SPOTLIGHT: BEHIND MHERO - A LOOK AT INTEROPERABILITY, ARCHITECTURE, STANDARDS AND THE VALUE OF OPEN SOURCE SYSTEMS

May, 2016 | Interview by Dykki Settle with Amanda Puckett BenDor and Carl Fourie

We know the big picture: Health information systems (HIS) such as iHRIS for collecting and managing health worker data and DHIS 2 for tracking indicators on health services and surveillance are essential tools for improving health systems. These systems also provide platforms for new tools that use their data to meet additional needs such as informing health policy and advocacy. mHero is an example of one such tool. mHero uses facility and location information from DHIS 2 and health worker data from iHRIS to connect ministries with health workers providing real-time information for decision making. A part of the Ebola response, mHero scaled in Liberia, has been included in the national health information strategic plan in Liberia and Guinea, and is soon to be piloted in Sierra Leone. While mHero started as an epidemic response tool, it has become a powerful health system strengthening component, creating two-way communication links between the Ministry of Health and health workers that never existed before.

With tools like mHero, you frequently find a technical team working to ensure the technology functions properly as well as a health team which uses the system to inform decision making about service delivery or supporting health workers. These teams often do not interact and have limited understanding outside of their technical or operational scopes. However, as we transition into a world where more health information systems will be implemented, linked together and used to strengthen health services, a common understanding of how these systems work needs to be developed between the technicians and health professionals. This includes a basic understanding of important components behind the technology, such as interoperability, standards and the architecture – topics we will speak about today.

Both teams need to understand why essential global guidelines, such as the Principles for Digital Development, are important and how their work can be strengthened by them. We also need to facilitate dialogue between technical and health teams as they increasingly depend on each other to strengthen health systems.

My name is Dykki Settle and I am the Director of Digital Health Solutions at PATH. I am proud to be an active contributor to all of the technologies mentioned above and excited about the value they bring to health systems around the world. Bridging the gap between "techies" and "healthies" is a passion of mine - as health becomes increasingly digital, we can't afford misunderstandings or slowdowns due to cultural and language

differences. At PATH, the "packaging" and "consumability" of digital health technologies is as important as the technologies themselves. Technology and innovation only has value to the degree it is adopted and used.

Seeking to understand this gap better and how we might transcend it, I sat down to talk with Carl Fourie from Jembi Health Systems, who can be seen as a techie, and Amanda Puckett BenDor from IntraHealth, who can move between the health and technology worlds. We spoke about how to build capacity and increase understanding of health information system concepts for both technical and health-oriented implementers.

Dykki: Carl and Amanda, interoperability has been a "buzz word" for a while. What does this mean and why is it so important for health information systems and global health? What are some of the challenges with interoperability?
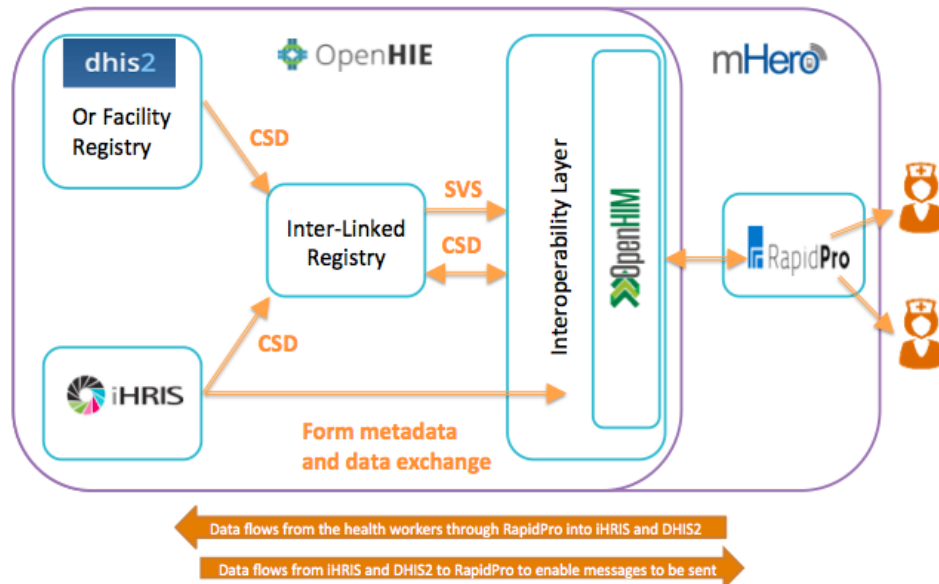
Amanda: Over the past 10 to 20 years, we have invested a lot of time and money into developing robust information systems to support global health. Interoperability is about leveraging those investments and essentially connecting those systems so they can share information. So if data is collected in one system on the number of deliveries at a health facility, that system can be made to exchange data (be interoperable) with a system that collects information on health workers so you can see how many qualified health workers are at that facility to support maternal and child health services such as safe delivery. With interoperable systems, so much more data can be mined for trends.

