



Cloud-based Semantic Platform for Dynamic Management of Context-aware mobile ERP applications

Hamza Reffad, Adel Alti, Philippe Roose

► To cite this version:

Hamza Reffad, Adel Alti, Philippe Roose. Cloud-based Semantic Platform for Dynamic Management of Context-aware mobile ERP applications. ACM MEDES, 2016, Hendaye, France. 10.1145/3012071.3012076 . hal-02436984

HAL Id: hal-02436984

<https://univ-pau.hal.science/hal-02436984>

Submitted on 13 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Cloud-based Semantic Platform for Dynamic Management of Context-aware mobile ERP applications

Hamza REFFAD
LAS Laboratory University of
SETIF-1
Sétif, 19000 - Algeria
reffadh@yahoo.fr

Adel ALTI
LRSD Laboratory University of
SETIF-1
Sétif, 19000 - Algeria
alti.adel@univ-setif.dz

Philippe ROOSE
LIUPPA Laboratory
IUT of Bayonne
Anglet, 64000 - France
philippe.roose@univ-pau.fr

ABSTRACT

Mobile cloud computing (MCC) has attracted more and more attention from both Small and Medium Enterprises (SMEs). MCC provides access to cloud services for mobile devices. With the growing popularity of smart mobile devices, customers need efficient and new high quality services. In this context, mobile ERP services management and adaptation at real time are becoming a challenging task. However, it is necessary to handle in a dynamic intelligent and transparent platform various mobile cloud services provided by multiple cloud providers with several qualities. This paper presents CloudCxQERP platform, which is a cloud-based semantic platform for dynamic management of context-aware mobile ERP applications. It is based on semantic web technologies and cloud computing and provides fast ERP development process and “an ecosystem” of enterprises that ensures quality customer service. Moreover, our platform provides a middleware that offers to users more efficient services corresponding to current situations and adapts its ERP applications more quickly and cost effectively. Experiments are based on online packaging boxes services.

CCS Concepts

- **Distributed architectures** → **Cloud computing**.
- **Enterprise computing** → **Enterprise information systems** → **Enterprise resource planning**.

Keywords

Context-aware, semantic virtual business services, Cloud, ERP, QoS, multi-enterprises profiles.

1. INTRODUCTION

Mobile cloud computing (MCC) has attracted more and more attention from both Small and Medium Enterprises (SMEs). MCC is an emerging technology that integrates cloud computing technology with mobile devices. MCC provides access to cloud services for mobile devices such as computational resources, computing utilities, scalability. Today, the massive use of new

technologies has led to a dramatic multiplication of a wide range of ERP business services, configurations, redundant context and enterprises profiles. Moreover, there is an increasing usage of heterogeneous mobile devices as well as various customers searching efficient and new high quality interactive services. This could imply that each company needs to manage business processes on various devices. In many cases, this situation could overload a provider. Developing and managing different mobile cloud services for providers are costly. For instance, when a given purchasing service is provided by two or more providers, one company may want a fast transaction time, while another one may want a better price without extra travel and custom fees; which one improve company's satisfaction with relevant service among the provided services.

The objective of this work is to present cloud-based semantic platform called *CloudCxQERP*. CloudCxQERP allows a dynamic semantic management of context-aware mobile ERP applications in order to handle various mobile cloud services provided by multiple cloud providers with several qualities. It provides also more helpful to customers to intelligently provide up the right services to current needs and to current situations (*devices, location, etc.*). A platform for achieving this objective rests on three key elements: first, of course, there has to be an "ecosystem" of enterprises/customers profiles that semantically grouped according to their functional and contextual descriptions. Second, there has to be inference engine constraints analytics to intelligently match up current constraints with cloud semantic services, and finally there is the underlying technology to ensure the deployment or calls services are as efficient and high quality as possible. A platform needs to provide a middleware that offers to users more efficient services corresponding to their current situations and adapts its ERP applications more quickly and cost effectively. The proposed platform is built according to three primary layers. The first generic semantic core layer is a middleware itself and that sits in the Cloud. It provides the semantic reasoning and clustering facilities. Above a middleware, a platform accumulates a network of enterprises, costumers and on-line business affairs between both participants.

The rest of this paper organized as follows: Section 2 presents related works. Section 3 details ontology model. Section 4 describes our adaptation platform architecture, i.e., the CloudCxQERP platform. Section 5 validates our proposal and section 6 concludes the paper with some future works.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MEDES'16, November 01-04, 2016, Biarritz, France

© 2016 ACM. ISBN 978-1-4503-4267-4/16/11...\$15.00

DOI: <http://dx.doi.org/10.1145/3012071.3012076>

2. RELATED WORKS

In this section, we present works directly related to our proposal and point out the differences with our model.

