

The Future of Coding Assistants: Generative AI and Automation

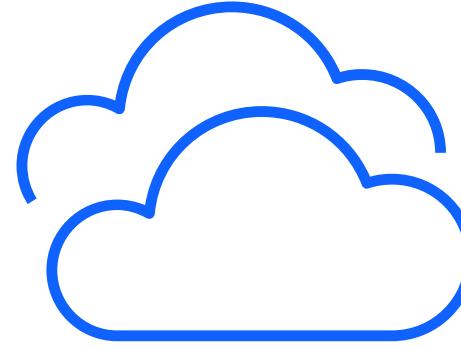
An Enterprise Technology Leadership Summit
Exclusive



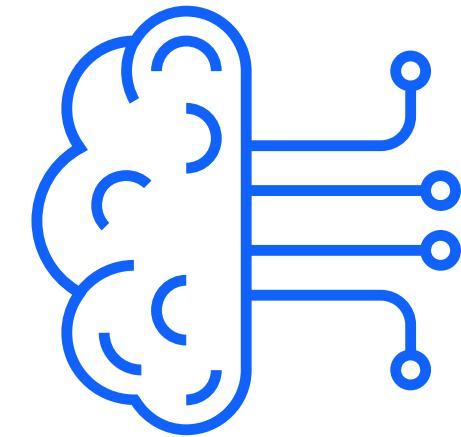
River Wong
Product Manager on IBM watsonx Code Assistant
Data & AI
IBM



River's To Do List for
the next 28min



Why your life is about to get difficult



IBM's perspective on Coding Assistants



What solutions we have for you

83%

of executives agree that application modernization is central to business.

yet only

27%

say their organization has modernized essential workflows.

Why application
modernization
projects fail

High costs

57% of leaders say they're
challenged by financial factors.

Technical Debt

51% say they're challenged by
technical factors.

Skills gap

45% say they're challenged by
expertise factors.

Successful modernization requires:

- Using continuous, targeted modernization to minimize the cost, risk and impact of legacy application optimization.
- Leveraging both generative AI and automation to enhance developer productivity and accelerate the application lifecycle.

70%

of modernization costs will be
reduced by gen AI¹

So how will we use gen AI for modernization?

Create a workout plan
for resistance training

Write a Python script
to automate sending daily email reports

Test my knowledge
on ancient civilizations

Create a morning routine
to boost my productivity

Please modernize the COBOL applications in my mainframe



ChatGPT can make mistakes. Check important info.

AI Models

- Trained on approved open-source licenses
- Trained on use-case relevant data

Products

- Purpose built
- Leveraging automation and gen AI
- E2E developer lifecycle

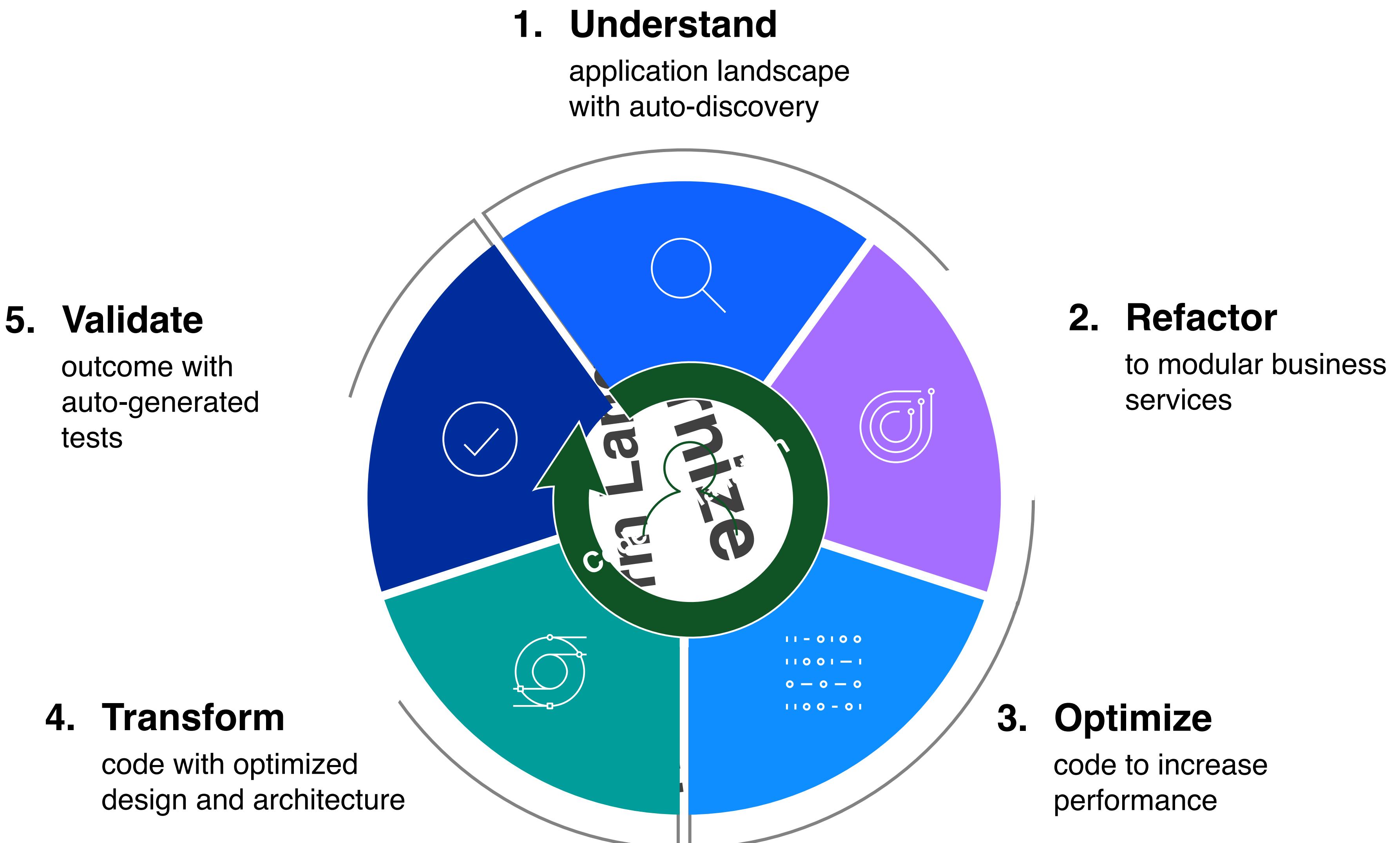
**IBM watson^X™ Code Assistant for
z**

Accelerated application lifecycle

Objectives:

- Address skills and productivity challenges with automation and AI
- Ensure IBM Z qualities of service with mixed language interoperability
- Align with industry standard DevOps approaches

IBM watsonx Code Assistant for Z modernization experience



Tailor your journey based on your application modernization and development needs

