National Database for Autism Research

Global Unique Identifier (GUID)
Web Service User Manual

12/18/2008

Revision

Version	Date	Description
1.5.3	7/7/2008	The document provides an overview of how the GUIDWS system works and how to obtain GUIDs from the system
1.5.4	8/1/2008	• Updated "Special Instructions" to reflect that name suffixes • Added requirement of Java 1.5 to run the client • Added information on the Batch functionality • Removed reference to an obsolete document • Updated document format
1.5.5	10/10/2008	• Section "Running the client using the Graphical User Interface (GUI), For MS Windows:"was updated
2.0.2	12/18/2008	• Section "Running the client using the Graphical User Interface (GUI), For MS Windows:" was updated to reflect the latest changes in GUID interface

Table of Contents

Introduction

The National Database for Autism Research (NDAR) is dedicated to enhancing collaboration in autism research and clinical practice. The NDAR Global Unique Identifier Web Service (GUIDWS) system provides a platform for users, such as clinics and/or different institutes, etc., to share information of their subjects, e.g. phenotypic data, genomic data, with each other, but without exposing personally identifiable information (PII) of the subjects.

The NDAR GUIDWS application is fundamental to NDAR and allows data (e.g. imaging data, genomic data, and phenotypic data) collected from one subject at different sites to be associated with the same subject without exposing PII. This document provides an overview of how the GUIDWS system works and how to obtain GUIDs from the system.

GUIDWS system: How it works

Typically researchers collect PII (sometimes referred to as Private Health Information or PHI) of their subjects and store it into their respective local databases. The GUIDWS system consists of two components:

GUIDWS server – a single web service application for creating GUID, updating and merging duplicated subjects, and notifying the client sites of a change to the GUID of a subject.

GUIDWS clients – a web service client application that encrypts the PII information collected by a client site using a secure one-way hash function[1] to produce hash codes for each subject. The hash codes are associated with a randomly generated GUID that is used to identify a single subject.

When the client site, using the GUIDWS client application, requests a GUID for the chosen subject, the client application sends the hash codes associated with the subject to the NDAR GUIDWS server. The GUIDWS server processes the hash codes and returns a GUID back to the client application. PII data is never transmitted or stored in the GUIDWS server application. Communication between the GUIDWS web server occurs via HTTP & SOAP. See also Figure 1**.**

---

[1] A hash function is a reproducible method of turning some kind of data into a (relatively) small number that may serve as a digital "fingerprint" of the data. The algorithm "chops and mixes" (i.e., substitutes or transposes) the data to create such fingerprints. The fingerprints are called hash sums or hash codes. Hash codes are used for various purposes in information security applications.
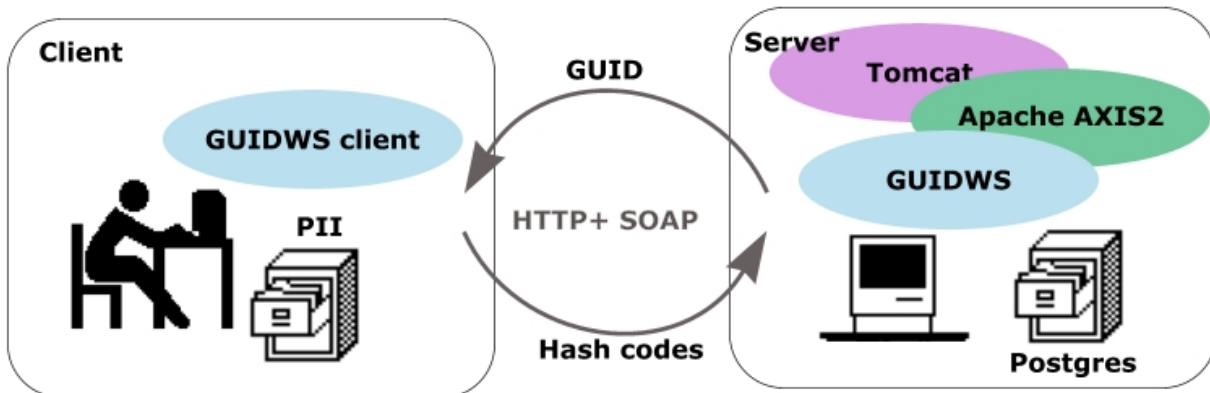
Figure 1. GUIDWS System Architecture

GUIDWS security

There are two layers to the GUIDWS security: application and network (IP restriction).

Security for GUIDWS is implemented within the application by using the Web Services Security (WS-Security), which allows a secure connection between a client and server by providing encryption & authentication support. Each client site is configured with the client's public-private key pairs that are used to identify a user to the server. This reduces the ability for unauthorized access.

The other security layer is the client IP restriction access. This layer maximizes security by preventing unauthorized client machines to the Web Service even if they have a valid client application with valid credentials. All sites that would like to gain access must submit the fully qualified domain name of the machine (*i.e. somehost.example.com*) and static IP address of each client machine that will run the GUID client application. A unique GUID client application will be created specifically configured for each client machine. If the client machine has a dynamic IP address and not a static IP address, then range of dynamic IP addresses with subnet mask is required. Each GUID client will only work on the client machine it was created for.

GUIDWS functions

The following are the service functions that compose the NDAR GUIDWS:

LOGIN – this function allows users to login to the GUIDWS. Users need to login first in order to perform any service functions within the application. Only authorized users are allowed to access the GUIDWS server.

GET – this function generates a valid GUID in a format that is used in the NDAR system provided the minimum required PII information is available. This module uses the hash function developed in NDAR to generate a unique GUID for each subject. See Section *Valid* versus invalid GUID for more information on valid GUIDs. For more information about GUID format, refer to Section *GUID composition*.

- **Subject Comparison** – this sub-module of the GUID Generator module compares the incoming subject hash codes to the existing subjects hash codes to determine if there are subjects in the database that match this subject according to rules configured as the system parameters.
- **Subject Merging** module activates when the hash codes of existing subjects in the GUID database match each other, this indicates those subjects are actually the same subject and need to be merged into one subject. The GUID of the subject that has the most complete PII information is then mapped to be the representative GUID for merged subject GUIDs.

