Cloud Customer Architecture for Mobile

Executive Overview

This paper describes vendor neutral best practices for hosting the services and components required to support mobile apps using cloud computing. The architectural elements described in the document are needed to instantiate mobile hosting environments using a private, a public or a hybrid cloud deployment model.

At a high level, cloud capabilities for mobile support the lifecycle of enterprise mobile applications that are deployed to employee or customer devices and provide managed access to backend business applications and enterprise data sources which support the mobile apps on the devices. These solutions allow companies to leverage emerging mobile technologies to reinvent customer relationships by engaging them anywhere and anytime the context is relevant.

Cloud computing and cloud services are a good match for supporting mobile devices. Mobile apps tend to have time variable usage patterns that are well handled by the scalability and elasticity of cloud computing - increasing and decreasing the backend resources to match the level of requests from the mobile devices. It is also characteristic of mobile apps to make use of server-side data that is unique to the apps. Some of this data is not enterprise data, such as social data (e.g., Twitter or Facebook data), and there are good cloud services available to incorporate such data.

Some data associated with mobile apps is accessed with a frequency and in a volume and format that can be difficult to accommodate with traditional enterprise transaction-based systems. It is common to support mobile apps with one or more databases containing the data for the app. Such databases typically hold copies of the necessary enterprise data in a form suited to serving the mobile apps, such as JSON data held in a NoSQL database. The elastic provision and support of these app specific databases is one of the notable capabilities of cloud computing. Using app specific databases also reduces the need to access enterprise systems and systems of record, along with a reduction in resource requirements.

Another factor influencing mobile app design is the often global nature of app usage. Users accessing the apps from many locations put pressure on the infrastructure requiring "local" endpoints around the globe to avoid latency issues. Cloud computing is well suited to running the same backend services in multiple datacenters around the world.

When developing and deploying a mobile app it is important to remember that mobile apps typically have a short lifecycle - they change fast to adapt to new devices and business markets. Therefore, planning for agility is another requirement of mobile apps - allowing for frequent, regular updates to the apps and the functionality that supports them. There is a need to support “2 speed IT” – where enterprises manage their on-premises systems of record enterprise systems at traditional change cycles while allowing applications at the edge or in the cloud to iterate faster – this includes being able to deploy quickly on devices and the mobile backend. Cloud services are good at supporting DevOps, agile development and operations capable of introducing new versions of apps and backend services rapidly

through use of automated test and deployment capabilities complemented with monitoring to validate operational quality.

As with earlier major transformative shifts in enterprise technology, a proliferation of implementation options and deployment topologies can make the adoption of mobile capabilities a challenge.

For example, different mobile device platforms, including Apple iOS, Google Android, Microsoft Windows Phone and BlackBerry, each come with their own application SDKs. They all support building apps using portable technology, however, native implementations involving a custom app implementation for each device type usually offer the best user experience. This means that organizations need skills and tools to support developing and deploying across all these devices and implementation types.

This proliferation also contributes to challenges with bring-your-own-device (BYOD) in the enterprise. Enterprises need ways to apply corporate policies to devices, which are allowed access to the enterprise network. This includes a means to distribute and update a portfolio of secure custom enterprise mobile applications for employees to use.

This paper highlights the mobile app lifecycle from the perspective of a cloud service provider and cloud service customer. The mobile cloud architecture guidance provided by this paper can help enterprises understand common architectures that have been proven in numerous successful enterprise deployments. Figure 1 shows the overall high-level logical architectural components for hosting a mobile app. It shows how a mobile device, managed by mobile device management, connects to the core cloud components including mobile gateway, mobile backend, mobile business applications, data services and security services while transformation and connectivity gets relevant data from enterprise systems and puts it in a format that can be leveraged on mobile devices.

