



QualityClouds

ServiceNow Development Report 2023



Contents

Key Findings	4
Customization vs Configuration	7
What are our customers fixing in their ServiceNow instances?	15
Analysis of Configured Applications	17
Emphasis on Functional Training	18
In Summary	19

Management Summary

As we navigate through the evolving landscape of ServiceNow development, this report presents a comprehensive analysis of trends, challenges, and opportunities that have emerged over the last year. Our focus has been on identifying patterns that can guide architects in optimizing ServiceNow implementations for efficiency, security, and scalability. The report took into consideration the configuration data gathered in more than 115 ServiceNow instances during the period July 2022 to June 2023.

Key Findings

Increase of Low-code/no-code techniques

The increase in the proportion of new applications developed as extensions on ServiceNow, showcases the platform's adaptability and innovation. A rising extension ratio, like the 8% seen in the first half of 2023, indicates increased use of low-code/no-code solutions within our customers

Customization vs Configuration

The ratio of custom code to configurations remains a critical factor. While there's an evident inclination towards low-code/no-code solutions, pro-code development continues to play a significant role, especially in complex customizations. From all the instances analyzed, the average configuration ratio was 67% (using built-in options and tools provided by ServiceNow, and avoiding the introduction of new code in the instance)

Code and Configuration Health

Lines of Code (LoC) have seen a year-on-year increase, raising concerns about potential technical debt and maintainability. Configuration Elements (CEs) show a significant increase. This may indicate either a wider use of the functionality in the ServiceNow instances or a growing complexity in their configuration.

Quality and Security

The number of security vulnerabilities in libraries has increased, as well those issues related to the maintenance of the user permissions tables. This necessitates a stronger focus on security in development practices, as well as enhanced vigilance by instance administrators in ensuring that all code runs with the minimum required privileges. In those instances scanned for the first time, technical debt has risen, with average tech debt hours per instance increasing. This highlights the need for more quality assurance processes.

Governance and Compliance

Despite improvements in addressing common issues, challenges remain in adhering to best practices, especially in areas like naming conventions and DOM manipulation. This calls for enhanced governance frameworks.

Key Findings

Applications and Modules

The most utilized applications include Human Resources and Incident Management, while applications like Service Catalog and Change Request show high configurability, which doesn't come as a surprise. The analysis of issues per application underscores the necessity for targeted training and support.

Upgradeability and Future-Proofing

The gradual adoption of newer ServiceNow versions implies a proactive approach to ensure smooth upgrades and maintain compatibility with future releases. This can be achieved by avoiding changes of out-of-the-box elements and having a proper regression testing strategy.



Strategic Recommendations



Emphasize Security - Integrate robust security practices at every stage of development to address the rising vulnerabilities. Check the code and configuration thoroughly before deploying into production.

Manage Technical Debt - Implement strategies for regular debt reduction and prevention to maintain a healthy codebase. This will help you to cut maintenance costs and increase delivery speed.

Strengthen Governance - Enhance governance frameworks to ensure adherence to best practices and compliance standards.

Focus on Training and Support - Develop targeted training programs for your developers to address common issues in heavily used and highly configured applications. Do the same for citizen developers so you can adopt low-code / no-code strategy at scale.

Based on our learnings gathered exploring more than 115 ServiceNow instances in the last year, and as you continue to evolve in your ServiceNow journey, these insights and strategies will be pivotal in steering our development efforts towards greater efficiency,

security, and overall success.

Customization vs Configuration

Within this dynamic landscape of ServiceNow development, understanding the balance between configuration and customization is vital for platform owners and architects. This section, derived from data across 115+ customers over the last year, presents the evolving trends in configuration versus customization, including extensions through new app developments.

Key Findings

Our analysis reveals notable trends over three consecutive semesters (Jan 2022 - June 2023), offering insights into the evolving preferences in ServiceNow development.

