

Design Documentation

This is one piece of the **Design Documentation** for Frontiers Beta 2026-2. For a guide to the full Design Documentation set and how to read it, see [Designing With Frontiers](#).

If you want the ruleset only, skip to **Default Rules** using the left-side menu, or read the [Frontiers Overview](#) for the condensed version. If you want the reasoning behind those rules, read top to bottom.

This section is the teaching part of the document. It exists to explain how tabletop games handle the special things characters can do beyond their baseline action economy. None of this is Frontiers-specific. If you already understand how Abilities, Talents, Powers, Feats, or similar systems work, you can skip ahead to [Part Two](#).

What "Special Abilities" Actually Are

Every TTRPG character can do baseline things. Move. Attack. Make a check. These are the actions any character can take just by existing in the system.

But characters need to be more than baseline. A wizard's fireball, a fighter's parry, a rogue's sneak attack, a face's silver-tongued reframe of a hostile situation. These are not things every character can do. They're learned, granted, or earned, and they make characters feel distinct from one another.

The mechanical layer that handles these things has many names:

- Spells (D&D-likes, Pathfinder)
- Powers (Mutants & Masterminds)
- Talents (Free League's Year Zero Engine)
- Feats (Pathfinder)
- Moves (Powered by the Apocalypse)
- Special Abilities (Blades in the Dark)
- Stunts (Fate)
- Edges (Savage Worlds)
- Tricks (Mothership)

Whatever they're called, they all answer the same question: **what can this character do that another character can't?**

This is the mechanical heart of character identity. Two characters with identical Attributes can play wildly differently if their Abilities differ. Abilities are how systems express "this is what your character is for."

How Different Games Handle Powers

The diversity here is enormous. A few examples to make the design space concrete:

Pathfinder (Paizo, 2009/2019) uses a deeply layered Feat system. Hundreds of Feats are available, sorted by prerequisites and tiers. A character picks Feats at specific Level milestones. Each Feat is a small, specific mechanical advantage. Character build is a long, deliberate sequence of Feat choices that shape the character's mechanical identity over time.

Powered by the Apocalypse (Vincent Baker, 2010) uses Moves: 5-7 character-specific actions, each tightly coupled to a roll-and-result table. A character has a small set of Moves and the Moves are the character's mechanical identity. A Battlebabe and a Brainer aren't just thematically different. Their Move lists are entirely separate.

Blades in the Dark (One Seven Design, 2017) uses Special Abilities: each Crew role has a list of 8-10 named Abilities, and characters pick a few. They're powerful, defining, and named after evocative concepts (Ghost Veil, Mule, Daredevil). Each Ability does something specific and dramatic.

Mothership (Tuesday Knight Games, 2018) uses Tricks and class-specific abilities sparingly. Characters have one or two class features and otherwise rely on raw stats and equipment. Powers are scarce and each one matters.

Savage Worlds (Pinnacle, 2003) uses Edges: small modular advantages picked during character creation and at advancement. Edges are atomic. Each one does one specific thing. But characters accumulate many over a campaign, building a tailored loadout.

Fate (Evil Hat, 2003) uses Stunts: small, character-defined advantages built around the system's Aspect framework. Stunts give bonuses to specific Skills under specific conditions. They're flexible and player-authored within structural rules.

Lasers & Feelings (John Harper, 2013) has no Powers layer at all. Each character's "thing" is a single descriptive line. Mechanical specialization doesn't exist beyond the one number on the character sheet.

The point is not that any of these is wrong. They are answering the same design question with different priorities. **Some games make Powers the center of character identity (PbtA, Blades). Some make them modular accessories (Pathfinder, Savage Worlds). Some treat them as scarce signature moments (Mothership). Some skip them entirely (Lasers & Feelings).**

The choice cascades into character creation, progression, encounter design, and balance. Get it right and characters feel distinct and interesting. Get it wrong and the system either feels samey or like a stack of bonuses without identity.

The Three Questions Every Powers System Answers

Underneath all the implementations, every Powers system makes three choices:

1. Where do Powers come from?

Some systems tie Powers to **class** (Pathfinder, PbtA Playbooks). Each class has its own list, and your class is your access point.

Some tie Powers to **species or origin** (D&D 5e racial features, Mutants & Masterminds origins). Powers express what the character is fundamentally.

Some tie Powers to **training or background** (Skills-as-powers in Cypher System). Powers express what the character learned.

Some tie Powers to **items** (artifacts, magic weapons, technological devices). Powers come from what the character has.

Some allow **free selection** (Savage Worlds Edges, Fate Stunts). Players pick Powers à la carte from a menu, regardless of class or origin.

Most systems combine sources. The choice of which sources are dominant tells the player what kind of stories the system is telling.

