

This document is part of the **Frontiers Design Documentation**, a set written to teach instead of to instruct. Each entry takes one subsystem, explains what that subsystem is in the wider language of tabletop design, surveys how games have solved it, and then shows what Frontiers does as one worked answer among many.

It is written for several readers at once:

- Designers using Frontiers who want to understand why it works the way it does.
- Designers using Frontiers who want to change it, and need to know what a change will cost.
- Designers building something else entirely, who want the design space mapped before they choose.

Frontiers is a toolkit, and a toolkit is only useful if you can see what each tool is for. None of this is a rule; the rules live in the Rulesets, and what follows is the reasoning, the history, and the alternatives.

There will be terminology here that is unfamiliar to casual players. Treat this documentation as advanced content, written for designers interested in building. It will not teach you every term or phrase it uses, but it will do its best to define them as they arrive.

What are Abilities

Abilities are the most nebulous subsystem a game has, and it is worth being honest about that before anything else.

Attributes are numbers and Skills are tags, where an ability has no fixed form at all. It can be a permanent bonus, a new action, a rewrite of how you take damage, a resource you spend, a thing that happens when a condition is met. Because it can be almost anything, there is no single definition that closes the case. Any attempt at one either excludes abilities that clearly count or grows so broad it means nothing.

So this document makes no claim to the final word on what an ability is, which is a much longer and more subtextual conversation than a teaching document should try to have. What it will give you instead is a simple, usable definition, one with as many clear bounds as can be drawn without drowning the idea in exceptions. It is a designer's-eye view, not a philosopher's, because a new designer needs a rubric they can build against, not a perfect account they cannot use.

Take everything here as one writer's practical framing rather than a universal law. It earns its place by making abilities easy to reason about.

The lock and the key

The simplest test for whether something is an ability is whether it is **gated**.

If everyone can do it by default, it usually falls outside the category: it is play, a basic action, or a roll anyone is allowed to attempt. In most systems, moving is not an ability, and attempting to pick a lock is not an ability, because these are things any character is assumed to be able to do.

But hold that lightly: whether those things are ungated is itself a design choice. A game can absolutely decide that movement is a gated resource, or that picking a lock requires a specific unlock, and some do. The default is not a law, only the common case.

If something is locked behind a class, a choice at creation, a level, or a resource, and you had to get it, then it is an ability. The gate makes the whole distinction. And this holds even when the ability is a better version of something everyone can already do.

Take a generic case: in many systems any character can Dash to move faster, so dashing is not an ability there. But a trait that lets only certain characters double their movement is the ability version of that same effect. Identical outcome, different access. One is available to all; the other was unlocked.

An ability need not grant anything new, either; it can change how you do something you could already do. A trait that gives you Favor when picking a lock has not handed you a new action, you could always try the lock, but it has modified the terms on which you attempt it, and that modification is itself the ability. Unlocking a new interaction and changing the terms of an existing one are both abilities. What they share is the gate.

Where a system draws that gate is a design decision, and nothing about the fiction settles it for you. That is the important part, and it is easiest to see across real games. In the 5e SRD, many of the core things a character does are formally features and actions the rules grant, so a great deal of ordinary play is, by that game's own framing, the use of an ability.

Pathfinder gates a number of its actions behind training, so an action you cannot even attempt untrained becomes available only once a skill unlocks it, which makes the action itself an ability that the skill hands you. Cypher draws the line somewhere else again, treating much of what a character can do as abilities purchased through their type and descriptor, so the character's whole capability set reads as a list of granted abilities, not a baseline everyone shares.

None of those three is more correct than the others. Each decided where its gate sits, and that decision shaped how the whole game gets built and played. The freedom is exactly why the word resists a fixed definition, and why the rest of this section maps the space instead of trying to pin a point in it.

The grid

Once something is an ability, two questions describe almost any of them: **how it presents**, and **what it costs**.

How it presents:

- **Passive.** You are told you have it, and it is simply true. It requires no activation and asks nothing of you in the moment.
- **Active.** You invoke it. It happens because you chose to make it happen, usually as part of a turn.

What it costs:

- **Free.** It costs nothing to use. A passive bonus, or an active effect you can invoke at will.
- **Single-cost.** It costs one kind of thing: a use, a point, an action, a resource.
- **Compound-cost.** It costs more than one thing at once, most often an action and a resource together.

