

JASON Task Force  
HIT Policy Committee  
US Dept Health and Human Services

This informal submission is provided in response to an approach to have a representative of the openEHR Foundation take part in public hearings relating to the JASON report<sup>1</sup> produced for HHS.

What follows are two sets of notes: firstly some comments on the JASON report, and secondly, responses to a set of specific questions received about how a 'JASON-like architecture' for could be achieved in the US.

This submission is not considered private, and may be used by HHS as it sees fit.

Sincerely

Thomas Beale

CTO Ocean Informatics  
Lead, specifications program, openEHR Foundation

---

<sup>1</sup> [http://healthit.gov/sites/default/files/ptp13-700hhs\\_white.pdf](http://healthit.gov/sites/default/files/ptp13-700hhs_white.pdf)

## Notes on the JASON Report

This report highlights some of the key requirements, and also key impediments to achieving shareable health data. We agree with the general concept, and many of the proposed ideas. However, it also appears to take a precipitative position on a 'solution' (e.g. section 1.4 is already talking about 'common markup' and 'APIs') without having covered a number of questions on the problem side that need to be addressed.

Since the report is being used to frame discussions and decision within the ONC on this topic, it may be worth pointing out what we consider to be some of the missing elements of both the problem and solution picture.

Our comments are not meant to be critical of either the report, nor of the current ONC activities. On the contrary, both are to be lauded. The very fact that a large scale interoperable Health IT infrastructure remains elusive in all countries today means that revisiting the question is entirely appropriate. We hope that our inputs here are taken in the spirit that they are meant, which is essentially this: that as much intellectual progress should be made on developing a common understanding of the problem, and the general shape of a solution approach, as possible, prior to any 'green light' policy announcements and large-scale expenditures.

### What kind of problem is this?

The first point we would make is that the report has conceived of the problem as a software engineering one, one that can be solved with a standard software architecture of some kind.

Although the lack of a standard architecture of some kind is clearly important at some level, there are various difficulties with using it as a way of framing the problem.

### Technical Aspects

Firstly, we have to ask the question: what software architecture are we talking about? That of a vendor product? Of some new product that is being proposed? Or perhaps of an infra-structure service like Google, Microsoft Cloud, or similar? We have to assume that in most cases, the 'software architecture' of any product is the business of its developer, typically a vendor, or sometimes an organisation like the VA. So 'software architecture' in the normal sense of the term is not likely to be something that can be prescribed centrally.

Secondly, we are talking about health data. There is an **inherent level of complexity** in clinical processes, workflows and information gathering and use that means that there are no simple answers, as implied by the statement "[search functions could pull data from the legacy systems and index those data so that they are more amenable to general queries](#)" (p4) or talk of a "[federated database](#)" (p5). Indeed, this is one of the problems in health IT – why can't data easily be transferred / transformed in this way? Over the last 20 years, we have discovered it is due to the complex semantics of clinical process and data (due in turn to the complexity of human workflows and ultimately biology itself).

Large coding systems such as LOINC, SNOMED CT and many others have been developed to deal with just this problem, and have not yet solved it. Another problem is: what should a common information model look like? Unfortunately, health data are not like airline bookings. Consider that a GP or admissions application has to deal with patients with injuries/trauma, infectious disease, chronic or acute problems in any body system, genetic disorders, depression, schizophrenia, childbirth, dozens of types of cancer, elective surgery, alcoholism and drug abuse, .... The list is endless. And that ignores different levels and types of clinical note-taking, reporting, levels of detail

according to specialisation, patient involvement in clinical trials and every other activity that can occur within a health provider establishment. Not to mention genomics, and the complex chemistry of drugs and their interactions, public health data requirements and so on.

One outcome of the last 15 years or so in health informatics research is that **no single information model will deal with the diversity**, indeed, an architecture predicated on that idea isn't viable - trying to create such a model would end up in a huge, incomprehensible and unmaintainable mess.