2. What gates their use?

Powers need limits, or they become baseline actions. The gates take many forms:

- **Resource cost.** Spend mana, spell slots, stress, hero points, action points to use the Power. Common in tactical systems.
- **Frequency limit.** "Once per encounter," "once per session," "once per long rest." Common in heroic systems.
- **Risk.** Use carries the chance of consequences (PbtA's mixed results, Blades' position/effect).
- **Tier requirement.** Higher Powers require higher character development.
- **Nothing.** Some Powers are passive and always-on. Others are at-will but functionally limited by the situation.

Most systems combine multiple gates. A spell might cost a slot AND require a roll AND have a tier requirement.

The choice of gating shapes how often the Power activates, how dramatic its activation feels, and how much it dominates a character's identity.

3. How do they interact with the resolution system?

Three broad patterns:

- **Replace the roll.** The Power lets the character do something automatically, no resolution required. (PbtA "advances," some passive abilities.)
- **Modify the roll.** The Power changes the math of an existing roll (bonus, reroll, advantage, change the die used).
- **Trigger an effect outside the roll.** The Power adds a consequence or capability that doesn't depend on dice (positioning shifts, narrative leverage, conditions applied).

Most systems use all three patterns across their Powers list, but the dominant pattern reveals the system's relationship to dice. A system where most Powers replace rolls feels narrative. A system where most Powers modify rolls feels tactical. A system where most Powers trigger effects feels strategic.

The Point

There is no correct answer to any of these questions. Each combination produces a different feel of play.

But there is a correct coherence: a Powers system should know what it's for. Powers that do too many different things, gate themselves too many different ways, or come from too many different sources end up feeling like an unstructured grab bag. Powers with a clear identity, whether scarce signature moments, modular bonuses, or tactical tools, feel intentional.

A designer's most consequential decision in this layer is **what role Powers play in the character's overall identity**. Are Powers the character, or do Powers adorn the character? Both are valid answers, but they're different design philosophies and they require different system support.

This section covers the specific choices Frontiers made and the reasoning behind them. The earlier section explained the field. This section explains the answer.

The Design Question

When designing Frontiers' Abilities, the goal was a Powers layer that:

- Could come from any source (Archetype, Origin, Background, Equipment, Vector Marks) without each source needing its own subsystem
- Had a single universal anatomy designers could write to consistently
- Used a resource gate (≠Energy) that felt tactical without being punishing
- Supported both tightly defined Abilities and loosely scaling ones

- Worked across PCs, NPCs, and item-driven activations with the same rules

Most existing systems either tie Powers tightly to one source (PbtA Moves are Playbook-locked) or use entirely separate subsystems for each Power source (D&D's spells, feats, racial traits, and item abilities each have their own rules). Frontiers wanted **one Ability framework for everything**.

That goal narrowed the design space:

Goal	Reasoning	Constraint
Universal anatomy	Designers writing Abilities should follow one template	Argues against multiple Power subsystems
Source-agnostic	An Ability from a sword should work like an Ability from an Archetype	Argues for a unified resource model
Energy as the gate	Resource pressure should connect to effort	Argues against per-day or per-rest limits as default
Level comparison instead of absolute Level	High-Level Abilities should be reachable at low Level if the player accepts the cost	Argues for a relative Level rule
Static AND dynamic Abilities	Some Abilities should stay fixed; some should scale with the character	Argues for both Ability Level types

The Tradeoffs Considered

Frontiers studied several Powers architectures:

Spell-slot systems (D&D, Pathfinder). Predictable resource management. But slots create per-day pacing pressure that doesn't fit Frontiers' Encounter-focused structure.

Move systems (Powered by the Apocalypse). Strong character identity. But Moves are tightly coupled to specific roll outcomes, which would force Frontiers to redesign its resolution layer.

Stress-driven systems (Blades in the Dark). Powers cost a meta-resource. But Blades' Stress is a single track that fuels everything; Frontiers wanted Energy specifically tied to effort beyond baseline, not all dramatic moments.

Free-selection systems (Savage Worlds). Maximum flexibility. But without structural gates, Powers can't scale by character growth meaningfully.

The answer Frontiers landed on: **one Ability anatomy, one resource gate (≠Energy), one Level comparison rule**.

Frontiers' Answer — One Anatomy, One Gate

Every Ability in Frontiers follows the same structural template, regardless of where it came from. An Ability granted by an Archetype, an Origin, an Item, or an NPC stat block all look the same on paper and follow the same rules at the table.

The two systems that make this work are **Ability Level** and **Energy cost**, calculated against the user's **Reference Level** at the moment of use.

Ability Level

Every Ability has a **Level** that represents its potency, complexity, and potential impact. Levels are not just labels. They're the mechanical hook that determines whether using the Ability costs Energy.

