A Reference Architecture for the Apprenda Platform

Document Revision 1.0 for Platform Version 5.0
Summary.................................................................3
What Is The Apprenda Platform, Logically Speaking?..............................................4
Planning for the Apprenda Platform ...........................................................................6
Infrastructure Hardware .............................................................................................7
Networking ...................................................................................................................8
Operating Systems and Software ..................................................................................9
A Production Ready Apprenda Platform: Recommended Hardware/Software ................9
Reference Architecture Topology (Single Cloud) ........................................................12
Apprenda Software Inventory .....................................................................................17
Hybrid Cloud Installations ..........................................................................................20
Reference Topology (Hybrid Cloud) .............................................................................21
Maintenance, Upgrades, and Infrastructure Modifications ..........................................22
Summary

Apprenda created and maintains this reference architecture guide to relay our recommendations for running production Platforms-as-a-Service with the Apprenda Platform technology. Apprenda is a Platform as a Service (PaaS) "engine" and framework that empowers any organization to operate a self-service cloud computing Platform for building, deploying, and running applications. Apprenda can be deployed on public or private infrastructure, providing freedom from lock-in and tremendous increases in developer productivity, infrastructure utilization, and app management agility. Whether your deployment of the Apprenda Platform serves as a private PaaS within a large organization, a hosting Platform for cloud-based SaaS applications, or a hosted public PaaS (all are valid uses of the technology), you’ll find useful and relevant information in this document.

Who Should Read This Reference Architecture?

This document is intended for enterprise IT departments, cloud service providers, hosting companies, and ISVs who intend to deploy and operate an Apprenda Platform instance. The information contained in this document will help you make hardware and software decisions to provision and configure the recommended initial footprint of an Apprenda-based solution. You should have some knowledge of cloud computing terminology, hardware configuration, networking, and Microsoft Windows-based server solutions. You should also be familiar with Linux-based server solutions and Oracle administration should you choose to configure your Apprenda Platform with support for these technologies.

Your Next Steps

Upon becoming familiar with this reference architecture, you should contact Apprenda’s support team at support@apprenda.com to construct a detailed deployment plan based on your specific needs.
What Is The Apprenda Platform, Logically Speaking?

Apprenda is an application server and container for .NET and Java applications where application components are first-class citizens of the architecture. Apprenda aggregates Windows and Linux infrastructure into a single logical layer that is a self-service computing Platform for one or more development teams. An application running on Apprenda is one that is composed of some number of traditional ASP.NET or Silverlight UIs, Windows Communication Foundation (WCF) web services, Windows Services, Java Web Applications packaged as WARs, and SQL Server-based or Oracle-based databases. Apprenda provides an execution fabric for your application components of all types, server farm management for a network of peer-based nodes, an inheritable distributed architecture that makes your .NET or Java application a scalable, high performance entity, and a management system for dealing with the various intricacies of an application’s lifecycle over time. Additionally, Apprenda offers various APIs and frameworks that provide capabilities such as message brokering, distributing caching, and application metering.

The best way to think of Apprenda is from a stack-wise point of view. Apprenda is analogous to an operating system, an application server like IIS or Tomcat, or any other pure software layer
that sits above physical resources (such as hardware or a network) and acts as a host and execution backdrop for an application that lives on top of it. In the case of an operating system, everything from memory subsystems to disk access and thread scheduling is provided as a set of services to the applications built for the OS. When you write code, you implicitly assume that the OS will take care of fundamental and difficult problems for you, and that the hardware resources are abstracted away from your code to some extent. Runtimes like the .NET CLR or the JVM provide further value through more abstraction for a large subset of applications being developed. Apprenda is similar in both form and spirit to an operating system or runtime.

Here are some of the Apprenda Platform’s notable features:

