Inventory Tweaks Documentation
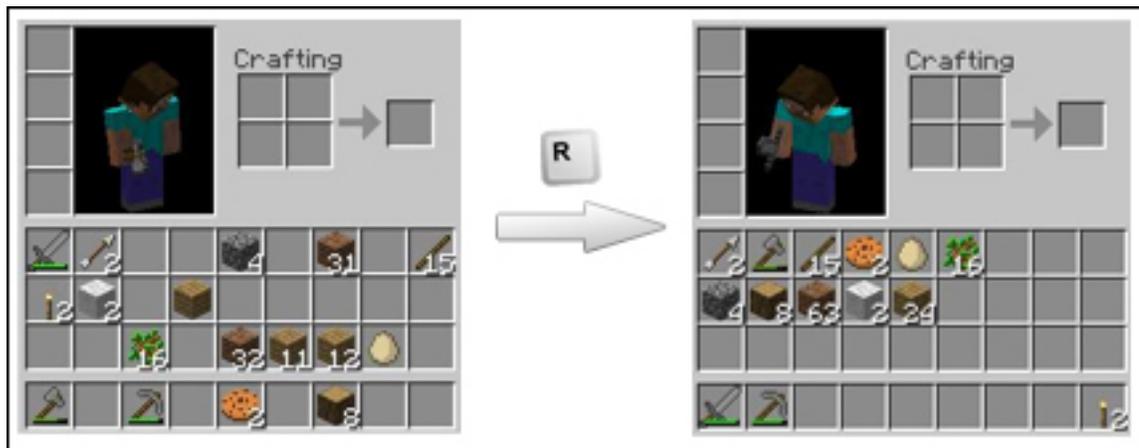
Release 1.57

Jimeo Wan

March 14, 2014

Contents

What is Inventory Tweaks?

This client mod for Minecraft allows you to **easily manage your items**. It works out of the box, and will make you gain a massive amount of time, both in **single and multiplayer**!

- Tools and items are automatically replaced

- Sort your inventory and your chests in a single click/keystroke

- Customize entirely the way your inventory is sorted, thanks to configuration files

Download

Inventory Tweaks 1.57 (API) for **Minecraft 1.7.2** (requires Forge)

Inventory Tweaks 1.56 (API) for **Minecraft 1.6.2** (requires Forge)

Inventory Tweaks 1.55b (API) for **Minecraft 1.6.2** (requires Forge)

Inventory Tweaks 1.55 (API) for **Minecraft 1.6.2** (requires Forge)

Inventory Tweaks 1.54 (API) for **Minecraft 1.5/1.5.1/1.5.2** (requires Forge)

Inventory Tweaks 1.53 (API) for **Minecraft 1.5/1.5.1** (requires Forge)

Inventory Tweaks 1.52 (API) for **Minecraft 1.5/1.5.1** (requires Forge)

Inventory Tweaks 1.50 for **Minecraft 1.4.6/1.4.7** (requires Forge or ModLoader) | Mirror

Inventory Tweaks 1.45 for **Minecraft 1.4.4/1.4.5** (requires Forge or ModLoader) | Mirror

Inventory Tweaks 1.43b for **Minecraft 1.3.2** (requires Forge or ModLoader) | Mirror

- Older versions
- Latest changes

Contact

- Get support on Minecraft Forums
- Join the project on Github (yep, it's Open Source!)

Getting started

4.1 Setup

Once you have MinecraftForge, this mod is installed by simply moving the JAR to the **mods** folder of Minecraft.

4.2 Basic use

Just start playing Minecraft, you can already use most features of the mod. Notice the "..." button that just appeared on the top-right corner of your inventory? This leads to the mod settings screen, if it's there it means that the mod is installed correctly.

4.2.1 Inventory sorting

Trigger the sorting in your inventory by using the **middle click**, or by pressing **R** (by default), at any time. If you want to change the shortcut, you can do that from the "..." menu in your inventory.

Tip: The middle click shortcut behaves differently if you target the **crafting input**: in that case, it will adjust the item amounts to balance them, helping optimize your recipe input. For example, if you want to make sticks, it will make sure your wood will be split equally between the top and the bottom stack.

4.2.2 Chests sorting

When you open a chest, you will see three additional buttons on the higher part of the screen. They allow to sort items in different ways ; but you can also use the middle click if you prefer (click several times to switch the sorting method).

4.2.3 Auto-refill

If you eat an item, break a tool or empty a stack by placing blocks, the mod will automatically fetch if possible a similar item from your inventory.

Note:

- Click on the "..." button from your inventory to explore the features (yay, **inventory shortcuts**!) and customize the settings of the mod.

- You don't even have to be in your inventory to use the **R** shortcut.

Working with rules

One of the most powerful features of this mod is its ability to customize the sorting feature. For instance, you can make sure that each time you sort your inventory, the best pickaxe available is put in your hotbar, but there is really a lot that can be done.

5.1 First steps

Everything is done by editing a configuration file, which is in your Minecraft folder, at *config/InvTweaksRules.txt*. You can also open it by opening your inventory in-game, clicking the *"..."* button, then choosing *"Open the sorting rules file..."*.

In this file, each line you type is a new constraint you add for sorting items. Basically each rule you will add (such as `D LOCKED` which is here by default) is a new line with the format: `XX YYYYYY` where `XX` is the *target*, and `YYYYYYY` is the *keyword*. For instance, if you add a new line that says `D5 pickaxe`, then go back to Minecraft and sort your inventory, it will put the best pickaxe available in the `D` row (your hotbar), `5th` column. Awesome, right?

Note:

- There's no need to restart Minecraft when you edit your configuration. Just save, switch to Minecraft, sort!

