

# Magic Quadrant for Technical Debt Management Tools

20 May 2026 - ID G00841444 - 35 min read

By Tigran Egiazarov, Howard Dodd, [and 1 more](#)

Rapidly accumulating technical debt reduces quality, predictability, and delivery speed, limiting innovation and AI scale across the SDLC. Use this Magic Quadrant to identify vendors that align with your strategy for measuring and monitoring technical debt at both the architectural and code levels.

## Strategic Planning Assumption

By 2027, architectural technical debt will account for 80% of all technical debt.

## Market Definition/Description

Gartner defines technical debt management tools as software solutions that analyze source code, architecture and dependencies to identify, visualize and prioritize technical debt, structural flaws, and security risks. Delivered via SaaS or self-managed models, these solutions utilize static and dynamic analysis — often augmented by AI — to provide actionable insights for remediation, automate code refactoring, and accelerate cloud migration or modernization initiatives.

Technical debt management tools provide a structured environment for identifying, measuring, and monitoring the costly structural and security compromises within software applications. These tools offer automated analysis at both the code and architectural levels, effectively revealing risks such as defects, “code smells,” dead code, and architectural drift from established best practices. By abstracting the complexities of

manual code reviews and dependency mapping, technical debt management tools enable product teams to maintain long-term delivery speed and application quality.

The technical debt management market reflects a consolidation of technologies across static and dynamic analysis, software composition, architecture observability to streamline the remediation process. While AI code assistants are increasingly effective at remediating code-level debt, the market is shifting toward managing architectural technical debt — debt that cuts across multiple systems or architecture layers, which is expected to account for 80% of all technical debt by 2027. These tools are essential for businesses aiming to achieve excellence in software engineering and prevent the “breaking point” where accumulated debt leads to unstable performance and soaring maintenance costs.

Technical debt management tool benefits include:

- **Improved agility and predictability:** By providing early visibility into architectural compromises, these tools prevent time-consuming rework and ensure teams can adapt quickly to new business opportunities.
- **Operational excellence:** These tools monitor for security vulnerabilities and performance bottlenecks, allowing organizations to focus on innovation rather than constant “firefighting” and unplanned maintenance.
- **Enhanced developer experience:** By reducing cognitive load across code and architecture, these tools improve developer satisfaction, lower onboarding time, and help maintain acceptable risk and cost when building modern solutions.
- **Strategic prioritization:** Through technical debt scoring and financial impact modeling, these tools allow leaders to incorporate debt remediation directly into product roadmaps alongside new customer features.

## **Mandatory Features**

The mandatory features for this market focus on providing visibility across both the architectural and code levels to prevent application “breaking points:”

- **Software architectural technical debt analysis:** Tools must provide high-abstract-level visibility into system dependencies using dependency graphs. This includes architecture complexity indicators and metrics panels to identify tightly coupled modules and architectural drift from established best practices.

- Technical debt visualization: Tools need to provide dashboards or reports that visualize and prioritize tech debt, often using metaphors like hot spots or heatmaps to highlight critical issues.
- Static code analysis: Tools must have the capability to automatically detect defects, code smells, and dead code. They must also analyze test coverage to identify insufficient verification of business logic, which increases feature-delivery lead time.

