PIQUE SOLUTIONS

Choosing the Right Vendor for Application Development

Customer Experiences with Amazon, Oracle, and Pivotal Cloud Services

PIQUE SOLUTIONS

August 2018

THE DEVELOPMENT OF THIS WHITE PAPER WAS SPONSORED BY ORACLE. THE UNDERLYING RESEARCH AND ANALYSIS WERE EXECUTED INDEPENDENTLY BY PIQUE SOLUTIONS.

795 Folsom Street, 1st Floor | San Francisco, CA 94107 | Tel.: 415.685.3392 | www.piquesolutions.com

Contents

Pique Solutions is a competitive research and market analysis firm supporting Fortune-500 companies in the information technology sector. Pique is based in San Francisco, California.

Executive Summary

Many enterprise customers are increasingly turning to the cloud for application development. In addition to cloud-native AppDev scenarios, these customers are working through a host of other types of AppDev scenarios. During the development of this white paper, Pique Solutions spoke to numerous enterprise customers to understand how they approach application development in the cloud. In general, their projects fall into the following six scenarios:

- Cloud-native
- Container-native
- Modernization of enterprise Java
- Software-as-a-service (SaaS) extensions
- High productivity (low-code/no-code)
- Mobile

Our in-depth discussions with these customers aimed to assess how three leading cloud AppDev providers meet the needs of these varied scenarios. These vendors include Amazon, Oracle, and Pivotal.

The key findings of the study were as follows:

- Surveyed customers shared that they have more than one AppDev use case, citing the need to support a wide spectrum of AppDev scenarios and the value of procuring those services from a single cloud provider. Customers shared that Oracle is one cloud provider that can meet their various needs and found significant integration with other cloud applications, platforms, and infrastructure elements.
- All three vendors have achieved significant adoption in cloud-native deployments. Many Amazon Web Services (AWS) infrastructure-as-a-service (IaaS) customers naturally use AWS to develop cloud-native apps. Pivotal is perceived as a go-to cloud-native application platform because of its support for Cloud Foundry (PCF). Many customers, however, chose to use Oracle Cloud for cloud-native application development for its integrated developer experience, support for diverse technologies and open standards, and interoperability with on-premises systems.
- Container-native development is a nascent but rapidly developing service area that constitutes a compelling approach for many companies. Compared to AWS and Pivotal, Oracle is perceived by study participants as a more recent entrant in this space, but it has advantages with respect to container-native development by adhering to open standards, providing transparency and control over services, and ensuring the integrated nature of their cloud services.
- Study participants cited Oracle's common tooling and central capabilities—such as DevOps, management, and integration—that work effectively across their portfolio of cloud services and hybrid operations, simplifying implementation, deployment, and management. AWS and PCF also provide these capabilities, but they are more limited from the end-to-end life-cycle perspective and customers often need to use a variety of other third-party services and tools for DevOps and management of their environments.

- Oracle provides comprehensive support for modernization of enterprise Java, SaaS extensions, and high-productivity application development, whereas these are noted gaps in AWS's and, to a greater degree, Pivotal's current offerings. Oracle also effectively supports the rapidly emerging blockchain opportunity with its new Autonomous Blockchain Cloud Service, which offers integration with existing cloud and on-premises applications, and AppDev environments and tools.

Table 1. Analysis of Use Case Coverage by Cloud Providers

App Dev Use Case	AWS	Oracle	Pivotal
Cloud Native	●	●	●
Container Native	●	●	◔
Modernization of Enterprise Java	○	●	◔
SaaS Extensions	○	●	○
High-Productivity	○	●	○
Mobile	●	●	○

As illustrated in **Table 1**, Oracle provides a more comprehensive support for cloud AppDev use cases as compared to AWS and Pivotal. Although not all use cases are likely to be employed in a single company, many companies we spoke with did cite the value of having the flexibility and uniformity of supporting various development models as they journey to the cloud. In addition, AWS and Pivotal customers related that they can benefit in leveraging Oracle Cloud services in a multivendor, multicloud environment to address the areas not currently serviced by augmenting their enterprise service capabilities. Lastly, surveyed customers also like having the option of running in the public cloud, on-premises, or in their own datacenters—and moving back and forth between them as needed. This is a noted gap in AWS's service strategy, and to a lesser degree, Pivotal's.

Introduction

Cloud services are changing and enhancing the way enterprises develop, test, deploy, and manage applications. The result is a far greater choice of options for developing and delivering applications, or even transformative capabilities, to the market. Some newer organizations are "born in the cloud" and from inception place their entire IT estate in the cloud. Most enterprises, however, are somewhere along the journey of moving to the cloud and likely operate in several different development modes. A recent *CIO* article cites that public cloud adoption in enterprises will cross 50% this year, suggesting a tipping point fueled by digital transformation initiatives.[1]

Pique Solutions has identified six key scenarios for cloud application development and options for which enterprises have to leverage cloud services.

