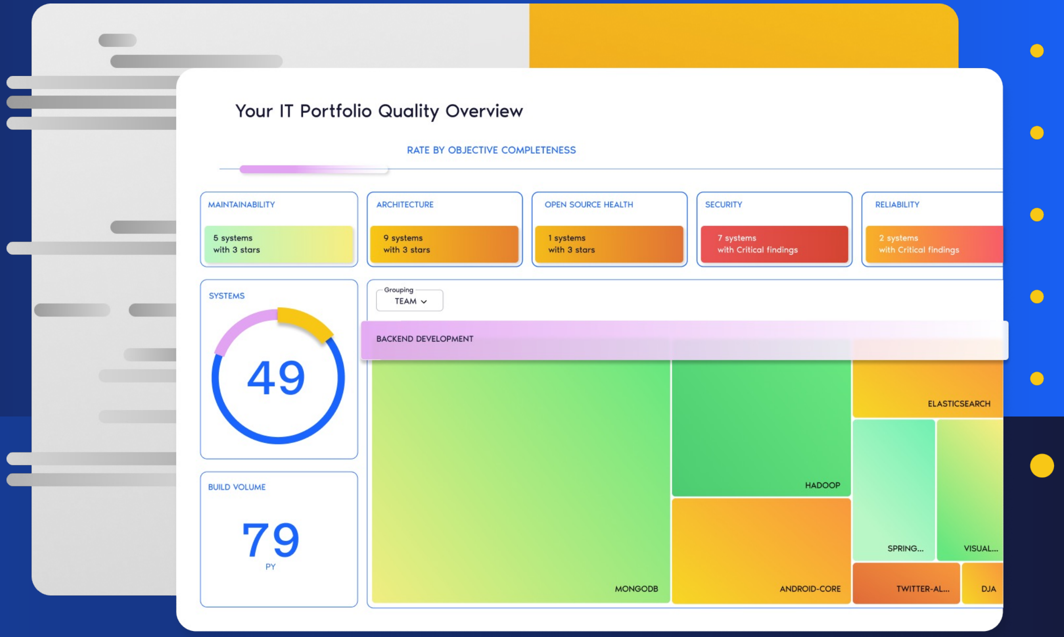
Software
Improvement
Group



Software Portfolio Scan report

Table of contents

Get a secure, agile, and cost-effective software portfolio

You’re reading this because you want to take control of your software portfolio—great move.

Your software powers your business, but hidden risks, inefficiencies, and outdated architecture could be holding you back, driving up costs, creating security blind spots, and slowing innovation.

This report gives you a preview of the insights you'll get from the Software Portfolio Scan: a fast, high-level assessment that benchmarks your software landscape, pinpoints risks, and delivers clear, actionable recommendations.

Let’s dive in.

Don't let your systems hold you back

Move faster. Innovate smarter. Cut costs.

Every day spent maintaining legacy systems is a **missed opportunity for growth.** Hidden risks, slow development, and rising costs will only worsen over time.

In just two weeks, the Software Portfolio Scan will help you:

- **Identify risks before they become problems**
- **Benchmark your software portfolio against competitors**
- **Get a clear, boardroom-ready roadmap**

Act now: Pinpoint risks, reduce costs, and future-proof your software portfolio with a Portfolio Scan.

Get your Software Portfolio Scan today

4.5x

FASTER TIME TO MARKET

-50%

LOWER MAINTENANCE COSTS

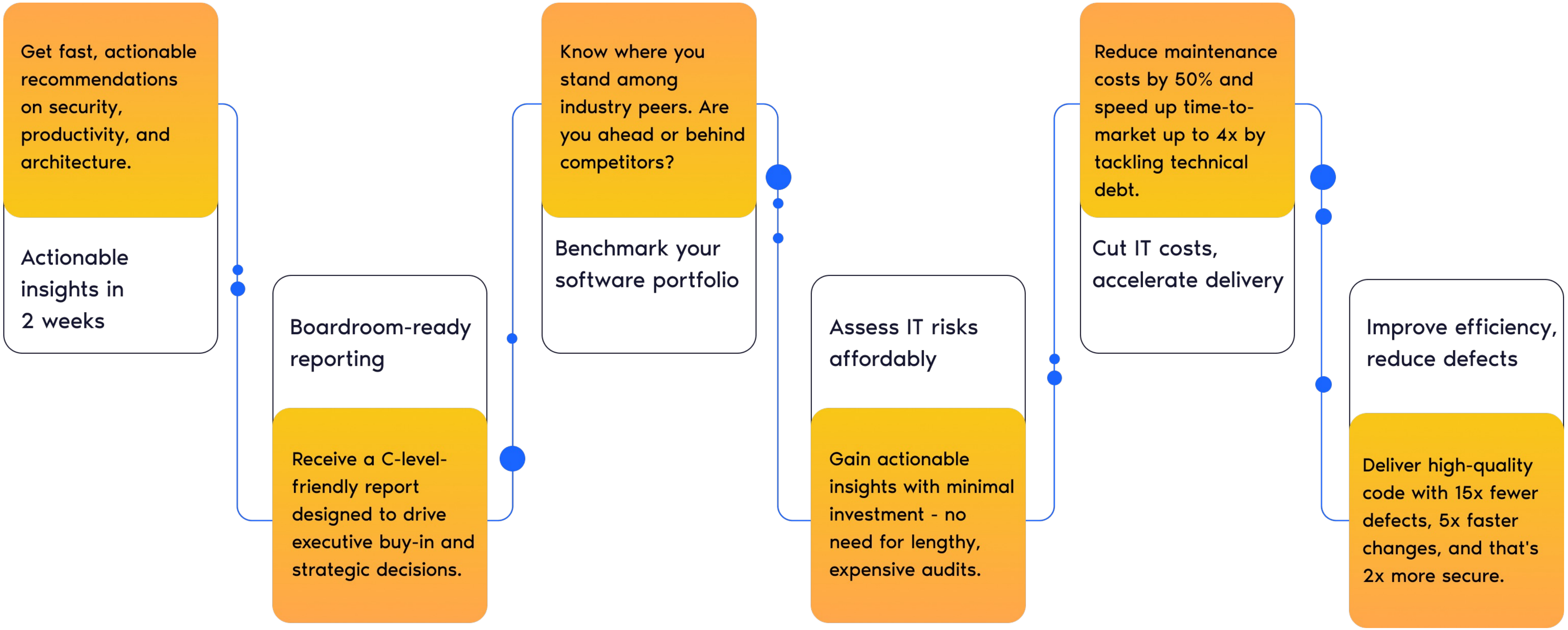
+30%

MORE DEVELOPMENT CAPACITY

2x

MORE SECURE

What do you gain from a Software Portfolio Scan?



