

Synchronizing Calendars and Address Books Between Macs for FREE

or

Making Any Mac into a Calendar and Contacts Server

One feature which is sorely missed by owners of multiple Macs is the ability to easily share their calendars and contacts between their computers. There are third-party software solutions, such as Mark/Space's *The Missing Sync* - but they cost money. There are also web services that offer calendars and contacts synchronization, but they either cost (such as MobileMe), or they are not completely dependable and compatible (such as Plaxo and Google). In the case of Google, the company is the first to admit that its calendar solution is not completely compatible with iCal, and worse yet - its contacts solution does not have individual first, middle and last name fields - which renders it useless for the purposes of an accurate synchronization between Macs.

Apple has another solution for you, except MobileMe - and that's to get a Mac OS X Server. Among other things, the server edition of OS X has a built-in calendar and contacts server. As of the Snow Leopard version, both are based on the newest standards - CalDAV for calendars, and CardDAV for contacts. All you need to do from your client computer is to subscribe to the server from within iCal and Address Book, and you are all set for bi-directional sync.

Fortunately enough, these portions of OS X Server are also open-source, and are known as the [Darwin Calendar and Contacts Server](#). The two servers are still separate projects (written in Python and based on the Twisted framework), and are very much under development. The CardDAV server was only released in early 2010. Consequently, a lot of the installation, configuration and operation is only half-baked, and it takes some work to get to the point of the famous Apple "It just works" (well, they do want you to buy OS X Server). Nevertheless, once you have everything up and running, this is exactly what you get - the most dependable way to sync your calendars and contacts, within a local network or over the internet, for free. You can designate any Mac on your network to be the server, and it will serve itself as well.

1. Setting Up the Calendar Server

My tutorial is based on the one found [here](#) (a Google translation from German is available

[here](#)).

First of all, the version of Darwin CalDAV under development now is version 3. Since it is actively developed, many times a check-out snapshot simply does not work. We will use, therefore, version 2.3

Using terminal, Create a new folder in your home directory, and cd into it:

```
mkdir calendar  
cd calendar
```

Now we will download the code from Apple:

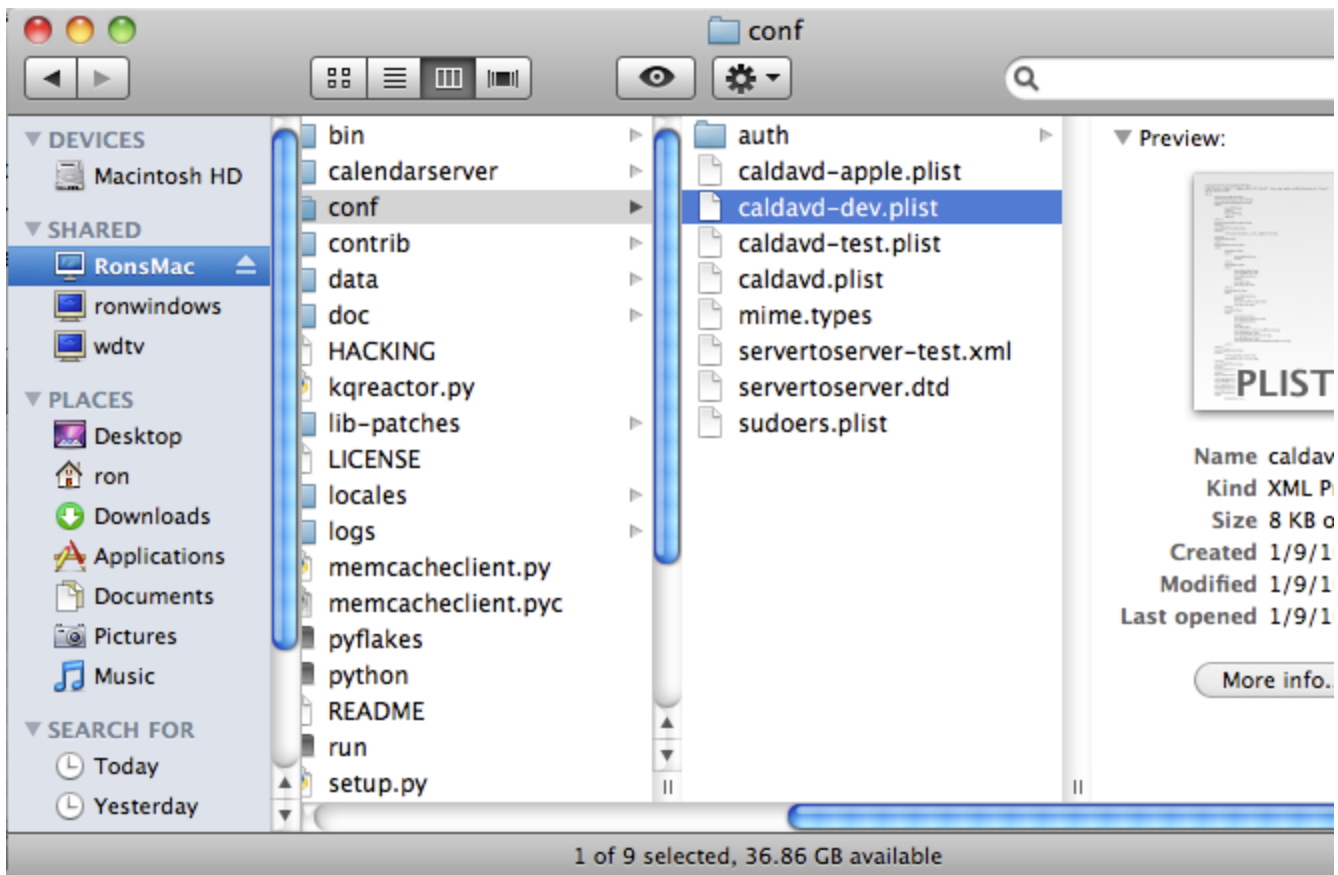
```
svn checkout http://svn.calendarserver.org/repository/calendarserver/  
CalendarServer/tags/release/CalendarServer-2.3 CalendarServer
```

This will take a few minutes, and at the end will display the revision number which was downloaded. Once it is done, cd into the subfolder `CalendarServer`, and run the command to perform initial installation:

```
cd CalendarServer  
./run -s
```

This can take up to 10 minutes. The computer will download the necessary extensions to Python, as well as the Twisted framework, and install them alongside `CalendarServer` in your calendar folder. When you see "Using python as Python" - the installation is done.

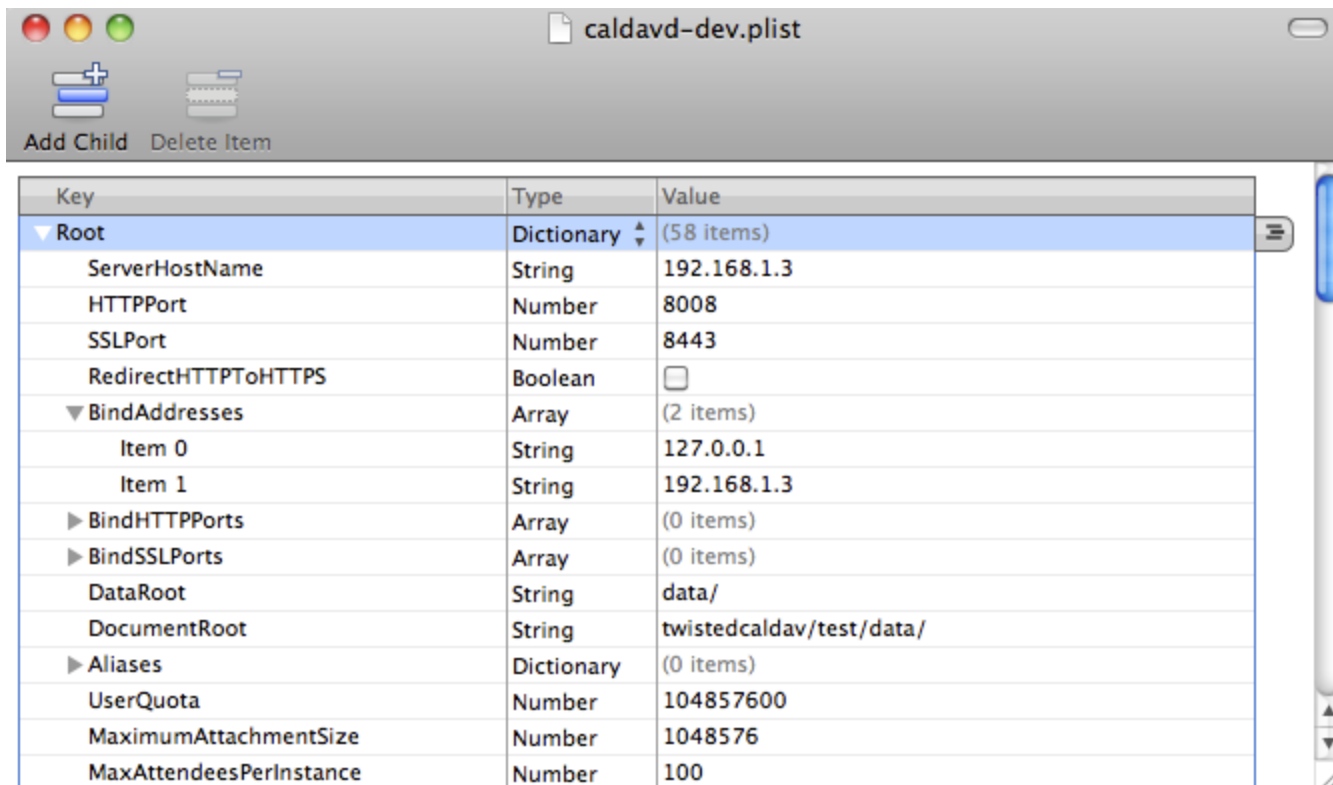
Now it is time to configure the server. The basic configuration files are stored in the folder `CalendarServer/conf`. The server first accesses a file named `caldavd-dev.plist`, but this file does not yet exist. Using the Finder, duplicate the file `caldavd-test.plist` and then rename it:



Double-clicking the file `caldav-dev.plist` will open it in the property list editor. We need to modify several properties:

1. The string `ServerHostName`. If you are going to use the server only on your home network, you can give it the computer's internal IP address, such as `192.168.1.3`. This will also work if you plan to connect to this computer over the internet by using a VPN. If you plan to connect to the computer using a dynamic DNS service, such as DynDNS or NoIP, you need to enter your assigned address here (make sure the router forwards the ports indicated below in the file to the proper computer). Finally, if you plan to use an SSH tunnel to connect to this computer, then put the local host's address: `127.0.0.1`.
2. The array `BindAddresses`. Click on the little arrow to the left of the array, so it points down. Now click on the button to the right of the line to add items to the array. We need to create two strings: one for the localhost (`127.0.0.1`) and one for the computer's address on the local network (in this example it is `192.168.1.3`). You may also need to add your dynamic DNS address as a third item, if you choose to use one.

The top of the file should look similar to this:



Key	Type	Value
▼ Root	Dictionary (58 items)	
ServerHostName	String	192.168.1.3
HTTPPort	Number	8008
SSLPort	Number	8443
RedirectHTTPToHTTPS	Boolean	<input type="checkbox"/>
▼ BindAddresses	Array (2 items)	
Item 0	String	127.0.0.1
Item 1	String	192.168.1.3
▶ BindHTTPPorts	Array (0 items)	
▶ BindSSLPorts	Array (0 items)	
DataRoot	String	data/
DocumentRoot	String	twistedcaldav/test/data/
▶ Aliases	Dictionary (0 items)	
UserQuota	Number	104857600
MaximumAttachmentSize	Number	1048576
MaxAttendeesPerInstance	Number	100

Before we can start testing our server, we need to add user accounts. The proper configuration file is located in the folder `CalendarServer/conf/auth`, and it is named `accounts-test.xml`. First duplicate it in Finder so that you have a backup of the original, and then open it in TextEdit.

It took me a while to figure out which is the best, most stable way of handling users and groups in this file. This is what I concluded: for each set of calendars (meaning a set of "Home", "Family" and "Work" etc. that an average person uses) we need to create a separate user and a separate group. Whoever connects with the credentials of this user later on will be presented with this group of calendars. Below you'll see the code you need, starting with the line `<accounts realm="Test Realm">`, to create two individual sets and one that can be used as a shared set (meaning, all users will login with the username "shared" and the proper password). Please note, that at this time you should NOT add a `guid` line for any of the users. We will do that later. You will need to substitute all the upper-case `uid` names and passwords with lower-case names and suitable passwords (the passwords can be upper- or lower-case, of course).

```
<accounts realm="Test Realm">
```

```
<user>
```

```
<uid>admin</uid>
```

```
<guid>admin</guid>
```

```
<password>admin</password>
```

```
<name>Super User</name>
```

```
<first-name>Super</first-name>
<last-name>User</last-name>
</user>
```

```
<user>
<uid>USER1</uid>
<password>PASSWORD1</password>
<name>USERNAME1</name>
<first-name>FIRST NAME1</first-name>
<last-name>LAST NAME1</last-name>
</user>
```

```
<user>
<uid>USER2</uid>
<password>PASSWORD2</password>
<name>USERNAME2</name>
<first-name>FIRST NAME2</first-name>
<last-name>LAST NAME2</last-name>
</user>
```

```
<user>
<uid>shared</uid>
<password>PASSWORD3</password>
<name>Shared Calendars</name>
<first-name>Shared</first-name>
<last-name>Calendars</last-name>
</user>
```

```
<group>
<uid>user1group</uid>
<password>PASSWORD4</password>
<name>USER1's GROUP</name>
<members>
<member type="users">USER1</member>
</members>
</group>
```

```
<group>
<uid>user2group</uid>
<password>PASSWORD5</password>
<name>USER2's GROUP</name>
<members>
<member type="users">USER2</member>
</members>
</group>
```

```
<group>
<uid>shared</uid>
```

```
<password>PASSWORD6</password>
<name>Shared Calendars Group</name>
<members>
<member type="users">shared</member>
</members>
</group>

</accounts>
```

Note that the first user in the code is the Administrator. Do not delete this user, and make sure you change the password.

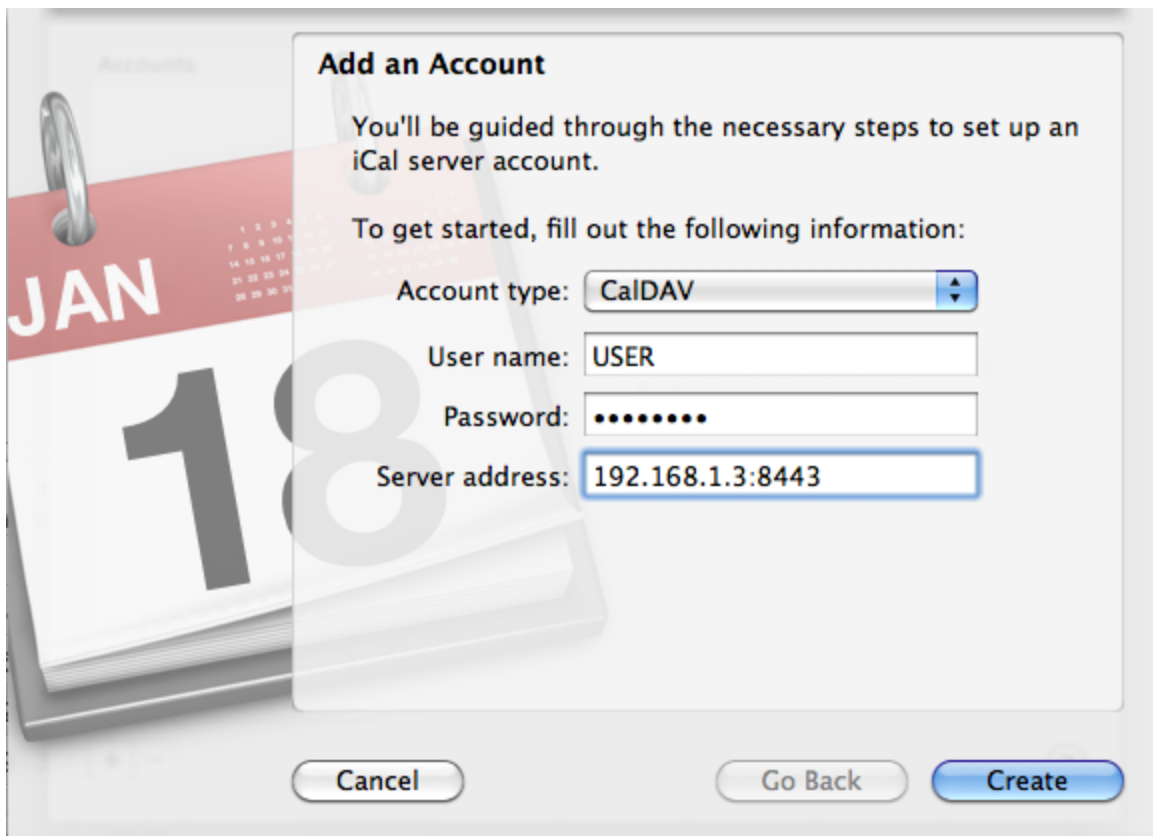
Now it is time for a first test run. Change the working directory and issue the command:

```
cd ~/calendar/CalendarServer
./run
```

You will see many messages, and finally the screen will stop with a message that looks like this:

```
2009-05-07 15:23:25+0200 [-] [caldav-8010] [-] AMP connection
established (HOST:UNIXSocket(None)
PEER:UNIXSocket('logs/caldavd.sock'))
```

