

Advanced Cloud Document System

Alberto Buschetti¹, Filippo Eros Pani² and Daniele Sanna¹

¹ Experteam srl, Spin-off of University of Cagliari, Via Zara 11, Cagliari, Italy
{alberto.buschetti, daniele.sanna}@e-xperteam.com

² Department of Electrical and Electronic Engineering, University of Cagliari,
Piazza d'Armi, Cagliari, Italy
filippo.pani@diee.unica.it

Abstract. Nowadays, public administration offices are faced with long and complex procedures. Frequently used tools such as email clients and document systems are seldom integrated. This leads to problems like slowness, large quantities of paper documents, redundancies, high operational costs, and a low level of citizen participation. Google Apps try to fill that gap by providing an integrated and usable environment, but it is not enough for typical public administration applications, where the existing IT system has to be taken into account. The aim of this project is to create a Software as a Service (SaaS) platform, based on open-source components and integrated with Google Apps. With such a platform, public institutions and companies could use an environment in which to work on the most frequent tasks. From an innovative point of view, the project uses the latest technologies and the latest patterns in both planning and development.

1 Introduction

The large work volume and the limited time in which public institutions need to provide their services represent a difficult issue to solve. The computerization of services means the reduction of paper documents with time, leading to a substantial step forward. The tasks that an organization has to perform are supported by the use of suitable software, such as email clients, document systems, work-flow managers, online publishing systems, etc. Users access these tools daily, but there are apparent limitations that hinder the improvement of the efficiency of a whole organization (either public or private). This is due mainly to the lack of integration, to the usage of non-open formats in interchanged data and to the impossibility of remote access to the work tools.

The proposed project aims to create, even as a prototype, a typical Cloud software system that can integrate already existing systems with the ecosystem of Google Apps¹. That would help the resources employed in daily work activity, improving the general efficiency by leveraging on the potential offered by Google Apps, such as their remarkable capability for document circulation among users in environments

¹ Google Apps, <https://www.google.com/work/apps/business/>

that use email clients and document managers. The advantages are outstanding, including the improvement of the interaction between public offices and other institutions as well as of its communication with citizens, creating an actual Virtual Organization.

The software system at the core of the project is categorized as SaaS, and represents the latest technology in software use and supply. The spirit of the project is to pursue the creation of software artifacts by using typical approaches of Open Source. A project of this kind, making use of innovation in software development through the use of non-traditional models, also needs a new approach to development. Namely, the project includes the use of an Agile Methodology derived from Kanban [1], given the experience in that context gained by the proponents, in particular by the Agile Group² of the Department of Electrical and Electronic Engineering (DIEE) at the University of Cagliari.

2 Context of Research Proposal

The project contributes to a substantial advancement to the state of the art in the field. It follows the logic of the process management issues in Service Oriented Architectures (SOA), with an integration to the management of the documents involved in those processes and the management of the data coming from the process together with the documents themselves. At a scientific level, this kind of industry research follows the approach proposed on Service Oriented Computing, Kanban methodologies [2], and evaluation methodologies for software solutions [3]. It focuses, however, on their usage in workflow processes that rely on a Cloud-based Web Service architecture, and pays specific attention to interesting themes such as the integration of a document logic and the management of end-to-end security. The problem of security in applications and data in a SOA will be addressed by taking into account solutions like WS-security. Another aspect that will be under investigation is the study of a system allowing the optimal management of all applications involved in these processes and elaborated in this SOA. That interaction could be described and formalized also through Business Process Management (BPM) solutions, which would allow for an efficient management and structuring of the information pertaining to analyzed processes and the compliance of the tool to prominent market standards like Business Process Modeling Notation (BPMN) and Business Process Execution Language (BPEL).

It is known that a large part of the data owned by organizations can be found non-structured documents (editable or non-editable text, pictures, videos, etc.), while the current systems, including integrated “data warehousing” systems, are based on structured data belonging to one or more database. A document-centered approach could guarantee the usage of the whole wealth of document, adding in-text search systems, semantic marking, etc., and represents a hot topic in research.

The system adopts an approach that has become a standard in many situations, that is the use of web-based software as a service (SaaS) in Cloud mode. Google can

² Agile Group, DIEE, University of Cagliari, <http://agile.diee.unica.it/>

be said to be the pioneer of SaaS of this kind, creating services like office automation, document management, calendar, web-mail, etc., currently widespread in public and private organizations.

In Cloud systems, the computing paradigm is no more valid, and certain approaches, namely object-oriented design, do not find a place. In fact, in this field it has been replaced by “service-oriented” design.