IBM watson^X™ Code Assistant for Z

Demo

IBM watsonx Code Assistant | +

wca4z-ra.techzone.ibm.com/#/

Seismic - Login Seismic w3.ibm.com Home • MURAL aIBM | Trello IBM Lighthouse Benevity IBM Learning Portal Tririga workplaceS... Shuttle service-Ho... Succeed with Clari... WCA Bookmarks TT-Backend-Feat... IBM watsonx Code... WCA4Z: SOL 24-2... Ideas list report |... IBM Corporate Car... All Bookmarks

IBM watsonx Code Assistant for Z Refactoring Assistant Workspaces

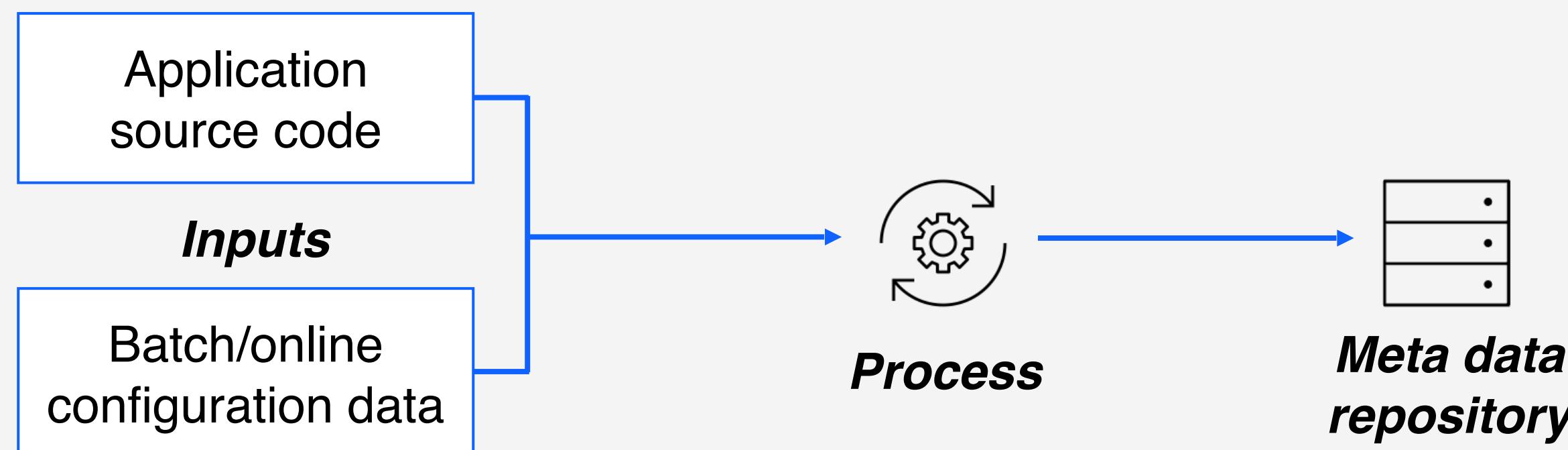
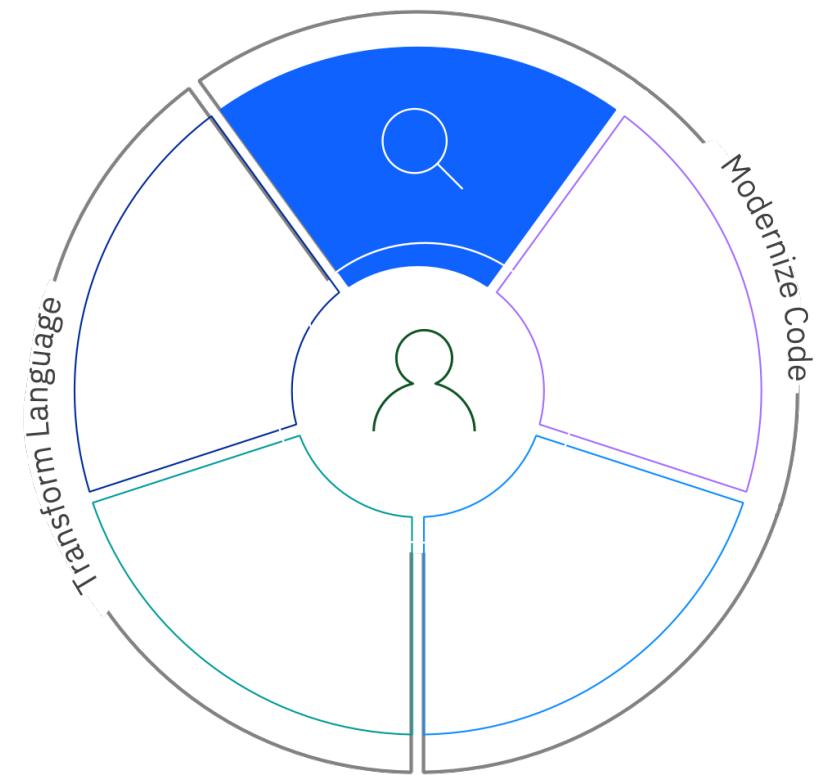
Demo (GenApp) Start typing to search for artifacts Graph Workbook

Let's search!
Begin exploring by searching for an artifact.

Understand: Begin continuous modernization of your tightly coupled applications

Visualize and auto-document your COBOL application at the enterprise level

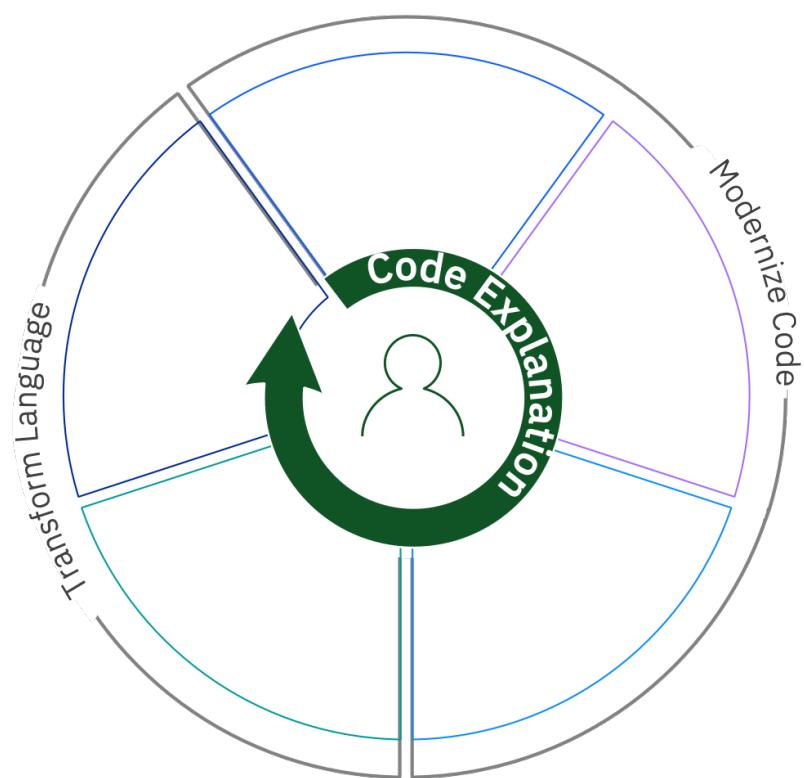
- Start of your application modernization journey with an inventory of applications, resource usage, and dependencies. Leverage COBOL explanation to improve understanding
- Build business alignment and confirm that your understanding of the application is valid – ensuring modernization efforts achieve expectations
- Mitigate the challenge of lack of application SMEs with automated analysis & visualized application flows to enable accelerated application understanding