A further impediment to defining health information semantics in software models and database schemas is the **rate of change in health, and its impact on health data**. Every one of the following sources of real world change has to be reflected in models health data if they are to stay up to date:

- Medical / scientific research – creating new types of tests and treatments
- Clinical research – new and changing protocols for managing different types of patients
- Business process – constantly changes to workflows, e.g. in A&E
- Legislative changes – altering what data need to be collected and when

If health information semantics were to be built into software in the classic textbook way, none of these changes could be addressed in a sustainable or clinically safe way – given that real provider systems contain peta-bytes of complex data at any point in time.

An alternative approach is needed, in which the **semantics of health information are defined primarily outside of the software**. This is partly addressed by the use of coded terminologies. However, other elements are needed, including 'detailed clinical models' (DCMs), and computerised guidelines, care pathways and checklists.

### Social and Economic Aspects

However, these technical difficulties are not the only reason that health IT interoperability is so elusive.

A sociological reason is that in a health system heavily oriented to hospital care, **sharing of health data across providers** (as opposed to inside provider institutions) **has not been a priority** for the sector as a whole. Health IT investment and procurement has been heavily oriented towards comprehensive solutions from large vendors, that actively compete by locking institutions in to proprietary data formats and interfaces, with standardisation being largely limited to lab results, radiology and (sometimes) prescriptions.

Other sectoral factors - particularly re-imbursement models, and the wrong application of fee-for-service and fee-for-outcome business models, as described in detail by Clayton Christensen in 'The Innovator's Prescription' - predispose providers against sharing data. Only providers with comprehensive control over all/most care settings and large numbers of clinics, such as Kaiser Permanente, have been able to take a different path.

Many analysts believe that the current system, which consumes around 18% of GDP compared with 9% in the best-served European countries isn't sustainable in the long term.

Christensen and others see a role for 'shared EHRs' in addressing the problem, although clearly there are deep structural economic problems in the sector that need to be addressed as well.

### Can the problem be framed in a better way?

The JASON report presents findings (p6) which we agree with, but Recommendations (p7) which we think could be improved by a reframing of the solution in terms not of a 'software architecture', but

as an **open platform definition**. This term not only reflects some concrete differences with the 'software architecture' concept, it may prove to be a more acceptable term in the marketplace, since vendors and developers are likely to see software architecture as their own business.

There are two levels at which the platform concept should be understood – the technical and the economic.

Technically speaking, a **platform can be defined as follows**:

- the '**computational**' view – what functions & procedures can be called in the interface? E.g. `create_patient`, `update_patient`
- the '**informational**' view – functions return data, and both functions and procedures take data as arguments. For the API to be fully specified, these arguments and returns need to be specified too
- a **protocol specification** – what order API calls need to be made, how exceptions are reported and so on.

This translates partly to a software architecture – although not the primary software architecture of a vendor product – but also an information and standards - what we may call a **content architecture**. The latter corresponds to what is **outside of the software**, which is *most clinical process and informational semantics*.

A further critical element is a **model of querying**. Ideally this should be based on models of content, as we have done in openEHR with Archetype Query Language (AQL)<sup>2</sup>, not on physical DB schemas.

Basic questions now need to be considered:

- Who defines the platform?
- Who implements it?
- How does it deal with ongoing change and complexity?
- Who maintains it?

Some of these questions are political – i.e. questions of responsibility and ownership. At this point we need to consider who the players are in the sector, in order to get an idea of who might do what.

An informal model of platform-related stakeholders is shown below.