Actionable insights in 2 weeks

Get fast, actionable recommendations on security, productivity, and architecture.

Boardroom-ready reporting

Receive a C-level-friendly report designed to drive executive buy-in and strategic decisions.

Benchmark your software portfolio

Know where you stand among industry peers. Are you ahead or behind competitors?

Assess IT risks affordably

Gain actionable insights with minimal investment - no need for lengthy, expensive audits.

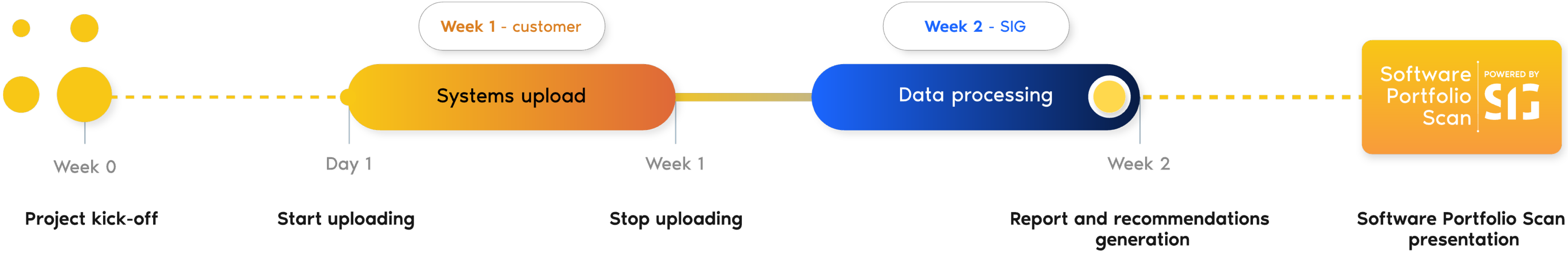
Cut IT costs, accelerate delivery

Reduce maintenance costs by 50% and speed up time-to-market up to 4x by tackling technical debt.

Improve efficiency, reduce defects

Deliver high-quality code with 15x fewer defects, 5x faster changes, and that's 2x more secure.

A clear and actionable software assessment, delivered in just 2 weeks



Week 1 - customer

Week 2 - SIG

Systems upload

Data processing

Software Portfolio Scan POWERED BY SIG

Week 0

Day 1

Week 1

Week 2

Project kick-off

Start uploading

Stop uploading

Report and recommendations generation

Software Portfolio Scan presentation

In just 14 days, you'll get a benchmark-driven software assessment with clear, actionable recommendations so you can cut maintenance costs, improve developer productivity, mitigate high-level security risks, and future-proof your architecture.

Why Software Improvement Group (SIG)?

300+
BILLION LINES OF CODE

20,000+
SYSTEMS ANALYZED

300+
TECHNOLOGIES SUPPORTED

25
YEARS OF EXPERIENCE

TRUSTED BY 400+ LEADING ENTERPRISE ORGANIZATIONS

NXP
KLM
SIEMENS
ING
DHL
PHILIPS

CERTIFICATIONS

ISO 27001

ISO 17025

AICPA SOC

For over 25 years, SIG has been a trusted leader in software quality assurance, supporting both traditional and AI-driven software portfolios.

Backed by the world’s largest commercial software benchmark, our platform, Sigrid®, analyzes your source code to uncover risks, improve maintainability, and ensure long-term resilience.

Combined with out team of expert IT consultants, we deliver reliable, data-driven insights to enhance software quality, reduce risk, and drive smarter decision-making.

That's why 400+ leading enterprises rely on SIG to ensure their software is a driver of success, not a source of risk.

Introducing: Horizon United, a fictional client with real-world challenges

About Horizon United

This sample report is based on Horizon United, a fictitious bank with 50 years of history and a global client base. Like many enterprises, its complex IT infrastructure relies on a mix of modern technologies, legacy systems, and open-source libraries.

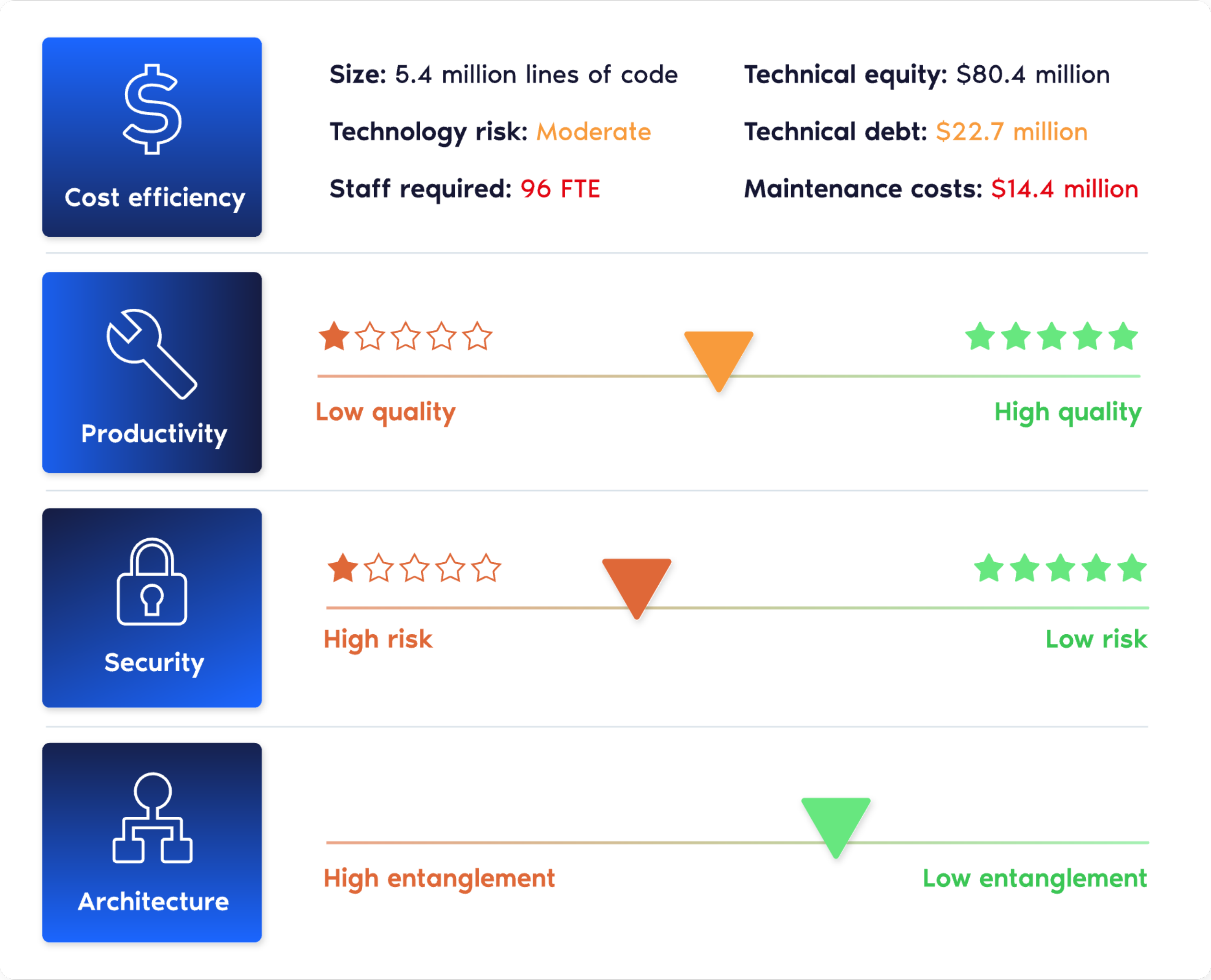
Why did they get a Software Portfolio Scan?

