

Adressbücher in ownCloud 6.003 samt Anbindung zu Thunderbird und Android

Inhaltsverzeichnis

Beliebig viele Adressbücher pro Benutzer.....	2
Wie ownCloud Kontakte speichert in der DB.....	2
URL's für CardDAV-Zugriff.....	3
Datenmigration bestehender TB-Adressbuch Kontakte zu ownCloud.....	3
Verteilerlisten und Kontakt-Kategorien.....	4
Thunderbird.....	4
Android.....	4
CardDAV-Sync App.....	4
DAVdroid App.....	5
Die Qual der Wahl.....	5
Contact Editor Pro.....	5
Anbindung.....	5
WICHTIGE Tipps eher Sie loslegen.....	5
Felder, Entsprechungen, Sync-Verhalten.....	7
CardDAV-Sync.....	10
DownSync (Server => Android).....	10
UpSync (Android=>Server).....	10
DAVdroid.....	10
DownSync (Server => Android).....	10
UpSync (Android=>Server).....	11

Ich wäre an dieser Aufgabe beinah verzeifelt, ich habe TAGE gebraucht. Wenn man dann alle Stolpersteine überwunden hat, funktioniert es letztendlich doch gut. Der Weg lohnt sich – aber gehe ihn mit Bedacht, überstürze nichts, lese die ganze Anleitung und plane ordentlich, eher gehandelt wird – dann dürfte es ohne allzuviel Frust auch für Sie klappen!

Beliebig viele Adressbücher pro Benutzer

OwnCloud erlaubt jedem Benutzer, beliebig viele eigene, benannte Adressbücher anzulegen. Dabei ist die „Settings“ Schaltfläche, wo man das macht, etwas versteckt: Wähle im ownCloud-Browser-UI zunächst (in der linken Spalte) „Kontakte“. Die Anzeige schaltet in Zwei-Spaltenmodus, links die Steuerung/Navigation und rechts die Inhaltsanzeige. In der Navigationsspalte steht **ganz unten** eine Schaltfläche mit Zahnrad – das ist es. Einmal gefunden ist die Bedienung sehr einfach.

Wie ownCloud Kontakte speichert in der DB

Tabelle **oc_contacts_addressbooks** : Liste der vorhandenen Adressbücher aller Nutzer

Tabelle **oc_contacts_cards_properties** :

- id	int(10)	
- userid	varchar(255)	Name des ownCloud-Nutzers e.g. "TimReeves" (Eigentümer)
- contactid	int(10)	Eindeutige num. ID des Kontaktes
- name	varchar(64)	Name der Kontakteigenschaft, z.B. CATEGORIES, EMAIL, N, FN, ORG, TEL
- value	varchar(255)	
- preferred	int(11)	

Hier stehen einzelne Kontakt-Eigenschaften, eine pro Tabellenzeile. Das sind Eigenschaftsnamen der Vcard 3.0 Spezifikation, siehe <http://de.wikipedia.org/wiki/VCard> or <http://en.wikipedia.org/wiki/VCard>

Tabelle **oc_vcategory** : Adressbuch CATEGORIES = Gruppen (aller Benutzer)

Tabelle **oc_vcategory_to_object** : [ObjId / CategoryId / Type]

Speichert die Beziehungen Kontakt <=> Kategorie, Pendant zu oc_contacts_cards_properties für die CATEGORIES.

Tabelle **oc_contacts_cards** :

- id int(10)
- addressbookid int(10)
- fullname varchar(255)
- carddata longtext
- uri varchar(200)
- lastmodified int(10)

Die Tabelle enthält pro Kontakt das VCard dazu (Vcard 3.0 Format im Feld "carddata"), fertig zum Ausliefern z.B. fürs Herunterladen.

Somit kann ownCloud alles speichern, was in Vcard 3.0 spezifiziert ist. Er kann sogar etwas mehr (Subset v. Vcard 4.0); siehe <https://github.com/owncloud/apps/issues/455> Übrigens: Vcard 3.0 enthält grundsätzlich UTF-8 Zeichen.

