



Case study

Project Rubicon

A five-engineer team, a microinverter built into the cable, and a working prototype in under six months.

Engagement series.

How lateralworks invested in and ran a corporate innovation program — organizing five engineers to prove a radical idea fast, define the right product with customers, and reach a validated proof of concept against a schedule half the industry norm.

Prepared by

lateralworks
Investor and program manager

Sector

Solar power electronics
Module-level power electronics

Online

lateralworks.com
Fast-time-to-market practice

Table of contents

Project Rubicon

Overview	03
The engagement at a glance	04
01 — The market, and what was at stake	05
02 — An inverter inside the cable	08
03 — Voice of the customer	12
04 — Removing every constraint	16
05 — Five engineers, one schedule	19
06 — Proof, validated in the field	21
07 — Shrinking to fit the cable	24
08 — What the prototype proved	28
09 — Why the program stopped	31
10 — The practices behind the result	33
References	36

Core thesis. The product a team ships is decided long before the engineering is done — by the conditions leadership sets, the customers the team listens to, and the schedule it runs against. Project Rubicon put five engineers in the right conditions, pointed them at a customer-defined target, and got a working proof of concept in under six months. The parent company chose not to commercialize it. The method still did exactly what it is built to do.

Overview

Overview

A large industrial wire manufacturer had a small power-electronics division and a radical idea: build the solar inverter into the cable itself. The company asked lateralworks to invest in the idea and to run the program. Our job was to organize a five-engineer team, prove the concept, and define a product the market would actually buy — fast enough to matter in a market that was moving underneath everyone.

The result was a working proof of concept in under six months, reached with about a third of the planned budget and in roughly half the time a program like this normally takes. The team breadboarded the core idea within three weeks of conceiving it, validated it on the bench as a simulated string, and then ran it outdoors into the grid on a working solar array. On cost per watt, the program's model put the design below the leading microinverters of the day. Its topology was also simpler and ran cooler, which is where reliability comes from in this class of product.

The parent company later paused, and ultimately declined to continue. That was a portfolio decision, not a verdict on the technology or the team. What the engagement demonstrates is the method: how a host organization, a small team, a customer-defined product, and a disciplined schedule combine to produce the right product at the right time — the outcome lateralworks calls fast-time-to-market, or FTTM.

This case study walks through the market the idea was aimed at, the customer research that shaped it, the creative-thinking challenge that produced the core invention, the way the program was organized and scheduled, the proof that emerged from the lab and the field, and the specific lateralworks practices that made a six-month result possible. It closes with an honest account of why the program stopped, and why the method behind it still counts as a success. The practices referenced throughout are the ones lateralworks has developed since 1991 and documents at lateralworks.com.

What the numbers say. Under six months to a working prototype. Three weeks from idea to breadboard. About one-third of the planned budget. Roughly half the schedule of a typical program. A cost-per-watt target below the category leaders. One idea, complete on the first attempt — which is rare, and is what the right conditions are supposed to produce.

Summary

The engagement at a glance

<p>Role</p> <p>Investor and program manager</p> <p>lateralworks co-funded the proof of concept and ran the program end to end — assembling the team, setting the conditions, and managing the schedule.</p>	<p>Scope</p> <p>Prove a radical idea, define the right product</p> <p>Take a distributed, cable-integrated microinverter concept from a challenge workshop to a validated proof of concept — while defining the product with customers in parallel.</p>
<p>The hard part</p> <p>A third of the cost, small enough for a cable</p> <p>Beat the category leaders on cost per watt, fit the electronics inside a solar cable, and prove reliability — all at once, on a compressed schedule and a constrained budget.</p>	<p>What we delivered</p> <p>A working prototype in under six months</p> <p>A breadboard proof in three weeks, a bench-validated string, a field test into the grid, an industrial design, and a cost model below competitive products at launch — at roughly a third of the planned budget.</p>

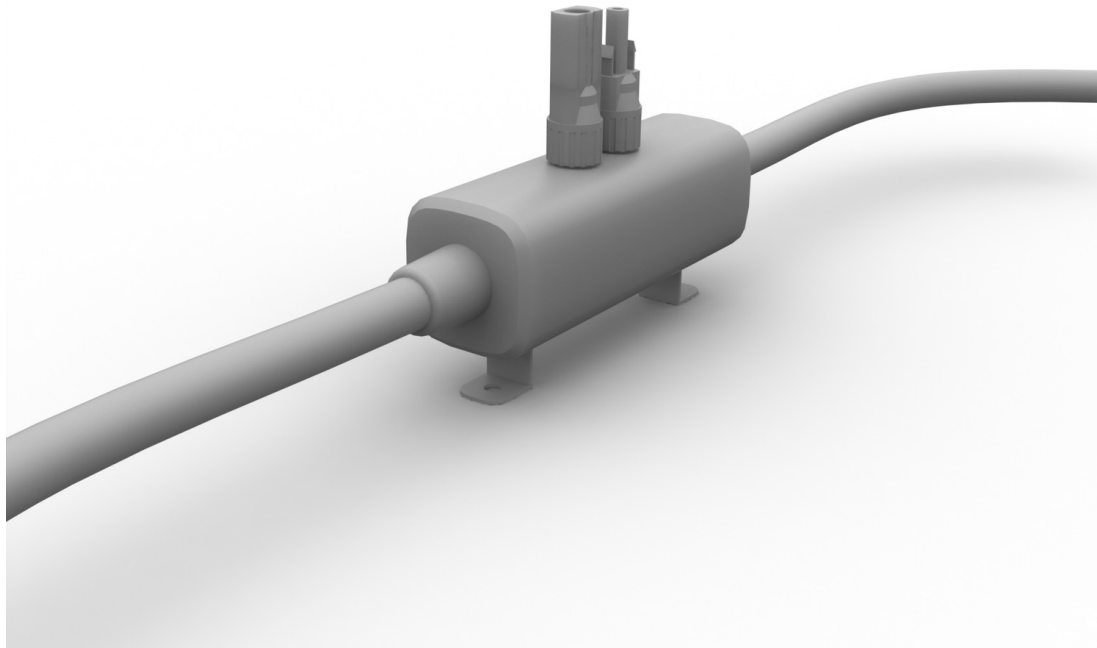


Figure 1. The concept in one image: a distributed microinverter node built onto the trunk cable itself, terminating in standard connectors. The parent was a wire company — so the cable became the product.

