SMART on FHIR: a standards-based, interoperable apps platform for electronic health records

RECEIVED 27 December 2014
REVISED 7 September 2015
ACCEPTED 7 November 2015
PUBLISHED ONLINE FIRST 17 February 2016

Joshua C Mandel,[1,2,4] David A Kreda,[3] Kenneth D Mandl,[1,2] Isaac S Kohane,[1,4] Rachel B Ramoni[3,5]

ABSTRACT

Objective In early 2010, Harvard Medical School and Boston Children's Hospital began an interoperability project with the distinctive goal of developing a platform to enable medical applications to be written once and run unmodified across different healthcare IT systems. The project was called Substitutable Medical Applications and Reusable Technologies (SMART).

Methods We adopted contemporary web standards for application programming interface transport, authorization, and user interface, and standard medical terminologies for coded data. In our initial design, we created our own openly licensed clinical data models to enforce consistency and simplicity. During the second half of 2013, we updated SMART to take advantage of the clinical data models and the application-programming interface described in a new, openly licensed Health Level Seven draft standard called Fast Health Interoperability Resources (FHIR). Signaling our adoption of the emerging FHIR standard, we called the new platform SMART on FHIR.

Results We introduced the SMART on FHIR platform with a demonstration that included several commercial healthcare IT vendors and app developers showcasing prototypes at the Health Information Management Systems Society conference in February 2014. This established the feasibility of SMART on FHIR, while highlighting the need for commonly accepted pragmatic constraints on the base FHIR specification.

Conclusion In this paper, we describe the creation of SMART on FHIR, relate the experience of the vendors and developers who built SMART on FHIR prototypes, and discuss some challenges in going from early industry prototyping to industry-wide production use.

Keywords: Electronic Health Records, Information Storage and Retrieval, Software, HL7 FHIR, Interoperability

RESEARCH AND APPLICATIONS

INTRODUCTION

In 2009, two of this paper's authors, Mandl and Kohane, recognized the clinical implications of then current, inflexible electronic health record (EHR) architectures.[1] Inspired by the agility with which the mobile developer community built novel, diverse, high-quality applications from well-defined platform-level application programming interfaces (APIs), Mandl and Kohane argued for EHR systems to become platforms capable of running third-party applications without expensive custom integration. The authors recognized that, unlike in the mobile app space, adoption of common, interoperable data specifications would be a key requirement for such an ecosystem.

In April 2010, Mandl and Kohane secured funding to pursue the vision of EHR-as-platform through the Office of the National Coordinator for Health Information Technology's (ONC) Strategic Health IT Advanced Research Project.[2] They dubbed the effort Substitutable Medical Applications and Reusable Technologies (SMART).[3]

ONC funded SMART to develop an open platform for substitutable third-party apps. The project was informed by, but operated independently of, other initiatives in ONC's interoperability portfolio (e.g., Blue Button,[4] Direct Project,[5] Nationwide Health Information Network[6]). What distinguished SMART from contemporaneous data exchange and interoperability efforts was a focus on creating immediately tangible, interoperability-supporting, vendor-independent apps (Box 1).

History of Standards Supporting Data Access and Semantic Consistency

The Health Level Seven (HL7) Version 2 (V2) Messaging standard[7] supported granular data payloads and had the advantage of wide deployment. However, V2 focused on messaging-based exchange and required significant site customization, which led to semantic inconsistencies across implementations.[8] The more recent HL7 Version 3 Reference Information Model (RIM) offered a framework for expressing clinical statements in a semantically consistent way, but implementation complexity led to incompatible systems and documents.[9,10] The Clinical Document Architecture ((CDA), based on RIM) and its C32 templates provided more detailed guidance during the Meaningful Use Stage 1 timeframe, but these specifications did not address granular data access, and also led to inconsistent implementation. In 2012 and 2013, SMART evaluated the new Consolidated CDA (C-CDA) specification for document exchange with a group of EHR vendors.[11] C-CDA, created in 2012 for Meaningful Use Stage 2, exhibited numerous challenges for data interoperability, with confusion among implementers about the implied semantics of RIM-derived elements leading to technically valid but heterogeneous and sometimes illogical data payloads. C-CDA, as a document-oriented specification, focused on rolling up disparate aspects of a patient's health record into a single summary-level document. It did not expose individual elements of data ("resources") as distinct artifacts for query or interpretation. In addition, C-CDA provided no official API for requesting documents from a source system.

First Generation: SMART Classic

The SMART team initially selected platform components that suited our mission, emphasizing Web standards (e.g., HTML, JavaScript, OAuth, Resource Description Framework (RDF)) and widely adopted terminologies for coded data (e.g., RxNorm,[12] Logical Observation Identifiers Names and Codes (LOINC),[13] Systematized Nomenclature of

Correspondence to Joshua Mandel, Department of Biomedical Informatics, Harvard Medical School; jmandel@med.harvard.edu. For numbered affiliations see end of article.

Box 1: Related work: Parallel Industry Efforts

Several interoperability-related efforts were present or emerged during our work, including

- *Blue Button*, a government-initiated branding and community engagement effort focused on encouraging provider organizations to give consumers access to their own health data.
- *CommonWell Health Alliance*, an exchange network focused on helping healthcare providers share documents across disparate clinical organizations and EHR products.
- *Microsoft HealthVault*, a private, consumer-facing, personal health record including a standalone data repository with an integrated application ecosystem.
- *S&I Framework*, an ONC-led community where individual workgroups address specific use cases by developing implementation guidance and working with standards-development organizations.

Medicine – Clinical Terms (SNOMED CT)[14]). The remaining need was a standard for sharing granular clinical data (e.g., a prescription, lab result, or blood pressure measurement) with a developer-friendly, semantically consistent, on-demand (vs. server-initiated messaging) approach. Based on 2010-era standards (see history above), no open specification offered a practical solution for enabling substitutable apps, and so we proceeded to define a small set of clinical data models in a specification that we now refer to as SMART Classic. We represented a set of common clinical statements using RDF, which we selected for its flexibility and adoption in Semantic Web applications. To expose patient-level data according to these models,[15] we built a corresponding representational state transfer API offering data access through a fixed uniform resource locator structure and hypertext transfer protocol (HTTP) verbs (e.g., GET, POST).[16] We added and refined our set of clinical statements and API calls over time to enable richer payloads with more expressive queries.