Program Managers may not think of it this way, but data tells a story and data exchange (and interoperability) allows for a much richer and detailed story. It benefits us all to find more ways of connecting systems, building capacity to interpret the data and increase demand for that data to inform decision making. We always talk about tearing down silos - this is one way to do so.

Figure 1: mHero environment and data flow

The mHero platform is the quintessential example of interoperability. **Figure 1** illustrates how the different systems work together to send SMS to health workers. mHero takes already existing open source systems - IntraHealth's health worker data management system iHRIS and UNICEF's interactive mobile messaging system RapidPro - and combines them so that Ministries of Health can create mobile messages, or SMS, workflows in RapidPro, use iHRIS to select which health workers they want to send an SMS and send them. mHero also utilizes health worker and health facility registries and links them with DHIS 2 for facility data.

Operationalizing interoperability is a challenge due to issues of data privacy, ownership, security and governance. If a health information system sits in one department of a Ministry of Health and works to share data with another system in another department, data access and ownership becomes a question. Countries are starting to explore data sharing agreements that elaborate on how and with whom data will be shared, the security of that data, and communicating about findings. I often tell people that the technology is the easy

part - it is the governance of these systems that is the most challenging because we are working in new and innovative ways.

Carl: I agree with Amanda, in comparison to people, the technology is the easier part of interoperability. We often talk of interoperability at 4 levels:

- Hardware: bare metal devices working together
- Syntactic: a common way of describing data that needs to be exchanged
- Semantic: common meanings of data (for example, the measure of Gallon varies depending on whether someone is in the US or UK. Can you imagine this in health data?)
- Organizational Interoperability: Amanda referenced it above, organizations policy, process and data exchange.

Before unpacking this too much I find it good to frame what is meant by "interoperability." The word interoperability means many things to many people and often gets interchanged with "integration" and vice-versa. To understand why it is important, we need a clear distinction and definition of it.

[Booby Woolf](#) has a nice [summary](#) of interoperable and integrated information systems:

> ... *interoperability means that two (or more) systems work together unchanged even though they weren't necessarily designed to work together. Integration means that you've written some custom code to connect two (or more) systems together. So integrating two systems which are already interoperable is trivial; you just configure them to know about each other. Integrating non-interoperable systems takes more work. The beauty of interoperability is that two systems developed completely independently can still work together. Magic? No, standards (or at least specifications, open or otherwise);*

So we see that Interoperability is more than just getting data to move between systems, it focuses on the "how" just as much as the "what" is been exchanged.

Having interoperable health systems is an important step in global health as it allows for incremental upgrades, updates or changes to an information system ecosystem without needing to rip it all out and replace it. More so it allows for new systems to leverage the expertise and content of existing systems, to add to the overall "digital picture" and to avoid unnecessary parallel systems.

It is important for the people who design and create health information systems – the developers and architects – to consider the current and future models when designing tools and solutions and being aware of the need and ability of their tools to be interoperable. Interoperability doesn't happen without thought, design and effort in software development, and health information system developers, particularly those focusing on low- and middle-income countries, should spend time considering how their tool could contribute or leverage off of existing infrastructure as well as for part of them.

Dykki: Describe the emergence of health information system architectures. What are some of the different types of architectures? How can Ministries of Health identify the best design to meet their needs?

Amanda: To describe the emergence of HIS architectures it's best to clarify the understanding of the word architecture for the conversation. An architecture is a formal description and representation of a system/solution, organized in a way that supports the operation and meets the required needs as well as expectations of the system. A HIS architecture is a logical description of how an information system is designed to meet the envisaged health business function(s) within the constraints of its environment.