GETBATCH – this function allows for the request of multiple subject GUIDs from a single call to the server. There are no limits to the number of GUIDs in a request. However, the batch functionality will process 50 subjects at a time with a 5 minute delay between each 50 subjects processed. The functionality takes as an input a CSV file that includes PII information (see Table 1. PII fields) and creates as an output with the list of GUID(s).

GETINVALIDGUID – this function generates an invalid GUID which has a different format from a valid GUID. An invalid GUID is a unique ID that has no basis on PII. No PII is required to get an invalid GUIID. This function should only be used if a user is unable to get a valid GUID using the GET function due to an insufficient PII for a subject. See Section *Valid versus invalid GUID* of this document for more information on invalid GUIDs. For more information about GUID format, refer to Section *GUID* composition.

TESTCONNECTION – this function tests the client connectivity to the GUID server. This function doesn't store any information to the GUID database and no input is needed.

DOESGUIDEXIST – this function allows users to check if a GUID exists in the database or not. Users need to provide a GUID, which can either be valid or invalid. If the submitted GUID exists in the database, the function returns true. Otherwise, it returns false.

UPDATE (future feature) – this function will be used to update the hash codes based on PII of a subject that has already been entered to the GUIDWS application. **This module is not yet implemented in the system.**

PII fields that are used to generate GUID in NDAR

The PII fields that are used to generate GUID are listed in Table 1 below. Note that the PII fields shall not change over the lifetime of the subject and shall be uniquely specified for the subject. Each PII field has an associated probability, thus, two different individuals drawn randomly from the subject population for the system share the same value for that field.

PII Field Name	Abbreviation used in the NDAR GUIDWS client	Required minimum for valid GUID
Complete legal given name of subject at birth	FIRSTNAME	Yes
Complete legal family name of subject at birth	LASTNAME	Yes
Complete additional legal name or names at birth	MIDDLENAME	Yes
Day of month of birth	DOB	Yes
Month of birth	MOB	Yes
Year of Birth	YOB	Yes
Physical sex of subject at birth [M/F]	SEX	Yes
Government Issued / National ID (For the United States, this is the Social Security Number)	GIID	RECOMMENDED
ISO 3166 STANDARD 2 letter abbreviation of the COUNTRY of origin of THE GIID. See APPENDIX 1 for a Listing of the ISO 3661 Country Abbreviations	GIIDCOUNTRY	RECOMMENDED
City of birth	COB	Yes
Mother's complete legal given name at her birth	MFIRSTNAME	
Mother's complete legal family name at her birth	MLASTNAME	
Father's complete legal given name at his birth	FFIRSTNAME	
Father's complete legal family name at his birth	FLASTNAME	
Mother's day of month of birth	MDOB	
Mother's month of birth	MMOB	
Father's day of month of birth	FDOB	
Father's month of birth	FMOB	
Indicator if subject has NO Middle Name AT BIRTH	SUBJECTHASNOMIDDLENAME	Yes

Table 1. PII fields

1   The "Last Name" field must contain the family name given at birth, prior to legal name change, or marriage. If there is any doubt as to the original legal name at birth, please locate the information on the birth certificate. Name suffixes such as "Jr.", "Sr.", "III", etc. should be ignored.

2   If "First Name" is compound, such as Anne Marie, or Jose-Luis, it may be unclear whether the second part of the compound is a first name or a middle name. In those cases, please use the first name as you would report it on other records, such as school transcripts, or credit card billing statements. If in doubt, please refer to the birth certificate.

3   If participant does not have a "Middle Name" (known not to have a middle name at birth), please enter "NOTAPPLICABLE". If in doubt, please refer to the birth certificate.

4   If the "City or Municipality of Birth" has undergone a name change during a participant's lifetime, participants must provide the city name at the time of their birth. Examples of this are Peking / Beijing, or Bombay / Mumbai. Again, if there is any doubt, please refer to the birth certificate.

Valid versus invalid GUID

The GUID server, using the GET function, will return a valid GUID only. A valid GUID is returned when all the required PII information is supplied to distinctly identify the subject. If the minimum required PII information is not supplied to create the hashcodes, then an error message is returned instead of a valid GUID. A valid GUID can be merged, and possibly changed through mapping (see Subject Merging for the GET method in Section *GUIDWS* functions), to another valid GUID based on the amount of PII supplied. Therefore, while all fields are not required for a valid GUID, it is strongly recommended that the maximum PII information be supplied to for a GUID to reduce the possibility of the valid GUID changing due to a merge. Table 1, above, denotes the required minimum PII inputs to ensure a valid GUID will be obtained.

An invalid GUID can be generated using the GUID server function, GETINVALIDGUID. An invalid GUID will always be unique. Invalid GUIDs are not compared to other subjects (including VALID) and are never merged since they have no PII associated with them (see Subject Comparison and Subject Merging for the GET method in Section *GUIDWS* functions). Invalid GUIDs are intended to be temporary IDs for subjects when sufficient PII for a valid GUID is not

available but a unique ID is desired. The advantage of using an invalid GUID as an ID within a study is that it can later be updated to get a new valid GUID while maintaining a mapping in NDAR for the invalid GUID to the valid GUID.

GUID composition

The GUID consists of a prefix, GUID pattern, and check character. The prefix chosen for this system is "NDAR". For the GUID a pattern is AANNNAA, where A represents alphabetic and N represents numerical characters. The check character can be numerical or alphabetic.

An example of GUID for a valid subject is

NDARCJ743PV3

Where,

Prefix –	NDAR	
Pattern –	CJ743PV	where the pattern is AANNNAA and A represents an alphabetic except letters I, O, Q, S are not used (because those letters are easier to confuse people with the digits) N represents a number both A and N are generated using a nondeterministic random number generator[2]
check character	3	

An example GUID for an invalid subject is

NDAR_INVZG542YHV

The format of the invalid is identical to that of a valid GUID with the exception that the prefix NDAR_INV is used instead of NDAR.

Obtaining the GUIDWS client

To obtain the NDAR GUIDWS client, first send a signed GUID software transfer agreement (STA), which can found at http://ndar.nih.gov/ndarpublicweb/datasubmission.go, to:

Matthew McAuliffe, Ph.D.
National Institutes of Health
Bld. 12a room 2041
South Drive
Bethesda, Maryland, 20892 -5624

---

[2] A variant of ISO7064 Mod 37, only difference is that I, O, Q, S are removed.

