**B9IS105 - Enterprise Information Systems**



***Evaluation of order management business process in Coca Cola***

Module Lecturer: Dr. Obinna Izima

Akhil Ravuri – 10610852

Sharon Farrell - 10581193

Shastri Telikicherla – 10606539

**Contents**

1. Glossary ………………………………………………………………………………………….……. 3
2. List of illustrations ………………………………………………………………………….……. 4
3. About Coca-Cola ………………………………………………………………………………. ….. 5
4. Introduction

* Scope of the document …………………………………………………………………...…. 5
* Limitations …………………………………………………………………………………….…. 5

1. Enterprise Information Systems & their Architecture ………………………….… 6
2. Coca Cola’s legacy order management process

* Brief explanation & Challenges …………………………………………………………. 11

1. Improvised/New order management process

* Technology & Tools …………………………………………………………………………. 12
* Architecture ……………………………………………………………………………………… 16
* Business Improvements ……………………………………………………………………. 17

1. Recommendations ………………………………………………………………………………. 19
2. References ……………………………………………………………………………………………. 20

**Glossary**

1. EIS – Enterprise Information Systems
2. IT – Information Technology
3. NFR – Non Functional Requirements
4. COM – Component Object Model
5. ERP – Enterprise Resource Planning
6. SOA – Service Oriented Architecture
7. MAS – Multi Agent System
8. SOAP – Simple Object Access Protocol
9. WSDL – Web Service Description Language
10. RAS – Remote Access Service
11. RPA – Robotic Process Automation
12. SCM – Software Change Management
13. CRM – Customer Relationship Management
14. UDDI – Universal Discovery, Description, and
15. CDN – Content Delivery Network

**List of Illustrations**

1. Figure 1 - Framework for scenario-based EIS software architecture analysis
2. Figure 2: Database-centred layered architecture
3. Figure 3: Web services architecture
4. Figure 4: Manual order management process before implementing RPA
5. Figure 5: A glimpse of Microsoft power apps
6. Figure 6: Microsoft Flows/Power Automate
7. Figure 7: Shows different scenarios to trigger the flow
8. Figure 8: shows how to create a sample workflow using power automate
9. Figure 9: displays several criteria’s/scenarios/connectors that can be configured in power automate workflow
10. Figure 10: Microsoft Power Platform landscape diagram
11. Figure 11: New automated process of order management system using Microsoft Power Platform

**About Coca Cola**

Coca-Cola bottling company also known as Coca-Cola united is one of the largest and privately held Coca-Cola bottlers in United States and its parent company is famously known as Coca-Cola, it is not just a brand but a household name, there might be hardly an individual who has not heard about Coca-Cola. It was founded in 1902 and has its headquarters in Birmingham, Alabama (United States). It serves over 200,000 customers and uses thousands of trucks to deliver more than 600 product lines ranging from small stores to super stores and mega retailers.

**Introduction**

We referred to Microsoft’s customer success story whitepaper on Coca-Cola bottling company (<https://customers.microsoft.com/en-us/story/845187-coca-cola-bottling-company-united-consumer-goods-power-automate>) and extensively studied their order management business process and understood how they used technology and automation to improvise the process. Based on our understanding, we have prepared this report which outlines their legacy order management systems, the steps they took in terms of technology to improvise it and what are our recommendations that can further improve or add value to the business process.

***Scope***

* The scope of this report is only limited to business process study of order management system of Coca-Cola bottling company/ Coca-Cola United.
* Based on our understanding of the innovation/new architecture design, we have discussed and provided our recommendations at the end of this report which are solely our understanding and might be debatable or require discussion before any actual implementations.

***Limitations***

* Our understanding on Coca-Cola’s order management business process and the technical innovations done by them are solely limited and are as described in the Microsoft’s whitepaper and other technical articles, whose links are provided under references.

**Enterprise Information Systems & their Architecture**

Information Technology (IT) plays a pivotal role in industries, it is essential to for enterprises to have a dedicated and qualified IT professionals to stand out in the market and develop products that not only solve business problems but also add value to the organization. For example, if we were to design a control system, an engineer in the early 19th century would for sure work only with mechanical and electronic components, whereas majority of the tasks would be devoted to the tasks on information systems, if we were to design the same at his point of time. Enterprise Information Systems are important assets for the enterprises to achieve their business objectives but also to organize, plan, schedule, and control their business processes. These information systems are very important for some processes like supply chain management, EIS is a critical enabler for modern enterprises for process improvement and to achieve effectiveness. Information Technology deals only with the software, tools and the technologies and related methodologies whereas information systems include people, management, processes, and business functions along with the information technologies, so information systems are a bigger subset to information technologies. So, when we discuss further about Information Technology in this report that also get applied to Enterprise Information Systems. In IT, software architecture is a vital component, software architecture describes the system components as well as their topological relations. A proper software architecture will not only help the development team to plan the application development but also helps to analyse and make early decisions about a software system's high-level design before the development team starts getting their hands on to get the system developed. Due to such importance of software architecture, this field is always emerging and requires continuous learning, upgradation of skills and a skilled workforce to stand ahead of other enterprises.

