

CP372 Group 29

Request for Comments

Category: Informational

Server/Client Bibliography Protocol – SCBP/1.0

Status of This Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Abstract

The Server/Client Bibliography Protocol is a design protocol for a network of communication between a server and multiple client applications. It is a connection-oriented protocol in which many tasks and requests can be executed with ease by connected clients. The server will be able to handle incoming bibliography data and store it for the duration of execution. The client will be able to add, remove, get, and update existing bibliography data, in which other clients can then interact with.

This specification represents the first design of this protocol, referred to as “SCBP/1.0”.

Table of Contents

Status of This Memo	1
Abstract.....	1
1.0 Introduction	2
1.1 Purpose	2
1.2 Server	2
1.3 Client	2
2.0 Communication.....	2
2.1 Overview	2
2.2 Message Formats	2
2.2.1 SUBMIT.....	2
2.2.2 UPDATE	3
2.2.3 GET	3
2.2.4 REMOVE	3
2.3 Responses	4
2.4 Synchronization Policies.....	4
2.5 Server-Side Data Structure.....	4
2.6 Server Error Handling.....	4
2.7 Client Error Handling.....	4

1.0 Introduction

1.1 Purpose

The Server/Client Bibliography Protocol is a design protocol for a network of communication between a server and multiple client applications. This protocol outlines the specifications required for server and client applications to remain consistent.

1.2 Server

The server will be responsible for handling multiple incoming connection requests, storing any new bibliography additions, and updating any existing entries. The server will be able to respond to the message formats described, and will gracefully handle error conditions. The server will perform the necessary tasks, and respond with the requested data for any client with an established and persistent connection.

1.3 Client

The client will be responsible for providing a user-friendly interface for the creation of bibliography entries. The client will communicate with the server using the message formats described and will forward tasks to the server for completion. The client will handle any error conditions in user input. The client must establish a connection with the server to begin operations, and must maintain said connection for the entire duration of the user session.

2.0 Communication

2.1 Overview

The client and server will communicate via message requests from the client. The client will send a total of four types of messages through an established connection:

1. SUBMIT messages containing book descriptions
2. UPDATE messages containing an update to a particular book description
3. GET messages containing requests for a particular book
4. REMOVE messages containing requests to remove a book from the bibliography file

The server will respond to these four messages as follows:

1. SUBMIT – the server will add the entry to the bibliography if entry is unique
2. UPDATE – the server will update the specified existing bibliography entry
3. GET – the server will send the client a list of entries matching the request
4. REMOVE – the server will remove the specified entry from the bibliography list

The message type must always be declared first. Parameters within a message can be entered in any order and must be separated by a newline.

2.2 Message Formats

2.2.1 SUBMIT

Required: SUBMIT, ISBN

Optional: TITLE, AUTHOR, YEAR, and PUBLISHER

Example:

SUBMIT
ISBN 9783161484100
TITLE Modular Algorithms in Symbolic Summation and Symbolic Integration
AUTHOR Gerhard
PUBLISHER Mir

Client expects to add a new entry to the database

2.2.2 UPDATE

Required: UPDATE, ISBN

Optional: TITLE, AUTHOR, YEAR, and PUBLISHER

Example:

UPDATE
ISBN 9783161484100
YEAR 2004
PUBLISHER Springer

Client expects to update book entry in the database

2.2.3 GET

Required: GET, value(s)

Values: ALL, ISBN, TITLE, AUTHOR, YEAR, and PUBLISHER

Examples:

GET
TITLE Modular Algorithms in Symbolic Summation and Symbolic Integration
AUTHOR Gerhard

Client expects to receive records of all books in the bibliography file with this title and this author

GET
ALL

Client expects to receive all entries of all books in the bibliography file

2.2.4 REMOVE

Required: REMOVE, value(s)

Values: ALL, ISBN, TITLE, AUTHOR, YEAR, and PUBLISHER

Example:

REMOVE
AUTHOR Gerhard

Client requests removal of all books in the bibliography file with this author

2.3 Responses

For SUBMIT, UPDATE, and REMOVE messages, a confirmation response will be sent to the user if the action outlined in the message was completed successfully and a failure response if for any reason the actions outlined in the message could not be completed.

For a GET message, a listing of the matching books will be sent as a response in the following format, if a book does not have any data for a certain entry, it will be omitted:

```
ISBN 9783161484100
TITLE Modular Algorithms in Symbolic Summation and Symbolic Integration
AUTHOR Gerhard
PUBLISHER Mir
YEAR 2004
```

If no books are found that match the given message, a failure response will be sent.

2.4 Synchronization Policies

Books entered by the client via the SUBMIT message will be added to a thread safe list. The thread safe list is a part of the synchronized collections Java library, which provides safe concurrent access to a singular list of book elements. Each client will pull data from the same list, meaning that additions from one client will be visible to another.

2.5 Server-Side Data Structure

The server stores each book in a class called Book. This object represents all the information needed to represent the data for a single item.

Properties:

```
Public int isbn;
Public int year;
Public String title;
Public String author;
Public String publisher;
```

The books are then stored in a synchronized list which serves to ensure that the data is synchronized for all users connected to the server.

2.6 Server Error Handling

The server will verify that duplicate elements are unable to be added to the list. The server responds to the client with a string response, either indicating success or providing an error message.

2.7 Client Error Handling

The Client will ensure that the message contains a valid command before sending. This includes checking that the first line is one of SUBMIT, UPDATE, GET, or REMOVE and that all parameters are formatted correctly. In addition, the client checks that both SUBMIT, and UPDATE contain an ISBN as a parameter and that any ISBN given as a parameter is a valid ISBN by performing an ISBN-13 check digit calculation.