The INFOR publisher [2] provided a semantic connection between some ERPs standards (*i.e. SAP Oracle and SAGE Microsoft*) through its oracle fusion middleware. French publishers with 15 years of experience in Oracle Fusion Middleware, Mobile and Cloud implementations, provided enterprises with scalable and extendable ERP services, including marketing management, customer relationship and online sales. More recently, this publisher was oriented to business solutions in the Cloud "CloudSuite as full SaaS". They didn't talk exactly about cloud model adaptation of the business process to have full control of how business transactions should be received in different devices (*Laptop, Tablet, Smartphone...*) and nor multi-parts enterprises profiles. This publisher provides richer online ERP services but did not take into account the business category semantics, multi-enterprises clustering profiles and QoS (*lower implementation cost, lower time for customer, meet most of enterprises requirements, flexibility to meet with context changes*). Gronlilet al. [5] proposed a cloud based framework for adapting mobile business services in pervasive computing environment by constructing a private mobile business service adaptation agent for each mobile user in the cloud. However, this work hasn't considered the contextual information of customers and enterprises. In addition, this work does not hold the services selection and then, it does not allow providing the appropriate service to a customer. Other works have used cloud computing paradigm as solution for the huge burden on user's mobile computing devices and overcoming obstacles related to the ERP performance. In [1] authors proposed an architecture to publish the business ecosystems of several enterprises, in the center's "cloud" for storage, processing, and distribution. In [4] authors proposed an infrastructure called ERP-Cloud infrastructure which can manage business process on IT infrastructure. In [3] authors proposed mobile cloud ERP based on the adaptive object model for providing business process scalability and reducing hardware costs. This work is more efficient on terms of memory usage. According to [9-10], the context information is indispensable to control the complexity of enterprise systems. Therefore, an enterprise system that uses mobile computing devices which connect over-time to the cloud in more efficient manner in terms of energy, CPU and bandwidth when the enterprise suffers. All these works have proposed interesting solutions for acquiring, storing and managing business affairs using cloud computing paradigm. However, they are constructing a specific business process for each enterprise profile. Business data will be processed in the public cloud and not in private cloud, so if a security attack problem will be met, the business data cannot be in a safe status in the cloud and that leads the security of business transactions. Rekik et al. [6] proposed a cloud description ontology for service discovery and selection. Cloud ontology and semantic techniques are used for modeling a generic customer requirements and contextual cloud services, which enables the system to guide a customer selecting the service that best fits his/her needs in the given context. This work is efficient and shared between different services using OWL as the best candidates to describe and format those services. However, this work uses manual constraints checking and not supports logic inference. They also not provide possibilities for using reasoning

techniques. Gupta et al. [7] have proposed critical organizational success factors for Cloud ERP. They based on market-based view (*i.e. budget of company and ERP project budget*), knowledge-based view (*i.e. economic strategies and implementations strategies*) and social-based view (*i.e. strategic goals and objectives*). They included also QoS criteria like implementation cost, customers' satisfaction, etc. without using contextual information of customers and enterprises which may be as relevant as the required market-based views. Another interesting work is combination of Cloud computing paradigm and optimization techniques [8] to manage resources over the across clients' access. This work based on genetic optimization method for selecting and composing best ERP services. It proposes a generic solution to manage and aggregate of quality business services from multiple providers. This work did not take into account the semantics of services and that of the category, QoS and context providers. Currently, Cloud ERP was developed with standard technologies that meet several qualities of service (*reliability, safety, efficiency*). However, mobility, constraints complexity and contexts of information will not be considered. This implies the following disadvantages: (1) - mobility introduces a degree of heterogeneity and highest dynamicity compared to traditional ERP systems (2) - a poor quality of service of Cloud ERP, consequence of the syntactic description without the context information and (3) - many enterprises profiles cloud has been specified, so many business process have to be computed and updated. In many cases, this situation could overload the provider. The methods and contributions of this paper are different from above related works. Our goal is looking for generic high level semantic abstraction business processes that satisfy the most contextual constraints of both enterprises and mobile customers (*e.g. same business constraints with same device and environment contexts*). Our approach supports deep transition from a generic approach to a more specific enterprise context (s) to deploy personalized business services. Preferences could both QoS and contextual parameters. Our architecture must be able to reason upon contextual business services that comes from different and heterogeneous providers. Therefore, our architecture uses ontology and rules-based approaches to represent and reason upon contextual information, infer the current situation, and select the quality services to the enterprise staff or a customer according to the inferred situations.

3. PROPOSED APPROACH

The main objective of our approach is to identify and reuse the concept of a virtual business process as combination of virtual business tasks for several mobile ERP applications. It allows also the fast identification process of relevant cloud business services and improves the efficiency and accuracy of ERP development process. These objectives are achieved through finite sets of semantic relevant cloud services. The main idea is taking the cloud as a way to group some specific business requirements and cloud services such that enterprises can share experiences, computation resources and bandwidth resources.

A mobile user prefers to adapt ERP services of specific task to his current context. Our goal is to select relevant business service that match user context with low deployment costs and to find new high-quality services to meet continuous companies' requirements and context changes.

To be efficient and shared between different cloud providers, mobile ERP applications must be as interoperable, extensible and laconic as possible. For this, ERP applications are described as OWL ontology. An overview of CxQCloudERP ontology is given in Fig.1. It is comprised of 9 key concepts: enterprise, business process, constraints, entity (*resource*), context entity, QoS, cloud platform, cloud provider, cloud service.

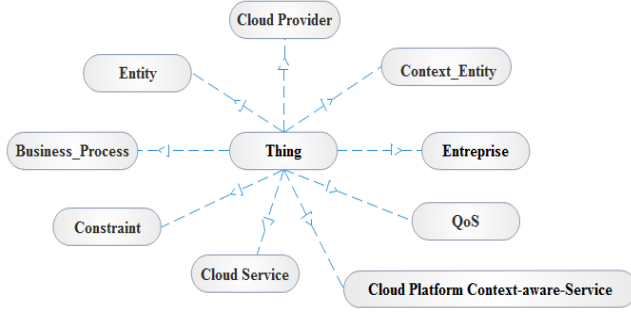


Figure 1. CxQCloudERP Ontology structure.

