For the license detail, please visit: <http://creativecommons.org/licenses/by-nd/4.0/>

You are free to:

- *Share* - copy and redistribute the material in any medium or format for any purpose, even commercially.

Under the following terms:

- *Attribution* - You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- *NoDerivatives* - If you remix, transform, or build upon the material, you may not distribute the modified material.

All rights reserved

The document is proprietary of the VAMDC consortium members.

This document reflects only the authors' view. The European Community is not liable for any use that may be made of the information contained herein.

Acknowledgements

VAMDC is funded under the "Combination of Collaborative Projects and Coordination and Support Actions" Funding Scheme of The Seventh Framework Program. Call topic: INFRA-2008-1.2.2 Scientific Data Infrastructure. Grant Agreement number: 239108.

CONTENTS

1	The VAMDC variant of the Table Access Protocol (TAP)	1
1.1	Required features	1
1.2	Query language	2
1.3	Format of results	2
1.4	Standard view of database	2
1.5	VOSI availability	3
1.6	VOSI capabilities	3
1.7	Registration	4
1.8	HTTP Header Information	4
1.9	HTTP Status Codes	5
1.10	Making a synchronous query	5
1.11	Volume limitation example	6
1.12	Standard Compliance Checklist	6
2	References	8
3	Change log for data-access protocol	9
3.1	Changes between v11.05 and v11.12	9
3.2	Changes between v11.12 and v12.07	9

THE VAMDC VARIANT OF THE TABLE ACCESS PROTOCOL (TAP)

Many data-sets in VAMDC include information that can be rendered in the VAMDC-XSAMS data model. Data in that common model could be transformed to and from a table model which uses the same columns for all data-sets. If all the data-sets had this table model as part of the schemata of their databases, then a SQL query to that model would work on all data-sets, and the results could be written in a common format.

VAMDC-TAP is a protocol for data-access services that provide the common table model matching VAMDC-XSAMS and which can return the results of queries in VAMDC-XSAMS. VAMDC-TAP services accept queries in a restricted form of SQL (VSS2: VAMDC SQL Sub-set #2) and return results in VAMDC-XSAMS or in certain tabular formats. Implementations of VAMDC-TAP map queries from the common table-model to the actual schemata of their databases.

VAMDC-TAP provides “virtual data”. I.e., it associates data selection criteria, defined by a query text, with an archived data-set, defined by the address to which the query is sent, the two combined in one URL. Each such URL represents the results of the query as if they had been pre-computed and stored on a web server. The data URLs are semi-permanent; they can be copied between application, bookmarked, emailed to colleagues, etc.

VAMDC-TAP is based on IVOA’s [Table Access Protocol \(TAP\)](#). TAP already provides virtual data and allows us to plug in our query language VSS2 and our data model VAMDC-XSAMS.

VAMDC-TAP is defined as a web-service protocol. That means that VAMDC-TAP services are driven by GET and POST requests to HTTP (or HTTPS) URIs. Low-level details of the protocol are defined by the HTTP RFCs. Further, the service can be implemented in any language and on any database engine without breaking interoperability.

Section 1.2 lists the features required in a conforming VAMDC-TAP service. Sections 1.3 to 1.9 inclusive define details of these features. Section 1.10 is not part of the specification but instead explains how to form and execute a VAMDC-TAP query.

1.1 Required features

A VAMDC-TAP implementation must be organized as a TAP service. This means that the implementation must be a RESTful web-service and must provide web resources in the pattern mandated by TAP.

In the TAP standard, some features are mandatory and some are optional. VAMDC-TAP uses a sub-set of the mandatory features and some of the optional ones.

A VAMDC-TAP service must support at least the following features:

- synchronous data-query, as specified in TAP.
- XML description of service capabilities as specified in TAP by reference to the [Virtual Observatory Support Interfaces \(VOSI\)](#) standard
- VSS2 query language

- XSAMS as an output format
- a standard view of the relational data (see below)
- VOSI availability readout
- VOSI capabilities readout

If the service provides all of these features then it may be registered as VAMDC-TAP.

A VAMDC-TAP service should support all the mandatory features of TAP. If it does so, it may be registered as both VAMDC-TAP and as TAP. (See section below for an explanation of the distinction.)

If a VAMDC-TAP service is to be registered as both TAP and VAMDC-TAP, then it should provide the VOSI output that specifies the details of tables and columns for use in queries. Without these metadata TAP is very hard to use.

A VAMDC-TAP service may support output formats other than XSAMS. The most-useful formats are tabular: VOTable for virtual-observatory integration and CSV for use in spreadsheets. If the service is to be registered as a TAP service, the TAP standard says that the service must support VOTable.

