



GAP Plug-in for Generic Application Data File Import

Plugin Configuration Guide

Contents

Introduction	2
GAP Configuration Checklist	2
Supported Collection Types	2
Data Sources	3
Overview of using the GAP Plug-in:	3
Creating a CSV file of backup data:	3
Limitations.....	3
Format and Contents of CSV Data Import File.....	4
¹ Bocada Errorset Usage	7
Installation of GAP Plug-in	7
Components and Installation.....	7
Considerations when Upgrading Bocada:.....	8
Optional: Modifying the GAP Plug-in to Appear as the Backup Product Name	8
Upgrading the GAP Plug-in	9
Configuring GAP Plug-in.....	9
TroubleShooting:	10
Data Collection Update fails with Error 309: Manual plug-in exited with status -1073741515	10
Data Collection Update fails with Error 545: The NVSAVE plugin is not compatible with this revision of BeUpdate.exe.....	11
Data Collection Update fails with Error 546: Incorrect number of bindings supplied	11
Technical Support	12
Appendix: Python License and Copyright Notice.....	12

Introduction

This document describes the Bocada Manual, or Generic Application (GAP) Plug-in for Bocada.

The GAP Plugin is used to bring backup job data into the Bocada database for reporting when a backup product specific plugin is not available. In some production applications the GAP plugin is used for situations in which the backup server cannot be reached by the normal Bocada Data Collection Service, but when the data can be shipped out of the backup server in a controlled CSV format.

The GAP plug-in mines data from a comma separated value (CSV) format. Typically, the file is populated by a script written by the Bocada User or by Bocada Professional Services. A script would extract data from the backup product data and dump the data into a CSV file that the Bocada plugin can access.

GAP Configuration Checklist

While detailed steps are included below, this is an overview of the steps to configure GAP collections on your Bocada Data Collection Server:

- ☐ Configure .csv file to be extracted from backup server to location on the DCS.
- ☐ Install the GAP plug-in (provided by Bocada Support). This adds product type “Manual”.
- ☐ If desired, rename “Manual” to actual product name.
- ☐ Add the “Manual” server to Bocada under Operations > Backup Servers, and set Server Properties.

Supported Collection Types

The plug-in currently supports the following collection types:

Collection Type	Supported	Description
Backup	✓	Collects transactional details about backup, duplication and restore jobs. Example metrics include, start times, durations, bytes, files, errors etc.
Storage		Collects point-in-time inventory information. Example metrics include, total recoverable gigabytes (storage), media volume count, media volume status, etc.
Policy		Collects and stores information on policy attributes, schedules, storage units, storage groups, storage lifecycle policies and clients.

Data Sources

The GAP Plug-in works similarly to other Bocada Plug-ins in most ways: Data is collected by a Bocada Data collection server. For the GAP plug-in, the data is mined from one or more comma-separated-value CSV data files, each one associated with a named backup server, as defined in the Bocada User Interface. Data mining runs according to the schedule as defined in Bocada for that backup server.

Overview of using the GAP Plug-in:

Installing the GAP Plug-in will create a new Backup Product Type called “Manual”. Using the standard Bocada user interface, you will add a new backup server of this type and specify a data file location from which the server will mine backup jobs.

Creating a CSV file of backup data:

Typically, the data input file will be created by a data extraction script as written by you. Bocada Professional Services may be able to assist in creating such a script; This activity is outside the parameters of standard Bocada Support. The import file may also be assembled using a text editor or Excel spreadsheet, and exported in CSV format to a file, but the Excel spreadsheet must be configured to export dates in the format specified below.

The CSV file should be placed in a folder on the Bocada Data Collection Server (DCS) or on a remote share accessible by the DCS. New data should overwrite the old data with the same filename as needed. Bocada data mining will insure that no duplicate records are created.

The GAP Plug-in will import the CSV file data to the Bocada database for reporting. Data collection updates should be scheduled to run regularly after the CSV file has been updated to automatically mine the data from the file.

The Bocada mining code handles detection of duplicate jobs, so the same CSV data can be collected by Bocada repeatedly and if the backup job already exists, then Bocada will not add the job to the database again.

