INTERNET-DRAFT
draft-dasl-requirements-01.txt
Feb 24, 1999
Expires August 24, 1999

Jim Davis
Xerox Corporation
Saveen Reddy
Microsoft Corporation
Judith Slein
Xerox Corporation

Requirements for DAV Searching and Locating

Status of this Memo

Abstract

The Distributed Authoring and Versioning protocol [WEBDAV] defines simple mechanisms to assign and retrieve values for properties. This

document presents requirements for a WebDAV extension to support efficient searching for resources based on WEBDAV properties and content. These requirements are intended to be the basis for the DAV Searching and Location (DASL) protocol.

1. Introduction

Motivation for DASL

WEBDAV and HTTP provide support for client-side search, but not server-side search. The GET method defined in [HTTP] allows clients to retrieve a resource's content; the PROPFIND method defined in [WEBDAV] allows clients to retrieve a resource's properties. Having retrieved a resource's properties and/or content, the client can compare them to its search criteria to determine whether the resource is of interest. Although this client-side searching is logically sufficient, and requires no modifications to the server, it comes at a significant cost, because it makes inefficient use of network resources. A client must retrieve properties and content for each resource under consideration. Furthermore, it does not take advantage of server intelligence. Servers capable of searching can use sophisticated mechanisms to generate results: internal caching of intermediate search results, content-indexing, etc.

Even simple, common queries may expose these limitations. Consider the query "find all text files modified during the last week." When such a query is extended to a large number of clients searching against a single server, the limitations become more apparent. Client-side searching has difficulties scaling in these cases.

DASL allows for server-side searching. Server-side searching allows the client to formulate a query and have the server perform task of selecting the resources that fit the criteria. This overcomes both of the limitations of client-side searching described above. The benefit is a searching solution that scales; the cost is that the server software becomes more complex.

This document presents requirements for any protocol that might be proposed for DASL. These requirements come from considerations of the scenarios presented in [SCENARIOS], from the need to support the WebDAV object model, the use of HTTP, and general IETF rules. We provide rationale for those requirements whose justification is not obvious. We assign each requirement a priority, one or two, where one is higher. The significance of the number is that priority one requirements are those that any protocol must define to be considered successful, where priority two requirements are those that are desirable but not necessary. There are no priority three requirements at present.

2. Terminology

scope
: a set of resources to be searched.

criteria
: an expression against which each resource in the search scope is evaluated.

result set
: a set of records, one for each resource for which the search criteria evaluated to True.

record
: a description of a resource. A result record is a set of properties, and possibly other descriptive information

result
: A result is a result set, optionally augmented with other information describing the search as a whole.

result record definition
: a specification of the set of properties to be returned in the result record

sort specification
: a specification of an ordering on the result records in the result set.

search modifier
: an instruction that governs the execution of the query but is not part of the search scope, result record definition, the search criteria, or the sort specification. An example of a search modifier is one that controls how much time the server can spend on the query before giving a response.

query
: A query is a combination of a search scope, search criteria, result record definition, sort specification, and a search modifier.

query grammar
: a set of definitions of XML elements, attributes, and constraints on their relations and values that defines a set of queries and the intended semantics.

schema
: a listing, for any given grammar and scope, of the properties and operators that may be used in a query with that grammar and scope.

Hit highlighting
: is a specification of the location(s) within a resource containing text that matched a content-query. It allows clients to provide visual cues to a user to identify segments in a text resource that cause them to match content-based queries.

paged results
: allows a client to request that the server return a subset of the result set rather than the entire set. In subsequent calls to the server, additional results from the same query can be requested. Paged results are intended to improve the

performance and manageability of search results.

In addition to the terms defined above, this document uses terminology consistent with [HTTP] and [WEBDAV].

Requirements are divided into five categories, and numbered within each category. The categories are Scope, Criteria, Record Definition, Other and Discovery.

3. Requirements: Scope

S1: It is possible to specify at least one resource in the scope (P1). It is possible to specify a set of distinct, unrelated resources in the scope (P2).

> As this is the first requirement in the document, we explain the notation. S1 means this is the requirement one in the Scope section, P1 means that the requirement to have at least one resource in scope is essential, and P2 means that allowing more than one is nice but not required.
>
> Rationale: Supporting multiple resources in scope could be difficult to define, because distinct resources may have different sets of metadata, support different operators, or have different access rights.

S2 It is possible to specify a WebDAV collection as a scope (P1).

S3: It is possible to specify other types of resources in a scope (P2).

> Rationale: A client might wish to determine whether a given resource was of interest without transferring it.