Overall Trend - There is a consistent increase in the configuration ratio, indicating a growing preference for leveraging native ServiceNow features and capabilities. This trend points towards a strategic shift in development practices, favoring configurations that are easier to maintain and upgrade.

Configuration Ratio - Starting from 64.09% in the first half of 2022, the configuration activities steadily increased to 66.96% by mid-2023. This rise reflects an industry-wide move towards optimizing existing ServiceNow functionalities to meet business needs, minimizing the need for extensive custom coding. The customization activities have seen a slight decline, from 27% in early 2022 to 25% in 2023. This gradual decrease suggests a cautious approach towards customizations, likely due to the complexities and long-term maintenance challenges they can introduce.



Customization vs Configuration

Extension through New Apps - The extension ratio, representing new app developments, showed variability - peaking at 11% in late 2022 before dropping to 8% in mid-2023. This fluctuation may indicate that we are still in a journey of adopting citizen developer practices and considering ServiceNow to build bespoke applications.



Interpretation

Configuration as a Preferred Approach - The increasing reliance on configuration highlights a strategic preference for scalable, maintainable, and less resource-intensive solutions. This approach aligns with a need for agility and adaptability in corporate IT environments.

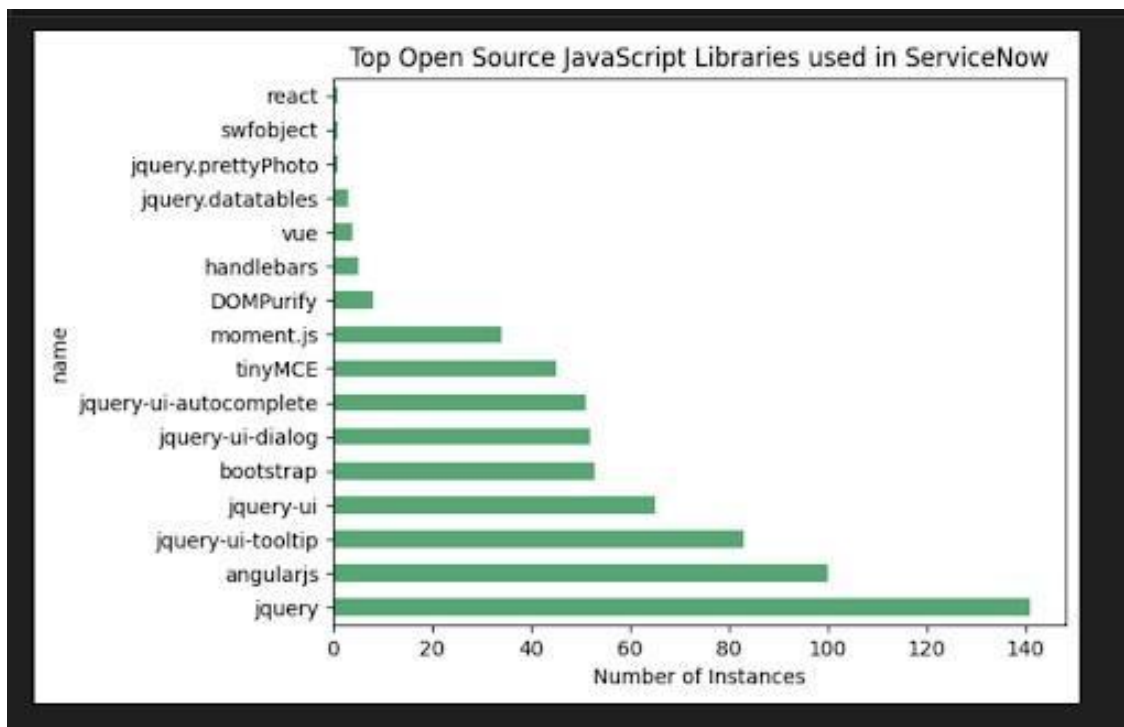
Customization for Specific Needs - While customization, expressed as the number of configuration elements which add lines of code to an instance, is on a slight decline, it remains an integral part of ServiceNow development. This sustained presence underscores the necessity for tailored solutions in areas where out-of-the-box functionalities fall short.