Facing rising maintenance costs and security concerns, Horizon United’s CTO sought an objective, data-driven assessment from SIG to:

- Identify areas for improvement in its software portfolio
- Free up developer resources and reduce technical debt
- Spot high-level risks before they escalate
- Ensure its IT architecture is future-ready and AI-ready
- Benchmark its software portfolio against industry standards

Key findings

Horizon United’s key findings at a glance



Cost efficiency

Size: 5.4 million lines of code

Technology risk: Moderate

Staff required: 96 FTE

Technical equity: $80.4 million

Technical debt: $22.7 million

Maintenance costs: $14.4 million

Productivity

Security

Architecture

Legacy COBOL consumes more than 50% of IT maintenance costs

Horizon United Bank’s legacy COBOL systems account for the majority of its $22.7 million in technical debt. Addressing this could cut annual IT maintenance costs – currently $14.4 million – by more than 50%. Modernization is critical to reduce expenses.

Low-quality core banking code slows innovation up to 30%

While most of Horizon United's systems perform above market average, its core banking system's low-quality code is delaying updates and innovation. The more time spent fixing outdated systems, the less capacity there is for new development. On average, higher-quality systems free up 30% extra capacity for innovation and improvement.

Security vulnerabilities in customer-facing systems increase risk

While Horizon United’s overall security rating is above industry average (3.3), customer-facing software systems contain vulnerabilities listed in the OWASP Top 10—the industry's most critical web application risks. In addition, weak open-source governance and insufficient security tooling further raise the risk of breaches, regulatory fines, and business disruption.

Rigid architecture makes system changes up to 30% slower

The digital banking system meets industry standards, but rigid dependencies are slowing modernization. A more flexible architecture could accelerate innovation. Above-average architecture scoring systems allow for changes to be made 30% faster than market-average systems.

Deep dive 1: Cost efficiency

The problem: Maintenance costs are holding Horizon United back

Technology	Technical equity		Technical debt	Projected maintenance	
COBOL	$28M	2.5M LoC	$15M	56 FTE	$8.3M
Java	$24M	1.4M LoC	$3.6M	19 FTE	$2.8M
C++	$10.5M	0.6M LoC	$1.5M	10 FTE	$1.5M
Python	$9.5M	0.5M LoC	$1.3M	5 FTE	$0.8M
Others	$8.4M	0.4M LoC	$1.3M	7 FTE	$1M
Total volume:	**$80.4M**	**5.4M LoC**	**$22.7M**	**96 FTE**	**$14.4M**

Recommendation:
Reduce reliance on COBOL

Horizon United should phase out COBOL to reduce costs and boost agility.

This will:

- Decrease maintenance costs
- Free up resources for innovation
- Speed up development and boost competitiveness

Key findings

- Horizon United has total Technical Equity of $80 million and 5.4 million lines of code, spanning both **modern and legacy technologies.**
- COBOL makes up only 35% of the codebase but accounts for 66% of total technical debt ($15 million).
- Horizon United spends $14.4 million annually maintaining outdated systems, with over 50% of those costs directly tied to COBOL.

Why this matters

- High costs: $8.3 million spent yearly on maintaining outdated technologies instead of driving innovation
- Slow development: Engineers waste time fixing legacy code rather than building new features.
- Competitive risk: Other banks move faster with modern technology.

Terminology

- **Technical equity:** The total value of proprietary software and technology.
- **Technical debt:** The cost of fixing outdated or inefficient code.
- **Projected maintenance:** The yearly cost to stay operational, assuming the industry average of:
  - 15% code change rate per system
  - $150,000 yearly salary per FTE

Deep dive 2: Productivity

The problem: Horizon United's maintainability is falling behind

What is maintainability?

Maintainability measures how easy it is to test, change, and evolve a codebase. A well-maintained system allows for quick modifications, reducing costs and improving development speed.

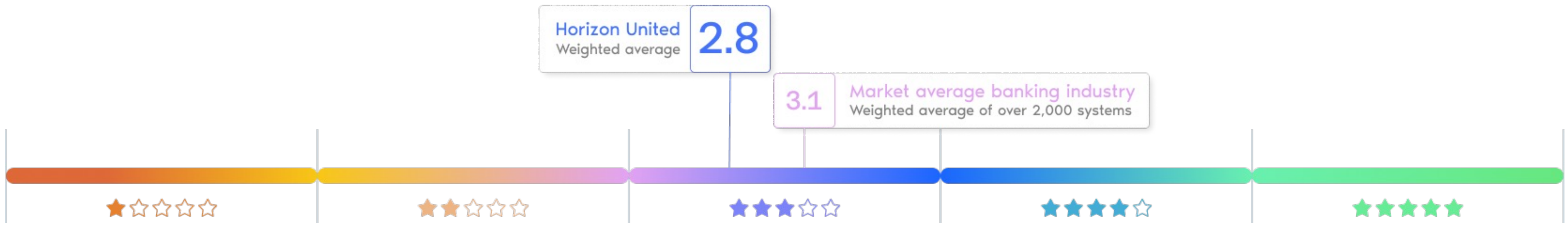
Why does it matter?