Abilities come in two types:

Static Level Abilities. The Ability's Level is fixed at design time. A Level 3 Parry stays Level 3 forever, no matter who uses it. As the user's Reference Level grows, this Ability becomes free to use at Reference Level 3 and remains free above that.

Dynamic Level Abilities. The Ability's Level scales with the user's Reference Level. A Dragon-Origin breath weapon written as "Ability Level = Reference Level + 1" always costs 1 Energy to use, regardless of when it's used. The Ability stays "one step ahead" of the user permanently.

Most Abilities are static. Dynamic Abilities are a tool for designers who want a feature to feel evergreen for that character. Origin Abilities and signature features are particularly good fits for Dynamic scaling.

Reference Level

The **Reference Level** is the comparison value used to determine an Ability's Energy cost for a particular user. It works differently for different actors:

- **Player Characters:** Reference Level = total Character Level (sum of all Archetype Levels).
- **Non-Player Characters:** Reference Level = the NPC's defined Level (set by the GM or system designer).
- **Items activating Abilities:** Reference Level = the Item's Level.

This unification is important. The same Ability with the same Level rule can be used by a PC, an NPC, or an Item, and the math always works the same way: compare the Ability's Level to the actor's Reference Level, and pay Energy for the difference.

The term "Reference Level" was chosen because it's neutral. It doesn't imply where the Ability came from (the source of the Ability is irrelevant once you're using it). It just names the Level you measure against.

The Energy Cost Rule

The rule is one sentence:

If an Ability's Level exceeds the user's Reference Level, the user pays ↗Energy equal to the difference.

If Ability Level ≤ Reference Level: no Energy cost. If Ability Level > Reference Level: Energy cost = Ability Level – Reference Level.

A Reference Level 3 character using a Level 5 Ability pays 2 Energy. The same character using a Level 10 Ability pays 7 Energy (which they probably can't afford and would have to Overspend for; see the Flow of Play design doc for Overspend rules).

The rule is **linear** by design. Each Level above costs 1 Energy. The math stays trivial regardless of how big the gap is.

Why Reference Level Beats Absolute Level Requirements

Most TTRPG Powers systems gate Powers by absolute Level: "you can't use a Level 9 spell until you're Level 17." Frontiers does it differently. Any character can attempt any Ability, but the cost scales with how far above their Reference Level the Ability is.

This buys two things:

Flexibility for high-stakes moments. A Level 3 character desperate to save the day can use a Level 8 Ability. They just have to pay 5 Energy (or Overspend, gaining Wounds or Fatigue). The engine doesn't say "no." It says "yes, but expensive."

Cost-driven scaling. As a character grows in Reference Level, their old Abilities become free, and they unlock cheap access to mid-tier Abilities. They never lose access to anything; they just gain more efficient access to bigger things over time.

This makes Energy a tactical resource that maps to ambition. Players can always reach for the big move. The question is whether they can afford it.

Frontiers' Answer — Universal Anatomy

Every Ability defines the same nine fields. Some may be omitted if not relevant, but the framework is consistent across every Ability in the engine.

Field	What It Specifies
Name	The Ability's identifier
Level	Static value or dynamic formula (e.g., "Reference Level + 1")
Action Cost	↗AP, ↗Free Action, Reaction, or Passive

Field	What It Specifies
Range	Self, Melee, Short, Medium, Long, or specific □Units
Target	What the Ability affects (one creature, area, self, etc.)
Resolution	Whether a roll is required, what Attribute, what target value
Effect	What happens when the Ability resolves
Duration	If applicable, how long the effect lasts
Frequency	If limited, how often it can be used

This consistency lets designers and players parse any Ability quickly. A new Ability is never confusing structurally. Only its specific Effect needs interpretation.

The Ability Level should always be visible and clear, marked with whether it's Static or Dynamic. Players need this information to calculate Energy cost at a glance.

Why Action Cost Is Part of the Anatomy

Most Abilities cost ♦AP (1 by default). But Frontiers explicitly allows Abilities to cost ♦Free Actions, to function as Reactions, or to be entirely Passive (no Action cost, always-on).

This was a deliberate choice. By making Action Cost a field rather than a default, Frontiers signals to designers that they can choose between high-frequency cheap Abilities (Free Actions, Passives) and high-impact expensive ones (multi-AP). The Action economy and the Ability system inform each other.

It also makes Reactions a normal Ability shape, not a special category. A Reaction is just an Ability whose Action Cost says "Reaction." The rest of the anatomy works identically. (See the Flow of Play design doc for how Reactions interact with the AP pool.)

Why No Default Frequency Limit

By default, an Ability with no listed Frequency can be used as often as ♦AP and ∕Energy allow. Designers add Frequency limits ("once per Encounter," "once per Scene") only when they want to gate something further than Energy already does.

