apprenda

Private PaaS Enables the Hybrid Cloud Era

APPRENDA *White Papers*

Private PaaS Enables the Hybrid Cloud Era

CONTEXT AND PRIVATE PAAS ADOPTION

By abstracting applications from infrastructure, private Platform-as-a-Service (PaaS) bridges public Infrastructure-as-a-Service (IaaS) and internal IT to empower hybrid cloud strategies.

88% of enterprises are interested in private PaaS while 34% of enterprises are already implementing or evaluating private PaaS.[1]

EXECUTIVE SUMMARY

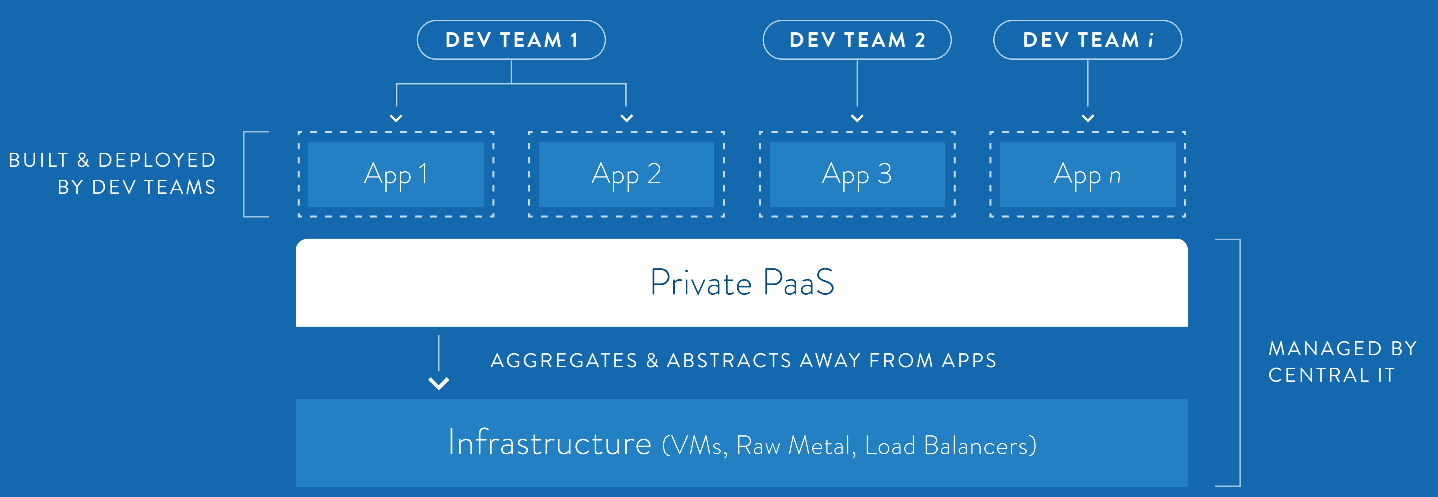
Custom software is now indispensable for business growth and competitive differentiation. Organizations need to reduce frictions between line-of-business application development and central IT in order to improve the efficiency of custom software development. Organizations also need to move forward on hybrid cloud strategies to stay agile. A private Platform-as-a-Service (PaaS) managed by central IT empowers developers by giving them the freedom and simplicity of a self-service, policy driven, world-class PaaS that overlays both internal IT and public cloud. In a 2015 Planning Guide, Gartner urges enterprises to start evaluating private PaaS in 2014 and 2015:

> *"PaaS improves developer productivity, reduces operational effort and increases hosting density. This value proposition is too compelling for large enterprises to ignore."*[2]

The word "private" in private PaaS means that the PaaS solution is managed by IT but can be installed on both publicly-managed (e.g. IaaS) and private infrastructure. In the hybrid or purely private model, private PaaS delivers:

- **EASY HYBRID CLOUD APP MIGRATION:** Private PaaS simplifies applications' back-and-forth migration between public clouds and internal IT by abstracting applications away from infrastructure and enforcing a common inheritable architecture in the different environments.
- **FASTER TIME TO MARKET:** Architecture patterns common to most applications and clouds are built into the PaaS as middleware to reduce coding time. The PaaS self-service console allows developers to upload compiled code and, with a button click, publish it to infrastructure dictated by policy automation.
- **INCREASED AGILITY:** Leveraging private PaaS simplifies application deployment and management and increases developer productivity through shared services.

