

active data



Calendar Event Import
User Guide

Active Data Calendar version 3.14.7

Active Data Calendar is a trademarked product of Active Data, Inc.
All other brand and product names mentioned in this guide are the trademarks, registered trademarks, or trade names of their respective holders.

© Copyright 2014
Active Data, Inc.
190 Brodhead Road, Suite 300
Bethlehem, PA 18017

All rights reserved. No part of this publication may be reproduced without the prior written consent of Active Data, Inc., with the exception of copies made for your internal office use.

Document Control

Doc/Prod Version	Description	Date	Author	Status
0.02/3.14	Addressed first-round review comments. New user guide template applied.	3/25/2013	David Wandelt	Draft
1.0/3.14.x	Addressed final review comments, including updates to field descriptions table and multiple iCalendar	6/6/2013	David Wandelt	Released
1.1/3.14.1	Added stepwise instructions for adding a persistent import. Reworked descriptions of suppressing event owner and subscriber emails. Reworked descriptions of recurrence options.	10/10/13	David Wandelt	Released
1.2/3.14.7	Updated XML for Output / Input formats and descriptions of new nodes. Added Country, State, and County validation.	12/1/2014	Gregory E. Oppenheimer	Released
1.2.1 / 3.14.7	Updated Import Update process to clarify fields needed to match records and avoid duplication.	3/6/15	Gregory E. Oppenheimer	Released

Contents

Active Data Calendar's event importing capabilities.....	6
Creating files to import.....	6
Supported formats	6
Purging	6
Data validation	6
Importing large files	7
One-time import procedure (small files only)	7
Creating a one-time import using the persistent import interface.....	7
Setting an Import Update Schedule.....	8
Important Note.....	8
Example 1: Quick completion	8
Example 2: Run-time overrun.....	9
Example 3: Multiple persistent imports	10
Creating a Persistent Import.....	12
Persistent Import Options	12
Facilities Module fields.....	13
No Facilities Module	14
Importing CSV	14
CSV guidelines	15
Importing XML	16
Import fields (CSV and XML).....	16
Sample files	16
CSV and XML field specifications	17
Field descriptions.....	20
<i>General: date and time formats.....</i>	<i>20</i>
<i>Event Description field</i>	<i>20</i>
<i>Department Name field</i>	<i>21</i>
<i>Categorization field.....</i>	<i>21</i>
<i>Private Flag</i>	<i>22</i>
<i>Highlight flag</i>	<i>22</i>
<i>Facilities field (CSV) and Locations structure (XML)</i>	<i>22</i>
<i>Room element (XML only).....</i>	<i>24</i>
<i>Room Link field.....</i>	<i>25</i>
<i>Internal Comments field.....</i>	<i>25</i>
<i>Start Date, Start Time, End Date and End Time fields.....</i>	<i>25</i>
<i>Recur Type field.....</i>	<i>26</i>
<i>Recur Days field.....</i>	<i>26</i>
<i>Recurring End Date field</i>	<i>27</i>
<i>Address fields</i>	<i>27</i>
<i>Location Url field.....</i>	<i>27</i>
<i>Import Series Id field</i>	<i>27</i>
<i>Import Occurrence Id field</i>	<i>27</i>
<i>Created On and Modified On fields.....</i>	<i>27</i>

<i>Room field elements (CSV)</i>	29
<i>Information Status field</i>	29
<i>Setup Notes field</i>	30
<i>User field elements</i>	30
<i>Internal Custom <n> fields</i>	31
<i>County fields</i>	31
<i>State fields</i>	31
<i>Country fields</i>	31
<i>Registration fields</i>	31
<i>Resource fields</i>	32
<i>External Series ID and External Occurrence ID fields</i>	35
<i>Event Owner field</i>	36
<i>External Import ID field</i>	36
<i>Event Owner Name field</i>	36
<i>DateUp element (XML Only)</i>	36
<i>DateDown element (XML Only)</i>	36
<i>HideEndDateTime element (XML Only)</i>	36
<i>EventGUID element (XML Only)</i>	36
<i>AttachmentBinary and ImageBinary fields (XML Only)</i>	37
Sample XML and CSV Import Files	37
Update Keys	37
Importing iCalendar files	38
iCalendar fields	38
iCalendar field descriptions	39
<i>BEGIN</i> property	39
<i>PRODID</i> property	39
<i>SUMMARY</i> property	39
<i>DTSTAMP</i> property	39
<i>DTSTART</i> property	39
<i>DTEND</i> property	40
<i>DESCRIPTION</i> property	40
<i>RESOURCES</i> property	40
<i>LOCATION</i> property	40
<i>CATEGORIES</i> property	40
<i>RRULE</i> property	40
<i>UID</i> property	40
<i>LAST-MODIFIED</i> property	41
<i>END</i> property	41
Example iCalendar import content	41
Information that can't be imported	42
FAQs and/or troubleshooting tips	42
Duplicate records	42
Appendices	42
Country List	42

Active Data Calendar's event importing capabilities

Active Data Calendar's *Import Events* feature lets you place events into your calendar by reading a file into it. You can do this as a one-time occurrence, or set up a persistent (repeating) job to do it on a schedule.

The primary use of Calendar's event importing capability is ensuring that multiple systems that use the same event data remain in sync with each other. Persistent import jobs run periodically (at an interval that you set) to ensure that the difference (latency) between these systems remains at an acceptable level.

Creating files to import

While you can create a file containing events to be imported any text editor, it's usually advantageous to create your structured event information in a higher-level tool. Microsoft Excel, for example, can export data in CSV (comma-separated value) format, and will correctly generate CSV fields in cases that require double-quote delimiting, such as values containing embedded commas.

Most often, the data being imported are first exported from another program where it already exists, such as a database for another application that handles calendar information. Exporting the data this way can save a lot of rework, and duplicate creation of event records in multiple systems (with the risk of mismatches in the information).

Supported formats

Calendar import process input files can be of type CSV, XML or iCalendar¹. Each file type has specific content and formatting requirements. These are described in below, in the *Import fields (CSV and XML)* and *Importing iCalendar files* sections.

Purging

The Calendar's event "purge" functionality allows events to be deleted using and import file or persistent feed. This feature facilitates iterative testing of import files during development. The file that was imported is fed to the purge process, ensuring that all inserted data is removed.

Data validation

Unless otherwise noted in the master table below (see *CSV and XML field specifications*), text field contents are not validated by the import process. For example, any two-letter combination is accepted for the State parameter, even though the value should be one of the standard values. Validation is the responsibility of the import file's creator. Please take care to provide valid

¹ Note that although conventionally referred to as "iCal", the IETF's RFC 5545 that defines it is careful to always call it "iCalendar". iCal® is a registered trademark of Apple Inc., for its calendar app.

data; wrong values for a registrant's address, for example, could result in incorrect sales tax calculations.

Importing large files

Files containing more than 1000 event occurrences are considered "large" files. Large files must be imported by creating a one-time import in the *persistent* event feed process, rather than the *one-time event import* process. This ensures that a long-running import job completes without timing out.

See *Creating a one-time import using the persistent import interface*, below.

One-time import procedure (small files only)

You can import files of not more than 1000 event occurrences using this one-time import procedure:

1. Log in to the Calendar's administrative area.
2. Click the [Add](#) link next to **IMPORT**:
3. Select the **One-Time Event Import** in the **Select an import option:** drop-down list.
4. Click the Browse button and locate your input file to upload.
5. Click the Open button.
6. Select the appropriate options on the rest of the page and click **Submit**.

Creating a one-time import using the persistent import interface

If your import file contains more than 1000 event occurrences, you can still import it just one time with the persistent import tool. Just create a one-time import by selecting the same date for the start and end dates, and specifying a start time, end time and import interval² that allow the import job to run only once. For example, here there are only 15 minutes between the beginning and ending times, so specifying an import interval of 30 minutes ensures that the import runs only once:

Start Date:	05/21/2012
End Date:	05/21/2012
Time to Begin Retrieving:	10:00 AM
Time to End Retrieving:	10:15 AM
Import Interval:	30 minutes

² *Import Interval* specifies how long the system waits from the successful completion of one import job run before kicking off the next run of the job, or the next (different) import job.

Note that if the time required for the import to run takes it past the “Time to End Retrieving” that you set, the job *does* complete its run. That 10:15 time to end just prevents it from running a second time; it does not prevent the completion of a job that has already started.

Setting an Import Update Schedule

This is the way to create a process that runs repeatedly, at times you specify. The Add Persistent Import dialog requires a *Start Time*, *End Time* and *Import Interval*, which tell the Calendar application when to import updates from the URL you specify. New import job runs can begin at the start time, and until the end time.

Active Data Calendar’s event import process is single-threaded, which affects the running of scheduled jobs you create, because only one import process can run at a time. If more than one import job is scheduled, each job starts only when the one before it completes.

