October 2012

ascent

Thought leadership from Atos

white paper

PaaS - Making the most of Clouds

Atos believes that a General Purpose PaaS would allow Cloud capabilities to be exploited to their full potential for applications. A General Purpose PaaS would be a comprehensive, open, flexible, and interoperable solution that simplifies the process of developing, deploying, integrating, and managing applications both in public and private Clouds.

This whitepaper introduces the concept of the General Purpose PaaS and describes the desired capabilities and building blocks that need to be established, and proposes an architecture for such a platform. It also offers an analysis of market trends and existing solutions, in order to establish a future vision and direction for PaaS, as well as outlining the business potential of such a solution.

PaaS - Making the most of Clouds

Contents

04
Introduction
An explanation of the need for PaaS and a definition of the concept.

05
Trends, market and technology overview
A look at the current PaaS market and forecasts for the future, including changes expected. PaaS user groups and current PaaS providers are also considered.

10
Building a General Purpose PaaS
An outline of the innovative capabilities that a General Purpose PaaS should provide in order to exploit PaaS to its full potential, including a proposed PaaS architecture.

18
Business Opportunities
A consideration of how PaaS could be used and by whom, including the benefits it could bring.

19
Conclusion
An overview of what PaaS providers need to consider in the near future.

About the Authors

Edited by Ana M. Juan Ferrer, Head of Lab at Research and Innovation at Atos in Spain (ana.juanf@atos.net), based on contributions from:

Clara Tejero Royes, System Architect at Atos in Spain (clara.tejero@atos.net)

Philippe Reynaud, VP Portfolio Management at Atos in Spain (philippe.reynaud@atos.net)

Paritosh Wechalekar, Sr. Technical Architect at Atos in India (paritosh.wechalekar@atos.net)

Jérôme Brun, Vice President Cloud Services at Atos in France (jerome.brun@atos.net)

Jordan Janeczko, Cloud Strategist for the Global Systems Integration at Atos International (jordan.janeczko@atos.net)

Purshottam Purswani , Prinicpal Architect at Atos in India (purshottam.purswani@atos.net)

Paul AlbadaJelgersma, Siemens Global Partnership at Atos (paul.albadajelgersma@atos.net)

Guy Lidbetter, Chief Technology Officer, Global Managed Services at Atos (guy.lidbetter@atos.net)

Thierry Caminel, Business Solution Manager at Atos in France (thierry.caminel@atos.net)

John Hall, Head of Portfolio & GKO at Atos in the UK (john.2.hall@atos.net)

Michael Kollar, Chief Technology Officer at Atos in the USA (michael.kollar@atos.net)

Francesco D'Andria, Project Manager at Research and Innovation at Atos in Spain and coordinator of the Cloud4SOA project (francesco.dandria@atos.net)

James Ahtes, Project Manager at Research and Innovation at Atos in Spain (james.ahtes@atos.net)

About the Atos Scientific Community

The Atos Scientific Community is a network of some 100 top scientists, representing a mix of all skills and backgrounds, and coming from all geographies where Atos operates. Publicly launched by Thierry Breton, Chairman and CEO of Atos, the establishment of this community highlights the importance of innovation in the dynamic IT services market and the need for a proactive approach to identify and anticipate game changing technologies.

Introduction

Cloud computing first emerged in the form of Infrastructure-as-a-Service (IaaS), boosted by the birth of Amazon Web Services (AWS). AWS began offering IT infrastructure services to businesses in the form of web services in 2006. At the same time, Salesforce.com was offering Software-as-a-Service (SaaS), based on the idea of application service provision (ASP). Its offering included a customization layer, force.com. Soon, driven by the existence of force.com and the entrance of Google's App Engine, the market erupted and it became clear that there was a need for a middleware layer (Platform as a Service – PaaS) between IaaS and SaaS. PaaS enables the simplified consumption of Cloud infrastructure and supports the viability of more complex and configurable Cloud applications.

NIST[1] defines Platform-as-a-Service as, "The capability provided to the consumer to deploy onto the Cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying Cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment."

IaaS vendors are now pushing up the Cloud stack to offer added-value PaaS programming frameworks on top of their infrastructure, in order to overcome the threat of increasing infrastructure commoditization. SaaS vendors are also offering platform tools to tailor their on-demand portfolio with the intention of creating customer loyalty and establishing a wider market for their offering.

As in any market for an emerging technology, there is a truly diverse array of capabilities being offered by PaaS providers, from supported programming tools (languages, frameworks, runtime environments, and databases) to various types of underlying infrastructure, even within the capabilities available for each PaaS.

IaaS vendors are now pushing up the Cloud stack to offer added-value PaaS programming frameworks on top of their infrastructure.

[1] SP 800-145, " A NIST definition of Cloud Computing", http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

Trends, market, and technology overview

Platform-as-a-Service Market in Figures

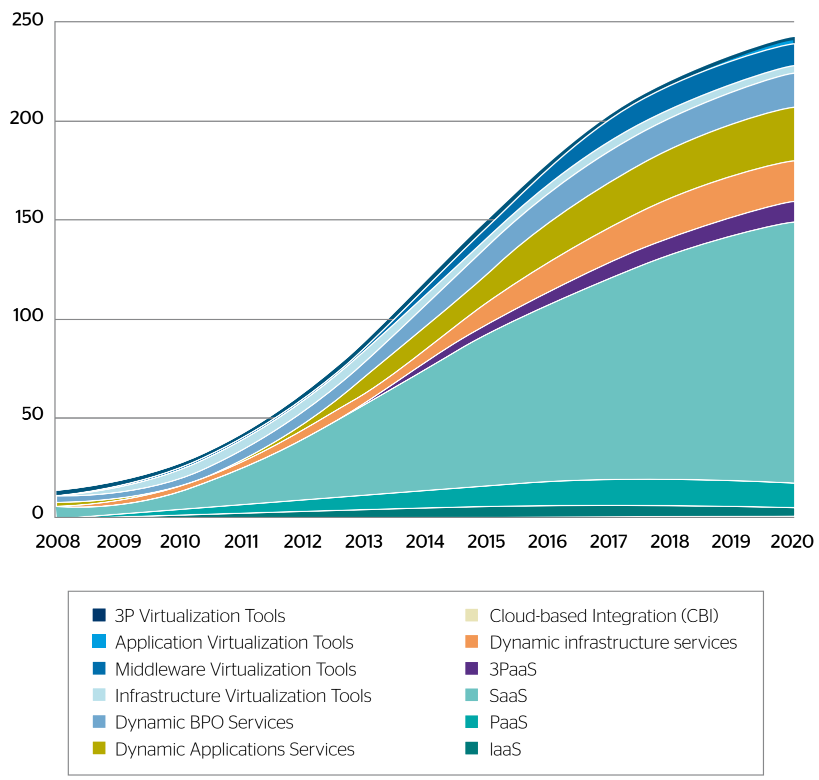
Generally speaking, sizing the Cloud Market is not an easy task. New trends, opportunities, actors, and business models are appearing all the time, leading to potentially unstable market segmentation and volatile forecasts.

