Game Collection Settings

Custom collections

Custom collections are deprecated and should be replaced by the more powerful **dynamic collections** described below. Batocera 36 still supports existing static custom collections, but the `batocera-create-collection` helper tool has been removed.

Personal custom collections are stored in `/userdata/system/configs/emulationstation/collections/`. They are simple text files that you can edit manually, or edit through the UI described above.

As an example, a `custom-mario.cfg` file that would look like:

```
/userdata/roms/gameandwatch/Mario Bros. (Nintendo, Multi Screen).mgw
/userdata/roms/nes/Mario Bros (NES).zip
/userdata/roms/gb/Super Mario Land (World) (Rev A).zip
```

If you are using Batocera **5.26** or later, you can use the command `batocera-create-collection` to help you create custom collections.

For example, if you want to create a collection for your “Mario” games like on the example above, you can use it with `batocera-create-collection mario` and it will create a `custom-mario.cfg` custom collection file, with all the games that have “Mario” in their names or descriptions.

You can tune your custom selection with command line switches to select only a specific system, or a specific genres of games, and invoking the command subsequently. For instance, if you want all “ball” games on NES and Gameboy in a “ballers” custom collection, you can start with `batocera-create-collection -c ballers -s nes ball` and then add GB games to the same custom collection by typing `batocera-create-collection -c ballers -s gb ball`. Note that you need to have your games correctly scraped in order to have this command search in the game names (especially for Arcade games on MAME of Final Burn Neo) or in the genres of the games.

From https://github.com/batocera-linux/batocera.linux/blob/master/package/batocera/core/batocera-scripts/scripts/batocera-create-collection

```
batocera-create-collection [ -c collection_name ] [ -g genre ] search_terms
  -c foobar  : The resulting collection will be stored
               as "custom-foobar.cfg" in the collection directory
  -g Shooter : Only games matching "Shooter" <genre> in your gamelist.xml
               files will be added to the collection
  -s snes    : Only games in the /userdata/snes/ directory (to filter out
               systems -- 1 system only after '-s')
  -h         : Display this help file

Note that existing game collections with the same name will have newly
found games added to the list.
```

```
Examples:
  "batocera-create-collection zelda" creates a custom collection
  of all your roms matching "Zelda" in their names.
  You can use it with "final fantasy", and it creates a collection file
  named "custom-final.fantasy.cfg".
  Invoke "batocera-create-collection -c basketball 'nba '" if you want a
list
  of NBA games named "custom-basketbell.cfg", without 'pinball' games in.
  "batocera-create-collection -c plateformers -g Plateform" creates a
  collection of all plarform games (into custom-plateformers.cfg).
  Finally, "batocear-create-collection -c all" creates a custom-all.cfg
  collection with all your games. It can take a while to create, if you have
  a lot of games on your system.
  If you want to create a collection with Zelda games for both nes and gb,
  you can start with "batocera-create-collection -s nes zelda" and then
  add gb games with "batocera-create-collection -s gb zelda".
Gotchas:
  - 'search_terms' for games amd 'systems' are case insensitive...
  - ... but 'genres' are case sensitive (blame scapers disparities)
```

Dynamic collections

Batocera **5.26** and later also let you create dynamic collections from EmulationStation.

Think of dynamic collections like rules you set up in your e-mail client to automatically create “smart folders”: when you receive new e-mails that match the predefined rules, they are showing up in the right smart folder automatically. Dynamic collections on Batocera are the same: you start by creating rules to set up your dynamic collections (match a string in the game name, a gaming system, a genre...) and when you add new ROMs matching those rules, they show up automatically in your dynamic collection. This way, you can share dynamic collections with your friends, even if they don't have the same ROMs, or if their ROM file names aren't exactly the same as yours.

Let's create a dynamic collection for all Capcom Games, for example. Go into **GAME COLLECTION SETTINGS → CREATE NEW DYNAMIC COLLECTION** and call it “Capcom”.

On the next menu you can select the filters your want to apply for this dynamic collection: systems, genre, publisher/developer, number of players, if the game is a favorite or not, rating, safe for kids, language, region already played or not… many options are available there. To pre-populate those option, all your `gamelist.xml` files are parsed, so it **might take a while to load**. For example “publisher / developer” on my system with 12,000 ROMs it takes over a minute to load. For my example here, let's select all the editors that seem relevant for Capcom.

For example, I'd select “CAPCOM”, “CAPCOM CO., LTD.” and “CAPCOM U.S.A., INC.”. Those options are depending on your scraping metadata, so you might have different wordings. Once you have selected your options, you're all set. You have a new Dynamic collection for all Capcom games.

If you want to tweak/edit this collection, from EmulationStation, go into your newly created collection, press **SELECT → EDIT DYNAMIC COLLECTION FILTERS**.

Dynamic collections are stored in `/userdata/system/configs/emulationstation/collections/` as XML files like `Capcom.xcc`

in my example. If you prefer, you can log through SSH on your Batocera and edit the file manually, which has the following format:

```
<?xml version="1.0"?>
<filter name="Capcom">
     <pubDev>CAPCOM CO., LTD.</pubDev>
     <pubDev>CAPCOM</pubDev>
     <pubDev>CAPCOM U.S.A., INC.</pubDev>
</filter>
```

From:
https://wiki.batocera.org/ - **Batocera.linux - Wiki**

Permanent link:
https://wiki.batocera.org/game_collection_settings?rev=1680034003

Last update: **2023/03/28 20:06**