CAPCO

A ROADMAP TO SOFTWARE CURRENCY MATURITY

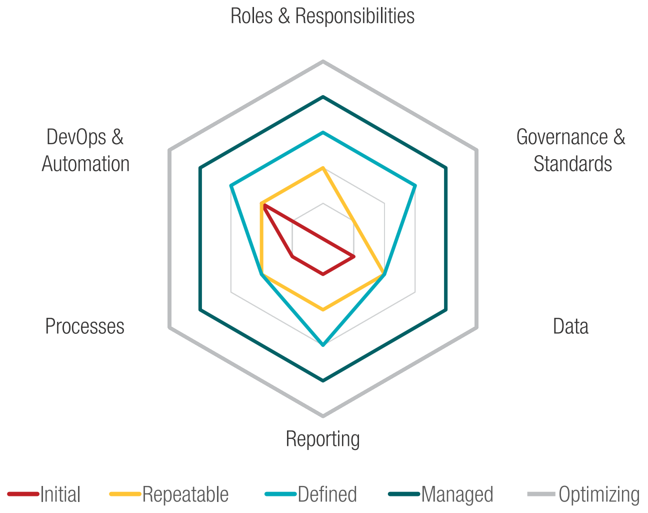
Capco has recently observed increased demand for technology currency strategy and maintenance engagements from our clients with varying levels of maturity in this space. The 2017 WannaCry ransomware attack, which is estimated to have cost $4 billion, targeted Windows XP.[1] Among others, the list of targets included Fortune 500 businesses like FedEx, Honda, and Hitachi as well as government bodies such as England's NHS. Despite the attack, unsupported technologies from all the major software companies continue to make up a large portion of the software estate being used by financial institutions today.

This article aims to provide a roadmap for financial institutions to develop a mature software currency capability that effectively eliminates these vulnerabilities in an ongoing fashion. Financial institutions can achieve this by following three key steps: creating a proactive mindset towards currency, developing a granular software inventory and relevant KPIs, as well as incorporating software currency management into performance reviews. Some future-looking capabilities that allow financial institutions to manage their software currency in a more automated fashion are equally explored.

THE CURRENCY MATURITY MODEL

Insufficient prioritization and funding have led to large portions of the technology stack becoming outdated, requiring large capital investment projects to resolve the issue. The below model provides insight into the five levels of maturity for software currency management.

Currency Maturity Model



[1] https://www.cbsnews.com/news/wannacry-ransomware-attacks-wannacry-virus-losses/

1. **Initial** – Software currency management is ad hoc and inconsistent; unpredictable and reactive.

2. **Repeatable** – Software currency is managed exclusively by the technology teams with no involvement from business. Lack of funding means that most technologies are outdated.

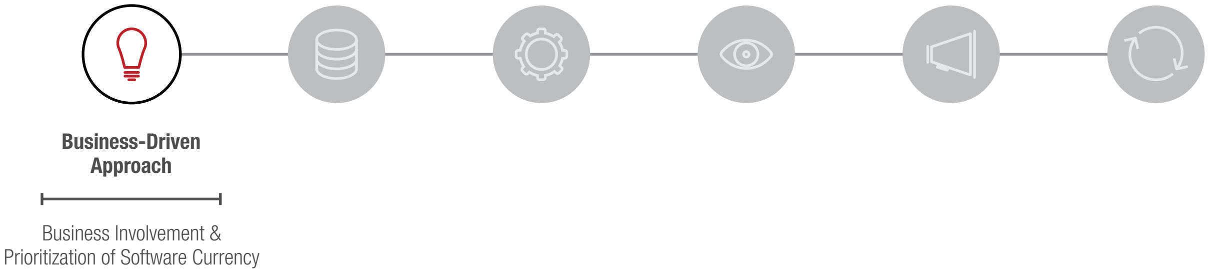
3. **Defined** – Organizational KPIs, roles, and standards have been defined; however, a lack of data and processes prevents the organization from understanding the breadth of the problem.

4. **Managed** – Organization has invested heavily in developing its currency management capabilities (data, reporting, DevOps, and processes) as well as clearly defined roles and responsibilities and standards. Software currency is well managed and tracked at the organizational level.