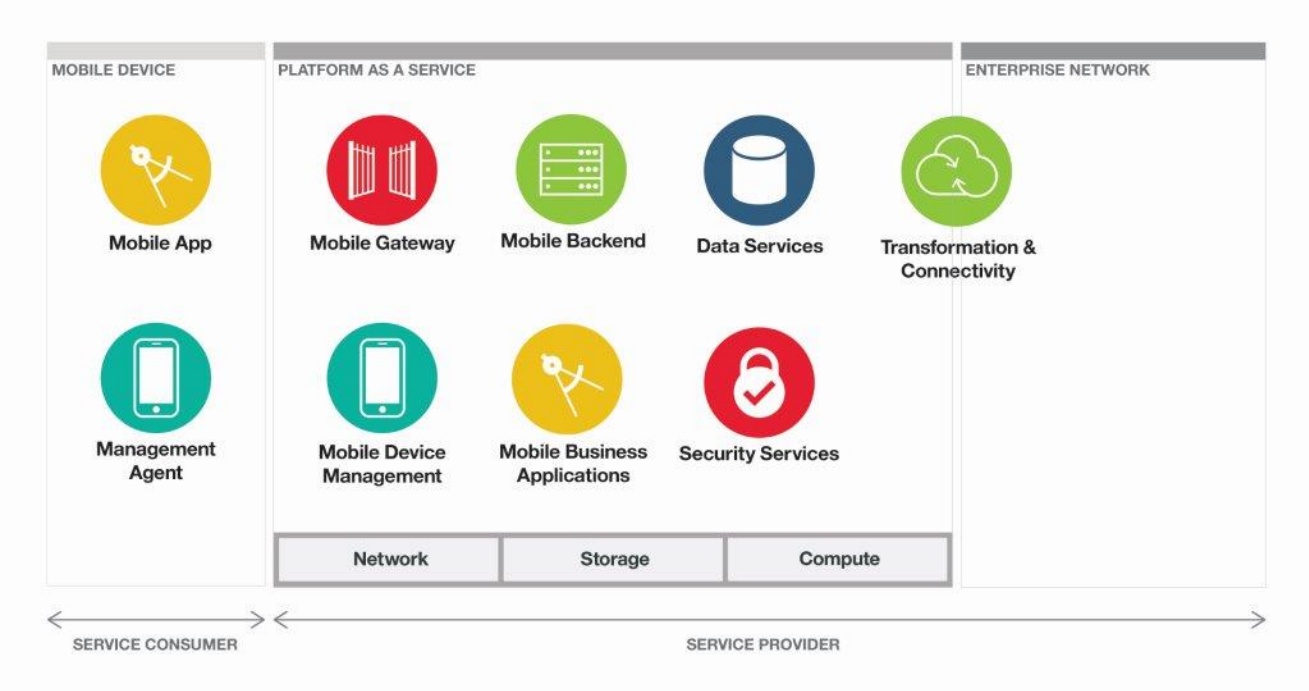


Figure 1: Mobile and Cloud – a high level view

Cloud Customer Mobile Architecture Components

As enterprises take their journey from viewing to transacting to collaborating using mobile devices, various patterns are identified and decomposed into a set of capabilities required to meet the end-to-end mobile app lifecycle. There are three stages in the mobile app lifecycle hosted on cloud that need to be considered:

1. Developing/deploying a mobile app
2. Running/ hosting a mobile solution
3. Supporting mobile app end users

This paper focuses on the core components of mobile hosting and consuming along with integration with the on-premises enterprise network. The components and processes for service/app creation are described separately from this paper. The architecture described in this section assumes that the mobile apps have already been developed and deployed.

Figure 2 illustrates the high level architecture of a mobile cloud solution. The architecture has 4 tiers, each containing a subset of the components:

- Mobile Device
- Public Network, which connects the device to the mobile cloud services
- Provider Cloud environment, where the various cloud services exist
- Enterprise Network, containing existing enterprise applications, services and data

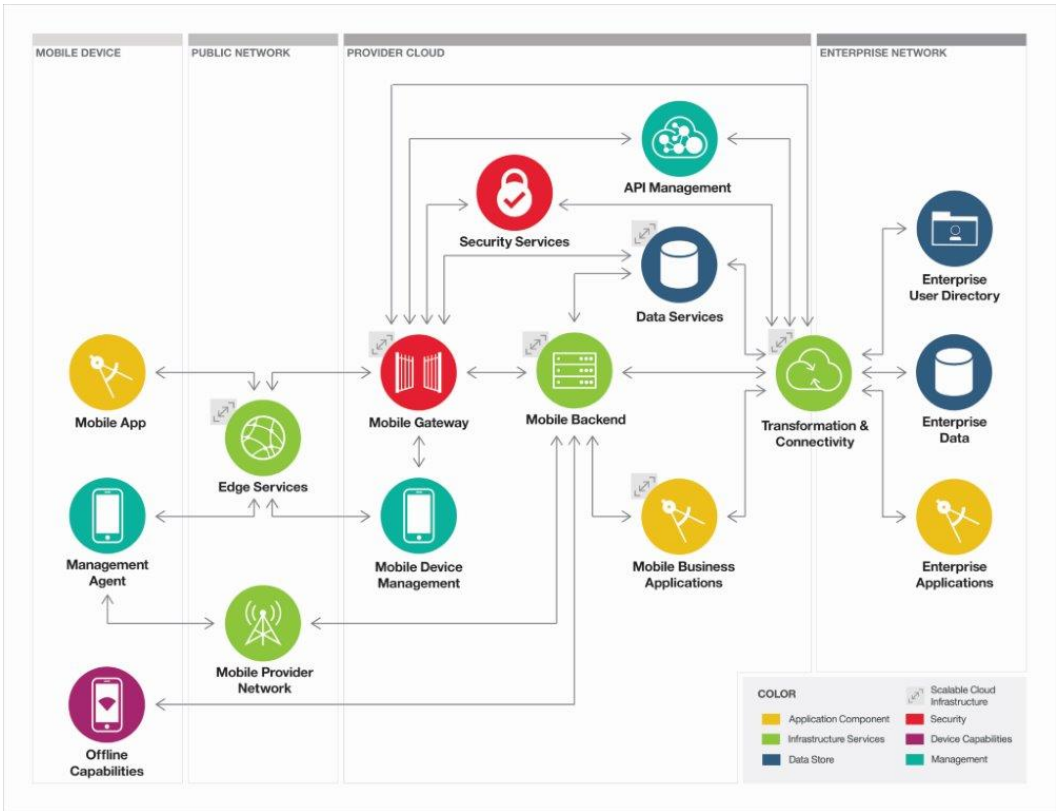


Figure 2: Cloud Customer Mobile Architecture

The following sections explain each major component along with their subcomponents. At the end of these explanations is a mobile architecture diagram with all of the components, subcomponents and relationships. Note that the arrow symbol next to several of the components in Figure 2 represents a scalable cloud infrastructure that is highly available and is not a single point of failure.

