



Productive  
Computing



## Functions Guide

Revised July 27, 2018

## Table of Contents

950 Boardwalk, Suite 205, San Marcos, CA 92078 • (760) 510-1200 • [www.productivecomputing.com](http://www.productivecomputing.com)

© Copyright 2013 Productive Computing, Inc.

<b>I. INTRODUCTION.....</b>	<b>3</b>
<b>II. FUNCTION DESCRIPTIONS .....</b>	<b>4</b>
A. Registration and Other Functions.....	4
PCIM_GetLastError( format ) .....	4
PCIM_GetOperatingMode.....	4
PCIM_Register( server ; port ; page ; licenseID ).....	5
PCIM_Version( type ) .....	5
B. Practical Functions .....	6
PCIM_CalOpenCalendarWithTitle( title ).....	6
PCIM_EventsFetch( StartDate ; EndDate ; Calendars ; uid ).....	6
PCIM_GetItemCountForType( type ) .....	6
PCIM_GetProperty( type ; property ; optSubProp ) .....	7
PCIM_NewItem( type ).....	7
PCIM_OpenItemAt( type ; index ).....	7
PCIM_OpenItemWithUID( type ; uid ; optOccurrence ).....	8
PCIM_RemoveItem( type ; uid ; occurrenceDate ; span ) .....	8
PCIM_SaveItem( type ; optSpan ) .....	9
PCIM_SetProperty( type ; property ; value ).....	9
PCIM_RemindersFetch( calendars ; status ; date ).....	9
C. Recurrence Related Functions .....	10
PCIM_AddRecurrence( type ; interval ; daysotw ; daysotm ; monthsoty ; endDate ; endCount ) .....	10
PCIM_RemoveRecurrence .....	10
D. Alarm Related Functions .....	11
PCIM_AddAlarm( type ; abstrigger ; action ; eaddr ; reltrigger ; sound ; url ) .....	11
PCIM_RemoveAlarm( type ; index ).....	11
PCIM_RemoveAllAlarmsFrom( type ) .....	12
<b>III. APPENDIX - AVAILABLE PROPERTIES.....</b>	<b>13</b>
Table 1: Available Calendar Properties .....	13
Table 2: Available Event Properties .....	14
Table 3: Available Reminder Properties .....	15
Table 4: Available Alarm Properties.....	16
Table 5: Available Attendee Properties .....	16
Table 6: Available Recurrence Rule Properties .....	17
Table 7: Available Recurrence End Sub-Properties .....	18
<b>IV. CONTACT US .....</b>	<b>19</b>

## I. Introduction

### Description

The iCal Manipulator plug-in offers functions that support a bi-directional data exchange between FileMaker® Pro and Apple® Calendar. With this plug-in FileMaker users are able to push new records into Apple Calendar, pull records from Apple Calendar, and modify existing records in Apple Calendar. 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 Guide
- 3) Download a demo: [http://www.productivecomputing.com/dl/iCal\\_Manipulator/iCalManip.zip](http://www.productivecomputing.com/dl/iCal_Manipulator/iCalManip.zip)
- 4) Watch video tutorials: <http://productivecomputing.com/video/>
- 5) Familiarize yourself with Apple Calendar and Reminders

## II. Function Descriptions

This section describes the functions that are available with the iCal Manipulator plug-in.

### A. Registration and Other Functions

#### PCIM\_GetLastError( format )

**Description:**

Returns either a verbose description of the last error or the number of the last error.

**Parameters:**

format - "Number" causes the error number to be returned, otherwise a verbose description (string) is returned.

**Return Values:**

The last error string or number.

#### PCIM\_GetOperatingMode

**Description:**

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

**Parameters:**

None.

**Returns:**

"UNREGISTERED," DEMO," "LIVE" or "EXPIRED." Licensed plug-ins will be running in LIVE mode.

### **PCIM\_Register( server ; port ; page ; licenseID )**

#### **Description:**

Registers the plug-in with Productive Computing's registration servers. Must call Register function once before using the plug-in.

#### **Parameters:**

server = "licensing.productivecomputing.com"  
port = "80"  
page = "/PCIReg/pcireg.php"  
licenseID = "your license ID" or "DEMO-IM2"

#### **Return Values:**

0 for success, anything other than 0 is an error

### **PCIM\_Version( type )**

#### **Description:**

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

#### **Parameters:**

type = "Short" or "Long" . The default version string returned is "short". The short version string includes only the version number of the plug-in. For example: "2.0.0.0". The long includes the plug-in name and version number. For example: "iCal Manipulator 2.0.0.0"

#### **Returns:**

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

## B. Practical Functions

### PCIM\_CalOpenCalendarWithTitle( title )

**Description:**

Opens the specified calendar for editing.