Limitations

The GAP plug-in includes the following limitations:

- Data errors in the input file will not have an automated removal method. If the manually created data input file has errors in the data, those errors will be imported to the Bocada database.
- If two distinct backup jobs have the same client name, same target name, and the start time is identical down to the nearest minute, then the jobs will be assumed to be duplicates, and only one will be loaded in to the Bocada database.

Format and Contents of CSV Data Import File

The data can be entered into a CSV file manually or programmatically. The table below shows the fields in each line of the file.

- There is exactly one job per line, and there cannot be any trailing blank lines at the end of the file, nor extra records at the end of each line.
- Fields deliberately left empty should not have any whitespace; one comma followed must immediately by following comma.
- An example import data file would contain lines that appear as follows:

```
aacwin04,c:\ Galaxy, 2017-05-03, 2018-11-09,-5,,11223344,Full,Galaxy,,0,This is a comment.
acywin05,SystemState, 2017-05-03,,1.5,90,222333444,Full,BackupExpress,,1,
aclientsap,/usr, 2017-05-03 03:33,,,-8,,333444555,Incremental,SAP,678,16384,TSM backup of SAP
aclsap2,/, 2017-05-03 16:16,-8,,,556677,Full, ViceVersa,678,6,ViceVersa
aclient1,/target1,2017-05-03 19:19:18,,,-8,180,556677,Full,SAP,678,7,Errorset 7 Test
```

Column	Field	Example(s)	Description	Notes	Bocada Field Name
A	clientname	acytxwin04	Name of application server whose data is being backed up.	Required field. Name must be exactly consistent from one data file import to the next or else data will appear to be from multiple clients.	clientfqname
B	targetname	c:\ /usr VM name	Data target backed up.	Required field. Only one target per line is accepted. The targetname will be treated as a string with no interpretation of meaning.	targetname
C	startdatetime	2018-06-30 or 2018-11-01 03:03:03	Date and time when backup job begins. Format is yyyy-mm-dd with optional hh:mm:ss. This can be in local time of backup server or GMT time.	Required field. Several reports accuracy depend on the accuracy of this start time field. Note that fields MUST have leading zeros for fields e.g. 3:00 AM must be entered as 03:00, first of the month must be 01. See appendix for how to force Microsoft	Jobdatetime or Jobdatetimelocal (depends on offset value below)

				Excel to write CSV files in this format.	
D	expirationdate	2019-06-21 or 2018-12-31 13:00 or <empty>	Date (time optional) when media data is set to expire, in the local time of the backup server.	Can be empty. GMT offset time will be applied. If retention dates are not available in backup job or if retentiondays (defined in next row of this table) instead of expiration date is provided.	expirationdatetime
E	hoursGMToffset	-5 or 1.5 or 0	Offset is in hours to add to GMT times to get to local time of startdate and expiration date.	UTC aka GMT time offset in hours is the number of hours to add to a GMT time in order to get local time. For example US Eastern Standard time is -5. This value should be identical for every line in the file and should correspond to the setting in the server properties dialog in the Bocada User Interface. If in doubt, set this value to zero, and input all times as GMT.	(used to calculate jobdatetime and jobdatetimelocal)
F	retentiondays	90 or <empty>	Number of days this backup data copy will be retained.	Can be empty. This will be ignored if expirationdate days is provided. If both expirationdate and retentiondays are empty, then the data will be assumed to never expire and 2100 will be put in Bocada® database as expiration.	(used to calculate expirationdatetime if expirationdate is not provided)
G	bytecount	112233344	Bytes backed up. Can be	Required field. This must be a plain byte	bytecount

			negative number.	count, not GB or MB. A negative number is a special-case use that will allow for changing aggregate retention calculations.	
H	canonicallevel	Full or Incremental	Bocada standard backup level names, one of: Differential, Full, Incremental, Manual, Partial, Archival, etc.	Required field. Restore and duplication jobs are not supported in this version of the plug-in.	canonicallevel name
I	groupname	WeeklyFull or ViceVersa or SAP	Backup group name if available, else schedule name or other.	Required field. See regular Bocada documentation for definition of group name. If you do not have a correct group name then you may wish to load this field with the name of the backup product or with the string ManualPlugin.	groupname
J	jobdurationse conds	678 or <empty>	Job duration in seconds.	Can be empty. This value is required for throughput and success and failures reports that need length of job to calculate. If the field is empty then the jobduration will be set to 60 seconds.	
K	errorset	0 or 16384	Whether there are errors in backup.	Required field. 0 means no errors, and the full Bocada errorset is supported.	errorset