Simply put: Productivity. A low maintainability score creates friction, slows development, and increases IT costs. A high score enables faster releases, better efficiency, and lower maintenance costs.

How do we measure maintainability?

SIG evaluates source code using the SIG/TÜViT Evaluation Criteria for Trusted Product Maintainability. Results are benchmarked against thousands of systems, recalibrated annually, and assigned a star rating (1 = low, 5 = high).

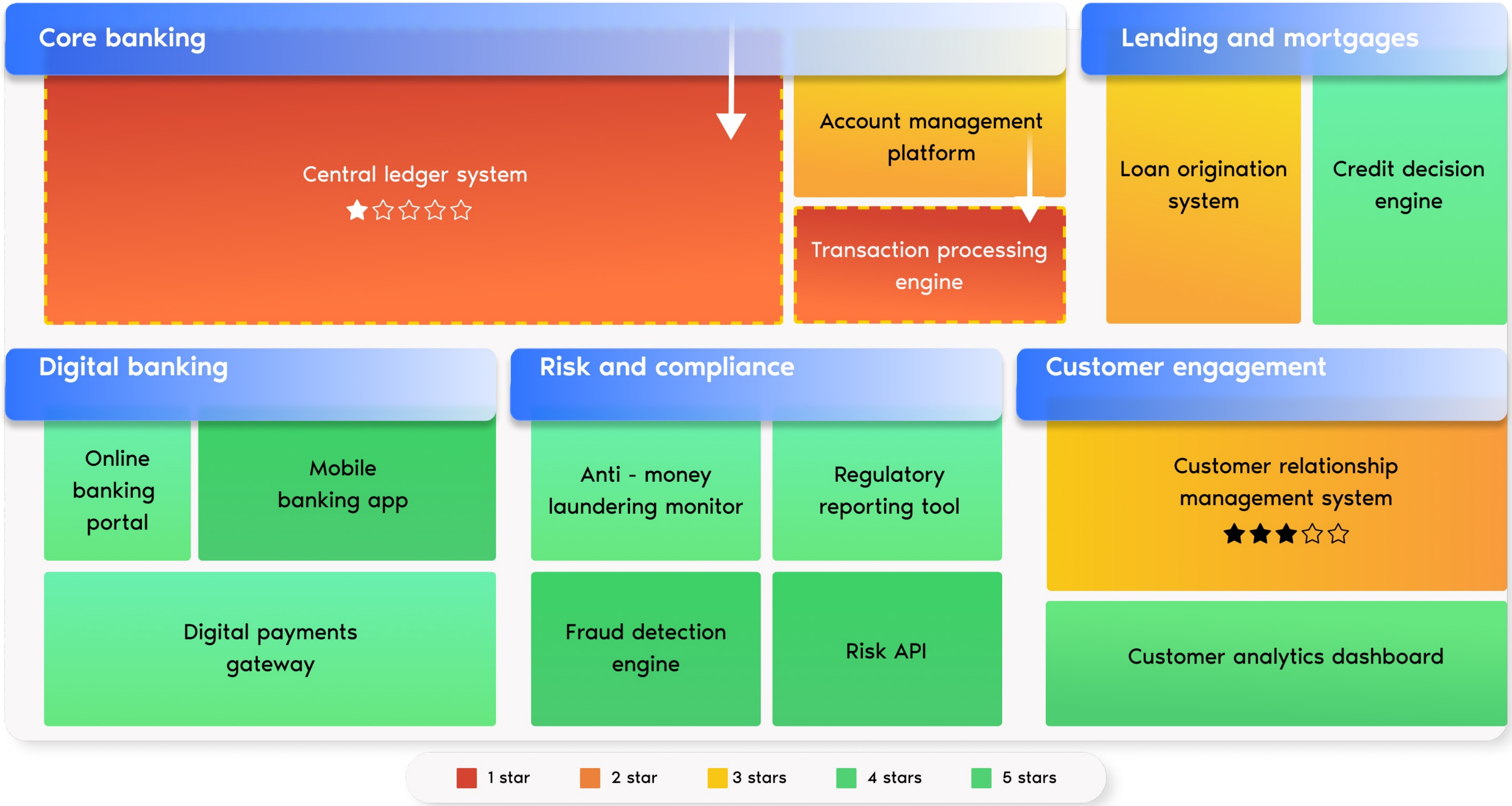
TÜVNORDGROUP



Key findings

As we can see, Horizon United has a **below average maintainability score (2.8)** compared to its direct competitors (3.1).

Core banking's low quality is a roadblock to efficiency



Core banking

Central ledger system

★☆☆☆☆

Account management platform

Transaction processing engine

Lending and mortgages

Loan origination system

Credit decision engine

Digital banking

Online banking portal

Mobile banking app

Digital payments gateway

Risk and compliance

Anti - money laundering monitor

Regulatory reporting tool

Fraud detection engine

Risk API

Customer engagement

Customer relationship management system

★★★☆☆

Customer analytics dashboard

🟥 1 star 🟧 2 star 🟨 3 stars 🟩 4 stars 🟩 5 stars

Key findings

- Horizon United's core banking system has the lowest maintainability rating (0.8 stars) in its software portfolio. High technical debt is making it costly, slow, and difficult to manage.
- After a code maintainability analysis, we see that 70% of code is highly complex and 56% is redundant.
- Digital Banking, Risk & Compliance perform well (4 stars), showing strong quality