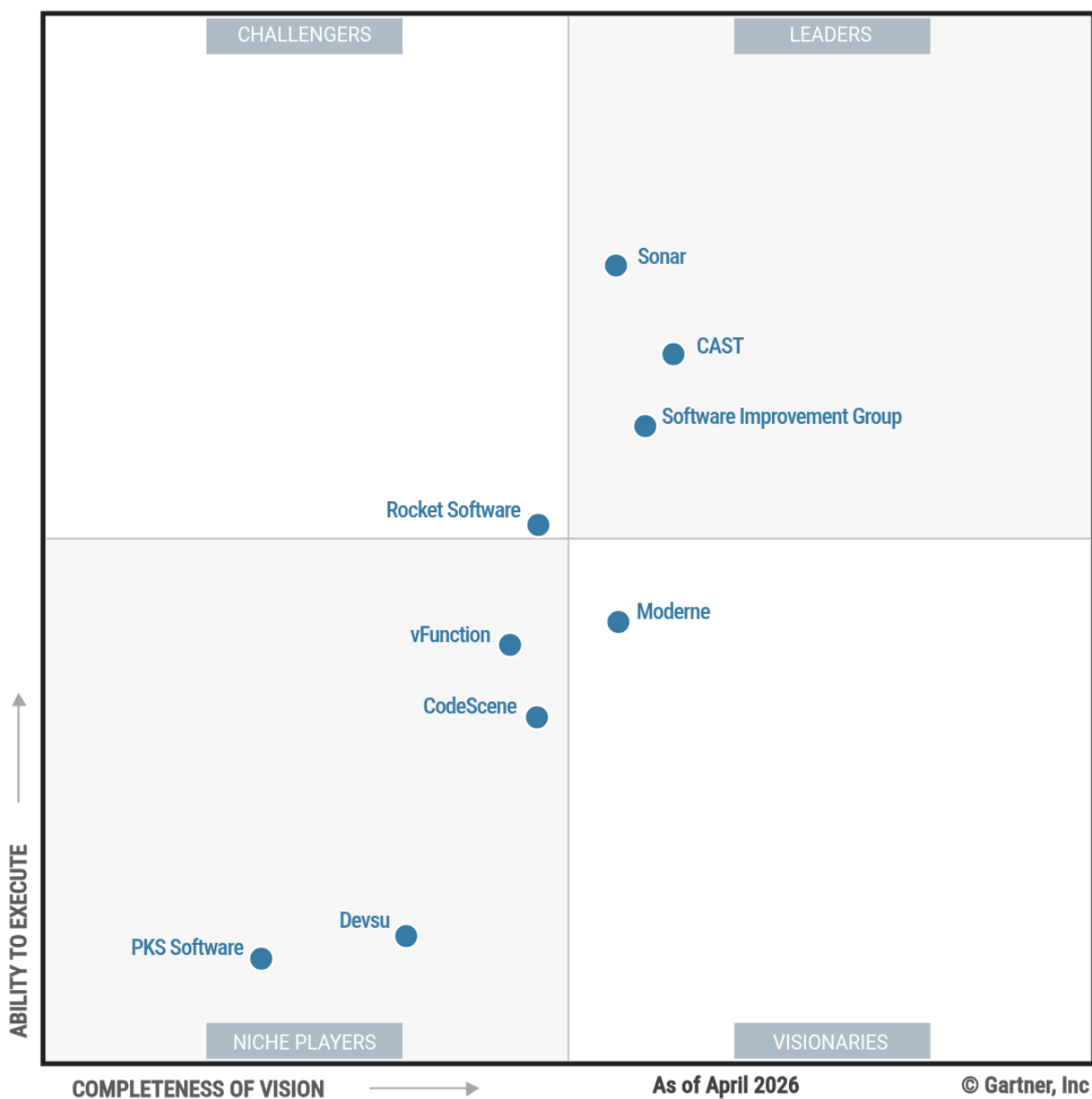
## Common Features

- Software composition analysis (SCA) of technology stack: Essential for monitoring the building blocks of an application, including outdated frameworks, unsupported technology stacks, and open-source libraries with known vulnerabilities.
- Static Application Security Testing (SAST): Ability to detect and report on compliance with various security measurement frameworks — for example OWASP, CERT, SSDF, hard-coded credentials, and hard-coded keys.
- Technical debt scoring and financial modeling: The ability to evaluate and score technical debt items (often using the SQALE method) and convert remediation costs into tangible financial terms to help leadership prioritize work on product roadmaps.
- Performance monitoring and observability integration: Capturing production telemetry (logs, metrics, and traces) through dynamic analysis to identify runtime symptoms of debt, such as unstable performance or reliability issues.
- AI-powered technical debt intelligence: This includes AI-augmented remediation suggestions for architectural debt and context-aware refactoring.
- Architectural modernization guidance: Intelligent recommendations specifically for monolith-to-microservices decoupling or identifying optimal candidates for coupling/decoupling to enhance scalability.
- Time-series technical debt visualization and reporting: Tracking and monitoring the progression of debt over time across an entire application portfolio to evaluate the success of long-term remediation efforts.
- Continuous integration and continuous delivery (CI/CD) integration: Tools must seamlessly integrate into DevOps platforms and CI/CD pipelines to measure technical debt continuously as part of the software development life cycle

- SaaS vs. on-premises delivery models: The flexibility to choose between SaaS for ease of maintenance or on-premises/containerized deployment for organizations with high security requirements regarding proprietary code and secrets.
- IDE integration: Extensions that provide real-time checks and feedback directly in the developer's environment to prevent technical debt as code is written.

## Magic Quadrant

Figure 1: Magic Quadrant for Technical Debt Management Tools



## Vendor Strengths and Cautions

### CAST

CAST is a Leader in this Magic Quadrant. Its offerings include CAST Imaging, CAST Highlight, and CAST Gatekeeper, which collectively provide source-code-based structural analysis, architectural technical debt detection aligned with software quality standard ISO 5055, and portfolio-level application risk and modernization insights. The platform supports code and architecture analysis across large multistack application portfolios and can be deployed as SaaS or self-managed on-premises.

CAST operates globally, and its customers tend to be large enterprises in regulated industries managing complex legacy systems. During the past year, CAST has increased investment in AI integration via Model Context Protocol (MCP) servers, positioning its architectural data for consumption by AI coding assistants and agentic technical debt remediation workflows.

#### *Strengths*

- **Product or service:** CAST delivers a broad and technically deep portfolio for managing technical debt across individual applications and application portfolios. Its products enable customers to assess architectural integrity, identify structural risks, and prioritize remediation using consistent, standards-aligned metrics. CAST's products are particularly effective for large enterprises with diverse, legacy-heavy application estates.
- **Innovation:** CAST is extending architectural intelligence into agentic software engineering workflows. By exposing detailed dependency, call-flow, and technical debt insights through protocols such as MCP, CAST enables AI tools and agents to consume deterministic architectural context. This innovation differentiates it from competitors' code-centric analyzers and supports emerging AI-driven modernization and remediation use cases.
- **Business model:** CAST's subscription-based pricing and partner ecosystem have enabled the company to build long-term relationships with large-enterprise customers who have complex application portfolios. This business model supports adoption of CAST products across these enterprises, enabling customers to detect and manage technical debt at scale.

