

Document Availability Information API (DAIA)

The **Document Availability Information API** (DAIA) defines a data model with serializations in JSON, XML, and RDF to encode and query information about the current availability of documents. The specification is being refined since 2008. See <http://purl.org/NET/DAIA> for a detailed specification. An XML Schema can be found at <http://purl.org/NET/DAIA/schema.xsd> (also in HTML) and an RDF ontology at <http://purl.org/ontology/daia>. A reference implementation in Perl is available at CPAN to create your own DAIA servers, wrappers, and clients. It includes a simple [DAIA validator and converter](#). A public validator is available at <http://daia.gbv.de/validator/>.

Introduction

More and more services of libraries are also offered through APIs that fit into a service oriented architecture (SOA). Most of the fundamental actions are covered by (web) APIs. For Search there is SRU/SRW, Z39.50, and OpenSearch, for Harvest and Syndicate there is OAI-PMH, RSS, and Atom Syndication, for Upload and Edit there is SRU Update, and the Atom Publishing Protocol, for Copy and Provide there is unAPI, COinS, and Microformats, and for Identity Management there is Shibboleth and OpenID. At the same time library formats evolve from semi-structured digital catalog cards (MARC) to more structured formats such as MODS and RDA. Availability of documents and holdings in libraries, however, has lacked a clear and open specification. The Document Availability Information API (DAIA) fills this gap.

Specification

DAIA was first designed as a simple REST-API with a conceptual data model, independent from serialization formats, in mind. It started as expression of basic availability concepts in XML and JSON, which are specified at <http://purl.org/NET/DAIA>. A serialization in RDF is being developed now, to be incorporated in the final DAIA specification. The current DAIA ontology is located at <http://purl.org/ontology/daia>.

Overview

The DAIA data model can be viewed from a tree-oriented data view like XML and JSON or from a graph-oriented data view like RDF. The following two diagrams illustrate both views

DAIA model as tree

Also available [On one page \(A4\)](#) for printing.

DAIA model as graph

What DAIA is not about

The basic question to be answered by the Document Availability Information API (DAIA) is:

Is a specific document available at a specific institution and how can I get it?

This implies that DAIA is not mainly about:

DAIA is not about loan - Traditional libraries only held physical objects that you were able to look at or even loan. In the digital world this is only one special case. As soon as you have digitized an object, the difference between viewing the digital document and copying the digital documents becomes senseless (copyright advocates may have a different view, but copyright as the right to prevent circulation of information that has already been published is hopefully becoming an antiquated concept). Availability information also contains information about the possibility to loan an object, but it's not the main point.

DAIA is not about discovery - Either you know which document you want - so you have an identifier, or you first have to find out which document you want - and its identifier. DAIA is not about this task. The same applies for the institution you ask for availability. Either you can refer to it with an identifier or you cannot. Please remember that in the digital world objects and concepts without a known identifier are as lost as a book that has been put to a wrong shelf in a large library magazine.

What DAIA is about: The concept of Availability

To understand DAIA you need to keep in mind some general design considerations:

- availability always depends on context - it is not a property of a document or item but a specific service to use the item that is provided by an institution at a specific time.
- The format contains as little as possible implicit assumption "how traditional libraries work". Some of these assumptions only apply to the pre-digital age and are changing. DAIA is not about libraries, it's about the availability of documents.
- The type of a document is not a property of its availability, but only one factor that influences available services. That's why publication types etc. are not part of DAIA.

In the end availability is a binary value about a service you want to do with a resource. In the library context the basic services can be defined as:

1. **presentation** - an item can be used inside the institution (in their rooms, in their intranet etc.)
2. **loan** - an item can be used outside of the institution (by lending or online access)

3. **interloan** - an item can be used mediated by another institution. That means you do not have to interact with the institution that was queried for this item. This includes interlibrary loan as well as public online resources that are not hosted or made available by the queried institution.
4. **openaccess** - an item can be used immediately without any restrictions by the institution, you don't even have to give it back. This applies for Open Access publications and free copies

In addition there can be multiple facets like delay time or specific limitations etc. but this is all optional.

Implementations

- [Perl Reference Implementation](#) in Perl, available at CPAN
- [DAIA Converter and Validator](#) (also available at CPAN)
- <http://daia.gbv.de/> - GBV DAIA Server
- [TU Hamburg-Harburg](#) - Example: http://www.tu-harburg.de/daia/daia_ws.php?ppn=540319058
- [doctor doc](#) (also Open Source)
- [BSZ](#) has implemented a DAIA-Server for [Koha library system](#)
- [VuFind DAIA-Driver](#) and [VuFind PICA-Driver](#) based on DAIA
- [First test server for Hamburg](#)
- ...
- [DAIA Implementations](#) phpDAIA, DAIA Perl, XSLTs, ...

Additional documents

- [Defining Availability Information with the Document Availability Information API \(DAIA\)](#). Lightning talk at the ECDL 2009
- [Reference implementation in Perl](#)
- [Icons für Bibliotheksanwendungen](#). Test to use PICOL-based icons for DAIA (in German)

In the context of [ILS Discovery Interface Task Force](#) and its official recommendation (revision 1.1, released in December 2008) DAIA fits to the [GetAvailability](#) method (section 6.3.1).

Existing standards beside DAIA

See notes at [DAIA/Review](#).

More information

This page is being worked on. Have a look at the German introduction [Verfügbarkeitsrecherche mit DAIA](#) and the [DAIA Format](#) specification. Please send comments and questions to jakob dot voss at gbv.de.

Diese Seite wurde zuletzt am 18. Dezember 2012 um 15:26 Uhr geändert.