Application Discovery is the starting point for z/OS application modernization

- Deep enterprise application analysis
- Auto discovery of data and program relationships
- Enable incremental refactoring of business services

Code explanation: understand and document application faster

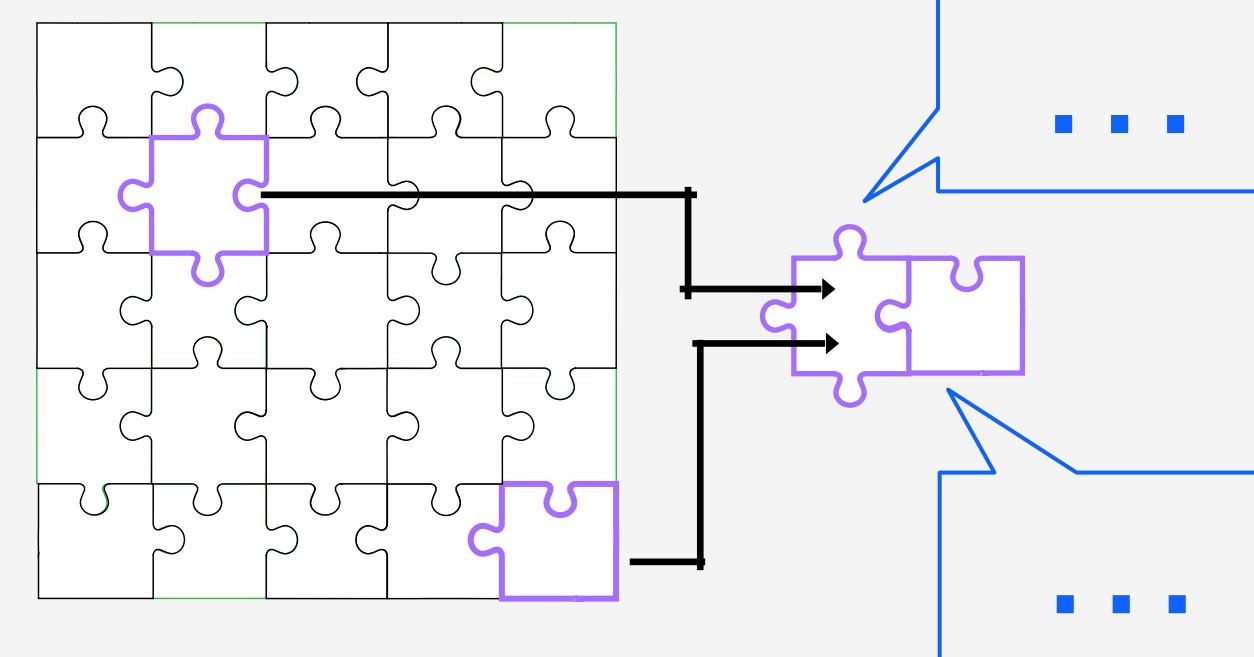


Leverage Generative AI for a natural language explanation of COBOL and JCL

- **Reduce knowledge gap:** Real-time code explanations of COBOL and JCL to aid developers and system programmers, accelerating development or modernization efforts.
- **Free up SMEs:** Less reliance on senior experts, reducing knowledge bottlenecks via real-time code explanations.
- **Streamline documentation:** Utilize code explanations to update application and JCL job step knowledge, reducing manual efforts.
- **Facilitate modernization strategy:** Gain deeper insights into programs, aiding in identifying optimal modernization approaches.

```
230 *====O=====
231 INSERT-CUSTOMER.
232 *=====
233 * Insert row into
234 *=====
235 MOVE 'INSERT
236 *=====
237 IF LGAC-NCS
238 EXEC SQL
239 INSERT
240
241
242
243
244
245
246
247
248
249
```

