

Author: Jayke Paver

Module Type: Additive Module

Compatible With: Frontiers Beta 2026-2 and any system using the default Equipment rules.

Version: Beta 2026-2 Revision

Original Release: April 24, 2026

Current Revision: April 24, 2026

This document is a **modular addition** to the Frontiers Engine. It is one possible implementation of inventory tracking within the engine, not the definitive one.

The core Frontiers rules do not include carrying capacity by default. The engine assumes a character carries whatever makes narrative sense for their situation. This works well for systems where inventory is not a central pressure (heroic fantasy, cinematic action, most social or investigative games), but falls short in systems where the question "what can you fit in your pack?" matters to the tone.

This module provides one slot-based answer to that question.

The Carrying Capacity Module is a reference implementation, not the "correct" way to handle inventory in a Frontiers-based system. Other modules might use weight-based tracking, encumbrance tiers, or entirely freeform approaches. This one uses discrete **slots** because slots are fast to track, easy to teach, and scale cleanly across genres. If your system needs a different approach, build your own module or adapt this one.

This module serves three purposes:

First, it offers one clear and adaptable framework for inventory tracking using the existing engine structure.

Second, it demonstrates how an inventory subsystem can attach cleanly to the Frontiers Engine without modifying its foundation.

Third, it provides a reasonable starting implementation that system designers may adopt as-is, modify, expand, or entirely replace.

Use of this module is optional. It is supported and endorsed, but not mandatory.

Systems built on Frontiers may:

- Adopt this module as written
- Modify portions of it

- Expand it with additional inventory layers
- Replace it entirely with a different approach
- Omit inventory tracking altogether

This module assumes use of the default Equipment rules. If a system alters those foundations, adjustments may be required.

Slot-based inventory trades fine granularity for speed. Weight tracking is more precise, but slot tracking is faster to adjudicate at the table. The table asks "does it fit?" rather than "what's the total in pounds?"

Slots also scale well across genres. A fantasy slot system tracks potions, scrolls, torches, and provisions. A sci-fi slot system tracks power cells, tools, weapon mods, and rations. The slot count stays roughly the same; only the items change. This makes the module genre-portable without rework.

The specific slot values in this module are tuned for a character who should feel prepared but not infinite. A starting character carries about 9-10 slots of gear. A character with high ♦Vigor carries up to 13. Filling every slot is possible but means every slot is committed.

Other modules might tune these numbers differently. A survival horror module might cut the capacity in half. A heroic module might double it. Both would be valid. This one sits in the middle.

Each character has a **Slot Capacity** determined by the following formula:

$$\text{Slot Capacity} = 8 + \text{♦Vigor Modifier}$$

This produces a baseline range from 8 slots (Vigor Modifier +0) to 13 slots (Vigor Modifier +5).

Slot Capacity is a Live Derivative: it updates immediately when ♦Vigor changes. If a character's ♦Vigor drops, their Slot Capacity drops, and they may become Overburdened as a result.

Systems may add Slot Capacity bonuses from other sources (Archetype features, Abilities, specific Equipment like backpacks or harnesses). Those bonuses stack additively with the base formula.

Slot Tracking Scope

This module tracks only items that are meaningful to carry. It does not track:

- Worn clothing and basic garments (unless genre-specific armor rules apply)
- Items marked **Negligible** (see below)
- Items inside Containers, which track against the Container's internal capacity

The intent is that the slot count represents operational gear, the kit the character would audit if asked "what do you have on you?"

Negligible Items

Some items are small, numerous, or cumulative in ways that make individual tracking pointless. These are treated as **Negligible** and do not consume slots.

Common Negligible items include:

- Ammunition (arrows, bullets, thrown knives, power cells) tracked by the system in bulk rather than per-slot
- Small personal effects (writing materials, a deck of cards, a journal, a smoking pipe)
- Rations assumed as a character's "standard supply" rather than tracked individually
- Coin, gems, or credit chips (unless a system specifies that wealth tracking uses slots)
- Trivial consumables (a flask of water, a few candles)

The system designer defines what counts as Negligible. The rule of thumb: if tracking it item-by-item would add overhead without meaningful decisions, mark it Negligible.