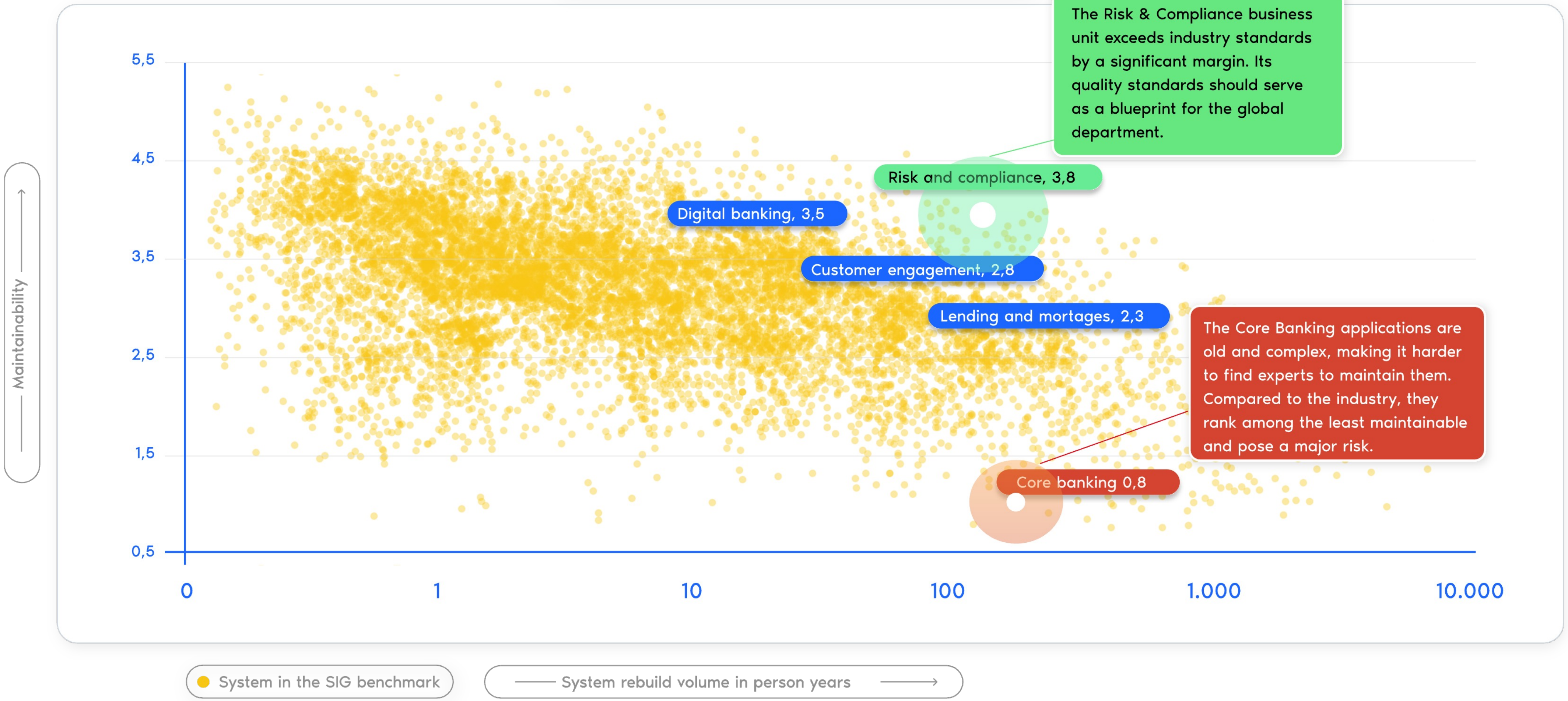
Recommendation: Modernize the core ledger and transaction processing engine systems

This can:

- Improve development speed
- Reduce costs
- Enhance system flexibility

Digging deeper: One system is impacting the maintainability score

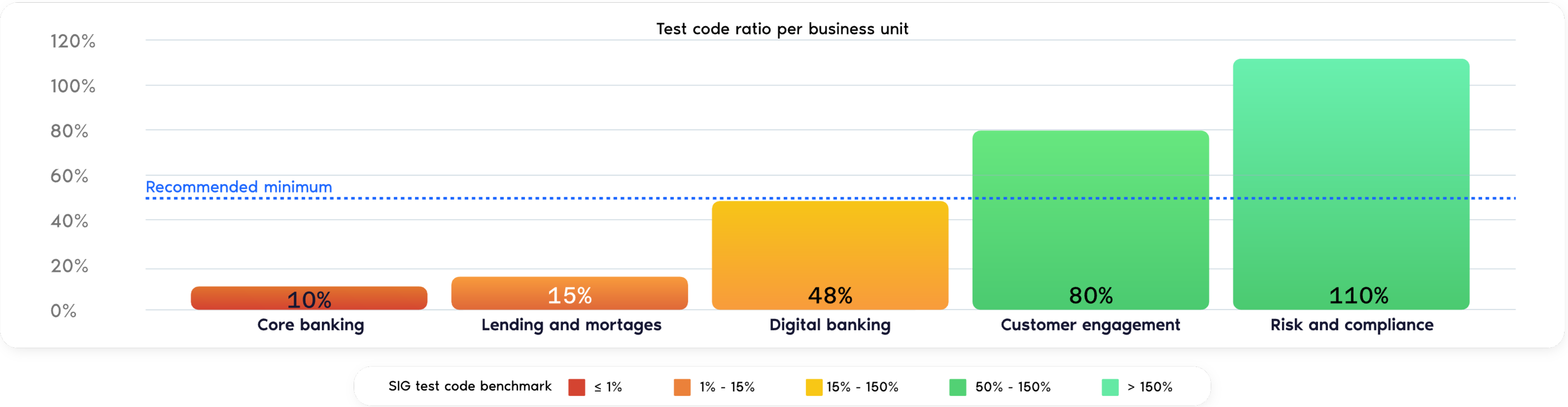
Maintainability benchmark



The Risk & Compliance business unit exceeds industry standards by a significant margin. Its quality standards should serve as a blueprint for the global department.

The Core Banking applications are old and complex, making it harder to find experts to maintain them. Compared to the industry, they rank among the least maintainable and pose a major risk.

Low test-code ratio is driving up risks and costs



Key findings

- Core banking, lending, and digital banking have low test automation, increasing risk.
- Heavy reliance on manual testing slows development and releases.
- Legacy systems lack unit tests, leading to undetected bugs and system failures.

Why this matters

- **Higher failure risk:** Critical bugs may go undetected, leading to outages and disruptions.
- **Slower releases:** Manual testing creates bottlenecks, making the company less responsive.
- **More defects, higher costs:** New updates are riskier, increasing customer-facing issues and maintenance expenses.

Recommendation: Boost test automation

Increase automated test coverage in core banking, lending, and digital banking.

This can:

- Reduce risk
- Accelerate releases
- Enhance software stability

Deep dive 3: Security

18

Security code reviews help identify vulnerabilities before they become threats

What are vulnerability and threats?

A vulnerability is a flaw or weakness in a software's system, processes, design, implementation, or deployment. Vulnerabilities can be technical, like outdated dependencies, or human-generated such as design or coding errors. A threat is anything that could exploit a vulnerability and cause harm.