#### *Cautions*

- **Customer experience:** CAST's product portfolio is not a cohesive platform but rather a loosely integrated collection of tools, with core capabilities spread across multiple products. Buyers should plan to invest notable time and expertise to configure and integrate these products, which can extend implementation timelines, delay time to value, and lead to inconsistent adoption across teams.
- **Offering (product) strategy:** CAST's business-level mapping is currently limited in scope. While its products support risk weighting at the code and architecture level, they have limited capabilities for business impact modeling.
- **Market understanding:** CAST does not natively support every phase of the technical debt management life cycle, as it relies on integrations with AI coding agents for technical debt remediation and requires other enterprise tools for technical debt tracking against business impact and revenue.

## CodeScene

CodeScene is a Niche Player in this Magic Quadrant. The CodeScene platform supports technical debt management and engineering intelligence, with capabilities for code health analysis, architectural change-coupling and delivery analytics. It integrates with DevOps platforms, integrated development environments (IDEs) and AI-assisted development tools. CodeScene can be deployed as SaaS or as a self-managed on-premises or private cloud installation.

CodeScene operates primarily in Europe and serves midmarket and enterprise customers. During the past year, it has expanded support for AI-assisted and agentic development through MCP-based integrations, allowing AI coding agents to discover and fix code health issues.

### *Strengths*

- **Product or service:** CodeScene differentiates its product by correlating code structure, change coupling, and commit history — revealing code health and security risks that static analysis alone cannot surface. CodeScene's integrations into continuous integration/continuous delivery (CI/CD) pipelines, IDEs, and AI-assisted development workflows enable it to deliver technical debt insights.
- **Market responsiveness:** CodeScene has responded to trends such as AI-assisted coding and governance concerns by extending its analytics to support AI readiness

and agentic workflows. CodeScene's capabilities complement existing developer tools and new AI coding agents while preserving deterministic analysis and auditability.

- **Innovation:** CodeScene is the only vendor in this Magic Quadrant that offers analysis of the relationship between a codebase and the developers working on it, such as change coupling, knowledge concentration and offboarding simulations. These features expose team dependencies and organizational fragility, helping engineering leaders to proactively reduce delivery risks and continuity gaps.

### *Cautions*

- **Sales execution:** CodeScene relies exclusively on direct sales, with no sales partners and limited third-party marketplace integrations. While this approach ensures a direct connection with clients and can improve client experience, it will limit the company's ability to expand quickly and reach new customers across regions.
- **Geographic strategy:** CodeScene's operations are concentrated in Europe, with limited presence in other regions. Multinational customers may encounter constraints related to localized sales coverage, regional procurement support, or in-region customer success resources.
- **Offering (product) strategy:** The CodeScene platform links code health scores to engineering outcomes and AI readiness. However, it has limited support for technical debt tracking against business impact and revenue.

### **Devsu**

Devsu is a Niche Player in this Magic Quadrant. Devsu's offering is Velx, which supports expert-led analysis, remediation planning, and modernization execution. Velx focuses on architecture- and system-level analysis, and is primarily delivered through forward-deployed engineering teams in a structured approach that enhances accuracy, governance and execution quality.

Devsu operates with a geographically distributed delivery model and supports customers across multiple regions and industries. During the past year, Devsu has invested in architecture discovery leveraging C4 standards, agentic remediation workflows, and enhanced analysis and reporting features.

### *Strengths*

- **Innovation:** Velx's primary innovation is creating an intelligence layer that maps architecture, dependencies, and technical debt for AI-assisted remediation. Alongside its forward-deployed engineering model, this approach helps customers leverage the Velx platform for technical debt management and modernization.
- **Offering (product) strategy:** Devsu differentiates Velx from episodic and static analyzers by providing an engine for ongoing architectural insight. These insights enable customers to understand how structural debt constrains delivery velocity, reliability and cloud readiness over time, so they can continuously prioritize modernization efforts and debt remediation.
- **Market understanding:** Devsu recognizes that many organizations lack the necessary specialized expertise to manage technical debt on their own. It has designed Velx around customizable workflows supported by forward-deployed engineering teams. This approach is well-suited to customers who need to remediate technical debt but lack the internal skills or architectural expertise to do so on their own.

### *Cautions*

- **Business model:** Devsu's Velx platform was productized from an engineering service platform in 2025 and is in active commercialization. As such, the long-term viability of Velx has yet to be proven. Devsu remains largely dependent on its professional services to generate revenue, and its partner ecosystem is relatively small compared to competitors.
- **Customer experience:** Devsu operates a structured and multilayered customer success model (CSM) embedded within its delivery organization rather than a stand-alone CSM function. Defined roles ensure clear accountability for account health, delivery governance and technical outcomes. However, this model can limit direct customer visibility into system understanding, architectural decisions and outputs, potentially increasing future transition complexity.
- **Product or service:** Within the technical debt management landscape, Velx offers capabilities that connect architecture intelligence to execution workflows, including AI-driven system reconstruction, multirepository contextual reasoning, and generation of implementation plans. However, these capabilities are emerging and primarily oriented toward modernization rather than core technical debt management, limiting Velx's differentiation in this segment.