After the STA has been received an email will be sent to you requesting the following information

- Organization Name
- Client Computer's Fully Qualified Domain Name
- Client Computer's IP Address
    - If the IP address given is a dynamic IP address, please specify the range of IP addresses used as well as the subnet mask
- If you have any questions please contact the NDAR Help Desk at ndarhelp@nih.gov

Running the GUIDWS client standalone application

You can run the GUIDWS client using two ways: the command line or the Graphical User Interface (GUI) on both UNIX like system and MS Windows. The GUIDWS client requires Java JRE 1.5 installed on your PC.

Running the GUIDWS client using the command line

To run the GUIDWS client using the command line on UNIX and Windows, you can use guid_client.sh shell script and the guid_client.bat respectively. User can get a GUID for a subject by using a file. Below is an example that shows how to request a GUID for a new subject by using a file, text_pii.txt in this example, as an input for the GUIDWS client for the GET command:

For UNIX systems

GUID_CLIENT_DIR%   ./GUID_client.sh   -a   get   -u   username   -p   password   -d 'http://host:port/path/to/GuidService' -f test_pii.txt

For MS Windows

GUID_CLIENT_DIR:\   GUID_client.bat   -a   get   -u   username   -p   password   -d "http://host:port/path/to/GuidService" -f test_pii.txt

Program options

-a <action> -- The action which the GUID client should perform: get, getInvalidGuid, or testConnection.

-u <username> -- The username to use when logging into the GUID webservice.

-p <password> -- The password to use when logging into the GUID webservice.

-d <url> -- The URL to the GUID webservice, including the hostname, port number and service path.

-c <charset> -- The charset for the input, default is UTF-16.

-b <batch_file>-- The comma separated value file containing the ordering of the PII fields on the first line and PII data for one subject on each subsequent line, without limit on the total number of subjects.

-f <filename> -- The file is used to store the PIIs for a single subject. Each piece of PII should be on its own line, in the format '<FIELD><=><VALUE>

-g <guid> -- The GUID that is submitted to check if a GUID exists or not.
-h -- Display this message.

The table below shows the options that are required for each command:

Commands	Options required
GET	-a, -u, -p, -d, -f
GETINVALIDGUID	-a, -u, -p, -d
TESTCONNECTION	-a, -u, -p, -d
DOESGUIDEXIST	-a, -u, -p, -d, -g
GET(BATCH)	-a, -u, -p, -d, -b

Program Input - GET

GET takes an input file, as shown below, with the PII field key-value pairs (see Table 1. PII fields). Please note that the all required fields must be entered in order to get GUID. The value for SUBJECTHASNOMIDDLENAME must be either YES or NO. If subject is known not to have a middle name at birth, please enter YES; otherwise, enter NO.

```
COB=Bethesda
DOB=20
MOB=10
YOB=2001
FDOB=17
FMOB=3
MDOB=8
MMOB=6
FIRSTNAME=Alexandra
```

```
MIDDLENAME=
LASTNAME=Smith
FFIRSTNAME=Philips
FLASTNAME=Smith
MFIRSTNAME=Danna
MLASTNAME=White
SEX=M
GIID=000-00-0000
GIIDCOUNTRY=US
SUBJECTHASNOMIDDLENAME=yes
```

Program Input−GETBATCH

GETBATCH takes an input file in comma separated format. The first line of the file should contain the PII field names as defined in Table 1. There is also an additional "ID" field required that is used in the GUID output to map from the original subject to each generated GUID. The content of this field is left to the user's discretion for output mapping but could be a site's local ID for the subject. If no such ID exists, then a simple ascending integer should be used for each subject. Below is an excerpt from an example input file format. The complete example file, viewed in MS Excel, is displayed in Figure 2.

Example batch CSV with select PII input data :

```
ID,FIRSTNAME,MIDDLENAME,SUBJECTHASNOMIDDLENAME,LASTNAME,
MOB,DOB,YOB,COB,SEX
Local_id1,John,Quincy,NO,Public,1,1,2007,Washington,M
Local_id2,Jane,,YES,Smith,2,2,2006,Baltimore,F
```


Figure 2. The input CSV file

Note that if you are sending subsequent requests for multiple GUIDs, the required wait time between sending each GUID batch request containing 50 subjects is 30 seconds.

Program Output

The output from the various GUID functions appears in three output files below each time they run. The "timestamp" is a number indicating the actual date and time the client was executed. Limited output will also be generated at the command line, however, these three files in combination will have the complete information of an execution.

- output_guid_timestamp.txt
- output_log_timestamp.txt
- output_error_timestamp.txt

output_guid_timestamp.txt

This file contains the normal/expected return values of the GUID functions.

For example, for the GET, the output is:

```
NDARTL435KVE
```

For GETBATCH, each line of output is prefaced by the "ID" field (see the Program Input−GETBATCH section):

```
Local_id1 - NDARTL435KVE
Local_id2 - NDARNX545XZE
Local_id3 - NDARVR604PPX
```

output_log_timestamp.txt

This is a log file. It contains the details of an execution and is useful for debugging. A sample of the file's contents is shown below:

```
Jul 31, 2008 3:45:08 PM org.apache.axis2.deployment.DeploymentEngine prepareRepository
INFO:    No    services    directory    was    found    under    C:\projects\ndar-
guid_client\client\Java\trunk\classes\repository.
Jul 31, 2008 3:45:08 PM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: addressing-1.1
Jul 31, 2008 3:45:08 PM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: addressing-1.2
Jul 31, 2008 3:45:08 PM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: rampart-1.2
Reading batch file column headers.
Reading subject number 1 from batch file line 2
Reading subject number 2 from batch file line 3
Reading subject number 3 from batch file line 4
Reading subject number 4 from batch file line 6
```

```
Reading subject number 5 from batch file line 7
ERROR - One or more required fields are missing/need modification, and therefore a valid GUID can
not be generated. Please enter and/or modify values for the following fields and resubmit: -
Middle name is missing
Reading subject number 6 from batch file line 8
ERROR - One or more required fields are missing/need modification, and therefore a valid GUID can
not be generated. Please enter and/or modify values for the following fields and resubmit: - City of
Birth is missing
Completed reading batch file subjects.
Number of subjects read from batch file:  6
Number of subjects read from batch file without errors:    3
Number of subjects read from batch file with errors: 3
Sending getBatch request number 1
Sending getBatch request of size: 3
Logging into GUID webservice.
Jul 31, 2008 3:45:12 PM org.apache.xml.security.signature.Reference verify
INFO: Verification successful for URI "#Id-12082199"
Jul 31, 2008 3:45:12 PM org.apache.xml.security.signature.Reference verify
INFO: Verification successful for URI "#Timestamp-28653851"
Jul 31, 2008 3:45:13 PM org.apache.xml.security.signature.Reference verify
INFO: Verification successful for URI "#Id-11729694"
Jul 31, 2008 3:45:13 PM org.apache.xml.security.signature.Reference verify
INFO: Verification successful for URI "#Timestamp-19002963"
Completed getBatch request number 1
Logging out of GUID webservice.
Outputting errored subject information.
Finshed outputting errored subject information.
Finshed GUID batch request.
```

