From Curiosity to Prototype: Using Claude to Explore Ensemble Forecasting

How an AI assistant helped me research a complex topic, build a presentation and sketch out a solution architecture—all in a single workflow.

Christine Fordham
Manager, Forecasting Solutions Delivery

THE STARTING POINT

For some time, I've been thinking about where operational forecasting needs to go next. Utilities are managing a grid that looks fundamentally different than it did ten years ago —rooftop solar, battery storage, electric vehicles (EVs) and smart HVAC have introduced far more variability at the edge of the system. The point-forecast models that performed well in a simpler era are increasingly showing their limits. When load is uncertain in both direction and magnitude, a single forecast number isn't enough. Operators need a range.

That led me to explore ensemble forecasting as a potential path forward, and I used Claude, Anthropic's AI assistant to help me do it—not just as a search engine, but as a thinking partner through the whole process: research, presentation development and solution design.

STEP 1: RESEARCH AND FRAMING THE PROBLEM

Before building anything, I needed to understand the landscape. I started by asking Claude to help me frame the operator's dilemma: the grid is increasingly shaped by "behind-the-meter" (BTM) assets that utilities can't directly observe. Solar panels generating power that never shows up as a meter read. Batteries charging and discharging based on customer preferences. EVs plugging in unpredictably.

The research surfaced a useful framing: utilities aren't really facing a forecasting problem—they're facing a visibility problem. The forecasting error is a symptom. What operators need is a model that tells them not just what is likely to happen, but how confident it is about that prediction, so they can make better decisions about reserves, dispatch and grid flexibility.

That distinction—between a point forecast and a probabilistic forecast—became the spine of everything that followed.

STEP 2: BUILDING THE PRESENTATION OUTLINE

With the conceptual framework in place, I worked with Claude to develop a presentation structured around two core questions operators should be asking:

1. Are you using weather uncertainty, or ignoring it?

Modern numerical weather prediction (NWP) systems like European Centre for Medium-Range Weather Forecasts (ECMWF) don't produce a single forecast—they produce an ensemble of 51 plausible weather scenarios. Most utility load models throw away that information, collapsing 51 scenarios into one average. Ensemble forecasting uses all 51, producing a distribution of load outcomes rather than a single number.

2. Can you see what's behind the meter?

Behind-the-meter (BTM) assets—solar, batteries, EVs, smart HVAC—are now significant enough to swing a feeder's load by 20–40%. A load forecast that can't account for them isn't really a load forecast anymore. The presentation explored each asset class and what data and modeling approaches are needed to incorporate them.

Claude helped me think through the logical flow of these ideas, draft the slide content and then—critically—apply our corporate PowerPoint slide master template so the final deck matched our visual standards. That last step, which would normally take a few hours of manual formatting work, took about 20 minutes.

STEP 3: DESIGNING THE SOLUTION ARCHITECTURE

After completing the deck, I asked Claude to help me sketch out what a real forecasting application supporting this approach would actually look like. The prompt was deliberately open-ended:

> *"Focusing on Ensemble Forecasting, lay out the structure of a forecasting application that supports both simulation ensembles with constrained quantile regression averaging and Bayesian Transformers."*

What came back was a layered architecture with two parallel modeling branches that converge into a single operational output:

Branch A —Forecast Ensemble + CQRA

Run the existing load model once per NWP weather member (51 parallel runs). Collect the resulting load traces and extract calibrated quantiles using constrained quantile regression averaging (CQRA). The "constrained" part means the averaging weights are optimized to minimize forecast error at each quantile level separately, rather than applying uniform weights across the board. Studies have shown the potential to deliver a 4.4% reduction in error versus the best single model.

Branch B — Bayesian Transformer

A deep learning mode—specifically a Temporal Fusion Transformer—paired with a Bayesian uncertainty layer—the model doesn't just produce a forecast, it produces a probability distribution over possible outcomes, with intervals that honestly widen when it's operating outside its training experience. This is particularly valuable for unusual events like heat domes or newly connected industrial loads that the model hasn't seen before.

Combined output

Both branches produce calibrated quantile forecasts. A mixing layer blends them using weights tuned on held-out data, weighted by the Continuous Ranked Probability Score, a metric that rewards forecasts for being both accurate and well-calibrated. The combined output feeds four operational decisions: reserve commitment, demand response activation, BTM asset dispatch optimization, and model governance (automated alerts when the model's calibration drifts).

STEP 4: GOING DEEPER ON THE CONCEPTS

One of the most useful things Claude contributed during this process wasn't code or slides, but clear explanation of concepts I needed to understand well enough to discuss credibly.

CQRA and pinball loss: I asked Claude to explain how CQRA works under the hood. The key idea is pinball loss: a scoring function that penalizes a forecast asymmetrically depending on the quantile being targeted. If you're trying to produce a P90 forecast (a threshold that load will fall below 90% of the time), getting it wrong by under-predicting is nine times more costly than over-predicting. Minimizing pinball loss across the training data forces the model to learn that threshold. An isotonic regression step on top of that ensures the resulting quantiles don't "cross"—that a P80 forecast is always lower than a P90 forecast, which sounds obvious but can easily be violated when combining forecasts from multiple models independently.

Fact-checking: When I asked about V2G (vehicle-to-grid) revenue figures included in the deck, Claude didn't just confirm the numbers—it traced them back to original sources, noted that one figure was a forward-looking projection rather than a measured result and surfaced the actual trial data showing higher measured revenue than the estimate I had used. That kind of self-correction was genuinely useful.

