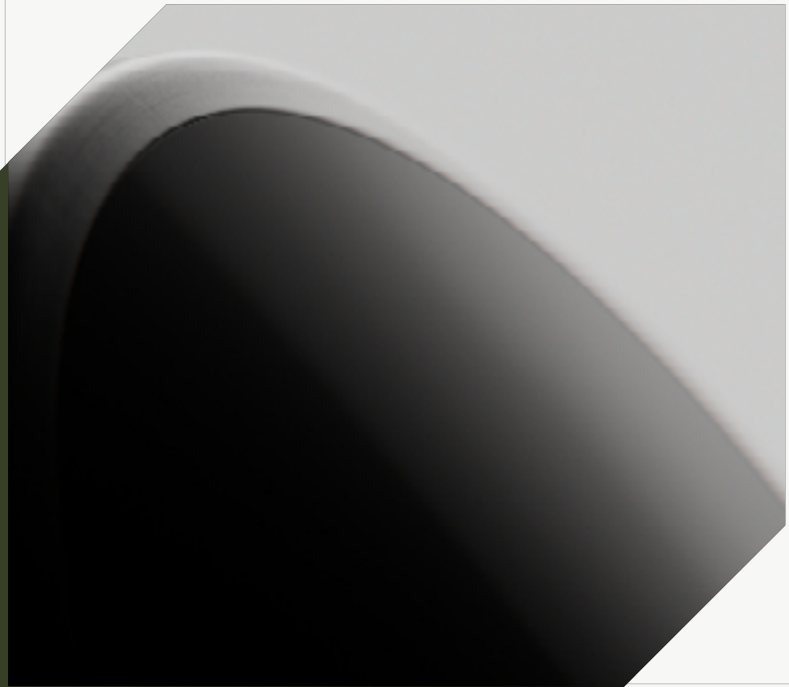
Qt Group

WHITE PAPER

Quality Commitment at Qt Group



Contents

Quality as a concept in software development has become more important and relevant in recent years. Quality is seen, as a whole and as a service, bringing transparency and credibility throughout the products' entire lifecycle. The definition of quality has also expanded to cover not only different areas of product development but also environments, processes, controls, data, and infrastructure. The qualitative aspects of the company's various functions also contribute to the quality needs of the products, as well as quality requirements from various standards and legislation. Quality is an important part of software products, both for customers and end users.

The importance of quality has been emphasized in Qt Group's work, and it is a strategy. Qt Group is a globally recognized leader in cross-platform software development, providing robust tools, frameworks, and quality assurance tools that empower developers to create innovative applications and devices. As technology advances and customer expectations grow, quality has become a pivotal differentiator in delivering exceptional user experiences.

Qt Group ensures that an organization's activities (all business functions) are conducted according to predefined rules, regulations, and standards. It involves adherence to both internal policies and external regulatory requirements (e.g., ISO standards and industry-specific regulations). This includes providing documentation and transparency into its development process, product performance, and internal validation and testing. The types of documentation provided to customers with commercial Qt licenses include but are not limited to, Qt's Quality System, Qt's development process and proof of internal testing, source code, QA practices, test reports, and standards certificates. This white paper outlines Qt Group's commitment to quality and the strategies employed to uphold the highest standards across its products and services.

Quality Vision and Commitment - Approach to the Regulatory Environment/Compliance and Certifications

Quality is just not the amount of bugs and the amount of bugs fixed - it is more than that. As a trusted industry leader, we prioritize maintaining rigorous standards to ensure our customers receive the highest level of service, support, and product quality. A key component of our Quality Assurance Framework and Information Security Management System (ISMS) are the internationally recognized ISO 9001:2015 [11] and ISO/IEC 27001:2022 certifications, together with legislative requirements. Compliance with these standards demonstrates our unwavering commitment to delivering secure and high-quality products. This certification is fundamental to our business validation process, ensuring that our operations consistently meet industry benchmarks and customer expectations. Before engaging in new business partnerships, we conduct thorough assessments using a structured evaluation questionnaire, reinforcing our commitment to operational excellence. By adhering to these stringent quality standards, Qt continues to provide innovative solutions while fostering trust and confidence among our customers. Our focus on quality ensures continuous improvement, allowing us to deliver products and services that consistently exceed expectations.

Qt Group is already actively working on the product's **EU Cyber Resilience Act (CRA)** compliance, and, in several areas, we believe that we are already close to the level that the CRA will require for the kinds of products Qt Group provides. However, we recognize that as a tool and framework provider, we have customers whose own products will need to comply with more strict requirements in the CRA. Given this, we are looking at how Qt Group could potentially better help our customers to be compliant. Qt Group has already made improvements to be compliant with requirements. Improvements are made for security issue handling (released new Support Terms [15] covering the topic); we have recognized the necessity to enhance the documentation of our utilization of third-party tools and to refine our secure software practices, acknowledging that this is an ongoing endeavor. We are releasing SBOM (Software Bill of Materials) documentation as part of our release starting from Qt 6.8.0 (October 2024) onwards; other products are due later. For CVE (public industry de-facto process for Critical Vulnerability and Exploit) handling, we achieved CNA (CVE Number Authority) status in March 2025. We are in the process of implementing and formalizing guidelines for all our software vendors (Third Party Software in Agreements) to ensure compliance with CRA regulations, or to establish a CRA steward for OSS software as necessary. Our software products are currently at different stages of CRA readiness, with the Qt Framework leading our research and development efforts. for OSS software as necessary.