EHR Vendor Response

From 2011, when we first released SMART Classic, through mid-2013, EHR vendors were not receptive. At the time, third-party medical apps were not yet an important near-term business driver. In addition, vendors raised objections to the SMART data model and API because they had not been developed in conjunction with the health IT standards community. Vendors also identified specific technical obstacles to adoption, such as our focus on clinical data to the practical exclusion of other types, e.g., administrative data; no perceived advantage to RDF over "plain" extensible markup language (XML) or JavaScript Object Notation (JSON); concerns about our solution to a write API; and the lack of population-level API.

Progress in Healthcare Data Standards: HL7 FHIR

In 2011 the HL7 community began to recognize that HL7 Version 3 was not gaining traction and launched a "Fresh Look Task Force" including participation from the SMART team to explore new approaches to interoperability.[17,18] Two distinct initiatives emerged: the Clinical Information Modeling Initiative led by Stan Huff,[19] which built closely on the work of openEHR's archetypes[20]; and Resources For Health designed and published by Grahame Grieve.[21] Ultimately, Grieve transferred the Resources For Health specification to HL7, re-branding it as "Fast Healthcare Interoperability Resources" (FHIR) and securing an agreement that HL7 would publish it under an open license.[22] FHIR focused on providing an API for healthcare that was "simple and easy to implement and manage," inspired by contemporary Web APIs.[23] The earliest version of FHIR defined data models to support laboratory result exchanges, and by 2012, a growing community began to participate in expanding FHIR's scope.

FHIR represents clinical data as resources, where each resource is a coherent expression of meaning stated in terms of well-defined fields and data types. FHIR's clinical resource definitions are concrete, intuitive concepts such as MedicationPrescription, AdverseReaction, Procedure, and Condition. These resources constitute a graph of clinical data by explicit inter-resource references. For example, a MedicationPrescription resource explicitly references its prescriber (a FHIR Practitioner), its patient (a FHIR Patient), and the drug prescribed (a FHIR Medication). FHIR does not include detailed models for every aspect of a clinical record, but provides a built-in extensibility mechanism to enrich existing resource definitions.

FHIR resource definitions do not directly promise semantically consistent data out-of-the-box. Instead, to serve distinct contexts (e.g., EHRs, public health reporting workflows, wearable devices), FHIR resources might use specific data payloads with distinct terminologies. Semantic consistency relies on an abstraction layer called FHIR profiles that constrain and extend resource definitions in particular contexts. From SMART's perspective, profiling can enable interoperability for app substitutability by locking down data element requirements, prescribing coding systems, and imposing conventions on data representation in resources.

The FHIR API is a contemporary, resource-oriented HTTP interface to search for, create, read, update, and delete FHIR resources representing clinical, administrative, and infrastructure data (ie, constraint definitions and grouping structures), including single-patient queries and population-level queries. FHIR uses idiomatic XML and JSON to serialize resources.

METHODS

In July 2013, we reviewed FHIR in light of the EHR vendor response to SMART Classic. We determined that FHIR's API offered a superset of SMART Classic functionality, and FHIR resources played a role similar to SMART's clinical statements but with more comprehensive domain coverage. At that time, HL7 was preparing a September 2013 ballot to approve FHIR as an initial "Draft Standard for Trial Use" (DSTU1).[24] Thus, we began to collaborate with the core FHIR team and became involved in the design of FHIR resources, core infrastructure, and support tools such as automated source builds and publication of chat logs.

To adopt FHIR, we deliberately took an informal, iterative approach to platform design, without explicit gathering of requirements. We released open-source prototypes, soliciting feedback through discussion forums, conferences, and personal communication with app developers and EHR vendors. This informal process guided the re-platforming of SMART with FHIR's pre-DSTU data models and supported our engagement leading into Health Information Management Systems Society (HIMSS14) (see below).

We identified FHIR profiles to deliver semantic guarantees analogous to SMART Classic to enable substitutability while isolating app developers from EHR-specific details. We aimed to keep profiling minimal to achieve a balance between the needs of app developers and the practical reality of system vendors exposing APIs to existing data. For example, we defined a single profile for quantitative lab results rather than distinct profiles for each analyte (see below for details).

Table 1: SMART on FHIR vs. FHIR Alone

	SMART on FHIR	FHIR DSTU1 Alone
Authorization	OAuth2	None built in
Authentication	OpenID Connect	None built in
Data Models	(From FHIR)	FHIR Resources ($\sim$50 DSTU1 resources)
Profiles	SMART profiles ($\sim$10 use cases)	None built in
Data Access	(From FHIR)	FHIR REST API
Data Format	(From FHIR)	FHIR JSON or XML
EHR UI Integration	SMART launch specification including EHR context and UI embedding for Web apps	None built in
Documentation	http://docs.smarthealthit.org/	http://hl7.org/fhir

Box 2: SMART on FHIR technology components

Definitions

1. A SMART on a FHIR system is a health IT system that has implemented the SMART on a FHIR specification, including our profiled versions of FHIR, OAuth2, and OpenID Connect. Such a system is capable of running SMART apps.
2. A SMART on a FHIR app (also application, service) runs against a SMART on a FHIR system, extending its functionality through the use of clinical and contextual data.
3. OAuth2 is a Web standard for authorization. Its key function is to enable an end user (in our case, patient or clinician) to approve a third party app (e.g., a SMART app) to access a specific set of data from a service provider (e.g., EHR).
4. OpenID Connect is a Web standard for authentication. It defines an OAuth2-based protocol allowing end users to sign into apps using external identity providers.

SMART on FHIR defined a way for health apps to connect to EHR systems with appropriate security guarantees (Box 2). In addition to FHIR models and API, components included authorization, authentication, and UI integration (Table 1).

SMART on FHIR Profiles

Because substitutability requires a common set of FHIR profiles, our SMART on FHIR "starter kit" adopts conventions from Meaningful Use Stage 2.[25] The profiles specify that data must be coded according to Meaningful Use terminologies, including RxNorm for medications, LOINC for observations, and SNOMED CT for problems. Such high-level guidance does not ensure the predictable payloads required to enable substitutable apps. Therefore, we filled in details where needed. For example, our allergy profiles use distinct terminologies to communicate an allergy to a drug ingredient, drug class, or food/environmental substance. Guided by what we had done in SMART Classic, we proposed a parsimonious set of profiles, aligned with industry best practices, to foster semantic interoperability. These profiles needed to express the following kinds of constraints: terminology restrictions (e.g., requiring LOINC codes on lab observations), element cardinality restrictions (e.g., requiring the name field on a Patient resource), data type choice restrictions (e.g., requiring numerical values on a quantitative lab result), and hierarchical structuring of resources (e.g., requiring a blood pressure observation to link to sub-components for systolic and diastolic readings).