Negligible does not mean powerless. A Negligible item may still require Competency, may still have Level interaction, and may still affect play. It simply doesn't consume a slot.

Each non-Negligible item has a size that determines how many slots it occupies.

Size	Slots	Examples
Small	1	Daggers, potions, hand tools, medkits, books, torches, pouches
Medium	2	One-handed weapons, light armor, shortbows, kits and toolsets, rope coils
Large	3	Two-handed weapons, heavy armor, shields, longbows, large instruments
Massive	5	Siege components, rare signature items, unusually bulky gear

The boundary between sizes is determined by the system designer. Most items fall into Small or Medium; Large is for substantial gear; Massive is rare and usually involves GM adjudication.

System designers may subdivide these categories if they want finer granularity (e.g., Light 1-slot / Standard 2-slot / Heavy 3-slot / Massive 5-slot), or merge them if they want less. The four-size model above is the recommended default.

Containers are items that hold other items. They reduce slot pressure by consolidating multiple small items into a single tracked element.

A Container has:

- An **External Size** (how many slots it takes in the character's inventory when carried)
- An **Internal Capacity** (how many slots of gear it can hold)

Example Containers

Container	External Size	Internal Capacity	Notes
Belt Pouch	1 slot	2 slots	Quick access
Satchel	1 slot	4 slots	Single shoulder strap
Backpack	2 slots	6 slots	Both shoulders, more capacity
Saddle Bag	2 slots	8 slots	Mounted or attached to a vehicle
Large Chest	5 slots	15 slots	Usually left at a home base

Containers are system-defined. The values above are examples a designer can adopt or modify. Enhanced Containers (bags of holding, dimensional pockets, tech-stowage modules) follow Enhanced Equipment rules and may have much larger internal capacities.

Accessing Items in Containers

Items inside a Container are **Carried**, not **Equipped**, by default. (See the Equipment design doc for the Equipped / Carried / Stored distinction.)

Retrieving an item from a Container during Encounter Play typically costs **1 ♦AP** unless the Container's design specifies otherwise. Some quick-access Containers (belt pouches, holsters, combat harnesses) may reduce or remove this cost.

Items inside a Container do not apply passive bonuses, even if the item would normally do so when Equipped. A character must retrieve and Equip the item first.

Container Overflow

A Container cannot hold more slots of gear than its Internal Capacity. If a character tries to stow an item that doesn't fit, they must either:

- Drop the item
- Remove something else from the Container
- Choose a different Container

Overflowing a Container does not produce Overburdened penalties on its own, but if the total slot count across the character and all Containers exceeds Slot Capacity, Overburdened applies.

If the sum of a character's carried slots exceeds their Slot Capacity, they become **Overburdened**.

Overburdened imposes:

- **-1 Movement Point per Turn** (reducing starting MP by 1)
- **Disfavor on physical Resolution Rolls** (any roll where the governing Attribute is ♦Vigor or □Finesse)

These penalties persist until the character reduces their carried slots to Slot Capacity or below.

Overburdened and Initiative

If a character becomes Overburdened **before an Encounter begins**, they also suffer **-2 to their Initiative Value** for that Encounter.

If a character becomes Overburdened **during an Encounter** (by picking up heavy gear mid-fight, by a ♦Vigor reduction, or by any other means), the Initiative penalty does not apply to the current Encounter. Their locked Initiative for that Encounter stays as it was. The -2 applies to their next Encounter if they are still Overburdened at that time.

This rule matches the engine's Live vs. Anchored Derivative philosophy. Initiative, once locked, does not recalculate mid-Encounter. Movement Points and Resolution Rolls are Live Derivatives and update immediately.