The PaaS segment itself is still largely immature. It could be said that PaaS services delivered directly to end customers is not a significant business in itself today, especially when compared to IaaS or SaaS. However, PaaS is clearly a major enabler for upper-layer Cloud business (SaaS, BPaaS); the second generation of Cloud applications is increasingly based on a PaaS in order to optimize the cost of software development and maintenance.

Overview Cloud market

The following graph shows the total market size for Cloud (US $billions)

It seems that the current PaaS market is rather small. Even in 2020 it is anticipated to be significantly smaller than the markets for SaaS and even dynamic application services and BPO. Nevertheless, Gartner[2] estimates that by 2015 competition between PaaS vendors will produce new programming models, new standards, and new software market leaders.



Source: Forrester Research Inc. "Sizing the Cloud" Report March 2011

PaaS Market... currently a €1 bil. business

In 'Cloud Computing Market Figures' July 2011, Pierre Audoin Consultants[3] estimated the worldwide PaaS business to be €1.202 mil. in 2011 and forecast a rise to €7.427 mil. by 2015 (annual growth between 50% and 100%). For Europe, the same consulting firm estimates PaaS revenue to have been €223 mil. in 2011 and predicts growth to €1.462 mil. in 2015.

Gartner anticipates[4] that "the worldwide enterprise market for PaaS platforms will grow from $900 mil. spent in 2011 to $2.9 bil. in 2016, representing a 26.6 percent CAGR (combines annual growth rate). Growth rates per PaaS sub-segment include: Application Development (22%), Database Management Systems (48.5%), Business Intelligence Platforms (38.9%), and Application Infrastructure and Middleware (26.5%). Application Infrastructure and Middleware is expected to be the largest revenue source in PaaS for the next four years." Gartner reports that this sub-segment generated $649 mil. in 2011 and projects it to grow to $2.1 bil. in 2016, generating a 26.5 percent CAGR. With 76 percent of the entire 2012 public Cloud estimated to be in the BPaaS (Business Process-as-a-Service) segment, it is clear that Gartner sees strong interest from enterprise clients to spend in this area.

It can therefore be said that while the standalone PaaS business will remain relatively small within the overall Cloud market, the ratio of the PaaS value embedded in SaaS and BPaaS revenues is becoming significant.

[2] Gartner, Predicts 2012: More PaaS Vendors and More PaaS Confusion, G00226255, November 2011.
[3] PAC, Pierre Audoin Consultants, https://www.pac-online.com/pac/pac/live/pac_world/home/index.html;jsessionid=E9EDC029E52EFDCD1B9D52ACAD65918D
[4] Forbes, Forecasting Public Cloud Adoption in the Enterprise, http://www.forbes.com/sites/louiscolumbus/2012/07/02/forecasting-public-cloud-adoption-in-the-enterprise-2/, February 2012

Platform-as-a-Service Trends

Expectations within the Platform-as-a-Service market are that there will be significant changes between now and 2014. Users, providers, and industry analysts are all still evaluating the current state of play and although they all see a need for improvement, the point at which stability will be reached in the areas of feature set, architecture, and pricing models is still a topic of debate. Despite this, there are a number of identifiable trends, driven primarily by the needs of different user groups and the motivation for PaaS providers to address this market space.

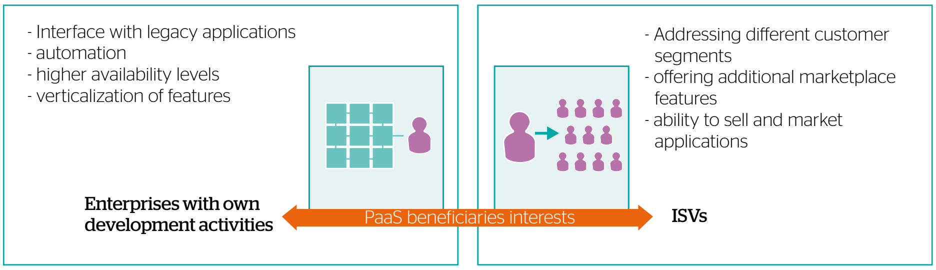


Figure 1: PaaS beneficiaries' main interests

There are two primary user groups that benefit from using Cloud at the PaaS level (compared to at an IaaS level): Enterprises with their own internal software development activities and independent software vendors (ISVs) interested in selling SaaS services on top of a hosted PaaS.

Enterprises expect several benefits from PaaS, primarily from standardizing around a specific platform. In many software development organizations (regardless of whether the programming is done in-house or by a third party) development projects make use of a very heterogeneous toolset. This tends to increase the overall cost for development and deployment, and also reduces the flexibility of team members who may need to work on several different projects at once. Ideally, enterprises should focus on one, or maybe two, different platforms to decrease costs (the use of two platforms may help avoid lock-in) allowing them to more easily transition to another platform if they become unhappy with the predominant platform.

Enterprises will also want to increase the speed that finished applications can be reliably deployed. Similar to their logic for deploying IaaS, they may prefer to begin with a private PaaS infrastructure which can then be extended into the public Cloud depending on the nature of the application and the company's overall Cloud strategy.

In spite of the long-term desire to have the option to use a public Cloud provider, heavy integration features may still be required for existing and legacy application sets. Enterprises tend to have a large installed base of existing applications and PaaS must be able to integrate easily into those existing applications. In addition, development teams must be able to start using PaaS easily. This means it must be inexpensive and teams must be knowledgeable in their domains in order to start using the new PaaS environment. Not only must PaaS frameworks fit current development models, but where possible, should offer additional features that address the specific market that the enterprise is operating in. For example, in the financial services market a PaaS with increased feature sets for security, transactional processing, and data mining may be an absolute necessity. For enterprises in pharmaceuticals, additional features covering compliance to GxP (a general term for Good Practice quality guidelines and regulations) and FDA (Food and Drug Administration) standards will be required.

Private PaaS vendors (i.e. companies who sell software enabling a private PaaS environment for their customers) are already moving toward:

- Increasing the ability to interface with legacy applications.
- Increasing support for automated deployment.
- Adding features/methods for reaching higher availability levels.
- Achieving market-tailored 'verticalization' of features.

There is an important second user group; ISVs who are interested in selling their products as SaaS. ISVs are more interested in using a public PaaS provider because of the possibility to address multiple customer segments. Public PaaS, for example force.com, often offers an additional marketplace for integrating the different applications that are using their PaaS and in the case of force.com, with the application suite of salesforce.com. This opens up new channels for ISVs to sell their products. Depending on the cost of deploying to a new PaaS provider's framework, ISVs may be willing to develop on multiple platforms in order to take advantage of new sales possibilities. There is a similar phenomenon within the mobile device space where application developers produce the same app for Apple iOS devices, Android devices, and Windows Mobile devices. These types of PaaS users are more interested in having standardized interfaces and decreasing migration costs as their aim is not to standardize their development around one platform to decrease costs, but to find a cost-effective way to use as many different platforms as possible to support a successful business case.