Explain code in natural language (English)
on-premises* or as-a-service



*COBOL code explanation on-premises availability coming

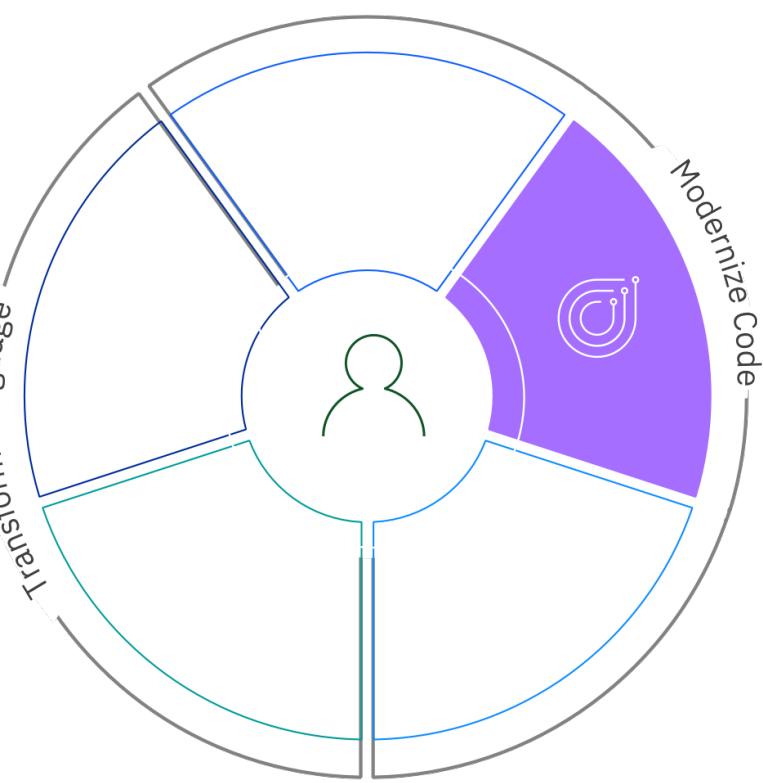
2H

* JCL code explanation availability coming in fall 2024

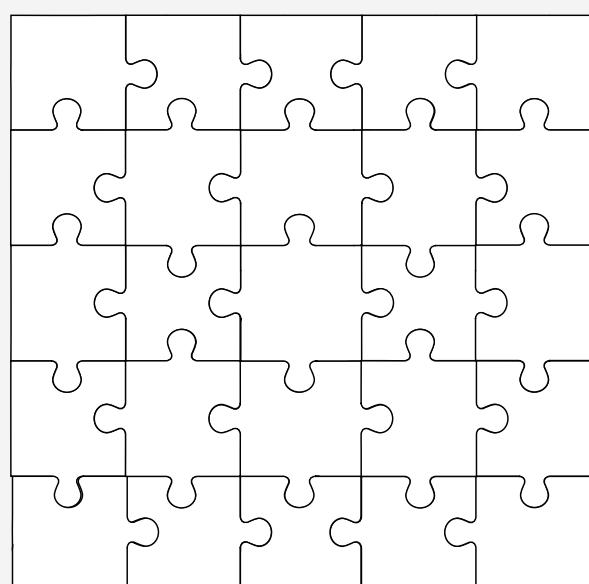
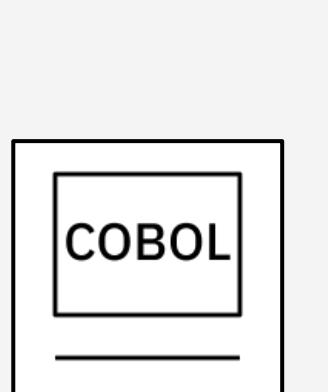
Refactor: Automated tooling to identify services within an application to modernize

Discover programs and data needed for a refactored business service within a large application

- Separate code needed into a refactored service which will be easier to maintain and reuse
- Automate the service creation process to improve accuracy and reduce time and skill required for manual developer analysis
- Unlock modernization development agility and ease of integration

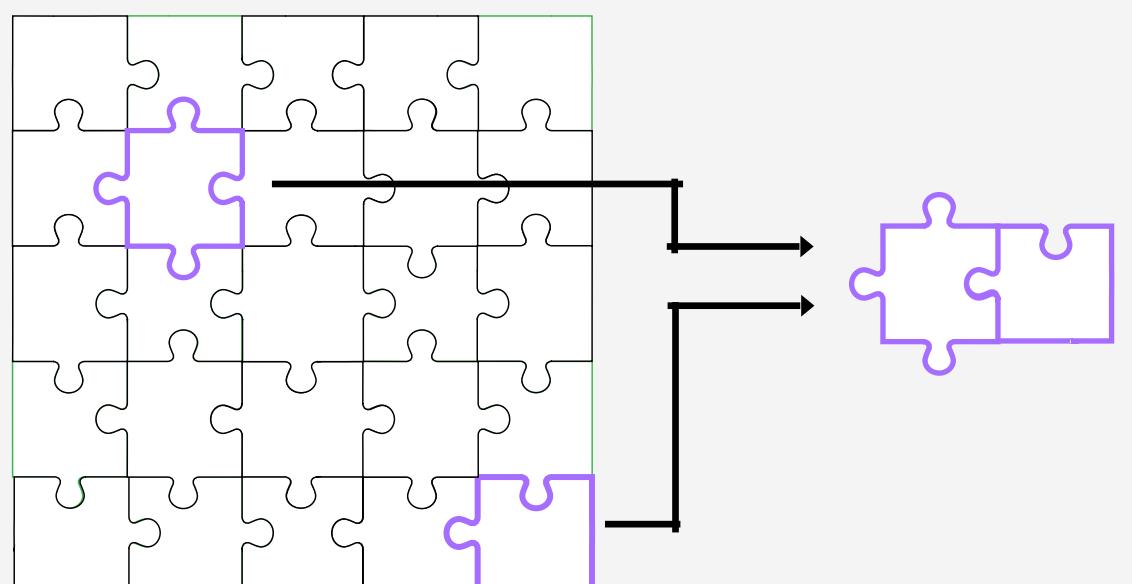


Monolithic application



New automated refactoring capability

New Refactoring Assistant can quickly identify parts of an application to refactor and extract into modular, reusable services via deep functional analysis of the source code.

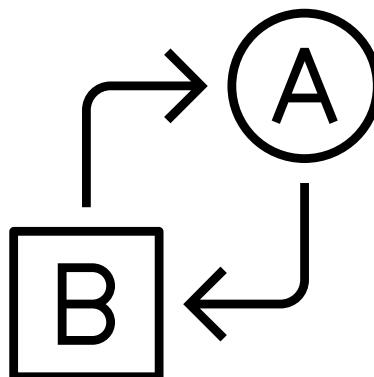


Optimize: Optimize your COBOL code with prioritized performance insights



Performance analysis and recommendations

Conducts in-depth analysis of COBOL modules through static and dynamic analysis, providing actionable recommendations to optimize performance.



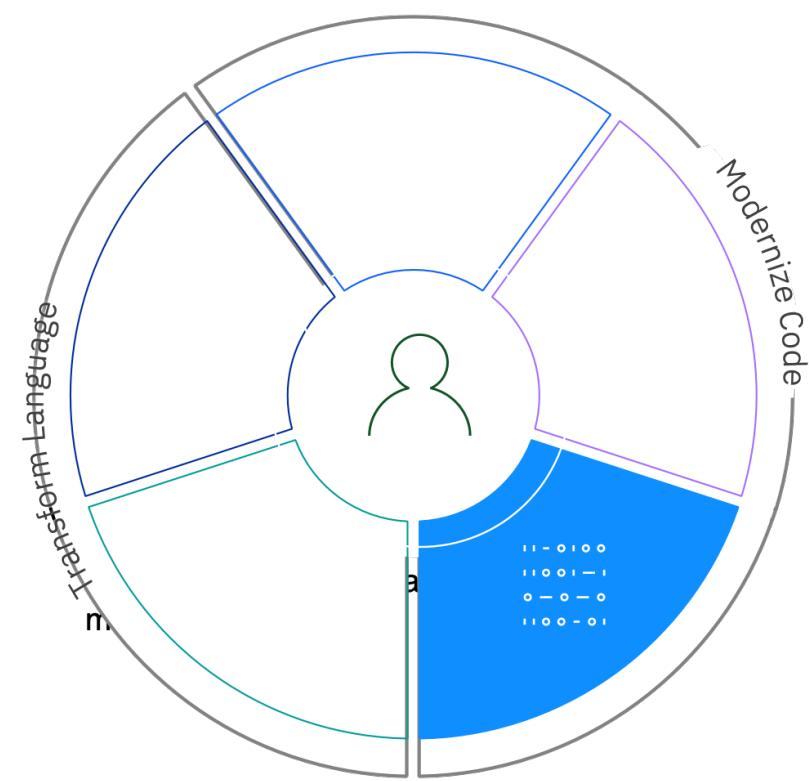
Source-code matching

Offers line-to-line analysis for targeted fixes and enhancements, ensuring precise improvements, all within your IDE.



