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 difference between what a character is and what they have learned to do. None of this is Frontiers-specific. If you already understand how skills, training, and proficiency systems work, you can skip ahead to Part Two.

What Training Layers Do

Tabletop characters need a way to express the difference between **innate capability** and **learned expertise**.

A strong character is strong. That's what their stat says. But a character who has spent years training in unarmed combat is not just strong; they're trained, and that training is a different kind of capability than raw strength.

Most TTRPGs handle this through a layer of mechanics that sits on top of attributes. The names vary wildly:

- Skills (most d20 lineage games)
- Proficiencies (D&D babies, Pathfinder)
- Action Dots (Blades in the Dark)
- Talents (Free League's Year Zero Engine)
- Specialties (Mothership)
- Tags (Risus)
- Approaches (Fate Accelerated)
- Or absent entirely (some narrative-forward systems collapse this layer into pure stats)

Whatever they're called, the training layer answers the same question: **how does practiced expertise differ from raw capability?**

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

This question is one of the most consequential in TTRPG design, because the answer shapes how characters get differentiated. Two characters with identical attributes should still feel like distinct people. The training layer is usually how that happens.

How Different Games Handle Training

The diversity here is enormous. A few examples to make this concrete:

Call of Cthulhu (Chaosium, 1981) uses an extensive percentile skill list. Each Skill has its own value (often 30-70%), and a character rolls under the Skill value to succeed. Skills here are the mechanical identity of a character. Two investigators with identical characteristics but different Skill spreads play completely differently. The training layer dominates resolution.

Pathfinder (Paizo, 2009/2019) uses Proficiency tiers (Untrained, Trained, Expert, Master, Legendary). Each tier adds a flat bonus on top of attribute Modifier. A character is trained or not, and how trained they are matters. The system rewards depth of investment in specific Skills.

Blades in the Dark (One Seven Design, 2017) uses Action Dots: 0 to 3 dots in twelve broad action verbs (Skirmish, Hunt, Survey, Sway, etc.). Action Dots simultaneously serve as attribute and skill. A character with 3 Skirmish dots is both naturally combative AND trained in violence. The two layers are deliberately fused.

Mothership (Tuesday Knight Games, 2018) uses Skills as flat percentile bonuses (+10%, +15%, or +20%) applied to Stat checks. Skills are explicit but lightweight. A character without the relevant Skill rolls the raw Stat; a character with it rolls Stat + Skill bonus. Training narrows uncertainty without dominating it.

Risus (S. John Ross, 1993) collapses everything into Cliches: short descriptive tags like "Brave Knight" or "Cunning Thief" with dice values. There's no separation between attribute and skill. The Cliche is both.

The point is not that any of these is wrong. They are answering the same design question with different priorities. **Some games make training a heavy mechanical layer. Some make it light. Some skip it entirely.** The choice cascades into how character creation feels, how progression feels, and how players differentiate their characters from one another.

The Three Questions Every Training System Answers

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

1. How granular is training?

Some systems define dozens of specific skills (Call of Cthulhu's list runs to over fifty). Others define a handful of broad domains (Blades' twelve verbs, Fate's six approaches). Still others define training in even broader strokes (Risus's freeform Cliches).

Why this matters. Granular systems let players build precisely-defined characters but require more bookkeeping and slower character creation. Broad systems are faster to learn and play but

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

can make characters feel less distinct. The granularity choice determines how much texture the training layer adds to character identity.

2. Is training binary or scaled?

Some systems treat training as a yes/no: either you have the skill or you don't (Mothership's specialty system, Mörk Borg's knack-style approach). Others treat it as a scale: you can be a little trained, somewhat trained, very trained, or legendarily trained (Pathfinder's proficiency tiers, Call of Cthulhu's percentile growth).

Why this matters. Binary systems are faster at the table and create cleaner build choices ("do I have this or not?"). Scaled systems support deeper progression and let trained characters feel meaningfully more competent than novices. The choice determines whether training is a qualitative or quantitative signal.

