

# Powering GenAI modernization

Transform complex Java and .NET applications into cloud-native systems at least **15x faster**



```

8 import org.springframework.web.bind.annotation.PathVariable;
9 import org.springframework.web.bind.annotation.PostMapping;
10 import org.springframework.web.bind.annotation.PutMapping;
11 import org.springframework.web.bind.annotation.DeleteMapping;
12 import org.springframework.web.bind.annotation.GetMapping;
13
14
15 @RestController
16 @RestController("/order")
17 public class OrderController {
18
19     @Autowired
20     OrderService orderService;
21
22     @Autowired
23     Logger logger;
24
25     @GetMapping
26     @RequestMapping("/{customerOrderId}")
27     public SalesOrder fetchSalesOrder(@PathVariable String customerOrderID) {
28         logger.log(this.getClass().getName());

```

**CLASS DOMAIN APPLICATION**

**DEAD CODE** Drift **TODO**

15-Jun- 9:41  
OrderController in Application

Class `OrderController` was not spotted during the learning although it has other dependencies. If it's no longer used, remove the class, or unmark it as dead code.

Effort Estimate: ?

Send to Copilot
 Send to Amazon Q
 Generate Prompt

Companies face a growing backlog of complex, monolithic applications that are difficult, risky, and expensive to refactor. However, lifting and shifting alone doesn't deliver the expected cloud benefits: cost reduction, agility, scalability, and the engineering velocity to compete in an AI-driven world.

Too often, modernization focuses on code and infrastructure, reducing licensing costs and fixing vulnerabilities, but misses the bigger opportunity. True elasticity, speed, and innovation require architectural modernization.

## Impactful modernization focuses on architecture

vFunction delivers AI-driven architectural modernization that powers GenAI code assistants like Amazon Q Developer and GitHub Copilot to transform Java and .NET monoliths into modular, scalable, microservices-based applications at least 15 times faster than traditional methods.

## How vFunction works

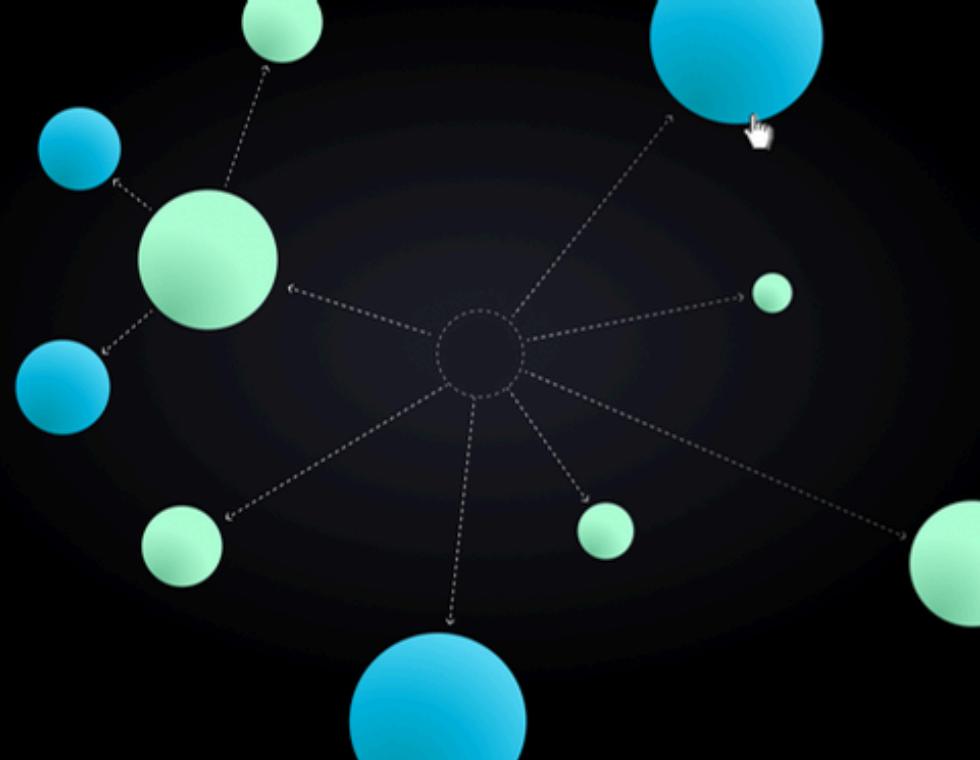
While other tools generate code for greenfield projects, vFunction cuts through the complexity of **brownfield applications**. For continuous modernization, vFunction:

- **Combines static and dynamic analysis with data science** to identify logical domains, components, and architectural context
- **Uncovers architectural technical debt** across massive codebases
- **Enables users to design the target state** of the modularized app and generate a "to-do" plan to achieve it
- **Provides relevant context to code assistants** for automated refactoring based on the "to-do" plan
- **Breaks monoliths into scalable cloud-ready services**

## Key benefits

- ✓ **Accelerates modernization**  
Refactors complex Java and .NET applications into modular, cloud-ready services
- ✓ **Visualizes architecture and enables DDD**  
Uses patented static and dynamic analysis combined with data science, integrated via the MCP server and IDE, to enable domain-driven design and uncover architectural technical debt
- ✓ **Guides and automates refactoring**  
Architectural intelligence powers GenAI workflows that help code assistants fix technical debt and break down monoliths into modular components.
- ✓ **Streamlines service transformation**  
Extracts services with speed and accuracy, generating APIs, and supporting framework upgrades made for the cloud.
- ✓ **Enables continuous modernization**  
Integrates with your SDLC to detect, monitor, and remediate architectural drift—keeping systems resilient, scalable, and aligned release after release.

