WEB SERVICES AS A SOLUTION FOR CLOUD ENTERPRISE RESOURCE PLANNING INTEROPERABILITY

Djamal Ziani and Nada Alfaadhel

King Saud University, College of Computer since, Department of Information Systems, Riyadh, Saudi Arabia

ABSTRACT

Recently, organizations have shown more interest in cloud computing because of the many advantages they provide (cost savings, storage capacity, scalability, and speed of loading). Enterprise resource planning (ERP) systems are one of the most important systems that have been upgraded to cloud computing. In this thesis, we focus on cloud ERP interoperability, which is an important challenge in cloud ERP. Interoperability is the ability of different components to work in independent clouds with no or minimum user effort. More than 20% of the risk rate of cloud adoption is caused by interoperability. Thus, we propose web services as a solution for cloud ERP interoperability. The proposed solution increases interoperability between different cloud service providers and between cloud ERP systems with other applications in a company.
.

KEYWORDS

Cloud computing, ERP, interoperability, web services.

1. INTRODUCTION

"By 2020, organizations with no-cloud policy will be rare" [Gartner]. In this research, we focus on cloud ERP challenges and, more specifically, on interoperability. Enterprise resource planning (ERP) is a technology that combines and integrates all business activities of an organization. The adoption of an ERP system at any organization can improve the performance, business flow, and availability of information. With the rise of cloud computing technology, small and medium enterprises (SMEs) became interested in ERP systems. Cloud ERP became more popular than on-premise ERP for many reasons, and low investment is definitely one of them. The interoperability issue is one of the major challenges when adopting cloud ERP systems. A company can decide to use cloud ERP as a solution to automate its business processes, but it should think about how the new cloud ERP will be integrated with the other applications and systems in the company. In this thesis, we suggest web services as a solution to help in solving the interoperability issue.

2. BACKGROUND

2.1. What is ERP?

ERP is a technology that provides solutions for all business activities in an enterprise, with a single application and a single data repository [1]. It combines and automates all business

activities across an organization [2]. It also assists managers and employees in reaching their goals of reducing the product cost and elevating efficiency [3].

In another word, ERP provides organizations with integrated solutions to manage their resources in an efficient and effective way. Therefore, the purpose of ERP systems is to facilitate all business function processes inside the organization and to manage all the connections outside the organization [4].

One of the benefits that an organization can get from using ERP is it enables the organization to re-engineer its business processes in order to achieve long-term success. ERP can also improve the process flow, customer service, and enterprise performance [5].

The ERP evolution started in the 1960s, where most enterprises were interested in inventory control systems under the inventory control package. In the 1970s, material requirements planning (MRP) systems were developed, and their main function was scheduling the production and material requirements. Later, in the 1980s, MRP was upgraded to its next generation (MRP II), which expanded to cover more business processes, such as finance, project management, and HR. MRP II also focuses on synchronizing production requirements with materials.

In the early 1990s, ERP was founded based on the MRP and MRP II systems. ERP systems took enterprises to a higher level in integrating business processes, including all enterprise systems [6].

2.2. Cloud computing

Cloud computing is a model that enables access of a convenient, on-demand network to a shared pool of configurable computing resources (e.g., networks, servers, storage facilities, applications, and services) that can be rapidly provisioned and released with a minimal effort of management or interactions of the service provider [7].

The National Institute of Standards and Technology defines cloud computing as "an access model to configurable computing resources at any location appropriately and throw the network based on demand" [8].

Organizations can benefit from cloud computing in terms of cost reduction, flexibility increase, automation, high availability, and hardware independence [4].

Cloud computing can be classified under two main aspects: cloud by services and cloud by deployment.

- Cloud service models are:

    - Software as a service (SaaS): Provides users with the ability to access the application of the service provider, which is located in the cloud side through a web portal
    - Platform as a service (PaaS): Provides a platform to support the building, operation, and modification of applications. This platform includes providing development, testing, and production environments. An example of this platform is Google App Engine.

- Infrastructure as a service (IaaS): Provides a computing environment that includes hardware and system software, such as servers, operating systems, storage, communication networks, and any supporting tool where the cloud provider is responsible for the installation and maintenance. An example of this platform is Amazon EC2 [9, 10].

Cloud deployment models are:

- Public cloud: In this type of infrastructure, the cloud is available for the public or large industrial groups, and it is owned by an organization that sells cloud services.

- Private cloud: This type of cloud is operated for a single organization. It can be managed by an organization, IT department, or third party.

- Community cloud: It supports a specific community interest and shared by several organizations.

- Hybrid cloud: This type is a composition of two or more cloud deployment models (at least one private and at least one public), which are bonded together to enable data and application portability [7, 11, 12].