We will also examine how Qt Group's products support the regulatory requirements of the critical area industries. Qt Group also supports customers with safety-critical systems-related products. **Qt Safe Renderer** is certified to ISO 26262:2018-6; ASIL D, -8 section 11; ASIL D (Road vehicles), IEC 61508:2010-3 – 7.4.4; SIL 3 and IEC 61508-3 (electrical/electronic/programmable electronic safety-related systems), EN 50128:2011 6.7.4; SIL 4 (Railway applications), ISO 25119-3 AMD 1:2020 AgPL e (Tractors and machinery for agriculture and forestry), IEC 62304:2015 (2006+A1) C.7, fit-for-use (Medical device software) [13]. **Axivion Static Code Analysis (SCA)** tool is certified for the following safety-relevant standards: ISO 26262 (up to ASIL-D), IEC 61508 (up to SIL 4), EN 50128 (up to SIL 4), EN 50657 (up to SIL 4) and IEC 62304 (up to Class C) [14].

Product Overview with Quality

At Qt Group, quality is not merely a process but a core value embedded in every phase of the software development lifecycle. Our vision is to deliver reliable, secure, and high-performing solutions that exceed customer expectations. Qt Group's products themselves bring customers qualitative know-how to the various stages of product development. The processes, guidances and industrial best practices used by Qt Group in its own product development bring customers certainty that the products are made qualitatively, and transparency guarantees reliability [12].

Qt Development Framework and Tools

Qt is in its sixth generation (Qt 6 was released in December 2020), and the first generation of Qt was released in May 1995, 30 years ago. During this period, Qt has evolved extensively and Qt and its processes have followed suit. Qt Project is based upon an open development model. Anyone who shares that interest can join the community, participate in its decision-making processes, and contribute to Qt's development [1].The KDE Free Qt Foundation has the purpose of securing the availability of the Qt Toolkit.

Qt Development Framework

Qt Framework contains a comprehensive set of highly intuitive and modularized C++ library classes and is loaded with APIs to simplify application development. It produces highly readable, easily maintainable, and reusable code with high native performance and a small footprint—and it's natively cross-platform.

Qt Development Tools

Qt Framework comes with a comprehensive set of developer tools, compilers, internationalization and localization tools, and design tools. Qt Framework can be used with several platform integrated development environments (IDE), Visual Studio, Visual Studio Code as well as with Qt based Qt Creator IDE. Qt Creator is a cross-platform IDE designed to maximize developer productivity when working on embedded solutions. With the Qt AI Assistant, developers can now enjoy smarter coding support. From auto-completion and code optimization to automatic generation of test cases and documentation, the Qt AI Assistant transforms software development for desktop, mobile, and embedded platforms. Qt Design Studio revolutionize your development process by bridging the gap between designers and developers to turn your design visions into production-ready UIs.

Quality Assurance Tools

Squish

Take the complexity out of testing graphical user interfaces (GUIs) and human-machine interfaces (HMIs) – even in the face of product evolution and safety-critical applications. Squish supports agile-oriented teams. Schedule routine or custom-triggered test executions, identify regressions before builds get to QA, and get fast feedback on the commits the team is looking for. Seamlessly automate multi-technology applications or applications with more than one toolkit. Interact with UI controls of each type natively and automatically and focus your efforts on application quality. Squish fully supports Behavior-Driven Development (BDD), an agile testing method that brings together technical and business project stakeholders to bring high-quality products to market.

Coco

Cross-platform & cross-compiler toolchain. Linux, Windows, RTOS, and others. Using gcc, Visual Studio, embedded compilers, and more. Merging multiple execution reports to provide advanced analysis and more outstanding features. Assess and optimize code performance with Coco's built-in Function Profiler.

Test Center

A centralized test result management platform connecting your test automation with your entire development process. View test suite statistics, visualize trends, and analyze historical data of your test executions with built-in, automatic statistical reporting of your imported data. Achieve traceability by optimizing tests and ensuring all requirements are fully realized by synchronizing your results with your tools. Implement automatic result uploads from your Jenkins server for instant analysis of your pipeline projects. Test Center was built from the beginning to be a central, lightweight web database – access it from any device at any time.

