

## Office of Technology Strategies (TS), Architecture, Strategy & Design (ASD)

### A VA Executive's Guide to Device-Independent Development

#### INTRODUCTION

In CTS Note #5, we introduced the concept of device-independent development, specifically as it pertains to secure mobile applications. This CTS Note takes a quick step back to describe device independence as a concept, and focuses on how it will help VA adapt to changing business, technology and industry trends.

#### BACKGROUND

As VA moves from a facility-centric, clinical services delivery operating model to a geographically constrained health management delivery model, we will see more and more devices used to deliver healthcare, manage benefits, and provide self-service to Veterans. The applications each user runs in order perform these functions must be able to run on any device, from the traditional workstation to the constantly upgraded smartphone.

Even with "Agile," Government software application lifecycles will never match consumer hardware lifecycles. Even if we could develop software as quickly as commercial independent software vendors (ISV), we will never be able to afford to replace or enhance them as often as new devices will roll out to the consumer market. This reality requires VA (and all large enterprises, as well as business application ISVs) to develop device-independent applications.

Device independence is the property of an application allowing it to run on different devices regardless of the hardware or operating system of the device.

First generation "mobile apps" had to be developed for specific devices, for three major reasons:

- Device-independent standards and tools enabling applications to use device features like cam-

# Technology Strategies

Defining OI&T's  
"To Be"  
Technology  
Vision



This newly established office within OI&T's Architecture, Strategy & Design (ASD) interacts not only with the ASD pillar offices, but also with multiple stakeholders within OI&T and with strategic offices across the enterprise. TS works closely with IT and business owners to capture business rules and provide technical guidance as it relates to Data Sharing across the enterprise, specifically for inter-agency operability.

eras, microphones, etc. did not exist.

- Building user interfaces (UI) that automatically reconfigured themselves for different sized screens was very complicated and required lots of code.
- Securing mobile apps was very difficult.

Today, numerous open standards and open source tools make it not only possible, but relatively easy to use local device hardware features, build "responsive" user interfaces and implement device-independent Federal Information Processing Standard (FIPS)-compliant security with only pure HTML5, CSS and JavaScript.

#### HOW DOES DEVICE INDEPENDENCE WORK?

Legacy applications rely on devices for key features such as processing resources (e.g., memory storage), device sensors (e.g., location), and user interfaces (not just a screen,

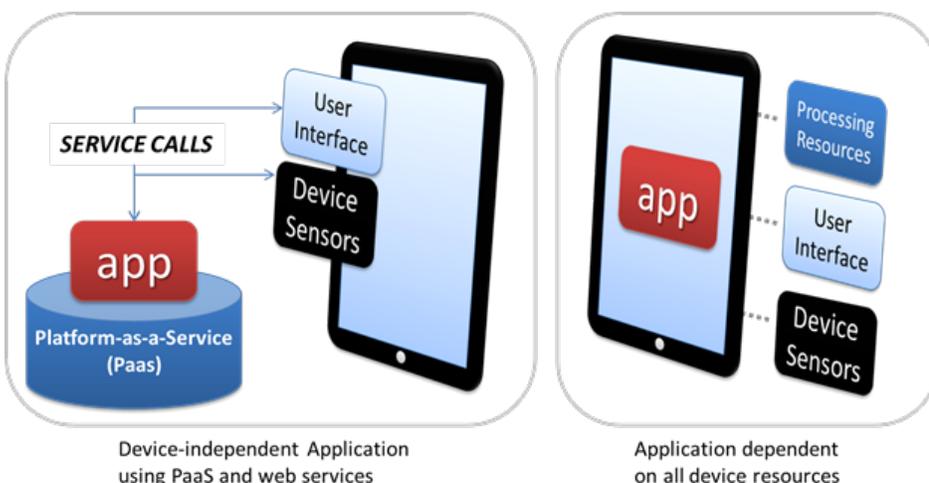


Figure: Device-Independent Architecture

## A VA Executive's Guide to Device-Independent Development

*Continued from Page 1*

but also cameras, speakers, etc.). Using service-oriented architecture (SOA) concepts, device-independent development decomposes these resources into services, either provided external or internal to the device. In this way the application no longer relies on how such features are implemented in a device (which can vary), but how the application accesses or uses the resources as services.

Developers use a platform-as-a-service (PaaS) to provide processing resources on which the application will run; and each app can rely on different PaaS implementations, further increasing the agility of the application. The other resources are also decomposed into services, meaning the application can use web services to access device sensor data or user interface specifications (see Figure). From the point of view of an application, then, the end-device is not seen as a static resource, but rather as a set of dynamic services.

### **BENEFITS**

Device independence is an important technology strategy for VA as it moves forward. The key benefits to adopting a device-independent development framework are as follows:

#### **Improved Information Security**

Removing applications from the secure networks in which they sit now may not seem like a beneficial move. But as network security updates fail to keep up with emerging mobile security threats, those devices and applications cannot rely on the network to be secure. Applications can now be developed with native security features, such as device-independent FIPS 140-2 encryption. This fit-for-purpose development means device-independent applications are better suited to protect against constantly evolving cyber threats.

#### **Increased Information Agility**

Device independence is perhaps one of the best examples of agile technology. VA and private medical centers may use different devices, but in a device-in-

dependent world, medical professionals can access health data captured on multiple devices through a single application. Further, non-medical staff can rely on a device-independent app to reuse medical data for administrative purposes.

#### **Decreased Time & Cost**

IT departments cannot customize existing legacy applications for each device. By the time developers have spent their entire budget to make the app compatible with a specific device, the device will no longer be adopted by the most users and the cycle will continue. If each application is device-independent, it requires only one development lifecycle to create the core application. Future enhancements, either to the app itself or to its interoperability, can be made by reusing existing services.

Device independence ties directly into the key business and IT drivers VA is facing today. Previous CTS Notes have also discussed technologies and strategies that rely on device independence. Device-independent architecture relies on SOA principles to decompose computing resources into services that applications can access regardless of the device running it. Even innovations in fitness devices and patient-generated data (PGD) rely on the ability to run apps that collect data from one device and store data in medical records accessed on another device. Ultimately, the growing use of mobile devices and mobile applications across VA requires device independence in order to ensure apps are interoperable and secure across any mobile device.

If you have any questions about device-independent development, don't hesitate to ask CTS ([askCTS@va.gov](mailto:askCTS@va.gov)) for assistance or more information.

Check out earlier CTS Note editions [here](#)

([vawww.blog.va.gov/oit360](http://vawww.blog.va.gov/oit360)).