Software architectures and development should balance the system's flexibility and the performance while achieving the functional/non-functional requirements for which they are intended and built. Although there are several articles and proposals on the software architectures and the different options, but there are only a handful that can help evaluate it against the requirements of an application.

There are a lot of software architectures described or proposed by different researchers. These descriptions can be specific to a business domain, could be with different aspects of Information Technology like automation, IoT, cloud, software development to solve business use cases etc. on the other hand these can also be for real time projects or can be service-oriented architecture. In designing and implementing an Information Systems, software architecture must adhere to the key business drivers, which can be either functional or non-functional requirements, this will ensure that information systems is aligned with the enterprise missions and adds value to the organization. In this report we have detailed on the software architecture that support the functional requirement of Coca-Cola United.

### Types of Software Architecture

Software architectures provide a view of the system components and their properties. It came into existence when enterprises started adapting Information Technology to support their businesses and different aspects such as programming languages, formal logic, architectural styles, design patterns. It represents the business structures and processes of an EIS and is a vital tool to support the assessment of design operations at an early stage.

A significant contribution to the development of software architecture is the patterns codification which can be used as the blueprint of components, constraints, and their relations. Patterns define the general solutions that can be reused to accelerate the software development process. Some methods to apply the operational patterns for the EIS design are as follows.

General purpose software packages encapsulate data structures and algorithms to implement a generic but customizable solution of business problems based on the best practices. Market-leading providers include SAP AG, Oracle Corporation, and Baan Co. In these packages, many operational patterns are exploited: database-centered data sharing, pipeline-based data processing, event driven message invocation, to name a few. The packaged software tools have been adopted by a variety of enterprises to optimize their business processes.

Domain-specific software architecture is tailored to EISs in a specified domain. Such architecture includes some special components which differ from common components of generic software architecture. For example, an industry-oriented ERP can accommodate the requirements especially for a certain industry domain, some insignificant software elements and tools included in generic software package can be removed to reduce the complexity.

Distributed computing involves several interacting elements coordinated to achieve a system-level goal. Distributed programming typically falls into one of the following architectural options: client-server, n-tier architecture, and peer-to-peer. Programming languages with parallel and concurrency supports, and middleware technologies are among the key enablers for distributed computing. Agent is an autonomous entity situated in the environment, multi agent system is composed of a group of agents, the agents within a MAS can cooperate or compete each other to achieve the goals at the system level. MASs have been successfully applied in manufacturing, workflow management etc.

Service-oriented architecture (SOA) has been a recent advance in integrating heterogeneous platforms such as legacy software tools. A SOA allows an EIS to extend its capabilities by applying reusable software modules so that the development cost can be reduced without reinventing a wheel. When equipped with methods like Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), and Universal Discovery, Description, and Integration (UDDI), the SOA has become an important technology to EISs.

### Scenario-Based Software Architecture Analysis

The user-oriented method facilitates the decision-makings related to EIS software architecture. The evaluation in this kind of method is performed based on the intended business and quality attribute goals. This is because the way software architecture supports the driving NFRs determines how an EIS will behave. The performance of an EIS in turn will shape the business strategies and technical capabilities of enterprises.

There are huge benefits of using non-functional requirements to build an EIS. However, there are also many unsolved practical issues when NFRs are considered in the implementation of EISs. For example, NFRs are usually subjective and hard to be quantified. This calls for qualitative methods to reason how well the EIS software can meet the NFRs. Moreover, users express their missions with different terminologies, even though there are several standards related to NFRs. EIS must balance a set of the conflict objectives to determine its software architecture. Some examples of the conflict objectives are flexibility vs. productivity, scalability vs. reliability. Moreover, all of these objectives contribute to the cost factor. It is necessary for industrial information engineers to consider all of the objectives simultaneously at the system level.