**Parameters:**

title - The name of the calendar to open.

**Return Values:**

0 for success, otherwise an !!ERROR!!

### PCIM\_EventsFetch( StartDate ; EndDate ; Calendars ; uid )

**Description:**

Creates a "found set" of events in Apple Calendar which matches the parameters. It is highly recommended that the max range between the StartDate and EndDate is four years.

**Parameters:**

StartDate - A date indicating items with a date later than or equal to this will be "found".

EndDate - A date indicating items with a date earlier than or equal to this will be "found".

Calendars - A list of calendar names to be searched. The list is a FileMaker formatted list (return delimited).

uid - Indicates the desired found set only include items that belong to a specific recurring event.

**Return Values:**

0 for success, otherwise an !!ERROR!!

### PCIM\_GetItemCountForType( type )

**Description:**

Returns the number of available items of the specified type.

**Parameters:**

type - calendar, event, reminder, attendee, alarm.

**Return Values:**

The count of available items or an !!ERROR!!

### **PCIM\_GetProperty( type ; property ; optSubProp )**

#### **Description:**

Retrieves the specified property for the currently opened item of the specified type.

#### **Parameters:**

type - calendar, event, reminder, alarm, attendee, recurrenceRule, or recurrenceEnd.

property - One of the available properties for the currently opened item. See "Appendix" for available property names for each type.

optSubProp - Reserved for properties of recurrence end properties. To be used if property parameter is "recurrenceEnd." Values are usesEndDate, occurrenceCount, or endDate.

#### **Return Values:**

The property value or an !!ERROR!!

### **PCIM\_NewItem( type )**

#### **Description:**

Creates a new item of the specified type.

#### **Parameters:**

type - calendar, event, or reminder.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **PCIM\_OpenItemAt( type ; index )**

#### **Description:**

This function is used in conjunction with PCIM\_GetItemCountForType. Opens the item of specified type at the specified index. Once opened the item may be read and/or written. Note that attendees are read only.

#### **Parameters:**

type - calendar, event, reminder, attendee, or alarm.

index - The one based index of the Apple Calendar item (one-based index means that the first item is identified by a 1 instead of a 0 ).

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

## **PCIM\_OpenItemWithUID( type ; uid ; optOccurrence )**

### **Description:**

Opens the item of the specified type with the specified uid (unique identifier). For recurring items the optOccurrence value indicates the date of the recurrence.

### **Parameters:**

type - calendar, event or reminder.

uid - The unique identifier of the item.

optOccurrence - Optional parameter is specific to recurring event types. The parameter indicates the date of the recurrence to open for the specified item.

### **Return Values:**

0 for success, otherwise an !!ERROR!!

## **PCIM\_RemoveItem( type ; uid ; occurrenceDate ; span )**

### **Description:**

Removes or deletes the item of the specified type.

### **Parameters:**

type - calendar, event or reminder.

uid - The unique identifier for the item.

occurrenceDate - Required if the specified item is an event item. Indicates the date of the specific occurrence of the event to delete.

span - Required if the specified item is an event. Indicates which of the events should be deleted. Valid values are "this," "future" and "all." "This" will affect only the current item in the recurrence pattern. "Future" will affect all future items in the recurrence pattern. "All" will affect all items in the recurrence pattern.

### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **PCIM\_SaveItem( type ; optSpan )**

#### **Description:**

Saves item or commits the changes to the currently opened item of the specified type.

#### **Parameters:**

type - calendar, event or reminder.

optSpan - For recurring items only. Determines which set of the recurrence should be changed. Valid values are "this," "future" and "all." "This" will affect only the current item in the recurrence pattern. "Future" will affect all future items in the recurrence pattern. "All" will affect all items in the recurrence pattern.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **PCIM\_SetProperty( type ; property ; value )**

#### **Description:**

Sets the specified property for the currently opened item of the specified type.

#### **Parameters:**

type - calendar, event, reminder, or alarm.

property - The property to set. See "Appendix" for available property names for each type.

value - The value to which the property should be set.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **PCIM\_RemindersFetch( calendars ; status ; date )**

#### **Description:**

Creates a "found set" of reminders in Apple Calendar which match the parameters.

