

# Real Estate Transaction Standard v1.x Technical Review:

## *Analysis and Findings*

Prepared for:



December 6, 2007

SOA Software, Inc.  
12100 Wilshire Blvd., Suite 1800  
Los Angeles, CA 90025  
866-SOA-9876

**SOA**  
software

## Table of Contents

<b>1</b>	<b>OVERVIEW</b>	<b>3</b>
<b>2</b>	<b>SUMMARY OF ANALYSIS PROCESS</b>	<b>4</b>
<b>3</b>	<b>RETS AS AN INDUSTRY STANDARD</b>	<b>6</b>
3.1	RETS IS A GATEWAY TO DATABASES	6
3.2	RETS SPECIFIES A LOW-LEVEL PROTOCOL BASED ON HTTP 1.1	6
3.3	RETS DOES NOT DEFINE REAL ESTATE TRANSACTIONS	6
3.4	RETS DOES NOT DEFINE A REAL ESTATE DATA MODEL	7
3.5	RETS DOES NOT DEFINE A REAL ESTATE REFERENCE ARCHITECTURE	7
<b>4</b>	<b>RECOMMENDED CHANGES TO RETS 1.X STANDARD</b>	<b>8</b>
4.1	CLARIFICATIONS AND CORRECTIONS	8
4.1.1	<i>Corrections to the BNF definitions for DMQL</i>	8
4.1.2	<i>Include valid values for Boolean data</i>	8
4.1.3	<i>Be more specific on use of MaximumLength in Table Metadata</i>	9
4.1.4	<i>Be more specific on valid value ranges in DMQL</i>	9
4.1.5	<i>Clarify server behavior for Restricted Indicator</i>	9
4.1.6	<i>Be specific on when server should return &lt;MAXROWS/&gt;</i>	9
4.1.7	<i>Clarify foreign key handling for COMPACT-DECODED result sets</i>	9
4.2	MINOR ENHANCEMENTS AND IMPROVEMENTS	9
4.2.1	<i>Expand the definition of valid lookup values</i>	9
4.2.2	<i>Add a time zone indicator to all date/times</i>	10
4.2.3	<i>Add the time to all versioning date stamps</i>	10
4.3	LARGER ENHANCEMENTS AND IMPROVEMENTS	10
4.3.1	<i>Improve the implementation of paged result sets</i>	10
4.3.2	<i>Include the ability to deliver role-based metadata</i>	10
4.3.3	<i>Make RETS more strict in the proper encoding of XML content</i>	11
4.3.4	<i>Develop a light-weight version of RETS</i>	11
4.4	MAJOR CHANGE: THE HANDLING OF NAMES	12
4.4.1	<i>Require the use of Well-Known Names</i>	12
4.4.2	<i>Include a full data dictionary of Standard Names and definitions</i>	12
4.4.3	<i>Require the use of Standard Names</i>	12
4.4.4	<i>Rename System Names to Server Names</i>	12
4.4.5	<i>Assign a "namespace" to all Server Names</i>	13
4.4.6	<i>Allow both Standard and Server names in a DMQL query</i>	13
<b>5</b>	<b>AREAS FOR FUTURE THOUGHT</b>	<b>14</b>
5.1	SERVER RESULTS NOT MATCHING METADATA	14
5.2	ISSUES WITH DIGEST AUTHENTICATION	14
5.3	REDUCE BINDING TO HTTP	14
5.4	STATUS OF UPDATE CAPABILITY	14

## 1 Overview

---

NAR's Center for Realtor Technology has engaged SOA Software to perform an evaluation of the technical content and capabilities of the Real Estate Transaction Standard (RETS) version 1.x. This assignment grew out of analysis of the RETS standards development process performed by SOA earlier this year. That work culminated in a presentation to the RETS community at their April 2007 meeting in Austin, Texas.