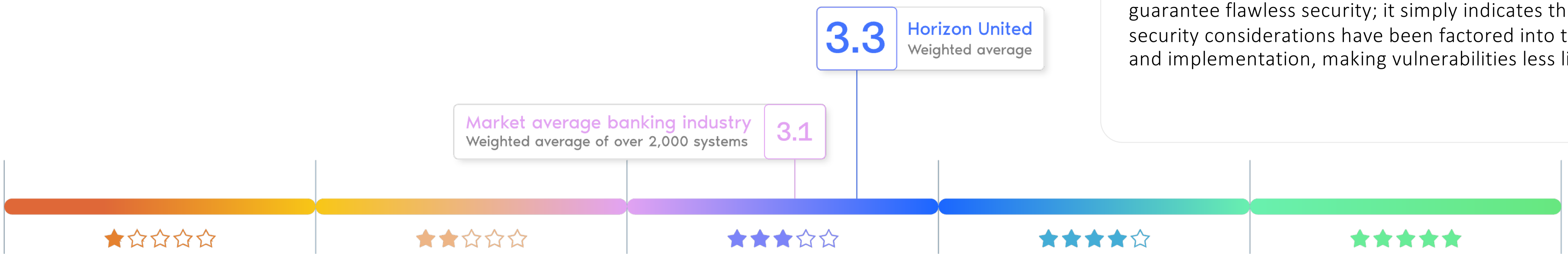
Why does it matter?

Security is often treated as a final checkpoint rather than a fundamental part of the software development process. A common misconception is: "We have penetration tests, so we're secure." However, by adopting Security by Design, organizations can proactively identify vulnerabilities before they're costly and time-consuming to fix.

How do we measure security?

We evaluate system properties through a thorough analysis of the source code and infrastructure. This includes reviewing the codebase and other artifacts (such as documentation) to derive scores for various system characteristics. These characteristic scores are then mapped to the OWASP Top 10 –which identifies the ten most critical risks in web application security– to rank software systems from 1 to 5 stars.

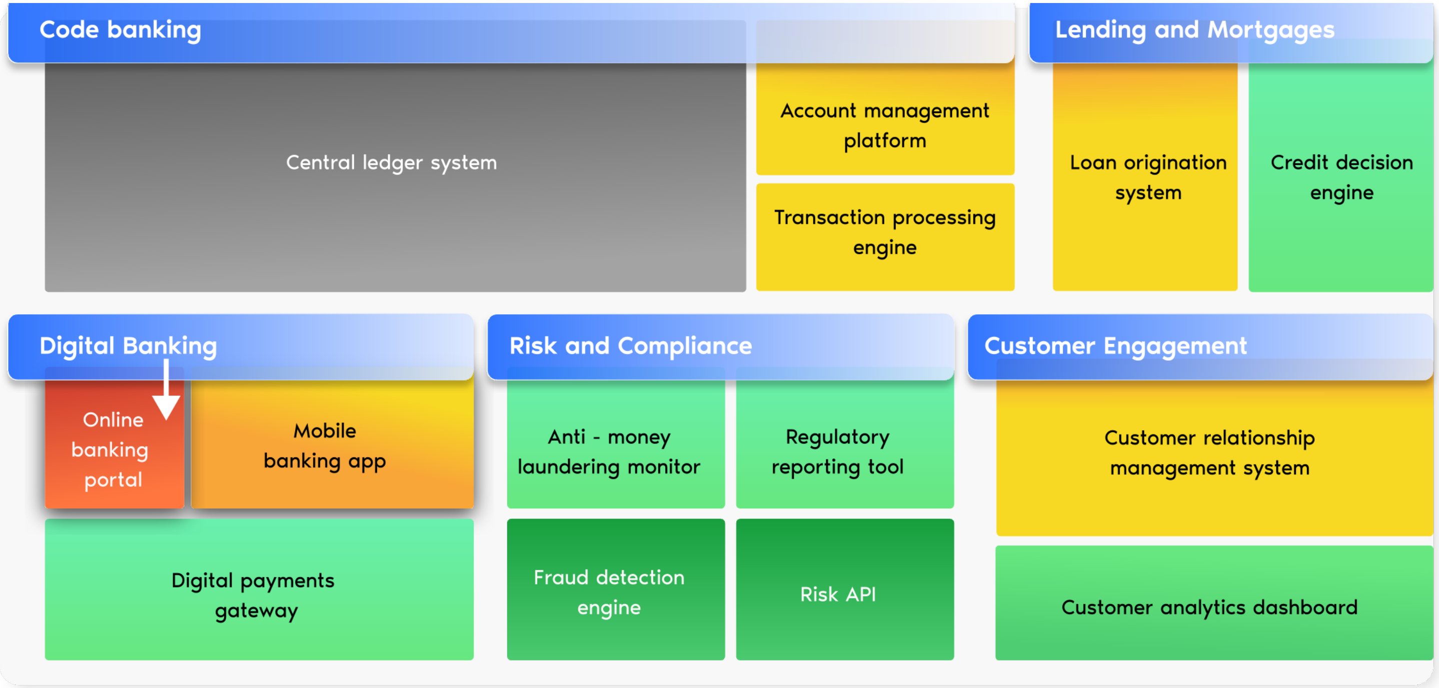
It is important to note that a 4- or 5-star rating does not guarantee flawless security; it simply indicates that security considerations have been factored into the design and implementation, making vulnerabilities less likely.



Key findings

- Looking at the weighed average, Horizon United has an above-average security (3.3) rating, outperforming the market average (3.1).
- However, one application: Online banking portal rates low and poses potential risks.
- Important note: Not all technology could be automatically scanned. An in-depth cybersecurity risk assessment is recommended to understand all potential security issues.

Security gaps in online and mobile banking put Horizon United at risk



Code banking

Central ledger system

Account management platform

Transaction processing engine

Lending and Mortgages

Loan origination system

Credit decision engine

Digital Banking

Online banking portal

Mobile banking app

Digital payments gateway

Risk and Compliance

Anti - money laundering monitor

Regulatory reporting tool

Fraud detection engine

Risk API

Customer Engagement

Customer relationship management system

Customer analytics dashboard

■ Technology could not be automatically scanned. An in-depth cybersecurity risk assessment is recommended to understand all potential security issues.

Key findings

- **The Online Banking Portal and the Mobile Banking App systems** have a high number of critical security findings, increasing the potential security risks.
- While Horizon United's overall security posture is above industry average, these applications remain vulnerable
- COBOL-based technology in core banking could not be automatically scanned, meaning undetected security gaps may exist and require further review.