- **Cloud-native development** is an approach to building and running applications that fully exploits the advantages of the cloud computing model. It involves 12-Factor applications, polyglot programming (e.g., node.js, ruby, Go, Java SE), and microservices.
- **Container-native application development** is a more recent focus for many development organizations based largely on the success and adoption of Kubernetes. Container-native means that daily development practices include the standard building blocks of containerized applications and the orchestration of those containers in a true DevOps methodology.
- **Modernization of enterprise Java applications** is a scenario in which enterprises that have a significant investment in Java applications on-premises seek to leverage cloud services to migrate those applications to the cloud without having to refactor or lose any enterprise-grade benefits that Java brings relative to performance, security, scalability, and availability.
- **SaaS extensions** refers to using cloud services, primarily platform as a service (PaaS), to extend, customize, and integrate SaaS applications, which by their very nature and architecture are the same for all SaaS customers. Customers can take advantage of extensions built by others in a marketplace scenario to build their own customizations.
- **High-productivity application development** refers to capabilities designed for rapid prototyping and development with more visual, less-coding use cases for either "citizen developers" or even application IT resources. It extends the developer base beyond the "pro coders" to enable agile customization and extension of applications.
- **Mobile application development** can be enhanced and augmented using cloud services for the development and deployment of mobile and chatbot applications and even mobilizing traditional on-premises applications in a sustainable model. Companies struggle with mobile development when they approach it as a standalone process disconnected from enterprise development.

[1] https://www.cio.com/article/3137946/cloud-computing/6-trends-that-will-shape-cloud-computing-in-2017.html.

Cloud Providers Take Different Approaches

Given the variety of AppDev scenarios, Pique Solutions evaluated three common cloud providers relative to enterprise deployments. Through secondary research and interviews with customers, we sought to document the approach, the service coverage across use cases, and direct feedback based on customer experiences with AWS, Oracle Cloud, and PCF.

Amazon Web Services

AWS has demonstrated market leadership in public cloud infrastructure and in recent years has augmented its IaaS with additional PaaS services. The company has traditionally targeted developers and IT professionals and provides good tooling for developing cloud-native applications residing on the AWS cloud. AWS also provides a comprehensive marketplace of surrounding PaaS services, but it is largely up to customers to assemble the broader solutions, which may involve a mix of AWS and third-party vendor offerings. AWS's own PaaS-related services are more limited, and they do not currently offer any enterprise SaaS (e.g., Enterprise Resource Planning, Human Capital Management, Customer Relationship Management) nor ready SaaS extensions and low-code tooling for citizen developers or line-of-business users. AWS does provide technical tooling and services for DevOps, management, and monitoring, but they are best suited to cover AWS service elements.

Oracle

Oracle, with its roots in enterprise software and systems, supports a broad spectrum of AppDev scenarios from cloud- and container-native, to blockchain, to modernizing enterprise Java applications, to extending SaaS applications, and even to high-productivity and mobile cloud services. Oracle was considered a more recent entrant to the cloud IaaS arena, although they started their journey to cloud in 2006 when they embarked on building their portfolio of SaaS applications. The company has been aggressive in rolling out IaaS and PaaS services in the last few years. Customers shared that these cloud services were well integrated and enterprise-ready. They provide AppDev tools for both technical and citizen developers. Finally, Oracle's DevOps, management, and monitoring offerings cover the spectrum of cloud, on-premises, and even non-Oracle environments.

Pivotal

PCF is an application PaaS built on top of Cloud Foundry, an open-source cloud platform that allows developers to deploy, operate, and scale cloud-native applications. Compared to other vendors and cloud providers, Pivotal is focused narrowly in the application PaaS (aPaaS) layer and lacks ancillary PaaS capabilities such as integration, process automation, data management, analytics, and APIs. That said, they do also provide service brokers that allow customers to integrate with other cloud providers' PaaS services. Pivotal's DevOps and management capabilities are generally more limited and they do not, as a vendor themselves, address scenarios for modernizing enterprise Java, SaaS extensions, high-productivity, and mobile/chatbot application development. As such, Pivotal customers often need to work with multiple vendors to cover the full SaaS, PaaS, and IaaS layers.

Study Approach

Methodology

The primary research phase consisted of an in-depth data collection and multiphase interview process. Pique identified and qualified 14 customers and partners involved in implementations inside medium and Fortune-500 organizations. These experts provided detailed primary research and data. The research focused on the cloud application development options and the impact of choice of vendor platform on the IT environment and the broader business.

The research process and methods were as follows:

- Reviewed publicly available information and secondary research on cloud application development trends, drivers of adoption, use cases, and key value drivers.
- Identified and qualified 14 customer interviewees who participated in in-depth interviews for each of the different cloud solutions.
- Synthesized data and research findings.

Table 2 lists the companies analyzed and interviewed in the data-gathering phase of the research project.