This was deliberate. Energy is doing the heavy lifting of resource gating. Adding mandatory Frequency limits would create a second layer of restriction that overlaps with Energy and forces designers to think about both. Keeping Frequency optional means designers add it when they specifically want pacing pressure beyond Energy.

Powerful Abilities that should only fire once per fight get a Frequency limit. Routine Abilities don't need one.

Why Static AND Dynamic Levels

The two Level types serve different design purposes:

Static Levels are for tightly tuned Abilities. A Parry that grants +2 Defense at Level 3 makes sense at Level 3 and might not at Level 8. Static Level lets the designer pin the Ability to its intended power band.

Dynamic Levels are for evergreen identity Abilities. A Dragon's breath should always feel impressive, regardless of whether the dragon is young or ancient. Dynamic Level keeps the Ability relevant by always staying just above (or at) the user's Reference Level.

Dynamic Abilities should generally include scaling effects (damage that grows with Ability Level, range that increases, duration that extends) to justify the Energy cost they always carry. A Dynamic Ability that's "Reference Level + 1" but does a flat 4 damage forever is a bad design; the player pays 1 Energy per use for an Ability that doesn't grow with them.

Designers writing Dynamic Abilities should write them as "always doing more for the cost they carry."

Everything below this point is the actual mechanics. The earlier sections explained the theory. This section explains how it runs at the table.

What an Ability Is

An **Ability** is a discrete mechanical feature that lets a character do something beyond baseline action.

Abilities can: - Produce an effect beyond what raw Attributes alone could - Modify existing mechanics - Introduce tactical, defensive, or narrative leverage - Express character identity beyond Attributes and Skills

Abilities are explicitly defined, mechanically bounded, and limited primarily by ♦AP, ⚡Energy, and (when applicable) Frequency.

Ability Anatomy

Every Ability defines nine fields:

1. **Name** — what the Ability is called
2. **Level** — the Ability's Level, marked as Static or Dynamic
3. **Action Cost** — ♦AP, ♦Free Action, Reaction, or Passive

4. **Range** — Self, Melee, Short, Medium, Long, or specific □Units
5. **Target** — what the Ability affects
6. **Resolution** — whether a roll is needed, and what kind
7. **Effect** — what happens
8. **Duration** — how long the effect lasts (if applicable)
9. **Frequency** — usage limits (if applicable)

Fields not relevant to a specific Ability may be omitted, but the framework is universal.

Ability Level

Every Ability has a Level. Levels are not skill checks or experience requirements. They're the value used to compare against the user's Reference Level when calculating ≠Energy cost.

Ability Levels come in two types:

Static Level Abilities

The Ability's Level is fixed at design. The Level does not change based on who uses the Ability or when.

A Level 3 Parry costs 3 Energy when used by a Reference Level 0 character, 2 Energy at Reference Level 1, 1 Energy at Reference Level 2, and 0 Energy at Reference Level 3 or higher.

Static is the default Ability type and should be used unless a Dynamic Level serves the design better.

Dynamic Level Abilities

The Ability's Level is calculated at the moment of use, based on the user's Reference Level.

A Dynamic Ability's Level is written as a formula relative to the user. Common patterns:

- **Reference Level + 1** — always 1 above the user (always costs 1 Energy)
- **Reference Level + 2** — always 2 above (always costs 2 Energy)
- **Reference Level + 0** — always equal (always free)
- **Reference Level – 1** — always 1 below (always free, never any Energy cost)

Dynamic Level Abilities should generally include scaling Effects (damage, range, duration) that grow with the Ability's current Level. This justifies the consistent Energy cost.

When writing an Ability, designers should always mark whether the Level is Static or Dynamic, and Dynamic Abilities should specify the formula clearly.

Maximum Ability Level

The default engine assumes a soft cap of **Ability Level 15**. This corresponds to the soft cap of Character Level 10 plus 5 Levels of "stretch" for Abilities that exceed any reasonable character's Reference Level.

Systems built on Frontiers may extend the cap, but should also extend Energy and Reference Level scaling proportionally. An engine that supports Level 30 characters needs Level 35+ Abilities and Energy pools that can absorb the costs.

Reference Level

The **Reference Level** is the value compared to an Ability's Level when calculating ⚡Energy cost.

How Reference Level is determined:

Actor	Reference Level
Player Character	Total Character Level (sum of all Archetype Levels)
Non-Player Character	Defined Level set by GM or system designer
Item activating an Ability	The Item's Level

The same Ability with the same Level rule applies the same math regardless of who's using it.

⚡ Energy Cost

The Energy cost of using an Ability is calculated at the moment of use:

If Ability Level > Reference Level, pay Energy equal to the difference. Otherwise, no Energy cost.