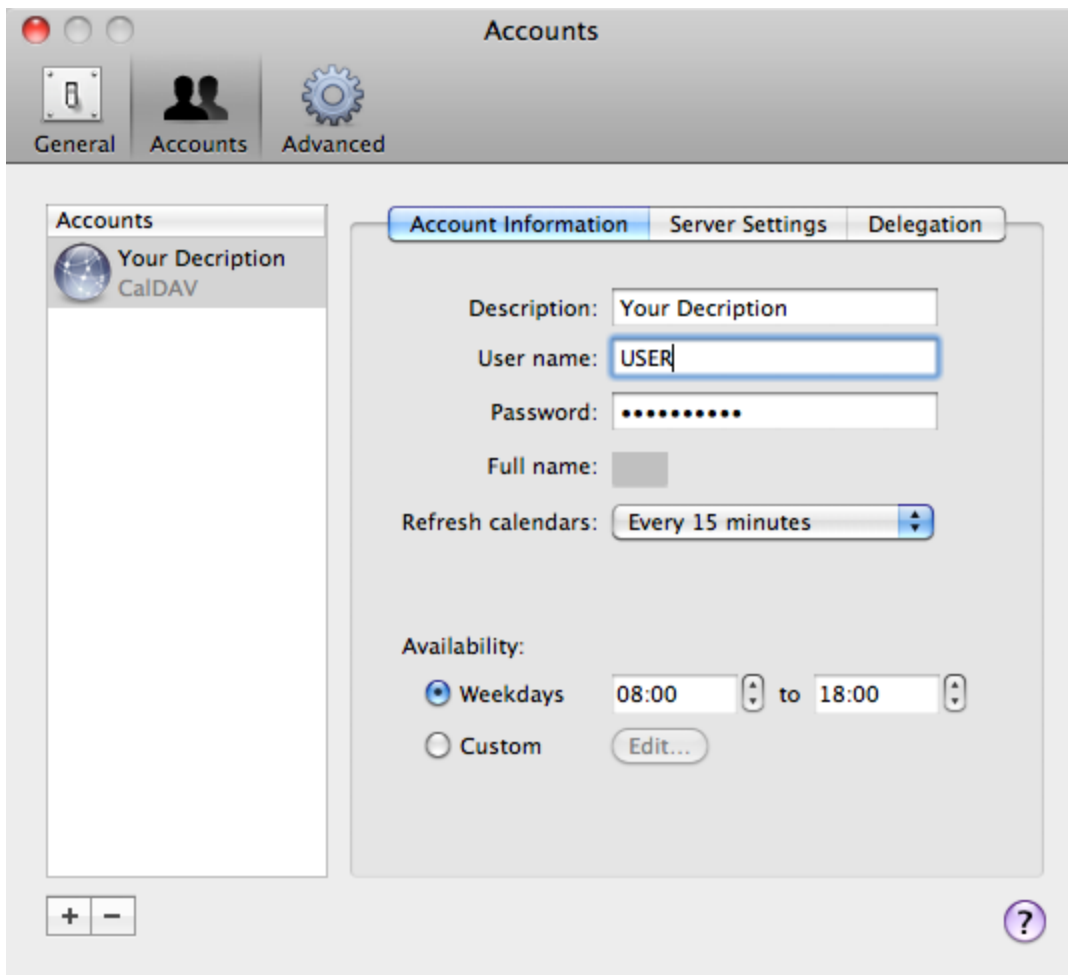
At this point, without closing the terminal, open iCal. Go to the iCal menu, and choose Preferences. Open the Accounts tab, and click the + sign on the bottom left. A dialogue will open, asking you to configure the new account:



Choose to add a CalDAV account type, and fill in the username and password for one of your users. The address should be the same one you defined earlier as `ServerHostName` in the plist file. The port chosen above is for SSL connection, which is always recommended. NOTE: Do NOT add more than one user at this time - the reason for this will become clear in a moment.

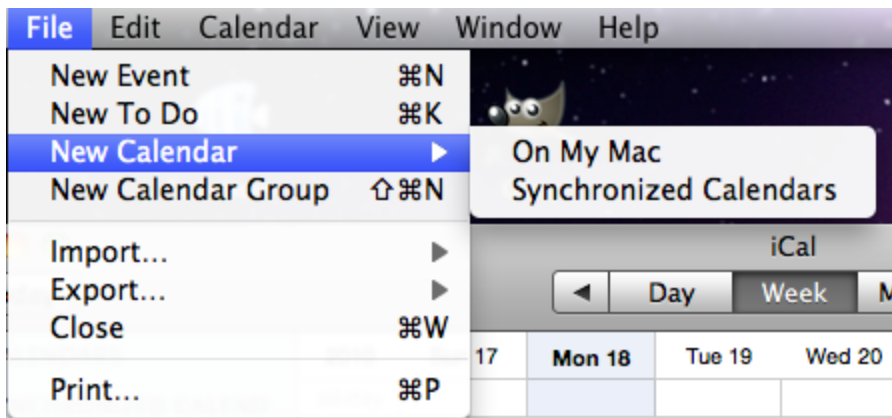
Next you will be presented with an SSL certificate warning. By clicking the little arrow for details, you can choose to add this certificate as a permanent exception, and thus this message will not bother you again.

After iCal finds the settings on its own and configures the account, you will be presented with a screen like this:



The only thing that you may wish to modify is the description. The description you choose will be the title of the calendar group on the left bar of iCal. After modifying it, you may close the Preferences dialogue.

You will now see that a new group of calendars appears at the left iCal bar. It includes only one calendar, named "calendar". You may rename it and edit its color if you wish, but this is the main calendar of this group, and can only be deleted if you unsubscribe through the Preferences dialogue. You can, however, add additional calendars to this group, through the File menu:

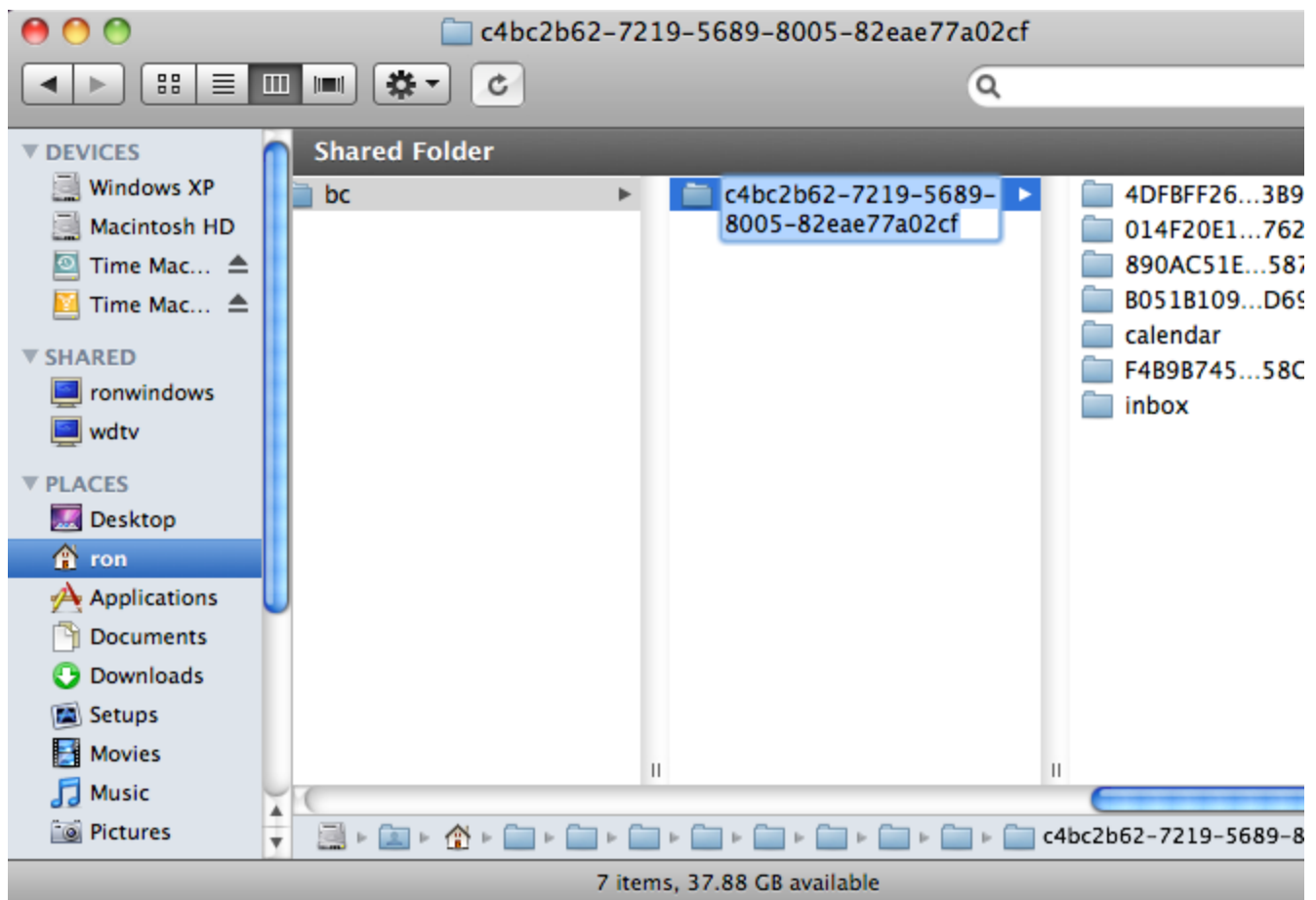


You can now export your old, local calendars one at a time, and import them to your newly created calendars. When you are done, quit iCal.