Axivion Static Code Analysis

The fully customizable Axivion Static Code Analysis is a cornerstone for your CI-based quality gate. It complements the use of automated testing tools and ensures high-quality code - right from the start. The automated analysis of your software projects identifies violations of coding guidelines according to MISRA (including MISRA C: 2023 and MISRA C++:2023) and AUTOSAR C++14. Security-relevant violations are analyzed with coding guidelines according to CERT®, C Secure Coding and CWE. Metric violations are displayed and documented, as are clones, cycles or unreachable code. The results of Axivion Static Code Analysis support you in the continuous quality assurance accompanying the development of software created in the programming languages C and C++. By automating the quality checks, developers can focus on the tasks that require human intelligence and creativity. Repetitive tasks can simply be handed over to a machine, which repeatedly performs full-scale checks and delivers comprehensive reports.

Axivion Architecture Verification

Implementing architecture verification ensures that the system's structure remains clear and clean. To create the structural model of your software architecture, you can either set it up manually or import it from a UML model or from any other form of graph structure. Software architecture erosion and architecture debt are made transparent and can thus be effectively combated. Deviations are flagged to check if the code complies with the architecture, and further architecture erosion is stopped. Already existing architecture debt can be corrected via targeted refactoring measures, which you can directly monitor in the results report of the architecture verification.

Quality in Software Development

Qt Group follows industry best practices in software development, and internal process descriptions are the starting point for high-quality development. Quality requirements with different controls are part of every transparency phase in software development.

Qt Project Governance Model: The Qt Governance Model is a set of principles and guidelines that define how decisions are made within the Qt project. It is designed to ensure transparency, accountability, and community participation in the development and direction of the Qt software framework. The Qt Governance Model includes structures such as the Qt Project Governance Board, which oversees the project's strategic direction and decision-making processes, as well as community-elected representatives who help shape project policies and priorities. Overall, The Qt Governance Model aims to foster a collaborative and inclusive environment for all contributors and stakeholders involved in the Qt Ecosystem [1].

Requirements Management: The Requirement Management process covers the Life Cycle Management of requirements related to Qt Product(s). The Product and Requirements Management workflow consists of several parts. We maintain a JIRA project with customer and market feedback, as well as User Stories for items which we want to evaluate their relevance.

Software Design and Development: Depending on the conditions, the software is developed based on the maintainership model or based on an adapted version of the Scrum model. Qt SW development teams may utilize various methods from Scrum's agile development methodology. For example, daily work is organized within 2-week sprints and related follow-up practices like sprint planning, sprint reviews, and sprint retrospectives. In the Scrum teams, a defined set of acceptance criteria called Definition of Done is used. These criteria can be set depending on the needs and characteristics of the project. Qt Quality Gate Criteria defines the minimum standard for submitting code to any Qt branch. The submission is any change (addition, modification, or removal) that is submitted to a Qt branch. It may consist of code, comments, documentation, resources, or other content. The development teams are responsible for ensuring all criteria are met before any code is submitted.

Review Process: The changes are reviewed and improvements are proposed in the Gerrit Code Review tool [5]. Once a change is found to be of good quality without obvious defects, it may be approved by the approver or maintainer. In the case of technical disagreements, discussions may be resolved by a maintainer's decision. A maintainer is a person who knows an area of code well and can thus judge other people's changes.

Testing and Validation Process: Qt Group's testing and validation process is designed to detect and resolve defects early, ensuring the highest level of product reliability. Our process includes Automated Unit and Integration Tests, Manual Exploratory Testing, Performance and Scalability Testing, Security and Vulnerability Assessments, and Cross-Platform Compatibility Testing [9].

Release Process: The Release Process provides a set of packages of the developed software that follows the roadmap and fulfills the agreed-upon quality targets before release, supported by robust documentation and customer support. Qt Group releases various software products, ranging from application frameworks to design and developer tools to quality assurance tools. Due to the variety of products and team setups, the individual release processes can differ, but unless defined otherwise in the product-specific instructions, they must still follow the common patterns presented in this process description. Qt Releases are done via a time-based development cycle or a feature-driven one. In time-based releases, the release date is scheduled in advance, and the development effort aims to produce a high-quality release on that predefined date. Features that are not done in time get pushed to the next release. On feature-based releases, the target is to publish the release whenever the needed feature set is ready to be released. Qt Wiki page [2] provides information about the schedule and content of the releases. any code.

- **Major Release (Main Releases):** Change the first digit of the version number. E.g. Qt 5 and Qt 6 are Major releases. Moving from a Major Release to another Major Release is called an Upgrade in Qt Product Licensing Agreements. Supported for one year from the date of the release, unless they are defined in documentation as LTS (Long-Term Support) releases. This can introduce architecture changes, binary breaks, discontinuity in features or functionality.