While SOA Software has no established position in the MLS or real estate industry, we do have extensive experience participating in the standards creation process. SOA Software is also a producer of software products that are extensively involved in implementing industry standards. The SOA staff that performed this evaluation also has broad experience in developing and implementing industry standards that dates back to some of the early EDI efforts and moving forward to include HIPAA, ACORD, and OAG standards.

This document, along with a presentation at the December 2007 RETS meeting in Miami, Florida, reports on the RETS 1.x technical review. This document discusses the actual review process. An evaluation of RETS in relationship to other industry standards is presented. Finally, the detailed findings and recommendations of the technical review are included along with some areas for future study.

## 2 Summary of Analysis Process

---

The technical review of RETS 1.x consisted of the following phases:

- ❑ A detailed study of RETS 1.5 and 1.7 documentation including:
  - ❑ The actual standards documents
  - ❑ The RETS 1.7 Standard Names Data Dictionary
  - ❑ The Real Estate Domain Model at [http://retsserver.realtors.org:8080/protege/protege\\_applet.html](http://retsserver.realtors.org:8080/protege/protege_applet.html)
  
- ❑ A review of standards that have been implemented in other industries
  - ❑ The Open Application Group standards for CRM
  - ❑ The ACORD standards for the insurance industry
  - ❑ The Mortgage Industry Standards Maintenance Organization (MISMO) standards used in mortgage and related transactions
  - ❑ The RosettaNet standards used in the electronic component supply-chain industry
  - ❑ The Department of Justice “Global Justice” standards such as JXDM (Justice XML Data Model) and JRA (Justice Reference Architecture).
  - ❑ A sampling of HIPAA related standards such as Health Level 7 (HL7), the National Council for Prescription Drug Programs (NCPDP) and the National Uniform Billing Committee (NUBC).

Every one of these standards was created for the purpose of fostering interoperability and information sharing within an industry. Each of these standards addresses this goal by specifying the message formats and messaging protocols for conducting business transactions. For instance, the ACORD standards contain the definition of XML messages for such things as changing the beneficiary of a life insurance policy and the MISMO standards specify the message formats and exchange protocols for initiating a residential mortgage application and for ordering an appraisal.

- ❑ The detailed findings of the RETS 1.x Technical Review Workgroup were studied. This group consisted of experienced developers of products and tools that implement the RETS standards. The findings from this group have been summarized into the analysis presented in this document.

Finally, one-on-one interviews with a select group of current RETS developers were conducted at the recent RETS Working Group Trimester meeting in Chicago. The people interviewed were all experienced in developing and supporting products and tools that implement the RETS standards. The people interviewed are shown in the following table. They represented a mix of both software vendor and MLS developers as well as having experience with implementing both RETS servers and RETS clients.

The interviews covered experiences and difficulties encountered developing RETS applications. Many of the interviewees were able to provide a historical perspective on the evolution of the RETS 1.x standards. Talks focused on the benefits RETS had provided, what needs to change with RETS, how RETS had shaped the functionality and implementation of their products, and the future outlook for RETS.

To summarize the general outcome of these interviews, there were several recurring observations:

1. Ambiguities and inconsistencies in the specifications were a serious impediment to initial adoption of RETS. Some of that has been corrected in later revisions but there is still a large body of “RETS lore” that must be understood in order to successfully implement RETS applications. Fortunately, that lore is fairly accessible in the form of reference implementations and the RETS email list server.
2. The lack of a consistently implemented data model is viewed as one of the major shortcomings of RETS 1.x. It seems servers do not always implement the existing Standard Names in any consistent manner. Data mapping and translation is sometimes viewed as a significant performance handicap. One of the largest efforts in any deployment of RETS applications is mapping of data between clients and servers.
3. There was the general feeling that RETS 1.x still has at least five years of useful life. None of the interviewees were actively looking at implementing RETS 2 and were unsure what sort of market forces might drive a migration to the new version of the standard.
4. Comments regarding DMQL centered around the need to correct several inconsistencies, inaccuracies, and omissions. When questioned specifically as to usability of DMQL, the consensus was that it was adequate for the task of accessing real estate data. Any initial difficulty early developers had with preparing or parsing DMQL requests were largely eliminated once common reference code was published.