				A simplified set of errors can be used". ¹	
L	comment	This is a comment	Optional comment up to 32 characters, omit quotation characters.	Can be empty.	groupsession

¹Bocada Errorset Usage

For more sophisticated error reporting, please contact Bocada.

Installation of GAP Plug-in

Installation is performed simply by adding four files, obtained from Bocada Support, to your bin directory. The default Bocada installation bin folder is:

C:\Program Files (x86)\Bocada\DataCollection\bin

Components and Installation

You will need the following 4 components added to your Bocada\DataCollection\bin folder to install the Manual Plug-in, be sure to restart your Bocada services after adding these files:

1. manual.box² file.
2. manual.exe file.
3. manual.plugin.xml file.
4. manual.properties.xml file.

You may find the following additional components useful:

5. A sample CSV input file that you can use for testing.
6. This document.

You must create the final components:

7. Your CSV data input file: with default name of `Manual_Import.csv`, with optional default location of `C:\Program Files (x86)\Bocada\DataCollection\data`

² This will be created by your Bocada engineer by renaming a generic.box file compatible with your version of Bocada. This file must be updated with each version of Bocada!

Considerations when Upgrading Bocada:

Upgrading your Bocada installation is currently done by uninstalling the Bocada Application and Data Collection servers, then installing the new versions. Installing the new Data Collection Server version also installs new versions of the default plug-pins, but not the four 'manual' plug-in components listed above. These files are available from Bocada Support; please contact us if you are planning an upgrade so that we can provide you with the updated versions of these four files.

Optional: Modifying the GAP Plug-in to Appear as the Backup Product Name

By default, jobs collected by the GAP Plug-in will appear in the Bocada database, and in Bocada reports, with the Product Name *"manual"*. You can modify or copy the GAP Plug-in to match the name of the backup product whose data you are processing and the plug-in is collecting. This renaming will cause the new name to be stored in the productname field in the Bocada database so that will be displayed in Bocada reports instead of *"manual"*. The new name that you select must be a single token and not have any spaces in it. Here are instructions how to do this using the example product name of 'RealProductName':

- Copy or rename manual.box file to RealProductName.box
- make certain there are no spaces in RealProductName
- Copy or rename manual.plugin.xml file to RealProductName.plugin.xml
- Copy or rename manual.properties.xml file to RealProductName.properties.xml
- Edit the RealProductName.plugin.xml file, changing the word Manual to your product name in 3 places. The original file will appear similar to this:

```
<?xml version='1.0'?>
<!DOCTYPE Plug-inDescriptor>
<Plug-inDescriptor>
  <Name>Manual</Name>
  <NickName>Manual</NickName>
  <ProductName>Manual Plug-in</ProductName>
  <Version>1</Version>
  <TierCode>0</TierCode>

  <!-- capabilities indicate which types of updates are supported by the plug-in -->
  <Capability name="MediaInventory" value="no" />
  <Capability name="MediaActivity" value="no" />
  <Capability name="BackupActivity" value="yes" />

</Plug-inDescriptor>
```


Your changed lines will be these 3 and the new lines should appear similar to this:

```
<Name>RealProductName</Name>  
<NickName>RealProductName</NickName>  
<ProductName>RealProductName Plug-in</ProductName>
```

Upgrading the GAP Plug-in

The GAP plug-in is version-dependent, but is not updated by the Bocada Installer. When upgrading Bocada to a new version, please contact Bocada Support for an upgraded version of the plug-in files. Especially you will need a new copy of manual.box file.

Configuring GAP Plug-in

After you install the GAP plug-in, add one backup server per CSV input file to Bocada via the Bocada Application: Navigate to Operations > Backup Servers, and select Add Server to open the Add Server dialog, selecting “Manual” for setting Application Type (or the Product Name you set, if the optional steps above were implemented).