Mobile Device Components

Mobile App – Mobile apps are the main vehicle for user engagement with services on mobile devices. Although users can interact with websites through mobile browsers, the use of native mobile apps is the predominant use case. Mobile apps communicate with backend services using APIs, typically based on REST interfaces. Mobile apps contain two key components:

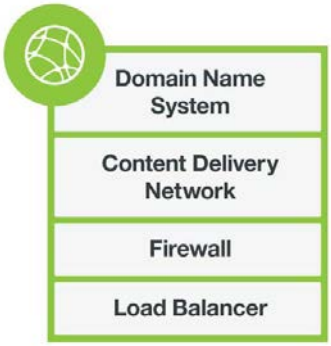
- **Vendor Frameworks** – Provide access to device capabilities and features from the device manufacturer and/or mobile network provider, like Apple Pay, Google Wallet, and Core Data.
- ***Enterprise Software Development Kits (SDKs)*** – Provide the ability to support communication with mobile backend services through SDKs that are consumable for mobile developers and encapsulates client flows needed to access backend systems.

Management Agent – Management agents apply the policies of the enterprise, typically for devices used by employees of the enterprise where the apps deal with sensitive enterprise data. The agent is a part of the SDK that stores, enforces, and manages policies, including security policies, on the device.

Offline Capabilities – Offline capabilities provide the ability to store data securely on devices and sync to the backend when the network is available. Since mobile networks are not always available to the device, a mobile app may use offline capabilities such as an encrypted database to access and store secure data.

Public Network Components

Edge Services – Edge services include services needed to connect the mobile device and its apps to the right mobile gateway through the Internet using Wi-Fi or mobile provider networks. These include:



- **Domain Name System Server:** Resolves the URL for a particular web resource to the TCP-IP address of the system or service which can deliver that resource.
- **Firewall:** Controls communication access to or from a system permitting only traffic meeting a set of policies to proceed and blocking any traffic which does not meet the policies. Firewalls can be implemented as separate dedicated hardware, or as a component in other networking hardware such as a load-balancer or router or as integral software to an operating system.
- **Load Balancers:** Provide distribution of network or application traffic across many resources (such as computers, processors, storage, or network links) to maximize throughput, minimize

response time, increase capacity and increase reliability of applications. Load balancers can balance loads locally and globally. Load balancers should be highly available without a single point of failure. Load balancers can support any of the mobile components, but support is especially important for the mobile gateway and mobile backend.

- **Content Delivery Networks (CDN):** Provides geographically distributed systems of servers deployed to minimize the response time for serving resources to geographically distributed users, ensuring that content is highly available and provided to users with minimum latency. Which servers are engaged will depend on server proximity to the user, and where the content is stored or cached.

Mobile Provider Network: The mobile provider network is the provider of wireless communications that owns or controls all of the elements necessary to sell and deliver services to an end user including radio spectrum allocation, wireless network infrastructure, back haul infrastructure, billing, customer care, provisioning computer systems and marketing and repair organizations.

Provider Cloud Service Components

Mobile Gateway – The mobile gateway marks the entry point from a mobile app to the mobile specific services for the solution, typically offering a set of internet-accessible APIs. The mobile gateway may also use data services and/or the enterprise user directory. A mobile gateway may be implemented by a common gateway across all channels into an API ecosystem. It provides:

- **Authentication/Authorization:** Provides the ability to identify, authenticate and authorize the user, using a variety of methods and token types using security services. Mobile authentication services provide the ability to handle different token types, like OAuth or OpenID as well as biometric technologies like Voice ID or voice authentication.
- **Policy Enforcement**: Enforces corporate policies during invocations from mobile devices
- **API/Invocation Analytics**: Captures analytical data of API invocation by a variety of clients (e.g., how often an API is invoked and who is invoking the API).
- **API/Reverse Proxy**: Provides the entry point of an API, usually in a DMZ. API proxy routes an API call to an implementation instance such as an application in the mobile backend.

Mobile Backend – The mobile backend provides runtime services to mobile applications which implement server-side logic, maintain data, and use mobile services. The mobile backend provides an environment to run application logic and the implementation of APIs. Application logic hosted here can communicate with the enterprise network as well as other services and applications outside the service provider. It provides:

- **Application Logic/API implementation**: Provides the implementation of the business logic being requested by the mobile app via defined APIs. The implementation may call on other services to provide required functions. A variety of runtimes such as Java or Node JS can be used to code the business logic.
- **Mobile App Operational Analytics**: Provide the ability to do analytics on runtime flows. The mobile backend collects and logs information from mobile apps, such as what client pages were visited, what backend functions were called, what device type called a particular backend, offline storage statistics, and so forth.
- **Push Notifications**: Provide support for subscription and sending of push notifications. Mobile apps allow users to register and receive push notifications while a mobile backend provides APIs for backend logic to push notifications to devices using the mobile provider network.
- **Location Services**: Provide the ability to collect and use location data from mobile apps running on a device.
- **Mobile Data Sync**: Provides the ability to synchronize data on a device that is stored in the backend.
- **Mobile App Security**: Interacts with Security Services to check authorization of users to perform app specific tasks.