URL's für CardDAV-Zugriff

Hier ist das User-Manual dürftig bis inadequat. Dort steht nur:

- Android: `carddavs://<domain>/remote.php/carddav/`
- iOS: `http://<domain>/remote.php/carddav/principals/username/`

Beide o.g. Formen sind für Thunderbird + SOGo Connector nutzlos, zeigen nicht, wie man gezielt ein bestimmtes Adressbuch auswählt. Schade, denn es geht durchaus:

- **`https://<IhrDomain>/remote.php/carddav/addressbooks/<Benutzername>/<Adressbuch-Name>`**

Vorsicht beim Umbenennen von Adressbüchern! owncloud ändert dabei nur den Sichtnamen, nicht den URI-Namen! Letzterer bleibt so, wie er aus dem ursprünglichen Namen erzeugt wurde; er kann jedoch in der DB, in `oc_contacts_addressbooks::uri`, ohne Probleme angepasst werden.

Datenmigration bestehender TB-Adressbuch Kontakte zu ownCloud

- Thunderbird exportiert (von sich aus) nur LDIF oder CSV; ownCloud importiert nur Vcard)-:
- eGroupware exportiert zwar Vcard 2.1, aber leider nicht Vcard 3.0

- Es geht einfacher: Neues Adressbuch via SOGo in TB-Adressbuch anlegen [Datei | Neu | Remote-Adressbuch], synced zu ownCloud, URL s. oben, und Kontakte vom alten Adressbuch per Drag + Drop zum Neuen hinzufügen. So werden sie von SOGo Connector hochgeladen. Übrigens, in der ownCloud Doku wird fälschlicherweise behauptet, man können in TB-Adressbuch Kontakte per Drag & Drop nur verschieben – stimmt nicht, halte die Strg-Taste gedrückt zum Kopieren.

Verteilerlisten und Kontakt-Kategorien

Die in Thunderbird bekannten, herkömmlichen Verteilerlisten werden von Sogo Connector nicht gesynct. Sie passen nicht ins Konzept.

Generell ist man dazu übergegangen, Kontakt-Kategorien zu verwenden. Man definiert die Kategorien (explizit oder implizit) und kann ein Kontakt mehreren Kategorien zuordnen (Vcard „CATEGORIES“).

ownCloud, Thunderbird-Adressbuch und Android-Kontakte kennen alle die Kontakt-Kategorien - sie werden als Eigenschaften der einzelnen Kontakte generell auch gut synchronisiert zwischen den Geräten.

Thunderbird

- Das Plugin "Inverse SOGo Connector" von [heruntergeladenen](#) .xpi Datei installieren (zur Zeit des Schreibens: 24.0.5 v. 29. Mai 2014)
- Extras | Einstellungen erhält einen neuen Hauptreiter "SOGo" ganz rechts – für Kategorien. Das Einzige, was man hier angeben kann, ist der Name der Kategorie.
- Datei | Neu | Remote-Adressbuch
- SOGo Connector ist eher das „Kümmerkind“ unter den hier involvierten SW-Komponenten – es wird weiter entwickelt, doch die Energie dahinter ist eher mittelprächtigt. Den aktuellen Status findet man im [Bugtracker](#). Der Quellcode ist offen auf [GitHub](#).

Android

CardDAV-Sync App

- Installiere aus [Google Playstore](#): [CardDAV-Sync beta](#) von Marten Gajda (kostenpflichtig)
Zur Zeit des Schreibens Vn. 0.4.5 vom 4. November 2013. Liste der [Supported servers & services](#); Wiki-Seite zu [ownCloud](#).
- Es gibt auch eine Gratis-Version von CardDAV-Sync, die jedoch folgende Elemente nicht synchronisiert: websites, organizations, notes, anniversaries, nicknames, instant messengers, categories (mapped to groups in Android), phonetic names. Also zahlen Sie lieber.

