Responses to specific questions from Carl Dvorak – Epic Systems Corporation

Below are the questions for the panel:

1. How do you, or would you, define a "public" API (attributes and utility)?

   APIs (Application Programming Interfaces) specify a way for one software program to invoke the capabilities of another software program. The API specification includes the name of the API, the parameters to pass to it and the return value or actions one can expect from the API's invocation. An API typically also presumes some "business logic" is being executed beyond simple data retrieval method to access data from a database as these can be accommodated in a simpler manner such as an SQL query. An API allows the using party to benefit from the programming of the offering party beyond typical data access.

   A public API is therefore one of the many APIs within a software application that has been designated as being available for use outside the development team managing the source code and overall integrity and security of the core application. (e.g. being made available to the "public" or to the "outside") A private API or system API is often the name for an API that the developer chooses not to expose to the public.

   For a public API to be most useful, "interface stability", also occasionally referred to as the "interface contract", should be clearly defined and maintained through time to the extent practical. This means that the name of the API, the parameters being passed to the API and the return value from the API should remain stable through time and should they need to change, a pre-agreed upon communication method should be used to notify those who might be using the public API that they need to change their use of that API and the timing requirements for such a change.

   An API in one software application may be accessed by other software applications through a variety of methods which range from program code written on the same computer in the same programming environment calling the API in the simplest case to a program on an entirely different computer, written in an entirely different programming environment calling the API through a layer of standards based management of such "remote procedure calls – aka RPCs". Today, these remote calls to an API are often associated with Web Services – SOAP or RESTful for example. Historical models of such calls include CORBA (Common Object Request Broker Architecture).

   Regarding utility, public APIs are not new to healthcare and in fact form the basis of billions of transactions each year from EHR vendors to services such as ePrescribing networks like SureScripts, user authentication services such as LDAP (local directory access protocol), CCOW (clinical context object working group) and others including CCDA exchanges with both Direct and Connect protocols.

The utility of public APIs is greatly enhanced when they conform to industry standards. For example, IHE (Integrating the Healthcare Enterprise) standards for exchanging information via APIs. This allows vendors and other innovators to create products or services that use these APIs or support these APIs within their own products with some assurance that they will last through time and will not require constant re-engineering to maintain working products.

2. Have you deployed, are developing, or are planning any of these types of API's and for what purpose (e.g., CCDA, basic MU content, PACS, medications, referrals, billing etc)?

Yes, and as described above in question 1, Epic utilizes public APIs and provides public APIs for all of what you describe and more. In addition to what you describe, we provide public APIs for biometric security devices such as finger print readers, palm scanners and iris scanners. We also provide public APIs for third party document imaging vendors to integrate their software closely with Epic software. We have had over a dozen such document imaging vendors providing products that use these APIs. And, many more use cases including consumer engagement and monitoring devices for example in conjunction with Apple and direct with device manufacturers such as FitBit and Withings.

Epic provides hundreds of public APIs available through either a click through agreement or a simple non publication agreement as described below for both read and write use. In addition, we expose thousands of APIs to our customers for their internal development to extend or customize our system to meet their specific desires. In fact, we provide our source code to customers exposing all of our internal APIs for their use in development. This does not allow them to re-disclose or commercialize these APIs, but it does specifically allow them to extend or adapt our system to meet their needs.

3. Do these API's affect push or pull functionality?

These APIs typically do both write (push) and read (pull) functions.

4. Are your API's bidirectional?

Yes, most of these APIs are bidirectional – they provide us some information (here is a scanned document to associate with the currently selected patient) and we provide them with some information (here is the internal Epic reference and provenance to associate with this document for your use).

5. What type of business agreements were developed prior to initiating the build?

For public APIs, the business agreement is typically very small and simple. Often this might be a “click through” agreement. Or, for some situations, a simple agreement to not further distribute or published without any requirements for confidentiality.

In some cases, if we choose to expose an internal or private API to a partner who we expect to do some companion development, but are not interested in supporting that API to a broader public audience, we often agree on a non-disclosure and restricted use requirement in the agreement. That agreement would also typically clearly identify the responsibilities of each party to manage changes to that API over time.

In addition, many of the third parties who provide APIs we use at Epic require proprietary treatment of their APIs and specifically prohibit Epic from disclosing their APIs to a vendor who competes with them. We therefore have to program to two different APIs for those two vendors and switch between them based on the customers selection of that third party product.

6. If you have an API, how do you manage patient identification across entities?

In most cases public APIs assume that patient and user context are already established and being passed as parameter to these APIs for patient identification and user access auditing.

In situations where an API to establish patient identity between computer systems is needed, there is a specialized set of APIs that allow for a “patient lookup” to occur between the computer systems.