3. What does training mechanically do?

This is the question most designers underthink. Training can:

- Add a flat bonus to a roll
- Unlock the ability to attempt something at all
- Grant Favor (or advantage, or rerolls)
- Reduce the cost of an action
- Allow narrative permissions ("you know this NPC's secret")

Most systems pick one mechanic and stick with it. Some pick two and feel cluttered. Almost no system picks three or more without becoming hard to teach.

Why this matters. The mechanical identity of training determines its feel. A flat-bonus system feels precise and incremental. A Favor-grant system feels surge-y and dramatic. An access-grant system feels permission-based. Each produces a different relationship between the character sheet and the fiction.

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 clarity: the training layer should have one mechanical identity that's easy to explain. A trained character should do something specific that an untrained character can't, and that something should be communicable in one sentence.

A designer's most consequential decision in this layer is **what to make training do**, mechanically, and how to balance it against the natural pull of attributes. Get it wrong and the system either feels like attributes are everything (training is decoration) or feels like training is everything (attributes are decoration).

The goal is for both layers to matter, in different ways, for different moments.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

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' Skills, the goal was a training layer that:

- Felt distinct from raw Attributes (training should do something attributes don't)
- Was easy to learn (one core mechanic, not a menu of options)
- Supported broad character differentiation without forcing every character into long Skill lists
- Worked across many genres without prescribing a Skill list the engine forces on designers
- Created design hooks for Abilities, Vector Marks, and Archetypes to interact with

Most existing approaches either go deep (Call of Cthulhu's percentile-skill system) or go flat (Mothership's small Skill bonus). Frontiers wanted something between: tight enough to teach quickly, layered enough to support distinct character builds.

That goal narrowed the design space:

Goal	Reasoning	Constraint
One mechanical identity for training	A trained character should do a specific, nameable thing	Argues against multi-benefit menus
Two-layer differentiation	Categories of training and specific training should both matter	Argues for paired systems
Binary specific Skills	Have/don't have is faster than tiered training	Argues against Skill ranks
Designer-defined Skill list	Different genres need different Skills	Argues against engine-prescribed Skill lists
Narrow but expandable	The default state should be small; Abilities/Marks expand it	Argues for a low-overhead baseline

The Tradeoffs Considered

Frontiers studied several training architectures:

Long Skill lists (Call of Cthulhu, older Fate editions). Rich character differentiation. But character sheets fill up fast, and most Skills only matter occasionally.

Tiered proficiency (Pathfinder). Strong sense of progression, clear mathematical scaling. But adds tracking complexity and forces designers to balance every tier.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Broad action verbs (Blades in the Dark). Elegant, fast, dramatic. But fuses the training and attribute layers, which Frontiers wanted to keep separate.

Skill-as-bonus (Mothership). Lightweight, easy to teach. But a single mechanical identity doesn't give designers much to hook onto with Abilities or progression.

Skill-as-permission (some narrative-forward systems). Strong fictional grounding. But hard to use as a base for tactical resolution math.

The answer Frontiers landed on: **two layers, two mechanics. Categories grant flat modifiers, and specific Skills grant Favor.**

Frontiers' Answer — Categories and Skills

Frontiers' Skill system has two layers that work together but operate differently.

Layer	Mechanic	How You Get It
Skill Category	Flat modifier (+1 to +5) added to rolls within the Category	Vector Marks, Archetype features, Abilities, Equipment
Specific Skill	Favor on rolls when the Skill applies	Character creation, Vector Marks, Background, Archetype

A character with a Category modifier benefits broadly across all rolls in that Category. A character with a specific Skill benefits sharply on rolls where that Skill applies. The two layers stack: a character with **+3 Body Category** AND **the Athletics Skill** gets both a +3 modifier and Favor on Athletics rolls.

Why Two Layers

A single-layer Skills system forces a tradeoff between breadth and depth. If Skills are broad (one Skill covers many rolls), characters feel underspecialized. If Skills are narrow (each Skill covers few rolls), characters need many Skills to feel competent and the sheet bloats.

Two layers separate these concerns. **Categories provide breadth.** A character with +3 Body is reliably competent across all physical rolls (running, climbing, fighting, surviving). **Specific Skills provide depth.** A character with the Athletics Skill is genuinely trained in that specific domain, getting Favor on the rolls where it applies.