5. **Optimizing** – Organization has reached a state where a very small portion of the stack runs on outdated technologies. Where outdated technologies exist, there is a clear risk-based remediation plan to resolve it. Overall focus on continuous improvement, while organizational stability provides a platform for innovation and agility to adapt to new technological developments.

Given the importance of effectively managing software currency to reducing cyber risk and critical vulnerabilities, this POV will provide a strategic roadmap to help organizations evolve their Software Asset Management capability towards the "Optimizing" level.

OBSERVATION #1: CHANGING THE MINDSET TOWARDS SOFTWARE ASSET MANAGEMENT - PREVENTATIVE TO PROACTIVE



Many enterprises have delayed upgrades to the technology stack. These delays often exist due to a lack of prioritization for remediating software currency issues. Business owners usually prefer to invest in new features on existing applications or developing net-new applications rather than investing in currency. The enterprise needs to implement strategies to change the business mindset towards currency from reactive to proactive to resolve this. Business teams need to become the drivers of the mindset change by increasing their visibility into the currency portfolio, including understanding the operational resiliency of their platforms, as well as increased accountability for the business teams to manage the currency.

Recommended Approach – Implementing a Business-Driven Hub and Spoke Model

To drive this change, a 'Software Currency Spoke' should be identified within each business line, and a 'Software Currency Hub' should be created as a new team within InfoSec.

1.1 The Software Currency Spoke

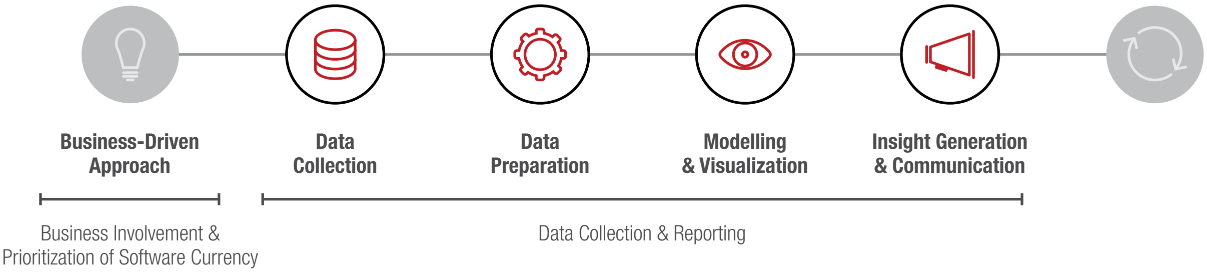
The Spokes should be a group of business leaders at the VP/ Executive Director Level or above who take on additional responsibility in addition to their existing roles. The Spokes are responsible for ensuring that the software currency portfolio is managed and reported effectively within their LOB. This includes supporting the technology teams in gaining sufficient funding from their respective business owners, providing oversight into the overall portfolio, and reviewing LOB-level reporting on their portfolio. Furthermore, this should help transfer some of the Software Currency accountability to the business, which should increase funding for the technology teams to remediate software currency. Accountability and funding will equally increase software currency's priority, which should lead to a better-managed portfolio.

1.2 The Software Currency Hub

The Software Currency Hub should be accountable for working with the Spokes to track enterprise reporting, acting as an arbitration board for currency decision-making, creating currency standards and defining the enterprise direction for critical technologies. The Hub should meet with its Spokes on a monthly cadence to track progress on the Software Currency Portfolio and help develop the constantly evolving currency standards. Clearly defined currency standards should encourage upgrading to technologies where the vendor manages the currency through the constant deployment of new versions such as Office 365. In the case where these are not available, further currency standards should be defined by the organization. Additionally, currency standards should encourage deploying containerized shared services that allow for individual application upgrades without considering the external impact and dependencies on other applications.