If everything worked so far, then we are almost set. Almost, but not quite. The problem is that the CalDAV server identifies the calendar groups (meaning, the users) by assigning them unique, long `guid` values. We have to locate the newly-created `guid` value for our first user, and add it to the file `accounts-text.xml`, or the server will not know which calendar belongs to that user the next time we run it. This is why we must add one user at a time to iCal - so that we can identify the proper `guid` for this user.

You must perform the following for each user you add through iCal, after each addition.

Using Finder, go to the following folder: `HOME/calendar/CalendarServer/twistedcaldav/test/data/calendars/__uids__`. If you did everything correctly so far, there should be only one subfolder (see note below if there are more), with two random letters as its name. Enter that subfolder, and you will find another, similar one. Enter that one as well. The next subfolder's name is the `guid` we are looking for. Click on its name once, make sure you don't delete it, and click Command-C to copy it to the clipboard:



Note: If there were more than one folder inside `__uids__`, then go inside the subfolders until you reach one called "calendar". If you already imported your calendars, it should be full. If it is empty, then this is the wrong subfolder and you should go back to `__uids__` and try another one.

So now you have the proper `guid` in your clipboard. Reopen the file `HOME/calendar/CalendarServer/conf/auth/accounts-test.xml` in TextEdit, and add the following line just below the `uid` line of the proper user:

```
<guid>PASTE_THE_GUID_HERE</guid>
```

Save the file, and now go back to the terminal. To shut down the current instance of the server press CTRL and C. Issue the `./run` command again, wait for the messages to stop, and open iCal to see if everything remained as it should. If it did, you can go on and add the next user, repeating the process for identifying and copying the `guid` value (yes, some of the times this will involve adding a user on iCal on a different computer, and then coming back to the server computer in order to edit the xml file). Don't forget to restart the server after every such addition. This process needs to be done only once for each `uid` in the xml file, regardless of the computer iCal was run on.

We are almost done! Everything is functioning, all the users have been added, and all we need to do is to make sure that the server runs in the background. Check and see if there is a folder named `LaunchAgents` in `HOME/Library`. If there isn't one, create it. Download [this file](#) and place it there. The file will make sure that when you log on to the computer, the server will start running in the background (give it a minute before you open iCal).

If you use iSync to synchronize your calendar with your mobile device, then launch iSync and click on the icon of that device. The window will expand, and you'll notice that the list of calendars to choose from now includes the ones on the server. Choose the calendars you wish to synchronize, click on "Device" at the top menu bar, and choose to reset the device, so that you are spared resolving conflicts in your appointments. This is important, as iSync will view all of the appointments you previously copied to the server as new ones, and will not know to relate them to the existing information on your mobile device.

If you plan to configure the Contacts Server as well, you may want to wait until the end of the next section before you perform the reset.

That's it! Now it is time to set up the very very new CardDAV Server for contacts.

2. Setting Up the Contacts Server

The CardDAV Server, which was released in the beginning of 2010, is so new, that it has no stable version. Additionally, the most recent versions (when these lines are being written) of Address Book (5.0.1, revision 864) and of iSync (3.1.0, revision 585.0) are not yet fully compatible with the CardDAV Server. Nevertheless, we can still get it to work without any glitches.

Basically, the setup of the CardDAV Server is very similar to that of the CalDAV Server. It is obvious that these two projects are going to be merged, and that actually creates, at present, collisions between the two servers.

The first thing we need to make sure of, therefore, is that we install the CardDAV Server in a different folder than the CalDAV Server. If they are installed in the same folder, both will try to impose the revisions of the Python extensions and the Twisted framework that they work with, and consequently - disable each other.

We'll start by creating a new folder:

```
cd
mkdir contacts
cd contacts
```

Now we will download the most recent snapshot of the server:

```
svn checkout http://svn.macosforge.org/repository/calendarserver
```

```
/ContactsServer/
```

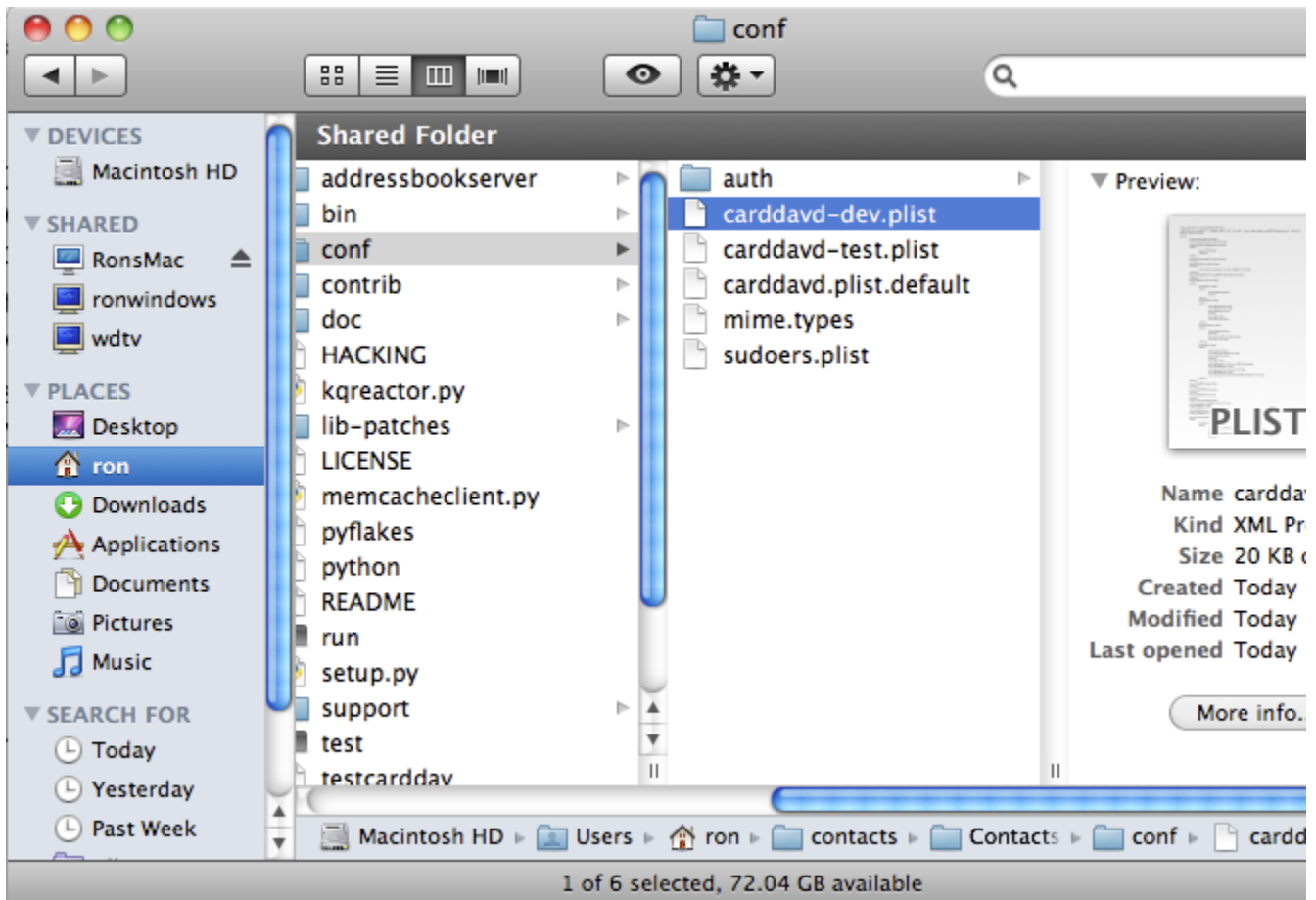
This will take a couple of minutes, and will be finished with the announcement of the revision that was checked out (in my case, it was 4964). Next we need to run the basic installation:

```
cd ContactsServer
```

```
./run -s
```

All of the supporting libraries will be downloaded and configured. The final message, after close to 10 minutes, will be "Using python as Python". If for some reason you get an error message, that means that the revision you checked out is broken. Delete the ContactsServer folder, wait a couple of days for the developers to upload a new revision, and try again. Sorry, I have no better advice.

