



Productive  
Computing

Creating Efficiency Through Automation

# Address Book Manipulator



## Functions Guide

Revised November 13, 2015

# Table of Contents

I. INTRODUCTION.....	4
II. FUNCTION DESCRIPTIONS .....	5
1) Registration and Other Functions.....	5
PCAB_GetLastError( optType ).....	5
PCAB_GetOperatingMode.....	5
PCAB_Register( ServerName ; ServerPort ; ServerPage ; LicenseID ) .....	6
PCAB_Version( Type ) .....	6
2) Record Specific Functions.....	7
PCAB_New( RecordType ; optName ).....	7
PCAB_Open( UID ) .....	7
PCAB_Save( optType ).....	8
PCAB_Delete( UID ).....	8
PCAB_CompareProperty( Property ; Value ; optUID ; optKey ) .....	9
PCAB_GetFirstParentID.....	9
PCAB_GetNextParentID.....	10
PCAB_GetImage.....	10
PCAB_SetImage( BinaryData ) .....	10
PCAB_Display( UID ).....	11
3) Group Specific Functions .....	12
PCAB_AddPerson( UID ).....	12
PCAB_RemovePerson( UID ).....	12
PCAB_AddSubgroup( UID ) .....	12
PCAB_RemoveSubgroup( UID ).....	13
PCAB_GetFirstMemberID .....	13
PCAB_GetNextMemberID.....	13
PCAB_GetFirstSubgroupID.....	14
PCAB_GetNextSubgroupID .....	14
PCAB_GetListOfValues( RecordType ; GroupOrMember ; FieldName ).....	15
PCAB_ClearAll( GroupUID ; Whom ; ExcludeMe ) .....	16
4) Single Valued Property Functions (Group and Contact Records) .....	17
PCAB_SetValueForProperty( Property ; Value ; optType ).....	17
PCAB_DeleteValueForProperty( Property ; optType ) .....	17
PCAB_GetValueForProperty( Property ; optType ) .....	18
5) Multi-Valued Property Functions .....	19
PCAB_AddMV( Property ; Value ; Label ).....	19
PCAB_AddAddress( Label ; Street ; City ; State ; Zip ; Country ; CountryCode ) .....	19
PCAB_AddService( IMorSocialProfile ; Label ; Type ; Username ; optSocialURL ; optSocialUserID ).....	20
PCAB_GetLabelForUID( Property ; UID ).....	21
PCAB_SetLabelForUID( Property ; UID ; Label ).....	21
PCAB_GetValueForUID( Property ; UID ; optKey ) .....	22
PCAB_SetValueForUID( Property ; UID ; Value ; optKey ) .....	22
PCAB_DeleteValueForUID( Property ; UID optKey ) .....	23
PCAB_OpenFirstMVProperty( Property ; optType ).....	24
PCAB_OpenNextMVProperty .....	24
6) Search Functions.....	25

PCAB_Search( Type ; Property ; Label ; Key ; Value ; Comparison ; optExpandConstrain ) .....	25
PCAB_OpenFirstRecord( optType ) .....	25
PCAB_OpenNextRecord( optType ) .....	26
7) Custom Field Related Functions.....	27
PCAB_AddCustomField( RecordType ; FieldName ) .....	27
PCAB_GetCustomField( RecordType ; FieldName ) .....	27
PCAB_SetCustomField( RecordType ; FieldName ; Value ) .....	28
III. AVAILABLE PROPERTIES.....	29
Single Valued Properties .....	29
Multi-Valued Properties.....	29
IV. AVAILABLE KEY VALUES AND COMPARISON OPERATORS.....	30
Available Key Values.....	30
Available Comparison Operators .....	31
V. CONTACT US.....	32

# I. INTRODUCTION

## Description

The Address Book Manipulator plug-in offers functions that support a bi-directional data exchange between FileMaker® and Apple® Address Book or Contacts. With this plug-in FileMaker users are able to add, edit, and delete records into the address book, pull records from the address book, and search for specific records in the address book. These operations are accomplished by using FileMaker function calls from within FileMaker calculations. These calculations are generally determined from within FileMaker “SetField” or “If” script steps.

## Intended Audience

FileMaker developers or persons who have knowledge of FileMaker scripting, calculations, and relationships, as proper use of the plug-in requires that FileMaker integration scripts be created in your FileMaker solution.