- If you didn't guess it, `A` is the top row of your inventory, while `D` is the bottom (i.e. hotbar). `1` and `9` are respectively the left and right columns.

5.2 Available targets

- **A particular slot**: `D1 sword`

- **A whole row**, to sort from left to right: `D sword`

- **A whole column**, to sort from bottom to top: `1 sword`

- **A rectangle**, defined by two slots: `A5-C9 stuff`

Note: Regarding rectangle targets, "A1-C3", "C3-A1", "A3-C1" and "C1-A3" don't behave exactly the same, choose the one that best suits your needs.

5.2.1 Target modifiers

Add them at the end of the targets (example: `Ar`) to apply their effect.

- **Reverse modifier**, to sort from right to left or top to bottom: `r`
- **Vertical sorting**, for rectangle rules: `v`

5.3 Available keywords

All item names are supported, and even a whole lot of category keywords, such as `edibleFood`, `blocks`, `items`, `equipment`... The whole list is stored in the file *config/InvTweaksTree.txt* ; an online version is available here. Also, don't worry about syntax, for instance `woodenSword` works as well as `WOODSWORD` or `Wooden Sword`.

Here are the special keywords you can use:

- **Locked slots**: `LOCKED`. This make items from your selection not move during sorting (e.g. `D LOCKED`). However, the LOCKED keyword is overriden by rules with finer targets (`D LOCKED` is overriden by `D1 torch` or `1 swords`).
- **Frozen slots**: `FROZEN`. Items in frozen slots never move during sorting or auto-refill (e.g. `D8 FROZEN`).
- **Items by IDs**: If an item doesn't have a keyword (such as mod items), but you know its ID, you can still select it thanks to ID rules (e.g. `D1 489`, or `D1 489-5` for a specific damage value).

5.4 Multiple configurations

It is possible to have several set of rules you can swap between. Start a new configuration block by adding a rule in a single word ending with a colon:

```
Mining:
D1 pickaxe

Fighting:
D1 sword
```

You can then swap between those configurations by either:

- Holding **R** for 1 second
- Pressing **1-9 + R**

Here, **1 + R** would load the Mining config, **2 + R** the Fighting config, etc.

5.5 Example

Here is the config the mod author uses:

```
Default:

D1 pickaxe
D2 axe
D3 shovel
D4 sword
D5 edibleFood
```

```
D9 torch

A1-C5v stuff
A9-C6v blocks
```

Tweaking the item tree

6.1 First steps

This file is where are stored the keywords you can use to create rules. It also specifies the item order in the categories. It can be modified as you want, giving you the opportunity to make custom keywords for your rules:

```
<myCategory>
  <dirt id="dirt" />
  <cake id="cake" />
</myCategory>
```

Now, if you create a rule using the "*myCategory*" keyword, it will match only dirt and cake.

Note:

- Here, you don't need to remove "dirt" and "cake" from their default categories.

- It's recommended to add any custom category to the top of the item tree, just below the `<stuff>` tag.

- Within a category, the order of the items determines which item is preferred. For instance, if you make a `D1 sword` rules, iron swords will be picked first, since it appears first in the item tree. Feel free to tweak that!

6.2 Items defined by IDs and damage

Some items are not only defined by IDs, but also by specific "damage values":

```
<sexyWools>
  <magentaWool id="wool" damage="2" />
  <pinkWool id="wool" damage="6" />
  <purpleWool id="wool" damage="10" />
</sexyWools>
```

6.3 Categories defined by Forge ore Dictionary Names

You can also define categories by entering an ore dictionary name. Useful for mods:

```
<camelOre>
  <camelOreBlock oreDictName="oreCamel"/>
```

```
  <camelOreIngot oreDictName="ingotCamel"/>
</camelOre>
```

Adding configured compatibility info

Additional compatibility info can be added to the file 'config/InvTweaksCompatibility.xml'. This will not work with all mods, as it is only capable of the most basic info required. Ask mod authors for API compatibility if their containers do not behave correctly. The format for entries is fairly simple:

```
<root>
    <chest class="mod.container.MyChestContainer" row_size="9" large_chest="false"> <!-- Chests to sho
    <inventory class="mod.container.MyInventoryContainer" disable_buttons="false"> <!-- Other invento
</root>
```

F.A.Q.

If you're having troubles making the mod work, or can't manage how to customize things as you want, tell your problem on the Github Issue Tracker, or the Minecraft Forum topic, other users and I will be glad to help!

- **I can't sort items from my mods like I want!**

Currently, you have two solutions, the first being to make rules according to the item ID (e.g. `D1 somemod:someItem` or `D1 somemod:someItem-5` for a specific damage value).

Second solution, edit the *InvTweaksTree.txt* file from the config folder, to manually add the items you want to sort. Each item is defined by its ID, and the easiest way to find out which item is which ID is to use something like the Not Enough Items mod.

- **The last update broke my InvTweaksTree.txt file!**

Don't worry, you'll find a backup in your config folder. It's just that InvTweaks replaced the tree with a new version, up-to-date with the latest Minecraft items.

- **The config files don't appear!**

You're probably running a platform that is not 100% supported by Java. In any case, what you can do is extract manually these files from the Zip: *DefaultConfig.dat* should go to **{MinecraftDir}/config/InvTweaksConfig.txt**, and *DefaultTree.dat* to **{MinecraftDir}/config/InvTweaksTree.txt**.

- **The "..." icon looks bad with my texture pack!**

By default, the mod uses *assets/minecraft/textures/gui/widgets.png* to draw the small buttons for the inventory and chests. It looks good most of the time, but you can override this by adding a **assets/inventorytweaks/textures/gui/button10px.png** file to your pack. Here is a template.