Mobile Device Management (MDM) – MDM focuses on managing devices, mostly in Business to Employee (B2E) scenarios. MDM provides services to keep track of enterprise owned devices and also manage devices that connect to corporate networks using management agents on the devices. MDM provides:

- **Enterprise App Distribution**: Provides the ability to host enterprise catalogs and to distribute enterprise applications to mobile devices. If enterprise apps are not deployed to public app stores then enterprise catalogs are needed.
- **Mobile Device Security**: Interacts with Security Services to support enterprise security policies that need to be applied to devices. This includes policies on accessing enterprise networks, password standards, encrypted documents, device wiping, and so forth.
- **Device Management**: Provides the ability for an enterprise to view its organization-wide device usage as well as enabling administrators to add, remove, wipe, and perform actions across all of those devices.
- **Device Analytics**: Captures metrics on the actions performed by employees on devices that can help improve management of devices.

Mobile Business Applications – Mobile business applications represent the enterprise or industry specific capabilities that need to be available to devices that consume mobile services or drive

communications with users of devices. These can provide the gateway to enterprise applications and data, and include their own analytics components to track usage. They can include:

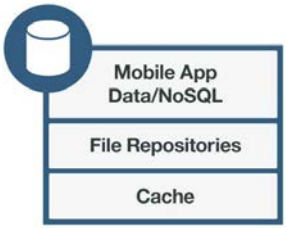
- **Proximity Services and Analytics:** Provides analysis and insight into patterns of activity in a physical location to optimize operations or facilitate next best actions. It connects insights from digital activity and physical presence to enable unique engagement with populations and the individual. It also enables contextually relevant mobile communications delivered at the right place at the right time.
- **Campaign Management:** Delivers contextually relevant experiences to connect with customers using mobile apps. This includes using different styles of *push* including (Apple/Android) Passbook, Wallet and SMS solutions. It connects with the mobile backend services and helps to send personalized messages to mobile device users and dynamic sets of individuals based on expressed preferences. This component applies deep analytics to help marketers and app developers understand mobile user behavior, preferences and usage, thus enabling them to quickly deploy mobile campaigns with relevant offers. It includes the ability to personalize mobile offers in real time, and execute cross-channel marketing campaigns.
- **Business Analytics and Reporting:** Provides complete mobile visibility by capturing user information for mobile websites including both network and client interactions and touch-screen gestures such as pinching, zooming, scrolling, and device rotation. This component can be used to build and manage an early warning system to detect mobile user problems and provide proactive awareness into mobile application failures, usability issues or other obstacles that lead to failed transactions, abandonment, poor app store ratings and negative feedback. It can also help quantify revenue impact and segmentation by analyzing specific mobile user behaviors or device attributes.
- **Workflow/Rules:** Orchestrates the flow of information at various points in the mobile architecture. A mobile client is integrated and synchronized with mobile business applications, mobile backend and enterprise systems that are potentially based on different workflow /rules engine.

API Management – API management capabilities advertise the available services endpoints to which the mobile gateway has access. It provides API discovery, catalogs, connection of offered APIs to service implementations and management capabilities, such as API versioning.

- **API Discovery/Documentation**: Provides the ability for mobile developers to find and use APIs securely.
- **Management**: Provides a management view into API usage by mobile apps using information from mobile gateway, backend, etc.

Data Services – Data services enable mobile app data to be stored and accessed. Mobile applications deal with data from many different sources. For example, a user's information exists in enterprise systems, on social networks, and a variety of other sources. Data is often stored in a form suitable for

rapid access by mobile apps and sometimes includes (potentially transformed) extracts of enterprise data. Data services can include:



- **Mobile App Data / NoSQL**: Stores data in a form that is easily and rapidly consumed by mobile apps.
- **File Repositories:** Provide the ability to store static files, such as PDFs and content.
- **Caches:** Provide the ability to cache data for fast access by mobile apps.

Security Services – Security services enable management of access so that only authorized users can securely access mobile cloud services. This component also provides protection of data across mobile devices and cloud services, and enables visibility to have actionable security intelligence across cloud and enterprise environments.

- **Identity and access management -** Identifies and authorizes the user providing risk and context based access to mobile and cloud services, including user management, authentication, identity federation, single sign-on, and mobile access management capabilities. These capabilities are leveraged by other components in this architecture – for instance, mobile gateway enforces user authentication and mobile access management, while enterprise secure connectivity enables security services to connect to enterprise security systems like LDAP registries.