Trends in the PaaS space will address these users and scenarios. Public PaaS feature sets will address ISVs' needs by offering additional marketplace features and focusing on the ability to sell and market applications. As usual in 'land grabs' of emerging and growing markets, where large players try establishing themselves early on as market leaders, PaaS hosts have not invested substantial effort in standards, instead rely on establishing their platform as the de facto industry standard in order to solidify their long-term position. It remains to be seen how quickly the market will address the desire for standardized interfaces that will allow SaaS application developers to quickly switch between public PaaS providers.

Private PaaS vendors will address other issues based on the needs of their end users. They will focus on improving the robustness of the runtime/productive environment and on decreasing development costs for new applications. Similar to the classic middleware market, feature set expansion will emerge in areas of orchestration and business process automation (in this case, Cloud orchestration).

Evaluation of PaaS Providers

Below is an analysis of the main features of existing Cloud offerings with a comparative analysis made among them.

AWS Elastic Beanstalk

Elastic Beanstalk, currently in Beta version, is built on top of existing AWS' services. It targets Java developers using the Apache Tomcat software stack and enables the direct upload of a J2EE application. AWS Elastic Beanstalk allows the user to control the core elements of underlying infrastructure. Flexibility is offered for the selection of the appropriate Amazon EC2 instance type and the selection of database options. The integration of AWS Elastic Beanstalk with the AWS Auto scaling service provides the interesting capability of automatically scaling applications up or down, enabling automatic handling of peaks in the application's workload and traffic.

CloudBees

CloudBees allows the building, testing, and deployment of Java web applications in the Cloud. It consists of DEV@Cloud, a framework that not only enables the deployment of an application in the Cloud like a traditional PaaS, but also continuous integration of a project in the Cloud, and RUN@Cloud that offers support for the deployment and administration of Java applications in the Cloud. RUN@Cloud also offers traditional application server functionality, providing load balancing, scalability, and high availability for web apps. CloudBees includes an Ecosystem of tools from partner companies, some of which are provided free of charge.

Cloud Foundry

Cloud Foundry, a VMware-led project, is the world's first 'open' PaaS offering. It provides a platform for building, deploying, and running Cloud apps using Spring for Java developers and other JVM languages/frameworks, including Rails, Sinatra, Node.js, Groovy, Grails, and Scala. Cloud Foundry is an application platform which includes a self-service application execution engine; an automation engine for application deployment and lifecycle management; a scriptable command line interface (CLI) for integration with development tools to ease development and deployment processes; an open architecture for quick development framework integration; an application services interface; and a Cloud provider interface. One of the core tenets of Cloud Foundry is that it is free from underlying infrastructure. This gives users the option to use their existing infrastructure (desktop, data center and private Clouds) whilst still leveraging all the benefits of PaaS.

Google AppEngine

Google App Engine is a PaaS offered by Google. Its main value proposition is that developers can quickly build small applications locally (on developer machines) and deploy them to the Cloud in the same environment that powers Google applications. It offers fast development and deployment and simple administration. Supported languages are currently Java, Python, and Go with an appropriate Software Development Kit available for each. This platform provides an execution environment where applications run on a virtualized technology foundation that scales automatically on demand. Google AppEngine is often criticized for not providing transparency to the user to allow control of the infrastructure it uses. Developers do not have direct control over resource allocation because the underlying system and hardware resources are masked by the AppEngine layer depending on the degree to which they rely on direct access to Google data persistence.

Microsoft Windows Azure

The Windows Azure Platform is a PaaS for applications built using the .NET framework. The platform consists of various on-demand services hosted in Microsoft data centers and commoditized through three product brands:

- Windows Azure: an operating system providing scalable compute and storage facilities.
- SQL Azure: a Cloud-based, scale-out version of SQL Server.
- Windows Azure AppFabric: a collection of services supporting applications both in the Cloud and on premise.

The Windows Azure Platform provides an API (application programming interface) built on REST, HTTP, and XML that allows developers to interact with the services provided by Windows Azure. It also provides a client-side, managed class library which encapsulates the functions for interacting with services, and it integrates with Microsoft Visual Studio allowing it to be used as the IDE (integrated development environment) to develop and publish Azure-hosted applications.

Simple applications that just use web pages, web services, and SQL Azure Database can be moved between on-premise data centers to the Azure Services Platform with minor configuration changes. However, applications that take advantage of other Azure Services and Cloud-specific features of Windows Azure, like Blobs, Tables, Queues, and Worker roles, require rewriting specifically for the target platform.

Comparative Analysis

There are clear differences with the PaaS offerings, in terms of supported programming tools (languages, frameworks, runtime environments and databases), underlying infrastructure (private/public IaaS Clouds), and even the application administration toolsets available for each.

With regards to programming tools, providers often follow the strategy of supporting as many languages and frameworks as possible in order to attract multiple programmer communities. Not surprisingly, Java EE is widely adopted for web application development, it is currently supported by Google AppEngine, CloudBees, Cloud Foundry, and VMforce from Salesforce.com, amongst others. The management of the underlying infrastructure is also diverse. Cloud Foundry and CloudBees, for example, can use private or public IaaS and even support a degree of federation, or Cloud bursting, among providers. Some PaaS offerings are bound to specific IaaS offerings, i.e. AWS for AWS Elastic Beanstalk or Engine Yard, while others like OpenShift support interoperability among IaaS providers. In some cases, developers need to manage a directly generated, virtual infrastructure, in other cases, such as with Google App Engine, this process is completely transparent.

There is one common characteristic; PaaS users are currently bound to the specific platform they use, making portability of software (and data) created on top of these platforms difficult in some cases.

Current diversities among PaaS result in portability and interoperability issues.

The following table provides a comparison of the platforms studied:

PaaS Diversity/Type	AWS Elastic Beanstalk	CloudBees	Cloud Foundry	Google AppEngine	Microsoft Azure
Programming Languages	Java	Java, Grails, JRuby, Coldfusion, Scala	Java, Ruby, Node.js, Scala	Phyton, Java, GO	C#, Java, PHP, Ruby
Runtime Environment	Apache Tomcat	JVM based		Jetty	Windows Azure
Supported Frameworks	-	Spring	Spring, Rails, Sinatra, Groovy, Scala	Djiango	.NET
Database and Data Types	Amazon RDS, Amazon SimpleDB, Microsoft SQL Server, Oracle	MySQL	MySQL, Redis and MongoDB	SDK	SQL Azure
Development Tools	AWS Toolkit for Eclipse	IDE,SVN	Cloud Foundry integration Extension for Eclipse and STS	Web, API	SDK
PaaS Controlling	Web CLI, API	Web CLI, API	Web, API	Google Apps Support	Web
Tools and Plugins	New Relic	Sauce Labs Ondemand, JFrog, SonarSource, New relic, Cloudant			
Infrastructure Portability	-	Amazon, OpenStack, vSphere	Free from the underlying infrastructure, can be used for private Cloud and AWS		
Payment Model	PaaS not charged, payment on pay per use of underlying AWS resources used (EC2, S3)	Combination of subscription + pay per use based on services usage	Under establishment	PaaS not charged. Basic hosting with limitations, free of charge. Additional services under pay-per-use + subscription model	Access control and Service bus per transaction,