#### **Parameters:**

calendars - A list of calendar names to be searched. The list is a FileMaker formatted list (return delimited).

status - A string indicating the type of the reminders to be "found." Acceptable values are "complete" or "incomplete." May also be left empty to return all complete or incomplete reminders.

date - A date indicating reminders with either a due date before this date (status is incomplete) OR a completed after this date (status is complete). A date is required when status is set to "complete."

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

## C. Recurrence Related Functions

### PCIM\_AddRecurrence( type ; interval ; daysotw ; daysotm ; monthsoty ; endDate ; endCount )

#### **Description:**

Adds a recurrence rule to the currently opened item. The recurrence rules will be set according to the parameters passed to the function. See the documentation for valid parameter combinations.

#### **Parameters:**

type - daily, weekly, monthly, or yearly.

interval - A number indicating the period measured in days, weeks, months, or years between occurrences. Period is determined by the rrtype.

daysotw - A list of numbers indicating the days of the week on which the occurrences will occur. Used only with weekly recurrence rules. Valid values are 1 thru 7.

daysotm - A list of numbers indicating the days of the month on which the occurrences will occur. Used only with monthly recurrence rules. Valid values are 1 thru 31. The parameter is ignored if nthweekday has a value.

monthsoty - A list of numbers where each number is an integer value ranging from 1-12 representing the month of a year. Used only with yearly recurrence rules.

endDate - A date indicating the day the recurrence should end. May be empty.

endCount - The number of times a recurring event should repeat. Valid values are greater than 0. Used only if endDate is empty. May be empty. If both endDate and endCount are empty, then the recurrence never ends.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

### PCIM\_RemoveRecurrence

#### **Description:**

Removes the recurrence rule from the currently opened event item. Please use the PCIM\_SaveItem function to commit the item and reset the recurrence rule.

#### **Parameters:**

None.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

## D. Alarm Related Functions

### **PCIM\_AddAlarm( type ; abstrigger ; action ; eaddr ; reltrigger ; sound ; url )**

#### **Description:**

Adds an alarm to the currently opened event or reminder.

#### **Parameters:**

type - event or reminder.

abstrigger - A timestamp indicating the date and time to trigger the alarm. If this value is set, then all other triggers are reset.

action - The action that the alarm should perform. Possible values are "display," "email," "procedure" and "sound." Please see "Available Alarm Fields" section for further descriptions and actions of setting these values.

eaddr - If the action is email, then this is the address that the email will be sent to.

reltrigger - The number of seconds before or after the event or reminder date from which the alarm will be triggered. If this value is set, then all other triggers are reset.

sound - The name of an application or system sound.

url - The url to open when the alarm triggers. When you set the url property, then the action property is set to procedure, and the email and sound properties are set to nil.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **PCIM\_RemoveAlarm( type ; index )**

#### **Description:**

Removes the specified alarm from the currently opened item of the specified type. To obtain the number of alarms for an item, open the item and use PCIM\_GetItemCountForType( "alarm" ).

#### **Parameters:**

type - event or reminder.

index - The one-based index of the alarm.

#### **Return Values:**

0 for success, otherwise an !!ERROR!!

## **PCIM\_RemoveAllAlarmsFrom( type )**

### **Description:**

Removes all alarms from the item type specified. Requires a call to PCIM\_OpenItemWithUID or PCIM\_OpenItemAt as you must know what event or reminder you are working with before you can delete all the alarms.

### **Parameters:**

type - event or reminder.

### **Return Values:**

0 for success, otherwise an !!ERROR!!

### III. Appendix - Available Properties

This section contains a list of available properties. Each item in Apple Calendar contain various properties. These properties may be read or written to. Typically in order to read or "get" an Apple Calendar property value, you call PCIM\_GetProperty( type ; property ; optSubProp ). In order to write or "set" an Apple Calendar property, you typically call PCIM\_SetProperty( type ; property ; value ). However, some areas have a specific function to set these properties. This will be outlined in each are below. Please note that property names typically ARE case sensitive.

**Table 1: Available Calendar Properties**

Use PCIM\_GetProperty( type ; property ) and PCIM\_SetProperty( type ; property ; value ) to read (get) and write (set) these properties.

