

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

It is written for several readers at once:

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

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

There will be terminology here that is unfamiliar to casual players. Treat this documentation as advanced content, written for designers interested in building. It defines each term as that term arrives, so you can read straight through.

---

## What are Skills

---

Skills are how a game represents trained capability, as distinct from what a character is naturally suited to. Most systems draw some line between the two, and a Skill system is the machinery on the trained side of it.

To give a well-known example, Call of Cthulhu hands every investigator a long, itemized list: Library Use, Spot Hidden, Psychology, Locksmith, Drive Auto, and dozens more, each rated as a percentage the player rolls under. The list is the character sheet's center of gravity. What an investigator is comes through largely in what they are trained in and how deeply, and the granularity is the point in a game where knowing to check the newspaper archive is a real decision.

That answer sits at one extreme. Others go the opposite way, replacing the list with a handful of broad approaches, or dropping skills entirely and letting an attribute plus the fiction carry the weight. All of them are answering the same three questions:

1. **What counts as trained?** How finely capability is divided, and how many named things a character can be trained in.
2. **How much does training matter?** Whether it is a small nudge, a large bonus, or the primary number being rolled.
3. **Does training gate anything?** Whether lacking a Skill means doing it worse, or means being unable to try.

## Why it matters

A Skill system is the most visible place a game tells players what kind of competence it cares about.

The list is a statement of scope. A game with Locksmith, Forgery, and Cryptography as separate Skills is telling you it expects investigation to be granular enough that those distinctions matter. A game with a single Larceny Skill has decided that grain is finer than it needs. Neither is wrong, but the list is read as a promise: players will assume the things named are things the game intends them to do.

A lot of character identity ends up living there too, whether or not the designer intended it. Attributes tend to be few and shared, so two characters are often mechanically distinguished by what they know, not what they are. The more granular the Skill system, the more identity it carries, and the more time it takes at creation.

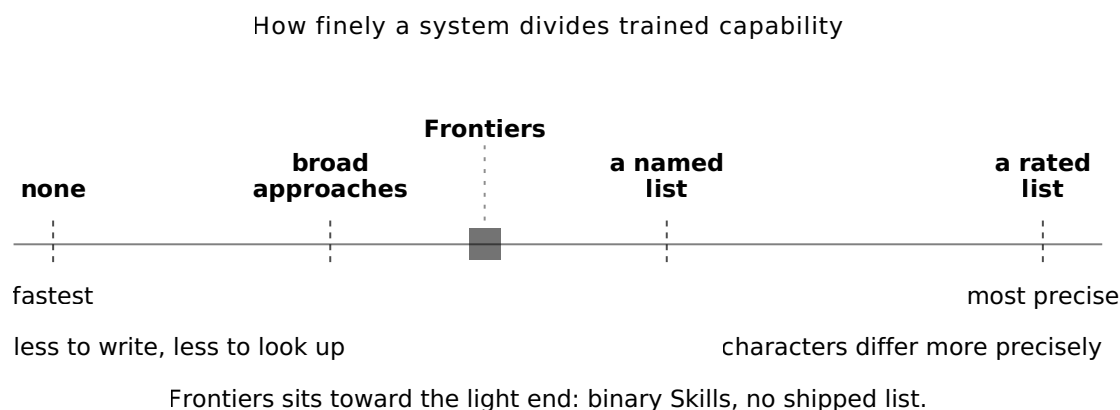
And it interacts directly with resolution. A Skill has to attach to the roll somehow, and how it attaches decides how much weight training carries against luck. A system where the Skill is the target number makes training the dominant factor. A system where a Skill grants a modest bonus makes it one input among several.

For the designer, the question underneath is:

How much of a character's competence should come from what they trained in, instead of what they are?

## The Approaches

Skill systems mostly differ along one axis: how much of a character's capability the Skill carries, and how finely it is divided. Everything else follows from that.



A few shapes worth knowing.