The Software Currency Hub will be equally responsible for holding ad-hoc arbitration meetings for risk acceptances. Risk acceptances will largely focus on allowing certain teams to forego their currency management if they are going to be consolidating or sunsetting a given group of applications to not waste funding. Furthermore, the Hub will need to hold ad-hoc sessions with enterprise architecture. These ad-hoc sessions will define the upgrade path when technologies reach the end of life, meaning that the vendor will no longer be releasing new versions and all existing versions are out of support. Currency standards, the enterprise technology direction, and funding will all need to be considered for these decisions.

OBSERVATION #2: RECOGNIZING THE IMPORTANCE OF A SOFTWARE CURRENCY INVENTORY – DATA COLLECTION, ANALYSIS, KPIS AND KRIS


Business-Driven Approach
Data Collection
Data Preparation
Modelling & Visualization
Insight Generation & Communication
Business Involvement & Prioritization of Software Currency
Data Collection & Reporting

Once business teams have increased their visibility and accountability over currency, it becomes a business-driven activity, organizations need to begin developing a software currency inventory. First, organizations are often unaware of their position as it relates to currency maintenance due to a lack of data on their applications' software currency. This, in turn, limits the organization's ability to generate effective insights, identify where support should be directed, and where the interdependencies exist both internally and across applications.

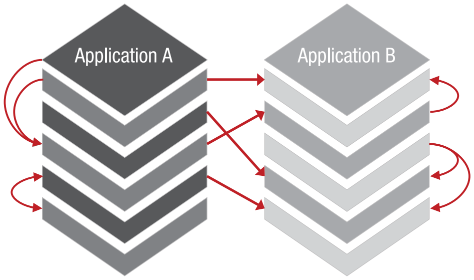
Recommended Approach – Developing a Software Currency Inventory

2.1 What is in the environment?

Technology teams and vendor management need to work together to identify all of the technologies that exist in the environment. Additionally, they should identify the full technology stack for every application and store this data in a consolidated enterprise CMDB. Further, the software currency data needs to be kept up to date through an established checkpoint in the technology delivery process requiring the delivery team to identify the full technology stack and update the CMDB with this data every time a new application or technology is deployed.

2.2 Defining Application Dependencies

Application Dependency Map



Second, the enterprise needs to define the internal and external relationships to their application to perform impact analyses. External relationships (straight arrows) identify what servers and technologies might be 'owned' by another application are relevant to another. This data gives organizations a line of sight into where vulnerabilities exist outside the application's immediate and obvious scope. Internal relationships (curved arrows) will define how different layers of the tech stack depend on others. This will help identify the difficulty associated with upgrading a certain technology as it may incur compatibility issues that require other layers to be upgraded. To keep the data accurate and complete, updating the CMDB with the relationships should be included as a checkpoint in the delivery process for all applications, infrastructure, and technologies to identify where these relationships exist.

2.3 Tracking technology lifecycles

A third key piece of data is gathering the lifecycle data of technologies currently deployed in the environment and those on the horizon in an automated fashion. The most crucial data points include end of life/support, release dates of upcoming versions, and technology certification/ assessment dates. Each of these data points will enable enterprises to identify where and when upgrades are needed. They equally increase the depth of the assessment process when selecting what upgrade path an application should take. These data points will help the Currency Decision Board make assessments on the scale of upgrading a given version or moving away from technology.

2.4 Creating Currency Scorecards

By leveraging this data, organizations should develop Software Currency Scorecards at the application, line of business, and enterprise levels. This will require developing currency KPIs to monitor how the technology lifecycle is being managed end-to-end. Additionally, currency KRIs should be developed to identify where the largest infosec vulnerabilities exist based on each application's current technology stack, and their associated resolution plans. Scorecards will shed light on where an otherwise opaque process is breaking down, how the organization can improve, and where funding can be best directed. The following is a non-exhaustive list of important metrics for organizations to measure:

1. Technology Lifecycle Status (in percent): unsupported, more than one year remaining of support, and less than one year of support

2. Planned upgrade dates in comparison to the technologies end of support dates: upgrade is after the end of support, the upgrade is within six months of the end of support, or upgrade is more than six months ahead of the end of support