Selective Extension Development - The variability in new app development suggests a selective, need-based approach rather than a consistent investment. This trend could reflect a focus on developing extensions that offer significant value or address specific business challenges. The trend analysis from the past year reveals a clear shift towards maximizing the use of built-in ServiceNow capabilities through configuration. While customization still plays a significant role, there is a cautious approach towards it, likely due to the challenges in long-term management and the potential to introduce technical debt. Extension development through new apps appears to be driven by specific business needs rather than a consistent strategy. These insights should guide platform owners and architects in making informed decisions about balancing configuration and customization in their ServiceNow environments.

OSS Libraries and Top 10 Vulnerabilities

OSS Libraries and top 10 vulnerabilities

Open-source JavaScript (JS) libraries are integral to extending functionality and enhancing user experiences. However, the use of such libraries comes with the responsibility of managing potential security vulnerabilities. This section delves into the prevalent JS libraries in the ServiceNow ecosystem and the top vulnerabilities associated with them.



Prevalent JavaScript Libraries

Our analysis indicates a diverse range of open-source JS libraries utilized within ServiceNow. The most commonly incorporated library is jQuery, which serves as the foundation for several other plugins like jquery-ui-dialog, jquery-ui-autocomplete, and jquery-ui-tooltip. Bootstrap and AngularJS are also widely adopted, providing robust frameworks for UI components and single-page applications, respectively. Other libraries such as Vue, React, and moment.js are also utilized, albeit to a lesser extent, showcasing the platform's flexibility in integrating with various JS tools.

Top Vulnerabilities

Alongside the utility these libraries offer, our findings reveal a series of vulnerabilities that development teams must be vigilant about:

Cross-Site Scripting (XSS) - A significant number of libraries, including AngularJS, jQuery, and Bootstrap, have been identified with XSS vulnerabilities. These weaknesses could allow attackers to inject client-side scripts into web pages viewed by other users, potentially leading to information theft or unauthorized access.

Regular Expression Denial of Service (ReDoS) - Libraries like moment.js have been flagged for ReDoS vulnerabilities, where inefficient regular expressions can lead to disproportionately high CPU usage, causing a denial of service.

Security Misconfigurations - Various versions of Bootstrap and AngularJS have been noted for security misconfigurations, potentially exposing applications to various attack vectors.

Addressing the Vulnerabilities

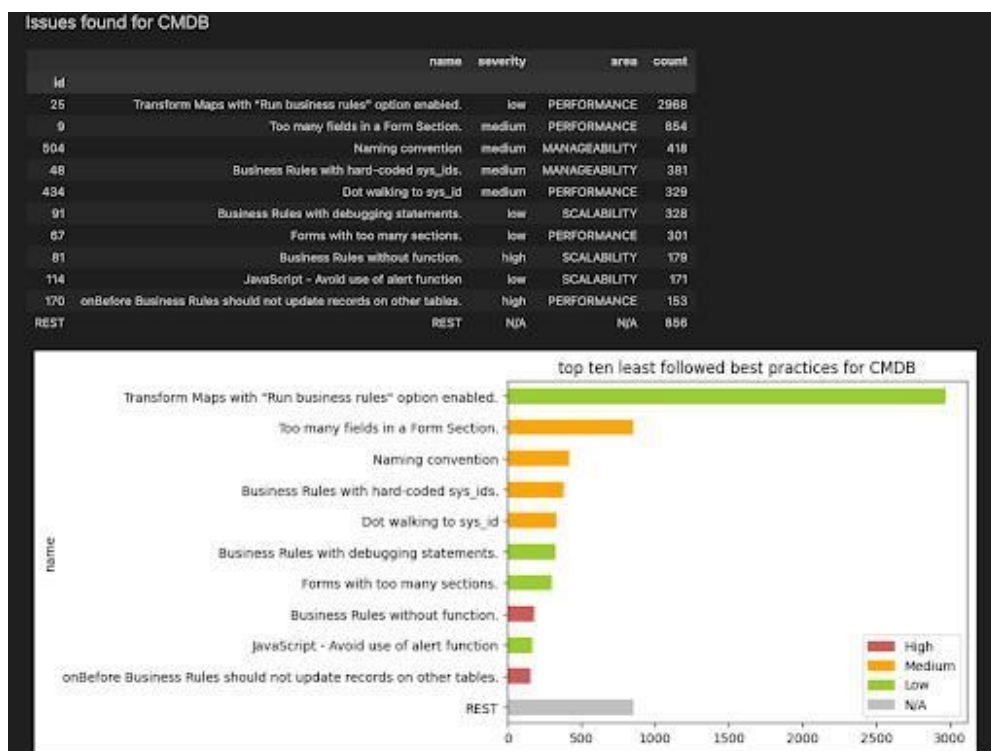
To mitigate these risks, it is crucial for ServiceNow developers to:

