

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 organize the numbers on a character sheet, why those numbers exist, and what choices designers face when picking them. None of this is Frontiers-specific. If you already understand attribute and derivative design thoroughly, you can skip ahead to Part Two.

What Character Numbers Do

Every tabletop character is, mechanically, a small bundle of numbers. Some games go bigger (Pathfinder 2e characters have dozens of trackable values). Some go smaller (Lasers & Feelings characters have one number). Most sit somewhere in between.

The numbers exist to do three jobs:

- **Define what a character is good at.** A high number in one area, a low number in another, signals identity.
- **Drive resolution.** Numbers feed into rolls, into Defenses, into thresholds. They turn the abstract "this character is strong" into a mechanical advantage at the table.
- **Track state changes.** Some numbers go up and down during play. They're the game's shorthand for "what is happening to this character right now."

These three jobs are usually distributed across two layers: **stats that describe the character** and **values derived from those stats**. The first layer is fixed (or near-fixed) during play. The second layer changes constantly.

This is not a universal architecture, but it shows up often enough that it's worth examining.

The Layers

Most TTRPGs organize character numbers into a layered structure. The most common pattern, present in some form across many design traditions, looks like this:

The Capability Layer

The character's **fundamental traits**. What they are, intrinsically.

These get many names across systems:

- Attributes (Pathfinder, World of Darkness)
- Stats (Mothership)
- Approaches (Fate Accelerated)
- Action Dots (Blades in the Dark)
- Tags (Risus)

The shape varies wildly. Call of Cthulhu tracks **eight** characteristics on a 0-100 scale. Free League's ALIEN RPG uses **four** attributes on a 1-5 scale. Lasers & Feelings tracks **one** number from 2 to 5. That's the entire character.

Each shape is a design statement.

Why this layer matters. This is where character identity lives. A high-Strength, low-Charisma character feels fundamentally different from a low-Strength, high-Charisma one. The capability layer is the foundation everything else builds on. Get it wrong and characters feel either samey or incoherent.

The Derivative Layer

Numbers **computed from** or derived from the capability layer. Things like Hit Points, Defense values, saving throws, perception, initiative.

This is where games differ most. Some games have many derivatives (D&D 5e has at least eight per character). Some have very few. Some have effectively none, computing everything on the fly.

Derivatives serve as bookkeeping convenience. Rather than recalculating "how much damage can this character take" every time, you write down a Hit Point number and update it as it changes. This is purely a usability decision, not a mechanical one, but it shapes how the game feels at the table.

Why this layer matters. Derivatives are what players actually track. Every "minus 2 to your defense" effect, every "regain 1d4 hit points" healing, every "you take 3 stress" complication touches the derivative layer. The more derivatives a system has, the more alive the character sheet feels, and the more bookkeeping the table has to manage.

The Sheet

The **physical (or digital) record** of all of the above.

This is the most underrated layer in design. The sheet is the interface between the player and the system. A sheet with seven rows of clearly-named values feels different from a sheet with thirty cluttered ones, even if the underlying math is identical.

Sheets express priorities. The biggest, most prominent boxes on a sheet tell the player "these matter most." The cluttered margins tell them "these matter less." Designers often spend more time on prose than on sheet layout, but the sheet is where the system meets the player every session.

Why this layer matters. A clean sheet teaches the system. A muddy sheet hides it.

How Different Games Use the Layers

The relationship between Capability, Derivative, and Sheet layers determines the entire feel of a system. A few examples to make this concrete:

Call of Cthulhu (Chaosium, 1981) uses eight characteristics, a long list of skills (each a derivative of characteristic + training), and tracks Sanity as a separate slowly-decaying pool. The derivative layer is huge. Most of what a player rolls is a derivative, not a raw characteristic. The sheet is dense. The feel is investigative, slow, lived-in. Numbers describe a person's whole capability profile.

ALIEN RPG (Free League, 2019) uses four attributes (Strength, Agility, Wits, Empathy), twelve skills, and tracks Health, Stress, Air supply, Radiation, and other survival pressures separately. The capability layer is small but the derivative layer is thematically loaded. Every tracked value reinforces the survival horror genre. The sheet feels heavy because every number on it is a way the character can die.

Mothership (Tuesday Knight Games, 2018) uses four stats (Strength, Speed, Intellect, Combat) and four saves (Sanity, Fear, Body, Armor) on a percentile scale, plus Wounds and Stress as separate damage tracks. Compact capability layer, deliberately split derivative layer. The sheet fits on a single page. The feel is brutal and quick. Every number is something to be afraid of.