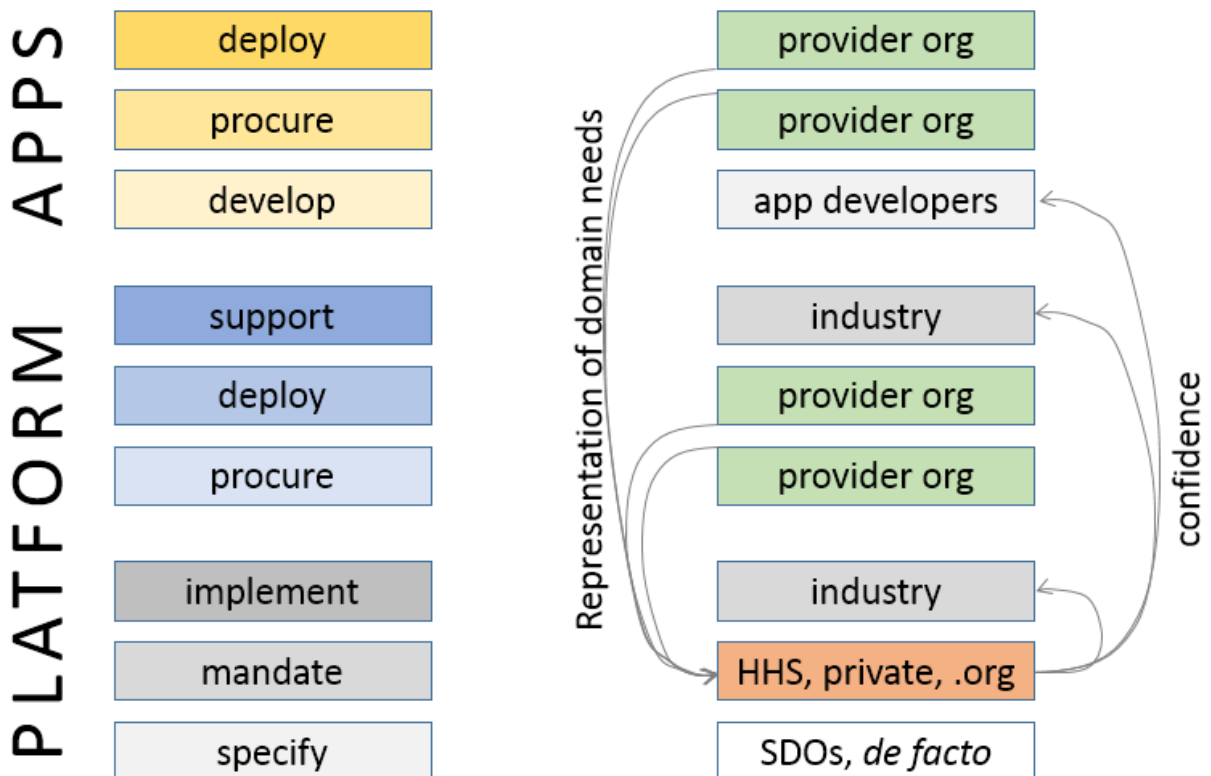
---

<sup>2</sup> <http://www.openehr.org/wiki/display/spec/Archetype+Query+Language+Description>



|          |           |  |
|----------|-----------|--|
| APPS     | deploy    | runs the apps  |
|          | procure   | buys the apps  |
|          | develop   | develops apps that talk to the platform              |
| PLATFORM | support   | support deployed system                              |
|          | deploy    | actually deploys the platform implementation         |
|          | procure   | pays for platform production deployments             |
|          | implement | implements the platform (i.e. back-end(s))           |
|          | mandate   | mandates the platform specification in the ecosystem |
|          | specify   | defines and publishes the platform specification     |

Mapped to the health IT sector, this might look like the following:



These diagrams are taken from the author's own blog<sup>3</sup>, and are in no way meant to be prescriptive, or even heavily theoretically based. The intention here is to simply point out the plurality of players, and that an ONC strategy based around an open platform needs to take develop a picture such as

<sup>3</sup> <http://wolandscat.net/2014/05/07/what-is-an-open-platform/>

this, and determine the roles in the platform ecosystem. The JASON report recognises the problem of who does what as the 'turf problem' (item 2, p15).

