



Pain Points Won't Tell You What to Build

Product discovery for complex B2B software



"Who here has built something a customer asked for, and then they didn't use it?"

The Claim

A **pain point** is what a customer experiences when an existing solution fits their situation badly. A **feature request** is a solution someone has imagined. Neither one describes the underlying customer problem, so they cannot tell you what to build. They are where discovery starts, not where it ends.

A **customer problem** is a concrete situation: what someone is trying to achieve, why, in what circumstances beyond their control, and by what criteria they judge the result. **Understand and describe the situation exactly, and the right solution mostly follows by reasoning.**

"The hard part is no longer writing code but finding the problem to solve and understanding it better than the customer alone can."

Mikko Kärkkäinen, CEO of RELEX, Helsingin Sanomat, 9 April 2026

Problem and Solution Are Different Kinds of Thing

A customer problem	A solution
Exists as facts in reality	Comes from imagination
Discovered by research	Must be invented
Changes slowly ; thus a solid basis for long-term strategy	Changes fast with technology and fashion
Vendor cannot change it	Entirely under vendor's control
Independent of any solution	Its fit depends on the problem ; poor fit produces pain points and feature requests
Not directly visible while a solution is being used	Plainly visible when a solution is used

Exactly like a **foot and a shoe**. A blister tells you the shoe doesn't fit the foot. It doesn't tell you the shape of the foot, and you can't design a good shoe based on blisters. Shoe shops have measured feet with the Brannock device since 1927. Software companies still ship a solution and then ask about blisters.



The Backlog Test

Take the top five items on your roadmap. Give each one a few minutes.

1. **Describe the concrete situation** in which the item arises. Who is doing what, why, in what circumstances beyond their control, judged by what criteria.
2. **Cross out** everything referring to today's tools, how they are used, and the processes, roles and hand-offs of a particular organisation.
3. **Keep only what reality imposes.**

If you can't write the description, you don't yet know whether that item is the right thing to build.

Measuring a Design Before You Build It

A good solution requires the customer to take **only the actions genuinely necessary** to reach the best available result. Since the situation is concrete, you can count them: how many actions does a proposed solution require in that situation at a minimum? Fewer necessary actions means better fit. You don't need users or a built solution.

I watched one experienced user complete a 30-minute workflow with existing retail software. It took around 300 steps. Only three or four were genuinely necessary for a human to take in that situation. She only asked for a shortcut button that would have saved two of them. Most of the features she was complaining about should not have existed.

The optimal solution is often **smaller and simpler** than what is already on the market, which makes it cheaper to build, once you know what it is.

This works where customer value comes from accomplishing something (instrumental value), which covers nearly all B2B software. It works much less well where the value is in the experience itself (intrinsic value).

Three Starting Points, One Move

What it looked like	What the situation turned out to be
A feature request: build a profitability calculation so we can delist unprofitable products	The buyer is deciding a whole category, not one product. Poor profitability has about a dozen possible causes, often fixable without delisting. The product had to support the diagnosis, not output a number.
A pain point: the space planner comes back three weeks later to say the new range won't fit on the shelves	Buyer and space planner are measured on opposing things. Estimate the space and cost impact while the buyer decides, and nothing has to go wrong.
A scope assumption: assortment planning is a separate module, because that's how the market sells it	Assortment and pricing are largely overlapping problems. If they are split into separate products, the tool won't fit the buyer's job. Unusable tools are what every failed competitor had shipped.

Same move every time: **discover, understand, and describe the situation.**

The Case Study

RELEX Solutions entered assortment management, a market where every competitor had tried and failed and where retailers didn't use the software they had bought.

Discovery: 6 months, one person. 40 sessions across 6 retailers, with buyers, macro and micro space planners, category managers, pricing specialists, supply chain people and data analysts. Plus process and data-flow analysis, industry literature, scientific papers and data science.

Build: a team of 3–4, 18 months.

5

enterprise customers in year one

20+

pipeline globally

IDC Leader

on the first release

"The insights and results from Antti's product discovery helped us find the right approach. The results have been surprisingly good given how little resources we have spent on it."

Tommi Ylinen, Chief Product Officer, RELEX Solutions



Antti Latva-Koivisto

Helps B2B SaaS product leaders work out what customers actually need, rather than what they ask for. 25+ years in B2B software at Comptel, Nokia, and as a consultant to RELEX and others.

deductivediscovery.com — newsletter on B2B SaaS product discovery

linkedin.com/in/anttilk · anttilk@antti.lk · +358 40 578 2747



Read the
newsletter