It’s worth noting that your job’s actual running time will vary in connection with many factors, so a job that runs only two minutes on one occasion could run considerably longer another time. It’s wise to build some slack into your schedule if completion time is a critical dependency for you. The examples below assume a constant run time; “Your mileage may vary”!

Important Note

The *Import interval* setting is based on job completion, not job start. Therefore, when you enter times, you are not creating an actual run *schedule*, but rather, a sequence of job runs followed by a wait time between the end of one run, and the start of the next run of the same or another job. The examples below should help to clarify this concept.

Example 1: Quick completion

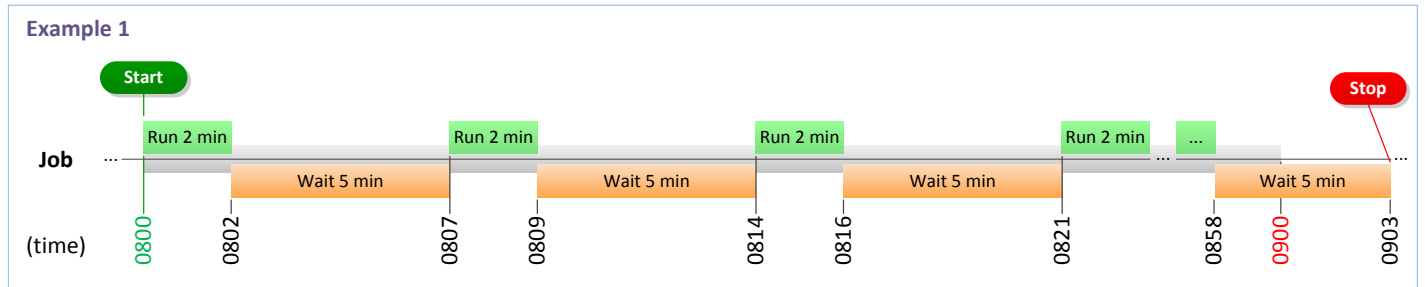
2-minute run time, 5-minute interval setting

Let’s say you set a start time of 08:00, end time of 09:00, and interval of 5 minutes, for a job you believe will run for about 2 minutes. You might think you’ve created a schedule that will run from 08:00 to 08:02, then 08:05 to 08:07, followed by 08:10 to 08:12, and so on, with a last run from 09:00 to 09:02.

But here’s what actually happens:

Suppose the run that starts at 08:00 does complete in just 2 minutes, as you expect. At 08:02, the 5-minute interval timer begins counting. At 08:07, the next 2-minute run starts. That second run completes at 08:09, the timer counts until 08:14, and begins the next run, and so on, until a the interval timer would start a new run some time after 09:00. The runs then cease, either completely (if you have reached your end date) or until the next day at 08:00. If you extend this example out, the last run ends at 08:58 and the interval timer stops counting at 09:03. Because that’s beyond your 09:00 stop time, no further runs take place on that day.

Here's what this basic example looks like, represented as a time line:



In any event, all your runs complete (though fewer iterations may have taken place than you expected). All the data get in, and everyone's happy.

Example 2: Run-time overrun

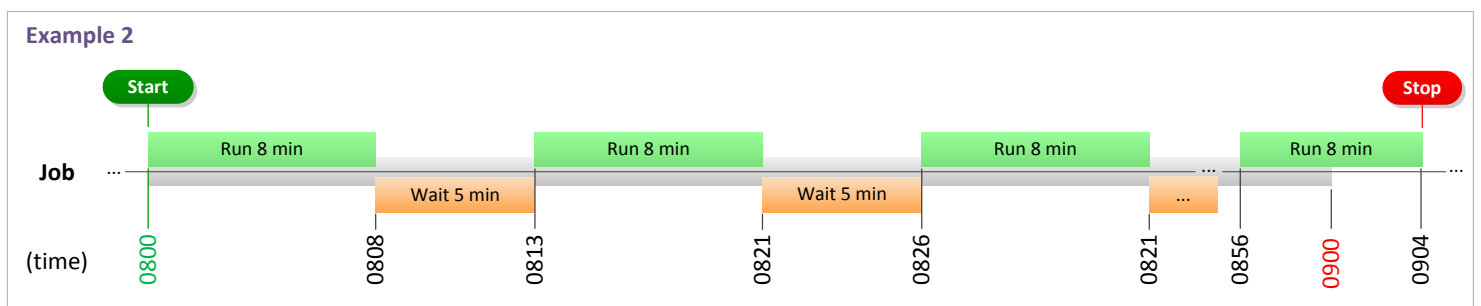
8-minute run time, 5-minute interval setting

In this example, you set a start time of 08:00, end time of 09:00, and interval of 5 minutes—but your job actually takes 8 minutes to run. If the second run kicked off at a hard-scheduled 08:05, 3 minutes before the first run ended, the results would be unpredictable at best—possibly catastrophic, if the records in the two imports started stepping on each other. That could create a real mess.

Now you see why the interval timer start is based on the run's completion time, not its start time. The timer begins counting to the next run at 08:08, counts for 5 minutes until 08:13, then kicks off the next run. The job runs until 08:21, and the timer begins counting again until 08:26, then kicks it off again, and so on. You don't have to accurately predict the job's run time, because run this way, the runs can never overlap. The next run will *not* start at 8:05, but at a time 5 minutes after the finish of the last run. That's why we programmed the logic as $[\text{finish time}] + [\text{interval}] = [\text{start time}]$.

The last run of that day is the one that starts at or before your end time (09:00), and finishes after the end time. You may get fewer runs this way, but they always import reliably.

Here is this example, shown as a time line:



And wonder of wonders, all the runs complete, all the data get in, and everyone's still happy.
Even the boss!

Example 3: Multiple persistent imports

Now here is where it gets a little tricky. Active Data Calendar allows you to create more than one persistent import job, but it runs only one job at a time. Here is how the system handles that situation.

Let's say you have two persistent jobs (it could be more than two), set up to run concurrently, as:

Job 1:

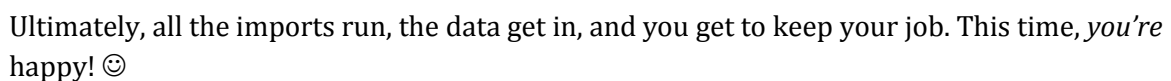
Time to Begin Retrieving:	08:00 AM
Time to End Retrieving:	08:30 AM
Import Interval:	5 minutes
Time required to complete import:	5 minutes

Job 2:

Time to Begin Retrieving:	08:00 AM
Time to End Retrieving:	09:00 AM
Import Interval:	10 minutes
Time required to complete import:	6 minutes

- 8:00 AM: Job 1 kicks off (Job 1 has priority because it was created first)
- 8:05 AM: Job 1 completes
- 8:05 AM: Job 2 kicks off
- 8:11 AM: Job 2 completes
- 8:11 AM: Job 1 kicks off
- 8:16 AM: Job 1 completes
- 8:21 AM: Job 1 kicks off (Job 1 has priority because it was created first)
- 8:26 AM: Job 1 completes
- 8:26 AM: Job 2 kicks off
- 8:32 AM: Job 1 wait time ends after a new run can begin, so Job 1 ends
- 8:42 AM: Job 2 kicks off
- 8:48 AM: Job 2 completes
- 8:58 AM: Job 2 kicks off (Job 1 was set to only poll until 8:30AM, so Job 2 continued in its own cycle until its end time of 9:00 AM)
- 9:04 AM: Job 2 ends its final run of this day.

Here is the time line for this more-complex example:



Creating a Persistent Import

To create a persistent import job:

1. Log in to the Calendar's back end.
2. Under the Events tab, click **IMPORT: Add**.
3. Click on **Select an import option** and choose Persistent Import Feed.
4. Give the persistent import a name by entering it in the **Import Name** box. This name is how the import will later be chosen from a list, so make it sufficiently descriptive.
5. Choose from the options listed in *Persistent Import Options*, below. Be sure to supply values for the required ones.
6. Click **SUBMIT** to place the import into the Calendar's library of persistent import jobs. If the Active option is set to Active, the job begins running within the dates and times you selected.

Important Note

When using the persistent Import feature, the file must be located on a hosted web location, accessible via a direct URL. Files housed on FTP servers, even with a direct URL will not be imported and are not supported.

Persistent Import Options

The persistent import process has the options and fields described below. Entries marked with an asterisk (*) in the list, as on the actual set-up page, are required.

- ***Import Name:** The name under which the import will be stored in the library of import jobs.
- ***Import Type:** The format of the file being imported. CSV, iCalendar and XML are supported.
- ***Import URL:** The uniform resource locator (URL) where the Calendar will find the file being imported. The format for this is **Error! Hyperlink reference not valid.>**. The file's extension should be **.csv**, **.ics** or **.xml**, as appropriate.
- **Username:** If the server at the Import URL requires a user name to access the file, enter it here.
- **Password:** If the server at the Import URL requires a password to access the file, enter it here.
- ***Assign Events to:** The user account to assign the imported events to as their owner.
- ***Disable Workflow Emails:** Setting this option to Yes suppresses sending the workflow e-mails normally associated with the event action being performed (add or modify).
- ***Disable Subscription/Notification Emails:** Setting this option to Yes suppresses e-mails normally sent to subscribers to the events being imported.