We provide EHR implementers with the details to produce consistent data feeds by describing clinical data in terms of concrete use cases like quantitative lab results, vital signs, and medication allergies. App developers still need to delve into individual terminologies to make sense of these data (e.g., LOINC has 116 codes describing measurements of serum cholesterol, of which only a subset are likely to be relevant in any particular context).[26]

Our profiling approach promotes common semantics by explicit, detail-oriented, prescriptive guidance. It does not rely upon ontologies or automated reasoning except where these constructs are embodied in the underlying terminologies (e.g., a full understanding of SNOMED CT requires description logic). Critical semantics are pushed to well-known external terminologies, so developers must learn how these terminologies work.

At the time we adopted FHIR (pre-DSTU through DSTU1), its Profile mechanism was incompletely implemented. Thus, we did not express SMART on FHIR constraints as FHIR Profile instances per se. Instead, we wrote a human-readable implementation guide with examples published at http://docs.smarthealthit.org/profiles/. Implementers were able to view this guide and quickly verify that their data translation routines were aligned with platform expectations. We also embodied validation logic in the form of sample apps, to ensure that data meet expectations. For example, vendors using the Blood Pressure Centiles app were effectively guided to structure systolic and diastolic blood pressure readings in the correct format, otherwise the app could not "see" them (e.g., render them in tables and graphs). With the transition to FHIR DSTU2, we anticipate using FHIR's automated profiling mechanisms to map SMART-specified terminologies and units, presence of specific values, and path-specific extensions.

Implementing SMART on FHIR Servers

Server implementers can create FHIR implementations with a small set of FHIR resources and incrementally expand coverage to address priorities. Implementation entails creating EHR-specific logic to transform internal data structures to corresponding FHIR resources and with SMART-specified profiles.

Example. To support apps related to blood pressure, the FHIR Observation resource provides a starting point. The Observation resource could, for example, expose separate measurements (systolic and diastolic) each with metadata (time of observation, identity of observer, etc.) or a single observation with two components and shared metadata. SMART specifies a tree structure where each set of vital signs includes a root-level "blood pressure" observation with a specified LOINC code that captures the explicit link between systolic and diastolic values (Figure 1). This resource references systolic and diastolic values as two "component" observations, each with its own specified LOINC code and metadata to be interpreted individually. The payoff is a predictable payload that avoids implicit pairing algorithms (e.g., timestamp-based joins) and supports fine grained queries (e.g. a risk calculator might require only systolic blood pressure values, while a vital signs viewer might require a complete set of vital signs).

Payload Validation

We did implement general-purpose automated data validation of FHIR payloads. The core FHIR tooling includes some automated validation packages that, as they improve over time, could be incorporated into the platform (see Discussion section).

Authentication and Authorization

To allow apps that run across heterogeneous security environments, SMART on FHIR specifies how apps obtain authorization tokens, but allows servers to apply any necessary policies to determine a user's permissions.

Every representational state transfer (REST) API call includes an authorization token obtained and transmitted via OAuth 2.[27] The scope of access tokens is kept narrow so that, for instance, an app working with a single patient record requests a limited-scope access token that is only valid for querying that patient's data (Figure 2).

Some apps rely on information about *who* is running them, and thus require an end user to authenticate or "sign in" to function properly. To accomplish this, SMART on FHIR uses OpenID Connect.[28] Via OIDC, an app can request an "openid" access scope at launch time. Upon approval, the app will receive a set of claims (name, email address, FHIR Profile uniform resource locator, etc.) packaged in a signed OIDC identity token.

User Interface Apps

We have focused on user-facing SMART on FHIR apps. We built these demonstration apps as HTML5 web apps that run in all modern web browsers on desktop or mobile devices. The platform also supports native apps on iOS, Android, and desktop operating systems. Web apps may be integrated into EHR systems in different ways: in web-based EHRs, by adding an inline frame to the web interface; and for native client EHRs, by adding a browser widget or launching an external browser. When apps launch from an existing EHR session, SMART on FHIR defines a protocol to communicate EHR context (i.e. current patient, encounter, and user identity).

Background Apps and Services

Background apps and services can request access to the same individual and population-level data FHIR API calls as user-interface apps, obtaining authorization tokens through a fully automated OAuth2 flow with no end user. This functionality uses the OAuth2 client credentials grant. Examples of uses for background functionality include computing quality data metrics, generating reminders for appointments, and alerts for abnormal lab results.

Developer-Friendly Interfaces

With respect to authentication, authorization, and clinical data retrieval, SMART on FHIR provides a straightforward app development experience. An example active medication list app needs only 25 lines of JavaScript to instantiate the SMART on FHIR JavaScript client, determine the patient in-context at the time of launch, fetch demographics and active medications, and produce a bullet list (Figure 3).

RESULTS

We created a reference implementation and other software to aid vendor implementations.

SMART on the FHIR Reference Platform

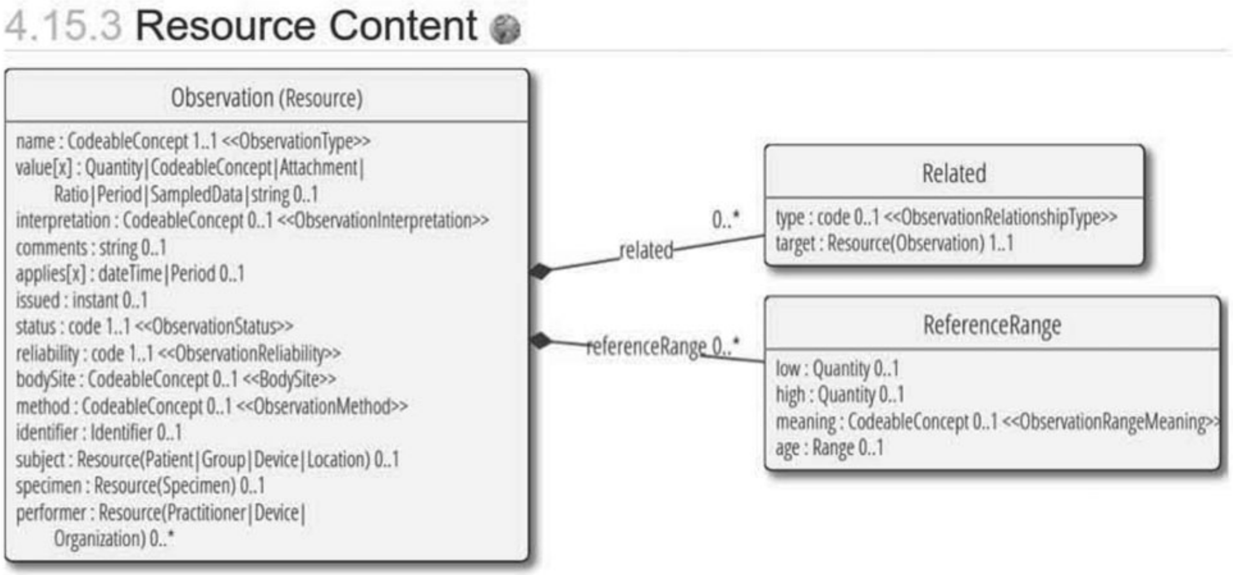
Our SMART on FHIR reference implementation (Table 2) has three key components:

1. API server
2. Authorization server
3. Apps server

Table 2: SMART on FHIR Reference Servers.