1. **Update Libraries Regularly:** Ensuring that all JS libraries and their dependencies are up-to-date is fundamental in protecting against known vulnerabilities.
2. **Code Auditing:** Implementing thorough code reviews and automated security scans can help detect and rectify security flaws.
3. **Adopting Secure Coding Practices:** Training developers in secure coding practices can prevent common vulnerabilities from being introduced in the first place.
4. **Leveraging Security Plugins:** Utilizing plugins and tools designed for security can help safeguard applications against exploitation.

The use of open-source JS libraries in ServiceNow development brings considerable benefits in terms of functionality and user experience. However, it is accompanied by the responsibility of managing security risks. By staying informed about the prevalent vulnerabilities and taking proactive measures to address them, development teams can ensure the robustness and security of their ServiceNow applications.

CMDB Development Insights

The Configuration Management Database (CMDB) is a fundamental component within ServiceNow, serving as the backbone for managing an organization’s IT infrastructure. A well-maintained CMDB supports various IT processes, but misconfigurations can lead to significant challenges. The insights from our data pinpoint critical areas where CMDB configurations often fall short.



Common CMDB Configuration Issues

Our findings from an analysis of CMDB-related practices across ServiceNow applications reveal a series of common issues:

Performance Hurdles - The most frequently occurring problem involves ‘Transform Maps with “Run business rules” option enabled’, which can dramatically impact performance due to unnecessary processing.

CMDB Development Insights

Manageability Concerns - A substantial number of instances were flagged for having 'Too many fields in a Form Section' and for violations of 'Naming conventions'. These manageability issues can complicate the administration of the CMDB and lead to inefficiencies. It can also imply performance problems.

Customization Pitfalls - 'Business Rules with hard-coded sys_ids' and 'Dot-walking to sys_id' were identified as prevalent problems. These practices hinder scalability and can cause difficulties during upgrades or when integrating with other systems.

Scalability and Maintainability - Instances of 'Business Rules without function' and the misuse of 'JavaScript - Avoid use of alert function' indicate a disregard for scalable and maintainable code practices. Additionally, 'onBefore Business Rules should not update records on other tables' is a high-severity issue that can lead to performance degradation and data integrity concerns.