- ***Start Date:** The first date on which the import job should run.
- ***End Date:** The last date on which the import job should run.
- ***Time to Begin Retrieving:** The local time each day when the import job will start running to look for new event updates from this event feed.
- ***Time to End Retrieving:** The local time each day when the import job will stop running to look for new event updates from this event feed. Jobs begun before this time will complete.
- ***Import Interval:** Specifies how long the system waits from the successful completion of one import job run before kicking off the next run of the job, or the next (different) import job.
- ***Active:** Selecting Active tells the calendar to run this import job at the specified times, between and on the specified start and end dates.
- ***Department:** Select a department from the list if you want to apply that department name to events based on one of the next two options:
 - Apply the selected Department for event(s) where no Department is specified in the import file. Selecting this option places the department you chose from the **Department** list into the **Department** field for any event being imported with a blank **Department** field.
 - Apply Department Information Globally. Select this option to set the **Department** field in all events imported to the department you selected from the list, even if their **Department** field is not blank.
- ***Category/Subcategory:** Select a category or subcategory from the list if you want to apply that selection to events based on one of the next two options:
 - Apply the selected Category/Subcategory for all event(s) where no Category/Subcategory is specified in the import file. Selecting this option places your selection from the Category/Subcategory list into the **Categorization** field for any event being imported with a blank **Categorization** field.
 - Apply Category Information Globally. Select this option to set the **Categorization** field in all events imported to the category or subcategory you selected from the list, even if their **Categorization** field is not blank.

Facilities Module fields

Active Data Calendar offers an optional *Facilities Module* that tracks information about the locations and other facilities (including *resources* such as PA systems, computers, and video projectors) that relate to events. Clients that have the facilities module can provide the facilities-specific fields in their event import feed (for events that include facilities).

Facilities-specific fields include:

- Room Setup Name
- Room Setup Time

- Room Tear Down Time
- Room Capacity
- Setup Notes
- User Setup Time
- User Tear Down Time
- User Setup Count
- User Setup Notes
- Resources – CatSubcat
- Resources – Resource
- Resource – Quantity
- Resource – Notes

These fields are all described in the subsections under *Field descriptions*.

No Facilities Module

If your installation doesn't include the Facilities Module (FM), you are not *required* to provide these fields.

- Room Setup Name
- Room Setup Time
- Room Tear Down Time
- Room Capacity
- Setup Notes
- User Setup Time
- User Tear Down Time
- User Setup Count
- User Setup Notes
- Resources – CatSubcat
- Resources – Resource
- Resource – Quantity
- Resource – Notes

However, you may wish to send them anyway, if you anticipate purchasing the module at some future time. The data you supply will be stored in the database, and be immediately available once the FM is installed and running.

Importing CSV

CSV files have a very specific format, which must be followed exactly if the import operation is to succeed. The field names are listed in the block below, followed by some guidelines for making the import process easier. The table in the next section defines each field's format, maximum length, and allowable values.

Event Name,Event Description,Contact Name,Contact Phone,Contact
Email,Department Name,Categorization,Private Flag,Highlight,Facilities,Room
Link,Internal
Comments,ExternalField1,ExternalField2,ExternalField3,ExternalField4,All Day

Flag,Start Date,Start Time,End Date,End Time,Recur Type,Recur Days,Recurring
End Date,Address 1,Address 2,City,State,Zipcode,Phone,Location Url,Import
Series Id,Import Occurrence Id,Created On,Modified On,Room Setup Name,Room
Setup Time,Room Tear Down Time,Room Capacity,Information Status,Setup
Notes,User Setup Time,User Tear Down Time,User Setup Count,User Setup
Notes,Internal Custom 1,Internal Custom 2,Internal Custom 3,Internal Custom
4,County,Country,Registration - Enabled,Registration - Max
Registrants,Registration - Display Available,Registration - Type,Registration
- Template,Resources - CatSubcat,Resources - Resource,Resource -
Quantity,Resource - Notes,External Series Id,External Occurrence Id,Event
Owner,External Import ID,Event Owner Name

See the example file on the Client Portal's *Documents* page, named Active Data Calendar 3.14.7
Event Import CSV Sample.

CSV guidelines

When creating your CSV file, please observe these guidelines:

- Each field must always be accounted for by a comma, whether supplying a value, or as a null (appears as two adjacent commas).
- Supplied values must properly handle embedded commas and quotes. Encase values containing either double quotes or commas within double quotes. Further, use a pair of double quotes to represent one double quote character.

Some examples:

- Maybe a single field contains a first name, last name and suffix, such as *Edward Brown, Jr.*, with an embedded comma as shown. In CSV, this field appears in double quotes as ...,`"Edward Brown, Jr."`,...
- Or let's say that same field contains a nickname, such as *Eddie "screwball" Brown*. In this case, the CSV reads ...,`"Eddie ""Screwball"" Brown"`,...
- `Eddie 'screwball' Brown`, with the nickname in *single* quotes, doesn't require the surrounding double-quote characters. Its CSV representation is ...,`Eddie 'Screwball' Brown`,...
- A single letter in double quotes, `"A"`, looks strange in CSV, but it's actually correct, as: `""A""`
- Beware of "smart quotes." Data pasted into Excel from a program like Microsoft Word or Adobe® FrameMaker® will retain those non-ASCII "smart" characters, but they are not translated in CSV to their respective "dumb" ASCII equivalents (0x22 for double, 0x27 for single quotes). These characters appear in your event text fields as non-printable (□).
- To skip (omit) a value for a given field, your CSV file must still supply the comma that represents its place in the CSV record structure.

- Each record line must be terminated by a CRLF³ to be recognized as a valid record.
- The first record must be a header row, containing the above list of fields exactly as shown, without double quote characters surrounding the record.
- For more information on CSV format, please see <http://tools.ietf.org/html/rfc4180>

Importing XML

XML files have some flexibility in their format, in Calendar imports they do accommodate all of the fields that CSV supports, but in a node-and-name structure. XML also accommodates two additional fields, used for importing images and attachments, plus a few more (shown in the master table).

A reference example is provided on the Client Portal's *Documents* page, named Active Data Calendar 3.14.7 Event Import XML Sample.

Import fields (CSV and XML)

The table provided here lists all of the fields used for importing event (and related) information into Active Data Calendar, using either CSV or XML format files. Some additional elements are required, such as header information and time zone.

Important: *Values imported into Calendar must conform to the specifications for the database fields they are being placed into. This includes the field length, type, and allowable value(s). Deviations from this requirement may result in errors and/or rejected records or imports.*

Also, the CSV field and XML element and attribute names are case-sensitive. They must appear as shown in the table for the import to succeed.

Sample files

The *Documents* page of the Client Portal provides sample import XML and CSV files. These contain the complete set of elements required in a valid file. The sample files are called:

- Active Data Calendar 3.14.7 Event Import XML Sample
- Active Data Calendar 3.14.7 Event Import CSV Sample

³ CRLF: *carriage return, line feed* sequence. In ASCII, this is a hexadecimal 0x0D, 0x0A (decimal 13, 10).

CSV and XML field specifications

In this table:

- An asterisk (*) following a field name indicates that additional information is provided in the sections following the table.
- An asterisk (*) following a “N” in the “Required?” column indicates that it may be required in certain cases. See the corresponding additional information section for the explanation.
- CSV fields marked with “n/a” are either fields not supported in CSV files, or are XML structure elements that CSV doesn’t require. Please see the above-referenced example files in the Client Portal’s *Document* area for specifics.

