TU WIEN Informatics

Impact of automation in Software Asset Management Systems

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

David Rodriguez-Pastrana Parareda, BSc
Matrikelnummer 01227980

an der Fakultät für Informatik
der Technischen Universität Wien
Betreuung: Univ.Prof. Dr. Wolfgang Kastner

Wien, 2. Oktober 2023

David Rodriguez-Pastrana Parareda	Wolfgang Kastner

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

TU WIEN Informatics

Impact of automation in Software Asset Management Systems

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Business Informatics

by

David Rodriguez-Pastrana Parareda, BSc
Registration Number 01227980

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof. Dr. Wolfgang Kastner

Vienna, October 2, 2023

David Rodriguez-Pastrana Parareda

Wolfgang Kastner

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

David Rodriguez-Pastrana Parareda, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 2. Oktober 2023

David Rodriguez-Pastrana Parareda

Contents

∞

Danksagung

Ich bin meinem Betreuer zutiefst dankbar für seine kontinuierliche Unterstützung bei diesem aktuellen Thema der Automatisierung im Software Asset Management und den bestehenden Prozessen. Ein besonderer Dank gilt meinen Teamkollegen bei Crayon, Tilo Probst und Markward Franz, für ihre wertvollen Erkenntnisse und meiner Freundin Charlotte Somborn, die mich auf dieser gesamten Reise unterstützt hat.

Ich bin dem Unternehmen Crayon, einem Marktführer im IT Asset Management[1], für ihre Unterstützung, Schulungen und die mir gebotenen Möglichkeiten auf diesem Gebiet des IT Asset Managements dankbar. Ihre Einblicke, insbesondere von ihren globalen Teams in Oslo, Wien und München, waren außerordentlich hilfreich. Als Teil des Crayon SAM-Tool-Teams konnte ich mein Wissen im Bereich Softwarelizenzierung erweitern.

Zu den wichtigsten SAM-Partnern von Crayon gehören: Snow, Flexera und ServiceNow.

Crayon war von entscheidender Bedeutung, bot Einblicke in jedes SAM-Tool und half, meine Forschungsziele zu verfeinern.

[1] *Gartner 2023 - Crayon als Marktführer für SAM benannt*

Acknowledgements

I'm deeply thankful to my supervisor for his continuous support of this on-demand topic of automation in software asset management and existing processes. Special thanks to my team colleagues at Crayon, Tilo Probst, and Markward Franz for their valuable insights and to my girlfriend Charlotte Somborn who supported me during the whole journey.

I am grateful to the company Crayon, a leader in IT Asset Management [2], for their assistance, training's, and opportunities provided in this field of IT asset management. Their insights, especially from their global teams in Oslo, Vienna, and Munich, have been invaluable. Being part of Crayon's SAM tool team allowed me to learn and expand my knowledge in software licensing.

Crayon's key SAM partners include: Snow, Flexera, and ServiceNow.

Crayon has been essential, offering insights about each SAM tool and helping to refine my research goals.

[2] *Gartner 2023 - Crayon named Leader for SAM*

Kurzfassung

Effektives Software Asset Management (SAM) hat sich in der heutigen Unternehmenslandschaft als eine kritische Herausforderung herauskristallisiert. Da Softwareverträge zunehmend komplexer werden und Anbieter spezialisierte Dienstleistungen anbieten, hat sich die Aufgabe, präzise IT-Ausgabenberichte, Budgets und Prognosen zu erstellen, intensiviert. Diese Arbeit untersucht den Übergang von einem geschäftszentrierten Ansatz zu einem service-technologiezentrierten und hebt die Vorteile der Automatisierung und proaktiven Berichterstattung, zur Minderung von Überprüfungsrisiken, hervor. Insbesondere prognostiziert Gartner, dass bis 2025 etwa 70% der Unternehmen die Infrastrukturautomatisierung umsetzen werden.

Das korrekte Implementieren von SAM kann versteckte Kosten aufdecken und die Produktivität in der Organisation steigern. Gartner weist darauf hin, dass optimierte SAM-Praktiken zu einer erheblichen Reduzierung der Softwarekosten führen könnten, um bis zu 30%. Darüberhinaus deuten Prognosen auf einen Anstieg von Organisationen hin, die fortlaufende SAM-verwaltete Dienstleistungen übernehmen, mit einer erwarteten Steigerung von 20% im Jahr 2023 auf 40% bis 2026, als Reaktion auf die steigenden Kosten und Komplexitäten.

Schlüsselwörter: Software asset management, IT asset management, Self-service automation, Business process management, Service-oriented architecture, Software procurement, Artificial intelligence, OpenAI, ServiceNow, Crayon, SAM tools, ITAM program, FinOps.

Abstract

Effective Software Asset Management (SAM) has emerged as a critical challenge in the contemporary corporate landscape. As software contracts become increasingly complex and vendors offer specialized services, the task of producing precise IT expenditure reports, budgets, and forecasts has intensified. This research explores the shift from a business-centric to a service-technology-oriented approach, emphasizing the advantages of automation and proactive reporting to mitigate audit risks. Notably, Gartner predicts that by 2025, approximately 70% of organizations will implement infrastructure automation.

Implementing SAM correctly has the potential to unveil hidden costs and enhance organizational productivity. Gartner suggests that optimized SAM practices could lead to a significant reduction of software expenses, up to 30%. Moreover, forecasts indicate a rise in organizations adopting ongoing SAM-managed services, with an expected increase from 20% in 2023 to 40% by 2026, in response to the escalating costs and complexities involved.

Keywords: Software asset management, IT asset management, Self-service automation, Business process management, Service-oriented architecture, Software procurement, Artificial intelligence, OpenAI, ServiceNow, Crayon, SAM tools, ITAM program, FinOps.

CHAPTER 1

Introduction

"Based on our experience within Asset Management, we have the opportunity to create a complete overview of the customer's license entitlement and actual usage. The result is better control, better economy, and better basis for decision making."

Rune Syversen, back in 2002 upon founding Crayon.

In today's business environment, managing software assets has become a significant challenge for firms. As software contracts increase in complexity and software suppliers specialize more in their services, it has become very difficult to create the desired reports, budgets, and forecasts on IT expenditure, as noted in the research paper [1]. Numerous studies highlight the importance of evolving from a technology-oriented to a service-oriented approach [19, 25, 30]. The majority of studies recommend including more automated activities and proactive reporting to avoid the risks associated with vendor audits [7, 27]. Failing to properly manage software IT assets and their lifecycle results in hidden costs and reduced productivity for the entire organization [6, 8]. In this case, a proper Software Asset Management is key to mitigate risks and minimize financial liabilities by proactively monitoring unused, underutilized, or outdated licenses.

According to ***Gartner*** *"many organizations can cut spending on software by as much as 30%"* by utilizing best practices to optimize software licenses by using SAM tools [1].

Gartner, on September 5, 2023, states the following: *"By 2026, the growth in spend and complexity will drive 40% of organizations to utilize continuous software asset management (SAM) managed services from expert third parties, up from 20% in 2023."* [2]

[1] *Gartner - Cut Software Costs by 30% Using SAM tools*
[2] *Gartner - Software Asset Managed Services in 2023*

Managing software assets without using the correct tool creates a high cost for **IT**, and businesses acknowledge the importance of this investment. Usually, a firm will start by optimizing software license costs in its first program stages and later include optimization practices for hardware governance and procurement strategies. Having the right IT program allows firms to fulfill the objectives specified by the executive team, including the Chief Information Officer (**CIO**), as IT costs expand and require a successful **ITAM** strategy as presented in the success handbook [30]. This applies to managing the complete lifecycle of purchasing software and hardware, from procurement to disposal, where service-oriented methods can ease and simplify activities, support license cost management, reduce complexity in contract renewals, and eliminate tasks that become repetitive to the operations unit team. Several studies highlight the effectiveness of combining business-driven processes (a top-down approach) with IT-driven, bottom-up processes through Representational State Transfers (**REST**)ful *Web Service* [32, 4, 24]. In this context, the integration of Business Process Management (**BPM**) and Service-Oriented Architecture (**SOA**) improves the flexibility and effectiveness of the business operations. Together, these two frameworks transform enterprise functions from mere "automation" with "managed flexibility", allowing applications to be invoked as needed for delivering optimal IT services. When aligning **SOA** within SAM systems, different workflow actions may require automation to enrich tasks such as license imports, refreshes, and transfers, contract renewals, product allocation, identification of hardware disposals, or even suggesting license types with ***AI*** as demonstrated in this thesis. Leading tools like ***Flexera***, ***Snow***, ***USU***, and ***ServiceNow*** can support efforts to reduce operational and maintenance IT costs, as highlighted in a recent study [33].

Moreover, as licenses become more difficult to manage, the major *"Big 4" **Tier-1*** software vendors - *Microsoft, Oracle, IBM, and SAP* [3], have significantly raised their prices and specialized their products based on functionality and support [4]. For instance, cloud applications licensing depends on deployment factors and allocated resources, such as Software as a Service (**SaaS**) for subscribed software, Infrastructure as a Service (**IaaS**) for virtualized computing, Platform as a Service (**PaaS**) for platform provisioning, and Business Process as a Service (**BPaaS**) for automating business processes. ***Gartner*** forecasts a 21,7% growth in cloud and SaaS spending by 2023 [5]. This holds the importance of monitoring and ***Rightsizing*** cloud services to align with actual usage. Collaboration between **ITAM** and **FinOps** can optimize software and SaaS use, potentially saving millions on cloud spending when proactively reporting.

Gartner predicts that infrastructure automation will be adopted by 70% in the next three years [6]. *Flexera's* research, based on 500 IT executives, reveals 74% increased focus in **SaaS**, and 72% in public cloud computing, and leading software audits coming from *Microsoft, VMware, Adobe, Oracle, SAP, and IBM* [7].

[3] *SAM Managed Services - Tier 1/2 vendors*
[4] *ITAM Report 2023 - CIO Uncertainty for 2023/24*
[5] *Gartner: 21.7% Growth in Cloud/SaaS, 2023*
[6] *Gartner: 70% to Adopt Infrastructure Automation by 2025*
[7] *Flexera 2023: ITAM-FinOps Intersection Saves Millions*

1.1 Problem Statement

One of the biggest challenges for **IT** executives is to estimate and come across department costs, there is still a significant amount of waste spent across the IT landscape. According to a survey conducted in 2020, 43% of IT executives continue to track IT assets on spreadsheets [22], failing to use the appropriate tools for software tracking which leads to unsuccessful outcomes.

As technology infrastructure expands, there's a higher urgency for digital transformation, where well-performing enterprises tend to move soon into an **ITAM** program to address their emerging challenges in IT spending. The 2022 ITAM Report conducted by ***Flexera*** [8] offers insights into the core priorities of **ITAM** within enterprise-level organizations. The report also provides a broad overview of IT investments relative to public cloud, hybrid cloud, and SaaS technologies.

Being part of the program can help organizations save millions of dollars on information technology (**IT**) and operations. Usually, license costs stand unnoticed because there isn't a clear understanding of where the expenses originate, or because there's no tool in place for properly tracking the licenses, applications, and IT services. It's common to observe unforeseen costs originating from different unit areas, such as the use of licenses very highly priced and without approval, low usage, lack of formality via service request or approval process, improper tracking of licenses for possible disposals, or building own support tools that are overlooked by the financial team when estimating forecast budgets.

Some of the key zones where to track hidden IT costs are:

1. Software
2. Shadow IT
3. Hardware
4. Downtime
5. Inefficiency

Below, potential hidden and indirect costs associated with technology are identified, which organizations should recognize.

Software

The software significantly impacts an organization's total cost of ownership (*TCO*) expenditure. Owning software involves more than only licensing compliance, requires tracking an active software subscription or renewal process. During the software procurement process, organizations encounter costs such as licensing fees, usage model fees, contract negotiation expenses (including legal fees), and operational costs to start functioning. After licensing the assets, other hidden costs can come into play, which include:

- Software upgrades and maintenance
- ***Shelfware*** (unused software)
- Software support
- IT training (for the newly licensed software)

Additionally, enterprises frequently have to monitor over-licensing (or under-licensing) problems to witness any audit risks and to avoid paying for unused software - ***Gartner*** states that expenses can be cut by an average of 30% [10]. Based on the size and nature of the software *"gap"*, an organization can reduce its costs by half through the implementation of a specific process. This is likely achievable when evaluating annual maintenance costs or indirect costs with the right tools and accounting methods such as the Activity-Based Costing (**ABC**) approach - a method that assigns indirect costs to products and services by identifying and measuring the activities that consume resources.

Shadow IT

The term ***Shadow IT*** refers to the phenomenon where individuals or departments independently use information technology systems, devices, software, applications, and services without formal approval from the IT department. This practice has grown significantly, driven by the increasing adoption of cloud-based applications services [37].

While shadow IT can enhance employee productivity and drive innovation, it also brings substantial security risks to organizations, including potential data breaches and compliance violations. It often arises from employee's need for more efficient tools or applications to fulfill their job requirements. In a 2012 **RSA** study [8], 35% of employees reported working around their company's security policies to accomplish their tasks. For instance, an employee might discover a more effective file-sharing application than the officially sanctioned one and share it with colleagues.

Managers should be able to transform ***Shadow IT*** into a governed, business-located IT, enabling solutions with an informal character and a certain scope to address the opportunities and risks of specific applications. Additionally, a mature Business IT Alignment (**BITA**) analysis can also support the reduction of shadow IT [40].

This suggests identifying software spending on applications owned by the organization and understanding which ones are actually used or needed to support and invest. This process is known as ***Software Rationalization*** - a process that determines and scores the usage of applications by gathering metrics such as cost, usage, risk, and quality.

Furthermore, shadow IT extends to personal devices, such as smartphones and laptops, as part of the Bring Your Own Device (**BYOD**) trend. While this tendency offers flexibility

[8] *2012 RSA study*

and convenience for employees, it also presents challenges for organizations in terms of security and data management on personal devices.

According to ***Gartner***, ***Shadow IT*** can account for a significant portion of IT spending within enterprise organizations, estimated to range between 30% and 40%. ***Everest Group*** has suggested a higher figure for 2023, around 50% [9].

Hardware

The impact of hardware expenditures in organizations is often not well estimated, where multiple organizations focus only on the initial purchase price. However, ***Gartner*** highlights what are the best practices to measure and improve hardware spending by using the total cost of ownership (*TCO*) and identify areas of optimization [10]. The real expenses go beyond the initial purchase price, the *TCO* takes into account both direct costs, including hardware, software, operations, and administration, as well as indirect costs, such as end-user operations and downtime. Apparently, *TCO* is frequently overlooked and left out of budget considerations.

According to ***Gartner***'s research, the initial purchase price of a computer represents less than 20% of its *TCO*. The bulk of the *TCO*, approximately 80%, is attributed to indirect expenses such as technical support, maintenance, and labor costs. These expenses are the primary contributors to the *TCO*, highlighting the importance for organizations to concentrate on optimizing hardware expenditures by adopting best practices.

Computers are subject to constant configuration and maintenance, incurring ongoing costs from security measures, software updates, repairs, and general support. Enabling **IT** infrastructure and managing processes, however, can significantly reduce *TCO* while reducing timely operations for productivity.

[9] *Shadow IT Statistics for 2023*
[10] *Gartner TCO to Optimize Costs*

Downtime

IT outages are a common occurrence in the business world, resulting from various factors such as planned maintenance, software bugs, ***ISP*** issues, hardware failures, or even cyber-attacks. These disruptions can lead to significant financial losses, decreased productivity, compliance violations, legal costs, and in some cases, even business collapse.

Gartner's research [9] has shown that **IT** downtime costs companies an average of 5.600€ per minute, which can escalate to between 140.000 - 540.000€ per hour, depending on the size and scale of the business.

The annual survey 2022 reveals that on-site power issues continue to be the primary cause of significant site outages, as confirmed in the *Annual Outages Analysis 2023* report [21]. Other causes may include cooling failures, software/IT errors, and network disruptions that occur less frequently but are still of concern.

Inefficiency

Inefficiencies arise when there's a lack of extended automation in workflow-capable systems. This often results in escalated operational costs for organizations due to non-standardized software data, information quality matters, or restricted system integrations. Automation can assist with repetitive tasks, as explained below.

(a) **Data quality:** Automating tasks depends on the availability of accurate and reliable data stored in various systems. However, the required licensing data is often dispersed across diverse systems and applications. This leads to problems in integrating and ensuring data quality, usually possible to mitigate in tools by standardizing the data with normalized value names, versions, and editions - which are matched within the software content library provided by the tool. Having poor-quality data or not manually normalizing exposes compliance risks to firms. Having an inaccurate license position brings the tool investment into a loss if it's not properly configured.

(b) **Lack of standardization:** One of the primary challenges is the lack of data standardization coming from the software industry, where data is not well formatted. Different software vendors will expose their own licensing standards, pricing forms, and renewal procedures on different matters. This variety makes it difficult to develop a unified system capable of managing various scenarios across different software vendors.

(c) **Limited integration with other systems:** Multiple organizations operate their software assets on multiple platforms which carries various complexities. As the systems are required to exchange data across each other, it is important that the diversified number of tools can communicate via *Web Service* or **API**. For instance, some applications important to integrate are: software procurement, discovery,

CMDB for asset inventory, financial **ERP** systems for software purchases like *SAP Ariba*, *Coupa Software*, and *Crayon Cloud-iQ*. Integrating these systems with a **SAM** tool can be challenging, especially when systems are not built in the cloud or do not equip an **API** for communicating between platforms. A potential automated process might be limited based on the absence of *Web Service* communication.

The complexities and challenges in IT Asset Management (**ITAM**) are rising fast, and there is a demand for automated scenarios to improve operating activities and productivity, lower hidden software costs, and maximize business value. One notable concern to firms is the financial impact caused by underutilized licenses based on a lack of monitoring and the spike arising from subscription services **SaaS**. This not only limits resources but also raises questions about how well the firm controls software, hardware, and IT services. Additionally, a lack of transparency about licensing regulations increases the risk of having financial penalties. The rise of ***Shadow IT*** is a result of uncoordinated IT processes due to the low flexibility and response to employee's demands.

Further, managing software licenses manually has raised attention to using alternative and trustworthy options through a verified vendor system. In those regards, **SAM** tools are available to satisfy the demands of reclaiming unused software, reporting on a legacy vendor, optimizing spending, and allocating resources to other valued assets. Relying on traditional methods or manual processes, specifically during audits, indicates poor outcomes on productivity and overall spending risks. It is also common to observe corporate silos with specific applications or technology which also enlarge the lack of control. The option of procuring software on a shopping portal allows firms to proactively report and forecast software demands, having a much clearer budget plan.

In summary, effective IT Asset Management can greatly benefit firms by monitoring cloud vendors and subscription services using legacy **SAM** tools. The key factor is the urgent need for a more efficient, automated, and integrated approach to avoid overlooking potential cost savings and risks and to ensure optimal resource allocation.

1.2 Outline

This thesis is divided into six chapters. It begins with an introduction, which outlines the current difficulties observed in Software Asset Management (**SAM**). Chapter 2, presents an in-depth analysis of the current challenges, control and optimization strategies, and the utilization of automation to address everyday tasks when using SAM systems.

Chapter 3 presents the methodology applied, based on the research questions, and evaluates the methodologies of using **BPM** with **SOA** standards. Chapter 4 describes the methods implemented, detailing the process diagram and the developed scripts in ***ServiceNow*** as *Proof of Concept* (***POC***) which can influence a service-oriented approach.