SMART on FHIR App server (UI)	https://fhir.smarthealthit.org
SMART on FHIR Authorization server	https://authorize.smarthealthit.org
*SMART on FHIR API Server 1 (*authenticated access)	https://fhir-api.smarthealthit.org (access control via OAuth2)
SMART on FHIR API Server 2 (no-authentication access)	https://fhir-open-api.smarthealthit.org (open-access for testing)
SMART on FHIR Source code	https://github.com/smart-on-fhir
Examples	
Using the no-authenticate sandbox, API Server 2, a given resource is located by its type and patient ID relative to the base server's URL: Patient 1288992: <https://fhir-open-api.smarthealthit.org/Patient/1288992>	
Data belonging to a given patient can be queried by providing search parameters, e.g., a feed of all medication prescriptions for patient 1288992: <https://fhir-api.smarthealthit.org/MedicationPrescription?patient=1288992>	

Figure 1: A FHIR Observation Resource Definition for systolic blood pressure with example in JSON.



4.15.3 Resource Content

Observation (Resource)

name : CodeableConcept 1..1 <<ObservationType>>
value[x] : Quantity|CodeableConcept|Attachment|
Ratio|Period|SampledData|string 0..1
interpretation : CodeableConcept 0..1 <<ObservationInterpretation>>
comments : string 0..1
applies[x] : dateTime|Period 0..1
issued : instant 0..1
status : code 1..1 <<ObservationStatus>>
reliability : code 1..1 <<ObservationReliability>>
bodySite : CodeableConcept 0..1 <<BodySite>>
method : CodeableConcept 0..1 <<ObservationMethod>>
identifier : Identifier 0..1
subject : Resource(Patient|Group|Device|Location) 0..1
specimen : Resource(Specimen) 0..1
performer : Resource(Practitioner|Device|
Organization) 0..*

Related

type : code 0..1 <<ObservationRelationshipType>>
target : Resource(Observation) 1..1

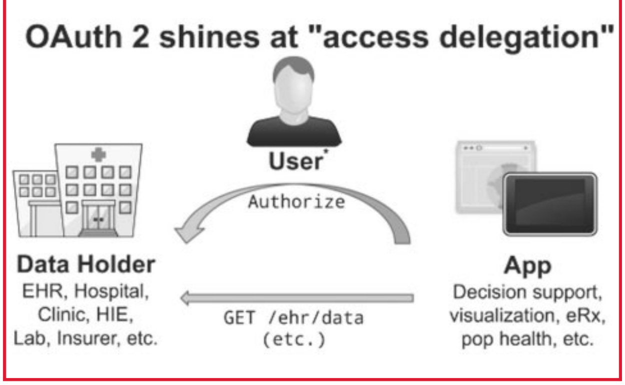
ReferenceRange

low : Quantity 0..1
high : Quantity 0..1
meaning : CodeableConcept 0..1 <<ObservationRangeMeaning>>
age : Range 0..1

Source: Resource Content diagram from FHIR specification (http://hl7.org/fhir/observation.html)

```
{
  "resourceType": "Observation",
  "name": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "8480-6",
        "display": "Systolic blood pressure"
      }
    ]
  },
  "valueQuantity": {
    "value": 109,
    "units": "mm Hg",
    "system": "http://unitsofmeasure.org",
    "code": "mm[Hg]"
  },
  "appliesDateTime": "1999-07-02",
  "status": "final",
  "subject": {
    "reference": "Patient/example"
  }
}
```

Figure 2: SMART on FHIR use of OAuth2 for access delegation.


OAuth 2 shines at "access delegation"
User
Authorize
Data Holder
EHR, Hospital, Clinic, HIE, Lab, Insurer, etc.
GET /ehr/data (etc.)
App
Decision support, visualization, eRx, pop health, etc.

Each component is open-source and presented in a public sandbox hosting a hybrid of anonymized and synthesized clinical data for about 60 sample patients.

Reference API server

In 2013, we implemented the first open-source FHIR API server in about 3000 lines of Groovy code. It takes advantage of lower-level open source FHIR parsers, serializers, and object models. It runs on the Java Virtual Machine using the Grails Web development framework backed by a PostgreSQL database. The server supports create, read, update, and delete operations for all FHIR DSTU1 resources and implements the FHIR search API, including chained search and path-based inclusion. We do not currently support composite search parameters, a FHIR construct to query internal data structures (e.g., a name-value pair nested within a FHIR resource). The server supports open access control, HTTP Basic Auth, and OAuth2 (with decisions delegated to an external server – see below). Other open source and proprietary servers have subsequently been written.[29]

Reference authorization server

We built a Reference Authorization server by modifying MITREid Connect, an actively developed open-source OAuth2 and OpenID Connect server.[30] Our server implements SMART on FHIR's application launch protocol, passing EHR context to apps that launch from an EHR session. We also support a "standalone launch" where native apps can request that context (e.g., a patient or an encounter) be established before launch.

Reference app server

Our app server exposes an EHR-like environment for developers to browse a patient list and launch apps on a given record. The environment is a SMART on FHIR UI app written in HTML5 and JavaScript. For example, we used population-level FHIR queries to implement the patient search screen.

SMART on FHIR Vendor Platforms

In January 2014, in preparation for HIMSS14, four corporate exhibitors, Cerner Corporation, Intermountain Healthcare, Hewlett-Packard Company (on behalf of the Veterans Administration), and Harris Corporation, produced prototype implementations of SMART on FHIR on their respective test systems. The collaboration came together with coordination from the Healthcare Services Platform Coalition, a provider-led group of organizations promoting a healthcare app ecosystem. By omitting authorization and exposing only those FHIR resources needed to support a small suite of sample apps, each vendor completed necessary work with one or two software engineers in under 2 months. To our knowledge, no participating vendors had previously implemented any portion of the FHIR API, so these estimates describe the time required to implement SMART on FHIR "from scratch."