CSV Field Name	XML Element or Attribute Name	Length	Format and Possible Values	Required?
n/a	EVENT	n/a	(top-level group element)	Y
Event Name	Name	100	Alphanumeric	Y
Event Description*	Description	8000	Alphanumeric	Y
Contact Name	ContactName	255	Alphanumeric	N
Contact Phone	ContactPhone	50	Alphanumeric	N
Contact Email	ContactEmail	255	Alphanumeric	N
Department Name*	Department	50	Alphanumeric	Y*
Categorization*	Categorization	50	Alphanumeric	Y*
Private Flag*	PrivateFlag	1	“Y”, “N”, “B”	Y
Highlight*	Highlight	1	“Y”, “N”	Y
Facilities*	Locations*	n/a	(group element)	N
	See Facilities field (CSV) and Locations structure (XML)			
n/a	Location	n/a	(group element)	N
(Subfield of Facilities)	LocationName	500	Alphanumeric	N
n/a	Latitude	15	Double (41.5543374456746 or -55555555.666415)	N
n/a	Longitude	15	Double (41.5543374456746 or -55555555.666415)	N
(see Setup Notes, below)	SetupNotes	255	Alphanumeric (gets linked to the lowest level of the facility.)	N
n/a	Building	n/a	(group element)	N
(Subfield of Facilities)	BuildingName	75	Alphanumeric	N
n/a	Latitude	15	Double (41.5543374456746 or -55555555.666415)	N
n/a	Longitude	15	Double (41.5543374456746 or -55555555.666415)	N

CSV Field Name	XML Element or Attribute Name	Length	Format and Possible Values	Required?
n/a	Room*	n/a	(group element)	N
(Subfield of Facilities)	RoomName	75	Alphanumeric	N
n/a	RoomDivision	n/a	(group element)	N
(Subfield of Facilities)	RoomDivisionName	75	Alphanumeric	N
Room Link*	roomlink (an attribute of the Room or RoomDivision group element)	265	Alphanumeric	N
n/a	LocationsOnTheFly	n/a	(group element)	N
n/a	Location	n/a	(group element)	N
n/a	Name	150	Alphanumeric	N
n/a	Address 1	100		
n/a	Address 2	100		
n/a	City	100		
n/a	State	100		
n/a	ZipCode	100		
n/a	County	225		
n/a	Country	225		
n/a	Latitude	15		
n/a	Longitude	15		
Internal Comments*	InternalComments	1000	Alphanumeric	N
ExternalField1	ExternalField1	500	Alphanumeric	N
ExternalField2	ExternalField2	500	Alphanumeric	N
ExternalField3	ExternalField3	500	Alphanumeric	N
ExternalField4	ExternalField4	500	Alphanumeric	N
All Day Flag	AllDay	1	"Y", "N"	Y
Start Date*	StartDate	n/a	"m/dd/yyyy" date	Y
Start Time*	StartTime	n/a	"h:mm AM" or "h:mm PM" time	Y
End Date*	EndDate	n/a	"m/dd/yyyy" date	Y
End Time*	EndTime	n/a	"h:mm AM" or "h:mm PM" time	Y
Recur Type*	RecurType	n/a	See below	Y
Recur Days*	RecurDays	n/a	See below	N*
Recurring End Date*	RecurEndDate	n/a	See below	N*
Address 1	Address1	50	Alphanumeric	N
Address 2	Address2	50	Alphanumeric	N
City	City	30	Alphanumeric	N
State	State	2	Alphanumeric	N
Zipcode	Zipcode	10	Alphanumeric	N
Phone	Phone	25	Alphanumeric	N
Location Url*	InternetAddress	110	Alphanumeric	N
Import Series Id*	ImportSeriesId	255	Alphanumeric	Y
Import Occurrence Id*	ImportOccurrenceId	255	Alphanumeric	Y
Created On*	CreatedOn	n/a	Valid date and time (see format below).	N*
Modified On*	ModifiedOn	n/a	Valid date and time (see format below).	N*

CSV Field Name	XML Element or Attribute Name	Length	Format and Possible Values	Required?
Room Setup Name*	roomsetupname (an attribute of the Room or RoomDivision group element)	50	Alphanumeric	N
Room Setup Time*	roomsetuptime (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
Room Tear Down Time*	roomteardowntime (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
Room Capacity*	roomcapacity (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
Information Status*	InformationStatus	1	Character	N
Setup Notes*	setupnotes (attribute of the Location group element above)	255	Alphanumeric	N
User Setup Time*	usersetuptime (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
User Tear Down Time*	userteardowntime (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
User Setup Count*	usersetupcount (an attribute of the Room or RoomDivision group element)	n/a	Integer > 0	N
User Setup Notes*	usersetupnotes (an attribute of the Room or RoomDivision group element)	255	Alphanumeric	N
Internal Custom 1	InternalCustom1	1000	Alphanumeric	N
Internal Custom 2	InternalCustom2	1000	Alphanumeric	N
Internal Custom 3	InternalCustom3	1000	Alphanumeric	N
Internal Custom 4	InternalCustom4	1000	Alphanumeric	N
County*	County	255	Alphanumeric	N
Country*	Country	255	Alphanumeric	N
Registration – Enabled*	RegistrationEnabled	1	“Y”, “N”	Y
Registration – Max Registrants*	RegistrationMaxReg	n/a	Integer	N*
Registration – Display Available*	RegistrationDisplaySpaceAvail	1	“Y”, “N”	N*
Registration – Type*	RegistrationType	7	“Single”, “Partial” or “Series”	N*
Registration – Template*	RegistrationTemplate	n/a	Alphanumeric	N*
n/a	Resources	n/a	(group element, zero or one per event)	N
n/a	Resource	n/a	(group element, one or more per Resource structure)	N

CSV Field Name	XML Element or Attribute Name	Length	Format and Possible Values	Required?
Resources – CatSubcat*	ResourceCat and ResourceSubCat	50	Alphanumeric (50 allowed for Category and 50 for Subcategory)	N*
Resources – Resource*	ResourceName	100	Alphanumeric	N
Resource – Quantity*	ResourceQuantity	n/a	Integer > 0	N*
Resource – Notes*	ResourceNotes	1000	Alphanumeric	N*
External Series Id*	ExternalSeriesId	255	Alphanumeric	N
External Occurrence Id*	ExternalOccurrenceId	255	Alphanumeric	N
Event Owner	EventOwner	n/a	Integer (validated against the Account table)	N
External Import ID	ExternalImportID	255	Alphanumeric	N
Event Owner Name	EventOwnerName	62	Alphanumeric (not used in import file.)	N
n/a	DateUp	n/a	m/d/yyyy date	N
n/a	DateDown	n/a	m/d/yyyy date	N
n/a	HideEndDateTime	1	Alphanumeric	N
n/a	EventGUID	36	Alphanumeric	N
n/a	ImageBinary	n/a	Binary or Reference Link	N
n/a	AttachmentBinary	n/a	Binary or Reference Link	N
n/a	EventURL	n/a	Alphanumeric	N

Field descriptions

These notes apply to the above table of CSV and XML values. Not all fields are described, as many are self-evident for those familiar with Active Data Calendar’s functions and operation.

General: date and time formats

Formats for times and dates in the table above are shown with only a single hour or month placeholder (that is, “h:mm” and “m/dd/yyyy”) if a leading zero is not required for single-digit hours in times, and single-digit months in dates.

Event Description field

The free-form text description of the event.

- Holds up to 15000 characters (including HTML markup, if any)⁴
- If omitted, the *Event Name* is copied to this field

⁴ Active Data, Inc does not recommend using Microsoft® Word® for composing your event descriptions, as the HTML it generates, even in a “filtered” file, is too large.

Department Name field

The department to which the event is being posted.

- If the value provided is not already in the Calendar database, a new department is created.
- When you create the import, you select how to process a missing department field value. You can specify:
 - That the import does not supply department information, and to leave the Department field blank in the imported records.
 - A department to be forced into all imported records, regardless of the contents of the Department field in the import file.
 - A department to be forced into only those imported records having a missing Department field value.

Important: *Department selections cannot be modified in individual occurrences of a series. This is true whether you are working directly in the Calendar application, or importing data using the processes described here.*

Categorization field

The category and subcategory for the event being posted.

- If the value provided is not already in the calendar database, the import process creates a new category.
- To specify subcategory, place a double colon (::) between the category name and subcategory name.
- To specify many category-subcategory pairs, place double bar (||) between each pair. Note that a subcategory is not required.

Important: *Category and subcategory selections cannot be modified in individual occurrences of a series. This is true whether you are working in the Calendar application, or importing data using the processes described here.*

Category field examples:

- Specify an event with category *Alumni* and subcategory *Upcoming Events*:
`Alumni::Upcoming Events`
- Specify an event with category *Alumni* and subcategory *Upcoming Events*; also with category *Student Events* and subcategory *Future Events*:
`Alumni::Upcoming Events||Student Events::Future Events`
- Specify an event with category *Sporting Events* and category *Fall Activities*:
`Sporting Events||Fall Activities`

Private Flag

Indicates whether to show the event in the public calendar, the private calendar, or both.

Values indicate:

- **Y** = Private calendar only
- **N** = Public calendar only
- **B** = Both private and public calendars

Note: If your installation does not include the private calendar module, set this flag to **N**.

Highlight flag

Indicates whether to highlight the event. Values indicate:

- **Y** = Highlight the event
- **N** = Do not highlight the event

Highlighted events appear at the top of the list in an *Event List* view of a day or month, surrounded by a thick, colored line.

Facilities field (CSV) and Locations structure (XML)

This field is encoded differently in CSV than it is in XML (see the subsections below).

Facilities information for events is optional.

- If supplied, any location, building, room, and room division names are validated against records in the corresponding Active Data Calendar facility table (location, building, room or division). If the facility in the imported event record is not found, a new record is created for it.

The Calendar system links facility information in its proper hierarchy. Therefore:

- If you provide a building, you must provide the building's location
- If you provide a room, you must provide the building's location and building information
- If you provide a room division, then you must supply a location, a building, and a room to divide

If you do not provide one or more of the corresponding required fields, the system produces an error for that import record.