Problems in cloud computing:

Despite of all opportunities and advantages, cloud computing raises several issues:

- Security and privacy problems: The security challenges at various levels, namely, network, host, and application levels, are not specifically caused by cloud computing but instead are exacerbated by its use. The issues of infrastructure security and cloud computing can be addressed by clearly defining trust boundaries by understanding which party provides which part of security [14]. The security of data and identity of users are the most serious problem in cloud computing.

- Data storage problems: When an organization choses to store data on the public cloud, it loses its ability to have physical access to the servers hosting its information. As a result, potential business sensitive and confidential data are at risk from insider attacks. According to a recent Cloud Security Alliance report, insider attacks are the third biggest threat in cloud computing. Therefore, cloud service providers must ensure that thorough background checks are conducted for employees who have physical access to the servers in the data center. In addition, data centers must be frequently monitored for suspicious activities.

Moreover, cloud service providers often store data of more than one customer on the same server. As a result, one user's private data can be viewed by other users (possibly even competitors). To handle such sensitive situations, cloud service providers should ensure proper data isolation and logical storage segregation.

- Interoperability and portability: The cloud ecosystem is large, with many providers offering a wide variety of cloud services. Understanding the portability and interoperability "of what" is the necessary first step of planning and designing the use

of any cloud service. Clarifying the specific interoperability and portability concerns accelerates the identification of the "best fit" options and potential development of solutions. These issues have a huge impact on the usage and adoption of cloud. However, many experts says that interoperability is the greatest challenge that is faced in cloud computing adoption [15].

Cloud computing has a huge impact on enterprise software. Moving from traditional on-premise ERP to cloud-based ERP provides enterprises with a high demand of reliability, scalability, and availability of computing services [16, 17].

ERP systems were traditionally focused on large organizations. However, with the rise of cloud computing technology, SMEs became interested in the field. Nowadays, many organizations are turning to cloud ERP. As ERP is provided as a ready-to-use software solution, cloud ERP falls into the SaaS category of cloud services [16, 17].

Figure 1 shows the huge growth in the cloud service market from 2010 to 2017 (in billions) [18]:

Cloud service market size

	2010	2011	2012	2013	2014	2015	2016	2017
cloud service market size	80	90	109	132	150	180	200	250

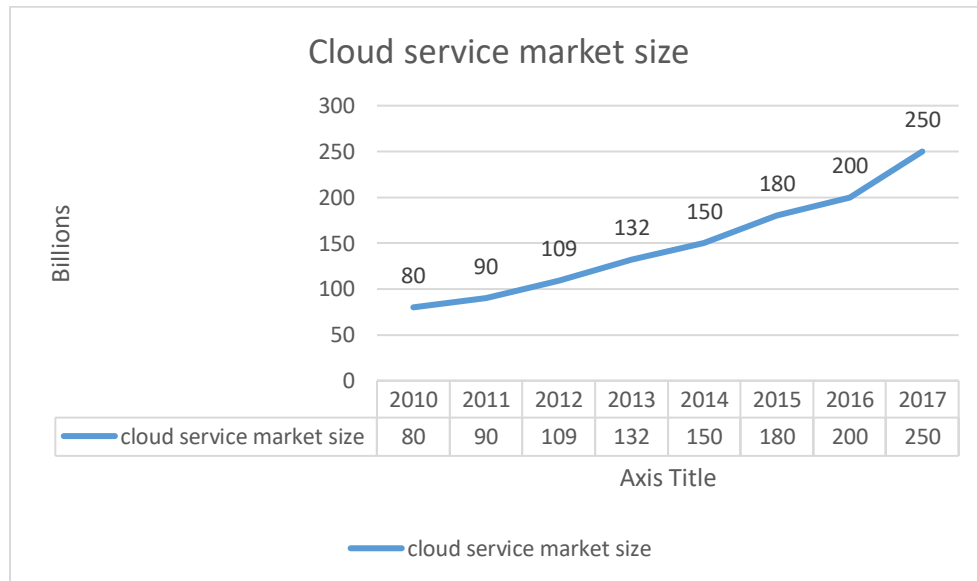Axis Title

cloud service market size

Figure 1. Cloud service market size

Cloud-based ERP is a development of integrated business suite supporting ERP, CRM, and e-commerce capabilities, where application modules can be accessed using an SaaS delivery model and where application users have flexibility to configure and subscribe to a set of application modules.

Vendors can provide organizations with three types of cloud ERP:

1. SaaS using a cloud infrastructure: This type provides maximum flexibility to switch vendors.
2. SaaS using a cloud platform: In this type, the vendors move applications to a separate cloud platform provider.