These prototypes incorporated diverse functionality. Cerner demonstrated the ability to dynamically determine, based on patient demographics, which apps should be offered to the user. Harris demonstrated how federation services could produce real-time queries combining patient data from distinct FHIR servers to generate a more complete longitudinal medical record.

In the case of its HELP2 EHR, Intermountain Healthcare engaged an external software development firm to write a Java-based integration framework that assisted in exposing services once data translation plugins, unique to HELP2 were implemented. The framework was shared with Hewlett Packard, where it was used to expose FHIR services on the Veterans Administration's Vista system, as well as Harris for their multi-system integration framework.

SMART on FHIR Apps

Six SMART on FHIR apps were shown at HIMSS14 (Figure 4).[31,32] We ported three from SMART Classic as part of our reference implementation, writing a FHIR-specific open-source JavaScript library to handle common functionality including authorization and data access. This library helped us port each of three apps in just a few hours.

Our work provided a model for the third party developers building SMART on FHIR apps for HIMSS14. Intermountain developed a server facade around its HELP2 EHR, and a bilirubin assessment app. Because bilirubin is depicted in a value-over-time graph, re-using code from the SMART on FHIR Growth Chart sped implementation. Intermountain also leveraged SMART's JavaScript client library.

Polyglot Systems ported its SMART Classic Meducation app to SMART on FHIR in under one day using SMART's client library. The app provides simplified, personalized medication information for patients in multiple languages.

VisualDx ported its existing service to SMART on FHIR in 2 days using SMART's client library. The app passes patient information to a back-end component based on SNOMED-CT and RxNorm, providing diagnostic clinical decision support, including medication-induced disease and adverse events, differential diagnosis, and visual references.

Cerner prototyped FHIR data services so that the above apps could run with patient context inside its PowerChart EHR.

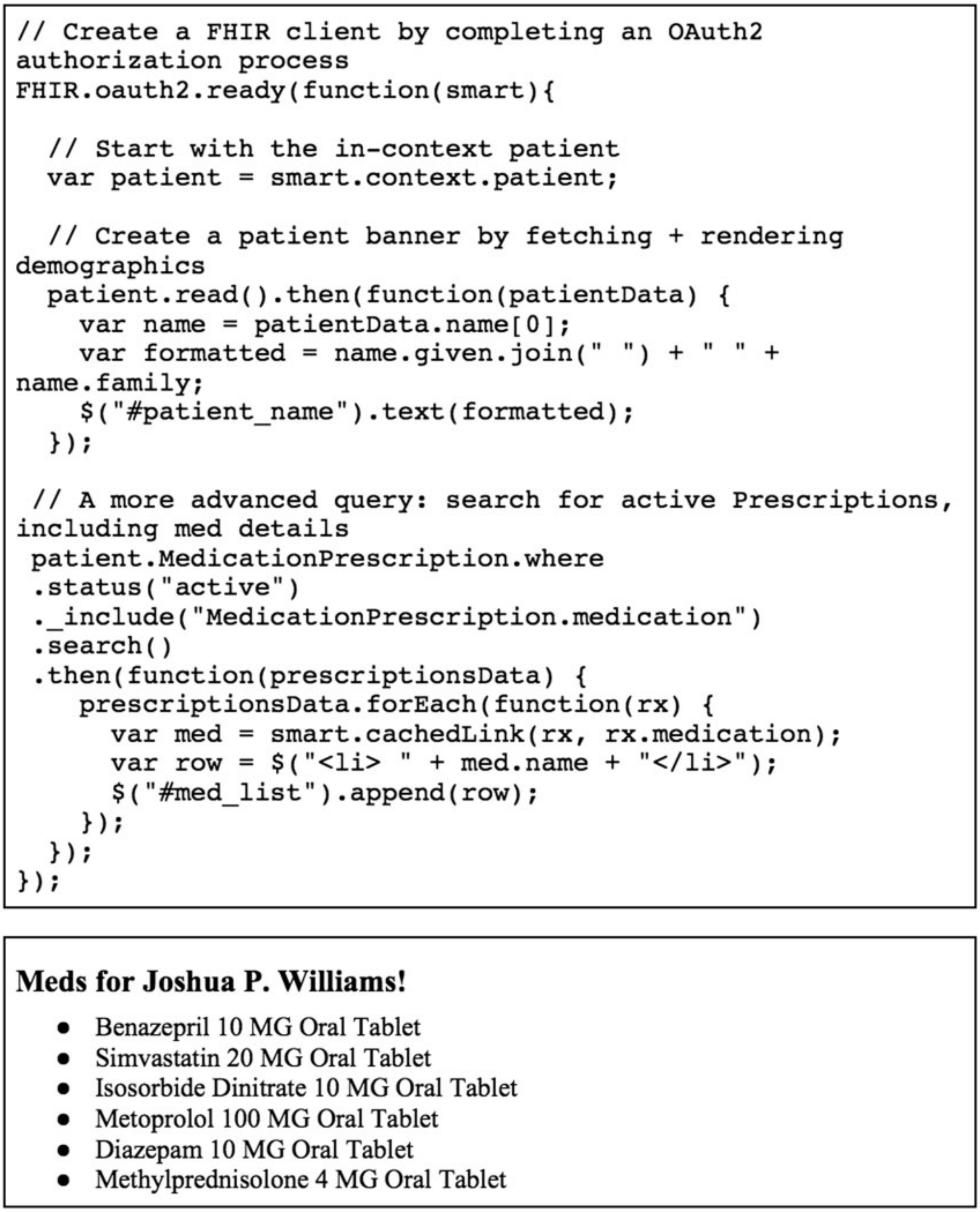
DISCUSSION

We created SMART on FHIR as a technical and market experiment to test whether standards-based data models could gain sufficient EHR vendor interest to influence the trajectory of the industry. HIMSS14 served as early evidence of the success of this experiment, demonstrating substitutability using a small set of data profiles.