Implications and Recommendations

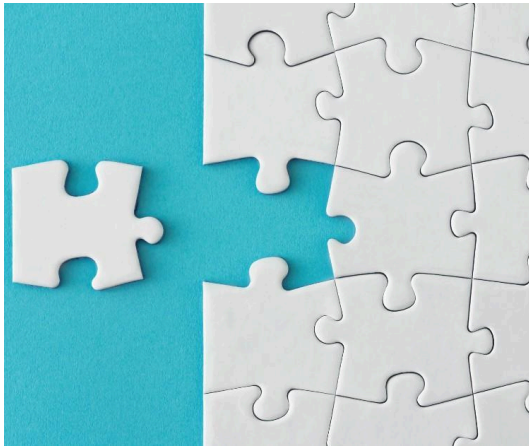
The recurring issues in CMDB configurations suggest a need for better adherence to best practices in ServiceNow development. Enhancing performance, manageability, and scalability should be prioritized through:

1. **Regular Audits** - Conducting periodic reviews of the CMDB setup to identify and rectify common misconfigurations.
2. **Training and Governance** - Establishing strong governance around CMDB changes and providing training on best practices to developers and administrators.
3. **Automated Tools** - Implementing automated tools for enforcing naming conventions and identifying performance bottlenecks can help



maintain the CMDB's integrity.

ServiceNow HR Application



The ServiceNow HR module is a critical application for managing employee services. 33% of the instances analyzed had the plugin active and some configuration on top.

Organizations adopt ServiceNow's HR application to streamline and centralize HR processes, enhancing efficiency and data accuracy. Its automation and self-service capabilities significantly

reduce administrative workload, allowing HR teams to focus on strategic tasks. The application's service management approach improves employee experience and satisfaction. Additionally, its robust analytics tools provide valuable insights for informed decision-making in HR strategies. This integration of functions and data-driven management modernizes HR operations and aligns them closely with organizational goals

Our annual analysis has identified recurring configuration issues that can affect the performance, scalability, and manageability of the HR module. Addressing these issues is paramount for ensuring a streamlined HR management process and ensuring a good employee experience.

Predominant Issues in HRR Module Configuration

Hard-Coded Elements - The prevalence of hard-coded `sys_ids` in Inbound Email Actions and Catalog Client Scripts poses significant risks to scalability and maintainability, locking configurations to specific instances that are prone to failure upon changes.

Alert Function Usage - The use of JavaScript alert functions within the HRR module, although less severe, still raises concerns for scalability as they can interrupt automated workflows and degrade user experience. As Quality Clouds normally analyze pre-production instances, this may be

something

known by development teams but quite often we see the same issue in productive environments.

Trends and Observations

Excessive Fields in Form Sections - Forms with too many fields can overwhelm end-users, negatively impacting the usability of the HHRR application.

Unnecessary Tables in Scoped Applications - The creation of unnecessary tables within scoped applications can increase licensing costs and complicate data architecture. The fact that this issue is often detected in the HHRR module points to the specificity of these processes in each company, which seems to require additional data points not available in the out of the box version of this plugin.

Recommendations for Improvement

To enhance the efficiency and reliability of the ServiceNow HHRR module, the following recommendations should be considered:

Data Dictionary Audits - Regularly review the data dictionary to ensure all entries correspond to existing tables to prevent data integrity issues. This will also help the organization's Chief Data Officer in their work (data quality and integrity, data standardization, identifying gaps and redundancies and facilitating data integration and interoperability)

Refactor Hard-Coded Elements - Replace hard-coded sys_ids with dynamic references to maintain flexibility and ease of maintenance.

Streamline Forms - Simplify forms by ensuring only necessary fields are included, enhancing user experience and data entry efficiency.

Cost Management - Evaluate the necessity of each table within scoped applications to manage and reduce unnecessary licensing costs. Consider re-designing internal processes to adapt to industry standards whenever possible.

Trends and Observations

Gradual Adoption of Newer Versions: There appears to be a progressive shift towards newer ServiceNow versions over the months. Early months show higher percentages for older versions (like Quebec and Rome), while later months indicate increased adoption of newer versions such as San Diego and Tokyo.