## Successful Integration Practices

- 1) Read the Developer’s Guide
- 2) Read the Functions Guides
- 3) Download a demo: <http://www.productivecomputing.com/plugins/address-book-manipulator-id-39/>
- 4) Watch video tutorials: <http://www.productivecomputing.com/video>
- 5) Familiarize yourself with Apple’s Address Book/Contacts

## II. FUNCTION DESCRIPTIONS

This section describes the external plug-in functions that are available with the Address Book Manipulator.

### 1) Registration and Other Functions

#### PCAB\_GetLastError( optType )

**Purpose:**

This function returns a description of the last error encountered. The parameter specified causes the function to return either a numeric or textual description.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
optType	The type of format the error should be displayed as	"Number" or "Text"	"Text"

**Return Values:**

The numeric or textual description of the last error encountered.

#### PCAB\_GetOperatingMode

**Purpose:**

Used to identify the operating mode of the plug-in. Operating mode indicates if the plug-in has been properly registered.

**Parameters:**

None.

**Return Values:**

"DEMO" for demo mode, "LIVE" for live mode, "UNREGISTERED" for unregistered mode, "EXPIRED" for expired mode. Licensed plug-ins will be running in "LIVE" mode.

### **PCAB\_Register( ServerName ; ServerPort ; ServerPage ; LicenseID )**

#### **Purpose:**

Registers the plug-in with Productive Computing's registration server. Must call the Register function once before using the plug-in on a machine.

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
ServerName	The name of Productive Computing's registration server	"licensing.productivecomputing.com"	
ServerPort	The port number for registration	"80"	
ServerPage	The registration page	"/PCIReg/pcireg.php"	
LicenseID	The license ID to pass along for registration	"your license ID" or "DEMO-AB3"	

#### **Return Values:**

0 for success, otherwise !!ERROR!!

### **PCAB\_Version( Type )**

#### **Purpose:**

Used to identify the product and product version installed on a machine.

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
Type	Determines the type of version string to return	"Short" or "Long". The short version string includes just the number of the plug-in. The long version string includes the plug-in name followed by the version number.	Short

#### **Return Values:**

Product version and/or name, otherwise "!!ERROR!!"

## 2) Record Specific Functions

### PCAB\_New( RecordType ; optName )

#### **Purpose:**

This function creates a new record of the desired type and adds it to the Address Book. Any new records will not be committed until the PCAB\_Save function is called.

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
RecordType	The type of record to create	"Person" or "Group" depending on the type of record to create. "Contact" is also an acceptable value for backwards compatibility with the previous plug-in version.	
optName	The name of the group to be created	When a new group record is being created, this value will contain the name of the new group record. This field is NOT optional for the group record.	

#### **Return Values:**

The Address Book unique identifier (AddressBook UID) for the new record, otherwise "!!ERROR!!"

### PCAB\_Open( UID )

#### **Purpose:**

This function opens an Address Book record for reading and modifying. The desired record is indicated by the Address Book ID passed as a parameter. Use this function to open an Address Book record directly.

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier of the desired record		

#### **Return Values:**

0 for success, otherwise "!!ERROR!!"

### PCAB\_Save( optType )

**Purpose:**

This function commits any changes to the currently open record to the Address Book. Since a group and a person record can both be open at the same time, they will both be saved.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
optType	Determines which type of record to save.	"Group" operates on the currently opened group record. "Person" operates on the currently opened person record. If empty, the operation is performed on the last record opened/created.	

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### PCAB\_Delete( UID )

**Purpose:**

This function deletes an Address Book record from the Address Book. This function should be used with caution as the delete action cannot be reversed. The desired record indicated by the Address Book ID passed as a parameter is deleted.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier for the desired record		

**Return Values:**

0 for success, otherwise "!!ERROR!!"



### PCAB\_CompareProperty( Property ; Value ; optUID ; optKey )

#### **Purpose:**

This function compares "Value" to a value stored in Address Book. The value stored in the Address Book to be compared is identified by "Property", "optUID", and "optKey".