The two layers answer different questions:

- How broadly capable is this character? → Category modifiers
- What is this character's actual training? → Specific Skills

A character can be a generalist (broad Categories, few Skills), a specialist (narrow Skills, no Category investment), or a focused expert (one Category + matching Skills). All three feel mechanically distinct.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Why Specific Skills Are Binary

Specific Skills are have/don't-have. There are no "Athletics +1" vs. "Athletics +5" tiers. You're either trained or you're not.

This was deliberate. Tiered Skills add tracking complexity that didn't serve the engine's goals. Binary Skills support faster character creation, cleaner build choices, and simpler GM adjudication ("Does the player have the Skill? Yes or no.")

Depth still exists in the system. It lives in **Categories**, in **Vector Marks**, and in the combination of Categories and Skills. A character with +5 Body and three Body Skills is meaningfully deeper than one with +1 Body and one Skill, even though all the Skills themselves are binary.

Why Category Bonuses Are Build Rewards, Not Defaults

A character does not "start trained" in a Category. There is no baseline Category modifier. Category bonuses are earned through Vector Marks, Archetype features, Abilities, and (in some systems) Equipment.

This was deliberate. Making Categories a baseline training layer would dilute their value. Every character would have some Category bonus, making the system feel like a flat math tax. Making them a build reward keeps them meaningful: a character with +3 Body has invested in being broadly physical. That investment shows.

It also keeps the system tight at character creation. A Level 1 character has specific Skills (from Background, Archetype, etc.) and no Category modifiers unless their Archetype explicitly granted one. Categories grow over time as the character makes build choices.

Why Designer-Defined Skill Lists

Frontiers does not ship with a fixed Skill list. The default Skill list is left to the system designer.

This was a structural choice, not laziness. A fantasy game's Skills (Athletics, Stealth, Arcana, Survival) and a sci-fi game's Skills (Zero-G, Hacking, Xenobiology, Demolitions) are fundamentally different. Forcing one list on both genres would constrain the engine in exactly the wrong place.

What Frontiers does prescribe is the Category structure. Three Categories (**Body, Mind, Voice**), each holding the system designer's choice of specific Skills. A designer building a system picks the specific Skills, sorts them under the appropriate Category, and the engine handles the rest.

This gives designers structure (the three-Category framework) and freedom (the specific Skills inside each).

The Three Skill Categories

Frontiers uses three Skill Categories: **Body, Mind, and Voice.**

Category	Covers
Body	Physical rolls — athletics, stealth, fighting, endurance, manual work
Mind	Mental rolls — perception, knowledge, investigation, focus, reasoning

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Category	Covers
Voice	Social rolls — persuasion, deception, performance, negotiation, presence

These names are deliberately broad. They cover the three major modes a character can engage with the world: physically, intellectually, and socially. The names do not connect to specific Attributes (Body is not "the Vigor Category"; physical rolls might use Vigor, Finesse, or both depending on the task). This decoupling keeps the systems independent and prevents cascading balance issues.

A note on naming: Body/Mind/Voice are functional placeholder names. Systems built on Frontiers may rename them to match genre and tone (a horror game might prefer Form/Thought/Presence; a martial arts game might use Stance/Focus/Spirit). The Category structure is what matters; the specific labels are a system-level choice.

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

What a Skill Is

A **Skill** is trained expertise in a specific domain.

Skills are binary: a character either has a Skill or they do not. There are no Skill ranks or tiers in the default engine.

A character without a Skill may still attempt the relevant action. The difference is access to **Favor on the roll**.

Skills do not determine whether a character may attempt something. They determine how effectively training enhances that attempt.

Skill Categories

Every Skill belongs to one of three Categories:

- **Body** — physical Skills
- **Mind** — mental Skills
- **Voice** — social Skills

A character may have a **Category modifier** (from +1 to +5) that applies to all rolls within that Category, regardless of whether a specific Skill applies. This is in conjuction and additive to whichever Attribute is applied to that roll as well, making a simple dual-modifier system for most rolls.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Category modifiers are gained from: - Archetype features - Vector Marks (see the Characters design doc) - Abilities - Equipment (in some systems)

A character does not start with any Category modifier. Category investment is a build reward, not a baseline.

