

PBUGG 2026 – BERLIN – 19-20 October

From writing code to instructing AI

The role of AI in PowerBuilder development

Marco Meoni

PBwink CTO & Appeon MVP
marco.meoni@gmail.com

Before PB and AI

PhD THESIS · 2018, CERN

"Mining Predictive Models for Big Data Placement"

Machine learning on the access history of large-scale scientific data, to predict which datasets will be needed and where to place them.

The problem

Huge data volumes, finite storage and bandwidth
[50PB/y, 3B+ datafiles, across worldwide sites]

The approach

Predictive and deep learning models trained on
usage logs

The lesson

Models can learn patterns we never wrote down as
rules

UNIVERSITY OF PISA · 2018

**"Deep learning is working so well
and we don't even know why."**

Antonio Gulli, Google

Closing remark, Seminar on Deep Learning, University of Pisa (as I remember it)

2018 → 2026: from predicting to generating

2018 · THE NEXT BIG THING

Machine & deep learning

Recognize patterns.
Predict indicators.
Specialist models, specialist teams.



2026 · THE STANDARD

"AI", a broad term

Understand intent.
Generate entire applications.
Everyday tools for every developer.

I can't remember the last time I spent hours writing code.

Today I describe the job. AI assistants and LLMs do it.

Not yet officially supported in the PowerBuilder IDE, and already a game changer.

Why it works now: PB 2025 speaks text

THE ENABLER

Solution & text-based workspace

Objects live as readable source files on disk, not inside binary PBLs. An LLM can read, write and diff them like any other code base.

THE CONVERSATION

Claude Desktop

Design discussions and entire components from a description: windows, user objects, DataWindows, PowerScript functions.

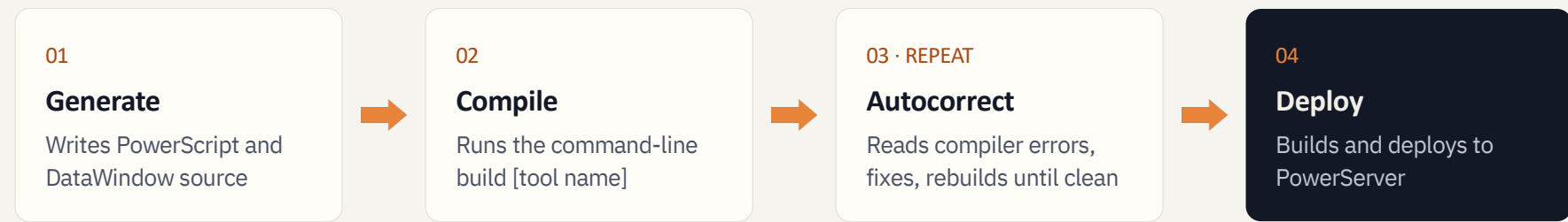
THE AGENT

Claude Code

Works inside the project folder: reads the whole code base, edits files, runs the build tools itself.

Will use PB 2025 R3

The Claude Code loop



My job: describe, review, approve.

Then wait and watch the job get done.

Demo time

A legacy PB app is no longer legacy.

Every large client/server PowerBuilder application can now be regenerated as a modern web application.

Why I'm still faithful to PowerBuilder

BACKEND · KEEP IT

DataStore Web API

DB transactions, business rules and years of tested DataWindow logic. A foundation AI can support, but should not re-engineer.

FRONTEND · REGENERATE IT

AI-generated web UI

Screens, navigation, forms and lists against the existing REST endpoints. A task Claude Code does in minutes.

The real constraint is people

Adopting an AI-generated frontend is only a matter of **time** and **human resources**, not of technology.

A new stack to own

JavaScript framework, build tooling, web hosting and its updates

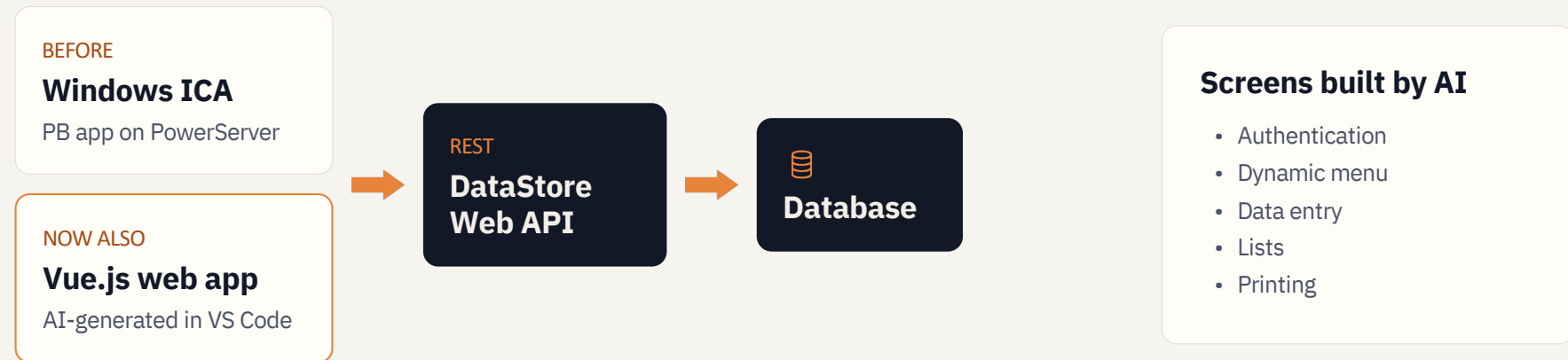
Skills often missing

PB-based companies may not have web developers to dedicate

No need to rush

The backend stays; the new frontend can arrive when the team is ready

Beyond the installable cloud app



Is PowerServer still worth it?