3. KRIs (in percent): applications without resolution plans, applications which have missed their resolution date

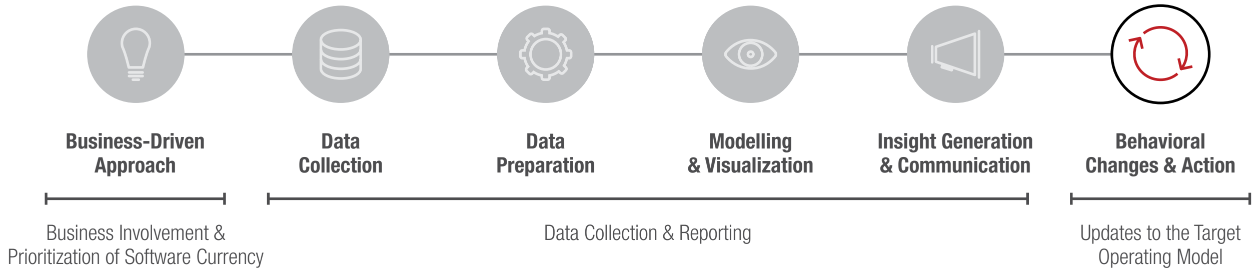
The Software Currency Scorecards should be used in conjunction with the application rankings in terms of business criticality and risk to inform priority and inform the business teams on software currency status. Scorecards at the application and line of business level should be used as the primary form of reporting that is shared from the application teams with the Software Currency Spoke. In contrast, the line of business and enterprise-level scorecards should be the main concern for the Software Currency Hub.

2.5 Updating Resiliency Assessments

Software Currency Scorecards and a more detailed CMDB should be used to update the resiliency assessments. There is a significant correlation between an organization's software currency and the application or enterprise resiliency to breaches. Organizations running more than 50% of their computers on outdated browsers double the chance of a breach, while organizations running more than 50% of their computers on outdated operating systems were three times as likely to experience a breach than those with less than half on outdated OS.[2] With currency included in the resiliency framework, organizations will be better positioned to identify address vulnerabilities. Additionally, including software currency in the resiliency framework will increase its visibility across the organization and further encourage it to drive its remediation.

[2] BitSight Technologies. "A Growing Risk Ignored: Critical Updates - Exploring the Prevalence of Outdated Systems and their Link to Data Breaches." 2017. Online.

OBSERVATION #3: ORGANIZATIONAL SHIFT – REWARDS AND RECOGNITION


Business-Driven Approach
Data Collection
Data Preparation
Modelling & Visualization
Insight Generation & Communication
Behavioral Changes & Action
Business Involvement & Prioritization of Software Currency
Data Collection & Reporting
Updates to the Target Operating Model

Organizations with a higher level of maturity in the technology currency maintenance process often struggle to make the insights they generate lead to fundamental changes in behavior and action. This is often a key point where organizations struggle to make their investments worthwhile. Without the appropriate updates to the organizational operating model, accountability will remain unclear, allowing the currency issues to persist. With the Software Currency Hub and Spoke Model established and a detailed Software Currency inventory, organizations are on their way to an effectively managed currency portfolio.

Recommended Approach – Including Software Currency in Performance Goals

Organizations need to include software currency management in the performance review goals and objectives for business, technology, and InfoSec employees involved in managing the enterprise software currency. The following roles should all have currency as part of their performance goals:

- Application Owners (Business and Technology)
- Software Currency Spokes
- Software Currency Hub members

Including software currency in the review of individuals helps create accountability and personal incentive to manage currency across the organization effectively. This is another opportunity to leverage the Software Currency Scorecards as they make for a measurable and consistent means for assessing successful management of software currency.

LOOKING TOWARDS THE FUTURE

Leveraging this guide will help organizations reach the ‘managed’ phase of the above software currency maturity model. To continue developing towards the Optimizing phase, organizations will need to embrace innovation through in-house or third-party products.

4.1 Liquid Software

Liquid software is a relatively new concept that is based on reaching a point where new software and new versions of existing software are being continuously released. This achieved through two improvements to the DevOps process :

1. **Short release cycles:** Automation in the release management process has reduced the effort required to deploy new software, leading to more software being released more quickly.