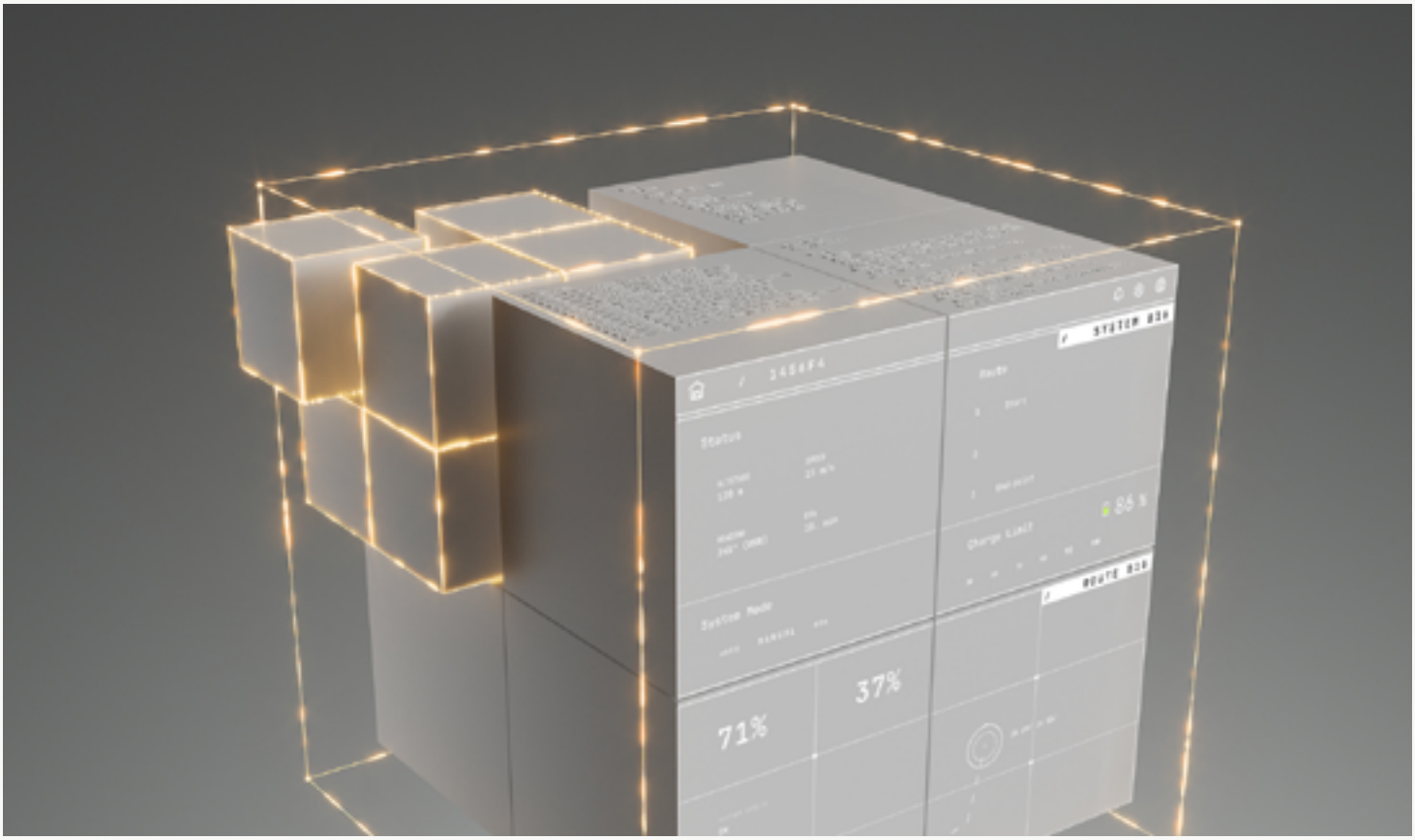
- **Minor Release:** Also known as Feature Releases. Minor Releases introduce new features within a Major Release. Minor Releases change the second digit of the version number. E.g. Qt 6.7 and Qt 6.8 are Minor Releases. Moving from a Minor Release to another minor release is called an Update in Qt Product Licensing Agreements. Supported for one year form the date of the release, unless they are defined in documentation as LTS releases.

- **Maintenance Release:** Also known as Patch Release or Micro Release. They do not introduce new features but focus on quality improvements. Exceptions are updates of 3rd party components or adaptation to operating system updates. Maintenance Releases are delivered when required, they often do not have a roadmap or communicated schedule. Maintenance release does not change the Major or Minor release support period.

- **Long-Term Support Release (LTS):** Standard Support for a release is 12 months (one year) from the date of the Major or Minor Release. Long-Term Support (LTS) Releases are Major or Minor Releases that will be supported for a longer period of time, than the defined 12 months. Examples: LTS period for Qt Framework versions 5.15, 6.2 and 6.5 is 3 years. For Qt 6.8 onwards it is 5 years. LTS releases provide customers with prolonged usability and predictability throughout a clearly defined timeframe. Additionally, The Qt Community also plays a role in maintaining Qt by contributing code, reporting bugs, and providing feedback to help improve the framework.