- **AUTOMATED POLICY AND GOVERNANCE:** Private PaaS automates application provisioning policies. For example, an application can be developed in the public cloud and moved using policy to the internal data center for production.
- **REDUCED COSTS:** Private PaaS allows for greatly improved infrastructure utilization, removes many manual human configuration tasks and provides self-service interfaces.
- **STREAMLINED APPLICATION MANAGEMENT:** Private PaaS enables the organization to manage all its applications from a central location and never worry about being outside the bounds of IT governance.


DEV TEAM 1
DEV TEAM 2
DEV TEAM *i*
BUILT & DEPLOYED BY DEV TEAMS
App 1
App 2
App 3
App *n*
Private PaaS
AGGREGATES & ABSTRACTS AWAY FROM APPS
Infrastructure (VMs, Raw Metal, Load Balancers)
MANAGED BY CENTRAL IT

UNDERSTANDING PRIVATE PLATFORM-AS-A-SERVICE (PAAS)

Private PaaS (sometimes referred to as enterprise PaaS) is analogous to a single application server that instead overlays most infrastructure and provides built-in functionality for the cloud era. Apprenda's private PaaS is managed by central IT and may envelop both private data centers and public IaaS to create a layer of abstraction that allows easy application migration among the different infrastructures.

The PaaS offloads a majority of the heavy lifting related to deployment, configuration, and scaling of custom Web and SOA applications. Apprenda is app-centric, which means the application components are first-class citizens. In the Apprenda model, server infrastructure and operating system images are merely commodity resources. Apprenda stands up the configuration of the application to reduce the complexity of DevOps and IT automation. This architecture differs from the current application delivery model, where infrastructure and virtual machines (VMs) are primary, applications are subservient to infrastructure constraints, and complex DevOps and corporate workflows try to bridge the gap.

The Apprenda platform pools together infrastructure resources such as servers and networking components and exposes them to developers through a self-service mechanism, like a command line or graphical user interface. Apprenda's middleware components include high availability, scalability, load-balancing, and multi-tenancy to decrease development time for existing and new custom applications.

Apprenda uses lightweight containers for both Linux and Windows Servers to provide isolation, increased resource usage, and better scalability. A container is similar to a virtual machine, but does not require a separate guest operating system. Other resources, such as database servers and network components, can also be managed to varying degrees by the platform to automate provisioning and connectivity activities.

Private PaaS use cases include:

- **HYBRID CLOUD** – Apprenda acts as a control plane for applications in internal data centers and the public cloud. The application containers for the PaaS abstract the infrastructure details, making migration seamless, while the application policy placement can ensure the apps are hosted on the right infrastructure during each phase of the lifecycle.
- **CLOUD ENABLEMENT OF EXISTING AND NEW APPS** – Applications using Apprenda inherit cloud features such as scalability, elasticity, and high availability. New application development can take advantage of the PaaS API for advanced cloud features, such as self-monitoring of resource constraints and auto-scaling.
- **PRIVATE CLOUD STRATEGY** – Private PaaS should be included as part of an overall cloud strategy to reduce frictions between developers and internal IT. The enterprise PaaS may include IaaS (public or private) or may be deployed without those solutions.
- **MICROSERVICES AND APIS** – Apprenda is often used to manage API access and host microservices. API layers are built up to abstract access and tasks that cross disparate systems to enable business level developers to quickly create mash-up applications of enterprise services.

WHEN TO USE PRIVATE, PUBLIC, AND HYBRID CLOUD

Many organizations are pursuing a hybrid cloud strategy. One Gartner poll found that nearly 50% of enterprise organizations will implement a hybrid cloud by 2017.[3] Almost all organizations in the poll were pursuing private cloud because they wanted improved agility, reduced costs, and improved quality of services. When trying to mesh private cloud initiatives in the push for public cloud, most of those polled indicated that there were serious operational and technical challenges. These challenges are mitigated by Apprenda's private PaaS.