- A multi-machine application server and runtime for hosting and managing applications.
- REST and SOAP exposure of API endpoints for web service integration and automation.
- Instrumentation and transformation engines that virtualize your application’s components across tenants so that you can achieve high efficiency. This means that parts of your application, such as your database, can be written as if they were for one tenant or customer (e.g., storing row data naively with no notion of tenant ownership), but, when running on Apprenda, can be transformed to behave in a multi-tenant fashion (e.g., your database will be instrumented to support multiple tenant data storage in a way that is transparent to your application queries). This instrumentation is optional, and a development team may choose to use it in scenarios where their application architecture calls for it.
- A user and tenant management layer that your applications can leverage so that the idea of users and account information can be shared across applications.
- An API and set of associated services that provide fundamental building blocks for user/account management and provisioning (either by you or through pre-built storefront widgets that allow your customers to self-provision), caching, and message brokering (among other things).
- Enterprise class server mechanics such as automatic high availability of web services, linear scale-out across your server farm, automatic and seamless IIS-based web farming, and failure isolation for both Windows and Linux servers.
- A Resource Allocation Policy (RAP) management system that allows IT to “slice” the sum of the compute power under control by the Apprenda Platform into discrete units of allowed CPU and memory allocation. Development teams can then choose these “slices” and assign them to their application components. The Platform will monitor and throttle resource utilization based on assigned policies. This gives IT advanced capacity planning capabilities and development teams concrete expectations while shielding them from the underlying infrastructure in the spirit of a PaaS.
- A set of web portals and tools to manage your applications and customers.
- An administrative web portal that gives IT a detailed view of how the Platform “sees” the infrastructure and how it has distributed application components accordingly.
Planning for the Apprenda Platform
Because the Apprenda Platform fills such a well-defined and abstracted role in a complete PaaS solution, you have many options for the infrastructure configuration on which it will run. In some ways, you should approach this planning process as you would a standard hosting or cloud solution rollout. Take into account the things about your operations that are most important to you. Disaster recovery? Scalability and/or high availability? Ease of access? Before you read through this reference architecture, take a few minutes to identify the aspects of your cloud strategy that will be most important to you, or your customers, or your internal development teams.

Public PaaS vs Private PaaS
Your intended use of the Apprenda Platform will play into how you apply the things you’re learning from this reference architecture guide. Perhaps you’re a cloud service provider looking to operate Apprenda as a public PaaS - allowing your customers to develop apps and deploy them into your cloud ecosystem. If so, you may encounter a high number of deployed apps from smaller software companies with many of them tapping into the Platform’s SaaS engine for metering and subscription management. Your focus may be on the availability of Platform business services and meeting the SLAs you offer your customers. On the other hand, if you’re in IT at a large enterprise and planning to run Apprenda in-house for your development teams and the thousands of apps they offer to employees, you’ll probably value features like the Resource Allocation Policy (RAP) and utilization engines instead. In yet another scenario, you may be looking for a way to incorporate the best features of public and private PaaS into your IT landscape. In this case, Apprenda’s hybrid cloud capabilities are invaluable to your organization. How you interpret this reference architecture and formulate an implementation should be informed by your intended use case(s), cloud computing goals, and primary Platform constituents.

Optional Platform Infrastructure
The Apprenda Platform per se consists of a series of Windows Services, an Apprenda Core DB deployed on an instance of MS SQL Server, and a number of WCF services and .NET UIs. This means that a single Windows server with IIS and SQL Server is capable of hosting an Apprenda Platform installation—as well as three-tiered .NET guest applications—but it also means that Windows-based infrastructure is required for core Platform functionality. Conversely, while additional infrastructure can be configured for the Platform in order to support Java components and Oracle-based storage for guest applications, such additions are superfluous to core Platform functionality. While hardware/software recommendations have been made for supporting non-essential Platform infrastructure, they are noted as “optional” in this reference architecture.

A Note about Infrastructure Sizing
The Apprenda Platform is architecturally interesting in that it was designed to manage as little as one server or scale to manage thousands: from your laptop to a datacenter with thousands of CPUs and terabytes of RAM. The Platform’s job is to maintain a comprehensive view of the infrastructure under its control and usher guest application components around to suitable host servers. Because applications are first-class citizens to the Apprenda Platform, sizing relies quite heavily on the requirements of guest apps. The Platform’s resource
allocation and utilization tracking engine exists to help operators “slice” the infrastructure into trackable units, which is a great aid in analyzing true app resource usage vs allocation and thus infrastructure sizing and capacity planning. This reference architecture is a starting point.

**Infrastructure Hardware**

The hardware needed for the deployment of the Apprenda Platform in this reference architecture covers various Platform roles: web servers, application servers, database servers, and ancillary servers. Our reference architecture is intended to cover deployment scenarios for high availability. We’ve configured away all single points of failure. In most cases, the Platform’s architecture accounts for high availability; however, in the few cases where the product’s design itself does not supply HA, we rely on standard practices in Windows environments such as SQL Server clustering.

**Physical Servers or Virtual Machines?**

The Apprenda Platform sits atop the Microsoft Windows and Linux operating systems, and as such, relies on the OS to abstract away specific hardware interaction and concerns. Apprenda can run on bare metal, virtual machines, IaaS, or a hybrid approach so long as OS and networking requirements are satisfied. This reference architecture includes a recommendation for spreading VMs across hosts for failure containment should your approach be virtualized.