Crossing the two gives a small grid, and most abilities in most games fall somewhere inside it:

	Free	Single-cost	Compound-cost
Passive	A permanent trait or standing bonus	A passive that draws on a resource to stay active	Rare: an always-on effect that continuously spends more than one thing
Active	An at-will action anyone with it can use	An action that spends a point, a use, or a resource	An action that spends a resource and an action cost together

A few examples show how the same idea lands in different cells depending on the system:

- A permanent +1 to a defense is **passive / free**. You have it, it costs nothing, it never switches off.
- A 5e SRD spell is **active / single-cost**: you take an action and spend a slot. Cantrips move to **active / free**, since they cost the action but no slot.
- A Pathfinder skill-gated action is often **active / single-cost**, spending an action once the training gate is cleared.
- A Cypher ability that you push by spending from a stat pool is **active / compound-cost** when it also takes your action for the turn: a resource and an action at once.
- A stance that continuously drains a resource to stay up is **passive / single-cost**, the rare living entry in the passive row.

Two things about the grid matter more than the cells themselves.

First, neither a system nor a single ability is confined to one square. A game can spread its abilities across the whole grid, and one ability can move between cells depending on the situation, which is a possibility worth designing around instead of against.

Second, the grid maps the whole space, and no single game occupies all of it. Where a system's abilities cluster tells you what kind of game it is, and the section that follows uses real systems to show that.

Why it matters

Abilities are where a system does most of its balancing, and the grid is why.

An ability's power comes from three things at once: what it does, what it costs, and how often it can be brought to bear. A free passive and a compound-cost action can produce the same effect and be worth wildly different amounts, because one is always on and the other is rationed twice over. When abilities feel unbalanced, the effect is rarely the culprit. The cost and the presentation usually are.

It is also the subsystem a game built on an engine defines most heavily for itself. An engine can set how abilities are costed and written, though the actual abilities, what they do and who gets them, are content. That boundary is where the Frontiers section picks up.

When you give a character something they could not do before, what did it cost them, and how often can they do it?

The Approaches

Abilities are rarely a place where a system picks one approach, and the grid already showed why: games spread across it, and a single character often occupies several cells at once. So the useful question is not "which cost model does a game choose," it is "how heavily does a system lean on its abilities being managed versus simply used." Every game answers by mixing the cells, and the mix is what gives it its feel.

Three questions decide that mix.

How much should players manage, versus simply use?

This is the master dial, and most systems sit somewhere in the middle of it rather than at either end.

At the light end, abilities cost nothing and are always available. The Apocalypse World family works this way: what a character can do is part of who they are, and the limit is the fiction and the dice, not a pool that empties. Nothing is tracked or hoarded, and a player never weighs whether an ability is worth spending on. The tradeoff is that all the balancing load falls on how the abilities are written, since there is no economy to catch an overpowered one.

At the heavy end, abilities draw on things a character can run out of, and using one is a real decision measured against every other use. This is where the 5e SRD sits for its casters, and it is worth seeing that even a single character there spans the grid: some of their options are just an attack, free to repeat; some are a resource they use up and it is gone until they rest; some are permanent traits that are always on. Free, single-cost, and spent-resource abilities, all on one sheet. The management is the play, and running low is a genuine state with its own tension.

Most games live between those poles, and where a system sits says what it wants a turn to feel like: a set of choices to optimize, or a set of tools to reach for.

Where does the limiter live?

When a system does meter its abilities, the limit usually lives in one of three places, and the interesting part is that games combine them rather than pick one.

- **In a pool.** A store the character draws down: a resource that depletes and refills on rest or over time. It makes power a budget, and gives the designer a clean lever, since an ability's cost can be tuned apart from its effect.
- **On a clock.** So many uses per encounter, per scene, per day. The ability is free in the moment but limited in frequency, so what is really being spent is the time until it returns. Intuitive, and it gives abilities a rhythm, but the length of the period sets the pacing whether the designer intends it to or not, and players learn to game the refresh.
- **In the turn.** The ability costs part of your turn, not a stored resource. The Pathfinder model is the one I find clearest: a turn holds only so much, and a bigger ability simply costs more of it. Everything stays in one currency and every turn becomes a small puzzle, but it only meters things inside a counted mode, so abilities used elsewhere need some other governor.

