JOURNAL OF APPLIED INTELLIGENT SYSTEMS & INFORMATION SCIENCES
VOL. 1, ISSUE 2, PP. 70-90, AUGUST 2020.
Available at: www.journal.research.fanap.com
DOI: 10.22034/JAISIS.2020.103704

ERP SYSTEMS ARCHITECTURE FOR THE MODERN AGE: A REVIEW OF THE STATE OF THE ART TECHNOLOGIES

Mohammad Amini Valashani[1,*], Arnold Mashud Abukari[2]
[1] *Research Center, FANAP Co., Tehran, Iran*; *m.amini @fanap.ir*
[2] *Computer Science Department, Tamale Technical University, Ghana.*

ABSTRACT

An Enterprise Resource Planning (ERP) system is a vital component of any successful organization for managing complex business processes. In order to successfully implement ERP solutions, the management and implementers should have a broad understanding of the ERP systems architecture and the specific components which might be required for any business need. In this article, we provide a comprehensive overview of the modern ERP systems architecture and investigate their underlying conceptual and structural models. In addition, we present the most significant technologies and tools used for the modern ERP systems which are provided by leader vendors in the market. Our research work can provide an insight for ERP project managers and consultants on what type of architecture is suitable for their business needs or how the implemented ERP system components work together. Moreover, it can assist ERP software designers to find the right type of software architecture which might satisfy any customer's changeable needs.

KEYWORDS: ERP Systems, ERP Architecture, Three-tier Architecture, Service Oriented Architecture, Cloud Architecture.

1. INTRODUCTION

In today's competitive and ever-changing business world, the success of any organization highly depends on how it implements and uses enterprise systems. The critical role of enterprise systems is undeniable since they have a significant impact on the organization's value chain in both primary and secondary activities. Many organizations in different industries use enterprise resource planning (ERP) systems which can integrate data across all functional units and comprehensively support major processes throughout an organization (Ganesh et al., 2014). An ERP system has a central database for the whole information flow in the organization reducing data redundancy and increasing flexibility. With the advent of web-based technologies, ERP systems could integrate the information within the organization as well as with its external partners, clients and suppliers.

The implementation of an ERP system in an organization enables all departments and functions across the organization to be integrated into one single platform that serves the needs of any particular functional area. As a result, all software applications allocated for different departments and units in the organizations can integrate their functions into a central and comprehensive software connected to a single database. The ERP comprehensive solutions will provide a common platform for the departments in the organizations as well as different functional departments located in partners', customers' and suppliers' sites to communicate effectively and share information as needed.

ERP systems can be considered as mission-critical systems for organizations. Thus, implementing and managing these systems requires good knowledge of their architecture and understanding how their components

[*] *Corresponding Author*

RECEIVED: 23 MAY, 2020; **ACCEPTED**: 29 JULY, 2020; **PUBLISHED ONLINE**: 29 JULY, 2020

interoperate with each other. Most managerial aspects such as cost, maintenance complexity, and ease of use are influenced by how an ERP architecture is implemented. The architecture defines the relationships among the complex information technology components, including hardware, software, and data with complicated organization components as company structures, business rules and people. In addition, understanding the enterprise systems architecture gives the management a clarified vision on the system infrastructure requirements, training requirements, change management requirements, and business process reengineering requirements, etc. so that they can develop a better information technology (IT) plan. Therefore, it is imperative for any project manager or consultant specializing in ERP systems setup, to gain sufficient and accurate knowledge about the technical architecture of the ERP systems in the modern age.

There are some research works proposing a specific type of architecture for ERP systems. They include simple architectures such as on-premise two-tier architecture Hall & Hall (1994) up to modern web-based (Sandoe & Boykin, 2007), SOA-based (Erdiyana, 2019) or cloud-based architectures (Pareek, 2014; Lee &Wang, 2019). To the best of our knowledge, no specific research has ever been devoted to studying all types of architecture for ERP systems and providing an insightful perspective on how ERP systems architecture has evolved in the market. In this paper, we aim to provide a comprehensive overview of the modern ERP systems architecture and investigate their underlying conceptual and logical models. Moreover, we present different technologies used for the modern ERP systems by leader vendors in the market. Our research work can provide an insight for ERP project managers, consultants on what type of architecture is suitable for their business needs or how the implemented ERP system components work together. Finally, it can assist ERP software designers to find the right type of software architecture which might satisfy any customer's changeable needs.

The rest of this article is organized as follows: In Section 2, we provide an overview of ERP systems and their role in today's organizations. Section 3 includes a well-formed review of the ERP systems architecture and the most common design models. The most innovative technologies applied for the architecture of ERP systems are presented in section 4. Finally, in Section 5 we conclude the paper.

2. ERP Systems Background

2.1. ERP systems overview

An ERP system is a packaged business software system that allows a company to "automate and integrate the majority of its business processes; share common data and practices across the enterprise; and produce and access information in a real-time environment." (Sumner, 2014). ERP systems tie together a multitude of business processes and enable the flow of data between them. By collecting an organization's shared transactional data from multiple sources, ERP systems eliminate data duplication and provide data integrity with a single source of truth (Oracle, 2020a).

To be competitive, organizations must improve their business practices and share information with their suppliers, distributors, and customers. ERP systems are different from legacy systems in that organizations use ERP to integrate enterprise-wide information supporting many business functions, including financials, human resources, manufacturing, logistics, and sales and marketing, etc. (Seddon et al., 2003). An ERP system also provides a central enterprise database by which all business transactions can be recorded, processed, monitored, and reported. The usage of a central enterprise database forms the foundation for creating a unified view of the whole organization.

One objective of implementing an ERP system is to apply an efficient information system by integrating and automating business process activities to make them faster and transparent. The degree of integration decides simultaneous dissemination of information with different stakeholders and helps improve the performance of the organization. The significance of an ERP system is that it can integrate information throughout the supply chain leading to cost reduction, inventory reduction, and improved operating performance (Sumner, 2014). A nice integration also becomes the source of competitive strength and can be a differentiating factor with respect to its competition degree. The scope of the ERP determines the extent to which we automate different business processes (Ganesh et al., 2014).

ERP systems can bring many benefits to the organizations regarding system and business perspectives. From the system perspective, they can help maximize the information throughput, minimize response time to customers and suppliers, and provide timely information to decision makers (Sumner, 2014). ERP systems eliminate the need for legacy systems that maintain incompatible data and cause fragmentation. The consistent information provided by integrated systems can be shared effectively across the organization and be used to monitor business performance (Davenport, 2000; Muscatello & Chen, 2008). From the business perspective, ERP systems are able to augment and optimize the operational performance throughout the business, from customer order through distribution and service. ERP as an integrated information system presents a better interaction between interrelated functions because of Business Process (BP) automation (Shirazi, 2019). The inventory levels and procurement costs are reduced, on-time delivery is improved and manufacturing throughput is increased (Davenport, 2000; Sumner, 2014). ERP can increase the efficiency of business processes, reduce cycle time and improve decision making through automation of business functions (Piturro, 1999; Ganesh et al., 2014).

ERP systems can provide support for organization’s business processes through separated modules. Each module is an enterprise software application built specifically for each functional area of business operations. Most important business modules provided by well-known ERP vendors in the market include Production Module, Purchasing Module, Supply Chain Management Module, Financial Module, Sales & Marketing Module, Human Resource Module, and even other complementary modules such as e-commerce, self-service and project management modules (Sumner, 2014; Monk et al., 2012). ERP software modules include the best business practices for the functional areas in order to allow organizations to implement their policies and procedures.

There are many ERP systems vendors in the market which provide ERP systems with various modules for organizations with different sizes from large enterprises to small companies. The most widely used ERP products are SAP products (SAP, 2020), Microsoft Dynamics family (Microsoft, 2019), Oracle ERP products such as NetSuite (Oracle Netsuite, 2020) and People Soft (Oracle, 2020b) as well as Sage ERP solutions (Sage, 2020). While these vendors offer many ERP solutions each with their own requirements and features, the strength of each vendor is different regarding each ERP functional module and for supporting organizations with different sizes (Monk et al., 2012; Nah, 2001). Thus, one of the big challenges of the management in an organization is to decide which vendor to choose for buying an ERP product and which modules services of a particular ERP solution are suitable for their business needs.

2.2. *The significance of ERP architecture*

Every type of enterprise system needs an architecture to reflect the blueprint of its software and hardware design. For designing and implementing an ERP system we need a blueprint of the actual implemented system which determines the high-level ERP implementation strategy and depicts the information flow within the organization’s subsystems and their interrelationships. The ERP implementation team can best build the ERP system for an organization when a flexible and well-designed architecture is defined for the system. The architecture of the ERP system shows how different components of the system are combined and interact with each other in order to provide the easiest and most efficient information flow in the organization. ERP architecture defines the relationships among the complex information technology components, including hardware, software, and data with complicated organization components as company structures, business rules, and people. Understanding the ERP systems architecture provides a well-defined vision for the management about the system infrastructure requirements, training requirements, change management requirements, and business process reengineering requirements. Thus, they will be able to develop a better IT plan for ERP deployment.

