How Open is Your System Architecture?

Gavin Tong

MBA, CPHIMS-CA, Standards and Interoperability Editor, is an Associate Managing Partner with Gevity in Toronto, Ontario

Rod Thurber

BCS, CPHIMS-CA, is a Senior Architect with Gevity in Halifax, Nova Scotia

8 billion dollars. That's the amount invested in Ontario's EHR between 2002-2015 according to the Ontario auditor general.[1] The good news is that according to Ed Clark, Ontarians have received 5.7 billion dollars in value between 2007-2015, and the benefits received will continue to outpace investments. However, overall costs of care will continue to rise and everyone in the system needs to find ways to be more efficient so that we can help achieve the triple aim of care.

This challenge isn't Canada's alone. Developed and developing countries around the world face the same trade offs of the cost of providing care outpacing the ability to pay for it. The only difference is the magnitude of the dollars available. In Myanmar, GDP per capita is $1,308.70 USD[2] compared to Ontario's at $43,317.13 USD.

Whether you're deploying an eReferral system in Ontario or a public health surveillance system in Myanmar, everyone is trying to design, develop, and deploy their healthcare IT systems in the most cost efficient way.

We've previously discussed the role of innovation in IT to help drive down costs to the system overall[3], and in this article, we'll discuss how open system architecture facilitates adoption by the service consumers and lowers IT costs related to maintenance and scalability.

Open System Architecture

Open system architecture is essentially the design of systems – such as jurisdictional EHRs and their various sub components - to be vendor independent, non-proprietary, and use popular standards[4]. One of the goals of open architecture is to make it easy for users to swap components or add new ones to extend the useful life, functionality, interoperability, and overall value provided to the organization and its stakeholders.

In simpler terms, open system architecture helps organizations create the technological platform that supports innovation by facilitating external stakeholders to build products that consume the organization's services and data. It is a purposeful way of designing IT systems so that organizational data and services can be exposed to external users who want to innovate.

Open system architecture and SOA-based architectures share the same principles of decoupling components and using multiple, purpose-built, services that effectively decompose the system into discrete and isolated subsystems that communicate over well- defined protocols. In an open systems architecture this decomposition creates an ecosystem of microservices that can be easily adopted by consumer systems.

The ecosystem of microservices contributes directly to longer term cost savings. For example, if the EHR components are deployed in a 'bricks and mortar' setting, microservices facilitates the ability to swap out backend components (e.g. the various registries and repositories), with minimal disruption to the consumers of the microservices.

Microservices also facilitate the switch over from 'bricks and mortar' back end components to lower-cost cloud based services. This is particularly important in developing nations that receive one time funding to design and implement systems, but need to find ways to lower ongoing maintenance costs.

Open System Architecture helps achieve the benefits above through adherence to several key principles:

1. **Uses proven, open, standards:** An open architecture enables the deployment of multiple integration methods based on established standards that are consensus based, widely used, and easily accessible to potential users.
2. **Uses emerging/popular standards:** This may seem contradictory to the first principle, but that's only if you view data exchange as a choice of one standard over another. Systems using open architecture support the ability to stand up new microservices or adapters to support emerging and popular standards, which allow new consumers to connect to the back-end systems without forcing existing consumers to change their applications. It also has the added benefit that popular and emerging standards tend to have larger labour pools to draw from, which helps lower human resource costs.
3. **Non-Proprietary:** An open system architecture is designed as an official standards-based architecture and does not expose any proprietary technologies or integration methods for service delivery. Removing proprietary integration points and services allows for a less complicated and more responsive integration environment.
4. **Abstraction:** One of the foundational architectural concepts of an open system is abstraction. With purpose built integration and access services sitting between participating applications and back-end services, any changes made to the back- end won't affect connections to participating applications. The use of abstraction and standards based integration ensures that the solution architecture provides vendor agnostic access to participating applications and services. Abstraction is enabled through the use of integration and service access layers which are underpinned by an enterprise class services where required and the functional alignment of the solution components. Abstraction of the service interfaces means participating applications are loosely coupled. Loosely coupled solutions have the additional benefit of reducing change and release management activities and reduction of the risk and complexity of any changes made.
5. **Adoption:** The architectural principle that enables rapid adoption is API-led connectivity. The fundamental building blocks of open system architecture are development of APIs (services) in order to meet application requirements, while establishing policies and managing access to backend data through the deployment of a configurable set of security and privacy services.
6. **Separation of Concerns:** The architecture should separate microservices into conceptual functional categories, such as Integration Layer,

Service Access Layer, Privacy and Security, Core business services, and Reporting. Implementing microservices within each of these categories helps simplify the development and maintenance of the services while supporting reuse and independent scalability. These architectural outcomes will also lower the operational cost of the solution.

The international development community has recognized the importance of open systems architecture, particularly because the countries that need integrated health IT systems the most simply can't afford to implement, let alone maintain, the monolithic systems we see in Canada. Perhaps the best example of this is in the architectures promoted by the Open Health Information Exchange (Open HIE).

Open HIE is a community of practice based on the experiences of developing the Rawanda HIE. Open HIE established reference architectures, guiding principles, and standards that adhere to Open Architecture principles. We're seeing a number of the global funding agencies and their partner countries request that digital health initiatives follow Open HIE approaches because it helps ensure they make smart IT investments.

Adherence to the principles of open system architecture isn't always easy. Organizations will always be saddled with legacy systems that break at least one of the principles open system architecture and can't be easily replaced. However, the goal is to use the principles of open system architecture in shaping IT roadmaps and procurements to ensure future IT investments yield their maximum benefits.

---

[1]http://www.auditor.on.ca/en/content/annualreports/arreports/en16/v1_303en16.pdf
[2]http://www.tradingeconomics.com/myanmar/gdp-per-capita
[3]http://www.healthcareimc.com/main/government-role-in-open-innovation/
[4] http://www.businessdictionary.com/definition/open-system-architecture.html