Table 2. Companies and Roles of Participants Included in Primary Research

Company	Title
Cloud Consulting Firm	Database Administrator and Systems Engineer
Global Retailer	Director of Software Engineering and Architecture
AI/Cognitive Computing Platform Provider	Vice President of Marketing and Vice President of Engineering
Global Financial Services Company	Executive Director of Technology
Insurance Provider	IT Director
Regional Bank	Chief Information Security Officer
Enterprise Java Mobile Platform Provider	Chief Technology Officer of Cloud Products
Global Systems Integrator	Cloud Practice Manager
Transportation and Warehousing Company	Director of Enterprise Architecture
Recreational Sports Equipment Manufacturer	Director of Global ERP
Integration Cloud Services Provider	Chief Executive Officer
Large South American University	Chief Information Officer, Provost
AWS and Pivotal Systems Integrator	Cloud Developer
Large Middle Eastern Airport	Head of Manpower Planning

AppDev Use Cases and Customer Experiences

The customers Pique Solutions interviewed spanned the six use cases, with many customers having experience with more than one in their organization. In this section, we share the customer experiences along with some independent analysis of the vendors' support for each of the use cases.

Cloud-Native Application Development

Cloud-native development is an approach to building and running applications that fully exploits the advantages of the cloud-computing model. Cloud-native development involves services-based architecture and microservices, or any modular loosely coupled model, for independent scalability, flexibility, and maintenance of 12-Factor applications using polyglot programming languages such as node.js, Python, PHP, Ruby, Go, and Java SE.

A global systems integration company shared their experience of leveraging cloud application development to create microservices that support multiple languages to better support their end customers. The cloud practice manager of the global systems integrator shared the analogy of a bento box approach to provide a full library of services so customers do not have to install, configure, or build themselves. Specifically, he shared, "What we're trying to build is a chat bot bento box, where we have all the libraries, all the languages, all deployed on a container. It just so happens to be on the Oracle's app container, which we just give to clients so they can run their own chat bots without actually going through the cycle of installing, updating libraries, basically providing this accelerated platform for them to connect their chat bot into their back office day one as opposed to starting from scratch and trying to build all these different pieces together."

> "I don't know if there's any other cloud out there that allows you or gives you the flexibility to deploy various technologies onto one cloud as Oracle does."
>
> **Cloud Practice Manager**
>
> Global Systems Integrator

He also highlighted the corresponding need and ability to run multiple technologies on one cloud, saying "I don't know if there's any other cloud out there that allows you or gives you the flexibility to deploy various technologies onto one cloud as Oracle does."

Finally, he touted the benefits of his experience using Oracle Developer Cloud Service (Dev CS) for DevOps, citing the end-to-end CI/CD functionality of Oracle Dev CS. Features that they leveraged included the task tracking system, agile project management, SCRUM and Kanban dashboards, and team collaboration tools that are built into Oracle Dev CS.

Cloud-native application development is considered the "table stakes" in terms of the use cases, so it is not surprising that all three vendors' offerings were found by customers to be effective and productive in terms of deployments in this area. For AWS and Pivotal customers, this is the most common use case and has been the focus of both companies. That said, customers or solution providers deliver cloud-native applications in a variety of ways.

Customers related that all three vendors provide good support for cloud-native application development by supporting polyglot programming and 12-Factor methodologies. Oracle offers an extensive and integrated set of tools for customers to easily build and deploy cloud-native applications. AWS and Pivotal, however, have historically been lacking in a complete and comprehensive DevOps capability but, based on our research, are augmenting their services to better address these gaps.

> "AWS and Pivotal have some gaps within IDE integration for containers and end-to-end code in the CI/CD pipeline in DevOps environments."
>
> **Cloud Developer**
>
> AWS and Pivotal Systems Integrator

A cloud developer experienced with both AWS and PCF shared that "AWS and Pivotal have some gaps within IDE integration for containers and end-to-end code in the CI/CD pipeline in DevOps environments. However, with recent announcements from AWS regarding Cloud9 and CodeStar it looks like AWS is going to close the gaps in its CI/CD pipeline components and make it a more complete DevOps solution," even though its tooling is still lacking integrated issue tracking system, Sprint management and dashboard (SCRUM and Kanban), peer code review, merge requests, and Wiki capabilities.

Container-Native Application Development

Container-native application development is a more recent focus for many development organizations based largely on the success and adoption of Kubernetes. Container-native means that daily development practices include the standard building blocks of containerized applications, utilizing Docker/Kubernetes, serverless Functions, and microservices. Some key characteristics of container-native development include the following:

- Builds are done using Docker/Kubernetes, including compiling and even dependency management.
- Tests are run in containers.
- The artifact produced is a container image, not a binary, tarball, or bundle.
- The testing and debugging cycle is done against containers, not applications running locally.
- The target deployment environment (Docker/Kubernetes) is the same as the development environment.
- Deployments often include serverless functions and microservices.

