

BRIDGING THE KNOWLEDGE & SKILLS DIVIDE WITH AI-POWERED ASSISTANTS



Picture two mainframe professionals at the height of their careers: Rebecca, an application developer pushing critical business updates, and Sebastian, a system programmer safeguarding the heartbeat of a global enterprise. Despite working in an industry that powers the world's most secure transactions and high-speed analytics, both often feel stuck in the past, tangled in aging code, scattered knowledge, and a shrinking pool of experts to turn to when the clock is ticking.

This is their story. It's also the story of thousands like them, working quietly behind the scenes to keep the world running, and how [conversational and contextual AI-powered assistants](#) are rewriting what's possible for the next generation of mainframers.

Rebecca: The developer with modern ambitions and heritage code

Rebecca's day begins the same way it often ends, squinting at lines of COBOL written before she was born. Each update she pushes carries the weight of millions in potential impact. One wrong line, and entire services can go offline, taking customers and trust with them.

She's talented, eager, and trained in modern development frameworks and tools, yet every time she tries to integrate a new cloud-native feature with mission-critical mainframe backends, it feels like hammering mismatched puzzle pieces into place.

Her frustrations echo a familiar refrain:

"I work in one of the most advanced industries, yet my development tools feel outdated. I wish I had a mentor who could answer my mainframe development questions instantly."

She's not alone. Across industries, the push to deliver new digital experiences collides daily with the reality of aging but indispensable systems, opaque codebases, and vanishing expertise.

Sebastian: The system programmer racing against time

Down the hall, Sebastian scans a sea of system logs, watching for threats that evolve faster than the defenses built to stop them. He knows every outage is unacceptable, every second counts. Yet as his veteran colleagues retire, so does the institutional knowledge of decades-old configurations, quirky fixes, and unwritten workarounds.

When a performance bottleneck arises at midnight, or a job failure threatens financial transactions, Sebastian must dive into layers of system data, all while customers expect perfection. One slip in service reliability can ripple into frustrated users, eroded trust, and lost revenue that's hard to recover.

"There's always pressure to make systems faster, efficient, and more reliable, but where do I even start? Troubleshooting a mainframe issue feels like finding a needle in a haystack."

He can't monitor everything 24/7. He can't clone himself. And he can't plug the brain drain overnight. Something has to change.

The changing face of the mainframe workforce

Their challenges aren't isolated stories, they're data points in an industry-wide shift. According to the latest [BMC Mainframe Survey](#), the share of mainframe professionals with 20+ years of experience dropped from 36% to 13% in five years. Meanwhile, the proportion of staff with mid-level experience (6–10 years) has grown significantly—talented, but less seasoned in the dark art of deciphering deep-rooted systems.

This demographic change is rewriting the rules. Organizations can no longer afford to be passive caretakers, hoping old skills will last forever. They must either commit to migrating everything off the mainframe (an expensive and risky path for workloads that run best where they are) or find a new way to modernize and optimize the mainframe in place.

The AI-powered path forward

Enter [AI-powered assistants](#)—not just search bars, but contextual, task-focused experts built into the very tools that Rebecca and Sebastian use every day. These assistants blend large language models (LLMs), domain-specific small language models (SLMs), and decades of institutional knowledge into an always-on, conversational ally.

For Rebecca, an AI assistant means she doesn't have to dig through cryptic COBOL or JCL manuals at 2 AM. She can simply ask, *"What does this subroutine do?"* and get an instant, plain-language explanation. She can push updates faster and with confidence, knowing her AI assistant has her back, suggesting code improvements, generating test cases, and even converting legacy code to modern languages like Java when needed.

By eliminating hours or even days spent on basic research and documentation, AI-powered

assistants free developers to focus on building new features and tackling strategic initiatives. The result is a direct boost to productivity and a stronger pipeline of innovation that drives competitive advantage.

For Sebastian, it means no more late-night panic hunting for system log correlations or calling a retired colleague for help. His assistant surfaces probable causes, recommends next steps, and even explains complex automation workflows in simple terms. If a problem recurs, the assistant remembers, preserving that critical fix for the next time, for the next person.

The technology inside the assistant

This transformation isn't magic. It's a deliberate layering of advanced AI technologies:

- **Natural Language Processing (NLP)** understands everyday questions and delivers technical explanations in conversational language.
- **Task Automation** can carry out repetitive tasks autonomously or semi-autonomously.
- **Context Awareness** knows the environment, user role, and historical context, making every answer relevant.
- **Retrieval-Augmented Generation (RAG)** combines live search through trusted data sources with generative AI to deliver accurate, evidence-backed responses.
- **Hybrid AI** blends machine learning, rules engines, and generative models for resilience and trustworthiness.