output_error_timestamp.txt

This file contains a list of errors that appeared while the program is running. The sample file is shown below:

```
ERROR - No ID field found in batch submission file (line: 4).
Local_id4 - ERROR - One or more required fields are missing/need modification, and therefore a valid
GUID can not be generated. Please enter and/or modify values for the following fields and resubmit: -
Middle name is missing
Local_id5 - ERROR - One or more required fields are missing/need modification, and therefore a valid
GUID can not be generated. Please enter and/or modify values for the following fields and resubmit: -
City of Birth is missing
```

Running the client using the Graphical User Interface (GUI)

To run the GUIDWS client using the Graphical User Interface (GUI) on UNIX and Windows, you can use guid_client_gui.sh shell script and the guid_client_gui.bat respectively. Below is the command to run the User Interface:

For UNIX systems:

[username@domain]~/temp/guid_java_client_DIR% ./GUID_client_gui.sh

For MS Windows:

guid_java_client_DIR:\ GUID_client_gui.bat

The NDAR GUID Client dialog box appears, see Figure 3.

Figure 3. NDAR GUID Client dialog box

Connection settings

To login to the GUIDWS, the user should enter the username and password (supplied with the client package) in the Connection Settings dialog box, see Figure 4. To open the Connection Settings dialog box, call the Settings > Connection Information menu.

Figure 4. Connection Settings dialog box

Testing connection

To verify if the user's computer is still connected to the NDAR GUID server, one should call the Settings > Test Connection menu (Figure 3). If the user's computer is connected to the NDAR GUID server, the following message appears (Figure 5):
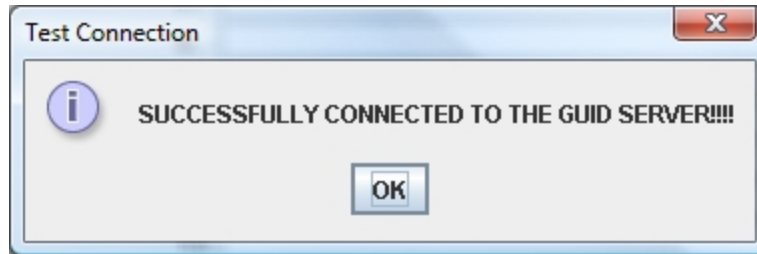
Figure 5.Test Connection window

Obtaining GUID

To obtain a GUID for a chosen subject, the user needs to enter subject information into the dialog box. This should be done twice − once in each panel of the dialog box. There are total 19 fields that should be filled out. The fields 3−11 (marked with asterisks) are **required** to obtain the GUID, see Figure 7. The required field names are listed below:

- Complete legal given (first) name of subject at birth*
- Complete legal family (last) name of subject at birth*
- Select YES if the subject DOES NOT HAVE a Middle Name, NO othervise*
- Complete Middle Name*
- Day of birth [1−31]*
- Month of birth [1−12]*
- Year of birth [####]*
- Physical sex of subject at birth [M/F]*
- Name of city/municipality in which subject was born*

To complete the NDAR GUID dialog box, and obtain the GUID the user should,

1   In the left panel of the dialog box:
- Enter the government issued or national ID number into the first field. Note that the list box in the second field becomes available;
- Select the country of issued ID from the list (Figure 6):
- Complete the required fields 3−11;
- Complete the optional fields 12−19, (Figure 7).

2   Enter the same information into the right panel of the dialog box, see Figure 7.

3   Press the Generate GUID button.

4   The GUID appears in the GUID filed at the bottom of the dialog box, see Figure 8.

The PII fields and the Generate GUID button will be grayed out only if enough PII is provided and the newly generated GUID is valid. The result appears as shown in Figure 8.

Note that spaces, hyphens and apostrophes can be used in the non-numeric GUID fields in the dialog box and are stripped silently by the application.