3.   SaaS using a cloud application: In this type, applications, infrastructure, and hardware are tightly integrated and cannot be separated [45].

What are the reasons for choosing cloud-based ERP?

1.   Cloud ERP does not require hiring IT professionals, buying servers, or having in-house software.
2. Managing the ERP is too complex and challenging for some organizations.
3. Cloud ERP is much economical [13].

3. RELATED WORK

Cloud interoperability problems first arose when multiple cloud providers exchange their data applications and Virtual machines (VMs) between heterogeneous cloud systems [19].

First, we have to understand the interoperability problem to suggest a good solution.

Arunkumar and Neelanarayanan [20] defined interoperability as the ability of different components to work in independent clouds with no or minimum user effort. They also stated that 22.9% of the risk rate of cloud adopting is caused by interoperability. In their paper, some approaches to solve interoperability were mentioned: one was the integrated approach (all cloud providers agree to follow the same standers), a unified approach (cloud provider has an equivalent model to map other standers), and a federated approach (all cloud venders agree to share the working semantics). However, interoperability issues can be raised at different levels.

From a user perspective [21] Described cloud interoperability as follows: Cloud users want to have all their needs in one cloud and one vendor or any other interoperable clouds without paying any extra effort in the development.

However, interoperability issues can be raised at different levels:

• Application interoperability: to port an application from its traditional infrastructure to cloud environment or from cloud to cloud or from IaaS cloud provider to PaaS cloud provider, reengineering the application is needed.

• Platform interoperability: there are two methods to achieve platform interoperability: platform source portability (in this method the platform is ported with the application), and machine image portability (machine image is a group of operating system, platform, configuration and application which can be moved across different infrastructure clouds that follows the same visualizing technology)

• Storage interoperability: a solution for storage interoperability issues was proposed by the Storage Networking Industry Associating (SNIA) that was called Cloud Storage Management Interface (CSMI) which is capable of accessing, administrating and self-provisioning data elements in different clouds by using Restful web services.

• Management and configuration interoperability: the cloud providers should manage and configure the physical system to avoid any system failure which will increase the uptime of their virtual servers.

The authors of [20] introduce 4 other levels of interoperability :

1. Technical interoperability: machine to machine communication.
2. Syntactic interoperability: ability to exchange data.
3. Semantic interoperability: ability to operate on that data.
4. Organizational interoperability: capability of organizations to communicate and transfer meaningful data.

Istvan Mezgar and Rauschecker [15] mentioned that cloud computing interoperability is a must for a good cooperation of multiple Internet resources. Some cloud computing experts say that interoperability is the greatest challenge faced in cloud computing adoption. Other IEEE cloud experts say that standers must be developed to overcome the interoperability challenge. They define interoperability as "the ability of two or more systems or applications to exchange information with not much effort from the customer."

Moreover, they introduce three different approaches in solving enterprise interoperability problems:

- Integrated approach: In this approach, a common format for all models should be agreed on by all parties.
- Unified approach: The common format exists in the meta-level and allows mapping between models.
- Federated approach: Partners have to share an ontology to map their concepts at the semantic level. This approach has no common format.

On the other side, they define cloud system interoperability as "the ability of writing a code that can work with more than one cloud provider, and the user is concerned about the capability to communicate between multiple clouds regardless of the differences between their providers."

Then, they classify cloud interoperability into two groups: interoperability of the cloud system (cloud layer) and interoperability of the applications (application layer).

Cloud interoperability is the ease of integration and migration of data and application between multiple clouds. Scott Dowell and his colleagues discussed the following cloud interoperability challenges [14]:

- Portability and mobility: This is about the image, application, and data interoperability. Portability is the moving of image—in a down state—from one host to another, and then this image is booted at its destination. Mobility is "the ability to move the workload of a live computer from one host to another without the in-flight state or losing the client connection." Portability and mobility are very important in indicating the degree of interoperability between clouds.
- Cloud service integration: Any enterprise may need to have its on-premise applications, and SaaS applications interact with each other; web services and SOA principles are able to solve this problem.
- Security, privacy, and trust: Having a well-defined cloud security policy and building a trust relationship between clouds are important for users to control and access remote resources across cloud boundaries.

They also presented a five-level cloud-to-cloud interoperability (C2CI) model:
- Level 0 (domain-based interoperability): At this level, we have an integrated environment with shared data, wide area networks, and separated applications, where the cloud services are limited to single cloud providers.
- Level 1 (enterprise-based interoperability): At this level, we have a universal environment with a wide area network, shared data, and applications and cross-domain information sharing and advanced collaborations via inter-cloud web services.
- Level 2 (portability interoperability): At this level, we have a public, private, or hybrid cloud environment, where cloud artifacts can travel across multiple providers in down states.
- Level 3 (security interoperability): At this level, we have a public, private, or hybrid cloud environment, where the policies and procedures from one cloud provider will interact with other policies and procedures from other cloud providers.
- Level 4 (mobile interoperability): At this level, we have a public, private, or hybrid cloud environment, where cloud artifacts can travel across multiple providers in in-flight states.