Severe Overburden

If a character carries more than **150% of their Slot Capacity**, they are **Severely Overburdened**. In addition to the standard Overburdened penalties:

- They cannot Run or Dash (movement is walking pace only)
- They cannot voluntarily enter Encounter Play without dropping gear first (they are mechanically unprepared for combat)
- The GM may restrict specific actions entirely based on the fiction

At the GM's discretion, a Severely Overburdened character may be unable to move at all if the weight is truly extreme (a character carrying 4x their Slot Capacity is functionally immobile).

The Equipment design doc defines three item states. This module interacts with all three:

State	Slot Interaction	Usability
Equipped	Counts toward Slot Capacity	Passive bonuses apply; usable without draw cost
Carried	Counts toward Slot Capacity	Requires 1 ♦AP to retrieve and Equip
Stored	Does NOT count toward Slot Capacity	Not immediately accessible; may require travel, retrieval, or time

Stored items are typically kept at a home base, in a vehicle, on a mount left at the edge of the dungeon, or in a safe location the character cannot reach quickly. They are still the character's property but not their operational kit.

The distinction between Carried and Stored is fictional as much as mechanical. A character walking through a dungeon with a heavy chest on a pack mule has the chest Stored, not Carried. A character walking through a dungeon with a backpack on their shoulders has the backpack and its contents Carried.

Transitioning Between States

- **Equipped → Carried:** 1 ♦AP (removing a weapon from hand, unbuckling armor)
- **Carried → Equipped:** 1 ♦AP (drawing a weapon, donning a piece of armor quickly). Full armor changes may take longer (see below).
- **Carried → Stored:** Requires access to a storage location (base, vehicle, mount, etc.)
- **Stored → Carried:** Requires physical retrieval from the storage location

Full armor swaps typically cannot be done mid-Encounter. Changing from light to heavy armor (or vice versa) takes minutes, not seconds. This is GM adjudication.

Slot Capacity is a Live Derivative and updates immediately when:

- ♦Vigor changes (permanently or temporarily)
- An Archetype feature adds or removes Slot bonus
- An Ability granting Slot bonus is activated, deactivated, or ends
- Equipment granting Slot bonus is Equipped or Unequipped

When Slot Capacity decreases, the character may become Overburdened. When Slot Capacity increases, Overburdened may be lifted.

If a character is mid-Encounter when Slot Capacity changes, the effects on ⇒Movement and Resolution Roll Disfavor apply immediately. The Initiative penalty does not apply mid-Encounter (as noted above).

The following sections are optional guidance for system designers integrating this module. Use them as needed; they are not part of the core module rules.

Tuning Slot Capacity for Tone

The default formula ($8 + \text{♦Vigor Modifier}$) targets a middle-ground feel. Systems with different tones may want to adjust:

Tone	Suggested Formula	Effect
Survival / Horror	$5 + \text{♦Vigor Modifier}$	Characters feel constantly underequipped; loot choices are painful
Default	$8 + \text{♦Vigor Modifier}$	Balanced. Characters feel prepared but finite
Heroic / High Fantasy	$12 + \text{♦Vigor Modifier}$	Characters feel capable; inventory rarely forces hard choices
Minimalist / Lifestyle	$10 + \text{♦Vigor Modifier}$, but halve item slot costs	Characters carry more total, less bookkeeping

The specific value chosen should reflect what kind of decisions the designer wants players to face.

Integrating with Equipment Levels

The default Equipment rules already create pressure through Reference Level interaction: Enhanced items above a character's Reference Level reduce their maximum $\nearrow$ Energy. This module adds physical inventory pressure on top of that.

A character may be Slot-limited (can't carry any more gear) while still having $\nearrow$ Energy headroom (can Equip more Enhanced items), or vice versa. Both pressures apply simultaneously and independently.

Systems may choose to link the two more tightly (e.g., "Enhanced items count as one additional slot"), but the engine default keeps them separate.