Ranking and prioritization

Ranks performance issues based on impact, enabling developers to focus on high-priority tasks for maximum efficiency.



Benefits of Performance Insights



Save time, money, and resources through early detection and problem resolution



Reduce skill gap by allowing entry level developers to fix performance issues independently.



Deliver robust and efficient COBOL applications by quickly detecting and fixing issues.

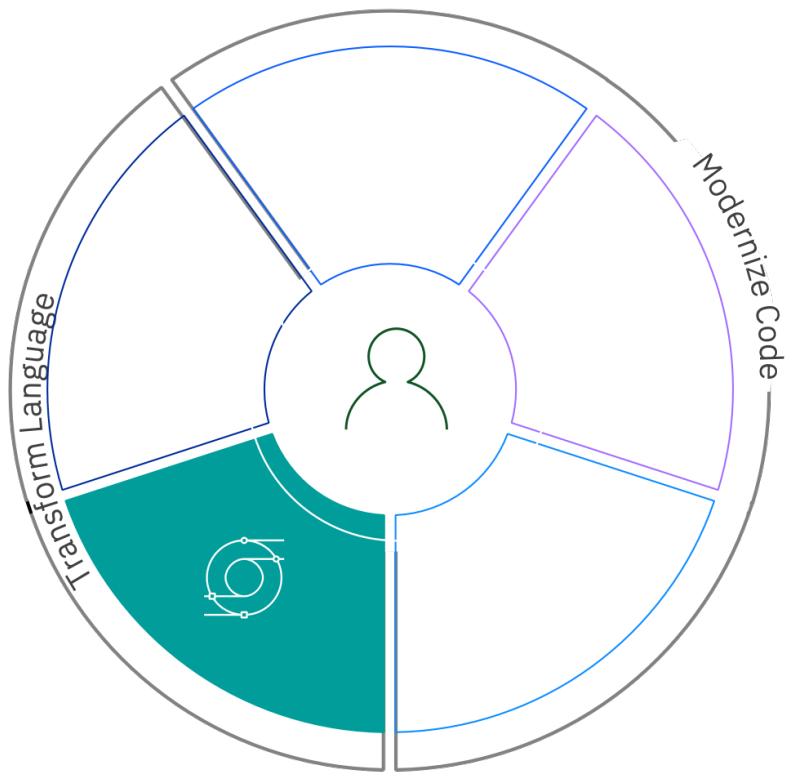
Top performance problems

Line	Problem name	Priority
		Critical
		High
		Medium
		Low

Transform: Leverage generative AI to accelerate COBOL to Java conversion

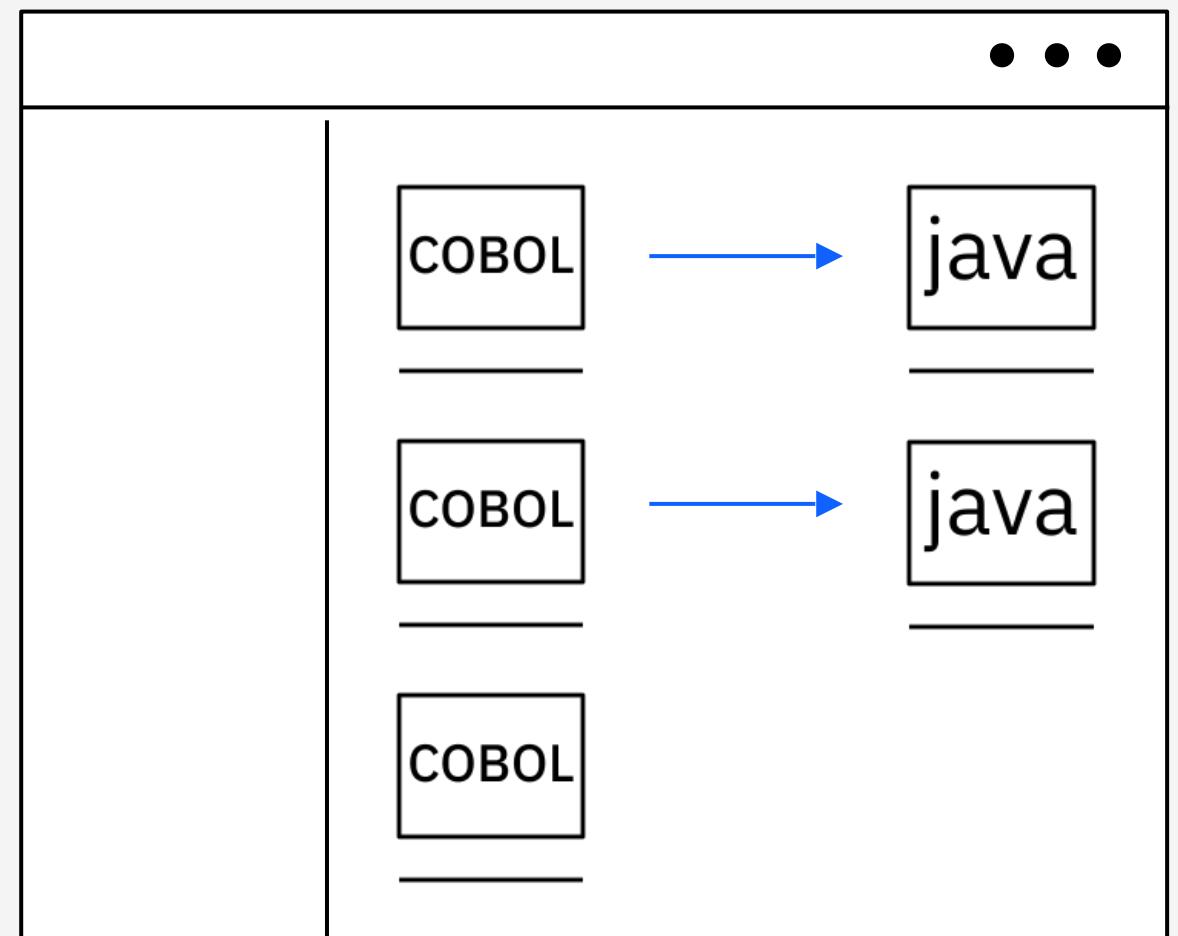
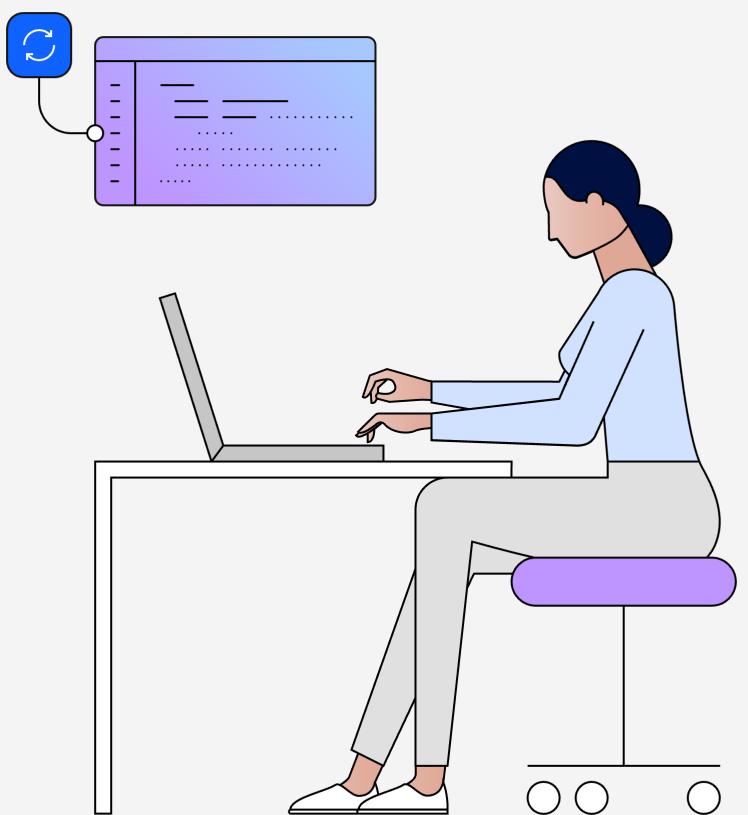
AI assistant to generate Java code in minutes, not months