Another successful approach is the Model Driven Architecture (MDA) [4], based on platform-independent models, which is especially suited in literature for an Agile approach based on Lean-Kanban [5]. The use of this model has remarkable implications in terms of interoperability, a crucial aspect in environments such as True SaaS. It is in the nature of an application that works on the Cloud to possess a high interoperability, which needs to be planned starting with its design when an MDA approach is at its natural place.

The literature has defined some best practices for the creation of a SaaS, whereas research and experience made it possible to find some fundamental points. First of all, software customization tends to be replaced by configuration. In fact, SaaS feature highly configurable, multi-tenant applications. This characteristic is very well finished, so much so that Automatic Tuning (AT) [6] and Dynamic Adaptive Systems are emerging. They allow an automated creation of the optimal configuration for each user (tenant) or group of users. Its modular nature, its templating functionalities and its Role Access Management (which allows to assign specific access policies to users, together with function and resource use policies) are important features for software subject to configuration. The functionalities of SaaS are distributed on different service levels, where each service level to be assigned to a functionality takes into account the use of cloud resources, the business value of the functionality, and of a possible purchase method. Currently, the SaaS that provide an integrated environment where email, document manager, office automation suites and other tools such as certified email, electronic signature, fax, etc. can be used are few, and are usually released as desktop applications. Integration is therefore a key element: according to Forrester Research [7], the main reason why organizations do not adopt SaaS lies in problems with the integration of in-house solutions with the SaaS, especially for real-time application communication.

Another important aspect of the proposed research concerns the comparison with other tools in existence, namely Cloud and non-Cloud. NVAF and its extensions [3] are valid frameworks to make a comparison without discrimination, using metrics taken from scientific literature.

3 Description of Research Project

The purpose of the project is to create an integration solution for typical Cloud tools such as those provided by Google, a solution that would use services and Apps like Google Drive, Google Documents, Google Maps, Gmail, Google Calendar, Google+, etc., to offer the necessary software equipment to circulate documents in an integrated environment from/to other organizations, in a perspective of Virtual Organization. With that Cloud approach, users can provide services without the use of desktop

software, but directly through the Cloud, speeding up their tasks and making data and documents immediately available. This can be applied to typical use cases in the interactions between a citizen and a public office, or between two public offices. It would mean to provide a solution that integrates tools with which the user is familiar, with other tools that add legal standing to the communication (in particular, an integration between tools like certified email, electronic signature, Italian National Service Cards (CNS), fax, IT protocol, but with a more familiar interface for a digital native citizen). It is an area where traditional issues related to the introduction of new technologies merge with new issues related to legal risks in data storage and potential reduction of the costs on competing proprietary software, in a context of continuing variation and advancement of information technology.

The project covers the creation of software artifacts through approaches that are typical of Open Source, and using the tools provided by that paradigm in order to apply the state of the art to the design and implementation of the system.

To create such a project, with its particular problems related to software development on the Cloud, something that differs from traditional models, that is an Agile Methodology derived from Kanban, will be experimented, in light of the experience gained by the proposers in this field.

3.1 Project Subdivision

The project includes a number of operational stages, which involve:

- the analysis and evaluation of the state of the art in the industry relevant to the project (available technologies and services, competitors, etc.);
- the verification of the options in the definition of the software development tools in this type of application-service;
- the evaluation of industry risk and the selection of development options;
- an in-depth study of the selected technology;
- the definition of the basic services to be developed;
- the definition of the integration architecture;
- lastly, the development and validation of a demonstration prototype.

3.1.1 Evaluation of State of the Art

The initial activity entails the study of a broad range of needs related to innovative Cloud applications, in order to perform an appropriate selection and focus on the developments of higher impact. One of the possible options for technology is the use of Google App Engine, the hosting and development platform that allows to create high-traffic web applications without the need to handle their issues. This environment provides a number of development tools: Google App Engine SDK for Java, Google App Engine SDK for Python, and Google Plugin for Eclipse.

A second option involves developing the software in J2EE architecture using the API provided by Google to make its integration possible. Many software houses with a presence on Google Apps Marketplace offer their applications and services on their own portal, in addition to the one provided by Google. Reference technologies will be

picked for an early selection of the architectures for the support system to the development of the services. The following operations will be carried out:

- the evaluation of available technologies and the comparison with the use of a support framework;
- the verification of active services and competitive systems.