One of the key conclusions we can draw (see blog article for details) is that, with the right assignment of responsibilities, the platform concept can fulfil an economic purpose, which is to **open up the health ICT economy**. It does this by:

- Enabling small, innovator companies to take part, since they can rely on the mandation of a platform
- Encouraging big incumbents to implement the platform and enable data sharing
- Enabling provider organisations to take control of procurement by making published interfaces, information models, terminologies and protocols contractual obligations.
- Enabling providers to procure from multiple sources, and rely on certification to know that the components / products will interoperate.

This concept is recognised in the JASON report (recommendations item 3.)

Some larger corporations are already seeing a platform-based future, as evidenced by the Health Services Platform Consortium<sup>4</sup> (lead by Harris, Cerner, Intermountain) and the Commonwell Alliance<sup>5</sup>.

Some of our other key conclusions with respect to the platform concept are:

- The technical paradigm and definition of the platform is critical to success; wrong choices will cause failure
- The platform is a process, not a product (i.e. that can be finished and published) – the required concept is a 'standards factory'
- Ontologies and terminologies are needed in the platform to ensure the reliable and clinically safe computability of data
- Developer usability will greatly influence adoption.

Obviously much more could be said than we have space for here. However we would suggest that some of the recommendations can be improved by recasting them (and adding others) within an open platform paradigm, rather than the more limited 'software architecture' notion.

Some recommendations such as 2.2 are unlikely to be realisable without some centralised / collaborative activity that works to determine the 'small set of necessary interfaces between functions' (p7). This is a non-trivial activity, and is not something that can be 'done' and 'published' – it will be a process which has to publish a series of definitions over time.

Other recommendations such as 3.1 could be improved by considering the platform concept. Currently this recommendation says: "EHR software vendors should be required to develop and publish APIs for medical records data, search and indexing, semantic harmonization and vocabulary translation, and user interface applications" (p7). This is unlikely to realise the intended goal, because what is required is not 'APIs' but 'standardised APIs'. Given the inherent complexity of APIs, particularly the information dimension, these two are completely different things. Instead, the APIs need to be defined by a more central player, and possibly *implemented* by vendors, but more likely, to be implemented in an open source community, and deployed by vendors with a minimum of integration work on their side.

---

<sup>4</sup> <http://healthcaresoa.org/>

<sup>5</sup> <http://www.commonwellalliance.org/>

Detailed recommendations such as item 6 (p 28) which says “EHRs should be represented as a collection of atomic data items and associated metadata.... The atomic data elements can be reassembled in various ways” tend to hide complexity. Experience over the last decades shows that ‘atomised’ i.e. fully structured data are preferable, but the details of how this are done are not as simple as implied here, particularly the idea that data can easily be re-assembled for the purposes of different users. In fact, this is a complex technical problem relation to information models, content models and terminology.

## Comments on Key Technical Issues

### APIs and the Hidden Complexity of Data

One of the seductive ideas about APIs is that they seem simple to define, and as such, an easy route to interoperability. It is reasonable to suggest that the *functions* of an API might be relatively simple, e.g. a function definition like `get_medications (filter): Medications` seems easy enough to understand. However, the definition of ‘filter’ and ‘Medications’ may easily be complex, and is dependent on what a ‘medications list’ is understood to be. Does it include only active medications? Suspended medications? Medications that have been reconciled on admission or discharge? Is it a single source of truth medications list, or just one or more copies of lists held in different places?

The model may become quite complex. We might assume that the HL7 FHIR Medications resource definitions – there are five – might form the basis of a group of ‘medications’ API functions. However, it has to be assumed (and is assumed by FHIR) that local and specialist uses of these resources will involve ‘profiling’ and ‘extensions’. In reality, the number of variants of just the FHIR medications data (and filtering criteria) may be open-ended, even if the number of API functions is small and fixed.

Now if we then start to consider other clinical data, we can see that two approaches are likely. For certain key data items (problem lists, allergies, demographics, procedures etc), custom-defined information resources are likely to be defined, as indeed they are by the HL7 FHIR effort, in a similar way as the medication list concept.