- *Enterprise* has a name, unique location, functional and non-functional requirements.
- *Business processes* describes various business processes such as sales, purchasing, opportunity, order, etc.
- *Cloud Service* has a name, a version and unique ID (URI).
- *QoS* which describes the quality of service parameters such as efficiency, security, availability, etc.
- *Entity* which describes different types of resources: staff, tasks, equipments, machines, raw materials, etc.
- Context Entity which defines all information that can be collected from the company (needs and preferences), process (deployment requirements and execution) and resource (resource type and constraints).
- *Constraint* which describes a set of OWL logical expressions of various mobile agents and staff.
- Cloud provider that provides business services
- *Cloud Platform Context-aware-Service* is used for the deployment of several business services.

3.1 Enterprise sub-ontology

Fig.2 illustrates the enterprise sub-ontology. The Enterprise concept is a core concept of our CxQWCloudERP ontology. It can be either a customer service enterprise or production enterprise. We propose a new concept (Virtual Enterprise) which includes a family of enterprises. Each enterprise has functional and non functional requirements. A company has no knowledge about the published services. In addition, when a particular service is provided by two or more cloud providers in the same area of a business, a decision should be taken to select the most appropriate service based on contextual description.

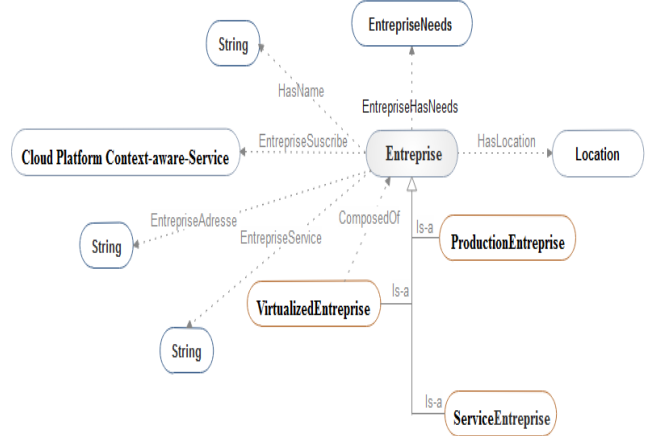


Figure 2. Enterprise sub-ontology.

3.2 Entity sub-ontology

An entity is defined as inputs/outputs of business process. We distinguish two types: physical entity (*resource*) and virtual entity. Physical entities are StorageResource, ComputingResource, DataResource, TaskResource, ActorResource and MoneyResource. Storage resource is disks (*Large, Medium, X-large, XX-large*) and ram (*Large, medium, large X, XX wide*). A virtual entity describes all families of resources in terms of hardware or software.

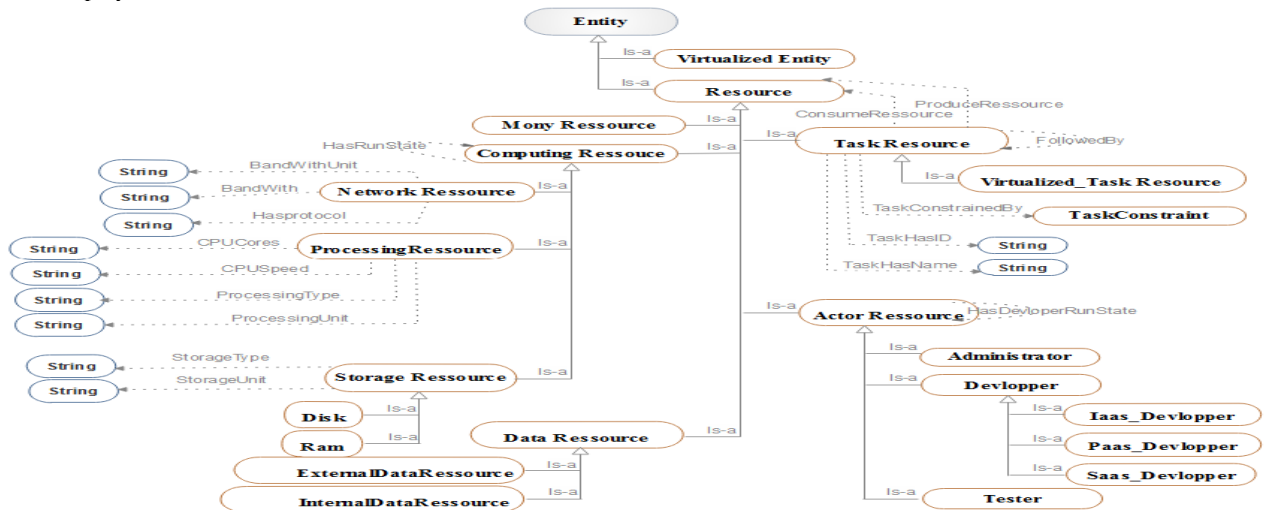


Figure 3. Entity sub-ontology.

3.3 Context Entity sub-ontology

We divided context sub-ontology into four essential sub-contexts:

- The ExecutionContext which describes the current state of equipment (*memory size, CPU load, energy, etc.*).
- The QoSContext which describes the quality of any mobile-based service can be represented on our ontology, which is defined as set of metadata parameters. These QoS parameters are: 1) – continuity of service, 2) – speed and efficiency of service and 3) – safety and security.
- The HardContext is the resources (*mobile devices*) accessible by users such as tablets, Smart phones, etc.

- The Environment Context: which describe spatial and temporal, B2B collaboration informations:

- Temporal information can be a date or time used as timestamp. Time is one aspect of durability so it is important to date information as soon as it is produced.
- The Place describes related informations about B2B collaboration {longitude and latitude}, in a given location, we can find available resources and services.
- The B2B collaboration: according to a B2B domain, a company can engage in B2B activity.