4. SOME PROPOSED SOLUTIONS FOR CLOUD INTEROPERABILITY:

Gupta and Bhatia in [12] discussed interoperability for information exchange and how standardization among communication protocols is very helpful in assuring Internet-wide application interoperability. They proposed the use of simple and standard XML protocols in web services as the best solution for interoperability over existing web communication protocols. They also mentioned two other options: using a pre-existing non-web-based protocol and designing a new non-web-based protocol. The first option may be undesirable because it suffers from a lake of universal deployment, whereas the second option requires more effort to be done on designing a new protocol, which also makes it undesirable.

Many companies have standardized their processes, data, and systems through the implementation of their ERP. This process can help create high integrated connections between the instance of managing the consistency of transaction data and that of producing reliable information [23]. This can be a solution for interoperability, but in our study, we will concentrate on web services as a solution.

Some recently published studies address interoperability issues as a major challenge faced by enterprises to survive continually changing environments. Moreover, they discussed the need for a continued research in enterprise interoperability. Panetto and his colleagues [24] explained how traditional systems manage the interoperability issues by developing collaborative interfaces and negotiations between systems to enable a meaningful interoperation, where the new generations of enterprise information systems are expected to reduce the pre/agreements needed to be established to enable interoperability. Romero and Vernadat [25] highlighted the future trend of the enterprise technical architecture and how web services and XML-based approaches can support interoperability and integration of systems, and this is what our solution is about.

[26] and [19] introduced a cloud broker solution for cloud interoperability. A cloud broker is a third-party mechanism or technique used to enable interactions among heterogynous cloud systems that are not able to interact with one another.

[27] presented another solution for interoperability between clouds, which is based on an integration of the monitoring technique and holistic message bus system. This solution provides a complete solution for communications in clouds and achieves interoperability between clouds.

[28] developed a framework for cloud interoperability based on compliance and conformance. He discussed how different cloud vendors can have different services and how APIs will lead the users to face interoperability problems if they will not relay on one cloud provider. He also discussed how standardization can help in solving cloud interoperability problems.

4.1 Why web service is the solution?

What is web services?

SOA is an approach in software architecture that is used to build platform-independent software component (services) that carry out small functions. This approach will improve the quality and productivity of a system [32].

SOA enables the integration of applications and resources to be flexible through the following: 1) SOA represents every application and resource as a service; 2) SOA enables the service to exchange structured information; and 3) SOA coordinates and meditates between services.

Web services are based on SOA. They are software components that provide services to other applications. W3C defines a web service as a software application that is identified by a URL, and its interfaces and bindings can be defined, described, and discovered as XML artifacts [33].

Bertolino and Polini [36] discussed the importance of web services in supporting interoperability between software applications. They define a web service as a "software system designed to support interoperable machine-to-machine interaction over a network." They proposed an audition framework to raise awareness on web services' interoperability issues and to study the behavior of the services, especially their interaction with other services, before they become publicly available. Their idea is still in its primary stage, and they have to work on it to make it more reliable and effective.

Curbera and his colleges[26] define a set of properties that should be displayed by web services:

1. Gray box component: web services are encapsulated and have a description file.
2. Web services are loosely coupled.
3. Flexible integration.
4. Massages instead of API's.

Because web services are platform and language independent, they become concerned with the problem of application to application interaction and for their interoperability they become a recommended solution for enterprise application integration [31, 32].

Using web service for interoperability is not a new solution. Basar daldal and his collages suggested   using RESTful Web Services for webRTC signaling interoperability[33]..

Cristian Danila and his colleagues proposed a web service architecture to achieve interoperability in SCM. They discuss how can a supply chain in manufacturing increases its productivity and lower the costs by using the proposed architecture that would provide context awareness and interoperability [34]. [35] Suggested a web service interoperability solution for cloud computing and provide a case study by connecting google app engine and Microsoft windows azure platform. They conclude their paper with the fact how web services are very effective in obtaining interoperability in cloud computing. All of these studies enrich our proposed solution how can web service be used as a solution for cloud ERP.