## 3 RETS As an Industry Standard

---

As part of the technical review of RETS 1.x, we evaluated the content and intent of the existing standard against standards established by other industry segments. This analysis has resulted in the following observations about RETS as an industry standard.

### 3.1 RETS Is a Gateway to Databases

The RETS 1.x standard essentially specifies a gateway or conduit to a database of real estate listings and related information. In that role, it provides a means for information suppliers to describe their data and make that data available for use. It also provides mechanisms for information consumers to discover information content and form and to retrieve the information needed by the application. Specific evidence that RETS is a data access specification is the fact that one of the tools provided for RETS actually converts its DMQL query language directly to a SQL query to run against a real database.

### 3.2 RETS Specifies a Low-Level Protocol Based on HTTP 1.1

The RETS protocol is modeled as a web application. Each transaction is identified by the URL that it is sent to. For instance, the *GetObject* transaction might be sent to <http://www.mymls.com/rets/getobject> and the Search transaction would be sent to <http://www.mymls.com/rets/search>. The parameters for the request are passed in HTTP headers and the “query string”.

RETS transaction responses are, again, tightly coupled to the HTTP protocol. Some parts of the result are returned in HTTP headers but the majority of the result is returned as the HTTP document body. This can consist of a full XML document but the more commonly used format consists of structured, delimited data contained inside an XML wrapper document. One exception to this is the *GetObject* transaction which returns binary results like images as MIME attachments.

### 3.3 RETS Does Not Define Real Estate Transactions

In spite of its name, RETS does not actually define any real estate business transactions. RETS does define message formats for real estate database searching and information retrieval which, from some viewpoints, might be considered a form of business transaction. However, from a larger perspective, one would expect something called a “transaction standard” to include a more complete set of transactions used in the real estate business such as “renew listing” or “place offer”.

In reviewing standards from other industries, this is one of the first areas many of those standards focus on since facilitating the exchanging of business transactions between members of an industry was one of the primary motivations for establishing the standards in the first place

### 3.4 RETS Does Not Define a Real Estate Data Model

The first objective of most other industry standards analyzed has been to establish a common vocabulary to allow participants to exchange information. This common vocabulary also needs some structural definition. Some industry standards address this by providing a business data model that includes such things as:

- ❑ A domain model that has clear definitions for data elements. Definitions for elements such as “currency” or “telephone number” or “postal code” would be contained in this model.
- ❑ An entity model that specifies the principle data entities, their basic composition, and their relationships to other entities.

The current RETS standard addresses this by only specifying a mechanism for exchanging abstract (or metadata) information that allows a server to communicate its data model to a client program. RETS provides little or no guidance on the structure or content of the data supplied and consumed through its interfaces. Such guidance would give RETS data a uniformity that would promote easy interoperability. It would also foster a separation of data and tools that would allow product developers an opportunity to make use of a predictable set of data in new and innovative ways.

### 3.5 RETS Does Not Define a Real Estate Reference Architecture


Some of the newer, more comprehensive industry standards have begun including specifications of “reference architectures” that combines the definition of standard business processing components, the data model, and business transaction flows into a cohesive unit. This architecture provides a standardized model for performing business information processing across an industry. One notable example of this level of standardization can be seen in the Department of Justice’s Global Justice Reference Architecture <http://it.ojp.gov/globaljra>.

A reference architecture for the real estate industry would be a technical implementation architecture that would address a range of information sharing use cases, and provide a blueprint for implementing interoperable data sharing services and capabilities. This dynamic interoperability strategy will help to prevent incompatibilities, guide vendors and organizations on how to fit components together, and facilitate communication and interoperability between disparate sources.