Current PaaS offerings also fit into at least one of the following classifications[5,6]:

Classification	Description	Examples
SaaS with extensions	Customize and extend the capabilities of a SaaS application.	Force.com
Purpose-built PaaS	A framework that simplifies the development of a specific class of applications.	Microsoft Windows Azure
PaaS tied to an application paradigm	Provides general capabilities, but supports only one programming model or development/deployment environment.	CloudBees, OpenShift Google AppEngine
PaaS tied to an IaaS Cloud	May provide general capabilities, but can be used only in the context of a determined IaaS Cloud; either a single public Cloud or a single type of private Cloud infrastructure.	Cloud Foundry AWS Elastic Beanstalk

[5] Cloud4SOA project, D1.1 Requirements Analysis, http://www.cloud4soa.eu/sites/default/files/Cloud4SOA%20D1.1%20Requirements%20Analysis.pdf
[6] Forrester Blog, Platform-As-A-Service, Chapter 2, http://blogs.forrester.com/john_r_rymer/10-05-11-platform_as_a_service_chapter_2

Building a PaaS: General Purpose Paas

A General Purpose PaaS is described as:

A comprehensive, open, flexible, and interoperable solution that simplifies the process of developing, deploying, integrating, and managing applications both in public and private Clouds.

A general purpose Paas is comprehensive, open, flexible, and interoperable.

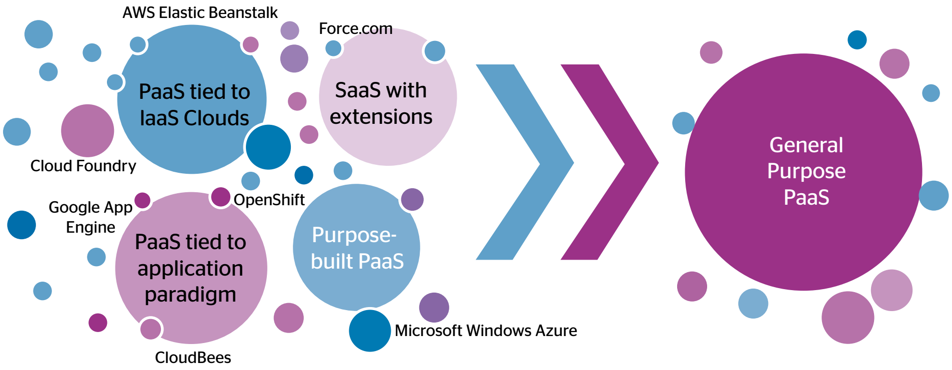


Figure 2: PaaS Evolution

Below follows an elaboration of the innovative capabilities that this next-generation General Purpose PaaS should provide in order to exploit PaaS to its full potential. These include:

- Increased ability to interface with legacy applications.
- Increased support for automated deployment.
- Added features/methods for reaching higher availability levels.
- Interoperability at all levels.
- Market-tailored 'verticalization' of features, i.e. adaptation of the platform to tailor it for specific uses.

Capabilities are categorized by PaaS core elements, presenting both current state of practice and further developments required. Finally, a potential architecture addressing the challenges presented is described.

General Purpose Paas Core Elements

Phil Wainewright's[7] model that defines and groups the four core elements of Cloud from a provider's perspective is used to describe the core element of a General Purpose PaaS:

- **Multi-tenancy** is a key factor in making the most of a PaaS Cloud environment. By supporting multi-tenancy, a platform can more easily achieve economies of scale, a reduction in cost per user, and the ability to share continuous improvements across a wider community. However, it also demands additional requirements for isolation into its architectural design to ensure that the execution of an application does not have any impact on others with regards to security, performance, availability, and administration.
- **Cloud Reach** represents the elasticity factor, commonly the capability to scale, not only in the traditional sense of more infrastructure capacity, but also:
  - The capability to support multiple platforms.
  - The possibility to extend the execution of processes across Clouds and to connect environments[8].
- **Service delivery** includes service management, monitoring and provisioning, pay-as-you-go pricing, and billing capabilities.
- **Functional scope** represents the traditional software platform capabilities present in any development platform (not Cloud-specific).

[7] Redefining Software Platforms - How PaaS changes the game for ISVs, http://ippblog.intuit.com/blog/2009/10/redefining-software-platforms---how-paas-changes-the-game-for-isvs.html, http://developer.intuit.comuploadedFiles/Developer/MyIDN/Technical_Resources/QBPD/Redefining%20software%20platforms%20Sep09.pdf

[8] *"Cloud Orchestration, a real business need"* by Wechalekar, Juan Ferrer & Albada Jelgersma, March 2011.

A Deeper Look: Desired Capabilities

Features and desired capabilities for each of the core elements of the General Purpose PaaS are described below. It should be noted that although some of the PaaSs that exist today offer partial coverage of the capabilities described, there isn't any that offers the full functionality.

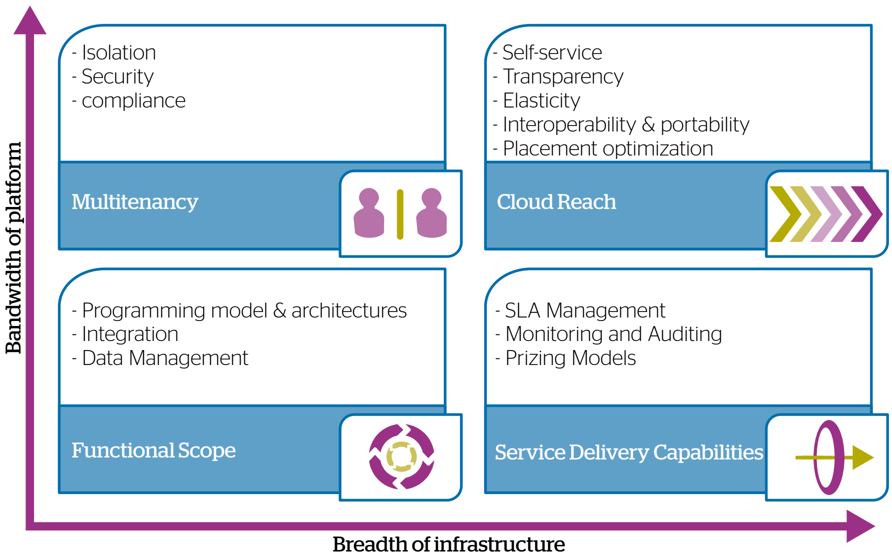


Figure 3: The two dimensions of Clouds (adapted from[13])

Multi-tenancy

- **Execution isolation:**

  In the context of Cloud computing, multi-tenancy is a Cloud's ability to share computing resources that are being used by different concurrent users. Isolation is the capability of perceiving a shared environment as dedicated and safe. Complete isolation among applications executed in PaaS environments can be achieved using different strategies[9]:

  - Virtual Multi-tenancy: This approach simply relies on the isolation provided by resource virtualization (VMs) and hypervisors in the infrastructure management layer. Although it is by far the most common approach to isolation, research demonstrates some degree of performance interferences in virtualized environments[10-11]; the coexistence of several VMs in the same physical resource leads to influences among VMs that can affect performance. Current virtual infrastructure managers do not take these influences into account for VM scheduling, nor do Cloud providers for billing[12]. New approaches to minimizing interference and performance accounting would be required to achieve complete isolation by this means.
  - Organic Multi-tenancy: This approach is based on isolation being achieved at different PaaS component levels, such as application servers, DBMS (database management system), etc. Logical isolation is useful for PaaS environments that do not rely on a specific IaaS infrastructure, but requires more management work than virtual isolation for the assignment of physical resources and the migration of user instances to new physical resources. This approach also relies on a high degree of trust among PaaS users and PaaS providers, since PaaS users do not implicitly have the means to verify complete isolation of their respective user environments.