**Using a SAN or NAS for Storage**

The use of a SAN or NAS is recommended as the physical storage mechanism for the following:

- The Platform Repository: a network share that will serve as the central storage location for all Platform and guest application binaries. This location must be configured prior to installation, should be specified as a network path in the Apprenda Installer, and must contain the following three folders:
  - Applications
  - Apprenda
  - SAC
- A shared IIS configuration network share location: a folder that must be set up prior to installation and then specified in the Apprenda Installer. The Installer will then create the shared configuration per se required to support Load Manager high availability. It is acceptable to configure this location on the same share as the Platform Repository.
- A central location for SQL failover cluster storage; this will be required to set up the recommended SQL failover clusters prior to running the Apprenda Installer
Networking
There are three primary networking concerns when working with the Apprenda Platform:

- Networking the participating servers together in a primary LAN configuration so that they have uninterrupted access to each other via TCP.
- Providing HTTP and HTTPS access to external parties for the purpose of accessing guest application components running atop and within your Platform.
- If installing on multiple clouds where one or more cloud is outside of the primary LAN, establishing a VPN tunnel to enable cross-cloud communication (see the Hybrid Cloud Installation section below for more information).

Establishing a Private LAN for Your Platform Servers
The servers being deployed to support a unique instance of the Apprenda Platform should be setup in a DHCP network configuration whereby they obtain IP addresses from a gateway server or networking appliance. For Platform operations, the servers communicate with each other using DNS-resolvable hostnames, and therefore require an internal DNS structure to be in place that resolves IP assignments. This can be an existing DNS infrastructure or one dedicated to the Apprenda Platform servers. Our reference architecture specifies a dedicated Domain Controller that also acts as the primary DNS server for all servers on one Platform instance’s private network.

Inbound Traffic from External Sources (HTTP, HTTPS, etc.)
The first point of contact for any inbound request to a Platform hosted application component URL is one of the Load Manager servers. This is true for UI workloads running on frontend servers, service workloads running on application servers, and Java Web application workloads running on Linux servers. We also recommend placing a hardware load balancer of your choice in a DMZ in front of the entire environment, routing traffic via ports 80 and 443 (HTTP and HTTPS respectively) between the Load Manager servers in either an active/active (HA) configuration or an active/passive (failover) configuration. As in standard web hosting environments, the Platform, its portals, and guest application UIs will all be accessible at a URL specified by you during Platform installation. You will need to configure external DNS that routes requests for your Platform URL to the hardware load balancer or the Load Manager server(s) directly. Traffic to Load Manager servers from all Apprenda-provided utilities and products occurs over ports 80 and 443. The Apprenda Load Manager distributes requests to the proper frontend content servers based on guest app workload location, and uses ports 8080 and 8081 by default internally (this is considered “inside” Apprenda proper). A typical HTTP request on port 80 will reach the Load Manager and be routed internally on port 8080 to a frontend content server that has the appropriately requested guest application installed and configured (this may be multiple web servers due to web farming). The response is sent back out through the proxy on port 80 so translation and internal routing is transparent to the requester.
Operating Systems and Software

The Apprenda Platform makes use of the Microsoft family of technologies for core Platform functionality, as well as specified flavors of Linux for optional Java support. Optional Oracle RDBMS installations are OS agnostic. Fundamentally, its goal is to perform ordinary application deployment tasks more quickly and more efficiently than ever before possible. The Platform works well within the following software specifications, classified by role. In most cases, the required software configuration can be achieved on Windows Servers by using the Server Administration tools and applying server Roles and specific Features. The Apprenda Platform Installer will validate your configuration prior to installation. In some cases, software customizations and/or unique configurations will require changes to the Apprenda Platform that the Installer cannot accommodate. If you think you have the need for a specialized install, contact support@apprenda.com. In order to avoid incompatibilities, no additional software should be installed on your Platform servers unless approved by Apprenda.