Category Modifier Cap

Category modifiers cap at **+5 per Category**. All sources count toward this cap.

Example: a character with +2 Body from their Archetype and +3 Body from Vector Marks is at the cap. An additional +1 from an Ability would not increase their modifier.

The cap is per-Category, not total. A character could in theory invest up to +5 in each of Body, Mind, and Voice, though the opportunity cost (Vector Marks could go to other things) makes this rare in practice.

Skill Benefits

When a Skill applies to a roll, the character gains **Favor on the roll**. That is the entire mechanical effect of having a specific Skill.

If a Category modifier also applies, both apply: the character gains the Category modifier AND Favor.

Only one Skill applies per roll, even if multiple Skills could plausibly fit. The player picks which Skill to apply (and therefore which Skill grants Favor).

When Skills Apply

A Skill applies when: - The character's approach clearly relies on trained technique - The character is leveraging specialized knowledge - The character is performing something they have practiced deliberately

The GM determines whether a Skill applies. The standard guideline:

- The **task** determines the Attribute.
- The **approach** determines whether a Skill applies.

If a character lacks any relevant Skill, they may still attempt the action. They use the governing Attribute Modifier (and any applicable Category modifier) but do not gain Favor.

Favor and Stacking

Favor from a Skill stacks with Favor from other sources, up to the engine cap of 3 stacks of Favor per roll. (See the Resolution System design doc for full Favor rules.)

If a character has more than 3 sources of Favor on a single roll, additional Favor is wasted. This is rare in normal play but worth knowing for designers building Favor-heavy Abilities.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Gaining Skills

The default engine grants Skills through:

1. Origin / Background / Archetype

Character creation choices grant a fixed number of Skills. (See the Characters design doc for specifics.)

2. Acuity Modifier

A character gains additional Skills equal to their **Acuity Modifier**.

If the Modifier is 0, no additional Skills are gained from this source. (Acuity Modifier cannot be negative under the floor-of-1 rule.)

These additional Skills are chosen at character creation unless the system specifies staged acquisition.

3. Advancement

Vector Marks may grant new Skills. (See the Characters design doc for Vector Mark options.)

Systems may also grant Skills through narrative milestones, training arcs, or downtime study.

Losing Skills

Skills are **permanent** once gained. Temporary loss of access may occur through narrative or specific effects, but Attribute reduction does not remove known Skills.

Systems may introduce Skill loss for specific thematic reasons (amnesia plots, corruption mechanics, long-term injury). Such mechanics are system-defined, not engine-defined.

Skills and Resolution

When making a Resolution Roll where a Skill applies:

1. Determine the governing Attribute.
2. Add the Attribute Modifier to the roll.
3. Add the Category modifier (if any) to the roll.
4. Apply Favor from the Skill.
5. Apply any other Favor or Disfavor sources.
6. Resolve the roll normally.

Example: A character with Acuity 6 (Mod +3), +2 Mind Category, and the Investigation Skill is searching for clues. The roll is d20+3 (Acuity) +2 (Mind) with Favor (Investigation Skill).

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Skills and Contested Resolution

When a roll is contested:

- Skills apply normally on both sides.
- Favor applies to the Resolution Die.
- If the contest includes a Gradient roll, Favor applies to the Gradient Die instead if the player declares so before rolling.

Skills and Multiple Rolls in One Action

If a single ✦Action requires multiple rolls:

- A Skill's Favor applies to only one roll within that Action.
- The player declares which roll receives the benefit.
- Category modifiers apply to all rolls within the Action.

Skills and Action Economy

Using a Skill does not cost ✦AP. Skills are passive enhancements that activate when relevant. They do not need to be declared as Actions.

Example Skill Categories

A system designer building on Frontiers will define their own Skill list. The example below shows what a baseline fantasy/adventure system might use, sorted by Category.

These are example Skills, not engine-prescribed Skills.

