

What SMB-Friendly IBM i Modernization Actually Looks Like

If IBM i modernization conversations sometimes feel distant to SMB teams, it's often because the examples assume resources that many smaller environments simply don't have.

Dedicated modernization budgets, specialized development teams, and multi-year transformation programs are real — but they're not the starting point for most IBM i shops.

For many small and mid-sized organizations, modernization doesn't begin with transformation programs.

It begins with clarity.

Step 1: Understanding What Already Exists

Before any new tools or platforms enter the discussion, many SMB teams start with something much simpler:

Understanding their own systems.

That can mean:

- Mapping application dependencies
- Documenting business rules buried in code
- Identifying unused objects or dead logic
- Clarifying how data actually flows through the system

This kind of work rarely appears in modernization marketing, but it is often the most valuable step.

When teams understand their systems better, they gain the confidence to change them.

Step 2: Reducing Fear of Change

One of the biggest barriers to modernization in SMB environments isn't resistance. It's risk.

When a system runs payroll, shipping, accounting, and customer operations, even small changes can feel dangerous.

Modernization therefore often begins by making change safer:

- Adding version control
- Improving testing practices
- Cleaning up fragile areas of code
- Reducing hidden dependencies

The goal isn't speed. It's stability during improvement.

Step 3: Improving the Developer Experience

Only after systems become easier to understand and safer to change do many teams begin exploring new tooling.

That might include:

- Modern IDEs
- Git integration
- Better deployment processes
- API exposure of business logic

At this stage, modernization starts to accelerate — not because tools appeared, but because the system is now understandable enough to evolve.

Step 4: Incremental Modernization

For SMB teams, modernization usually happens in small, steady steps:

- Modernizing interfaces
- Improving integration points
- Gradually refactoring core logic
- Making systems easier for new developers to learn

None of these steps require abandoning the platform.

They simply make the system healthier over time.

The Real Pattern

In many SMB environments, modernization looks less like disruption and more like stewardship.

The goal isn't replacing what works.

It's making sure what works **can keep working for the next generation of operators and developers.**