Examples: - Reference Level 5 character using a Static Level 3 Ability: 0 Energy - Reference Level 5 character using a Static Level 5 Ability: 0 Energy - Reference Level 5 character using a Static Level 8 Ability: 3 Energy - Reference Level 5 character using a Dynamic Level (Reference Level + 1) Ability: 1 Energy

If the user does not have enough Energy to pay the cost, they may **Overspend** (see the Flow of Play design doc for Overspend rules).

If the user is already at Negative Energy and using the Ability would exceed the negative limit, the Ability cannot be used.

Action Cost

Every Ability specifies its Action Cost. The four possible types:

Standard ♦AP Cost

The Ability requires 1 or more ♦AP, spent on the user's Turn during Encounter Play.

Most Abilities cost 1 ♦AP. Some powerful Abilities cost 2 or 3 ♦AP (consuming most or all of a Turn's economy).

If an Ability requires multiple ♦AP, all required AP must be available at declaration. The cost is paid before resolution.

✧ Free Action

The Ability does not cost ♦AP and may be used once per Round per Ability. Multiple different Free Action Abilities may be used in the same Round, but the same Free Action Ability cannot be used twice in one Round (unless a rule explicitly allows it).

Free Action Abilities are typically minor effects: small bonuses, quick repositioning aids, brief environmental manipulations.

Reaction

The Ability triggers off an external event during another character's Turn. Reactions cost ♦AP from the user's current pool (the same pool used on their own Turn).

A user with no available ♦AP cannot use a Reaction. A user who uses a Reaction has fewer ♦AP available on their next Turn.

(See the Flow of Play design doc for the full Reaction-AP interaction rules.)

Passive

The Ability has no Action Cost and is always active. Passive Abilities do not need to be declared and apply continuously.

Passive Abilities still have a Level. If the user's Reference Level is below the Passive Ability's Level, they continuously pay the difference in Energy as long as the Ability is "on." This makes high-Level Passive Abilities expensive to maintain at low Reference Level.

Most Passive Abilities are Static Level 0 or low. They're meant to be free at any reasonable Reference Level. Designers writing high-Level Passive Abilities should consider whether the design intent is to pressure the user's Energy pool continuously.

Resolution

An Ability's Resolution field specifies whether a roll is required and what kind:

- **No roll required.** The Ability succeeds automatically. The Effect resolves as written.
- **Resolution Roll required.** The Ability requires a standard Resolution Roll. The Ability specifies the governing Attribute and the target (DR or Defense).
- **Defense check.** The Ability declares a value compared against a target's Defense. If the target's Defense exceeds the value, the Effect is reduced or negated as specified.

Some Abilities use Gradient (see the Resolution System design doc for Gradient rules). When an Ability calls for Gradient, follow standard Gradient interpretation unless the Ability specifies otherwise.

Range and Target

Range specifies how far the Ability can reach. Common categories:

Range	□Units
Self	The user only
Melee	1 □Unit
Short	2-5 □Units
Medium	6-10 □Units
Long	11+ □Units

Systems may use specific □Unit values instead of category names.

Target specifies what the Ability affects: one creature, multiple creatures, an area, a point in space, an object, the user themselves, etc.

Effect

The Effect is the Ability's actual mechanical and narrative result. Damage dealt, conditions imposed, positions changed, allies aided, environment altered.

Effects should be specific. Vague effects ("the target is hindered") create adjudication friction. Clear effects ("the target loses 1 ♦AP on their next Turn") resolve cleanly.

Designers writing Abilities should aim for one or two clear consequences per Effect, not a long list of conditional outcomes.

Duration

If an Ability creates an ongoing effect, the Duration specifies how long it lasts:

- **Instantaneous** — resolves immediately
- **Until end of Turn** — expires when the user's Turn ends
- **Until end of Round** — expires when the current Round ends
- **For X Rounds** — expires after the specified number of Rounds
- **Sustained** — lasts as long as the user maintains it (often costing AP per Round to maintain)
- **Until dispelled** — lasts until specifically removed

Abilities without a stated Duration are Instantaneous.

Frequency

If an Ability has a usage limit, the Frequency field specifies it:

- **Once per Turn** — can only be used once during the user's Turn
- **Once per Round** — can only be used once between the user's Turns
- **Once per Encounter** — can only be used once per Encounter
- **Once per Scene** — can only be used once per narrative scene
- **Once per Downtime** — can only be used once between Downtimes

If no Frequency is listed, the Ability can be used as often as ♦AP and ♠Energy allow.

Frequency limits exist for Abilities that should be gated beyond what Energy alone provides, usually because the Ability is too powerful to fire repeatedly even at full Energy expense.

Example Abilities

The following examples show the universal anatomy in practice, covering the major Ability types: Static, Dynamic, Active, Passive, Reaction, and Free Action.