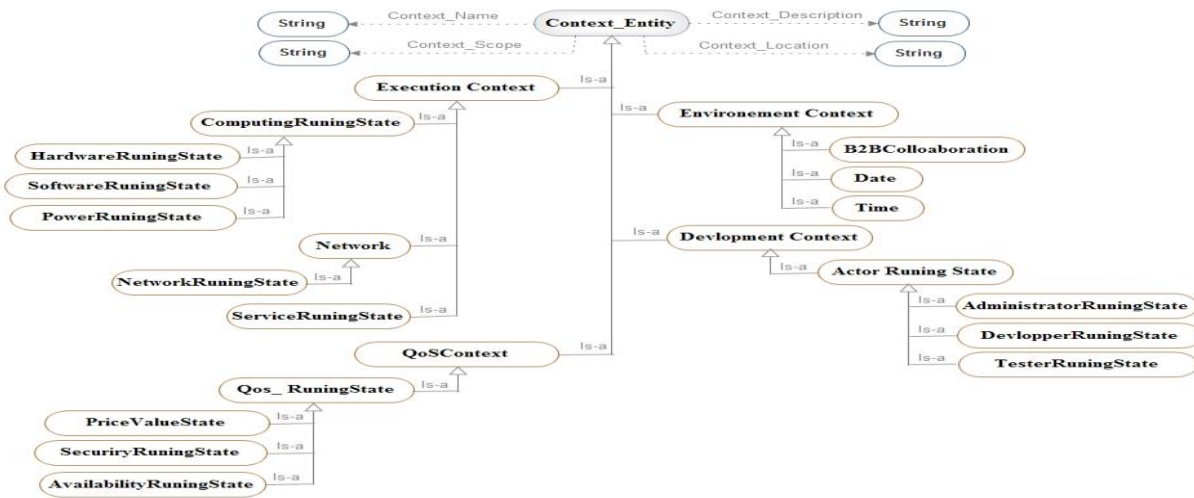


Figure 4. Context-Entity sub-ontology.

3.4 Business process sub-ontology

A business process is a sequence of steps to perform an identified business objective. These steps are actions carried out by different actors and synchronized through business exchanges. It would provide users with an improved services classification in one or more semantic criteria: business domain, business roles (*sales, purchasing, CRM, finance, etc.*), quality of service (*security, deployment and execution time*). We use cloud computing for semantic reasoning and clustering facilities. Our goal is to provide better use of resources in more optimum manner.

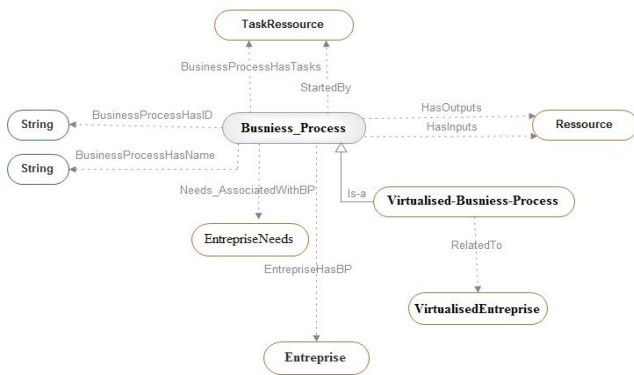


Figure 5. Business process sub-ontology.

3.5 Cloud provider sub-ontology

A cloud provider relates to local and remote data centers. It is a source provider (*servers, storage and network*) in terms of enhanced security, high availability, and confidentiality. Each provider provides three types of services: IaaS_Service, PaaS_Service, SaaS_Service in three modes: private, public and hybrid. When a given service is provided by two or more services providers in the same mobile client area, a decision should be taken to select the most appropriate service based on contextual description with a better quality of service. Depending on the availability of resources for the mobile device, our aim is the best service from the list of services with equivalent functionality. Specifically, in case of changes context, we dynamically switch to the next service in the list of services offering the same service functionality and requires lower resources.

3.6 Cloud service sub-ontology

The Service Cloud is made up of four sub-classes: three traditional services IaaS_Service, PaaS_Service, SaaS_Service and now virtual cloud service. Fig.5 shows following concepts:

- Service has a name, a version and unique ID (*URI*).
- Service_Location: the location which the service facilities are located.