For everything else, some kind of generic approach to defining the information is needed. A corresponding API function might be something like `get_record_item (criteria): RecordItem`. This simple function hides **potentially thousands of types of information** whose structure, coding and complexity are not knowable directly from the API definition.

In fact, **most of the semantics of health data are not in the API definition**. This tells us that while APIs are useful, even crucial, they are not the main entity in a platform that defines semantics. For this we would need to look at semantic health models, such as terminologies, so-called detailed clinical models (DCMs) and then higher order entities such as guidelines. In terms of DCMs, two well-known source of content models are available: openEHR.org archetypes<sup>6</sup> and the Intermountain Clinical Element Models (CEMs)<sup>7</sup>. A detailed description of these models is beyond the scope of this document, but for both the openEHR vendor and clinical community, and Intermountain Healthcare, the clinical models – which are completely separate from the software – are a foundational element of their health data platforms. Some government programmes now use archetypes: Australia, Scotland NHS, Norway, Slovenia, New Zealand, and Brazil.

---

<sup>6</sup> See e.g. <http://www.openehr.org/ckm/> and <http://dcm.nehta.org.au/ckm/>

<sup>7</sup> See <http://www.clinicalelement.com/>

The Clinical Information Modelling Initiative (CIMI) has been working toward an international standard version of the DCM concept, underpinned mainly by openEHR.org archetypes and Intermountain CEMs.

One of the important lessons from experience is that content models, guidelines and terminologies (particularly terminology value sets) are **defined primarily by clinical professionals**, i.e. people with professional knowledge of their area of clinical care (or health research). This is a major departure from IT-led developments in which the majority of a system's functionality and data are defined by technical people who have obtained a very imperfect idea of complex clinical reality by interviewing domain experts during product development.

The lesson here is that **an open platform requires not just APIs, but a sustainable way of defining clinical content** (and ultimately, process) formally. Additionally, an implementation approach in which APIs serving such content needs to be defined. HL7 FHIR provides some of the latter machinery.

### A Common Failure: Forgetting about how to get Data out – Querying

In the many national and other efforts to define e-health architectures or infrastructure, one of the most important aspects is invariably left out – querying. This is despite the fact that data are captured once but may be retrieved many times.

Querying is a complex subject, and we cannot go into a great deal of detail here. However we recommend one principle for consideration: the use of **content model-based querying**. openEHR created such an approach, known as the Archetype Query language (AQL), referred to above, in which queries can be developed solely on the basis of published models of content (archetypes) and terminology, i.e. **independently of physical DB schemas**. This is important for at least 2 reasons:

- Queries can be developed once, rather than for each implementation
- Decision support and Analytics applications that are query-dense start to become more economic, because their vendors can rely on published models to develop queries.

Most querying in openEHR production systems is done this way today, and there are user-friendly tools emerging in the openEHR community for authoring queries graphically.

### The Role of Standards

A key element of the open platform infrastructure is clearly 'standards'. The JASON report makes the following statement: **"The architecture must be based on open standards and published application program interfaces (APIs) and protocols"** (p26). This is a reasonable statement as far as it goes, but of course it depends on which standards we are talking about. Health IT standards cannot simply be pulled off the shelf and magically combined, as has been wrongly thought in the past, in some cases to very great cost (e.g. the UK<sup>8</sup>).

The work of creating a high-quality standardised platform definition cannot be under-estimated. Such a definition has to be:

- **self-consistent and coherent** – a mashup of off the shelf standards will not achieve the desired result;
- **sustainable** – the work of defining and maintaining it must be economically feasible and realisable without unreasonable human resource requirement;

---

<sup>8</sup> See National Audit Office report <http://www.nao.org.uk/report/the-national-programme-for-it-in-the-nhs-an-update-on-the-delivery-of-detailed-care-records-systems/>



- **scalable** – it must be able to scale to the massive diversity of real world clinical data and workflows to realise its full potential.