5. THE PROPOSED SOLUTION:

We propose a web service generator wizard that can generate web services to connect between cloud ERP applications. Our web service wizard help finding, combining and creating web services that can be helpful in minimizing the interoperability problem in cloud ERP. This wizard consists on a step by step procedure helping the customer, who want to integrate the new ERP system with the existing legacy systems, to identify how and what to create as web service to build the channel of communication. This procedure is done as follows:

- Identify which business object in ERP is involved in integration, for example Sales business Object.
- Identify which functions in the business object are involved, for example Goods Issue.
- Identify the existing APIs or web services in ERP system that can be used for the integration.
- If there is no web services in ERP system, create a new web service using the corresponding API.
- Identify which legacy system we want to integrate
- Identify what business object in legacy system is involved in the integration
- Searching for existing web services, APIs or database tables related to the function that we want to integrate.

There are 3 ways of creating the integration web service:

- Web service mapping: Create a composite web service based on the ERP web service and legacy web service.
- API mapping: Create a web service based on the ERP web service and by using the legacy API to extract the input output and by calling the API from the web service.
- Database mapping: It is used when there no legacy web service and API, in this case, the system will create a legacy web service based on the involved database table, and then create a new integration web service based on the ERP web service and the created legacy web service.

Company sales system

- The previous system contains a database of old customer accounts.
- The new system manages old and new customers in a cloud-based system.

The API will link the old and new systems together.

Example 1: Get customer info

Process:

- The new system sends a request to the API.
- The API receives a customer ID.
- The API sends a request to the old system.
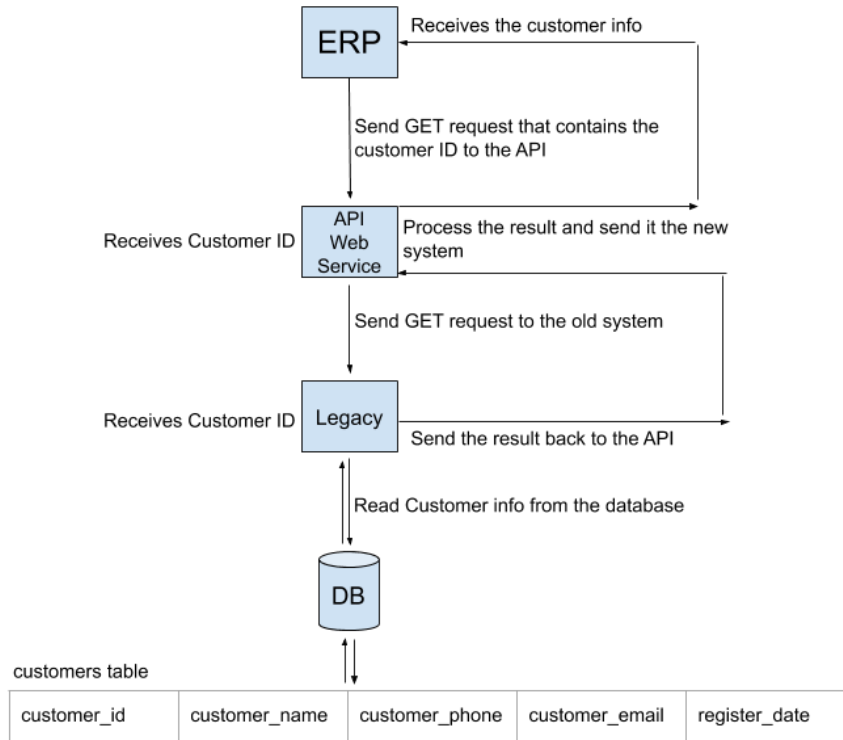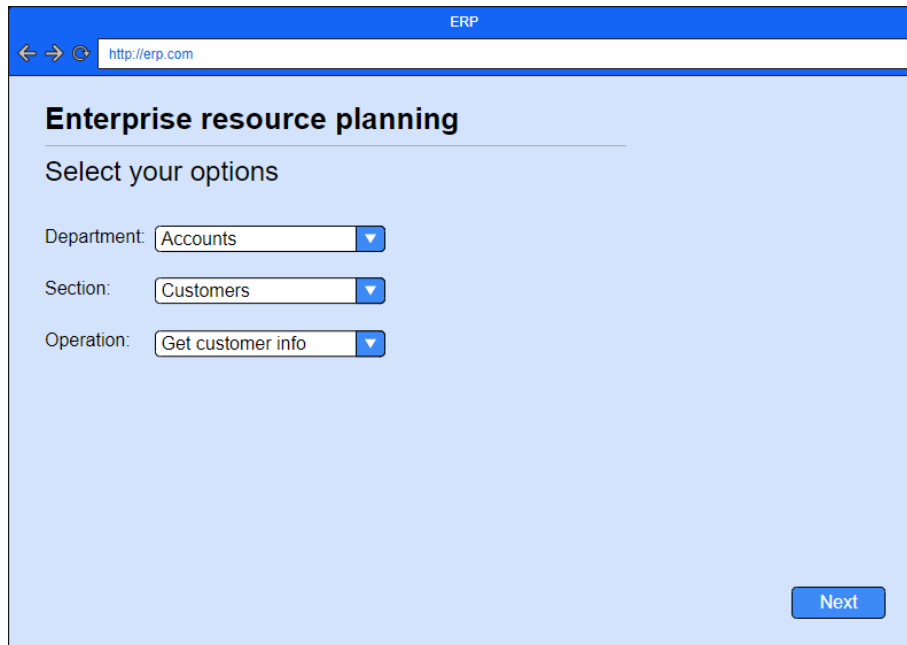- The old system sends back the customer info using the customer ID.
- The API sends the results back to the new system.