Lasers & Feelings (John Harper, 2013) is the extreme minimalist example. One number, from 2 to 5. Below it you roll for Lasers (logic, science, cool); above it you roll for Feelings (passion, intuition, charm). No derivative layer. No tracking. The sheet is index-card sized. The feel is fast and wild. A one-shot game where the math takes ten seconds to learn.

Pathfinder 2nd Edition (Paizo, 2019) is the maximalist example. Six attributes, a deep skill list, multiple saving throws, AC, hit points, focus points, hero points, conditions, and dozens of tracked values across abilities and equipment. Every layer is dense. The sheet is multiple pages. The feel is detailed, tactical, and rewards mastery.

The point is not that any of these is wrong. They are answering the same design question with different priorities. **A design choice on the capability layer cascades into the derivative layer, which cascades into the sheet, which determines how the system feels.** A change in any layer reshapes the whole.

The Point

There is no correct number of attributes. No correct number of derivatives. No correct sheet layout.

But there is a correct consistency: the layers should reinforce each other. A horror game should not have a dense, optimization-heavy character sheet. A tactical game should not have a single number. A narrative-improv game should not have twelve tracked pools.

A designer's most consequential decision in this system is what to track and what to leave implicit. Too few tracked values and the game feels shapeless. Too many and the game feels like a spreadsheet. The right number depends entirely on what kind of game it is.

It is also genuinely fun to make these choices. Picking attributes for a system is one of the most identity-defining moves a designer makes. The choice between "Body, Mind, Spirit" and "Strength, Dexterity, Constitution, Intelligence, Wisdom, Charisma" is not just mechanical. It's a statement about what kind of stories the system wants to tell.

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*, the goal was a character number system that could:

- Support broad character identity without forcing twelve attributes
- Drive resolution and defensive math from the same source
- Track meaningful state changes (damage, exhaustion) without becoming a spreadsheet
- Stay easy to learn while supporting depthful build choices

Most existing systems either go small (3-4 attributes, light derivatives, narrative focus) or large (6+ attributes, dense derivatives, tactical focus). *Frontiers* wanted something between: light enough to learn quickly, deep enough to support distinct character builds across many genres.

That goal narrowed the design space:

Goal	Reasoning	Constraint
Few attributes, broad coverage	Players shouldn't have to learn twelve stats to start playing	Argues against detailed-attribute systems
One stat per concept	A character's "strength" should not be split across three different numbers	Argues against overlapping attributes
Derivatives that mean something	Each tracked value should be a real lever, not a bookkeeping byproduct	Argues against numerous derivative bloat
Damage tracks that escalate distinctly	Physical and mental harm should feel different mechanically	Argues for two damage tracks, not one

The Tradeoffs Considered

Frontiers studied several attribute architectures:

Three-attribute systems (Body/Mind/Spirit-style, common in indie design). Maximum simplicity. But characters end up either generalists or forced into a single specialty, with little nuance between the two.

Six-attribute systems (D&D babies, Pathfinder 2e). Rich differentiation. But most characters dump 1-2 attributes entirely, making them effectively four-attribute systems with extra baggage.

Four-attribute systems (ALIEN RPG, Free League's Year Zero Engine, and others). Compact enough to learn in a sitting, broad enough to support meaningful build differences. Used widely in Nordic indie design.

Skill-as-attribute systems (Blades in the Dark, Forged in the Dark). Replaces attributes entirely with broad skill domains. Elegant for narrative-forward play but harder to use as a base for tactical Defense math.

The four-attribute model offered the right balance. Each attribute becomes meaningful (no dump stats), each character has clear identity, and the derivative math stays clean.

The answer Frontiers landed on: **four attributes, ten derivatives, one consistent scaling pattern.**

Frontiers' Answer — Four Attributes, Layered Derivatives

Frontiers uses four core attributes, each on a 1-10 scale.

Attribute	Domain
❖ Vigor	Physical strength, endurance, bodily resilience
□ Finesse	Coordination, agility, reflexes, tempo

Attribute	Domain
▢ Acuity	Perception, reasoning, mental sharpness
* Resolve	Mental fortitude, composure, emotional endurance

Each attribute produces two values: an **Attribute Score** (the raw 1-10 number) and an **Attribute Modifier** (Score ÷ 2, rounded down).