Defect Management: The Qt Defect Management system is Qt Bug Tracker JIRA [6]. The Qt Wiki page [3] provides instructions for reporting bugs and tips for making high-quality reports. Qt Customer Portal (former Qt Account) enables commercial customers to report issues. Qt Support personnel triage customer-reported defects via Salesforce and, if they are valid, transfer the defect description without customer confidential information to the bug tracker. Qt Support also informs The Qt Company developers about customer-reported defects via a weekly Microsoft Teams notification to the Development channel in the Qt Space. To improve its security and comply with legal requirements, Qt Group has further hardened the robustness of the Qt Framework's protocols and practices as described in Support Terms [15]. Upon identification, reported issues are evaluated to ascertain whether they constitute a genuine security threat. Verified security issues are promptly rectified, with the severity of the issue dictating the urgency of the response. In instances where a security issue originates from a third-party library, Qt Group takes the initiative to inform the concerned party and integrates the resolution in the subsequent maintenance release of the Qt software. All verified security issues are comprehensively documented, also in public Common Vulnerabilities and Exposures (CVE) databases. In addition, The Qt Framework offers an Early Warning List (EWL) to commercial Qt customers for advance notice of verified security issues. The EWL process will be expanded to all Qt Group commercial software products in the future and available for commercial customers regardless of product type.

Documentation: Software product documentation is essential for providing users with the information they need to effectively use and benefit from the software, while also contributing to the overall success and sustainability of the product. Product documentation provides several benefits, including **Improved User Experience:** Clear and comprehensive documentation helps users understand how to use the software effectively, leading to a better overall user experience. **Reduced Support Costs:** Well-documented software can reduce the need for customer support inquiries, as users can find answers to their questions in the documentation. **Faster Onboarding:** Good documentation helps new users quickly get up to speed with the software, reducing the learning curve and increasing productivity. **Enhanced Productivity:** Users can refer to the documentation for guidance on using specific features or troubleshooting issues, saving time, and improving efficiency. **Increased Adoption:** High-quality documentation can make a software product more attractive to potential users, leading to increased adoption and usage. **Improved Collaboration:** Documentation can serve as a central source of information for team members working on the software, promoting collaboration and knowledge sharing. **Better Maintenance and Updates:** Documentation that is kept up-to-date makes it easier for developers to maintain and update the software over time, ensuring its longevity and relevance [3].



Qt Product Security Overview

With over 30 years of successful use (the first public version of Qt Framework was released in 1995) across more than 70 industries worldwide, Qt Group delivers battle-proven technology particularly suited for building high-quality, secure applications. Providing access to the Qt Framework source code, Qt Group is a reliable partner with dedicated QA tools offering and a large customer and partner ecosystem whose collaboration, use cases, and feedback help mitigate cybersecurity risk.

Qt Group has implemented regular processes to reduce the risk of introducing security vulnerabilities into the Qt product's code-base and releasing them to other Qt users. A Core Security Team of Qt developers was established with the responsibility of ensuring that this policy is followed. The established code review process and commit policy prevent bad or compromised actors from committing malicious code or backdoors to the Qt codebase.

The security architecture and process of the software development environment consists of different controls in SW development transparency phases:

- SW development process describes the methods applied during development
- Qt Quality Management manual is a document of SW development phases and practices. It is available for commercial customers by request.
- Qt Framework source code is publicly available in the Qt Project Git Repository [4].
- Code review results: Qt's development is both auditable and transparent with a mandatory code review process with clearly defined requirements, a merit-based review system, a continuous integration system, and frequent releases with security updates. Software development is organized in The Qt Project under open governance. The Qt Code Review Process includes mandatory review for all changes (patches), merit-based approver/maintainer roles, and code review and code commit policies. Qt security-critical code is reviewed and tested according to cybersecurity industry best practices. Software development processes are consistently reviewed and updated so that critical code is captured and identified for this process.

- Public documentation gives transparency to the products [3].

- Continuous Integration: All changes must pass the Continuous Integration System. This includes compilation tests in various platforms and configurations, unit tests to ensure no functional regressions, and documentation and license scans. This approach enhances efficiency, reduces human error, and enables rapid feedback loops.

- Test results/metrics are available from different levels. Those are part of release-specific delivery reports and are available for commercial customers by request.

- The Core Security Team monitors the defects/bugs, Common Vulnerabilities and Exposures (CVE) database for vulnerabilities in third-party components and coordinates the application of necessary patches with the module maintainers. EU CRA requirements are followed. JIRA is a follow-up tool. Release notes include information on known issues.

- Product-Level Risk Management is a part of processes.

- The Qt Code goes through extensive static analysis and fuzzing. Static Code Analysis tools used include Qt's own static analysis tool, Axivion, along with other tools independent of the Qt product line. Qt API is fuzz-tested using OSS-fuzz, and the results are publicly available. Additional measures taken include clone management, cycle detection, and detecting unreachable code. High-priority issues discovered through static code scans and fuzz testing are reported as security issues and addressed before the next release.