To get another GUID,

1 Press New;

2 Complete the dialog box for another subject;
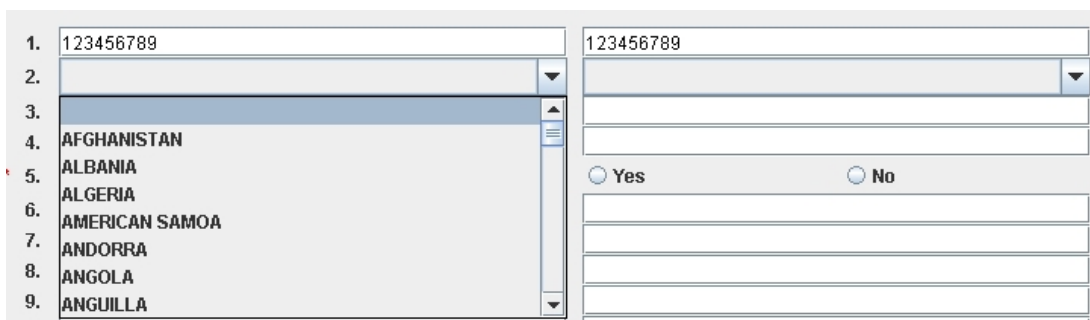
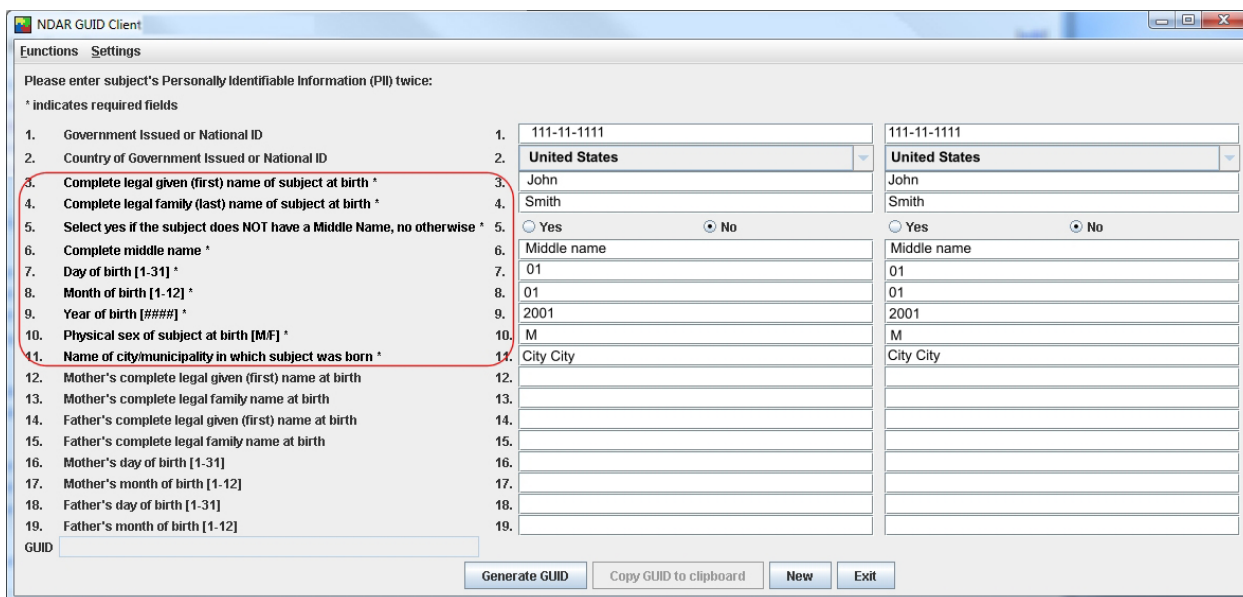**3** Press Generate GUID.

Figure 6. The list of countries

Figure 7. Completing the NDAR GUID client dialog box

Figure 8. Obtaining a GUID

To copy the GUID to the Clipboard

Once the valid GUID appears in the GUID field (see Figure 8), the user can copy it to the Clipboard by pressing the Copy GUID to Clipboard button. This allows the user to use the GUID in other applications.

What happens if the user did not provide ehough information?

The GUID will not be issued if the user did not provide the required subject information, for example, did not fill out all the required fields. In that case, the Error window appears listing the missed fields. The missed fields will also be highlighted in red in the dialog box. See Figure 9.
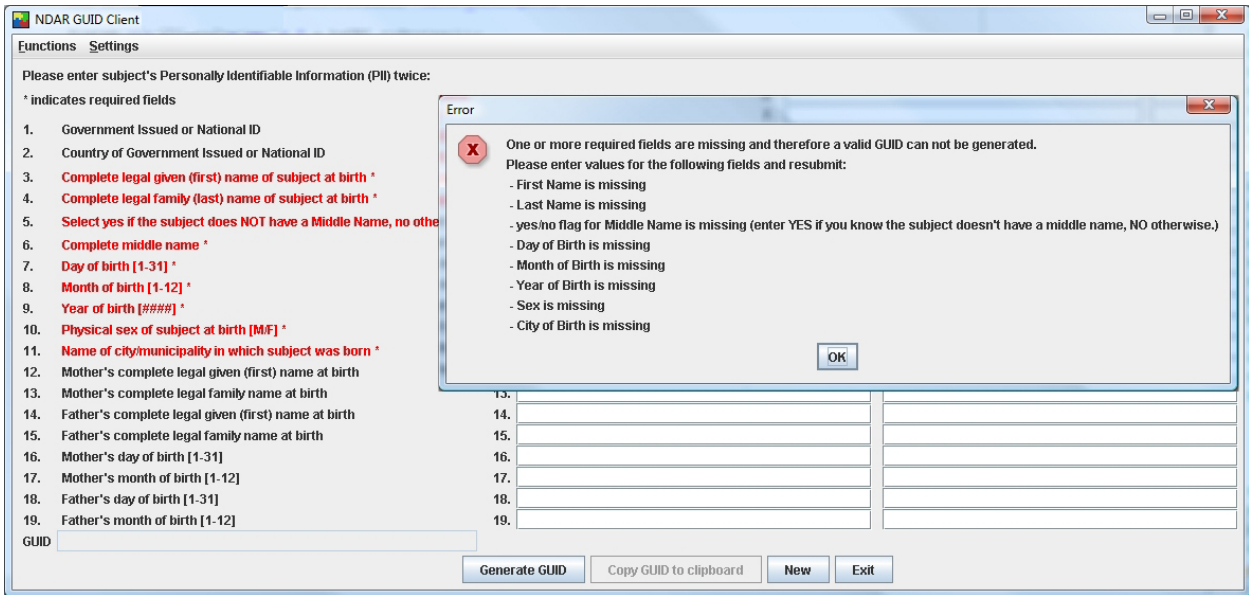
Figure 9. The Error message window

To obtain the GUID, the user should fill out all missed fields, and then press Generate GUID again.