Figure 2. Get customer info
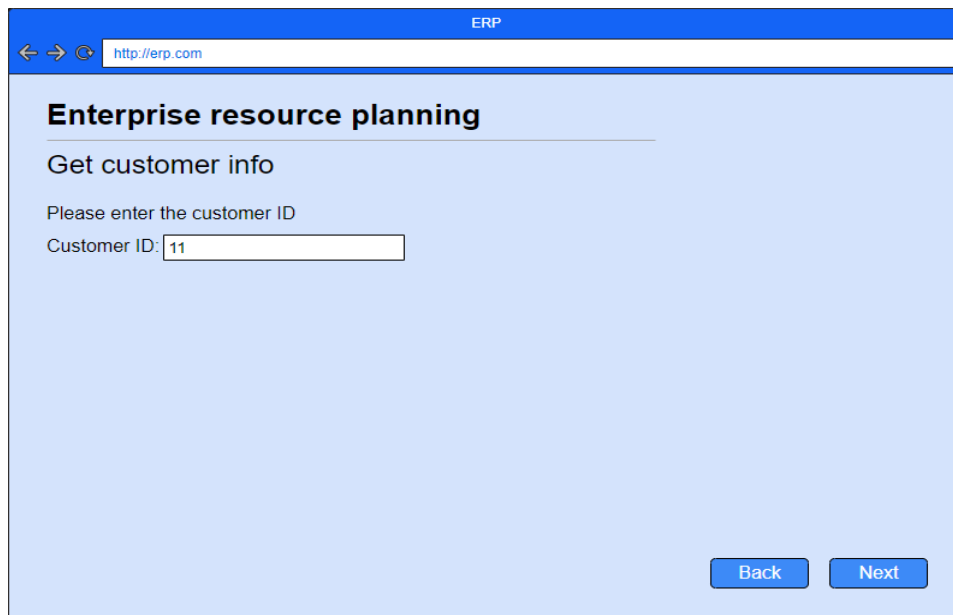
Figure 3. User interface 1

```
.
.
<form action="#" method="post">
  <h3>Select your options:</h3>
  <div>
    <label>Department:</label>
    <select name="department">
      <option value="accounts">Accounts</option>
      <option>...</option>
    </select>
  </div>
  <div>
    <label>Section:</label>
    <select name="section">
      <option value="customers">Customers</option>
      <option>...</option>
    </select>
  </div>
  <div>
    <label>Operation</label>
    <select name="operation">
      <option value="get_customer_info">Get customer info</option>
      <option>...</option>
    </select>
  </div>
  <div><button>Next</button></div>
</form>
.
```

Figure 4. Code 1

Figure 5. User interface 2

```
.
.
<form action="#" method="post">
    <h3>Get Customer Info</h3>
    <h5>Please enter the customer ID</h5>
    <div>
       <label>Customer ID:</label>
       <input type="text" name="customer_id">
    </div>
    <div>
       <button>Next</button>
       <button>Previous</button>
    </div>
</form>
.
.
```