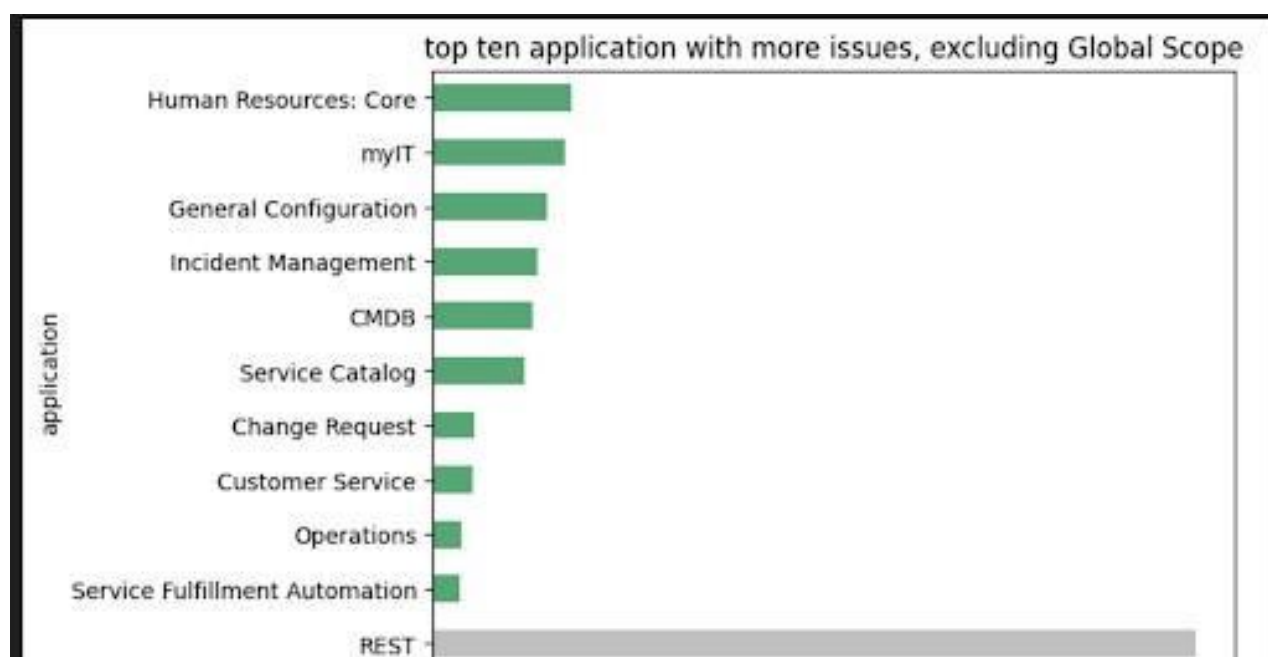
What are our customers fixing in their ServiceNow instances?

Phased Out Versions - The older versions, notably Quebec, show a diminishing trend as the year progresses, suggesting that instances are being updated to newer versions or that fewer scans are being conducted on these older versions.

Dominance of Specific Versions - Throughout the year, certain versions remain dominant. For example, the Rome version appears to have the highest percentage of scans consistently across most months, indicating widespread usage and possibly a preference for stability over immediate updates.

Introduction and Growth of Newest Versions - The latest versions, San Diego and Tokyo, start appearing partway through the year, with their presence increasing towards the later months. This implies an ongoing transition phase where organizations are adopting newer releases.

Upgrade Strategies - Since ServiceNow enforces a policy of only supporting “latest minus one” releases to remain in support, there are basically only two available upgrade strategies, which are to upgrade to every new version as it becomes available, or to skip one version and upgrade every two. Our data shows that the first strategy is the most popular, with only 27 of the over 115 instances scanned having skipped a version in their upgrade journey. Of the skipped versions, San Diego was most often skipped, with 19 of the 27 instances having skipped that version.



Insights from Issue Resolutions

Our customers addressed a variety of issues, reflecting focused efforts to optimize system functionality and governance. The following insights are derived from the most frequently resolved issues:

User and Group Integrity - A high number of fixes were related to roles assigned to invalid users and groups with inactive members or managers. This signifies a keen awareness of the importance of accurate user role management to maintain system security and integrity.