A Production Ready Apprenda Platform: Recommended Hardware/Software

<table>
<thead>
<tr>
<th>Platform Role</th>
<th>Description</th>
<th>Hardware Recommendation</th>
<th>Software Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Manager</td>
<td>Runs IIS for the purpose of acting as a reverse proxy to servers that host guest application UI workloads.</td>
<td>• Single socket dual core CPU&lt;br&gt; • 4GB RAM&lt;br&gt; • 15K rpm 80GB SAS HDD, RAID 1&lt;br&gt; • 2 NICs - 1Gbit or greater&lt;br&gt; • Network share for housing Shared IIS Configuration files (this can be the same share as the Platform Repository)</td>
<td>• Windows Server 2012/2012 R2 Datacenter or Enterprise&lt;br&gt; • .NET Framework 4.5&lt;br&gt; • IIS 8/8.5 (Web Server Role)&lt;br&gt; • MS Application Request Routing 2.5 and its dependencies:¹&lt;br&gt;   ○ URL Rewrite Module 2.0&lt;br&gt;   ○ Web Farm Framework 1.1&lt;br&gt;   ○ External Cache 1.0</td>
</tr>
<tr>
<td>Platform Coordination Node</td>
<td>Runs the Apprenda Control agent that orchestrates routing and deployment of guest application workloads. Requires an odd number of instances.</td>
<td>• Single socket dual core CPU&lt;br&gt; • 8GB RAM&lt;br&gt; • 15K rpm 80GB SAS HDD, RAID 1&lt;br&gt; • 2 NICs - 1Gbit or greater</td>
<td>• Windows Server 2012/2012 R2 Datacenter or Enterprise&lt;br&gt; • .NET Framework 4.5</td>
</tr>
</tbody>
</table>

¹ The optimal way to install ARR 2.5 is through the MS Web Platform Installer. If this is not a viable solution, manual installation instructions can be found at http://blogs.iis.net/wooyoo/archive/2011/04/20/how-to-install-application-request-routing-arr-2-5-without-web-platform-installer-webpi.aspx
<table>
<thead>
<tr>
<th>Platform Role</th>
<th>Description</th>
<th>Hardware Recommendation</th>
<th>Software Recommendation</th>
</tr>
</thead>
</table>
| Cache Node                       | Runs the Apprenda Caching service that provides in-memory caching for Windows and Linux servers. | • Single socket dual core CPU  
• 8GB RAM  
• 15K rpm 80GB SAS HDD, RAID 1  
• 2 NICs - 1Gbit or greater | • Windows Server 2012/2012 R2 Datacenter or Enterprise  
• .NET Framework 4.5 |
| Windows Application Server       | Employs the .NET runtime to host guest application WCF and Windows Services. | • Single socket quad core CPU  
• 8GB RAM  
• 15K rpm 80GB SAS HDD, RAID 1  
• 2 NICs - 1Gbit or greater | • Windows Server 2012/2012 R2 Datacenter or Enterprise  
• Microsoft SQL Server 2012 Shared Management Objects and requisite System CLR Types |
| Windows Web Server               | Runs IIS for the purpose of hosting .NET guest application UI workloads.     | • Single socket quad core CPU  
• 8GB RAM  
• 15K rpm 150GB SAS HDD, RAID 1  
• 2 NICs - 1Gbit or greater | • Windows Server 2012/2012 R2 Datacenter or Enterprise  
• .NET Framework 4.5  
• IIS 8/8.5 (Web Server Role) |
| MS SQL Node                      | Employs SQL Server to support guest application storage needs as well as the Apprenda Core Database. | • 2 x quad core CPU  
• 32GB RAM  
• 15K rpm 150GB SAS HDD, RAID 10  
• 2 NICs - 1Gbit or greater  
• SAN or NAS space for failover cluster storage | • Windows Server 2012/2012 R2 Datacenter or Enterprise  
• Microsoft SQL Server 2012 configured as a failover cluster |
| Platform Repository              | Network share that will serve as the repository for all Platform and guest application binaries. | • SAN or NAS  
• Network share containing the following folders:  
  o Applications  
  o Apprenda  
  o SAC  
• 2GB for Platform binaries plus storage for projected guest application binaries | |
<p>| | | | |
|                                 |                                                                             |                                                                                          |                                                                                             |</p>
<table>
<thead>
<tr>
<th>Platform Role</th>
<th>Description</th>
<th>Hardware Recommendation</th>
<th>Software Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Oracle RDBMS Installation</strong></td>
<td>2 x quad core CPU 8GB RAM 15K rpm 150GB SAS HDD, RAID 10 2 NICs - 1Gbit or greater</td>
<td>- Oracle Database 11g</td>
<td></td>
</tr>
<tr>
<td>(Optional)</td>
<td>Runs Oracle to support guest application storage deployments</td>
<td>2 NICs - 1Gbit or greater</td>
<td></td>
</tr>
<tr>
<td><strong>Quantity:</strong> 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Linux Server</strong></td>
<td>Single socket quad core CPU 8GB RAM 15K rpm 80GB SAS HDD, RAID 1 2 NICs - 1Gbit or greater</td>
<td>- Red Hat Enterprise Linux 6</td>
<td>- Libcgroup</td>
</tr>
<tr>
<td>(Optional)</td>
<td>Runs Tomcat through Apprenda’s custom Linux container to host Java Web Application workloads (WARs).</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Quantity:</strong> 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>AD FS Node</strong></td>
<td>Single socket quad core CPU 8GB RAM 15K rpm 80GB SAS HDD, RAID 1 2 NICs - 1Gbit or greater</td>
<td>- Windows Server 2012 Datacenter or Enterprise</td>
<td>- .NET Framework 4.5</td>
</tr>
<tr>
<td>(Optional)</td>
<td>Runs the Apprenda Federation Service in conjunction with AD FS, thereby allowing Identity Federation during user authentication. This server will also act as a Windows Application Server.</td>
<td>- IIS 8 (Web Server Role)</td>
<td>- Active Directory Federation Services Role configured as an AD FS Web Farm</td>
</tr>
<tr>
<td><strong>Quantity:</strong> 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Domain Controller and DNS Server</strong></td>
<td>Single socket dual core CPU 8GB RAM 15K rpm 80GB SAS HDD, RAID 1 2 NICs - 1Gbit or greater</td>
<td>- Windows Server 2012/2012 R2 Datacenter or Enterprise</td>
<td>- .NET Framework 4.5</td>
</tr>
<tr>
<td><strong>Quantity:</strong> 2</td>
<td>Uses Microsoft’s Active Directory and DNS server technologies to provide network-level accounts and DNS routing.</td>
<td>- Active Directory Server Role</td>
<td>- Active Directory Server Role</td>
</tr>
<tr>
<td><strong>Note that this requirement may already be satisfied by existing infrastructure.</strong></td>
<td></td>
<td>- DNS Server Role</td>
<td></td>
</tr>
<tr>
<td><strong>Managed Switch</strong></td>
<td>Establishes a 1Gbit or greater LAN between the machines.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Quantity:</strong> 1</td>
<td><strong>Note that this requirement may already be satisfied by existing infrastructure.</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Reference Architecture
Topology (Single Cloud)