3.1.2 Verification of Software Development Tools and Study of Technologies

This activity is based on an innovative Lean approach to software development in a Cloud environment, also in distributed and collaborative environments. Cloud practices will be also used, according to the idea: “developing software for the Cloud using the opportunities offered by the Cloud itself”, to determine some relevant modules in a standalone environment. Another subject of investigation is the risk analysis of the use of Cloud solutions of this kind. In fact, one of the factors that acted as a deterrent to the adoption of Cloud solutions was, until recently, the risk stemming from storing company data in third-party remote systems, and the risk of uncontrollable downtime. But the Cloud does not only offer the advantage of online storage: it also allows to create new mechanisms of communication and collaboration.

A study was conducted on Google’s environment to analyze its first-level characteristics, such as its data exchange format, its exposed functions, and its authentication method.

3.1.3 Definition of Applications and Services

The issues related to the Cloud, and the effectiveness of the solutions currently available in the market, will be studied, also considering the developments expected in the medium-long term. This phase of the research project will have a scientific character, but also an empirical one, since it requires the knowledge of market orientation and of potential competitors to the current developments as well as of future ones. The activity is thus made of the analysis of the reference market and of the services offered by competitors, the definition of priority requirements of the services, and lastly an evaluation of the evolution expected in the medium term.

3.1.4 Definition of Solution Architecture

This activity aims to study valid solutions and design a technology infrastructure that overcomes the basic issues together with the application issues. The Cloud Computing paradigm needs to be taken into account, from both a technological point of view of cooperation between applications, and from the logic of cooperation between applications and data, which could come from different systems and entities. The studied infrastructure will have to be able to manage document flows supporting an effective engine to manage applications, other than a presentation through Google Apps, in a SOA context.

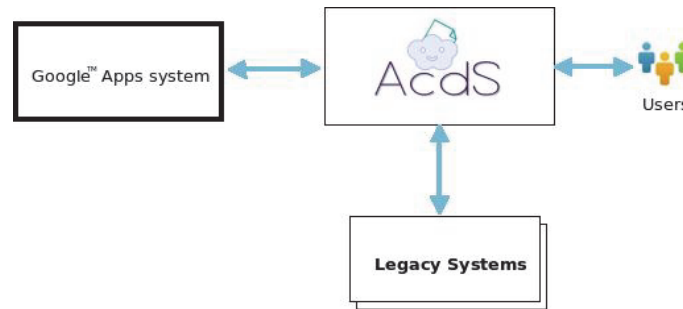


Fig. 1. General structure of the solution architecture.

3.1.5 Prototype Development

In this phase, the software components of the systems will be designed, created and validated, in prototype form. The development will require specific modules that will satisfy the need for modularity, interoperability, flexibility through the use of open standards. Particular care will be given to the building of a prototype that would solve the complex issues raising from the interaction of distinct, yet interdependent, processes in applications.

Another aspect that will be tackled at a prototypical level is the management of the security integrated in SOA supporting the Cloud, in a dynamic form, especially concerning access authorization. An architectural pattern of API Gateway³ type will be also studied. Such a component provides an intermediate layer that allows the platform components to perform requests to the Google Apps API through calls that have a preset name, independent from Google. The internal routing system, with its rules, allows the transparent use of the correct Google API.

4 Project Schedule

The project officially began on March 3, 2014, and its conclusion is estimated for the end of December 2015.

The current phase is the implementation of the prototype covered by the project. Special attention was given to the study of Document Management and interoperability between systems, as well as of integration techniques. That is a crucial aspect, since the adoption of complex software means having to face existing software systems with which dialogue is needed. Interesting results came to light in the comparison among software systems, which was necessary to evaluate the potential of competitors and of the Open Source software adopted in the project. SaaS is actually a new domain, which requires new approaches and new metrics compared to traditional software.

³ API gateway, <http://microservices.io/patterns/apigateway.html>

4.1 Interoperability Issues

The topic of interoperability is fundamental in environments such as True SaaS [8][9]. An application that runs on the Cloud must necessarily have a high interoperability. This feature needs to be planned from the design stage, with an MDA approach and the use of open technologies, as it was found during our study. It is a neutral and open approach for the development of enterprise applications, where modeling leads to the development of the software. MDA encourages the evolution of solutions through subsequent transformations, from high-level models to low-level models, down to the generation point of the code. Due to the considerable impact coming from the sudden changes related to the alternation of new technologies, the creation of Cloud services with a specific technology raises some issues. In fact, even when the same Cloud Service Provider is confirmed, it could update its own technologies. Therefore, MDA makes it possible to develop technology-independent Cloud services where the business logic does not depend on technical details.