- **Data and Application Protection -** Enables protection of enterprise data using a multi-level defense approach across infrastructure, application and data layers. Application security enables security as part of the development, delivery and execution of mobile apps, including libraries/tools to secure and scan mobile apps as part of the application development lifecycle. This component helps eliminate security vulnerabilities from mobile apps that access critical data before they are placed into production and deployed. Protecting deployed applications against application threats can be achieved through deploying web application firewalls. Data security capabilities support securing and monitoring access to data in mobile devices, enterprise databases, file shares, document-sharing solutions, and big data environments that may be accessed through the mobile platform, including encrypting data at rest integrated with enterprise key management, secure data in motion through secure connectivity architectures, and data activity monitoring that provides both real time data monitoring as well as vulnerability assessment. Infrastructure security capabilities are enabled by the edge services and the mobile device management components in this architecture.

- **Security Intelligence –** Enables comprehensive visibility and actionable intelligence that can help detect and defend against threats through analysis of events and logs and correlation and detection of high risk threats which in turn can be integrated with enterprise incident

management processes. These same capabilities could also enable with automated regulatory compliance and audit with collection, correlation and reporting capabilities.

Enterprise Transformation and Connectivity – The enterprise transformation and connectivity component enables secure connection to enterprise systems and the ability to filter, aggregate, or modify data or its format as it moves between mobile components and enterprise systems. Data transformation may be required when the native format of enterprise data is not appropriate for transfer to mobile devices.

- **Enterprise Security Connectivity**: Leverages Security Services to securely integrate with enterprise data security to authenticate and authorize access to enterprise systems.
- **Transformation**: Transforms data between enterprise systems and mobile components.
- **Enterprise Data Connectivity:** Provides the ability for mobile components to connect securely to enterprise data. Examples include VPN and gateway tunnels.

Enterprise Network Components

Enterprise User Directory – Provides storage for and access to user information to support authentication, authorization, or profile data.

Enterprise Data – One or more systems of record, for example, transactional data or data warehouses that represent the existing data in the enterprise.

Enterprise Applications – Applications that run enterprise business processes and logic within existing enterprise systems.

The Complete Picture

Figure 3 illustrates the complete picture for the ***Cloud Customer Mobile Architecture*** with all of the components, subcomponents and their relationships.