Getting GUIDs for multiple subjects

To obtain GUIDs for multiple subjects,

1   Complete the subjects' information in the MS Excel CSV file, see Figure 2;

2   Run the NDAR GUID client for MS Windows;

3   Call the Functions > Get GUIDS for Multiple Subjects menu.
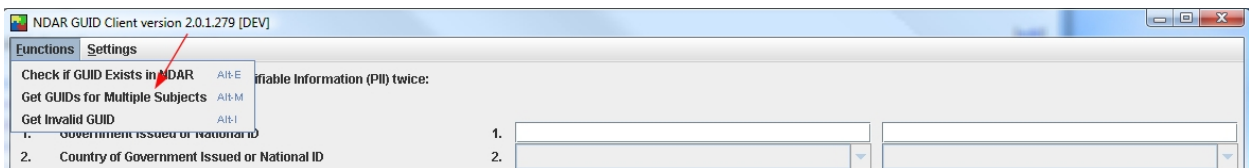
Figure 10. Calling the Get GUIDS for Multiple Subjects menu

The dialog box appears where the user is prompted to select the CSV file. Select the file and press OK. The Multiple Subject GUID Request Output dialog box appears displaying the following information:

- Name of the CSV file that has been processed;
- Name of the file where obtained GUIDS are stored. Note that the GUID file is stored in the same catalog where the source file is;

- List of "subject Local ID −GUID" for subjects that obtained NDAR GUIDs;
- List of "subject Local ID −Error(s)" for those subject records which information was processed with errors, and therefore, they did not obtain NDAR GUIDs. See Figure 11.

Press Close to close the dialog box. See Figure 11.
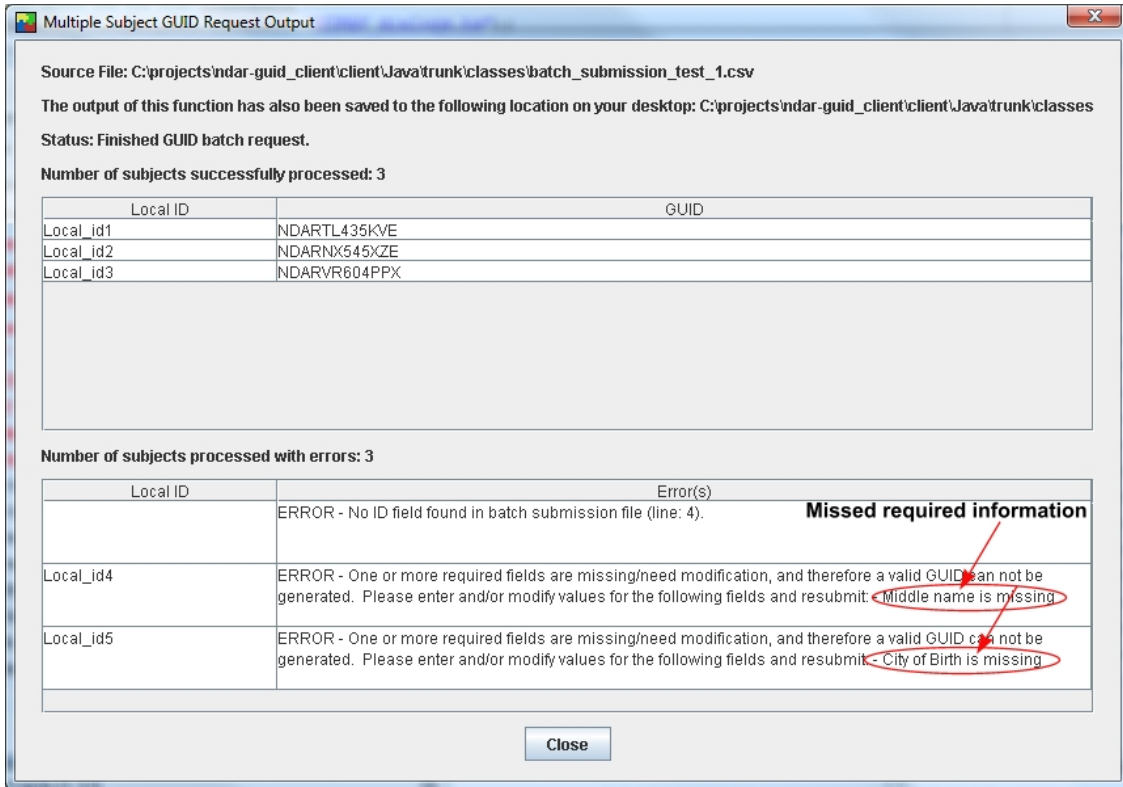
Figure 11. Multiple Subject GUID Request Output dialog box, note that the eror messages contain missed field names

Note that if you are sending subsequent requests for multiple GUIDs, the required wait time between sending each GUID request containing 50 subjects is 30 seconds.

Checking if GUID exists in NDAR

To check if a GUID y exists in NDAR, call the Functions > Check if GUID Exists in NDAR menu (Figure 10). The Check if GUID Exists in NDAR dialog box appears. Enter the GUID in the GUID filed, and press Check GUID. The result appears in the Result field. See Figure 12.
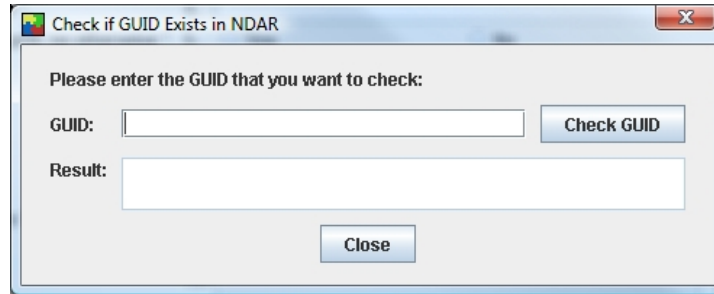
Figure 12. Check if GUID Exists in NDAR

Getting an invalid GUID

To get an invalid GUID, call the Functions > Get Invalid GUID menu. The Get Invalid GUID dialog box appears. Press OK. The software will generate an invalid GUID and display it in the dialog box.
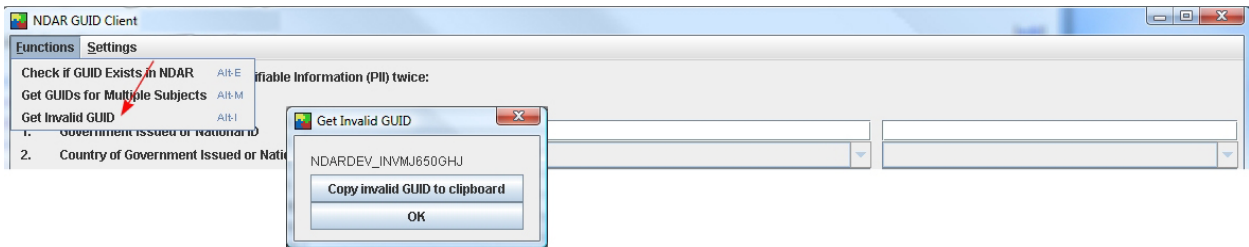
Figure 13. Getting an invalid GUID

To copy the invalid GUID to the Clipboard, use the Copy Invalid GUID to Clipboard button.