4.2 Google Apps Integration

An analysis of competitors was performed, especially of those operating in the ecosystem of Google Apps. The analysis highlighted the great interest many companies have for applications completely based on the integration with Google Apps. The apps that resulted to be most interesting were Google Drive and Gmail. A large part of the systems created by competitors seek the highest integration with those applications, exploiting their potential. One kind of widespread application in terms of Google Apps usage is a Workflow Management (using BPMN) that uses Google Drive as document storage and email as an alert system on the various tasks that make up a workflow. This meant a further confirmation of the validity of the project, and very good prospects for the company that hosts it. The same analysis sheds light on the strong effort from competitors to port the main functions onto a mobile environment.

Real case studies were evaluated in order to set apart the functionalities that would have the biggest impact on a public institution. The research made it possible to review the best practices and the patterns on development in a Cloud environment, giving the opportunity to be based on the state of the art in both design and development. The development of this type of applications requires some features that may not exist in traditional desktop applications [10], such as multitenants, that is the presence of all the users who share the same instance of the application.

Processes represent a fundamental aspect for an organization, and many companies do not have any process management systems. We therefore considered that as a priority functionality to implement.

A SaaS application can improve existing processes, or create new ones. The possibility for users to operate on the move could open new scenarios, which would be impossible were its availability limited to the company offices only.

A process manager is one of the software systems that most require a strong interoperability between systems. Consider, for example, the recurring need processes have for email, document resources, accountancy data, etc.

4.3 Architecture Choices

During the first phases of infrastructure design, our studies led us to define the optimal patterns for both the supply of the SaaS and the functionalities of the application. In particular, an API Gateway-type architecture pattern was used to isolate the application from possible future changes made by Google on Google Apps. Such a component provides an intermediate layer, which allows the platform components to perform requests to the Google Apps API through calls that have a preset name, independent from Google. The internal routing system, with its rules, allows the transparent use of the correct Google API.

The following figure roughly shows the way in which software clients can use an API server with an indirect communication thanks to an API Gateway.

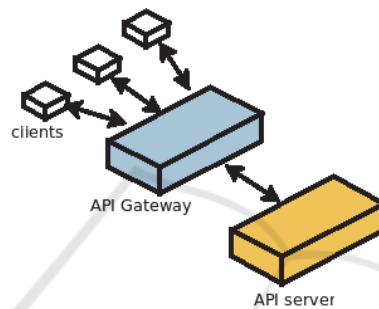


Fig. 2. Communication with API Gateway.

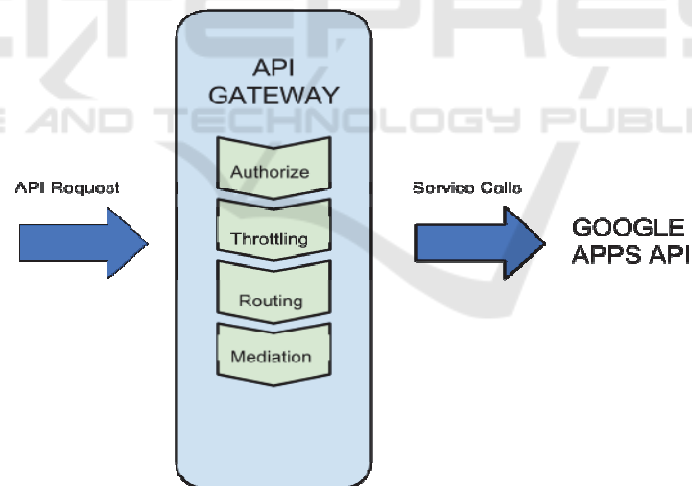


Fig. 3. API Gateway layers.

The API Gateway validates and authenticates requests to Google, transforms data before they are sent (logic validation), behaves as a mediator and performs caching. It makes it possible to manage:

- who is making the request (via OAuth);

58

- when it requests (Error-rate, payload-size, traffic shaping);
- where requests are managed (in-line transformation, orchestration);
- how to request (flow logic, transformation and validation logic, caching logic).

4.3.1 Authentication/Authorization

Two important actions performed by the API Gateway are authentication and authorization. The gateway has the account data of each user of the platform (tenant). Every request of use of Google Apps passes through an appropriate authentication module of the API Gateway, which performs all the necessary operations on Google's authentication system to authorize a tenant to use the extensions through Google Apps.

