Open and Safe HIT Platforms

How much do you really want a HIT platform to be like an iPhone?

Authors

- Ben Adida

Published

Version 12

Last edited: Sep 21, 2009

Exported: Aug 16, 2012

Original URL: http://knol.google.com/k/-/-/37idooo44z8da/1

Introduction

Kohane and Mandl [1] argue that health information technology should follow the iPhone example and move towards a platform architecture with ***substitutability***, i.e. modular components that can be substituted for others, with the user as the ultimate decider. The iPhone analogy is particularly powerful, because the broad strokes of such a platform are immediately clear to millions of existing users – and millions of others who have at least seen an iPhone television commercial declaring "there's an app for that!": if you don't like one app for job X, just download another app that does job X better.

However, Fred Trotter argues that the iPhone analogy is poor [2], because Apple:

- plays favorites with advance access to hidden APIs
- approves applications using an inscrutable process
- forbids applications that duplicate core functionality, e.g. phone/SMS.

- is the sole source of approved applications

Kohane, Mandl et al. followed "No Small Change" with the "Ten Principles" [3]. Interestingly, most of the design principles they recommend directly conflict with Apple's regulation of the iPhone ecosystem, almost exactly as detailed by Trotter, in particular:

Principle #4 states that "Third-party vendors should be able to develop [...] without barriers." This conflicts with the iPhone's strict and unpredictable approval process, and Apple's apparent decision to play favorites with some applications.

Principle #5 states that "platform vendors should not have control over what is installed on the platform. " This conflicts with Apple having sole control over which applications are allowed into the ecosystem, since even corporations who use the iPhone in a business setting are limited to

applications Apple approves.

Principle #8 states that "All applications should be removable and none should be required." This conflicts with Apple's stated intent to reject applications which duplicate built-in functionality, like SMS and VOIP.

Principle #10 states that "Certification requirements [...] should be kept minimal to maximize substitutability." This conflicts with Apple's practice of rejecting applications for subjective reasons, such as the use of "confusing icons."

Open-Source as the Solution?

Some have suggested that other platforms like Android are more compatible with the Ten Principles [4], because the Android platform is open-source, and Android phones typically allow users to install applications of their own choosing. Open-source, however, does not guarantee compatibility with the ten principles: a phone vendor could easily sell Android-enabled phones where only approved applications are installable, and where the phone's operating system, though open-source, cannot be physically updated on the phone without vendor approval, e.g. using digital signatures to approve software updates.

A Platform can be "too open."

Platform designers and operators seek, for various reasons, some amount of control over the direction of the ecosystem they foster. When announcing third-party application support for the iPhone, Steve Jobs wrote [5]:

"It will take until February to release an SDK because we're trying to do two diametrically opposed things at once: provide an advanced and open platform to developers while at the same time protect iPhone users from viruses, malware, privacy attacks, etc. This is no easy task."

Though it is clear that Apple's regulation of the iPhone apps market has gone far beyond malware prevention, the *goal* of malware prevention is certainly reasonable. An iPhone that can no longer receive calls because an application misbehaves would significantly reduce the functionality of the device and justifiably tarnish Apple's reputation. More generally, an application ecosystem that doesn't provide some basic measure of safety would slow innovation and cast a doubt on extensibility in general, as users would shy away from installing applications.

This inherent conflict between openness and safety will inevitably play out in the health IT space, especially as the importance of Personally Controlled Health Records grows and end-users are involved in the application selection process. A key question is whether we can overcome the unproductive deadlock of extreme measures on either side: *neither closed platforms that stifle innovation, nor fully anarchic platforms that provide no safety guarantees are likely the right answer.*

A balanced approach for PCHRs: Open/Branded/Trademarked

One approach worth serious consideration, especially in the tricky PCHR space, is the "Open Software / Branded Platform / Trademarked Interoperability" model.

In this model, the core software, e.g. Indivo, is open-source, and multiple installations are deployed by independent organizations (Children's Hospital Boston, Dossia, ...) Because the core software is shared between these installations, the platform API remains relatively consistent, and applications built for one installation are easily ported to another. This constitutes the "Open Software" portion of the model.

Then, individual platform deployments can design their customized safety conditions. Children's Hospital Boston may be particularly conservative with the applications it enables within its ecosystem, given that most patients are minors with complex privacy constraints. On the other hand, a commercial Indivo installation targeted at adults with serious medical conditions might choose to be far more permissive in its list of acceptable applications, given that sick adults tend to focus on care before privacy. Users of each installation would immediately know which organization they are dealing with, given that each installation is branded by the deploying organization. Thus, each installation puts its brand at stake when it bans or promotes a particular application. This is the "Branded Platform" portion of the model, where end-users come to trust particular PCHR deployments based on their ecosystem policies.

Finally, users must have the ability to migrate from one platform deployment to another when they decide they would prefer to live under a different safety regime, either more or less permissive. To ensure that all platform deployments respect the user's decision to migrate his record, each deployment would be required to pass a battery of compatibility tests to be branded "API compatible" with the core release. This requirement would be expressed in the API's trademark, e.g. the Indivo trademark for some PCHRs: only those deployments that remain compatible with key portions of the Indivo API would be legally permitted to call themselves "Indivo compatible." Users would then instantly recognize the set of PCHRs which clearly enable migration from one PCHR deployment to another. This is, of course, the "Trademarked Interoperability" portion of the model.

The Open/Branded/Trademarked model is designed to provide the right incentives for interoperability and user choice, balanced with patient safety, in the PCHR market. Specifically:

- if the open-source project deviates from its core mission, a separate team can fork the code
- if existing deployments do not provide the range of features and safety regimes that some users seek, then anyone can create a new PCHR deployment using the open-source code
- if some deployments attempt to entrap users by allowing migration in, but not migration out, these deployments would be in violation of the core project API's trademark

Conclusion

We do not claim that this model is the only viable approach: other models may well balance these competing interests. It is clear, though, that when balancing innovation and safety in a distributed application environment, a single feature, such as open-source, doesn't capture the complexity. Some level of compatibility enforcement is likely needed, so that not just code, but also data, can be ported.

It will be interesting and important to imagine how closed-source systems can play a role in balanced ecosystems. It will also be crucial to determine how much of this balance is required in enterprise environments, where increased control by local administrators is expected.

References

1. Mandl KD, Kohane IS. No small change for the health information economy. N Engl J Med. 2009 Mar 26;360(13):1278-81. PubMed PMID: 19321867.
2. The iphone, a poor HIT platform analogy
3. Ten Principles for Fostering Development of an "iPhone-like" Platform for Healthcare Information Technology
4. Ten Principles for Fostering Development of Platform for Healthcare Information Technology
5. Steve Jobs Announces 3rd Party SDK for iPhone for February 2008