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property containing the value to be compared		
Value	The value that is to be compared to the Address Book property value		
optUID	Optional - The universal identifier for the Address Book property		
optKey	Optional - In the case of an Address Book address, this is the key to identify the portion of the address	Acceptable values are "Street", "City", "State", "Zip", "Country", and "Country Code".	

#### **Return Values:**

0 if the values match, -1 if they do not match, or "!!ERROR!!"

### PCAB\_GetFirstParentID

#### **Purpose:**

This function and the PCAB\_GetNextParentID functions are used to iterate through the ABGroup record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the first group to which this record belongs.

#### **Parameters:**

None.

#### **Return Values:**

An Address Book UID, otherwise "!!ERROR!!"

## PCAB\_GetNextParentID

### **Purpose:**

This function and the PCAB\_GetFirstParentID functions are used to iterate through the ABGroup record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the next group to which this record belongs.

### **Parameters:**

None.

### **Return Values:**

An Address Book UID, END, or "!!ERROR!!"

## PCAB\_GetImage

### **Purpose:**

This function retrieves the binary image data for the currently opened person record. The returned value must be stored in a FileMaker Container field.

### **Parameters:**

None.

### **Return Values:**

The image data, blank if no image is present, or "!!ERROR!!"

## PCAB\_SetImage( BinaryData )

### **Purpose:**

This function sets the image for the currently opened person record using binary data from a FileMaker Container field.

### **Parameters:**

Parameter Name	Purpose	Values	Default Value
BinaryData	The binary data content of a FileMaker Container field	This data must be stored either from a previous call to PCAB_GetImage, or an image file inserted directly into the Container field, not by reference.	

### **Return Values:**

0 for success, otherwise "!!ERROR!!"

### **Notes & Examples:**

The Container field must have stored its binary data contents by value, and not by reference.

Supported image types: .bmp, .gif, .jpg.

## **PCAB\_Display( UID )**

### **Purpose:**

This function displays the provided Contact/Person UID in Address Book.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
UID	The Address Book unique identifier of a Contact/Person record in Address Book		

### **Return Values:**

0 for success, otherwise "!!ERROR!!"

### **Notes & Examples:**

This function will display the Contact/Person record only if the Address Book application is running.

### 3) Group Specific Functions

#### PCAB\_AddPerson( UID )

**Purpose:**

This function adds an ABPerson record to the currently open group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier for the desired record		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

#### PCAB\_RemovePerson( UID )

**Purpose:**

This function removes an ABPerson record from the currently open group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier for the desired record		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

#### PCAB\_AddSubgroup( UID )

**Purpose:**

This function adds an ABGroup record as a subgroup to the currently open group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier for the desired group record		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### **PCAB\_RemoveSubgroup( UID )**

**Purpose:**

This function removes an ABGroup record as a subgroup from the currently open group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
UID	The Address Book unique identifier for the desired group record		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### **PCAB\_GetFirstMemberID**

**Purpose:**

This function and the PCAB\_GetNextMemberID functions are used to iterate through the ABPerson record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the first ABPerson record which is a member of the currently active group record.

**Parameters:**

None.

**Return Values:**

An Address Book UID, "END", or "!!ERROR!!"

### **PCAB\_GetNextMemberID**

**Purpose:**

This function and the PCAB\_GetFirstMemberID functions are used to iterate through the ABPerson record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the next ABPerson record which is a member of the currently active Group record.

**Parameters:**

None.

**Return Values:**

An Address Book UID, "END", or "!!ERROR!!"

### **PCAB\_GetFirstSubgroupID**

**Purpose:**

This function and the PCAB\_GetNextSubgroupID functions are used to iterate through the ABGroup record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the first ABGroup record which is a member of the currently active group record.

**Parameters:**

None.

**Return Values:**

An Address Book UID, "END", or "!!ERROR!!"

### **PCAB\_GetNextSubgroupID**

**Purpose:**

This function and the PCAB\_GetFirstSubgroupID functions are used to iterate through the ABGroup record IDs of which the currently open record is a member. Call this function to return the Address Book ID of the next ABGroup record which is a member of the currently active group record.

**Parameters:**

None.

**Return Values:**

An Address Book UID, "END", or "!!ERROR!!"