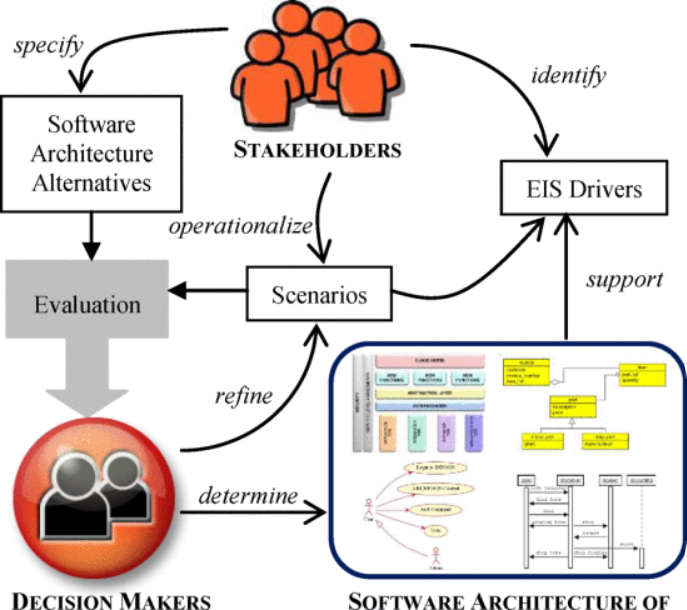


Figure 1: Framework for scenario-based EIS software architecture analysis

Above figure represents a scenario which explains the practical challenges in evaluating software architecture based on the given NFRs, here the boxes and arrows represent entities and activities respectively. The core component called evaluation is highlighted in a shaded callout. As depicted in this figure, the scenarios play two important roles in the evaluation. Firstly, they allow the abstract NFRs to be concretely defined, operationally measured, and meaningfully communicated among the stakeholders. Secondly, they link architecture choices to the satisfaction of the EIS drivers, which helps the management to make an informed decision about the system that is best tailored to meet their requirements.

### Eliciting Scenarios

A scenario is an abstracted description of system to be designed, both the user and designer's perspectives must be considered. Scenarios play an important role in requirements elicitation and analysis. Scenarios are frequently used in the system development process, for example, use case scenarios are a part of the rational unified process and the primary sources of the definition of requirements in the agile software development.

When dealing with subjective concepts like NFRs, scenarios can be used to evaluate if a set of subjective attributes can be satisfied by software architecture. For this reason, the quality attributes scenarios are created to evaluate the interactions of system from the perspectives of the subjective attributes. In contrast to the terminology scenarios used by others, the scenario here must relate to subjective attributes.

In other words, NFRs have to be defined clearly in a scenario. For example, a statement that says that “a system is flexible' is invalid since it is vague and meaningless. All systems are meant to be flexible and should accommodate certain type of changes. On the contrary, the following statement might be valid for a scenario:

“A user expects to insert an editable field for searching and add an active in the graphical user interface, the icons in the toolbar must be scaled, and the changes should be completed within 3 hours, these changes address the issues 4 and 12 raised in the bug report so that usability will be improved.”

The above scenario makes NFRs measurable and help resolve terminological ambiguities by capturing the stakeholders' precise concerns and thus the inclusion of agile development and user stories makes it more beneficial in building a strong EIS as agile suggests considering the efforts, requirements, and business values that are brought aboard by the user stories.

There are other factors be considered. For example, a scenario should associate the tasks with the roles of participators. In this way, system usages can be evaluated from multiple perspectives.

### Evaluating System Architecture

Trade-off on the conflict objectives must be made in an engineering situation that involves competing contingencies. Trade-off on software architecture is about how to make the decisions with a full comprehension of both the upside and downside of a particular choice. We examine two software architectures as an example:

* The first example is database-centered architecture presented in Fig. 2 below, where the design considerations are arranged in a grid. The horizontal axis depicts the types of conceptual features involved in the application such as Entity, Activity, and Constraint. The vertical axis describes the features of the organizations in a semantic model: Classification, Aggregation, and Specialization. The other narrative contents in the plot have shown how the objectives of the application are translated and related to the system components. The architecture, as shown in Fig. 2, is a layered structure where the information flow occurs only in two adjacent layers. This modular property of information hiding, together with the centralized data model, makes the database-centered architecture a suitable support for the caching organizational workload requirement.

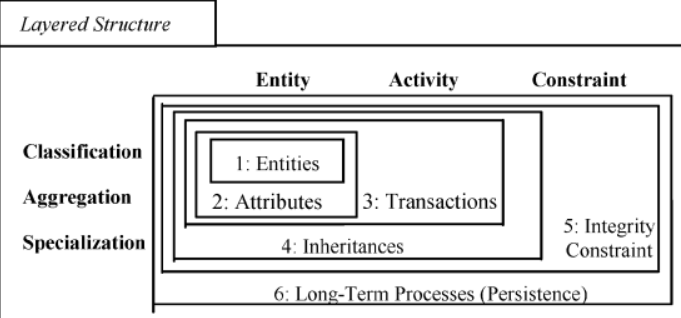


Figure 2: Database-centered layered architecture

* The second option of software architecture for is a distributed service-oriented infrastructure that exploits the Web services. The interactions amongst different systems/machines are supported by a web service, which is WSDL-based machine-processable interface. As a result, the Web services architecture is particularly suited for supporting the remote communication.