Real systems stack these. A system might meter an ability in the turn and on a per-day clock at once, so it costs part of this turn and can only be done so often in a day. Cypher leans its whole design on a pool, letting a character spend from their own stats to push an ability harder, so the pool and the effect are tied directly together. The lesson is that the limiter is not a single choice but a set of them, layered until the abilities feel as rationed as the game wants.

How are abilities written?

Cutting across all of the above is a question of expression: is each ability spelled out in full, or assembled from a shared vocabulary?

Detailed abilities read on their own. Everything the ability does is in its text, so a player can understand it without cross-referencing, at the cost of length and the risk that two abilities meant to work alike drift apart in wording.

Keyworded abilities are built from defined terms instead. Lancer leans on this: a compact ability references a shared glossary, and once a player knows the vocabulary they can read any ability at a glance. A term defined once behaves identically everywhere, rules text gets shorter, and a large list stays navigable. The price is the upfront wall: a new player has to learn the vocabulary before the abilities read as anything but jargon, and the designer has to keep the glossary airtight, since one loose keyword misfires across every ability that uses it.

This choice runs independently of cost and limiter. A costless system can be keyworded, and a heavy resource system can be written longhand. It comes down to whether a reader meets the rules all at once or learns a language first.

Running under all three questions is a trade between **freedom and control**. The more a system meters and layers, the more precisely a designer can balance and the more a player must manage;

the less it meters, the faster and more character-driven play becomes, and the more the balancing rests on ability design alone. Where a system lands is not one decision but the sum of these three, and it is worth making each of them consciously rather than inheriting a familiar bundle without noticing.

What Frontiers Does

Now, with that overview to catch you up to speed, here is the approach Frontiers takes, and the reasoning that led there.

Frontiers meters abilities with a single resource, ⚡ **Energy**, on one universal rule:

- Everything usable has a **Demand**: the level of power it operates at.
- Every character has a **Capacity**: the level they sustain freely.
- Using something at or below your Capacity is **free**. Using something above it costs ⚡ **Energy equal to the difference**, every time.

Full rules live in the Rulesets; what follows is why the shape looks like this.

A second action economy

The clearest way to understand ⚡ Energy is as a second action economy, layered over the first. Where a turn's ♦AP meters how much a character can do, ⚡ Energy meters how far past their level they can reach while doing it. It is the same idea as any action economy, a universal cost that gates power, aimed at a different axis.

That framing is the whole design. Energy is an expression of effort: the more a character strains beyond what they have mastered, the more it costs, and the things they have grown into cost nothing at all. It is the game's universal resource, one rule applied to every ability and every item in the game.

Paying for the gap

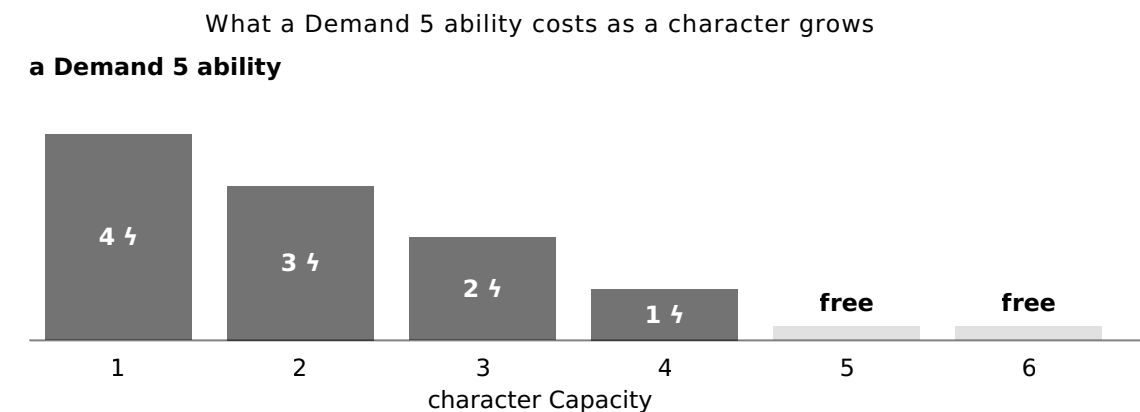
Most systems attach a flat cost to each ability: this spell costs this slot, that power costs that many points, fixed regardless of who is using it. Frontiers costs the distance instead, the gap between what you can sustain and what you are reaching for.