In our experience with SMART Classic as well as the SMART C-CDA Collaborative[11] and now SMART on FHIR, we have come to understand that an app platform must address the following concerns:

- **Constraining resources.** FHIR's base specification does not constrain resources. It leaves almost all data optional and most coding decisions open. For example, FHIR's MedicationPrescription resource leaves every field optional, including prescription ID, date, prescriber, patient, and drug.[33]

Figure 3: Active Medication List app sample code with sample output.



```
// Create a FHIR client by completing an OAuth2
authorization process
FHIR.oauth2.ready(function(smart){

  // Start with the in-context patient
  var patient = smart.context.patient;

  // Create a patient banner by fetching + rendering
demographics
  patient.read().then(function(patientData) {
    var name = patientData.name[0];
    var formatted = name.given.join(" ") + " " +
name.family;
    $("#patient_name").text(formatted);
  });

 // A more advanced query: search for active Prescriptions,
including med details
 patient.MedicationPrescription.where
 .status("active")
 ._include("MedicationPrescription.medication")
 .search()
 .then(function(prescriptionsData) {
    prescriptionsData.forEach(function(rx) {
      var med = smart.cachedLink(rx, rx.medication);
      var row = $("<li> " + med.name + "</li>");
      $("#med_list").append(row);
    });
  });
});
```

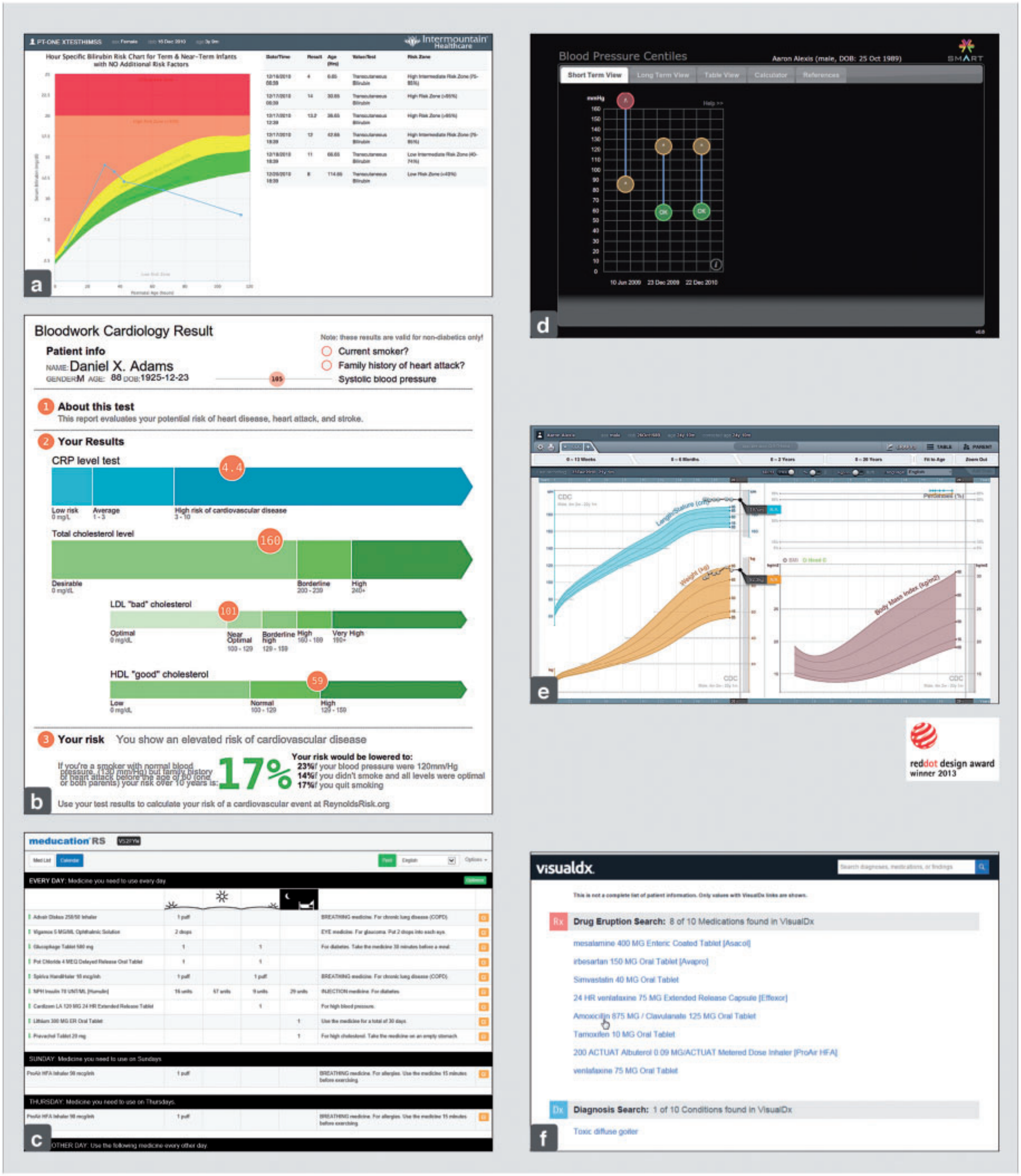
Meds for Joshua P. Williams!

- Benazepril 10 MG Oral Tablet
- Simvastatin 20 MG Oral Tablet
- Isosorbide Dinitrate 10 MG Oral Tablet
- Metoprolol 100 MG Oral Tablet
- Diazepam 10 MG Oral Tablet
- Methylprednisolone 4 MG Oral Tablet

Furthermore, FHIR takes no stance on coding of medications, conditions, lab results, procedures, or allergies; profiles must do that work. SMART on FHIR profiles, therefore, constrain FHIR resource definitions to enable substitutable apps, providing predictability that may also support non-app-oriented interoperability (e.g., peer-to-peer exchange of clinical records).

- **Curbing semantic fragmentation.** SMART on FHIR aims to identify widely applicable constraints, with an emphasis on the terminologies that systems already implement for compliance with the US Meaningful Use program. It is too early to know whether practical broad-scale interoperability can emerge with agreements covering only a small set of high-level profiling decisions in lieu of a large set of highly detailed profiles. To our knowledge, the trade-off between number and complexity of detailed profiles vs. practical interoperability has never been formally explored. As a best practice, we should be parsimonious

Figure 4: SMART on FHIR Apps at HIMSS14. (**a**) Bilirubin App by Intermountain Healthcare uses time of birth and serum bilirubin levels to monitor and flag risk for kernicterus; (**b**) Cardiac Risk App by SMART Health IT uses cholesterol lab, demographics, and other risk factors to estimate aggregate risks (concept by David McCandless and Stephanie Posovek)[31,32]; (**c**) Meducation App by Polyglot Systems, Inc. uses a medication list to produce patient-friendly instructions in 12 languages; (**d**) Pediatric Blood Pressure Centiles App by SMART Health IT uses age, gender, height, and blood pressure data to graph trends and flag hypertension per NIH guidelines (specified and used by Boston Children's Hospital clinicians); (**e**) Pediatric Growth Chart App by SMART Health IT uses gender, date of birth, available height, weight, head circumference, and body mass index data to plot growth against CDC, WHO, and disease-specific statistical norms; (f) VisualDx App by VisualDx, Inc. uses age, gender, problem list, and a medication list to provide diagnostic CDS (medication-induced disease and adverse events) and general differential diagnosis through visualization of disease. (Images courtesy of their respective authors.)