- Generative AI to build data structures and business logic in Java from your refactored COBOL code
- Well-architected object-oriented Java – not JOBOL
- Maintains IBM Z runtimes and qualities of services with interoperability, integration, and enterprise standardization



IBM watsonx Code Assistant for Z

- State of the art granite.20b.code large language model with a 32k token context window
- Trained with 1.6T tokens across 115 programming languages
- Tuned for Cobol to Java use case



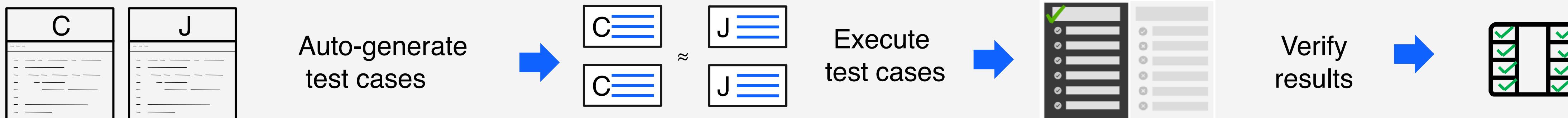
Validate: Automated testing capability

Streamlined and accelerate testing of new code

- Auto generated testing to compare semantic equivalence of new Java service to, providing confidence in a successful Java translation and de-risking the process
- Accelerate developer productivity
 - Enables incremental testing if the Java code is working vs waiting to test broader code path flows in a later test cycle where it's harder to determine errors
 - Tool automation automating tests and enabling them to run in isolation without requiring the middleware to execute the test
 - Junit tests generated can be reused and integrated in the DevOps pipeline per standard practice as the application evolves



Validation Scenario: Tests compare COBOL paragraph and Java method verify equivalence



Uses AI to generate the tests using the same input/output data for both the COBOL and Java tests

Automatic mocking enables unit tests to run on z/OS in isolation without middleware!

Westfield Insurance

With IBM watsonx Code Assistant for Z:

80%

less time spent
understanding the
application

30%

less time explaining
and documenting
code

But wait, there's more...

Generative AI-assisted software development with

GA: October 2023

**IBM watsonx Code
Assistant for Red Hat®
Ansible Lightspeed**

Generative AI to help developers
create Ansible Playbook content
more efficiently

GA: October 2023

**IBM watsonx Code
Assistant for Z**

Automated, generative AI–
assisted mainframe application
modernization

In the works:

Accelerate your Java application
lifecycle with generative AI and
automation

IBM watsonx Code Assistant

Ansible-tuned
model

COBOL-to-Java-
tuned model

Java-tuned model

IBM Granite Code LLM

IBM watsonx Code Assistant

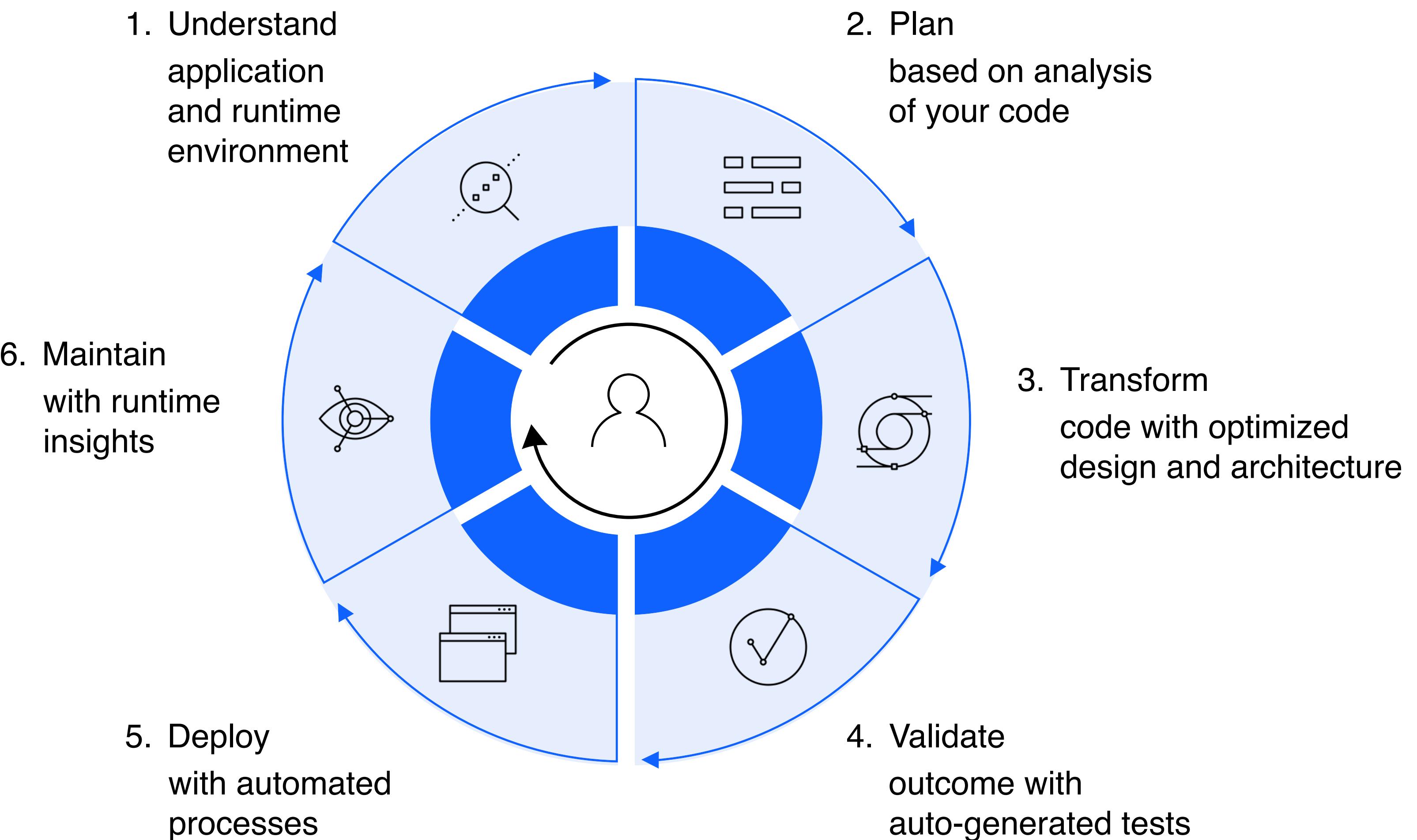
Benefits

Enhance developer productivity

Streamline application modernization

Improve code quality and manageability

Generative AI-assisted Java application lifecycle



Take the next step

Visit us at the IBM Booth!

- Connect with the experts (and/or me!)
- See watsonx Code Assistant in action
- See how your business can accelerate application modernization with automation and generative AI

Demo:

Request a live demo of
watsonx Code Assistant for Z



Webinar:

Address performance issues & optimize COBOL with
watsonx Code Assistant for Z



Learn more:

watsonx Code Assistant product website



Notices and disclaimers

© 2024 International Business Machines Corporation.

All rights reserved.

This document is distributed “as is” without any warranty, either express or implied. In no event shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.

Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM.

Not all offerings are available in every country in which IBM operates.

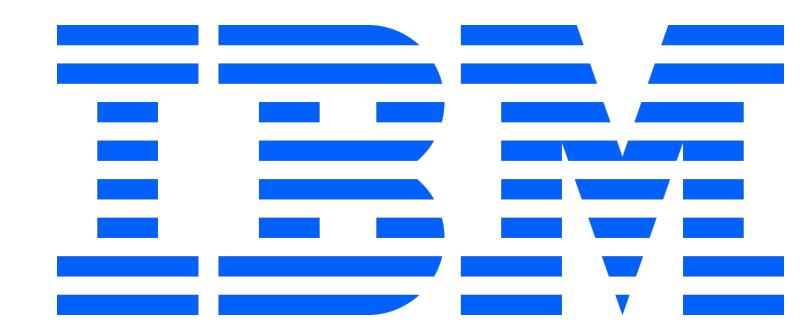
Any statements regarding IBM’s future direction, intent or product plans are subject to change or withdrawal without notice.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at: www.ibm.com/legal/copytrade.shtml.

Certain comments made in this presentation may be characterized as forward looking under the Private Securities Litigation Reform Act of 1995.

Forward-looking statements are based on the company’s current assumptions regarding future business and financial performance. Those statements by their nature address matters that are uncertain to different degrees and involve a number of factors that could cause actual results to differ materially. Additional information concerning these factors is contained in the Company’s filings with the SEC. Copies are available from the SEC, from the IBM website, or from IBM Investor Relations.

Any forward-looking statement made during this presentation speaks only as of the date on which it is made. The company assumes no obligation to update or revise any forward-looking statements except as required by law; these charts and the associated remarks and comments are integrally related and are intended to be presented and understood together.

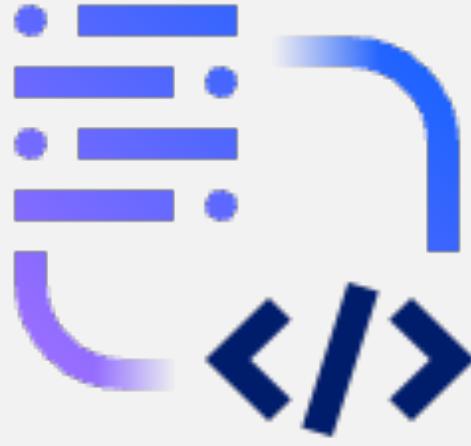


Appendix A:

watsonx Code Assistant products

IBM watsonx Code Assistant for Red Hat Ansible Lightspeed

Accelerate Ansible Playbook Creation with Generative AI



IBM watsonx Code Assistant for Red Hat Ansible Lightspeed helps automation teams create, learn, and maintain Red Hat Ansible Automation Platform content more efficiently.

Objectives

Increase productivity for automation developers

Expand who can create automation content

Extend trust in the automation content

Benefits

Increased time to value

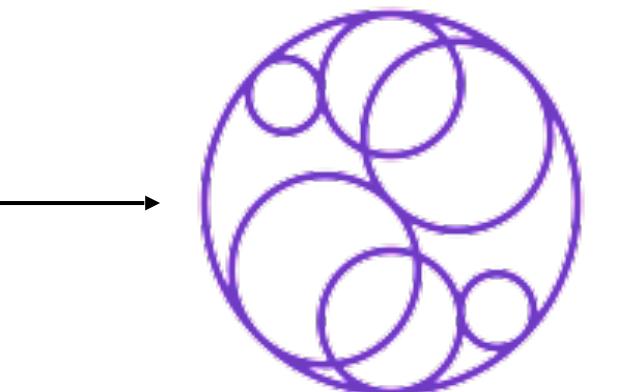
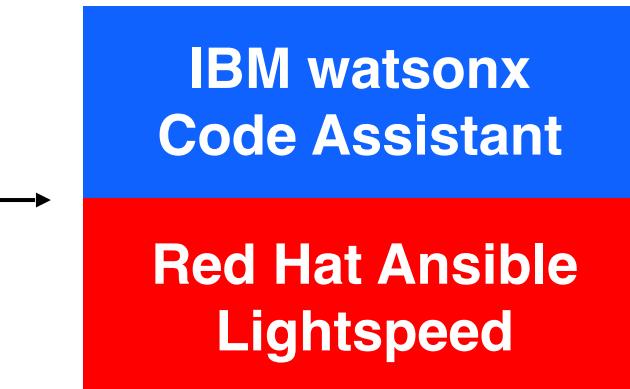
Reduced learning curve

Enhanced content quality

AI-powered Ansible content generation

```
local_dev > lightspeed_tp > deploy_monitoring.yml
1 --- 
2   - name: Deploy monitoring
3     hosts: monitoring
4     become: true
5
6     # module_defaults:
7     #   ansible.posix.firewall:
8     #     permanent: true
9
10    tasks:
11      - name: Include redhat.rhel_system_roles.ansible_builtin_include_role
12        name: redhat.rhel_system_roles.ansible_builtin_include_role
13
14      # - name: Copy files/cockpit.conf to /etc/cockpit/
15
16      # - name: Restart cockpit service
17
```