- **Security at multiple levels:**

  While Cloud computing offers a paradigm-shifting technological solution for computational resources and software, concerns about data privacy and confidentiality, as well as the underlying security and resilience of resources delivered in the PaaS and IaaS environments, are certainly considered obstacles to the uptake of Cloud-based delivery models. The main concerns relating to security issues are:

  - Inherent loss of control over enterprise data.
  - Inability to enforce security policies due to the de-perimeterization of the enterprise boundary.
  - Security issues associated with a brokerage and federation based multi-Cloud model.
  - Inability to perform thorough audits and evaluations of the Cloud environments.
  - Absence of security standards and certification.
  - Regulatory compliance with data privacy and mobility of data across multiple legal boundaries.
  - Confidentiality and integrity of data hosted in the Cloud.
  - Availability and resilience of the services and the associated need for infrastructure protection along with the security of the various technology layers of the Cloud model.

- This topic is already addressed in detail in a dedicated Atos Scientific community whitepaper[14].

---

[9] Steve Bobrowski. 2011. Optimal Multitenant Designs for Cloud Apps. In Proceedings of the 2011 IEEE 4th International Conference on Cloud Computing (CLOUD '11). IEEE Computer Society, Washington, DC, USA, 654-659. DOI=10.1109/CLOUD.2011.98 http://dx.doi.org/10.1109/CLOUD.2011.98.

[10] Younggyun Koh; Knauerhase, R.; Brett, P.; Bowman, M.; Zhihua Wen; Pu, C.; , "An Analysis of Performance Interference Effects in Virtual Environments," Performance Analysis of Systems & Software, 2007. ISPASS 2007. IEEE International Symposium on , vol., no., pp.200-209, 25-27 April 2007.

[11] Xing Pu; Ling Liu; Yiduo Mei; Sivathanu, S.; Younggyun Koh; Pu, C.; , "Understanding Performance Interference of I/O Workload in Virtualized Cloud Environments," Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on , vol., no., pp.51-58, 5-10 July 2010.

[12] *Ibid.*

[13] Redefining Software Platforms - How PaaS changes the game for ISVs, http://ippblog.intuit.com/blog/2009/10/redefining-software-platforms--how-paas-changes-the-game-for-isvs.html, http://developer.intuit.com/uploadedFiles/Developer/MyIDN/Technical_Resources/QBPD/Redefining%20software%20platforms%20Sep09.pdf

[14] Jordan Janeczko, Scientific Community Whitepaper, Risk Analysis Framework for Cloud environments.

- **Compliance:**
  Public and hybrid Cloud scenarios are characterized by a constant flow of data which usually cannot be specifically allocated to a particular geographical location. This results in uncertainty with regards to various data protection legislation, which may transcend national borders and therefore complicate compliance with global Data Protection legislation. Enterprises or individuals using PaaS to develop applications that handle confidential and private data need to safeguard its privacy. Therefore, from a legal point of view, providing mechanisms to enable data protection and privacy in Cloud environments should be a fundamental requirement of the General Purpose PaaS. To ensure the ongoing integrity of this privacy, it is also important to contribute to and align with standards and policies created by industry organizations, commercial enterprises, and governments.

Cloud Reach

- **Self-service:**
  This requires the complete automation of provisioning, configuration, and administration of resources in order to bridge the gap between the development of an application and its operation. Firstly, it has to include offer capabilities to deploy and promote application source code, software, and artifacts along the whole application lifecycle (considering different environments, such as development, testing, and production environments). Secondly, it has to offer capabilities for the automated deployment and configuration of application binaries and application infrastructure components, such as OS, Application containers, DBMS, Load Balancers, etc. Finally, virtual service provision based on hypervisor-specific VM instantiation from templates delivered in a hypervisor agnostic way has to be provided for multi-Cloud deployments (considering multiple IaaS providers). As a result of the current lack of interoperability among VM formats, multi-Cloud deployments (using multiple IaaS providers) require virtual image transformation processes.

- **Transparency (independence) and full control of the underlying infrastructure:**
  - With regards to management of the underlying infrastructure, PaaS offerings will require transparency, which can be achieved through the complete automation and self-service management of the underlying infrastructure, while allowing visibility and control over application execution, for advanced users.

- **Application Elasticity:**
  Elasticity is an essential core feature of PaaS application execution. Elasticity dictates the ability of the underlying infrastructure to adapt to an application's demand, ideally considering both functional and non-functional requirements (i.e. number of concurrent users, application response times, and number of open DB connections, etc.). Strictly speaking, elasticity is an indication of the ability to seamlessly increase and decrease resources, while scalability is a measure of the ability to increase capacity through the addition of resources, although commonly they are used as synonyms. Two types of scalability are usually considered: horizontal and vertical. Horizontal scalability refers to the number of instances to be satisfied (e.g. changing volume of requests). Vertical scalability refers to the size of the instances and is thus linked to the number of resources needed to achieve the required configuration. For true Cloud elasticity, both vertical and horizontal elasticity are called for, supported by rapid up- and down-scaling.

  The network is often neglected despite the fact that the Cloud's essential characteristics for all of its service models are its availability over the network. Network performance significantly influences a user's perception of applications and services in the Cloud; broadband, latency, and throughput all impact services' performance. Network management has significant influence on scalability. High-performance application execution in PaaS environments requires of the holistic management of all resources involved (network, storage, and compute). Common approaches to controlling scalability on Cloud infrastructures rely on VM replication and/or ignoring underlying infrastructure. However, on demand elasticity on network pipes and connecting elements is necessary in order to take an overall approach to Cloud scalability.

- **Interoperability and Portability:**
  A significant barrier to Cloud computing adoption is interoperability and portability across providers and products. In the scope of PaaS, there are two approaches to interoperability:
  - Interoperability and portability of applications among PaaS environments: the ability to migrate applications across PaaS offerings. Current issues can be summarized as:
    - Lack of common/standardized Cloud PaaS APIs.
    - Too much diversity in frameworks, languages, toolsets, and SDKs (proprietary).
    - Different levels of services (different types of PaaS solutions).
    - Heterogeneous data types and storing methods.
  - Non-interoperable accounting, billing, metering, and advertising services.
  - Interoperability and portability of applications among PaaS execution environments (IaaS): Allowing a PaaS deployment to interact using the same API and the same application with several IaaS Clouds. More details on this approach are provided in the Placement Optimization section below.