Iron Stance (Static Level, Active)

Name: Iron Stance

Level: Static Level 2

Action Cost: 1 ♦AP

Range: Self

Target: Self

Resolution: No roll required

Effect: Until the start of your next Turn, gain +2 to □Physical Defense. You cannot use ⇒Movement Points while in this stance.

Duration: Until start of your next Turn

Frequency: None

A defensive maneuver. Reference Level 2 characters and above use this for free.

Cinder Breath (Dynamic Level, Active)

Name: Cinder Breath

Level: Dynamic, Reference Level + 1

Action Cost: 2 ♦AP

Range: Short

Target: Cone, 3 □Units

Resolution: Targets roll ♦Vigor vs DR (8 + Reference Level)

Effect: On failure, target takes (Reference Level) damage and gains the Burning condition for 2 Rounds. On success, half damage, no condition.

Duration: Damage instantaneous; Burning persists 2 Rounds

Frequency: Once per Encounter

A dragon-like signature attack. Always costs 1 ⚡Energy regardless of Reference Level. Damage and DR scale with the user.

Counterstrike (Static Level, Reaction)

Name: Counterstrike

Level: Static Level 4

Action Cost: Reaction (1 ♦AP from current pool)

Range: Melee

Target: Triggering attacker

Resolution: Roll □Finesse vs attacker's □Physical Defense

Effect: Triggered when an enemy in Melee range attacks you. On a successful roll, deal weapon damage to the attacker before their attack resolves.

Duration: Instantaneous

Frequency: Once per Round

A reactive fighter Ability. Costs 1 ♦AP from the user's current pool. The user must have AP available. Having spent all 3 AP on their Turn means no Counterstrike available until their next Turn refresh.

Battlefield Awareness (Static Level, Passive)

Name: Battlefield Awareness

Level: Static Level 1

Action Cost: Passive

Range: Self

Target: Self

Resolution: No roll required

Effect: You cannot be flanked. Enemies do not gain bonuses from attacking you from positions other than directly in front of you.

Duration: Always active

Frequency: N/A

A baseline defensive Passive. Free at Reference Level 1+. Costs 1 Energy continuously to maintain at Reference Level 0, which is unusual but not impossible for a freshly created character.

Quick Step (Static Level, Free Action)

Name: Quick Step

Level: Static Level 2

Action Cost: ♦Free Action

Range: Self

Target: Self

Resolution: No roll required

Effect: Move 1 □Unit without spending ⇒MP. This movement does not provoke Reactions that would normally trigger from movement.

Duration: Instantaneous

Frequency: Once per Round

A small mobility tool. Free Action, so it doesn't compete with ♦AP. Once per Round per Ability rule applies.

Tuning the Feel

The default rules give Frontiers' baseline feel. The knobs below let designers tune that feel without breaking the engine.

Ability Density

How many Abilities a character has at each Level shapes whether the system feels Power-heavy or Power-light.

Density	Effect	Good for
Low (1-2 Abilities at Level 1, +1 every 2 Levels)	Each Ability matters individually, slow build pacing	Lethal or low-magic systems
Default (2-3 Abilities at Level 1, ~1 per Level)	Balanced. Characters always have meaningful options	Most Frontiers-based systems
High (4-5 Abilities at Level 1, +2 per Level)	Rich tactical options every Turn, longer character creation	Heroic or high-magic systems

Energy Cost Sensitivity

How tight ✂Energy budgets are shapes how often above-Reference-Level Abilities get used.

- **Generous Energy.** Above-Level Abilities are common moves, not exceptional ones. Powers feel abundant.
- **Default Energy.** Above-Level Abilities are tactical investments. Players use them in important moments.
- **Scarce Energy.** Above-Level Abilities are rare and dramatic. Powers feel precious.

Static-Dynamic Mix

How many Abilities are Dynamic vs Static shapes whether the system feels evergreen or tier-locked.

- **Mostly Static.** Characters outgrow early Abilities and unlock new ones at higher Levels. Strong sense of progression. Old Abilities can feel obsolete.
- **Mostly Dynamic.** Characters' starting Abilities scale with them and stay relevant. Less sense of progression but more identity continuity.
- **Mixed (default).** Some signature Abilities are Dynamic; most progression Abilities are Static. Old favorites stay relevant; new options expand the kit.

Frequency Limit Frequency

How often Frequency limits appear shapes whether Abilities feel always-available or pacing-controlled.

- **Few Frequency limits.** Most Abilities can be spammed (gated only by Energy). Characters always have their full kit available.
- **Default.** Powerful Abilities have Frequency limits; routine Abilities don't. Pacing pressure on big moments only.

- **Many Frequency limits.** Most Abilities are gated. Players manage uses carefully across Encounters.
-

Edge Cases

The following rulings apply unless a specific rule explicitly overrides them.