Below figure depicts the components involved in a web service architecture:

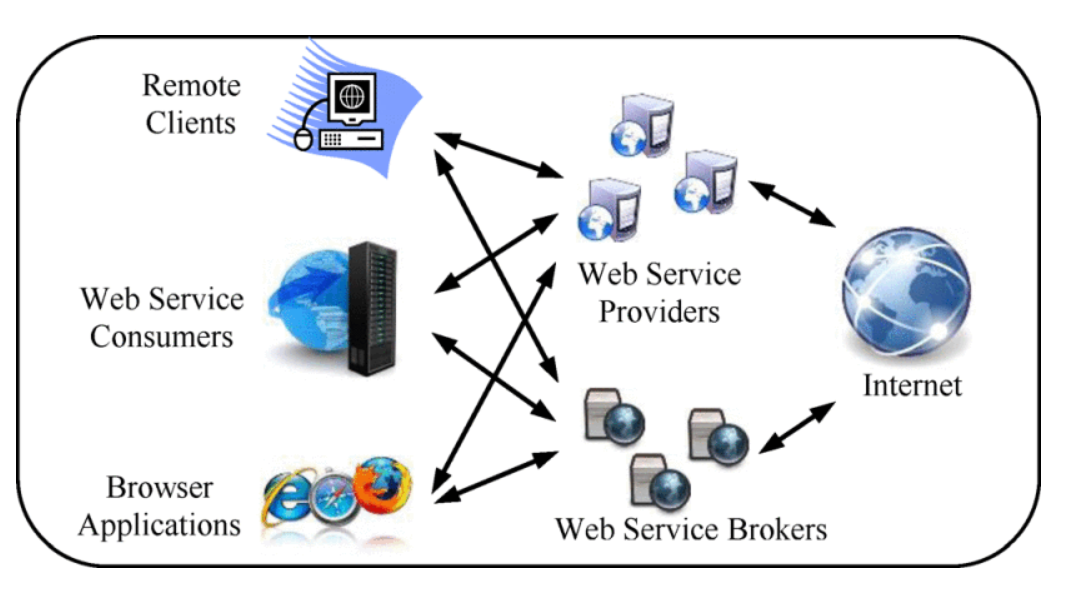


Figure 3: Web services architecture

In the above section we explained how different scenarios are explicitly supported by certain design alternative and the software architecture options can be evaluated accordingly as per these scenarios. The main idea here is to use the scenarios to connect software architecture with the driving NFRs and to propagate the contribution relations through qualitative design reasoning. For reasoning, the evaluation is focused on finding a solution that is sufficiently good rather than fully optimal in all aspects. The underlying rationale of reasoning in such a matter is that the satisfaction of NFRs is not a straight true or false answer.

Effective design analysis depends on the ability to aggregate and present the relevant information in an insightful way to help decision makers to find the right balance among the NFRs. One can use an intuitive and integrated graphical representation to serve this purpose.

**Order management process in coca cola**

***Brief explanation & challenges***

* In 2014, Coca-Cola had refranchised and due to which the sales increased a lot and expanded into different new areas.
* They introduced Coca-Cola freestyle which was a self-serve dispenser which was deployed to fast food stores and other suppliers, it offers more than 100 products of coca cola for example sprite, bottled water etc. But this complicated the distribution system as “Coca-Cola innovation collided with their supply chain system”.
* Below figure shows the order management process used by Coca-Cola:

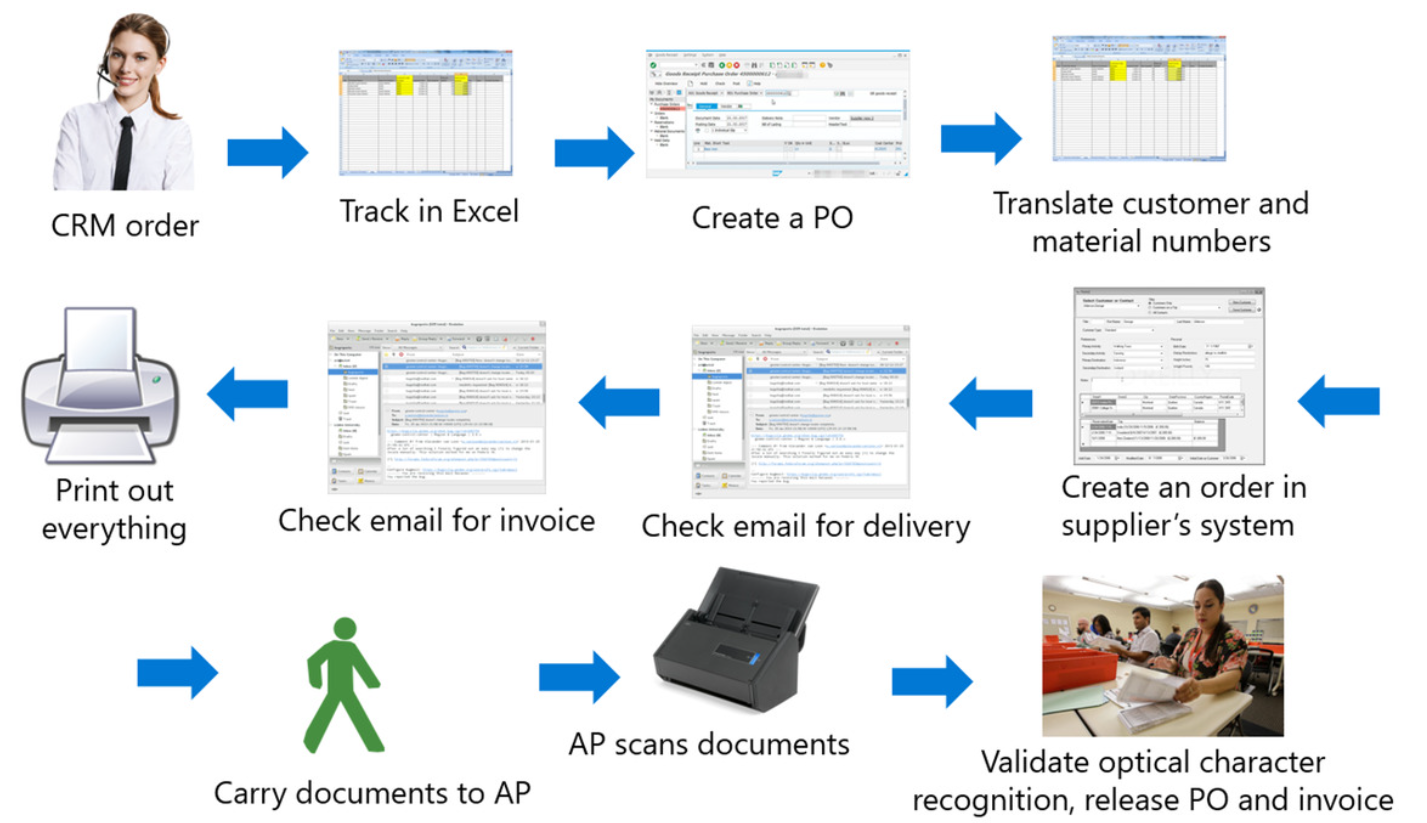


Figure 4: Manual order management process before implementing RPA

* The order management process consisted of 11 steps, at first a CRM order is received which Coca-Cola stored in excel sheets like the Customer and material numbers. PO was created manually after agent take order and push manually through CRM system, then scanned the invoice and walked to accounts payable. Internal numbers were kept in SAP system by Coca-Cola. Delivery and invoice emails were tracked manually by email.
* A single CRM agent can handle only 50 Freestyle orders, and nearly 50,000 more used to be in waiting. These problems were solved by Coca-Cola by implementing RPA with Microsoft Power Automate.

**New order management process**

***Technology & Tools***

Below are the list of Technologies/Platforms/Processes that were used by Coca-Cola while adopting to the new automated process of order management:

* CRM
* RPA
* Microsoft Power Platform
* Azure Virtual Machines
* Azure Blob storage
* Azure key vault
* AI Builder: Microsoft Power Automate (part of Microsoft Power Platform)
* Azure DevOps (Power Platform Build Tools for Azure DevOps)

Below table shows the old system and their corresponding new system:

|  |  |
| --- | --- |
| ***Old System*** | ***New System*** |
| Excel Spreadsheet (customer orders) | Microsoft Azure SQL |
| Creation of Purchase Order in SAP involved human intervention | ‘Asa’ RPA Bot (Automated Service Agent) |
| Extracting customer/material number into spreadsheet with human intervention | Automated Service Agent |
| Manual check of delivery and invoice emails | Automated Service Agent |
| Manual creation of order for the supplier | Automated Service Agent |
| Manual intervention for documents handling of Accounts Payable | Automated Service Agent |