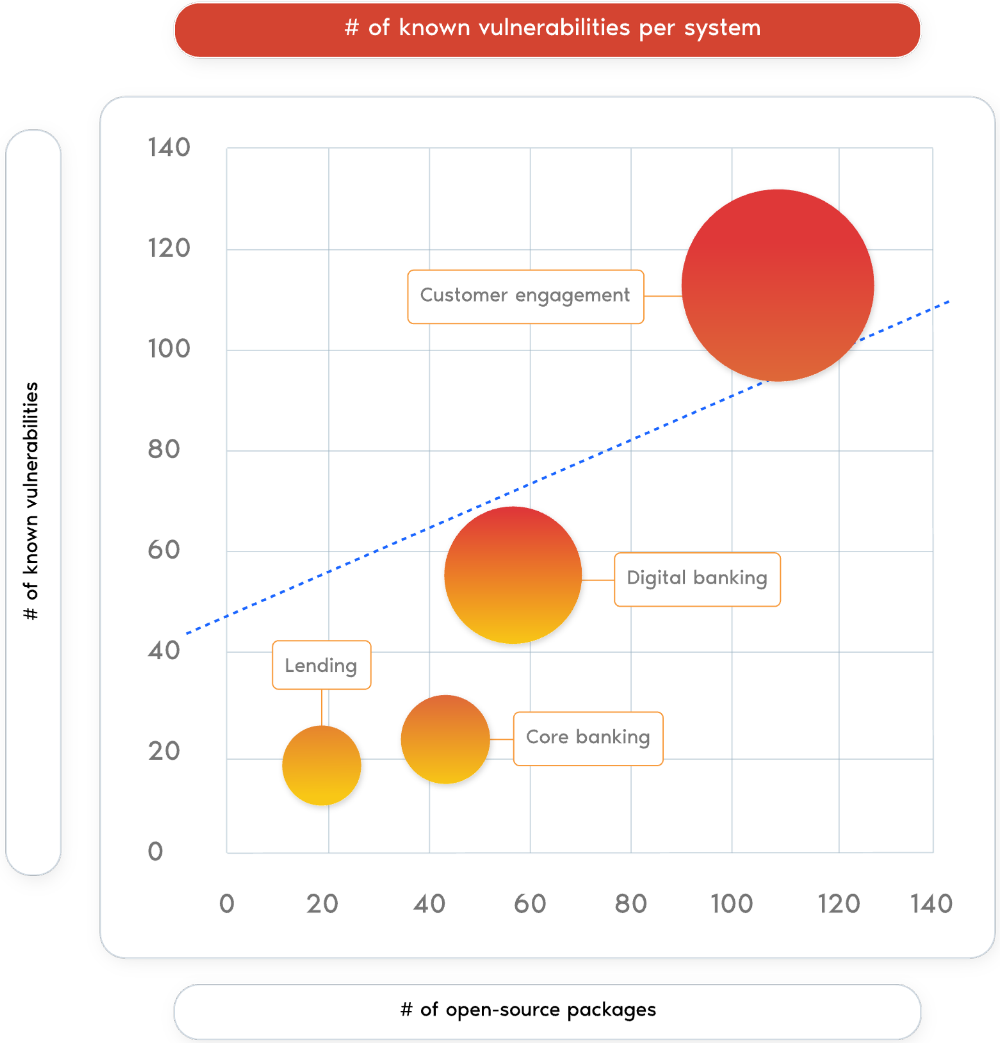
Why this matters

- **Higher risks:** A security breach could lead to financial losses, fraud, and legal consequences.
- **Customer trust:** Online and mobile banking are customer-facing applications, meaning any vulnerability could directly impact user confidence and retention
- **Regulatory compliance:** Addressing vulnerabilities is essential to avoiding fines and legal issues.

Recommendation: Strengthen security by addressing open-source risks

Horizon United Bank should prioritize a security code review for the Online Banking Portal, as it is a known risk. As a customer-facing application, it increases the overall likelihood of a security breach.

Open-source management gaps are creating vulnerabilities



While using open-source libraries saves time and money, poor management can introduce Horizon United to security vulnerabilities, compliance violations, and legal complications.

Key findings

- Customer engagement systems contain over 100 known open-source vulnerabilities, increasing exposure to security threats
- Other applications perform better but still contain between 20 and 60 known vulnerabilities, indicating a lack of consistent governance.
- Many systems rely on 1-2 open-source dependencies with restrictive licenses, which could create legal and compliance risks.

Why this matters

- Unmanaged vulnerabilities can be exploited, leading to data loss or downtime.
- Restrictive licenses in commercial applications could result in legal issues.

Deep dive 4: Future-proofness

While the architecture rating is above average, scalability is a problem

What is architecture?

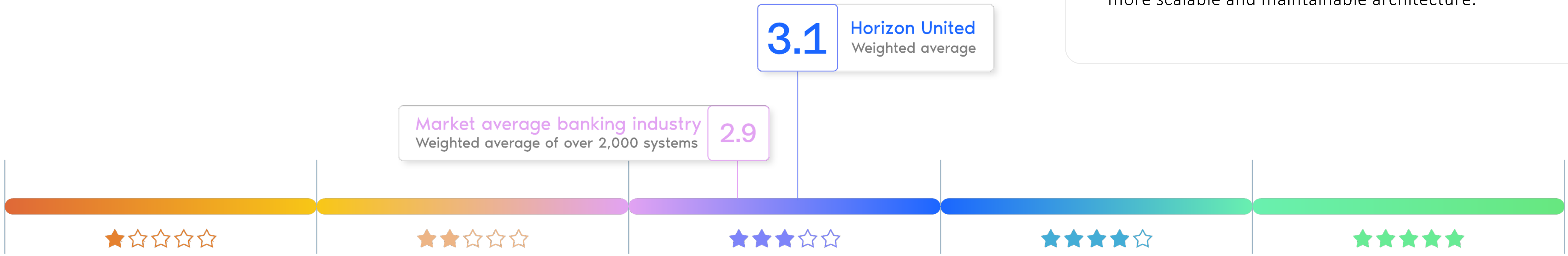
Software architecture defines how a system is structured and how its components interact. A well-designed architecture enables scalability, adaptability, and efficient system evolution, while a rigid one can limit growth, slow development, and increase costs

Why does it matter?