It takes someone with vision and understanding of the overall health system to see how different health information systems fit together. This vision should align with national health priorities and strategies which are different in every country. An HIS system architecture is roadmap of how systems and the people behind those systems will work together. A well thought out HIS architecture will function to link data and systems, be a guide for governance of data and data sharing and clearly illustrate how health information systems will be used within a group of stakeholders like staff from the Ministry of Education and other areas in national governments. This may be in the form of frameworks, standard operating procedures or other agreed upon guidance documents that make the architecture implementable. By this I mean that data flows through different systems but more importantly the right people have access to the right data when they need as sound evidence to inform budgets, policy etc.

Carl: From a technical aspect the word architecture brings to mind a range of different architectural patterns. A list of some which are found in [Appendix A](#) below.

It takes someone with vision and understanding of the overall health system to see how different health information systems fit together. This vision should align with national health priorities and strategies which are different in every country.

This is not an exhaustive list and there are more approaches and combinations of these styles being used as designs are informed by experience and understanding. More often than not a system (HIS Systems inclusive) is architected using a combination of styles to meet the challenges and objectives of the functions and domains it caters to. From a technical perspective, when designing a system it is important to match the approach to the problem at hand and take into account the non-functional requirements – criteria that act as goals for the system, such as "the system shall be backed up daily" – as well as the functional requirements, criteria that specific how the system will meet that goal, such as "the system will run specific backup software like [Bacula](#).

In our experience, and tendency, in the design and implementation of health information systems for developing country settings, is to use a combination of patterns that allow us to meet the needs of scalability, control and auditing of data through the system, functional domains and the dynamic addition and "swap out" of components. It is important for countries to consider not just what needs to be accommodated now in the design of the system but also to design with the idea of what may come. This helps to ensure that the design can accommodate change in an appropriate way. For example, when designing a national-level system to register pregnancies, is one also considering how the system can be leveraged to track vaccinations or ANC visits? Questions like these help to encourage a flexible and expansible design that can be incrementally built (a build-as-you-go approach) to make optimal use of time, effort and resources while leveraging existing infrastructures to support new and existing use-cases.

Dykki: Let's talk about data standards, particularly those which support health information system interoperability. What are some of the new standards for health information systems? Why are these important for developers and program managers to understand?

Amanda: Standards are necessary for interoperability. As we discussed earlier, interoperability is how all of our systems can work together to share and exchange data. System standards provide guidance to help the developers write the correct code to make this possible. I think it is important for program managers to understand that technical standards exist for interoperability and data sharing. This improves data quality, and we all know, no matter how robust a system is, without reliable, valid and quality data, it is useless in terms of informing health policy or strategic decisions.

Carl: What better place to find the definition of a standard than the [International Standardization Organization](#) (ISO), an international, independent body that sets guidelines covering issues such as manufacturing to food safety, agriculture and health care:

> *A standard is a document that provides requirements, specifications, guidelines or characteristics that can be used consistently to ensure that materials, products, processes and services are fit for their purpose.*

This basically means a description/set of rules that describe how something should be done and respond to inputs. Health interoperability standards help to govern the syntactic and semantic exchange of health data within a health information system. (As a reminder, syntactic exchange deals with the data itself and semantic exchange deals with the description of data.) As Amanda touched on, standards help technical teams better support interoperability and a more component-based architecture, which is something we promote for HIS. This often requires additions and extensions to meet new use cases.

As Amanda touched on, standards help technical teams better support interoperability and a more component-based architecture, which is something we promote for HIS.

In the health field there are standards that help guide HIS tools to be interoperable. The selection of the appropriate set to which your system subscribes should be driven by the need and environment. One example of standards in use in health care is [Integrating the Healthcare Enterprise](#) (IHE), an initiative by healthcare professionals and industry to improve the way computer systems in healthcare share information. IHE promotes the coordinated use of established standards to address specific clinical needs in support of optimal patient care, such as DICOM – a standard for handling, storing, printing, and transmitting information in medical imaging – and HL7 – a set of international standards for transfer of clinical and administrative data between software applications used by various healthcare providers. These standards and specifications are grouped as profiles that allow software vendors to claim adherence to and to be able to perform a defined function in a specified way.