DAVdroid App

- Installiere aus Google Playstore: [DAVdroid](#) alpha von bitfire web engineering – Hirner, Stockmann GesnBR (FOSS + kostenpflichtig)
Zur Zeit des Schreibens Vn. 0.5.13-alpha, 15. Mai 2014
- Für Samsung-Geräte besser direkt aus dem Samsung Store besorgen, sonst braucht man bei Android 4.1 JellyBean auch ein weiteres App das verhindert, dass Konten beim Booten gelöscht werden.

Die Qual der Wahl

Sie unterscheiden sich in der Ausrichtung: Während DAVdroid sich sehr puristisch verhielt, und das Android-Betriebssystem in keiner Weise umgehen oder „linken“ will, verhält sich CardDAV-Sync eher pragmatisch, und wenn ein wenig „gemogelt“ werden muss, um das gewünschte Nutzerverhalten zu erreichen, dann tut er das. Beides hat etwas für sich, hier muss man selbst eine Entscheidung treffen.

DAVdroid synchronisiert nur **ein** Adressbuch pro Benutzerkonto (d.h. Benutzerkonto am Server). Dies liegt am Aufbau der Kontakte-Realisierung direkt im Android-Betriebssystem, da können die Apps nichts dafür! CardDAV-Sync hingegen ermöglicht durchaus das Anbinden mehrerer Adressbücher für ein Benutzerkonto – man muss sie nur hintereinander einpflegen, mehrere (die er durchaus erkannt und auflistet) können nicht auf einmal ausgewählt werden.

Eine weitere wesentliche Unterscheidung ist bei der Behandlung von Kategorien, sprich Groups. Android sieht für 3rd-Party Apps keine saubere Schnittstelle zu den Gruppen vor, weswegen DAVdroid hier nichts synchronisiert; wem das aber wichtig ist, kann das gewünschte Verhalten von CardDAV-Sync bekommen, der hier „fünfe gerade sein lässt“. Mir sind die Gruppen wichtig, daher nahm ich CardDAV-Sync.

Contact Editor Pro

Das Standard-App Kontakte von Android schwächelt: Das UpSync von Kontaktgruppen funktioniert nicht, der Geburtstag kann nicht gesetzt werden. Mit dem Zusatz-App Contact Editor Pro, ebenfalls vom Pragmatiker Marten Gajda (kostenpflichtig), klappt alles.

Anbindung

- URL: [<https://>] <IhrDomain>/remote.php/carddav/addressbooks/<Benutzername>

WICHTIGE Tipps eher Sie loslegen

- Wer von z.B. eGroupware kommt, muss sich erstmal in ownCloud einfinden: Die Benutzeroberfläche ist sehr aufgeräumt, sehr praktisch, und ermöglicht dank viel jQuery-Einsatz viel mehr an Möglichkeiten als man auf den ersten Blick vermuten würde.