01

Section one

The market, and what was at stake

In 2016 the solar microinverter was a maturing product with an unsettled market. Module-level power electronics — microinverters and DC optimizers, or MLPE — had gone from novelty to default on American rooftops in less than a decade. Enphase Energy had effectively created the category when it shipped the first commercial microinverter in 2008 [1]. By 2014, some form of MLPE was built into more than half of all U.S. residential solar installations [2].

Section one

A category that was winning and a leader that was hurting

The tailwinds were real. A residential rooftop is a hard place for a single string inverter: multiple roof planes, shifting shade from chimneys and trees, and small system sizes — typically five to seven kilowatts. Putting a small converter on every module lets each one track its own maximum power point, so one shaded panel no longer drags down the whole string, and it hands the installer per-panel monitoring and a simpler design. Code accelerated the shift: the 2014 National Electrical Code introduced the rapid-shutdown rule of section 690.12, and the 2017 revision tightened it to an array-boundary limit that, in practice, pushed installers toward module-level hardware [3, 4].

Two companies had pulled away from the field. By the third quarter of 2015, Enphase and SolarEdge — the DC-optimizer challenger that had gone public that March — together held roughly 80 percent of the U.S. residential inverter market [5]. SolarEdge was climbing; by 2016 its revenue had passed Enphase's [6]. The category had two clear leaders, and a long tail of everyone else.

The leader was under pressure

Enphase, the name most customers knew, was struggling where it hurts. It finished 2015 with revenue of about \$357 million but a net loss of \$22 million, after a profitable 2014 [7, 8]. It shipped a record 706 megawatts that year — more than three million microinverters — and passed its ten-millionth unit in the fourth quarter [7]. Yet gross margin was collapsing under a price war: from about 30 percent for the full year to the high teens by mid-2016 [7, 8]. The company had cut microinverter prices roughly 19 percent in November 2015, and its chief executive was publicly committing to cut device cost in half and chase parity with string inverters at around ten cents a watt [8, 9].

That is the opening a challenger looks for: a category that customers had accepted, a dominant brand whose economics were strained, and a cost target — well under twenty cents a watt for an integrated device — that no incumbent had yet reached. A wire company with a distributed, cable-integrated design and a genuinely lower bill of materials could enter exactly there.

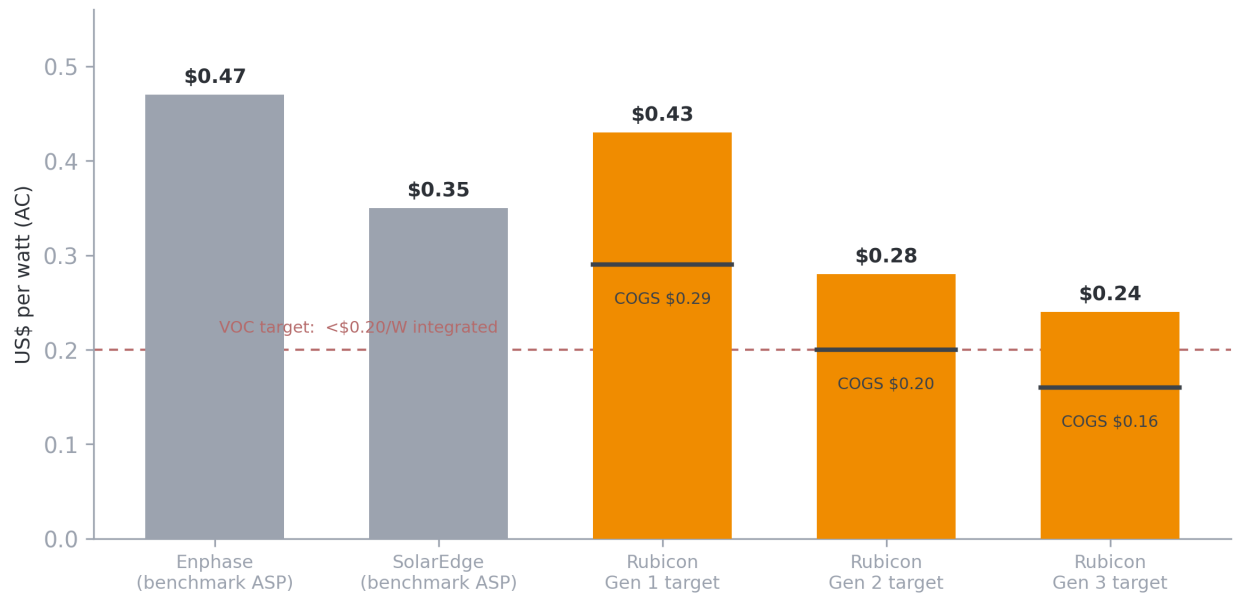


Figure 2. Where the program aimed. The two bars on the left are the leading products the team benchmarked against; the three on the right are Project Rubicon's internal cost-per-watt targets across three generations. The dashed line is the integrated-device price customers said the category needed to reach. Benchmark and target figures are lateralworks and program internal estimates from 2015–2016; competitor list prices varied by channel and volume.

The reliability question

Underneath the cost war ran a quieter argument about reliability, and it cut both ways. Putting power electronics on the roof, behind each panel, means exposing them to heat and weather for the life of the array — and replacing a failed unit means pulling a module. Electrolytic capacitors, the components most sensitive to heat, were the classic knock against microinverters [10]. Enphase answered with a 25-year warranty designed to match the panel's life and carried tens of millions of dollars of warranty liability on its balance sheet to back it [7, 11]. Installers had lived the difference: several told trade researchers in 2015 that early microinverters had failed in the field at rates that pushed them toward optimizers, where the inverting happens off the roof [2]. A design that ran cooler and simpler had a real story to tell here — and the Rubicon topology was built to tell it.