An AI/cognitive computing platform provider of services for facial recognition, video/audio processing, NLP/transcription, and eDiscovery shared their needs relative to building container-native applications. The vice president of engineering shared, "We have a software platform that acquires end content, whether it's structured or unstructured, such as audio and video. Then we run it through a variety of what we call engines that provide cloud services. That allows us to do transcription, facial recognition, and text processing, et cetera. Then it's all built on top

of an existing platform that allows us to start coordinating and orchestrating all those AI services to keep on refining that information so that we can actually get it to something we can then run through a business process."

The firm leveraged Oracle Container Cloud services as well as serverless capability with a team of 40-plus engineers and developers primarily developing in Node.js and Go languages and writing scripts for infrastructure using Terraform. "We have over 400-plus projects on Oracle Pipelines. We've run all of our processing of content, whether structured or unstructured data, through the serverless architecture that's based on containers. The whole container-native approach allows us to onboard all these AI engines that use a variety of different technologies, such as TensorFlow, Keras, GPUs, et cetera. Being able to run it through the orchestration system we built in conjunction with Oracle Functions allowed us to get to an immense scale. We couldn't do it if we didn't have a container-native approach."

"Being able to run [our content] through the orchestration system we built in conjunction with Oracle Functions allowed us to get to an immense scale. We couldn't do it if we didn't have a container-native approach."

Vice President of Engineering

AI/Cognitive Computing Platform Provider

Like cloud-native application development, all three vendors in our study support container-native application development but to varying degrees. They use different service and technical approaches depending on how they introduced container services in the context of their existing cloud application development services.

For example, support for Kubernetes was only recently released by Amazon (Elastic Container Service for Kubernetes) and Pivotal (Pivotal Container Service). By contrast, Oracle has engaged and invested substantially in the Cloud Native Computing Foundation (CNCF) efforts and, in addition to the acquisition of Wercker for CI/CD, has developed and released an open container engine for Kubernetes, container pipelines, serverless functions, container microservices, and a container registry.

According to surveyed customers, the key difference in Oracle's approach is the technical transparency and cloud-neutrality it provides. Customers also like the fact that Oracle's Kubernetes is not only based on standards, but Oracle's serverless functions (Fn) and microservices are also open source and standards-based. Managing Kubernetes clusters is difficult but Oracle provides a fully managed Kubernetes service to make it easier to deploy and manage. Oracle's Kubernetes is also available to run on bare metal. In contrast, AWS Lambda is considered proprietary and AWS Kubernetes is not currently available on bare metal. Finally, AWS and Oracle provide fully managed Kubernetes, while Pivotal leaves it to the customer to manage.

Table 3. Container-Native Service Characteristics

AWS	Oracle Cloud Services	PCF
Fully managed Kubernetes Several approaches to container-based development including Elastic Container Service (ECS) and Elastic Container Service for Kubernetes (EKS) Services such as Lambda and EKS lack transparency and level of control A "roll-your-own" approach using various AWS services and third-party tools; customer must integrate Kubernetes service not yet available on their nascent bare metal service Lambda and Function Steps are proprietary to only AWS and cannot deploy on other cloud infrastructure	Fully managed Kubernetes Transparent, standards-based approach using Docker/ Kubernetes Integrated, very lean experience, with prebuilt dashboards, monitoring, and DevOps pipeline Build and run natively on Oracle bare metal instances for performance, scalability, and security High Availability (HA) and runs three masters across three Availability Domains Oracle Fn and microservices based on open source and standards-based Flexible and open to run on other cloud infrastructure	Customer-managed Kubernetes Container support geared toward deployment and not as integrated in the developer experience Architecture of product is proprietary and hard to fully support new community architecture Relies on third-party monitoring, logging, and management Multicloud support

Table 3 summarizes the experiences and sentiment of our study participants as they relate to AWS, Oracle, and PCF. In short, companies we talked to found AWS container-native service offerings lacking in integration consistency across the AWS service portfolio and lacking in transparency and granular control of their services. In fact, many cited the inability to determine what was "going on under the hood" from both configuration and performance perspectives.

The AI/cognitive platform provider discussed earlier also shared their evaluation of AWS/Lambda and found it not suitable for their needs. "Originally, we looked at AWS Lambda, which is an obvious thing you look at, but it really didn't allow us to support a lot of AI services. For example, if you need GPUs, if you need special libraries involved, I found it couldn't support it at the time. I really just had a Node.js and Python .zip file I could upload. I didn't really have a full control over what was running, which we need to do to support all these different environments like TensorFlow, or Keras, or any of the others."

Further, as compared to Oracle services, AWS and PCF were found lacking on DevOps capabilities, which led many companies to assemble solutions themselves using third-party tools, including Chef, Puppet, GitHub, Octopus, Ansible, and others. Pivotal's support for Kubernetes is very new and, in the context of their broader application development platform, participants found it challenging to achieve truly open standards. They also found it more suited toward deployment and not fully integrated with the full lifecycle developer experience.