Once the basic setup is done, we need to create the configuration file that the server will use. Similarly to what we did with the Calendar Server, use the Finder to go to `HOME/contacts/ContactsServer/conf`, duplicate the file `carddavid-test.plist` and rename it to `carddavid-dev.plist`:



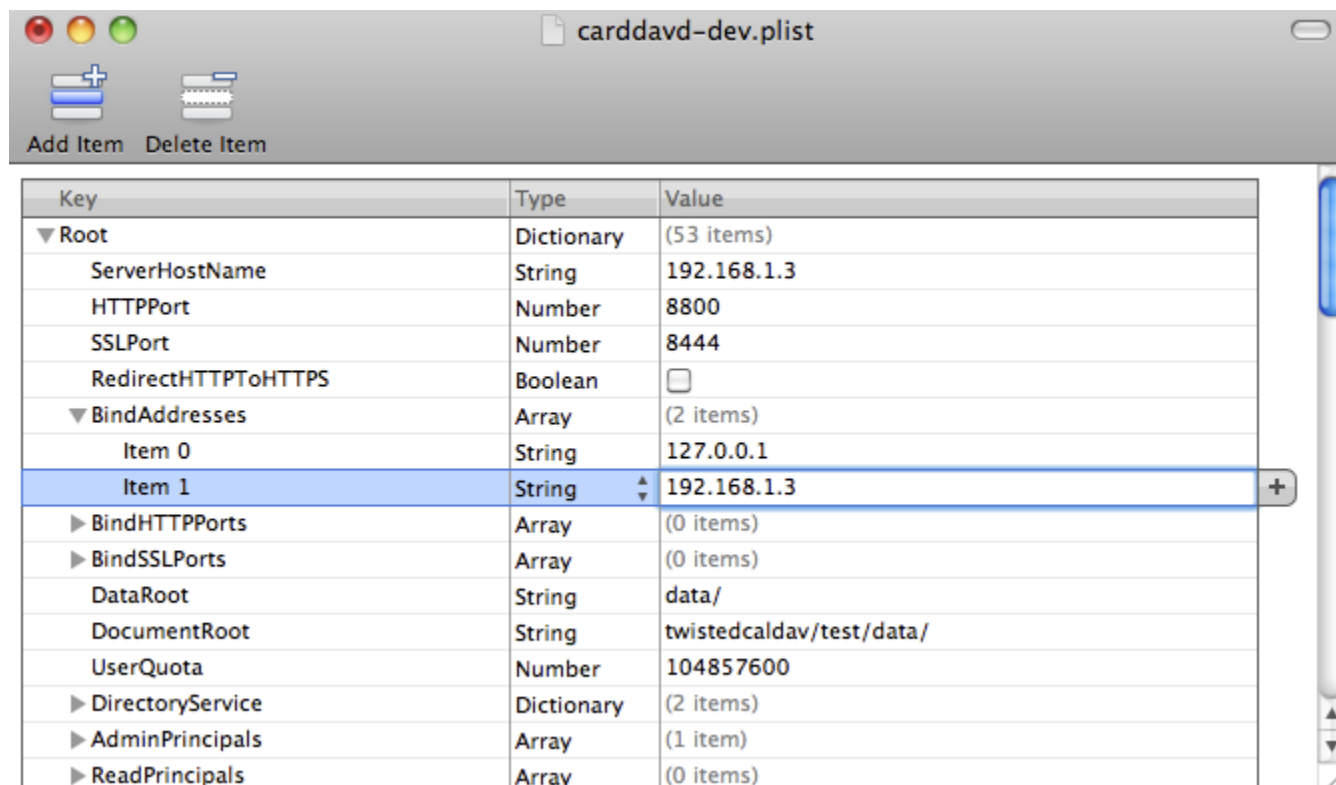
Now we need to edit this file. Double-clicking on it will open it in the property list editor. As

before, the string `ServerHostName` needs to be either your computer's IP address within the local network, its dynamic DNS address if you are going to contact the server remotely through the internet, or the localhost address (127.0.0.1) if you plan to use an SSH tunnel for the connection.

Although we did not change the values of the `HTTPPort` and the `SSLPort` when we configured the Calendar Server, now they are of concern. This is because as long as the two servers are separate entities, and for the time being they are, their ports must be different. The default `HTTPPort` is indeed already different, but we must change the `SSLPort` here to a different value, such as 8444.

Next we need to expand the array `BindAddresses` by clicking the small triangle to its left. We need to add new strings (by clicking on that line and then on the little icon to its right) with all the addresses that apply to this computer: 127.0.0.1, the internal network's IP address, and the dynamic DNS address, if it is used. Press the Enter key when you are done editing a line, and don't forget to save the file.

The edited file should look similar to this:



Key	Type	Value
▼ Root	Dictionary	(53 items)
ServerHostName	String	192.168.1.3
HTTPPort	Number	8800
SSLPort	Number	8444
RedirectHTTPToHTTPS	Boolean	<input type="checkbox"/>
▼ BindAddresses	Array	(2 items)
Item 0	String	127.0.0.1
Item 1	String	192.168.1.3
▶ BindHTTPPorts	Array	(0 items)
▶ BindSSLPorts	Array	(0 items)
DataRoot	String	data/
DocumentRoot	String	twistedcaldav/test/data/
UserQuota	Number	104857600
▶ DirectoryService	Dictionary	(2 items)
▶ AdminPrincipals	Array	(1 item)
▶ ReadPrincipals	Array	(0 items)

Before we can start testing our server, we need to add user accounts. The proper configuration file is located in the folder `ContactsServer/conf/auth`, and it is named `accounts-test.xml`. First duplicate it in Finder so that you have a backup of the original, and then open it in TextEdit. The following is an example of what you should change the file

into, starting from the line `<accounts realm="Test Realm">`:

```
<accounts realm="Test Realm">

<user>
<uid>admin</uid>
<guid>admin</guid>
<password>NEW_PASSWORD</password>
<name>Super User</name>
<first-name>Super</first-name>
<last-name>User</last-name>
</user>

<user>
<uid>USERNAME</uid>
<password>PASSWORD</password>
<name>FULL_NAME</name>
<first-name>FIRST_NAME</first-name>
<last-name>LAST_NAME</last-name>
</user>

<group>
<uid>group01</uid>
<password>PASSWORD</password>
<name>Group 01</name>
<members>
<member type="users">USERNAME (FROM BEFORE!)</member>
</members>
</group>

</accounts>
```

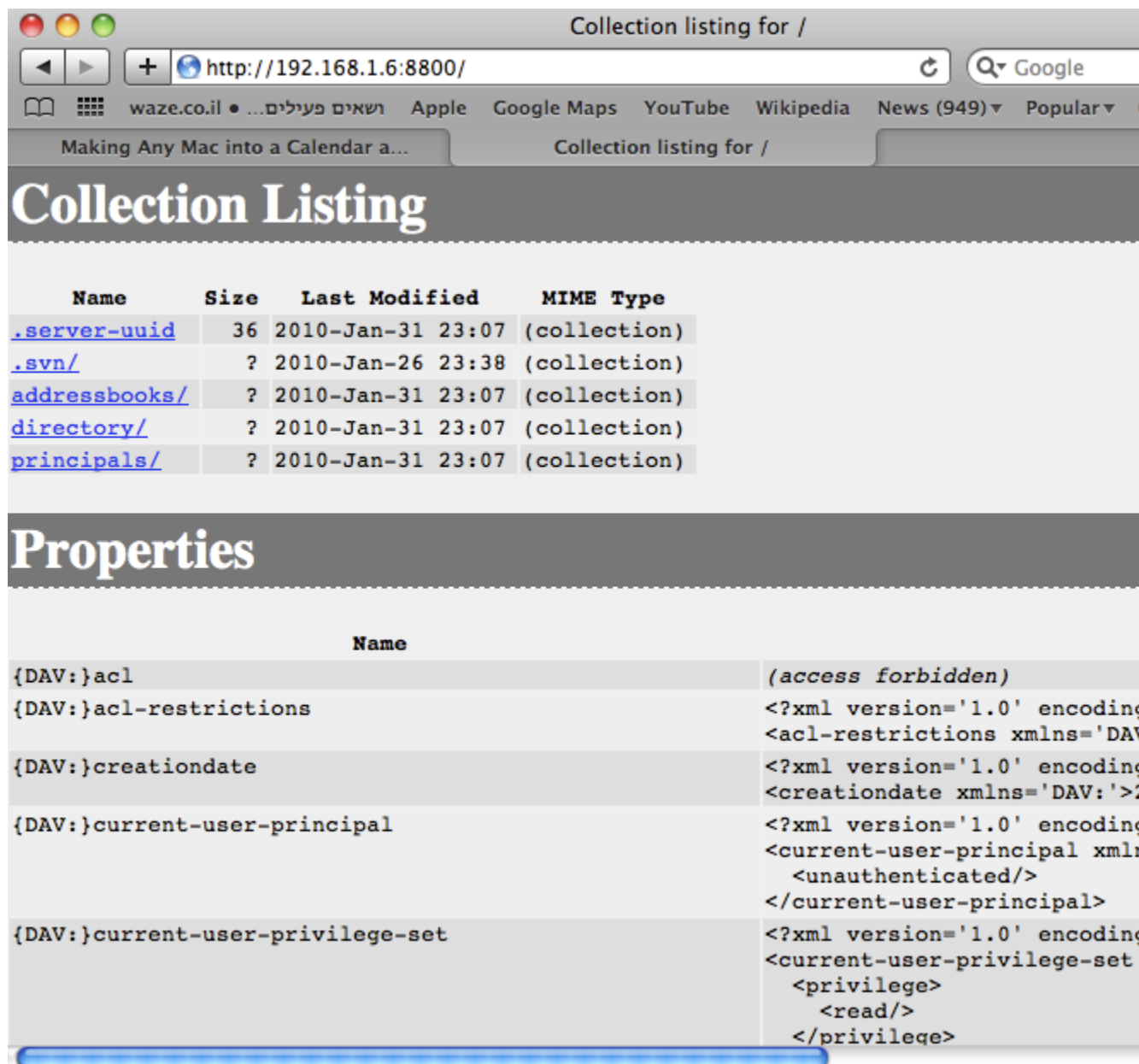
Once again, we keep the `admin` user, and add another for our own purposes; change everything in capital letters to suit your needs. Do not add `guid` lines yet - we will do that soon. For each shared contacts database use its own username and its own group, and configure later on the address book applications on all of your computers to use the same username for the same database.