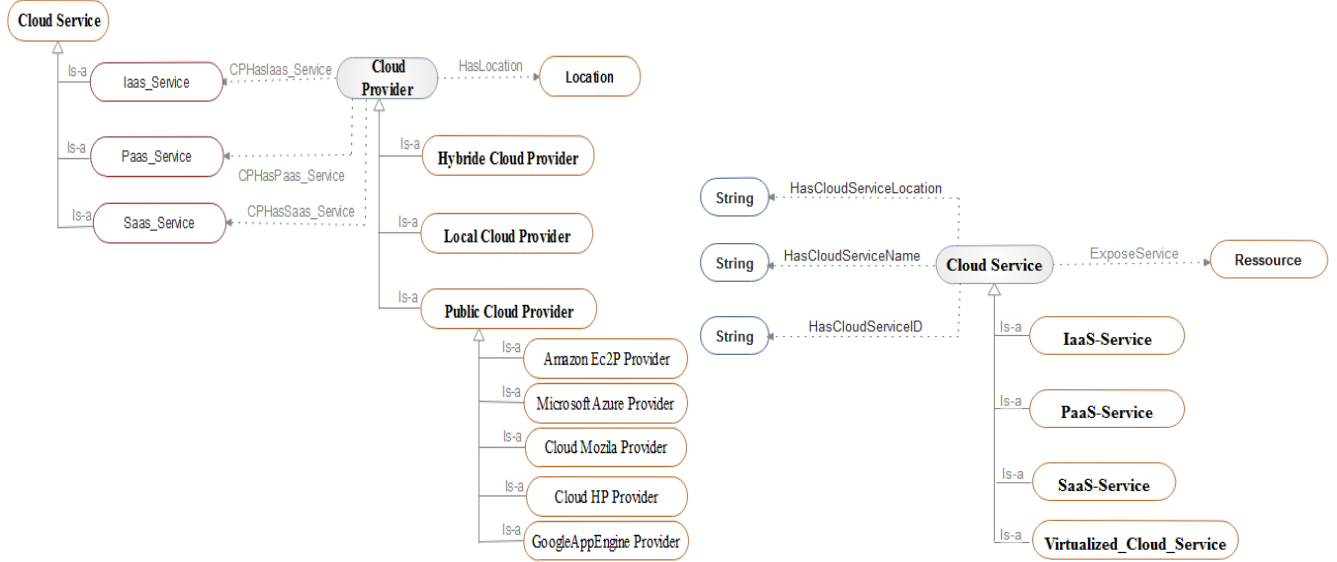


Figure 6. Context sub-ontology (a) Cloud provider sub-ontology (b) - Cloud service sub ontology

- Service_Category = {SaaS, IaaS, PaaS, virtual service.}.
- Service_Type = {local, remote, hybrid}.
- QoS_parameters: list of QoS parameters. This concept does not only relate to services but also the mobile device resources where business activity takes a place.
- Service_parameters: Each service category has its own parameters and service state. For instance, the PaaS service related to the database platform, implementation languages and runtime tools, etc. SaaS has a name, version, role, unique ID (*URI*), location and execution time, business category (*CRM, sales, shopping, production, finance, etc.*). The IaaS service has a capacity, computing and communication resources, etc.

The Virtualized Cloud Service allows mobile customers and staffs for improving the efficiency of a business process and hiding heterogeneity of platforms in one or more criteria: planning and scheduling strategies, federation of business partners, routing paths, business monitoring and service migration.

3.7 Constraint sub-ontology

Each context entity has its context parameters. For example, the context related to the hardware resource is a collection of parameters: machine speed, min/max of consumption. Certain context parameters may use semantically similar terms, for instance the speed of the machine and the number of units produced / minute.

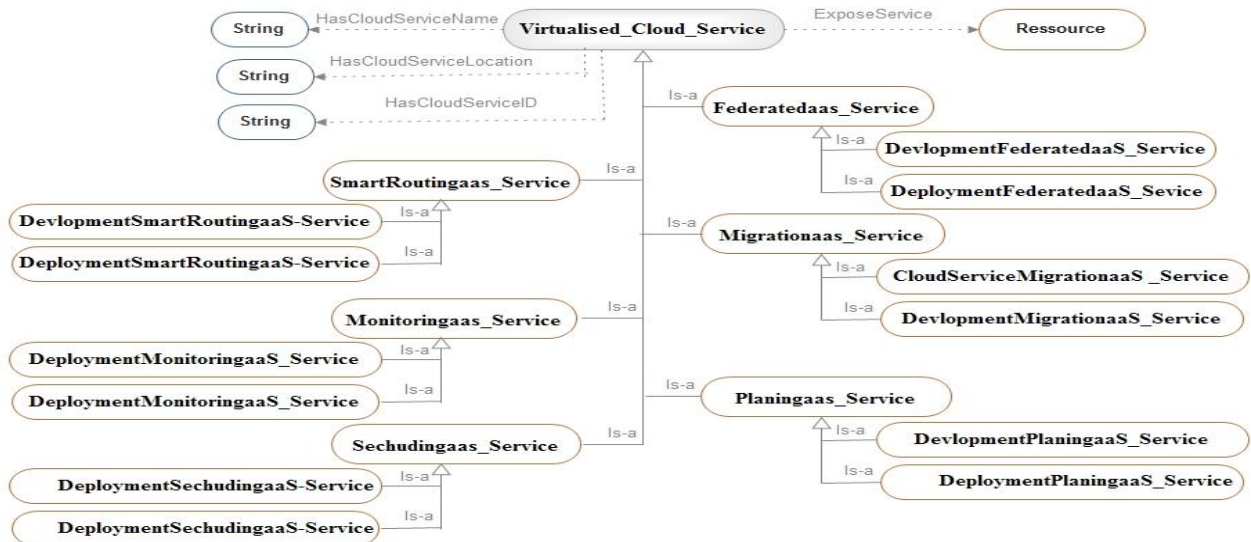


Figure 7. Virtual cloud service ontology.

4. The CloudCxQERP PLATFORM

4.1 The general architecture

Here is a general overview of a cloud-based semantic context-aware ERP architecture (Figure 8):

- *Enterprise customers*: this is the device that the customer is currently using.
- *ERP Cloud Proxy*: it stores or serves as an ERP Cloud proxy for cloud business services and ERP resources. It also executes cloud business services. This component relies on the following sub-components: (1) - the *Semantic Customer Context Monitor* is responsible of verifying user's context change by contacting the *Cloud Context Management component*, (2) - the *Semantic Enterprise Context Monitor* is responsible of verifying enterprise resources and context changes and (3) - the *ERP Inference Engine* supports the building of personalized business process from available services.

