

# UNTANGLING INTRACTABLE MAINFRAME APPLICATION CODE WITH GENERATIVE AI



The greatest roadblock to modernizing applications on the mainframe is *fear*. Fear that making any change, no matter how minor, will break something.

Given that older, monolithic mainframe applications are typically mission-critical – core bank transaction processing, airline reservation systems, insurance ratings, and claims management, to name a few – IT executives are understandably reluctant to touch such apps at all.

It's no wonder: Many of these applications are extraordinarily complex, consisting of layers of code from decades of development work, in many cases following no consistent coding or documentation practices over the years or decades they have been running the business.

Furthermore, as mainframe professionals retire, their organizations risk the loss of their tribal knowledge – perhaps the only knowledge of how many mainframe applications work. What was once a maintainable application suddenly becomes frighteningly opaque once the experts leave the building.

Despite such challenges, evolving business needs mandate ongoing updates to legacy applications on the mainframe. Regardless of their fear or loss of seasoned personnel, it's time for today's mainframe teams to take the plunge and leverage modern tooling to modernize mainframe applications in place.

# A New Generation of Mainframe Tooling

The first step in untangling complex, monolithic mainframe code is to gain visibility into its inner workings. Tools like [BMC AMI DevX Code Insights](#), for example, leverage BMC's expertise with runtime observability to provide visualizations of mainframe code, enabling developers to understand interactions, dependencies, and relationships among programs.

BMC AMI DevX Code Insights gives developers the power to save and replay execution traces, providing insight into program calls, behaviors, and input/output types. The tool also provides data flow analysis, as well as code structure and logic charts, giving developers an understanding of how data flow through an application.

BMC AMI DevX Code Insights breaks down the impenetrability of complex, monolithic mainframe applications – but it doesn't go far enough. Even with such powerful tooling, developers are left with an essential question.

*What does the program actually do?*

Tools like BMC AMI DevX Code Insights need additional capabilities to answer this fundamental question.

Tracing data flows is one thing, but connecting the numerous lines of code to the business requirements that originally defined their behavior can be elusive, especially for more complex, layered applications.

The missing piece of the puzzle: *generative AI* (GenAI). GenAI feeds massive data sets into large language models (LLMs), which, in turn, answer user queries in natural, conversational language.

With GenAI, tools like BMC AMI DevX Code Insights can finally answer this question.

## Extending BMC AMI DevX Code Insights with GenAI

BMC has incorporated the power of GenAI into BMC AMI DevX Code Insights via its [BMC AMI Assistant](#) capability. The assistant takes as input existing source code, then outputs conversational English that describes what the code actually does – not only to developers, but to business analysts, managers, or anyone else with a vested interest in the behavior of mainframe applications.

BMC AMI Assistant works seamlessly with the rest of BMC AMI DevX Code Insights, augmenting its code structure and data flow analysis capabilities with insight into the behavior of the applications themselves.

The result is quick understanding of even complex, unfamiliar code – with clear explanations of what the code is doing and how it's doing it.

In fact, the output from BMC AMI Assistant is so clear that developers can use it to add comments to code that may have been lacking adequate documentation – giving future developers a leg up on further modernization work.

## Beyond Code Explanations

GenAI can be useful for more than code explanations on the mainframe. Here are some examples.

*Code reviews* – GenAI can augment human code reviews by providing on-demand, interactive

feedback based on corporate rules, policies, and best practices.

*Test assistance* – GenAI can create test cases with optimal coverage while also creating the minimal quantity of test data necessary to run the tests. The test assistance functionality can analyze the source code itself to determine the best strategy for testing it.

*Code assistance* – GenAI can provide a "copilot" that can leverage knowledge of best practices, design patterns, and coding standards, both industrywide and specific to the organization.

## **The Intellyx Take**

The two main reasons why people are so afraid of touching monolithic mainframe code are the lack of visibility into what that code is doing and insufficient control over the code's behavior.

With its new GenAI-powered capabilities, BMC AMI DevX Code Insights addresses both causes. Not only does BMC AMI DevX Code Insights with BMC AMI Assistant help developers understand the structure and behavior of complex mainframe applications, it also provides vital insights into what those applications actually do for the business.

BMC AMI DevX Code Insights has now become an even more vital tool of empowerment for mainframe development teams – not only dispelling the fear of breaking something, but also instilling the confidence that developers can make the updates and changes their organizations require, regardless of the complexity of the code.

*Copyright © Intellyx BV. BMC is an Intellyx customer. Intellyx retains final editorial control of this article. No AI was used to write this article.*