- Qt Source Code contains code from third-party origins. The third-party code is regularly updated before Qt Releases. Security-critical third-party code is blocked from Qt Releases unless it is up to date with its most recent release.

- For each release, antivirus tools scan the Qt Installer and other binary content in the released packages.

- Long-term Support (LTS) releases are especially important for regulated industries, which must adhere to additional quality and safety requirements. With its LTS policy, Qt Group ensures that example medical software built with older versions of Qt will be maintained over long periods of time [10]. Medical device manufacturers can trust that Qt code is always supported without the need to move to new versions and request new certifications. EU CRA requirements are followed.

Ecosystem

The Qt Software Ecosystem offers a powerful and versatile platform for developing cross-platform applications with rich features, performance, and community support. A large ecosystem offers several benefits. **Cross-Platform Development:** Qt enables developers to write code once and deploy it across multiple platforms, including desktop, mobile, and embedded systems, saving time and effort in development. **Rich Library of Tools and Libraries:** Qt provides a comprehensive set of tools, libraries, and APIs that make it easier for developers to create feature-rich and interactive applications. **Performance and Efficiency:** Qt is known for its high performance and efficiency, making it suitable for developing applications that require fast response times and smooth user experiences. **Community Support:** The Qt Community is active and vibrant, offering support, resources, and collaboration opportunities for developers working with the Qt framework. **Modular Architecture:** Qt's modular architecture allows developers to use only the components they need, reducing bloat and improving application performance. **Integration with Third-Party Tools:** Qt integrates well with various third-party tools and technologies, enabling developers to leverage existing resources and workflows. **Customizability and Flexibility:** Qt provides a high degree of customizability and flexibility, allowing developers to tailor their applications to meet specific requirements and design preferences. **Long-Term Support and Stability:** Qt is known for its long-term support and stability, ensuring that applications built with Qt will remain functional and relevant for years to come.

Processes

Qt Group has processes available for all needed functions so that processes can help organizations achieve better quality by providing structure, consistency, and opportunities for continuous improvement. Processes are based upon standardization, regulatory requirements, and industry best practices. Implementing well-defined processes helps standardize activities, making it easier to identify and address quality issues. Processes ensure that tasks are carried out consistently, reducing the likelihood of errors and variations that can impact quality. By establishing processes that include feedback mechanisms and performance metrics, organizations can continuously monitor and improve quality over time. Well-designed processes can help identify potential sources of errors and deviations and implement controls to prevent them, thus enhancing quality. Streamlining processes can help eliminate waste and inefficiencies, leading to improved quality by focusing resources on value-adding activities. We are continuously improving and evolving our processes, latest due to CRA.

Data Handling Quality

Qt Group incorporates customer data handling into quality aspects, ensuring data security, accuracy, compliance with regulations, proper data retention and deletion practices, and strict access controls. By prioritizing these aspects, organizations can maintain the quality and integrity of their customer data-handling processes. Customer data handling is an important aspect of quality for any organization that deals with sensitive or personal information. Data Security, ensuring that customer data is stored securely and protected from unauthorized access, is crucial for maintaining quality. Implementing encryption, access controls, and regular security audits are key components of data security. Quality customer data handling involves ensuring that the data stored is accurate and up-to-date. Regular data validation and verification processes can help maintain data accuracy.

Quality aspects of customer data handling include compliance with data protection regulations such as GDPR, CCPA, or industry-specific regulations. Ensuring that customer data is handled in accordance with these regulations is essential for maintaining quality and trust. Proper handling of customer data includes defining data retention policies and procedures for securely deleting data when it is no longer needed. Managing data retention and deletion processes is important for maintaining quality and compliance. Implementing strict access controls and permissions to ensure that only authorized personnel have access to customer data is crucial for maintaining data quality. Regularly reviewing and updating access controls can help prevent unauthorized access to customer data.

Risk Management

Risk Management is an integral part of quality management that involves identifying, assessing, and mitigating risks that could impact the quality of products or services. Risk Management Process ensures that Qt Group proactively identifies, mitigates, and monitors risks while maintaining compliance, cybersecurity resilience (CRA), and sustainability transparency (CSRD) or other industry-specific requirements. The Risk Management Process provides high-level central guidance for the primary execution threads controlling workflow. Subprocesses described here are secondary processes that perform specific tasks and are dependent on the primary process (but can sometimes run independently). Our primary process governs the entire system's operation, whereas subprocesses run assigned tasks and report back to the primary process.

The Risk Management Process provides a structured framework for identifying, assessing, and managing risks that may impact Qt Group's operations, objectives, software products, and compliance obligations. This process aligns with corporate risk management policies and ensures compliance with legislation, industry standards, and requirements from valid certifications. The Risk Management Process aims to identify, analyze, treat, and monitor risks continuously. The process determines the scope and strategy of risk management to be performed. Overall, Risk Management plays a critical role in ensuring product quality by helping organizations anticipate and address potential threats before they impact product quality. This enhances customer satisfaction and minimizes risks to the organization's reputation and bottom line.