Adherence to Naming Conventions - The widespread resolution of issues regarding naming conventions, created by our customers, underscores the value of a standardized approach to naming for easier management and clarity.

Job Scheduling and Execution - There was a notable focus on correcting configurations of scheduled jobs, either lacking a dedicated integration user or being run by deleted users. This reflects an understanding of the need for reliable job execution and the risks associated with improper user assignments.

Security and Privacy Concerns - The resolution of issues like making reports public, utilizing alert functions in JavaScript, and having dictionary entries for non-existent tables indicates a prioritization of system security and data privacy.

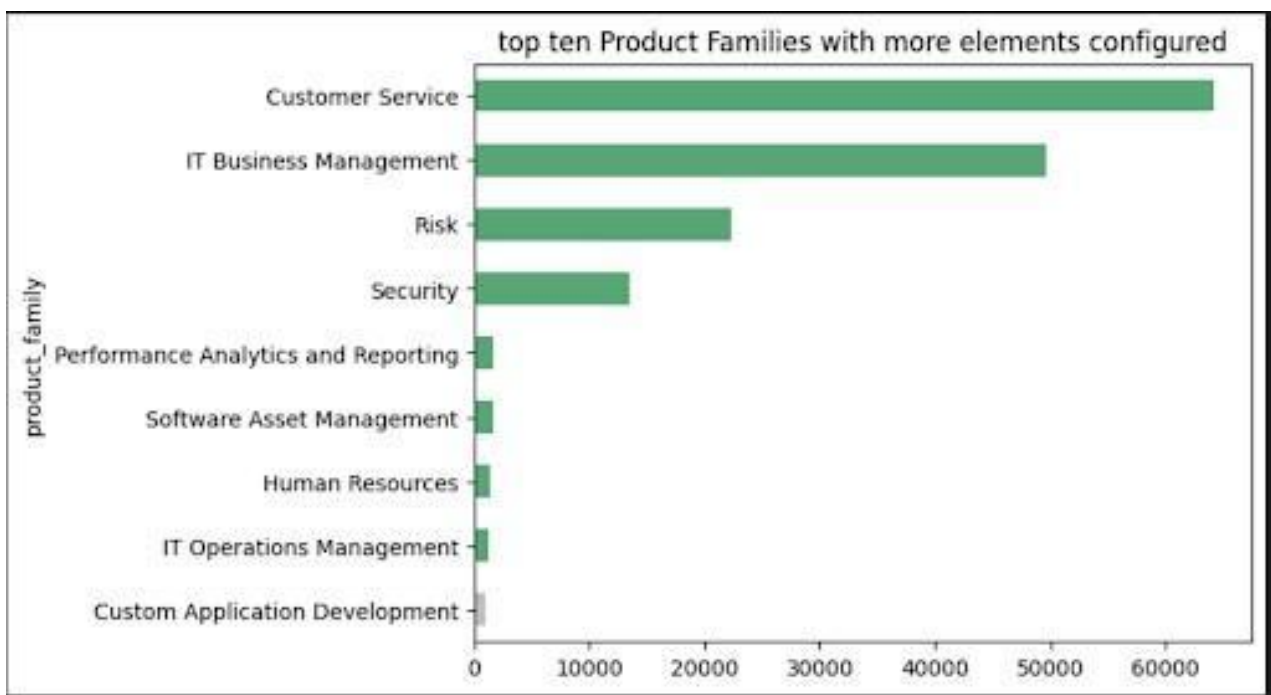
Best Practices in Development - The correction of development practices, such as the avoidance of hard-coded sys_ids and the use of debugging statements in script includes, demonstrates a commitment to clean, maintainable, and scalable code.