The **Score** is used for thresholds and defensive values. The **Modifier** is added to active rolls.

Why Four

Three felt too few. Most three-attribute systems force every concept into a single bucket, so a character's "physical capability" must cover both raw strength and reflexes. That creates either generalists (one good physical stat covers everything) or false specialization (the strong character is also automatically agile).

Six felt too many. Six-attribute games consistently produce dump stats: characters who put 8 in three attributes and 16 in three. That's not deeper play; it's wider math with the same number of meaningful choices.

Four lets each attribute have a clear, distinct identity:

- ♦ **Vigor** and ▢ **Finesse** split the physical: durability versus agility
- ▢ **Acuity** and * **Resolve** split the mental: perception versus composure

A character can be physically strong but slow. Quick but fragile. Sharp-witted but easily rattled. Composed but oblivious. Each combination produces a recognizable archetype without needing more attributes to express it.

Why 1-10 Scale

The 1-10 scale was chosen because it maps cleanly to a Modifier of 0 to +5 (Score ÷ 2 rounded down). That gives the system six distinct competence tiers that fit comfortably alongside a d20 resolution roll.

A 1-20 scale would have given finer granularity but made each point of investment matter less. A 1-5 scale (like ALIEN RPG) would have been simpler but produced fewer differentiation tiers.

The 1-10 range is also intuitively read. "Acuity 7" feels meaningfully different from "Acuity 4" without needing a chart to interpret.

Attributes never go below 1. This is a hard floor. No effect, temporary or permanent, can push an attribute to 0. A character with ♦Vigor 1 is fragile but functional. A character with ♦Vigor 0 would be a mathematical edge case the engine refuses to produce, so to include this would need additional expansion beyond a simple drag-and-drop, this is for designer simplicity primarily.

Why Score AND Modifier

Some systems use only the raw stat. Some use only a derived modifier. Frontiers uses both, deliberately.

The **Score** is used where the full magnitude matters: Defense values, Maximum Wound and Fatigue thresholds, derivative calculations.

The **Modifier** is used where the bonus is added to a die: resolution rolls, attack rolls, derivative rolls like Awareness and Intuition.

This split lets the engine apply the right granularity in the right place. A Defense of 13 versus 14 is meaningfully different. A roll of d20+3 versus d20+4 is meaningfully different. The Score gives one; the Modifier gives the other. Same source, two pressures.

Frontiers' Answer — The Derivative Layer

Frontiers tracks ten derivatives. They cluster into four functional groups:

Group	Derivatives
Resilience	⊕Health Pool, ⊗Wounds, ⊖Fatigue
Defense	□Physical Defense, □Mental Defense
Tempo	➤Initiative, ⇨Movement Points, ⚡Energy
Sense	🕒Awareness, ∞Intuition

Each derivative is computed from one or two attributes plus other contributors (Equipment, Archetype, Abilities). None of them are random per character. Once the formula is set, the value is set.

Why Two Damage Tracks

Frontiers splits damage across ⊗Wounds (physical) and ⊖Fatigue (mental).

This is a deliberate departure from single-pool systems where everything that hurts the character pulls from one HP bar. The two-track model recognizes that "you are stabbed" and "you are terrified" are different kinds of harm, and a character should be able to be resilient to one without being equally resilient to the other.

It also gives the engine clean attack vectors. Combat pressures Wounds. Horror, isolation, and overextension pressure Fatigue. Designers building Encounters have two distinct levers to pull, and characters can build for either.

This decision is shared with games like Mothership (Wounds and Stress), Burning Wheel (separate physical and emotional tracks), and ALIEN RPG (Health and Stress). Frontiers' implementation is

closest in feel to ALIEN's: the two tracks are independent, each has its own max threshold, and exceeding either has serious consequences.

Why ●Awareness and ∞Intuition Are Separate

□Acuity already exists as an attribute. *Resolve already exists as an attribute. So why track ●Awareness and ∞Intuition as their own derivatives?

The answer is **rollability**. A character's perception in any given moment is not just their raw □Acuity. It's their □Acuity modified by their current Fatigue. A composed, alert character notices things a stressed, exhausted one would miss, even if their underlying □Acuity is identical. Additionally, it operates as a secondary stat layer where abilities or traits can be built on in a way that doesn't directly effect the core Attribute itself, but instead, a Derivative, opening the door for more complex ability design.