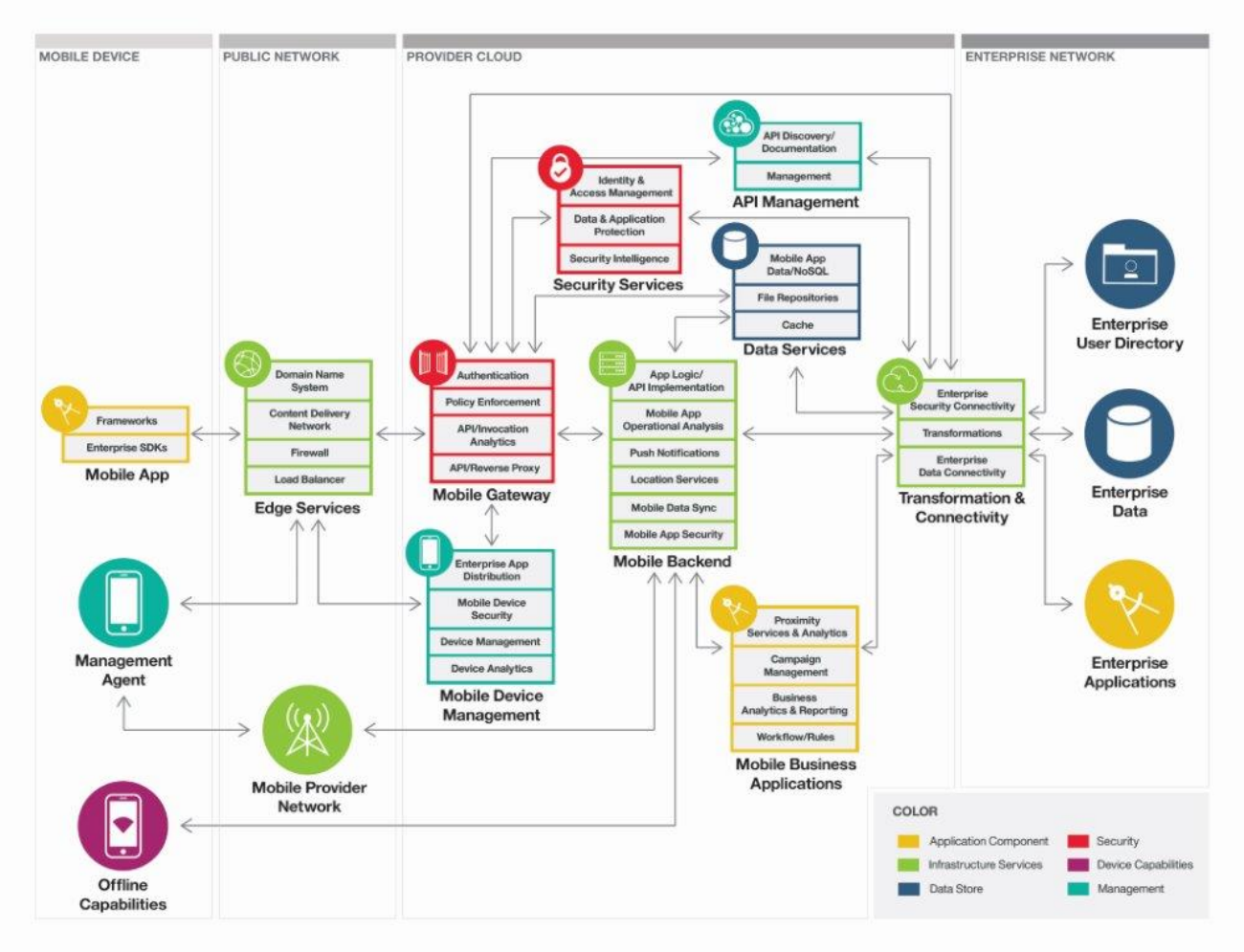


Figure 3: Cloud Customer Mobile Architecture Sub-Components

Runtime Flow

Figure 4 illustrates the flow of a typical use case for mobile banking. The mobile user installs the mobile app on their device, and then uses it to deposit a check to an account by taking a picture of the signed check from the mobile device. The bank also offers services to subscribe for text or email notification when certain events occur, such as an account falling below a minimum balance or possible fraud alerts. This scenario has 3 different flows:

1. Mobile app installation flow number 1 in blue.
2. Check deposit flow numbers 2-8 in yellow.
3. Push notification flow numbers 9-10 in green.

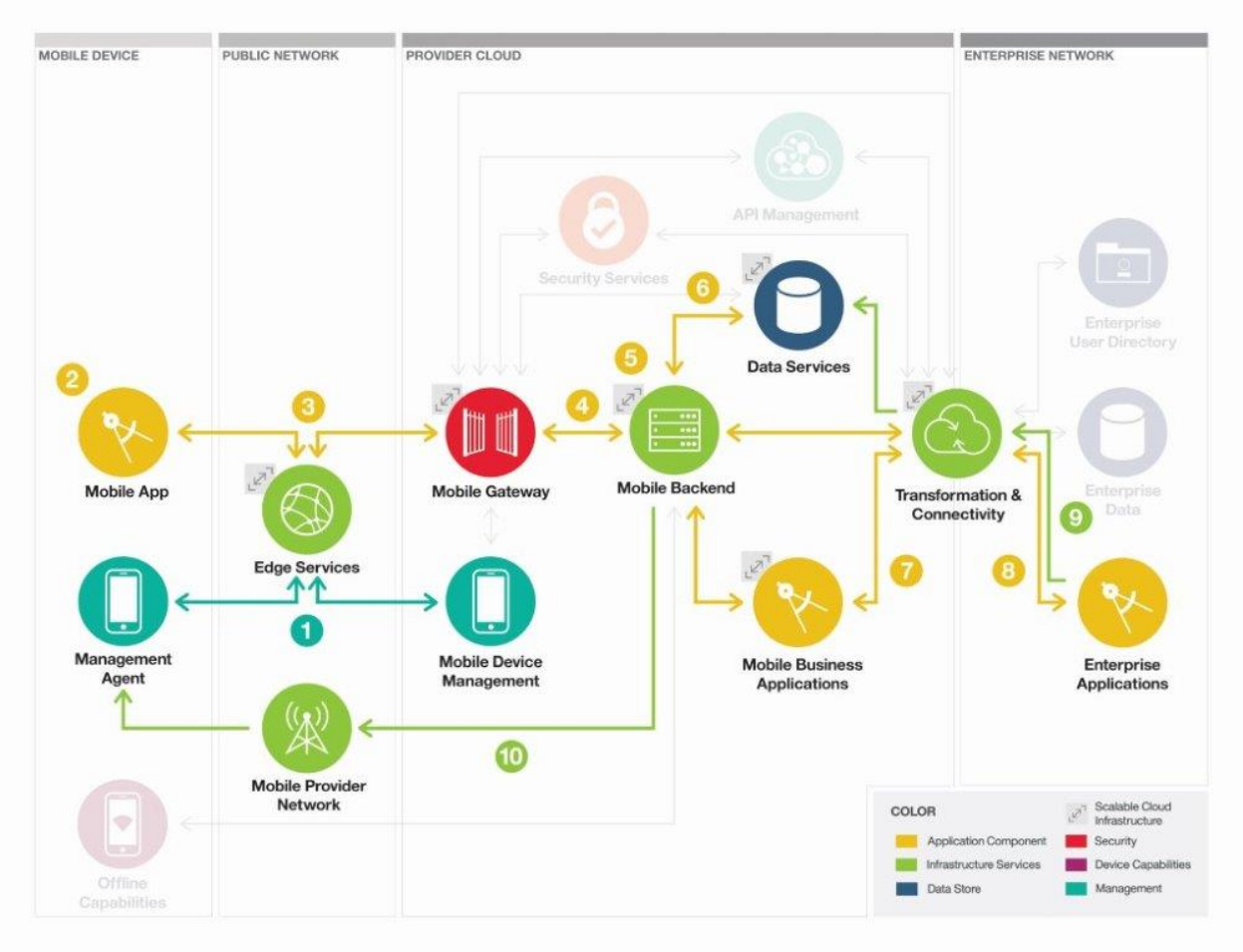


Figure 4: Flow of Interactions

1. The banking customer installs the mobile app onto their device after browsing a public application marketplace such as Google Play or the Apple App Store. In an enterprise usage scenario, the company - using their own enterprise app store and corporate mobile device manager - may instead push the application to the device over the public network. As part of the installation process, the user can opt-in for location aware services and sign up for push notifications on account balance changes or fraud alerts, for example.