02

Section two

An inverter inside the cable

The insight that started the program was almost embarrassingly literal. The sponsor was a wire company. If the inverter could be made small enough, it did not need its own box bolted to the racking — it could live on the cable that was already going to be there. The company would be selling its own core product, with the electronics along for the ride.

Section two

No central inverter, only building blocks

The conventional architectures each carried a penalty. A string inverter is cheap but centralizes the conversion, so shade anywhere hurts everywhere. A microinverter fixes that by putting a full inverter behind every panel, but a full inverter is expensive and hot. A DC optimizer splits the difference and still needs a central inverter somewhere in the system.

Project Rubicon took a different route. Instead of one big inverter, it distributed the inverting function across many small AC building blocks strung in series along the cable. Each node did its own maximum-power-point tracking and contributed a slice of the conversion; the nodes added up along the string; and all that remained at the end was a small grid interface box near the service panel, sized so that eight to twelve nodes made a branch within a 20-amp limit [12]. There was no central inverter to buy, and no full inverter behind each panel to pay for and cool.

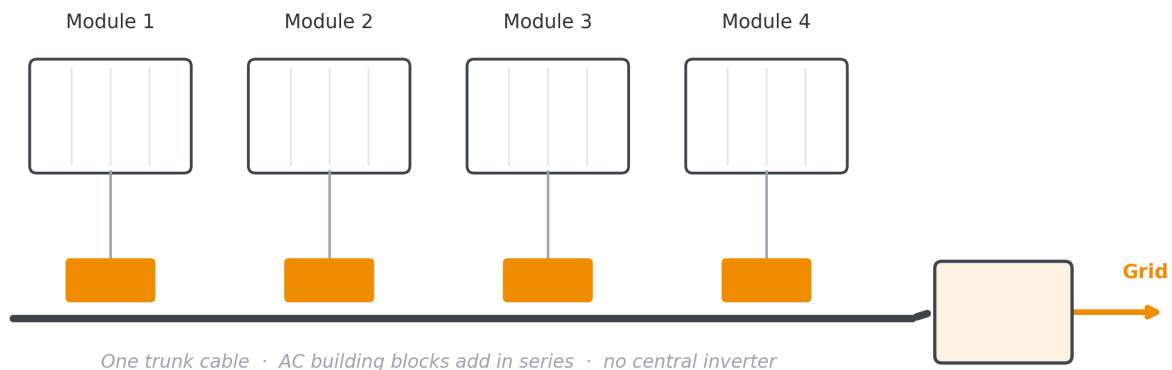


Figure 3. The distributed architecture. Small AC-optimizer nodes on the trunk cable each perform module-level tracking and partial conversion; they add in series along the string; a single small grid interface box completes the tie to the grid. No central inverter, and no full inverter behind each module.

Why the cable changes the economics

Building the electronics into the cable did three things at once. It removed the separate enclosure, the separate mounting hardware, and the separate installation step — the cable, the optimizer, and the inverter became one item to buy and one thing to install [12]. It gave the distributed design somewhere to shed heat along its length rather than in a single hot box. And it turned the sponsor's existing manufacturing and channel strength — making and selling cable at volume — into the product's cost advantage rather than a side business.

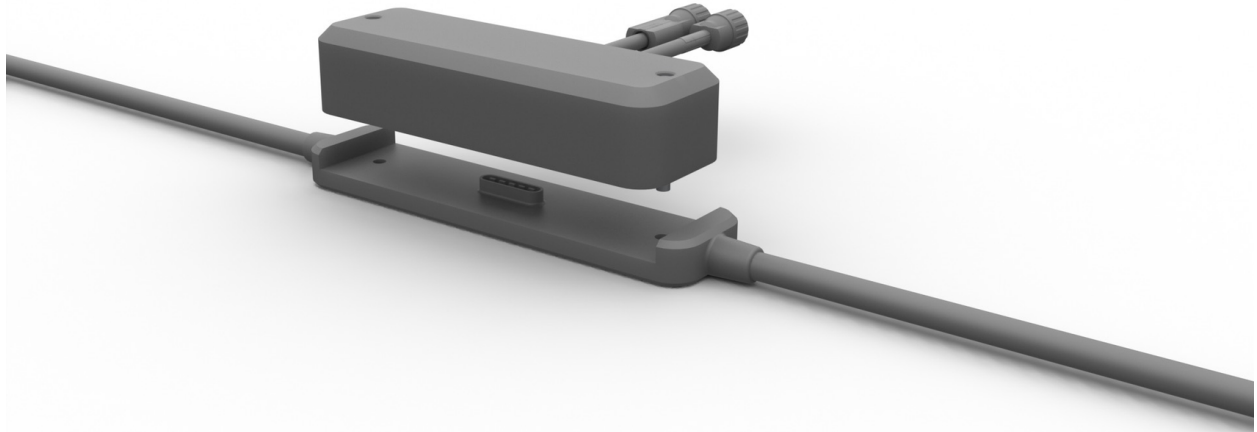


Figure 4. A single node inline on the cable. The unit tracks its module and contributes its share of the conversion, then passes AC down the trunk to the next node in the string.

On paper the datasheet claim was blunt: the level of integration and the cost per watt had no direct competitor. It offered the shading immunity and per-module tracking of a microinverter or optimizer, at closer to the cost of a plain string inverter, with the simplicity of a system that needed only a small grid interface box rather than a central inverter [12]. The task for the next several months was to prove that the paper claim was real — and to make sure it was the product customers actually wanted.