Body Skills (Example)

- **Athletics** — running, jumping, climbing, swimming
- **Stealth** — moving unseen, hiding, sneaking
- **Acrobatics** — balance, tumbling, agility under pressure
- **Endurance** — sustained physical effort, resisting exhaustion

Mind Skills (Example)

- **Perception** — noticing details, spotting hidden things
- **Investigation** — analyzing evidence, deducing patterns
- **Knowledge** — recalling specific facts (history, lore, etc.)
- **Survival** — tracking, navigation, foraging

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Voice Skills (Example)

- **Persuasion** — diplomacy, sincere negotiation
- **Deception** — lying, misleading, disguising intent
- **Performance** — entertainment, distraction, command of attention
- **Insight** — reading people, sensing motive (overlaps with ∞Intuition)

A system designer will adjust this list to fit their genre. A sci-fi system might add Hacking (Mind), Zero-G (Body), or Xenolinguistics (Voice). A horror system might add Composure (Mind) or Will (Voice). The Category structure handles all of these. Designers just pick the Skills.

Alternatively, a designer could just work with their playes on what skills are allowed on a character-by-character basis if wishing to replicate the open-ness of skill design something like Cypher System or Daggerheart has.

Competencies

Competencies are a separate system from Skills. They handle access and proficiency, not roll bonuses.

What a Competency Is

A **Competency** is the formal training, familiarity, or licensure required to use a specific tool, weapon, armor type, language, or system without penalty.

Competencies are binary: a character has a Competency or they don't.

Competencies do not grant Favor. They do not grant Category modifiers. They grant access: the ability to use a thing properly.

Examples of Competencies

Common Competencies include:

- **Weapon Competencies** (Light Weapons, Heavy Weapons, Ranged Weapons, Exotic Weapons, etc.)
- **Armor Competencies** (Light Armor, Heavy Armor, Shields, etc.)
- **Tool Competencies** (Healer's Kit, Alchemy Kit, Thieves' Tools, etc.)
- **Languages**
- **Vehicle Operation**
- **Magical Systems** (in systems where magic requires structured access)

The specific Competency list is system-defined. Frontiers does not prescribe one.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Competencies vs. Skills

These two systems answer different questions:

Question	System
Can the character use this thing properly?	Competency
Does the character get a bonus when doing it?	Skill

A character with Heavy Armor Competency can wear plate without penalty. A character with the Athletics Skill gets Favor on physical rolls when training applies. The two are independent.

A character can have both, one, or neither. Each handles a different aspect of capability.

Lacking a Required Competency

If a character attempts to use something they lack the Competency for, the GM may rule that:

- The item cannot be used effectively
- The action functions at Disfavor
- The action increases Difficulty
- The benefit is reduced
- The action carries narrative limitation

The specific penalty is system-defined. The principle is that a character without Competency in a thing cannot use that thing as if they did.

Gaining Competencies

Competencies are typically granted by:

- Origin (cultural baselines, like a starting language)
- Background (professional training, like Weapon or Tool Competencies from a soldier or smith Background)
- Archetype (class-driven Competencies, like Heavy Armor for a warrior Archetype)

Vector Marks may grant Competencies in some systems, but this is less common than Skill or Category gains.

Competencies are permanent once gained, like Skills.

Tuning the Feel

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

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Skill List Density

How many Skills a system defines shapes the entire feel of character creation.

Density	Effect	Good for
Few (5-8 Skills total, ~2-3 per Category)	Each Skill matters often, characters specialize quickly	Lightweight or one-shot systems
Default (10-15 Skills, ~3-5 per Category)	Balanced. Characters can be specialists or generalists	Most Frontiers-based systems
Many (20+ Skills, ~7+ per Category)	Granular character building, slow to learn	Detail-rich or long-campaign systems

Category Modifier Frequency

How often Category modifiers appear in the system shapes whether Categories feel like normal investments or rare achievements.

- **Generous** — Multiple Vector Mark options grant Category modifiers, Archetypes commonly include them. Players regularly accumulate +2 to +4 Category bonuses.
- **Default** — Vector Marks include Category modifier options, Archetypes occasionally grant them. Most characters have +1 to +2 in their primary Category by mid-game.
- **Scarce** — Category modifiers come almost exclusively from rare Abilities or Equipment. Most characters never accumulate more than +1.