Our recommendation is to consider *standardisation* as a part of the platform definition work, and to carefully assess and use existing standards and specifications (or elements thereof) only where they are fit for purpose. Where no ‘standard’ appears to exist, develop one, i.e. identify what is missing and either build it (if it is a finite entity, e.g. a specification for dates and times in health) or create a sustainable ‘standards factory’ for more open-ended requirements (e.g. standardised library of clinically safe standard UI elements).

### Legacy Systems

The JASON report appears to assume some kind of system that would be populated by legacy data: “These APIs would allow the new architecture to be populated from the legacy systems” (p 29). This is probably a simplistic view, and a more likely one is that EHR / EMR and also HIE products may implement elements of the platform so as to connect to a national health data infrastructure.

### Conclusions

There are many details in the JASON report in which we have not commented here. For the purposes of this response, we re-iterate the basic suggestion, which is to:

- recast the ‘software architecture’ notion as an open platform concept;
- start to identify relevant platform-related stakeholders;
- understand the definition of the platform (including architectural elements) as an ongoing process rather than a finishable product;
- expect that the majority of what needs to be defined is not a software artefact, but will instead:
  - be formal models of content, workflow, process, queries or other domain semantics, that can be used to generate code and also be consumed by software at runtime;
  - be defined by domain professionals using tools, not IT workers.

## Responses to JASON Task Force Questions

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?

The underlying challenges for vendors and other owners of current systems will be the standard ones: mapping existing data and system interfaces to standardised data and interfaces. How much of a real challenge this is, and whether vendors will accept it depends entirely on getting the central strategy right. For example, creating a small number of core data and API specifications around e.g. 'medications list', 'problem list', 'allergies' and basic demographics could work, if done in the right way. The APIs should probably be developed in an open source fashion, and the formal content models need to be openly available. If the technical or semantic gaps are too great or costly, industry will not adopt.

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

There are certainly evolutionary paths available for industry. These have to be carefully determined, trialled and maintained. Previous national programmes have found to their cost that mandating certain kinds of across-the-board change were simply not viable.

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

Most aspects of security are just a case of doing normal security engineering. The presence of PKI and biometric identification would be useful.

Privacy is the more difficult subject, because it potentially relates to specific types of content (e.g. a patient's mental health notes) and requires a way of evaluating 'legitimate relationships' (of care – patient) among the possibly numerous and changing set of carers who might once / ever have had the right to see patient notes; with the added complication of emergency services and end of life access.

Over-complicated models of privacy most likely won't work, because they require comprehensible ways of setting and re-evaluating settings – potentially far more complex than the typical Android or iOS settings when you install a new app on your phone.

Therefore simpler models e.g. whereby a patient can simply move some category of data in and out of a 'sensitive' area of the EHR which can only be seen by specifically named healthcare professionals will probably be needed.

However, the most important issue is to establish what citizens actually see as the main risk:

- Access of their health data by a physician not treating them?
- Access of their data by insurance?
- Access of their data by employers?
- Access of their data by commercial enterprises, e.g. drug research companies?
- Access of their data by media?
- Access of their data by friends and family?

These potential 'threats' to privacy involve different motivations, mechanisms, and consequences. We could therefore suggest that the first task in the privacy policy space is to develop a statement of

the problem as the public (and others) really see it. This would become a cornerstone of Information Governance policy for health data.