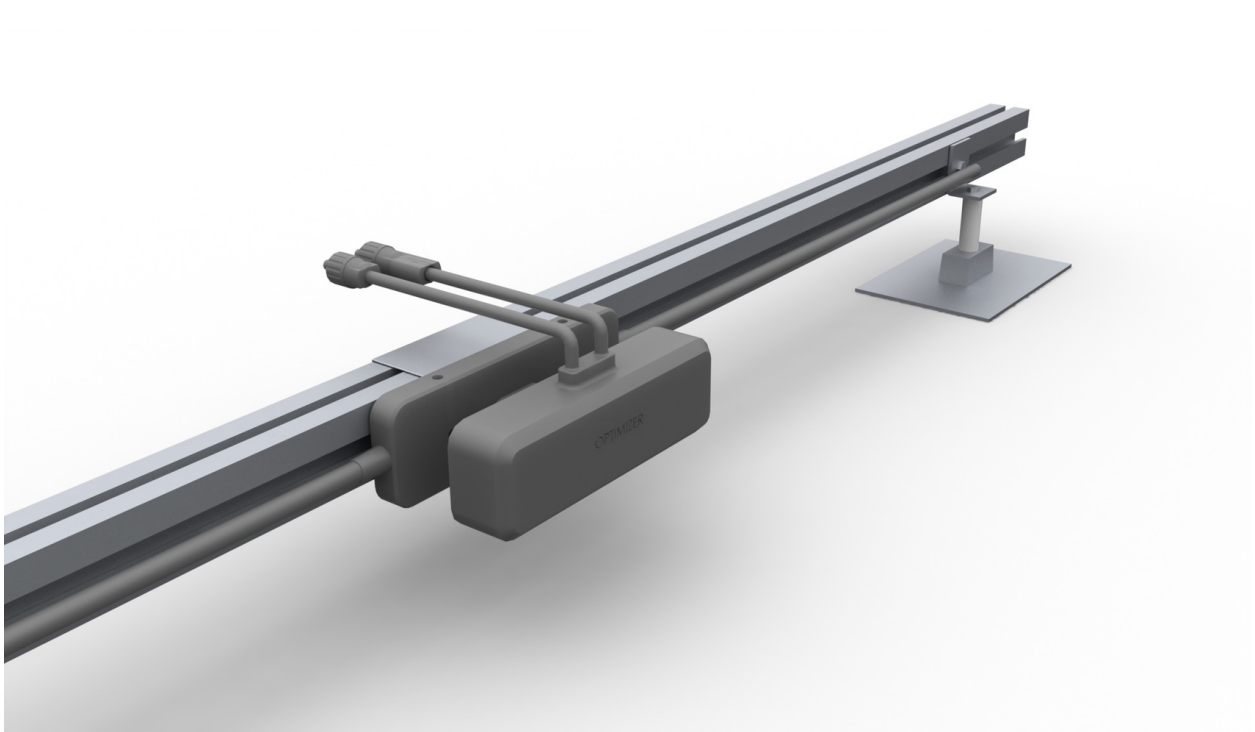


Figure 5. The node shown mounted along a rail with the module cabling — the installed picture the industrial design had to serve: no separate box, no extra bracket, nothing to ground on its own.

03

Section three

Voice of the customer

A radical architecture is worth nothing if it solves the wrong problem. Before and during the build, lateralworks and the team ran a deliberate voice-of-the-customer effort — talking to the people who specify, finance, and install these systems, and letting what they said reshape the product.

Section three

What the buyers actually told us they needed

The developers and module makers were consistent and specific. In a 2013 conversation that set the tone, a senior strategist at one of the largest solar developers laid out his buying criteria in order: cost, reliability, bankability — the confidence that the warranty company will still exist in five years to stand behind a twenty-year system. His verdict on the integrated microinverter was a hard threshold: it had to be built into the module, it had to prove twenty-year reliability, and it had to come in under twenty cents a watt. Above that price, in his words, it would never happen.

A module maker's U.S. general manager, interviewed the same year, drew the same line from the other side of the table. He saw the market moving toward inverters integrated into the junction box once the reliability risk was understood and the price reached the low twenties of cents per watt, and he weighed cost at the system level — inverter, cable, and monitoring together. The residential system, four to seven kilowatts, was the sweet spot everyone named.

What the customer set as the target. Integrated with the module. Under twenty cents a watt for the integrated device. Twenty-year reliability, proven. Bankable. Simple enough to be a consumer product. Those four requirements — not an internal specification — became the fixed end state the team designed against.

Credibility opened the doors

The team could ask hard questions because it had standing to ask them. One member had co-founded a leading inverter company and advised many others; he brought both a sharp read on the breakthrough requirements and direct relationships with the segment's most demanding buyers — the large residential installers and developers and the established power-electronics firms. That access turned VOC from a survey into a working collaboration: customers helped refine the product definition and the market timing rather than simply rating a concept.

The listening had a cost the team accepted on purpose. Customer input kept changing the design during development — the normal, healthy pattern of a fast program, and uncomfortable for engineers who would rather freeze and finish. The discipline lateralworks holds here is simple and hard: the wrong product at the right time fails just as surely as the right product at the wrong time, so the team keeps talking to customers and keeps absorbing change, right up until the point where converging matters more than improving.

The clock had a price

To make those trade-offs concretely rather than by argument, the program ran a cost-of-delay model built on the business case — the method Reinertsen and Smith formalized and that lateralworks has used for years to put a number on time [13, 14]. For this product the model put the cost of delay at about \$173,000 a day: roughly \$5.2 million in profit for every month the product reached the market late. A number that size changes behavior. It tells an engineer that paying to save a week is almost always right, and it lets the whole team weigh a new feature, a cost reduction, or a schedule slip on the same scale — the business case — instead of by opinion.

The idea

Ignore the constraints

Innovation comes when thinkers ignore the constraints — which is exactly backwards from how development normally begins.

lateralworks program assessment
Project Rubicon, 2016

04

Section four **Removing every constraint**

The core invention did not come from a specification. It came from a room deliberately set up to break one. In September 2015 lateralworks opened the program with a structured creative-thinking challenge — a proven technique for getting a team to suspend the assumptions it does not know it is making.