Similarly, Intuition is *Resolve under the same Fatigue pressure. A confident, clear-headed character reads social situations more accurately than the same character would after a brutal day.

By giving ●Awareness and ∞Intuition their own derivative formulas ($1 + \text{Modifier}$, with -2 per Fatigue applied), the engine creates two rollable values that respond to the character's current state, not just their permanent stats. They also each serve their own gameplay functionality in addition, but that goes without saying.

This means ⊖Fatigue actually pressures perception and social acuity, not by reducing the underlying attribute, but by pushing the derivative below useful thresholds. A ⊖Fatigue-3 character with □Acuity +3 has ●Awareness of $1+3-6 = -2$, which is roll-against-able and bad.

Why Anchored Derivatives Don't Recalculate Mid-Encounter

This is one of the engine's least-visible but most important rules.

When an attribute changes mid-Encounter (from a temporary effect, an ability, environmental damage), some derivatives update immediately. Others don't.

Live derivatives (Defenses, MP, Awareness, Intuition) update immediately. They reflect the character's current state and exist to be rolled.

Anchored derivatives (Maximum HP, Maximum Wounds, Maximum Fatigue, Initiative once locked) **don't update mid-Encounter**. They exist as stable thresholds that the rest of the engine references.

The reason is to prevent edge cases. If a character takes a temporary -1 ♦Vigor and their Maximum Wounds shrinks from 3 to 2 mid-fight, and they currently have 3 Wounds, do they suddenly die? In a strict-recalculation system, yes. In Frontiers, no. The threshold was set when the encounter started, and temporary changes don't retroactively trigger death conditions.

This rule trades a small amount of mathematical purity for a large amount of edge-case safety. It is unlikely to come up in normal play, but when it does, it prevents some genuinely awful moments.

Although, in stricter fiction that might be a benefit, so feel free to change that as you please from the position of a system designer.

Everything below this point is the actual mechanics. The earlier sections explained the theory. This section explains the values that go on the sheet.

Attributes

Every character is defined by four core Attributes:

- ♦ **Vigor**
- □ **Finesse**
- □ **Acuity**
- ✱ **Resolve**

Each Attribute is on a **1-10 scale**.

Each Attribute produces two values:

- **Attribute Score** — the full 1-10 value
- **Attribute Modifier** — $\text{Score} \div 2$, rounded down

Score	Modifier
1	0
2-3	+1
4-5	+2
6-7	+3
8-9	+4
10	+5

The **Score** is used for thresholds and defensive values. The **Modifier** is used for resolution rolls and rollable derivatives.

Attribute Floor: No Attribute Score can be reduced below 1 by any means, temporary or permanent. If an effect would reduce an Attribute below 1, it stops at 1.

Attribute Ceiling: Attribute Scores are capped at 10 by default. Systems may explicitly allow values beyond 10, but this is a system-level override.

Attribute Definitions

❖ **Vigor**

Vigor represents physical strength, endurance, bodily resilience, and resistance to trauma. It governs how much punishment a character can withstand and how physically demanding burdens affect them.

Vigor is commonly applied when brute force, stamina, durability, or survival under physical stress are involved.

□ **Finesse**

Finesse represents coordination, agility, reflexes, balance, and control of tempo. It governs how efficiently a character moves and how precisely they act under pressure.

Finesse is commonly applied when speed, precision, positioning, dexterity, or reaction time are involved.

□ **Acuity**

Acuity represents perception, reasoning, adaptability, pattern recognition, and mental sharpness. It governs how clearly a character observes the world and how quickly they process information.

Acuity is commonly applied when investigation, analysis, problem-solving, or environmental awareness are involved.

✱ **Resolve**

Resolve represents mental fortitude, emotional stability, composure under pressure, and psychological endurance. It governs how much mental strain a character can endure.

Resolve is commonly applied when composure, conviction, morale, or resistance to psychological pressure are involved.

Derivatives

Derivatives are statistics shaped by Attributes (and sometimes other sources) but functioning independently during play.

The ten Derivatives in Frontiers:

Derivative	Function
✚ Health Pool (HP)	Physical damage capacity
⊗ Wounds	Lasting physical injury track
⊖ Fatigue	Lasting mental strain track
➤ Initiative	Turn order in Encounter Play
⚡ Energy	Effort and exertion currency

Derivative	Function
□ Physical Defense	Defense against physical attacks
□ Mental Defense	Defense against mental attacks
● Awareness	Perception rolls
∞ Intuition	Social and emotional reading rolls

When an Attribute contributes to a Derivative, the rule states whether it uses the **Score** or the **Modifier**.