- Wenn wenige Teilnehmer Adressbücher miteinander teilen wollen, so lohnt sich der Einsatz von ownCloud-Gruppen kaum, jeder Teilnehmer kann per Klick auf dem Teilen-Symbol Benutzer direkt auswählen, die Sicht darauf bekommen, und zudem bestimmen welche Operationen erlaubt sind.
- Geteilte („shared“) Adressbücher erhalten für den CardDAV-Zugriff durch den „Beschenkten“ spezielle URLs, in etwa `https://<domain>/remote.php/carddav/addressbooks/<Beschenkten>/<Adressbuchname>_shared_by_<Eigentümer>`
- Bei geteilten Adressbüchern sehen Mitnutzer die Gruppen, die der Eigentümer den einzelnen Kontakten zugeteilt hat, nicht. Denn diese Gruppen beziehen sich auf den Gruppen des Eigentümers; ggf. muss der Mitnutzer eigene Gruppenzugehörigkeiten einrichten.
- Vorsicht: Verschiedene Nutzer sollten besser unterschiedliche Namen für ihre Adressbücher wählen. Wenn z.B. zwei Benutzer je ein Adressbuch „Kontakte“ haben besteht die Gefahr, dass wenn beim Anbinden der „_shared_by_User“ Suffix im URL vergessen wird, dann pustet man seine eigenen Kontakte in das Adressbuch des anderen Benutzers!
- Mir ist passiert, dass ein Photo in einem Kontakt die Synchronisation von TB-Adressbuch nach ownCloud torpediert hat – ich glaube das Bild war einfach zu groß. In so einem Fall bricht SOGo Connector beim UpSync immer an diesem Kontakt ab, die Nachfolgenden werden nie hochgeladen. Verschieben Sie zunächst das Problem-Kontakt in ein lokales TB-Adressbuch und stoßen Sie die Sync nochmal an.
- Gescheiterte Syncs protokolliert SOGo Connector dankenswerterweise in der Fehlerkonsole von Thunderbird: Extras | Fehlerkonsole.
- Ein weiterer Grund für scheiternde UpSyncs kann eine falsche Uhr-Einstellung am PC sein: Ich hatte auf einem Linux-PC mit englischer Sprache das Datum falsch eingetragen, weil es im Amerikanischem Format „mm.dd.YYYY“ einzugeben wäre, d.h. 5. Juni kommt dann als 6. Mai an)-: Besser gleich ein NTP Daemon installieren (Network Time Protocol). Ubuntu: `sudo apt-get install ntp`.
- Ein Kontakt, das ich gerade in ownCloud erfasst hatte, wurde brav zum TB-Adressbuch gesynct – und war dann hast-du-sie-nicht-gesehen sofort wieder in ownCloud verschwunden! Ich konnte jedoch eine Mini-Änderung des Kontakts in TB-Adressbuch vornehmen (Änderungen im Kategorien-Dropdown werden sofort gesynct), der Kontakt wurde dann brav hochgeladen, samt Bild.
Fazit: Lieber Änderungen in TB-Adressbuch vornehmen – dort können zur Sicherheit Kontakte per Drag + Drop (Strg!) zu einem lokalen Adressbuch kopiert werden, so dass einem im Fall eines Unglücks mit der Synchronisierung nichts passiert was nicht wieder hergeholt werden könnte :)
- Eingabe des Geburtstags in ownCloud-Browser-UI auf Deutsch geht wie gewohnt: tt.mm.JJJJ – aber wer auf English bedient, muss das Datum ganz anders eingeben: Month tt, YEAR (also Monat ausgeschrieben, Leerschritt, Tag (Ziffern), Komma, Leerschritt, Jahr).

- ETags durcheinander, falsches Datum an einem UpSync-PC => Syncs gehen nicht, Datenbestände am Server korrupt? Die Notbremse:
Per SQL die entsprechenden Tabellen der ownCloud-DB leeren, alle Kontakte erneut hochladen.
TRUNCATE oc_contacts_cards oc_contacts_cards_properties oc_vcategory_to_object

Felder, Entsprechungen, Sync-Verhalten

ownCloud / Thunderbird-Adressbuch (TBA) / Android-Kontakte (ADK) via CardDAV-Sync

Thunderbird	Owncloud	Android	OC >TBA	OC >ADK	TBA> OC	ADK> OC
Reiter: Kontakt						
Vorname	OK (Klicke auf Stift neben Anzeigename)	OK	J	J	J	2
Name	OK (Klicke auf Stift neben Anzeigename)	OK	J	J	J	2
Anzeigename	OK (FN)	OK	J	J	J	J
Spitzname	OK	OK	J	J	J	J
E-Mail Adresse	Arbeit	OK	J	J	J	3
Weitere E-Mail Adresse	Home	OK	J	J	J	3
[Chatname: Reiter „Chat“]	-	-				
Tel. Dienstlich	Arbeit	OK	J	J	J	J
Tel. Privat	Home	OK	J	J	J	J
Tel. Fax	Fax	Falsch, als Mobile gesynct	J	F	J	4
Tel. Pager	Pager	OK	J	J	J	N
Tel. Mobil	Mobil	OK	J	J	J	J
Bevor. Nachrichten-Format	In DB-Vcard als „X-MOZILLA-HTML:TRUE“	-	-	-	J	-
Reiter: Kategorien						
Mehrere Dropdowns	OK - Multi-Auswahl-Dropdown	Groups: Contact Editor Pro nötig!	J	J	J	J
Reiter: Privat	(Klicke auf „Home“)					
Adresse	OK	<Block: Home>	8	J	J	6
(Adresse Zusatz)	Anzeige nur in Zusammenfassung	<Block: Home>	8	J	J	6