Section four

The workshop that produced the idea

The method is counter-intuitive, and that is the point. Normal development starts with constraints and tries to solve within them. The challenge workshop does the opposite: it states the end goal in the most demanding terms — here, a device a third the cost of the competition and small enough to fit inside a cable — and then removes every assumed constraint about how to get there. We asked the engineers to list every assumption behind their earlier attempt at an inverter, and then to challenge each one and free-think the alternatives.

One engineer on the team, released from the conventional way of thinking about inverters, arrived at the idea that carried the whole program: a half-sine-wave switching scheme and a distributed topology that spread the conversion across many small nodes instead of concentrating it in one. Because the performance targets were radical, they demanded an equally radical concept — and that is exactly what the right conditions and the right person produced.

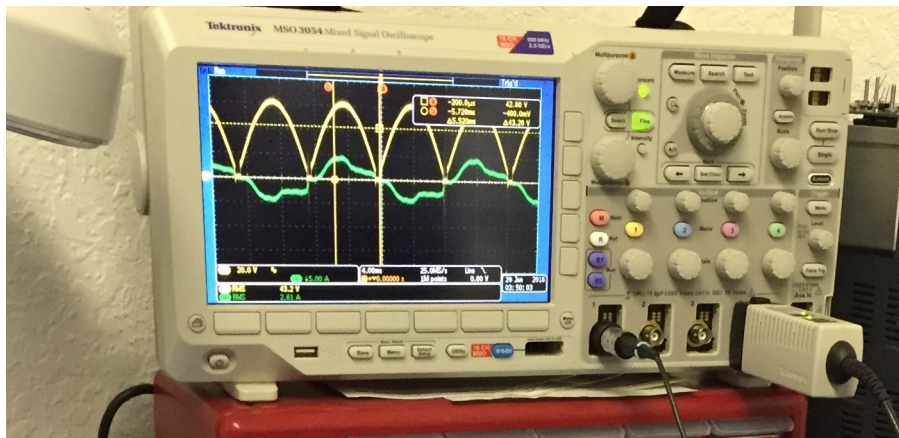


Figure 6. The half-sine-wave signature on the bench oscilloscope — the switching behavior at the heart of the distributed design. Seeing the waveform hold on the scope was the first hard evidence the concept was real.

Three weeks from idea to breadboard

The team did not wait to see whether the idea would survive review. By October — less than three weeks after conceiving it — they had breadboarded the concept and shown that it worked. It met the performance targets, it was cost competitive, and it could hit the size target. A team arriving at an idea that complete on the first attempt is rare; it is the outcome the challenge method is designed to produce, and it happened here because the team had total freedom to create and no constraints to design around.

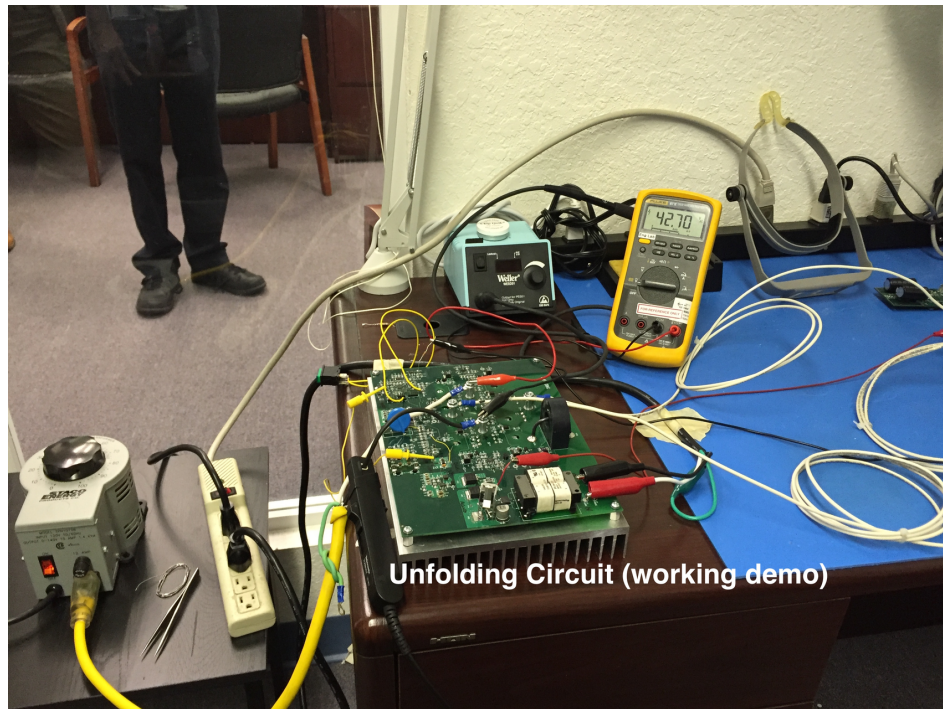


Figure 7. The working breadboard. A benchtop supply, a hand-wired gate driver board, and a multimeter — the unglamorous hardware behind a three-week proof that the half-sine, distributed concept converted power as intended.

Everything after that first three weeks — roughly the next several months — was a different kind of work: refining the idea into a product, keeping it from diverging, and getting it ready for field test and customer validation. The invention was fast. Turning it into the right product, without losing the cost and size that made it worth building, was the long part.

05

Section five

Five engineers, one schedule

A good idea is necessary and not sufficient. The reason Rubicon moved at the speed it did is that the program was organized — the team, the host conditions, and the schedule — to let the idea travel without friction. This is the part of the story that transfers to any program, in any industry.

Section five

A small team, isolated, and fully provisioned

The core team was small — five engineers — and, unusually, intact: they had worked together for years and had many products behind them. lateralworks immediately separated them from the parent’s day-to-day operations and co-located them, relieved them of all other duties, and let them think about exactly one thing: how to make a very low-cost, very small inverter. Normal corporate reporting and overhead were removed. The team’s only job was the product and the technical problems in front of it.

Around that core we built a team of external specialists rather than hiring a large internal one — mechanical and industrial design, contract manufacturing and supply chain, industry and technical advisors. The default was to outsource, which saved time and brought in more diverse expertise than any five people could hold. Critically, the contractors were treated as full members of the team, not as vendors: they planned with everyone else and owned the outcome with everyone else.