The ERP systems are often designed by ERP vendors with a package-driven architecture (Motiwalla & Thomson, 2012). It means most ERP solution vendors provide their built systems as a software package in order to satisfy business needs of the customer organizations. Therefore, the architecture of these systems is driven by the vendor design strategy and not by the customer organizational strategy and business process requirements. Most vendors in the competitive ERP market try to use best practices of the industry’s business processes in their system logic. This type of architecture is useful for the organization that wants to purchase a full-fledged software package ready to work with the industry's standards. As a result, companies need to implement ERP software according to the requirements contained in the package so that they can obtain the best possible performance and maximize their return on investment (ROI).

The organizations who want to implement a complete ERP product bought from a vendor with a package-driven architecture should not ignore their organizations architecture and rush into a "vanilla" or "as-is" implementation (Ganesh et al., 2014; Monk et al., 2012). They should plan accurately for the long-term maintenance and subsistence of the system. The ERP system maintenance, the required modifications, and customizations in the system are significantly influenced by the ERP system architecture and can have a great impact on how to support the organization's policies and procedure, data transformation, upgrades, backups, security and controls. Therefore, it is imperative for the ERP designers and implementers to conceive the ERP architecture after they select and buy software from any vendor.

There are also companies and organizations whose business requirements or organizational structure do not completely match the available ERP products in the market. In addition, they may not want to buy an out-of-the-box ERP system for specific reasons such as implementing and change management costs, high customization needs, integration challenges, or data sensitivity. Therefore, it may become inevitable for these organizations to start building their proprietary software in order to be able to respond to their particular business needs and be able to scale their system effectively. Designing and building a proprietary ERP solution from scratch is a big challenge for organizations and needs intensive planning and analysis phases. Defining and specifying a proper ERP architecture according to special business needs, the organizational structure, and available technology and skills for the ERP development is highly important for the success of the custom ERP software solution.

3. Research Methodology

In order to conduct a comprehensive study about the architecture of ERP systems, a wide range of articles related to ERP architecture were reviewed. Given that the subject of our research was the study of architecture in terms of design models as well as technologies used, we needed to gather both academic and industrial resources to select informative articles.

For academic articles, we searched through the most important academic research databases which usually publish articles in the fields of engineering, technology, and management. The selected academic research databases were Scopus, Web of Science, Science Direct, IEEE Xplore, and Springer. The keywords used for searching included *ERP architecture*, *ERP architecture models*, *enterprise systems*, and those keywords related to architecture such as *tired architecture*, *logical* and *physical architecture*.

ERP systems play a significant role in today's industry and businesses. Technology advancements have made great changes in these systems over the years. Meanwhile, large companies have moved to the path of design and production of ERP systems as a promising and profitable software product and have made a name for themselves in the market. One cannot avoid searching the resources of these companies as pioneers of the ERP products market to gain knowledge on the ERP systems, specifically their architecture. Therefore, we performed a full research on the websites and documentations of the most prevalent ERP software products vendors, such as SAP, Oracle, Microsoft, Sage, Epicor, and so on. We also went through many white papers and blog posts in the industry-focused website to find informative material about ERP architecture styles, methods and technologies.

To sum up, we gathered and studied 65 articles about ERP systems architecture and their related technologies. Moreover, several books about ERP systems were selected and studied. We analysed the resources according to their canonical concepts and decided that two important subjects are of high importance in ERP systems: architecture styles and architecture technology. The ERP systems architecture are analysed from these two perspectives in the next sections.

4. Architecture Styles

Generally, the architecture of the ERP systems can be defined in two types: logical Architecture and Physical (or tiered) architecture (Markus et al., 2000). The logical architecture, as depicted in Fig. 1, demonstrates how the system is organized to support the functional business requirements and related end users. The hardware infrastructure and database systems are located at the lowest layers followed by the core business logic which encodes the real-world business rules and constraints in the next layer. The fourth layer provides the details of functional business applications that are built into ERP system. Between the end users and business applications lies the client user interface. It is responsible for all user interactions with the system. Therefore, the implementation of other layers is hidden for the end user.

On the other hand, the physical architecture depicted in Fig. 2 shows the composition of different physical components of the system in a way that the whole system can deliver the best performance and reduce costs.

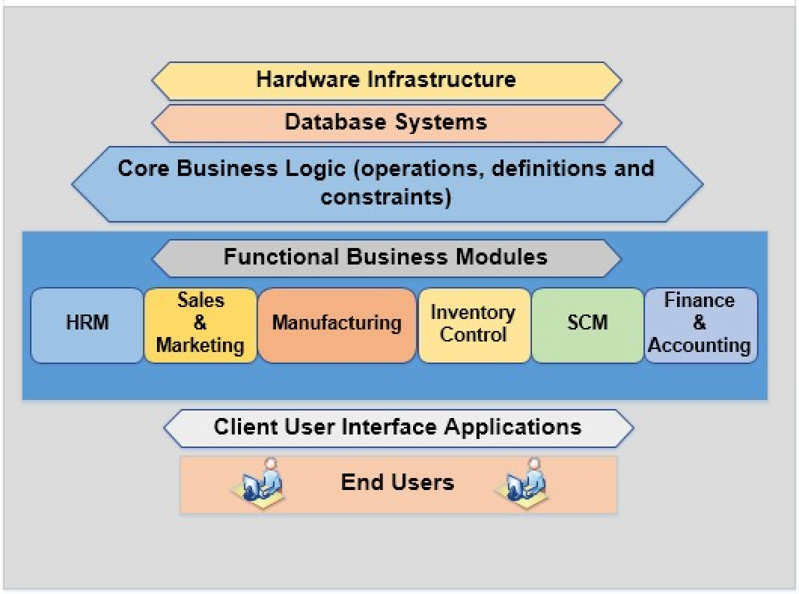
Hardware Infrastructure
Database Systems
Core Business Logic (operations, definitions and constraints)
Functional Business Modules
HRM
Sales & Marketing
Manufacturing
Inventory Control
SCM
Finance & Accounting
Client User Interface Applications
End Users

Fig. 1. Logical architecture of an ERP system

4.1. *The tiered architecture*

The physical architecture of the ERP systems is usually designed in a layered style based on the client-server model also called Multi-tier or N-tier architecture (Buschmann et al. 1996, Jour & Zhaolin, 2013). In this model, an ERP system component are organized in layers or tiers so that the complexity of the system functionality can be better managed and the system can support scalability and flexibility for the organization business needs. The tiered architecture also enables the designers to separate the resources into multiple layers providing security and reducing the risk of system failure.

Traditionally ERP architecture has followed the same pattern as the software architecture in the last decade. In the early days of ERP systems, they were built using a two-tier architecture in the form of client-server interaction (Hall & Hall, 1994). In this type of architecture, the ERP systems components are grouped into two layers: servers and clients (Fig.2-a). The servers usually are responsible for handling database requests and the clients include the applications which request the data from the servers and use it for different functional module. This model can also implement with *thin clients* where the servers take care of both database and application tasks and the clients are only responsible for providing a user (graphical) interface to the end user for interaction (Wortmann, 1998).

The benefits of the two-tier architecture include ease of access to the information, low cost for the requirements of infrastructure, and the high performance for a limited number of work stations. However, this model is not flexible enough for adding more clients and software and expensive middleware are needed for integration of the system with other components. Furthermore, any change or modification in the database has an impact on applications (or clients) (Motiwalla & Thomson, 2012). Another drawback is that the implementation of business function modules depends a lot on the database servers or client applications implementation (Ganesh et al., 2014).

The pressure for adding more efficient and effective functionality while providing ease of access for more end users in the enterprises led the ERP vendors to move into the more efficient three-tier architecture (Fowler 2002, Rashid et al. 2002). It is the most reliable, flexible, and scalable architecture. The three tiers in this architecture are called data tier, business logic (or application) tier and presentation (or web) tier (Fig.2-b). The descriptions of the tiers are presented as follows:

- The data tier includes data repositories and database servers that business functions can use to store and retrieve their required information. The structure of all organizational data and its relationships with both internal and external systems are defined in this tier. Usually, the data is stored in relational databases capable of handling SQL queries. As an example, we could say that the sales order, items, delivery, and remarks on an order are linked together in this layer.
- The application tier implements the business logic of the system functional modules and includes the servers and mechanisms to provide the required functions with the best performance. In fact, this tier implements business process rules in the form of reusable objects that can be reassembled into many functional applications (developer, 2004). The servers in the application layer handle users' request, retrieves data from the database tier, and processes data as needed.
- The presentation tier is responsible for delivering the information with appropriate formats and provide integration tools in order to facilitate user interactions from the clients. This layer provides flexible GUI with which the users can interact with the components of application tier and access their required services. The reason this layer is also called web tier is that it can provide it services through web server technologies. The end users can access the business functional modules anywhere from outside of the organization using various web client programs.