Coding facilities in CSV

In the CSV file format, the **Location Name**, **Building Name**, **Room Name**, and **Room Division Name** fields are combined into a single **Facilities** "super" field. These fields are delimited by double colon characters (ASCII 0x3a), within the **Facilities** field, as shown in this example:

```
...,Bethlehem::Hotel::200::200B,...
```

Multiple facilities are still entered into this one field, but separated by a different delimiter, the double pipe character (ASCII 0x7c).

Here is an example. An event occurrence is assigned to the **location** “Bethlehem”, **building** “Hotel”, with two different room facilities. The first is room 100, the second is the division of room “200” known as “200B”. The CSV **Facilities** field for importing this facility information is:

```
...,Bethlehem::Hotel::100||Bethlehem::Hotel::200::200B,...
```

Coding facilities in XML

The XML import format supports one or more facilities for a single event occurrence. The tree structure of the multiple facilities is broken down into elements under a node named **Locations**, as shown in the XML snippet below. This example contains two locations.

```
<Locations type="group">
  <Location type="group">
    <LocationName type="text">Rouses Point</LocationName>
    <SetupNotes>Facility Notes</SetupNotes>
    <Building type="group">
      <BuildingName type="text">RP-IT</BuildingName>
      <Room type="group" roomsetupname="Double Conference Table" roomsetuptime="0"
        roomteardownntime="0" roomcapacity="1000" roomlink="" usersetuptime="0"
        userteardownntime="0" usersetupcount="1000" usersetupnotes="">
        <RoomName type="text">101</RoomName>
      </Room>
    </Building>
  </Location>
  <Location type="group">
    <LocationName type="text">Bethlehem</LocationName>
    <SetupNotes>Facility Notes</SetupNotes>
    <Building type="group">
      <BuildingName type="text">Hotel</BuildingName>
      <Room type="group">
        <RoomName type="text">200</RoomName>
        <RoomDivision type="group" roomsetupname="" roomsetuptime="0"
          roomteardownntime="10" roomcapacity="6" roomlink=""
          usersetuptime="0" userteardownntime="0" usersetupcount="0"
          usersetupnotes="Open the Murphy bed">
          <RoomDivisionName type="text">B</RoomDivisionName>
        </RoomDivision>
      </Room>
    </Building>
  </Location>
</Locations>
```

(Ad-Hoc Locations)

The XML import format supports adding “Ad-Hoc” or one time use facilities for a single event occurrence. These “Ad-Hoc” locations use a structure noted as “Locations On the Fly” tree

structure of the multiple facilities is broken down into elements under a node named **LocationsOnTheFly**, as shown in the XML snippet below. This example contains two locations.

```
<LocationsOnTheFly type="group" >
  <Location type="group">
    <Name type="text">City Park</Name>
    <Address1 type="text">123 Mainstreet</Address1>
    <Address2 type="text">Lot 23</Address2>
    <City type="text">Bethlehem</City>
    <State type="text">PA</State>
    <ZipCode type="text">18017</ZipCode>
    <County type="text">Lehigh</County>
    <Country type="text">United States</Country>
    <Latitude type="text">40.6745574</Latitude>
    <Longitude type="text">-75.3837705</Longitude>
  </Location>
  <Location type="group">
    <Name type="text">City Mall</Name>
    <Address1 type="text">123 Mainstreet</Address1>
    <Address2 type="text">Lot 36</Address2>
    <City type="text">Bethlehem</City>
    <State type="text">PA</State>
    <ZipCode type="text">18017</ZipCode>
    <County type="text">Lehigh</County>
    <Country type="text">United States</Country>
    <Latitude type="text">40.6745574</Latitude>
    <Longitude type="text">-75.3837705</Longitude>
  </Location>
</LocationsOnTheFly>
```

The attributes **building type**, **building name**, **roomtype**, **room name**, **roomsetupname**, **roomsetupptime**, **roomteardowntime**, **roomcapacity**, **roomlink**, **usersetuptime**, **userteardowntime**, **usersetupcount**, and **usersetupnotes** are not supported for Ad-Hoc Locations.

Room element (XML only)

This XML group element provides details of the room in which the event takes place. Accommodates the attributes listed below.

- **RoomName** is an element within this **Room** group element.
- Attributes of the **Room** group element include these optional fields defining the room's details:

Attribute

- **roomsetupname**
- **roomsetupptime**
- **roomteardowntime**
- **roomcapacity**
- **roomlink**

- See the corresponding CSV field names in the *CSV and XML field specifications* master table above for the specifics of each element's format and length. The fields are described in *Room field elements (CSV)*.
- In facilities having a [RoomDivision](#), the above attributes appear in the [RoomDivision](#) element.
- The example XML import file on the Client Portal's *Documents* page shows this structure. This file is named Active Data Calendar 3.14.7 Event Import XML Sample

CSV

The CSV import record has no [Room](#) field as such. Each room-related attribute in the XML attribute table above is placed into its own field in CSV, as described in *User field elements* subsections.

Each occurrence of the field can contain multiple values. Separate the values with two pipe characters, || (ASCII 0x7c).

Room Link field

A URL pointing to information about the event location, such as a description, or a link to map-based directions.

CSV

In CSV format, the [Room Link](#) field contains the URL.

XML

In XML format, [roomlink](#) is an attribute of the [Room](#) group element (or the [RoomDivision](#) group element, if there is one).

Internal Comments field

This free-form text field in the Calendar contains comments posted by the various parties processing the event (the user creating the event, administrator approving it, etc.). Active Data Calendar's front end time-stamps these records to preserve the order and timing of such comments. Just as each staff member touching the event can add a comment to it on each occasion, the comments accumulate to create a log.

During imports of event update records, each update replaces the comments already in the Calendar. Accumulation of the comment log is the responsibility of the system or person creating the import file.

Start Date, Start Time, End Date and End Time fields

Every event must have a [Start Date](#). [Start Time](#), [End Date](#) and [End Time](#) are optional. These additional rules apply:

- Times must be in "h:mm AM" or "h:mm PM" format. Leading zeros are optional for single-digit hours. The import process validates these times.

- Dates must be in “m/d/yyyy” format. Leading zeros are optional for single-digit month and day numbers. The import process validates these dates.
- If an end date and/or time are specified, they must be *after* the start date and start time.
- A start time is required if an end time is provided. [A start time without an end time is valid.](#)
- For all-day events, [Start Time](#) and [End Time](#) must be either set to **12:00 AM** or left blank.

Recur Type field

Active Data Calendar supports the event recurrence types shown in this table. Calendar’s processing generates a schedule of occurrences from the parameters you specify. Recurrence’s capabilities are very powerful, but require careful execution to get the results you want.

Each pound sign (#) shown below must be replaced by an integer, as described. It represents the number of units (days, weeks or months) to skip before the next occurrence.

Valid [Recur Type](#) values include:

Recur Type	Description
One Time	A single event with no recurrence.
Custom	A series that does not have regular intervals. In Calendar, this kind of recurrence is set up using the CUSTOM SCHEDULE button, on the <i>Schedule & Request</i> dialog. You specify each occurrence date using the ADD OCCURRENCES button, and they do not have to follow any pattern. Each selected start date can have its own end date, and start and end times. Each event has one import record.
Interval#	Sets a number of <u>days</u> from the start date before the next occurrence. So, Interval1 is a daily event, Interval2 recurs every other day, Interval3 recurs every 3 rd day, and so on. Use Interval7 for events recurring weekly.
Monthly by Date#	Events in this series type recur on the same numeric day (taken from the start date’s day (“dd”) parameter) of each recurrence month, repeating every # months. Example: Recur Type of Monthly by Date2 , with a Start Date of 4/25/2013 recurs every two months, on June 25 th , August 25 th , and so on through the Recurring End Date .
Monthly by Position#	Events in this series type recur every # months on the specific <i>position</i> day of the day on which the start date falls. Example: Monthly by Position3 with a Start Date of 2/14/2013 (the second Thursday of that month) occurs quarterly (every 3 months) on the 2 nd Thursday through the Recurring End Date .
Weekly#	A series recurring on one day of each week. Example: Weekly2 with a Start Date of 2/14/2013 occurs every Thursday through the Recurring End Date .
Yearly by Date	A series recurring on the same day (“m:dd”) of every year.

Recur Days field

Used only in conjunction with [Recur Type](#) of [Weekly#](#), this parameter specifies the day(s) of the week on which the event occurs.

- Specify multiple days in a week, such as Tuesday and Friday, by separating them with commas. [If using CSV file format, enclose the list in double quotes.](#)
- Each day is specified by its standard, fully spelled-out name {[Sunday](#), [Monday](#), [Tuesday](#), [Wednesday](#), [Thursday](#), [Friday](#), [Saturday](#)}.
- If the start date and end date of an event are not on the same weekday, this value *must* be blank as this does not apply to *multi-day* recurring weekly events.