*Microsoft Power Automate* helps organizations build multi-stage workflows which automate repetitive business processes to help employees save time and increase productivity. With template workflows available for many common processes, the capability to create custom workflows from scratch, and integration with hundreds of systems and applications, Power Automate offers the flexibility to meet a diverse range of needs. Like the rest of Power Platform, all of this can be accomplished via intuitive, user-friendly interfaces which do not require technical knowledge to use, allowing your organization to save time, increase productivity, and reduce the potential for human error. Though basic Power Automate functionality is accessible to individuals without development expertise, engaging a trusted partner to help you develop your workflows enables organizations to leverage advanced functionality involving complex, multi-system workflows and fully custom flow parts. Data prise DevOps offers the knowledge and experience you need to unlock the full potential Power Automate has to offer and securely automate your business processes at scale. (Power, B.I., 2020)

Below are a series of figures which show how to work with power automate and create a workflow that can run automatically and perform the actions as defined:

Graphical user interface, application, Teams

Description automatically generated

Figure 5: A glimpse of Microsoft power apps – that offers chatbots, flows (power automate), dataverse

A screenshot of a computer

Description automatically generated

Figure 6: Microsoft Flows/Power Automate – offers built in templates to connect with external systems such as JIRA, CRM etc., one can click on new flow to create a custom flow

Graphical user interface, application

Description automatically generated

Figure 7: Shows different scenarios to trigger the flow, that can be selected by workflow builders

Graphical user interface, application

Description automatically generated

Figure 8: shows how to create a sample workflow using power automate

Graphical user interface, text, application

Description automatically generated

Figure 9: displays several criteria’s/scenarios/connectors that can be configured in power automate workflow

*Power Virtual Agent* is a paralleloffering to Microsoft Power Platform and is used to create intelligent conversational bots to provide business outcomes. For example, one can create powerful AI-powered chatbots for range of requests from providing simple answers to common questions to resolving issues requiring complex conversations

***Architecture***

Microsoft Power Platform is a complex architecture and can be hardly depicted on a single image. However, we have listed some of the essential components below that help us understand how power platform works:

* It is built on top of Microsoft Azure infrastructure, it uses various Azure services, including Azure SQL, Azure Virtual Machines, and Azure Cache for Redis etc. Microsoft Power Platform offers more than 400 connectors to connect to various services of other third-party systems such as SharePoint, Office 365, Salesforce, Dynamics 365, OneDrive, Azure etc.

A picture containing diagram

Description automatically generated

Figure 10: Microsoft power platform landscape diagram

* Microsoft Power Platform uses Azure Content Delivery Network (CDN) to optimize performance and user experiences when using static content such as images and icons.
* Microsoft Power Automate uses Azure Relay to establish connectivity between Power Automate cloud flows and desktop flows in Power Automate for Desktop. It uses HTTPS and data is encrypted during its transmission between the computer and the cloud.
* Using an [on-premises data gateway](https://learn.microsoft.com/en-us/data-integration/gateway/service-gateway-onprem) enables us to connect to on-premises services with Microsoft Power Platform in a very secure manner and uses Azure behind the scenes. Any data transmitted via the on-premises data gateway is sent via Azure Service Bus.

***Business Improvements/Advantages***

* The company faced two challenges as it explored solutions. “It was too costly to create APIs for our legacy applications like SAP and others, and we also had to navigate a third-party website,” explains Allan McDaniel, Manager of Development for BI and Master Data at Coca-Cola Bottling Company United. “We used Power Automate RPA bots to fill these gaps and automate the process.” Desktop flows in Power Automate repetitive processes in Windows and web applications—a perfect fit for high-volume but mundane data entry and transfer.
* The solution was built by the company’s fusion teams of citizen developers and professional developers and its partner, Happiest Minds Technologies. Coca-Cola United started by synchronizing data between the company’s SAP CRM system and Azure SQL using Azure Data Factory. Happiest Minds added the automation, creating a master automated service agent they’ve dubbed “Asa.” Developed on Microsoft Azure and Microsoft Power Platform, Asa consists of several bots and uses Azure Key Vault to help secure and control passwords and other sensitive data.
* Power Platform Build Tools for Azure DevOps were used to co-ordinate work between Coca-Cola United and Happiest Minds, making it easier to drive continuous improvement and development across the project. These tools also enabled automation of common build and deployment tasks. “Power Platform and Azure DevOps enabled our citizen developers, pro developers and partner to build as a team – and that accelerated the entire development process,” says Means.
* Now, when a CRM agent enters an order into the CRM system, Asa takes it from there and signs into the company’s SAP system without human intervention. Asa easily accesses orders, which are now tracked in a [Microsoft Azure SQL database](https://protect-us.mimecast.com/s/gTGOC9r70Ys3rZxCEANfU) rather than in an Excel spreadsheet. Asa reads the database and creates a PO in the company’s SAP system. Asa then submits the order to the supplier’s web application, validates successful entry, monitors the email system for invoice and delivery emails, matches them to the correct order, and then stores the attachments in [Azure Blob Storage](https://azure.microsoft.com/en-us/services/storage/blobs/) for future reference. After that, Asa uses form processing in [AI Builder](https://flow.microsoft.com/en-us/ai-builder/) to extract information from those email attachments that’s necessary to close the process in the Accounts Payable system, and finally, it releases the invoice and PO from SAP. These steps occur with bots within the Asa bot running in unattended mode, a key capability of Power Automate RPA that takes mundane, tedious tasks from humans and shifts them to bots. With RPA unattended mode, everything is fully automated. After deploying the unattended mode in [Azure Virtual Machines](https://azure.microsoft.com/en-us/services/virtual-machines/), Coca-Cola United can now schedule and trigger events that increase end-to-end automation of high-volume tasks—like its suddenly expanded orders for Freestyle. And because Asa is cloud-based, it can automatically scale to any job and interoperate with any application. Just as importantly, Power Platform and Azure provide a rich set of monitoring and alerting capabilities that facilitate debugging.
* The new, simplified process frees the dedicated CRM agent, allowing orders from all channels, such as inbound and outbound call centre agents, field service sales representatives at customer sites, and via a customer self-service portal.