A systems integrator familiar with container development using AWS and Pivotal tools shared his experience in terms of the container-native development process. He explained, "Both AWS and PCF offer building images with the application, putting the image in the registry, deploying the image to container environments in a CI/CD pipeline. The difference is how easy it is to set up and use the pipeline. Overall, AWS has more in the way of process to set up and use the CI/CD pipeline. Though PCF provides support for containers in two ways—Pre-Built Images and Platform Built Containers via Buildpacks—PCF really provides container-native *deployment* support, and *not* direct development support, since all development including coding, testing, compiling, and application building are done in other ways."

> "[AWS] Lambda is not open source and we have found some performance issues in using it."
>
> **Chief Technology Officer**
>
> Mobile Application and Integration Platform Provider

With respect to serverless capabilities, study participants also cited differences in the vendors. A chief technology officer of a mobile app and integration platform provider talked about the use of serverless computing: "Serverless can be a very compelling feature for customers to adopt the Oracle Cloud. Oracle's Fn is open source. We believe it will run best on the Oracle Cloud. To compare it to our evaluation of AWS, Lambda is not open source and we have found some performance issues in using it."

Lastly, customers cited the benefit of Oracle providing container-native service capability as one of many modes of development needed for an effective migration to the cloud over time.

> "Oracle is way ahead of competitors with respect to container-native development and managed Kubernetes. We have been impressed with the level of support we have gotten in terms of knowledge of their resources, and they are committed to our success."
>
> **Director of Software Engineering and Architecture**
>
> Global Retailer

For example, a large global retail organization shared their accelerated path to move their datacenter applications to the cloud over the next three to five years, and they will pursue many new development projects using a container-native approach. Their director of engineering explained, "Any new development that we are doing we use microservices architecture. We also develop 12-Factor applications that are cloud ready."

They value the flexibility of Oracle Cloud to support applications from packaged to modern Java-based applications. They are considering a move to SaaS longer term but are focused on lift-and-shift as a first step. "If it is a lift-and-shift application, it is easy for us to spin up a bunch of servers in Oracle and use them as our IaaS provider. But if it is a modern

application and we plan to migrate that to Oracle, our strategy is to get to a more microservices, container-based architecture and that is where we look to more of their PaaS services for AppDev."

He went on to say, "Oracle is way ahead of competitors with respect to container-native development and managed Kubernetes. We have been impressed with the level of support we have gotten in terms of knowledge of their resources, and they are committed to our success."

Finally, he shared the positive impact on developer productivity: "My developers are excited about container-native development and the Oracle cloud services. They feel like the tools are very developer friendly and easy to use. Our developers are surely more productive."

Modernization of Enterprise Java Applications

Given the longevity and success of Java, particularly in the context of enterprise applications, many companies have a significant investment and installed base of enterprise Java applications in their IT estates. With an estimated 8 to 10 million Java developers, there is also a significant labor base of professionals skilled in Java concepts, standards, and technologies. Rather than start from scratch in the cloud or move or migrate to a completely different architecture, many companies remain interested in leveraging their Java investments in conjunction with cloud deployment.

Modernizing enterprise Java applications to support the development and integration of mobile users and data is a significant opportunity for many companies. A mobile and integration platform provider shared an interesting scenario of leveraging enterprise Java for cloud development and making enterprise Java capabilities available in the cloud. According to their chief technology officer, many companies' mobile capabilities are in a "state of panic" due to traditional approaches tied to mobile devices (e.g., Swift, JavaScript) that have a short shelf-life and are very challenging to sustain. Mobile developers in mobile centric approaches are often disconnected from enterprise developers and want to "open all ports" for access, which is extremely risky. Healthcare, military, and finance organizations are evaluating alternatives because of the importance of enterprise security.

The CTO shared the key advantage for Oracle: "The technology is well ahead of the competition. On Oracle, you can run on Java 9, which is a huge advantage for developers. It is much more productive for developers and makes it much easier for us as a Java development company." Java also enables developers to have more control over applications. He went on to say, "Running on Oracle Cloud allows Java developers to leverage their skills to mobile, and the fact that it is integrated well to Oracle Cloud components is important."

"Running on Oracle Cloud allows Java developers to leverage their skills to mobile, and the fact that it is integrated well to Oracle Cloud components is important."

Chief Technology Officer

Enterprise Java Mobile Platform Provider

Table 4. Support for Modernizing Enterprise Java

AWS	Oracle Cloud Services	PCF
No native support for **e**nterprise Java **a**pps but can run enterprise Java apps in EC2 containers in customer self-managed environment Lacking integration of cloud-native and on-premises enterprise apps	Leverages existing enterprise Java skillsets Native support for enterprise Java apps Support for latest Java version Support for seamless migration of on-premises enterprise Java apps to cloud or in hybrid-cloud deployment Tools to profile customers' Java apps with Java Flight Recorder and gain insight into their Java app behavior and performance with Java Mission Control	More rooted in Spring framework; not ideal for lift-and-shift of enterprise Java apps Development does not align with traditional enterprise Java development

> "If you want to build an enterprise Java application with PCF, it's all on your own. There is no IDE integration and you need to develop your own buildpack, and dependency management is very hard."
>
> **Cloud Developer**
>
> AWS and Pivotal Systems Integrator

As illustrated in **Table 4**, support for moving enterprise Java applications to the cloud is limited for both AWS and Pivotal customers. AWS customers can run Java apps in EC2 containers in a customer self-managed environment, but they lack support for native enterprise Java capabilities in the cloud. This means that developers are only able to take advantage of the visibility and level of control for security, performance, and scalability inherent in enterprise Java if they take the time to set up the environment, build their own High Availability/Disaster Recovery (HA/DR), and tune and manage the environments themselves. This presents a significant challenge from an operations perspective.