❖ Vigor Derivatives

+ Health Pool (HP)

+HP represents a character's current physical vitality.

When a character advances a Level (including Level 1), their Maximum +HP increases by:

❖Vigor Modifier + Archetype HP Modifier

The **Archetype HP Modifier** is a value (typically +4 to +8) defined by the character's Archetype. It represents the durability identity of that Archetype.

There are no other contributors to Maximum HP in the Default Rules, although in some systems Origin and Background could also contribute, but that is not true in the base engine.

If ❖Vigor changes permanently (through levelup or similar), recalculate Maximum HP retroactively as if the new ❖Vigor had been in effect since Level 1. This represents character development. Permanent Attribute increases reward earlier choices.

If ❖Vigor changes temporarily, Maximum HP does not change (see Anchored Derivatives below).

⊗ Maximum Wounds

Maximum ⊗Wounds = 1 + ❖Vigor Modifier

Each ⊗Wound imposes: - -2 to □Physical Defense - -2 to □Mental Defense

Penalties stack.

Wounds are gained when a character drops to 0 +HP, when an Ability inflicts one, or when a player chooses Wound during an Overspend.

If a character would gain a Wound that exceeds their Maximum Wounds: **the character dies**. This is the engine's hard death threshold.

Wounds are removed during □Downtime at a rate of 1 per hour.

▢ Physical Defense

▢ **Physical Defense = 5 + the higher of (♦Vigor Modifier or ▢Finesse Modifier) + Equipment Bonuses**

Defense acts as a Difficulty Rating against incoming attacks. Attackers must roll equal to or higher than the relevant Defense to succeed.

The "highest of" structure means a character can specialize in either ♦Vigor or ▢Finesse without being penalized for not maximizing both. A ♦Vigor-focused character defends with ♦Vigor; a ▢Finesse-focused character defends with ▢Finesse.

▢ Finesse Derivatives

⇒ Movement Points (MP)

⇒ **MP = 3 + ▢Finesse Modifier**

1 ⇒MP = 1 ▢Unit moved.

⇒MP refreshes at the start of each Turn. Movement does not normally cost ♦AP.

► Initiative

Initiative determines turn order in Encounter Play. Each character has an Initiative Value (IV) on a 0-10 scale. **Higher acts first.**

At the start of an Encounter: - The character starts with their base IV (provided by Archetype). - They may adjust their IV by adding or subtracting up to their ▢Finesse Modifier. - The final IV is locked for the duration of the Encounter unless altered by effects.

Initiative also contributes to ▢Physical Defense via the "highest of ♦Vigor or ▢Finesse Modifier" structure above.

▢ Acuity Derivatives

● Awareness

● **Awareness = 1 + ▢Acuity Modifier**

●Awareness is a rollable value used to notice hidden threats, environmental changes, concealed objects, or subtle sensory details.

When making an Awareness check, roll d20 + ●Awareness. The current value is reduced by 2 for each ⊖Fatigue the character has.

Systems may allow passive use of Awareness in certain contexts (a passive value of 10 + Awareness is a common implementation), but it is fundamentally roll-based.

Additional Skills

A character gains a number of additional Skills equal to their $\square$ **Acuity Modifier**.

These Skills represent expanded mental breadth and specialization beyond a character's baseline training. (See the Skills design doc for full Skill rules.)

$\square$ **Mental Defense**

$\square$ **Mental Defense** = 5 + the higher of ($\square$ **Acuity Modifier** or $\ast$ **Resolve Modifier**) + **Equipment Bonuses**

Same structure as $\square$ Physical Defense. A character can specialize in $\square$ Acuity or $\ast$ Resolve without being penalized for not maximizing both.

$\ast$ **Resolve Derivatives**

∞ **Intuition**

∞ **Intuition** = 1 + $\ast$ **Resolve Modifier**

∞ Intuition is a rollable value used to assess deception, detect manipulation, read social dynamics, and interpret emotional nuance.

When making an Intuition check, roll d20 + ∞ Intuition. The current value is reduced by 2 for each $\ominus$ Fatigue the character has.

$\ominus$ **Maximum Fatigue**

Maximum $\ominus$ Fatigue = 1 + $\ast$ **Resolve Modifier**

Each $\ominus$ Fatigue imposes: - -2 to $\odot$ Awareness - -2 to ∞ Intuition

Penalties stack.