## Moderne

Moderne is a Visionary in this Magic Quadrant. It focuses on AI-assisted code modernization and large-scale refactoring, with an emphasis on policy-driven change, organizational standards enforcement and agent efficiency through deterministic tools. It enables enterprises to define and operationalize engineering standards across teams and applies AI analysis to source code. Moderne offers SaaS, on-premises, and secure delivery options capable of operating using customer-provided AI tokens.

Moderne's operations are based in North America, and its customers are primarily large organizations in the software, banking and retail industries. Moderne is the commercial steward of OpenRewrite, using it as the source of its deterministic refactoring recipes while translating community-driven demand into reusable solutions (recipes) for consistent, enterprise-scale automation.

### *Strengths*

- **Product or service:** Moderne's platform supports large-scale, AI-driven code analysis and transformation. It allows customers to codify and enforce organizational standards into reusable "recipes" that function as centrally governed policies. Moderne's platform closely aligns with how developers reason about codebases, as it anchors software composition analysis (SCA) and transformation logic to file and folder structure. Moderne also excels in multirepo changes, such as replacing log4j.
- **Innovation:** Moderne's platform enables AI agents to use MCP-exposed tools to explore and analyze large codebases, evaluate modernization options, and compose modernization plans by selecting and combining OpenRewrite recipes. Then, Moderne's refactoring engine distills these outputs into low-cost, deterministic transformations from solutions derived from OpenRewrite. This innovative hybrid AI model enables customers to modernize and transform systems at scale.
- **Market understanding:** Moderne supports a composite view of customer-defined value that separates engagement, outcomes, loyalty and service experience metrics, rather than relying solely on activity-based metrics. This design reflects an understanding of how large organizations measure engineering productivity and modernization value.

### *Cautions*

- **Offering (product) strategy:** Moderne lacks explicit frameworks for prioritizing architectural, code, or risk-based technical debt, as it primarily focuses on addressing code and technology-stack levels of technical debt. Buyers looking to remediate architectural technical debt should be prepared to supplement with other technical debt management tools.
- **Sales execution/pricing:** Moderne's platform requires customers to provide their own AI tokens. While this offers flexibility and avoids vendor lock-in at the model level, customers are responsible for managing AI infrastructure, costs and governance.
- **Customer experience:** Moderne does not have a customer advisory board, and the company performs limited analysis of customer journey friction points. These gaps restrict systematic feedback loops.

## PKS Software

PKS Software is a Niche Player in this Magic Quadrant. Its platform, eXplain, focuses on deterministic analysis of mainframe-centric environments and related legacy ecosystems. It specializes in deep static analysis of mainframe code and associated technologies, producing a point-in-time, queryable representation of applications, dependencies, database interactions, and data flows. eXplain uses a client/server architecture and can be hosted in the cloud, or accessed through web browsers or IDEs.

PKS Software operates primarily in Europe, with a growing presence in North America, and serves customers largely in regulated, legacy-intensive industries such as banking, insurance, logistics, and utilities. During the past year, PKS Software delivered mostly incremental updates, including minor tooling support and integration improvements, rather than substantial product innovation.

### *Strengths*

- **Product or service:** PKS Software's eXplain consolidates code, APIs, configuration, and runtime data into a centralized system model, enabling powerful querying and analysis across architectural, technical, and business abstraction layers.
- **Innovation:** PKS Software's primary innovation is business capability clustering, driven by static and dynamic code-call analysis. Programs are grouped into technically cohesive units based on how they invoke and depend on one another, revealing logical application and business domains. This clustering enables decomposition and

prioritization of large monoliths and supports modernization roadmapping where architectural boundaries are poorly documented.

- **Market understanding:** PKS Software's platform capabilities demonstrate a strong understanding of mainframe and IBM Power I (AS/400) environments, where accuracy, determinism, and explainability outweigh speed or automation. Its design helps enterprises address knowledge loss, regulatory scrutiny, and risk associated with undocumented legacy logic.

### *Cautions*

- **Offering (product) strategy:** PKS Software's product strategy is narrowly focused on mainframe and associated technology analysis rather than end-to-end application modernization. The offering emphasizes static insight generation and does not enable execution, automation, or continuous improvement workflows.
- **Operations:** eXplain lacks native integration with a software development life cycle (SDLC) but can be fully integrated via documented APIs for use in schedulers, pipelines, and automated workflow processes. Its IDE support is limited to Eclipse, VS Code or a web browser. The platform does not support centralized administration or provide a monitoring dashboard for shared enterprise governance, reducing its suitability for large, federated organizations.
- **Business model:** As a subsidiary of the TIMETOACT GROUP, PKS Software is positioned more as a services-adjacent analytical offering than a fully productized modernization platform. Buyers should expect to augment PKS Software with additional tools to support governance, tracking, execution, and modernization automation.

### **Rocket Software**

Rocket Software is a Challenger in this Magic Quadrant. Its portfolio of products, including Enterprise Analyzer and COBOL Analyzer, deliver technical-debt-related capabilities supporting application modernization, operations, and life cycle management, with particular strength in mainframe and legacy-centric environments. Rocket embeds debt analysis in long-term modernization and operational continuity programs for mission-critical systems. Its products can be deployed in the cloud or self-managed on-premises.

