

Research and Development

The Software Engineering Handbook

A Judgment and Decision-Making Approach

James C. Davis

Chapter 10 of the full book · Edition 1 · 2026

Research and Development

Premise. *Research and development invests scarce engineering effort in uncertain opportunities for asymmetric benefit.*

DEFINITION • Research and development

Research and development (R&D) is systematic work undertaken to increase knowledge and develop new applications of knowledge (OECD 2015). The knowledge sought may concern what is possible, how something works, or how an existing limitation can be overcome.

R&D begins with a constraint: there are more worthwhile problems than there is time to pursue them. A researcher may see opportunities to improve performance, remove a limitation, support a new use, exploit a new capability, or understand a phenomenon more deeply. An industrial R&D group faces the same abundance of possibilities. Pursuing one means not pursuing another.

The unusual feature of R&D is that the outcomes can be asymmetric. An investigation may fail and produce little beyond what was learned, while a successful new idea can enable capabilities, improvements, or understanding far beyond the effort invested. Most research projects can fail without doing much harm; a successful one can change what becomes possible.

The preceding chapters followed a purpose into a system: engineers decide what to promise (Requirements), make those promises precise (Specification), organize a realization (Architecture) and resolve the choices inside its parts (Design), and decide what evidence justifies delivery (Validation). R&D asks where engineering effort should go when we do not yet know what is possible. Before asking how to solve a problem, we must decide whether the problem deserves the effort. The answer depends partly on what success could accomplish and partly on how likely success appears. A modest improvement with an obvious solution may be nearly certain. A larger advance may require an idea we do not yet have.

There is no formula that resolves this tradeoff. R&D exists precisely because important facts are unknown. But four questions organize the judgment.

KEY-IDEA • The four questions of R&D

- **What is worth pursuing?** Decide which uncertain opportunity deserves scarce engineering effort: seek asymmetric upside under opportunity cost.
- **Does it require a new idea?** Determine whether the limitation in what exists is essential to the approach or merely a property of its current embodiment.
- **How large is the advance?** Judge what would be different if the work succeeds: an increment, an advance, or occasionally a transformation.
- **What would establish it?** Determine what evidence would support the claimed advance.

The first two concern choosing and perceiving which opportunity deserves your attention and whether it actually requires a new idea. The latter two are judgments about understanding what you have actually accomplished. The chapter takes them in turn.

What is worth pursuing?

A useful R&D problem offers the possibility of an outcome worth the effort required to investigate it. You must weigh two factors:

- **Potential impact.** If this works, how large is the upside? An incremental improvement may be useful; an advance may change what can be done at all.
- **Likelihood of success.** How plausible is it that the investigation will produce something useful?

Cost, timing, available expertise, competition, and dependencies also matter. The point is not to assign each factor a score and select the largest number. It is to compare opportunities explicitly rather than allowing the easiest visible project to consume the time available.

R&D is easily distorted by its visible outputs. A patent, paper, prototype, or product can become the objective rather than evidence that useful knowledge was created. Organizations face the same temptation when they subordinate research to immediate product needs. The work becomes easier to justify in the short term, but less able to discover ideas that would change what the organization can do.

The object of R&D is knowledge. Pursue an important question whose answer is not yet known and whose answer could matter. Papers and patents are valuable ways to communicate, protect, and establish what was learned. Products may eventually put that knowledge to work. They are outcomes of a successful R&D agenda, not substitutes for one.

Here, hard does not mean technically difficult for its own sake. It means the most potentially transformative — for the discipline, the organization, or for you; any and all.

My advice is to choose the hardest important problem you can plausibly make progress on. You will sometimes fail. That is part of accepting the asymmetry. You are giving up some probability of producing a result in exchange for the possibility of producing a much larger one. You will also find that difficult problems change what you are capable of doing: they force you to learn more deeply, acquire techniques you would not otherwise need, and encounter questions you could not have seen from the easier problem. When the hard problem succeeds, it has more room to matter. Over a research career, I would rather see you take serious swings than become very efficient at producing results you already knew how to obtain.

This advice deliberately does not maximize the probability of producing a result. Research can tolerate failed attempts in a way ordinary delivery work often cannot. The relevant question is not simply *How likely am I to produce something?* It is *What might become possible if this works, and what will I learn by seriously attempting it?*

Does it require a new idea?

Finding an important problem does not establish that a new approach is needed. Existing systems are imperfect in innumerable ways. They may be slow, support the wrong formats,

expose awkward interfaces, lack useful features, or have implementations that were never optimized for the situation now under consideration.

Any of these limitations can motivate engineering work. They do not necessarily motivate R&D. The useful distinction is between essence and accident.

DEFINITION • Essential and accidental properties

An **essential** property follows from the underlying concept: an embodiment that lacked the property would no longer faithfully realize the same idea. An **accidental** property belongs to a particular embodiment but need not belong to another embodiment of the same idea. The distinction between essential and accidental properties goes back at least to Aristotle; here we apply it to engineered systems and approaches.