The result? An [AI knowledge expert](#) that learns, adapts, and grows alongside your mainframe teams, bridging generational divides and preventing knowledge loss as experts retire.

Forward-looking companies are taking it a step further: combining vendor-supplied assistants with their own domain-specific knowledge bases and custom-trained models. This approach keeps proprietary expertise secure while expanding the assistant's IQ with each incident resolved and each question answered.

Some are also mixing large LLMs with lighter SLMs, optimizing cost, latency, and precision for different tasks. The idea is simple: powerful general reasoning when you need it, efficient task execution when you don't.

From use case to reality

Organizations today are moving beyond generic chatbots and stand-alone AI experiments by embedding production-ready assistants directly into their core mainframe environments. For developers, this means having an AI-powered experience within their [application development and modernization tools](#), where the assistant explains legacy code, suggests modern design patterns, and automatically generates clear documentation.

For operations teams, the same intelligent technology is woven into monitoring and incident management solutions, [using hybrid AI to explain probable causes](#) of mainframe system issues and recommend next best actions, all in natural language, directly within the operational workflows they rely on every day.

Unlike generic chatbots, these assistants are domain specific. They understand mainframe nuances, jargon, and context that broad-purpose AI models can't grasp out of the box. And because the

models are curated, accuracy and trust remain high, with no hallucinations, just practical, actionable insights.

Why it matters

For enterprises, the stakes are existential. Banks, airlines, and governments can't afford downtime. They can't gamble on brittle migrations or hope junior hires will become experts overnight. AI assistants transform how institutional knowledge is captured, organized, and reused, ensuring each new mainframer inherits not just a job, but a digital mentor. More IT leaders are choosing to stay on the mainframe because AI gives them the confidence to modernize in place, overcome workforce challenges, and unlock new business value without the risk of starting over.

They also deliver confidence: to deploy faster, fix faster, and innovate fearlessly. Freed from repetitive grunt work, developers and operators alike can spend more time on high-value projects that move the business forward.

Rethinking talent strategy

This shift is reshaping talent strategies too. Instead of focusing solely on hiring unicorns with deep legacy experience, organizations can build mixed-skill teams, relying on AI to close the knowledge divide. According to a recent [Forrester report](#) on the ROI of Mainframe application development modernization, many organizations are already cutting new-hire onboarding times by 50 percent simply by adopting modern development tools. Now, imagine the impact when AI assistants handle code explanation, documentation, and even code conversion automatically. New hires onboard faster. Veteran experts spend less time answering the same questions and more time solving novel problems.

As a result, the mainframe doesn't just survive the talent crunch, it thrives, with a workforce supercharged by [digital knowledge expertise](#) on demand.

Beyond the assistant: The role of AI agents

AI assistants are powerful on their own but they become even more impactful when they work hand-in-hand with [AI agents acting as digital intelligent workers](#) behind the scenes. When a task needs to go beyond answering questions—like diagnosing a recurring job failure, checking dependencies, rerunning processes, and verifying results—an assistant can call on an agent to handle it autonomously.

Together, assistants and agents blend guidance with action, combining conversational help with task execution to deliver continuous, intelligent automation. This synergy keeps the mainframe as agile as modern cloud environments, without losing the reliability that makes it essential.

A tale still being written

Rebecca no longer dreads updates. Sebastian no longer works alone. Together, human plus machine, they're setting a new standard for what mainframe work can be: faster, smarter, and more resilient.

This is the real promise of AI for the mainframe: not replacing experts, but multiplying their impact,

preserving their wisdom, and turning every new hire into a seasoned pro from day one.

Closing thoughts

By 2030, analysts predict that AI assistants will be as common in mainframe operations as automation tools are today, an invisible backbone that does the heavy lifting while people focus on what people do best: thinking creatively, solving big problems, and building what's next.

For organizations still deciding whether to retire, maintain, or fully leverage their mainframes, the message is clear: standing still is no longer an option. Modernize in place. Optimize fearlessly. And empower every mainframer with an AI partner that never sleeps, never forgets, and never stops learning.

Rebecca and Sebastian's story could be your story too, a tale not of old systems dragging you down, but of timeless reliability supercharged with new intelligence. To learn more about the [award winning BMC AMI Assistant](#) watch this [video](#) or checkout our modern mainframe tools on the [BMC AMI webpage](#).

You've never mainframed like this. And that's exactly the point.