Ansible content creation

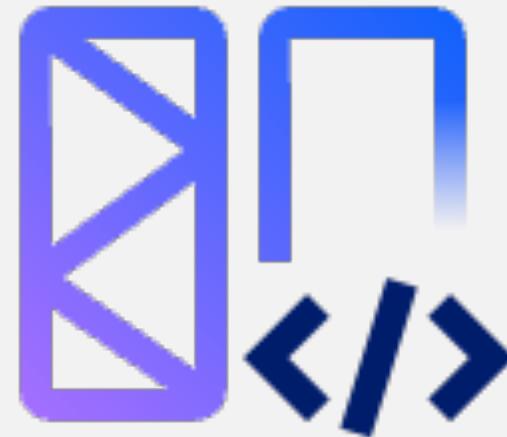


IBM Granite code model
for Ansible

- ✓ **Generate** high-quality Ansible content with IBM's purpose-built Granite code model
- ✓ **Accelerate** automation scripting with natural language to Ansible AI-assisted tasks
- ✓ **Personalize** content generation via prompt tuning of IBM Granite code model
- ✓ **Match** content with potential sources for trust and transparency
- ✓ **Explanation** for the purpose of an Ansible Playbook
- ✓ **Integrate** with popular IDEs starting with VS Code

IBM watsonx Code Assistant for Z

AI-assisted mainframe application modernization



IBM watsonx Code Assistant for Z helps modernize mainframe COBOL applications with capabilities that include application discovery, auto refactoring to business services, transformation of COBOL code to Java using generative AI, and auto-generated tests to validate new Java code, all while leveraging code explanation.

Objectives

Accelerate the application lifecycle

Create AI-generated high-quality Java code

Deliver flexibility and rich interoperability

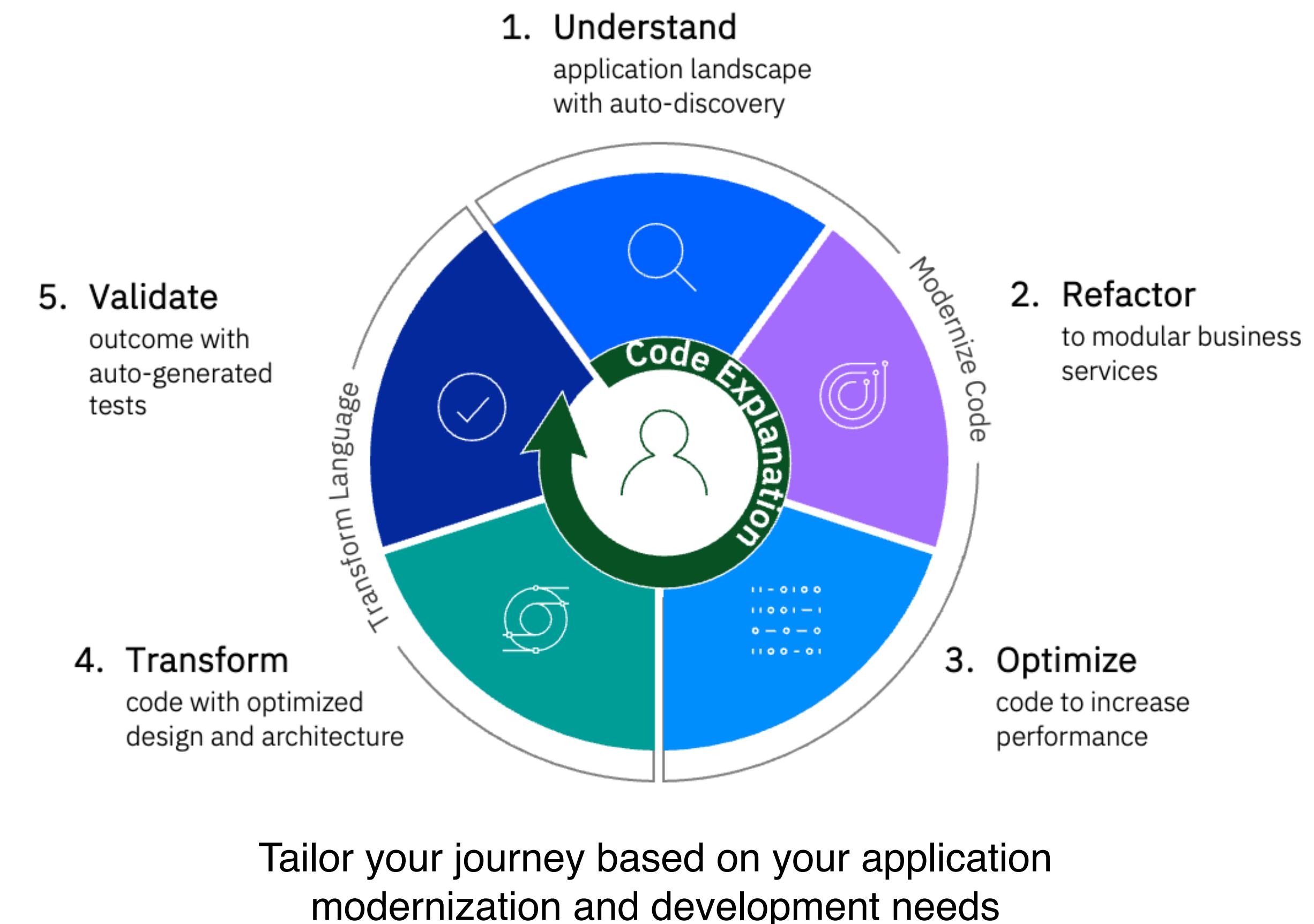
Benefits

Increased developer productivity

Greater business agility

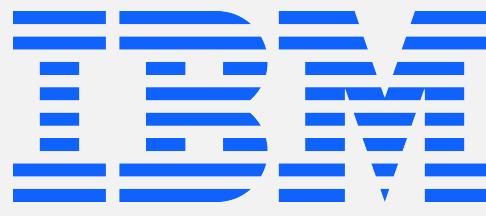
Expanded talent pool

Generative AI-assisted application lifecycle



IBM watsonx Code Assistant for Enterprise Java Applications

AI-assisted Java application modernization



IBM watsonx Code Assistant for Enterprise Java Applications leverages generative AI and automation to accelerate your Java application lifecycle and automatically transform your code to modernize Java applications.

Objectives

Gen AI-assisted Java development

Upgrade Java application code

Application runtime modernization

Benefits

Enhanced developer productivity

Streamlined application modernization

Improved code quality and manageability

Generative AI-assisted application lifecycle

1. Understand application and runtime environment



2. Plan based on analysis of your code

3. Transform code with optimized design and architecture

4. Validate outcome with auto-generated tests

5. Deploy with automate processes