WHAT THIS PROCESS SHOWED ME

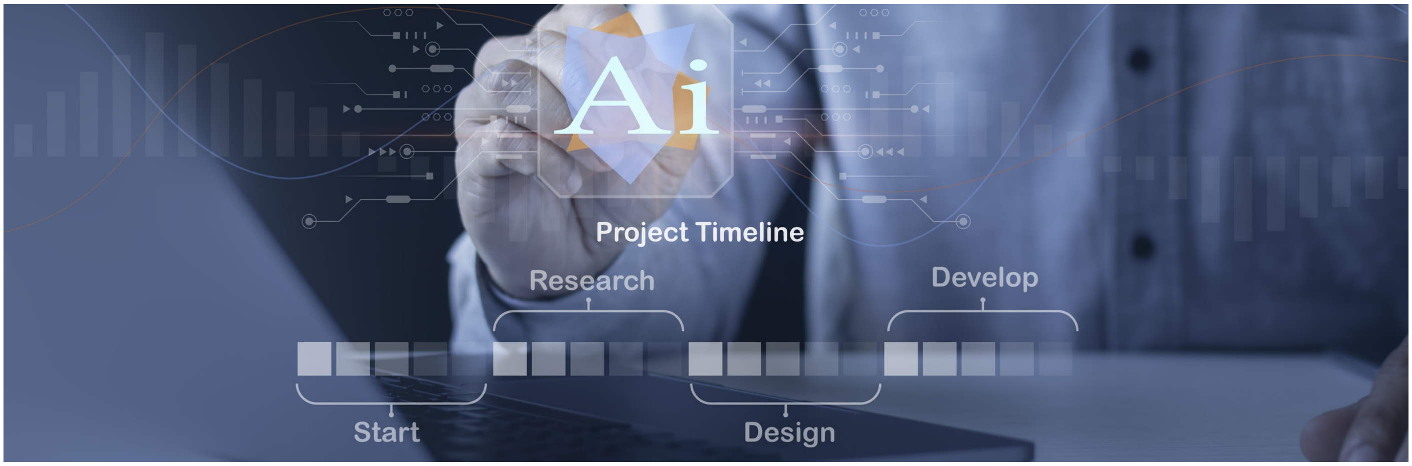
A few takeaways from working through a domain-exploration project like this with an AI assistant:

The value is in the iteration. I did not get the right framing on the first prompt. The presentation structure emerged through back-and-forth—asking Claude to explain something, pushing back on an answer, requesting a simpler version, then using that as input to refine the next slide. The conversation itself became part of the work.

It's a force multiplier for technical breadth. I understand forecasting well, but I don't have deep working knowledge of Bayesian neural network architectures or the specific mathematics of quantile regression. Claude helped me get to a credible, detailed understanding of those topics fast enough to build a coherent presentation—and to prototype a solution structure I could actually defend in a technical conversation.

Formatting and document production are nearly free. Rebuilding the deck to use the corporate slide master—something that would have taken several hours manually—became a single prompt. The AI handled the XML manipulation of the PowerPoint file directly. The productivity gain for anyone who spends meaningful time on polished deliverables is significant and often overlooked.

The prototype isn't a finished product. The architecture Claude sketched out is a starting point, not a specification. It identifies the right components and their relationships, but the decisions about data pipelines, latency requirements, infrastructure and model governance still require domain expertise and real engineering work. The prototype helps clarify what to build; it doesn't build it for you.



STEP 5: IMPLEMENTING A WORKING SIMULATOR FOR BRANCH A

We already have a scalable solution for executing the models for each weather member, but we need to extend our solution to provide operationally useful results. Building a prototype ensemble forecasting engine using real-world data and base models that I already understood allowed me to focus on the unfamiliar part of the problem, CQRA, and how to structure data inputs for efficient operational execution.

I then started a new Python project in Claude Code, providing the high-level prototype and a prompt to focus on implementing a simulator for the Forecast Ensemble + CQRA.

The first pass used simulated data and returned numeric outputs and produced a structure that was easy to extend. Through a series of prompts, I enhanced the simulator to use real data from a historical data file and our weather service. In the process, I discovered some blockers that I might not have identified if this had remained a thought exercise rather than a working prototype grounded in familiar data. There were also bugs, including problems with units and time zones that I was able to catch and ask Claude to correct. I also added many diagnostic plots to support bugging and deepen my understanding of how the CQRA model behaved.

WHERE THIS GOES NEXT

Next, I plan to continue developing the ensemble simulator to help inform how we will design the features to support CQRA in MetrixIDR. Implementing the Bayesian Transformer and the combined output is new territory for our team, and I am looking forward to doing additional research to prototype and validate that approach with real data. I'll continue documenting that work as it progresses.

If you work in utility forecasting and thinking about how to move from point forecasts to probabilistic outputs, I would encourage you to start with the question Claude helped me articulate at the outset: What decisions are you actually making with this forecast, and what would you do differently if you had a range instead of a single number? For me, that question shaped everything that followed.

Christine Fordham
Manager, Forecasting Solutions Delivery

Ms. Fordham manages the forecasting solutions delivery team, overseeing all aspects of consulting projects and software implementation. In addition, she manages Itron's forecasting product portfolio. She has more than 30 years of experience in the forecasting area, across the delivery, product management and R&D areas. In her previous role, she led the development team for Itron's suite of forecasting products. She is a certified SAFe Leader and applies the principles of the Scaled Agile Framework to product development at Itron. Recently, Ms. Fordham has been instrumental in the design and development of our new high-volume services, enabling both operational ensemble forecasting as well as forecasting at the transformer and service point levels, which are key components of the Itron Low Voltage DERMs solution.

Itron
2111 North Molter Road
Liberty Lake, WA 99019 USA