A VAMDC-TAP service may support query languages other than VSS2. If the service is to be registered as a TAP service, the TAP standard says that the service must support Astronomical Data Query Language (ADQL).

1.2 Query language

A VAMDC-TAP service must support VAMDC SQL sub-set #2 or VSS2 as its primary query-language. This language is specified in a query by setting the HTTP parameter LANG to VSS2. The service must not be sensitive to the case of the parameter value, but clients should use upper case for this value.

1.3 Format of results

A VAMDC-TAP service must support XSAMS as its primary output-format. This format is specified in a query by setting the HTTP parameter FORMAT to XSAMS. The service must not be sensitive to the case of the parameter value, but clients should use upper case for this value.

When the format is XSAMS, the service must return the results with the MIME type `application/x-xsams+xml`. The service must not use `text/xml` for these results.

TAP allows the value of the FORMAT parameter to be either the MIME type of the results or a keyword denoting the format. The MIME type for XSAMS is ambiguous and can easily be confused with VOTable or other XML formats. Clients must not use the MIME type when specifying XSAMS output.

1.4 Standard view of database

A VAMDC-TAP service must provide a relational view of its database that can be queried using the SQL-sub-set language VSS2. VSS2 has no FROMs clause, so VSS2 queries implicitly address the database as a single table. A VAMDC-TAP service must be arranged to support this.

When implementing a VAMDC-TAP service for a particular database, the implementor must define the *restrictable* quantities: these are the columns of the standard view on which constraints may be placed in the WHERE clause of a query. The restrictables must be taken from the set defined in VAMDC's standard dictionary, in which the names, data types, units and semantics are specified. A given implementation need not support every item in the dictionary. The restrictables for a service must be noted in the service's registration.

The implementor of a service must also define the *returnables*: these are the terms from the VAMDC dictionary which will appear in the results of a query. The returnables must also be noted in the registration.

If a service supports both TAP and VAMDC-TAP, the tables available for a TAP query need not be related to the standard view. One of the main reasons for supporting TAP is to give access to a wider range of tables.

1.5 VOSI availability

Notes on service availability, current and planned, are provided in an XML document. The availability metadata help in monitoring the VAMDC system and in managing downtime. A service installation may use the availability document to announce a number of conditions, including planned down-time for maintenance and unavailability of the database when the web service itself is available.

A VAMDC-TAP service must provide the availability document, in the form defined by IVOA's Virtual Observatory Support Interfaces (VOSI) standard, at the location mandated by the TAP standard.

1.6 VOSI capabilities

A service "capability" is an XML element describing the use of one aspect of the service. The capability states the URL for accessing that aspect and may add other metadata. A VAMDC-TAP installation will have a sequence of capabilities for different aspects, including a primary capability for the query protocol itself; the capabilities are distinguished by their `standardID` attributes. This sequence of capabilities is combined into the capabilities (XML) document and that document is copied from a URL on the VAMDC-TAP service into the VAMDC registry to form the machine-readable part of the registration.

A VAMDC-TAP service must provide the capabilities document as defined by Virtual Observatory Support Interfaces (VOSI) standard, at the location mandated by the TAP standard.

A VAMDC-TAP service must include the following capabilities in its capabilities document. (The notation `{x}y` for an XML type indicates the type `x` in the namespace `y`.)

- The VAMDC-TAP protocol, with structural type `{http://www.vamdc.org/xml/VAMDC-TAP/v1.0}VamdcTap` and standard ID `ivo://vamdc/std/VAMDC-TAP`.
- The generic protocol TAP, with structural type `{http://www.ivoa.net/xml/VOResource/v1.0}Capability` and standard ID `ivo://ivoa.net/std/TAP`.
- The capabilities, with structural type `{http://www.ivoa.net/xml/VOResource/v1.0}Capability` and standard ID `ivo://ivoa.net/std/VOSI#capabilities`.
- The availability, with structural type `{http://www.ivoa.net/xml/VOResource/v1.0}Capability` and standard ID `ivo://ivoa.net/std/VOSI#availability`.

In the capabilities document, structural types must be stated using the `xsi:type` attribute, except where the default type, `{http://www.ivoa.net/xml/VOResource/v1.0}Capability`, is used.

The XML schemata defining the parts of the registration are available on-line.