Method	Type	Property Name	Description
Get	calendar/ reminderlist	allowsmodification	Returns "true" if the you can add, modify, or delete calendar items or "false" if calendar items can't be modified.
Get/Set	calendar/ reminderlist	color	The color of the calendar. The value of this property is a string representing a generic RGB color along with its transparency. The format of the string is "R G B" where R is the red component, G is the green component, and B is the blue component. R, G, and B take values between 0 and 255 inclusive. Example color strings: red = "255 0 0" or green = "0 255 0" or silver = "204 204 204" etc.
Get	calendar/ reminderlist	iseditable	Returns "true" if the calendar's properties can be edited or "false" if the calendar's properties cannot be edited.
Get	calendar/ reminderlist	issubscribed	Returns "true" if the calendar is subscribed or "false" if the calendar is not subscribed.
Get	calendar/ reminderlist	notes	Textual notes about the calendar.
Get/Set	calendar/ reminderlist	title	The title of the calendar.
Get	calendar/ reminderlist	type	The type of calendar. Note that only Local calendars can be created. Possible returned values are: Birthday, CalDAV, Exchange, Local, or Subscription
Get	calendar/ reminderlist	uid	The unique identifier for the Calendar.

**Table 2: Available Event Properties**

Use `PCIM_GetProperty( type ; property )` and `PCIM_SetProperty( type ; property ; value )` to read (get) and write (set) these properties.

Method	Type	Property Name	Description
Get	event	attendeecount	The number of attendees for this event.
Get/Set	event	enddate*	The end date and time for the event.
Get	event	externalUID	The external unique identifier; this is common across multiple machines
Get	event	hasalarms	True if there are any alarms for this item
Get	event	hasattendees	True if there are attendees associated with this event
Get/Set	event	isallday	True if this is an all day event, otherwise false.
Get	event	isdetached	Returns whether or not an event is detached.
Get	event	isrecurring	True if the event has a recurrence rule, it is a recurring event.
Get/Set	event	location	A description of the location of the event.
Get	event	ModifiedDate	Last modification timestamp of the event
Get/Set	event	notes	The notes about the item.
Get	event	occurrence	The occurrence date of an event.
Get/Set	event	startdate*	The start date and time for the event.
Get/Set	event	title	The title of the event item.
Get	event	uid	The event item's unique identifier.
Get/Set	event	url	The URL for the calendar item.

\*When setting the "startDate" "endDate," or "isAllDay" fields, then the fields will be adjusted accordingly to Apple Calendar's rules. The value of one field can override and change the value of another field in Apple Calendar regardless of what has been scripted in FileMaker. For example, in FileMaker if you set the "isAllDay" field to True AFTER you set the "startDate" and "endDate" fields, then the event will be marked as an all day event without setting the specific start and end date and times as the event is now an all day event.

Since our plug-in adheres to rules and functionality of Apple Calendar, please be aware of the design and order you set these fields in your FileMaker script steps. We recommend setting these fields in the order shown below, however the field order of the script steps should be carefully designed and tested by a developer according to your solution and use.

- `PCIM_SetProperty( "event" ; "startdate" ; SomeStartDateValue )`
- `PCIM_SetProperty( "event" ; "endeate" ; SomeEndDateValue )`
- `PCIM_SetProperty( "event" ; "isallday" ; SomeAllDayEventValue )`

**Table 3: Available Reminder Properties**

Use `PCIM_GetProperty( type ; property )` and `PCIM_SetProperty( type ; property ; value )` to read (get) and write (set) these properties.

Method	Type	Property Name	Description
Get/Set	reminder	calendartitle	The title of the calendar that this item belongs to.
Get/Set	reminder	calendaruid	The uid of the calendar that this item belongs to.
Get/Set	reminder	completeddate	The reminder's completed date. If you set <code>completeddate</code> to nil, then <code>iscompleted</code> is set to False. If you set <code>completeddate</code> to a valid date, then <code>iscompleted</code> is set to true. Place <code>completeddate</code> AFTER <code>iscompleted</code> field in FileMaker script steps.
Get/Set	reminder	duedate	The due date and time for the reminder.
Get	reminder	hasalarms	True if there are any alarms for this item.
Get/Set	reminder	iscompleted	True if the reminder is completed, otherwise false. If you set <code>iscompleted</code> to true, then <code>completeddate</code> is set to the current date. If you set <code>iscompleted</code> to false, then <code>completedDate</code> is set to nil. Place <code>iscompleted</code> BEFORE <code>completeddate</code> field in FileMaker script steps.
Get	reminder	nextalarmdate	Returns the next alarm date for the reminder.
Get/Set	reminder	notes	The notes about the item.
Get/Set	reminder	priority	The priority of this reminder. Acceptable values are: none, high, medium, low.
Get/Set	reminder	title	The title of the reminder item.
Get	reminder	uid	The calendar item's unique identifier.
Get/Set	reminder	url	The URL for the calendar item.