Date/Time	Result	Age (hrs)	Value/Test	Risk Zone
12/16/2010 08:39	4	6.65	Transcutaneous Bilirubin	High Intermediate Risk Zone (75-95%)
12/17/2010 06:39	14	30.65	Transcutaneous Bilirubin	High Risk Zone (>95%)
12/17/2010 12:39	13.2	36.65	Transcutaneous Bilirubin	High Risk Zone (>95%)
12/17/2010 18:39	12	42.65	Transcutaneous Bilirubin	High Intermediate Risk Zone (75-95%)
12/18/2010 18:39	11	66.65	Transcutaneous Bilirubin	Low Intermediate Risk Zone (40-74%)
12/20/2010 18:39	8	114.65	Transcutaneous Bilirubin	Low Risk Zone (<40%)

Bloodwork Cardiology Result

Note: these results are valid for non-diabetics only!

Patient info

NAME: Daniel X. Adams

GENDER: M AGE: 88 DOB: 1925-12-23

○ Current smoker?

○ Family history of heart attack?

185 Systolic blood pressure

1 About this test

This report evaluates your potential risk of heart disease, heart attack, and stroke.

2 Your Results

CRP level test: 4.4 — Low risk 0 mg/L; Average 1 - 3; High risk of cardiovascular disease 3 - 10

Total cholesterol level: 160 — Desirable 0 mg/dL; Borderline 200 - 239; High 240+

LDL "bad" cholesterol: 101 — Optimal 0 mg/dL; Near Optimal 100 - 129; Borderline high 129 - 159; High 160 - 189; Very High 190+

HDL "good" cholesterol: 59 — Low 0 mg/dL; Normal 100 - 129; High 129 - 159

3 Your risk You show an elevated risk of cardiovascular disease

If you're a smoker with normal blood pressure, (130 mm/Hg) but family history of heart attack before the age of 60 (one or both parents) your risk over 10 years is: 17%

Your risk would be lowered to:
23% if your blood pressure were 120mm/Hg
14% if you didn't smoke and all levels were optimal
17% if you quit smoking

Use your test results to calculate your risk of a cardiovascular event at ReynoldsRisk.org

meducation RS

Med List | Calendar | English | Options

EVERY DAY: Medicine you need to use every day

Advair Diskus 250/50 Inhaler	1 puff				BREATHING medicine. For chronic lung disease (COPD).
Vigamox 5 MG/ML Ophthalmic Solution	2 drops				EYE medicine. For glaucoma. Put 2 drops into each eye.
Glucophage Tablet 500 mg	1		1		For diabetes. Take the medicine 30 minutes before a meal.
Pot Chloride 4 MEQ Delayed Release Oral Tablet	1		1		
Spiriva HandiHaler 18 mcg/inh	1 puff		1 puff		BREATHING medicine. For chronic lung disease (COPD).
NPH Insulin 70 UNT/ML [Humulin]	16 units	57 units	9 units	29 units	INJECTION medicine. For diabetes.
Cardizem LA 120 MG 24 HR Extended Release Tablet			1		For high blood pressure.
Lithium 300 MG ER Oral Tablet				1	Use the medicine for a total of 30 days.
Pravachol Tablet 20 mg				1	For high cholesterol. Take the medicine on an empty stomach.

SUNDAY: Medicine you need to use on Sundays.

ProAir HFA Inhaler 90 mcg/inh	1 puff				BREATHING medicine. For allergies. Use the medicine 15 minutes before exercising.

THURSDAY: Medicine you need to use on Thursdays.

ProAir HFA Inhaler 90 mcg/inh	1 puff				BREATHING medicine. For allergies. Use the medicine 15 minutes before exercising.

OTHER DAY: Use the following medicine every other day.

visualdx

Search diagnoses, medications, or findings

This is not a complete list of patient information. Only values with VisualDx links are shown.

Rx **Drug Eruption Search:** 8 of 10 Medications found in VisualDx

- mesalamine 400 MG Enteric Coated Tablet [Asacol]
- irbesartan 150 MG Oral Tablet [Avapro]
- Simvastatin 40 MG Oral Tablet
- 24 HR venlafaxine 75 MG Extended Release Capsule [Effexor]
- Amoxicillin 875 MG / Clavulanate 125 MG Oral Tablet
- Tamoxifen 10 MG Oral Tablet
- 200 ACTUAT Albuterol 0.09 MG/ACTUAT Metered Dose Inhaler [ProAir HFA]
- venlafaxine 75 MG Oral Tablet

Dx **Diagnosis Search:** 1 of 10 Conditions found in VisualDx

- Toxic diffuse goiter

in creating new profiles. A risk is that diverse organizations may produce incompatible profiles that lead to fragmented semantics and serious interoperability challenges.

- **Data validation.** We have not yet produced a formal, automated validation process to determine whether SMART on FHIR data payloads meet expectations defined in our profiles. Pragmatically, viewing data through the lens of apps (e.g., a blood pressure percentile calculator) exposes common errors, such as apps that depend on profiled data can fail in visible ways when data are (even subtly) noncompliant (e.g., JSON numbers incorrectly serialized as strings). While this approach is helpful for debugging, it is unsuitable for real world deployments. A future objective will be to automate data validation with FHIR's Profile mechanism.

With respect to semantic fragmentation, community consensus for FHIR profiles will be key for the successful deployment of substitutable apps. To this end, since our HIMSS 2014 demonstrations, the SMART team has worked closely with the Argonaut Project,[34] whose implementation program is currently underway with upward of 40 vendors actively participating (as of September 2015) in trial implementation of SMART on FHIR's authorization and UI integration specifications. Argonaut uses a set of data FHIR profiles from the ONC's Data Access Framework project,[35] which we hope may eventually obviate the need for SMART on FHIR to maintain its own profiles.

Technical progress notwithstanding, the ultimate success for SMART on FHIR depends upon progress in two other areas. First, a critical mass of vendors must complete production level implementations to anchor the effort. Second, vendors and providers will have to extend their business and operational models to embrace a healthcare "app economy." We are encouraged by the fact that our specification and reference platform motivated several clinical system vendors to prototype SMART on FHIR platforms and several third party developers to create SMART on FHIR apps.