These APIs are most often associated with a Master Patient Index API set and IHE provides a standard for resolving patient identity through matching and selection algorithms.

7. Does your API affect data extraction from discreet fields and the placing of that data into similar discreet fields in the receiving application?

Our APIs certainly extract data from discrete fields within our application database and affect placement of outside data directly into discrete fields after appropriate validation through business logic. We have no direct control over what another party does with the data once we provide it.

8. Does your API affect specific available data to be actively selected by the sender? By the requester?

As previously stated, we have hundreds of public APIs. Some of these are generic data query APIs that allow the calling program to send us parameters selecting what they would like to have us send them or how they would like us to handle the data they supply to us.

Generally, these types of generic data query APIs are quite complicated to manage in that an EHR is required to maintain role-based access to patient protected health information and has responsibilities for accounting for disclosures. Most health systems would apply great care in managing such APIs to ensure they are not later prosecuted for inappropriate PHI disclosures.

If time permits, please consider answering as many of the additional general questions below.

- Given currently implemented information technology (IT) architectures and enterprises, what challenges will the industry face with respect to transitioning to a JASON like architecture? What challenges will your organization face?

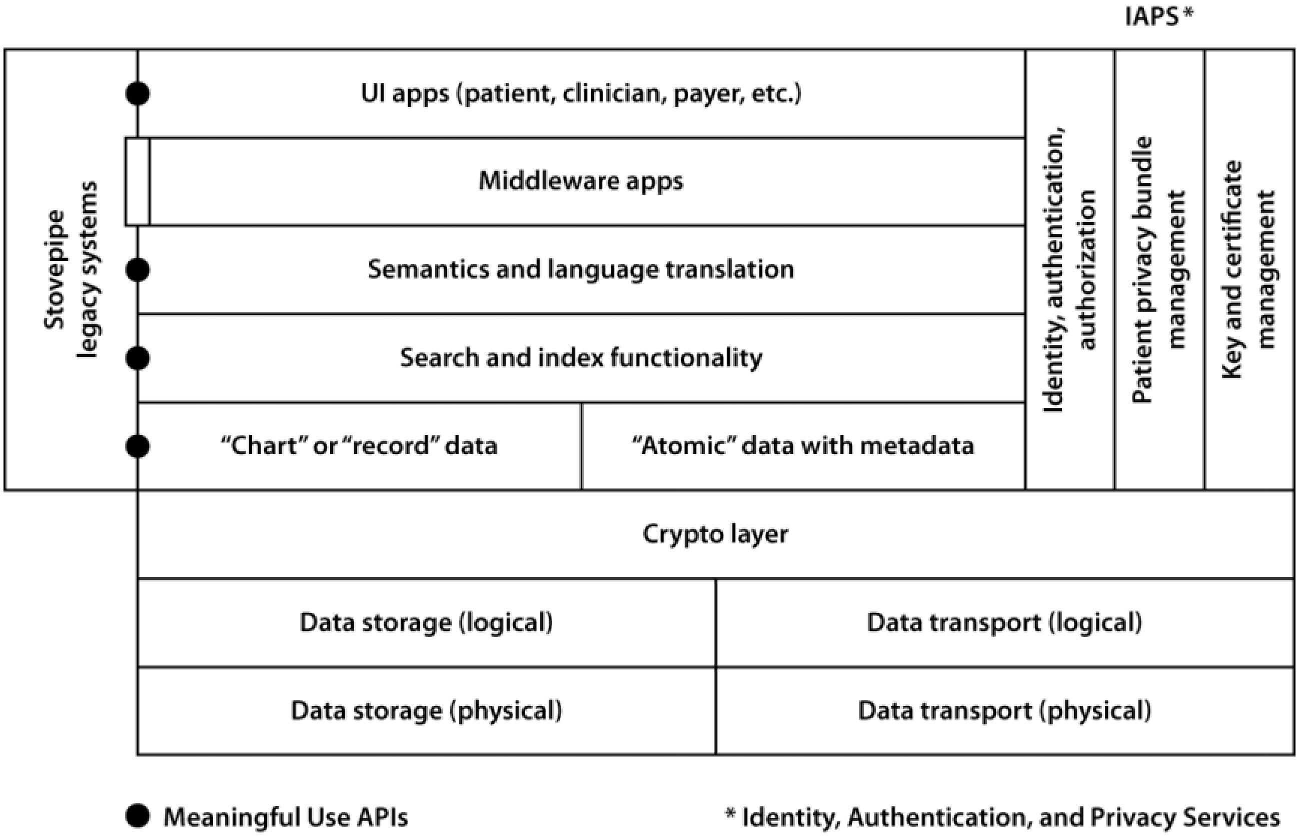

IAPS*
UI apps (patient, clinician, payer, etc.)
Middleware apps
Semantics and language translation
Search and index functionality
"Chart" or "record" data
"Atomic" data with metadata
Stovepipe legacy systems
Identity, authentication, authorization
Patient privacy bundle management
Key and certificate management
Crypto layer
Data storage (logical)
Data transport (logical)
Data storage (physical)
Data transport (physical)
● Meaningful Use APIs
* Identity, Authentication, and Privacy Services

Figure 1. JASON's proposed software architecture for the exchange of health information.