## The skill is the roll

The Skill is the number you roll against, and the attribute behind it is secondary or absent. Call of Cthulhu works this way, as does much of the percentile tradition.

An investigator's Locksmith might read 45. You roll percentile dice and come in at or under 45, and that is the entire resolution. There is no attribute to add and nothing to compute. Degrees of success come from the same number: rolling under half of it is a Hard success, under a fifth an Extreme one, so a single value carries both whether you succeeded and how well.

At the table this makes competence completely transparent. A player knows before they commit that they are a 45% locksmith, which turns "do I try this" into a real decision rather than a got a hopeful one. The cost is that the list must be comprehensive, because anything unlisted has no number at all, and comprehensive lists run long. It also nudges players toward their best numbers, since attempting something untrained is close to pointless.

## Skills as a graded modifier

The Skill is a rank added to a roll that an attribute also feeds. GURPS, Shadowrun, and Traveller all do versions of this, though they feel quite different in play.

**Traveller** rates Skills from 0 upward and adds them to 2d6 against a target, usually 8. What makes it distinctive is the gap at the bottom: having a Skill at level 0 means trained-but-unremarkable, while having no Skill at all applies a significant penalty. The interesting space is therefore between nothing and something, and a single level of training changes a character's odds more than any later level will.

**GURPS** ties each Skill level to a governing attribute and rolls 3d6 under the result. Because 3d6 is a steep bell, a point of Skill near the middle of the range is worth far more than a point at the edges, so the system rewards getting a Skill to competent and then broadening instead of pushing one number to extremes.

**Shadowrun** converts Skill into dice. Attribute plus Skill sets how many d6 you throw, and you count successes. Training therefore widens the range of possible outcomes rather than shifting a single number, and a large pool feels reliable in a way a large bonus does not, because it is many small chances, not one big one.

Grading in general gives you a long runway and legible progression, and charges you arithmetic plus a certain inevitability at the top, since once numbers are large enough, specialists stop meaningfully failing at their speciality.

## Skills with fictional weight attached

The Skill is not only a number but a claim about what the character has done. Burning Wheel is the clearest case.

Skills arrive through lifepaths, the character's actual history, so a sheet reads more like a biography than an inventory. More unusually, Skills improve by being tested under pressure rather than by spending points: a character logs the difficulty of the tests they attempt, and advancement comes

from accumulating the right mix of routine, difficult, and challenging attempts. Failing while trying something hard is progress.

The feel at the table is that players go looking for trouble in their speciality, because a comfortable campaign is one where nobody improves. It binds mechanics to fiction tightly, and asks a lot in return: it is heavy to learn, and it works poorly for tables that want Skills to be a simple competence rating.

### **Broad approaches instead of a list**

A small set of wide categories replaces the itemized list. Lancer uses skill triggers, not a granular list, and many modern indie systems do something similar.

Rather than naming a discipline, a trigger names a way of approaching things, and applies whenever that approach fits. A character picks a handful and improves them in steps, adding the trigger's bonus to a `d20` when it applies. Two characters with the same nominal expertise can therefore hold quite different triggers, because what is being rated is method instead of subject.

Creation is fast, nothing falls outside the categories, and no player has to search a list. The cost is resolution: with few enough categories, different characters produce similar sheets, and identity has to come from somewhere else, usually abilities or equipment.

### **No skills at all**

Some games decline the question entirely. Cairn has three attributes and no Skill list whatsoever. Competence lives in equipment, background, and what the player describes, and the GM decides whether a roll is even needed.

That relocates the system rather than removing it. Because there is no Skill to point at, the argument for why a character should succeed is made in the fiction, and having the right tool matters more than having the right number. Play tends to run fast and lethal, and preparation matters more than optimization.

It suits short, dangerous, improvisational games, and suits long campaigns poorly, since there is little to grow into and few numbers to move.

Running under all of these is a trade between **resolution and speed**. The more finely you divide capability, the more precisely characters can differ, and the more the system asks of everyone using it: more to write, more to choose from, more to look up. There is no correct point on that line, only the one that matches how much of your game is about what characters know.