It may turn out that the real estate industry does not have a level of complexity sufficient to warrant the effort required to create a reference architecture. However, the fact that other industry standards are beginning to address interoperability issues from the architectural level should cause the RETS community to consider such directions when planning the standard’s future direction.

## 4 Recommended Changes to RETS 1.x Standard

---

These recommendations are the result of RETS 1.x technical review and analysis. Many of the changes proposed here originated as findings from the RETS 1.x Technical Review Workgroup. These recommendations are organized a little differently than the original findings document and, in many cases, have been combined or summarized. Other recommendations, particularly some those with larger scope, are the result of the one-on-one interviews and the independent study of the RETS standards. These recommendations are identified with a little SOA Software icon (  ) in the margin.

The recommendations for changes to the RETS 1.x standards are grouped into the following categories:

1. Changes representing the continuing effort to improve the clarity and accuracy of the standards. In many cases, these changes are being recommended to make the standards more accurately reflect how actual implementations behave.
2. Some relatively small enhancements are being recommended to improve the usability of RETS. Some of these also reflect the changing nature of the companies using RETS
3. Some larger enhancements and extensions are also being recommended. While most of these recommended changes would be logical extensions the existing RETS functions, there are some which might require changes to existing implementations in order to remain compatible and compliant.
4. Finally, we make a set of recommendations for major changes that address what has shown through the interview process and other inputs to be a significant weakness in the RETS standard.

### 4.1 Clarifications and Corrections

Most of the findings from the RETS 1.x Technical Review Workgroup focused on clarifying or correcting the existing RETS standards. All of the items in this category came from those findings and it our recommendation that all of these clarifications and corrections be implemented.

#### **Editorial Corrections:**

##### **4.1.1 *Corrections to the BNF definitions for DMQL***

The RETS 1.x Technical Review Workgroup found several syntax errors in the BNF definitions for DMQL. None of these seems to have any negative impact on existing implementations

#### **Making Standards More Specific:**

##### **4.1.2 *Include valid values for Boolean data***

Surprisingly, the current standards are not specific as to what constitutes a valid Boolean value. It has been left up to the individual implementers and this has lead to some interoperability issues.

#### **4.1.3 Be more specific on use of *MaxLength* in Table Metadata**

The RETS 1.x Technical Review Workgroup suggested several changes to the standard that improve the treatment of the *MaxLength* attribute in the Table metadata. These suggested changes also clarify the values this attribute should have in various situations.

#### **4.1.4 Be more specific on valid value ranges in DMQL**

The standards are not as clear as they could be on what makes up valid value ranges in DMQL. The Technical Review Workgroup points out where the standards fail to make these rules clear but does not always suggest how the rules should change. These ambiguous areas need to be clarified so it is clear what constitute valid value ranges.

### **Changing SHOULD/MAY to MUST:**

#### **4.1.5 Clarify server behavior for *Restricted Indicator***

The current version of the standard allow servers to choose how they are going to represent data that a user is not allowed to see when the *RestrictedIndicator* has not been set. The Technical Review Workgroup recommends that this optional behavior should be made mandatory and we concur.

#### **4.1.6 Be specific on when server should return *<MAXROWS/>***

The current standard also allows a server to optionally send the *<MAXROWS/>* element in some situations. We agree with the Workgroup's recommendation that this element must always be returned whenever the server is not returning all rows that satisfy a search.

### **Eliminating Ambiguous Language:**

#### **4.1.7 Clarify foreign key handling for *COMPACT-DECODED* result sets**

This is also a Workgroup recommendation that the discussion about generating the *COMPACT-DECODED* format specifically state that foreign keys are not to be expanded.

## **4.2 Minor Enhancements and Improvements**

#### **4.2.1 Expand the definition of valid lookup values**

RETS 1.x Technical Review Workgroup has suggested that the definition of lookup values be expanded to include blanks, underscores, etc. Two things should be considered if this change is going to be made:

1. Handling of white space can be problematic in some environments. For instance, should multiple spaces be allowed? What if the number of spaces in the DMQL query value does not actually match the value stored on the host?
2. Comma should *not* be included in the list of additional characters to be allowed in lookup values. If it were, then DMQL parsing would be impacted.