TODO (34) Q		Expand TODO
DOMAIN CHANGES	15-Feb-2024 13:46	Drift <span>TODO</span>
ShippingScannerJob		
STATIC CLASS EXCLUSIVITY	15-Feb-2024 13:46	Drift <span>TODO</span>
Architectural Complexity has decreased from 3 to 1		
DEAD CODE	16-Feb-2024 13:46	Drift <span>TODO</span>
InventoryController in Application		
DEAD CODE	16-Feb-2024 13:46	Compat <span>TODO</span>
ProductController in Application		
CLOUD READINESS	16-Feb-2024 13:46	Compat <span>DONE</span>
com.oms.util.Logger		
DOMAIN CHANGES	16-Feb-2024 13:46	Drift <span>DONE</span>
DinersPayment Service		

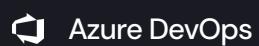


## Integrate architectural modernization into your SDLC

vFunction connects its architectural modernization engine to modern developer environments, enabling teams to query architectural issues, generate GenAI prompts, and trigger remediation directly from the command line. Leveraging vFunction's Visual Studio code extension or its MCP (Model Context Protocol) server, vFunction bridges the gap between architecture and code assistants, making architectural transformation fast, actionable, and fully embedded into your SDLC.

**With optimized support for Amazon Q Developer and GitHub Copilot**, teams can resolve complex issues like circular dependencies, god classes, resource exclusivity and domain entanglement using prompts enriched with run-time architectural data.

**Through integrations with Jira and Azure DevOps**, vFunction pushes prioritized architectural to-dos directly into your backlog, ensuring that nothing gets lost between discovery and execution while automatically validating that these to-dos are remediated.



## Examples of issues vFunction helps resolve with GenAI workflows



Refactor domain dependencies and cross-domain calls



Simplify and split "god" classes



Untangle circular flows and domain-level loops



Split classes used differently across multiple domains



Remove dead code



Resolve circular library dependencies



Upgrade aging frameworks and outdated libraries



Transform Apache Struts Forms to REST

## Fully funded application modernization

Modernizing complex applications doesn't have to be a budget blocker. vFunction partners with leading cloud providers through a range of programs—including Proof of Concept (POC) funding and modernization acceleration programs to reduce risk and accelerate cloud native adoption.

Through programs like the [AWS ISV Workload Migration Program](#), qualified organizations can receive **fully funded vFunction licenses** to jumpstart stalled migrations and transform Java and .NET monoliths into modular services on cloud platforms like AWS Lambda, EKS, ECS with Fargate, and beyond.

vFunction also works closely with other leading cloud providers to support application modernization across a variety of environments.



Unblock stalled migrations



Adopt advanced cloud services (e.g. serverless) and agentic capabilities



Modernize faster with no upfront license cost



Microsoft Azure

Google Cloud

ORACLE  
Cloud Infrastructure

Trusted by engineering teams to modernize their most critical applications



"Without vFunction, we never would have been able to manually tackle the issue of circular dependencies in our monolithic system."

Martin Lavigne

R&D Lead, Trend Micro®



15x

faster full application modernization

[Global financial leader](#)

2X

increase in cloud ROI

[CDL Insurtech leader](#)

90%

decrease in deployment time

[Trend Micro](#)

Showing 30 TODOS (7 Filters Applied)

	Priority	Date	Event	Details
▼	!/m	1	15-Jun-2025 09:41	Dead Code ShippingService
Class <code>ShippingService</code> is a dependency of class <code>com.oms.service.ProductService</code> and while class <code>\$com.oms.service.ProductService</code> was seen to be running in the context of domain <code>ProductController</code> , Class <code>ShippingService</code> wasn't and should be considered as dead code.				
Effort Estimation: Medium				
Baseline: eldad - Analysis Completed - 05 May 2025 08:56 Measurement: Scheduled - 27-May-2025 11:24 (UTC+00:00)				

**Copy Prompt**

The diagram illustrates a distributed system architecture. A central black circle represents a hub, with several green circles representing different services connected to it via dashed lines. The services labeled are: ShippingPriceCo... (top left), OrderController (top left), ModifyFulfillLine... (top right), PaymentService (top right), InventoryContro... (bottom right), ProductController (bottom center), and StoreSearchCont... (bottom left). This visualizes the interconnected nature of modern distributed applications.

## Visualize, modernize, and govern your distributed applications on one platform

Beyond monoliths, vFunction reduces architectural technical debt and modernizes complex distributed environments. By leveraging OpenTelemetry tracing and supporting a variety of technology stacks, vFunction provides deep architectural insights to remove architectural technical debt that threatens the resiliency and scalability of your distributed system.

Addressing technical debt in distributed systems enables seamless cloud migration. vFunction's governance capabilities prevent sprawl, enforce service boundaries, and monitor architectural drift, so your distributed architecture stays resilient and scalable.

Ready to tackle complexity and scale modernization?  
Visit [vFunction.com](https://vFunction.com) to get started.

[Get started](#)

Trusted by:



## About vFunction

vFunction® is the pioneer in powering GenAI modernization, transforming complex apps into cloud-native systems. Built to solve one of the toughest challenges in software today, vFunction enables code assistants to turn Java and .NET monoliths into modular, microservices-based applications—at least 15x faster than traditional approaches. vFunction also supports the architectural modernization of complex distributed systems. Global system integrators and top cloud providers partner with vFunction to help leading companies like Intesa Sanpaolo, CDL, and Trend Micro modernize their applications—accelerating innovation and amplifying business impact. vFunction is headquartered in Menlo Park, CA, with offices worldwide. To learn more, visit [www.vFunction.com](https://www.vFunction.com)