---

# What Frontiers Does

---

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

Frontiers' Skill system is deliberately the thinnest layer in the engine:

- Skills are **binary**. You have one or you do not.
- A Skill grants **Favor**, which is the whole of its mechanical effect.
- **One Skill** applies per roll.
- Every Skill belongs to **Body**, **Mind**, or **Voice**, and those categories carry no modifier of their own.
- The engine ships **no Skill list**.

Full rules live in the Rulesets; what follows is why so little is here.

## Binary, because simple is easier to build on

Most systems grade training, and grading has a known endpoint: once the numbers are large enough, a specialist stops meaningfully failing at their speciality. Frontiers avoids that by never starting down the ladder. You are trained or you are not, untrained being the default and trained being the bonus.

The main reason is comprehension. Nobody really wants to hold ninety graded proficiencies in their head, each keyed to a different Attribute, before their character can do anything. Skills should be readable at a glance, and binary is as readable as it gets.

The second reason is what happens downstream. Shipping a specific, complicated Skill system means anyone whose game wants something else has to tear it out first. A designer building percentile-style granular expertise and a designer building a broad approach-based system would both begin by deleting whatever I shipped. Starting lean means you add as gaps appear, and adding is the easier direction to move in.

It helps that the resolution layer already carries the weight Skills would otherwise be asked to carry. The reasoning there is its own document, but in short: the curve, the Gradient, and the option to take a flat result all shape how reliable competence feels, so Skills do not need to be a second solution to the same problem.

## Favor, because the dice are the point

A Skill could have been a flat bonus. It grants Favor instead, and there are a few reasons.

The plain one is that people like rolling dice, and Frontiers is a system that wants dice on the table. Adding a number is arithmetic, where adding a die gives the table something to watch.

The structural one is that every Frontiers Ruleset rolls two dice, and both of them matter. Favor is a choice about which die to improve, and because the Gradient is read off one of those dice, that choice is between succeeding and succeeding well. Expertise buys either confidence or impact,

declared before the dice answer. A flat bonus could not offer that, and in a system without the Gradient the argument for Favor would be much weaker.

There is a third reason, and it holds in the roll-over Rulesets. A character can take their Score instead of rolling, succeeding at anything within their competence without touching the dice. Favor is therefore a choice, never a requirement: a trained character does not have to gamble to benefit from their training. They can take the sure thing when the sure thing is enough, and reach for the dice when it is not. The d100 has no passive value to take, so this argument does not apply there, and its Trivial tier covers the same ground from the other direction. The full reasoning lives in The Resolution System documentation, and it is worth reading alongside this one.

Favor is a modest effect by design, so training stays a real edge without ever becoming the dominant factor.

## **One Skill per roll**

A soft cap, and an obvious one once you have seen it abused. Without it, a player writes Weaponry and Blades on the same sheet and collects both on every swing, which is bookkeeping with extra steps.

One Skill per roll keeps Skills broad and honest. If several could apply, the player picks. Favor from other sources still stacks normally, so the cap constrains Skills specifically instead of capping how favorable a situation can get.

## **Categories as handles, not modifiers**

Body, Mind, and Voice do nothing on their own, which is the point of them rather than an oversight.

Because Skills are lightweight, the useful thing the engine can offer is a way to address them. Three stable categories give designers somewhere to hook effects: Favor on all Voice rolls for a scene, ignore Disfavor on Body rolls while this is worn, choose a Mind Skill and hold Favor on it permanently. None of that is writable if every Skill is an island.

They help at the table too. Three buckets give a player a frame for what kind of character they are building, and give a designer a frame for which Skills their game actually needs. And because they carry no mechanics of their own, they are among the easiest things in the engine to remove if you do not want them.

## **No Skill list, on purpose**

The engine names categories and ships nothing to put in them.

