

# How Generative AI Will Radically Change Application Development

Bern Elliot

© 2023 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates. This publication may not be reproduced or distributed in any form without Gartner's prior written permission. It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by [Gartner's Usage Policy](#). Gartner prides itself on its reputation for independence and objectivity. Its research is produced independently by its research organization without input or influence from any third party. For further information, see "[Guiding Principles on Independence and Objectivity](#)."

**Gartner**®

**AI-augmented development uses GenAI methods to aid software engineers in creating, testing, delivering and maintaining applications.**

# Topics

- AI Code Generation
- AI-Augmented Testing
- Design-to-Code
- GenAI — Changing the Nature of Applications
- Recommendations

# AI Code Generation

# **Generative AI tools are most useful when generating code in the following scenarios:**

01



Boilerplate  
code

02



Refactoring  
code

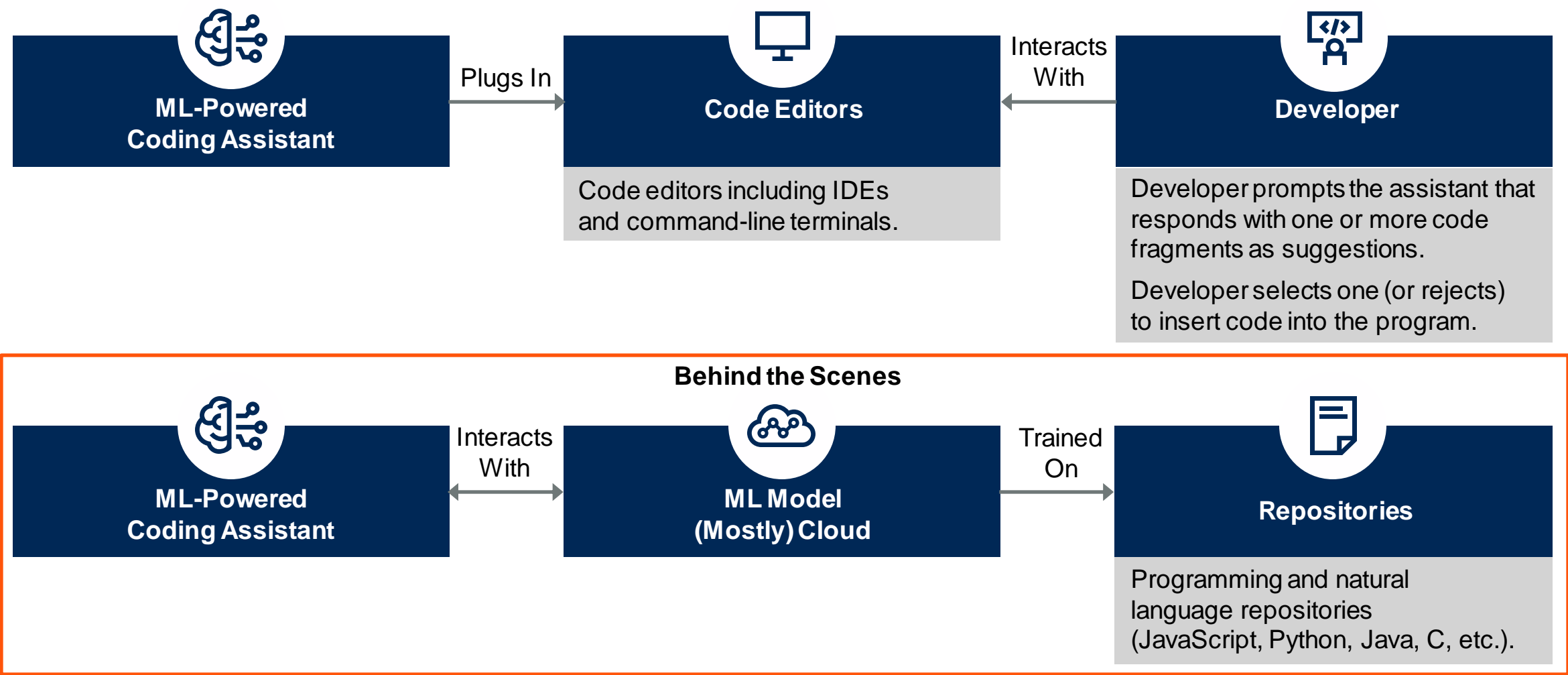
03



Working &  
learning  
older  
frameworks  
& languages

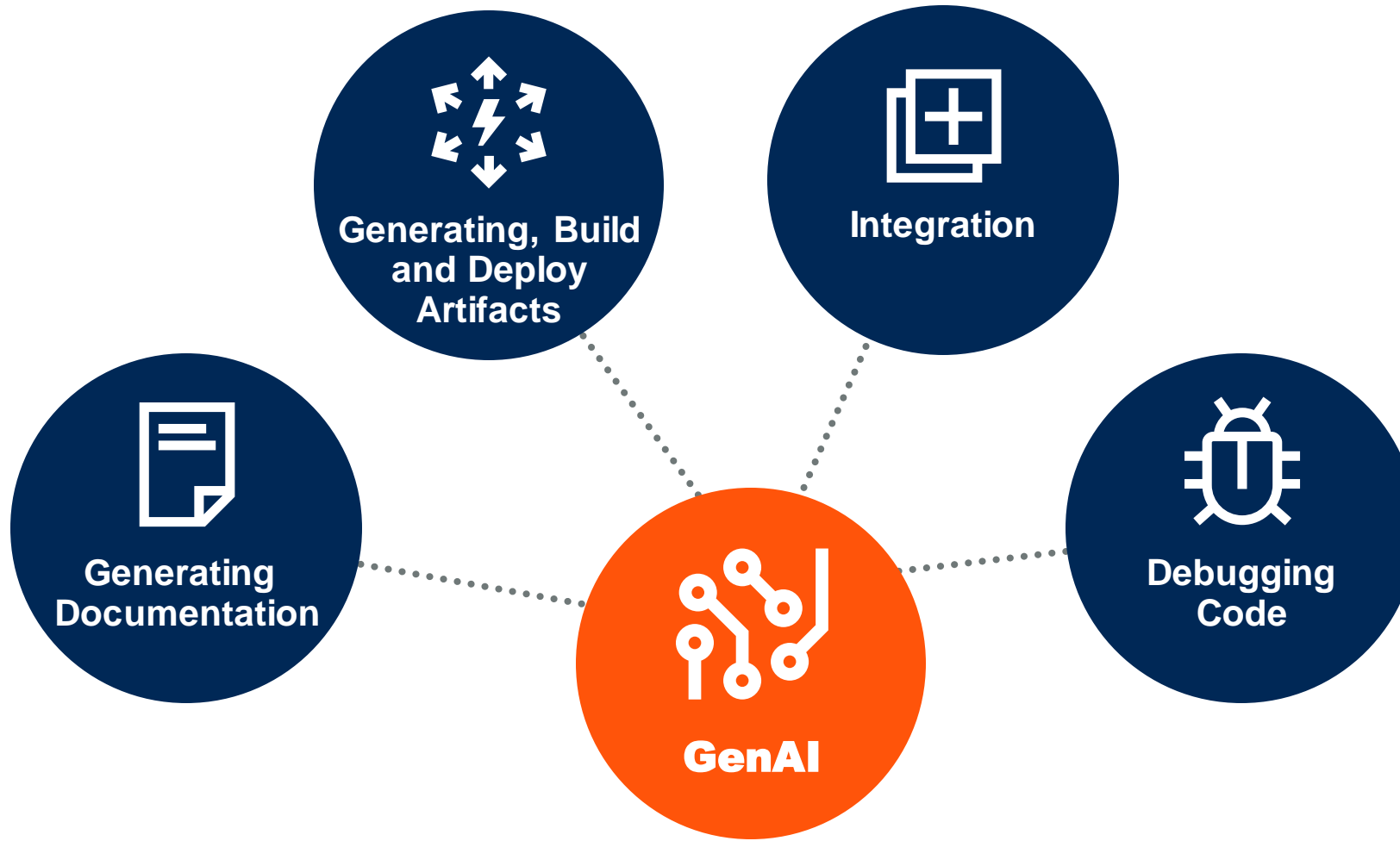
**Gartner®**

# How Developers Work With ML-Powered Coding Assistants



Source: Gartner  
777224\_C