Namespace	Location of schema
http://www.vamdc.org/xml/VAMDC-TAP/v1.0	http://www.vamdc.org/xml/VAMDC-TAP/v1.0
http://www.ivoa.net/xml/VOResource/v1.0	http://www.ivoa.net/xml/VOResource/v1.0
http://www.ivoa.net/xml/VODataService/v1.1	http://www.ivoa.net/xml/VODataService/v1.1
http://www.ivoa.net/xml/VODataService/v1.0	http://www.ivoa.net/xml/VODataService/v1.0
http://www.ivoa.net/xml/VOSICapabilities/v1.0	http://www.vamdc.org/downloads/xml/VOSI-capabilities-1.0.xsd

The capabilities document should refer to these schemata using the `xsi:schemaLocation` attribute on the document element. This makes it easier to validate the document. However, the registration process will still work in the absence of `xsi:schemaLocation`.

1.6.1 VAMDC-TAP specific capabilities information

The following information must be included in VAMDC-TAP capability registration block:

- Service access URLs, including mirrors addresses, in `interface/accessURL` elements
- Implemented version of standards in `versionOfStandards` element, **12.07** for current standards.
- Node software implementation used, including version, in `versionOfSoftware` element

- Collection of sample queries in *sampleQuery* elements, that may be used for node monitoring task or to give an overview of contained data and query strategies for the node.

Sample queries must correspond to the following criterias:

- must be executed quickly (within seconds)
 - must result in valid XSAMS documents provided as the response
 - must result in documents that return a portion of all the specific database content, i.e. all the elements that may be returned in documents for any query should be returned in at least one document returned as a response to sample query.
- A set of *returnable* keywords, indicating the major elements filled in XSAMS. If node software is not using *returnable* keywords internally, only a brief set of most important keywords that are specific to this database may be returned.
 - A set of *restrictable* keywords that may be used to query the node. See the **Dictionary** document for supported **restrictable** and **returnable** element values.

1.7 Registration

A VAMDC-TAP service must be registered. The registration document must be of type [CatalogService \(v1.0\)](#) or [CatalogService \(v1.1\)](#) as defined by IVOA (i.e. it must use the VODataService standard in either of two versions).

The registration must include the capability elements copies from the capabilities document described above.

1.8 HTTP Header Information

1.8.1 Statistics

A VAMDC-TAP service should provide information/statistics about the amount of data that will be returned for a specific query in the HTTP headers of the reply to the query. This allows a client (e.g. the portal) to use the HEAD method (instead of GET) on the same query-URL to gather information before the query is actually executed and the data transferred.

The names of the headers to be used are

- VAMDC-COUNT-ATOMS
- VAMDC-COUNT-MOLECULES
- VAMDC-COUNT-SPECIES
- VAMDC-COUNT-SOURCES
- VAMDC-COUNT-STATES
- VAMDC-COUNT-COLLISIONS
- VAMDC-COUNT-RADIATIVE
- VAMDC-COUNT-NONRADIATIVE

Their values should be the count of the corresponding blocks in the XSAMS schema, e.g. the number of radiative transitions that will be returned for this query. With a reasonable database layout the nodes should easily be able to gather these numbers by running COUNT queries on their corresponding tables.

1.8.2 Volume limitation

A VAMDC-TAP service can limit the amount of data it returns via the synchronous interface, for example to prevent the fetching of the whole database or for performance reasons. The service must then fill the HTTP-header of the response with the field VAMDC-TRUNCATED that contains the percentage that the returned data represent with respect to the total amount available for that query. It is up to each service to decide both where and if to put the limit and how to implement it, for example the number of states or processes. Response documents for the queries that lead to volume limitation must remain valid XSAMS, including all references between elements.

1.8.3 Document size estimate

VAMDC-APPROX-SIZE HTTP header is intended to provide the estimation of the size of the response document. It should return an integer value, representing estimate uncompressed document size in megabytes.

1.8.4 Data modification time

A VAMDC-TAP service may add the **Last-Modified** header to a response. This header has the syntax and general semantics specified for HTTP 1.1, but also has special meaning within VAMDC-TAP.

If this header is used, it must refer to the modification time of the database supplying the data extract. The value of the header should be the time of the last change to the data actually used in forming the response. If the service installation cannot supply this specific time it may use instead the time of last modification to any relevant part of the database.

1.9 HTTP Status Codes

Following HTTP1.1 Status Codes must be implemented by the node software for the SYNC TAP endpoint:

HTTP Code	meaning	Content type	Response body
200	Request processed normally, data is present.	application/x-xsams+xml	XSAMS instance document
204	Request processed, but no matching data found	none	none
400	bad request with malformed query string or missing restrictable	unspecified, may be application/x-votable+xml	unspecified, may be a votable with error message
404	not used, will be encountered if the endpoint is wrong	unspecified, may be application/x-votable+xml	unspecified, may be a votable with error message
500	internal crash	unspecified, may be application/x-votable+xml	unspecified, may be a votable with error message

1.10 Making a synchronous query

The base URL for a TAP service can be discovered from the registry. From this, the access URL for the query can be derived: add /sync¹ to the base URL and then add parameters to define the specific query.