There is no list of Skills that suits every game that will ever be built on Frontiers, and committing to one would mean either a list too generic to be useful or a list that encodes a genre. Official Frontiers systems and Starter Sets define their own, because what counts as trained expertise depends entirely on what the game is about. A survival game and a courtly intrigue game do not want the same list, and neither should have to work around one.

It is the same boundary the engine draws elsewhere, with the structure as chassis and the content belonging to the system built on it.

## **Skills scale with ● Acuity**

Skill count is not a flat allotment. It rises with a character's ● Acuity Rank, so a player who wants a broadly trained character can build one by investing in it.

Flat counts make characters more alike. Tying the number to an Attribute turns breadth of training into a real build direction instead of a fixed line on everyone's sheet. The rule runs both ways: a negative ● Acuity Rank removes Skills, describing a character who never absorbed much training, though the floor never drops below one. Everyone brings something.

The choice of ● Acuity specifically solves a problem most systems have. Intelligence-style Attributes are notoriously hard to justify: physical Attributes have obvious mechanical work to do, while mental ones often end up governing a narrow slice of rolls and little else. Pathfinder ties languages to Intelligence for much the same reason. Giving ● Acuity control over Skill count means it has a use every character can feel, including martial ones who would otherwise have no reason to invest, and it makes the "do I take this or that" decision at creation a genuine trade instead of an obvious one.

## **The GM decides when a Skill applies**

When it is arguable whether a Skill fits, the recommendation is to allow it. Favor is a modest benefit, and a player engaging with what their character trained in is worth more than a strict ruling.

More broadly, the GM has final say. Many recent systems have narrowed that role considerably; Frontiers keeps it, and the Skill system is one of the places that shows.

## **Is this the best solution for Skills?**

Skills are the most stripped-back part of the engine, and they are stripped back on purpose.

It does put the onus on the designer. A Frontiers system will want to say more about Skills than the engine does, and the Starter Sets and official settings all do. But Skills are one of the few subsystems a system can genuinely define on its own terms, and the engine would rather hand that over than enforce an ideology about what expertise looks like.

One consequence worth naming plainly: because only one Skill applies per roll, a specialist's tenth Skill is worth less than their third. That is working as intended rather than a flaw to patch. It is the same as owning fifteen weapons, where the fifteenth adds less than the third did, because what additional Skills buy is range. A system where each new Skill made you stronger at what you already did would be a different and, in my view, worse game.

There is a good chance you disagree with some of these answers, which is a reasonable place to land. The final section below sets out variant and alternative rules you can use in their place, along with what each one changes and what it will cost you elsewhere.

---

## Variants

---

Variants keep the philosophy intact. Skills stay binary, stay light, and stay the layer a system fills in for itself; what changes is how they attach. The groups below are dials, not packages, and they are meant to be mixed.

### Tune the effect

A Skill grants Favor, and only one applies per roll. Both are levers.

#### GRANT A FLAT BONUS INSTEAD

A Skill adds a fixed number to the roll, not a die. Training becomes arithmetic, which some tables prefer, and it makes expertise fully predictable. It also removes the confidence-or-impact choice, since a bonus cannot be placed on the Gradient, and it puts numeric modifiers back into a system built to avoid them.

#### LOWER THE TARGET INSTEAD

A Skill reduces the DT instead of improving the roll. Mathematically similar to a bonus, but it reads as the task being easier for a trained character rather than the character being better, which suits some fictions.

#### ALLOW MORE THAN ONE SKILL PER ROLL

Lift the cap and let every applicable Skill contribute. Less dangerous than it sounds, since Favor still caps at 3, so the practical ceiling holds. What you lose is the pressure that keeps Skills broad: players will write narrower, overlapping Skills specifically to collect several at once.

### Tune the count

Skills scale with ● Acuity Rank, with a floor of one.

#### USE A FLAT ALLOTMENT

Everyone gets the same number regardless of Attributes. Simpler, and it removes ● Acuity's build incentive, which will make that Attribute noticeably less attractive unless you give it work elsewhere.

#### CHANGE THE RATE

One Skill per 2 points of ● Acuity Rank is a default. Faster scaling makes trained characters dramatically broad; slower makes expertise rare and precious.

