

Synergistic Office Solutions, Inc.



Topic: Connecting to the SOS database with MS Access
Document ID: # 406
Product: All
Date: Sept 18, 2006
Author: Seth Krieger

To use any non-SOS program to read SOS data, you must have a user account that has been set with the “Grant read-only access from third-party products” option set. In the SOS Admin module, check the user account you plan to use and make sure that this option has been set for that user ID.

1. Open Access (these instructions are based on Access2003. Other versions may differ somewhat.)
2. Select **File > New**
3. In right pane select “Blank database...”
4. Browse to a suitable folder and give the new Access database a name, such as “SOS”
5. Click **File > Get External Data > Link Tables**. By using “Link Tables”, rather than “Import” you will always have up-to-date data when you run your Access reports or queries. If you import the data, you will only have the data as it is at the time you do the import!
6. At the bottom of the selection window, drop the list for “Files of type” and select “ODBC Databases”. You will probably have to scroll down in the list.
7. In the next window, called “Select Data Source”, click the **Machine Data Source** tab and select “SOSDATA”.
8. An ODBC window will probably appear. Enter the SOS user ID and password that you have decided to use for this purpose and click OK.
9. A list of tables and views in the SOS database will appear. Scroll down to the ones that start with “SOS.” and tag those that you will be using. If you are not sure, the following assortment of pre-programmed views will probably cover most of the bases. Views represent data spread across several tables, but the links between the tables have already been done for you, making reporting much easier. Resist the temptation to select every table and view in the list that Access displays or you will have to click “OK” hundreds of times!
10. When you OK your selections you will be asked to provide a “unique record identifier” for each one. You can ignore the warning and just click “OK”. The ID you are using does not have the rights to add or modify anything to the SOS database so there is no opportunity for data corruption.

Name of View	Description
rv_appts	Appointments view
rv_credits	Credit transaction header information
rv_creditsplits	Credit transactions at the split level
rv_charges	Charge transactions at the split level
rv_journal	Transaction header info for both charges and credits
rv_patients	Patient demographics and other data, including UD fields
rv_policies	Patient insurance policy information
rv_ptpayors	Patient payor information, including mailing address and aging

Another way to select the tables and/or views you need is to find an existing query from the Query Samples document (<http://www.sosoft.com/fod/doc403-sosqueries.pdf>) or from the many queries that appear in message exchanges in the user group (<http://groups.google.com/group/sosoft>) that is similar to what you would like. The tables and/or views used by the query are listed in the FROM clause. Here is an example, with the tables underlined:

```
SELECT  
trim(refname + ', ' + b.firstname) as "RefName",  
b.address1,
```

Synergistic Office Solutions, Inc. 352/242-9100 Fax 352/242-9104
<http://www.sosoft.com>

```

b.address2,
b.city,
b.zip,
b.phone,
b. Defaultid,
e.payorname,
d.providerid,
d.secondaryid,
COUNT(distinct a.ptnum) AS "PtCount"
FROM patients a
JOIN ptcsu c ON c.ptnum = a.ptnum
JOIN refsrcs b ON b.refsrcnum = c.refsrcnum
LEFT OUTER JOIN refpayorids d ON b.refsrcnum = d.refsrcnum
LEFT OUTER JOIN payors e ON e.payorname = d.payorname
WHERE
a.flag=0
AND a.intakedate > '1980-10-01'
AND a.licnum='101'
AND payorname IN (<list payornumbers here>)
GROUP BY refname,b.address1,
b.address2,b.city,b.zip,b.phone,b.defaultid,e.payorname,d.providerid,d.secondaryid
ORDER BY refname

```

The query also tells you how these tables and views are joined to one another, something you will have to specify in Access when you design a report or query that uses more than one table or view. In the example above, each table of view is assigned a shorthand alias, which is used instead of the longer name everywhere else in the query that the table or view name would normally be required. So ...

```

FROM patients a
JOIN ptcsu c ON c.ptnum = a.ptnum

```

... tells us that the alias for the patients table is “a”, the alias for the ptcsu table is “c” and that the two tables are linked together from the value called “ptnum” in “c” to the value called “ptnum” in “a”.

For more information about the SOS database see the following resources:

1. The last chapter of SOSTECH.PDF, which you will find in your SOS folder.
2. <http://www.sosoft.com/files/downloads/sosdpics.pdf> - relationship diagrams
3. <http://www.sosoft.com/files/downloads/sosddct.pdf> - annotated data dictionary
4. <http://www.sosoft.com/files/downloads/sosviews.pdf>