Thunderbird	Owncloud	Android	OC >TBA	OC >ADK	TBA> OC	ADK> OC
Ort	OK	<Block: Home>	J	J	J	6
Bundesland	OK	<Block: Home>	J	J	J	6
PLZ	OK	<Block: Home>	J	J	J	6
Land	OK	<Block: Home>	J	J	J	6
Webseite [Privat]	OK (Home)	OK	J	J	J	7
Geburtstag	OK Geburtstag	OK (Events: Birthday)	J	J	J	5
Alter (errechnet)	-	-				
Reiter: Dienstlich						
Titel	OK	Organisation [1] (Job Title)	J	J	J	J
Abteilung	Nur in Kombi-Anzeige:	Fehlt	-	N	J	-
Organisation	ORG [0], [1]	Organisation [0] (Organisation)	J		J	J
Adresse	OK	<Block: Work>	8	J	J	6
(Adresse Zusatz)	Anzeige nur in Zusammenfassung	<Block: Work>	8	J	J	6
Ort	OK	<Block: Work>	J	J	J	6
Bundesland	OK	<Block: Work>	J	J	J	6
PLZ	OK	<Block: Work>	J	J	J	6
Land	OK	<Block: Work>	J	J	J	6
Webseite [Geschäft]	OK (Arbeit)	OK	J	J	J	7
Reiter: Sonstiges						
Benutzerdef. 1	Keine Anzeige, DB = Vcard CUSTOM1		-	-	J	-
Benutzerdef. 2	Keine Anzeige, DB = Vcard CUSTOM2		-	-	J	-
Benutzerdef. 3	Keine Anzeige, DB = Vcard CUSTOM3		-	-	J	-
Benutzerdef. 4	Keine Anzeige, DB = Vcard CUSTOM4		-	-	J	-
Notizen	OK	OK	J	J	J	J

Thunderbird	Owncloud	Android	OC >TBA	OC >ADK	TBA> OC	ADK> OC
Reiter: Chat						
9 Felder... !	Kommen von TBA nicht durch, bis auf X-AIM in DB. OwnCloud speichert bel. Instant Messenger korrekt ab - als Vcard IMPP Felder, d.h. ownCloud macht es richtig, SOGo Connector nicht.	X-AIM => IM: AIM, d.h. alte Konvention wird erkannt. Doch CardDAV-Sync erzeugt IMPP Vcard Sätze, die korrekt ankommen. In beiden Richtungen.	N	J	N	J
Reiter: Foto						
Foto	OK	OK	J	J	J	J