Recurring End Date field

Used with recurring events, the event appears on the Calendar up to and including this date. You can think of it as the “repeat until” date. **Note** that all events in a series *must* have the same [Recurring End Date](#), and it must be the last date in the series.

Address fields

The fields for [Address1](#), [Address2](#), [City](#), [State](#), [Zipcode](#) and [Phone](#) contain the event owner’s details.

Location Url field

Supplies a URL linking to information describing the facility where the scheduled event will take place.

Import Series Id field

A string value that uniquely identifies an event series. Used to link related individual events in the Calendar into a series, so they can be updated collectively. Every event in the series has the same value in this field.

Please note: This field is required when using a persistent import, see *Update Keys* for additional information.

Import Occurrence Id field

A string value uniquely identifying a single event occurrence within a series, so the event can be updated separately from the other event occurrences in the series.

Please note: This field is required when using a persistent import, see *Update Keys* for additional information.

Created On and Modified On fields

- These are the timestamps on which the event was created, and last changed, in the calendar
- Although a CSV import can succeed with just comma placeholders (i.e., no data) for these fields, they are required for updating events that were previously imported, and for grouping events in a series.

Import process logic

Active Data Calendar finds existing events during an import by matching on the event's [External Series ID](#), [External Occurrence ID](#) and [External Import ID](#).

- If the import process does not find a match, it creates a new event. Even if the event doesn't already exist, populating the [External Series ID](#), [External Occurrence ID](#) and [External Import ID](#) fields enables Calendar to group the events it creates into a series, not just add disconnected events that follow a recurrence pattern.
- If the import process *does* find a match, it compares the [Modified On](#) timestamp in the event in the import file with the *last import* timestamp. If the [Modified On](#) timestamp in the import file record is greater than the last import run date, the import process updates the event record from the corresponding record in the file being imported.

Created On field

- Requires both date and time elements
- Consists of a valid date in [m/dd/yyyy](#) format, then one ASCII blank character ([0x20](#)), followed by a valid time, coded as [h:mm:ss](#) [[AM](#)|[PM](#)], with a blank between the seconds and the AM/PM indicator. Leading zeros are not required in the month, day and hour.

Example: March 12, 2013 at 11:42:27 in the morning appears as
[..., 3/12/2013 11:42:27 AM, ...](#)

The specified format must be followed; if this field is not formatted as required, import-related functions dependent on a valid date and time, such as updating an existing event, will not function correctly.

Modified On field

- Requires both date and time elements
- The real world date on which the event was last modified
- Consists of a valid date in [m/dd/yyyy](#) format, one ASCII blank character ([0x20](#)), followed by a valid time, coded as [h:mm:ss](#) [[AM](#)|[PM](#)] , with a blank between the seconds and the AM/PM indicator.

Example: November 4, 2013 at 8:23:05 in the evening appears as [..., 11/04/2013 8:23:05 PM, ...](#)

The specified format must be followed; if this field is not formatted as required, functions dependent on a valid date and time, such as updating an existing event, will not function correctly.

Note: The event will only update if the [Modified On](#) date-time is greater than last import date-time.

Room field elements (CSV)

The Calendar import process supports four parameters that define the kind of room accommodations required by the participants. These include:

Room Setup Name

A name that typically describes how the room is arranged, such as “Large open rectangle”, “Multiple columns of chair rows” or “Large round tables with ten chairs each”.

Room Setup Time

The estimated lead time needed to set the room up before the event begins, usually specified in minutes.

Room Tear Down Time

The estimated time needed to dismantle the setup after the event ends, usually specified in minutes.

Room Capacity

The number of people this setup needs to accommodate.

Information Status field

Indicates whether the event is active, and if not, what the system should do with it. Options include marking it as canceled or rescheduled, or deleting it entirely.

Valid **Information Status** values:

Information Status	Meaning
A (or blank):	Approved: The default value for approved events, requiring no additional processing.
C	Canceled: Mark the event as canceled, but leave it in the Calendar.
D	Delete: Delete the event from the Calendar entirely.
R	Rescheduled: Mark the event as rescheduled, but leave it in the Calendar.

Information Status notes

C and **D** values: For these functions to update the existing event record, the import record must contain:

- **External Series ID**
- **External Occurrence ID**
- **External Import ID**
- **Modified On**

The system finds the existing event record using these fields. It also determines whether the record you want to cancel (**C**) or delete (**D**) has been updated since the last import run.

Outcomes

- If the import process finds an event occurrence matching the criteria, the existing occurrence record updates correctly.
- If the import process *does not* find the event:
 - If the imported **Information Status** is **C**, the process creates a new record in the Calendar, with a **C** status, to display the canceled event.
 - If the imported **Information Status** is **D**, the process skips the record, since it is presumed to have been already deleted.

***Important:** This field constitutes the overall Marketing Status of an event. In installations with an enabled Facilities Module, the status of associated resource and facility requests reference this status. For example, if **Information Status** is set to **A**, all facilities and resources associated with the event are imported as **Approved** also.*

Setup Notes field

Free-form text to help the person setting up the room know how you want it done.

User field elements

The Calendar import process supports four parameters that define the kind of accommodations required by the Calendar user for setting up and tearing down the facility (that is, before and after the actual event). These are entered at the event level in the Calendar (not the room), and include:

User Setup Time

The number of minutes the event owner needs when setting up the facility (say, a presenter's computer, projector, passing out hand-outs, etc.), in addition to the time facilities management needs to set up tables and chairs (or whatever), which is in **Room Setup Time**.

User Tear Down Time

The estimated time the event owner needs for dismantling equipment or the like, after the event ends, in addition to the time facilities management needs to take away their setups, clean, etc., which is in **Room Tear Down Time**.

User Setup Count

The number of setups (i.e., the number of participants) the event owner anticipates having to set up and tear down.

User Setup Notes

Any additional set-up information that the event owner thinks would be helpful.

CSV

In a CSV file, these attributes are split into their own fields, with the above names. Each field can contain multiple values separated by two pipe characters, || (ASCII 0x7c).

Note: The number of values being imported must be equal to or less than the number of rooms. If there are fewer values than rooms, the import links the supplied values to the rooms in order, until there are no more setup values; the excess room(s) will not have these parameters.

The example CSV import file in the Client Portal's *Documents* area provides an example of these fields. The file's name is Active Data Calendar 3.14.7 Event Import CSV Sample.

XML

In an XML file, these fields appear as attributes in the `<Room>` node (or the `<RoomDivision>` node, if one exists for the `Location` instance). See *Room element (XML only)*. The names of the corresponding XML attributes are:

Attribute

- `usersetuptime`
- `userteardowntime`
- `usersetupcount`
- `usersetupnotes`

Internal Custom `<n>` fields

These four fields can contain event information useful for internal operations, such as filtering. The information they contain is not displayed to visitors. This field is linked to the lowest level of the facility (`Room` or `RoomDivision`).

County fields

This field accepts up to 255 characters of free-form text, without validation.

State fields

This field accepts 2 characters. If the country field is populated, the state field will be validated against the Country field.

Country fields

This field accepts up to 255 characters and is validated in the Countries list at the end of this document.

Registration fields

These fields indicate whether registration is enabled for the event, and contain registration information for events that do have registration enabled.

Registration Enabled field

- Required for all events
- Indicates whether the event is set up to accept registrations
- Valid values are `Y` (yes) and `N` (no)

- If this field = **Y**, all the other **Registration** fields are required

Registration Max Registrants field

- Required if the event accepts registrations
- Limits the number of registrants, if desired
- Indicates the limit to the number of registrants the event accommodates
- Must be a positive integer

Registration Display Available field

- Required if the event accepts registrations
- Controls whether the Calendar displays the number of unfilled registrations remaining for the event. If enabled, this information appears in the list view (if configured), and the event details.
- Valid values are **Y** (yes) and **N** (no)

Registration Type field

- Required if the event accepts registrations
- Indicates the type of enforcement for series-event registrations
- Valid values are **Single**, **Partial**, **Series** or <blank>
- Must be **Single** for a one-time event
- Blank sets registration type to **Multiple Occurrence Registration Allowed**

Registration Template field

- Required if the event accepts registrations
- Contains a valid form ID integer value, which the import process validates against the registration forms table
- A <blank> or zero denotes the Standard Form
- The import process returns an error if the **Template** value is not valid

Resource fields

Resource records are optional. Resources include items such as desktop computers for users, a video projector or chairs. To find the resource name values for your system, look in the *Resource Category/Subcategory* drop-down list of the *Request Resources for this Event* dialog. Note that resource category and resource subcategory are separate elements in XML, but in CSV are combined into one field.