For example, Care Services Discovery, or CSD, a profile used in mHero, specifies how a health worker, facility, organization or service data can be synchronized and linked within a central data store. This linked data can then be queried in a standardized way.

Other IHE standards include:

- PIX/PDQ - to support patient lookup queries
- XDS.b - to submit and query clinical documents
- mACM - to support alerts within health systems
- ADX: Aggregate data exchange - to exchange aggregate health data
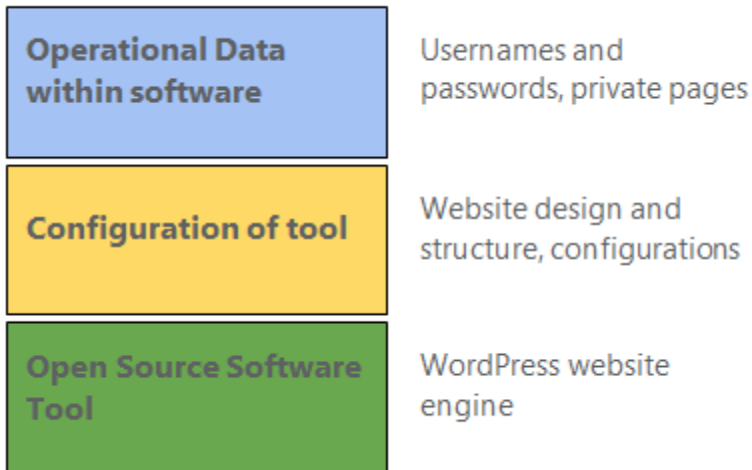- ATNA - For securing transaction and providing an audit log of transaction that have transpired

All these standards and interoperability profiles are important for software developers to be knowledge about since compliance to recognized standards supports a multi-software approach for health information systems development and deployment. They also allow for fit-for-purpose solution design that can tailor tools to the environment and needs. Most important of all standards allow data to be shared.

As a proverb states: "if you want to go fast, go alone; but if you want to go far, go together." Health isn't something that we want to just get done, it is something that we want to get done right.

Dykki: Data Security and Privacy are one of the Principles for Digital Development. What are some of the challenges in data security and how are they being mitigated? How do Open Source systems impact on security?

Carl: This is one of the topics that we have seen arise time and again; the confusion or assumption that open source software means open data (i.e. data is available to anyone). It is important to differentiate between the software and the implementation configuration and data within the software and implementation. Each of those three levels can be managed separately.

Figure 2: Figure: implementation and solution breakdown and examples

Operational Data within software	Usernames and passwords, private pages
Configuration of tool	Website design and structure, configurations
Open Source Software Tool	WordPress website engine

As can be seen in **Figure 2**, working on an open source tool doesn't automatically force data to be open. As technologists we need to manage this understanding with implementers. Can we imagine a financial institution opening its data? So to debunk the concept that open source implies open data it is important to address data security in information systems and particularly in systems that exchange information.

In health systems that exchange data, such as mHero, the question of security and privacy in both system installation and data exchange does come up. A general challenge faced in any information system installation is that the base installation may not be secure, i.e. easy to guess passwords, operating systems and servers not secured with firewalls or a lack of software updates. These are challenges for both open source and closed source systems and are addressed by following good practices in system installation and setup.

Data exchange and privacy and security raise an interesting set of questions and challenges (not uncommon in other industries). These include "how do I know who I'm sending my data to?", "can anyone intercept my data?", "who is exchanging what data?", "where is this data coming from?" Many of these are grouped into the areas of "access control" and "auditing."

mHero leverages the OpenHIE's architectural model of an information mediator (in this case the OpenHIM). In this model we see a middleware component playing the role of access control, audit log and router. Here we see systems connecting to a trusted agent (the middleware) and sending data to secure end-points. This pattern allows for a central point of control and view into the system, while it can be argued that it also introduces a central point of failure, implementation needs to mitigate risks on either side. A well instantiated middleware should have redundancies and failovers to mitigate point-of-failure concerns; and by its nature provide the implementers a controlled interface to manage access to data in and out of the system.