Continuing to Chapter 5, the final findings are taught after measuring the potential benefits of each automation using **KPI**s to estimate records saved during operations. Lastly, Chapter 6 provides a brief summary of the achieved results and discusses future considerations relevant to this topic.

In this study, the aim is to explore and discuss the benefits of using automation within Software Asset Management tools to mitigate risks in standard, repetitive, or error-prone tasks. The analysis is performed using the **DSR** methodology, which involves assessing and refining the model through organized and planned interviews with SAM tool consultants at ***Crayon***. The designed model is based on the **BPM** agile process and **SOA** for simplifying the most time-consuming activities. During development, the results are compared and evaluated by experts with over five years of experience in the topic of IT Asset Management, who have collaborated with major international and enterprise firms.

CHAPTER 2

State of the Art

"We have a good platform to take part in that consolidation, but we need to stay relevant by investing in the right technology."

Rune Syversen, Founder of Crayon

In this chapter, we explore a range of challenges that organizations face in managing their software assets, along with different Software Asset Management solutions offered to address the complexity. From ***ServiceNow***'s SAM Pro to ***USU*** LiMa, ***Snow*** License Manager, and ***Flexera*** FlexNet Manager Suite, we examine their features, benefits, and capabilities. These solutions offer robust tools to optimize license spending, ensure compliance, and streamline software asset management processes.

2.1 SAM and License Challenges

To understand the overall challenges in license management and the efforts to maintain license alignment, firms must consider common matters related to 2.1.1 data complexity, 2.1.2 vendor compliance, and 2.1.3 ensuring a license balance.

2.1.1 Data complexity

When it comes to managing software licenses, many organizations depend only on vendor tools such as Microsoft's System Center Configuration Manager (**SCCM**), IBM License Metric Tool (**ILMT**), and Oracle License Management Services (**LMS**) for their yearly software audits. Instead of investing in a dedicated SAM system, they use customized spreadsheets or unmaintained, self-developed tools for SAM assessments, often due to cost constraints. Since vendor tools have limited capabilities for tracking IT assets, organizations may incur additional fees when they require deeper evaluations by software vendor specialists. Ideally, a tool should be able to organize information and reduce dependence on external experts, especially when software products (sourced from multiple

data sources) require certain data normalization, calculating which users and devices are assigned, and applying optimization rules to align with the consumption of **SaaS** usage data.

A primary concern when analyzing licenses is the excess noise and mismatches from the sourced Enterprise Resource Planning (**ERP**) systems. Deviations in the spelling of incoming publisher names, hotfixes, versions, and patch names complicate the analysis, making it difficult to determine which products require the correct allocation of licenses. Without a dedicated tool, this process requires substantial manual effort, including counting licensed user or device allocations. This not only consumes significant time but also increases susceptibility to human errors.

To overcome these challenges, organizations need to invest in the appropriate SAM tool that offers visibility into their main software ***Tier-1*** vendors. Choosing the right tool is crucial for conducting the needed steps of normalization and software matching and determining an accurate license position. By addressing data challenges and configuring the appropriate tool, organizations can achieve better control over software licenses, optimize their spending, and enhance standard processes like procurement, approval, and software disposal.

2.1.2 Vendor compliance

During yearly audits, negotiating from a position of technical know-how is vital. Not understanding license entitlements can favor vendors in pricing. It's essential to optimize usage, understand all aspects of software, and avoid excess costs. Technical readiness can offer firms substantial advantages, especially before a ***True-Up*** audit.

It's also key to understand the vendor's license types, as highlighted in Table 2.1.

Licensing Types	
Licensing Type	**Description**
Capacity-Based Licensing	Software licensing are based on the power (**CPU**, **cores**, **sockets**) of the hardware and/or groups of hardware.
Client Access Licensing	Requires any users and/or devices connecting directly or indirectly to a server to be licensed. Often a declarative secondary metric. Examples: Windows Server + **CAL**s Licensing, Oracle Named User Plus Licensing
Concurrent Licensing	Referred to as "floating licenses", this model allows simultaneous users access to the software running on a server within a threshold. Often used for engineering and specialty applications.
Consumption Licensing	A software subscription where an advance fee is consumed for one or more services, drawing down on the prepaid fee. Examples include Digital Licensing based on the number of signatures in document signature software and a subscription fee consumed for various services on the vendor's platform.
Device Licensing	A licensing type (node-locked) where the software is licensed per device.
Device Subscriptions	A software subscription where the software is licensed per machine, and usage is calculated.
Indirect or Digital Access	Access to software or systems from humans or non-humans via **API**s, devices, bots, **IoT** sensors, etc.
User-Based Licensing	Software allocated and licensed to a specific named user.
User Subscriptions	Software subscriptions are allocated to a named user, with usage calculations.
Subcapacity Licensing	Software licensed for less than the full capacity of one or more servers, reducing licensing costs for virtualization technologies.

Table 2.1: Overview of Licensing Types

Note: This is not an exhaustive list of all software licensing types. Source: Gartner

2.1.3 Challenges in maintaining a license balance

In Software Asset Management, challenges with license balance occur when there is a discrepancy between owned software licenses and their actual usage, potentially resulting in financial inefficiencies or compliance risks.

The two figures below, Figures 2.1 and 2.2, illustrate the concept of being over-licensed by having unused software or ***Shelfware***.

Supply exceeding demand results in waste

Figure 2.1 illustrates common gaps and scenarios that firms usually face based on actual software usage and entitlement rights. During a ***True-Up*** audit, costs with the software vendor (software publisher) may lead to an excess of licenses than anticipated, being an extremely important aspect to be well prepared for defending what is really needed. This may require finding optimal scenarios to save costs and developing an **ITAM** strategy to mitigate increased prices.

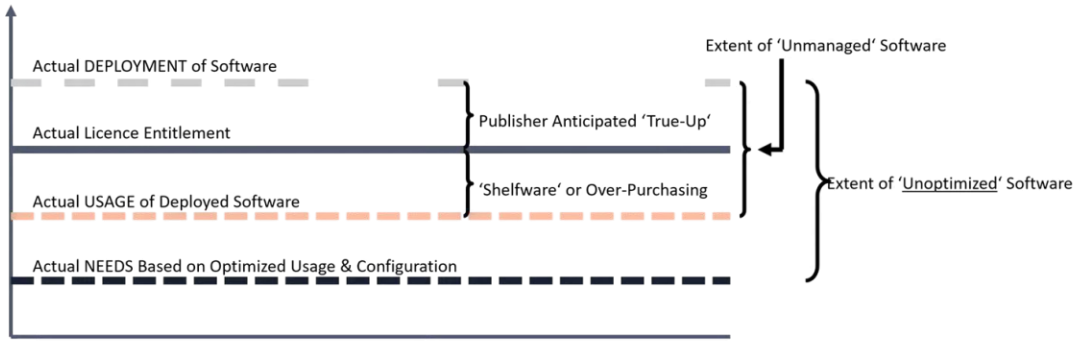

Figure 2.1: Entitled Licenses vs. Actual Needs - highlighted with a dashed line

Challenges to avoid compliance risks

It is crucial for a firm to continuously monitor licensing under the entitlements; otherwise, it can result in excessive penalty fees, as indicated in [8]. In this report, 15% of 500 IT executives reported spending more than 5 million euros during the last three years on software vendor audits. Maintaining a compliance position not only prevents risks but also reduces costs and enhances IT efficiency across the organization, enabling the allocation of resources to other critical areas of innovation.

Below, some of the key aspects of solving SAM risks are outlined, along with strategies on how to mitigate them in section 2.2.

- **Mismatched software purchases:** Buying **SaaS** subscription software that doesn't match user growth or company expansion can lead to unused licenses and compliance problems.
- **Lack of SAM technology:** Effective SAM requires tools for comprehensive assessments, including expert reviews beyond automation.
- **Licensing compliance issues:** SAM prevents legal actions or penalties by tracking licenses and expiration dates, especially for third-party or custom software.

- **Overpayment for software assets:** SAM identifies underutilized licenses, overlaps, and product types, and monitors usage and user access.

To overcome these challenges, it's important to understand the owned software assets and how they are being used. Developing a **SAM** governance strategy, allows firms to ensure cost-efficient spending.

Figure 2.2 illustrates the *"ideal"* scenario during an audit or ***True-Up***, marked in light blue as *"Installed; Licensed; Needed"*. This represents perfect license usage and optimization without risks or overpaying for licenses.

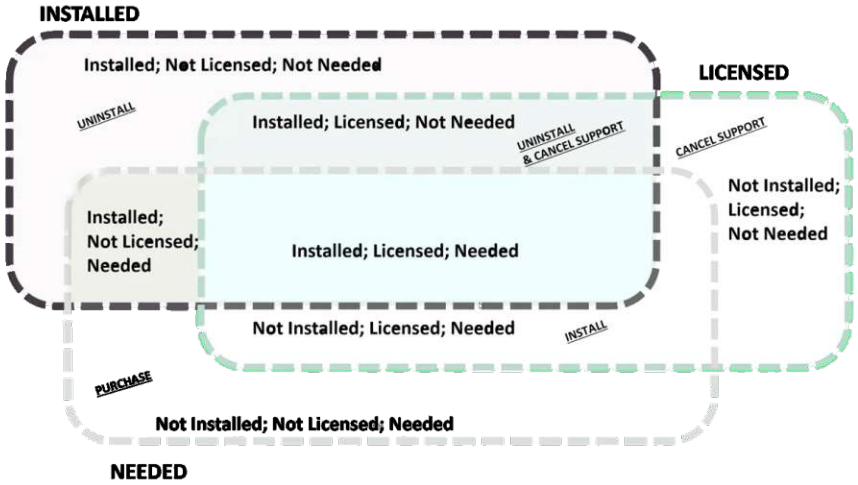

Figure 2.2: Ideal audit True-Up - highlighted in light blue

2.2 IT Cost Optimization and Risk Mitigation

Before diving into optimization practices, it's important to understand how **ITIL** defines Software Asset Management as *"all the infrastructure and processes necessary for the effective management, control, and protection of software assets within an organization throughout all stages of their lifecycle"* [1].

SAM is integrated into an organization's IT strategy to reduce costs, minimize legal risks associated with software ownership and usage, and improve the demand for IT and employee productivity.

The processes are guided based on the following **ISO** standards: ISO 19770-1 [2], ISO 19770-2 [3], and ISO 19770-3 [4].

[1] *An introduction to SAM*
[2] *ISO 19770-1 ITAM Requirements*
[3] *ISO 19770-2 Software identification tags*
[4] *ISO 19770-3 ITAM entitlement schema*

- **ISO 19770-1** provides processes for proper software asset management, aligning with governance standards and demonstrating its impact on IT Service Management.
- **ISO 19770-2** focuses on software identification tags, allowing the identification of the machines where software is installed and which application licenses are active.
- **ISO 19770-3** provides guidelines for defining a common schema in software entitlements between vendors, customers, and tool providers. This may include licensing terms, rights, limitations, and transport methods (**XML**, **JSON**, etc.).

The **SAM** framework provides a governance strategy with key objectives including compliance and cost optimization. It plays a crucial role in structuring workflows, optimizing data foundations and processes, and defining task allocations across various **IT** departments.

Within the framework, **ITIL** offers the following advantages:

1. **Lifecycle Management**: Provide structured and efficient administration of the entire software asset lifecycle through standardized and streamlined processes.
2. **Cost Optimization**: Achieve savings through license optimization and avoid duplicate purchases.
3. **Compliance and Risk Mitigation**: Ensure adherence to license agreements and identify security vulnerabilities.
4. **Efficient Resource Utilization**: Centralize the management of software licenses.
5. **Improved Negotiation Position**: Engage vendors for more favorable terms.
6. **Enhanced Security**: Facilitate regular updates of software products.
7. **Transparency and Control**: Offer complete oversight over software inventory.
8. **Streamlined Business Processes**: Ensure optimal provision of software tools and resources.

In Software Asset Management, automation is essential for managing core processes following **ITIL** best practices. However, it's important to initially have a clear understanding, which is often lacking, of the most time-consuming tasks in each phase with the highest productivity inefficiencies.

Below, the relevant **SAM** core processes for potential automation scenarios are explained.

1. **Procurement Processes**: Determine the software needs of the organization to ensure the right tools are acquired.
2. **Technical Inventory**: Assess how software is currently utilized across platforms, such as clients, servers, and the cloud.
3. **Licensing**: Manage contracts and licenses to ensure the organization can legally use the software.
4. **Compliance**: Ensure the software in use matches existing licenses.
5. **Optimization**: Manage software efficiently from acquisition to disposal.
6. **Additional Processes**: Depending on the organization's unique requirements, other processes may be necessary.

2.2.1 Optimizing Software License Rights

An important factor for optimizing software usage rights is understanding the different options available defined in the vendor's license agreement, which specifies the authorized usage scenarios. While different types of usage rights exist, below we detail the standard types applicable to major software vendors (such as *Microsoft and Adobe*), which include:

License Usage Rights	
Type	Description
Upgrade	Install the latest version at no extra cost. Requires effective management tools due to varying maintenance in licenses.
Downgrade	Purchase the latest version but install an earlier one. Useful for standard desktop images with obsolete versions.
Server Virtualization	Install on multiple virtual machines, counting as one license.
Secondary Use	Installation on both desktop and laptop counted as one license.
Disaster Recovery	Simultaneous installation on live and backup servers using one server's license.
Multiple Installations	Install the same application multiple times, often for different versions. Note: Older ITAM tools may incorrectly identify unclean uninstalls as multiple installations.

Table 2.2: Types of Software Product Use Rights

After detailing the **SAM** core processes, the licensing types as presented in Table 2.1, and the usage rights in Table 2.2, the subsequent pages will detail the key concepts. These concepts occur during the operations when using a Software Asset Management tool and the phases required for maintaining and optimizing spending.

2.2.2 Software Normalization / Application Recognition

The normalization process is the blueprint for standardizing and properly formatting software products using various recognition rules from the SAM tool content library, which is regularly updated. This process involves consolidating multiple software titles into standard fields, such as publisher name, product title, and version, or when available, using the direct identifier publisher part number (PPN). This ensures consistency and reduces the ambiguity of having duplicate entries when detecting application allocations to track user consumption.

In Figure 2.3, it is evident that without the application of normalization, data remains unidentifiable, appearing fragmented and disorganized. However, following the normalization process, as depicted in Figure 2.4, applications can be identified within the software content library database. The data becomes more cohesive and organized, enabling the tool to perform reconciliation and display compliance status.

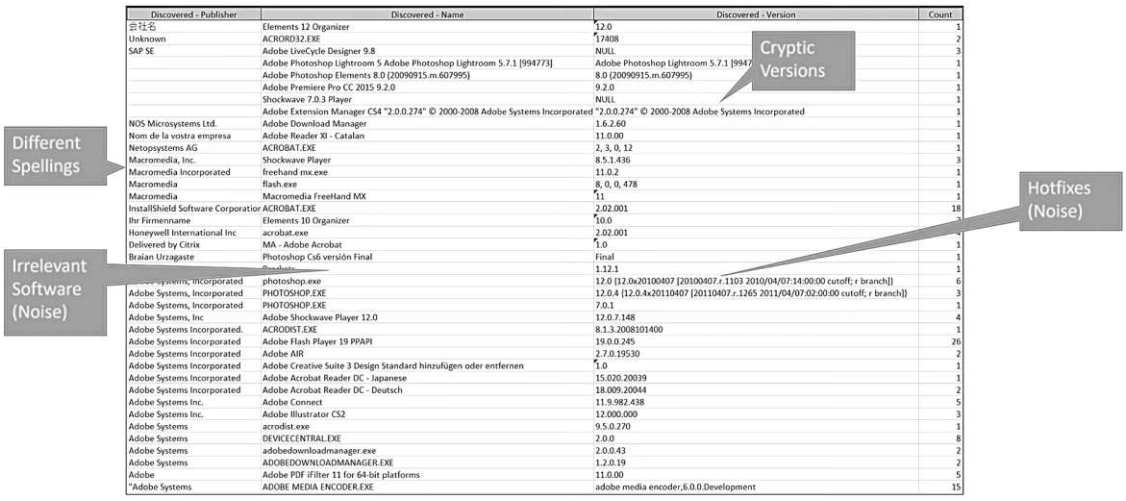

Figure 2.3: Products not normalized - Fragmented & disorganized

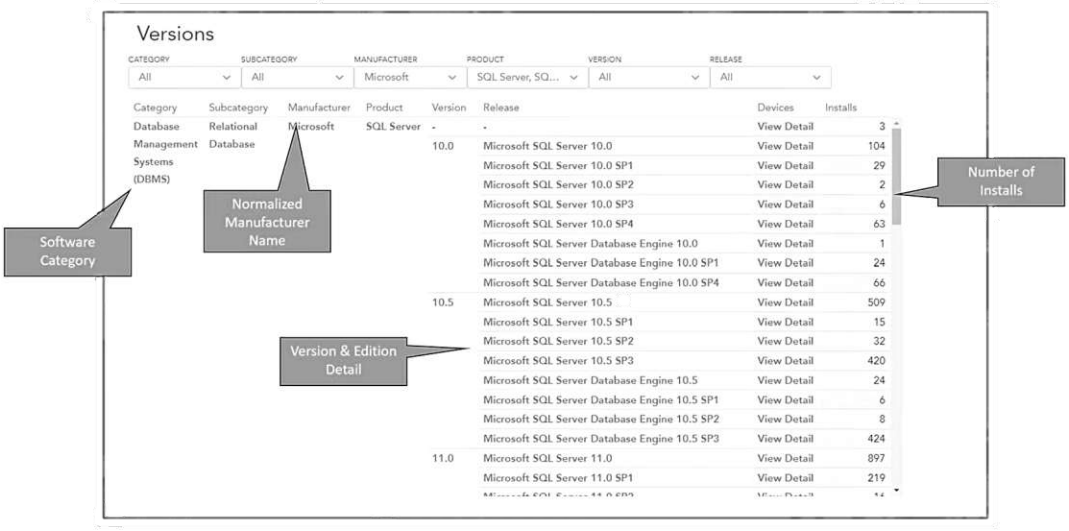

Figure 2.4: Products recognized after normalization - Cohesive & organized

2.2.3 Software Reconciliation

Reconciliation in SAM tools involves matching software installations and usage with the purchased licenses. It can be seen that the organization remains compliant with software licensing agreements and the software vendors. An effective reconciliation allows recognition of unused allocated resources and avoids penalties for overused licenses on yearly vendor audits. Gathering high-quality data from multiple sources is essential during this process. Defining optimization rules and automated processes can benefit much in the costs after are performed.

Following reconciliation, understanding the term **ELP** is important, as it's when an audit is done with the software vendor for a ***True-Up*** audit.

Results of an Effective License Position (ELP)

For many organizations, a SAM-managed service provides a series of detailed reports on the effective license position (**ELP**) of major software vendors deployed across their IT organization. These reports aim to optimize usage and avoid compliance risks by showing the current effective status of license consumption versus entitlement for software publishers like Microsoft, IBM, Oracle, SAP, and others. **ELP** reporting helps identify risks, such as compliance issues (using more software than being entitled) and overspending (having more entitlements than needed).