**PCAB\_GetListOfValues( RecordType ; GroupOrMember ; FieldName )**

**Purpose:**

This function is used to retrieve a value list of group members, or groups to which a record belongs. The value list contains either the names or UID's of the groups or members for the currently opened person or group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
RecordType	Indicates which of the currently opened records to retrieve the value list from	"Group" or "person". If "person" is specified, then the "GroupOrMember" parameter must be "groups".	
GroupOrMember	Indicates the contents of the value list	"Groups" or "members". "Members" will return a list of persons that are members of the currently opened group record. When specifying "members", the RecordType parameter must be "group".	
FieldName	Determines which field the value list will be based off of	"uid" or "name". When retrieving members of groups and specifying "name", the full name including the name prefix, middle name, and name suffix will be included in the list. Otherwise the group names or record UIDs will be returned in the list.	

**Return Values:**

The desired value list, otherwise "!!ERROR!!"

**PCAB\_ClearAll( GroupUID ; Whom ; ExcludeMe )**

**Purpose:**

This function is used to delete person or group records from the Address Book according to which group they belong. It may also be used to delete all records form the Address Book.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GroupUID	The UID for the group from which to delete select records for deletion	If left empty, it is assumed that all records from the Address Book are desired to be deleted.	
Whom	Indicates which record types to remove	Empty (""), "groups", or "members". "groups" will cause only subgroups (or groups if the UID is omitted) to be deleted. "members" will cause only person records to be deleted. Leaving the parameter empty will cause all records to be deleted.	
ExcludeMe	Indicates that the record marked as "me" in the Address Book will or will not be deleted	"True" or "False". True indicates the record will not be deleted. False indicates that the record will be deleted.	

**Return Values:**

0 for success, otherwise "!!ERROR!!"



## 4) Single Valued Property Functions (Group and Contact Records)

### PCAB\_SetValueForProperty( Property ; Value ; optType )

**Purpose:**

This function sets a property value stored in the Address Book to that of "Value" for an opened record. Note that "Value" must contain a value. To empty or remove a property value, use the "PCAB\_DeleteValueForProperty" function.

**Dependencies:**

Must have a record (person or group) opened.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property that accepts the value		
Value	The value that is to be stored in the property	Any non-empty value.	
optType	The type of record to which the property belongs	"Group" or "person". Group operates on the currently opened group record, Person operates on the currently opened person record. If empty, the operation is performed on the last record opened/created.	

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### PCAB\_DeleteValueForProperty( Property ; optType )

**Purpose:**

This function deletes a property value stored in the Address Book for an opened record.

**Dependencies:**

Must have opened a record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property that holds the value		
optType	The type of record to which the property belongs	"Group" operates on the currently opened group record. "person" operates, on the currently opened person record. If empty, the operation is performed on the last record opened/created.	

**Return Values:**

0 for success otherwise "!!ERROR!!"

**PCAB\_GetValueForProperty( Property ; optType )**

**Purpose:**

This function retrieves a property value stored in the Address Book for a currently opened record.

**Dependencies:**

Must have opened a record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property that contains the value		
optType	The type of record to value the property's value from	"Group" operates on the currently opened group record. "person" operates, on the currently opened person record. If empty, the operation is performed on the last record opened/created	

**Return Values:**

The value stored in the property or "!!ERROR!!"

## 5) Multi-Valued Property Functions

### PCAB\_AddMV( Property ; Value ; Label )

**Purpose:**

This function adds a new value to the multi-valued property identified by the “property” parameter for the currently opened person record.

**Dependencies:**

Must have opened a record and the property must be a multi-value property.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property that accepts the value		
Value	The value to be stored in the property		
Label	The label for the property (Home, Work, etc. )		

**Return Values:**

Either the unique identifier for the property entry, or "!!ERROR!!"

### PCAB\_AddAddress( Label ; Street ; City ; State ; Zip ; Country ; CountryCode )

**Purpose:**

This function adds a new address to the currently opened person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Label	The label for the address (Home, Work, etc. )		
Street	The street portion of the address		
City	The city portion of the address		
State	The state portion of the address		
Zip	The zip portion of the address		
Country	The country portion of the address		
Country Code	The country code portion of the address. This value affects how Address Book prints and displays the address		

**Return Values:**

Either the unique identifier for the address entry, otherwise "!!ERROR!!"