Running the GUIDWS client programmatically

An Application Programming Interface (API) for GUIDWS client functions is available (see the Javadocs). The API shows the functions that are used by the GUID client to communicate with the GUID web server. In order to run the above mentioned functions (refer to Section) programmatically, you will need to extend the AbstractGuidClient.java class (gov.nih.ndar.ws.guid.client.AbstractGuidClient). This class and other supporting classes are available from the GUID_client.jar library file supplied as part of the GUID client distribution.

Troubleshooting

Timestamp error

The time between the GUID server and a client computer should be synchronized within five minutes difference in order to access the server. Assuming that users are in the United States, users should set their local computer clocks to match the time in http://www.time.gov/ to avoid an error that can be caused by a time difference between the user's computer and the GUID server. In order to ensure that the user's clock remains correct, it is recommended that users utilize a clock synchronization tool, such as Network Time Protocol (NTP), http://www.ntp.org/, or some other local synchronization tools.

If the time is not synchronized within five minutes, the generated error looks like below:

```
Nov 26, 2007 9:40:15 AM org.apache.axis2.deployment.DeploymentEngine
prepareRepository
INFO: No services directory was found under
C:\projects\guid_client\Java\trunk\classes\repository.
Nov 26, 2007 9:40:15 AM org.apache.axis2.deployment.ModuleDeployerdeploy
INFO: Deploying module: addressing-1.1
Nov 26, 2007 9:40:15 AM org.apache.axis2.deployment.ModuleDeployerdeploy
INFO: Deploying module: addressing-1.2
Nov 26, 2007 9:40:16 AM org.apache.axis2.deployment.ModuleDeployerdeploy
INFO: Deploying module: rampart-1.2
Nov 26, 2007 9:40:16 AM gov.nih.ndar.ws.guid.client.PWCBHandler getPasswordFile
INFO: File Path for pwcbHandler properties is repository\conf\pwcbhandler.properties
org.apache.axis2.AxisFault: An error was discovered processing the <wsse:Security> header.
(WSSecurityEngine: Invalid timestamp The security semantics of message have expired)
    at org.apache.axis2.util.Utils.getInboundFaultFromMessageContext(Utils.java:434)
    at
org.apache.axis2.description.OutInAxisOperationClient.send(OutInAxisOperation.java:373)
    at
org.apache.axis2.description.OutInAxisOperationClient.execute(OutInAxisOperation.java:294)
    at org.apache.axis2.client.ServiceClient.sendReceive(ServiceClient.java:520)
    at org.apache.axis2.client.ServiceClient.sendReceive(ServiceClient.java:500)
    at gov.nih.ndar.ws.guid.client.AbstractGuidClient.login(AbstractGuidClient.java:200)
    at gov.nih.ndar.ws.guid.client.CmdLineGuidClient.main(CmdLineGuidClient.java:152)
```

Transport out error

A transport error will be observed if the entered command line command is incorrect. This error is often encountered when the entered command appears to be correct but contains special characters. If you encounter the error/exception below, verify you have transcribed the command correctly.

```
Feb 7, 2008 11:21:09 AM org.apache.axis2.deployment.DeploymentEngine prepareRepository
INFO: No services directory was found under D:\NDAR\GUID_client\repository.
Feb 7, 2008 11:21:09 AM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: addressing-1.1
Feb 7, 2008 11:21:09 AM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: addressing-1.2
Feb 7, 2008 11:21:12 AM org.apache.axis2.deployment.ModuleDeployer deploy
INFO: Deploying module: rampart-1.2
org.apache.axis2.AxisFault: Transport out has not been set
 at org.apache.axis2.engine.AxisEngine.send(AxisEngine.java:439)
 at org.apache.axis2.description.OutInAxisOperationClient.send(OutInAxisOperation.java:330)
 at
org.apache.axis2.description.OutInAxisOperationClient.execute(OutInAxisOperation.java:294)
 at org.apache.axis2.client.ServiceClient.sendReceive(ServiceClient.java:520)
 at org.apache.axis2.client.ServiceClient.sendReceive(ServiceClient.java:500)
 at gov.nih.ndar.ws.guid.client.AbstractGuidClient.login(AbstractGuidClient.java:365)
 at gov.nih.ndar.ws.guid.client.CmdLineGuidClient.main(CmdLineGuidClient.java:174)
```

Appendix 1. ISO 3166-1, English Country Names and two letter abbreviations

Country	Abbrv	Country	Abbrv	Country	Abbrv
A		C continued		FRENCH POLYNESIA	PF
AFGHANISTAN	AF	CANADA	CA	FRENCH SOUTHERN TERRITORIES	TF
ÅLAND ISLANDS	AX	CAPE VERDE	CV	G	
ALBANIA	AL	CAYMAN ISLANDS	KY	GABON	GA
ALGERIA	DZ	CENTRAL AFRICAN REPUBLIC	CF	GAMBIA	GM
AMERICAN SAMOA	AS	CHAD	TD	GEORGIA	GE
ANDORRA	AD	CHILE	CL	GERMANY	DE
ANGOLA	AO	CHINA	CN	GHANA	GH
ANGUILLA	AI	CHRISTMAS ISLAND	CX	GIBRALTAR	GI
ANTARCTICA	AQ	COCOS (KEELING) ISLANDS	CC	GREECE	GR
ANTIGUA AND BARBUDA	AG	COLOMBIA	CO	GREENLAND	GL
ARGENTINA	AR	COMOROS	KM	GRENADA	GD
ARMENIA	AM	CONGO	CG	GUADELOUPE	GP
ARUBA	AW	CONGO, DEM.REP.	CD	GUAM	GU
AUSTRALIA	AU	COOK ISLANDS	CK	GUATEMALA	GT
AUSTRIA	AT	COSTA RICA	CR	GUERNSEY	GG
AZERBAIJAN	AZ	CÔTE D'IVOIRE	CI	GUINEA	GN
B		CROATIA	HR	GUINEA-BISSAU	GW
BAHAMAS	BS	CUBA	CU	GUYANA	GY
BAHRAIN	BH	CYPRUS	CY	H	
BANGLADESH	BD	CZECH REPUBLIC	CZ	HAITI	HT
BARBADOS	BB	D		HEARD AND MCDONALD ISLANDS	HM
BELARUS	BY	DENMARK	DK	HOLY SEE (VATICAN CITY	VA