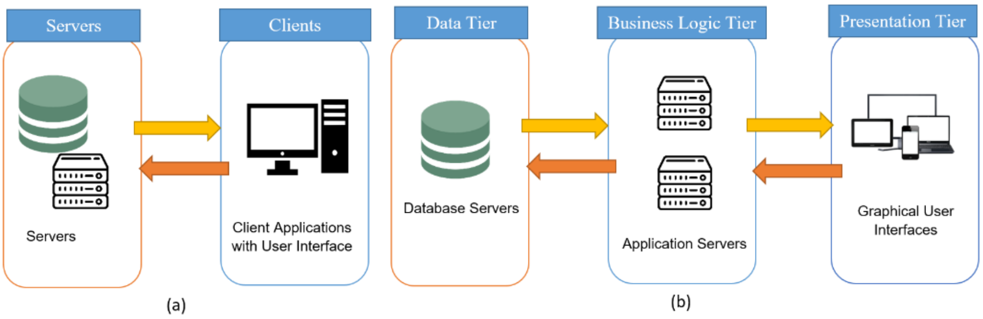

Fig. 2. The physical architecture of ERP systems: (a) Two-tier ERP architecture, (b) Three-tier ERP architecture

There are many advantages in applying the three-tier architecture of ERP systems for the organizations over the traditional client-server architecture. The three-tier architecture can provide scalability, reliability, flexibility, maintainability, reusability and security for the organizational applications. Another benefit brought by this architecture is that in the case the management decides to outsource maintenance of the ERP, the transition becomes easier and more controllable. However, this architecture implies its own challenges for the implementers. For example, more hardware and software resources are needed to support the functions of the middle layer and the complexity of development environments is increased dramatically compared to the visually obvious configuration of two-tier client-server architecture (Motiwalla & Thomson, 2012; Wortmann, 1998).

ERP modern architectures have shifted to apply new technologies in the 2010s, considering the emergence and development of innovative technologies, particularly the increasing use of the web and virtualization technologies in business processes. Thus, the three-tier model of ERP architecture has become richer and more powerful using these modern technologies. In the following sections we take a glance at the new developing architectures.

4.2. *Web-based architecture*

The web service architectures make use of web technologies and concepts to deliver simple, pervasive and real-time access to ERP modules for the end users and streamline the flow of information between systems. Web service technologies can offer many capabilities to enterprise software systems encouraging more and more ERP vendors to adapt and transfer their architecture in order to use the advantages of these technologies and provide powerful functionalities. In some new architectures the four-tier model has been developed in which the presentation tier was decomposed into two-tiers: services tier and browser tier (Sandoe & Boykin, 2007). This modern architecture provides its functionality using the following four components:

- Web server
- ERP portal
- Back-end server integration
- Browser plug-ins or applets

There are two important advantages regarding the use of web services: ease of integration and cost reduction by using a hosted application model (Acumatica, 2016). Whenever the end users who might be from customers, suppliers or within the organization request to access information from an organization ERP software, such as order status, inventory levels and invoice data, the web services can provide easy access for them without the need of a particular client software. The end users can simply access ERP applications using a web browser over the Internet infrastructure and easily integrate their applications with existing internal systems and external partners or suppliers' systems.

Figure 3 shows an example of web-based architecture proposed by Tarantilis et al. (2008). In this architecture each user logs into the system through a web portal and then an information flow engine using object oriented technology navigate the user to the individual system functionalities. Each functionality is developed using a web-based component which is easily pluggable to the system.

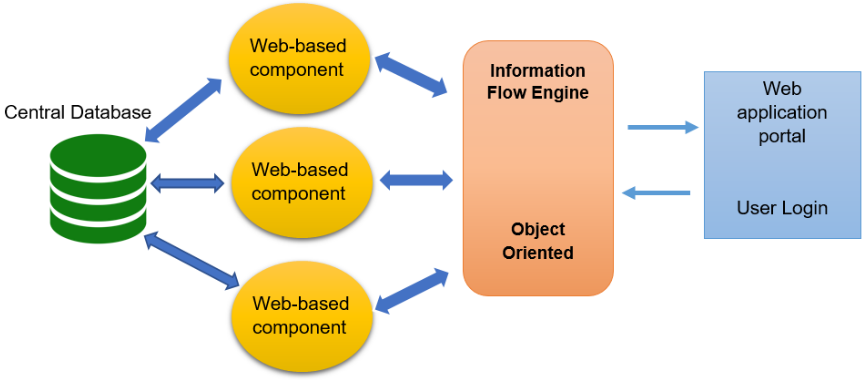

Fig. 3. Web-based architecture (adapted from Tarantilis et al. (2008))

Web-based architecture use object-oriented technology to apply a mixture of data and functions available in web service methods in order to provide services to different applications. It can help different vendors' proprietary applications to integrate with other tools and services. However, web services are strictly dependent on web protocols and standards. This might impose some limitations on the development on different modules in ERP. Furthermore, using web services might create more complexity in developing the presentation layer structure and make extra burdens on the network bandwidth (Acumatica, 2016).

4.3. *Service oriented architecture*

SOA is a style of software design where different functions are provided to the other components in the form of services through a communication protocol over a network. SOA is also intended to be independent of vendors, products and technologies (Lawler & Howell-Barber, 2007).

A service is a well-defined, self-contained and discrete unit of functionality that can be accessed remotely and acted upon and updated independently. For example, retrieving a credit card statement online can be implemented as a service. ERP systems can implement SOA by decomposing the business logic layer into smaller distinct units of functionalities as services (Datix Blog, 2018). These services can collectively provide the modular functions of ERP systems. A service can then be individually used anywhere in the modules of the ERP systems. The services in SOA must comply with certain design principles and standards. For each service there is one provider and one or more consumer. Two or more services can coordinate some activity for a business function. Since services are independent of the operating system platform, a consumer from a device with any operating system written in any language can use any service provided that it uses the SOA standard protocols and specifications. In SOA software development model, a contract is implemented between a consumer (client) and a provider (server) that specifies the following:

- A description of the service function
- A specification of service input and output
- The state of the environment before service can be called
- The state of the environment after service has been provided
- Error handling when there is an exception

A systematic survey of SOA adoption maturity models has been presented by Pulparambil &Baghdadi (2019). It is worth noting that although SOA is similar to Web services, it is not the same. Web services can be considered as an application of SOA with Web-based technologies such as SOAP and REST. However, SOA is more than a set of technologies; it is a standard that defines how to provide functionalities in software solutions independent of any specific technology. For ERP systems, SOA enables the system to provide agile resources for the business by a set of best practices which help the designer hide the complex implementation of each technology in the system. It decomposes and reorganizes the structure of different business applications so that they can offer different capabilities as independent services. Therefore, SOA can bring agility and flexibility for an organization in a way that it can improve business visibility, reduce the cost of integration, and increase the reuse of services. In a recent study, SOA was used to integrate an organization’s ERP system with mobile applications to automate the data import and export to the ERP and support the creation of real-time data communication between ERP systems and mobile applications (Erdiyana, 2019).

4.4. *Cloud architecture*

When organizations decide to use an ERP system in their business environments, they need to cope with many important problems related to purchasing and licensing costs, installation, maintenance and troubleshooting of the system. As a result, they need to plan and provide sufficient budget and hire specialized IT staff in order to take care of many tasks such as data storage, networking, technical support, security, high availability, privacy protection and compliance measures. Cloud computing has recently emerged as a powerful solution to satisfy the growing demands of many organizations that are not able or willing to purchase and maintain full software systems for their business environments.

Cloud computing is a mature technology build on top of advanced virtualization solutions for facilitating the access to computing and storage resources for any business processes. It can be best defined as “on-demand access to virtualized IT resources that are housed outside of your own data center, shared by others, simple to use, paid for via subscriptions, and accessed over the Web” (Brynjolfsson et al., 2010). Customers can access virtualized resources whenever they want and wherever they are using a simple browser without worrying about any other technical problems. Cloud computing offers three forms of virtualized resources as services for customers: Infrastructure as service (IaaS), Platform as a service (PaaS) and Software as a service (SaaS) (Armbrust et al., 2010).