- 1) ownCloud speichert Bilder direkt in oc_contacts_cards::carddata wie folgt: PHOTO;ENCODING=b;TYPE=JPEG:<enkodiertes Bild>, was Vcard 3.0 entspricht. Eine Pendant in oc_contacts_cards_properties speichert ownCloud nicht.
- 2) ADK bietet bei der erweiterten Namens Erfassung gleich 5 Felder an: Prefix, Vor, Mid, Nach, Suffix. Diese kommen in ownCloud alle im Anzeigenamen (FN) an, aber in der erweiterten Namensansicht kennt ownCloud nur Vor, Mid und Nach.
- 3, 4) Work Fax und Home Fax kommen als vom Typ Work bzw. Home an - das „Fax“ geht verloren - und zwar weil CardDAV-Sync sie mit zwei Attributen exportiert z.B. „TEL;TYPE=WORK;TYPE=FAX:...“, womit owncloud anscheinend nicht umgehen kann. Zu erwarten wäre gemäß [RFC6350](#) eine Komma-getrennte Werteliste hinter einem TYPE-Keyword, z.B. „TEL;TYPE=fax,work:...“. Das Gleiche Spiel gibt es bei E-Mails: sie kommen an, aber Work und Home beide vom Typ „Internet“; die kann man in ownCloud leicht händisch korrigieren
- 5) Events (Birthday) und Groups werden am ADK beim Erfassen mit dem Standard-App zu einem 3rd-Party Online-Adressbuch gar nicht angeboten. Groups können nachträglich festgelegt werden, nach dem ursprünglichen Erfassen, werden aber nicht zum Server übertragen. Beides [funktioniert mit CardDAV-Sync](#), aber nur wenn der Zusatz-App „Contact Editor Pro“ benutzt wird.
- 6) ADK kennt nur ein Feld für Adressen)-: Man kann zwar dort mittels ENTER die Adress-Elemente auf mehreren Zeilen verteilen, doch sie werden dennoch weder von CardDAV noch DAVdroid auf die richtigen „;“ Positionen gebracht beim VCARD-Export. Siehe [hier](#) (ganz unten) und [hier](#) und [hier](#) (d.h. die Android-Entwickler sind sich des Problems bewusst - aber wann ändern sie etwas?).
- 7) CardDAV-Sync exportiert „EMAIL;TYPE=INTERNET;TYPE=HOME:...“ - gleiches Problem wie (3,4).

8) ownCloud benutzt nur ein Feld für Adresse + Adresszusatz, so dass diese zwangsmäßig bei Thunderbird ins erste der beiden Adressfelder ankommen.

9) Thunderbird unterscheidet bei Fax und Mobil nicht zwischen dienstlich und privat. Er importiert z.B. „TEL;TYPE=FAX,HOME“ sowie „TEL;TYPE=CELL,HOME“ in seine einzigen Mobil- und Fax-Felder, da kann ownCloud nichts dafür! Mir wäre eine Bevorzugung der „WORK“ Nummern lieber, aber da werden sicher die Bedürfnisse auseinandergehen. Dafür exportiert Thunderbird unspezifisch „TEL;TYPE=cell“ und „TEL;TYPE=fax“, welche ownCloud beim Import korrekt interpretiert - weil ownCloud anscheinend von der Zuordnung mehrerer Eigenschaften (Fax + Home) keinen Gebrauch macht.

- CardDAV (Android) gibt seine Anfrage so ab, dass alle VCARDs auf einmal enthalten sind (REPORT)

- SOGo Connector (Thunderbird) macht einfach ein GET auf das VCARD-URL.

Nach etwas Hin- und Her (propfinds) geht die Datenübergabe in einem vergleichsweise kleinen Datenfenster vonstatten - und zwar in [VCARD VERSION 3.0 Format](#). Dieses Format enthält diverse [Properties](#).

CardDAV-Sync

DownSync (Server => Android)

- Adressen werden auf sinnvoller Weise übernommen, mit ENTER zwischen Elementen: Street1 Street2 / Ort Bundesland PLZ / Land

UpSync (Android=>Server)

- Adressen mit ENTER zwischen Elementen werden nicht in Adresselementen getrennt.

DAVdroid

DownSync (Server => Android)

- CATEGORIES im Vcard werden ignoriert (auch wenn es gleichnamige Groups gibt). Siehe [hier](#) - es scheint schwierig zu sein.
- Adressen werden auf sinnvoller Weise übernommen, mit 2 x ENTER zwischen Elementen: Street1 Street2 / Bundesland / PLZ Ort
- Autotelefon kommt als weiteres „Mobile“ an (privates Mobiltelefon ebenso) - OK.

UpSync (Android=>Server)

- Groups werden ignoriert, es wird kein Vcard CATEGORIES erzeugt; eingegeben Gruppen werden gelöscht.
- URL;Type=home wird nicht erzeugt, stattdessen eine eingegebene Home-Website wieder gelöscht
- Adressen mit ENTER zwischen Elementen z.B. „Street1 Street2 / Bundesland / PLZ Ort“ werden nicht in Adresselementen getrennt.