Resource record structure

- If the import does include resource record groups, each group must include the fields shown as *required*, below.
- Only **Resource Name** is optional. The other three fields are required.
- **CSV:** If the event has multiple resources, separate them with a double pipe (||), observing that every field in a resource record must account for the same number of resources in the overall record. The CSV version of *Resource record example* below the field definitions for this record illustrates this, showing one placeholder for a resource that doesn't have the optional resource name, and another for a resource with no **Notes** text. If the event has just a single resource, the double pipes are not required.
- **XML:** An event can have only one **Resources** structure, though it may have many **Resource** instances for the event. Elements within the structure are required as noted in the field descriptions below.

Resources - CatSubcat field

- Required for each resource listed
- **CSV:** Separate category from subcategory with a double-colon sequence. For example, **Computer::Desktop**, for category of **Computer** and subcategory of **Desktop**.
- **XML:** This is two separate fields, one called ResourceCat, the other, ResourceSubCat.

Resources - Resource field

- Optional for each resource listed
- Gives the name of the resource, such as **Dell 660s**, for a CatSubcat of **Computer::Desktop**
- **CSV:** Must include placeholders if a given resource does not have a name

Resource - Quantity field

- Required for each resource listed
- Gives the number needed of the particular resource
- Must be a non-zero integer

Resource - Notes field

- Required for each resource listed
- Free-form text providing information to the person setting up the room
- Can be blank, but must be accounted for by inserting the double-pipe placeholder (||) where it is blank. See the example shown next.

Resource record example

Let's say you're holding a software training event with a presenter and 20 participants in one room. The participants pair up, two sharing one desktop computer to perform the exercises. The presenter needs a laptop computer, a podium with a microphone, and a video projector.

The resource inventory for this event is:

- 10 desktop computers (participants)
- One video projector
- One laptop computer (presenter)
- One microphone-equipped podium

The room is a dedicated training room, so desks and chairs are already there, so we don't need to name them as resources.

If you were to create a resource request for this directly in the Calendar front end, here is what it would like:

Resources/Items Selected for your Event			
Qty	Resource Name	Notes	
<input type="text" value="10"/>	Dell GX745	For the participants	<input type="button" value="DELETE"/>
<input type="text" value="1"/>	Epson 540		<input type="button" value="DELETE"/>
<input type="text" value="1"/>	Mac Air	The presenter gets a classy Mac laptop	<input type="button" value="DELETE"/>
<input type="text" value="1"/>	Podium 2	This podium has a built-in gooseneck microphone	<input type="button" value="DELETE"/>

The **CSV** record for this setup is:

```
...,User Computer::Desktop|AV Equipment::Projectors|AV
Equipment::Laptops|Furniture::Podiums,Dell GX745|Epson 540|Mac Air|Podium
2,10||1||1,For the participants|||The presenter gets a classy Mac
laptop||This podium has a built-in gooseneck microphone,...
```

Note that there are four fields here (shown in alternating green and red, so you can see them easily), hence the three commas (in blue) separating the fields within the example.

If there were no name for the `User Computer::Desktop` resource, the second field (name) would begin with a placeholder (instead of the `Dell GX745` shown), like this:

```
,||Epson 540|Mac Air|Podium 2,
```

Similarly, if the `AV Equipment::Projectors` resource had no name, the name field would have a placeholder in the projector's position (instead of the `Epson 540` shown), like this:

```
,Dell GX745|||Mac Air|Podium 2,
```

The equivalent **XML** structure for this same setup is:

```
<Resources type="group">
  <Resource type="group">
    <ResourceCat type="text">User Computer</ResourceCat>
    <ResourceSubCat type="text">Desktop</ResourceSubCat>
    <ResourceQuantity type="text">10</ResourceQuantity>
    <ResourceNotes type="text">For the participants</ResourceNotes>
  </Resource>
  <Resource type="group">
    <ResourceName type="text">Epson 540</ResourceName>
    <ResourceCat type="text">AV Equipment</ResourceCat>
    <ResourceSubCat type="text">Projectors</ResourceSubCat>
    <ResourceQuantity type="text">1</ResourceQuantity>
    <ResourceNotes type="text" />
  </Resource>
  <Resource type="group">
    <ResourceName type="text">Mac Air</ResourceName>
    <ResourceCat type="text">AV Equipment</ResourceCat>
    <ResourceSubCat type="text">Laptops</ResourceSubCat>
    <ResourceQuantity type="text">1</ResourceQuantity>
    <ResourceNotes type="text">The presenter gets a classy Mac
laptop</ResourceNotes>
  </Resource>
  <Resource type="group">
    <ResourceName type="text">Podium 2</ResourceName>
    <ResourceCat type="text">Furniture</ResourceCat>
    <ResourceSubCat type="text">Podiums</ResourceSubCat>
    <ResourceQuantity type="text">1</ResourceQuantity>
    <ResourceNotes type="text">This podium has a built-in gooseneck
microphone</ResourceNotes>
  </Resource>
</Resources>
```

Note that the first `Resource` group (for the `User Computer`) lacks a `ResourceName` instance. This is valid, since `ResourceName` is optional.

External Series ID and External Occurrence ID fields

These two string-value fields uniquely identify the event in the system you used to create the import file. Data that you export from Active Data Calendar includes these fields, so that if you need to update the information in the system from which these event records came, that system can find the event.

- **External Series ID** is the overall event identifier
- **External Occurrence ID** identifies single occurrences within the event series

By combining these fields with the **External Import ID** (see below), the Calendar system can correctly process updates from multiple external systems.

Event Owner field

The event owner's Login ID as it appears in the Account table. It is not required, but if its populated, it must be an active Login ID. You can find this information on the Calendar front end, through the *Workflow* tab's **ACCOUNT(S): View** function.

External Import ID field

An import identifier that you supply. The Calendar system uses this field to relate imports to each other. To be successful, calendar event update imports must include this field, and it must match the value used to import the events. If not supplied, it uses the URL from which the file was pulled in a persistent import.

By combining this field with the **External Series ID** and **External Occurrence ID** fields (see above), Calendar can correctly process updates from multiple external systems. These three fields together enable Calendar to determine the exact information from the previous import that is being updated.

Event Owner Name field

A text field used during the *export* process, **Event Owner Name** enables calendar exports to be *re-imported* into the same calendar or a central calendar. This field must be present, though its value may be omitted.

DateUp element (XML Only)

Corresponds to the *Publish Date/Time* field on the Scheduling and Facilities page in event creation and modification. Tells the Calendar to begin showing the event on this date.

DateDown element (XML Only)

Corresponds to the *Unpublish Date/Time* field on the Scheduling and Facilities page in event creation and modification. Tells the Calendar to stop showing the event on this date.

HideEndDateTime element (XML Only)

Indicates whether to hide the end date and time of an event when displayed in the Calendar.

EventGUID element (XML Only)

Uniquely identifies the event across all platforms. See [this article](#) for more information on globally unique identifiers (GUIDs).

AttachmentBinary and ImageBinary fields (XML Only)

These fields may contain:

- base64-encoded image and attachment data if the event has an associated image and/or attachment(s)
- A URL pointing to an image or attachment related to the event

Note: CSV files do not support binary inclusions.

Sample XML and CSV Import Files

Sample files are provided on the Client Portal's *Documents* page. They are named:

- Active Data Calendar 3.14.7 Event Import CSV Sample
- Active Data Calendar 3.14.7 Event Import XML Sample

Update Keys

For event records to update, the fields uniquely defining them must match the corresponding Calendar database keys. Record updates are based on these fields:

- ImportSeriesID
- ImportOccurrenceID
- CreatedOnDate
- ModifiedOnDate
- ExternalImportID

Importing iCalendar files

While an iCalendar (.ics) file has a more flexible format than either CSV or XML, it still must be followed carefully to ensure the events that you import get created in the Calendar as you intend.

- The complete iCalendar file format specification is available at <http://tools.ietf.org/html/rfc5545>. Active Data Calendar implements only those components and properties listed in this User Guide, not the entire specification.
- Although Calendar's import process does not require an iCalendar-format file to have any particular name and extension, .ics is the extension recommended in RFC 5545 (and generally used in the industry) for files containing schedule information (that's the iCalendar name for an *event*).

The table below provides a list of all required fields, with some guidelines to make the import process easier.

Note: The iCalendar file format does not provide a "department" field. When setting up the import, you must specify a destination department to place all imported events under.

All iCal events must supply values for SUMMARY (provides the event name), DTSTART (start date), and DTEND (end date). The remaining fields are defined as optional. The field lengths shown are constrained by the Calendar's database column sizes.

iCalendar fields

This table lists the subset of iCalendar-format properties (i.e., fields) that Active Data Calendar's event import process supports.

- Though shown capitalized for readability, the keywords and data in the definitions and examples are not case-sensitive.
- Except for BEGIN and END, the properties do not have to appear in any specific order.

Note: The T and Z components of date-and-time values are literal, but not case-sensitive. The Z represents "Zulu" time, meaning UTC (coordinated universal time), formerly known as GMT (Greenwich Mean Time). Times in iCalendar files are, accordingly, interpreted by the import process as UTC.