# Other Areas Where You Benefit From Generative AI



# Generative AI Vendors — Software Engineering

## Code Generation:

Amazon CodeWhisperer,  
AskCodi, Codeium, GitHub  
Copilot, Replit, Tabnine

## Code Analysis:

Codiga, Codeball,  
Metabob

## Search:

CodePilot, Wizi AI

## Documentation:

Mintlify, Stenography,  
Thenéo

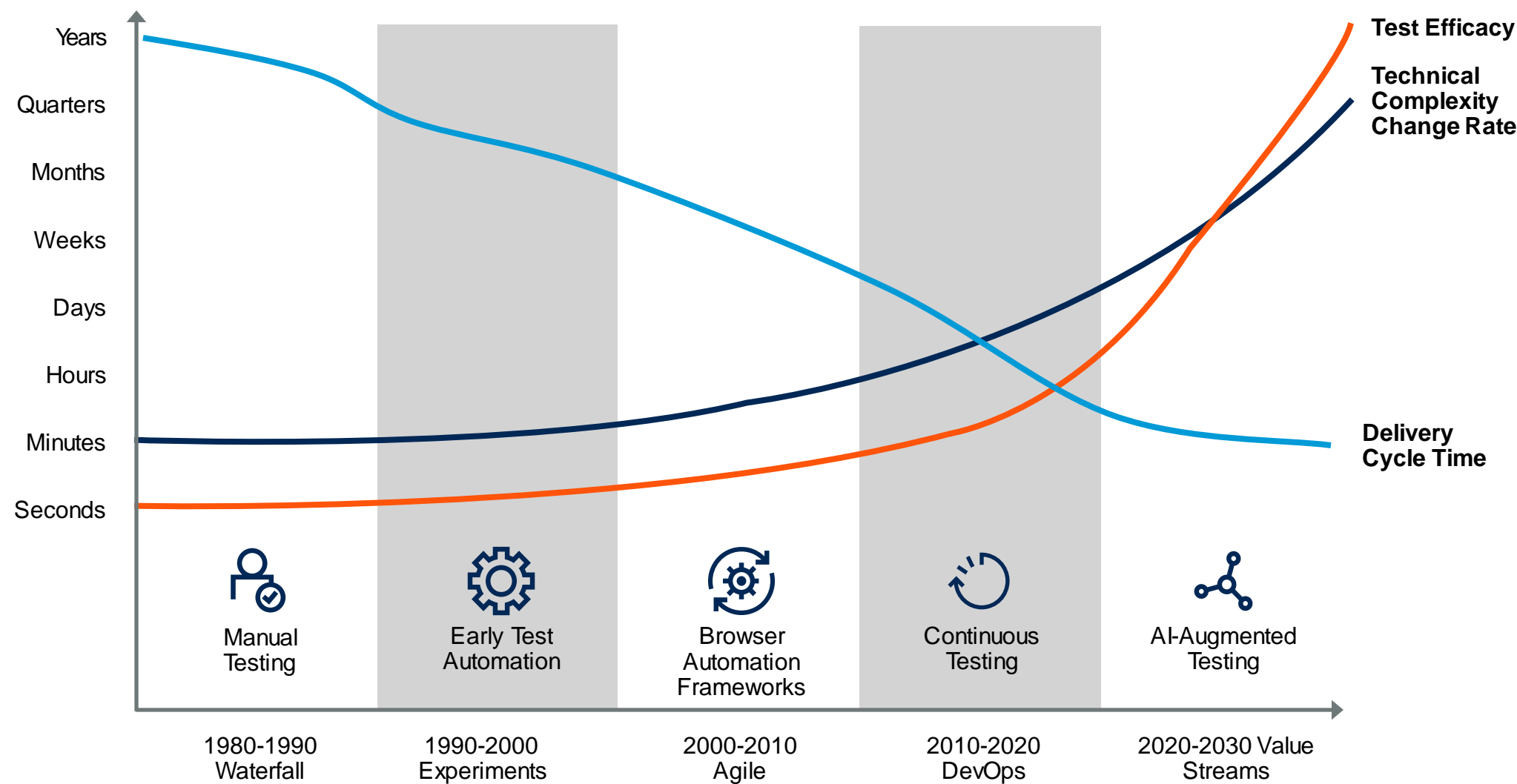
## DevOps Automation:

Kubiya, Red Hat Ansible  
Automation Platform/  
IBM (Red Hat)



# AI-Augmented Testing

# Evolution of Testing



Source: Gartner  
744939\_C

# AI-Augmented Testing

## AI-augmented software-testing tools:

- Improve software quality and reduce cycle times
- Provide early feedback about the quality of release candidates
- Improve consistency.