The results obtained in an **ELP** report involve reconciling the software licenses purchased with the software installations allocated to users or devices, enabling organizations to assess their compliance status. This helps determine whether the organization is under-licensed (at risk of a compliance audit, also known as under-purchased, as it has more software installs than purchased licenses) or over-licensed (wasting money on unnecessary

software purchases, also known as over-licensed or with unused licenses, often referred to as), negatively impacting the company and benefiting the vendor through increased profits.

Figure 2.5 presents an example in green, indicating a positive or over-purchased situation (small excesses beyond the purchase rights are considered healthy). In red, it illustrates an under-purchased scenario where the metric **PVU** or **CPU cores** exceed the purchase rights, requiring immediate attention to avoid audit penalties. Proactively reviewing the **ELP** report, helps organizations to mitigate risks, avoid compliance penalties, and identify optimization opportunities, reducing hidden or unnecessary software costs.

Manufacturer	Product	Version	Edition	Install	Users	PVU	Cores	Licenses	ELP
Microsoft	Project o365	2013	Pro	50	1210			1210	0
Oracle	Database	11g	Ent				100	80	-20
IBM	WSCS					8000		7800	-200
Unicom	RAD	8			12			15	+3

Figure 2.5: Example of Effective License Position (ELP) Report

2.2.4 Software Procurement

Procurement is a strategic process for acquiring the products and services essential for the operations of a business. It enables companies to source, organize, and purchase goods from various vendors for the best financial outcomes. It facilitates growth by improving processes and providing greater visibility into company spending to prevent cost inefficiencies and mitigate risks. This cross-functional process will involve multiple departments, including purchasing, finance, legal, and IT.

In a traditional procurement system, the accounting department handles each purchase-related task manually, including:

- Routing purchase requests
- Verifying approvals
- Entering tracking data
- Managing vendors
- Processing invoices
- Executing payments (often via paper checks)

Tracking these manual processes often relies on spreadsheets, which are not the most flexible tools, limiting the organization's visibility during negotiations for future supplies. Modern procurement systems will automate many of these manual tasks by:

- Centralizing vendor and invoice information
- Implementing *optical character recognition (OCR)* for paper invoice conversion
- Establishing automated workflows for approval and order management
- Digitizing purchase requisitions, order processing, and payment for goods

Additionally, the rise of **SaaS** applications has enabled automation via **API** connectors, simplifying the software license procurement process. However, this has led to some unforeseen consequences, including increased risk of ***Shadow IT*** as users encounter fewer restrictions, reduced formal approval processes, and heightened compliance risks.

SaaS has emerged as the fastest-growing cloud application category, with businesses widely adopting it since 2020. According to ***Gartner***, cloud spending is projected to increase by approximately 21.7% in 2023 [5].

The **SAM** team should oversee the procurement process by integrating the right sourcing tools (e.g., SAP Ariba, Coupa Software, Crayon Cloud IQ).

When developing a procurement solution, Heckman's research [16] includes the following steps:

1. Identify need
2. Put together a cross-functional team and identify roles and responsibilities
3. Continuously refine requirements and specifications in accordance with user needs
4. Gather information regarding alternative solutions
5. Perform cost-benefit analysis or other analytic technique to justify expenditure
6. Evaluate alternative solutions (including build/buy, in-house/outsource, etc.) and associated risks and benefits
7. Develop procurement plans that are integrated with project plans
8. Gain approval for the expenditure
9. Develop preliminary negotiation strategies

[5] *Gartner forecasts worldwide public cloud to grow 21.7%*

According to Heckman [16], some of the important questions to raise before planning the procurement process are:

1. What are the important components of an appropriate procurement plan? *Strategy*
2. How much planning (front-end loading) is appropriate or necessary for different types of acquisitions (e.g., commodity purchases vs. complex, unique acquisitions)? *Planning*
3. How should project teams be configured for different types of acquisitions (appropriate internal and/or external resources, project leader, etc.)? *Teams*
4. How should changes in scope, and changes in orders be handled? *Changes*
5. What are the important cost vs. budget considerations? *Costing*
6. What are the most effective methods of obtaining executive commitment? *Commitment*
7. Can requirements be separated from wants? *Requirements*
8. Should performance specifications and other outputs be captured for use in later phases such as quality management? *Specs*

Actions to consider for Software Portfolio and Vendor Management (**SPVM**) are [6]:

- **Cost Optimization:** Adopting a continuous, proactive approach to cost management.
- **Agile IT Procurement:** Prioritizing an agile framework for procurement, balancing speed, risk, and cost.

This implies that centralizing the procurement process using an automated solution focused on vendor selection, can result in better outcomes and discounts. The **SAM** licensing team's expertise, together with the financial team, can improve price negotiation and required standards, secure better licensing terms, and potentially achieve cost savings from software investments.

[6] *Sourcing, Procurement and Vendor Management Leaders*

2.2.5 License Management - Understanding the differences

SAM is the practice that covers software assets throughout their entire lifecycle, including the people, processes, and technologies used for reporting an effective license position. It aims to optimize the license from its purchase and software deployment to its monitoring, contract renewal, and disposal step.

In contrast, license management is the practice wherein multi-vendor experts focus on understanding the complexities of different models, metrics, terms, upgrades, and conditions. Software vendor specialists must be certified to understand the legal terms and conditions required. Their expertise allows them to identify mechanisms to optimize product costs by allocating resources to other purchased products. Their main goal is to support the business in cost-saving without breaching the agreement between the firm and the software supplier.

While **SAM** experts may also engage in software licensing aspects, their primary focus is on supporting the final reports and optimization practices. Their knowledge is more technical, and they are the ones who can define rules, processes, and configurations for an effective operation of the lifecycle.

Organizations often require support from both SAM tool experts and license specialists from vendors such as *Microsoft, IBM, Oracle, and SAP* to address both complexities together.

Evaluating Required Applications Using APM

Application Portfolio Management (**APM**) is a strategic approach used by enterprise architects to evaluate and manage the collection of applications and services in an organization.

APM views all applications as part of a collective portfolio, and utilizes specific metrics; considering factors like an application's age, use frequency, maintenance costs, and inter-dependencies - assessing value and overall health of the **IT** infrastructure.

Assessment helps to decide whether to maintain, update, retire, or replace particular applications. **ITAM** focuses on tasks such as software license allocation and hardware management. In contrast, APM evaluates whether certain software and hardware assets are essential for business operations.

APM's goal is to optimize the portfolio by controlling and replacing costly applications and identifying the underperforming ones - assists in minimizing the number of applications in use, and addresses issues like ***Shadow IT***.

In essence, APM can guide important decisions like choosing specific communication platforms for the entire organization.

2.3 SAM Governance

In the domain of **SAM**, the complexity of software licensing has brought specialization in multiple diverse tools. Organizations and professionals usually rely on the license metrics of certain vendors, particularly *"Tier-1"* vendors with complex, and costly licensing. This has led to dedicated **SAM** consultants who can support software licensing and optimization practices with knowledge of specific SAM tools.

2.3.1 ITAM Strategy

An effective **ITAM** strategy is required for *(1) optimizing the operational level of decision-making, (2) asset lifecycle management, (3) strategic asset management, (4) organizational unit aspects, and (5) asset information management* [12]. This will ensure optimal use and cost management across the entire **IT** infrastructure of the company.

When implementing an effective ITAM program, the organization is able to reduce budget costs on multiple departments, where enterprises have a big dependency on IT Service Management tools. Numerous readings explain how collaboration between **IT** and finance is fundamental for a successful ITAM program. Defining goals, having a clear objective, and communicating with the organization's bottom line can reach the desired results. It's important to recognize that ITAM is more than only solving an IT problem or goal - *it's a strategic approach for everyone to form part and effectively impact the whole business objectives.*

When discussing the significance of IT Service Management, it's essential to recognize that all employees and stakeholders play a role as organizational resources. A lapse in IT directly affects the bottom line, necessitating its financial implications to be measured with the appropriate tools. Many organizations have dedicated teams to oversee tasks within Software Asset Management, enhancing service quality, which in turn positively influences customer success. Effective IT services, driven by well-implemented company strategies, often lead to favorable customer impacts. Automation plays a crucial role in this ecosystem by cutting down time and ensuring timely support for both employees and customers.

IT Service Management and IT Asset Management have historically been separate disciplines, each with different business objectives and goals. However, their processes are highly interrelated. An existing **ITSM** process, supported by an **ITAM** program, enables risk mitigation, software cost optimization, ensured compliance, and improved **IT** operational costs as part of enhancing the IT architecture [1]. Additionally, it's crucial to pay increased attention to the risks associated with unattended software or ***Shadow IT*** to prevent audit penalties and ensure proactive reports on software spending.

On the other hand, as IT infrastructure expands rapidly, numerous papers highlight challenges in optimizing cloud usage [15, 3, 17]. Additionally, other papers address the emerging topic of tracking costs associated with Machine Learning usage [20]. Being well-prepared with the appropriate procedures enables a firm to refine its existing processes.

For example, there are modern techniques to track assets in real-time for Internet of Things (**IoT**) or smart devices [18, 35]. In sectors like healthcare, real-time monitoring is crucial for tracking the status of devices [31].

Having a robust ITAM strategy is essential for enterprises given the dominant software asset costs from third-party vendors, who often adjust their tactics for increased profits [28]. With rising IT costs, effective license management is necessary, especially given the magnified risks of maintaining compliance. Adopting SAM tools brings significant advantages to both large and mid-sized corporations [6, 27].

2.3.2 Steps for Implementing an ITAM Program

The importance of ITAM for both the bottom and top lines of an organization leads to the question: *how can finance get involved?* Several strategies can enhance the cooperation between finance and **IT**, starting with recognizing their mutual dependence [30].

1. **Scheduling recurring meetings:** Finance should always know what is going on with IT, and vice versa. Monthly or even quarterly update meetings will be mutually beneficial for both parties. Discuss new projects coming up, and new initiatives being taken, and use each other to help accelerate your mutual goals. This should also be an opportunity for the finance team to check in on projects that they have previously approved. Most executives approve a budget and leave it at that. Both parties should always be in the know regarding ongoing and even completed projects.
2. **Consulting the tool technicians:** IT is usually brought in at the last minute when additional software licenses or hardware are needed due to significant business changes. This is a significant mistake. When IT is given the chance to plan ahead, they have greater leverage and the potential to save money for the organization. Last-minute decisions will end up costing more in the short and long term. Bring in the IT department from the beginning, and make it a collaborative process.
3. **Making ITAM a priority for everyone:** As discussed earlier, ITAM can result in significant cost savings. The next step is to get buy-in across the organization. An ongoing ITAM program needs to be a priority for everyone to work effectively. This includes the C-Suite. Although alignment between finance and IT is crucial, to sustain the project, these two departments must work together to keep the entire organization on the same page.

By applying Kotter's principles [26], the following steps promote the transition to a successful **IT** governance culture and environment.

1. **Proactively design and manage the IT governance program:** Requires an executive sponsorship, a promoter, and a shared, measurable vision. Goals and strategies should be linked to the vision and performance evaluations.

2. **Mobilize commitment and provide incentives:** There should be a strong commitment from key managers and professionals. Creating a "Task Force" team to collaborate, develop, and coordinate plan executions.

3. **Make trade-offs and choices:** IT governance requires choices impacting resources, costs, and priorities. Clarify who approves choices and to whom issues are escalated. Determine when a task has reached its limit.

4. **Make change last, assign ownership:** Changes should be reinforced, supported, and promoted by accountable owners to ensure it continue throughout the organization.

5. **Monitor progress and common processes:** Developing consistent policies and technologies that enable progress and learning. Making IT governance an objective in employee evaluations and rewarding progress.

The following Figure 2.6 provides an overview of the **ITAM** process transition using the appropriate assessments, metrics, and methodologies.

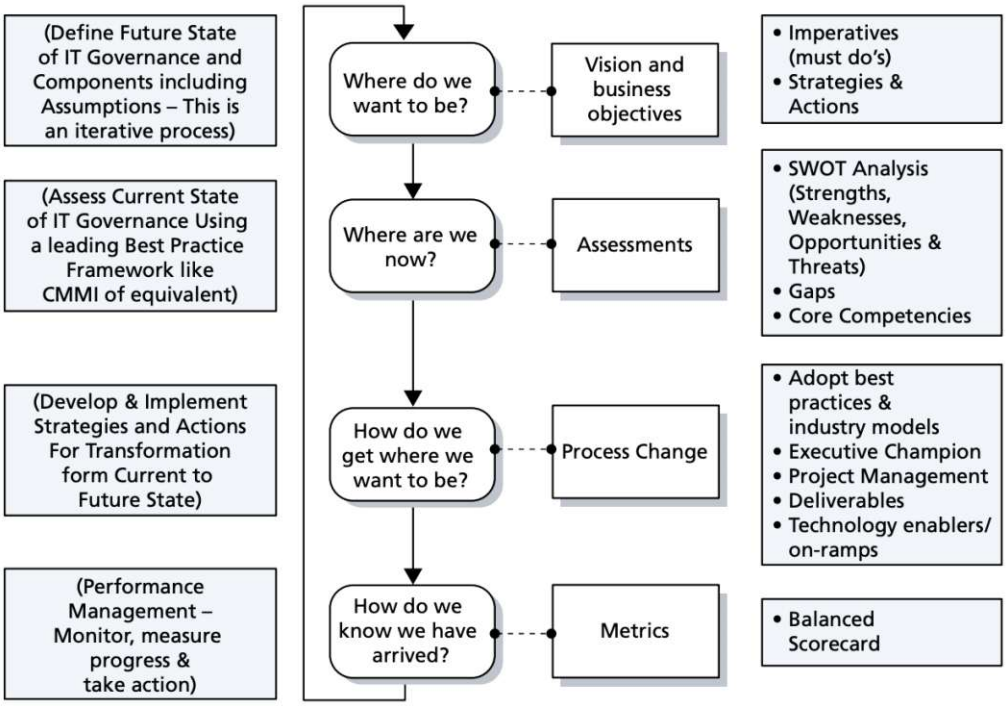

Figure 2.6: ITAM Governance Transformation - Current to Future State

Source: Implementing IT Governance [36]

2.3.3 SAM Lifecycle

When initiating an IT Asset Management program, understanding the phases of the software lifecycle is crucial. Two important terms to consider are *"entitlement granted"* (rights associated with a user or device) and *"license consumption"* (software deployed and in use). Installing software means creating an investment action that initiates a financial transaction between the software publishers and the firm.

Ideally, all licenses should be acknowledged and monitored to maintain the correct license balance and track their acquisition until disposal, as depicted in Figure 2.7. This process includes keeping a list of procurable software applications that meet user needs with minimal approval requirements. It's also essential to have policies for reclaiming software that is not used within a specified time frame.

To promote efficient software usage, it's important that firms underline that software belongs to the business and not to individual users, and prevent behaviors that lead to unnecessary software retention.

It's important to remark that technologies lacking feature releases may represent gaps in the IT Architecture, leading to potential risks [28].

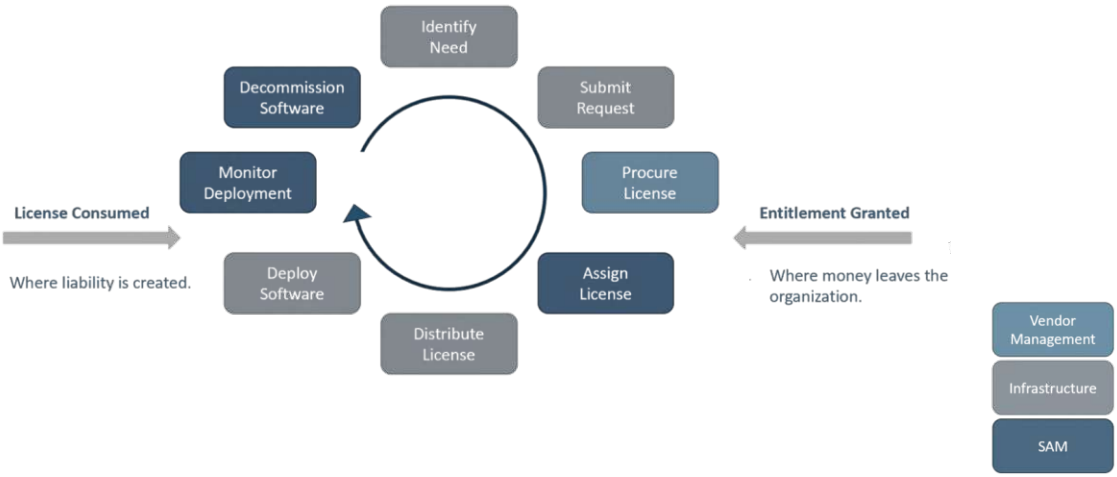

Figure 2.7: Important phases in Software Asset Lifecycle

Source: Anglepoint

2.4 The Advantages of SAM Tools

Due to the increasing complexity of software-licensing metrics, **IT** leaders are required to invest in **SAM** tools instead of using traditional practices which are usually done with *Excel* spreadsheets for calculating and tracking entitlements [27].

In contrast, tools constantly evolve to incorporate new features like **SaaS** connectors, *Web Service* integration, and future plugins. The decision of which tool to invest in is a common challenge, usually related to which platform performs better for network discovery and vendor features for achieving the greatest compliance results, with easy administration, maintenance, and great support are the most common demands.

For assisting with the tool selection, ***Gartner*** has introduced a *Tool Decision Framework* for Software Asset Management, which divides it into nine core activities:

1. Discovery and software purchases from various sources
2. Identify specific software entitlements from contracts and records
3. Organize entitlements into a structured inventory
4. Find both physical and virtual platforms running software
5. Recognize the software's use on these platforms
6. Obtain software usage data into an organized record
7. Compare and reconcile software asset data against entitlements
8. Adjust and refine software entitlements and consumption
9. Centralize and share software details, allowing efficient management during the lifecycle

Furthermore, extensive research and comparisons of **SAM** tools for automating services it was done for implementing a Service-Oriented Architecture approach.

Gartner's positions *ServiceNow, USU, Snow, and Flexera* as the leading candidates in their ranking from validated IT executive reviews [7].

Their primary goal is how well can discover software and devices and maturity for measuring metrics from ***Tier-1*** software publishers such as *Microsoft, Oracle, SAP, IBM, VMware, and Adobe.* Determining which tool works better for a firm depends on key factors - the need for IT Service Management system and workflow feature, or the inclusion of a **CMDB** system for managing inventory and address **ITIL** processes [23, 29].

[7] *Gartner - SAM tool comparisons*

Tools like *Snow and Flexera* may still continue being the most advanced for specific vendors, such as *IBM, Oracle, and SAP* as they have existed longer in the market than *ServiceNow and USU*.

Gartner calls ***ServiceNow*** as the best **ITSM** system [11], which can fulfill **ITOM** processes and includes a complete **CMDB** tool to manage inventory, promissing high expectations [2].

Based on the trending reviews gathered from ***G2*** platform reviews [34], SAM tools can be evaluated based on the below key factors.

- **Quality of Support:** 9/10 average user rating
- **Ease of Use:** 8.9/10 average user rating
- **Ease of Setup:** 8.4/10 average user rating

Users on the ***G2*** platform also conduct comparisons of SAM tools among the following most important features on daily usage.

- **SaaS Visibility:** 97% average rating
- **Change Logging:** 96% average rating
- **Logging:** 96% average rating
- **Administration Console:** 96% average rating
- **Alerting:** 95% average rating

These ratings can support when having to decide or make a *Proof of Concept* (***POC***) among different SAM tools.

