

# Input handling

- [Types of Input](#)
- [Input Buffers](#)

# Types of Input

## **Loose Item Input:**

The simplest form of input, loose item input is when items are placed directly into the system as-is, either by throwing them into a water stream or into a chest/barrel. Loose items inputs sometimes automatically sort out unstackables (such as tools, armor, and potions) and redirect them to nonstackable storage.

In more complex systems, loose items are placed into shulker boxes, either by a manual loader or chest into a mixed loader.

## **Shulker Box Input**

In nearly all cases, it is more efficient to input into a storage using shulker boxes (note that this requires an unloader, see the Unloaders page for details). Because of this, most large storages will have an input UI (user interface) that allows direct input of mixed-item-type shulker boxes into the system.

This input type usually requires protection from accidental loose item input using a shulker box sorter (see the nonstackable sorting page). The items sorted out are usually returned to the player or loaded into a box.

## **Direct Bulk Input**

The following section only applies to encoded and box sorting-based bulk systems.

High volumes of single-item-type, full shulker boxes will slow down a storage and waste time splitting, unloading, and merging. If the item type in the box is in the bulk storage, then it is sometimes impossible to manually place a box into a chest (as is recommended for already full, single-item-type boxes). To solve this issue, input systems sometimes include a chest that leads directly to the input of the bulk hall, bypassing the splitting array and unloading array.

This is not possible for bulks that use loaders at the top of the slice, as they only accept loose items as input.

# Input Buffers

Input buffers are an optional component of an automated storage system. The input buffer will accept items from the input UI (usually in the form of mixed-item-type shulker boxes) and hold them until the storage is given the instruction to "run".

Input buffers are usually used in the case that a storage does not have a ticking area attached to it, as a player cannot walk away from a running storage without danger of unloading it. The buffer allows the player to choose when the storage will run, and therefore allows them to leave immediately after inputting items.

Input buffers are also sometimes used as a speed optimization. Unloading arrays can only be fully utilized when their input volume is larger than a single box, and the more shulker boxes past that single one, the higher the rate that items will be stored. Thus, holding the input until at least a few boxes have built up is better for raw speed.

Input buffers have a minor drawback in that, until the storage is allowed to run, the most recently inputted items are inconvenient to access.