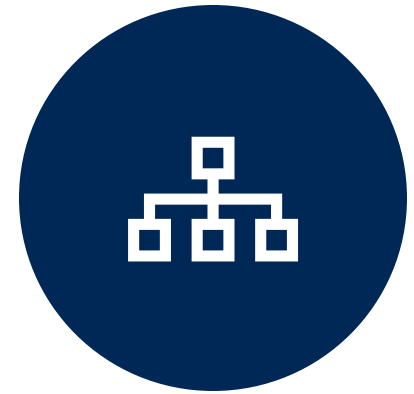
# Generative AI Models Can Help With Three Aspects of Testing



**Write  
Test Code**



**Generate  
Test Data**



**Generate  
Stub**

# Design-to-Code

# Design-to-Code Evolution

## Early Design-to-Code Tools

Styles, fonts, images, communicated as:

- Code snippets
- Redlines
- Optimized images

## Current Design-to-Code Tools

Platform assets from design systems deliver:

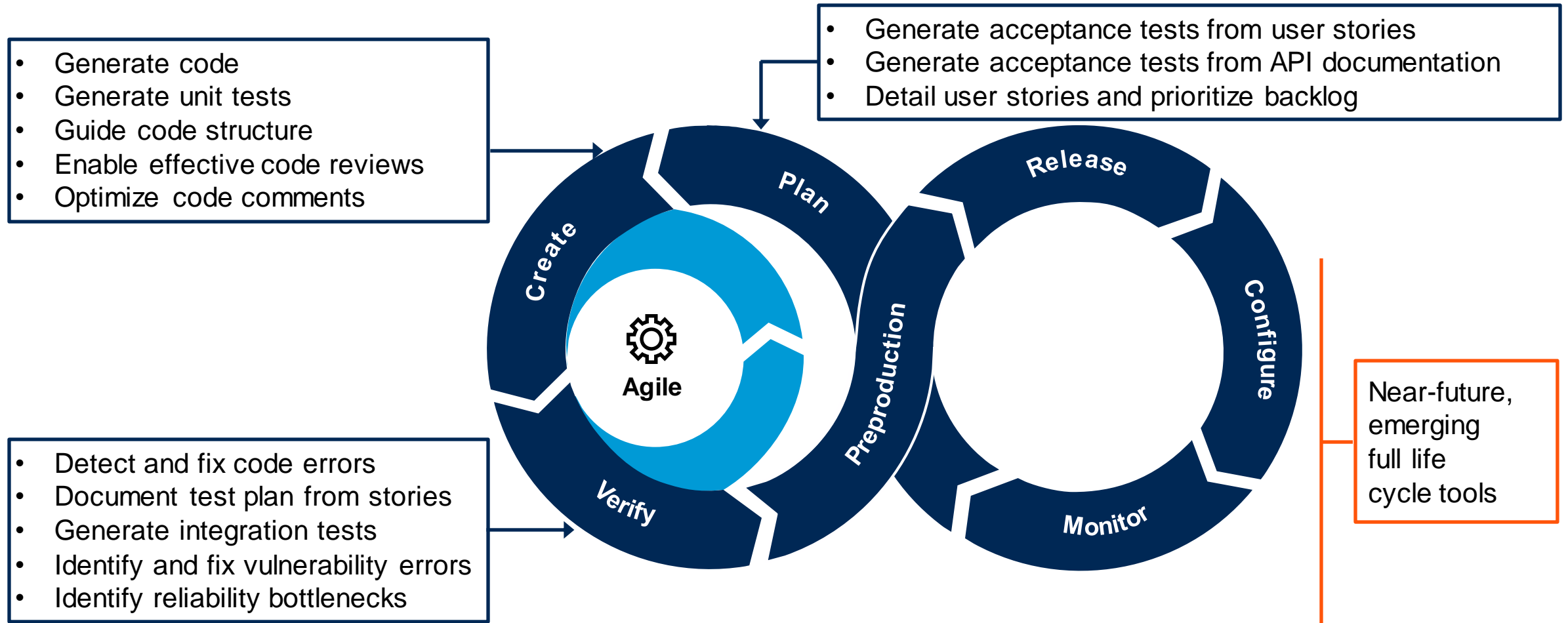
- Fully coded screens
- Minimal or no refactoring

