

SDSSDbBeanRemote Methods

Java Libraries Required:

```
commons-httpclient-2.0-rc2.jar
commons-logging.jar
jakarta-oro-2.0.5.jar
mysql-connector-java-2.0.14-bin.jar
```

The server (on which SDSSDbBeanRemote is running) IP (what it broadcasts as) must be set up with an SDSS Verification Server Record in order for SDSSRemotedbBean to work:

- Server IP must be set up for [SDSS Verification Server](#) (see Step #5 in [SDSS Verification Server Process](#)) access.

2003-04-17

```
userLoggedInNFO(source, userName, ip, infobase);
```

```
//
// infobaseFound will return "y" or "n";
//
```

```
infobaseFound = output[0];
errorMessage = output[1];
```

```
//
// If user is not already logged in, then Check for IP recognition
//
```

```
ipRecognizeNFO(source, iaAddress, infobase);
```

```
String found = output[0];
String subscription_dldir = output[1];
String subscription_start_url = output[2];
String subscription_login_url = output[3];
String subscription_code = output[4];
String sublogin = output[5];
String errorMessage = output[6];
```

```
String subloginURL = sublogin + "?subcode=" + subscription_code +
"&autologin=Y&filedownload=Y&sub_start=" + subscription_start_url +
"&sub_login=" + subscription_login_url;
```

```
//
// If found, then send to subloginURL to log this user in;
//
```

2003-07-14

Attached is modified SDSSDbBeanRemote which will allow you to list infobases/ebooks and/or subscriptions any particular user has access to.

The bean has 4 new methods:

1. **userSubsCount(String source code, String username)** : Returns an integer which contains the number of subscriptions a user has access to.
2. **userNFOsCount(String source code, String username)**: Returns an integer which contains the number of infobases a user has access to.
3. **userSubs(String source code, String username, int subscription count)**: Returns an Array[subscription count][3] containing the following:

Array[x][0] - Subscription Code
Array[x][1] - Subscription Start URL
Array[x][2] - Subscription Title

4. **userNFOs(String source code, String username, int infobase count)**: Returns an Array[infobase count][5] containing the following:

Array[x][0] - Infobase Name
Array[x][1] - Infobase Title
Array[x][2] - Subscription Code
Array[x][3] - Subscription Start URL
Array[x][4] - Subscription Title

The process to get the list (array) of subscriptions a particular username has would be to:

- a) Get the subscription count using `db.userSubsCount("C126", username);`
- b) Get the subscription array using `db.userSubs("C126", username, subscription count);`

The process to get the list (array) of infobases a particular username has is:

- a) Get the infobase count using `db.userSubsCount("C126", username);`
- b) Get the infobase array using `db.userSubs("C126", username, infobase count);`

You should test this immediately to make sure it is working properly. You'll need to copy this bean to your "classes" directory and re-start Resin.

2003-07-23

Attached is updated SDSSDbBeanRemote class file. It contains a new method:

verifyPassword(String source, String username, String password)

It returns "y" or "n" (lower case).

This method verifies that an SDSS customer account exists with the username and password you send.

Please test at your earliest convenience. It contains some fancy encryption stuff, but there shouldn't be a problem. I've tested it and it works in my environment.

2003-08-18

Attached is an updated SDSSDbBeanRemote class file. It contains two new methods:

getUserIPCount(String source, String ip)

It returns a count (an integer, i.e., "int count") of usernames that are IP recognized using the IP you send.

getUserIPList(String source, String ip, int count)

This returns an Array[username count] of usernames that are IP recognized using the IP you send. You must also send the count which is returned by getUserIPCount() above. The array is: Array[count].

2004-02-02

Here is the new SDSSDbBeanRemote bean with an additional method:

public static int testURL(String source, String url, int timeout)

This method will return the status code of the url (website) within the timeout period (in milliseconds) that you send. For example:

```
db.testURL('C126',http://gem.greenwood.com,8000)
```

will return an Integer containing the Status Code of the website (i.e., 200, 404, etc...). The request will timeout after 8 seconds. If it returns -1, that means the program could not get to the site at all. If it returns 0, then there is some internal program error.

2004-04-17

```
//  
// Replace Strings. Enter the entire String, the String value to look for,  
// and the String value to replace with.  
//  
// Returns "TheLine" with String values replaced;
```

```

//
replaceString(String TheLine, String LookFor, String ReplaceWith)

//
// Check to see if this IP address is logged into sddllog for access to the fileName;
//
// Returns "y" or "n";
//
checkSDDLLOG(String source, String cookieIP, String iaAddress0, String
iaAddress1, String iaAddress4, String iaAddress5, String fileName)

//
// Return File Path Information
//
// Returns:
//
//     output[0] = fileFound;
//     output[1] = path;
//     output[2] = fileType;
//     output[3] = contentType;
//     output[4] = hostpath;
//     output[5] = sourceIn;
//     output[6] = remote;
//     output[7] = remotePath;
//     output[8] = "";
//     output[9] = "";
//     output[10] = "";
//     output[11] = "";
//     output[12] = "";
//     output[13] = "";
//     output[14] = errorMessage;
//
returnFILEPath(String source, String fileName)

// Return File Login Information
//
// Returns:
//
//     output[0] = fileInfo;
//     output[1] = file_filename;
//     output[2] = file_title;
//     output[3] = file_description;
//     output[4] = file_source;
//     output[5] = src_name;
//     output[6] = src_url;
//     output[7] = src_email;

```

```
//      output[8] = sdsbub_code;
//      output[9] = sdsbub_title;
//      output[10] = sdsbub_desc;
//      output[11] = sdsbub_descurl;
//      output[12] = "";
//      output[13] = "";
//      output[14] = errorMessage;
//
returnFILEInfo(String source, String fileName)
```

Added – Dates Unknown

```
//
// Get the count of subscriptions that a particular infobase belongs to;
//
// Returns numeric count of subscriptions.
//
getSubCount(String source, String infobase)

//
// Get the subscriptions that a particular infobase belongs to;
//
// Returns list of subscriptions in this String [][] array format:
//
// Subscription Code
//      output[counter][0]
// Subscription Start URL
//      output[counter][1]
// Subscription Title
//      output[counter][2]
//
getSubList(String source, String infobase, int countLimit)
```

2004-05-11

```
//
// Verify if the username exists.
//
userNameExists(String source, String username)
```

It returns "y" or "n" (lower case).

This method verifies that an SDSS customer account exists with the username and password you send.