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

Efforts that we know of that contain a general platform-oriented vision and have implemented some or most aspects of such a vision:

#### White-paper

- Microsoft Connected Health Framework – a few years old now, but a comprehensive piece of work that gives a good idea of the sophistication required in a realistic platform - <http://www.microsoft.com/health/ww/ict/Pages/Connected-Health-Framework.aspx>

#### Relevant standards- / specifications:

- HL7 / OMG HSPS specifications - <http://hssp.wikispaces.com/>
- HL7 FHIR - <http://www.hl7.org/implement/standards/fhir/>
- IHE – <http://www.ihe.net/>
- IHTSDO – <http://ihtsdo.org> – terminology development SDO
- LOINC – <http://loinc.org> – terminology for lab observations
- WHO, WONCA etc – other terminology issuers
- openEHR specifications - <http://www.openehr.org/programs/specification/releases/currentbaseline> - including the Archetype Definition Language (used by openEHR and CIMI, and an ISO standard)
- Archetype Modelling Language (AML) RfP about to go into OMG - <https://github.com/opencimi/AML> ; this work was started as an offshoot of CIMI, and when a standard emerges, would result in mainstream UML tools being able to do archotyping.

#### Implementation-based:

- Harvard SMART - <http://smartplatforms.org/> - now working in/with HL7 FHIR
- HL7 FHIR - <http://www.hl7.org/implement/standards/fhir/> - open source implementation of core FHIR specifications
- Clinical Information Modelling Initiative (CIMI) - <http://opencimi.org/> - an effort to find an international common content modelling approach, largely based on approaches developed (independently) by Intermountain Healthcare and openEHR.
- Intermountain Healthcare – similar knowledge-based architecture to openEHR, based on Clinical Element Models and code generation; currently adding FHIR integration.
- openEHR - [http://www.openehr.org/what\\_is\\_openehr](http://www.openehr.org/what_is_openehr) - primarily around high quality EHR information model, formalised content definitions (archetypes), terminology use, with APIs being standardised currently among vendors; will integrate with FHIR, HSPS and some IHE interfaces.

#### New / getting started

- Healthcare Services Platform Consortium (HSPC) - <http://healthcaresoa.org/>
- CommonWell Alliance - <http://www.commonwellalliance.org/>

There are undoubtedly others. As mentioned in the notes above, it is absolutely essential not to treat available standards as something that can be picked off the shelf and combined with other standards. The highest priority in a vision like JASON is a coherent, maintainable architecture.

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?

We don't agree with this recommendation as it is stated. Given that directive, EHR vendors will most likely develop and publish their own APIs, and particularly their own content and terminology models, and the situation will be as bad as it is now. Not only would they build their own, but the program they follow would be individual to each vendor.

Instead, the platform specification (including APIs) needs to be primarily outside the vendors, and managed by organisation(s) on which vendors (and other experts) can be members. See the comments above about platform-related roles & stakeholders.

Open source implementation of interoperability components should be built – these have two effects:

- Establishes a computable form of each 'standard'
- Removes the need for each vendor to 'implement' the standard(s) from scratch, thus reducing industry resistance.

It is crucial to establish an architectural board for any JASON-like architecture, as would be found in efforts like FHIR, openEHR, and emerging in HSPC for example.

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?

Firstly, as described above, APIs are only a small part of the story. Models of content, terminology, and eventually process are the main challenge, and these mostly don't appear directly within an API definition.

What is needed is to establish an owner / custodian of a platform definition, then establish within that organisation

- base standards for EHR information, content modelling, data types
- API definitions
- a 'standards factory' process for generating content models and terminology value sets (something like CIMI's goal)

This requires tooling, and open source implementations, so that industry can easily take up the outputs.

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

As mentioned above:

- development: establish / identify a custodian organisation outside of the vendors to manage / oversee the platform definition
- dissemination: fund open source implementation efforts that create definitive (and free) implementations of standards (schemas, APIs, DCMs etc)

Certification needs to be serious, yet lightweight. We would suggest based on open source reference implementations and related resources that allow vendors to unofficially attempt self-certification, prior to a streamlined approach to obtaining official certification. The process needs to be designed so that small specialist vendors are not disadvantaged.

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

Possibly by acting as something like a coordinator / custodian of the platform definition, i.e. over the top of existing SDOs and other organisations. No single SDO or other organisation has all the pieces to the puzzle, and a coordination / mandation role will need to be relatively central – even if it is actually performed by an independent .org.

#### 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?

See above responses.