Parameter name	Meaning	Supported values in VAMDC-TAP
REQUEST	Requested operation	doQuery
LANG	Name of query language	VSS2
FORMAT	Format for results of query	XSAMS, VOTABLE, application/xml ²
QUERY	Text of query	As per query language

¹ This access-URL identifies the web-resource for synchronous queries. Asynchronous queries are sent to a separate web-resource.

² Implies VOTable.

Parameter names are insensitive to case: service implementations must accept any mix of case.

The parameter values are URL-escaped to replace illegal characters with hexadecimal codes (e.g. each space is replaced by %20). In practice, only the QUERY parameter needs to be escaped. Clients of the service must escape the parameters before sending the request.

This is a plausible example of a query URL, fully decorated with parameters:

```
http://some.server/some/path/sync?REQUEST=doQuery&LANG=VSS2&FORMAT=XSAMS&QUERY=select%20*
```

Here, the base URL, found in the registry, is `http://some.server/some/path`. The query is `SELECT *`.

The query is initiated by an HTTP-GET request to the access URL. The HTTP response carries the results of the query in the specified format.

1.11 Volume limitation example

Volume example is implemented for the Django-based prototypes and activated for the VALD node which now returns max 1000 transitions (plus corresponding states and sources, of course). Similar limits are easily done for the other nodes in a few lines of code. In addition to the HTTP-header, the VAMDC-XSAMS generator also puts a comment into the beginning of the XML-document which also notifies of the truncation.

For example, a query like this:

```
wget -S -O bla.xml "http://vamdc.fysast.uu.se:8888/node/vald//tap/sync/?REQUEST=doQuery&LANG=VSS2"
```

will show the HTTP-header as:

```
VAMDC-TRUNCATED: 2.9 %
```

In Django node software implementation you will also find the following section at the top of the returned XML:

```
<!--
  ATTENTION: The amount of data returned has been truncated by the node.
  The data below represent 2.9 percent of all available data at this node that
  matched the query.
-->
```

1.12 Standard Compliance Checklist

Each VAMDC node implementing VAMDC-TAP protocol must pass the requirements of the following checklist to be fully interoperable within VAMDC infrastructure and do not interfere with other nodes. This list is not exclusive, but if a node does not comply with any of the checklist items, it can not be included into 12.07 public system.

1. Node **must** return valid XSAMS documents as defined by the latest VAMDC-XSAMS standard in any case when the response document is required.
2. When database contains no data corresponding the query, node **must** respond with HTTP 204 Status Code, with empty response body, both for HEAD and GET requests to the /sync endpoint of VAMDC-TAP protocol.
3. Node **must** respond to the vss2 query **SELECT SPECIES** with an XSAMS document containing only species information about all molecules, atoms and particles contained in node database, without states and processes data. Response should be given within a reasonable amount of time, no more than 30 seconds.
4. Node **must** support gzip content encoding of transferred data, as defined in HTTP specification. This is employed to preserve the bandwidth and speed up the transfer of data, since XSAMS documents compress very well.

5. Node **must** support HTTP HEAD requests to the TAP sync endpoint, giving sensible values in VAMDC-COUNT-* headers. Response to HEAD requests should be generated within a reasonable amount of time, no more than 30 seconds. Values may be inaccurate, but should give a view on how much data will be returned by the node for GET request.
6. If node contains transitional data, it **must** support queries by **RadTransWavelength**, defining transition wavelength in vacuum in Angstroms. Use of this keyword is a common convention for clients querying the transition data.
7. Node **must** provide sensible sample queries in Capabilities registration.

REFERENCES

- International Virtual Observatory Alliance (IVOA) : <http://www.ivoa.net>
- Table Access Protocol (TAP) : <http://www.ivoa.net/Documents/TAPRegExt/20110127/index.html>
- Virtual Observatory Support Interfaces (VOSI) : <http://www.ivoa.net/Documents/VOSI/20100311>

CHANGE LOG FOR DATA-ACCESS PROTOCOL

3.1 Changes between v11.05 and v11.12

The details of VOSI capabilities and registration were clarified. The XML schema with namespace <http://www.vamdc.org/xml/VAMDC-TAP/v1.0> is now explicitly in force for the VAMDC-TAP registration. This formalizes the approach used in the 11.09 system.

3.2 Changes between v11.12 and v12.07

- Added mention of Last-Modified HTTP header.
- Replaced VSS1 mention with VSS2.
- Changed Mime-Type of XSAMS to be returned as `application/x-xsams+xml`
- Added Quality Assurance checklist chapter
- Explained VAMDC-TAP registration capabilities elements