The JASON report showed a misunderstanding of modern computing architectures already employed by most major EHR vendors including Epic. Most EHR vendors already provide a plethora of APIs and could easily activate thousands more APIs of all manners should they choose to.

This isn't a technical challenge in any way for Epic and unlikely to be a technical challenge to any other EHR vendor either.

It is however a challenge of definition, semantics, efficacy and intention. Vocabulary and context are significant challenges for the entire industry. Applying the "big data" techniques of advertising, consumer sentiment, espionage or other fields of interest would lead to the generation of significant misinformation in healthcare when contemplating the very specific needs and care of an individual. What information is "good enough" to show you an advertisement for the latest microwave oven on the market is not sufficient to advise a patient on which chemotherapy regimen they should undergo at this point and likely will never be sufficient unless context is completely understood and factored in to the equation. The patient preference would need to also be fully considered.

The notion of “Atomic data with metadata” is an oxymoron. Data is either atomic (of or forming a single irreducible unit) or it has context (metadata). In healthcare, all data has context. And, the more context the better when it comes to understanding and interpreting that data.

Provenance (of a piece of data) is simply an arbitrary point on the continuum of context.

How much context or provenance you desire is dependent on what you want to do with the data. What context is appropriate is almost always in the mind of the consumer of the data.

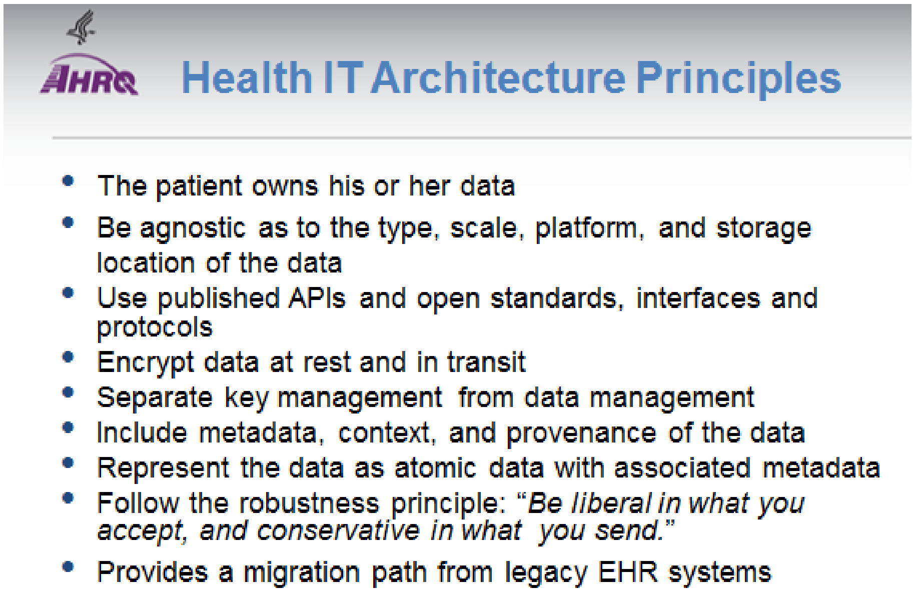
- Do you see an evolutionary path for the industry to move from currently implemented approaches to a JASON like architecture?

Most of the technical capabilities JASON described are already available in today’s EHRs. What is missing is the policy aspects related to the use of these capabilities.

Must a health system supply all its patient data to a Google or Microsoft search engine, where patient specific information can be used for purposes other than direct healthcare, in order to be considered compliant? This is an important policy question for our future.

The notion that JASON provides a migration path from legacy EHR systems is odd. The assumption that a migration path is needed is based on a false assumption that EHR vendors, including Epic as well as our many competitors, do not continue to invest heavily in advancing the art and science of healthcare informatics.

Articulating that AHRQ and ONC have a principle to “Provide a migration path from legacy EHR systems” (last bullet on slide below) demonstrates a lack of understanding and respect that EHR vendors share a common mission to improve patient care and engagement through technology.



Health IT Architecture Principles

- The patient owns his or her data
- Be agnostic as to the type, scale, platform, and storage location of the data
- Use published APIs and open standards, interfaces and protocols
- Encrypt data at rest and in transit
- Separate key management from data management
- Include metadata, context, and provenance of the data
- Represent the data as atomic data with associated metadata
- Follow the robustness principle: "*Be liberal in what you accept, and conservative in what you send.*"
- Provides a migration path from legacy EHR systems

- What policy and technology developments would be necessary to assure the privacy and security of information in a JASON like architecture?