According to the latest momentum reviews on ***G2*** platform trends [8], *ServiceNow and USU* are currently the leading tools, closely followed by *Snow Software and Flexera FlexNet Manager*.

2.5 Comparing Leading SAM Tools

The diversified SAM tools market has seen traditional SAM tool vendors take a platform approach, moving into adjacent offerings such as **IaaS** management, unified endpoint management, SaaS management, workflow/request management, IT service management, vulnerability management and hardware asset management [9]. Organizations heavily rely

[8] G2 - SAM Tool Trending
[9] *Gartner - Market Guide for SAM Tools*

on software, but the governance and strategic prioritization of software asset management are in many cases undervalued. The capabilities of SAM tools are set to expand, especially when can support cloud, virtual, and mobile software. The rapid changes in software licensing and platform diversity introduce investment risks. IT tool vendors lacking strong SAM capabilities are now collaborating with SAM tool vendors to enhance their offerings. Due to these trends, ***Gartner*** predicts more partnerships and acquisitions in this sector. It's important for IT leaders to select tools that align with both present and future needs, keeping in mind these market shifts. Before purchasing a SAM tool, it's common to conduct a *Proof of Concept* (***POC***) test to ensure its efficacy, especially in measuring software consumption.

In the following Table 2.3, according to qualified reviewers validated by ***Gartner*** [10], the most popular tools based on reviews written for each of the highest-rated SAM tools in the Gartner Magic Quadrant [11]. Snow is in first place with 305 reviews, followed by Flexera (294 reviews), USU (124 reviews), and ServiceNow (95 reviews). Some important factors to consider are: all tools have the same rating for optimizing software licenses, ServiceNow is the winner in support and satisfaction (noted with the label 'success'), and Snow and Flexera FlexNet are the most mature in Software Discovery and with **ELP** reporting. USU leads in its pricing model, keeping its other numbers quite balanced with the others.

IBM has certified Flexera One ITAM as an alternative to its **ILMT** for subcapacity reporting, particularly for software on Red Hat OpenShift and Kubernetes, but only in environments, Flexera One ITAM supports. Flexera FlexNet Manager, similar to ServiceNow isn't IBM-certified but can be used under special terms or within IBM's **IASP** initiative if managed by an authorized provider.

SAM Tool Rating by Gartner - 08.2023							
SAM Tool	**Discovery**	**ELP**	**Optim.**	**Impl.**	**Success**	**Support**	**Price**
ServiceNow SAM Pro	4.1	4.2	**4.2**	**4.2**	**5.0**	**4.5**	4.1
USU LiMa	3.9	4.2	**4.2**	**4.2**	4.3	4.4	**4.5**
Snow License Manager	**4.4**	**4.3**	**4.2**	**4.2**	4.4	4.3	4.4
FlexNet Manager Suite	4.3	**4.3**	**4.2**	4.1	4.4	4.3	4.4

Table 2.3: Ratings for each SAM Tool by Gartner

A complete insight into the leading SAM tools is provided below.

[10] *Gartner - SAM tool comparisons between Snow, Flexera, ServiceNow, USU*
[11] *Gartner - Last Magic Quadrant for SAM Tools*

2.5.1 ServiceNow Software Asset Management (SAM) Pro

Description: Offers a comprehensive set of software asset management tools designed to help organizations get the most out of their IT investments by optimizing license spend and minimizing the risk associated with non-compliance issues. This platform features automated workflows, powerful analytic dashboards, detailed reports on utilization trends, built-in security controls, and more – all designed to improve visibility into the IT environment while driving cost savings.

- **Features:** Alerts and actionable insights, automatic app identification, detection, and removal of unauthorized apps.
- **HQ location:** Santa Clara, California, U.S.
- **Pros:** Intuitive and easy to start for beginners, excellent support.
- **Cons:** Third-party integration slows it down, and lacks some essential SAM features.
- **Licensing model:** Subscription Unit
- **Products offered:** Software Asset Management Professional, Software Asset Management Enterprise
- **IBM capable:** IBM's Authorized SAM Provider (**IASP**) program
- **Oracle capable:** Database, Database options and Fusion Middleware
- **SAP capable:** Subscription Unit
- **SAM verified rating:** Gartner: 4.2/5 (95 reviews)

2.5.2 USU LiMa (License Management)

Description: Is an enterprise-grade solution mainly for global organizations that need advanced asset tracking capabilities tailored to meet their specific industry requirements while still offering robust automation options in order to ensure accurate compliance with internal policies as well as external regulations like GDPR or HIPAA. USU SAM leverages big data analytics along with predictive models powered by AI/ML algorithms in order to support IT investments while keeping costs under control.

- **Features:** Ensures license compliance, and cost optimization by removing excess and unused licenses.
- **HQ location:** Möglingen, Germany
- **Pros:** Highly customizable, user-friendly, excellent support, quick, and easy to deploy.

- **Cons:** Requires experience to drive value, outdated GUI.
- **Licensing model:** Per device, per **FTE** or managed accounts of dedicated SaaS vendors, per managed Oracle server
- **Products offered:** USU License Management, USU Discovery, USU Oracle Optimization, USU SAM Analytics, USU Optimization for SAP, SaaS Optimization
- **IBM capable:** IBM's Authorized SAM Provider (**IASP**) program
- **Oracle capable:** Database, Database options and Fusion Middleware
- **SAP capable:** Per SAP user
- **SAM verified rating:** Gartner: 4.4/5 (124 reviews)

2.5.3 Snow License Manager

Description: Provides a singular view of the data. It can track the lifecycle of applications and has a built-in calculator that can effectively capture the licensing position, even across different licensing metrics. The program is fast and can automatically identify more than 95.000 software publishers and over 550.000 applications. It also comes with flexible deployment options and the ability to access data from multiple points.

- **Features:** Visibility of software assets, license optimization, compliance with licensing rules, contract management.
- **HQ location:** Stockholm, Sweden
- **Pros:** Fast to deploy, simple to operate, excellent customer support.
- **Cons:** Slow and crashing reporting, steep learning curve, outdated UI.
- **Licensing model:** Per device
- **Products offered:** Snow Adoption Tracker, Snow Spend Optimizer, Snow Optimizer for SAP Software
- **IBM capable:** IBM's Authorized SAM Provider (**IASP**) program
- **Oracle capable:** Database and Database options
- **SAP capable:** Per SAP user
- **SAM verified rating:** Gartner: 4.5/5 (305 reviews)

2.5.4 Flexera FlexNet Manager Suite

Description: Is a comprehensive suite of digital asset management solutions specifically for medium and large enterprises that need a powerful but easy-to-use system that can handle complex licensing requirements across multiple vendor products and locations around the world. The platform offers integrated automation capabilities, enhanced control over license utilization data, cost savings opportunities through policy enforcement, and more to keep your organization compliant while helping to maximize **ROI** on IT investments over time.

- **Features:** Insights about risks and costs, integration with inventory solutions, visualization of software estate.
- **HQ location:** Itasca, Illinois, U.S.
- **Pros:** Flexible, customizable, maintains license compliance, accurate data, and easy cloud management.
- **Cons:** Poor customer service, steep learning curve, time-consuming initial setup.
- **Licensing model:** Per device
- **Products offered:** Flexera One ITAM, FlexNet Manager Suite or FlexNet Manager for SAP, SaaS Management
- **IBM capable:** IBM's Authorized SAM Provider (**IASP**) program
- **Oracle capable:** Database, Database options, and Oracle Fusion Middleware
- **SAP capable:** Per SAP user
- **SAM verified rating:** Gartner: 4.2/5 (292 reviews)

2.6 AI and Software Asset Management

The integration of Artificial Intelligence within various industries has shown a new era of innovation. The benefits of ***AI*** in IT Asset Management can be in multiple areas as outlined by ITAM review [12].

In this research, two potential scenarios of using AI in the domain of Software Asset Management (**SAM**) were implemented, as detailed in Chapter 4, section 5.

- **Automation:** AI can simplify SAM by automating tasks such as inventory and license management, guiding tasks for potential optimization.
- **Predictive Analytics:** AI can analyze usage data enabling future trend predictions, aiding optimal license allocation and informed software procurement decisions.
- **Risk Management:** AI detection of risks, such as non-compliance or security vulnerabilities, can allow timely mitigation strategies, enhancing software asset security and compliance.
- **Cost Optimization:** AI can identify underutilized software, guiding organizations to optimize software expenditure, leading to potential cost savings.
- **Cloud Software Management:** As businesses shift to the cloud, AI can benefit in monitoring cloud software usage, license management, and expenditure optimization, navigating the increased complexity of cloud-based software assets.

[12] *ITAM review - The Power of AI*

2.7 Importance of Automation

Figure 2.8 showcases the *"IT Talent Quadrant"* based in the US from *Gartner*'s 2021 *IT Skills Roadmap*. It ranks IT skills by demand, emphasizing the importance of automation, and contrasts job openings with candidate availability as of July 2021.

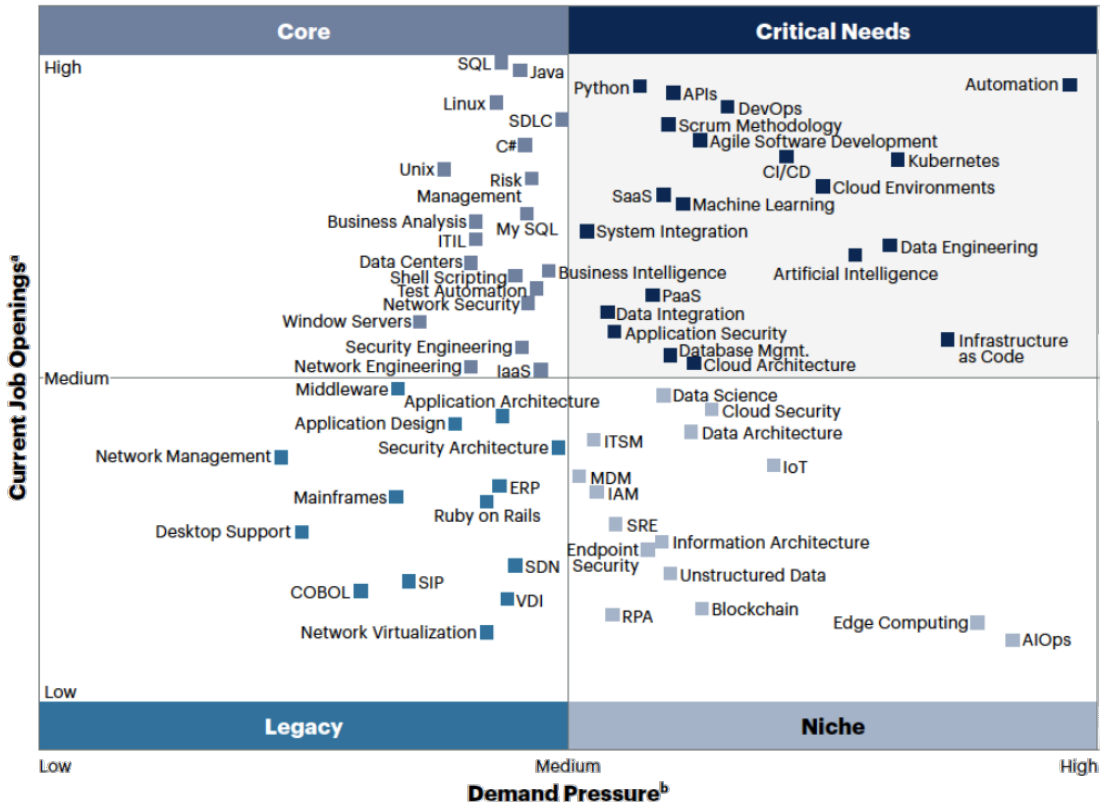

Figure 2.8: IT Talent Quadrant

Source: Gartner - 2021 IT Skills Roadmap

Software Asset Management tools automate multiple tasks required to maintain compliance with software license usage rights, thereby controlling software spending. They facilitate an in-depth analysis of software assets by analyzing software license entitlements. Furthermore, they automate the import of software usage, including independent software vendors (ISVs) for effective license position (**ELP**)'s, and optimize software consumption.

Figure 2.9 shows that SAM is closely tied to the **ITIL** process. The **CMDB** is where software installations and devices (e.g., Windows/macOS operating systems, servers, workstations) are discovered. The inventory must work seamlessly with the discovery model (application recognized) to integrate automation during the most difficult stages.

Stages where the SAM operation team typically spends the most time are marked in red.

This is similar to the time-consuming manual import of software licenses or entitlements when no procurement mechanism is defined.

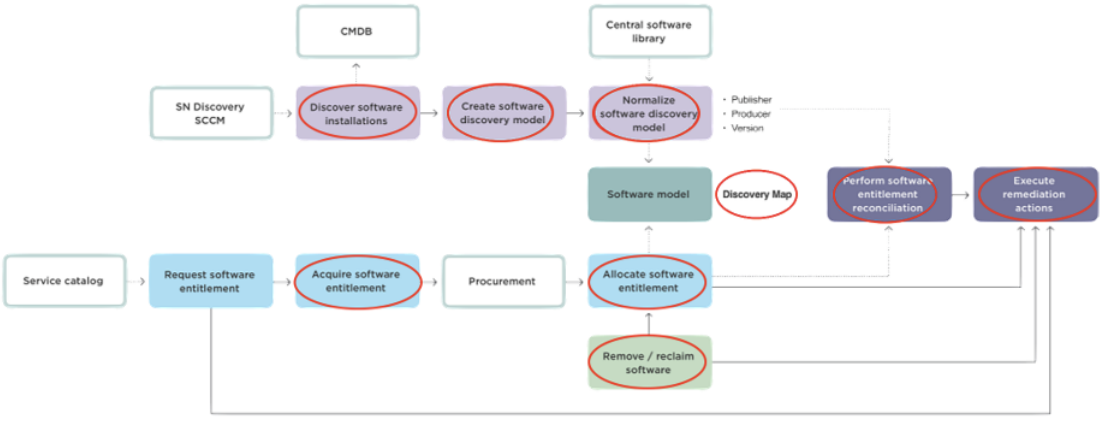

Figure 2.9: Integration of ITAM - Stages in Software Asset Management

Source: ServiceNow

Automated **SAM** simplifies the **IT** department's workload by handling repetitive tasks and streamlining software asset management. As organizations transition to the cloud and adopt user-based licensing, software licensing has become more straightforward. However, tracking user data and usage can be challenging without the proper tools. In the past, determining an organization's effective license position (**ELP**) would require significant time and effort.

Through automation, organizations can collect and analyze data on software usage, licensing, and costs, enabling them to gain insights into vendor performance in terms of compliance, cost optimization, and overall value.

This allows organizations to identify which tool performs better for their assets, providing a more reliable benchmark for major vendors and leading to improved vendor management and optimized software asset utilization.

Automation has been identified as a key strategy for improving the proactive and efficient functioning of IT Asset Management (**ITAM**). By automating certain processes, the **ITAM** team can reduce their workload and allocate their attention to other critical areas.

Key tasks that can be automated include the software recycling process, software deployment process (including patching and updates), software removal, licensing or software reports, compliance management, software request process, and software approval process. Several solutions available in the market can assist organizations in automating these time-consuming tasks, enabling **ITAM** professionals to focus on more strategic activities.

For instance, tools like **SCCM** can aid in software deployment and removal, while **ITAM** tools can automate the uninstalling of unused software based on predefined parameters.

Reports can be generated and automated using SCCM or specialized SAM tools, providing insights into software usage, compliance, and vendor-specific metrics. These reports can be scheduled for distribution at any time, reducing the manual effort required for compiling and sending them.

SAM tools also play a crucial role in highlighting compliance risks and non-compliance issues. They can provide alerts and notifications for breaches of compliance, reducing the need for constant manual monitoring. However, it is essential for users to remain vigilant and monitor compliance, as no system can guarantee optimized license usage or ensure correct user assignments.

Certain processes still require manual intervention, such as software license management, software license data entry, internal audits, contract management, and risk management. The overall management of software assets and licenses should remain a manual process, as machines can make mistakes. **SAM** professionals need to oversee these processes to ensure accurate license management, conduct internal audits, negotiate contracts, and address risks effectively.

As well, automation has its limitations. Some processes cannot be fully automated due to the lack of available systems. The need for human verification to ensure accuracy, the requirement for logical decision-making, and the necessity for SAM professionals to stay updated with their environment are essential. Manual interaction is crucial for assessing and analyzing data provided by SAM tools, as technology lacks the same rational thinking as humans.

While automation can significantly enhance **ITAM** processes, manual intervention remains necessary for critical tasks that require human expertise and decision-making. **SAM** professionals need to maintain an up-to-date understanding of their **SAM** estate to make informed decisions that optimize software usage, spending, and overall organizational benefits.

CHAPTER 3

Methodology

"The way we're engaging with customers is becoming more digitized, self-serving, and API-driven."

Rune Syversen, Founder of Crayon, 29.08.2020

In this study, to determine the appropriate methodology for analyzing the findings and validate the activities required for the SAM tools, the **DSR** methodology was applied. This involved measuring and validating activities using a service-oriented architecture process (SOA), identifying reiterative tasks on a BPM diagram, and validating the **KPI**s, which consists of records saved for each manual entry required from the *UI*. A process design for the activities enables us to understand, build, evaluate, and prioritize different scenarios. This also helps to present use cases for validating the results and later contrast them with IT experts and the SAM operations team.

3.1 Approach

Understanding the lack of existing processes that require automated activities, our research questions try to address the following:

RQ1: Which manual tasks can be replaced by a process?
RQ2: Which tasks require a certain prioritization over others?
RQ3: Which tool is more capable of reducing manual actions?

By addressing these research questions, this study seeks to offer a comprehensive understanding of task automation and prioritization within organizations. The potential for replacing manual tasks with automated processes will be assessed, with a focus on real-world applicability. Additionally, the study will explore the tools that minimize effort in manual interventions and how tasks can be ranked based on their importance to an

organization. The findings from the study aim to provide both a theoretical perspective and actionable insights for businesses looking to optimize their operations.

By this means, the most suitable practice for this research was the DSR framework to generate knowledge of how things can and should be constructed or arranged in a SAM tool (e.g., process design), and achieve a desired set of goals (e.g., removal of iterative common tasks), which are referred to as design knowledge (DK).

In the **DSR**, the environment defines the problem space in which the phenomena of interest reside. It is composed of people, organizations, and existing or planned technologies. In it are the goals, tasks, problems, and opportunities that define needs as they are perceived by stakeholders within the organization [39]. The knowledge base is composed of foundations and methodologies, to provide guidelines used in the built and evaluation phases. In Figure 3.1, Hevner's model for each of the phases is outlined.