One integrated schedule, used to drive — not to report

The whole program ran on a single integrated schedule that every member helped build, internal and external alike. It was refreshed every week. The team used the schedule’s trend to see where it was slipping and to create urgency before the fact, not after. It always knew the gap between when a result was wanted and when it could realistically arrive — a real plan, never a happy one built for someone else to read.

That last distinction matters. The schedule existed for the team to manage itself, not to feed a management reporting cycle. Reporting to the host happened only when budget or schedule crossed an agreed threshold. The weekly planning refresh doubled as an automatic status report to anyone outside the team, at no extra cost to the people doing the work — so the reporting overhead that slows most programs simply was not there.

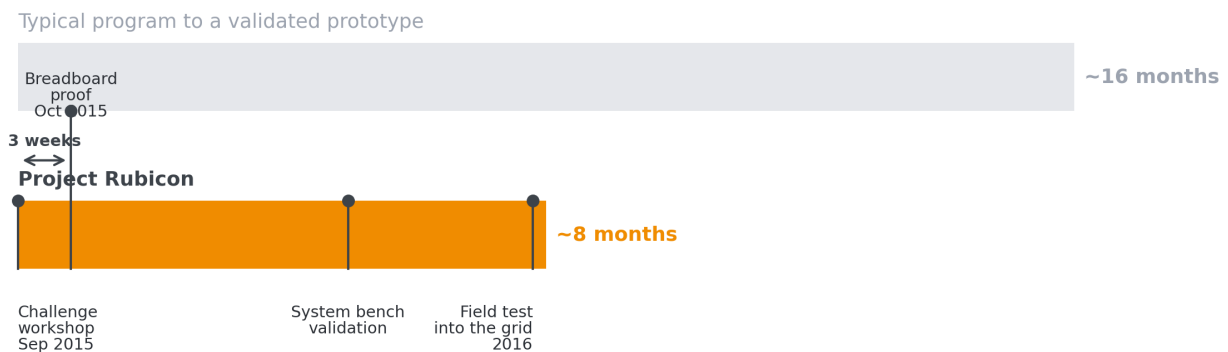


Figure 8. The schedule that resulted. Against a typical ~16-month path to a validated prototype, Rubicon reached a working proof of concept in under six months and field validation at roughly eight — from a challenge workshop, to a breadboard three weeks later, to a bench-simulated string, to a field test into the grid.

None of these moves is exotic. Isolate and focus the team, provision it fully, reward challenge, integrate the outsiders, and run one honest schedule to drive the work — they are ordinary practices, applied together and without exception. Section ten sets them out in full.

06

Section six

Proof, validated in the field

A concept that works on a single breadboard is a long way from one that works as a system, outdoors, tied to the grid. The team closed that distance in steps, each one raising the stakes and each one passed.

Section six

From the breadboard to the grid

The first step was to prove the distributed idea as a distributed system. On the bench, the team wired several optimizer boards together to simulate the start of a real string — nodes adding in series, exactly as they would on a roof — and confirmed that the architecture behaved as intended when the building blocks were stacked. This is where a distributed topology either holds together or falls apart, and it held.



Figure 9. The system on the bench: multiple optimizer nodes wired as a simulated string, instrumented and driven, feeding the full chain that a real array would present. Proving the nodes add up in series was the milestone that turned a circuit into an architecture.

The final step was to take it outside. The team ran the system on a working solar array — real modules, real sun, real grid connection — and watched it perform the way the model and the bench had promised. Testing a power-electronics prototype in the field, in daylight, into a live grid, is the moment a program stops being a lab exercise. Rubicon reached it in months, not years.



Figure 10. Field testing on a solar array — instrumenting the prototype outdoors, under load, into the grid. The step that separates a promising circuit from a product candidate.

What field validation established. The distributed nodes tracked and converted as designed; they added correctly in series along the cable; and the system delivered into the grid from a real array. The proof of concept was no longer a claim on a datasheet — it was a working system, validated where it would have to live.

07

Section seven

Shrinking to fit the cable

Proving the circuit was only half the product. The other half was physical: the electronics had to survive on a rooftop for decades, shed their own heat, and still be small enough to belong on a cable. This is where the program's discipline about the fixed end state was tested hardest.

Section seven

Keeping it small when physics pushed back

The predictable tension appeared as soon as thermal reality met the size target. To manage heat, the node wanted to grow. When it did — when the enclosure crept up in size to solve the thermal problem — the team was sent back to the challenge mindset and asked to make it smaller again, and removable. They did both, and held the cost target while doing it. Neither cost nor size was traded away to make a problem easier; the end state did not move.

