

# Information Dispersal Using SQRP: Short Query Response Protocol

Reza Ferrydiansyah, Veriyanta Kusuma, Etta Erinda Enggarini

Distributed Systems Laboratory, Informatics Department, Institut Teknologi Bandung, Indonesia

[reza@informatika.org](mailto:reza@informatika.org), [veriy@informatika.org](mailto:veriy@informatika.org), [rinda@informatika.org](mailto:rinda@informatika.org)

## Abstract

Currently there is so much information available on the Internet that it is increasingly difficult for a user to find the information he needs immediately. Part of the problem is the growth of the World Wide Web and the large amount of websites currently in existence.

We believe that for information to be more accessible to the general public, it must be easily and freely accessed and exposed in many sites simultaneously. Therefore in this paper we propose and describe the design, verification and implementation of a new protocol called SQRP (acronym for Short Query Response Protocol) that runs over the widely used TCP/IP. This protocol controls the sharing of information between Internet hosts to encourage faster information dispersal.

**Keywords:** Internet, Protocol, Information Dispersion, World Wide Web

## 1. Introduction

The growth and the accessibility of the Internet and in particular the World Wide Web has brought us to the age of information overflow. By looking at the number of pages on the web available we can surmise the number of information available.

One search engine, Google [1] reports that it can search for data in over 2 billion web pages. A typical search on Google will return more than 1 information source or page, sometimes even as many as 100 thousand pages or even more depending on the search parameters.

Clearly, there is so much information available that it is increasingly difficult for a user to find the information he needs immediately. Especially since a great deal of information is useful to only a very few users, and often for only a short period of time [2].

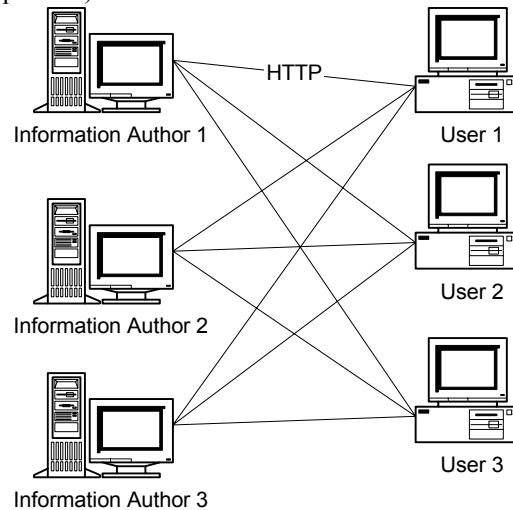
This problem is not limited at the user's end; there are some entities that need to give out information to the users. In this paper, we will refer to these entities as information authors. We believe that most information authors use the World Wide Web to give out their information. Due to the disorderly state of the WWW today it is also difficult for them to get their message across.

Figure 1 shows the current layout for information retrieval. Each user must find and connects directly to the information author's website via HTTP.

There are some problems with this approach, one being it is difficult to locate the correct website and afterwards locate the correct information in that website. Another problem is that the information author must have the knowledge to design a site and also update the site with the most current information. There is also the expense of having to have a server space where he can put his information.

In our opinion most websites are more suited to giving long term and relatively static information

(for example a product description manufactured by a certain company, or a company's organization) instead of the most up to date and dynamic information (the current trading price of a specific product).



**Figure 1.** Current layout for information flow

One way to simplify the distribution of information is that it must be easily and freely accessed and exposed in many places/sites simultaneously such as via various web portals. The process of distributing the information to various other sites is a specific instance of information dispersion.

The most common way of information dispersion is via parsing the information author's site or with an agreement to exchange or give information between the two parties involved. We believe that parsing websites and making agreements are difficult and should not be the primary way of obtaining the information in the first place.

Therefore we propose a new protocol that controls the sharing of dynamic information between Internet entities. This protocol is called SQRP an

acronym for Short Query Response Protocol. The protocol used is a simple ASCII based request-reply protocol and runs in the application layer of the TCP/IP stack, this is consistent with many of today's Internet protocols.

## 2. Related Work

The main method of information dispersion is that of information gathering. Web portals and web summary type-sites sends http requests to get data from websites, parses and extracts the information from those data according to its needs [3].

Automatic content extraction has the advantage that they can provide an immediate base of usable information but will generate some inappropriate keywords and miss generating others[2]. Some sites have also tried to ban automatic content extraction due to the resource that automatic content extraction uses [3].

Another problem concerning automatic content extraction is the time and regularity the extraction of content should take place. There is no way for an external entity of knowing when the data in an information author's website is updated.

## 3. SQRP Analysis

In the current information distribution methods, it is the collector who is actively retrieving data from various sources. An information author only puts the available data on the website according to its own needs and does not feel obligated to present the information in such a way that it would be easy for other sites to summarize.

We believe the information author must take an active role in dispersing information. One way to do this is to format and place the data in a predetermined location so that it can be easily taken by an information gatherer application.

Another important aspect is that information authors must be able to quickly and easily update the information in the predetermined location to keep the information up to date.