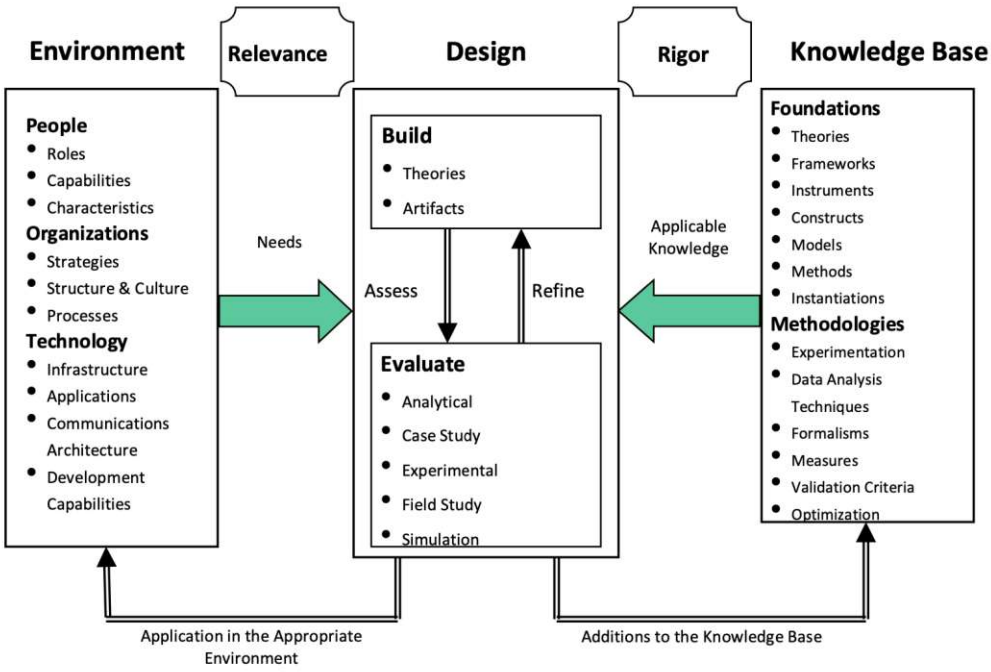

Figure 3.1: Design Science Research Framework (Adapted from (Hevner et al. 2004))

Source: Introduction to Design Science Research [39]

Some thoughts about the DSR knowledge publication schedule are presented in Table 3.1. It outlines a publication pattern on the DSR study [14]. The aspects of patterns suggest that they are not meant to be overly prescriptive.

In the DSR invention quadrant (e.g., producing algorithms for item associations, identifying common patterns, conducting proven experiments, introducing new methods, and

advancing new theoretical work), when a new product or idea is developed, it's first introduced and then observed in real-world settings. Researchers study its practical applications and use. In the field of information systems, if extensive knowledge already exists about such a product or idea, it reduces its perceived uniqueness.

Schema for a Design Science Research Study	
Section	**Contents**
1. Introduction	Problem definition, problem significance/motivation, introduction to key concepts, research questions/objectives, the scope of the study, an overview of methods and findings, theoretical and practical significance, and the structure of the remainder of the research. For DSR, the contents are similar, but the problem definition and research objectives should specify the goals that are required for the artifact to be developed.
2. Literature Review	Prior work that is relevant to the study, including theories, empirical research studies, and findings/reports from practice. For DSR work, the prior literature surveyed should include any prior design theory/knowledge relating to the class of problems to be addressed, including artifacts that have already been developed to solve similar problems.
3. Method	The research approach that was employed. For DSR work, the specific DSR approach adopted should be explained with reference to existing authorities.
4. Artifact Description	A concise description of the artifact at the appropriate level of abstraction to make a new contribution to the knowledge base. This section (or sections) should occupy the major part of the paper. The format is likely to be variable but should include at least the description of the designed artifact and, perhaps, the design search process.
5. Evaluation	Evidence that the artifact is useful. The artifact is evaluated to demonstrate its worth with evidence addressing criteria such as validity, utility, quality, and efficacy.
6. Discussion	Interpretation of the results: what the results mean and how they relate back to the objectives stated in the introduction section. Can include: a summary of what was learned, a comparison with prior work, limitations, theoretical significance, practical significance, and areas requiring further work. Research contributions are highlighted and the broad implications of the paper's results to research and practice are discussed.
7. Conclusions	Concluding paragraphs that restate the important findings of the work. Restate the main ideas in the contribution and why they are important.

Table 3.1: Publication Schema for a Design Science Research Study

To address these challenges and insights, a mixed-method approach will be used, combining quantitative and qualitative data collection and analysis techniques.

The methodology will consist of the following steps:

Data Collection

(a) **Survey:** A survey will be designed and administered to gather data on company size, software spending, and the tools currently in use. The survey will also include questions related to the perceived convenience of the tools based on various factors such as ease of use, integration capabilities, and scalability.

(b) **Interviews:** Semi-structured interviews will be conducted with SAM experts, IT managers, and stakeholders from different companies. These interviews will provide in-depth insights into the effectiveness of SAM automation in lowering IT expenses and detecting Shadow IT. The interviews will also explore the challenges faced in managing software assets and the requirements for effective repair mechanisms.

Quantitative Data Analysis In this thesis, part of the analysis is done using a spreadsheet. This choice is based on several advantages:

(a) **Cost-effective:** *Excel* is widely available and accessible, either as a pre-installed program or as a free download. This makes it a cost-effective option for data analysis.

(b) **Portable:** *Excel* files can be easily shared via email, allowing for convenient collaboration and communication with others involved in the research. Additionally, *Excel* files can be viewed on most smartphones, ensuring accessibility and flexibility.

(c) **All-in-one program:** *Excel* offers a comprehensive set of tools and functions for data analysis, including formulas, charts, pivot tables, and statistical functions. This makes it versatile for conducting quantitative analysis.

Validation and Verification

(a) **Expert Validation:** The research findings and conclusions will be reviewed and validated by SAM experts and IT professionals. Their feedback will help ensure the accuracy and reliability of the results.

(b) **Peer Review:** The thesis will undergo a peer review process to receive feedback and suggestions from other researchers and academics. This review will help enhance the quality and rigor of the research.

Ethical Considerations

The research will adhere to ethical guidelines, ensuring the confidentiality and anonymity of the participants. Informed consent will be obtained from survey respondents and interviewees, and their privacy and data protection rights will be respected.

3.2 Research Methodology

Design Science Research (**DSR**) is a methodology used in the field of information systems and computer science to develop and evaluate innovative artifacts, such as software systems, models, frameworks, or methods.

By adopting **DSR**, a problem-driven approach can be taken, identifying the specific issues and requirements within **SAM** that need to be addressed. With this approach, innovative artifacts, such as SAM tools, models, or frameworks can be designed and created. Based on feedback and empirical evidence continuous refining and improvement are enabled by the iterative nature of DSR.

By using KPIs to record the number of records automated, the effectiveness, performance, and possible cost savings can be measured. This approach allows for the evaluation and validation of the impact of the solutions, enabling informed decisions and comparisons with existing approaches.

The design science research methodology (DSRM) process model is shown in Figure 3.2, including 6 activities:

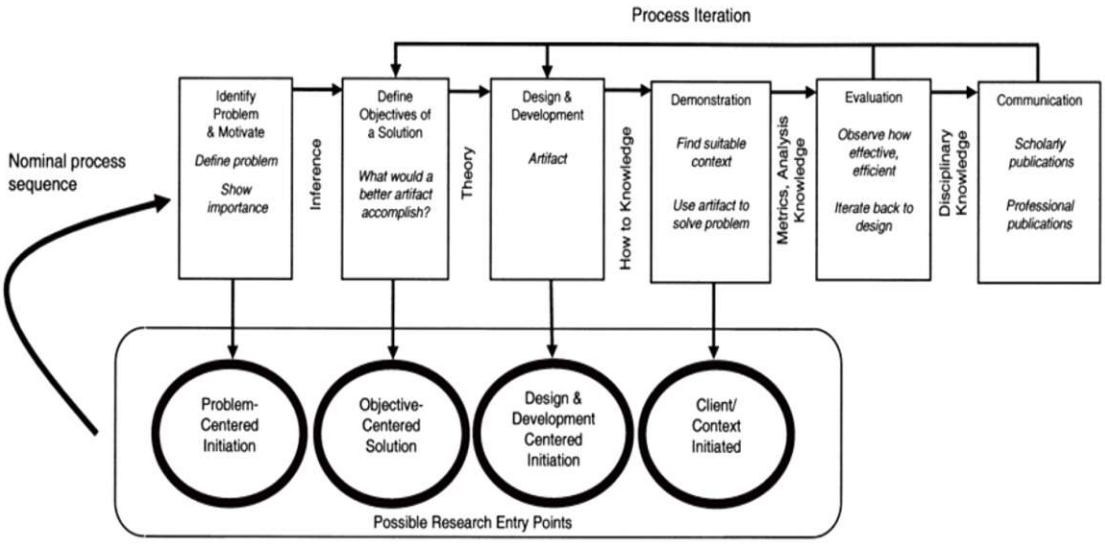

Figure 3.2: DSR Methodology Process Model (Adapted from Peffers et al. (2008))

Source: Introduction to Design Science Research [39]

1. **Identification and Motivation**: Define the problem and stress the importance of a solution, using knowledge of the problem's current state.
2. **Solution Objectives**: Establish quantitative or qualitative objectives based on problem definition.

3. **Design and Development**: Create an artifact with specific functionality and architecture.
4. **Demonstration**: Illustrate the artifact's utility through methods like experiments or simulations.
5. **Evaluation**: Compare the solution objectives with the artifact's actual performance, possibly leading to refinement.
6. **Communication**: Share details with stakeholders through tailored channels.

To quantitatively measure the results, multiple metrics are used through KPIs, providing details about their usage.

KPIs are important because they provide a fact-based methodology for measuring progress toward business goals. They help companies make informed decisions, focus on strategic priorities, and manage technology's impact on business performance. By tracking and analyzing these metrics, IT leaders can optimize costs, improve decision-making, and drive performance.

Various roles within an organization, such as the Office of the **CIO**, IT Finance, Infrastructure and Operations, Applications and Services, and Business Relationship Manager, can benefit from using **KPI**s to manage costs, performance, innovation, and business value.

When it comes to choosing which **KPI**s to track, it's important to focus on the ones that provide meaningful insights and align with specific business outcomes. Prioritizing essential metrics ensures that IT teams can effectively understand and communicate the impact of their actions on desired outcomes.

CHAPTER 4

Design and Implementation

Automation plays an important role in the Software Asset Management lifecycle, offering multiple advantages and simplifying regular tasks. This involves managing and optimizing software usage, ensuring compliance, and maximizing cost efficiency. By implementing automation throughout the lifecycle, organizations can reduce manual effort, improve accuracy, and achieve greater control over their software IT assets.

4.1 Holistic BPM-SOA Model Design

In the past, Business Process Management and Service-Oriented Architecture were treated as separate initiatives. However, it is now widely recognized that combining BPM and SOA is the most effective way to align business processes and IT resources, enabling businesses to become more agile and responsive to changes.

4.1.1 SOA and BPM

BPM is a modern approach to managing business services, enhancing agility and performance for digital transformation [38]. In contrast, **SOA** supports model changes to meet new business requirements by interconnecting workflows through loosely coupled, invokable, and business-oriented software components. [5]

By integrating BPM-SOA [32, 4, 24], businesses can analyze their processes to determine which parts should be re-engineered to increase efficiency and reduce costs.

The objective of combining both practices is to enable businesses to adapt quickly to changes and facilitate scalability. In Service-Oriented Architecture, there are two main approaches, **SOAP** and **REST**, which are used to support Business Process Management strategies.

Some significant benefits of using SOA include faster time to market, lower costs, improved application consistency, and increased agility. This results in a modular system that is easier to use and maintain, with the goal of increasing agility and reducing the total cost of ownership (*TCO*). It's interesting to read the interview done with Ismael Ghalimi, ex-CEO of Intalio [13], where he discusses the history of Business Process Management, industry challenges, and how Service-Oriented Architecture and *Web Service* enable infrastructure to make business-processes work on a large scale.

When exploring the combined benefits of BPM-SOA, different tools (e.g., using models and methods) can be designed/implemented to evaluate the usage of **SOAP** and **REST** architectures. This approach draws from a design-based methodology that has already proven success in generating new knowledge about BPM.

Gartner recommends starting with a small-scale Service-Oriented Architecture implementation, identifying specific business needs, and focusing on achieving them. The following steps will determine the main benefits:

- Improved efficiency, particularly with respect to business process execution
- Lower process administrative costs
- Higher visibility on existing/running business processes
- Reduced number of manual, paper-based steps
- Better service-level effectiveness
- Quicker implementation of processes
- Quicker time to market
- Shorter (overall) project cycles
- Overall reduction in the total cost of application development and maintenance

The API-managed tools market is rising, driven by SaaS and mobile apps. In 2021, the full life cycle API management market grew by 26.7%, attributed to global increases in API usage [1].

Gartner defines the full life cycle as the market for software that supports all stages of an API's life cycle - *planning and design, implementation and testing, deployment and operation, versioning and retirement.* This enables organizations to discover, design, develop, manage, and secure APIs, regardless of their size, region, or industry.

[1] *Gartner 2022 - Full Life Cycle API Management*

4.1.2 Using BPM and SOA Together

While SOA and BPM can operate independently, together they streamline API-enabled application management. Table 4.1 highlights their five key differences.

Business Process Management vs. Service-oriented Architecture	
BPM	**SOA**
- Business-driven - Top-down process approach - Reuses process model - Project-oriented - Success measured by business metrics and KPIs	- IT-driven - Bottom-up architectural approach - Reuses service implementation - Enterprise infrastructure-oriented - Success measured by architectural metrics, logical consistency

Table 4.1: Five main differences between BPM & SOA

According to ***Gartner***'s report, Table 4.2 outlines the advantages and implications of sharing technical services via *Web Service* platforms using SOA.

Service-oriented Architecture	
Benefits	**Implications**
Higher Upfront Costs - Diverse life cycle "speeds" - Synergy of different technologies - Optimal tech skills allocation - Processes visibility - Greater maintainability - Easier outsourcing/offshoring	**Architectural Partitioning** - Cultural change - Infrastructure (SOA backplane) - More formal methodology - Longer design time for services - Testing (unit/end-to-end)
Incremental Deployment - Gradual migration - Cost "spreading" across projects - Reduced maintenance cost	**More Distributed Infrastructure** - Extensive use of middleware - Transaction management - Debugging/troubleshooting - End-to-end management - More granular security - Metering/logging
Reuse of Services: - Faster time to deployment - Lower development cost - Consistent application implementation - Consistent process execution - Reduced risk	**Tighter Management/Governance** - Ownership/accountability - Taxonomy (service classification) - Metering/logging - Application/configuration - Greater transparency

Table 4.2: SOA Business Benefits and Implications

Source: Gartner's Roadmap to SOA

4.1.3 Process design for Software Asset Management

The **BPM-SOA** model is illustrated in Figure 4.1. This figure represents the six most significant activities in Software Asset Management, identified through workflow revisions and meetings with SAM tool experts. It includes the implemented methods for each activity.

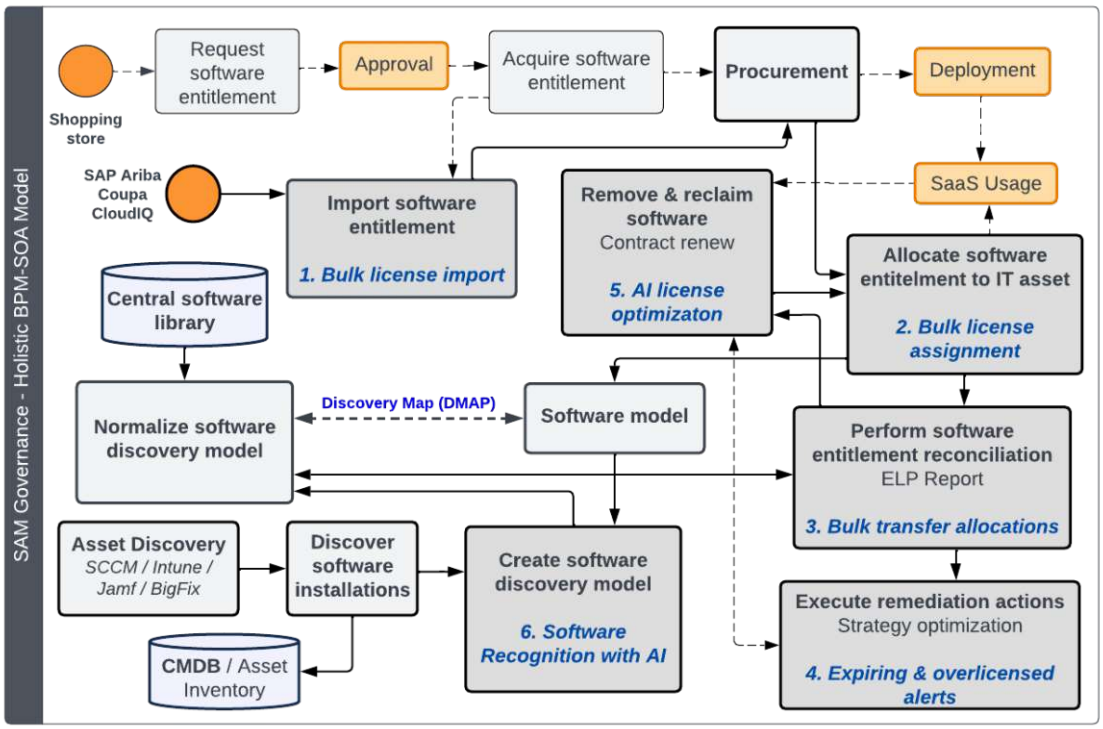

Figure 4.1: Lifecycle of processes with developed automations

In the presented diagram, two scenarios are observed: one where software is purchased from a shopping store (service catalog), and another when it is procured through an **ERP** system like *SAP Ariba*, *Coupa Software*, or *Crayon Cloud-iQ*. It's evident how each of the main six activities in the Business Process Management model can be made more flexible and effective by automating them through *Web Service* communication or by adapting the workflow process using the Service-Oriented Architecture model.

After consulting with experts, it was found that the activities consuming the most time are the imports of software entitlements and the creation of recognizable application products using software discovery models from the central software library. These activities have been simplified with the proposed developments in our implementation steps.

4.2 Developed Scripts for Automation

In ServiceNow, a *Proof of Concept* was defined to show key actions in the **SOA** model. These actions should work for all SAM tools and are listed with numerals identifying each business process. After speaking with **SAM** experts and the operations team, a library called *SAMAutomationUtils* was implemented to include all the methods required. This library shows how manual tasks can be replaced more easily for each of the key activities. In each developed function the code is also shared in the repository of ***GitHub*** using **XML** format. In ***ServiceNow***, after downloading the files, all settings can be imported by following the instructions provided in the *link*.

1. **Bulk license import:** This implementation simplifies the process of importing raw data on any data table. The solution is based on two parameters "jsonData" with the exact field names and values to read and the "tableName" for inserting the data. This function can save multiple processes in a workflow process which requires automating data insert in multiple tables, either using REST calls, service request forms, or maintenance scripting.