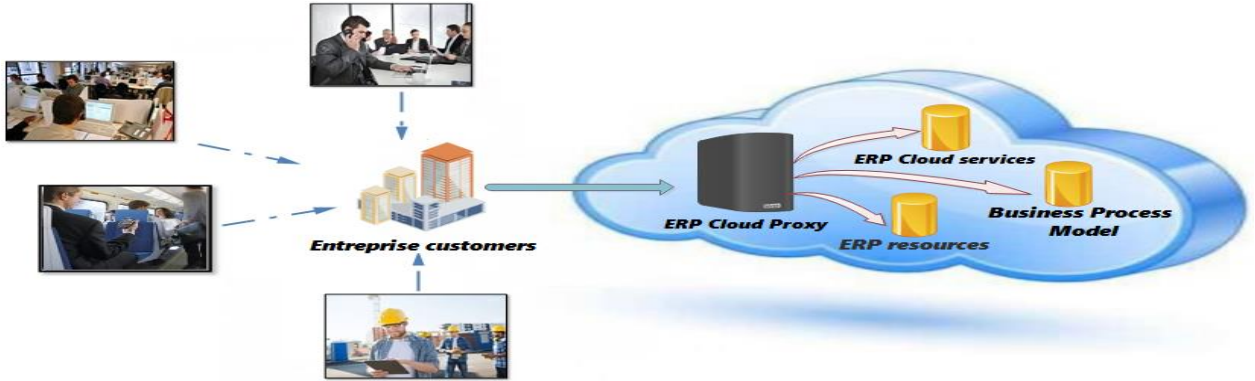


Figure 8. The general architecture of CloudCxQERP platform

4.2 Functional model of the platform

We have specified the development process that exploits our ontology in order to generate several business processes according to the mobile users and their current context. This process uses a semantic inference engine that is responsible of planning and optimizing the selection, the deployment and the execution of business processes based on a customizable quality of service. In order to accelerate the development process and to better discover and compose cloud business services, we propose to enhance the expressiveness of the semantic descriptions of cloud business services, specifically to provide the "best" business service composition according to semantic properties based upon common functional and non-functional needs provided by specific companies. Our work provides a semantic description of cloud business services (e.g., *name*, *location*, *role*), allows a dynamic selection of cloud business services based on context metadata parameters (user, mobile device, service provider). The process of selection and dynamic composition of ERP processes is a 7-step process:

Step.1 – Business requirements clustering: The first phase is to cluster into common constraint similar enterprises based on their functional and non-functional descriptions. Therefore, the service discovery component matches set of services that share similar functionality with a most of the target enterprises needs. The clustering and classification of new services are performed offline

- *ERP Cloud Registry*: stores the description of Cloud business services, business process patterns and ERP resources of all providers in the Cloud. When a Cloud provider is active, it sends the description of all its cloud services to the ERP proxy. Therefore, the ERP proxy has the descriptions of all services hosted on providers involved in the ERP application, and the ERP generic composition algorithm is executed. In the next section, we will detail the algorithm.

When the ERP proxy receives these informations as XML files, it calls the ERP Inference Engine to cluster possible constraints of equivalent customers. This later researches for Cloud business services able to make partial or full generic services composition that satisfies most of the target customers' constraints. The algorithm gives all the possible chains satisfying the most costumers' constraints. The ERP engine selects the best chain. We use various politics based on the weight (*energy saving*, *customer profile*, *low band-width*, *CPU charge etc.*) of each chain.

to eliminate any overhead during service discovery. The goal is to accelerate the search process of relevant cloud business services and to reduce overall response time. The idea of the requirements clustering is to take into account the description of the functional and non-functional business requirements of two or more enterprises. Two descriptions are semantically equivalent if all the matching values between parameters are exact. We propose a clustering method based on inference rules to cluster common needs of different enterprises. If no similarity has been detected, go to step 6. Otherwise, if some similarities have been detected, it produces a common requirements guide containing common requirements. The inference of common needs of several companies expressed in SWRL as follows:

```
Common_ComplexConstraint (?cex) ∧
BusinessType(?cex, "Buy") ∧ Enterprise(?e1) ∧ Enterprise(?e2) ∧
EnterpriseName(?e1, ?name1) ∧ EnterpriseName(?e2, ?name2) ∧
swrlb:notEqual(?name1, ?name2) ∧ EnterpriseHasNeeds(?e1, ?en1) ∧
ConstraintExpressedBy(?en1, ?cex1) ∧ EnterpriseHasNeeds(?e2, ?en2) ∧
ConstraintExpressedBy(?en2, ?cex2) ∧
ExpressionComposedOf(?cex1, ?sex1) ∧
ExpressionComposedOf(?cex2, ?sex2) ∧ ConstraintID(?sex1, ?IDs1) ∧
ConstraintID(?sex2, ?IDs2) ∧ swrlb:equal(?IDs1, ?IDs2) ∧
BusinessType(?cex1, ?type1) ∧ BusinessType(?cex2, ?type2) ∧
swrlb:equal(?type1, "Buy") ∧ swrlb:equal(?type2, "Buy")
→ ExpressionComposedOf(?cex, ?sex1) ∧
ExpressionComposedOf(?cex, ?sex2) ∧
EnterpriseSet(?cex, ?e1) ∧ EnterpriseSet(?cex, ?e2) ∧
IsCommonConstraint(?sex1, "1") ∧ IsCommonConstraint(?sex2, "1")
```

Step.2 – Business tasks semantic matching: The task manager component matches between the virtual task and the common constraint properties. The semantic matching process exploits the categorization of the task hierarchy to find suitable matches. For that purpose, it matches the virtual task properties (*task category, location, time, QoS parameters, task time, inputs/outputs, etc.*). Each constraint can be expressed as an AND-OR clause of a simple expression, like the following one: (*taskCategory = 'Buy' AND Price = 'Low' AND location = 'Sétif' AND TransactionType='Internet'*). The inference engine brings out set of business virtual tasks.