Rocket operates globally, with a strong presence across North America and Europe, and its customers are mainly multinational enterprises. During the last year, Rocket has added code analysis and AI features that enhance predictive detection of performance issues, architectural insight, and impact analysis.

### *Strengths*

- **Market responsiveness:** Rocket has established integrated, multichannel customer insight and support models, including onboarding, advisory and sustained enterprise support, that surface challenges for new and long-standing customers. This feedback approach enables Rocket to deliver stability, predictable delivery, and risk-managed modernization.
- **Market understanding:** Rocket's suite of products focuses on identifying and remediating technical debt within highly complex, entrenched legacy and mainframe environments. This pragmatic understanding of technical debt as an operational constraint aligns well with enterprises that prioritize controlled modernization over aggressive refactoring or legacy platform replacement.
- **Offering (product) strategy:** Rocket provides a range of complementary tools that allow customers to address technical debt, modernization, and operations through a single vendor relationship and procurement model. Rocket's support for open-source initiatives such as Zowe reinforces its alignment with the needs of organizations seeking to modernize mainframe environments while preserving existing platforms and skills.

### *Cautions*

- **Product or service:** Rocket's technical debt capabilities are distributed across multiple products rather than delivered as a unified, purpose-built platform. While its cloud and distributed offering are highly integrated with modern SDLC practices, its mainframe-focused products are not. They offer limited support for fine-grained debt prioritization, remediation modeling, and developer-centric workflows.
- **Innovation:** Rocket has been relatively slow to release innovative features for technical debt management, being more focused on AI-assisted modernization for mainframe and hybrid environments. Its product features lag competitors in delivering AI-driven analysis, agentic workflows, and advanced architectural intelligence.

- **Business model:** Rocket's business model and technical debt management capabilities span cloud and distributed environments but remain heavily centered on mainframe and adjacent legacy technologies. While this focus supports a loyal installed base, it limits feature depth and flexibility as buyers expand technical-debt-management efforts into broader cloud-native use cases.

## Software Improvement Group

Software Improvement Group (SIG) is a Leader in this Magic Quadrant. Its flagship offering, Sigrid, provides standards-based assessment of software quality and technical debt at the architecture and portfolio levels. SIG offers broad support for more than 300 technologies and emphasizes explainable metrics, longitudinal insight, and decision support. Sigrid can be deployed as SaaS or self-managed on-premises.

SIG has a well-established presence in Europe, with an expanding U.S. presence and global reach. Its customers are mainly large enterprises in regulated industries. During the past year, SIG has delivered integrations with AI-assisted workflows and strengthened business insight reporting in Sigrid.

### *Strengths*

- **Innovation:** SIG's architectural modeling, metric tracking over time, and explainable technical debt metrics align to multiple industry standards like ISO 25010, OWASP and ISO 5055, driving confidence in objective measures. These measures are a strong fit for organizations that require stable, auditable, and defensible assessments to support compliance, risk management, and regulatory oversight.

Market understanding: Sigrid treats technical debt as an economic and governance challenge, providing business capability value and impact views that engineering leaders need to operationalize debt management. This focus on decision support will appeal to buyers in enterprises that value longitudinal analysis and compliance-oriented quality management.

- **Product or service:** Sigrid integrates software quality analysis into CI/CD pipelines, governance workflows, and audit processes, and modern IDE's, MCP and agent workflows, enabling continuous tracking of risk signals. This appeals to enterprise buyers who prioritize oversight, life cycle decisions and business capability alignment over speed of tactical debt remediation.

### *Cautions*

- **Offering (product) strategy:** SIG's platform has limited native support for remediation execution, workflow orchestration, or developer-centric tooling. Customers will need to procure and integrate complementary tools or services for end-to-end remediation.
- **Market responsiveness:** SIG's platform roadmap has started to address the limited integration of AI-assisted development and agentic workflows into its platform compared to its competitors. As a result, domain-specific guidance and remediation suggestions may lag more innovative products, increasing integration effort for organizations adopting AI-driven engineering tools.

**Business model:** SIG's business model combines platform subscriptions with advisory and deployed engineering teams to deliver immediate, client-specific value through custom configurations and integrations. As the platform capabilities evolve, it becomes increasingly important to actively engage buyers on the product roadmap, helping them balance short-term customization investments with a clear understanding of when those capabilities will be standardized within the core platform.

## Sonar

Sonar is a Leader in this Magic Quadrant. Its SonarQube platform provides automated static code analysis, technical debt management, and architecture insights embedded directly into developer and CI/CD workflows. SonarQube Server can be deployed as a self-managed platform (on-premises or containerized), a distributed server configuration, or a fully managed SaaS offering.

Sonar's operations are geographically diversified, and its customers are mainly large enterprises in regulated industries focused on software quality, security, and maintainability at scale. During the past year, Sonar has increased investment in AI-augmented development workflows, including agentic integrations, quality context for AI coding assistants, and closed-loop agentic remediation automation embedded into pull-request workflows.

## Strengths

- **Product or service:** The SonarQube platform delivers deep static analysis and developer-centric SDLC integration, including architecture visibility and AI-powered remediation capabilities. SonarQube MCP server enables AI agents to navigate through architectural, quality and security issues in code that SonarQube detects.