Consider a parser. Parsers accept input and determine whether it conforms to some language or format; that role is essential to the concept. A particular parser might accept JSON but not XML, use one parsing algorithm rather than another, or expose a particular API. Those properties may simply characterize that implementation. Another parser can differ in all of them while remaining recognizably a parser.

This distinction matters because a claimed advance often attacks the embodiment when it needs to attack the idea. Suppose an existing system lacks a capability your proposed system provides. If the existing system could acquire that capability through an ordinary extension while retaining the same conceptual approach, you have identified an engineering task, not necessarily an advance. A long list of deficiencies does not change that fact if every deficiency can be repaired without changing the idea.

The stronger question is therefore: *Could a faithful embodiment of the existing idea avoid this limitation?*

If yes, the limitation is probably accidental. Improve the embodiment.

If no — if the limitation follows from an assumption, abstraction, algorithm, representation, or other defining property of the approach — then eliminating it requires a conceptual change. That is where the opportunity for an advance begins.

Can this be solved with elbow grease?

There is a practical way to develop intuition for this distinction. When you believe you have found a limitation worth researching, ask whether it can be solved with **elbow grease**.

Imagine giving the existing system and the limitation to two competent junior engineers for six months. They can write code, optimize components, replace libraries, support additional formats, collect more data, improve deployment, and clean up awkward parts of the implementation. If you expect sustained engineering effort to make the limitation disappear while leaving the underlying idea intact, be suspicious of the claimed advance. You may have found substantial work without finding a substantial advance.

If the limitation survives that thought experiment, look deeper. Perhaps every faithful realization of the approach inherits it. Removing the limitation might require changing an assumption about the problem, introducing a different representation, replacing the governing algorithm, moving a boundary, or otherwise altering something integral to the idea. Now you may have found the opening for R&D.

Elbow grease is a diagnostic, not a definition. An accidental limitation can be expensive to remove, especially in a large or poorly engineered system. Conversely, a deep conceptual change can occasionally be easy to implement once somebody sees it. The definitive question remains whether eliminating the limitation requires changing the idea.

Generative AI also changes the elbow-grease test because it changes the price of engineering effort (Software Engineering). “Two junior engineers for six months” once represented a substantial amount of implementation capacity. Increasingly, some of that work can be accomplished by a much smaller team working with capable agents in days or weeks. Yesterday’s impressive implementation effort can become tomorrow’s routine engineering task.

This raises rather than lowers the importance of identifying the conceptual advance. As implementation becomes cheaper, difficulty that comes primarily from producing implementation becomes weaker evidence of an advance. The underlying test has not changed: can additional engineering effort remove the limitation without changing the idea? Commodity intelligence simply makes accidental limitations cheaper to eliminate and therefore harder to defend as the basis for R&D.

How large is the advance?

R&D can produce improvements of very different magnitude. An **increment** improves an existing approach while preserving its essential idea. An **advance** changes something essential and thereby makes something meaningfully different possible. Occasionally, an advance is **transformative**: it changes the space of problems people can solve or the way a field approaches them.

The vocabulary follows from the essential-accidental distinction. Work that removes accidental limitations produces increments: a faster implementation, a broader format, a cleaner interface — valuable, often necessary, and conceptually conservative. Work that removes an essential limitation produces an advance, because every faithful embodiment of the old idea inherited the limitation and the new idea does not. Transformation is rarer still, and mostly recognized in retrospect; few projects should be judged by whether they achieve it.

This vocabulary also disciplines the potential-impact judgment made when choosing what to pursue. When you claim an opportunity has a large upside, say which magnitude you mean. An increment can be worth pursuing when the approach it improves is important enough. But the asymmetric upside that justifies a hard, uncertain investigation usually lives at the advance level: the work does not merely improve what exists, it changes what can be done.

What would establish the advance?

Once an opportunity appears worth pursuing and requires a genuine advance, the remaining question is what would establish that advance. Four pieces must hold together:

- **Problem.** Identify the important limitation or opportunity the work addresses.

- **Essential limitation.** Show why additional engineering of the existing approach cannot adequately resolve it.
- **New idea.** State what changes in the underlying concept rather than merely in its embodiment.
- **Evidence.** Demonstrate that the new idea produces the claimed improvement and establish the conditions under which the claim holds.

Weak R&D often leaves one of these links unsupported. An important problem does not establish that a new idea was required. A new idea does not establish that it works. A successful implementation does not establish that the improvement came from the claimed idea. The evidence must support the particular advance being claimed.

Measurement for decision-making

R&D is undertaken because something consequential is not known. Measurement is how an investigation finds out what it has learned, and whether the next unit of effort is worth spending.

Consider a team investigating a new program-analysis technique because the existing technique cannot scale to the programs they care about. The first measurement belongs before the new idea, not after it. How do analysis time and memory grow as programs get larger? At what size does the existing tool stop being usable, and what gives way first: the constraint solver, a memory ceiling, or a timeout somebody configured years ago? The answer can go two ways, and both are useful. It may locate the limitation in the approach itself, which is the opening the chapter has been looking for. It may instead expose an accidental limitation that ordinary engineering can remove, in which case the team has found substantial work rather than an advance. This is the elbow-grease question (“**Can this be solved with elbow grease?**”) asked empirically rather than imagined.