Public cloud is a model that has existed since at least 2006 and is often the one organizations are most familiar with. Public PaaS provides an application development platform while the service provider manages the infrastructure. Public PaaS has not been widely adopted in large organizations. These large enterprise organizations prefer private PaaS over public PaaS because public PaaS:

- Increases lock-in chances due to the dependence on the full IT stack from the provider
- Comes with greater cost than internal infrastructure and other public cloud initiatives
- Complicates regulatory or compliance issues
- Lacks enterprise-grade security or identity management features
- Fails to give specific control over the hardware and often does not meet specifications

On the other hand, public infrastructure-as-a-services (IaaS) is becoming a highly competitive, low cost market that has attracted many enterprise organizations. For example, both AWS and Microsoft Azure have been named Leaders in Gartner's Magic Quadrant[4] and have ample case studies.[5]

Use of IaaS resources from public providers to create a hybrid cloud is seen as a way to efficiently deal with variable usage and is potentially more cost-effective at times. Public IaaS allows organizations to maintain a high level of control over network and server configuration. Apprenda provides middleware, a policy and abstraction layer over both internal and external internal resources, giving operations the ability to extend the service capability yet maintain control.

COMMODITIZE COMPLEX ARCHITECTURE PATTERNS & SERVICES

In order to fulfill project requirements, enterprise application development requires complex architecture patterns and application services. Currently in organizations, development teams need to rewrite pieces of solutions any time a new requirement is defined. For example, should a project require in-memory caching, a cache is built or downloaded. The net result is that development teams spend significant amounts of time trying to build or source this functionality, causing not only a severe negative impact on time to market, but often a severe impact on software quality if the development team has little to no expertise in creating or integrating those subsystems.

Much like the application server did for web applications and the OS did for desktop, PaaS, as a runtime and abstraction, helps aggregate common functionality rather than leaving it to development teams to re-invent the wheel each time.

Problems arise when a development team needs to tackle more complex architectural issues such as high availability and resiliency, scalability, or modern cloud architecture patterns like multi-tenancy. These architectural facets increase complexity exponentially, slowing down project roadmaps and adding to a project's overall brittleness.

In a private PaaS model, these services are provided by the platform. Guest applications can either explicitly consume these services via API calls or implicitly take advantage of them by virtue of their being deployed in the PaaS container. Rather than tackling complex development issues, guest applications on a private PaaS can simply expect the PaaS to provide commoditized, shareable standard services. Development teams benefit by being able to narrow their development focus to the business requirements of a project, speeding up development time and improving software quality by accessing well-proven, well-integrated components. As a result, reductions in development cycle investments of 30%-90% can be typically experienced through leveraging a PaaS.

A private PaaS approach to commoditizing standard architectures and services is much more effective than using shared libraries. Shared libraries tend to require active participation by the development team to adopt since they are not part of a runtime. Much like a Java or .NET application inherits memory management by virtue of running in its runtime, PaaS guest applications implicitly inherit architectures and capabilities by virtue of running on a PaaS.

IMPLICIT FUNCTIONALITY (INHERITED)

- Scalability
- High Availability
- Configuration Independence
- Dynamic Error Handling
- Advanced Message Routing
- Requirements Driven App Placement
- Contextual Logging
- Session Replication
- Basic Multi-tenancy
- Lifecycle Management
- Infrastructure Independence

EXPLICIT FUNCTIONALITY (API CALLS)

- Distributed Caching
- Messaging Services
- Advanced Logging
- Advanced Multi-tenancy
- Application Self Manipulation (e.g. auto-scaling an app that is self-aware)
- Advanced Authorization

BUILDING VS. BUYING PAAS

Apprenda helps organizations realize the value of an enterprise PaaS without the complex and expensive engineering required to build a PaaS. According to Gartner analyst Richard Watson,

> *"[Organizations should not] underestimate the scale of the engineering that goes into building out this layer of cloud platform management services."*[6]

Apprenda brings PaaS to the enterprise rather than the enterprise to the PaaS. Apprenda is a PaaS solution used by McKesson and JPMorgan Chase to deploy a truly multi-tenant PaaS on existing infrastructure. For organizations that have already built a PaaS, many peers have found immense cost, structural, and efficiency gains by using Apprenda instead.

