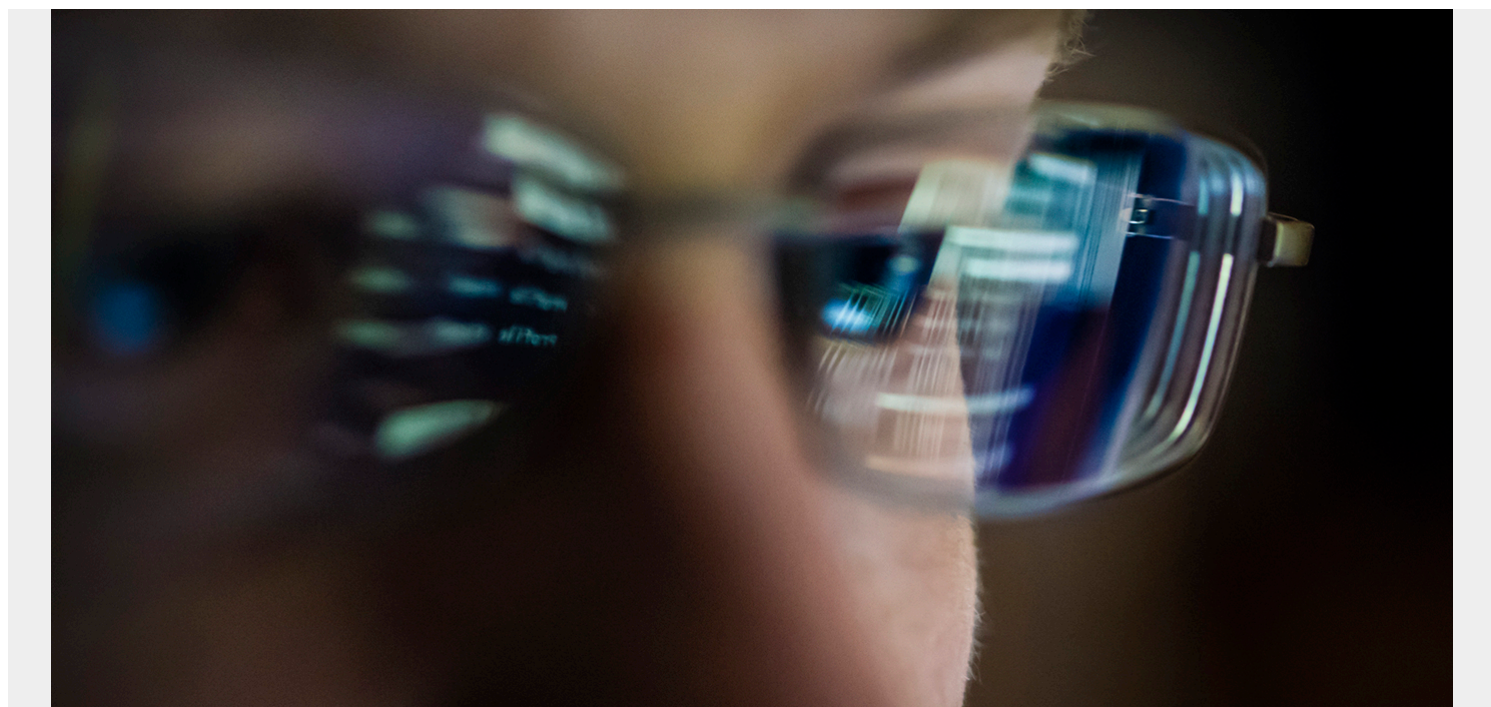


DO YOUR DEVELOPERS HAVE THE DROIDS THEY ARE LOOKING FOR?



In the Star Wars movies, starfighter pilots are commonly assisted by astromechs, a type of repair droid that serves as an automated mechanic. Astromechs have many appendages — tools that can do almost anything. Pilots rely on their astromech copilots to control flight and power distribution systems while also calculating hyperspace jumps and performing simple repairs. The best example of this is the loyal R2-D2, always there for Luke Skywalker.

When problems arise, and they always do, the astromech droid is there to fix the problem. Couldn't developers benefit from similar automated assistants as they work on code? Luckily, such tools do exist.

Here are some examples of BMC AMI technology that assists developers in delivering quality code – fast.

- When confronted with code they do not understand, a developer working in [BMC AMI DevX Code Insights](#) merely highlights a section of the code, right-clicks, and selects “Explain.” BMC AMI Assistant returns a short, artificial intelligence (AI) generated summary of the business logic and a detailed description of the code’s logic flow. Developers now have an easily available way to work with confidence on the code using the BMC AMI DevX Workbench Editor in Eclipse or [VS Code](#). They can also see charting to understand the structure of the program and the flow of the logic, and trace data from its arrival to its departure—right from the editor they use every day.
- Another assistant is the Runtime Visualizer in BMC AMI DevX [Code Insights](#), which enables developers to visualize their applications in real time. With it, developers can quickly see

exactly how the application works.

- When entering code in the BMC AMI DevX Workbench Editor, a type-ahead feature anticipates an automatically completes reserved words, allowing the developer to select one, saving time and avoiding typos.
- When it comes time to debug a batch program using [BMC AMI DevX Code Debug](#), developers can right-click the JCL member, select 'Debug as' and they're good to go, with the configuration already filled out. Also, the configuration settings are all visible in one dialog, and if important information is missing, the dialog will point it out.
- Creating test data can sometimes be difficult, but in BMC AMI DevX Workbench, developers can use the 'Copy To' function of the Host Explorer. Being able to copy multiple files and rename them, and even copy them to another LPAR with no shared DASD is a big help.
- When there is a compile error, developers can use 'Show Compile Diagnostics' in [BMC AMI Workbench Host Explorer](#) or [BMC AMI DevX Code Pipeline. It takes them](#) straight to the line(s) in their program that caused their compile to fail. This capability saves having to page through the compiler output and then go open the program and locate the line(s) that caused the issue.

Whether you're a starfighter in a galaxy far, far away or a developer working on mainframe applications, it's best not to go it alone. Thanks to these tools, developers have their own faithful assistants to help them reach their goals.

To learn more about these features and how to use them, turn to [the BMC Education Courses for AMI DevX](#).