The default produces a Category layer that feels earned but achievable.

Skill Application Tightness

How broadly the GM rules Skills apply determines how often Favor enters play.

- **Broad rulings** ("Athletics covers any physical force") — Favor activates often, specific Skills feel powerful but Categories feel less essential
- **Narrow rulings** ("Athletics is running, jumping, and climbing only") — Favor activates rarely, Categories carry more weight
- **Default** — Skills apply when the character's approach clearly draws on training. Most rolls in a Category get Favor only some of the time.

This is a tuning knob that lives with the GM rather than the designer, but designers can guide it through Skill descriptions and example use cases.

Competency Strictness

How harshly the system penalizes lacking Competency shapes how essential Competencies feel.

- **Strict** — Lacking Competency means the item cannot be used at all. Competencies are gates.
- **Default** — Lacking Competency means Disfavor or partial functionality. Characters can attempt without training but pay for it.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

- **Loose** — Lacking Competency carries only narrative consequence. Mechanically, anyone can use anything.

Edge Cases

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

Multiple Applicable Skills. If more than one of a character's Skills could apply to a roll, the player picks one. The chosen Skill grants Favor; the others do nothing for this roll. Skills do not stack.

Skill Without Category. A character with a Skill but no Category modifier still gains Favor when the Skill applies. Categories and Skills are independent layers.

Category Without Skill. A character with a Category modifier but no specific Skill in that Category gets the modifier on every roll within that Category. They do not gain Favor. The Category modifier alone is the benefit.

Multiple Sources of the Same Category Modifier. If a character has Category modifier from multiple sources (e.g., +1 from Archetype, +2 from Vector Marks), the modifiers stack toward the +5 cap. Once the cap is reached, additional sources do not increase the modifier.

Skill That Applies to Multiple Categories. A specific Skill always belongs to a single Category, defined by the system. If a roll could plausibly draw on multiple Categories, the GM picks which Category modifier applies. The Skill's Favor still applies regardless.

Skill Granting Favor on the Gradient Die. By default, the Favor from a Skill applies to the Resolution Die. If the player wishes to apply it to the Gradient Die instead, they must declare so before rolling. (See the Resolution System design doc for Favor application timing.)

Skill Without Competency. If a task requires a Competency and the character lacks it, having a Skill does not override the requirement. A character with the Athletics Skill but no Heavy Armor Competency cannot wear heavy armor without penalty, even though they're physically trained.

Competency Without Skill. A character with a Competency but no relevant Skill may use the item normally and rolls without penalty. They do not gain Favor on the use unless they also have an applicable Skill.

Creating New Skills Mid-Play. If a system allows custom Skill creation during play (e.g., from training arcs or narrative milestones):

- Define the Skill's Category clearly.
- Prevent overlap with existing Skills.
- Record the definition for future consistency.

Temporary Loss of Skills. Some narrative effects may temporarily suppress Skill access (curses, conditions, environmental factors). Suppressed Skills do not grant Favor while suppressed but are not lost. They return when the suppressing condition ends.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Variants

Variants modify the Skills system while preserving the Category-and-Skill structure.

Tiered Specific Skills Variant

What changes: Specific Skills become tiered (Trained, Expert, Master) instead of binary.

- **Trained** — Favor on the Resolution Die (as default)
- **Expert** — Favor on the Resolution Die, plus +1 to the roll
- **Master** — Favor on the Resolution Die, plus +2 to the roll, plus an additional Category modifier of +1

What stays: Category structure, Category modifier cap, Competency separation.

What shifts: Adds depth to specific Skill investment. Characters with deeply trained Skills feel mathematically distinct from those with broad shallow training.

Use when: Building a system that wants long-progression Skill mastery.

Skill List Lock Variant

What changes: The system designer locks a fixed Skill list. Players cannot create new Skills during play, even with GM approval.

What stays: All other Skill mechanics.

What shifts: Tighter system control, easier to balance Abilities that interact with specific Skills, less player creativity.

Use when: Building a competitive or tightly-balanced system where Skill selection should feel like a defined build choice.