2. The customer then uses the mobile app to deposit a check by taking a picture with the camera built into the mobile device. The user logs into the app (which will communicate with mobile gateway for authentication) and then sends the "deposit check" request to the bank with the check image. The user interaction is logged for understanding customer behavior and for understanding operational efficiency.

3. This service is located using DNS, load balancers, and other public network boundary components collectively known as edge services. For all transitions from the mobile app on a device to the mobile gateway through the public networks, which can be wireless or mobile networks, the mobile app sends requests using a URL resolved by a DNS to an IP address. The IP

address may be the IP address of a CDN server, load-balancer, firewall, or proxy service in front of the mobile gateway. The CDN server determines if the requested content is in the CDN storage network. If the CDN server cannot satisfy the request the request is sent to the firewall. The firewall evaluates the request and allows the request to continue forward to the mobile gateway if it meets the firewall rules.

4. The mobile gateway receives the deposit request and checks security rules for access to the "deposit check" service and uses an API service lookup to direct the request to the right service implementation in the mobile backend. The security check validates credentials and authorization of actions and passes a successfully validated user request on to the mobile backend. It then logs the activity for analytical purposes. Security services ensure that the user is authenticated and has appropriate access to the mobile application. The service ensures that the user is signed on to back-end services and systems.

5. The mobile backend executes the "deposit check" business logic to store the check image and send the check information to backend processes and systems to deposit the check in the customer's account. The service retrieves information through the transformation and connectivity components that enforce enterprise application security and ensure the account is valid. The mobile backend provides location services and manages subscription services for push notifications. It logs the activity for analytics usage. The mobile backend uses the workflow/rules service to start the deposit check process flow.

6. Data services may be used to speed up response time - e.g., the account balance could be stored in a NoSQL database and check images may be cached in the file repository. The "deposit check" business logic now stores the account, image of the check, and the deposit amount using the data services APIs. The information is logged for analytics

7. The mobile business application workflow sends the deposit check transaction through the transformation and connectivity components that enforce enterprise application security rules and grant access. The process flow uses different services to check the validity of the check, store the image of the check in the enterprise document repository, and deposit the check using the core banking application. The process execution step is logged for analytics

8. The enterprise account application stores the image of the check in the enterprise database for tracking purposes and applies the deposited amount to the customer's account in the core banking application. Control returns to the mobile backend.

   When the mobile backend "deposit check" service application completes its tasks, the resulting content is delivered through the mobile gateway (which logs information for analytics again) and the public network to the mobile app.

9. Some-time later, once the amount is added in the enterprise application, a request is sent through the transformation and connectivity components which uses the data services API to update the account with amount deposited and the balance. This information is now cached in

data services for expedited access and to save resources by reducing accesses to enterprise systems.

10. Depositing of the check information in data services invokes the push notification service in the mobile backend to send an alert to the customer that the check was successfully deposited. The mobile backend manages subscription services to determine and send the alert via a push notification on the public network to the device. The push notification service takes care of connecting to and using the right mobile provider network.

    The customer receives the notification that their deposit has been accepted and continues to interact with the banking application. The context of app usage is recorded for analysis by the bank to ensure ongoing excellent customer service.

Independently, the analytics being collected can be used for a variety of business purposes, including campaign management, fraud detection, and business presence needs. For example, as the analytics determines that this was a large deposit that can change the customer status to an elite customer, it sends information about elite customer service to the customer through push notification. Alternatively, locations service showing lots of activity from a particular geography may cause the bank to add an ATM locally or to engage them in local investment opportunities. The logs and events from the infrastructure, security components (firewall, mobile gateway, mobile device management, etc.) data and application access logs are sent to the security intelligence component for analytics. Correlating the collected data and application access logs, the security intelligence component can detect anomalies and report on unauthorized access as well as suspicious behaviors.

Deployment Considerations

Mobile capabilities can be deployed in a number of different ways. One dimension is whether to deploy the mobile services using an Infrastructure as a Service (IaaS) cloud service or whether to use mobile services provided by a Platform as a Service (PaaS) cloud service. A second dimension is whether the mobile services use a private cloud deployment model, a public cloud model, or some form of hybrid cloud model.

Using IaaS cloud services means that the cloud service supplies basic resources such as compute nodes and data storage capabilities – it is the customer's responsibility to install the required software for each of the necessary service components and to configure them to work as a cohesive whole. Some of the components can be bought as off-the-shelf software, while others (such as the mobile backend applications) must be purchased or developed by the customer.