Two things came out of that choice. The first is uniformity. A designer never has to invent a cost for an ability; they set its Demand, and the engine does the rest. Every ability in the game can be explained in about two sentences, because every ability follows the same rule. It looks more complex on the surface, precisely because it is unfamiliar, and it is far simpler underneath, because there is only ever one thing to learn.

The second is that competence becomes something a player feels. In a flat-cost system, a level-fifteen character pays the same for a basic effect as a level-three character did, which never quite

makes sense: mastery should make the basics effortless. Under Demand and Capacity it does. As a character's Capacity rises, the abilities they have grown into fall to free, and the pool is spent only on what still stretches them. Growth reads as things getting easier, on top of a bigger number to spend.

This is the ability migration the grid pointed at earlier. A Frontiers ability does not sit in one cell for a character's whole life. An ability that is compound-cost at low level, taking a turn and a chunk of Energy, becomes free-to-use once Capacity catches up to its Demand. Abilities move across the grid as characters grow, which is unusual, and it is the most distinctive thing the system does.



The cost is the gap. Once Capacity reaches 5, the ability is native and stays free forever.

How gear draws Energy

Items work on the same Demand, with one difference: a wielded item's drain is **held** for as long as you carry it, and returns when you put it down. Carrying it online reserves that much Energy for as long as it is equipped, instead of costing Energy each time it is used.

This is because gear is a constant draw where an ability is a moment of effort. Holding a powerful weapon ready is a sustained strain, so it sits on your Energy the whole time you hold it. The design payoff is control over one of the oldest problems in the hobby: the too-strong item handed to a character who has not earned it. In Frontiers, a powerful item takes up room in you for as long as you carry it.

If its drain is more than your Capacity can hold, you cannot bring it online at all, and even when you can, it costs you the headroom you would otherwise spend on your own abilities. Raw power is throttled by what a character can afford to keep running, which makes gear a real decision instead of a pile of bonuses, and it scales as cleanly as everything else on the same one rule.

Overspend, the agency valve

A character can spend Energy they do not have. Pushing past zero is allowed, and each point below costs a ☉ Wound or ☹ Fatigue, the player's choice.

This is an opinionated inclusion, and it is there on purpose. Frontiers avoids strict walls where a meaningful choice could sit instead, and would rather hand a player a painful option than a locked door.

Overspend is the same move the engine makes with Reactions, which are paid for out of the same ♦AP a character would spend on their turn: an escape valve that trades a resource you do not quite have for agency in a moment that matters. The reasoning behind that pattern is set out in the Flow of Play documentation. Here it means a character is never simply out, they can always reach further, at a cost they can feel.

The pool is small on purpose

A character's ♣ Energy pool is tight, and grows slowly with Level. This is structural rather than stingy. A generous pool has no tension, so nothing about it would matter; a tight one makes every point spent a real decision. And because abilities fall to free as Capacity rises, the same pool stretches further over a character's life without ever needing to be large. What cost Energy at low level is free later, so the pool frees up for whatever still stretches the character, instead of being inflated to keep pace.

Is this the best solution for Abilities?

The mana-and-slot model this reacts against is well understood, and that is part of the problem with it: it is so tied to one kind of magic in one kind of game that few designers extend it past that. Demand and Capacity is an attempt at the general case, the rule a slot system is a special case of, built to be mass-applied to any ability or item a designer can imagine without inventing new machinery each time.

Whether it is best is a matter of taste, but it is clear, uniform, and easy to build on, which is what an engine owes a designer. Everything runs off one universal resource and one universal rule, so nobody meets this system and has to learn fifteen ability types before they can play. They learn one thing, and it applies to everything.

There is a good chance you want a different economy, or none at all. The final section below sets out variant and alternative rules you can use in their place, along with what each one changes and what it will cost you elsewhere.

Variants

Variants keep the philosophy intact. ♣ Energy stays the universal resource, Demand and Capacity still set what things cost, and cost is still the reach past what a character sustains. What changes is the numbers. The groups below are dials, and they are meant to be mixed.

Tune the pool

The base engine assumes a maximum character Level of 10, a widest possible reach of 5, and a pool that grows to 8 at the ceiling.