4.3.2 Caching

Caching controls the requested resource and the client requesting it. According to that information, in compliance with an appropriate Expiration Model and Validation Model, it is possible to return a local version of data or perform an interrogation towards Google Apps. That local version is called cache entry and can be automatically created by the gateway, or its creation can be inhibited with a no-cache flag related to the call arriving from the client.

We opted to store the cache gateway side, that is, storing it in the gateway. This decision was made from a performance perspective, which would be lacking when using an external module to manage the cache, for example a key managed server.

4.3.3 Mediator

This part of the application executes business operations on the resource request called by a client, executing the manipulations needed to prepare the final request to send to Google Apps. The mediator module includes a listener of the path of the request incoming from the client. If a handler, i.e. an API definition, corresponds to that path, all the operations that prepare data before the next Router/Proxy phase are performed. The operations include a manipulation of the body data of the request coming from the client.

4.3.4 Router/Proxy

This component includes a mapping between functionalities and the URL of Google API, to redirect and launch the final call that will reach the endpoints of Google's API. There is a register inside it, where each entry has the parametric URL and a number of features used to locate the entry according to the request pre-pared by the Mediator. From this module, the request is launched through a Request Rewriting operation, and the response is returned. The response will be appropriately recomposed by the mediator to be eventually returned to the client. A single request from a client could comprise more synchronous requests to Google Apps, the URL of which are included in the router/proxy module.

5 Conclusion

The project represents an advancement of the state of the art in a domain still not completely explored, namely the development of True SaaS integrated with third-party systems like Google Apps. This kind of applications will become increasingly common, and will change the way to work with software tools. The proposed solution aims to become a reference tool for the daily tasks to be performed by users of either public organizations or private companies. This solution also offers basic tools (email managers, office automation, document manager, etc.), using a suite of proven effectiveness such as Google Apps. The innovation that the completion of the project can bring is remarkable, and articulated on more facets. On the one hand, it is the answer to a current need with innovative technologies (SaaS delivered system). On the other hand, innovation is brought into the solution itself. The use of open implementation technologies and of standards of the same kind addresses classic integration and sync issues with legacy systems, and facilitates data migration (document or other) towards the proposed solution. The solution will guarantee a high security, and lower downtime levels compared to other systems in existence. In fact, through synchronous replication, users' data and actions will be copied in real time on a number of data centers, with a switching function between data centers.

Acknowledgements. This paper has been produced as part of the research project entitled "Servizi Avanzati in Cloud (Advanced Cloud Services)" developed at Ex-perteam s.r.l. company. The project is financially supported through a research grant funded by the Autonomous Region of Sardinia with European local funds (ROP SARDINIA ESF 2007-2013 – Objective: Regional Competition and Employment, Axis IV Human Resources, Line of Activity I.1.1. e I.3.1.).

References

1. Anderson, D. J.: Kanban: Successful Evolutionary Change for Your Technology Business. Blue Hole Press (2010)
2. Corona, E., Pani, F. E.: A Review of Lean-Kanban Approaches in the Software Development, Transactions on Information Science and Applications, Vol.10, No.1, Print ISSN: 1790-0832, E-ISSN: 2224-3402 (2013)
3. Pani, F. E., Sanna, D., Marchesi, M., Concas, G.: Transferring FAME, a methodology for assessing open source solutions, from university to SMEs. In: D'Atri, M. De Marco, A. M. Braccini, F. Cabiddu, (Eds.), Management of the Interconnected World, ItAIS - The Italian Association for Information Systems, A 1st Edition, XIV, 534 p., Hardcover, LNBIP, ISBN: 978-3-7908-2403-2 (2010)
4. Sharma, R., Sood, M., Sharma, D.: Modeling Cloud SaaS with SOA and MDA, Advances in Computing and Communications - Communications in Computer and Information Science (2011)
5. Poppendieck, M., Poppendieck, T.: Lean software development: An agile toolkit, Addison Wesley, Boston, Massachusetts, USA (2003)
6. Naono, K., Teranishi, K., Cavazos, J., Suda, R.: Software Automatic Tuning, From Concepts to State-of-the-Art Results, Chap. 15, pp. 255-274. Springer, Berlin (2010)
7. Forrester, The Forrester Wave: Hybrid Integration, Q1 (2014)

60

8. Lewis, G. A.: The Role of Standards in Cloud-Computing Interoperability, Software Engineering Institute, Carnegie Mellon (2012)
9. Almeida, F., Oliveira, J., Cruz, J.: Open standards and open source: enabling interoperability, International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.1, January (2011)
10. Critical Requirements for Cloud Applications: How to Recognize Cloud Providers and Applications that Deliver Real Value - Workday (2015)