Pivotal is rooted in the Spring framework but is lacking a developer experience that integrates the IDE and build process to the deployment and management for enterprise Java. A cloud developer shared his experience with Java and Pivotal and cited the issue relative to native enterprise Java support. He explained the process: "If you want to build an enterprise Java application with PCF, it's all on your own. There is no IDE integration and you need to develop your own buildpack, and dependency management is very hard. In theory, you can do development activities with your own IDE with WebSphere, WebLogic, JBoss, and Glassfish app servers supported. Then you can compile and build to .jar, .war, or .ear files, then take the

prebuilt images approach with those enterprise Java applications and deploy them to PCF containers. In my opinion, I wouldn't recommend this approach. Instead I would suggest deploying prebuilt images directly with IaaS providers or on your private cloud. There is no need for the container on top of PCF on top of IaaS."

Things could be changing, however, as noted in a recent announcement highlighting a Pivotal partnership with IBM on enterprise Java.[2] If successful, this could make the development and deployment on PCF easier, but it still may not adequately offer some of the business-critical enterprise Java features such as enterprise grade transaction, security, concurrency, and management.

The cloud developer summed it up saying, "Most importantly, from my experiences, there are lots of JEE or even earlier version J2EE applications in the banking, retail, and insurance industries running business-critical workloads. Increasingly, executives want to move those applications to private or public clouds. But there is not any intention and capabilities from PCF to support those. After all, PCF is rooted in Spring Framework, which is a competitor with JEE framework." Despite sharing some of the specific challenges relative to enterprise Java, the system integrator did acknowledge that PCF is good for new cloud-native applications with short life cycles and frequent releases.

Finally, another aspect of modernization of Java involves moving application services to the cloud and making it easier for end users to access and interact with enterprise capabilities. A good example of this was shared by a large university in South America. The university serves 20,000 traditional students and has opened nearly 100 smaller facilities to support long-distance learning across the region. Its "virtual university" enrolls an additional 10,000 students and plans to increase its enrollment by as much as 60%, from 30,000 to 50,000 students in the coming years.

A significant part of that virtualization and outreach involved the implementation of the Oracle Java Cloud Service and Oracle Database Cloud Service to extend their ERP application capabilities to new groups of users. Based on the same set of technologies they were using on-premises, it proved very cost-effective and easy to deploy and manage. The chief information officer cited, "By relying on a single set of deeply integrated [Oracle] technologies, we are achieving all the scalability and reliability we need—even with a limited IT

"By relying on a single set of deeply integrated [Oracle] technologies, we are achieving all the scalability and reliability we need—even with a limited IT team."

CIO

Large South American University

[2] https://adtmag.com/articles/2017/12/05/pivotal-ibm-java.aspx.

team." But for the provost, the cloud services are having an even more profound impact, powering what he calls innovation centers.

Extending SaaS applications

Another key use case as companies move to the cloud is the need to customize or extend SaaS applications that are, by nature, lacking in the ability for individual customers to modify them to fit their unique needs. PaaS-based integrations and extension philosophies ensure that base applications remain separate from extensions, thereby reducing complexities of applications upgrades and patching-related dependencies. PaaS platform components can not only connect with SaaS applications but also with on-premises applications, which essentially means that one can orchestrate an extension by combining different capabilities all connected in a standardized service-oriented manner. This affords companies with greater flexibility and leverage of SaaS applications in the context of their IT portfolio.

Some examples we identified of Oracle customers extending SaaS applications include the following:

- JavaScript Extension Toolkit (JET) based custom application to extend the functionality of Oracle Procurement Cloud for a custom Supplier Portal or Travel Management module for HCM Cloud.
- Managed attachments and document printing solutions for Sales Cloud using Document Cloud and Java Cloud Service.
- Custom employee hiring and onboarding process and orchestration module, using Process Cloud that integrates hiring (Taleo recruit) with Core HR (HCM Cloud) and Payroll (Fusion Payroll).
- Extend and provide mobile applications customized for the mobile workforce using Mobile cloud service.
- Analytics dashboard using BI Cloud Service that collates data from disparate sources including SaaS applications to deliver comprehensive reports online and over mobile using Analytics Cloud.
- Out-of-the-box single sign-on between the extension applications and SaaS applications, including self-service user management and access control.