ERP systems with cloud architecture can be categorized in the SaaS type of cloud computing services. A Cloud-based ERP system is a full suite solution with all the necessary ERP modules, where client organizations can pay as per the usage or a monthly fee for the type of service they get for their business (Washington Frank, 2017). In other words, it is as if companies lease the hardware and software of the vendor to avail the benefits of ERP. The ERP software system and database reside on the vendor's servers or at a third party-infrastructure and the ERP system can be accessed from a remote desktop connection on a computer, or even a simple browser on tablet or mobile device with an internet connection. Companies leasing cloud-based ERP services can save money by eliminating the ownerships and licensing costs associated with their purchased software and other installation and maintenance costs of on-premise ERP systems (Acumatica, 2018). Also, the organizations have the chance to choose the ERP modules they want and donot pay any money for other unnecessary modules which might not be needed in their business. Using the Cloud-based ERP offers customer organization high availability of resources, scalability, reliability and fault tolerance as well as security and ease of use (Pareek, 2014).

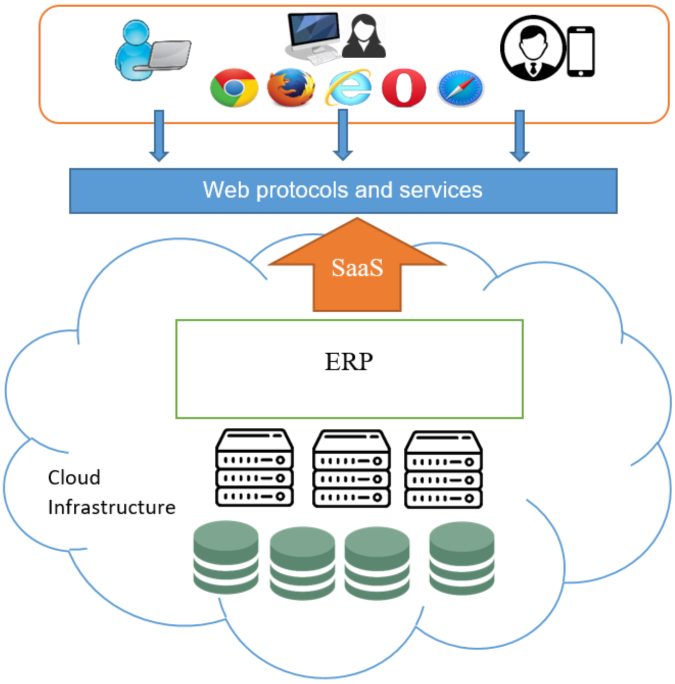

Fig. 4. Cloud based ERP architecture

Table 1 shows different architecture styles of ERP systems which are applied in ERP design for the modern age. For each architecture style, significant characteristics, the benefits and drawbacks are presented.

5. Technologies

In this section, we investigate the most important technologies applied in ERP solutions which are provided by some of the most popular vendors in the ERP markets. With all the benefits that the three-tier architecture brings for ERP software, most ERP vendors have moved to build their solutions based upon this beneficial architecture and tried to enrich it using modern technologies in the IT world. On the other hand, the rapid growth of cloud computing technology has made it possible to host any type of software system, including ERP systems, in the cloud using the benefits of transparency, scalability and ubiquity of the cloud environments.

In the following, we explain some of the most successful ERP systems along with their useful technological components. We have a look at SAP ERP software, Oracle NetSuite, SAGE ERP products, and Microsoft Dynamics ERP platforms. By presenting and describing the architecture of ERP solutions which are provided by well-known vendors and the diverse technologies used in them, the reader can gain an insight on how ERP systems architecture has evolved and continue to be more flexible and efficient for today's ever-changing business needs.

Table 1. ERP architecture style, characteristics, benefits and drawbacks

Architecture Style	Significant features	Benefits	Drawbacks
Two-tier	Using the client-server model	• Ease of access to the information, • Low cost for the infrastructure, High performance for limited number of work stations	• Lack of scalability • High cost for integration • High Impact of database modification on applications
Three-tier	Separation of functionalities in three layers: data, application, and presentation	• More scalability, reliability, and security • Easier maintenance of the software reusability of components	• Need for more resources Complexity of development environment
Web-based	Use of web services to provide functionality over web protocols	• Cost and time reduction • Ease if access to information • Less disruption to business operations Ease of integration	• Dependence on web protocols and standards Increased complexity in development process
SOA	Decomposing the business logic layer into smaller distinct units as services	• Increased scalability and flexibility • Decreased cost of integration Increased services reusability	• High cost of implementation • Extra overload on service controls
Cloud-based	Hosting ERP infrastructure and services outside the company's premise	• More scalability, reliability, and security • Easier maintenance of the software reusability of components	• Need for fast and permanent Internet connectivity • Customization Limitation • Data Security and ownership issues • Price Contemplation

Generally, it can be said that the benefits of three-tier architecture for the organizations are more than its limitations in the long run. Many large-scale distributed systems and enterprise applications including e-commerce solutions are increasingly applying this type of architecture. The Application tier is supported by modern component technologies such as Enterprise Java Beans (EJB) and CORBA Component Model (CCM) which provide powerful frameworks for application development with components. The three-tier architecture is also used by many web services based on hypertext transfer protocol (HTTP) and extensible mark-up language (XML) (Motiwalla & Thomson, 2012; Fowler, 2002).

5.1. SAP ERP system

SAP is the best-known ERP software (Gartner, 2019) vendor that has continuously improved its enterprise software solutions in order to respond to the ever-changing needs of its customers. SAP products for ERP systems has evolved in a wide spectrum from *R*/2, *R*/3 to ECC and its database systems has experienced advancements into HANA DB from its traditional RDBMS. This company has made great steps toward the world of enterprise mobile applications software and cloud enterprise solutions. SAP Business One is a highly scalable and powerful ERP software which covers all aspects of an organization functional needs and provides critical data to help managers make better business decisions (SAP Business One, 2018).

SAP ERP architecture has been designed based on the three-tier architecture described in section 4.1. The three-tiers in this architecture has the following functions (Kalaimani, 2016; STechies, 2008):

- The Database layer includes the relational database and database management systems (RDBMS).

- The Application layer is responsible for executing all SAP applications and processing the requests from presentation layer. This layer is also known as Kernel layer and Basic layer.
- The Presentation or user interface layer provides GUI for the end users so that they can interact with the applications.

Figure 5 depicts SAP systems architecture with its three layers (Kalaimani, 2016). The database layer which stores all the data needed for the ERP system can include a relational database such as Microsoft SQL Server, Oracle database or SAP native HANA DB (Baumgartl, 2016). The Application layer includes a set of servers which are responsible for background processing and requests management. The presentation tier provides GUI for end users' access to the system using front-end clients which serve as an interface. The front-end clients present SAP screen and provide tools for them to enter inputs to the system and receive appropriate outputs.

The application layer provides a range of services for the operation of the SAP system. The application servers' processes are carried out through work processes (WP) (STechies, 2008; ERPdb.info, 2020). The work processors are components that are able to execute a desired request. Each work processor is registered as a user in the database system for the entire runtime of the SAP system. A *dispatcher* collects the requests and distributes the work load to the different work processes.

SAP has developed a modern web-based, open integration, application platform called NetWeaver which serves as the technical foundation for many SAP applications (SAP NetWeaver, 2020). This technology applies a service-oriented architecture to integrate enterprise application services. The SAP NetWeaver Application Server (sometimes referred to as WebAS) is a runtime environment for the SAP applications.

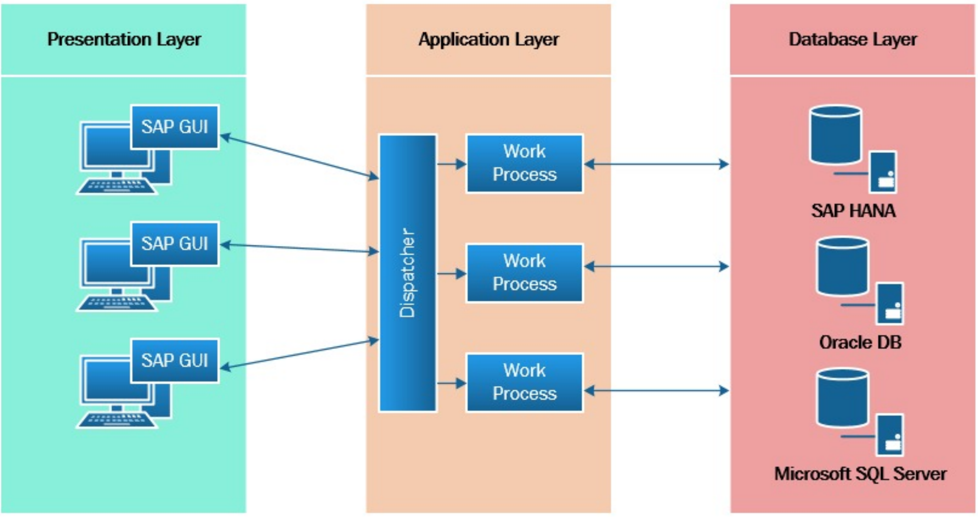

Fig. 5. SAP ERP three-tier architecture

All of *SAP* business management suite runs on SAP WebAS including supplier relationship management (SRM), customer relationship management (CRM), supply chain management (SCM), product lifecycle management (PLM), enterprise resource planning (ERP), transportation management system (TMS). The applications in SOA are designed to use Web services as the standard way to communicate well-defined information with an array of other applications.