Ability Level vs Reference Level Timing. The Energy cost is determined at the moment of declaration, comparing the Ability's Level to the user's Reference Level at that moment.

If the user's Reference Level changes mid-Encounter (rare), already-declared Abilities are unaffected. Future Ability use applies the new Reference Level immediately.

Energy Payment Timing. Energy is paid immediately upon declaration. If insufficient Energy is available, Overspend rules apply.

If the Ability is interrupted after declaration, Energy is not refunded unless explicitly stated by the Ability or another rule.

If the Ability is canceled before declaration completes, no Energy is paid.

Ability Interruption. If an Ability is interrupted after declaration: - ♦AP cost remains spent. - ⚡Energy cost remains spent. - Effects that have already resolved are not undone. - Unresolved portions do not occur.

If interruption occurs before any resolution begins, the Ability fails without effect (but resources are still spent).

Multiple Abilities in a Turn. A user may use multiple Abilities in a Turn if: - They have sufficient ♦AP. - No Frequency limits are violated. - No rule explicitly prohibits it.

There is no inherent limit on the number of Abilities used per Turn beyond Action economy and Frequency.

Passive Abilities and Energy. Passive Abilities do not require declaration but still apply Energy cost continuously if the user's Reference Level is below the Ability's Level.

Continuous Energy drain happens at the start of each of the user's Turns during Encounter Play. Outside Encounter Play, Energy drain is tracked at GM discretion (typically per hour or per scene).

Stacking Abilities. Unless an Ability explicitly states it stacks: - Identical named effects do not stack. - Different named effects stack normally.

If two Abilities modify the same statistic for the same Duration, both apply unless one explicitly replaces the other.

Ability Duration Overlap. If the same Ability is activated again before its previous Duration expires, the new Duration replaces the old Duration unless the Ability explicitly states Durations stack.

Ability and Skill Interaction. If an Ability requires a Resolution Roll, Skill Benefits apply normally to that roll.

If an Ability modifies a roll (rather than requiring one), Skill Benefits apply to the underlying roll before the Ability's modification, unless explicitly reversed.

Ability and Gradient. If an Ability requires a roll and does not specify Gradient behavior, Gradient applies normally if deviation meaningfully affects the outcome.

If an Ability modifies damage or effect magnitude, Gradient modifies the final result unless explicitly stated otherwise.

Simultaneous Ability Activation. If two Abilities trigger simultaneously (both as Reactions to the same event, for example), resolve per Action Timing Priority rules. Reaction chains follow reverse chronological order.

Suppressed or Lost Abilities. If an effect suppresses Abilities: - Passive Abilities are disabled. - Active Abilities cannot be declared. - Ongoing Ability effects remain unless the suppression effect explicitly ends them.

If an Ability is permanently lost (narrative or mechanical), ongoing effects immediately end unless specified otherwise.

Ability Without Explicit Frequency. If an Ability lists no Frequency, it may be used as often as ♦AP and ♠Energy allow. Narrative plausibility may still constrain repeated use at GM discretion.

Using an Ability at Negative ♠Energy. A user can use an Ability while at 0 or Negative Energy by Overspending, as long as the negative limit would not be exceeded.

Each instance of Overspend triggers its own consequences (gain 1 ☹Wound or 1 ☹Fatigue, player's choice). See the Flow of Play design doc for full Overspend rules.

Dynamic Ability Level Calculation. A Dynamic Ability's Level is calculated at the moment of declaration. If the user's Reference Level changes (rare, but possible from temporary effects), already-declared Abilities use the Level value at declaration. Future declarations recalculate normally.

Variants

Variants modify how Abilities behave while preserving the core structure: Abilities have Levels, ↗Energy gates above-Level use, ↗AP gates active use.

Energy Surcharge Variant

What changes: Above-Reference-Level Energy cost is no longer linear.

Instead of paying the full Level difference, the user pays: - 1 ↗Energy when using an Ability up to 3 Levels above - 2 ↗Energy when using an Ability 4-6 Levels above - 3 ↗Energy when using an Ability 7+ Levels above

What stays: Reference Level, Ability Level, the comparison rule.

What shifts: Powers feel more available at low Reference Level. High-Level Abilities are accessible without massive Energy investment.

Use when: Building a heroic or high-fantasy system where players should regularly reach above their Level.

Strict Level Lock Variant

What changes: Users may not use Abilities above their Reference Level under any circumstances.

↗Energy cannot bypass this restriction.

What stays: Same-Level and below Abilities work normally.

What shifts: Removes Overspend tension from Ability use. Makes Level progression more rigid and gated.

Use when: Building a tightly balanced or competitive system where Level becomes a hard wall.

Auto-Scaling Abilities Variant

What changes: All Abilities have implicit scaling tied to Reference Level. Damage values, ranges, durations, and area sizes auto-scale as the user grows.