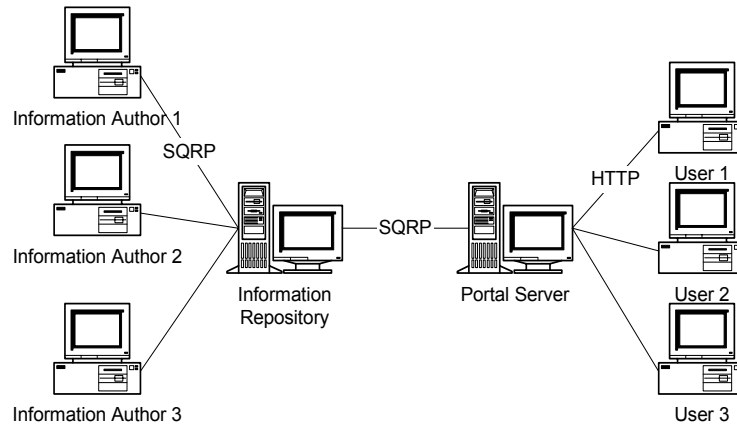
SQRP (Short Query Response Protocol) is a response-request communication protocol between an information author and an information repository, and between information repository and information gatherer entities. The architecture of information flow using SQRP is shown on figure 2.

In SQRP, an information author determines a set of query  $Q \{Q_1, Q_2, Q_3, \dots, Q_n\}$  and its corresponding answers  $A \{A_1, A_2, A_3, \dots, A_n\}$  where

$Q_i$  corresponds to  $A_j$  if and only if  $i=j$ . Both  $Q$  and  $A$  is limited in length (thus the word 'short' in SQRP). By limiting the length of the query and the answer we hope that the information placed in the repository is essential and to the point. Both  $Q$  and  $A$  is also ASCII text based with no formatting information embedded in it.

As shown in figure 2, an information author uses SQRP to update data in an information repository. This can be done via any application which supports SQRP, the information author does not even need a website. By using a protocol to update information, the physical storage of information is hidden from the information author.

A portal server or an information gatherer connects to the information repository via SQRP. There are two ways an information gatherer may request information using SQRP; the first method is by asking  $Q_i$  and receiving exactly one  $A_i$ . The second method is subscribing to  $Q_i$  and then receiving  $A_i$  every time the information author updates  $A_i$ .



**Figure 2.** Information flow using SQRP

The information gatherer then formats the information taken from the information repository to the need of the information gatherer's site.

With this structure a user does not need to know the exact website of the information he is looking for. The information gatherer can act as a central information server giving information from many information repositories. The user will only need to remember one website, which is the information gatherer's page.

SQRP is suited to give short information that are often updated such as theater shows, product price, stock price, and currency exchange information.

#### 4. SQR Design

SQR provides three main services:

1. Information Storage  
Used by the information authors, this service enables the information owner to add a query definition and to update the response to a particular query. The information author shall be an authenticated user.
2. Query Response  
Information repository returns short responses to predetermined queries sent by the information gatherers.
3. Information Subscription  
Lets an information gatherer entity subscribe to a query. In this service, a persistent connection is opened between the server and the gatherer; and whenever the information author updates the response to a particular query the server will send the updated information to the subscriber.

Each information author receives a space in the repository. Those spaces are divided into directories, which is used to place the Queries and the answers.

SQR does not specify how the information author (list of users) or list of directory is created nor does it specify how the implementation puts the data into the storage.

##### 4.1. Authentication Commands

These following commands specify the user authentication and are used mainly in the information storage service.

- a. USER NAME  
Syntax: USER <SP> <username> <CRLF>  
The argument <username> is a user identification that is required to authenticate user. This should be the first command transmitted by the user after control connections are made. In order to change the access control, information repository may allow a new USER command to be entered at any point. This has the effect of flushing any user, password, and account information already supplied and beginning the login sequence again.
- b. PASSWORD  
Syntax: PASS <SP> <password> <CRLF>  
The argument field is the user's password. This command must be immediately preceded by the user name command, and, for some sites, completes the authentication phase.

##### 4.2. Interaction Commands

These following commands specify the interaction between users, both information gatherers and the information authors, and the information

repository. These commands are used in the three services.

- a. QUIT/EXIT  
Syntax: EXIT <CRLF>  
This command causes the server to close the control connection and the user application shall be terminated.
- b. LIST ORGANIZATIONS DIRECTORY  
Syntax: LD <CRLF>  
The result of this command is a list of organization's directories sent by server to the user's application.
- c. LIST QUERY  
Syntax: LQ [<SP> <directory>] <CRLF>  
This command causes a list of queries in certain organization directory to be sent from server to user's application
- d. CHANGE DIRECTORY  
Syntax: CD <directory> <CRLF>  
This command causes the working directory to be changed into another organization directory specified in the argument.
- e. PRINT WORKING DIRECTORY  
Syntax: PWD <CRLF>  
This command causes the name of the current organization directory to be returned in the reply.

##### 4.3. Specific Commands

These following commands specify how to add a query definition and to update the response to a particular query. These commands are particularly used in the information storage functions.