Table 5. Support for Extending SaaS Applications

AWS	Oracle Cloud Services	PCF
Not an enterprise SaaS provider; does not provide out-of-the-box extensions and integrations for SaaS	Leverages enterprise SaaS apps (e.g., ERP, CRM, HCM, SCM, EPM) Oracle AppDev provides out of-the-box extensions and integrations for SaaS	Not an enterprise SaaS provider; does not provide out-of-the-box extensions and integrations for SaaS

As illustrated in **Table 5**, Oracle has an extensive set of SaaS applications across ERP, HCM, CRM, SCM and others, and has provided tools and capabilities to easily extend these SaaS applications with extensions.[3] Further, Oracle PaaS can also be leveraged to extend non-Oracle SaaS applications and integrate those applications with Oracle SaaS and on-premises enterprise applications. In contrast, AWS and Pivotal do not offer enterprise SaaS applications and do not have the expertise or capabilities to extend Oracle or other third-party enterprise SaaS applications.

One company interviewed was a large Middle Eastern airport using Oracle PaaS to extend their HCM applications by implementing integration, application extensions, and mobile capabilities to "automate all of the employee life-cycle processes end to end." According to the head of manpower planning and HRD, "We started with the Java Cloud Service which we used to develop some of the extensions to fill the functionality gaps or to feed our custom processes, then we invested in SOA Cloud Service to build the HCM replica to provide as a base to fit into the extensions, then we built a custom mobile app using the Mobile Cloud Service to provide the functionalities on the mobile devices to the employees."

High-Productivity Applications

High-productivity applications refer to capabilities designed for rapid prototyping and development with more visual, less-coding use cases for either citizen developers or line-of-business application developers. They want to be able to self-service and quickly create an app without involving IT, which means a shorter turnaround for capabilities, quicker innovation, and a lower cost for developing and maintaining applications. Generally speaking, this extends the developer base beyond the pro coders to enable customization and extension of applications. Analyst firms refer to these solutions as high-productivity application PaaS (hpaPaaS).

The hpaPaaS is still in its early phase, but the need for these services was confirmed by many of our study participants. Analyst firms also suggest a large market for these services, with Holger Mueller, an analyst at Constellation Research, recently saying, "The addressable market for low-code developers is 15 to 20 times larger than the developer market."

An integration and cloud services provider shared how their marketing organization needed to develop a prize drawing application on short notice to support a trade show coming up in a matter of weeks. Given the short timeframe, they did not have the time to engage IT, articulate the requirements, and go through the traditional development process. Using Oracle's high-productivity cloud development tools—Autonomous Visual Builder Cloud Service (VBCS)—their marketing team was able to quickly build the application, on their own, in time for the trade show. They want to empower their marketing team to be more agile. Their chief executive officer explained, "It just created instant value, so not only did we save on cost, especially explicit cost, but we also had a marketing team that was enabled via true self-service. What

[3] https://cloud.oracle.com/en_US/java/java-cloud-service-saas-extension/features.

really impressed us was just how quickly they got up to speed given that they weren't developers, but they did have decent technical skills with their use of programs like Photoshop and others."

Table 6. Support for High-Productivity Application Development

AWS	Oracle Cloud Services	PCF
Limited support for high-productivity application development Signaled that a low-code offering may be forthcoming, but no details on timing or service definition	Mature hpaPaaS services includes Autonomous VBCS and MAX, which empower citizen developers and line of business users for rapid development Low-/no-code augments broader development community, particularly extending or cloud-enabling on-premises systems	No offering in this space and no announcements to deliver one Require third party hpaPaaS tooling

As documented in **Table 6**, our research finds that AWS and PCF currently do not have their own service offerings in the low-code arena. A customer needing to use low- /no-code application development will need to use another third-party tool. AWS recently announced a partnership with Bonitasoft and has hinted about a forthcoming offering, but no details have been provided. Oracle has mature offerings in the high-productivity areas such as Autonomous VBCS and MAX, which enable citizen developers to rapidly develop and deploy cloud applications and support IT developers to enable high productivity.

Mobile Application Development

Mobile application development is another important aspect in a complete cloud platform and one area that has been traditionally disjointed from the broader development processes. In a rush to "mobilize," enterprises often relied on device-specific approaches to deploying mobile capabilities and often created unsustainable practices. The reason is that having separate mobile and nonmobile development creates application silos and limits leverage, integration, and governance. Mobile development can also be accelerated in many cases by leveraging back-end applications and systems that already exist in an enterprise.

An integration and cloud services provider shared the experience of a pharmacy customer who wanted to rapidly extend their existing ERP system and create a shopping cart–type experience to simplify the ordering and replenishment of inventory and other supplies from tablets and mobile devices. Rather than build a completely separate mobile application or embark on a costly ERP development project to provide the needed capability, they took advantage of Oracle services for mobile development. In doing so, they could quickly model the user experience through a drag-and-drop interface while still being able to access the data model and broader application framework via Java scripting to provide additional logic and required validation.