The SAP NetWeaver is a comprehensive platform for development, composition and maintenance of the enterprise systems. As shown in Fig. 6, SAP NetWeaver application server hosts all applications such as business process management, data warehouse, Online Analytical Processing (OLAP), and extending applications to mobile devices. The architecture of SAP NetWeaver is composed of four-layers:

- Application platform layer
- Process Integration layer

- Information Integration layer
- People Integration layer

Therefore, SAP NetWeaver can provide a framework for integrating people, processes, and information. The application platform has been improved with SAP's native database named "HANA", SAP solution bundle, and IS-solutions, which are capable of integrating people (business) and processes (key business operations). Recently, the SAP NetWeaver platform has extended its functionality to support mobile business activities with the Fiori-based mobile platform. SAP Fiori is a modern user experience (UX) for SAP software and applications (SAP Fiori, 2020). The goal of designing SAP NetWeaver platform was to support heterogeneous software in order to provide more flexibility for a customer's solution.

SAP NetWeaver is a technical layer for all business applications running on it. For example, one may run a Java-based application or any other home-grown customer application which can consume SAP applications via XML as consumable services, and vice versa. You have a choice as a customer to develop your own software applications in Java (J2EE) and/or ABAP with customized solution.

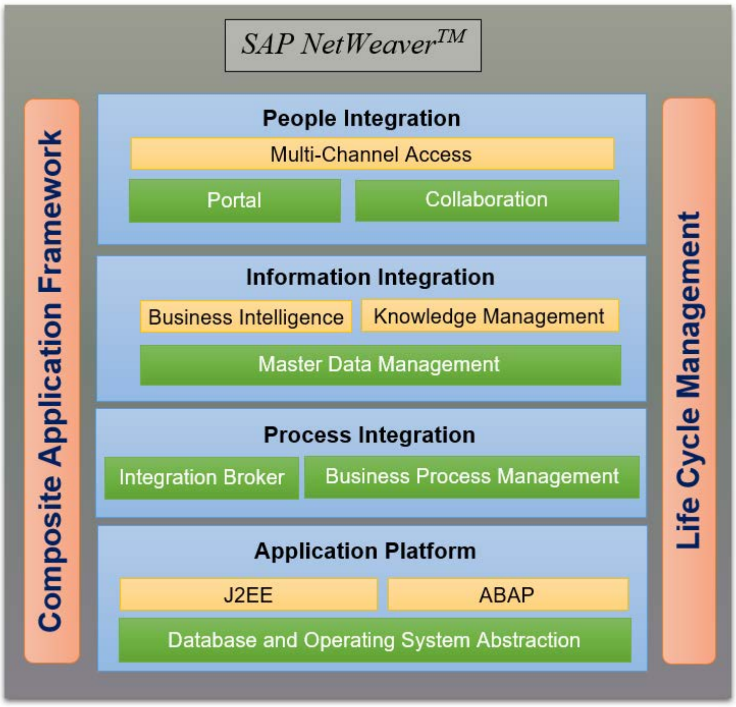

Fig. 6. SAP NetWeaver Architecture (Oio, 2003)

VMware, a giant vendor for cloud computing services, has cooperated with SAP company to provide a virtualized version of entire SAP landscape which can be used for testing, development, Q&A, training, and production environments (VMware, 2019). VMWare cloud technologies can improve efficiency, scale, flexibility and agility for the entire business environment enabling organizations to implement and deploy SAP HANA more easily. The SAP virtualized solution can bring many benefits for the client organization: (1) high availability with automated fault tolerance and zero downtime or data loss (2) Faster time to value with rapid and automated provisioning (3) Lower total cost of ownership (TCO) through automated data center operations, application provisioning and infrastructure delivery (4) better data protection, security and compliance.

5.2. Oracle NetSuite

Oracle NetSuite is one of the most popular ERP vendors and the world's largest cloud ERP provider which has consolidated its position as a market leader for ERP products (G2 Guide for ERP Sytem, 2020). With more than 19,000 customers running NetSuite ERP across more than 200 countries, this Cloud-based ERP system has been trusted by some of the world's best-known brands to successfully migrate their financial and operational processes to the cloud (Oracle Netsuite, 2020).

NetSuite provides a complete suite of Cloud-based applications in the form of SaaS, including financial management, Enterprise Resource Planning (ERP), Human Capital Management (HCM), Customer Relationship Management (CRM), professional services automation, and omnichannel commerce for many organizations in different sizes from small start-ups to global enterprises. In addition, the components of NetSuite are modular. Thus, it can be easily deployed and integrated with existing on-premise or cloud enterprise software as needed.

NetSuite ERP has been built on top of a flexible, complete and powerful infrastructure called *SuitCloud* platform (Oracle, 2019). NetSuite has implemented SuitCloud platform in order to provide comprehensive data privacy, data management, security, scalability and availability for running mission-critical ERP, CRM and ecommerce applications. Customers and software developers are allowed to use this platform components in order to maximize the benefits of the cloud capabilities.

SuiteCloud is basically a multi-tenant cloud platform that includes Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) technology (NetSuite, 2012). Figure 7 depicts the architecture of SuitCloud platform with three layers abstracting infrastructure, application and developer tools. In order to interact with NetSuite cloud and customize the applications, SuitCloud offers developer tools to both business analysts and software developers to rapidly extend, customize and integrate NetSuite capabilities for financials/ERP, CRM, ecommerce, etc.

SuiteCloud Developer Tools include workflow management, scripting, analytics, web services and more (Oracle Netsuite, 2017). SuiteBuilder provides graphical customization tools for powerful record and filed management and building sophisticated business forms without any coding. SuiteFlow is a work flow engine which provides users with easy-to-use, point-and-click tools to customize and automate business processes across NetSuite and between other business applications. SuiteScript offers a great set of scripting models and full-featured application-level scripting capabilities for developers and administrators to create flexible business logic within NetSuite tailored to their specific business needs. SuiteTalk provides web services tools which facilitate the integration of NetSuite with existing on-premise systems and third-party cloud applications. Furthemore, *SuiteAnalytics* provides embedded business intelligence (BI) tools to the applications built or customized with SuiteCloud, eliminating the need for any third-party BI tools. Finally, *SuiteBundler* allows developers to group and package up any customized or built applications on the SuiteCloud platform in order to easily deploy them in the production environment.

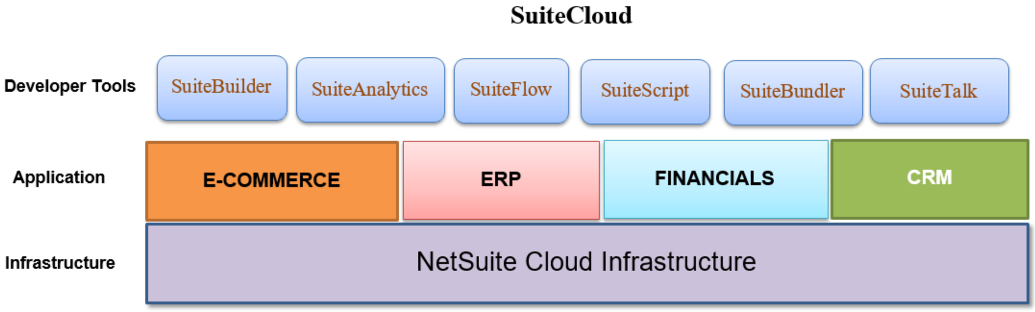

Fig. 7. NetSuite SuitCloud platform architecture (Oracle Netsuite, 2017)

5.3. *Microsoft Dynamics*

Microsoft Dynamics is the umbrella under which Microsoft develops and markets its ERP and CRM software products. The main benefit of any product designed within the Dynamics suite is that it can be easily integrated with other Microsoft products such as Microsoft Office, Windows SharePoint Services and Microsoft Outlook (Microsoft, 2019). Microsoft has recently bundled its ERP and CRM products with its cloud computing technology (using Microsoft Azure) and Artificial Intelligence capabilities (with Power BI and Cortana) providing a new family of enterprise software solution called Dynamics 365 (Encorebusiness, 2020).

Currently, there are two main Dynamics 365 ERP solutions provided by Microsoft (Microsoft, 2020; Encorebusiness, 2020):

- Dynamics 365 Business Central: ERP software-as-a-service solution well-suited for small and medium-sized businesses. It has been built based on previous *Dynamics NAV* ERP solution.
- Dynamics 365 Finance & Operations: ERP software-as-a-service solution meant for large or diversified Enterprises. It is particularly provided for companies with multiple lines of business, international operations, multi-currency needs, or complex manufacturing processes. It has been built based on previous *Dynamics AX* ERP which was mainly an on-premise solution.