- **Placement Optimization: Best venue execution selection:**
  This provides important benefits as the user is not locked into a single IaaS provider, but can decide the best choice for the deployment of each service among the different IaaS Cloud offers. The selection can be optimized based on the type of service to be deployed and different application execution requirements, such as the level of trust in the IaaS provider, or based on previous application execution experiences, including eco-efficiency, risk, cost, security levels offered, legal constraints, and quality of service parameters (e.g. availability, performance, operation, etc.).

A user may even want to launch a cross-site application that will be deployed in different providers at the same time. This can be beneficial for the distribution of the different components of a given application among different Cloud providers, for example, to improve fault tolerance in case of Cloud service disruption, to implement application load balancing features among different sites, to enable the possibility of expanding the application capacity from one Cloud to another, to implement proximity policies regarding the location of service consumers, or for any other reason (to lower costs or improve security, for better ecological performance, or enhanced performance or security, etc.). Two of the main technological challenges of this approach are the creation and management of cross-site private networks and LANs using simple, standard procedures to interconnect different service components, and the creation and management of virtual storage systems across site boundaries to store service data.

Service Delivery

- **Service Level Agreement (SLA) Management:**

  SLA Management is a key aspect of providing a commercially-viable PaaS offering. There are two sides to SLA management. The first is the SLA between the PaaS provider and the PaaS user. The second is the SLA between the PaaS provider and different Infrastructure (IaaS) providers it may use for application execution. It should be noted that the latter strongly influences the SLA that the PaaS provider can offer and is therefore taken as the focus for this section.

  In order to assure proper Quality of Service (QoS) for application execution, SLAs that describe application execution requirements between the PaaS provider and IaaS providers have to be established. QoS support is a relevant capability that is essential in many use cases where specific requirements have to be met through application execution. Basic QoS metrics like response time, throughput, etc. must be guaranteed so as to ensure that the quality guarantees of the user are met, at the very least. Other parameters have been widely investigated, such us:

  - Trust: Trust is a multifaceted aspect related to areas such as risk and security, and is strongly influenced by perception and previous experience. In essence, trust is a subjective measure and to cope with this subjectivity it can be assessed by using reputation mechanisms based on how well promised levels of service are achieved.
  - Risk: Risk corresponds to the probability of the occurrence of hazardous events that would have a negative impact on service provision under the agreed service levels. Identifying, assessing, treating, and monitoring risk is imperative for the proactive operation of IaaS.
  - Environmental concerns reflected in upcoming legislation have increased awareness of the ecological (Eco) footprint of the ICT industry. Level of ecological awareness can now be a determining factor between competing IaaS providers. Furthermore, rising electricity prices may guide the execution of services to locations in which requested services can be provided in a more efficient way. It is therefore desirable to specify and enforce power consumption limits for application execution in SLAs, to decide where applications are to be executed based on electricity prices, and to monitor and assess ecological factors in executing applications.
  - Cost limits are necessary to balance the previous three parameters and to take a pragmatic approach to controlling application execution.

- **Application execution monitoring and auditing**

  This refers to the capacity of PaaS users to assess and validate SLAs and the real QoS provided during application execution, as well as to gain insights about PaaS and IaaS providers' internal procedures. Aspects such as regulatory and security controls (incident inventory, handling and corrective actions, disaster recovery plans, business continuity, etc.) and data location traceability are crucial for users to overcome the loss of control inherent to the use of a Cloud platform.

- **Consideration of multiple pricing models**

  For both providers and users flexibility in pricing models is essential. Potential pricing models consider price per VM, per hour, per hour of CPU time, per user per month, etc.

Functional scope

- **Programming models and application architectures**
  The essential characteristics of Cloud infrastructures are inherent support for elastic, scalable, and resource-friendly application execution. However, these characteristics are not independent of application architecture and platform; application has to support the full exploitation of the Cloud's benefits and characteristics. The main objective of a General Purpose PaaS is to allow users to develop applications in a seamless way with regards to the underlying infrastructure. In addition, the General Purpose PaaS has to consider:
  - Automatic extraction of parallelism and scalability to get Business Process Objectives for the provided service.
  - Support for orchestration among newly developed software and existing services (executing in the same or a different infrastructure).
  - Seamless migration of existing applications to Cloud execution environments.
  - Complete independency from the execution environment.
  - Transparent provision of metering and billing tools, so that developed software (services) can be easily exploited in SaaS models.
- **Integration**
  Integration refers to the ability to integrate with legacy software and on-premise assets (i.e. licensed software), as well as with third-party Cloud services (Cloud Orchestration). As part of considering the development of new services, PaaS providers have to enable the adaptation and combination of legacy and licensed software in addition to the composition of applications within larger contexts, including existing in-house applications and other third-party provided Cloud services. By using a General Purpose PaaS, applications will be able to execute licensed software in hybrid-Cloud scenarios by means of distributed license validation mechanisms, as well as integrating with existing services, enabling enterprise software to exploit the potential of a Cloud ecosystem.
- **Application Data Management Capabilities**
  Most PaaS providers already offer data management services with limited scalability for MySQL, PostgresSQL, CouchDB, and other open-source data management solutions that are easy to integrate with the platform's backend. Scalability and functionality are even more limited for users. Other approaches rely on integrating the PaaS offering with Data-as-a-Service offerings, such as AWS S3, SimpleDB, or Microsoft Azure. For any approach, given the already identified data security and compliance criticality in PaaS, required capabilities are:
  - Configuration: The ability to build, package, deploy, install, configure, and verify data sources.
  - Resource logging and monitoring: Event capture, propagation, analysis and reporting, and escalation.
  - Scheduled maintenance downtime: The capacity to add new data and ability to remove old data.
  - Usage reporting management: Dashboards, billing/chargebacks, long-term trending and performance throughput, and response-time.
  - Security data multi-tenancy: Degrees of isolation, identity management, authentication, authorization, and auditing.
  - Provisioning and de-provisioning: Flexible scale-out and scale-in, and locality-awareness of data.
  - Online retention data migration: Tiering, information lifecycle management ,and CAR (Compliance/Archival/Retrieval)
  - Backup and recovery replication: Snapshots
  - Integrity tampering