Figure 6. Code 2

Figure 7. User interface 3

```
.
.
<h3>Get Customer Info</h3>
 <h5>Customer ID: <b>11</b></h5>
 <h5>Customer info:</h5>
 <table>
    <thead>
      <tr>
        <th>Name</th>
        <th>Email</th>
        <th>Phone</th>
        <th>Register Date</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Ahmad Mohammad</td>
        <td>ahmad@gmail.com</td>
        <td>123456789</td>
        <td>10/05/2015</td>
      </tr>
    </tbody>
  </table>
  <div>
    <button disabled>Next</button>
    <button>Previous</button>
  </div>
```
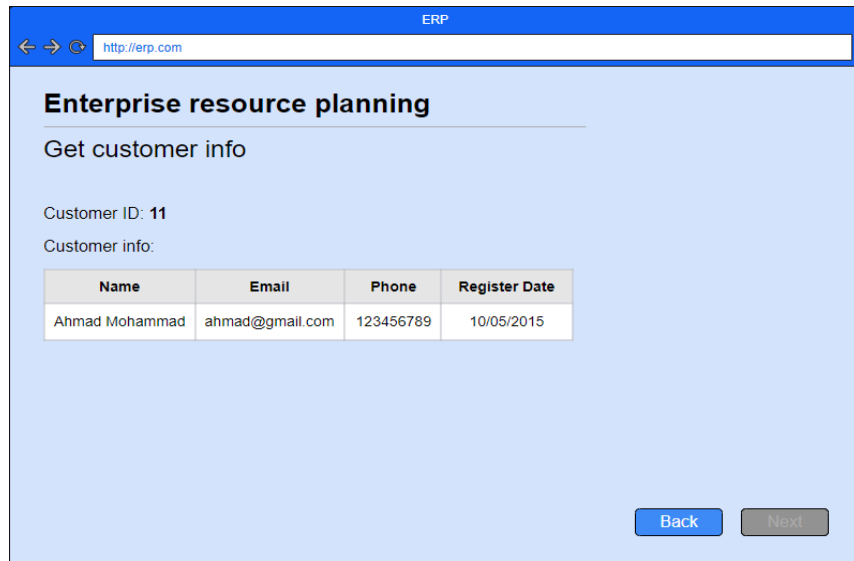
Figure 8. Code 3

Example 1 Back-End (Server) Code:

ERP (Send request to the API):

```php
<?php
$customer_id = $_POST['customer_id'];
$request = sendAPIRequest('get_customer_info', $customer_id);
$result = json_decode($request);
if ($result['status'] == 'success') {
    $customer_info = $result['customer_info'];
    //display customer info table (screen 3 above)

}
?>
...
```

Figure 9. Code 4

API (Receives an ERP request):

```php
<?php
$customer_id = $_POST['customer_id'];
$request = sendRequestToLegacy('get_customer_info', $customer_id);
$result = json_decode($request);
if ($result['status'] == 'success') {
    echo $result; //Send back to ERP
}
?>
```

Figure 10. Code 5

Legacy (Receives an API request):

```php
<?php
$customer_id = $_POST['customer_id'];
$query = "select * from customers where customer_id = $customer_id";
$result = mysqli_query($conn, $query);
if($result){
    $result['status'] = 'success';
    return json_encode($result);
}

return json_encode(array('status' => 'fail'));
?>
```

Figure 11. Code 6

6. CONCLUSION AND FUTURE WORK

In this paper we focused on the interoperability problem in cloud ERP and stated the drawbacks that ERP faces with interoperability. The literature review discussed some of the previously proposed solution for interoperability issues.We proposed a web services generating wizard that

can be helpful in solving cloud ERP interoperability issues. As a future work, we aim to extend the methodology of solving cloud ERP interoperability by combining our results with other research on the field and to consider other useful alternative methods. A possible extension of our work is to build a full version of the solution and make some minor adjustments if needed to improve the functionality of the wizard. Moreover, a set of tests should be establish to provide a good assessment of our solution. Last but not the least, developers should give more focus and attention to interoperability issues when planning and designing any ERP system

REFERENCES

1. Purohit, G., M. Jaiswal, and S. Pandey, Challenges involved in implementation of ERP on demand solution: Cloud computing. International Journal of Computer Science Issues (IJCSI), 2012. 9(4): p. 481.

2. Weng, F. and M.-C. Hung, Competition and challenge on adopting cloud ERP. International Journal of Innovation, Management and Technology, 2014. 5(4): p. 309.

3. Yang, N., D. Li, and Y. Tong, A cloud computing-based ERP system under the cloud manufacturing environment. International Journal of Digital Content Technology and its Applications, 2012. 6(23): p. 126.

4. Wang, S. and H. Wang, A Survey of Open Source Enterprise Resource Planning (ERP) Systems. International Journal of Business and Information, 2014. 9(1): p. 1.

5. ALdayel, A.I., M.S. Aldayel, and A.S. Al-Mudimigh, The critical success factors of ERP implementation in higher education in Saudi Arabia: a case study. Journal of Information technology and economic development, 2011. 2(2): p. 1.

6. Goyal, S., Public vs private vs hybrid vs community-cloud computing: A critical review. International Journal of Computer Network and Information Security, 2014. 6(3): p. 20.

7. Khanghahi, N., R. Nasiri, and M. Razavi, A New Approach toward Locating ERP Components on Cloud Computing Architecture. International Journal of Advanced Research in Computer Science, 2014. 5(1).