Although, the above two ERP solutions offer the best flexibility and usefulness for the organization when deployed on the cloud, Microsoft has also allowed its customers to implement these solutions on premise for specific needs. The on-premise architecture of the Microsoft dynamics 365 ERP solutions follows the pattern of their predecessors. In other words, Dynamics 365 Business Central uses an architecture similar to the common three-tiered architecture of its predecessor Dynamics NAV (Microsoft, 2017) and Dynamics 365 Finance & Operations mostly is deployed based on Dynamics AX software which also conforms *three-tiered* model (Msdynamics, 2018).

The core components of the three-tiered architecture of Microsoft Dynamics 365 Business Central includes Web server, Server (application server), and Microsoft SQL Database. The ERP application modules are provided by Web server components of this architecture for end users through a series of web services run by the Microsoft Internet Information Services (IIS) (Microsoft, 2020). The developers also have the opportunity to use web services such as SOAP and OData to develop their customized applications and extract their required information from the ERP application server.

5.4. Sage ERP software

Sage is another market leader for enterprise software and integrated business management solutions (Sage, 2020). This company offers flexible ERP products for different type of customer organizations depending on the size and the business functional aspects required in their processes. Currently there are three major ERP products provided by Sage with extended use of cloud computing technology: Sage 100cloud, Sage 300cloud, Sage X3. Among these three products, Sage X3 is a comprehensive ERP solution specifically designed to support the demands of medium to large enterprises.

Sage X3 is very flexible as far as implementation and deployment are concerned. The platform powering Sage X3’s applications may be deployed in a variety of environments: as a Sage managed cloud SaaS, on customers’ own infrastructure on-premises, or even in a private cloud environment of their choice (Sage, 2017).

The modern architecture of Sage X3 allows customer organizations to support an open and flexible network of mobile users collaborating on the company data, without compromising the integrity and security of the enterprise information.

The usage of full cloud architecture managed by Sage eliminates the need for customers to install and maintain the solution infrastructure and to take care of integration, security, upgrade and backup procedures. Users may access the system in a usual web browser, and benefit from the full capabilities of the solution hosted in the secure and scalable Amazon Web Services (AWS) cloud platform, hosted in the Amazon data centers (Amazon, 2020).

However, Sage X3 does not oblige its customers to only use the proprietary cloud platform of this vendor. It is also flexibly possible for customer organizations to take full control of their software infrastructure and make the best use of their pre-existing environments by choosing to deploy Sage X3 on premises or in a private data center. The companies can use this option as an alternative deployment especially when it is necessary to integrate legacy applications with the new ERP system and customize the solutions for specific needs.

Sage X3 applies a multi-layer architecture to better organize the structure of the functions and services related to data management, application executions, and user interactions. This multi-tier architecture combined with SOA ensures that the ERP operations are highly reliable in all situations (Sage, 2016). Figure 8 shows the summarized architecture of Sage X3.

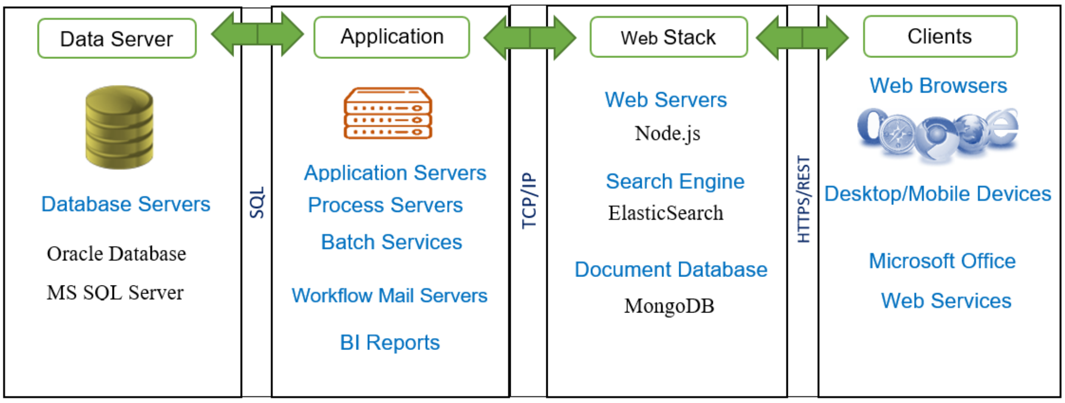

Fig. 8. Sage X3 ERP multi-tier architecture (adapted from Sage (2017))

With the usage of service-oriented architecture (SOA) Sage X3 ERP, all business management functionalities are accessible as interoperable services which be easily called by external applications. Moreover, Sage ERP X3 architecture provides Web Services in order to extend interoperability and flexible accessibility with web-based applications using popular languages such as .NET or Java. Sage ERP X3 web services support XML, UDDI, WDSL, SOAP, and .NET standards. The ERP implementers can dedicate an external server to Web Services and integrate it with the Sage X3 system.

As shown in Fig. 8 the architecture of Sage X3 comprises of four tiers of which three tiers are known for the reader and are well described in previous sections. The Web Stack layer is added by Sage to better leverage the modern technologies of Big Data processing and Cloud services. The components described below are scalable and provide high availability for the services using clusters:

- *Node.js* is an open source, cross-platform java-script runtime environment for server-side scripting and scalable networking applications (Node.js, 2020).
- Elasticsearch is a search server based on Lucene, a free and open source search engine software library. It provides a distributed, multitenant-capable full-text search engine with a RESTful web interface and schema-free JSON documents (Elastic, 2020).
- MongoDB is a scalable NoSQL database for administration of data in the form of JSON-like documents with load balancing and data replication features over multiple machines (MongoDB, 2020).

5.5. *Other ERP products*

In addition to well-known ERP products, there are other ERP solutions in the market to satisfy the needs for ERP solutions. For example, Oracle also provides another ERP system under the name of PeopleSoft, a newly acquired brand (Oracle, 2020b). The Oracle PeopleSoft application suite is an ERP system which applies web-based architecture profoundly. ERP systems combined with web services can become a flexible application platform with various components that are really helpful for multiple business functions. Recently, Oracle has also merged PeopleSoft ERP with its cloud platforms to better provide services for its customers. PeopleSoft application runs within the PeopleSoft Internet Architecture (PIA), which requires a variety of software and hardware elements, including relational database management system (RDBMS), application server(s), Process Scheduler server(s), web server(s), and web browsers (Oracle, 2016).

Epicor, is another popular ERP vendor which has implemented a complete service-oriented architecture for its ERP software and applied the best practices of service-oriented design for constructing its solutions (Epicor, 2020). Epicor ERP is delivered either on-premise, hosted or as a cloud-based software as a service (SaaS) offering.

Table 2 summarizes the information related to the architecture of some well-known ERP systems in the markets. As previously mentioned, the world market has commenced and will continue a fast movement to make the best use of the cloud services and infrastructure for providing ERP products and services.

With the emergence of cloud computing technologies and their tremendous capabilities and ease of use, most ERP vendors provide cloud-based architecture of their ERP so that more customers can afford to use their flexible ERP software. Of course, on-premise deployments are still available from some well-known vendors in order to satisfy the requirements of large organization. However, the flexibility and pervasiveness of cloud-computing technology is shifting the ERP market to providing more SaaS based ERP solutions.

Table 2. Well-known ERP products architecture

Vendor	ERP Product	Architecture Type
Oracle	NetSuite (OneWorld)	Cloud (SaaS)
	PeopleSoft	Web-based architecture + Cloud
SAP	S/4 HANA	On-Premise three tier with NetWeaver and SOA
	Business One	Cloud (SaaS), three tire + SOA for On-premises
Microsoft	Dynamics 365 Business Central	Cloud (SaaS), Three tier for On-premises
	Dynamics 365 Finance & Operations	Cloud (SaaS), Three tier for On-premises
Sage	Sage X3	Cloud (SaaS), Multi-tier + SOA for On-premises
	Sage 100cloud	Cloud (SaaS)
	Sage 300cloud	Cloud (SaaS)
Epicor	Epicor ERP	Complete SOA+ Cloud

6. Future Trends

ERP systems architecture has experienced a long journey from simple two-tier architecture with fully on-premise implementation to more modern styles with three tiers or more benefiting from new technologies and innovations. Today, it is very usual to see web-based and cloud-based architecture in many organizations' ERP implementations. Moreover, SOA can help ERP designers to make the system more scalable, reliable and flexible by decomposing the functionalities into small reusable components as services. ERP systems can benefit a lot from SOA when greater number of business processes are to be automated.