   Available at:
   Method *bulkDataImport*: sys_script_include.xml

   a) Client calls

```
var samAuto = new SAMAutomationUtils();
var jsonData = [{
    "asset_tag": "DEMO TEST MS VISIO",
    "software_model": "b5a4ec6287bde9109120fd98cebb3541", //Microsoft Visio
    "agreement_type": "generic",
    "product_type": "subscription",
    "subscription_period": "monthly",
    "unit_cost": "30.23", //euros
    "start_date": new Date("August 6, 2023"),
    "end_date": new Date("August 6, 2024"),
    "metric_group": "6e2a21987f222200fa0d328c4efa915c", //Common
    "license_metric": "f60759f5c31222006081face81d3ae7b", //Per User
    "unlimited_license": false, //if not to define purchased_rights field
    "purchased_rights": 10,
}];
var tableName = "alm_license";
samAuto.bulkDataImport(jsonData, tableName); //main function call
```

b) Server-side library function

```
var SAMAutomationUtils = Class.create();
SAMAutomationUtils.prototype = Object.extendsObject(
    AbstractAjaxProcessor, {

    //import software entitlements in one run
    bulkDataImport: function(jsonData, tableName) {

        var readJson = JSON.parse(JSON.stringify(jsonData));

        gs.print('total elements: ' + readJson.length);
        gs.print('JSON file read is: ' + JSON.stringify(readJson
    , null, 2));

        var jsonFields = {};
        var tableCallGR = new GlideRecord(tableName);
        tableCallGR.initialize();

        for (var i = 0; i < readJson.length; i++) {

            var obj = readJson[i];

            for (var key in obj) {
                //gs.print('key: ' + key + ', value: ' + obj[key
    ]);
                jsonFields[i] = obj[key];
                tableCallGR.setValue(key, obj[key]);
            }
            var response = tableCallGR.insert();
        }

        //gs.print('\n\nJSON stored in table [' + tableName +
    '], with response: ' + response);
        gs.print(JSON.stringify(jsonFields, null, 2));

    },
    type: 'SAMAutomationUtils'
});
```

c) Output result

```
insert  alm_license 1
*** Script: total elements: 1
*** Script: JSON file read is: [
  {
    "asset_tag": "DEMO TEST MS VISIO",
    "software_model": "b5a4ec6287bde9109120fd98cebb3541",
    "agreement_type": "generic",
    "product_type": "subscription",
    "subscription_period": "monthly",
    "unit_cost": "30.23",
```

```
    "start_date": "2023-08-06T07:00:00.000Z",
    "end_date": "2024-08-06T07:00:00.000Z",
    "metric_group": "6e2a21987f222200fa0d328c4efa915c",
    "license_metric": "f60759f5c31222006081face81d3ae7b",
    "unlimited_license": false,
    "purchased_rights": 10
  }
]
```

2. **Bulk license assignment** This automation allows the allocation of software licenses to users and devices in bulk based on a service request or workflow automation.

   Available at:
   Method *oneLicenseToManyUsers*: sys_script_include.xml
   Method *oneLicenseToManyDevices*: sys_script_include.xml

   a) Client calls

```
var samAuto = new SAMAutomationUtils();

//one license to multiple users
var userList = "David Pastrana, Alexander Cora";
var softwareLicense = "E3 - Microsoft Windows Server Standard";
samAuto.oneLicenseToManyUsers(userList, softwareLicense);

//one license to multiple CI (devices)
var devicesList = "ACME Citrix License server, HR Database";
var deviceLicense = "E3 - Microsoft Windows Server Standard";
samAuto.oneLicenseToManyDevices(devicesList, deviceLicense);
```

   b) Server-side library function

```
var SAMAutomationUtils = Class.create();
SAMAutomationUtils.prototype = Object.extendsObject(
    AbstractAjaxProcessor, {

    //create PO notification before expiring a contract

    //allocate one software entitlement to multiple users
    oneLicenseToManyUsers: function(userList, entitlement) {

        userList = userList.split(',');

        gs.print('userList ' + userList);

        var validLicense = new GlideRecord('alm_license');
        if (!validLicense.get('display_name', entitlement)) {
            gs.print('Exit: entitlement provided does not exit.'
    );
```

```
16              return;
17          }
18
19          gs.print('license: ' + entitlement);
20
21          var userAllocGR = new GlideRecord('alm_entitlement_user'
    );
22          userAllocGR.initialize();
23
24          for (var i in userList) {
25
26              gs.print('user: ' + userList[i]);
27
28              var validateUser = new GlideRecord('sys_user');
29              if (validateUser.get('name', userList[i])) {
30                  userAllocGR.assigned_to = validateUser.sys_id;
31                  userAllocGR.licensed_by = validLicense.sys_id;
32                  userAllocGR.quantity = 1;
33                  userAllocGR.insert();
34                  gs.print('License ' + entitlement + ', properly
    assigned to user ' + userList[i]);
35              }
36          }
37      },
38
39      //allocate one software entitlement to multiple devices
40      oneLicenseToManyDevices: function(deviceList, entitlement) {
41
42          deviceList = deviceList.split(',');
43
44          gs.print('devices: ' + deviceList);
45
46          var validLicense = new GlideRecord('alm_license');
47          if (!validLicense.get('display_name', entitlement.
    toString())) {
48              gs.print('Entitlement name does not exit.');
49              return;
50          }
51
52          validLicense = new GlideRecord('alm_license');
53          if (!validLicense.get('display_name', entitlement))
    return;
54
55          var deviceAllocGR = new GlideRecord('
    alm_entitlement_asset');
56          deviceAllocGR.initialize();
57
58          for (var i in deviceList) {
59
60              gs.print('device: ' + deviceList[i]);
61
62              var validateDevice = new GlideRecord('cmdb_ci');
63              if (validateDevice.get('name', deviceList[i])) {
```

```
64                  deviceAllocGR.allocated_to = validateDevice.
    sys_id;
65                  deviceAllocGR.licensed_by = validLicense.sys_id;
66                  deviceAllocGR.quantity = 1;
67                  deviceAllocGR.insert();
68                  gs.print('License ' + entitlement + ', properly
    assigned to device ' + deviceList[i]);
69              }
70          }
71      }
72      type: 'SAMAutomationUtils'
73  });
```

c) Output result

```
1 update  alm_license 1
2 *** Script: License E3 - Microsoft Windows Server Standard,
    properly assigned to user David Pastrana
3 *** Script: user:  Test user
```

3. **Bulk transfer allocations:** This automation can automate the process of retiring allocations to licensed software when they are no longer needed or in use. It helps in optimizing license usage, ensuring compliance, and avoiding unnecessary costs.

   Available at:
   Method *moveLicenseAllocations*: sys_script_include.xml

   a) Client calls

```
1 var samAuto = new SAMAutomationUtils();
2 samAuto.moveLicenseAllocations('E3 - Microsoft Windows Server
    Standard','SW000060 - Adobe Systems Acrobat Catalog','user');
```

   b) Server-side library function

```
1  var SAMAutomationUtils = Class.create();
2  SAMAutomationUtils.prototype = Object.extendsObject(
       AbstractAjaxProcessor, {
3
4      //move all license allocations (user or devices) from an old
        to a new license, type can only be of type "user" or "asset"
        for devices
5      moveLicenseAllocations: function(oldEntitlement,
       newEntitlement, type) {
6
7          var itemsList = [];
8
```

```
9           if (type != 'user' && type != 'asset') {
10              gs.print('type can only have values: user or device.
    ');
11              return;
12          }
13
14          var oldValidLicense = new GlideRecord('alm_license');
15          if (!oldValidLicense.get('display_name', oldEntitlement)
    ) {
16              gs.print('Old entitlement name does not exit.');
17              return;
18          }
19
20          var newValidLicense = new GlideRecord('alm_license');
21          if (!newValidLicense.get('display_name', newEntitlement)
    ) {
22              gs.print('New entitlement name does not exit.');
23              return;
24          }
25
26          var newSoftwareModel = newValidLicense.software_model +
    '';
27          gs.print('New model: ' + newSoftwareModel + ', new
    license: ' + newValidLicense.sys_id);
28
29          var allocationGR = new GlideRecord('alm_entitlement_' +
    type);
30          allocationGR.addQuery('licensed_by.display_name',
    oldEntitlement);
31          allocationGR.query();
32          while (allocationGR.next()) {
33              allocationGR.licensed_by = newValidLicense.sys_id;
34              allocationGR.allocated_model = newSoftwareModel;
35              allocationGR.license_key = '';
36              allocationGR.update();
37              itemsList.push(allocationGR.assigned_to.name + '');
38              //gs.print(allocationGR.assigned_to.name + ' added
    .');
39          }
40
41          if (itemsList.length == 0) {
42              gs.print('No allocations.');
43              return;
44          }
45
46          var msg = 'All ' + type + ' allocations have been moved
    from (' + oldValidLicense.display_name.toString() + ') to ('
    + newValidLicense.display_name.toString() + ') - ' + type + '
    s reallocated: ' + itemsList.toString();
47          gs.print(msg);
48
49          oldValidLicense.work_notes = msg;
50          oldValidLicense.update();
```

```
51      }
52
53      type: 'SAMAutomationUtils'
54  });
```

c) Output result

```
1  update  alm_license 1
2  update  alm_entitlement_user  1
3
4  *** Script: New model: 79c1767f37601000deeabfc8bcbe5d4e, new
       license: ce3d8bda3784200044e0bfc8bcbe5d54
5
6  *** Script: All user allocations have been moved from (E3 -
       Microsoft Windows Server Standard) to (SW000060 - Adobe
       Systems Acrobat Catalog) - users reallocated: David Pastrana
```

4. **Expiring & overlicensed alerts:** Two functions are implemented. One allows the creation of a notification when a contract in use will soon expire, notifying that a purchase order is required (5 days ahead). The second function could be extended to take action on licenses that are over licenses and require action to fix them.

   Available at:
   Method *createPurchaseOrderNotification*: sys_script_include.xml
   Method *overlicensedNotification*: sys_script_include.xml
   Notification Email Script: sys_script_email.xml
   Notification action: sysevent_email_action.xml

   a) Client calls

```
1  var samAuto = new SAMAutomationUtils();
2
3  //notification for any expring contract (e.g., run every day)
4  //to take action and make a Purchase Order (PO)
5  samAuto.createPurchaseOrderNotification();
6
7  //notification for any overlicensed vendor (e.g., every week)
8  //to take action and optimize/reallocate licenses
9  samAuto.overlicensedNotification();
```

   b) Server-side library function

```
1  var SAMAutomationUtils = Class.create();
2  SAMAutomationUtils.prototype = Object.extendsObject(
       AbstractAjaxProcessor, {
3
```

```
4       //create PO notification before expiring a contract
5       //default is 5 days of week ahead
6       createPurchaseOrderNotification: function() {
7           var itemsGR = new GlideRecord('ast_contract');
8           var maxDaysRange = 5;
9           var query = 'endsRELATIVEGT@dayofweek@ahead@5^
        endsRELATIVELT@dayofweek@ahead@' + maxDaysRange;
10          itemsGR.addEncodedQuery(query);
11          itemsGR.query();
12
13          if (!itemsGR.hasNext()) return;
14
15          var totalItems = 0;
16          var totalVendors = [];
17          while (itemsGR.next()) {
18              totalItems++;
19              totalVendors.push((itemsGR.vendor.name) ? itemsGR.
        vendor.name + '' : 'empty');
20          }
21
22          var urlLink = gs.getProperty('glide.servlet.uri') + '
        ast_contract_list.do?sysparm_query=' + query;
23
24          var emailData = {
25              'currentDate': this.getCurrentDate(),
26              'totalItems': totalItems,
27              'totalVendors': totalVendors,
28              'urlLink': urlLink
29          };
30          emailData = JSON.stringify(emailData);
31
32          gs.eventQueue('sam.automation.createpo.notification',
        current, emailData);
33      },
34
35      //overalicensed notification using event template
36      overlicensedNotification: function() {
37          var itemsGR = new GlideRecord('samp_product_result');
38          var query = "over_licensed_amount>javascript:global.
        getCurrencyFilter('samp_product_result','over_licensed_amount
        ', 'EUR;0')";
39          itemsGR.addEncodedQuery(query);
40          itemsGR.query();
41
42          if (!itemsGR.hasNext()) return;
43
44          var totalItems = 0;
45          var totalCosts = 0;
46          while (itemsGR.next()) {
47              totalItems++;
48              totalCosts += parseFloat(itemsGR.
        over_licensed_amount.getValue());
49          }
```

```
50          totalCosts = this.addCommas(totalCosts) + this.
        getCurrency(itemsGR);
51
52          var urlLink = gs.getProperty('glide.servlet.uri') + '
        samp_product_result_list.do?sysparm_query=' + query;
53
54          var emailData = {
55              'currentDate': this.getCurrentDate(),
56              'totalItems': totalItems,
57              'totalCosts': totalCosts,
58              'urlLink': urlLink
59          };
60          emailData = JSON.stringify(emailData);
61
62          gs.eventQueue('sam.automation.overlicensed.notification'
        , current, emailData);
63      },
64
65      addCommas: function(value) {
66          return value.toFixed(2).toString().replace(/\B(?=(\d{3})
        +(?!\d))/g, ",");
67      },
68
69      getCurrency: function(object) {
70          return ' ' + object.over_licensed_amount.getCurrencyCode
        ().toString();
71      },
72
73      getCurrentDate: function() {
74          var gdt = new GlideDateTime();
75          return gdt.getDisplayValue();
76      }
77
78      type: 'SAMAutomationUtils'
79  });
```

c) Notification email script

```
1  //Email Body: ${mail_script:email.script.createpo.notification}
2  //Event: email.script.createpo.notification
3  (function runMailScript(template, event) {
4
5      var emailBodyData = JSON.parse(event.parm1);
6      template.print('The identified <b>' + emailBodyData.
       totalItems + '</b> contracts nearing expiration for the
       following vendors: <b>' + emailBodyData.totalVendors + '</b
       >.<br><br>Details can be accessed from the following link: <a
        href="' + emailBodyData.urlLink + '">link</a>.<br>');
7      template.print('Date of Notice: '+emailBodyData.currentDate)
       ;
8
9  })(template, event);
```

```
10
11 //Email Body: ${mail_script:email.script.overlicensed.
       notification}
12 //Event: email.script.overlicensed.notification
13 (function runMailScript(template, event) {
14
15     var emailBodyData = JSON.parse(event.parm1);
16     template.print('The identified <b>' + emailBodyData.
       totalItems + '</b> over-licensed products with a total cost
       of <b>' + emailBodyData.totalCosts + '</b>.<br><br>Details
       can be accessed from the following link: <a href="' +
       emailBodyData.urlLink + '">link</a>.<br>');
17     template.print('Date of Notice: '+emailBodyData.currentDate)
       ;
18
19 })(template, event);
```

d) Output result

```
1  <html>
2  <body><div>Hello,</div>
3  <div>
4  <p>The identified <b>4</b> contracts nearing expiration for the
       following vendors: <b>empty,Roxio,Borland,APC</b>.<br><br>
       Details can be accessed from the following link: <a href="
       https://crayondeutschlandgmbhdemo1.service-now.com/
       ast_contract_list.do?sysparm_query=
       endsRELATIVEGT@dayofweek@ahead@5^
       endsRELATIVELT@dayofweek@ahead@500">link</a>.<br>Date of
       Notice: 30.09.2023 05:00:24</p>
5  <p>Warm regards,</p>
6  <p>SAM Automation Team</p>
7  </div>
8  </body></html>
9
10 <html>
11 <body><div>Hello,</div>
12 <div>
13 <p>The identified <b>52</b> over-licensed products with a total
       cost of <b>30,061,468.50 USD</b>.<br><br>Details can be
       accessed from the following link: <a href="https://
       crayondeutschlandgmbhdemo1.service-now.com/
       samp_product_result_list.do?sysparm_query=
       over_licensed_amount>javascript:global.getCurrencyFilter('
       samp_product_result','over_licensed_amount', 'EUR;0')">link</
       a>.<br>Date of Notice: 30.09.2023 05:00:24</p>
14 <p>Warm regards,</p>
15 <p>SAM Automation Team</p>
16 </div>
17 </body></html>
```

5. **AI license optimization:** This automation can automate the process of retiring allocations to licensed software when they are no longer needed or in use. It helps in optimizing license usage, ensuring compliance, and avoiding unnecessary costs.

   Available at:
   Method *openAIHelper*: sys_script_include.xml
   REST Message: sys_rest_message.xml
   REST Method: sys_rest_message_fn.xml

   Setting max tokens to 1024 provides complete and extended answers, while a temperature of 0.8 offers variability without interpretation.

   a) Web Service configuration

```
REST Message: ChatGPT
Endpoint: https://api.openai.com/v1/chat/completions

HTTP Headers:
    Authorization: Bearer sk-{your_openai_key}
    Content-Type: application/json

HTTP Method:
    Name: POST message
    Method type: POST
    Content: {
        "model": "${model}",
        "messages": [
            {
                "role": "system",
                "content": "${prompt}"
            },
            {
                "role": "user",
                "content": "${question}"
            }
        ],
        "max_tokens": ${max_tokens},
        "temperature": ${temperature}
    }
```

b) Client calls

```
var samAuto = new SAMAutomationUtils();
var licGR = new GlideRecord('alm_license');
//licenses with a total cost exceeding 1M euros
licGR.addEncodedQuery("cost>javascript:global.getCurrencyFilter
    ('alm_license','cost', 'EUR;1000000')^install_status=1");
licGR.query();

var list = [];
var data = {};
while(licGR.next()) {
    data.license_name = licGR.display_name+'';
    data.license_metric = licGR.license_metric.name+'';
    data.license_type = licGR.product_type+'';
    data.license_unit_cost = licGR.unit_cost+'';
    data.license_total_cost = licGR.cost+'';
    data.licence_available_allocations = licGR.
    allocations_available+'';
    list.push(samAuto.jsonParse(data));
}

var tableContent = JSON.stringify(list).replace(/"/g, "'");
var promptMessage = 'As a SAM expert using only the json data,
    provide a possible optimization of licenses';
message = promptMessage + ' from the following json data: ' +
    tableContent;
samAuto.openAIHelper(message, 1024, 0.8);
```

c) Server-side library function

```
var SAMAutomationUtils = Class.create();
SAMAutomationUtils.prototype = Object.extendsObject(
    AbstractAjaxProcessor, {

    openAIHelper: function(message, maxTokens, temperature) {

        gs.print('Your message: ' + message);
        if (gs.nil(message) || gs.nil(maxTokens) || gs.nil(
    temperature)) {
            gs.print('Invalid call. Use the following example:
    new SAMAutomationUtils().openAIHelper("hello world", 1024,
    0.8);');
            return;
        }

        var restMsg = new sn_ws.RESTMessageV2('ChatGPT', 'POST
    turbo');
        restMsg.setStringParameterNoEscape('model', 'gpt-3.5-
    turbo');
        restMsg.setStringParameterNoEscape('prompt', 'Act as an
    expert in software licensing. Create the answer always in
    JSON format.');
```

```
15          restMsg.setStringParameterNoEscape('question', message);
16          restMsg.setStringParameterNoEscape('max_tokens', 2024);
17          restMsg.setStringParameterNoEscape('temperature',
    temperature);
18          var msgResponse = restMsg.execute();
19
20          var msgBody = msgResponse.getBody();
21          var httpCode = msgResponse.getStatusCode();
22
23          if (httpCode == 200 || httpCode == 201) {
24
25              var parsedBody = JSON.parse(msgBody);
26              gs.print('AI Response: ' + parsedBody.choices[0].
    message.content);
27              gs.print('Tokens used: ' + parsedBody.usage.
    total_tokens + ', prompt_tokens: ' + parsedBody.usage.
    prompt_tokens + ', completion_tokens:' + parsedBody.usage.
    completion_tokens);
28              return JSON.parse(parsedBody.choices[0].message.
    content);
29
30          } else {
31              gs.print('Connection error. code: ' + httpCode + ',
    response: ' + JSON.stringify(msgBody, null, 2));
32          }
33      },
34
35      jsonParse: function(data) {
36          return JSON.parse(JSON.stringify(data));
37      }
38
39      type: 'SAMAutomationUtils'
40  });
```

d) Output result

```
1  /*** Script: AI Response: Based on the provided JSON data, here
       is a possible optimization of licenses:
2
3  1. To optimize the licensing cost for IBM Content Collector for
       SAP Applications, consider reducing the number of available
       allocations to match the actual usage. Currently, there are
       3000 available allocations, which might be higher than the
       required number. Adjusting this number can help optimize the
       license usage and reduce the cost.
4
5  2. For Oracle WebLogic Server Enterprise Edition, consider
       reviewing the license metric. Currently, it is licensed based
        on "Per Processor." If there are instances or servers where
       the license is not fully utilized, consider switching to a
       different license metric that aligns with the actual usage,
       such as "Named User" or "Core."
```

```
6
7  3. Similar to the previous case, for Oracle DB Server Enterprise
       Partitioning, Oracle DB Server Enterprise OLAP, and Oracle
       DB Server Standard, review the license metric and switch to a
       more suitable metric if the current one is not aligned with
       the actual usage.
8
9  4. For Adobe Systems Acrobat 7 Standard, which is licensed on a
       per-user basis, review the license allocations. If there are
       users who no longer require the software, consider
       reallocating the licenses to other users or reducing the
       total number of licenses to optimize costs.
10
11 5. For IBM Workload Scheduler, check the available allocations
       and adjust them based on the actual usage. If the current
       number of allocations is higher than required, reducing it
       can help optimize licensing costs.
12
13 6. For SAP Sales/Service Order Processing for SAP_BASIS Releases
       <= 6.40 Contract, review the license allocations and adjust
       them based on the actual usage. If the current number of
       allocations is higher than required, reducing it can optimize
       licensing costs.
14
15 Note: The above recommendations are general suggestions based on
       the provided JSON data. It is recommended to consult with a
       software licensing expert to perform a comprehensive analysis
       and provide specific optimization strategies tailored to
       your organization's needs.
16 *** Script: Tokens used: 983, prompt_tokens: 599,
       completion_tokens:384*/
```

6. **Software Recognition with AI:** Usage of artificial intelligence techniques, primarily machine learning, to automatically identify and categorize software products and versions within a system. By analyzing patterns in data, AI models can detect and classify software assets, enhancing accuracy and efficiency in Software Asset Management processes. This automation reduces manual effort, improves compliance, and ensures up-to-date software inventories.