## PCAB\_AddService( IMorSocialProfile ; Label ; Type ; Username ; optSocialURL ; optSocialUserID )

### **Purpose:**

This function adds a new service (instant message or social profile) to the currently opened person record.

### **Dependencies:**

Must have opened a person record.

### **Parameters:**

Parameter Name	Purpose	Values	Default Value
IMorSocialProfile	Determines if the service to be added is an instant message service or a social profile	"InstantMessage" or "SocialProfile"	
Label	The label of the service visible on the Address Book UI	Any string value, such as "Facebook" or "My IM Account"	
Type	The type of instant message or social profile service	For Instant Message services: "AOL", "ICQ", "Jabber", "MSN", "Yahoo", "Facebook", "Gadu-Gadu", "GoogleTalk", "QQ", "Skype". For Social Profile services: "Twitter", "Facebook", "Flickr", "LinkedIn", "Myspace", "Sina Weibo", "Tencent Weibo".	
Username	The service user's username		
optSocialURL	Optional parameter of the social profile URL		
optSocialUserID	Optional parameter of the social profile user ID		

### **Return Values:**

Either the unique identifier for the service, or "!!ERROR!!"

### **Notes & Examples:**

The properties for "Sina Weibo" and "Tencent Weibo" are only available in OS X 10.8+ and 10.9+, respectively.

To add a new Yahoo instant message service, you would make a call like this:

```
PCAB_AddService( "InstantMessage" ; "Personal Yahoo" ; "Yahoo" ; "JohnSmith123" )
```

To add a new Facebook social profile, you would make a call like this:

```
PCAB_AddService( "SocialProfile" ; "My Facebook" ; "Facebook" ; "j.smith123" ; "http://www.facebook.com/j.smith123" )
```

You can leave the optSocialURL and optSocialUserID blank; doing so will inform Address Book to generate them for you.

### **PCAB\_GetLabelForUID( Property ; UID )**

**Purpose:**

This function is used to retrieve the Label for a particular property value for the currently opened Person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property type for the property		
UID	The unique identifier for the property entry		

**Return Values:**

The label of the property, or "!!ERROR!!"

### **PCAB\_SetLabelForUID( Property ; UID ; Label )**

**Purpose:**

This function is used to change the label of a particular property entry for the currently opened Person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property type for the property		
UID	The unique identifier of the property entry		
Label	The label to set for the property specified		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### PCAB\_GetValueForUID( Property ; UID ; optKey )

**Purpose:**

This function is used to retrieve the value for a particular property value stored in the Address Book for the currently opened person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property type for the property		
UID	The unique identifier for the property entry		
optKey	Optional. Identifies the portion of the address or service, if the property is an Address or a Service	For Addresses: "Street", "City", "State", "Zip", "Country", and "Country Code", For Instant Message Services: "InstantMessageService", "InstantMessageUsername", For Social Profile Services: "SocialProfileService", "SocialProfileUsername", "SocialProfileURL", "SocialProfileUserIdentifier"	

**Return Values:**

Either the value stored in the property entry, or "!!ERROR!!"

### PCAB\_SetValueForUID( Property ; UID ; Value ; optKey )

**Purpose:**

This function is used to change the value stored in a particular property entry for the currently opened person record. Note that the Value cannot contain an empty value. To remove a value for a particular property call the PCAB\_DeleteValueForUID function.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property type for the property		
UID	The unique identifier for the property entry		
Value	The value for the property	Must not be empty	
optKey	Optional. Identifies the portion of the address or service, if the property is an Address or a Service	For Addresses: "Street", "City", "State", "Zip", "Country", and "Country Code", For Instant Message Services: "InstantMessageService", "InstantMessageUsername", For Social Profile Services: "SocialProfileService", "SocialProfileUsername", "SocialProfileURL", "SocialProfileUserIdentifier"	

**Return Values:**

0 for success, otherwise "!!ERROR!!"

**PCAB\_DeleteValueForUID( Property ; UID optKey )****Purpose:**

This function is used to delete the value stored in a particular property entry for the currently opened person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property type for the property		
UID	The unique identifier for the property entry		
optKey	Optional. Identifies the portion of the address or service, if the property is an Address or a Service	For Addresses: "Street", "City", "State", "Zip", "Country", and "Country Code", For Instant Message Services: "InstantMessageService", "InstantMessageUsername", For Social Profile Services: "SocialProfileService", "SocialProfileUsername", "SocialProfileURL", "SocialProfileUserIdentifier"	

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### **PCAB\_OpenFirstMVProperty( Property ; optType )**

**Purpose:**

This and the PCAB\_OpenNextMVProperty is used to iterate through the values stored in a multi-valued property for the currently opened person record. Use this function to open and access the first value stored in the property for the currently opened person record.