Considering the advantages of ERP systems, more companies get willing to use them for their business environment. The companies like to benefit from the advantages of ERP and usually do not like to spend a lot of money and time on implementing them. They also like to see their ERP system is integrated soon with their existing subsystems and give them a high degree of customization capabilities. Therefore, the challenge of ERP architecture is to provide more scalability, flexibility, ease of deployment, customization and integration (ERPdb, 2020). As a result, ERP vendors will focus more on providing ERP products which are able to serve customers with less cost and time and be flexible enough to integrate with the rest of the system as soon as possible. For this reason, web-based architectures, cloud computing technologies, specifically SaaS and SOA will play a significant role in the future of ERP architecture design.

Future trends of ERP architecture are formed by future needs of the businesses, the market competitions, and technological advancements that might change the way organizations expect ERP to meet their needs. Here, we explain some of the key trends.

6.1. Mobile ERP

As more people use mobile devices to connect to others, do business and access information, companies are more willing to use mobile applications to provide constant access to information and use this opportunity to expand their reach for the customers. Moreover, managers and end users expect a higher level of mobile connectivity and faster response time for performing their tasks (FDM4, 2020). They desire simple to use, configurable dashboards that give them one click to required and relevant information (Acumatica, 2018). Furthermore, the fast growth of social media and the massive user base that social media will enjoy in the coming years encourages new opportunities for digital marketing. Businesses may use these opportunities only when their ERP is properly integrated across social media channels and messaging applications. As a result, there is going to be a strong push for creating mobile apps presenting the functionality of large-scale ERP software.

ERP software architecture will need to provide a framework to offer various mobile services such as sending out push notifications, keeping track of orders and expenses, reviewing and approving employee time and presenting the real time data in different mobile devices (Terillium, 2019). Web services and SOA style will be really helpful to integrate mobile services into the whole ERP architecture. Mobile services would be implemented as separate services which connect to the main architecture using APIs or connectors. The architecture of ERP applications will allow a lightweight platform formed by microservices to provide various functionalities to end users and managers. Also, the presentation layers will need to create frameworks for more user-friendly interfaces and better user experience.

6.2. More shifts to cloud

Cloud is the future of most business applications, including ERP. A recent survey shows that 78% of UK chief information officers (CIOs) believe that cloud is critical to their IT strategy and corporate strategy, and 50% say they are already running a hybrid cloud ERP and benefiting from Total Cost of Ownership (TCO) reduction and increased agility (Accenture, 2018). Today, more and more companies are looking for new opportunities to expand their business to multiple geography and attract more customers (Acumatica, 2018). Besides, the SME segment is very cost conscious, especially about expenses with the mostly long term and intangible benefits (ERPdb, 2020). This will make them find the on-premise ERP solutions inflexible and limited to support their desires. Therefore, the landscape of business applications is changing and creating a great shift to use the latest and most innovative technologies.

Cloud computing is one of the promising technologies which will eventually dominate ERP market. However, each company will move at their own pace. At the beginning of the journey, more companies decide to use a hybrid architecture, keeping their on-premise legacy solutions safe and implement new business functionalities using cloud solutions. This method will give them the assurance that all enhancements are compatible and there is no dangerous modification in their core business processes. New companies who want to implement an ERP from the very beginning will be more likely to adopt cloud-based ERP solutions and implement an omnichannel pervasive business management solution.

The ongoing shift to cloud computing will continue and accelerate. We can expect new ERP software in the future to be increasingly built on an innovative cloud architecture designed from the ground up to deal with the massive computational challenges of processing global enterprise data in real-time. These platforms will provide the following capabilities (Washington Frank, 2017; Accenture, 2018):

- Microservices and open API’s for easy integrations and extensions
- A foundational AI engine, which enables predictive and intelligent solutions
- Fast in-memory computing with the ability to process large volumes of data in an instant
- A unified information architecture that eliminates the need for a separate data warehouse, and simplifies access to analytics for all users

6.3. ERP with the Internet of Things

The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, each with a unique identifier (UID), which have the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction (Internet of Things Agenda, 2019). IOT includes everything from mobile phones, coffee makers, refrigerators to cars, traffic lights, and buildings. It expands the connection possibility to the Internet for all types of devices even the devices and machines working in a production environment. The Internet of Things can now be considered a mature technology with a multitude of options both in terms of connectivity, hardware and use cases already available in the market (Loriot, 2020).

IOT can have a great impact on ERP systems performance in today's business environment. It may reduce the time to implement system monitoring and data collection technologies. The IoT has the potential to greatly increase data availability and accuracy. This has significant implications for enhancing ERP in areas including customer service, forecasting, inventory and asset management, and business intelligence (Anegis, 2018). IoT is also able to improve communications in different arenas of a business. For example, those products which communicate directly with the ERP system can make the connection to both customers and suppliers closer. IoT connectivity simplifies data transmission and allows us to tap into previously inaccessible data. This helps expand a vast array of analytics and AI applications.

Future ERP systems will be designed based on an architecture which makes the best use of IoT-enabled technologies. The architecture should allow for as many IoT devices as needed to connect with the ERP system and create scalable data transmission mechanisms between data repositories and IoT devices. The ability of creating real-time interactions with ERP and handling massive amount of data that is generated by the IoT devices is of high importance for the improvement of ERP performance.

7. CONCLUSION

The Application of ERP solutions bring many benefits for organizations such as performance improvements, cost reduction, and facilitating interactions between core business processes. For any type of ERP implementation strategy, whether it is to buy full out-of-the-box ERP products or to design and develop a custom ERP solution, the organization's ERP project managers and specialists need to acquire a good knowledge of ERP software architecture. The architecture defines the relationships among the complex information technology components, including hardware, software, and data with complicated organization components as company structures, business rules, and people. In this paper, we presented a comprehensive overview of the modern ERP systems architecture and investigate their underlying conceptual and logical models. We also introduced different technologies used for the modern ERP systems. The comprehensive overview of ERP architecture can provide an insight for ERP project managers, consultants on what type of architecture is suitable for their business needs or how their implemented ERP system components work together.

The ERP systems architecture has evolved from simple two-tier architecture based on client-server into more modern models with three tiers or more using complicated infrastructures and platforms. These days, web-based and cloud-based architecture are very common and can be seen in many ERP design frameworks. On the other hand, SOA has enriched the ERP architecture with the standard protocols and guidelines for implementing the business logic layer with the concept of service and reusability. With the fast development of cloud computing technology and its tremendous benefits for organizations, we see more and more companies shifting their ERP deployment strategy into cloud-based ERP implementation.

It can be said that the benefits of three-tier architecture for the organizations are more than its limitations in the long run. Most famous ERP products vendors in the market have established the basic foundation of three-tier architecture combined with modern technologies such as new DBMS and NoSQL databases for Data layer, process integration and information integration platforms for the application layer and new technologies for designing user interfaces in the presentation layer. The usage of web services in order to extend interoperability and flexible accessibility with web-based applications using popular languages such as .NET or Java is becoming prevalent.

Selecting the right type of architecture fore managers depends on their business requirements, existing software and hardware infrastructure and the amount of budget available for the ERP implementation. It is obvious that the two-tire architectures are now obsolete because of their inefficiency and should not be used for a complex system such as ERP. The three-tier architecture will provide sufficient component to design an efficient ERP system. The real challenge facing business managers is to choose between on-premise and cloud-based ERP architectures. If data security and ownership are major concerns for the organization, they need to implement on-premise architecture. However, for most SMEs, the managers are more concerned about ERP implementation and maintenance costs. This will make them choose a cloud-based architecture. Regardless of where the ERP system is hosted, the usage of SOA and web-services for the ERP architecture is of high importance for complex ERP systems. If managers decide to develop a custom ERP for their organization, these two architectures can help design a flexible and scalable ERP system with better integration. If the organization' strategy is to buy an out-of-the-box ERP product from the ERP vendors, they would better investigate the architecture to make sure it supports SOA, offer good web service capabilities to support scalability and integration with other subsystems.

The future trend of ERP market will include providing more cloud platforms for the ERP implementation to satisfy the diverse business needs of the customers and facilitating the implementation of on-premise ERP deployments for those companies who need to have their ERP systems on their own location. Furthermore, integrating ERP architecture with mobile applications and IOT infrastructures will be prevalent in the near future.

ACKNOWLEDGEMENT

The authors would like to acknowledge the support they received from the Research Center at FANAP Co.

REFERENCES

Acumatia (2018), available at < *https://www.acumatica.com/blog/the-future-of-erp-systems/*>.

Amazon Web Services (2020), available at: < *https://aws.amazon.com/* >.

Anegis (2018), Available at: < *https://www.anegis.com/articles/how-iot-enhances-erp* >.

Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A View of Cloud Computing. *Communications of the ACM*, 53(4), 50-58.

Baumgartl, A., Chaadaev, D., Choi, N.S., Dudgeon, M., Lahiri, A., Meijerink, B., & Worsley-Tonks, A. (2016). *SAP S/4HANA: An Introduction*. New York: SAP PRESS.