Apprenda works with existing technology and is easily extensible. This architecture design allows organizations to use existing investments and expertise. It is the only enterprise PaaS on the market that supports both Red Hat's JBoss and Apache Tomcat servers. Apprenda works with public cloud investments like AWS' data warehousing RedShift, Azure's messaging bus, and others to provide developers with servers beyond compute and storage.

PRIVATE PAAS BENEFITS

Easy App Migration in Hybrid Cloud

Currently, most enterprise organizations are moving forward with private cloud plans and are implementing hybrid cloud. The different environments are not only spread across geographies, but also amongst external and internal hosting. Organizations, in yet another level of abstraction, divide each of the different resources by use cases, such as dev/test, staging and production.

Apprenda can help enterprises to combine resources across all clouds into a single, logical resource pool and apply flexible and powerful application deployment policies. These policies can be used to automatically map applications to infrastructure based on multiple factors including use case, security, compliance, business unit, and geography.

Many think the most important aspect of hybrid cloud is that it allows enterprises to very easily add public resources during times of high demand or service outage. Apprenda enables that on-demand provisioning and flexibility to efficiently orchestrate across internal and external clouds. For example, dev/test can be hosted on the public external cloud, whereas an application can move and become hosted internally when it goes into production.

Faster Time to Market

Huge advancements have been made in enterprise application development, allowing enterprise developers to write custom service-oriented architecture (SOA) and web applications faster than ever. These advancements have allowed different lines of business and external business partners to develop critical software faster. Unfortunately however, once a development team has written an application, the process of provisioning an application and having it ready for use tends to be slow and cumbersome.

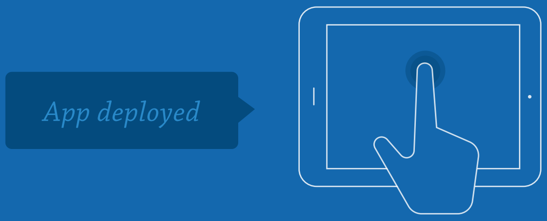
Typically, a development team can expect anywhere from 10-60 business days, with an average of around 30 days, to have an application provisioned on internal IT. During this time period, IT staff will provision a server or virtual machine, prep an operating system instance for each application layer, ensure that network dependencies such as DNS entries and load balancer configurations are updated, and also ensure that the application bits are installed and configured properly. If an application took 4 months to develop, the typical total time to market still averages 5-6 months - an increase of 25%-50% over the actual development phase! Minor changes to the application, such as new releases or necessary infrastructure updates, can each take just as long, adding significant direct and opportunity cost to an application over its lifetime.

The primary cause of these unfortunate economic profiles is the lack of architecture conformity between the different applications created by the different development teams. This leads to a scenario where each application is an exception to the rule and has different expectations and configuration needs.

In a typical enterprise IT environment, the various development teams likely made different decisions on how to handle concerns such as authentication, authorization, session management, caching, etc. Some may have built subsystems to handle these various cross-cutting application concerns, others may have bought off-the-shelf components, and still others might have used open source components. Because of these permutations, IT cannot automate management or provide broad-stroke policy enforcement. Additionally, developers have no way to expect generalizable value-added services from IT, since IT has no architecture expectations of the apps. On change requests, the entire approval process must be repeated and the static nature of the deployed state of the application means that drastic changes may require significant reconfiguration of the application.

Across the application portfolio, the lack of architecture conformity creates significant costs in managing and maintaining application deployments. Clearly, one solution to this problem is to establish an internal architecture and development standard, but enforcement of this standard is difficult and unlikely, particularly at scale. Furthermore, such a standard typically offers little adoption incentive, and is viewed as a barrier, rather than a benefit, by developers.