**Dependencies:**

Must have opened a person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Property	The Address Book property to be opened		
optType	The type of record to get the first MV property from	"Group" operates on the currently opened group record. "Person" operates on the currently opened Person record. If empty, the operation is performed on the last record opened/created.	

**Return Values:**

The unique identifier for the property, or "!!ERROR!!"

### **PCAB\_OpenNextMVProperty**

**Purpose:**

This and the PCAB\_OpenFirstMVProperty is used to iterate through the values stored in a multi-valued property for the currently opened person record. Use this function to open and access the next value stored in the property for the currently opened person record.

**Dependencies:**

Must have opened a person record and made a previous call to PCAB\_OpenFirstMVProperty.

**Parameters:**

None.

**Return Values:**

The unique identifier of the property, otherwise "!!ERROR!!"



## 6) Search Functions

### PCAB\_Search( Type ; Property ; Label ; Key ; Value ; Comparison ; optExpandConstrain )

#### **Purpose:**

The Search function locates Address Book records that match the criteria parameters. The resulting found record set is retained in memory for access by the OpenFirstRecord and OpenNextRecord functions. For example, use this function to search for records modified after a specific time interval or search for contacts by group name, zip code, or other specified criteria. This function proves quite useful when creating a bidirectional exchange as you can specify your specific search criteria. The new optExpandConstrain parameter allows for the function to expand or constrain the last search results.

#### **Parameters:**

Parameter Name	Purpose	Values	Default Value
Type	Indicates the type of records to search	"Contact"/"Person" or "Group"	
Property	Limit searches to the properties of this property type		
Label	Limit searches to properties with this label only		
Key	Limit searches to properties with this key only	Available Key values defined later in this guide	
Value	The value used to compare to the property values		
Comparison	This is the type of comparison between the "Property" and the "Value" parameters	Available Comparison operators defined later in this guide	
optExpandConstrain	Expands or constrains the previous search results with the results of this search	"Expand" performs an "OR" operation of the new search with the last search, similar to the "Expand Found Set" FileMaker command. "Constrain" performs an "AND" operation of the new search with the last search, similar to the "Constrain Found Set" FileMaker command.	"" , which indicates that a new find should be performed

#### **Return Values:**

The number of records in the found set, or "!!ERROR!!"

### PCAB\_OpenFirstRecord( optType )

#### **Purpose:**

This function opens the first record in the found set of person or group records depending on the optType parameter. If the optType parameter is empty, then the first record from the last call to PCAB\_Search is opened.

#### **Dependencies:**

Must have made a previous call to PCAB\_Search.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
optType	The type of record to open	"Group" operates on the first group record in the found set. "Person" operates on the first Person record in the found set. If empty, the operation is performed on the record type specified by the previous call to PCAB_Search.	

**Return Values:**

The unique identifier of the record, or "!!ERROR!!"

**PCAB\_OpenNextRecord( optType )****Purpose:**

This function opens the next record in the found set of person or group records depending on the optType parameter. If the optType parameter is empty, then the next record from the last call to PCAB\_Search is opened.

**Dependencies:**

Must have made a previous call to PCAB\_OpenFirstRecord.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
optType	The type of record to open	"Group" operates on the next group record in the found set. "Person" operates on the next Person record in the found set. If empty, the operation is performed on the record type specified by the previous call to PCAB_OpenFirstRecord.	

**Return Values:**

The unique identifier of the record, or "!!ERROR!!"

## 7) Custom Field Related Functions

### PCAB\_AddCustomField( RecordType ; FieldName )

**Purpose:**

Adds a custom field to the Address Book records. The field is added to group or person record according to the RecordType parameter and is user defined. Note that custom fields may not be removed from Address Book records, are not visible in the Address Book user interface, and are not synced with any other application (MobileMe, Mail, etc. ). This is a limitation imposed by Apple rather than the plug-in. Please take caution as this is a one way street. Any custom field created in Address Book cannot be deleted. In addition, any custom field created with this function will be only of single valued string type.