S4: When the scope is a collection, it is possible to specify the depth (P1).

> Users often intend to scope their searches either to the immediate children of a container or to extend the search recursively to the container's children. Furthermore, depth control is needed to prevent servers from performing unnecessary work.

4. Requirements: Criteria

Criteria generalities

C1: It is possible to search properties in a query (P1). It is possible to search both DAV-defined and application-defined properties in a query (P1).

Further requirements for properties are below.

C2: It is possible to search content in a query (P1).
Note that at this writing, unlike property searches, there is no single widely accepted semantics for content-based queries. Further requirements for content criteria are below.

C3: It is possible to search both properties and content in a single query.

C4: It is possible to combine criteria with Boolean operators (i.e. and, or, not) (P1).

Criteria for properties

C5: It is possible to include undefined properties in a query without error (P1).
Rationale:. This arises from the property model of DAV. Unlike the more familiar relational model, DAV does not define tables or schema for resources, hence there is no guarantee that all properties will be defined for all resources. Moreover, DAV allows an client to store arbitrary properties on arbitrary resources. Therefore DASL must support queries that use properties that are not defined on all resources in the scope. If such a query failed, there would be no way to locate the desired resources.

C5.1: It is possible to test whether a property is defined (P1).

C6.1: It is possible to compare a property value to a constant value (P1).

C6.2.1: It is possible to compare property values to other properties of the same resource (P2).

C6.2.2: It is possible to compare property values to other properties of other resources (P2).

Note that this may involve a "join". We do not expect the first version of the DASL protocol to meet this requirements.

C6.3: It is possible to compare property values to results of expressions (P2).

C6.4: It is possible to match property values with string-ending wildcards (P1). It is possible to match property values with pattern matching operators similar to the SQL "like" operator or regular expressions (P2).

The minimum is necessary to enable DASL to locate resources by content type, e.g. to locate all image files by comparison with

"image/*". More powerful comparisons are useful when strings encode structured data such as times or lists. Note that these are constraints on what the protocol must define, not on what servers must necessarily implement.

C6.5: It is possible to compare property values taking into account their structure (P2).

Explanation: Some WebDAV properties are defined to contain strings (e.g. DAV:getcontenttype), but others contain structured values (e.g., DAV:resourcetype, DAV:lockdiscovery). Support for structured value criteria is needed, for example, to locate resources locked in a certain manner by a certain principal. The working group consensus is that this feature, while undeniably very useful, is so difficult to define that it is better for DASL to proceed than attempt to define it. Also, there is much activity in the W3C to define an XML query language, and it was felt better to wait for this to complete than to define a competing standard.

C7.1: The protocol defines an equality operator (P1).

C7.2: The protocol defines relative operators (P1).

C8: The protocol defines means to specify case sensitivity (P1).

Note this does not say that all DASL servers must support both case-sensitive and case-insensitive comparisons, but only that the protocol must be able to express a client's preference, and define behavior in the case where the server cannot support that preference.

C9: The protocol supports language-specific definitions for string comparison and sorting (P1).

Different cultures define different rules for string comparison, e.g. for collating sequence and for significance of diacritics. Cross-language comparison is out of scope for DASL, but comparisons within the same language must be done with the appropriate semantics.

Requirements: Criteria for content searches

C10: It is possible to search content of any text media type (P1). The definition of "searching content" for DASL means locating sequences of characters in the contents of the resource.

DASL defines no requirements for searching for structure within text media types (e.g. for finding character strings only within certain HTML tags.) This functionality is too complicated to specify at the present time.

DASL defines no requirements for searching other media types that might contain text (e.g. subtypes of application). Searching non-text media types (e.g.images, audio) is out of scope for DASL.

C11.1: It is possible to search for words that are within a specified number of words (or, for some languages, characters) of each other (P1).

This is often called 'near' search. It is used to locate concepts that can be expressed in more than one way using the same set of words, e.g. one might locate both "the President's impeachment" and "the impeachment of the President".

C11.2: It is possible to search for words that occur within the same grammatical context, e.g. same phrase, sentence, or paragraph (P2).

This is sometimes called 'in' search.

C12.1: It is possible for a client to control whether content searches does or does not use a stemming comparison (P2).

C12.2: It is possible for a client to request comparisons using phonetic similarity (e.g. soundex) (P2)

C12.3: It is possible for the client to request keyword expansion (thesaurus expansion) (P2).

C13: It is possible for a client to conduct a relevance search (P2). In such a search, the query consists of a set of words (perhaps an entire resource), and the result is a list of resources whose contents most closely resemble the query, sorted in decreasing order of resemblance.