Fatigue is gained from Overspending Energy (when the player chooses Fatigue), from prolonged stress or deprivation, from specific Abilities, or from narrative strain at the GM's discretion.

If a character would exceed their Maximum Fatigue: they immediately fall **Unconscious**. They remain Unconscious until their Fatigue drops below maximum.

Fatigue is removed during $\square$ Downtime at a rate of 1 per hour.

$\square$ **Mental Defense (Resolve Portion)**

Resolve contributes to $\square$ Mental Defense through the "highest of $\square$ Acuity or $\ast$ Resolve Modifier" structure above.

⚡ Energy

⚡Energy is shared across all Attributes (none of them increase Maximum ⚡Energy). It is determined by character creation and grows through advancement.

Starting Maximum ⚡Energy is typically 3 at Level 1. (See the Characters design doc for advancement rules.)

For full Energy mechanics (spending, Overspending, recovery), see the Flow of Play design doc.

When Values Update

Frontiers separates Derivatives into two categories based on how they handle Attribute changes during play.

Live Derivatives

Live Derivatives **update immediately** whenever the underlying Attribute changes. They reflect the character's current state.

- □ Physical Defense
- □ Mental Defense
- ⇌ Movement Points
- ● Awareness
- ∞ Intuition
- Additional Skills granted by □Acuity Modifier
- Initiative adjustment at Encounter start (the choice to add or subtract □Finesse Modifier)

If ♦Vigor temporarily drops, Physical Defense drops with it. If □Acuity temporarily increases, Awareness increases with it. These changes persist as long as the underlying Attribute change persists.

Anchored Derivatives

Anchored Derivatives **do not update mid-Encounter** when Attributes change temporarily. They remain at the value they had when the Encounter began (or when last set during a stable moment).

- + Maximum HP
- ⊗ Maximum Wounds
- ⊖ Maximum Fatigue
- ► Initiative value once locked at Encounter start

This prevents edge cases where a temporary debuff suddenly reduces a maximum threshold below the character's current value, triggering death or incapacitation in ways that would feel unfair.

If an Attribute changes **permanently**, Anchored Derivatives recalculate normally: at the next reasonable pause if not under active pressure, or at the end of the current Encounter if pressure is active.

Tuning the Feel

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

Attribute Range

The default 1-10 scale produces 6 distinct competence tiers. Variations:

Scale	Modifier Range	Effect
1-5	0 to +2	Smaller range, less differentiation, faster math
1-10 (default)	0 to +5	Balanced
1-15	0 to +7	More tiers, more granular character investment
1-20	0 to +10	Heroic scaling, large gaps between novice and master

Wider ranges require recalibrating Difficulty Ratings and Defense formulas.

Archetype HP Modifier Range

The default range is +4 to +8. This produces clear durability tiers across Archetypes.

- **Tighter range** (e.g., +4 to +6): Archetypes feel more similar in durability. ♦Vigor dominates.
- **Wider range** (e.g., +3 to +10): Strong tank/glass-cannon distinctions between Archetypes.

Calibrate based on whether the system wants distinct durability roles or relatively interchangeable characters.

Defense Calibration

The default Defense formula (5 + highest Modifier + Equipment) targets ~80% hit rates at parity. Variations:

- **Lower base** (3 + highest Modifier): Defenses are easier to hit. Combat feels swingier.
- **Higher base** (8 + highest Modifier): Defenses are harder to hit. Combat feels grindier.
- **Two Modifiers added** (5 + ♦Vigor Modifier + □Finesse Modifier): Original 2026-1 formula. Specialists become extremely tanky; generalists fall behind. Not recommended unless paired with ranged attacker scaling.

Damage Track Granularity

Default Maximum Wounds and Maximum Fatigue are $1 + \text{Modifier}$ (range 1-6). Variations:

- **Higher base** ($3 + \text{Modifier}$): More buffer before death/Unconsciousness. Less lethal.
 - **Per-Level scaling**: Maximum Wounds increases by 1 per few Levels. Mid-Level characters become genuinely durable.
 - **Removed entirely**: Wounds become a binary "you are wounded" state. Drastic simplification.
-

Edge Cases

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

Attribute Reduced to 1. If an Attribute Score would be reduced below 1, treat it as 1. The Modifier is calculated normally (1 produces Modifier 0). All Live Derivatives update immediately. Anchored Derivatives do not change unless the reduction is Permanent.