The following is a holistic look at the topological construction of our Apprenda Platform reference architecture including servers, clustering, and networking. Note that the Apprenda Platform is architecturally in tune with the notion that applications are first-class citizens. The Platform is carefully designed to allow IT to independently scale different hosting tiers. This allows you to react to the demands of guest applications in a very scalable and granular way. As your cloud grows (or contracts), you have deep knowledge of actual component utilization and can manage accordingly. Have a lot of guest application ASP.NET websites, but not a lot of WCF service workloads? You’ll scale the Windows Web Servers differently than you will the Windows Application Servers. You can also scale Linux node independent from Windows nodes to accommodate differences in Java app deployment adoption vs .NET.

The Entire Architecture
Here’s a look at the entire stack for our reference architecture. Each Platform role is discussed in greater detail on the following pages. Not pictured here are the Primary and Backup Domain Controllers, and the DNS server(s). These roles may be served by existing infrastructure outside the Apprenda Platform architecture.
Load Managers
On the Apprenda Platform, the Load Manager (best described as a reverse proxy) serves as the initial receptor of an incoming HTTP requests. IIS configuration on the Load Manager is managed by the Apprenda Load Manager Service, a Windows service that creates and modifies URL Rewrite rules as various guest app components are deployed to internal frontend content servers. The Load Manager Service ensures the appropriate routing of inbound requests based on request URL patterns.

For high availability purposes, we recommend a topology consisting of two Load Manager servers, which requires the use of a shared IIS configuration housed in a network share. In order to optimize performance, a hardware load balancer should be placed in front of these Load Managers.

Platform Coordinator Nodes
Coordination of guest application workload deployment is handled by these nodes. The servers run a custom implementation of Apache ZooKeeper, running as a Windows service. Per the Apache ZooKeeper website, ZooKeeper is “a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services.” On Apprenda, the Platform Coordinator Nodes maintain knowledge of the topology of guest application workloads across all nodes, as well as any shared configuration in use by Apprenda components.

An optimal Platform installation requires an odd number of Platform Coordination Nodes, as a majority of extant Platform Coordination Nodes ((n+1)/2, where n=the number of nodes) must be up and running in order for the Platform to function properly. We suggest starting with 3 dedicated Platform Coordination nodes, which will allow the Platform to function as long as any 2 nodes are running; additional nodes can be added as needed once the environment is up and running.