- **Innovation:** Sonar's innovative closed-loop AI remediation agent fixes issues in pull requests and validates via SonarQube. In addition, its large language model (LLM) evaluation framework and public leaderboard continuously benchmark the quality, security, and complexity of code-generation models. Its new product, SonarSweep, extends these efforts by maintaining and optimizing the datasets used for training models. Sonar has introduced portfolio-level reporting and dashboards.
- **Overall viability:** Sonar shows exceptional financial and organizational viability, supported by a large installed base, sustained revenue growth, and continued investment in product innovation. This stability reduces customer risk and signals long-term commitment to advancing the technical debt management market.

### *Cautions*

- **Offering (product) strategy:** Sonar's product capabilities are geared toward software engineers. Its capabilities for advanced portfolio-level planning, business-aligned prioritization, or financial modeling of technical debt are less mature than those of some competitors.
- **Business model:** Sonar's business model remains primarily centered on developer- and engineering-led adoption, which can limit its appeal for organizations with executive-level budget ownership and portfolio-driven buying motions.
- **Market understanding:** Sonar's platform does not natively support the late phase of the technical debt management life cycle, as it relies on integrations with AI coding agents and other enterprise tools to manage technical debt tracking and remediation. Sonar's SonarQube platform is primarily designed for early stage prevention of technical debt.

### **vFunction**

vFunction is a Niche Player in this Magic Quadrant. The vFunction platform is designed to help organizations understand, prioritize, and refactor large, complex application portfolios. vFunction analyzes monolithic applications down to fine-grained technical elements, providing engineering-centric insights supported by AI-driven remediation guidance and inherited policy awareness. The platform can be deployed as SaaS or self-managed on-premises.

vFunction's operations are concentrated in Europe and North America, and partners extend its reach worldwide. Its customers are across multiple regions and industries, with a high focus on hyperscaler-driven modernization and migration initiatives. During the past year, vFunction has invested in agentic, semantics-driven automation, moving human involvement from designing solutions to validating results.

### *Strengths*

- **Product or service:** vFunction's ability to analyze applications top to bottom, including containers, database objects and stored procedures, reflects an awareness that modernization constraints often originate below the service or API layer. The platform's zoomable analysis model allows engineering teams to easily navigate between architectural, component and object-level views, supporting iterative refactoring. It is especially well-suited for organizations with predominantly Java and .NET environments.
- **Innovations:** vFunction supports automated technical debt management, as its platform creates and manages a backlog of technical debt actions for AI to remediate. Its support for OpenID simplifies integration into enterprise identity ecosystems.
- **Business model:** vFunction operates as an engineering-led, recurring modernization platform tightly embedded in AWS-centric cloud migration ecosystems. Its value proposition is built around continuous telemetry ingestion, analysis, and progress tracking, which makes it appealing for buyers seeking an ongoing modernization platform rather than a one-time assessment tool.

### *Cautions*

- **Market understanding:** vFunction lacks business capability mapping, which may limit its perceived value among executive stakeholders who require modernization decisions to be tied directly to business outcomes. It also lacks support for COBOL, which may limit its appeal for organizations with mainframes.
- **Offering (product) strategy:** vFunction's product capabilities are geared toward software engineers. Its capabilities for business-aligned prioritization or financial modeling of technical debt are less mature than those of some competitors.
- **Overall viability:** vFunction is a relatively small company and provides limited transparency into its scale and growth. It relies heavily on its AWS partnership for credibility and has limited scale in sales and customer success functions.

# Inclusion and Exclusion Criteria

To qualify for inclusion in this Magic Quadrant, a provider must meet the following criteria:

- Meet the market definition of technical debt management tools.
- Have all features applicable to these inclusion criteria and evaluated in this Magic Quadrant research generally available as of 1 February 2026 to all customers and fully documented. Custom development for specific customers will not qualify. General availability means the product or service is available on a public-facing price sheet/card for purchase directly by clients. Providers must be able to furnish the link to a pricing page for its technical debt management tool.
- Offer its technical debt management tool either as a direct-to-customer product or delivered through a forward-deployed engineering model that tailors and deploys the technical debt management tool for client use. This approach must not require the client to engage separate professional services for a broader modernization initiative.
- Demonstrate an active product roadmap, and go-to-market and selling strategy for the solution.
- Have phone, email and/or web customer support. They must offer contract, console/portal, technical documentation and customer support in English (either as the product's default language or as an optional localization).
- Fulfill one of the following size requirements as of 1 February 2026:
  - Have at least 100 paying enterprise customer organizations

OR

- Be a startup or scale-up in the area of technical debt management with the strong potential of growth, VC funded or revenue

We excluded vendors from this Magic Quadrant if their primary use case for the technical debt management is identifying technical debt in no-code or low-code applications, packaged business applications or SaaS-based applications (i.e., developing, extending, configuring or customizing applications such as Salesforce, Dynamics 365, Oracle, SAP

or ServiceNow). The market needs and expected platform capabilities for these use cases differ from the market definition for this Magic Quadrant.

## Evaluation Criteria

### Ability to Execute

We evaluated the following criteria to assess the vendor's Ability to Execute. Below is a description of what we specifically looked for in each criterion.