Apprenda offers a solution that establishes architecture conformity across all guest applications. The PaaS, and not human actors, provides fundamental workflows such as application provisioning, patch management, and middleware. IT can manage the PaaS and not each individual application. Developers can leverage self-service portals to ensure that they can deploy their applications in five minutes, rather than 30 days. Developers can also tap into platform-level services such as caching, authentication, and auto-scaling to drastically simplify projects.



Increased Agility

Routine tasks like provisioning appropriate hardware can be time-consuming and frustrating for enterprise developers. Time that could be spent coding is instead spent in lengthy discussions with IT operations teams to get server access and work out configuration requirements. Private PaaS eliminates these delays and frustration.

Through powerful developer self-service via APIs, standard IDE tooling, and web portals, Apprenda automates many aspects of the application lifecycle management process, spanning across creation, deployment, scaling, patching, and versioning.

Tasks that previously took hours, days, or weeks are reduced to seconds. Unlike IaaS, developers can focus on their code and applications, while the platform abstracts away infrastructure details like servers, load balancers, and storage. Teams of developers who are working together across one or more applications can also use the developer portal.

Apprenda ensures that web applications automatically inherit availability and guaranteed quality of service. Applications hosted on light-weight containers are liberated from the infrastructure and able to rehydrate on other servers, virtual machines, or public cloud when hardware fails. Load balancing across multiple instances on the platform fabric ensures that application experience is never diminished. Private PaaS' responsiveness to degradation of services from application telemetry monitoring gives users the performance and SLA they demand.

Policy and Governance Automation

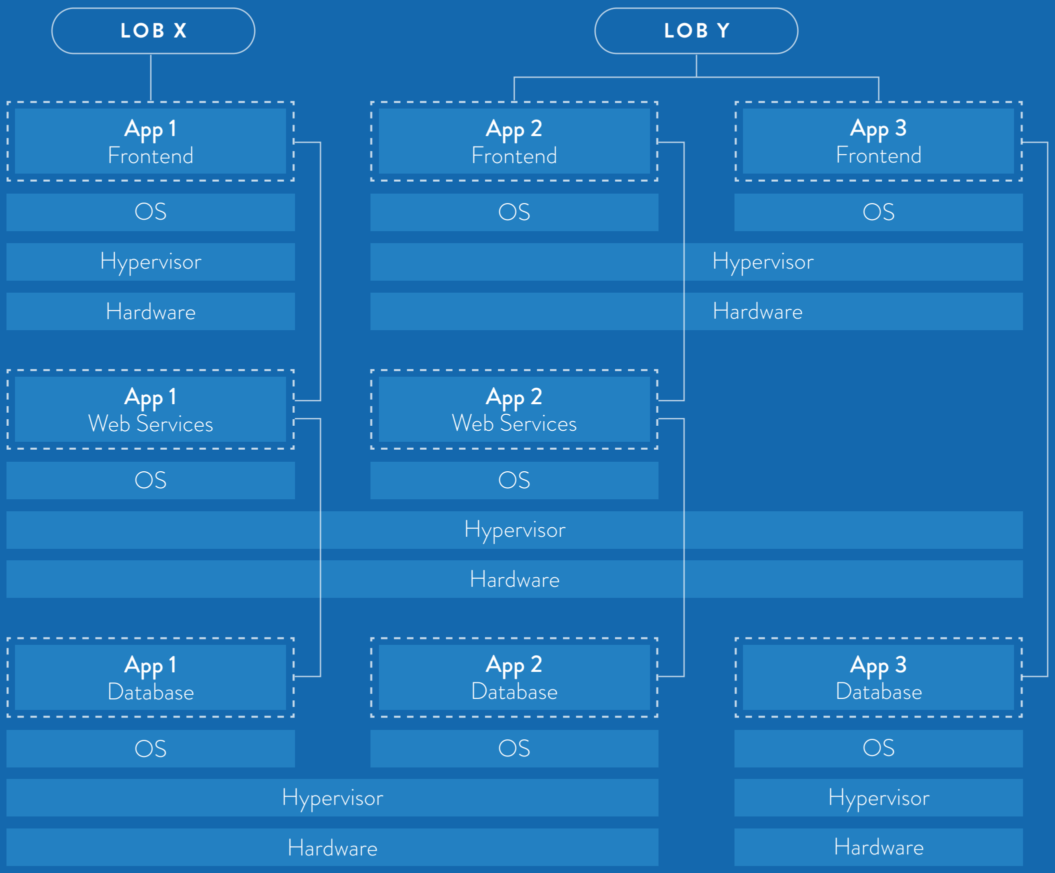
When enterprises manage large numbers of servers in a single, logical resource pool, situations often arise where applications need to be mapped to very specific infrastructure. In many cases, this is due to security or regulatory compliance.