**Dependencies:**

Must have opened a group or person record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
RecordType	The type of record which will receive the custom field	"Group" or "Person"	
FieldName	The name of the custom field to be added		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### PCAB\_GetCustomField( RecordType ; FieldName )

**Purpose:**

This function retrieves the value stored in a custom field.

**Dependencies:**

Must have opened a person or group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
RecordType	The type of record from which to get the custom field value	"Group" or "Person"	
FieldName	The name of the field from which to retrieve the value	"Group" or "Person"	

**Return Values:**

The value in the field, otherwise "!!ERROR!!"

**PCAB\_SetCustomField( RecordType ; FieldName ; Value )**

**Purpose:**

This function sets the value of a custom field in the currently opened group or person record.

**Dependencies:**

Must have opened a person or group record.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
RecordType	The type of record of which to set the custom field value	"Group" or "Person"	
FieldName	The name of the custom field for which to set the value		
Value	The value to which the custom field will be set		

**Return Values:**

0 for success, otherwise "!!ERROR!!"

### III. AVAILABLE PROPERTIES

\*\* Fields ARE case sensitive.\*\*

#### Single Valued Properties

A single valued property can only have one value in Address Book. Use PCAB\_SetValueForProperty function to set these properties and PCAB\_GetValueForProperty function to get these properties.

Module	Method	Field Name	Acceptable Values/Notes
All Records	Get	Modified Date	The date the record was last modified
All Records	Get	Creation Date	The date the record was created
Group Records	Get/Set	Name	The name of the group
Person Records	Get/Set	Birthday	Birth date
Person Records	Get/Set	Company Name	Company name
Person Records	Get/Set	Department	Department number
Person Records	Get/Set	First Name	First name
Person Records	Get/Set	Job Title	Job title
Person Records	Get/Set	Last Name	Last name
Person Records	Get/Set	Maiden Name	Maiden name
Person Records	Get/Set	Middle Name	Middle name
Person Records	Get/Set	Name Prefix	Title
Person Records	Get/Set	Name Suffix	Suffix
Person Records	Get/Set	Nickname	Nickname
Person Records	Get/Set	Note	Notes
Person Records	Get/Set	Person Flags	"ShowAsCompany" or "ShowAsPerson"
Person Records	Get/Set	Phonetic First	Phonetic representation of the first name
Person Records	Get/Set	Phonetic Last	Phonetic representation of the last name
Person Records	Get/Set	Phonetic Middle	Phonetic representation of the middle name
Person Records	Get/Set	Lunar Birthday	None Gregorian calendar Birthdate (IE: Chinese, Hebrew, or Islamic)

#### Multi-Valued Properties

Multi-valued properties can have more than one value. Use PCAB\_AddMV function to set these properties and PCAB\_GetLabelForUID and PCAB\_GetValueForUID to get the property label and property.

Module	Method	Field Name	Acceptable Values/Notes
Multi-value	Get/Set	Address	Physical address
Multi-value	Get/Set	Dates	Dates associated with a person
Multi-value	Get/Set	E-Mail	E-mail addresses
Multi-value	Get/Set	InstantMessage	Instant message services
Multi-value	Get/Set	SocialProfile	Social profile services
Multi-value	Get/Set	Web Pages	Web pages
Multi-value	Get/Set	Phone	Generic phone number
Multi-value	Get/Set	Related Names	Names of people related to a person

## IV. AVAILABLE KEY VALUES AND COMPARISON OPERATORS

### Available Key Values

Use PCAB\_AddAddress function to set an address with the values specified below. Use PCAB\_GetLabelForUID( "Address" ; "SomeUID" ) and PCAB\_GetValueForUID( "Address" ; "SomeUID" ; "SomeValueBelow" ) to get the address label and address. These key values are also used in the "Key" parameter of the PCAB\_Search function.