**Product/Service:** The capabilities of the provider's technical debt management tool, with particular emphasis on its ability to support debt identification (code health, visualization, defects, quality attribute measures); code-level issues (faults, security, complexity, maintainability); architecture observability (complexity, issues in architecture, dependency analysis); and portfolio visibility (debt aggregation, technology currency, capability linkage).

**Overall Viability:** Company financials, revenue growth, and customer growth and retention.

**Sales Execution/Pricing:** Capacity of sales organization, direct sales programs, and indirect sales programs.

**Market Responsiveness and Track Record:** Community engagement and contribution, customer interactions and responsiveness, and product customization and innovation.

**Customer Experience:** Training, onboarding, and certification programs; customer support; customer success and retention.

**Operations:** Customer operations structure, service-level agreements and support, and certifications and quality standards.

We did **not** evaluate **Marketing Execution**, as we determined that this criterion is not relevant to buyers in this market.

#### Ability to Execute Evaluation Criteria

| <i>Evaluation Criteria</i>   | <i>Weighting</i> |
|------------------------------|------------------|
| Product or Service           | High             |
| Overall Viability            | Medium           |
| Sales Execution/Pricing      | Medium           |
| Market Responsiveness/Record | High             |
| Marketing Execution          | NotRated         |
| Customer Experience          | Medium           |
| Operations                   | Medium           |
|                              |                  |

Source: Gartner (May 2026)

## Completeness of Vision

We evaluated the following criteria to assess the vendor's Completeness of Vision. Below is a description of what we specifically looked for in each criterion.

**Market Understanding:** Alignment with customer needs and use case, understanding of strengths and weaknesses, and market positioning.

**Offering (Product) Strategy:** Product roadmaps, customer contributions to product evolution and differentiating capabilities.

**Business Model:** Business model canvas, ecosystem and partnerships, and operating expenses.

**Innovation:** Current innovative capabilities, planned innovative capabilities and awareness of trends.

**Geographic Strategy:** Regional presence and capacity, product localization and regional partnerships.

We did **not** evaluate the following criteria, as we determined that these criteria are not relevant to buyers in this market:

- **Marketing Strategy**
- **Sales Strategy**
- **Vertical/Industry Strategy**

**Completeness of Vision Evaluation Criteria**

| <i>Evaluation Criteria</i>  | <i>Weighting</i> |
|-----------------------------|------------------|
| Market Understanding        | High             |
| Marketing Strategy          | NotRated         |
| Sales Strategy              | NotRated         |
| Offering (Product) Strategy | High             |
| Business Model              | Low              |
| Vertical/Industry Strategy  | NotRated         |
| Innovation                  | High             |
| Geographic Strategy         | Low              |
|                             |                  |

Source: Gartner (May 2026)

**Quadrant Descriptions**

## **Leaders**

Leaders provide mature offerings that meet market demand and have demonstrated the vision necessary to sustain their market position as requirements evolve. The hallmark of Leaders is that they focus on and invest in their offerings to the point where they lead the market and can affect its overall direction. As a result, Leaders can become the vendors to watch as you try to understand how new market offerings might evolve.

Leaders typically possess a large, satisfied customer base (relative to the size of the market) and enjoy high visibility within the market. Their size and financial strength enable them to remain viable in a challenging economy.

Leaders typically respond to a wide market audience by supporting broad market requirements. However, they may fail to meet the specific needs of vertical markets or other more specialized segments.

## **Challengers**

Challengers have a strong Ability to Execute but may not have a plan that will maintain a strong value proposition for new customers. Larger vendors in mature markets may be positioned as Challengers because they choose to minimize risk or avoid disrupting their customers or their own activities.

Although Challengers typically have significant size and financial resources, they may lack strong vision, innovation or an overall understanding of market needs. Challengers may offer products nearing the end of their lives that dominate a large but shrinking segment.

Challengers can become Leaders if their vision develops. Over time, large companies may fluctuate between the Challengers and Leaders quadrants as their product cycles and market needs shift.

## **Visionaries**

Visionaries align with Gartner's view of how a market will evolve, but their ability to deliver against that vision is less proven. In growing markets, this status is typical. In more mature markets, it may reflect a competitive strategy for a smaller vendor — such as selling an innovation ahead of mainstream demand — or a larger vendor trying to break out of a rut or differentiate itself.

For vendors and customers, Visionaries fall into the higher-risk-higher-reward category. They often introduce new technology, services or business models, and they may need to build financial strength, service and support, and sales and distribution channels. Whether Visionaries become Challengers or Leaders may depend on if customers accept the new technology or the vendors can develop partnerships that complement their strengths. Visionaries are sometimes attractive acquisition targets for Leaders or Challengers.

## **Niche Players**

Niche Players do well in a segment of a market, or they have a limited ability to innovate or outperform other vendors in the wider market. This may be because they focus on a particular functionality or geographic region, or because they are new entrants to the market. Alternatively, they may be struggling to remain relevant in a market that is moving away from them. Niche Players may have reasonably broad functionality, but with limited implementation and support capabilities and relatively limited customer bases. Compared to other quadrants, they do not demonstrate a strong vision for their offerings.

For end users, assessing Niche Players is more challenging than assessing vendors in other quadrants. Some could make progress, while others do not execute well and may not have the vision and means to keep pace with broader market demands.

