

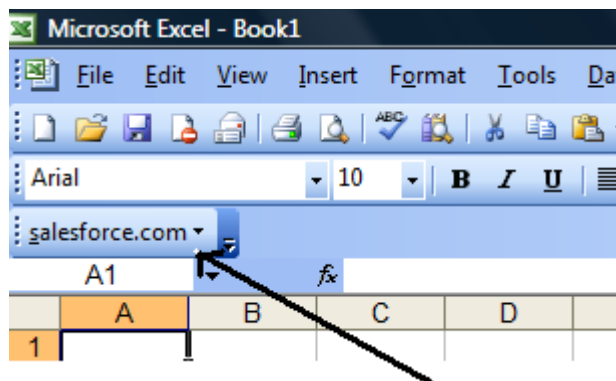
SalesForce.com makes it pretty simple to create most of the reports you need to manage your sales pipeline. You can create reports for 'My Teams Opportunities', summarized by account, opportunity owner, sales stage, etc. But in most cases, you have to choose between seeing the summary information and the detail that makes up the total. In my experience, it's still pretty common to run the report, export to Excel, and then have an analyst create the different views – usually pivot tables – required to run the business. That typically means you get the reports with some delay, and on somebody else's schedule.

SalesForce provides a great utility, Connect for Office, that reduces the steps required and forms the basis for self-service reporting. I'll focus first on getting Connect for Office up and running, and how it can simplify your reporting. Later, we'll discuss how to dynamically filter existing reports. Then we'll put these ideas together to create a self-serve Excel report to meet the needs of different users across your organization. Finally, we'll use Excel's hyperlink function to provide links from your Excel reports back to Salesforce data, much as if you were clicking on the links within Salesforce reports.

When you log in to Salesforce.com, you may have seen the Setup link near the top right. If you click on that, then expand the Desktop Integration list on the left-side navigation bar, you'll see 'Connect for Office'. Follow the steps provided to download and install the package. (Alternately, you can download the installation file directly from <https://na3.salesforce.com/setup/sforce/office/SForceOffice.exe>; save then run the file).

But what does that do? Fire up Excel to see!

Once you've installed Connect for Office, and open Excel, you should see a salesforce.com toolbar :



Click on the salesforce.com toolbar, then select Log In. You're prompted to enter your Salesforce.com user id and password. (The user id will be remembered so you don't have to reenter it each time). If you are not able to log in you may have to reset your security token (log in to Salesforce via your browser, select Setup > My Personal Information > Reset My Security Token) or you may need to append your security token to your password.

Once you've logged in, you can then click on the salesforce.com toolbar and select 'Import a Report.' After a few moments, you'll see the list of all reports your user permissions allow. Select one and

Import. Connect for Office will retrieve the report detail for the report you've selected. This works much like opening the report in Salesforce, then choosing to Export Details.

So far, this just seems like you've chosen to 'pull' (import) the report data rather than 'push' (export) it. What's the difference? Two things come to mind. We'll demonstrate the first. Save and close the Excel file, then open it back up. First make sure you're still logged in: click on the salesforce.com toolbar; if 'Log In' is grayed out, you're already logged in; if not, click 'Log In'). Then right-click in your report data and select Refresh.

Okay, so the report didn't change at all since you ran it two minutes ago. But, the Refresh caused Salesforce to go pull the report data anew. So if the underlying data *had* changed, the report would be updated to match. And now that the Excel file contains that reference to the underlying Salesforce.com report, the refresh is simpler than logging in via the browser, finding the report and exporting details.

A second advantage becomes readily apparent if you've worked with Excel macros. Turn on the macro recorder (Tools > Macro > Record New Macro or, in Excel 2007, Developer > Record Macro). Right-click in your report data and Refresh, then stop recording. Now open up the Visual Basic Editor (alt +F11), and you can see the simple code to refresh the report. While you're there, enter the following in the immediate window: `print activesheet.querytables(1).connection`. You'll see that the connection string references the Salesforce.com report via a URL, and the URL includes the 15-digit Salesforce report id.

In later papers, we'll dive into the SFDC report URLs in more detail, and show how they can be used to dynamically filter existing reports, or even pull from a different report!

For now let's address some issues you may have run into in using the toolbar to log in and import a report. First, you may have to set your firewall to allow Excel to access the internet. Second, your SFDC administrator may have denied you API access or permission run and/or export reports. To check this, log in via your browser. Select Setup > Personal Information. Click on your Profile name. Scroll down and check whether API Enabled and Export Reports are selected. If either is not selected, check with your administrator to see if this can be changed.

While we're sweating the initial details, this is a good time to check on your macro security in Excel. If you want to develop your own macros, you should **at least** use Microsoft's selfcert.exe to create a digital certificate to sign your macros. Then, in versions prior to Office 2007, you can set the macro security to Medium (Tools > Macro > Security). With Office 2007, you'll have to purchase a certificate from a trusted certificate authority and then set the security to 'Disable all macros except digitally signed macros' (Office Button > Excel Options > Trust Center > Trust Center Settings > Macro Settings). (The alternative under 2007 is to enable all macros, which I could never recommend.)

You should now be all set... SFDC permissions letting you run the API and export reports; the firewall letting Excel talk to the internet; macro permissions allowing you to develop and run macros without permitting unsigned macros. Next time we'll dynamically filter SFDC reports so that different users can see just what they need to know without having administrators create umpteen versions of the same report.