Cache Nodes
These nodes house the Apprenda Caching Windows service, a distributed Redis-based in-memory caching implementation for all Platform Windows and Linux servers. Each Cache node can support one instance of the Caching service per processor core; the number of processor cores/Caching service instances should be used as the number of service ports specified for the Cache in the Apprenda Installer. We recommend 2 dedicated Cache nodes with 2 cores/Caching service instances each to help with load distribution and mitigate infrastructure failure risk.
**Windows Application Servers**

All Windows Application Servers host the Apprenda Host Controller and Apprenda Service Container Windows services. The Service Container enables the hosting of WCF and Windows services, thereby allowing both key Platform and guest application service components to be hosted on these servers.

It is necessary that at least one Windows Application Server per cloud host Appenda’s Storage Controlling Services, which interfaces with SQL Server and Oracle to configure databases. These servers are required to have SQL Server Management Objects (SMO) 2012 installed. At installation, the Platform will mark any Windows Application Servers with SMO installed as capable of hosting the Storage Controlling Services and will deploy this component to those servers. It will install the required SMO version on a single Application Server if no suitable host is found. In order to ensure that the Storage Controlling Services is highly available, we recommend installing a supported version of SMO (version 11.0 or higher) on two servers that will be designated as Application servers prior to running the Apprenda Installer, as this will result in both servers being designated as Storage Controlling Services hosts. As needed, after installation additional Application servers can be configured as Storage Controlling Services hosts by installing SMO on the servers and then designating them as such in the System Operation Center (SOC).

**Windows Web Servers**

Windows Web Servers are frontend web servers that host .NET based UIs through IIS. Via portals, the Apprenda Platform allows developers and Platform operators to create ad-hoc web farms for .NET guest application UI workloads at any time (as long as there is sufficient infrastructure). The sum of your Windows Web Servers represent the compute power of your Platform specifically for hosting .NET guest application UI workloads.

It should be noted that all Windows Web Servers are capable of hosting WCF and Windows services. This is necessary for the nodes to host the Presentation Controlling Services (see the Apprenda Software Inventory below), which allows management of .NET UIs in IIS. As such, all Windows Web Servers will also be marked as Windows Application Servers (described in the section above) after installation even if they were designated as Web Servers only in the Apprenda Installer. By default, however, the Platform will deploy WCF/Windows services to Web Servers only if there are no dedicated Application Servers available.
**MS SQL Nodes**
The Platform manages SQL Server instances on your behalf to provision and configure guest application databases. Any number of SQL Server instances can be managed by a single Platform, and SQL instances can be added to the Platform at any time for capacity. Our reference architecture includes a single SQL Server instance that is configured as a SQL Server failover cluster; such a configuration typically relies on shared storage on a SAN or NAS. We’ve included two SQL nodes in the cluster as a standard recommendation for simple redundancy. The Platform will manage this cluster as it would a normal SQL Server instance. Expansion of the database tier of our reference architecture would come in two forms:

- Adding SQL Server nodes to existing cluster(s) will increase redundancy capabilities of existing Platform-managed SQL instance.
- Adding SQL Server nodes independent of existing cluster(s) or adding entirely new Platform-managed clusters will increase the capacity of the database hosting layer of your Apprenda Platform.

In addition to providing storage for guest applications, an Apprenda managed SQL Server instance is necessary for housing the Apprenda Core Database, which contains data necessary for Platform functionality.

**Platform Repository**
The Platform Repository is a network share that will serve as the central storage location for all Platform and guest application binaries. This location must be configured prior to installation, should be specified as a network path in the Apprenda Installer, and must contain the following three folders:

- **Applications**
- **Apprenda**
- **SAC**

It is also acceptable to configure each folder as a separate share so long as they are accessible through the same base path.

All guest application binaries, once uploaded to the Platform by developers, are stored in the Applications folder in the Platform Repository (in some parts of the Platform, such as the Repository Browser in the SOC, this folder is called the “Application Repository”). The Applications folder will also include binaries for Platform components that are themselves hosted on the Platform as guest applications, such as the Developer Portal and the System Operation Center. All other Platform binaries are stored in the “Apprenda” folder (which is sometimes referred to as the “System Repository”). Upon workload deployment, binaries that are needed for local execution are copied to the target server from their respective locations in the Platform Repository.
Linux Servers (Optional)
Linux Servers host Java Web Application components, which are deployed and managed on top of individual Tomcat instances by the Apprenda Linux Container.

Oracle RDBMS (Optional)
The Platform manages Oracle RDBMS installations on your behalf to provision and configure storage for guest applications.

AD FS Nodes (Optional)
As an install-time option, the Platform can be configured to support identity federation using Active Directory Federation Services (AD FS). If you choose to use AD FS for identity federation, we recommend creating an AD FS Web Farm consisting of no less than 2 AD FS Nodes backed by a SQL Server failover cluster. Please contact support@apprenda.com for information on setting up an AD FS Web Farm prior to running the Apprenda Installer.