Amanda: Carl said it best: Open source systems do not mean open data. Systems are quite sophisticated these days and allow the granting of permissions for data access. For software like iHRIS, someone may have permission to enter data but not run reports. All of this is very customizable and adaptable. Deciding whom has access to what is a governance process that should be included in HIS standard operating procedures.

I have also heard concerns about data living "in the cloud." People ask if data are truly safe and secure when hosted in off-site servers. Let's be honest, the best way to be 100% sure your data is safe is to lock it in a box and throw away the key. Cloud-based data storage is more secure than locally stored data because of complex security methods. Yes, data in the cloud is hackable but it is extremely difficult. Would-be hackers would need to break through sophisticated encryption methods - something that takes a lot of time, a huge amount of computer processing power and very sophisticated forensic software. It is also important to note that while the cloud may be gaining traction in many areas, in low-resource settings, it does need to be weighed against the country's constitutional law or internal policies about personal health care data leaving the geographic bounds of the country. Many a project has run afoul of the assumption that the tool can "just live in the cloud."

Dykki: What does the future look like for health information systems? Where are there trends, opportunities and emerging tools? What needs to be done to harness the work done in health information systems?

Amanda: Health information systems will only grow more robust, sophisticated and interoperable. I think we'll see more dashboards linking systems so that data will be easily retrieved and analyzed from one place. I also see mobile devices playing a much more prominent role in health information systems. Already, information is gathered using mobile devices but I foresee an increase in the utility of these devices to analyze and report data. This will facilitate more real-time data collection and analysis.

We need to be ready for the future, and in order to keep up with the changing technology, we need to have people with the right skills in the right places to support not only the technology development but also the process and implementation of health information systems. Capacity needs to be built globally for strategic systems thinking and design, as well as to address changing governance and leadership challenges.

> **We need to be ready for the future, and in order to keep up with the changing technology, we need to have people with the right skills in the right places to support not only the technology development but also the process and implementation of health information systems.**

Carl: I want to echo Amanda's point on capacity development here as a great place to start. ICT, information and communication technologies, is becoming increasingly part of our lives and hence more are going to start looking for the ICT in the health care they are receiving. Technically we are seeing the growth of the Mobile First approach to user facing systems. The open source state of the Android operating system is definitely seeing this as a major player in the health arena. We are also seeing the service based architectures coming out of the "app" age of the internet. This pattern leverages the principles of component reuse and data exchange between services to support a more complex work-flow and pairs with our previous conversation around interoperability in healthcare. We need to ensure that we are focused on building capacity to engage with these technical trends and design patterns in the countries and areas where we are deploying systems.

We have seen tools built to support sharing codes, like GitHub, and communities arising to share knowledge, community driven knowledge, through mailing lists, wiki's and discussion forums. These patterns are being adopted in the development of HIS software and particularly in open source software communities. We want to encourage HIS developers to leverage these tools and technologies to better support the development of their tools as well as suggesting the use of freely available code checkers and integration testing tools such as travis-ci to utilize the best practices in tool development and safe delivery of software.

From a technical aspect, the use of new and innovative tools and technologies paired with well proven and stable approaches and design principles will go a long way to deliver well-structured tools that are scalable and interoperable. It is important to remember that what is new today is old tomorrow. Some of the best things we, as technologists, can do is to build capacity and skills with local teams to empower them to continue the work and respond to the ever changing environment of real world and technical approaches.