#### 4.2.2 **Add a time zone indicator to all date/times**

As discovered in the one-on-one interview process, there is an increase in the number of RETS clients and servers that communicate across time zone boundaries. This increase is caused by the increase in MLS consolidations and the increase use of the Internet in real estate transactions among other factors. This tendency has pointed out an ambiguity in RETS regarding time values. When time values are communicated between RETS clients and servers in different time zones, there is simply no way of knowing what the actual time value is since neither the client nor the server necessarily has knowledge of what time zone the other one is in.

It is recommended that RETS change the definition of time values to incorporate a time zone designation. Along with this is the suggestion that clear rules be established for the interpretation of time values that have no time zone designation.

#### 4.2.3 **Add the time to all versioning date stamps**

The Workgroup has recommended that the Metadata version date be changed to a date/time to allow for multiple changes in a single day. This seems like a reasonable idea and should be expanded so that any versioning date stamp be changed to include the time as well.

### 4.3 Larger Enhancements and Improvements

#### 4.3.1 **Improve the implementation of paged result sets**

The ability of RETS clients and servers to process result sets in paged sections is currently compromised by inconsistencies and shortcomings of the standards that have produced inconsistent implementations. The RETS 1.x Technical Review Workgroup produced two important suggestions to help relieve this problem:

1. Require all compliant RETS servers to sort result sets by ascending value of the *KeyField* column.
2. Require all compliant RETS servers to support processing of the *Offset* attribute of DMQL searches.

In discussing this issue during interviews, the general feeling was that there is not a strong requirement for the deterministic cursor processing that is typical of database systems.



#### 4.3.2 **Include the ability to deliver role-based metadata**

Input received in the one-and-one interviews has shown that RETS servers are, increasingly, enforcing restrictions on information access based on the privileges granted to the authenticated user. This access control can have a large effect on the data that a server returns for a given request. Many times, clients need to be able to take different actions based on the access restrictions enforced on the server. These discussions in the interviews have created the recommendation that metadata be enhanced to provide the ability for the tables and fields disclosed in metadata to be adjustable based on the roles and privileges of the connected user.

There are still many details to be worked out in this recommendation not the least of which is the fact that RETS does not currently have the concept of a user's "role". Questions to be addressed in a working group that could be formed to design this feature would include:

How does one associate a given set of metadata with the access rights or a user?

How is metadata versioning affected by access rights?



#### 4.3.3 **Make RETS more strict in the proper encoding of XML content**

One recurring comment received during the one-on-one interviews was the issue of data content having a negative effect on message integrity. This is not a new issue for SOA Software which has a long and colorful experience with processing of XML messages. The solution is, fortunately, relatively straight forward but can be difficult to implement. Our suggested solution is to amend the standards to be very specific in how a RETS server (and to a lesser degree a RETS client) must encode the contents of XML attributes and elements. The answer to all of these complaints is not to restrict the data that can exist in a server's database. The answer is that properly encoded XML data can accurately represent any data content that is understandable to both the client and the server regardless of locale, language, codepage, or any other factor.

It is the recommendation that RETS be amended to require that clients and servers to observe the following:

- All XML document elements must obey Unicode UTF-8 (or UTF-16 depending on locale) encoding of all content.
- All XML content must use numeric character references for any character outside the range of 0x20-0x7F. Refer to *Section 4.1: Character and Entity References* in the latest W3C Recommendations for XML (<http://www.w3.org/TR/xml>).
- The characters ampersand ('&') and left angle-bracket ('<') can only be used as part of the actual XML markup. If those characters are needed in document content, then an entity reference must be substituted: use '&amp;' and '&lt;' instead of the forbidden characters.
- Quotation characters shall not appear in XML attributes. Entity references will be substituted if these characters are needed in attributes: use '&apos;' in place of a single-quote; use '&quot;' in place of a double-quote.