It should be noted that all AD FS nodes—including those that constitute an AD FS Web Farm—will automatically be designated as Windows Application Servers. This is necessary for the nodes to host the Federation Service (see the Apprenda Software Inventory below), which allows the Platform to interface with AD FS.
# Apprenda Software Inventory

<table>
<thead>
<tr>
<th>Package</th>
<th>Description</th>
<th>Type</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apprenda Platform Coordinator</td>
<td>Provides global lookup and name-server-like functionality to all other member nodes in Apprenda. It is responsible for maintaining awareness of all nodes, tracking relationships between nodes and what they each host, as well as maintaining responsibility for certain types of fabric-level management requests.</td>
<td>Windows Service</td>
<td>Platform Coordination Nodes</td>
</tr>
<tr>
<td>Apprenda Caching</td>
<td>A distributed Redis-based in-memory caching implementation for Windows and Linux.</td>
<td>Windows Service</td>
<td>Cache Nodes</td>
</tr>
<tr>
<td>Apprenda Load Manager</td>
<td>Manages IIS and Application Request Routing to distribute inbound HTTP request to the appropriate servers.</td>
<td>Windows Service</td>
<td>Load Managers</td>
</tr>
<tr>
<td>Apprenda Host Controller</td>
<td>The agent on each participating server that performs resource throttling of guest applications. Also handles Platform logging and licensing for Windows servers.</td>
<td>Windows Service</td>
<td>All Windows Web and Application Servers</td>
</tr>
<tr>
<td>Apprenda Service Container</td>
<td>The Platform's own WCF/Windows service hosting container; also houses Platform routing functionality.</td>
<td>Windows Service</td>
<td>All Windows Web and Application Servers</td>
</tr>
<tr>
<td>Apprenda Linux Container</td>
<td>Provides hosting and management services for Java Web Applications workloads packaged as WAR files, and manages licensing for Linux servers.</td>
<td>Linux Daemon</td>
<td>All Linux Servers</td>
</tr>
<tr>
<td>Account Portal</td>
<td>The user interface and web services that facilitate management of end user accounts.</td>
<td>UI, Apprenda Service</td>
<td>UI: Windows Web Servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Apprenda Service: Windows Application Servers</td>
</tr>
<tr>
<td>Package</td>
<td>Description</td>
<td>Type</td>
<td>Location</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>Apprenda Guest Application API Provider</td>
<td>Hosts an internal API used by the Linux Container to configure Java Web Application components for deployment and retrieve information for the Java Guest Application API.</td>
<td>UI</td>
<td>Windows Web Servers</td>
</tr>
<tr>
<td>Authentication Services</td>
<td>The user interface and web services that provide authentication mechanisms such as login/logout and signup capabilities.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers</td>
</tr>
<tr>
<td>Cataloging Service</td>
<td>Maintains an index of application components, as well as deployed workload metadata.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers</td>
</tr>
<tr>
<td>Common UI Resources</td>
<td>A collection of UI resources such as shared CSS, Javascript, and images used by the Apprenda portals.</td>
<td>UI</td>
<td>Windows Web Servers</td>
</tr>
<tr>
<td>Developer Portal</td>
<td>The user interface and web services that facilitate developer creation, publishing, monitoring, and management of guest applications.</td>
<td>UI, Apprenda Service</td>
<td>UI: Windows Web Servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Apprenda Service: Windows Application Servers</td>
</tr>
<tr>
<td>Extensibility Services</td>
<td>Runs all Add-ons and Bootstrap Policies in a safely controlled environment to prevent any unexpected side effects.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers</td>
</tr>
<tr>
<td>Federation</td>
<td>Interfaces with Active Directory Federation Services (AD FS) to establish a trust relationship and assist with authentication.</td>
<td>Apprenda Service</td>
<td>AD FS Nodes</td>
</tr>
<tr>
<td>Maintenance Interface</td>
<td>A user interface that is displayed in place of application that is being upgraded by the Platform.</td>
<td>UI</td>
<td>Windows Web Servers</td>
</tr>
<tr>
<td>Package</td>
<td>Description</td>
<td>Type</td>
<td>Location</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>Monitoring Service</td>
<td>Aggregates workload resource utilization data for persistence. Also manages dynamic and scheduled workload scaling, as well as min/max instance count enforcement.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers</td>
</tr>
<tr>
<td>Presentation Controlling Services</td>
<td>Provides IIS administrative management capabilities.</td>
<td>Apprenda Service</td>
<td>All Windows Web Servers</td>
</tr>
<tr>
<td>Scheduling Service</td>
<td>Provides scheduling services for time-based and delayed execution of web service calls.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers</td>
</tr>
<tr>
<td>Storage Controlling Services</td>
<td>Interfaces with SQL Server and Oracle to create, delete, and otherwise configure storage for your application's databases.</td>
<td>Apprenda Service</td>
<td>Windows Application Servers designated as Storage Controlling Services hosts</td>
</tr>
<tr>
<td>System Operation Center</td>
<td>The user interface that facilitates Platform management.</td>
<td>UI</td>
<td>Windows Web Servers</td>
</tr>
</tbody>
</table>
Hybrid Cloud Installations