Private PaaS leverages defined application deployment policies to enable a fine-grained mapping of applications /application components to infrastructure, which is based upon specific, configurable properties. Instead of creating multiple silos with different rules and configurations based upon business needs, private PaaS enables enterprises to consolidate applications on shared infrastructure, while using sophisticated deployment policies to honor specific business, security, or legal requirements.

Reduced Cost

Rather than relying on virtual machines as a container for providing application isolation, Apprenda uses a custom container model that runs inside operating system instances to isolate applications from one another: increasing server and virtual machine utilization and reducing sprawl. This fine-grained isolation is used to sub-divide each OS instance into multiple segments, increasing utilization and reducing hardware and licensing costs. Resource policies are the defining unit that allow platform operators to define how large these segments are from a CPU, memory, and storage perspective, and are published to developers as the set of options they may choose from when allocating resources to their applications. In addition to the significant cost savings, these resource policies also allow applications to go into production faster and ensure that the enterprise web hosting team is a source of developer satisfaction.

TRADITIONAL TOPOLOGY



LOB X

LOB Y

App 1 Frontend

OS

Hypervisor

Hardware

App 2 Frontend

OS

App 3 Frontend

OS

Hypervisor

Hardware

App 1 Web Services

OS

App 2 Web Services

OS

Hypervisor

Hardware

App 1 Database

OS

App 2 Database

OS

Hypervisor

Hardware

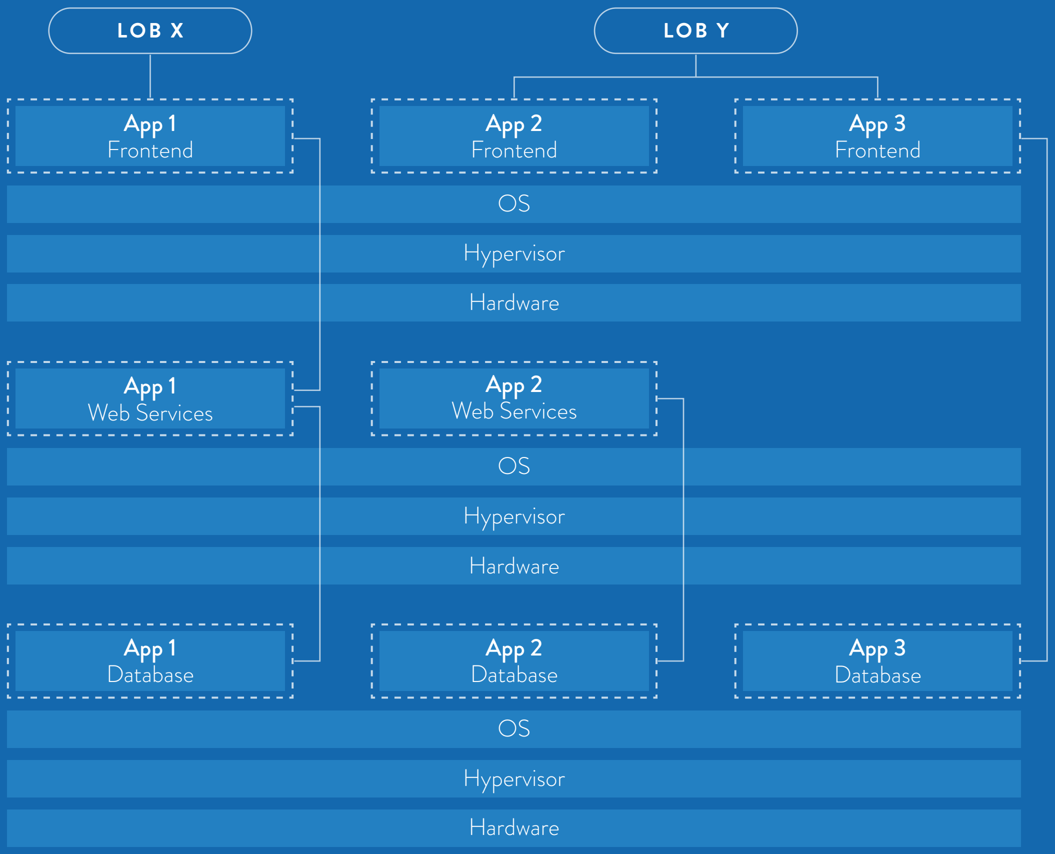
App 3 Database

OS

Hypervisor

Hardware

PRIVATE PAAS TOPOLOGY



LOB X

LOB Y

App 1 Frontend

App 2 Frontend

App 3 Frontend

OS

Hypervisor

Hardware

App 1 Web Services

App 2 Web Services

OS

Hypervisor

Hardware

App 1 Database

App 2 Database

App 3 Database

OS

Hypervisor

Hardware

Streamlined Application Management

Successful application projects are not "deploy once and forget" applications. Application owners will more than likely upgrade and change an application on a regular basis to provide end-users with new value and keep them ahead of the competition.