Brynjolfsson, E., Hofmann, P., & Jordan, J. (2010). Cloud Computing and Electricity: Beyond the Utility Model. *Communication of the ACM*, 53 (5), 32-34.

Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M. (1996). Pattern-Oriented Software Architecture: A System of Patterns, 1st Edition, Wiley.

Trillium (2019), Available at: < *https://terillium.com/future-trends-in-erp-systems-whats-next-for-erp* >.

Davenport, T.H., 2000. *Mission critical: realizing the promise of enterprise systems*. Boston: Harvard Business Press.

Accenture (2018), available at <*https://www.accenture.com/_acnmedia/pdf-90/accenture-unleashing-exponential-evolution-pdf.pdf*>.

ERPDB (2020), Available at: < *https://www.erpdb.info/sap-system-architecture/* >.

Elastic official website (2020), Available at: < *https://www.elastic.co/elastic-stack* >.

Epicor (2020), Available at < *https://www.epicor.com/en-eu/erp-systems/epicor-erp/* >.

Erdiyana, H. F. (2019). ERP System Integration with Mobile Applications Using Service Oriented Architecture. *In 2019 International Conference on Information Management and Technology (ICIMTech)*.Jakarta, Indonesia.

Fowler, M. (2002). *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley Longman Publishing Co., Inc.

Acumatica Official Web Site (2016), Available at: < *https://www.acumatica.com/blog/working-with-web-services-things-to-keep-in-mind-for-developers* >.

G2 Grid for ERP Systems (2020), Available at: <*https://www.g2.com/categories/erp-systems*>.

Ganesh, K., Mohapatra, S., Anbuudayasankar, S.P., & Sivakumar, P. (2014). *Enterprise Resource Planning: Fundamentals of Design and Implementation*. Berlin: Springer.

Hall, C.L., & Hall, C.L. (1994). *Technical Foundations of Client/Server Systems*. New York: Wiley.

Jour, J.H., & Zhaolin, F. (2013), The Design of ERP in the Multi-tier Architecture. *In 2013 Fourth International Conference on Digital Manufacturing & Automation*. Shandong, China.

Kalaimani, J. (2016). *SAP Project Management Pitfalls: How to Avoid the Most Common Pitfalls of an SAP Solution*. Berkeley: Apress.

Lawler, J.P., & Howell-Barber, H. (2007). *Service-oriented architecture: SOA strategy, methodology, and technology*. Florida: CRC Press.

Lee, H.Y., & Wang, N.J. (2019). Cloud-based Enterprise Resource Planning with Elastic Model–View–Controller Architecture for Internet Realization. *Computer Standards & Interfaces*, 64, 11-23.

Loriot (2020), Available at: < *https://www.loriot.io/blog/IoT-trends-2020.html* >.

Markus, M.L., Tanis, C. & Van Fenema, P.C. (2000). Enterprise resource Planning: Multisite ERP Implementations. *Communications of the ACM*, 43(4), 42-46.

Internet of Things Agenda (2019), Available at: < *https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT* >.

Microsoft Dynamics Official Website (2019), Available at: < *https://dynamics.microsoft.com/* >.

Microsoft Dynamics 365 documentation (2020), Available at <*https://docs.microsoft.com/en-us/dynamics365*>.

Microsoft Dynamics NAV Architecture Overview (2017), Available at: < *https://docs.microsoft.com/en-us/dynamics-nav/product-and-architecture-overview* >.

MongoDB official website (2020), Available at: < *https://www.mongodb.com/*>.

Msdynamics, (2018), Available at: <*https://msdynamics.net/dynamics-ax-updates/microsoft-dynamics-ax-architecture/*>.

Monk, E., & Wagner, B. (2012). *Concepts in enterprise resource planning*. Chicago: Cengage Learning.

Motiwalla, L.F., & Thompson, J. (2012). *Enterprise systems for management*. Boston: MA Pearson.

Muscatello, J.R., & Chen, I.J. (2008). Enterprise Resource Planning (ERP) Implementations: Theory and Practice. *International Journal of Enterprise Information Systems*, 4(1), 63-83.

Nah, F.F.H. (2001). *Enterprise Resource Planning Solutions and Management*. IRM press.

NetSuite (2012), Available at: < *https://www.netsuite.co.uk/portal/common/pdf/ds-suitecloud.pdf* >.

Ng, K.C. (2010). The Design of an Enterprise Information System Using Hierarchical Design Pyramid and Web-based Object Oriented Model. Doctor of Philosophy. *Hong Kong Polytechnic University*.

Node.js Official Website (2020), Available at: < *https://nodejs.org/en/* >.

Oracle Netsuite, (2020), Available at: < *https://www.netsuite.com/portal/assets/pdf/ds-netsuite.pdf* >.

Oracle Nesuite (2017), Available at < *https://www.netsuite.com/portal/assets/pdf/ds-suitecloud-platform-development-tools.pdf* >.

Oracle official Website (2020), Available at: < *https://www.oracle.com/applications/erp/what-is-erp.html* >.

Oracle (2020), Available at: < *https://docs.oracle.com/cd/E66686_01/pt855pbr1/eng/pt/tsvt/task_PeopleSoftArchitectureFundamentals-d27b40.html?pli=ul_d173e35_tsv t* >.

Oracle, (2020), Available at < *https://www.oracle.com/applications/peoplesoft/products.html* >.

Gartner (2019), Available at: < *https://www.gartner.com/doc/3913449* >.

Piturro, M. (1999). How Midsize Companies are Buying ERP, *Journal of Accountancy*, 188(3): 41–48

Pareek, R. (2014). Analytical study of cloud ERP and ERP. *International Journal of Engineering and Computer Science*, 3(10), 8710-871.

Pulparambil, S., & Baghdadi, Y. (2019). Service Oriented Architecture Maturity Models: A Systematic Literature Review. *Computer Standards & Interfaces*, 61, 65-76.

Rashid, M.A., Hossain, L., & Patrick, J.D. (2002). The Evolution of ERP Systems: A Historical Perspective. *In Enterprise Resource Planning: Solutions and Management* IGI global, 35-50.

Washington Frank (2017), Available at < *https://www.washingtonfrank.com/blog/future-of-erp-2018/* >.

Developer (2004), Available at: < *https://www.developer.com/java/ent/article.php/3446551/a-developers-overview-of-erp.htm* >.

Sage (2020), Available at < *https://www.sage.com/en-ae/erp/* >.

Sage, X3 Technology and Architecture Overview (2017), Available at: < *http://www.tangerinesoftware.com/documentation-SageX3-version7/Sage-X3-V11-Technology-and-Architecture.pdf* >.

Sage X3 Architecture and system requirements, (2016), Available at: < *https://partnerportal.sagex3.com/sites/default/files/sage_x3_partner/storage/architecture_guide_v11.pdf* >.

Encorebusiness (2020), Available at: < https://www.encorebusiness.com/blog/what-is-microsoft-dynamics-365/ >.

Sandoe, K.C., & Boykin, G.R. (2007). *Enterprise integration*, New York: John Wiley & Sons, 79–81

VMware (2019), Available at: < *https://www.vmware.com/solutions/business-critical-apps/sap-virtualization.html* >.

SAP Business One (2018), Available at: < *https://www.sap.com/uk/products/business-one.html?pdf-asset=2e4955be-937c-0010-82c7-eda71af511fa&page=1* >.

SAP NetWeaver (2020) Available at: < https://www.sap.com/uk/products/netweaver-platform.html >.

SAP Fiori (2020), Available at: < *https://www.sap.com/products/fiori.html* >.

SAP, (2020), Available at: < *https://www.sap.com/uk/products.html* >.

Shirazi, B. (2019). Cloud-based Architecture of Service-oriented MES for Subcontracting and Partnership Exchanges Integration: A Game Theory Approach. *Robotics and Computer-Integrated Manufacturing*, 59, 56-68.

Datix Blog (2018), Available at: < *https://blog.datixinc.com/blog/soa-erp* >.

Oio (2003), Available at: < *https://www.oio.de/sap-netweaver-components.htm* >.

Seddon, P.B., Shanks, G. & Willcocks, L. (2003). Introduction: ERP-The Quiet Revolution. *Second-wave Enterprise Resource Planning Systems: Implementing for effectiveness*, 1-16.

STechies (2008), Available at: < *https://www.stechies.com/types-of-work-process-in-sap/* >.

Sumner, M. (2014). *Enterprise resource planning*. New York: Pearson Education.

Tarantilis, C.D., Kiranoudis, C.T., & Theodorakopoulos, N.D. (2008). A Web-based ERP System for Business Services and Supply Chain Management: Application to Real-world Process Scheduling. *European Journal of Operational Research*, 187(3), 1310-1326.

Wortmann, J.C. (1998). Evolution of ERP systems. *In Strategic Management of the manufacturing value chain*. Springer, 11-23.