   Available at:
   Method *recognizeSoftwareWithAI*: sys_script_include.xml
   REST Message: sys_rest_message.xml
   REST Method: sys_rest_message_fn.xml

   Setting max tokens to 1024 provided complete answers, while a temperature of 0.8 or 1 would yields to the same results as variability is not required.

   a) Web Service configuration

```
REST Message: ChatGPT
Endpoint: https://api.openai.com/v1/chat/completions

HTTP Headers:
    Authorization: Bearer sk-{your_openai_key}
    Content-Type: application/json

HTTP Method:
    Name: POST message
    Method type: POST
    Content: {
        "model": "${model}",
        "messages": [
            {
                "role": "system",
                "content": "${prompt}"
            },
            {
                "role": "user",
                "content": "${question}"
            }
        ],
        "max_tokens": ${max_tokens},
        "temperature": ${temperature}
    }
```

b) Client calls

```
var useOpenAI = true;
var samAuto = new SAMAutomationUtils();

if (useOpenAI) {
    searchProductName = 'Red Gate Software SQL Compare';
    message = "Format the following product " + searchProductName
      + " with the following json structure: {'ppn': '', '
     publisher_name': '', 'product_name': '', 'product_version':
     '', 'product_edition': '', 'license_metric': '', '
     license_type': '', 'platform_type': '', 'product_language':
     ''} fill publisher and product name  leaving the other fields
      with empty value, do not use professional, return the answer
      as a JSON object.";
    var searchProduct = samAuto.openAIHelper(message, 1024, 0.8);
    gs.print(JSON.stringify(searchProduct, null, 2));
    var result = samAuto.recognizeSoftwareWithAI(searchProduct);
    gs.print(result);

} else {
    var searchProduct = {
        'ppn': '',
        'publisher_name': 'Postman, Inc.',
        'product_name': 'Postman',
        'product_version': '',
        'product_edition': '',
        'license_metric': '', //Per User, Per Device, User
     Subscription
        'license_type': '', //Perpetual, Subscription, Maintenance
     , Upgrade, Perpetual + Maintenance
        'platform_type': '',
        'product_language': ''
    };

    var result = samAuto.recognizeSoftwareWithAI(searchProduct);
    gs.print(result);
}
```

c) Server-side library function

```
var SAMAutomationUtils = Class.create();
SAMAutomationUtils.prototype = Object.extendsObject(
    AbstractAjaxProcessor, {

    //recognize, match a product license from the sam tool
    content library
    recognizeSoftwareWithAI: function(jsonData) {
        var json = JSON.parse(JSON.stringify(jsonData));
        if (json.license_type.toString().toLowerCase() == '
    perpetual') json.license_type = 'full';
```

```
8       if (json.license_type.toString().toLowerCase() == '
        perpetual + maintenance') json.license_type = '
        perpetual_maintenance';
9
10      //gs.print('JSON file read is: ' + JSON.stringify(json,
        null, 2));
11
12      var findProductGR = new GlideRecord('
        samp_sw_product_definition');
13      findProductGR.addEncodedQuery('publisher_part_numberLIKE
        ' + json.ppn + '^entitlement_definition.sw_product.publisher.
        nameLIKE' + json.publisher_name + '^entitlement_definition.
        sw_product.prod_nameLIKE' + json.product_name + '^
        entitlement_definition.version_displayLIKE' + json.
        product_version + '^entitlement_definition.
        edition_displayLIKE' + json.product_edition + '^
        product_typeLIKE' + json.license_type + '^license_metric.
        nameLIKE' + json.license_metric + '^entitlement_definition.
        platformLIKE' + json.platform_type + '^entitlement_definition
        .language.languageLIKE' + json.product_language);
14      findProductGR.query();
15      gs.print('Total: [ ' + findProductGR.getRowCount() + ' ]
        ');
16
17      var counter = 0;
18      var first50Results = [];
19      while (findProductGR.next()) {
20          var data = {};
21          data.ppn = findProductGR.publisher_part_number + '';
22          data.publisher = findProductGR.
        entitlement_definition.sw_product.publisher.name + '';
23          data.name = findProductGR.entitlement_definition.
        sw_product.prod_name + '';
24          data.version = findProductGR.entitlement_definition.
        version_display + '';
25          data.edition = findProductGR.entitlement_definition.
        edition_display;
26          data.type = findProductGR.getDisplayValue('
        product_type');
27          data.metric = findProductGR.license_metric.name + ''
        ;
28          data.platform = findProductGR.entitlement_definition
        .platform + '';
29          data.language = findProductGR.entitlement_definition
        .language.language + '';
30          first50Results.push(data);
31          if (counter == 50) break;
32          counter++;
33      }
34      return JSON.stringify(first50Results, null, 2);
35  },
36
37      gs.print('Your message: ' + message);
```

```
38          if (gs.nil(message) || gs.nil(maxTokens) || gs.nil(
    temperature)) {
39              gs.print('Invalid call. Use the following example:
    new SAMAutomationUtils().openAIHelper("hello world", 1024,
    0.8);');
40              return;
41          }
42
43          var restMsg = new sn_ws.RESTMessageV2('ChatGPT', 'POST
    turbo');
44          restMsg.setStringParameterNoEscape('model', 'gpt-3.5-
    turbo');
45          restMsg.setStringParameterNoEscape('prompt', 'Act as an
    expert in software licensing. Create the answer always in
    JSON format.');
46          restMsg.setStringParameterNoEscape('question', message);
47          restMsg.setStringParameterNoEscape('max_tokens', 2024);
48          restMsg.setStringParameterNoEscape('temperature',
    temperature);
49          var msgResponse = restMsg.execute();
50
51          var msgBody = msgResponse.getBody();
52          var httpCode = msgResponse.getStatusCode();
53
54          if (httpCode == 200 || httpCode == 201) {
55
56              var parsedBody = JSON.parse(msgBody);
57              gs.print('AI Response: ' + parsedBody.choices[0].
    message.content);
58              gs.print('Tokens used: ' + parsedBody.usage.
    total_tokens + ', prompt_tokens: ' + parsedBody.usage.
    prompt_tokens + ', completion_tokens:' + parsedBody.usage.
    completion_tokens);
59              return JSON.parse(parsedBody.choices[0].message.
    content);
60
61          } else {
62              gs.print('Connection error. code: ' + httpCode + ',
    response: ' + JSON.stringify(msgBody, null, 2));
63          }
64      },
65
66      jsonParse: function(data) {
67          return JSON.parse(JSON.stringify(data));
68      }
69
70      type: 'SAMAutomationUtils'
71  });
```

d) Output result

```
*** Script: Total: [ 4 ]
*** Script: [
  {
    "ppn": "P49U1M12",
    "publisher": "Red Gate Software",
    "name": "SQL Compare",
    "version": "-- Anything --",
    "edition": {},
    "type": "Perpetual + Maintenance",
    "metric": "Per User",
    "platform": "anything",
    "language": "-- Anything --"
  },
  {
    "ppn": "1922790",
    "publisher": "Red Gate Software",
    "name": "SQL Compare",
    "version": "Starts with 8.1",
    "edition": {},
    "type": "Perpetual",
    "metric": "",
    "platform": "anything",
    "language": "-- Anything --"
  },
  {
    "ppn": "S18U2M24",
    "publisher": "Red Gate Software",
    "name": "SQL Compare",
    "version": "-- Anything --",
    "edition": {},
    "type": "Maintenance",
    "metric": "",
    "platform": "anything",
    "language": "-- Anything --"
  },
  {
    "ppn": "S72U10M36",
    "publisher": "Red Gate Software",
    "name": "SQL Compare",
    "version": "-- Anything --",
    "edition": {},
    "type": "Subscription",
    "metric": "",
    "platform": "anything",
    "language": "-- Anything --"
  }
]
```

CHAPTER 5

Findings

Table 5.1 presents the findings obtained using the **DSR** methodology, focusing on different automated methods that show the potential monthly cost savings in Software Asset Management. Using a **BPM-SOA** approach, evaluations were conducted directly with the Software Asset Management team to engage them in challenging discussions and request feedback. The following methods estimate potential cost savings that could be achievable by applying the appropriate automated tasks.

- **Bulk License Import**: This process can import over 100 licenses simultaneously, saving 30 hours and resulting in cost savings of 3.000€.
- **Bulk License Assignment**: In this operation, licenses can be assigned to multiple users at once based on **API** by connecting a procurement source system. It leads to 5 hours saved and 500€ cost savings.
- **License Transfer (Bulk transfer allocations)**: This operation applies to allocate more than 200 users and devices to new licenses based on certain usage rights. Run by a SAM expert, it could save 10 hours and result in a cost savings of 1.000€.
- **Expiring & Over-Licensed Alerts**: This operation alerts about expiring contracts or over-licensed products, saving 5 hours and resulting in 500€ cost savings.
- **AI License Optimizations**: Run by a SAM expert, this operation could optimize software entitlements and save 25 hours of effort in identifying optimization scenarios, leading to a 2.500€ in cost savings.
- **Software Recognition with AI**: This operation recognizes applications and is run by a SAM expert. It saves 20 hours and leads to a cost savings of 2.000€.

For enterprises, automating the management of approximately 1.000 software entitlements monthly, as referenced in Table 5.1, could save an average of 95 hours and reduce costs by 7.000€; these figures are based on estimated automated records and serve as a reference.

Software Asset Management Operations (potential cost savings per month)							
Script name (flow ready)	**KPI**	**Success**	**Value (0-10)**	**Impl. (0-10)**	***h*ours saved**	**Cost savings (*h* * 100€)**	**SAM tools supported**
1. Bulk license import	Number of licenses to bulk import	> 100 records via SR & workflow approval	10	10	30h	3.000€	ServiceNow, Snow, Flexera, USU
2. Bulk license assignment	Number of licenses assigned to multiple users	> 50 records via SR & workflow approval	5	7	5h	500€	ServiceNow, Snow, Flexera, USU
3. Bulk transfer allocations	Number of users and devices moved to a new license	> 200 Script to run by the SAM expert	7	6	10h	1.000€	ServiceNow, Snow, Flexera, USU
4. Expiring & over-licensed alerts	Number of contracts expiring & Number of overlicensed products	> 50 Script to run by the SAM expert	8	8	5h	500€	ServiceNow, Snow, Flexera, USU
5. AI license optimization	Number of entitlement corrections	> 5 Script to run by the SAM expert	10	9	25h	2.500€	ServiceNow, Snow, Flexera, USU
6. Software Recognition with AI	Number of applications recognized	> 10 Script to run by the SAM expert	10	9	20h	2.000€	ServiceNow, Snow, Flexera, Usu
Total					**95h**	**7.000€**	

Table 5.1: Potential Cost Savings in SAM Operations

CHAPTER 6

Conclusion

In this study, a **BPM-SOA** model was introduced to optimize key activities in the business processes. Specific software components (services) were developed to support business activities, followed by an assessment of potential financial savings for businesses. The implemented model can be applied across multiple system products like *Flexera, Snow, USU (formerly Aspera), and ServiceNow*, showcasing potential savings that can go above 95 hours in managing software entitlements, licenses, allocations, and contract renewal operations for enterprise firms.

The defined process, which is supported by stored records and reviewed using **KPI**s measures, could provide savings exceeding 7.000€ by defining the right automated processes. Implementing a Service-Oriented Architecture design showed both monetary gains and strategic advantages, especially in automating service requests, procurement, and disposal workflows. This model allows the operations team to prioritize business-oriented tasks, increasing agility and efficiency within the business.

Numerous studies confirm that customizing workflow processes plays a significant role in IT Asset Management governance. The development of reusable components to support software-related operations within workflows can help minimize manual errors, ensure licensing agreement compliance, and allow staff to focus on other strategic initiatives. Based on the constantly changing and increasingly complex nature of software licensing models, automation has transitioned from being a basic feature to an essential component during the **ITAM** program.

For future research, it's worth exploring the potential of automation in managing hardware costs, especially as hardware becomes more service-oriented. Investing in automating hardware assets might reveal additional opportunities for cost savings and operational improvements in the overall **ITAM** strategy.

There are other areas worth exploring, particularly the sharing of resources among **SAM** tools. At present, all these tools are proprietary. This means each has its own repository

for software model matching and its own metrics specific to the software vendor. With the open data approach supported by state guidelines, there's potential to enhance collaboration between different **SAM** tools. Both could benefit from shared resources, such as software/hardware content libraries, accessible from public platforms.

It might be interesting to explore opportunities for comparing **SAM** tools as their demand continues to grow. A potential scenario would be standardizing a source of test data for all platforms from a shared repository and **API** where each **SAM** tool could import demo data. Firms often request comparisons and invest in costly ***POC***s to get a quick impression of the tool for specific products, metrics, reports, and dashboards from their ***Tier-1*** vendors.

Another promising area to research is the role of Artificial Intelligence in Software Asset Management. This thesis briefly touched upon two ***AI*** methods and their potential benefits for guiding licensing costs. By examining existing table records and fields that don't breach company privacy (such as purchased rights, license types, allocations used, and license costs) in **JSON** format and sharing them with **OpenAI**, more profound insights might be achieved. This could assist in identifying potential scenarios for optimization, which are yet to be investigated.

Glossary

Activity-Based Costing is a method that assigns indirect costs to products and services by identifying and measuring the activities that consume resources. 20, 93

Artificial Intelligence is the simulation of human intelligence in machines. It enables computers to perform tasks like learning, reasoning, and problem-solving. 48, 86, 93

Application Programming Interface is a set of protocols that enable different software components to communicate and transfer data. 93

Application Portfolio Management involves scoring, monitoring, and managing software application performance to inform and enhance decision-making processes. 37, 93

Business IT Alignment is the process of ensuring that an organization's information technology (IT) aligns effectively with its business goals and objectives. 20, 93

Business Process as a Service is a cloud computing model that delivers specific business processes or functionalities as cloud-based services to streamline and optimize business operations. 18, 93

Business Process Management is the optimization and automation of business processes to enhance efficiency and adaptability, often using SOA principles. 18, 59, 60, 62, 93

Bring Your Own Device refers to a policy where employees can use their personal devices, like smartphones, laptops or tablets, for work-related tasks and access company resources. 20, 93

Bring Your Own License licensing model that lets companies use their licenses flexibly, whether on-premise, or in the cloud. 93

Client Access Licensing is a licensing model used by many software vendors, where a license is required for each client device or user that accesses the software. 93

Chief Information Officer is a senior executive responsible for managing and implementing a company's information technology (IT) strategies and systems. 18, 93

Configuration Management Database is a database that tracks and manages an organization's IT assets and their configurations. 93

cores in software licensing, refers to a model where licenses are based on the number of processor cores running the software. 27, 34

Coupa Software is a cloud-based spend management platform that helps businesses optimize their procurement, expenses, and supply chain processes. 23, 62

Central Processing Unit in software licensing, refers to a model where licenses are based on the number of CPUs executing the software. 93

Crayon is the global leader in software asset management (SAM), volume licensing, and associated consulting and professional services, is also a global top 10 Microsoft **LSP** and worldwide top 3 **SPLA** distributor. 24

Crayon Cloud-iQ is a self-service procurement platform used for managing Cloud products, services, and economics across multiple vendors through a web portal. Cloud-IQ details. 23, 62

Design Science Research is a methodology used in fields like information systems and computer science to create new and innovative artifacts, such as models, methods, or systems, and to evaluate their effectiveness in solving real-world problems. 57, 93

effective license position is a report that details an organization's license compliance position with one or more software vendors. 33, 49, 50, 93

Enterprise Resource Planning software used by companies to manage daily activities like accounting, procurement, and supply chain operations. 26, 93

Everest Group is a consulting and research firm specializing in strategic IT, business services, and sourcing with a focus on offering insights and advisory to global clients. 21

Cloud Financial Operations is a framework and set of practices focused on managing and optimizing cloud costs and spending efficiently. 93

Flexera is a software company specializing in providing solutions for software asset management (SAM) and software license optimization. Flexera FlexNet Manager Suite datasheet. 18, 19, 25

Full-Time Equivalent is used to calculate the number of licenses needed for employees, whether they work full-time or part-time. 94

G2 is a software review platform where users can provide feedback and reviews about various software products and services. 43

Gartner is a global research and advisory firm providing information, advice, and tools for leaders in IT, finance, HR, customer service and support, legal and compliance, marketing, sales, and supply chain functions. 17, 18, 20–22, 35, 42–44, 49, 60, 61

GitHub is a web-based platform for version control and collaborative software development, allowing teams to manage, share, and track changes to code repositories. 63

Hardware Asset Management is the practice of tracking, managing, and optimizing physical IT assets, such as computers and servers, throughout their lifecycle. 94

Infrastructure as a Service is a cloud computing infrastructure that provides compute, network, and storage resources over the internet, via a subscription model that can scale. 18, 94

IBM's Authorized SAM Provider is a program or initiative by IBM that designates specific service providers - Anglepoint, Deloitte, EY, KPMG - as officially recognized and approved for Software Asset Management (SAM) related to IBM products. 45–47, 94

IBM License Metric Tool is an IBM tool that helps manage and track IBM software licenses, ensuring compliance and optimizing software investments. 25, 94

Internet of Things refers to the network of interconnected physical devices that collect and exchange data through the internet. 39, 94

Internet Service Providers company that provides individuals and businesses with access to the interne. 94