Architectural view

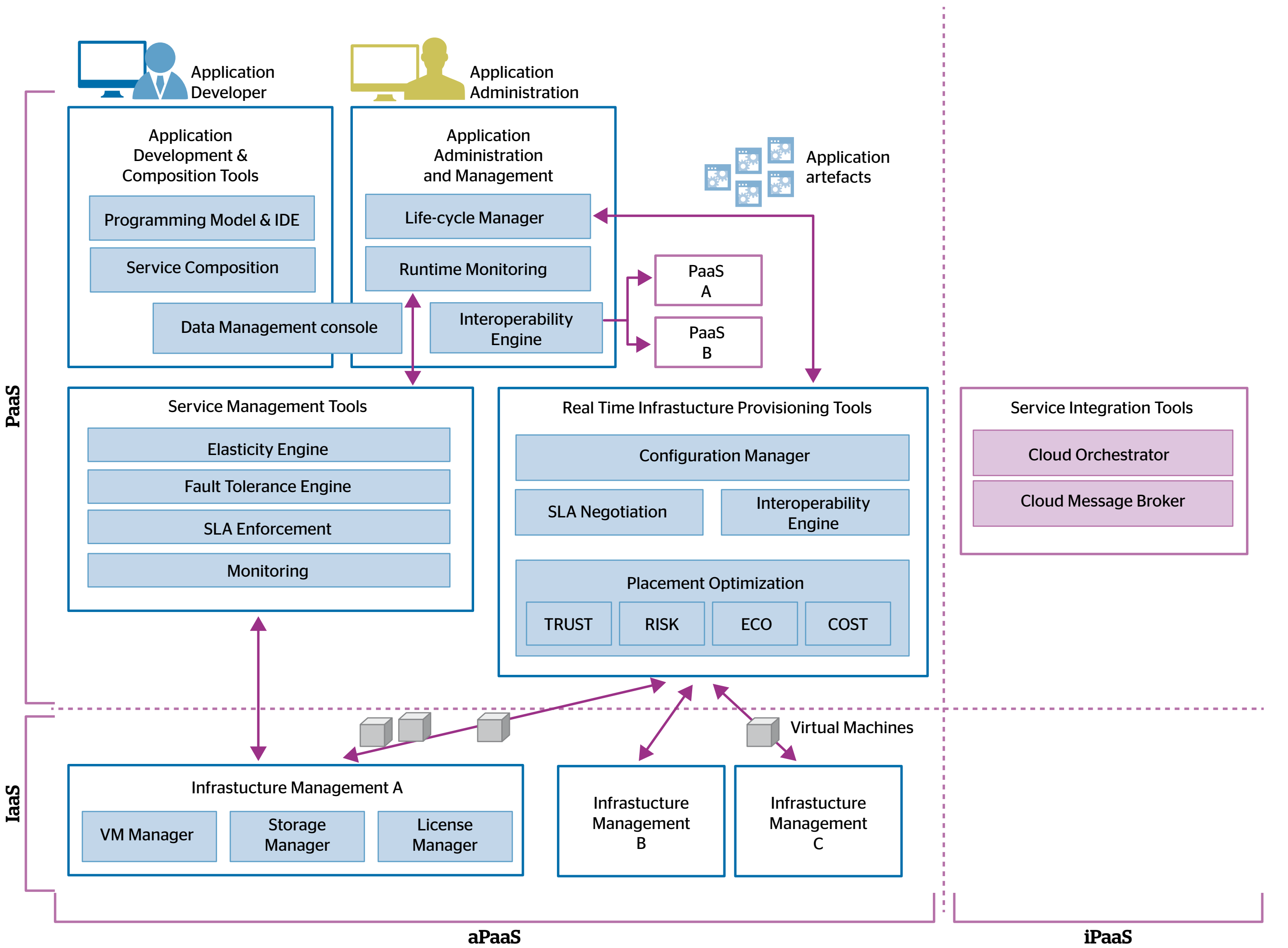


Figure 4: Proposed PaaS Architecture

Figure 4 shows the proposed solution for the General Purpose PaaS. The proposed architecture is split into both vertical and horizontal layers. Vertical layers represent differentiation between IaaS, infrastructure management components, and pure PaaS components, which rely on IaaS components.

Horizontal layers perform the differentiation suggested by Gartner in its reports[15,16,17]. In these, PaaS components are differentiated by:

- aPaaS (Cloud-enabled Application Platform – CEAP): "The platform for hosting and managing individual application services and data."
- iPaaS: "The platform for intermediation and integration of the application services hosted and point-managed by aPaaS".

[15] Gartner, PaaS Road Map, A continent Emerging, G00209751, January 2011
[16] Gartner, Reference Model for Integration PaaS, G00213749, June 2011
[17] Gartner, The Role of CSB in Cloud services Value chain, G00218960, October 2011

Some of the components described are not completely new developments at Atos. The available related and applicable software components for Scientific Community Proof-of-Concepts (PoCs) or Research and Innovation are described below. The architecture is built on the following building blocks:

Application Development and Composition Tools: An Integrated Development Environment (IDE) enables a programmer to develop code in widely-used programming languages, such as Java. Indications for the programmer about how to parallelize application execution parts should be given by annotations to the source. The same means could be used to indicate execution requirements for an application (i.e. access to private data). An IDE also has to allow for the composition of applications developed using existing services in the same or external platforms, by dynamically creating clients and stubs. The data management console has to provide tools to define data formats and schemas, as well as to define mechanisms to import and export data for testing purposes. Generated source code has to be stored in shared versioning code systems where visibility and access is provided per user or per organization using the PaaS.

Application Administration and Management: This building block provides the following tools:

- Service lifecycle manager: A management console that enables an application administrator to control application execution taking into consideration different versions for different environments (development, testing, and production). Control over execution has to include basic pre-configuration operations, such as configuration of the application's software pre-requisites, definition of SLA parameters for application execution, definition of elasticity rules and set-up of the application's data sources. Once the application has been configured, it has to allow the deployment, starting, resumption, stopping and un-deployment of the application in a given environment.
- Runtime monitoring: Visibility over users' applications during execution on a given infrastructure, including on-line information with regards to the parameters included in the SLA, as well as overall execution parameters, such as eco-efficiency, cost, and risk of breaching the SLA should be provided, in addition to console access to the virtual infrastructure in use at any given moment for an application. With regards to data management, visibility on data sources' physical location and resource logging and monitoring should be provided. Provisioning and de-provisioning of data sources and back-up and recovery procedures should also be allowed.
- Interoperability Engine: For a given application, based on programing language, data management schema, etc., alternative PaaS environments for specific applications should be provided. Once the user has selected their desired alternative PaaS the necessary conversion should be implemented to migrate the application between the two platforms.

Real-time infrastructure Provisioning: A General Purpose PaaS should provide the tools required to completely automate the process of acquiring virtual infrastructure for a given application. Based on the SLA parameters determined, the PaaS should be able to automatically negotiate the execution of the application on the infrastructure of one or more providers. Once the placement decision has taken place, the required virtual infrastructure should be generated using image templates, configuration, and contextualization tools. This makes the use of the interoperability engine twofold: conversion of API calls depending on the API exposed by the selected infrastructure manager; once the application is in execution, VM formats must be converted if the VM has to be migrated to a different infrastructure provider that relies on a different hypervisor technology than the original.

Infrastructure Management Toolset (to include VM managers, data managers and network managers already included within any IaaS infrastructure): The License Manager component must be used in cases of licensed software execution and the PaaS acts as an authorizing entity on behalf on the host domain license server.

General Purpose PaaS should provide the tools required to completely automate the process of acquiring virtual infrastructure for a given application.

Service Management Tools: Service management tools control the execution of an application over one or multiple infrastructure providers. These tools ensure that the agreed SLA is enforced and if not, takes the decision about the migration of the application VMs to a different infrastructure provider. They also use monitoring information to detect virtual infrastructure failures and apply defined elasticity rules to scale the virtual infrastructure assigned to application execution up and down.