It is time for us to start the server for the first time:

```
cd
cd contacts/ContactsServer
./run
```

There will be a long list of messages, which will stop with the words `'peopleNode': '/Search/Contacts', 'queryPeopleRecords': True}`. Do not close the terminal. Now we need to contact the server through its web-page, so that it will create the folders and

the `guid` for our account. Open a web browser, and enter the address `http://YOUR_LAN_ADDRESS:8800`. The page should look like this:



The screenshot shows a web browser window with the address bar set to `http://192.168.1.6:8800/`. The page title is "Collection listing for /". The main content area is titled "Collection Listing" and contains a table with the following data:

Name	Size	Last Modified	MIME Type
.server-uuid	36	2010-Jan-31 23:07	(collection)
.svn/	?	2010-Jan-26 23:38	(collection)
addressbooks/	?	2010-Jan-31 23:07	(collection)
directory/	?	2010-Jan-31 23:07	(collection)
principals/	?	2010-Jan-31 23:07	(collection)

Below the table is a section titled "Properties" which lists DAV properties and their values:

Name	Value
{DAV:}acl	(access forbidden)
{DAV:}acl-restrictions	<?xml version='1.0' encoding='utf-8'><acl-restrictions xmlns='DAV:'></acl-restrictions></?xml>
{DAV:}creationdate	<?xml version='1.0' encoding='utf-8'><creationdate xmlns='DAV:'>2010-01-31T23:07:00Z</creationdate></?xml>
{DAV:}current-user-principal	<?xml version='1.0' encoding='utf-8'><current-user-principal xmlns='DAV:'><unauthenticated/></current-user-principal></?xml>
{DAV:}current-user-privilege-set	<?xml version='1.0' encoding='utf-8'><current-user-privilege-set xmlns='DAV:'><privilege><read/></privilege></current-user-privilege-set></?xml>

Click on the link `principals`. You will be asked for a username and password - provide the ones you configured in the accounts file. Next click on `users`. Finally, click on your username. The following page provides us with plenty of necessary data, so keep it opened.

As was the case with the Calendar Server, we need to add the `guid` value to the file `accounts-text.xml`. This is the `guid` value that appears on the webpage under Principal Information:

Principal Details

Directory Information

Directory GUID: 86587b41-d080-5df3-8135-41b482cf220e
Realm: Test Realm

Principal Information

GUID: c4bc2b62-7219-5689-8005-82eae77a02cf
Record type: users
Short names: ron
Security Identities:
Full name:
First name:
Last name:
Email addresses:
' ()

Principal UID: c4bc2b62-7219-5689-8005-82eae77a02cf
Principal URL: /principals/_uids_/c4bc2b62-7219-5689-8005-82eae77a02cf/

Alternate URIs:
-> /principals/users/_/

Group members:
' ()

Group memberships:
-> [\(groups\) group01 - Group 01](#)

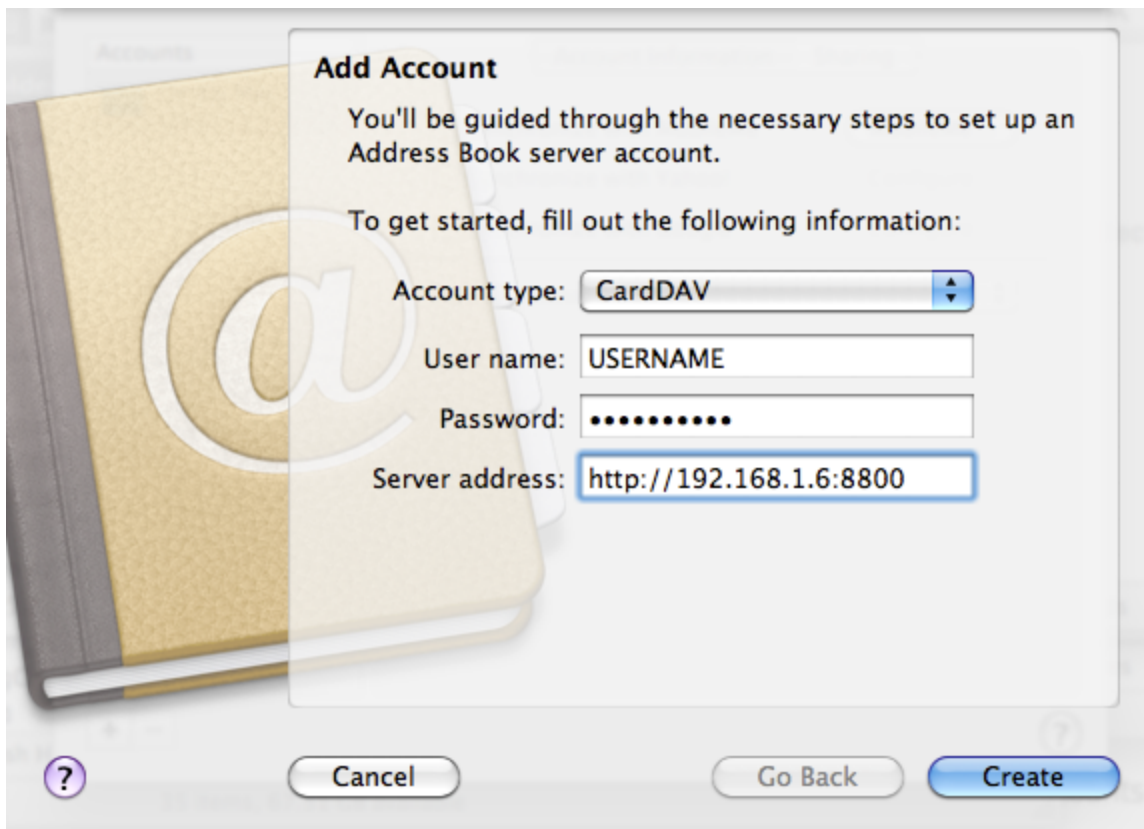
Address Book homes:
-> /addressbooks/_uids_/c4bc2b62-7219-5689-8005-82eae77a02cf

Mark that `guid` value, copy it to the clipboard with CMD-C, and paste it into the file `HOME/contacts/ContactsServer/conf/auth/accounts-test.xml` as a new line under your `username's uid`:

```
<guid>PASTE_THE_STRING_HERE</guid>
```

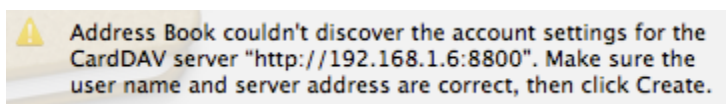
If you are using more than one user, then do the same for each of your users. Now restart the server: click on the terminal window, and press CTRL-C; then type `./run` again.

It is time for us to configure Address Book. Keep the server running, and open Address Book. Go to the Address Book menu at the top, and choose Preferences. Click on the Accounts tab, and then on the plus sign on the bottom left. Add your details to the following screen according to this example:

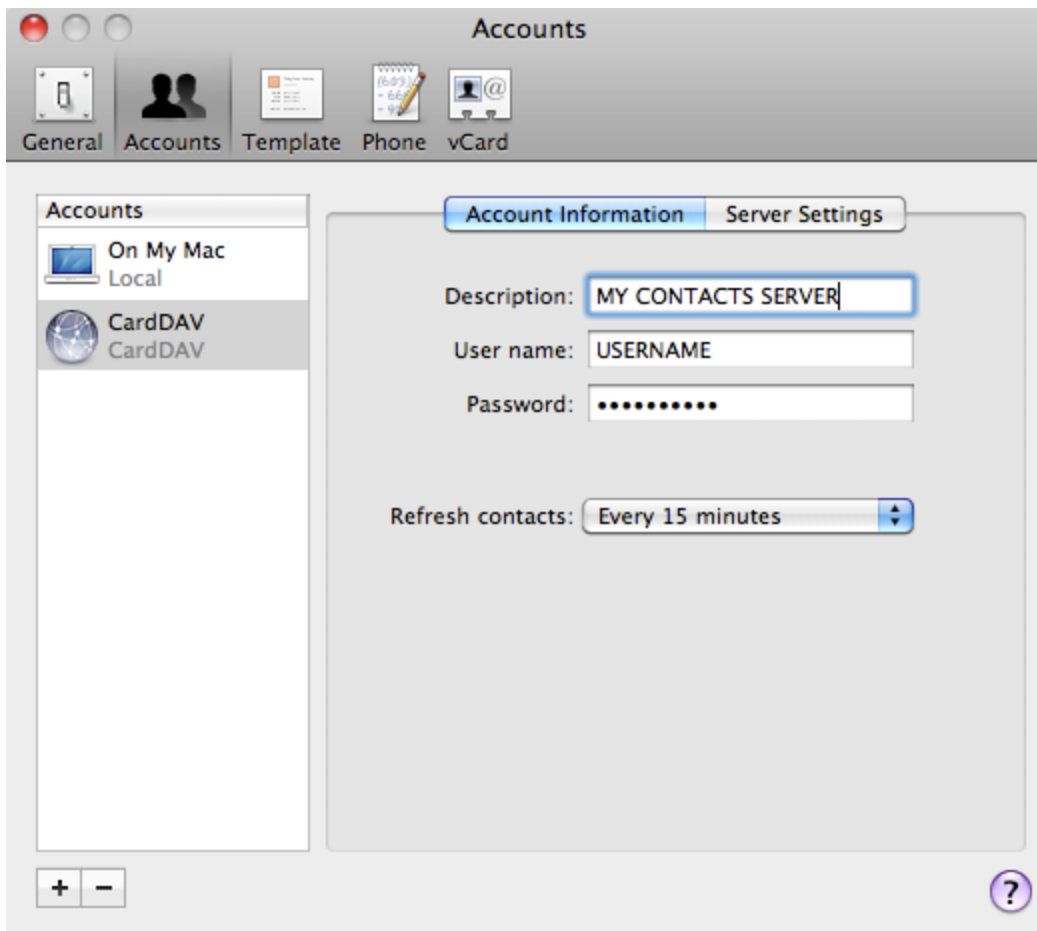


Now it is time for us to make two annoying discoveries:

1. Unlike the CalDav Server, the SSL implementation of the CardDAV Server is broken (as of the current revision). We have to use an unsecure connection, which is less than ideal (and thus it's recommended to connect to this server later through a VPN or an SSH tunnel).
2. Apple did not finish debugging and refining the CardDAV implementation of Address Book. You received, most likely, the following message:

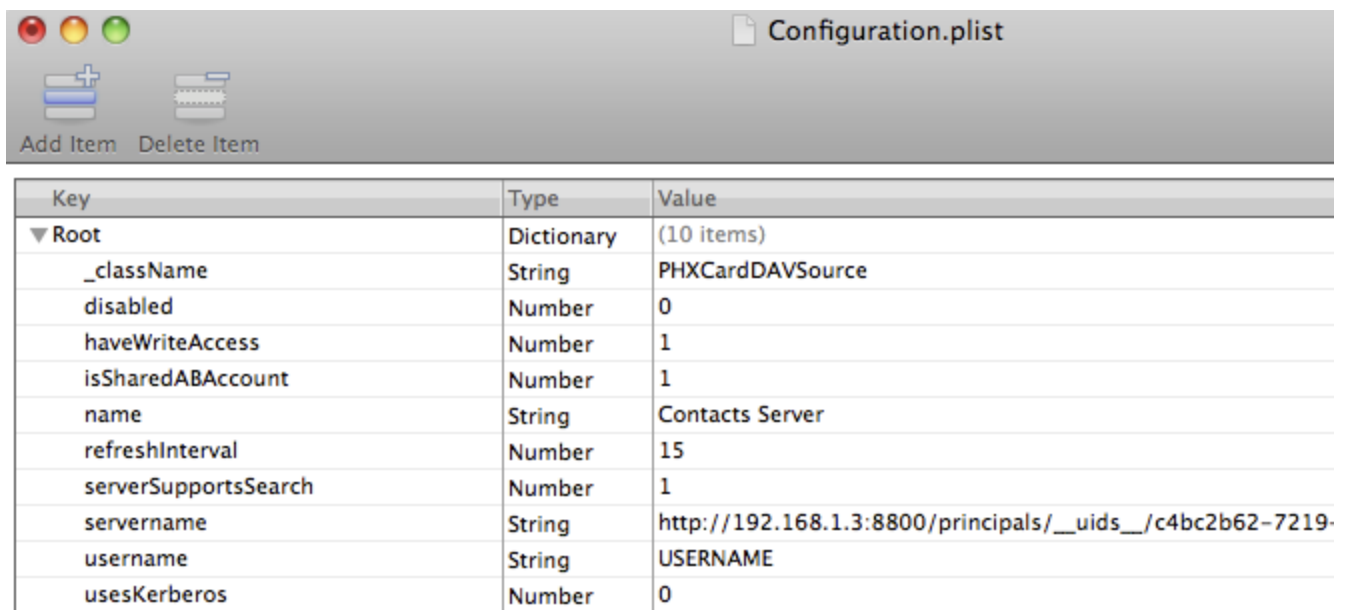


Indeed, click Create. In the screen that follows enter a description of the server you added, make sure the username and password are correct, and set a reasonable interval for synchronization. 15 minutes is a good idea. The "automatically" option, in my experience, does not work very well. Additionally, you will notice that whereas iCal has the option to manually refresh calendars, Address Book has no such option with contacts servers - for the time being.



Unfortunately, clicking on the "Server Settings" tab will be less than helpful, since the current version of Address Book does not let us edit the server path. We will have to do that manually. Quit address book, and using Finder, find the file HOME/Library/Application Support/Address Book/Sources/LONG_GUID_STRING/Configuration.plist. Double-click the file. We are interested in the string `servername`, which should be modified to this value:

`http://LAN_ADDRESS_OR_DYNAMIC_DNS:8800/principals/___uids___/YOUR_GUID`



Key	Type	Value
▼ Root	Dictionary	(10 items)
_className	String	PHXCardDAVSource
disabled	Number	0
haveWriteAccess	Number	1
isSharedABAccount	Number	1
name	String	Contacts Server
refreshInterval	Number	15
serverSupportsSearch	Number	1
servername	String	http://192.168.1.3:8800/principals/__uids__/c4bc2b62-7219-
username	String	USERNAME
usesKerberos	Number	0

Modify the other values according to the example above.

Launch Address Book. On the left column you should see an active spinning icon just to the right of the server's description - this means that Address Book was configured correctly and that it is synchronizing with the server. Once it's done you can drag all of your contacts onto the server's description, and after a short while they will all be copied. Do not delete them from "On My Mac", unless you are absolutely sure you will not need to use iSync. Using iSync requires a local copy of the contacts; more on the subject later.

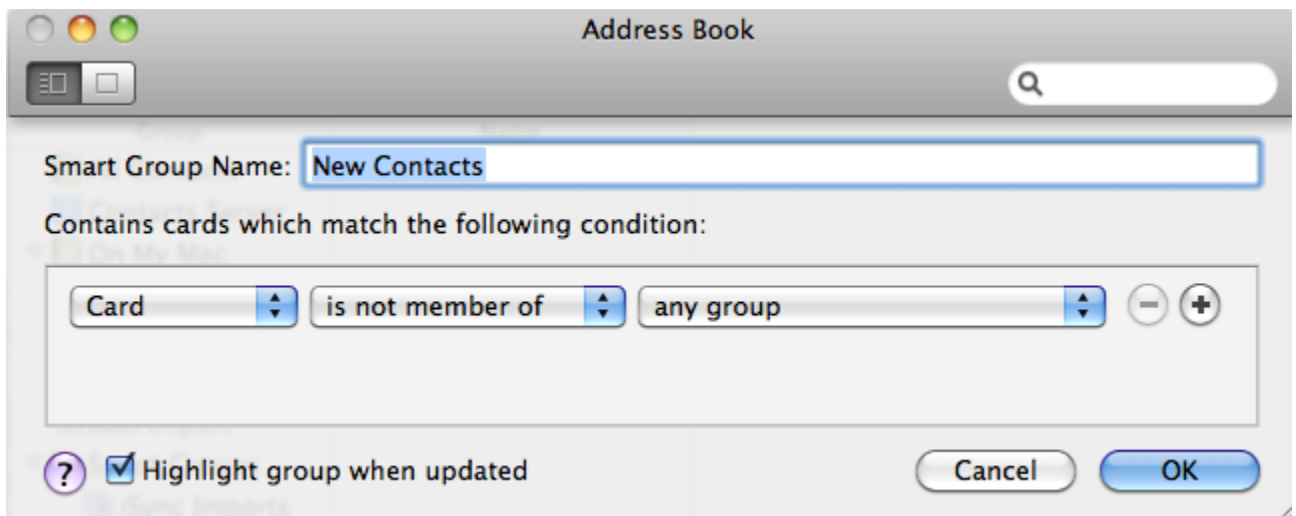
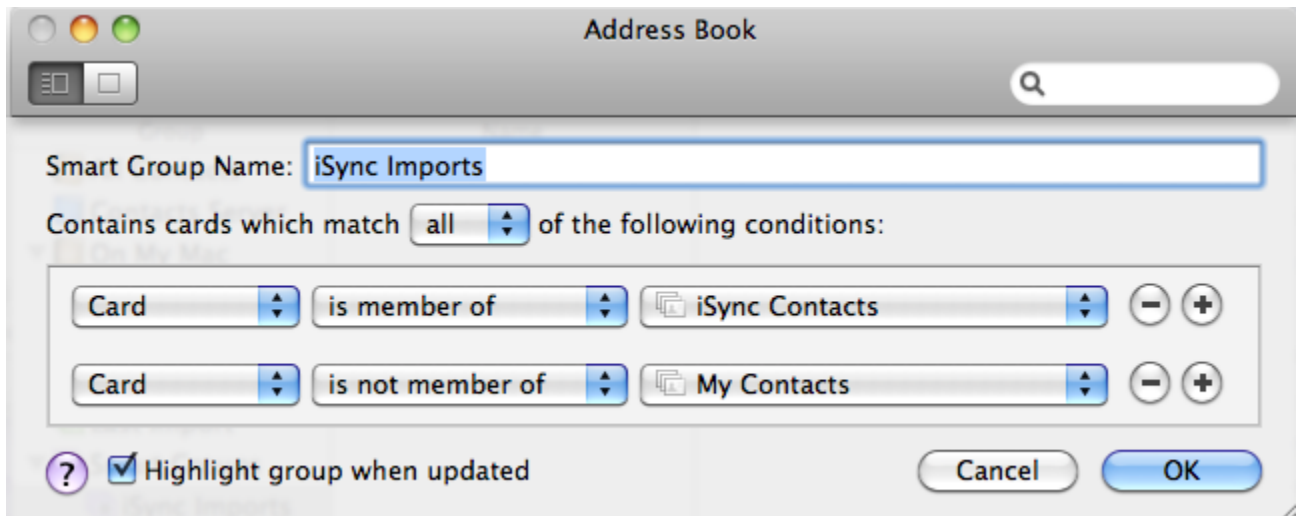
We are almost done. The only thing we need to do now is to configure the server to run in the background from the moment you log on to the computer. Use Finder to check whether the folder LaunchAgents exists in your HOME/Library folder. If it does not, create it. Download [this file](#) and place it there, and the next time you log on to your computer the server will run automatically in the background.