5. Requirements: Results

R1: It is possible to specify a sorting for the result set (P1).

R2: It is possible to specify a set of properties to be returned in the result records, distinct from the properties in criteria (P1).

For example, a query might ask for "the authors of those documents under 10K in size". In this case, the criterion relates only to the size, but the desired result record contains only the author.

R3: It is possible for a client to request limits on the resources consumed in creating of transmitting in the result set (P1).

Some queries can potentially return very large result sets. Clients that are good citizens will voluntarily limit the size of such results. In addition, some servers may charge money for queries.

R3.1: It is possible for a client to limit the number of records in the result set (P1).

This is the most meaningful unit of resource consumption to the client.

R4: It is possible for the server to return fewer result records than match the criteria (P1).

"Client proposes, server disposes".

R5: It is possible to a client to request paged results (P1).

Paged retrieval is necessary if result sets are very large and if clients must also present a responsive interface to a user. Note that this requirement is silent about whether a server implements paged results by storing results from a query or recalculating them as needed.

6. Requirements: Other

O1: It is possible to support multiple query grammars (P1).

Rationale: A particular query grammar may not expose all the useful searching functionality of a server. Clients should be allowed to query a server using any grammar that takes advantage of those special server capabilities. This requirement also allows DASL to define an initial limited query grammar which meets all the mandatory requirements without needing to address all the desirable, but non-mandatory requirements.

O2: It is possible to extend the basic grammar defined by DASL (P1).

03: It is possible for the server to redirect a query (P1).

This is useful when a server is not able to search a given scope, but can refer the client to another server which is able to search the scope.

O4: It is possible for the client to request hit highlighting (P2).

7. Requirements: Discovery

D1: It is possible for a client to discover the set of query grammars supported by a server (P1).

> Without this, it is not very useful for servers to support multiple grammars.

D2: It is possible for a client to discover the schema supported by a server for a particular grammar with a particular scope (P1).

> Note that the schema may differ depending on the scope. Query schema discovery allows a client to use optional properties and operators supported by a server.

D3: It is possible for a client to determine information about the properties within a scope (P2).

> This information can enable a user interface to help a user to construct a valid query, for example by providing meaningful names for properties, constraints on values, hints about data type, and so on, or information about expected performance, for example whether a property is indexed (and hence more quickly searched).

8. External Requirements

DASL must describe how to perform searches on internationalized content and properties. This is in keeping with IETF policy.

Information intended for user comprehension must conform to the IETF Character Set Policy [CHAR].

The WebDAV working group is currently addressing the standardization of mechanisms for authors to submit variants and version of resources, or for means of exposing access control. DASL should provide mechanisms that can query for variants, versions, and access control but can not do so until they are defined. Likewise, DASL may contribute requirements to access control (e.g. control over querying).

9. Related Work

Z39.50: "Information Retrieval (Z39.50): Application Service Definition and Protocol Specification".
http://lcweb.loc.gov/z3950/agency/

Z39.50 Profile for Simple Distributed Search and Ranked Retrieval

http://lcweb.loc.gov/z3950/agency/profiles/zdsr.html

The STARTS Protocol
http://www-db.stanford.edu/~gravano/starts.html

The Harvest Information Discovery and Access System
http://mordor.transarc.com/afs/transarc.com/public/trg/Harvest/

10. References

[CHAR] H.T. Alvestrand, "IETF Policy on Character Sets and Languages", June 1997, internet-draft, work-in-progress, draft-alvestrand-charset-policy-02.txt.

[HTTP] R. Fielding, J. Gettys, J. C. Mogul, H. Frystyk, and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2068, U.C. Irvine, DEC, MIT/LCS, January 1997.

[SCENARIOS] Henderson, R. et al Scenarios for DAV Searching and Locating. Work in progress. draft-henderson-dasl-scenarios-00.html, September 18, 1998 (Expires Mar 23, 1999)

[WEBDAV] Y. Y. Goland, E. J. Whitehead, Jr., A. Faizi, S. R. Carter, D. Jensen, "Extensions for Distributed Authoring and Versioning on the World Wide Web", IETF Proposed Standard, RFC 2518

11. Authors' Addresses

Jim Davis
Xerox Corporation
3333 Coyote Hill Road
Palo Alto, CA 94304
Email: jdavis@parc.xerox.com

Saveen Reddy
Microsoft Corporation
One Microsoft Way
Redmond WA, 9085-6933
email: saveenr@microsoft.com

Judith Slein
Xerox Corporation
800 Phillips Road 105-50C
Webster, NY 14580
Email: slein@wrc.xerox.com