Module	Method	Field Name	Acceptable Values/Notes
Keys	Get/Set	Street	Street portion of the address
Keys	Get/Set	City	City portion of the address
Keys	Get/Set	State	State/Province portion of the address
Keys	Get/Set	Zip	Postal code portion of the address
Keys	Get/Set	Country	Country portion of the address
Keys	Get/Set	Country Code	Two-letter ISO code for the country
Keys	Get/Set	InstantMessageService	Instant message service name (i.e. "AOL", "Yahoo", "Gadu-Gadu", etc.)
Keys	Get/Set	InstantMessageUsername	Instant message service username
Keys	Get/Set	SocialProfileService	Social profile service name (i.e. "Facebook", "LinkedIn", "Myspace", etc.)
Keys	Get/Set	SocialProfileUsername	Social profile service username
Keys	Get/Set	SocialProfileURL	URL link to the contact's social profile
Keys	Get/Set	SocialProfileUserId	Social profile service user ID

ISO codes: [http://developer.apple.com/mac/library/documentation/UserExperience/Reference/AddressBook/Classes/ABPerson\\_Class/Reference/CountryCodes.html#//apple\\_ref/doc/uid/TP40003945-RH2](http://developer.apple.com/mac/library/documentation/UserExperience/Reference/AddressBook/Classes/ABPerson_Class/Reference/CountryCodes.html#//apple_ref/doc/uid/TP40003945-RH2)

## Available Comparison Operators

These comparison operators are used to indicate the type of comparison to make in the “Comparison” parameter of the PCAB\_Search function.

Value	Meaning
Equal	Search for elements that are equal to the value
NotEqual	Search for elements that are not equal to the value
NotEqualCaseInsensitive	Search for elements that are not equal to the value, ignoring case
LessThan	Search for elements that are less than the value
LessThanOrEqual	Search for elements that are less than or equal to the value
GreaterThan	Search for elements that are greater than the value
GreaterThanOrEqual	Search for elements that are greater than or equal to the value
EqualCaseInsensitive	Search for elements that are equal to the value, ignoring case
ContainsSubString	Search for elements that contain the value
ContainsSubStringCaseInsensitive	Search for elements that contain the value, ignoring case
PrefixMatch	Search for elements that begin with the value
PrefixMatchCaseInsensitive	Search for elements that begin with the value, ignoring case
SuffixMatch	Search for elements that end with the value
SuffixMatchCaseInsensitive	Search for elements that end with the value, ignoring case
BitsInBitFieldMatch	Search for elements that match the bits in ABPersonFlags
DoesNotContainSubString	Search for elements that do not contain the value
DoesNotContainSubStringCaseInsensitive	Search for elements that do not contain the value, ignoring case
WithinIntervalAroundToday	Search for elements that are within a time interval (in seconds) forward or backward from today
WithinIntervalAroundTodayYearless	Search for elements that are within a time interval (in seconds) forward or backward from this day in any year.
NotWithinIntervalAroundToday	Search for elements that are not within a time interval (in seconds) forward or backward from today
NotWithinIntervalAroundTodayYearless	Search for elements that are not within a time interval (in seconds) forward or backward from this day in any year
WithinIntervalFromToday	Search for elements that are within a time interval (in seconds) forward from today
WithinIntervalFromTodayYearless	Search for elements that are within a time interval (in seconds) forward from this day in any year
NotWithinIntervalFromToday	Search for elements that are not within a time interval (in seconds) forward from today
NotWithinIntervalFromTodayYearless	Search for elements that are not within a time interval (in seconds) forward from this day in any year

## V. CONTACT US

Successful integration of a FileMaker plug-in requires the creation of integration scripts within your FileMaker solution. A working knowledge of FileMaker Pro, especially in the areas of scripting and calculations is necessary. If you need additional support for scripting, customization, or setup (excluding registration) after reviewing the videos, documentation, FileMaker demo and sample scripts, then please contact us via the avenues listed below.

Phone: 760-510-1200

Email: [support@productivecomputing.com](mailto:support@productivecomputing.com)

Forum: [www.productivecomputing.com/forum](http://www.productivecomputing.com/forum)

Please note assisting you with implementing this plug-in (excluding registration) is billable at our standard hourly rate. We bill on a time and materials basis billing only for the time in minutes it takes to assist you. We will be happy to create your integration scripts for you and can provide you with a free estimate if you fill out a Request For Quote (RFQ) at [www.productivecomputing.com/rfq](http://www.productivecomputing.com/rfq). We are ready to assist and look forward to hearing from you!