Software tools and solutions will come and go; the greatest assets we have are the people we have built up to address those challenges.
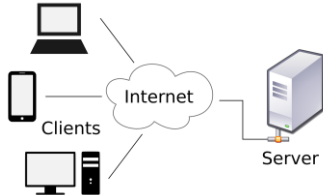
Dykki: Thank you both very much for this excellent discussion of some of the complex technology topics that underlie much of the innovation in health data systems today. If we can continue to unpack these concepts for the many different kinds of people who need to work together to implement these systems - stakeholders, developers, users - it will make their communication and collaboration much easier. The more we share a common understanding, the more successfully we can together improve the availability, quality and use of data and technology for better health.
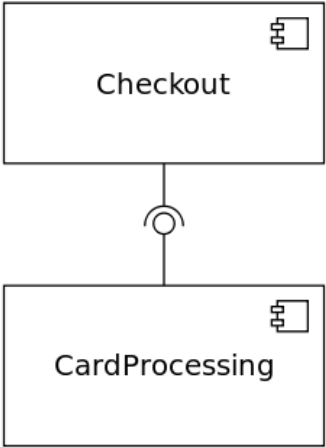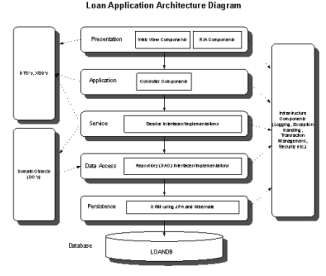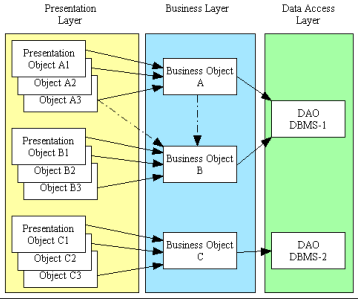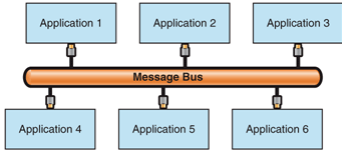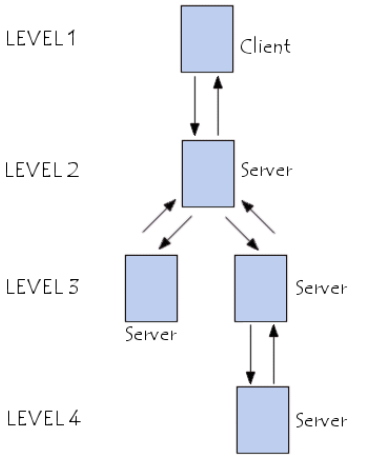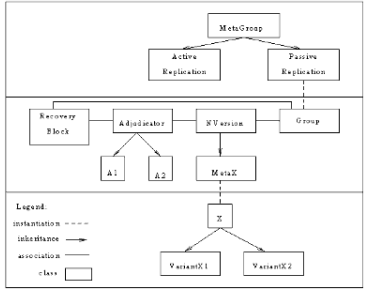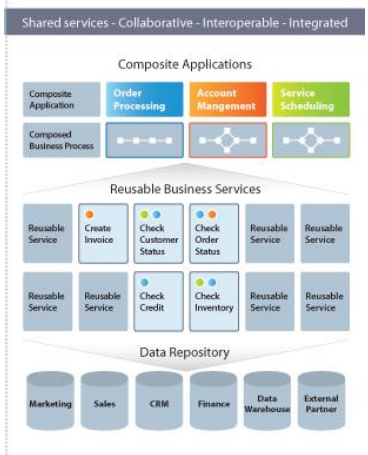
Appendix A

Appendix A has been adapted from the following document:

Microsoft Corporation. October, 2009. Summary of Key Architectural Styles in *Microsoft Application Architecture Guide, 2nd Edition*. https://msdn.microsoft.com/en-us/library/ee658117.aspx#KeyArchitectureStyles (Accessed June 2, 2016).

Return to document.

Architecture style	Description	Illustrative diagrams
Client/Server	Segregates the system into two applications, where the client makes requests to the server. In many cases, the server is a database with application logic represented as stored procedures.	

Component-Based Architecture	Decomposes application design into reusable functional or logical components that expose well-defined communication interfaces.	
Domain Driven Design	An object-oriented architectural style focused on modeling a business domain and defining business objects based on entities within the business domain.	
Layered Architecture	Partitions the concerns of the application into stacked groups (layers).	
Message Bus	An architecture style that prescribes use of a software system that can receive and send messages using one or more communication channels, so that applications can interact without needing to know specific details about each other.	

N-Tier / 3-Tier	Segregates functionality into separate segments in much the same way as the layered style, but with each segment being a tier located on a physically separate computer.	
Object-Oriented	A design paradigm based on division of responsibilities for an application or system into individual reusable and self-sufficient objects, each containing the data and the behavior relevant to the object.	
Service-Oriented Architecture (SOA)	Refers to applications that expose and consume functionality as a service using contracts and messages.	