A 30-step manual process for updating an application will not scale well over time. Poor application lifecycle management processes are not only costly, but they can cause business to grind to a halt when end-users lose faith in applications because of poor quality control and downtime.

Well-defined application lifecycle management processes, supported by sophisticated systems for managing the complexities of patching and updating a live application, are key to remaining agile and having the capability to respond to customers and the market instantly. Development platforms should integrate with world-class ALM software that is currently used in the enterprise to provide enhanced capabilities.

FIND OUT FOR YOURSELF

This paper has outlined how Private PaaS enables the Hybrid Cloud era. If you are interested in finding out more about Apprenda's Private Platform-as-a-Service, the best way is to experience it for yourself.

Apprenda offers three simple ways to do this:

1. Sign up for the monthly Open Demo webinar
2. Request access to Apprenda Express – and explore the free version of the platform
3. Arrange a Proof-of-Concept (PoC) for your organization

All three can be accessed by visiting the Apprenda website or contacting us.

ABOUT APPRENDA

Apprenda is the leading enterprise private platform as a service (PaaS) powering the next generation of software development in public, private and hybrid clouds. As a foundational software layer and application run-time environment, Apprenda abstracts away the complexities of building and delivering modern software applications, enabling enterprises to turn ideas into innovations faster. With Apprenda, enterprises can securely deliver an entire ecosystem of data, services, applications and APIs to both internal and external customers across any infrastructure. From the world's largest banks like JPMorgan Chase to healthcare organizations including McKesson and AmerisourceBergen, Apprenda's clients are part of a new class of software-defined enterprises, disrupting industries and winning with software.

[1] *"Detecting a Red Shift in Enterprise Software Development" (Press Release)*

[2] *2015 Planning Guide for Cloud Computing*

[3] *Gartner Says Cloud Computing Will Become the Bulk of New IT Spend by 2016*

[4] *Gartner Magic Quadrant for Cloud Infrastructure as a Service, May 28 2014*

[5] *AWS Case Studies*

[6] *Platform as a Service (PaaS) for Private Cloud*

Apprenda, Inc.
433 River Street
Troy NY 12180

WWW.APPRENDA.COM
1.877.PAASWEB