#### REMOVE THE FLOOR

Let a low enough ● Acuity leave a character with no Skills at all. Harsh, and appropriate for grittier games where being untrained is a real character state, not a slightly worse one.

## Tune the categories

Body, Mind, and Voice carry no mechanics and exist so effects have something to hook onto.

### ADD OR RENAME THEM

A fourth category costs nothing structurally. Genre-specific sets work well: a Tech category for science fiction, a Spirit category where the supernatural is a discipline of its own.

### GIVE THE CATEGORIES MODIFIERS

Let each category carry a bonus, penalty, or standing effect of its own instead of being a pure label. I tried this and did not keep it. It makes categories a second thing to balance, and it pushes players to pick Skills for their category, not for their character, but it is a reasonable direction if you want categories to be a real build decision.

### CUT THEM ENTIRELY

One of the cheapest removals in the engine, since nothing depends on them mechanically. You lose the ability to write effects that address a group of Skills at once, which is worth checking against your ability and equipment designs before committing.

---

## Alternatives

Alternatives replace the philosophy rather than tune it. Skills are the least entangled subsystem in the engine, so these are unusually safe compared with the equivalent swaps elsewhere: the main things to check afterwards are Abilities and Equipment, since both may reference Skills or their categories.

### Graded Skills

Rate Skills instead of holding them. This is the mainstream approach and the most common thing designers will want.

The cleanest fit is to grade in Favor, since the engine already caps Favor at 3:

| Rank | Name    | Effect  |
|------|---------|---------|
| 1    | Trained | 1 Favor |
| 2    | Expert  | 2 Favor |
| 3    | Master  | 3 Favor |

A Skill still applies once per roll, and the Rank sets how much Favor it grants. Nothing else in the engine needs to change, because the stacking cap already handles the ceiling: a Master with a

favorable situation is at the same maximum as anyone else, so the ranks compete with circumstance instead of compounding past it.

What it costs is the thing binary was avoiding. A Master rolling three dice and keeping the best rarely fails at their speciality, so reliability now comes from two sources at once, the curve and the Skill. It also makes advancement heavier, since Skill Ranks become another track to spend on and balance. Worth it if progression through expertise is what your game is about.

### **Skill as the target number**

Make the Skill the number being rolled against, as the percentile tradition does, with Attributes secondary or absent.

This is the deepest change on the list. Skills and Attributes would be competing for the same job, so one of them has to give ground, and everything derived from Attributes, Defenses, + HP, 4 Energy, would need rebuilding on a different foundation. In the d100 form this is close to writing a different game, though a coherent one. Only worth attempting if trained expertise is genuinely the center of your system.

### **Approach-based Skills**

Replace the list of disciplines with a small set of methods: Force, Finesse, Insight, Charm, or whatever fits your fiction. A Skill still grants Favor and is still binary; what changes is that it names a way of doing things rather than a subject.

This sits comfortably on the engine, since nothing about binary-plus-Favor cares whether the named thing is a discipline or a method. Creation gets faster and nothing falls outside the categories. The tradeoff is resolution: with few enough approaches, characters converge, and identity has to come from Abilities and Equipment instead. The Body, Mind, and Voice categories may become redundant under this model, since the approaches are already doing that grouping work.

### **No Skills**

Cut the subsystem. Favor comes from circumstance, Abilities, and Equipment only, and trained expertise is expressed through the fiction and what a character carries.

This is the least destructive removal available in the engine. Nothing depends on Skills existing, and ● Acuity simply loses one of its uses, so you may want to give it something else to do. Play becomes faster and more improvisational, and leans harder on the GM, since every question about whether a character's background applies is answered in the moment rather than looked up. Suits short, lethal, or heavily narrative games; suits long campaigns less well, since one of the main things characters grow into is gone.

Whichever you choose, Skills are the safest subsystem in Frontiers to rebuild. If you are going to make one substantial change to the engine, this is the place to start.