PaaS cloud services typically provide many if not all of the mobile service components as a set of cloud services. The customer selects and configures the services they need – and develops code for custom components such as the mobile backend applications. The deployment of the underlying resources is largely automated by the cloud service provider, with minimal effort from the customer.

For public cloud deployment, the components are instantiated in a shared datacenter of the cloud service provider. For private cloud deployment, the components are instantiated either on-premises within the enterprise or within an isolated environment in the datacenter of a cloud service provider. For hybrid cloud deployment, there is an element of choice of where to locate each component, either

in a public cloud environment or on-premises. It is typically the case that public cloud deployment is likely to be lower cost than private cloud deployment, but may have security risks that cannot be accepted by the customer.

In each case, the actual deployment topology used is driven by business factors with the choice typically governed by security and performance considerations. Security and data protection considerations will depend on the nature of the data associated with the mobile apps and the business impact of risks such as unavailability of data or of data breaches. Generally, the greater the sensitivity of the data, the more likely it is that on-premises private cloud deployment is used. However, even if some components are deployed on-premises, it is not necessary that all components are deployed in this way.

For example, an enterprise can choose to deploy the mobile backend components in a globally distributed shared public cloud while keeping the mobile business application services and data within private data centers to meet performance objectives while at the same time ensuring proper security and protection for sensitive data.

The DNS and CDN usually live in the public internet - these are typically purchased as services from a suitable provider.

For IaaS cloud services:

- The firewall and load balancer are deployed in the cloud service. Many cloud service providers have firewall services available. The load balancer can be run on one or more servers within the cloud service.
- The mobile backend components and the mobile business application services and data components are deployed onto virtual machines, containers, or bare metal nodes provisioned in the IaaS environment. Networks are configured to allow traffic through the mobile gateway to the mobile business application and API networks, and to the mobile provider networks. Components can be collocated or installed onto separate virtual servers as desired.
- Significant consideration is required concerning the number of instances deployed for each component. For resilience and redundancy, it is advisable to have at least 2 instances of each component, preferably in geographically separated data centers. To take advantage of the scalability and elasticity of the cloud services, it is also necessary to increase and decrease the number of instances of a component dynamically according to the work load placed on it. This requires monitoring and management components and also requires appropriate load balancing in place for the component.
- The transformation and connectivity component spans between the cloud computing environment and enterprise system and consideration must be given to how it is structured – is it largely within the enterprise network, or does it mostly exist in the cloud computing environment? Performance and security are key factors influencing the design.
- For any components that involve data storage, such as the data services, considerable thought needs to be given to the number and the location of copies of the data. Replication and backup are necessary design points, as is consideration of the number of compute instances allocated to the components reading and writing the data, and also to the consistency model applied to the data.

For PaaS cloud services:

- The firewall and load balancer are typically part of the cloud service.
- The mobile backend components and the mobile business application services and data components are typically provided by the PaaS itself, only requiring allocation and configuration by the customer. Custom code must be developed for the application components, but this is typically deployed into runtime environments provisioned by the PaaS.
- Data storage components are part of the PaaS and it is typical for these to be provided with options for replication and backup - in the best case with customer control over the locations used to store the data.
- The transformation and connectivity component may be supplied by the PaaS, but it typically requires at least the installation of some connectivity code within the enterprise network.
- The identity and access service, typically part of PaaS, provides user management, federation/single sign-on, and mobile access management capabilities. These cloud services could be integrated into mobile cloud architecture through open standard APIs resulting in faster implementation compared to traditional on–premises deployment.
- For a PaaS, scalability and elasticity are usually built in, although often requiring configuration – for example, establishing a set of policies for when to increase and when to decrease the allocation of resources. Similarly, it is common for a PaaS to support load balancing of replicated components, often done transparently when multiple instances are allocated.

Regardless of where components are deployed - public, private or hybrid – lifecycle, operations, and governance requirements need to be considered and addressed. Where components are deployed will strongly affect how management and governance are done. Private deployments may be able to use existing enterprise management and governance tools if they have access to the cloud infrastructure. Lifecycle operations (instantiate, initiate, terminate) for components instantiated outside the enterprise need to be agreed on and supported by the cloud service provider for public, hybrid, and externally hosted private deployments. In all cases, the key is automation – as much as possible should be completely automated and manual interventions should be reduced to a minimum.

Similarly, operational monitoring and management interfaces for gathering metrics, checking SLAs, status, notifications, and negotiating changes in capacity for these public components will need to be obtained and support for them should be added appropriately to existing management tools. This may include integrating data, information, tools and processes from multiple sources into common interfaces, reports, automation etc. for efficient and scalable operations.

Governance and compliance processes will need to accommodate the change in control and risk over externally hosted components. Optimally, lifecycle management solutions should integrate across deployment models and provide a common, integrated context that enables management of release, change, security, SLAs, problem diagnosis etc. in a complex, dynamic and potentially unreliable environment.

Acknowledgements

The major contributors to this whitepaper are: Roland Barcia (IBM), Tracie Berardi (OMG), Anshu Kak (IBM), Heather Kreger (IBM) and Karolyn Schalk (Garden of The Intellect LLC).