				STATE)	
BELGIUM	BE	DJIBOUTI	DJ	HONDURAS	HN
BELIZE	BZ	DOMINICA	DM	HONG KONG	HK
BENIN	BJ	DOMINICAN REP	DO	HUNGARY	HU
BERMUDA	BM	E		I	
BHUTAN	BT	ECUADOR	EC	ICELAND	IS
BOLIVIA	BO	EGYPT	EG	INDIA	IN
BOSNIA AND HERZEGOVINA	BA	EL SALVADOR	SV	INDONESIA	ID
BOTSWANA	BW	EQUATORIAL GUINEA	GQ	IRAN, ISLAMIC REPUBLIC OF	IR
BOUVET ISLAND	BV	ERITREA	ER	IRAQ	IQ
BRAZIL	BR	ESTONIA	EE	IRELAND	IE
BRITISH INDIAN OCEAN TERR	IO	ETHIOPIA	ET	ISLE OF MAN	IM
BRUNEI DARUSSALAM	BN	F		ISRAEL	IL
BULGARIA	BG	FALKLAND ISL. (MALVINAS)	FK	ITALY	IT
BURKINA FASO	BF	FAROE ISLANDS	FO	J	
BURUNDI	BI	FIJI	FJ	JAMAICA	JM
C		FINLAND	FI	JAPAN	JP
CAMBODIA	KH	FRANCE	FR	JERSEY	JE
CAMEROON	CM	FRENCH GUIANA	GF	JORDAN	JO
K		N		SAINT MARTIN	MF
KAZAKHSTAN	KZ	NAMIBIA	NA	SAINT PIERRE AND MIQUELON	PM
KENYA	KE	NAURU	NR	SAINT VINCENT AND THE GRENADINES	VC
KIRIBATI	KI	NEPAL	NP	SAMOA	PM
KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF	KP	NETHERLANDS	NL	SAN MARINO	VC
KOREA, REPUBLIC OF	KR	NETHERLANDS ANTILLES	AN	SAO TOME AND PRINCIPE	WS

KUWAIT	KW	NEW CALEDONIA	NC	SAUDI ARABIA	SM
KYRGYZSTAN	KG	NEW ZEALAND	NZ	SENEGAL	ST
L		NICARAGUA	NI	SERBIA	SA
LAO PEOPLE'S DEM. REP.	LA	NIGER	NE	SEYCHELLES	SN
LATVIA	LV	NIGERIA	NG	SIERRA LEONE	RS
LEBANON	LB	NIUE	NU	SINGAPORE	SC
LESOTHO	LS	NORFOLK ISLAND	NF	SLOVAKIA	SL
LIBERIA	LR	NORTH.MARIANA ISLANDS	MP	SLOVENIA	SG
LIBYAN ARAB JAMAHIRIYA	LY	NORWAY	NO	SOLOMON ISLANDS	SK
LIECHTENSTEIN	LI	O		SOMALIA	SI
LITHUANIA	LT	OMAN	OM	SOUTH AFRICA	SB
LUXEMBOURG	LU	P		SOUTH GEORGIA SANDWICH ISLANDS	SO
M		PAKISTAN	PK	SPAIN	ZA
MACAO	MO	PALAU	PW	SRI LANKA	GS
MACEDONIA	MK	PALESTINIAN TERRITORY, OCC.	PS	SUDAN	ES
MADAGASCAR	MG	PANAMA	PA	SURINAME	LK
MALAWI	MW	PAPUA NEW GUINEA	PG	SVALBARD AND JAN MAYEN	SD
MALAYSIA	MY	PARAGUAY	PY	SWAZILAND	SR
MALDIVES	MV	PERU	PE	SWEDEN	SJ
MALI	ML	PHILIPPINES	PH	SWITZERLAND	SZ
MALTA	MT	PITCAIRN	PN	SYRIAN ARAB REPUBLIC	SE
MARSHALL ISLANDS	MH	POLAND	PL	T	CH
MARTINIQUE	MQ	PORTUGAL	PT	TAIWAN, PROVINCE OF CHINA	SY
MAURITANIA	MR	PUERTO RICO	PR	TAJIKISTAN	
MAURITIUS	MU	Q		TANZANIA, UNITED REPUBLIC OF	TW
MAYOTTE	YT	QATAR	QA	THAILAND	TJ

MEXICO	MX	R		TIMOR-LESTE	TZ
MICRONESIA	FM	RÉUNION	RE	TOGO	TH
MOLDOVA	MD	ROMANIA	RO	TOKELAU	TL
MONACO	MC	RUSSIAN FEDERATION	RU	TONGA	TG
MONGOLIA	MN	RWANDA	RW	TRINIDAD AND TOBAGO	TK
MONTENEGRO	ME	S		TUNISIA	TO
MONTSERRAT	MS	SAINT BARTHÉLEMY	BL	TURKEY	TT
MOROCCO	MA	SAINT HELENA	SH	TURKMENISTAN	TN
MOZAMBIQUE	MZ	SAINT KITTS AND NEVIS	KN	TURKS AND CAICOS ISLANDS	TC
MYANMAR	MM	SAINT LUCIA	LC	TUVALU	TV
U		V		W	
UGANDA	UG	VANUATU	VU	WALLIS AND FUTUNA	WF
UKRAINE	UA	VATICAN CITY STATE	VA	WESTERN SAHARA	EH
UNITED ARAB EMIRATES	AE	VENEZUELA	VE	Y	
UNITED KINGDOM	GB	VIET NAM	VN	YEMEN	YE
UNITED STATES	US	VIRGIN ISLANDS, UK	VG	Z	
UNITED STATES MINOR OUTLYING ISLANDS	UM	VIRGIN ISLANDS, U.S.	VI	ZAMBIA	ZM
URUGUAY	UY			ZIMBABWE	ZW
UZBEKISTAN	UZ				