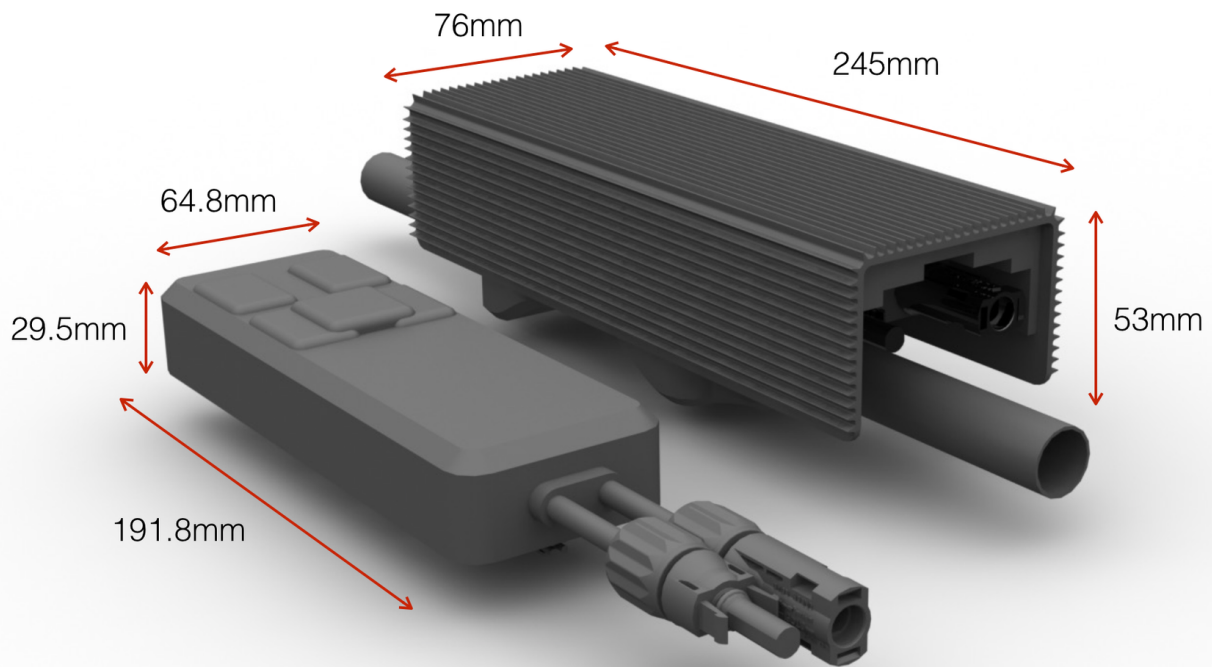


Figure 11. The volumetric study. As the design matured, every millimeter was argued against the thermal and connector requirements. The rule was constant: solve the physics without surrendering the size or the cost that made the product worth building.

Two ideas kept the package honest. A drop-in cable grommet let the node be assembled onto standard cable rather than demanding a bespoke molding on the critical path — a change that reduced risk, shortened the schedule, and preserved serviceability. And the industrial design deliberately broke from the category's houseful of anonymous gray boxes: the goal was a device that looked engineered, not improvised, because a product a customer trusts on their roof for twenty-five years has to look the part.



Figure 12. The grid interface box. Because the architecture pushed the inverting out to the distributed nodes, the only central hardware left was this small, simple unit near the service panel — one per branch, not one per system.

The mechanical work also surfaced the program's clearest lesson about outside help. The mechanical partner was one of the best engineering firms in Silicon Valley, but a service-based contract rewards billable hours, not breakthroughs — and on the hardest, most inventive part of the package, that misalignment showed. lateralworks' conclusion was blunt and reusable: when significant innovation is asked of a third party, the incentive has to be aligned with cash and equity, not a time-and-materials rate. Novel work needs skin in the game.

The result

Right product, fast

**A working prototype,
at a third of the
budget, in about half
the time — defined by
the customer, not by
the org chart.**

lateralworks program assessment
Project Rubicon, 2016

08

Section eight

What the prototype proved

Set against what it set out to do, the program delivered. It is worth being precise about what "delivered" means here, because the honest accounting is more persuasive than a rounded-up one.

Section eight

What the prototype actually proved

Start with the schedule and the money. From the challenge workshop, the team reached a working proof of concept in under six months, and a field-validated system at roughly eight — against a typical program norm of about sixteen. It did so on roughly a third of the planned budget; the project was, in plain terms, underfunded by about two-thirds, yet it never behaved as though it were, because lack of resources was not permitted as an excuse and the host supplied what the team asked for when it asked.

Take cost first, the criterion the customer named first. The program's model put the first-generation design meaningfully under the category leaders, with a roadmap that drove cost per watt down further across later generations toward the sub-twenty-cent integrated-device price customers said the market required. The competitive benchmark the team worked against put the leaders well above that line. The cable integration and the distributed topology, not a spreadsheet assumption, were what made the number reachable.

Reliability decides bankability, and here the argument was structural. The distributed design ran cooler and used a simpler topology than a full microinverter behind every panel, and heat and complexity are the two things that shorten the life of rooftop electronics [10]. Field testing had to continue to turn that argument into qualified twenty-year data; the program never claimed otherwise. But the design started from the right side of the physics.

The scorecard. Under six months to a working prototype, three weeks to the first breadboard, about a third of the planned budget, roughly half the typical schedule, a cost-per-watt target below the category leaders, and a topology built to run cooler and simpler. One idea, complete on the first attempt, then held to its cost and size through every trade-off.

What remained ahead was the work every hardware program faces after proof of concept: converging and freezing the design, standing up manufacturing and supply chain — always the longest pole — and completing the qualification and certification that turn a validated prototype into a shippable product. Those were known risks with known owners. The program had earned the right to take them on.

09

Section nine

Why the program stopped

The program did not fail on the bench or in the field. It stopped in the boardroom. The parent company paused the project and ultimately chose not to carry it into commercialization. An honest case study has to sit with that, because the reasons are as instructive as the successes.

Section nine

A portfolio decision, not a technical one

The decision to pause came at close to the worst possible moment — as the design needed to converge and the manufacturing runway needed to be started. lateralworks flagged the cost of it plainly at the time: slowing a program and restarting it typically adds six to eight months, so every month of pause tends to cost roughly two months at the end of a schedule that was already a quarter behind. Deferring the contract-manufacturing setup compounded it, because supply chain is the longest lead item in any high-volume hardware program. And the thinnest funding fell exactly where thin funding does the most damage — at the transition from proof of concept into verification, validation, and ramp.

There were softer lessons too. A large, established company is optimized for its core business, and a genuinely new venture usually needs to be stood up as its own entity, with its own board and its own win-win incentives, to survive the narrow windows a new market allows. Rapid technology development rarely thrives inside a parent tuned for something else. And, as section seven noted, a service-based contract with a key innovation partner left incentives misaligned on the hardest part of the work. None of these is a criticism of the engineering; all of them are reasons a strong technical result can still be set down.

Why it still counts

Judge the engagement against what lateralworks was there to do, and it succeeded. The method was asked to prove a radical idea quickly, to define the right product with the people who would buy it, and to fold what they said back into a disciplined schedule. It did all three: an idea complete on the first attempt, a product shaped by real customer requirements rather than internal assumption, and a validated proof of concept in a fraction of the usual time and budget. A parent's portfolio choice is a different question from whether the work was good. The work was good, and the way it was produced is repeatable.