If you create one manual virtual server in Bocada per CSV from each actual backup server, your reports grouped by server will correlate to your actual backup servers. If you instead extract data from all actual backup servers into a single CSV file (not recommended), this will appear as a single manual virtual server in Bocada.

Each manual server configured may be set to its own time zone. Set the name and location of your CSV manual import data file, and the time zone for which the dates and times in the CSV file will be interpreted; This must match your time zone offset field ‘hoursGMToffset’ in the CSV file.

Edit Servers

Servers | Schedules | Finish

Server name: gap_01

Application type for these servers : Manual

Configure Server Properties :

CSV_File_to_Import: \\WIN08R-SRV16\data\sample_manual_in

time zone: (GMT-08:00) Pacific Time (US & Canada)

Show Advanced Properties:

Debug logging level : Disabled

Previous Next Finish Cancel

TroubleShooting:

Data Collection Update fails with Error 309: Manual plug-in exited with status - 1073741515

If your update fails with the following error:

```
309: Manual plug-in exited with status -1073741515
```

Then you may need to copy some delivered dynamic linked libraries from one Bocada folder to another.

Copy the following files from your DataCollection Folder to your DataCollection/bin folder.

- msvcp71.dll
- msucr71.dll

Data Collection Update fails with Error 545: The NVSAVE plugin is not compatible with this revision of BeUpdate.exe

The GAP plug-in is version-dependent, but is not updated by the Bocada Installer. If you are seeing this error after upgrading Bocada, please contact Bocada Support for the plug-in files appropriate to your version. The key file that needs to be updated is *manual.box*.

Data Collection Update fails with Error 546: Incorrect number of bindings supplied

If your collection fails with an error similar to the below then your CSV file has at least one row with the wrong number of columns. In this example there is one extra column in at least one of the rows:

```
GENPLUG: CSV_File_to_Import = C:\Program Files  
(x86)\Bocada\DataCollection\data\data.csv  
Error 546: Access denied connecting to File: Incorrect number of bindings supplied. The  
current statement uses 12, and there are 13 supplied.
```

To check for too many columns you can briefly load your CSV into Excel (but do not save) you will see that some row do not have exactly 12 columns. Because some cell entries might be empty this may not be sufficient check for too few columns, but it is worth investigating!

Technical Support

For technical support or a copy of our standard support agreement, please contact us.

E-mail: support@bocada.com
Support Portal: <https://bocada-support.force.com/s/>
Phone: +1-425-898-2400

Appendix: Python License and Copyright Notice

The Bocada Manual Plug-in is partially implemented in Python. Below is the Python license and copyright notice:

PSF LICENSE AGREEMENT FOR PYTHON 2.4

-
1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 2.4 software in source or binary form and its associated documentation.
 2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 2.4 alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004 Python Software Foundation; All Rights Reserved" are retained in Python 2.4 alone or in any derivative version prepared by Licensee.
 3. In the event Licensee prepares a derivative work that is based on or incorporates Python 2.4 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 2.4.
 4. PSF is making Python 2.4 available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 2.4 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.
 5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 2.4 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 2.4, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
 6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
 7. Nothing in this License Agreement shall be deemed to create any

relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python 2.4, Licensee agrees to be bound by the terms and conditions of this License Agreement.

Copyright © 2020 Bocada LLC. All Rights Reserved. Bocada and BackupReport are registered trademarks of Bocada LLC. Vision, Prism, vpConnect and the Bocada logo are trademarks of Bocada LLC. Other product names mentioned herein may be trademarks or registered trademarks of their respective companies.

Protected by U.S patents 6,640,217; 6,708,188; 6,745,210; 7,457,833; 7,469,269; 7,496,614; 8,407,227

The material in this manual is for information only and is subject to change without notice. While efforts have been made to ensure accuracy, Bocada LLC assumes no liability resulting from errors or omissions in this document, or from the use of information contained herein.

Bocada LLC reserves the right to make changes in the product design and documentation without reservation and without notification to its users. 2020-08-04