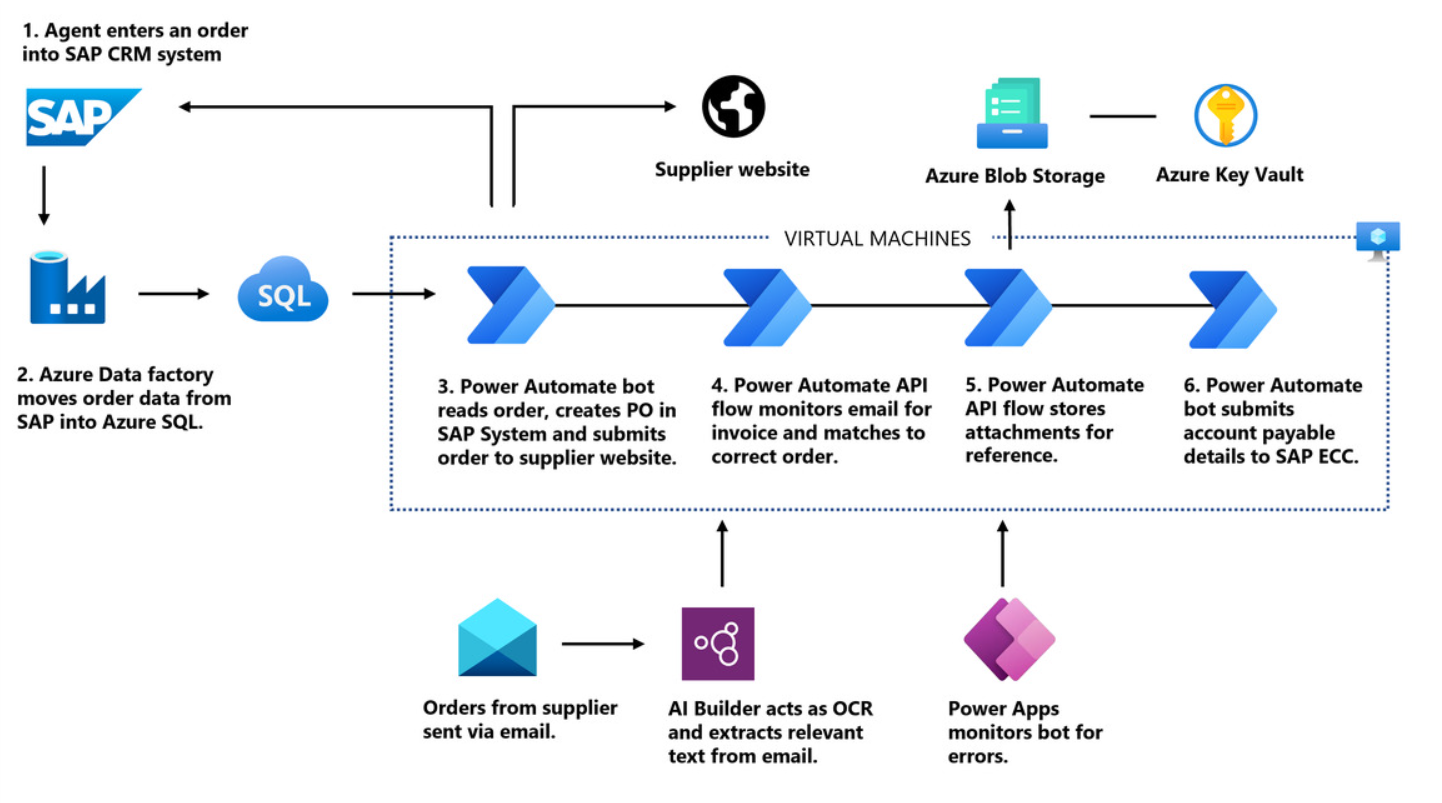


Figure 11: New automated process of order management system using Microsoft Power Platform

