

Write Legacy Custom Inventory Fields with an fwclد Fileset

What this legacy recipe does



This recipe writes the legacy numbered inventory fields with `fwclد -custom_write`. For new inventory requirements, use [Custom Fields](#), which offer more flexible definitions and reporting. Keep this method where an existing script, report, or integration still depends on the legacy field names.

The legacy schema provides 20 slots for each data type: String, Integer, Boolean, and DateTime. Choose an unused key from the complete matrix below, then make sure the command returns a value that matches that key's data type.

The downloadable Fileset supplies the write script. Configure it with a shell command that prints only the inventory value you want to store:

```
$ fwclд -custom_write -key <key_name> [-value <value_to_save>] [-silent]
```

Requirements

- FileWave Central
- A command that outputs the required inventory value
- Basic scripting knowledge

Directions

The FileWave Client can capture command output and store it in the selected legacy inventory field.

1. Run the command manually, confirm its output type, and select an unused legacy key from the matrix below.

String	Integer	Boolean	DateTime
custom_string_01	custom_integer_01	custom_bool_01	custom_datetime_01
custom_string_02	custom_integer_02	custom_bool_02	custom_datetime_02
custom_string_03	custom_integer_03	custom_bool_03	custom_datetime_03
custom_string_04	custom_integer_04	custom_bool_04	custom_datetime_04
custom_string_05	custom_integer_05	custom_bool_05	custom_datetime_05
custom_string_06	custom_integer_06	custom_bool_06	custom_datetime_06
custom_string_07	custom_integer_07	custom_bool_07	custom_datetime_07

custom_string_08	custom_integer_08	custom_bool_08	custom_datetime_08
custom_string_09	custom_integer_09	custom_bool_09	custom_datetime_09
custom_string_10	custom_integer_10	custom_bool_10	custom_datetime_10
custom_string_11	custom_integer_11	custom_bool_11	custom_datetime_11
custom_string_12	custom_integer_12	custom_bool_12	custom_datetime_12
custom_string_13	custom_integer_13	custom_bool_13	custom_datetime_13
custom_string_14	custom_integer_14	custom_bool_14	custom_datetime_14
custom_string_15	custom_integer_15	custom_bool_15	custom_datetime_15
custom_string_16	custom_integer_16	custom_bool_16	custom_datetime_16
custom_string_17	custom_integer_17	custom_bool_17	custom_datetime_17
custom_string_18	custom_integer_18	custom_bool_18	custom_datetime_18
custom_string_19	custom_integer_19	custom_bool_19	custom_datetime_19
custom_string_20	custom_integer_20	custom_bool_20	custom_datetime_20

- Create a fileset with a script where it uses the clients

macOS

```
/usr/local/sbin/FileWave.app/Contents/MacOS/fwclld -custom_write -key FIELD_TO_SAVE_TO -value INFORMATION_TO_SAVE
```

Windows (FW 15.4.2 and lower)

```
C:\Program Files (x86)\FileWave\fwclld -custom_write -key FIELD_TO_SAVE_TO -value INFORMATION_TO_SAVE
```

Windows (FW v15.5.0 or higher)

```
C:\Program Files\FileWave\client\fwclld -custom_write -key FIELD_TO_SAVE_TO -value INFORMATION_TO_SAVE
```

- Associate this fileset



Date format may be supplied as either YYYY/MM/DDTHH:MM:SS or YYYY-MM-DDTHH:MM:SS. E.g 2014/02/20T15:22:43 or 2014-02-20T15:22:43

Examples

Setting "custom_bool_13" to a false:

```
$ fwclld -custom_write -key custom_bool_13 -value 0
$ fwclld -custom_write -key custom_bool_13 -value false
```

Setting "custom_bool_13" to true:

```
$ fwclld -custom_write -key custom_bool_13 -value 1
$ fwclld -custom_write -key custom_bool_13 -value true
$ fwclld -custom_write -key custom_bool_13 -value something
```

Setting "custom_date_02" to a date:

```
$ fwclld -custom_write -key custom_date_02 -value 2014-02-20T15:22:43
```

To remove any key value, just leave off the -value parameter - so to reset the "custom_date_02" value back to it's default.

```
$ fwclld -custom_write -key custom_date_02
```

Example: Saving admins to string 01

```
#!/bin/sh
# This script is a verification sample
# benm @ fw

now=$(date +%Y-%m-%d-%H-%M")
echo "$now -- Writing current admins to inventory"

#writes the current administrators to an inventory field
currentadmins=$(dscacheutil -q group -a name admin |grep users)

/usr/local/sbin/FileWave.app/Contents/MacOS/fwclld -custom_write -key custom_string_01
-value "$currentadmins"
```

Notes

If you set your script to run at the "verification" phase then it will continue to run (default every 24hrs), for more on scripts see: [Fileset Scripts](#).

Useful Commands

- The current logged in user: `stat -f%Su /dev/console`
 - The Kernel version: `uname -r`
 - Battery Condition: `system_profiler SPPowerDataType | awk '/Condition/ {print $NF}'`
 - Current admins: `dscl . read /Groups/admin GroupMembership | cut -d " " -f 2-`
-

🕒 Revision #5

★ Created 2023-07-12 21:01:52 UTC by Josh Levitsky

✎ Updated 2026-07-30 13:00:24 UTC by Josh Levitsky