The most important development would be a new model of educating patients, providers and society in general about secondary use of data that currently drive the JASON and PCAST approach as described in the report and creating a national conversation about the use of personal health information would be necessary.

As far as technical developments, most EHR systems already support this type of access although in rationally limited ways given today's regulatory and legal climates.

Given that many patients wouldn't fully appreciate or understand the implications of secondary use of their data, special care must be given before mandating healthcare providers make their patient's data publically available to secondary uses beyond their understanding. Physicians need to help patients understand how the data they collect about them (under the perceived protection of the patient provider relationship) will be used.

- What existing efforts (standards, initiatives, pilots etc.) in the marketplace are advancing a JASON like infrastructure?

A standards-based initiative like FHIR (HL7's Fast Healthcare Interoperability Resources) provides a great platform for this type of work and is actively being supported by Epic as well as other vendors.

- A key recommendation of the JASON Report is that EHR vendors should be required to develop and publish APIs for medical records data, searching and indexing, semantic harmonization and vocabulary translation, and user interface applications. What existing efforts are underway in health care that could inform the implementation of this recommendation?

Most EHRs already support such APIs so this is more a discussion about mandating the use of these by search and indexing engines as they exist today. In addition, most EHR vendors provide SQL accessible normalized data that could already easily be provided to search and indexing companies exposing the protected health data of almost all Americans.

Initiatives to standardize such APIs by groups like IHE or SMART are reasonable efforts that most EHR vendors already recognize as useful industry initiatives and are making their R&D investments to support them in proportion to their understanding of their customers' need.

This is an area where ONC should continue to promote standards development and thoughtful requirements around use of such tools and technologies. The FHIR standards are reasonable standards to support.

ONC should not, however, serve the special interests of those who wish to commercialize use of protected health information and should instead focus on the needs of improving the health and healthcare of Americans.

- What standards, implementation specifications, certification criteria, and certification processes for electronic health record (EHR) technology and other HIT would be required to implement the JASON reports' recommendation that ONC require open published APIs through Stage 3 of Meaningful Use?

Because the JASON suggestions have a dramatic impact on patient privacy and patient-provider relationships, we would suggest that the concepts in the JASON report are not appropriate for inclusion in Stage 3 Meaningful Use and likely not appropriate for any following stage of Meaningful Use should one be proposed.

- What processes and approaches would facilitate the rapid development and use of these standards, implementation specifications, certification criteria and certification processes?

Market innovation has historically benefited from the light touch of government followed by an era of industry innovation. Thoughtful and restrained involvement of ONC and CMS will likely be the most appropriate role for the coming decade after such an unprecedented intervention (HITECH/MU).

We have an opportunity to let Meaningful Use serve as a spring board for the future. The future evolution of these standards is best left to market forces that will exist in a post Meaningful Use era. We will be at a turning point in the coming years where we could determine that the HITECH Stimulus program created either an infrastructure for market innovation or, conversely, a government-mandated forced march through technology for the future.

- How might ONC and other Federal agencies best integrate the changes envisioned by the JASON report into their future work?

Some aspects of what JASON postulates would be helpful, but much of what JASON suggests seems to put technology and secondary use to commercialize protected health information ahead of a thoughtful national policy relating to patient privacy and the sanctity of the patient-provider relationship. Creating suspicion and a culture of fear of how your personal data might be used would inhibit our very important national goals relating to healthcare.

A thoughtful and considered approach of supporting those technical concepts suggested by the JASON report would be the most appropriate course of action by ONC and CMS. I would suggest engaging patient advocacy, patient privacy and physician associations in a national dialogue on this topic before rushing to a technical requirement that might create widespread distrust of our nation's healthcare system by patients and providers alike.

- What actions would you recommend ONC take to help the industry advance towards a JASON like architecture that supports interoperability for primary and secondary uses of health information?

Focus on interoperability of the primary uses of healthcare data – treating patients to improve health and healthcare.

With regard to secondary use of protected health information, focus on policy issues relating to HIPAA, consumer expectations, commercialization, privacy, etc. at a national level.

Mandating technology solutions that would compromise protected health information should not be considered lightly. We have many more important issues on the immediate horizon to consider.

This is a topic that requires policy consideration and study beyond what the technical experts might suggest. Although JASON represents some truly gifted technical thought leaders, technology alone will not solve our nation's healthcare problems. We need a solution that increases the quality of patient care while increasing the role of the patient in informed decision making related to their care and fostering the role of primary care in preventing and managing chronic illness and facilitating thoughtful choices relating to end of life care.