

Datapack: Bundle-Based RuneForge System

Lj_1

2025-04-27

Contents

1	Player Guide	1
1.1	More Features	2
1.2	Warnings (IMPORTANT)	3
2	Map Maker Guide	3
2.1	Forgeable Items	3
2.2	Customizing Runes	3
2.3	Customizing Items	3
2.4	Configuration	3
2.5	Examples	4
A	Update Logs	4

1 Player Guide

When placing tools or armor (referred to as *items*) in the 2×2 crafting grid, you can modify their *runes* as if they were a bundle. Each *runes* enhances the *item* by adding enchantments, modifying attributes, or granting custom abilities.

This datapack is designed to be **collaboration-friendly**: in a collaborative map, each creator can design custom runes without altering the RuneForge system's core code.



Figure 1: Modifying runes on an item (bundle-like interface).



Figure 2: Example runes.

- `Lj_1 - [1.21.5] Bundle-Based RuneForge System` is the core datapack;
- `Lj_1 - [1.21.5] Bundle-Based RuneForge System - Examples` provides example runes. Execute `/function lj.bundle_forge.examples:get_runes` to obtain them. Also execute `/function lj.core:__install__` to initialize scoreboard objectives.

1.1 More Features

Each rune occupies $1/k$ (for some integer $k \geq 1$) of the bundle's capacity, limiting the number of runes per item. In the examples, rune weights are color-coded:

Color	Weight
Green	1/6
Blue	1/4
Purple	1/3
Gold	1/2



Figure 3: These runes conflict because both of them are Attribute Upgrade Runes

Some runes conflict with each other, which means they cannot be applied to an item simultaneously. In the examples, runes with the same yellow-labeled categories conflict.

1.2 Warnings (IMPORTANT)

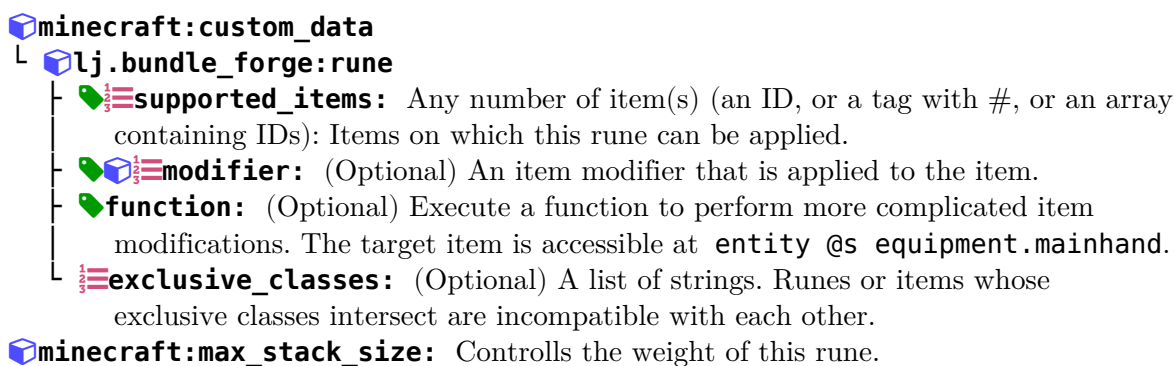
Other modifications (e.g., from anvils, smithing tables, or enchanting tables) **may not** persist when using the 2×2 crafting grid. The RuneForge system calculates results based on the original item, ignoring most external changes. Nevertheless, you can configure which fields persist. See Section 2.4 for more details.

2 Map Maker Guide

2.1 Forgeable Items

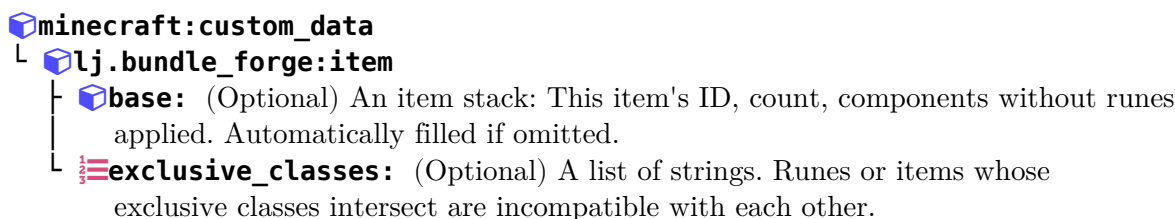
The tag `lj.bundle_forge:forgeable` determines which items display the bundle interface.

2.2 Customizing Runes

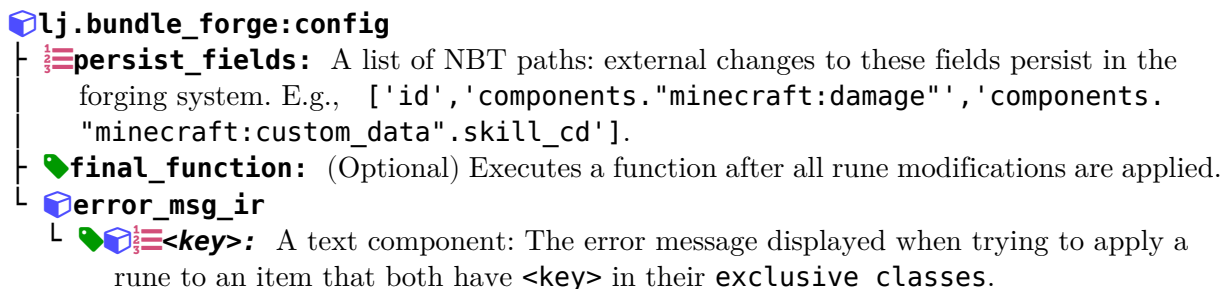


Also customize `minecraft:item_name`, `minecraft:lore`, etc., for visual polish.

2.3 Customizing Items



2.4 Configuration



```
[RuneForge] [Iron Sword] cannot accept [Example Rune - Basic Sharpness] - This item already has enchantments
```

2.5 Examples

The following command grants a rune that applies **Sharpness I** to swords:

```
/give @s minecraft:written_book[minecraft:custom_data={"lj.bundle_forge:rune":{"supported_items":"#minecraft:enchantable/sword",modifier:{function:"minecraft:set_enchantments",enchancements":{"minecraft:sharpness":1}},exclusive_classes:["enchant"]}},minecraft:max_stack_size=6]
```

This command creates a rune that sets swords' attack speed to 2.4:

```
/give @s minecraft:written_book[minecraft:custom_data={"lj.bundle_forge:rune":{"supported_items":"#minecraft:enchantable/sword",modifier:{function:"minecraft:set_attributes",modifiers:[{attribute:"minecraft:attack_speed",operation:"add_value",amount:-1.6,id:"minecraft:base_attack_speed",slot:"mainhand"}],replace:false},exclusive_classes:["attribute"]}},minecraft:max_stack_size=6]
```

Acknowledgments

This datapack is inspired by Ragecraft IV's RuneForge system and Cartographer 2.0's Encyclopedia.

A Update Logs

- **2024-04-27**
 - Fixed some bugs.
- **2024-04-10**
 - More complicated item modifications can now be executed through functions.
- **2024-04-08**
 - Added persistent fields configuration to support tracking of item durability, netherite upgrade, etc..
- **2024-04-05**
 - Initial version of Bundle-Based RuneForge System.