A Niche Player may be a perfect fit for your requirements. However, if it goes against the direction of the market — even if you like what it offers — then it may be a risky choice because its long-term viability will be threatened.

## **Context**

CIOs and software engineering leaders are seeking platform-based solutions that align closely with AI-native software engineering needs. Technical debt management solutions are uniquely positioned to support this shift to AI-native software engineering.

The demand for measurable ROI on technical-debt remediation is rising sharply. At the same time, the adoption of AI coding assistants and agents is accelerating code creation, which risks code sprawl and increasing architectural drift. This amplifies the need for

tools that can maintain control, ensure architectural integrity, and provide continuous visibility across evolving codebases.

When evaluating technical debt management tools, you should:

- Prioritize adoption of a technical debt management tool that addresses your most pressing use case: technical debt discovery, categorization and risk scoring, impact assessment, modeling and remediation, or planning and tracking.
- Favor tools that quantify impact, not just debt. Require capabilities for coupling analysis, change-impact modeling, and portfolio-wide structural risk assessment to clearly link technical debt to architecture qualities and business criticality.
- Prioritize solutions that turn technical debt insights into AI-assisted remediation actions, comparative benchmarks, and trackable outcomes.

## Market Overview

The rise of agentic AI is transforming how software engineers work, and it carries significant implications for how teams manage technical debt.

At first glance, AI coding tools appear to be reducing technical debt. In the 2025 Gartner AI in Software Engineering Survey, 39% of engineering leaders and teams noticed a reduction in technical debt within newly generated code due to use of AI.<sup>1</sup> AI has become proficient at detecting code-level defects and handling routine code hygiene, which has driven a decline in technical debt within individual codebases.

Upon closer inspection, however, it's clear that technical debt remains a pressing concern. As teams expand usage of AI coding agents across the SDLC, they are accumulating more architectural technical debt than ever before. By 2027, this pileup of architectural technical debt will account for 80% of all technical debt.

Architectural technical debt is even riskier than code-level technical debt. It accumulates under the radar across multiple systems and layers of applications, which will eventually cause systemic failures and soaring maintenance costs. Architectural technical debt will limit an organization's ability to deliver AI agents and agentic AI solutions.

The pervasiveness and riskiness of architectural technical debt are driving increased investment in technical debt management tools. Gartner estimates that the technical

debt management tools market will reach \$1.2 billion in annual revenue by the end of 2026, with double-digit annual growth rates projected through 2030.

## How to Navigate the Technical Debt Management Tools Market

Buyers of technical debt management tools should be aware of three trends when evaluating vendors.

### **Vendors are expanding their capabilities to address architectural technical debt.**

Technical debt management vendors are shifting their focus beyond code-level issues by releasing features such as:

- Continuous architectural observability, using static and dynamic analysis to monitor structural drift, dependency sprawl, and erosion of architectural intent.
- Software intelligence, enabling more contextualized reasoning and remediation automation.

These capabilities are fueling a new wave of modernization initiatives.

**Vendors are embedding AI capabilities into technical debt management tools.** AI-driven code remediation is evolving from experimental prototypes into fully productized capabilities. Examples of these capabilities include:

- AI-generated code fix recommendations, with clear enablement flows, policy controls, and language-specific guardrails.
- Automated refactoring that is paired with semantic behavior validation, reducing the risk associated with AI hallucinations.
- MCP services that expose code- and architecture-level insights, enabling AI coding agents to incorporate code health and architectural context directly into their workflows.

**Vendors are incorporating capabilities from adjacent technology markets into their platforms.** The technical debt management tools market increasingly overlaps with adjacent technology markets, such as:

- Static code analysis

- Static application security testing (SAST)
- Architecture observability
- Application portfolio management
- Code modernization tools

These capabilities are merging into unified remediation workflows, enabling continuous visibility from code to architecture, and AI-augmented recommendations grounded in both code semantics and architectural intent.

However, the majority of the players struggle to convert these insights into outputs that support prioritization, investment planning, life cycle governance, and executive justification. This misalignment explains why enterprises will continue to rely heavily on services, expert interpretation, and manual processes to translate tool output into action, despite increasingly sophisticated platforms.

## How to Use These Insights

Buyers of technical debt management tools should use our expert evaluation in this Magic Quadrant to identify vendors who align with your short- and long-term technical debt management strategy. Then, use the companion Critical Capabilities research to select tools that are best-suited for your highest-priority use cases.

*This Magic Quadrant for Technical Debt Management Tools replaces the **Market Guide for Technical Debt Management Tools**. It offers a rigorous, comparative analysis of vendors in this market, so buyers can make more informed decisions as they navigate the market.*

---

⊕ Evidence

⊕ Evaluation Criteria Definitions

## Unlock more AI value

With unmatched, proprietary AI insights from Gartner

[Explore the AI Hub ↗](#)

© 2026 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 research may not be used as input into or for the training or development of generative artificial intelligence, machine learning, algorithms, software, or related technologies.

[About](#) [Careers](#) [Newsroom](#) [Policies](#) [Site Index](#) [IT Glossary](#) [Gartner Blog](#)  
[Network](#) [Contact](#) [Send Feedback](#)

© 2026 Gartner, Inc. and/or its Affiliates. All Rights Reserved.