IT Asset Management is the practice of tracking and managing IT assets, including hardware and software, throughout their lifecycle. 23, 24, 38, 41, 48, 50, 85, 94

IT Business Management is the practice of aligning IT resources and operations with an organization's business goals and priorities to optimize value and drive growth. 94

IT Infrastructure Library is a set of practices for IT service management that focuses on aligning IT services with the needs of the business. 94

IT Operations Management is the practice of managing daily operations of an organization's IT infrastructure for optimal performance. 94

IT Service Management is the practice of designing, delivering, managing, and improving IT services to meet the needs of an organization or its customers. 38, 42, 94

JavaScript Object Notation is a text-based format used to store and transmit data objects consisting of attribute-value pairs, making it ideal for data interchange between a server and a web application due to its simplicity and readability. 94

Knowledge Base is a self-serve online library of information about a product, service, department, or topic. 94

Key Performance Indicator is a quantifiable measure that helps organizations track their progress toward achieving their key business objectives. 94

Oracle License Management Services division of Oracle that helps customers manage and optimize their Oracle software licenses and ensure compliance with Oracle's licensing policies. 25, 94

Licensing Solution Partner is a Microsoft Partner who is authorized by Microsoft to be named as the Partner of record in enterprise licensing deals. 94

Machine Learning is a branch of AI where computers learn from data to make predictions or decisions. It improves performance on tasks as it processes more data. 94

OpenAI is the organization behind ChatGPT, aiming to advance and ethically distribute artificial intelligence technologies. 86

Platform as a Service is a cloud infrastructure layer that provides resources to build user-level tools and applications. 18, 94

Proof of Concept in software asset management is a short-term project or demonstration that verifies the feasibility and effectiveness of a particular SAM tool or solution in a real-world environment. 24, 43, 44, 63, 94

Processor Value Unit is a metric used in IBM software licensing to quantify the computing power of a server's processors, which in turn determines the licensing cost for IBM software on that server. 94

Representational State Transfers is an architectural style for networked applications that uses standard HTTP methods for communication. It treats web services as resources accessed via URLs. 18, 95

Rightsizing refers to the process of optimizing or adjusting resources, such as computing infrastructure or software licenses, to match an organization's actual needs, thereby reducing waste and cost inefficiencies. 18

Return on Investment measures the financial return on software investments relative to their costs. 95

RSA RSA Security is a prominent cybersecurity firm specializing in encryption, identity, and access management solutions to safeguard digital assets and ensure secure user access. 20

Software as a Service is a software distribution model in which a cloud provider hosts applications and makes them available to end users over the internet. 18, 95

Software Asset Management is the practice of managing, optimizing, and ensuring compliance of software licenses within an organization. 17, 24, 25, 27, 29, 30, 32, 38, 42, 48, 49, 59, 62, 77, 83, 86, 95

SAP Ariba is a cloud-based procurement and supply chain management solution that facilitates business-to-business transactions and collaboration between buyers and suppliers. 23, 62

System Center Configuration Manager is a Microsoft tool used for managing systems, deploying software, and ensuring endpoint security within an enterprise. 25, 95

ServiceNow in the context of Software Asset Management (SAM), is a platform and suite of tools designed to help organizations manage and optimize their software assets. ServiceNow SAM Pro datasheet. 18, 24, 25, 43, 63

Shadow IT refers to employees or departments independently using software, services, or technology solutions without IT approval, often due to dissatisfaction with IT services or the need for specific tools. 20, 21, 23, 35, 37, 38

Shelfware refers to software that a company has purchased but is not actively using or implementing, often resulting in wasted resources and costs. 20, 28, 34

Software License Management is the process of controlling, documenting, and ensuring compliance with software licenses and their associated terms and conditions. 95

Snow is a SAM tool that provides data normalization, cleansing, enterprise mobility management, and software license optimization services. Snow License Manager datasheet. 18, 25

Service-Oriented Architecture is a design approach where software is organized into reusable services that communicate with each other. It's known for its flexibility and scalability. 18, 42, 59, 60, 62, 85, 95

Simple Object Access Protocol is a messaging protocol that uses XML for message format and usually HTTP or HTTPS for message negotiation and transmission. 95

sockets in software licensing, refers to a model where licenses are based on the number of physical CPU sockets on a server or machine. 27

Software Rationalization is the process of evaluating and optimizing an organization's software applications to cut costs and enhance efficiency by making informed decisions about which applications to keep or retire. 20

Services Provider License Agreement is a Microsoft licensing program for service providers and independent software vendors. 95

Software Portfolio and Vendor Management refers to the management of an organization's software portfolio and its relationships with software vendors. 36, 95

total cost of ownership is the sum of all costs related to a product or system, including acquisition, operation, and maintenance, over its entire lifecycle. 19, 21, 60, 95

Tier-1 is a top-level industry supplier, such as Microsoft, IBM, Oracle, SAP (in the software industry), and Dell, HP, and Cisco (in the hardware industry), known for offering a wide range of products or services and holding a significant market share. 18, 26, 38, 42, 86

True-Ups is the process of aligning software licenses with actual usage, done annually to avoid overpaying or violating agreements. 95

User Interface refers to the space where interactions between humans and machines occur, often relating to the design of computer screens, websites, and apps. 95

USU formerly known as Aspera, is a company that specializes in Software Asset Management (SAM) solutions.USU LiMa datasheet. 18, 25

Web Service is a method of communication between two electronic devices over a network. 18, 22, 23, 42, 60–62

Extensible Markup Language is a markup language and file format for storing, transmitting, and reconstructing arbitrary data. 95

Acronyms

ABC Activity-Based Costing. 20, *Glossary:* Activity-Based Costing

AI Artificial Intelligence. 18, 48, 86, *Glossary:* Artificial Intelligence

API Application Programming Interface. 22, 23, 27, 35, 83, 86, *Glossary:* Application Programming Interface

APM Application Portfolio Management. 37, *Glossary:* Application Portfolio Management

BITA Business IT Alignment. 20, *Glossary:* Business IT Alignment

BPaaS Business Process as a Service. 18, *Glossary:* Business Process as a Service

BPM Business Process Management. 18, 24, 59, 60, 62, 83, 85, *Glossary:* Business Process Management

BYOD Bring Your Own Device. 20, *Glossary:* Bring Your Own Device

BYOL Bring Your Own License. *Glossary:* Bring Your Own License

CAL Client Access Licensing. 27, *Glossary:* Client Access Licensing

CIO Chief Information Officer. 18, 58, *Glossary:* Chief Information Officer

CMDB Configuration Management Database. 23, 42, 43, 49, *Glossary:* Configuration Management Database

CPU Central Processing Unit. 27, 34, *Glossary:* Central Processing Unit

DSR Design Science Research. 24, 53, 54, 57, 83, *Glossary:* Design Science Research

ELP effective license position. 33, 34, 44, 49, 50, *Glossary:* effective license position

ERP Enterprise Resource Planning. 23, 26, 62, *Glossary:* Enterprise Resource Planning

FinOps Cloud Financial Operations. 18, *Glossary:* Cloud Financial Operations

FTE Full-Time Equivalent. 46, *Glossary:* Full-Time Equivalent

HAM Hardware Asset Management. *Glossary:* Hardware Asset Management

IaaS Infrastructure as a Service. 18, 43, *Glossary:* Infrastructure as a Service

IASP IBM's Authorized SAM Provider. 44–47, *Glossary:* IBM's Authorized SAM Provider

ILMT IBM License Metric Tool. 25, 44, *Glossary:* IBM License Metric Tool

IoT Internet of Things. 27, 39, *Glossary:* Internet of Things

ISO International Organization for Standardization. 29

ISP Internet Service Providers. 22, *Glossary:* Internet Service Providers

IT information technology. 18, 19, 21, 22, 30, 37–39, 42, 50

ITAM IT Asset Management. 18, 19, 23, 24, 28, 37, 38, 40, 41, 48, 50, 51, 85, *Glossary:* IT Asset Management

ITBM IT Business Management. *Glossary:* IT Business Management

ITIL IT Infrastructure Library. 29, 30, 42, 49, *Glossary:* IT Infrastructure Library

ITOM IT Operations Management. 43, *Glossary:* IT Operations Management

ITSM IT Service Management. 38, 42, 43, *Glossary:* IT Service Management

JSON JavaScript Object Notation. 30, 86, *Glossary:* JavaScript Object Notation

KB Knowledge Base. *Glossary:* Knowledge Base

KPI Key Performance Indicator. 24, 53, 58, 85, *Glossary:* Key Performance Indicator

LMS Oracle License Management Services. 25, *Glossary:* Oracle License Management Services

LSP Licensing Solution Partner. 88, *Glossary:* Licensing Solution Partner

ML Machine Learning. *Glossary:* Machine Learning

PaaS Platform as a Service. 18, *Glossary:* Platform as a Service

POC *Proof of Concept*. 24, 43, 44, 63, 86, *Glossary: Proof of Concept*

PVU Processor Value Unit. 34, *Glossary:* Processor Value Unit

REST Representational State Transfers. 18, 59, 60, *Glossary:* Representational State Transfers

ROI Return on Investment. 47, *Glossary:* Return on Investment

SaaS Software as a Service. 18, 23, 26, 28, 35, 42, *Glossary:* Software as a Service

SAM Software Asset Management. 17, 23–25, 27, 29–32, 35–38, 42, 48–51, 57, 59, 62, 63, 77, 83, 85, 86, *Glossary:* Software Asset Management

SCCM System Center Configuration Manager. 25, 50, *Glossary:* System Center Configuration Manager

SLM Software License Management. *Glossary:* Software License Management

SOA Service-Oriented Architecture. 18, 24, 42, 59, 60, 62, 63, 83, 85, *Glossary:* Service-Oriented Architecture

SOAP Simple Object Access Protocol. 59, 60, *Glossary:* Simple Object Access Protocol

SPLA Services Provider License Agreement. 88, *Glossary:* Services Provider License Agreement

SPVM Software Portfolio and Vendor Management. 36, *Glossary:* Software Portfolio and Vendor Management

TCO total cost of ownership. 19, 21, 60, *Glossary:* total cost of ownership

True-Up True-Ups. 26, 28, 29, 33, *Glossary:* True-Ups

UI User Interface. 53, *Glossary:* User Interface

XML Extensible Markup Language. 30, 63, *Glossary:* Extensible Markup Language

List of Tables

List of Figures

Bibliography

[1] Benno Albert, Rodrigo dos Santos, and Claudia Werner. Software ecosystems governance to enable IT architecture based on software asset management. In *Book*, pages 55–60, 07 2013. [*Cited on pages 17 and 38.*]

[2] Misho Markovski Albert Pang and Martin Zdravkovski. Top 10 ITSM Software Vendors, Market Size and Market Forecast 2021-2026. *www.appsruntheworld.com*, 09 2022. [Online; accessed 21-November-2022]. [*Cited on page 43.*]

[3] M. Banu and Saravanan Krish. Optimizing the Cost for Resource Subscription Policy in IaaS Cloud. *Journal for optimizing costs*, 02 2014. [*Cited on page 38.*]

[4] Gopala Krishna Behara. BPM and SOA: a strategic alliance. *Retrieved*, 5(6):2011, 2006. [*Cited on pages 18 and 59.*]

[5] Saida Boukhedouma, Zaia Alimazighi, Mourad Chabane Oussalah, and Dalila Tamzalit. SOA based approach for interconnecting workflows according to the subcontracting architecture. In *IADIS International Conference-Collaborative Technologies 2011-CT'2011*, pages 1–11, 2011. [*Cited on page 59.*]

[6] Dave Dawson. Software Asset Management: Time for a Reset. *www2.deloitte.com*, 11 2018. [Review online; accessed 21-November-2022]. [*Cited on pages 17 and 39.*]

[7] Ali Durdu and Abdullah Tutğaç. Examination of Risks in Management of Software Licenses in Enterprises and Cost-Benefit Analysis. *Bilişim Teknolojileri Dergisi*, 15(4):467–479, 2022. [*Cited on page 17.*]

[8] Flexera. State of IT Asset Management Report. *www.flexera.com*, 10 2023. [*Cited on pages 17, 19, and 28.*]

[9] Gartner. The Cost of Downtime. *www.gartner.com*, 07 2014. [Review online; accessed 19-August-2023]. [*Cited on page 22.*]

[10] Gartner. Gartner Says Organizations Can Cut Software Costs by 30 Percent Using Three Best Practices. *www.gartner.com*, 07 2016. [Review online; accessed 21-November-2022]. [*Cited on page 20.*]

[**11**] Gartner and Magic Quadrant. A Magic Quadrant Leader in ITSM Platforms for ninth year in a row. *www.servicenow.com*, 11 2022. [Online; accessed 21-November-2022]. [*Cited on page 43.*]

[**12**] Elizaveta Gavrikova, Irina Volkova, and Yegor Burda. Strategic aspects of asset management: An overview of current research. *Sustainability*, 12(15):5955, 2020. [*Cited on page 38.*]

[**13**] Ismael Ghalimi. SaaS 2.0 onward! Business process management as an on-demand capability—Interview with Ismael Ghalimi of Intalio. *Journal of Digital Asset Management*, 4:252–259, 2008. [*Cited on page 60.*]

[**14**] Shirley Gregor and Alan R Hevner. Positioning and presenting design science research for maximum impact. *MIS quarterly*, pages 337–355, 2013. [*Cited on page 54.*]

[**15**] Jashwant Raj Gunasekaran. Minimizing Cost and Maximizing Performance for Cloud Platforms. In *Proceedings of the 21st International Middleware Conference Doctoral Symposium*, Middleware'20 Doctoral Symposium, page 29–34, New York, NY, USA, 2020. Association for Computing Machinery. [*Cited on page 38.*]

[**16**] Robert Heckman. Managing the IT procurement process. In *Enterprise Operations Management Handbook, Second Edition*, pages 367–383. Auerbach Publications, 2020. [*Cited on pages 35 and 36.*]

[**17**] Leila Helali and Mohamed Nazih Omri. Heuristic-based Approach for Dynamic Consolidation of Software Licenses in Cloud Data Centers. *International Journal of Intelligent Systems and Applications*, 13:1–12, 12 2021. [*Cited on page 38.*]

[**18**] Wang Heng and Liang Xin. Innovation of Equipment Asset Management System Based on Internet of Things Technology. In *Proceedings of the 2020 4th International Conference on Management Engineering, Software Engineering and Service Sciences*, ICMSS 2020, page 164–168, New York, NY, USA, 2020. Association for Computing Machinery. [*Cited on page 39.*]

[**19**] Axel Hochstein, Gerrit Tamm, and Walter Brenner. Service oriented IT management: benefit, cost and success factors. *ECIS 2005 Proceedings*, page 98, 2005. [*Cited on page 17.*]

[**20**] Samuel Idowu, Daniel Strüber, and Thorsten Berger. Asset Management in Machine Learning: State-of-Research and State-of-Practice. *ACM Comput. Surv.*, 55(7), dec 2022. [*Cited on page 38.*]

[**21**] Uptime Institute. Annual outages analysis 2023 - The causes and impacts of IT and data center outages. *www.uptimeinstitute.com*, 03 2023. [Review online; accessed 19-August-2023]. [*Cited on page 22.*]

[**22**] Ivanti. 43Spreadsheets. *www.ivanti.com*, 02 2020. [*Cited on page 19.*]

[**23**] Mohammad Kajbaf, Negar Madani, Ali Suzanger, Shirin Nasher, and Mehrdad Kalantarian. An IT service reporting framework for effective implementation of ITIL continual service improvement process conforming to ISO/EC 20000. In *Proceedings of Fifth International Conference on Digital Society (ICDS)*, 2011. [*Cited on page 42.*]

[**24**] Faouzi Kamoun. A Roadmap towards the Convergence of Business Process Management and Service Oriented Architecture. *Ubiquity*, 2007(April), apr 2007. [*Cited on pages 18 and 59.*]

[**25**] Alan J Keel, Michael A Orr, Robin R Hernandez, Eduardo A Patrocinio, and John Bouchard. From a technology-oriented to a service-oriented approach to IT management. *IBM Systems Journal*, 46(3):549–564, 2007. [*Cited on page 17.*]

[**26**] John P Kotter et al. Leading Change: Why Transformation Efforts Fail. *Engineering Management Review, IEEE*, 37, 01 2007. [*Cited on page 39.*]

[**27**] Joakim Lindvall, Viktor Löfqvist, and Martin Ollinen. Software Asset Management (SAM) and IT risks. *Retrieved*, 2019. [*Cited on pages 17, 39, and 42.*]

[**28**] Yvan Luxembourg and Tim Sommer. IT Costs - The Costs, Growth And Financial Risk Of Software Assets. *Journal of IT Costs*, 05 2013. [*Cited on pages 39 and 41.*]

[**29**] Mauricio Marrone and Lutz Kolbe. ITIL and the Creation of Benefits: An Empirical Study on Benefits, Challenges and Processes. In *DBLP*, 01 2010. [*Cited on page 42.*]

[**30**] Phara McLachlan. *Pocket CIO–The Guide to Successful IT Asset Management: Get to grips with the fundamentals of IT Asset Management, Software Asset Management, and Software License Compliance Audits with this guide.* Packt Publishing Ltd, 2018. [*Cited on pages 17, 18, and 39.*]

[**31**] Charles Nel and Wyhan Jooste. A technologically-driven asset management approach to managing physical assets: a literature review and research agenda for 'smart' asset management. *South African Journal of Industrial Engineering*, 27, 12 2016. [*Cited on page 39.*]

[**32**] Jasmine Noel. BPM and SOA: Better together. *White paper, IBM*, 2005. [*Cited on pages 18 and 59.*]

[**33**] Alex Nyström. Knowledge Discovery in Databases for enhanced Software License Management: An activity-based Clustering model. *Retrieved*, 2023. [*Cited on page 18.*]

[**34**] G2 reviews. Best Software Asset Management (SAM) Tools for Enterprise Businesses in 2023. *www.g2.com*, 08 2023. [Review online; accessed 17-August-2023]. [*Cited on page 43.*]

[**35**] Kobkiat Saraubon, Panurut Chinakul, and Rittinan Chanpen. Asset Management System Using NFC and IoT Technologies. In *Proceedings of the 2019 3rd International Conference on Software and E-Business*, ICSEB '19, page 124–128, New York, NY, USA, 2020. Association for Computing Machinery. [*Cited on page 39.*]

[**36**] Gad Selig and Jayne Wilkinson. *Implementing IT Governance: A Practical Guide to Global Best Practices in IT Management*. ACM Digital library, 12 2008. [*Cited on page 40.*]

[**37**] Mario Silic and Andrea Back. Shadow IT–A view from behind the curtain. *Computers & Security*, 45:274–283, 2014. [*Cited on page 20.*]

[**38**] Bui Quang Truong, Anh Nguyen-Duc, and Nguyen Thi Cam Van. A Quantitative Review of the Research on Business Process Management in Digital Transformation: A Bibliometric Approach. *Software*, 2(3):377–399, 2023. [*Cited on page 59.*]

[**39**] Jan Vom Brocke, Alan Hevner, and Alexander Maedche. Introduction to design science research. *Design science research. Cases*, pages 1–13, 2020. [*Cited on pages 54 and 57.*]

[**40**] Stephan Zimmermann and Christopher Rentrop. On the Emergence of Shadow IT-a Transaction Cost-Based Approach. In *ECIS*, 2014. [*Cited on page 20.*]