Step.3 – Finding relevant context-aware virtual cloud services: After receiving the list of corresponding virtual tasks that match the most target mobile users' requirements. A cloud-based semantic service selection mechanism is based on both service semantic information and mobile users' context (e.g., *less free memory, less processor power, user location and available wireless protocols*) or the environment context (e.g., *less bandwidth*) or the mobile users' preferences. Relevant cloud business services with higher potential benefit are selected earlier and find a limited number of composition solutions to reduce the computation complexity. The following rule represent an example of inference rule which serve to answer the semantic compatibility of candidate cloud services which implements the virtual business task with the most target mobile constraints.

```
ComplexConstraintExpression (?cexp) ∧
ExpressionComposedOf (?cexp, ?sexp) ∧
IsCommonConstraint (?sexp, \"I\") ∧ LogicOperator(?sexp, ?logicoperator)
∧ NumericValue(?sexp, ?numvalue) ∧
SemanticValue (?sexp, ?semanticvalue) → sqwrl: selectDistinct (?cexp,
?sexp, ?parametrename, ?logicoperator, ?numvalue, ?semanticvalue
```

Step.4 – Generating optimized virtual business processes: This step consists in constructing all possible composition between virtual services based on the list of virtual services of Step 3. We try to find the next service by checking the services available. If the number of services is very large, at each step, we must check a lot of services. To reduce the checking time, we define the relation between a service and another one. This is based on semantic relationships between services; equivalent ≤ substitution those are used to implement rules based on their semantics. This step returns a list of composition paths.

Step.5 – Business Process refinement: Refinement is necessary to address some customization of relevant contextual parameters in order to fulfill specific needs of enterprises. For instance, one enterprise may want a fast transaction time with quality local raw materials providers, while another one may want money saving with an average quality of raw materials. The refinement component is responsible for adding/replacing specific business services for each business tasks

Step.6 – Selection of quality business processes: The goal of this step is to select a better optimal business processes from the primary process models identified in the previous step. We select the optimal business process in terms of cost with the best quality service. The process decision engine ensures the selection of the best business processes from the graph if there are several services provided by two or more cloud providers.

Step.7 – Process Execution: ERP proxy deploys cloud services and sends the result to each customer

5. A CASE STUDY AND EVALUATION

This section describes implementation details of our prototype. Firstly, based on OWL and SWRL languages, we have implemented our ontology and rules-based Cloud context model using Protégé tool [12]. Then, we use NetBeans IDE for development of our prototype for dynamic semantic service clustering and dynamic semantic service context monitoring.

Our prototype is based on three main parts: the first is a context monitor to simulate the semantic service behavior (*price, cloud provider, execution time*). The second part is contextual clustering of customers' and enterprises requirements. The last part is the dynamic business process management according to customer requirements. Our prototype generates virtual business process of both companies (*AGROFILM and SYNSET*) and offers to users relevant services to his current situations. Table1 illustrates requirements of a customer and Table2 shows requirements of both companies. Figure 9 shows an abstract business process diagram from a both companies' requirements and constraints of a customer. Each transaction involves five parts: a customer, order, manufacturing, a payment and a delivery gateway. The customer firstly configures it session by entering order details, browsing products informations and deciding whether to view more products information. Then the customer makes order by selecting packaging boxes models as well as filling in customer information and card credits details. The customer finally confirms the order and sends it to two companies (*AGROFILM and SYNSET*) and waits for a delivery response. Once the order is received, the delivery deadline is fixed for a customer and the collaborative manufacturing process started in both companies in order to fill a customer order. In parallel, companies purchase a supplier for raw materials in which the packaging boxes will be manufactured and the payment gateway verifies the credit card details. Depends on the returned status, a reply is either by email or phone call. Finally, companies deliver products to a customer in a specific location. The treatment of both companies' requirements creates an abstract business process. Some specific situations of customer (mobility of user, device resources) and/or a firm (equipment failure and no availability of raw materials) may be as relevant as the required business affairs. Several scenarios are experimented.

Table 1. Contextual constraints specified by a customer

Location	Quality and Contextual constraints
Any	Shortest delivery time and high-quality packaging boxes
Near a Company	Receives 10% discount based on a large quantity of packaging boxes without extra travel fees and notification by email or phone call
Car	Notification only by phone call and prefers high-security transaction

Table 2. Clustering contextual constraints of both companies

Company	Quality and contextual preferences
Both	City= SETIF AND Type=Deposit
SYNSET	Availability=High AND Supplier=Local AND Price=Low
AGROFILM	TransactionType=Confidential AND Supplier =Any AND Payment=Fast