We further recommend that any compliance tests be amended to rigorously exercise this capability.



#### 4.3.4 **Develop a light-weight version of RETS**

The concept of a "RETS-lite" is something we have encountered every time we have spoken with RETS developers. Once again, the issue was discussed in the one-on-one interviews conducted in Chicago. Based on this, we recommend that an effort be started to develop a version of RETS that is simpler and requires less resource for implementation. Suggestions for the areas of RETS that can be changed to produce this simpler version of RETS include:

- Only use Standard Names and a standard data dictionary. This eliminates the requirement to process RETS metadata.
- Establish well known values for lookup fields to further eliminate the requirement for metadata.
- Only use full XML results, thus, eliminating the more complicated processing of the COMPACT result formats.
- Support a number of shorthand queries or transactions that have a fixed set of search properties and a fixed set of data in the result sets.

One motivation for this suggestion is the proliferation of Internet-based access to real estate data. The increasing number and sophistication of IDX (Internet Data Exchange) clients is evidence of this trend. The other business area that was discussed in the interviews was the increasing presence of MLS affiliations and consolidations. This creates the need for a quickly-implemented exchange of data from dissimilar sources.

## 4.4 Major Change: The Handling of Names

One recurring comment in the one-on-one interviews was the difficulty in mapping and translating data between clients and servers. In examining this issue, we discovered that the primary source of the difficulty stems from the inconsistent way servers treat data names and formats along with the inconsistent way data relationships are treated in RETS servers. It comes down to the fact that one of the subjects the RETS standard focuses on, the flexibility created by the metadata-based processing, has become an implementation liability due to the lack of standardization in the data model. This section presents some recommendations for relieving this barrier to adoption.

The sections below present a series of recommendations that, combined, are intended to address the most prevalent comment encountered in the interviews.

### 4.4.1 **Require the use of Well-Known Names**

RETS 1.x presents the concept of “well-known” names in several contexts: returned values, resource names, table names, etc. The RETS 1.x Technical Review Workgroup made note of several inconsistencies in the standard in the use of these Well-Known names. It was the Workgroup’s recommendation, which we strongly endorse, that the use of Well-Known names should be required for all compliant clients and servers

### 4.4.2 **Include a full data dictionary of Standard Names and definitions**

The majority of the RETS developers interviewed in Chicago expressed a desire for an organized reference of Standard Names. The RETS web site contains a PDF and XLS file of Standard Names but these do not match each other. The, apparently, more comprehensive source seems to be the Excel workbook. However, this fails to adequately organize the Standard Names into any sort of structure.

The RETS standard contains a simple diagram of the structure of the information returned from the *GetMetadata* transaction. Another part of the standard presents a list of Well-Known resource names. There is no place in the standard that presents a list of “standard” real estate data objects and the “standard” information those objects should contain.

It is our recommendation that RETS be enhanced by including a data dictionary that would define Standard Classes, Standard Names along with content rules for members of those classes, and Standard Relationships describing the ways Classes relate to each other. The net outcome of this effort is a clearly defined set of Standard Names along with clearly defined content rules and relationships.

### 4.4.3 **Require the use of Standard Names**

If RETS is going to have a place as the standard for the exchange of real estate information, then once a data dictionary has been established, its use must be required. The current RETS standard permit servers to choose how much of the existing standard names, if any, they implement. For some servers, they do not even properly implement the few existing standard names they do support. This lack of consistency is a major weakness in RETS 1.x as voiced by several interviewees.

We recommend that, once established, the dictionary of Standard Names, content rules, and relationships be the required implementation of all compliant servers and clients. For example, if a server can deliver a data element that contains the asking price of a property then that data element must be contained in the proper Standard Class, be identified by the proper Standard Name, presented in the proper format, and be included in search results using Standard Relationships.