Environments and Infrastructure

Qt Group's environments and infrastructure are the starting point for high-quality product development. Quality aspects related to the environment and infrastructure are sustainability, so that infrastructure and development projects are sustainable and have minimal impact on the environment. Examples of these include:

- Using resources efficiently and minimizing waste in Construction and infrastructure projects.
- Incorporating green spaces, sustainable drainage systems, and renewable energy sources into infrastructure projects
- Building infrastructure that can withstand natural disasters and climate change impacts.
- Ensuring that infrastructure projects are designed and built with the safety and health of workers and the public in mind.
- Designing infrastructure to be accessible to all, including people with disabilities, the elderly, and those with mobility challenges. Following industry best practices and quality standards to ensure that infrastructure projects meet regulatory requirements and are built to last. Conducting thorough assessments of the environmental impact of infrastructure projects and taking steps to mitigate any negative effects. Ensuring that infrastructure is properly maintained and repaired to prolong its lifespan and prevent deterioration.

Sustainability as a Part of Quality

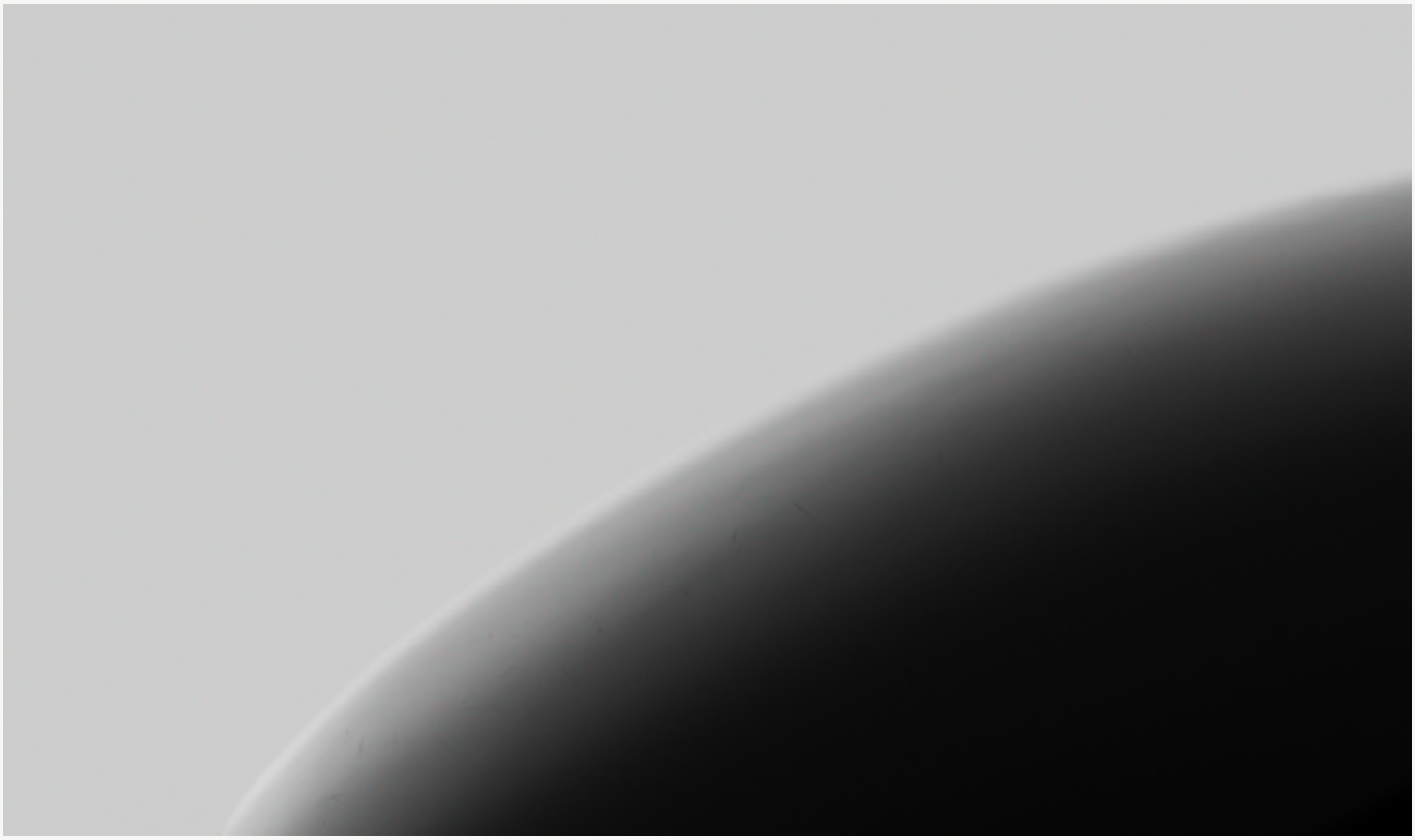
Sustainability reporting and global policies are part of Qt Group's quality practices. Qt Group is committed to transparent sustainability reporting, and we have now published our first sustainability statement in accordance with the European Union's Corporate Sustainability Reporting Directive. You can find the report on our website [8].