8. Xu, X., From cloud computing to cloud manufacturing. Robotics and computer-integrated manufacturing, 2012. 28(1): p. 75-86.

9. Sinjilawi, Y.K., M.Q. Al-Nabhan, and E.A. Abu-Shanab, Addressing security and privacy issues in cloud computing. Journal of Emerging Technologies in Web Intelligence, 2014. 6(2): p. 192-199.

10. Alali, F.A. and C.-L. Yeh, Cloud computing: Overview and risk analysis. Journal of Information Systems, 2012. 26(2): p. 13-33.

11. Fortinová, J., Risks of Cloud Computing. Systémová Integrace, 2013. 20(3).

12. Mather, T., S. Kumaraswamy, and S. Latif, Cloud security and privacy: an enterprise perspective on risks and compliance. 2009: " O'Reilly Media, Inc.".

13. Mezgár, I. and U. Rauschecker, The challenge of networked enterprises for cloud computing interoperability. Computers in Industry, 2014. 65(4): p. 657-674.

14. Boillat, T. and C. Legner, From on-premise software to cloud services: the impact of cloud computing on enterprise software vendors' business models. Journal of theoretical and applied electronic commerce research, 2013. 8(3): p. 39-58.

15. Rimal, B.P., et al., Architectural requirements for cloud computing systems: an enterprise cloud approach. Journal of Grid Computing, 2011. 9(1): p. 3-26.

16. Suciu, G., et al. ERP and e-business application deployment in open source distributed cloud systems. in The Eleventh International Conference on Informatics in Economy IE. 2012. Citeseer.

17. Mijac, M., R. Picek, and Z. Stapic. Cloud ERP system customization challenges. in Central European Conference on Information and Intelligent Systems. 2013. Faculty of Organization and Informatics Varazdin.

18. Gupta, V. and S.S. Bhatia, Developing Assurance Framework of Cloud Computing in the implementation of ERP: A Literature Survey.

19. Arunkumar, G. and N. Venkataraman, A novel approach to address interoperability concern in cloud computing. Procedia Computer Science, 2015. 50: p. 554-559.

20. Rezaei, R., T.K. Chiew, and S.P. Lee, A review on E-business Interoperability Frameworks. Journal of Systems and Software, 2014. 93: p. 199-216.

21. Yu, Q., L. Chen, and B. Li, Ant colony optimization applied to web service compositions in cloud computing. Computers & Electrical Engineering, 2015. 41: p. 18-27.

22. Nacer, H. and D. Aissani, Semantic web services: Standards, applications, challenges and solutions. Journal of Network and Computer Applications, 2014. 44: p. 134-151.

23. Cabral, L., et al., Approaches to semantic web services: an overview and comparisons, in The semantic web: Research and applications. 2004, Springer. p. 225-239.

24. Sheng, Q.Z., et al., Web services composition: A decade's overview. Information Sciences, 2014. 280: p. 218-238.

25. Bertolino, A. and A. Polini. The audition framework for testing web services interoperability. in Software Engineering and Advanced Applications, 2005. 31st EUROMICRO Conference on. 2005. IEEE.

26. Curbera, F., W. Nagy, and S. Weerawarana. Web services: Why and how. in Workshop on Object-Oriented Web Services-OOPSLA. 2001.

27. Avram, M.-G., Advantages and challenges of adopting cloud computing from an enterprise perspective. Procedia Technology, 2014. 12: p. 529-534.

28. Panetto, H., et al., New perspectives for the future interoperable enterprise systems. Computers in Industry, 2015.

29. Romero, D. and F. Vernadat, Enterprise information systems state of the art: Past, present and future trends. Computers in Industry, 2016.

30. Hanna, S., An Approach to Modeling Web Services Datatype Descriptions. Journal of Theoretical and Applied Electronic Commerce Research, 2016. 11(2): p. 64.

31. Narock, T., V. Yoon, and S. March, A provenance-based approach to semantic web service description and discovery. Decision Support Systems, 2014. 64: p. 90-99.

32. Tsalgatidou, A. and T. Pilioura, An overview of standards and related technology in web services. Distributed and Parallel Databases, 2002. 12(2-3): p. 135-162.

33. Metin, S. Using Web Services for WebRTC signaling interoperability. in Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP. 2016. IEEE.

34. Danila, C., et al., Web-service based architecture to support SCM context-awareness and interoperability. Journal of Intelligent Manufacturing, 2016. 27(1): p. 73-82.

35. Bhukya, D.P., R.A. Sony, and G. Muduganti, On web services based cloud interoperability. International Journal of Computer Science Issues (IJCSI), 2012. 9(5): p. 232.