* By removing the busywork of order and invoice processing, Coca-Cola United vaulted over the limitations it faced prior to creating its Asa bot.” We avoided having to hire 10 full-time employees,” says McDaniel. “Better still, the existing CRM agent is now free to work on other projects.” Kaylan Cannon, the company’s Customer Service Manager, is enthusiastic. “We are very excited about this solution,” she says. “It will dramatically reduce labour costs, minimize the various points of error in our current solution, and will allow us to rapidly expand the local Freestyle campaign to better support our customers.”
* For McDaniel’s team, Azure DevOps interoperability, with rich enterprise-ready security controls, is the gift that keeps on giving. “The low-code to no-code Power Automate RPA platform with built-in connectors significantly reduces development time,” says McDaniel. “The way that Azure DevOps interoperates with the solution plus the built-in administrative and governance controls in Power Platform is an important feature for us,” he adds. “We connect it not only with the Power Automate solution but also any other infrastructure needed. Once built, we can use a desktop flow in any other Power Automate flows, extending its value to solve other business challenges with legacy applications with little to no further development.” And Azure DevOps boosts collaboration with other bottling companies. “We can easily share this work with other organizations because Power Automate works with Azure DevOps.”
* Inspired by the productivity boost they achieved with Power Automate RPA, Means and his team are exploring other parts of the business platform, including [Microsoft Dynamics 365](https://dynamics.microsoft.com/en-us/). Azure DevOps and reusability and governance controls will make it easier to share solutions with other bottlers, but for now, Coca-Cola United is buoyed by the joint effort between itself, Microsoft, and Happiest Minds that resulted in an optimal solution, and the improvements continue. “While building this solution, we resurrected high-value strategic projects that we couldn’t tackle before because of the constraints of legacy apps,” says Means. “We feel empowered to take advantage of any future opportunities that the business provides us.”

**Recommendations**

1. ***Maintaining equilibrium between People, Process and Technology (Shastri):***

***Challenge:***

Although low code no code platforms like Microsoft Power Platform a wide range of connectors and are easy to use, for this reason the developers working on this platform are also referred within the enterprises as ‘*Citizen Developers*’. It brings in a challenging situation where citizen developers require development experience if the EIS requires development of components that require custom coding in some of the programming languages such as C#, React or Angular.

This might result an imbalance between the aspirations, expectations of citizen developers and the enterprise expectation in getting products developed at a faster pace.

***Suggestions:***

We suggest that enterprise such as Coca-Cola United in our scenario should have a development team with good mix of management, experienced developers, and citizen developers and constant training, development workshops must be organized regularly so that resources within team keep up with the learning curve.

1. ***Process Improvement (Sharon)***

***Challenge***:

In the current case study, Coca-Cola automated the same old process, with some changes which were not so significant.

***Suggestions***:

Whenever enterprises move from old system to a new system, they should work closely with business, vendors and all the stakeholders and look at the opportunities to change/improvise the system, particularly like the order management process in our scenario where many manual steps were automated using Microsoft Virtual Agent bots and Microsoft Power Automated based workflows, Coca-Cola leadership should encourage the teams involved on ground in such kind of projects so that they spend good amount of time in eliminating some of the manual steps or revamp and bring in some new changes to the system as this was a very good opportunity to do so.

1. ***Evaluating Competitors (Akhil):***

There are lot of other RPA tools in the market, which Coca-Cola can check and use

**References**

* Niu, N., Da Xu, L. and Bi, Z., 2013. Enterprise information systems architecture—Analysis and evaluation. *IEEE Transactions on Industrial Informatics*, *9*(4), pp.2147-2154.
* Power, B.I., 2020. Microsoft Power Platform.
* <https://learn.microsoft.com/en-us/power-platform/guidance/expressroute/understanding-architecture>
* <https://info.microsoft.com/ww-Landing-Coca-ColasJourneywithRPA.html?LCID=EN-US>
* <https://customers.microsoft.com/en-us/story/845187-coca-cola-bottling-company-united-consumer-goods-power-automate>