Multiple Attribute Changes Simultaneously. If multiple Attribute changes occur at the same time: 1. Apply all Score changes first (respecting the floor of 1). 2. Recalculate Modifiers. 3. Update Live Derivatives. 4. Anchored Derivatives follow Permanent/Temporary rules.

No intermediate recalculation occurs between stacked effects.

Modifier Stacking. If multiple effects modify the same Attribute Score: - Additive modifiers stack. - Multiplicative effects apply after additive adjustments. - If a rule modifies the Modifier directly (rather than the Score), apply that after recalculating from Score.

Unless stated otherwise, all bonuses and penalties stack.

Modifier Modifying a Modifier. If an effect modifies an Attribute Modifier directly: - It does not alter the Attribute Score. - It affects resolution rolls and any Derivative that explicitly references the Modifier. - It does not affect Derivatives that reference the Score.

Temporary Attribute Expiration Mid-Encounter. If a temporary Attribute change expires during Encounter Play: - The Attribute Modifier updates immediately. - Live Derivatives update immediately. - Anchored Derivatives remain unchanged unless explicitly recalculated by the effect.

Expiration does not retroactively invalidate resolved Actions.

Permanent Attribute Change During Encounter. If an Attribute permanently changes during Encounter Play: - Modifier updates immediately. - Live Derivatives update immediately. - Anchored Derivatives recalculate at the **end** of the current Encounter unless the rule explicitly states immediate recalculation.

This prevents mid-combat threshold collapse.

Anchored Derivative Reduction Below Current Value. If a permanent change reduces an Anchored Derivative maximum below the character's current value: - Reduce the current value to the new maximum. - If this causes Maximum Wounds or Maximum Fatigue to be exceeded, apply death or Unconsciousness accordingly.

Temporary changes cannot trigger this effect.

Defense Changes During Attack Resolution. If an Attribute change alters ☐Physical Defense or ☐Mental Defense during an attack: - Use the Defense value at the moment the attack roll is made. - Subsequent Defense changes do not retroactively alter that roll.

Initiative and Temporary ☐Finesse Changes. If ☐Finesse temporarily changes after Initiative has been established: - It does not retroactively alter current IV. - It may affect IV at the start of the next Encounter. - It affects ⇒MP immediately.

Attribute Change While Unconscious. If an Attribute changes while a character is Unconscious: - Apply recalculation normally. - If the change removes the condition causing Unconsciousness, the character regains eligibility to act.

Attribute Increase Beyond 10. If a rule explicitly allows an Attribute to exceed 10: - The Modifier continues scaling normally ($\text{Score} \div 2$ rounded down). - This is considered a system-level override of the default cap.

Such cases are rare and intentional.

Simultaneous Level Advancement and Attribute Change. If a character Levels up and also gains a permanent Attribute increase at the same time: 1. Apply the Attribute change first. 2. Recalculate Modifiers. 3. Apply Level-based Derivative scaling (e.g., HP increase).

This prevents double-counting inconsistencies.

Awareness or Intuition Reduced Below 0. If Fatigue stacking reduces ●Awareness or ∞Intuition below 0, the value is applied as a negative modifier to the roll. There is no floor. A character with Fatigue 5 and □Acuity 1 has Awareness of $1+0-10 = -9$, meaning they roll d20-9 for any Awareness check. This is intentional. Severe Fatigue is debilitating.

Variants

Variants modify specific parts of the Default Attribute & Derivative structure while preserving the four-attribute framework.

Linear Modifier Variant

What changes: Attribute Modifier formula becomes $\text{Modifier} = \text{Score}$ (no division).

What stays: 1-10 scale, all Derivative relationships, Anchored vs Live distinction.

What shifts: - Resolution Rolls add the full Score. - Derivatives referencing the Modifier use the full Score. - Scaling intensity dramatically increases. A high-Acuity character has +10 to perception checks instead of +5.

Use when: Building a heroic-tier system where character investment should produce dramatic mathematical advantages. Difficulty Ratings must be recalibrated upward.

Two-Attribute Defense Variant

What changes: Defense formula becomes $\text{Defense} = 5 + \text{relevant Modifier 1} + \text{relevant Modifier 2} + \text{Equipment}$.

What stays: Wound and Fatigue penalty effects.