- a. CREATE QUERY (CQ)  
Syntax: CQ <SP> <queryname> <CRLF>  
This command causes the query <query> specified in the argument to be created in the information repository. This command shall be followed by FILL INFORMATION command.
- b. FILL INFORMATION (FILL)  
Syntax: FILL <SP> <queryname> <SP> <QU> <information> <QU> <CRLF>  
This command causes the information of specified query to be stored. Query <queryname> specified in the argument must exist. If the query already has content information, the older content is replaced. The time sequence of a and b can be seen in figure 3c).
- c. DELETE QUERY (DELQ)  
Syntax: DELQ (<SP> <queryname> [...]) <CRLF>  
This command causes query specified in the argument and its information to be deleted from repository.

These following commands specify the query-response functions:

- a. SEND QUERY (QU)  
 Syntax: QU <queryname> <CRLF>  
 The result of this command is the content information of query specified in the argument was sent by information repository (figure 3b). The response by the server is in the form of +OK <queryname>: <information>
- b. SUBSCRIBE INFORMATION (SUBQ)  
 Syntax: SUBQ (<SP> <queryname> [...]) <CRLF>  
 This command causes content information of specified query(ies) to be returned. Every time the information updated, the new content will be sent while the connection is established. This command is specifically used by the information subscribe service (figure 3a). The response by the server is in the form of + <queryname> : <information>

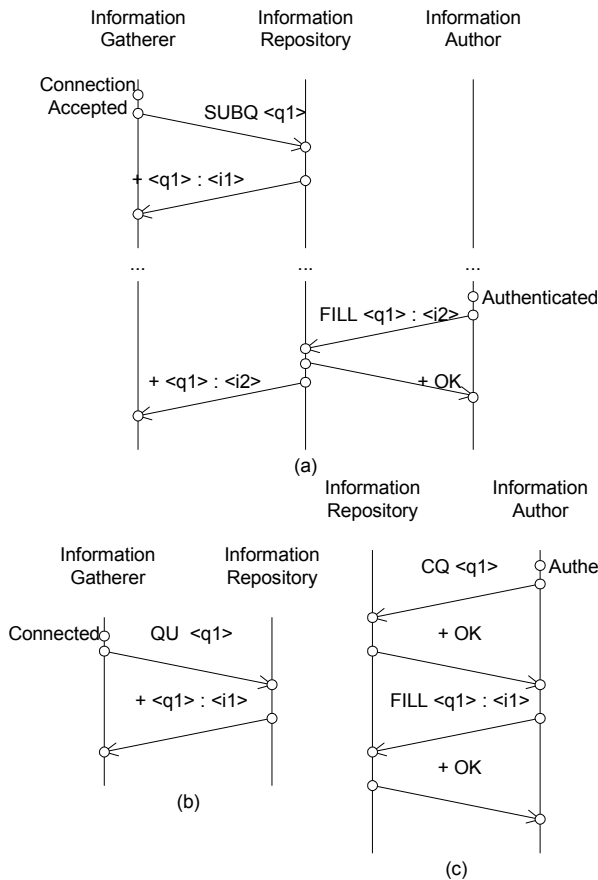


Figure 3: Time Sequence Diagram

#### 4.4 Syntax

The command arguments are given in Table 1.

No.	Arguments
1.	<username> ::= <string>
2.	<password> ::= <string>

3.	<information> ::= <string>
4.	<queryname> ::= <word>
5.	<commandname> ::= EXIT   LD   LQ   CD   PWD   USER   PASS   CQ   FILL   DELQ   QU   SUBQ
6.	<string> ::= <char>   <char><string>
7.	<char> ::= printable characters, any of the ASCII code 32 through 126
8.	<word> ::= printable characters, any of the ASCII code 33 through 126
9.	<SP> ::= ASCII character number 32 (Space)
10.	<CRLF> ::= ASCII character number 13 (Carriage Return Line Feed).
11.	<QU> ::= ASCII character quote (")

Table 1. Command arguments

#### 5. Conclusion and Future Work

Due to the number of information contained in the Internet it is difficult for a user to find given to the difficulty of finding the source containing the required information he needs. One way to solve this problem is to create distributed information repositories where each repository contains data from more than one sources.

Currently, information gatherers are actively looking for data. Information authors should also have an obligation to disperse its information to several repositories.

We have designed SQRP especially for the purpose of information dispersal. SQRP makes it possible to create a new method of information flow so that the above solution is possible. With SQRP, information gathering portals can easily take the data they need, while on the other hand information authors can also easily update the information and also better disperse the information to its user base.

#### 6. References

- [1] ---, [www.google.com](http://www.google.com), website, Accessed July 15, 2002.
- [2] Bowman CM, Danzig PB, Manber U & Schwartz MF: "Scalable Internet Resource Discovery: Research Problem and Approaches", *Comm of the ACM*, 37, 8, Aug '94. (1994)
- [3] Wagner C, Turban E. "Are Intelligent E-Commerce Agents Partners or Predators", *Comm of the ACM*, 45, 5, May 2002. (2002)
- [4] Estier TH, "What is BNF Notation" <http://cui.unige.ch/db-research/Enseignement/analyseinfo/AboutBNF.html>, website, Accessed July 18 2002.