BACKGROUND

One of the issues which surround the use of Audit Exchange (AX) is its inability to query users once the project has been initiated. While one can program questions into the analytic section of AX, these questions must be asked before the data can be queried. AX currently does not allow for interactive scripting.  
  
The EXECUTE command, however, changes this. Using the EXECUTE command, the ACL scriptwriter can create interactive scripts that can be utilized after the analytic section.

METHODOLOGY

The way the process works is that Audit Exchange creates one or more data files. These data files are exported to a specific folder. AX then uses the EXECUTE command to initiate a bat file. This bat file kicks off a desktop version of ACL, a second ACL project, and a specific script.

The second ACL project loads the data files that were created by AX. It then uses that data to perform dialog boxes. After obtaining the user input, the second project then creates a new output file. Audit Exchange then loads the file, reads the results and assigns new variables and acts upon the user input.

PROOF OF CONCEPT

To demonstrate this functionality, I have created three separate dialog boxes.  
  
The first dialog box will ask the user to confirm the various data paths. These values are initially defined in the script AA01. Replace the pathway provided for Slave 2 with your name. This script will also define variables identifying the version of ACL being used in the secondary project.

The second dialog box asks the user to enter a table name. Since there is no safe way to get a complete list of tables, this value has to be entered manually.

After entering the table name, a third dialog box is then loaded. This dialog box presents a list of the fields from the table selected. ACL will then EXPORT the data from the field selected, the table name selected, the field name selected, the value entered for Slave 2, and the ACL version of the second project to a table called “Proof of Concept.”

While I do not have Audit Exchange, I was able to test this using version 10 as my stand in for Audit Exchange and a second instance of ACL version 9.1 for the desktop.

SET UP

1. I have attached a zipped file that contains 11 separate scripts. The files that begin with a “W” should be placed in a folder for bat files. The other scripts should be copied into Audit Exchange (or into an ACL 10 project.) In the Audit Exchange file create a table called “EXTRACT\_TABLE”. The EXTRACT\_TABLE should be a file with a single record. The contents of that record are immaterial as the file will be used to EXTRACT values that are captured by variables.
2. Create an ACL desktop project. In that project create a table called “EXTRACT\_TABLE”. The EXTRACT\_TABLE should be a file with a single record. The contents of that record are immaterial as the file will be used to EXTRACT values that are captured by variables.
3. In the SCRIPT VA01\_demo\_dataonly, you need to change the variables that start with v\_path.
   1. V\_path\_data should be the pathway to a folder that can be used for the storage of data files.
   2. V\_path\_batfile should be the pathway to the folder where the “W” series of bat files are saved.
4. In the script WA01, WB01, WB02 change the path way for the data file in the EXPORT to the same value used in v\_path\_data
5. In the script vA01 change the path way for the batfiles.
6. In the three scripts with the names that end with “ACL\_startup” change:
   1. The pathway for the “ACLwin.exe” to the pathway where the desktop version of ACL resides.
   2. The Pathway and project name of the ACL project created in step 2.
   3. The pathway to where the bat files are saved.

In every script, I have replaced the values that I used when writing the project with notes indicating what value needs to be replaced and where. To find these places, search for a { or } and it will take you to the values that must be changed on each script.