As part of the Double Materiality Assessment conducted in 2024, Qt Group and its stakeholders did not identify any environmental standards as material within Qt Group's value chain; hence, the report does not include emission calculations or targets from scopes 1-3. However, Qt Group has recognized that such matters might become material in future double materiality assessments and will report them accordingly.

We have shared our office consumptions in our ESG report, the latest being from 2023. From the report, you can see that four of our offices already use 100% renewable energy and we are continuously seeking to find possibilities to extend the coverage to a more significant number of offices. Also, as a part of our CSRD-related work, Qt Group has recognized some improvement areas related to sustainability matters, and we are committed to and have already started developing internal processes and practicalities towards even more transparent reporting. Qt Group reports sustainability matters once a year in Q1 as a part of our annual report [8].

Customer Feedback Integration

Customer feedback plays a crucial role in our quality improvement initiatives. Qt Group maintains open communication channels with our users through forums, support portals, and beta programs, enabling us to identify pain points and prioritize enhancements.



Quality Benefits to Our Customers

Customers benefit from Qt Group's focus on quality, compliance, security, reliability, sustainability, and customer-centric practices, which enhance user experiences and build trust and confidence in the products and services provided. Customers benefit from transparency and credibility throughout the product lifecycle, ensuring trust in the products and services provided by Qt Group. Qt Group's adherence to internationally recognized ISO standards and industry-specific regulations ensures that customers receive secure and high-quality products, demonstrating the commitment to delivering exceptional service and product quality. Qt Group's commitment to meeting regulatory requirements, such as the EU Cyber Resilience Act (CRA), ensures that customers can rely on compliant products that meet industry standards and legislative requirements. Qt Group's focus on delivering reliable, secure, and high-performing solutions ensures that customers receive products that exceed expectations in terms of performance, security, and reliability. Customers benefit from robust security measures, accurate data handling practices, and compliance with data protection regulations, ensuring the confidentiality and integrity of their data. Qt Group's Software Ecosystem offers a powerful platform for developing cross-platform applications with rich features, performance, and community support, enabling developers to create innovative solutions efficiently. Qt Group's Risk Management Process helps identify, assess, and mitigate risks that could impact product quality, enhancing customer satisfaction and minimizing risks to the organization's reputation. Qt Group's commitment to sustainability and transparent reporting ensures that infrastructure projects are sustainable and have minimal impact on the environment, providing customers with environmentally responsible products and services. Qt Group's open communication channels with customers enable the integration of customer feedback into quality improvement initiatives, ensuring that customer needs and preferences are prioritized. Commitments to commercial customers are are available on Qt Groups's webpage [16].

Conclusion

At Qt Group, we are committed to delivering exceptional quality in all our products and services. As a global leader in software development, we strive to meet and exceed the expectations of our customers, partners, and stakeholders by fostering a culture of continuous improvement and innovation. Quality is the foundation of The Qt Group's success, driving customer satisfaction and long-term business growth. By fostering a culture of quality and leveraging cutting-edge technologies, The Qt Group continues to set new benchmarks for excellence in the software development industry. Quality is integral to Qt Group's mission to empower developers and businesses with world-class software solutions. By adhering to quality practices, we aim to deliver unmatched value, build lasting relationships, and maintain our position as a trusted industry leader.

Contact Information

For more information on our quality practices, please visit Qt Group's website [12] or contact Qt Group's Account Support Center [7].

More Information

[1] https://contribute.qt-project.org/quips/
[2] https://wiki.qt.io/QtReleasing
[3] https://doc.qt.io/
[4] https://code.qt.io/cgit/
[5] https://codereview.qt-project.org/
[6] http://bugreports.qt.io
[7] https://www.qt.io/qt-support/
[8] https://www.qt.io/investors/governance/esg
[9] https://testresults.qt.io/
[10] https://www.qt.io/terms-conditions
[11] https://www.qt.io/group/iso-9001
[12] https://www.qt.io/
[13] https://www.qt.io/product/functional-safety-and-qt
[14] https://www.qt.io/quality-assurance/axivion-static-code-analysis
[15] Support Terms 2024-10 or later; https://www.qt.io/terms-conditions/support-terms , Chapter 4 Premium Support, Chapter 5 Extended Support, Chapter 9 Handling of Security Issues and Exhibit for Qt Developer Framework, chapter 3 Extended Security Maintenance.
[16] https://www.qt.io/terms-conditions/support-terms

Qt Group

White paper | www.qt.io