That's it! If you don't use iSync, then you are done. You can delete the local copy of your contacts (although it is prudent to back it up first through File -> Export -> Address Book Archive), and set up Address Book in a similar way on all of your other computers. You will have to copy the file we modified manually, `Configuration.plist`, to each client computer.

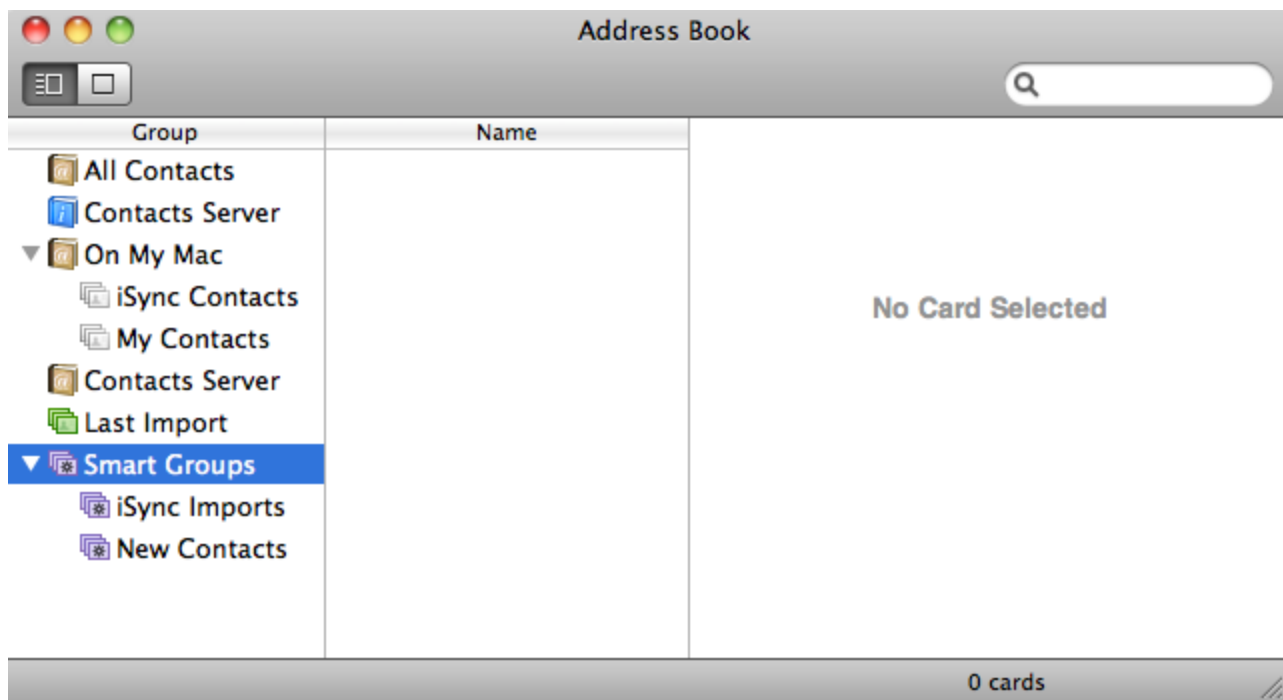
If you do use iSync to synchronize your contacts with your mobile device, then things are not going to be as simple as they were with the calendars. Once again, Apple's implementation of the CardDAV server is incomplete. iSync cannot synchronize contacts from a server as it can synchronize calendars. I tried all kinds of hacking methods to resolve this issue, and ended up with SmartFolders in Address Book as the most dependable solution.

Launch Address Book, and click on "On My Mac" in the left column. Then click on File at the top menu bar, and choose to create two groups (actual ones, NOT smart ones): call one "My Contacts" or something similar, and call the other "iSync Contacts". Now drag all of your contacts (or, rather, the ones you want on your mobile device) to BOTH groups.

Next click on "All Contacts" in the left column. Using the File menu once more, choose to create two smart groups according to these examples:



When you are done creating all the groups, your Address Book should look similar to this:



Next, launch iSync and click on the icon of your mobile device. The window will expand, and you will see that you can choose which of the groups you created to synchronize. Choose, naturally "iSync Contacts". Now you can use the "Device" menu to reset your device and make sure there are no conflicts between the information already on it and the newly created calendars and contacts on your computer.

In the future, any contact you create on your mobile device will appear, after synchronization, in Address Book's "iSync Imports" smart group. It will be simple enough to drag it into both the Contacts Server and into "My Contacts" group (it will already be in the "iSync Contacts" group). Once you dragged the new contact into both locations, it will be on your server and no longer a member of that smart group.

On the other hand, any contact you create on your Mac will appear immediately in the "New Contacts" smart group. You will need to drag it into three locations: the Contacts Server, the group "My Contacts" and the group "iSync Contacts".

I leave it to the enthusiastic reader to create an Automator workflow or an Applescript that automates all of this, and to Apple to fix that oversight in iSync.

Enjoy your synchronized calendars and contacts!

Comments:

Name: Ron

E-mail:

Date posted: April 07, 2010 - 09:30 am

Message: Dear Gads,

Thank you for the input. I am sorry for the delay in my reply. I did not try switching users on the client that the data directories are separate, and that different users and different groups are defined in the xml

Please post again and include a private email address, so that I can contact you. I want to hear more of y projects have merged - I need to update my guide.

Name: Gads

E-mail:

Date posted: *March 22, 2010 - 10:35 am*

Message: Thanks for your Response Ron!

The current problem is, the I setup on user on the mac like your tutorial. Now I switched to another use: contacts. Caldav works as expected.

In the accesslog on the server I only see some PROFINDs ... in Adressbook it just spins for a second. If

But nevertheless I'm trying to get it further :)

If you have any suggestions - please let me know.

btw.: I'm using svn trunk - as carddav was merged. It's running on a linux server. For the one user it now

Another bug in adressbook:

If you copy a Card with two phonenumbers - both in one category (f.i. work) - only the first one survive

Name: Ron

E-mail: [Private](#)

Date posted: *March 21, 2010 - 06:44 pm*

Message: I am afraid you did not provide enough information for me to know what's wrong. Try to go t anything.

Name: Gads

E-mail:

Date posted: *March 21, 2010 - 05:09 pm*

Message: Thanks :)

I got on vlient running with CardDav.

But the second doesn't see anything...

Name: Ron

E-mail: [Private](#)

Date posted: *March 11, 2010 - 12:51 am*

Message: It seems that the startup scripts don't always work. I am experimenting with adding delays to them to work.

-Ron

Feel free to enter your comments and/or questions

Name:

E-mail:

☐ Make Private

Message:



Add Comment

Reset

After submitting the form you will have to undergo a little turing test
(reading a randomly generated security code on an image) before your message is posted