If a new idea is required, measurement then establishes the size of the advance (“**How large is the advance?**”). The new technique analyzes programs ten times larger, or reduces an overnight run to minutes, or makes analyzable a class of systems that previously was not. Those observations matter only relative to the limitation that motivated the work. They also depend on what they are compared against: a claimed advance can disappear once the existing approach receives the same engineering attention the new one did, so the baseline should be the best existing approach rather than its most convenient embodiment.

Experiments and prototypes earn their cost when they are built around a question whose answer changes the next decision. A quick prototype may show that the idea cannot meet its memory budget, and the work should stop. A small experiment may show that the idea works only under an assumption that rarely holds, and the work should be redirected toward the cases where the assumption is plausible. Negative results are useful for this reason: they resolve an uncertainty before more effort accumulates behind it.

Activity is not the property we care about. Lines of code written, experiments run, papers read, and prototypes built measure effort, not learning. Before undertaking an R&D activity, an engineer should be able to complete the sentence:

If we observe _____, we will have reason to continue, redirect, or stop, because _____.

The blank need not be filled by a number. A counterexample can refute an assumption. A working construction can establish that something is possible at all. A benchmark can quantify an improvement against a baseline. What matters is that the observation bears on an uncertainty that would change what the team does next, and that the claim it supports is the one the work is actually making (“**What would establish the advance?**”).

R&D is a bet on learning

R&D begins before we know whether the proposed idea will work. That uncertainty is not a defect in the process; it is why the work is R&D.

The judgment is therefore not merely whether an idea sounds promising. You are deciding which uncertainty deserves your scarce time. Potential impact matters because success should justify the investment. Likelihood of success matters because some bets are implausible even when their imagined payoff is enormous. The need for a new idea matters because an apparent R&D problem that can be eliminated through ordinary engineering may not justify inventing a new approach at all.

Then the work itself produces information. A failed prototype may reveal that an assumed opportunity does not exist. An experiment may show that the supposed limitation of prior work was accidental. An attempted solution may expose a deeper problem than the one that motivated it. A failed research direction can therefore be valuable — but only if you recognize what it taught you and allow that evidence to change what you work on next.

You will spend a career making these bets with incomplete information. Choose them deliberately. Ask what could matter, whether the obstacle is real, whether it requires a new idea, and what evidence would tell you that you were right.

And when you have a choice between an easy problem whose answer you can already see and a hard important problem that might defeat you, I recommend the hard one.

Summary

Research and development invests scarce engineering effort in uncertain opportunities for asymmetric benefit. There are more worthwhile problems than time to pursue them, and the outcomes are asymmetric: an investigation may fail and produce little beyond what was learned, while a successful new idea can return far more than the effort invested. Four questions organize the judgment.

What is worth pursuing? weighs potential impact against likelihood of success, comparing opportunities explicitly instead of letting the easiest visible project consume the time — and, for a researcher, argues for the hardest important problem you can plausibly make progress on. *Does it require a new idea?* separates essential limitations, which follow from the underlying concept, from accidental limitations of a particular embodiment. The elbow-grease thought experiment supplies the diagnostic, and generative AI sharpens it: as implementation becomes cheaper, accidental limitations become weaker evidence of an advance. *How large is the advance?* calibrates the upside: an increment preserves the essential idea, an advance changes something essential, and a transformative advance changes what a field can attempt at all. *What would establish it?* requires four pieces to

hold together — problem, essential limitation, new idea, and evidence that supports the particular advance being claimed. Throughout, the object of R&D is knowledge: papers, patents, and products are outcomes of a successful agenda, not substitutes for one.

The bet is ultimately on learning. Even a failed direction pays, if you recognize what it taught you and let that evidence change what you pursue next.

READ FURTHER

Robertson Ishii, Teresa, and Philip Atkins. “Essential Vs. Accidental Properties.” Accessed September 25, 2026. <https://plato.stanford.edu/entries/essential-accidental/>. The ancient distinction between what belongs to a thing as the kind of thing it is and what merely belongs to one realization of it.

Hamming, Richard W. “You and Your Research.” 1986. <https://www.cs.virginia.edu/~robins/YouAndYourResearch.html>. A classic talk about choosing important research problems. Some of the advice has aged; the central question has not.

OECD. *Frascati Manual 2015: Guidelines for Collecting and Reporting Data on Research and Experimental Development*. OECD Publishing, 2015. <https://doi.org/10.1787/9789264239012-en>. The formal definition of R&D, from the manual used to decide what counts as research across countries and industries.

Bush, Vannevar. *Science, The Endless Frontier: A Report to the President*. United States Government Printing Office, 1945. <https://www.nsf.gov/od/lpa/nsf50/vbush1945.htm>. The report that argued for funding research whose object is knowledge and trusting applications to follow.