Simply put: Future-proofing. A system's architecture dictates how well it adapts to business changes, integrates new technologies, and supports long-term growth. When architecture is too rigid, even small changes require major effort, slowing development and making it harder to scale teams efficiently

How do we measure architecture?

Our Architecture Quality Model assesses five key factors that determine a system's ability to evolve: structure (modularity), communication (data exchange efficiency), data access (ease of retrieval), evolution (independence of changes), and knowledge distribution (how well architectural expertise is shared). Each system is benchmarked against thousands of applications and rated from 1 to 5 stars, with higher scores indicating a more scalable and maintainable architecture.



Key findings

Horizon United's architecture has a 3.1-star rating, slightly above the industry average of 2.9. However, as shown in more detail in the next slide: High component coupling makes even minor updates difficult, requiring significant time and effort. While several components are hardly changed, indicating active knowledge may no longer be available.

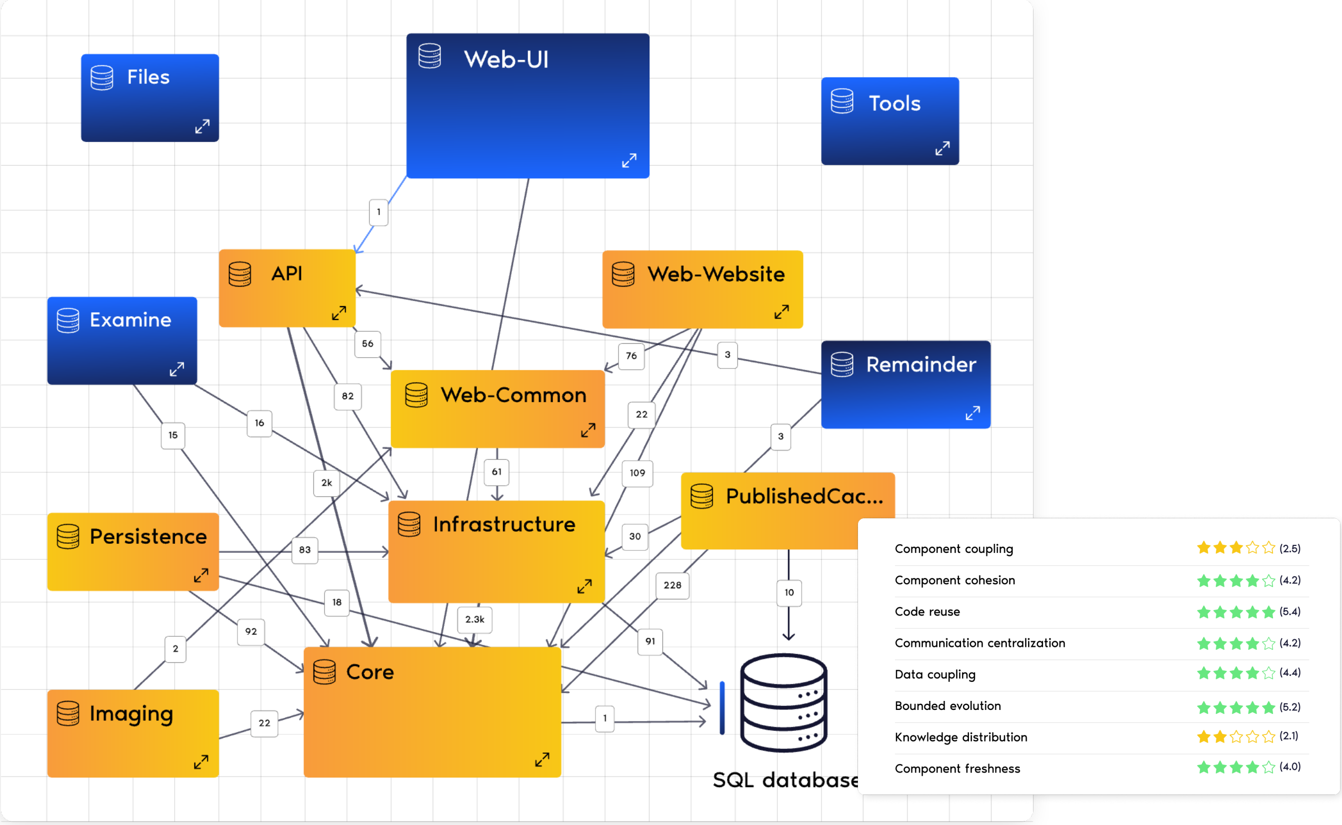
Zooming in: Architecture rigidity is limiting scalability and AI readiness

Key findings

- High component coupling means small changes trigger widespread modifications, increasing development effort and risk.
- Critical expertise is limited to a few people (low knowledge distribution (2.1/5), creating key-person risk and slowing onboarding.
- Outdated components (4.0/5 freshness score) could delay modernization efforts.

Why this matters

- **Higher development costs:** Tightly coupled components make changes expensive and time-consuming.
- **Reduced agility:** Harder to scale and adapt to business needs.
- **Operational risk:** Knowledge silos increase dependency on a few experts.
- **AI readiness is limited:** Tight coupling and outdated components make AI integration complex and costly.



Component coupling	★★★☆☆ (2.5)
Component cohesion	★★★★☆ (4.2)
Code reuse	★★★★★ (5.4)
Communication centralization	★★★★☆ (4.2)
Data coupling	★★★★☆ (4.4)
Bounded evolution	★★★★★ (5.2)
Knowledge distribution	★★☆☆☆ (2.1)
Component freshness	★★★★☆ (4.0)

Recommendation: Decouple, share knowledge, and modernize for AI readiness

To enhance flexibility and future-proof the system, gradually decouple core banking components, improve knowledge sharing to reduce key-person risk, and modernize outdated components for long-term scalability and seamless AI adoption.

What others say about us

“SIG provided us with useful insight regarding our core systems within just a few days.”

Claus Sprave, Head of IT LichtBlick SE (Eneco)

“Making sure your product is secure, protected and compliant throughout the entire lifecycle, from design to end-of-life, has become truly business-critical. This partnership with SIG offers strong support for cybersecurity.”

Joe Bohman, Senior Vice-president of Lifecycle Collaboration Software at Siemens Digital Industries Software

“We have been very impressed by the expertise that SIG has brought to the table and the way they translate their research and findings in clear, concise and simple to understand set of business recommendations.”

Harry van der Vossen, Director of Digital Delivery, RelyOn Nutec

Software Improvement Group

Identify. Act. Thrive.