**Table 4: Available Alarm Properties**

Use PCIM\_GetProperty( type ; property ) to read (get) these properties. Use PCIM\_AddAlarm( type ; abstrigger ; action ; eaddr ; reltrigger ; sound ; url ) function to add (set) alarm properties.

Method	Type	Property Name	Description
Get/Set	alarm	absoluteTrigger	The date and time the alarm will be triggered. When this property is set the relativeTrigger is set to nil.
Get/Set	alarm	action	The action to take when triggering the alarm. Valid Values are: display, email, sound.
Get/Set	alarm	email	An email address that is the recipient of an alarm that triggers an email message. An email message should be sent when the alarm triggers.
Get/Set	alarm	relativeTrigger	The number of seconds before the event to trigger the alarm.
Get/Set	alarm	sound	The system name of a sound to play when the alarm is triggered. A sound should be played when the alarm triggers. Value should be the name of a system sound. Sets the url and email properties to nil.
Get/Set	alarm	url	The URL to open when the alarm triggers.

**Table 5: Available Attendee Properties**

Use PCIM\_GetProperty( type ; property ) to read (get) these properties.

Method	Type	Property Name	Description
Get	attendee	address	A URL that can be used to contact the attendee.
Get	attendee	commonname	The user-entered name of the attendee.
Get	attendee	status	The attendees confirmation status. Possible Values: Needs Action, Accepted, Declined, Tentativedate.

**Table 6: Available Recurrence Rule Properties**

Use PCIM\_GetProperty ( type ; property ; optSubProp ) to read (get) these properties. Use PCIM\_AddRecurrence( type ; interval ; daysotw ; daysotm ; monthsoty ; enddate ; endcount ) to write (set) recurrence properties.

Method	Type	Property Name	Description
Get/Set	recurrenceRule	daysofthemonth	If the recurrenceType property is monthly, a list of integer values ranging from 1 to 31 representing days of the month in the pattern; otherwise, nil.
Get/Set	recurrenceRule	daysoftheweek	If the recurrenceType property is weekly, a list of integer values ranging from 1 to 7 representing days of the week in the pattern.
Get	recurrenceRule	firstdayoftheweek	This property affects the way the recurrence is expanded for weekly recurrence rules with an interval greater than 1. For these types of recurrence rules, Calendar Store sets this property to 2, for all other recurrence rules this property defaults to 0.
Get/Set	recurrenceRule	monthsoftheyear	If the recurrenceType property is yearly, a list of the months as integer values ranging from 1 to 12 representing the month of a year that the event occurs.
Get/Set	recurrenceRule	recurrenceinterval	The number of intervals between the specified pattern of a recurring event. The actual time between a pattern depends on the value of the recurrenceType property.
Get/Set	recurrenceRule	recurrencetype	The unit of measurement between occurrences of a recurring event. daily weekly monthly yearly
Get	recurrenceRule	hasend	Returns true if the recurrence rule has an end; otherwise returns false.
Get	recurrenceRule	recurrenceend	Returns 'never' if there is no end. Otherwise use the optSubProp parameter to get the recurrence end value. See Table 7 below.

**Table 7: Available Recurrence End Sub-Properties**

Use `PCIM_GetProperty( type ; property ; optSubProp )` to read (get) these properties.

Method	Type	Property	optSubProp	Description
Get	recurrencerule	recurrenceEnd	usesenddate	Indicates whether the endDate property is used to end the Recurrence Rule, or if the occurrenceCount is used to end the Recurrence Rule. Returns true if there is an endDate for the recurrence end.
Get	recurrencerule	recurrenceEnd	enddate	Indicates after a specific date the Recurrence Rule should end. The end date if the recurrence uses an end date value.
Get	recurrencerule	recurrenceEnd	occurrencecount	The occurrence count—number of times an event repeats—if the recurrence uses a number value; 0 if it uses an end date.

## IV. 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!