The founder and chief executive officer cited the significant benefit, "Once they built the [mobile] application using Oracle Cloud Services they were able to swap out their legacy ERP application with a new one without affecting the end user experience at all. They were using Oracle's integration capabilities to essentially expose functionality like get order details, get

replenishment details, et cetera. They simply needed to re-point to a set of different APIs on the back end that had integration. That transition was amazing."

Table 7. Support for Mobile Application Development

AWS	Oracle Cloud Services	PCF
Mobile services include the Mobile Hub and Mobile SDK, a collection of services customers can assemble based on unique needs Lacks prebuilt integration with other platform elements Services do not currently include customer experience analytics **or** built-in runtime container	All-in-one mobile and chatbot platform Built-in customer experience analytics Preintegration with Oracle SaaS, PaaS Built-in runtime container Support for emerging experience channels	No offering in this space and no announcement to deliver one

Table 7 summarizes Pique's research on mobile cloud services. PCF was identified as having no current offering in this space. As such, their customers rely on third-party tools to perform mobile development in the cloud in a process separated from the primary development efforts. In terms of AWS, they do provide mobile application development, but their offering is more of a collection of services that customers bundle together themselves to provision a mobile development solution versus a singular mobile platform. That said, AWS has some very high-profile customers using its mobile development platform, including companies like Netflix. They tout benefits such as the ease of development and the ability to scale mobile applications on AWS infrastructure. But some other customers found AWS mobile development lacking some of the more advanced capabilities, such as customer experience analytics and built-in runtime containers.

Oracle's approach to mobile development offers a service more "productized" than that of AWS. A combination of Oracle Mobile Cloud Service (MCS) and Autonomous VBCS provides a comprehensive set of mobile and intelligent chatbot functionality along with prebuilt integrations to Oracle PaaS and SaaS services.

According to the founder and chief executive officer of an integration cloud services provider, "That was probably the thing that highlighted why Oracle VBCS was a good choice, because when you're developing the app, you're essentially creating a data model through a really easy-to-use interface. Then you're able to use those data models and integrations with other systems in other apps." He added that the application development was further augmented by MCS and MAX, with MAX enabling the scanning of PLU codes on mobile devices and MCS providing both the entry point for APIs to the back-end systems and the analysis of the application usage. He went on to say that "Oracle VBCS and MCS go hand in hand when you're developing these native apps."

A recreational sports equipment firm shared their experience with Oracle MCS to modernize how they worked with its global network of 10,000 dealers. Specifically, they wanted a faster, more efficient way for their dealers to submit claims. This was a highly manual process using the existing on-premises ERP application, and dealers would have to either access it through a computer or call the central office. With MCS, they were able to develop a purpose-built, easy-to-use mobile front end to submit claims using features of the mobile device, such as incorporating serial numbers and photos of the equipment as part of the claim.

They also use Oracle PaaS to enable the integration and process orchestration with their on-premises JD Edwards system. The result was reducing the manual claims processing to just a few taps on their mobile screen—a time effort reduction of more than 70%. According to the global ERP director, "We are taking a lot of functionality that dealers can currently get only via their computers or by calling us on the telephone and putting it on their handheld devices."

"We are taking a lot of functionality that dealers can currently get only via their computers or by calling us on the telephone and putting it on their handheld devices."

Director of Global ERP

Recreational Sports Equipment Manufacturer

Conclusion

It is evident that cloud AppDev is being widely adopted in a variety of industries and in an increasingly diverse set of use cases. As our research and the body of this white paper illustrate, companies are looking for a cloud provider who can support a broad spectrum of cloud AppDev needs from a single integrated platform. We found that Oracle offers its customers a choice of AppDev services that support their cloud-native and container-native development as well as their journey in moving from on-premises environments to the cloud. Oracle offers a more open and standards-based approach to both technical developers and citizen developers. Oracle is also expanding its portfolio to support rapidly growing AppDev needs to include blockchain development. Finally, study participants shared that Oracle provides more complete and integrated DevOps and management capabilities that cover all of its growing portfolio of on-premises and cloud applications, platform, and infrastructure.

AWS and Pivotal services strengths lie in the traditional cloud-native development arena, providing professional developers with effective tools to build cloud applications. Both companies have experienced early success in this area and have also augmented and extended their service offferings to accommodate container-native and microservices development. Both companies also provide tooling for DevOps and management, but their CI/CD tools are less complete and their platforms are more restricted to support only portions of their own service portfolios. Finally, there are notable gaps in some of the AppDev use cases for modernization of enterprise Java, SaaS extensions, high-productivity, and mobile application development.

By championing a hybrid approach and giving customers a choice of deployment in the public cloud, on-premises, or in their own datacenters—and moving back and forth as needed—Oracle provides flexibility that Amazon, and to a lesser degree, Pivotal do not. The ability to use familiar tools, compatibility with their existing applications, and support for mission-critical requirements also helped to make Oracle a clear choice for customers moving to cloud and developing applications for and in the cloud.