10

Section ten

The practices behind the result

Everything in this case study rests on a body of practice lateralworks has developed since its first study of fast development teams in 1991, and documents at lateralworks.com. The practices are not a philosophy; they are a checklist that separates fast, successful programs from ordinary ones. Rubicon applied them, close to in full.

Section ten

The practices behind the result

lateralworks groups the practices into four conditions that all have to be in play at once to get fast-time-to-market — the right product at the right time. They are the host, the team, the product, and the time. Miss any one and speed leaks away; hold all four and a six-month proof of concept stops being remarkable and starts being expected.

Host	Team	Product	Time
The conditions leadership sets	Who does the work	What gets built	How the schedule is used
<ul style="list-style-type: none"> › Isolate and co-locate the team › One strategic focus, never deviate › Provision fully, remove excuses › Reward challenge, not compliance 	<ul style="list-style-type: none"> › A small, intact team with a track record › External SMEs as full members › No hierarchy, no reporting overhead › Everyone plans; everyone owns it 	<ul style="list-style-type: none"> › Fix the end state: cost, size, reliability › Iterate continuously on the how › Fail fast, recover faster › Listen to the customer, be willing to change 	<ul style="list-style-type: none"> › Plan realistically, know the gap › One integrated schedule › Weekly trend drives urgency › The schedule self-manages the team

Figure 13. The four FTTM conditions and the practices Rubicon applied under each. The full framework and its history are set out at lateralworks.com.

Host — the conditions leadership sets

The host isolated and co-located the team, held it to a single strategic focus that never changed, provisioned it as though resources were never the constraint, and rewarded challenge rather than treating it as insubordination. The cable-integration goal set on day one was the same goal at field test. When the design drifted, the answer was to challenge the drift, not to relax the target.

Team — who does the work

A small, intact team with a shared track record; external specialists carried as full members rather than vendors; no hierarchy and no reporting overhead; and a participatory plan every member owned. Fewer people with more skills beat more people with fewer skills, and everyone knowing the gap beat anyone managing the appearance of progress.

Product — what gets built

A fixed end state — cost, size, reliability — with continuous iteration on how to reach it; fast failure and faster recovery; and a standing willingness to change the design when the customer said so. The end state never moved; almost everything else did, on purpose.

Time — how the schedule is used

Realistic planning that always exposed the gap to target; one integrated schedule spanning internal and external work; a weekly trend used to create urgency before the fact; and a schedule owned by the team to drive itself, not built to satisfy a reporting request. Time was treated as the scarce resource it is — priced, in this program, at about \$173,000 a day.

How to read this case. The technology was specific to solar power electronics; the method was not. The host, team, product, and time practices apply to any program that has to turn a hard idea into the right product before the window closes. That is the transferable lesson of Project Rubicon, and the reason it earns a place in the lateralworks engagement series. The full framework is at lateralworks.com.

Sources

References

- [1] Enphase Energy, Inc. "Company history and first-generation microinverter (2008)." Corporate and investor materials, enphase.com.
- [2] GTM Research, as reported in industry trade press. Module-level power electronics adoption in U.S. residential solar (MLPE in more than half of residential installations by 2014); installer field-reliability commentary, 2015.
- [3] National Fire Protection Association. National Electrical Code (NFPA 70), Article 690.12 "Rapid Shutdown of PV Systems on Buildings," 2014 edition and 2017 revision.
- [4] GreenLancer / ExpertCE. "Solar rapid shutdown requirements: NEC 690.12," industry explainer summarizing the 2014 origin and 2017 array-boundary revision.
- [5] GTM Research. "SolarEdge and Enphase control roughly 80% of the U.S. residential solar inverter market," Q3 2015 data, reported by Greentech Media and photon.info.
- [6] Osborne, M. "SolarEdge revenue increased 15% in 2016." PV-Tech, February 16, 2017. <https://www.pv-tech.org/solaredge-revenue-increased-15-in-2016/>
- [7] Enphase Energy, Inc. "Financial Results for the Fourth Quarter and Fiscal Year 2015." Form 8-K, Exhibit 99.1, filed February 23, 2016. U.S. Securities and Exchange Commission, EDGAR.
- [8] pv magazine. "Enphase posts \$22 million losses during 'challenging' 2015." February 24, 2016. https://www.pv-magazine.com/2016/02/24/enphase-posts-22-million-losses-during-challenging-2015_100023354/
- [9] Greentech Media. "Enphase aims to reduce microinverter cost by 50% in two years" (target of parity with string inverters near \$0.10/W), 2015.
- [10] SolarQuotes. "Do electrolytic capacitors cripple microinverter reliability?" Discussion of rooftop heat exposure and capacitor lifetime, with manufacturer rebuttal. <https://www.solarquotes.com.au/blog/do-electrolytic-capacitors-cripple-microinverter-reliability/>
- [11] Enphase Energy, Inc. "Enphase Microinverter 25-Year Limited Warranty." <https://enphase.com/warranty/us>
- [12] Project Rubicon. "Cable-integrated microinverter data sheet," Version 3.0, internal program document, October 2015.
- [13] Reinertsen, D. G. The Principles of Product Development Flow: Second Generation Lean Product Development. Celeritas Publishing, 2009.
- [14] Smith, P. G., and Reinertsen, D. G. Developing Products in Half the Time. Van Nostrand Reinhold, 1991 (2nd ed. Wiley, 1998).
- [15] Wheelwright, S. C., and Clark, K. B. "Organizing and leading heavyweight development teams." California Management Review, vol. 34, no. 3, Spring 1992.
- [16] lateralworks. "Fast-time-to-market (FTTM) framework: host, team, product, and time." Practice methodology, developed since 1991. lateralworks.com.
- [17] lateralworks. "External project assessment of Project Rubicon." Internal engagement assessment, revised August 2016.
- [18] lateralworks. "Microinverter voice-of-the-customer interviews." Internal engagement research, 2013–2016.