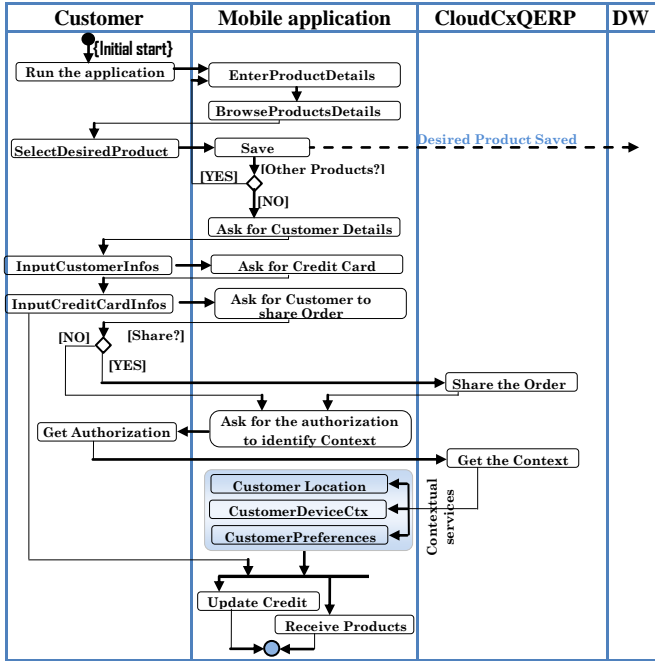


Figure 9. An abstract collaborative business process.

A First scenario: a customer in a Car using a Tablet. A customer cannot read his personal emails in a Car. The Semantic Customer Context Monitor has attached a discovery protocol and GPS service as client service. When a delivery time comes, the ERP Inference Engine reinitiates the composition process, inferred actions that of discovering the available text to speech services are supported by the user location: : update Text email service with Text to Speech service. After finding the adequate services from cloud and after its selection and integration, the user can follow the email service while taking a Car. When a User asked for the fast payment, the ERP Inference Engine discovered two payment types: a credit card and a bank transfer. The Inference Engine deployed the second one because absence of security factor and unavailability of Wi-Fi.

A second scenario: delay of raw materials reception. When a manufacturing process started in a SYNSET firm, a notification was sent to the Semantic Enterprise Context Monitor that there is not enough available raw materials from supplier to continue displaying such a customer order. In this case, ERP Inference Engine should evaluate another available semantic equivalent services paths but with different QoS in order to replacing supplier service at run-time. We are used [11] for classifying the relevant cloud services that have potential benefit. Table.3 shows the evaluation results, meaning that services orchestration “RawMateriel_eMail_SupplierLocal + PaymentOrder_Local” is selected as the best composition path. This selection based on the high score compared to other paths.

Table 3. Evaluation results

Services composition paths	Score
SaaS_Service_RawMateriel_eMail_SupplierLocal >> SaaS_Service_PaymentOrder_Local	0.79 ms/\$
SaaS_Service_RawMaterielSupplier_WebSite >> SaaS_Service_InputCustomerCreditCard	0.65 ms/\$

6. CONCLUSION

In this article, we have proposed a new hybrid approach based on semantic web technologies and cloud computing. We propose also CloudCxQERP platform, which is cloud-based semantic platform for dynamic management of context-aware mobile ERP applications. The main objective of this platform is the selection of relevant services and adaptation of ERP applications more quickly and cost effectively. In future works, we intend to refine costumers' requirements. Moreover, we will develop heuristic parallel genetic method at design and at run-time phases.

7. REFERENCES

- [1] A. Paya, D. Marinescu: Energy-aware Load Balancing and Application Scaling for the Cloud Ecosystem, *IEEE Transactions on Cloud Computing*, (99):2168-7161, 2015.
- [2] EuroCloud: Cloud Letter, Knowledge Base of Cloud, special Issue of ERP in Cloud Computing, pp. 16, 2014. DOI= <http://www.infor.fr/solutions/technology/cloud/>
- [3] Y.E. Gelogo, H.K Kim: Enterprise Resource Planning Asi-pacific *Journal of Multimedia Services Convergence with Art, Humanities and Sociolog*, 3(1):1-8, 2013.
- [4] K. Mohan, S.S. Shetty: Cloud Computing and Enterprise Resource Planning (ERP) Systems in Cloud Environment. *International Journal of Advanced Research in Computer Science and Software Engineering*, 5(4):721-726, 2015.
- [5] TM. Gronli1, G. Ghinea, M. Younas and J. Hansen: Automatic Configuration of Mobile Applications Using Context-Aware Cloud-Based Services. *Modeling and Processing Big-Data Technologies*, pp. 367-383, 2015.
- [6] M. Rekik, K. Boukadi, H. Ben-Abdallah: Cloud Description Ontology for Service Discovery and Selection, *In Proc. of the 10th Inter. Conf. on Soft. Eng. and App. (ICSOFT-EA'2015)*, pp. 26-36, 2015, Colmar, Alsace, France.
- [7] S. Gupta, S.C. Misra Implementation of Cloud ERP Moderating Effect of Compliance on the Organizational Factors. *In Proc. of the 5th Int. Conf. on Cloud Computing and Services Science (CLOSER-2015)*, pp. 194-198, 2015.
- [8] C.S. Chena, W.Y. Liangb, H.Y. Hsu, A cloud computing platform for ERP applications. *Applied Soft Computing*, Vol. 27: 127-136 (2015).
- [9] M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies: The Case for VM-Based Cloudlets in Mobile Computing, *IEEE Pervasive Computing*, 4(8) :14-23, October 2009.
- [10] M. Rekik, K. Boukadi and H. Ben-Abdallah: A decision-making method for business process outsourcing to the cloud based on business motivation model and AHP, *Int. J. Cloud Computing*, 4(1):47-62, 2015.
- [11] A. Altı, S. Laborie, P. Roose : USSAP: Universal Smart Social Adaptation Platform, *In Proc. of the 6th Int. Conf. on Ambient Systems, Networks and Technologies (ANT 2015)*, *Procedia Computer Science*, pp. 670–674.
- [12] Protégé. 2007. Why Protégé. <http://protege.stanford.edu>