Service Integration Tools: Single building blocks, like iPaaS, include components to enable across-Cloud application execution, such as for federated ESB (Cloud orchestration) and Cloud message broker.

Baseline Technologies

Below are details of Atos' related work for this whitepaper that was used as a baseline for the development of the General Purpose PaaS:

Service Composition and Cloud Orchestration: Scientific Community PoC: Demonstration of service orchestration among multiple Clouds.

Cloud Message Broker: Scientific Community PoC: The Cloud message broker (CMB) offers a shared Cloud-based message queuing framework (Cloud-message queuing), enabling messaging between various entities that wish to communicate with each other seamlessly and reliably using standard vendor neutral protocols.

Cloud4SOA Research and Innovation project [www.Cloud4soa.eu]: Cloud4SOA aims to enhance Cloud-based application development, deployment, and migration by semantically interconnecting heterogeneous PaaS offerings to facilitate their interoperability.

- The Cloud4SOA challenge is to open up the Cloud market to small-medium European PaaS providers and strengthen their market position, and to treat the vendor-lock in problem.
- The solution is to investigate and develop new technologies and standards for platform virtualization, thereby enabling their interoperability.

OPTIMIS Research and Innovation project [www.optimis-project.eu]: OPTIMIS aims to enable a dependable Cloud Service Ecosystem for the delivery of ecologically and economically sustainable IT services.

- OPTIMIS addresses the scenario of 2013+ where most companies use private and public Clouds in combination (hybrid Clouds).
- OPTIMIS considers full-service lifecycle optimization from construction, to deployment and operation both in private and public Clouds.
- Optimization is performed based on trust, risk, eco-efficiency and cost.

SmartLM Research and innovation project [www.smartlm.eu]: SMARTLM provides a generic and flexible licensing virtualization technology for new service-oriented business models across organizational boundaries.

OPTIMIS, Cloud4SOA, SmartLM research projects and Scientific Community Cloud Proof-of-Concepts are baseline technologies for Atos General Purpose PaaS.

Business Opportunities

New scenarios

The impact of Cloud Computing on a commercial or not-for-profit organization's strategy is expected to be significant, but is still an area of much debate and experimentation. There are companies that have already been using Cloud as the primary means of doing business, but it is still too early to predict whether or not these models are sustainable (although they seem to be very profitable).

The business benefits and strategic viability may be reasonably clear when considering an infrastructure Cloud or a software Cloud, but not so clear if looking at Platform-as-a-Service. Firstly, it is unclear what the platform's contribution is to the business strategy and secondly, the platform architecture's impact on the business model is not well defined. Because the platform resides in the middle layer, it bridges the gap between functionality and the underlying technology; it is linked to the technology and at the same time defines the available functionality of the application running on top of it that the user experiences.

By means of a General Purpose PaaS, an application running in the Cloud is expected to inherit all the aspects that make it 'Cloud'; on-demand and self-service, accessible through the Internet, pooled resources, elastic capacity, and most of all a usage-based billing method.

For Platform-as-a-Service, it is not enough to provide these capabilities to the end user of the application, these capabilities must also be given to the application developer.

This places PaaS users into two categories: software developers and software consumers. This opens up business opportunities for the provision of PaaS to companies in need of a Cloud platform to develop generic business applications, as well as the direct provision of PaaS to end users to build specialized applications to be used only within one company for a single specialized task.

This discloses PaaS as a toolbox that allows 'end users' to carry out quick-and-dirty programming, assuming that the platform is rich enough to provide most required components. In this way, PaaS could even fit a specific vertical market as it could be enriched with the generic components for that market already preloaded.

In some cases, this could bring the provision of a vertical PaaS 'dangerously' close to ERP system vendors' offerings which can be tailored to an organization by applying configurable variables or industry templates.

PaaS Benefits

Even without consideration of future business scenarios, it is clear that utilizing PaaS brings immediate business benefits and opportunities for development communities:

- Convergence of frameworks, and improvement of user management and of framework deployment.
- Facilitation of collaborative work between teams.
- Possibility to develop and test Cloud-ready applications.
- Immediate provisioning of environments for a new project.
- Immediate decommissioning of no longer needed environments.
- Improved management of licenses.
- No more fixed assets for computing and storage.

In addition, applications built on mature platforms will accelerate business strategy change to accommodate a more flexible and sustainable application landscape. Atos believes that most core internal processes have 'good-enough' IT and automation capability. Calling on mature PaaS will enable organizations to focus on strategic issues, because:

- It offers a response to the increasing speed of business and the necessity to be 'first-to-market'. A capable PaaS platform will potentially allow a significant shortening of the time-to-market for applications.
- The nature of PaaS enables it to bridge technology and user functionality with all kinds of built-in capabilities for billing, orchestration of workflow, auditability, and compliance reporting. That facilitates a focus on better and more flexible interfaces for users, specifically when coupled with mobile devices.
- The same PaaS capabilities will allow for a built-in connection between real-time business data and the business intelligence that can be derived from historic data and other data mining technologies.
- Finally, through inheritance of the underlying Cloud aspects, PaaS will force the business strategy to become service-driven.

Based on these viewpoints, PaaS enables an IT services company to provide both a generic platform and a very specialized environment for developers, for general purpose development and for specialist developers alike.

It also allows an IT services company to provide a flexible IT infrastructure layer, and makes possible the concept of all applications having a common messaging infrastructure, allowing for reduced overhead, and uniform communication, reporting, provisioning and billing.

Conclusion

Although standalone PaaS business is relatively small within the current overall Cloud market, it is clear that it constitutes a clear opportunity for IT service providers within the relatively short term. PaaS users have current and emerging need sets which must be addressed by PaaS in the near future in order to enable applications to exploit the full potential of Cloud:

- Increased ability to interface with legacy applications.
- Increased support for automated deployment.
- Added features/methods for reaching higher availability levels.
- Interoperability at all levels.
- Market-tailored 'verticalization' of features; adaptation of the platform to tailor it for specific uses.

The framework proposed in this whitepaper for a General Purpose PaaS aims to overcome current PaaS limitations.

This is completely in line with the announcement made by Atos, EMC, and VMware of a new strategic alliance that will offer packaged software, and platform and infrastructure Cloud services to global enterprise organizations. Together, with the creation of Canopy, we will provide a wide range of Cloud solutions and services designed to speed up delivery to help customers quickly take advantage of the benefits of Cloud computing.

About Atos

Atos is an international information technology services company with annual 2010 pro forma revenues of EUR 8.6 billion and 74,000 employees in 42 countries at the end of September 2011. Serving a global client base, it delivers hi-tech transactional services, consulting and technology services, systems integration and managed services. With its deep technology expertise and industry knowledge, it works with clients across the following market sectors: Manufacturing, Retail, Services; Public, Health & Transport; Financial Services; Telecoms, Media & Technology; Energy & Utilities.

Atos is focused on business technology that powers progress and helps organizations to create their firm of the future. It is the Worldwide Information Technology Partner for the Olympic Games and is quoted on the Paris Eurolist Market. Atos operates under the brands Atos, Atos Consulting and Technology Services, Atos Worldline and Atos Worldgrid.