Property Name	Length	Possible Values	Required?
BEGIN	15	BEGIN:VCALENDAR, BEGIN:VEVENT	Y
PRODID	??	Alpha numeric	N
SUMMARY	100	Alpha numeric	Y
DTSTAMP	N/A	Valid date in "yyyyMMddThhmmssZ" format	N
DTSTART	N/A	Valid date in "yyyyMMddThhmmssZ" format	Y
DTEND	N/A	Valid date in "yyyyMMddThhmmssZ" format	Y
DESCRIPTION	6000	Alpha numeric	N
RESOURCES	255	Alpha numeric	N
LOCATION	75	Alpha numeric	N
CATEGORIES	50	Alpha numeric	N
RRULE	N/A	Alpha numeric	N
UID	255	GUID	N
LAST-MODIFIED	N/A	Valid date in "yyyyMMddThhmmssZ" format	N
END	13	END:VCALENDAR, END:VEVENT	Y

iCalendar field descriptions

This section describes each of the import process' supported fields.

BEGIN property

A required, literal delimiter string placed at the beginning of the import file, and the beginning of each event component definition. Must appear exactly as shown above and in the example file below.

- **BEGIN:VCALENDAR** must be placed before any other properties.
- An iCalendar file can contain multiple event components, and **BEGIN:VEVENT** must be placed before each one.

PRODID property

The identifier for the product that created the iCalendar object.

- Begins with a literal hyphen-slash combination, - /
- Provides a place to name the vendor, product name, and product version. This information is delimited on both ends by a forward-slash character (see the example below).
- Ends with another slash, and the language code for English, /EN

SUMMARY property

The name of the event being imported. Placed into the Event Name field (see the CSV/XML table above).

DTSTAMP property

The event's creation date in the system supplying the file.

DTSTART property

The event's starting date.

DTEND property

The event's ending date.

DESCRIPTION property

The event's full textual description, and other free-form text you may wish to embed.

- If this is omitted or blank, Calendar uses the [SUMMARY](#) property
- This property is for supplying free-form text, but you can force line breaks using the "newline" escape sequence, `\n`, as shown in the example below
- The text in this property is not parsed by the import process beyond that required to interpret the newline sequences. (Despite the example looking like it contains CSV fieldnames, that is merely to help the human reader.)

RESOURCES property

An optional list of the resources needed for the event. Note that the import does not create Active Data Calendar resources from this information. It is instead added to the "Other Details" custom external field ([ExternalField2](#)) for the event, for display only. Note that even if custom field #2 has a label different from *Other Details* (the Calendar default), the data still goes into [ExternalField2](#).

NOTE: This property is not exported from Calendar when exporting events.

LOCATION property

The location name for the event (in the Facility table). Should not include building, room, or room division information.

CATEGORIES property

A comma-separated list of categories for the event. If this property is omitted or blank for an event, the process behaves as you specified when setting up the import. If you selected *Imported File includes Category/Subcategory information*, and the record does not contain this property, the import process returns an error.

RRULE property

The recurrence rule for the event. See <http://tools.ietf.org/html/rfc5545#section-3.8.5.3> for the recurrence rule specification.

UID property

A unique identifier (GUID⁵) for the VEVENT record (not the import).

- Acts like a database key to locate the event record in the Calendar
- Generated and maintained by the exporting application

⁵ See [this article](#) for more information on globally unique identifiers (GUIDs).

- Used to link records for updating by subsequent imports

Important: *Persistent imports must include the UID property in the initial import and subsequent updates. Failure to supply this will result in duplicate events.*

LAST-MODIFIED property

The date when the event was last modified. Used to check whether an existing record should be updated. If the record is not found, a new record is created.

END property

A required, literal delimiter string placed at the end of the import file, and the end of each event component definition. Must appear exactly as shown above and in the example file below.

- **END:VCALENDAR** must be placed after all other properties
- An iCalendar file can contain multiple event components, and **END:VEVENT** must be placed after each one

Example iCalendar import content

Below is sample iCalendar-format content. It contains some of the properties listed above, including a properly formatted header and footer. As mentioned above, each event must begin and end with its own **BEGIN:VEVENT** and **END:VEVENT** properties.

To be processed, each event in the file must have **SUMMARY**, **DTSTART** and **DTEND** properties. All other fields are optional. Each line in an iCalendar file must be no longer than 80 characters. If a property requires more than 80 characters (such as for the **DESCRIPTION** property in the sample), continue the property on a new line, and begin it with a single space character.

```
BEGIN:VCALENDAR
PRODID:-//ActiveDataExchange/Calendar V3.9.1//EN
BEGIN:VEVENT
DTSTAMP:20100505T165400
DTSTART:20100511T020000Z
DTEND:20100511T030000Z
LOCATION:Location Name
SUMMARY:This is the event name
DESCRIPTION:Event Description:\ntest\n\nContact Information:\nEmail:
email@activedatax.com\n\nLocation Information:\n \n\n
CATEGORIES:Category1 - SubCategory1,Category2 - SubCategory2
RESOURCES:Text listed here will display in the Other Details field
UID:487203995746
END:VEVENT
END:VCALENDAR
```

A sample file in iCalendar format is provided on the Client Portal's *Documents* page, named Active Data Calendar 3.14.7 Event Import iCalendar Sample.

Information that can't be imported

Not all information in Active Data Calendar can be imported. For example, the dimensions of a divided room can be stored in the Calendar's database, but there is no provision for importing this level of information into Calendar. If an item isn't addressed in this document, it probably isn't supported by the import process.

FAQs and/or troubleshooting tips

Duplicate records

Q1: I ran my import to update some events, but instead of updating my events, a lot of them are now showing up as duplicates. How did that happen, and how can I keep it from happening again?

A1: Check your key fields carefully. The key fields are those that the import process uses to find an existing record to update. In this version of Calendar, the fields in your import that must match those already in the Calendar database from the original import include:

- `ImportSeriesID`
- `ImportOccurrenceID`
- `CreatedOnDate`
- `ModifiedOnDate`
- `ExternalImportID`

If these fields in your update do not match those for the original import, the process will indeed create a new event record, and you will have duplicates.

Appendices

Country List

United States	Cape Verde	Germany	Liberia	Oman	Spain
Afghanistan	Cayman Islands	Ghana	Libya	Pakistan	Sri Lanka
Albania	Central African	Greece	Liechtenstein	Palau	Sudan
Algeria	Chad	Greenland	Lithuania	Palestine	Suriname
Andorra	Chile	Grenada	Luxembourg	Panama	Swaziland
Angola	China	Guadeloupe	Macedonia	Papua - New Guinea	Sweden
Antigua Barbuda	Colombia	Guatemala	Madagascar	Paraguay	Switzerland
Argentina	Comoros	Guinea	Malawi	Peru	Syria
Armenia	Congo Democratic	Guinea Bissau	Malaysia	Philippines	Taiwan
Australia	Congo	Guyana	Maldives	Poland	Tajikistan

Austria	Costa Rica	Haiti	Mali	Portugal	Tanzania
Azerbaijan	Cote d'Ivoire	Honduras	Malta	Puerto Rico	Thailand
Bahamas	Croatia	Hong Kong	Marshall Islands	Qatar	Togo
Bahrain	Cuba	Hungary	Mauricius	Reunion	Tonga
Bangladesh	Cyprus	Iceland	Mauritania	Romania	Trinidad and Tobago
Barbados	Czech Republic	India	Mexico	Russia	Tunisia
Belaruss	Denmark	Indonesia	Micronesia	Rwanda	Turkey
Belgium	Djibouti	Iran	Moldova	Saint Kitts and Nevis	Turkmenistan
Belize	Dominica	Iraq	Monaco	Saint Lucia	Tuvalu
Benin	Dominican Republic	Ireland	Mongolia	Saint Vincent	Uganda
Bermuda	Ecuador	Israel	Morocco	San Marino	Ukraine
Bhutan	Egypt	Italy	Mozambique	Sao Tome and Principe	United Arab Emirates
Bolivia	El Salvador	Jamaica	Namibia	Saudi Arabia	United Kingdom
Bosnia-Herzegovina	Equatorial Guinea	Japan	Nauru	Senegal	Uruguay
Botswana	Eritrea	Jordan	Nepal	Seychelles	Uzbekistan
Brazil	Estonia	Kazakhstan	Netherlands/Holland	Sierra Leone	Vanuatu
Brunei	Ethiopia	Kenya	New Caledonia	Singapore	Venezuela
Bulgaria	Fiji	Kiribati	New Zealand	Slovakia	Vietnam
Burkina Faso	Finland	Kuweit	Nicaragua	Slovenia	Western Sahara
Burma(Myanmar)	France	Kyrgyzstan	Niger	Solomon	Yemen
Burundi	French Guiana	Laos	Nigeria	Somalia	Yugoslavia
Cambodia	Gabon	Latvia	North Korea	South Africa	Zambia
Cameroon	Gambia	Lebanon	Norway	South Korea	Zimbabwe
Canada	Georgia	Lesotho			