2. **Distributed Software:** With software being increasingly deployed through containerized micro-services, there is no longer a shared ‘version’ that is used within an enterprise but rather a high volume of nodes, each using their own version of the software.

We can see a world where release cycles become so short that they tend to zero, and micro-services so small that software in effect becomes ‘liquid.’ Software would be constantly updated through automated pipelines. Although this is a utopian state where software is truly being released constantly, the real goal is to move towards products where the vendor manages the currency through automated version releases that can be easily and rapidly deployed to the whole host of micro-services in the enterprise. This presents a variety of steps that will need to be made to reach this level of maturity:

1. Accounting for how the software’s currency will be managed when teams are planning upgrades

2. Enabling DevOps processes to deploy software rapidly and automatically

3. Leveraging containerized micro-services rather than monolithic shared services, and

4. Developing sufficient security and automated review processes such that software can move from the pipeline the vendor has to distribute the upgrade, into financial institutions’ environments, and eventually into production

4.2 Scenario Analysis Tool

A Scenario Analysis Tool would be useful as an add-on product to the enhanced software currency inventory, Scorecards, and resiliency assessments. Users would be presented with an interface that would predict how their currency scorecards and resiliency scores would be impacted if they chose to delay one versus another layer of the technology stack. In its full form, this tool could leverage Artificial Intelligence to run enterprise-wide analyses and, in turn, return a prioritization of all technologies and applications to show which would deliver the greatest benefit to the Scorecards and resiliency of the enterprise.

4.3 Compatibility Assessment Tool

Many enterprises struggle to identify how upgrading one software component impacts the rest of that application’s stack. For example, upgrading an OS may require a full uplift of the entire stack due to compatibility concerns. This presents a significant opportunity for financial institutions to work with vendors to develop a compatibility database that lists all deployment combinations.

If used in concert with the Scenario Analysis Tool, these two could present upgrade options based on various parameters that the enterprise could identify, such as maximum support length, fewest upgrades, etc. By leveraging historical vendor contract data and project data, machine learning could be included to create estimates around how much each upgrade option would cost. This approach would create a further level of prioritization such that an increasingly accurate ROI (scorecard impact, cost, resiliency impact) could be driven out of the scenarios.

CONCLUSION

This article provides a roadmap for financial institutions to mature their software currency management capability. Software Currency is deservedly being given more attention at financial institutions as cyber security risks continue to be top of mind for executives. Reducing cyber risk not only protects an organizations' share price but equally its customers' most important data. By following this roadmap, FIs can implement a framework that provides the foundation for reducing the number of vulnerabilities that exist across environments and improve their organizational resiliency.

AUTHOR

Nathan Lautens, Associate
Nathan.Lautens@capco.com

CONTRIBUTOR

Tissany Jung, Principal Consultant
Tissany.Jung@capco.com

ABOUT CAPCO

Capco is a global technology and management consultancy dedicated to the financial services industry. Our professionals combine innovative thinking with unrivalled industry knowledge to offer our clients consulting expertise, complex technology and package integration, transformation delivery, and managed services, to move their organizations forward.

Through our collaborative and efficient approach, we help our clients successfully innovate, increase revenue, manage risk and regulatory change, reduce costs, and enhance controls. We specialize primarily in banking, capital markets, wealth and asset management and insurance. We also have an energy consulting practice in the US. We serve our clients from offices in leading financial centers across the Americas, Europe, and Asia Pacific.

To learn more, visit our web site at **www.capco.com**, or follow us on Twitter, Facebook, YouTube, LinkedIn and Instagram.

WORLDWIDE OFFICES

APAC
Bangalore
Bangkok
Gurgaon
Hong Kong
Kuala Lumpur
Mumbai
Pune
Singapore

EUROPE
Berlin
Bratislava
Brussels
Dusseldorf
Edinburgh
Frankfurt
Geneva
London
Munich
Paris
Vienna
Warsaw
Zurich

NORTH AMERICA
Charlotte
Chicago
Dallas
Houston
New York
Orlando
Toronto
Tysons Corner
Washington, DC

SOUTH AMERICA
São Paulo

WWW.CAPCO.COM

CAPCO