Integrating with Competencies

Carrying an item does not require Competency. Using it may. A character without the Heavy Armor Competency can still carry Heavy Armor in their backpack; they just can't Equip it without penalty.

This distinction lets characters transport gear for others, loot without committing, or prepare for situations where they might gain a Competency later.

Currency and Treasure

The default module treats coin and portable wealth as Negligible. For systems where wealth should have physical weight:

- Small amounts (pocket change, a day's earnings): Negligible
- Moderate amounts (a week's wages, a small haul): 1 slot per significant amount
- Large amounts (a treasure chest, a king's ransom): Multiple slots or a Container requirement

The specifics depend on the system's economy. A gritty treasure-hunting game might track every coin; a heroic game might abstract wealth entirely.

Vehicles, Mounts, and Shared Inventory

Vehicles and mounts typically have their own Slot Capacity (much larger than a character's) and can hold Stored items the party doesn't need to reach quickly. Common defaults:

- **Pack Mule / Riding Horse:** 30-40 slot capacity
- **Wagon / Cart:** 100+ slots
- **Small Vehicle (cart, rowboat):** 50-100 slots
- **Large Vehicle (ship, spacecraft):** 500+ slots, often tracked in different units entirely

Items on a vehicle or mount are Stored, not Carried, from the character's perspective. If the vehicle is destroyed or lost, the items may go with it.

Shared party inventory (a group chest, a communal pack) works the same way. It exists outside individual Slot Capacities but may require a specific character to retrieve items from it.

Adding inventory tracking to a system changes what the game is about, not just how it's tracked. Slot pressure is a storytelling tool.

When inventory matters, characters make choices that express their values. Do they carry the extra rations or the extra weapon? Do they bring the medkit for emergencies or the climbing gear for the route? Do they leave behind their loot to carry a wounded ally?

These choices emerge from scarcity. A system with no inventory pressure can tell many kinds of stories, but not stories about scarcity. A system with heavy inventory pressure tells those stories constantly.

Pick the pressure level that matches the game. A game about heroes should have light pressure. A game about survivors should have heavy pressure. A game about explorers might be in the middle.

To adopt this Carrying Capacity Module within a Frontiers-based system:

1. **Confirm engine compatibility.** This module assumes the default Equipment rules of Frontiers Beta 2026-2. If your system modifies those foundations, adjustments may be required.
2. **Decide on the Slot Capacity formula.** Default is $8 + \text{Vigor Modifier}$. Pick a different base value if your system's tone warrants it.
3. **Define your Negligible items.** List the items that should not consume slots. Be consistent.
4. **Define your item sizes.** Categorize gear as Small, Medium, Large, or Massive. Consider adding sub-categories if your system needs finer granularity.
5. **Define your Containers.** Decide what Containers are available in your system, their External Sizes, and their Internal Capacities.
6. **Decide on currency handling.** Abstract or tracked? If tracked, at what level of granularity?
7. **Decide on vehicle and shared inventory rules.** Include them if your genre calls for it; skip them if it doesn't.
8. **Playtest.** Inventory pressure is emergent from the combination of Slot Capacity, item sizes, and what players want to carry. Adjust based on actual play.

Attribution Line for Systems Using This Module:

Includes material from the **Carrying Capacity Module** by Jayke Paver, Beta 2026-2 Revision.

This module may stand alone as the inventory system for a system, or it may serve as a foundation for additional layers (weight tracking, durability interaction, specific gear-slot systems). Systems that find this module's approach too strict or too loose for their tone should feel free to build their own inventory module instead. Frontiers' openness extends to inventory just as much as to any other subsystem.

For the engine's broader design philosophy, modularity, and the Variant and Alternative convention, see **Designing With Frontiers**.

For the condensed working ruleset, see the **Frontiers Overview**. For full Equipment rules, see the **Equipment design doc**. This module is released under the **Open RPG Creative (ORC) License**.