What shifts: - Specialists who max both relevant Attributes become extremely tanky. - Generalists with one strong Attribute become more vulnerable. - The "highest of" rule no longer rewards single-stat specialization for Defense.

Use when: A system wants Defense to reward dual-Attribute investment, or wants combat to be slower and more deliberate.

Immediate Pool Recalculation Variant

What changes: All Derivatives, including Anchored ones, recalculate immediately whenever the underlying Attribute changes.

If recalculation reduces a maximum below current value, clamp current to the new maximum and apply death or Unconsciousness if thresholds are exceeded.

What stays: Score and Modifier rules, formula relationships.

What shifts: High volatility. Temporary debuffs can become lethal. Suitable for harsher or grittier systems where temporary effects should carry maximum weight.

Use when: Building a survival horror or grimdark system where every effect should feel dangerous.

Attribute Cap Removal Variant

What changes: Remove the 10 maximum cap. Attributes may scale beyond 10 indefinitely.

What stays: Modifier calculation ($\text{Score} \div 2$ rounded down).

What shifts: - Supports epic or mythic scaling. - Difficulty Ratings and Defense expectations require recalibration upward. - Build optimization can run away.

Use when: Building a system that intends to scale into superhuman or godlike tiers.

Alternatives

Alternatives replace major portions of the Attribute & Derivative architecture. Adopting these requires recalibration of Abilities, Difficulty Ratings, and Level progression assumptions.

Three-Attribute Alternative

Core change: Reduce to three Attributes (Body, Mind, Spirit, or similar).

Defense formulas, Resilience derivatives, and rollable derivatives all collapse into combinations of three Attributes instead of four.

What you gain: - Maximum simplicity at the Attribute layer. - Cleaner sheet, faster character creation.

What you lose: - The ♦Vigor/□Finesse and □Acuity/*Resolve distinctions. A "Body" stat must cover both raw strength and reflexes; a "Mind" stat must cover both perception and composure. - Some build identity. Specialists feel less differentiated.

Use when: Building a system where speed of play matters more than character build differentiation.

No Derivatives Alternative

Core change: Remove derived maximum calculations tied to Attributes. +HP, ⊖Fatigue, and ⊗Wounds become fixed by Level or Archetype only.

Attributes affect only resolution rolls.

What you gain: - Cleaner separation of identity (Attributes) from survivability (Levels). - Easier Encounter balancing.

What you lose: - ♦Vigor's downstream impact disappears. A high-Vigor character is no more durable than a low-Vigor one. - Build choices become primarily about resolution effectiveness, not survivability.

Use when: Building a system where Attributes should describe capability without affecting durability.

Unified Resilience Alternative

Core change: Replace +HP, ⊗Wounds, and ⊖Fatigue with a single track called Resilience.

Resilience Maximum = Base + Level. Attributes modify resolution rolls only, not survivability thresholds.

When Resilience reaches 0, apply escalating penalties. When it reaches a death threshold, the character dies.

What you gain: - Massively simpler resilience tracking. - One pool to deplete and recover.

What you lose: - The distinction between physical and mental harm. Combat and horror pressure the same pool. - The mechanical hooks for two-vector encounter design.

Use when: Building a rules-light system where the two-track damage model is overhead the system can't afford.

Attribute-Lite Alternative

Core change: Retain Attributes for narrative and roll guidance only. Remove Attribute Modifiers entirely.

Resolution Rolls add Skill or Level bonuses only. Defenses become static values determined by Level or Equipment.

What you gain: - Reduced math density. - Attributes become descriptive rather than mechanical drivers.

What you lose: - Most of the engine's tactical depth. Build differentiation now lives entirely in Skills and Abilities.

Use when: Building a strongly narrative system where Attributes should feel descriptive, not load-bearing.

A Note to System Designers

Attributes and Derivatives are the layer of the engine that touches the character sheet most directly. Players see these numbers every session. Every roll, every defense check, every damage event passes through them.

A system that adopts Frontiers' defaults inherits a particular feel: - Four broad attributes with no dump stats - A medium-density derivative layer - Two damage tracks that pressure independently - Defense math that rewards specialization - Anchored thresholds that protect against edge cases

This feel is deliberate and tuned.

Modifying the defaults with a Variant or an Alternative changes that feel. Attribute count cascades into every Skill, every Ability, every Defense calculation. Derivative count cascades into character sheet design and bookkeeping load. Anchored vs Live decisions cascade into how dangerous temporary effects feel.

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.

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**.