THE DOUBT

Is it still the only migration path for client/server PB apps?

Why pay a license if AI can generate a web frontend on the DS REST API, or transform the whole C/S project?

MY ANSWER: YES, DEPENDING ON WHAT YOU WANT

It moves the **whole application**, not a prototype: same code base, same DataWindows, transactions, deployment and **vendor support**.

Its value shifts from "the only way to the cloud" to **the lowest risk and pure PB code solution**.

Honestly, the risk is real: PowerServer must evolve with AI, not compete with it.

ASIDE

SnapDevelop and the fate of low-code

SnapDevelop 2025

A great low-code IDE. I used it for a while and built a couple of web apps with it.

About a year ago, Appeon stepped away from it.

MY READING

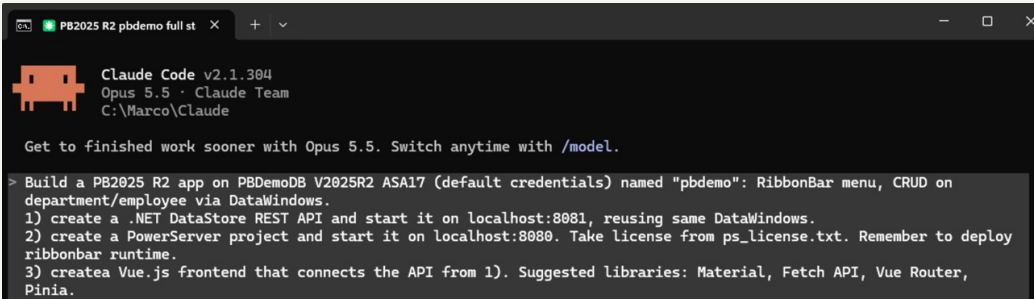
Low-code is squeezed by AI. Why drag and drop components when you can describe the whole app?

Demo time

DEMO

PB, PS, API and Web in minutes

THE PROMPT

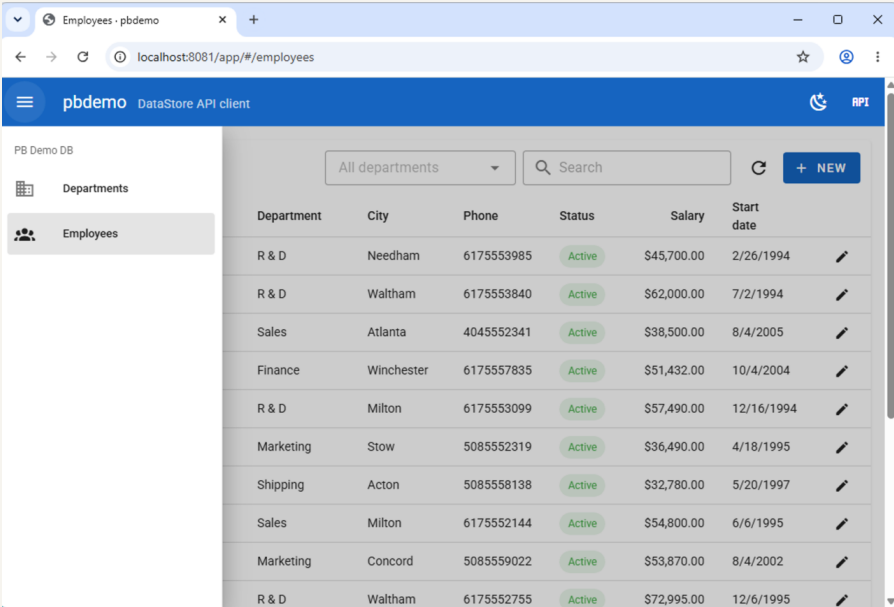


THE PROCESSING

- The pbdemo app is built and both servers are running. Test-Apis.ps1 prints All checks passed, and the Vue frontend shows live department and employee data in screenshots I took with a separate headless Edge, without touching your desktop.

What	Where
Native app (RibbonBar, department/employee CRUD)	C:\Marco\Claude\pbdemo\bin\pbdemo.exe
1) DataStore REST API, reusing the same DataWindows	http://localhost:8081/swagger
2) PowerServer client	http://localhost:8080/pbdemo/
2) PowerServer Swagger	http://localhost:8080/PowerServer
3) Vue.js frontend	http://localhost:8081/app/

THE RESULT



WHERE IS THIS ALL GOING?

Not coding. Instructing.

A totally new opportunity: our expertise moves from writing code to telling AI exactly what we need.

01

Our needs

Business rules and domain knowledge

02

Technology stack

PowerBuilder, PowerServer, Vue.js...

03

Security layer

Authentication, roles, data protection

04

Framework

Conventions, patterns and architecture

Thank you

- Questions? — **Marco Meoni**