Designers don't write explicit scaling into each Ability. The engine applies a uniform scaling formula across the board.

What stays: Static and Dynamic Level distinction; Energy cost rules.

What shifts: Dramatically simplifies Ability design. Reduces designer workload but limits per-Ability tuning.

Use when: Building a system where Ability balance should be uniform and predictable.

Frequency-Refresh Variant

What changes: Replace Frequency limits like "once per Encounter" with refresh-based mechanics:
- Refresh on Short Rest (between Encounters) - Refresh on Long Rest (after Downtime)

What stays: Energy interaction, Action costs.

What shifts: Creates a clearer pacing structure. Tracks "spent Abilities" instead of counting uses within a window.

Use when: Building a system with explicit rest beats.

Passive-Heavy Variant

What changes: Most Abilities are Passive or triggered, with fewer Active activations. Active Abilities are reserved for major signature moments.

What stays: Universal anatomy, Energy rules.

What shifts: Reduces Turn complexity. Speeds Encounter flow. Shifts character identity toward build design over moment-to-moment activation.

Use when: Building a system where Action economy should focus on movement and basic Actions, with Powers operating in the background.

Alternatives

Alternatives substantially restructure how Abilities function. Adopting these requires recalibration of pacing, Energy, and Action economy.

No Energy Alternative

Core change: Remove $\nearrow$ Energy interaction from Abilities entirely. Abilities are limited only by $\blacklozenge$ AP costs, Frequency limits, and cooldowns.

Above-Reference-Level Abilities are either prohibited or allowed at no extra cost.

What you gain: - Removes endurance tension from Powers. - Increases reliability. Players always know what they can do. - Shifts pacing pressure to positioning and Frequency.

What you lose: - The "ambition vs. cost" tactical layer Energy provides. - One of the engine's defining design choices.

Use when: Building a fast-paced, low-resource system.

Cooldown-Based Alternative

Core change: Replace Energy with cooldown timers. Each Ability defines a Cooldown duration (in Rounds, Scenes, etc.).

Using an Ability above Reference Level increases its Cooldown instead of costing Energy.

What you gain: - Predictable pacing. - Removes Energy volatility. - Encourages planning over resource gambling.

What you lose: - The flexibility of being able to spend Energy how the user chooses. - Energy as a unifying resource across Abilities and Equipment.

Use when: Building a system where Cooldowns fit the tone better than resource pressure.

Talent Slot Alternative

Core change: Replace Level gating with "Talent Slots." Characters equip a number of Abilities equal to their Level (or some derived value).

Abilities outside equipped slots cannot be used. Energy is removed or made optional.

What you gain: - Encourages between-Encounter build customization. - Emphasizes preparation over improvisation.

What you lose: - Spontaneous reach for high-impact Abilities in dramatic moments. - The tactical layer of Energy management.

Use when: Building a system with strong build-customization phases.

Narrative Authority Alternative

Core change: Reduce mechanical Abilities. Replace high-Level Abilities with narrative authority mechanics: spend a narrative token to alter the scene, declare environmental advantage, reframe consequences.

Energy may be replaced with narrative currency.

What you gain: - Strong narrative orientation. - Reduces mechanical complexity.

What you lose: - Tactical depth in Encounters. - Clear mechanical progression of Powers.

Use when: Building a story-focused system where Abilities should feel narrative-driven.

A Note to System Designers

Abilities are where character identity lives mechanically. Skills give Favor; Attributes give Modifiers; Abilities give characters their signature.

A system that adopts Frontiers' defaults inherits a particular feel: - One universal Ability anatomy across all sources - $\nearrow$ Energy gates above-Level use through linear scaling - Reference Level unifies the math across PCs, NPCs, and Items - Static Levels for tightly tuned Abilities, Dynamic Levels for evergreen identity - Frequency limits are optional, used only when Energy alone is insufficient

This feel is deliberate and tuned.

Modifying the defaults with a Variant or an Alternative changes that feel. Removing Energy turns Abilities into tactical menus. Replacing Levels with Slots turns them into loadouts. Each direction is valid; each produces a different system at the table.

Like everything in Frontiers, you are open as a designer to massively modify what Frontiers offers in favor of a method that works best for your system. This document is just giving you the tools and concepts Frontiers considered when deciding its own default engine rules.

In addition, it's important to note that ability and cost systems can be layered on top of each other. If a designer wants a spell-slot like system in their game, they can choose to add it on top of what already exists in the $\nearrow$ Energy system by merging it, or by making it a separate resource-intensive system in it's own right. $\nearrow$ Energy was designed to be adaptable to whatever needs a designer has, not to be constrictive.

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

For the condensed working ruleset, see the **Frontiers Overview**.