## Future Design-to-Code Tools

Integrated design development environment offering:

- Custom designer and developer workflows
- Fully integrated design assets that facilitate use of code from software engineering platform

# Emerging Full Life Cycle GenAI Tools



Source: Gartner

795126\_C

15 © 2023 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates.

**Gartner**

# **Generative AI — Changing the Nature of Applications**



# Generative AI App Dev Disruptions



**Prompt  
Engineering  
Will Become  
a Critical Skill**

**Applications  
Will  
Become  
Conversational**

**Applications  
Will Become  
Proactive/  
Interactive**

**The UX  
Will Need  
a Redesign**

# Prompt Engineering as Programming Approach

## 1 Out-of-the-Box Model Usage e.g., ChatGPT

- Uses ChatGPT or other service as is via language prompt.
- Pro: Fast to market, limited investments, gain experience.
- Con: Limited differentiation, control range is limited.

## Prompt Engineering

- Tools to create, tune, and evaluate prompt inputs & outputs.
- Pro: Better targeted ChatGPT & GPT-3 results, low startup costs.
- Con: Must integrate with business systems to introduce data.

## 2 Leveraging Custom Models LLM Models

- Leverage (build/license) LLMs that may be fine-tuned.
- Pro: Customize or optimized models, data, parameters & tuning.
- Con: Requires added funding and skills. This is not **ChatGPT**.

# All Applications Will Become Conversational

- Conversational interactions are inherent to people.
- LLMs are inherently conversational.
- Conversation can be extended to applications.
- The release of Microsoft 365 Copilot and GenAI enhancements to Google Workspace apps will drive demand.



# Applications Will Be Proactive/Interactive

## Current State:

- User-initiated.
- Entry of data/parameters, use of controls, via the UI.
- Results are responses to inputs.

## Near-Term Proactivity:

- User will request information in English.
- Prompt engineering alters behavior of the application.
- Application will suggest additional information related to the initial request.

## Medium-Term Proactivity:

- Launch of application will trigger presentation of relevant changes since last login.
- When combined with event manager, application will notify user of important changes.

# The UX Will Need a Redesign

## Text + Visuals + Proactive:

- Reenvision user interactions.
- Applications will have agency.
- Some traditional UI controls.
- Some hide preselected prompts.



# Recommendations

- ④ Establish a generative AI COE to monitor industry developments and establish priorities for adoption of GenAI-enabled capabilities.
- ④ Define the appropriate use guidelines for the use of AI-enabled coding tools.
- ④ Evaluate and adopt AI-augmented testing tools.
- ④ Use design-to-code platforms to help enable agile development and design workflows.
- ④ Develop prompt engineering skills to enable the use of large language models.
- ④ Assume application requirements will include generative-AI-enabled interactions and proactive recommendations.

# Recommended Gartner Research

- 🔍 [The Impact of AI and ML on Software Engineering](#)  
Van Baker and Anthony Mullen
- 🔍 [Innovation Insight for Generative AI](#)  
Brian Burke, Arun Chandrasekaran and Svetlana Sicular
- 🔍 [Innovation Insight for Artificial Intelligence Foundation Models](#)  
Arun Chandrasekaran, Magnus Revang and Arnold Gao
- 🔍 [Quick Answer: How Can Generative AI Tools Speed Up Software Delivery?](#)  
Hema Nair, Bill Bloesen and Others



# Recommended Gartner Research

- 🔍 [AI Design Patterns for Knowledge Graphs and Generative AI](#)  
Afraz Jaffri, Leinar Ramos and Others
- 🔍 [Quick Answer: What Impact Will Generative AI Have on Search?](#)  
Stephen Emmott, Hao Yin and Others
- 🔍 [Quick Answer: Can We Use ChatGPT for Code Transformation and Modernization?](#)  
Stefan Van Der Zijden and Howard Dodd