Apprenda’s hybrid cloud capabilities allow for the incorporation of infrastructure stationed at different physical locations—whether hosted on public IaaS or at separate private data centers—into a single Apprenda Platform that minimizes latency between locations.

In a hybrid cloud setup, the Apprenda components that comprise the Platform are categorized by cloud (preferably with one cloud per geographic location). Each cloud is capable of independently hosting all levels of a deployed application (effectively creating redundancy at each tier) and will include its own Load Management, Platform Coordination, and Caching configurations. All clouds share a single Core Database and Platform Repository. Support for optional guest application components, such as Oracle-based data storage and Java Web Application workloads, can be added to one or more clouds as dictated by location/load needs. If identity federation is configured for the Platform, a single AD FS web farm will be shared among all clouds.

Each cloud has its own unique URL, which enables routing to the appropriate application and/or service. However, guest application routing spans hybrid clouds, meaning if a user requests a guest application, the platform will ensure they reach the proper URL for that application based on its location. A hybrid cloud setup minimizes latency considering an application only needs to access services and data storage within its own datacenter. Applications are also “Cloud Aware,” meaning they maintain knowledge of their resource access pool and automatically deploy locally with these resources in order to maintain high reliability and availability.

Networking Requirements: Establishing a VPN tunnel (for Multiple-Cloud installs)

All Apprenda infrastructure located on a Platform must be accessible from a single LAN. In order to achieve this for a hybrid cloud setup, we recommend point-to-point VPN tunneling between datacenters in order to provide a secure medium for when inter-datacenter communications need to occur.
Reference Topology (Hybrid Cloud)

The following is the topological construction of a hybrid cloud Appenda Platform reference architecture. Please note that while this illustrates a two-cloud configuration, it is possible to set up three or more clouds on a single Platform. While the Platform roles discussed above for a single cloud topology still apply, the Platform as whole shares a single Platform Repository, Core DB, and AD FS farm. Although these items have been located on the same cloud, it is possible to locate them on different clouds.
Maintenance, Upgrades, and Infrastructure Modifications
The maintenance plan for your Apprenda Platform will be based largely on the designed architecture for your particular environment. It is also important to distinguish between infrastructure maintenance and Platform maintenance.

**Infrastructure Maintenance**
Much of the Apprenda Platform architecture is designed to be accommodating to ordinary and regular infrastructure tasks on the parts of the stack below it. For example, the Platform allows IT to shuffle frontend and web service application workloads across servers dynamically, which makes it easy to remove workloads from a particular server for maintenance. To apply Windows Update, for instance, IT can move all web service workloads off of an Application Server, perform updates (reboot if necessary) and allow the Platform to redeploy workloads when the maintenance period is complete. Likewise for UI workloads and Web Servers.

**Platform Maintenance**
The Apprenda Platform comes with a set of tools and utilities that facilitate lower-level Platform maintenance such as configuration changes not available in the management portals. Training for these tools is available and advised. Contact support@apprenda.com for more information.

Platform upgrades are available through the Apprenda Installer for the Platform version to which you would like to upgrade. Contact support@apprenda.com for information if you are planning to upgrade your Apprenda Platform.

**Adding or Removing Infrastructure**
With the exception of singular core Platform roles that can be designated at installation only (such as the Platform Repository, Core DB, and AD FS Web Farm), the “Modify” workflow in the Appenda Installer can be used to add or remove infrastructure as needed. Nodes may be added to existing clouds, or removed as long as the cloud is left with sufficient resources to a host component types essential to Platform functionality (.NET UIs, WCF services, and SQL Server databases). As needed, entire clouds may be added to the Platform, or removed provided that at least a single cloud (and infrastructure to support essential Platform functionality) remains. Please note that in order to add infrastructure, you must use the Apprenda Installer that corresponds to the Platform version (e.g., “5.0.3”) that you currently have installed.

The entire Platform can be uninstalled using the “Uninstall” workflow in the Appenda Installer.