Category-Only Variant

What changes: Remove specific Skills entirely. Characters only have Category modifiers.

What stays: Three-Category structure, +5 cap.

What shifts: Characters become broadly capable rather than specialized. Build differentiation moves entirely to Abilities, Archetype features, and Vector Marks.

Use when: Building a rules-light system where binary specific Skills feel like overhead.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

Specific-Skills-Only Variant

What changes: Remove the Category layer. Characters only have specific Skills, granting Favor when they apply.

What stays: Specific Skill rules, Skill list openness.

What shifts: Removes the breadth-of-training concept. Characters either have a Skill or they don't, with no in-between. Simpler to track, less layered.

Use when: Building a system where the Category layer adds complexity the system can't justify.

Granular Competencies Variant

What changes: Competencies become tiered (Familiar, Trained, Expert) rather than binary.

- **Familiar** — Can use without penalty
- **Trained** — Familiar, plus +1 to rolls using the equipment/system
- **Expert** — Trained, plus an additional Skill-like benefit defined by the system

What stays: Competencies are still about access first.

What shifts: Competencies become a deeper investment system. Adds Skill-like depth without using Skill slots.

Use when: Building a system where equipment and tool mastery is a significant build axis.

Alternatives

Alternatives replace the Skills system substantially. Adopting these requires recalibration of related systems.

Pure Attribute Alternative

Core change: Remove Skills entirely. Resolution rolls use only Attribute Modifier and situational bonuses.

Characters differentiate through Abilities, Archetype features, and Equipment.

What you gain: - Maximum simplicity. No Skill list, no Category structure. - Character creation is faster. - Abilities and Equipment carry full weight as identity layers.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

What you lose: - The training layer entirely. A trained character is mechanically identical to an untrained one with the same Attributes. - One of the engine's design hooks for differentiation.

Use when: Building a strongly attribute-driven or Ability-driven system where the training layer is overhead.

Numeric Proficiency Alternative

Core change: Replace Skills with a numeric proficiency bonus. Each character has Skill values that grow over time, applied as flat bonuses to relevant rolls.

What you gain: - Familiar to players coming from d20 traditions. - Progression feels mathematical and clear. - Easier to balance scaling.

What you lose: - The Favor mechanic for Skills. Skills become flat bonuses rather than dramatic surges. - The two-layer Category/Skill structure. Categories may become redundant.

Use when: Building a system that wants traditional d20 Skill scaling.

Descriptor-Based Alternative

Core change: Replace Skills with narrative descriptors or tags. When a descriptor meaningfully applies to a roll, the GM grants Favor, reduces Difficulty, or allows broader Narrative Edge.

Descriptors don't have fixed mechanical bonuses. They're permission and leverage tools.

What you gain: - Maximum narrative flexibility. - Reduced rigid skill lists. - Strong character identity through naming.

What you lose: - Mathematical predictability. - Tight balance. - Build clarity. Players have less mechanical signal about what their character is good at.

Use when: Building a strongly narrative-forward system where descriptors should feel character-defining.

A Note to System Designers

Skills in Frontiers do less than skills in most TTRPG engines. There is no Skill list. There are no Skill ranks. Each Skill does exactly one thing: grant Favor when it applies.

This is intentional. The Skill system is meant to be the least mechanically complex layer of character differentiation in the engine. Most of the texture lives elsewhere: in Abilities, in Archetypes, in Vector Marks, in Equipment.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.

A system that adopts Frontiers' defaults inherits a particular feel: - Two-layer training (Category modifier + specific Skill Favor) - Binary specific Skills, no tiers - Designer-defined Skill list per genre - Competencies as a separate access system - Category bonuses as build rewards, not baseline

This feel is deliberate and tuned.

Modifying the defaults with a Variant or an Alternative changes that feel. Adding tiered Skills, locking the Skill list, removing Categories, or replacing Skills with descriptors all produce systems that feel different 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.

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**.

© 2026 Jayke Paver · This product is licensed under the ORC License located at the Library of Congress at TX 9-307-067 and available online at various locations. All warranties are disclaimed as set forth therein.