RESIZE THE POOL

A larger pool makes reaching past yourself routine and raises the overall power level; a smaller one makes every point precious. The default is deliberately tight, so most changes here will be upward, and anything priced against the pool should be rechecked afterward.

CHANGE THE GROWTH

Grow the pool faster for a game where power ramps quickly, slower for a grind, or keep it flat so a character's reach never widens and all their growth comes from Capacity rising instead. Flat growth is an interesting tone: mastery makes old things free, but never lets you strain further than you could at first.

Tune the reach

Cost is the gap between Demand and Capacity, and that relationship is a dial.

CHANGE HOW STEEPLY THE GAP COSTS

The default is one Energy per point of gap. Making it steeper, say two per point past a certain distance, punishes over-reaching and keeps characters close to their level. Shallower makes reaching cheap and lets characters routinely punch above themselves.

CHANGE HOW CAPACITY TRACKS LEVEL

Widen the envelope so abilities stay costly for longer, or narrow it so characters grow into things fast. This sets the pace of the competence-migration: how quickly the abilities a character uses fall to free.

Tune overspend

A character can spend past zero, paying a $\otimes$ Wound or $\ominus$ Fatigue per point.

CHANGE THE PRICE

Charge both a Wound and a Fatigue per point for a harsher push, or cap how far past zero a character may reach so overspend cannot become a full second pool. Either keeps the valve open while raising its stakes.

REMOVE IT

Make zero a hard floor. This suits a stricter, more careful game, and it is a real change in feel: the agency valve closes, and a character who is out is simply out. Worth doing knowingly, since the option to push is one of the more distinctive things the economy offers.

Alternatives

Alternatives replace the philosophy and not tune it. Each changes what ζ Energy fundamentally is, and because Energy also governs gear, several of these reach into Equipment as well. They are listed roughly in order of how far they travel from the default.

Flat costs per ability

Give each ability a fixed price regardless of who uses it, in the manner of the mana and slot systems this design reacts against. A spell costs its points, a power costs its slot, and the number never changes with the user's level.

This is the most familiar option, and its familiarity is the whole appeal: players arrive already knowing how it works. What it costs is the two things Demand and Capacity were built to give. Uniformity goes, since each ability now needs its cost set and balanced by hand rather than following one rule. And the competence-migration goes with it: a master pays the same for a basic effect as a novice, so growth stops making the basics effortless. Choose it when familiarity matters more than either of those.

Prepared loadout

Change when Energy is committed. Instead of spending it in the moment, a character invests Energy into a set of ability cards during downtime; those cards are their available abilities until the next downtime, when everything resets and is re-slotted. Slotting a card reserves its Energy for the whole period, the way held gear does, so a character's live pool during play is whatever they chose not to commit.

This trades a spend-as-you-go rhythm for a prepare-ahead one, closer to loading out before a mission than managing a pool during it. It rewards planning over in-the-moment flexibility, and it makes downtime a real decision point and not only a recovery beat. Frontiers considered a version of this and did not ship it, preferring the fluidity of a live pool, but it is a coherent and appealing model for a game built around preparation, and the Demand and Capacity math carries over intact to decide how much a character can hold.

Uses per period

Drop the pool and meter abilities on a clock instead: so many uses per encounter, per scene, or per rest. Demand can still gate what a character can reach, but frequency, not a spent resource, becomes the limit.

The lightest bookkeeping on this list, and the most intuitive to a new player. What it gives up is the effort-expression the pool provides. Reaching past yourself is no longer a cost you feel accumulating, only a counter that resets, so the sense of a character straining and running low is gone. Suits a lighter game that wants abilities rationed without a resource to track.

Costless abilities

Remove ♣ Energy as an ability cost entirely, and lean on the ♦AP action economy plus careful ability design to do the balancing, in the manner of the costless systems surveyed earlier.

This is the most disruptive option, because Energy is not only an ability resource. Gear is throttled by it too, held drain is what keeps a powerful item from being a free upgrade, so removing Energy from abilities leaves that job unfilled. You would need another way to gate gear, which is a rebuild of the Equipment economy, not a change to this one. That reasoning belongs to the Equipment documentation; the point here is only that this alternative does not stop at abilities. Cut Energy and you have taken on two problems, not one.

Whichever you choose, ♣ Energy is one of the most connected systems in the engine, tied to both abilities and gear at once. A change here is rarely contained to abilities alone, so trace it into Equipment before committing.