### 4.4.4 **Rename System Names to Server Names**

The current RETS standard places undue emphasis on the native organization of the RETS server implementation. This emphasis comes about by calling the non-standard names “system” names.

This implies that these are the more important names in the implementation. Metadata elements such as *DeletedFlagField* in the Resource metadata currently reference a “system” name. Other relationships in the metadata are all based on the “system” names of data.

We recommend that these non-standard names be de-emphasized. The way we recommend this be accomplished is to call these names *Server Names*. This recasts these names as what they are – the individual implementations of a single server.



#### **4.4.5 Assign a “namespace” to all Server Names**

Once a stronger data dictionary has been made part of the RETS standard, there needs to be a mechanism to separate Standard Names from Server Names. The W3C standards for XML and XML-Schema have accomplished this with the concept of a “namespace” Before that, the C++ standard implemented the concept of namespace scopes.

Although we do not suggest an implementation mechanism, we do highly recommend that a namespace-like mechanism be established to separate the various Server Names from the common System Names. In this mechanism, the Standard Names might exist in the null or default namespace.



#### **4.4.6 Allow both Standard and Server names in a DMQL query**

Several times during the one-on-one interviews it seemed apparent that one of the causes of the Standard Names usability issues was the fact that a DMQL search request is limited to the existing “standard” names or “system” names, not both. Given the spotty implementation of standard names (some servers do not implement standard names at all), this places significant restrictions on a client application that wants to use standard names.

Once the concept of namespaces is introduced into the standard, it will be possible to intermix references to Standard Names and Server Names in the same DMQL search and the returned results.

We recommend that the *StandardNames* request attribute be removed from the standard in favor of a mechanism that allows a mixture of Standard Names and Server Names in the search and its results.

## 5 Areas for Future Thought

---

### 5.1 Server Results Not Matching Metadata

One comment that was made more than once during the one-on-one interviews was that there have been cases when the data returned from a RETS server did not match the characteristics that the same server had specified in its metadata. The comment was also made that, on several occasions, this mismatch has led to RETS client tools having to implement customized workarounds to compensate for this error. Integrating RETS clients and servers could be seriously jeopardized by this problem if not controlled.

The fact that this situation had been reported more than once and seemed to create more than a little amount of work when integrating products points to the possibility that this area of interoperability is not being adequately tested during the RETS certification process. Future consideration should probably be given to examining the RETS compliance tools and policy.

### 5.2 Reduce Binding to HTTP

As the RETS standard and features moves forward, consideration should be given to removing the current reliance on HTTP transport characteristics. We suggest that RETS will want to move in a direction that makes messages and exchanged data more self-defining. This will open up RETS to other message-passing models which, in turn, creates possibilities for new functional areas to be addressed. Some of the evolutionary changes that might be considered in moving away from a transport-centric message model are:

- Specify the transaction in the message structure rather than the HTTP URL.
- Move transaction properties into the message body instead of using the HTTP query string
- Return the processing result or error code as part of the reply message rather than using the HTTP return code.
- Specify a protocol and transport neutral alternative to MIME multi-part messages

### 5.3 Status of Update Capability

We were surprised at the low level of interest in the RETS update functions during the one-on-one interviews. Most respondents stated that they did not perceive any requirement for the feature in their own situation. Comments were also made that the feature is not yet ready for serious implementation. With only small exceptions, the overall feeling was that RETS update would be something to be looked at in the future. Maybe near-term efforts to improve and enhance RETS would be more profitably spent on other areas.

### 5.4 Issues with Digest Authentication

The RETS 1.x standards specify that Digest Authentication be used for the user login process. There is one significant issue with this technique in that it requires that both the client and the authentication agent must have access to the plain-text password. In many cases, this is not possible. For instance, most LDAP directories store only the one-way hash of the password. It is suggested that alternatives to Digest Authentication should be studied and considered for inclusion in RETS.