The active resolution of these issues highlights the dedication to maintaining a secure, efficient, and cost-effective ServiceNow platform. By continuing to prioritize these areas, platform owners and architects will not only ensure smooth operations but also support the overall strategic objectives of their organizations.

ServiceNow Application Configuration: Trends and Analysis

ServiceNow's flexibility is exemplified by the diverse range of its configured applications. The data on the most configured applications by our customers, encompassing both core platform components and plugins, reveals a strategic emphasis on enhancing customer service, IT business management, and risk assessment capabilities. This section provides an analysis of these trends and outlines key considerations for ServiceNow platform development.

Analysis of Configured Applications



The data indicates the following trends in application configuration:

Customer Service Dominance - Customer Service applications have seen the most configurations, highlighting the prioritization of customer support and engagement as central to business operations.

IT Business Management - IT Business Management (ITBM) applications are also highly configured, signifying the importance of aligning IT services with business needs, project management, and agile development processes.

Focus on Risk and Security - There's a significant configuration of Risk and Security applications, pointing towards heightened attention to compliance, threat assessment, and mitigation within digital workflows.

Asset and Operations Management - Software Asset Management and IT Operations Management configurations suggest that businesses are concentrating on optimizing software investments and ensuring the reliability of IT services.

Emphasis on Functional Training

Human Resources and Reporting - Human Resources (HR) and Performance Analytics and Reporting configurations indicate an investment in workforce management and data-driven decision-making.

Emphasis on Functional Training Prior to Configuration

Before embarking on configuration or development within ServiceNow modules, it is imperative to conduct thorough functional training. This preparatory step is crucial as it ensures that teams fully understand the existing capabilities of the ServiceNow platform, which often includes out-of-the-box functionalities that may already fulfill business requirements. Adequate training minimizes the risk of redundant customizations, promoting a more efficient use of the platform's extensive features. It is recommended that all team members tasked with configuration or development activities complete module-specific training. Furthermore, leveraging the expertise of a ServiceNow solution consultant is advisable whenever there is uncertainty. These professionals can provide valuable insights into the platform's capabilities, helping to clarify whether the functionality sought by the business is already available within ServiceNow, thereby streamlining the development process and aligning it more closely with organizational needs.



In Summary

As we conclude this comprehensive analysis of the ServiceNow development landscape in 2023, it is evident that the evolving complexities and escalating demands of ServiceNow environments present both challenges and opportunities. The increasing intricacies in customization vs. configuration, escalating concerns over code and configuration health, as well as the ever-evolving landscape of security vulnerabilities, underscore an urgent need for robust governance and adept management of ServiceNow platforms.

In this critical juncture, Quality Clouds emerges as a pivotal partner for platform owners and architects seeking to govern their ServiceNow development with precision and foresight. By leveraging Quality Clouds, organizations can expect a drastic reduction in costs, a significant decrease in time-to-market, and a marked increase in development speed. Our findings, drawn from an extensive analysis of over 115 ServiceNow instances, demonstrate that proactive management and optimization of these platforms are no longer a luxury but a necessity to stay competitive and efficient.

For those not yet utilizing Quality Clouds, this moment serves as a crucial wake-up call. The landscape of ServiceNow development is rapidly evolving, and the risks of falling behind are increasing. Delayed action could lead to compounded technical debt, security vulnerabilities, and inefficiencies that could be detrimental in the long run. Now is the time to act, to embrace the advanced capabilities of Quality Clouds, and to secure a future where your ServiceNow instance is not just a tool, but a strategic asset driving your organization towards unparalleled success.

We thank you for your continued support.

CONTACT

20-22 Wenlock Rd / London N1 7GU
224 W 35th St, Suite 500 PMB 112 / New York, NY 10001
Carrer de Torres i Amat 21, 1^o / 08001 Barcelona
LONDON / NEW YORK/ BARCELONA

www.qualityclouds.com | sales@qualityclouds.com | [@QualityClouds](https://www.instagram.com/QualityClouds)