Lessons Learned

1. Prospective ecosystem participants must see something real before they engage in productive discussions.
2. Successful community building benefits from multiple channels of engagement around specific, shared goals with tangible results. Active collaboration among SMART, app developers, EHR vendors, and Healthcare Services Platform Coalition enabled us to build critical mass around a technology demonstration at HIMSS14.
3. Properly designed data and authentication APIs can successfully shield health IT app developers from complexity in integrating with proprietary vendor systems.
4. Granular data access, as opposed to document-oriented access, provides a well-suited model to support apps that integrate into workflow at the point of care.
5. A solution that permits incremental implementation of resources and profiles provides vendors an efficient onramp to begin app platform implementation.

LIMITATIONS

Our references implementation did not convert large-scale data or demonstrate real time query translations on top of large data sets. We have not deployed SMART on FHIR apps in production clinical environments. We have been working with DSTU1, HL7's first draft specification of FHIR, which will evolve over the coming months with DSTU2, published on September 23, 2015. Finally, because our work focuses on technical challenges, we have not explored legal and commercial barriers to integrating third party apps and services into SMART on FHIR systems. These barriers may be considerable, but, it is worth noting, not new to the software industry or, with the advent of mobile clinical apps, entirely new to this sector.

CONCLUSION

SMART aimed to produce specifications that work for forward-thinking medical app developers *and* are implementable within today's evolving healthcare technology landscape. SMART Classic satisfied app developers but secured only limited vendor interest. The FHIR draft specification emerged at a propitious time, enabling us to create SMART on FHIR, which addresses the needs of end users and app developers while providing an open-standards-based platform that aligns with the needs of clinical system vendors. We see promising signs that vendors are treating this opportunity seriously. To build upon the momentum, we recommend a strong push toward early platform adoption in service of business cases that provide value today.

CONTRIBUTORS

J.C.M. and D.A.K. wrote the first draft of the manuscript. R.R., I.S.K., and K.D.M. contributed substantially to the manuscript. J.C.M. and D.A.K. contributed equally to this work. All authors were part of the SMART project. J.C.M. is the lead architect of SMART, co-chair of the FHIR Infrastructure Workgroup, a member of the FHIR Management Group, and served on HL7's Fresh Look Task Force. D.A.K. was the SMART translation advisor. I.S.K. and K.D.M. are the PI and co-PI of the SMART Project. R.R. was the Executive Director.

FUNDING

This work was funded by the Strategic Health IT Advanced Research Projects Award 90TR000101 from the Office of the National Coordinator for Health Information Technology.

ACKNOWLEDGEMENTS

We would like to thank the following individuals, their colleagues, and their organizations for assistance with SMART on FHIR. For developing test versions of their platform to run SMART on FHIR: David McCallie of Cerner Corporation, Stan Huff of Intermountain Healthcare, Rick Freeman of Isalus Consulting, Oscar Diaz of Harris Corporation, Bo Dagnall at Hewlett Packard; Sims Preston at Polyglot Systems, and Art Papier at VisualDx. For the SMART Pediatric Growth Chart Charlie Gower and Brian McLaughlin at Fjord for design, David Brick, Dean Karavite, Ross Koppel, Daniel Nigrin for subject matter expertise, and Vladimir Ignatov for coding. We would like to thank team members Nikolai Schwertner for coding the SMART BP Centiles app and Arjun Sanyal for coding the SMART Cardiac Risk App. We also wish to thank Mark Frisse of Vanderbilt, John D'Amore of Diameter Health, and David McCallie for their helpful comments on early versions of this paper, and Rachel Eastwood for producing figures for this paper.

REFERENCES

1. Mandl KD, Kohane IS. No small change for the health information economy. *N Engl J Med.* 2009;360:1278–12781.
2. Office of the National Coordinator for Health Information Technology. Strategic Health IT Advanced Research Projects (SHARP). 2013. http://www.healthit.gov/policy-researchers-implementers/strategic-health-it-advanced-research-projects-sharp. Accessed June 22, 2015.
3. Mandl KD, Mandel JC, Murphy SN, *et al.* The SMART Platform: early experience enabling substitutable applications for electronic health records. *J Am Med Inform Assoc.* 2012;19:597–603.

4. HealthIT.gov. *Your Health Data*. 2015. http://healthit.gov/patients-families/your-health-data. Accessed June 22, 2015.
5. HealthIT.gov. *Direct Project*. 2015. http://www.healthit.gov/policy-researchers-implementers/direct-project. Accessed June 22, 2015.
6. HealthIT.gov. *Nationwide Health Information Network*. 2015. http://www.healthit.gov/policy-researchers-implementers/nationwide-health-information-network-nwhin. Accessed June 22, 2015.
7. HL7 Messaging Standard Version 2.7. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=146. Accessed September 1, 2015.
8. Bender D, Sartipi K. HL7 FHIR: an agile and RESTful approach to healthcare information exchange. In: *2013 IEEE 26th International Symposium on Computer-Based Medical Systems (CBMS)*. 2013. 326–331.
9. Fyfe J, Bender D, Edwards HK. Everest: a framework for developing HL7V3 Applications. *SIGHIT Rec.* 2012;2:24.
10. Smith B, Ceusters W. HL7 RIM: an incoherent standard. *Stud Health Technol Inform*. 2006;124:133–138.
11. D'Amore JD, Mandel JC, Kreda DA, *et al*. Are Meaningful Use Stage 2 certified EHRs ready for interoperability? Findings from the SMART C-CDA Collaborative. *J Am Med Inform Assoc.* 2014;21:1060–1068.
12. National Library of Medicine. RxNorm. 2015. http://www.nlm.nih.gov/research/umls/rxnorm/. Accessed June 22, 2015.
13. Logical Observation Identifiers Names and Codes (LOINC®) — LOINC. 2015. http://loinc.org/. Accessed June 22, 2015.
14. International Health Terminology Standards Development Organisation. SNOMED CT. 2015. http://www.ihtsdo.org/snomed-ct. Accessed June 22, 2015.
15. Substitutable Medical Apps Reusable Technologies (SMART) Project. SMART Data Models. 2014. http://classic.smartplatforms.org:6002/framework/models/. Accessed June 22, 2015.
16. RFC 2616: Hypertext Transfer Protocol – HTTP/1.1. http://www.w3.org/Protocols/rfc2616/rfc2616.txt. Accessed September 1, 2015.
17. Health Intersections Pty Ltd. *HL7 Fresh Look Task Force*. 2015. http://www.healthintersections.com.au/?p=137. Accessed June 22, 2015.
18. Health Intersections Pty Ltd. *HL7 needs a fresh look because V3 has failed*. 2015. http://www.healthintersections.com.au/?p=476. Accessed June 22, 2015.
19. Clinical Information Modeling Initiative. *Mission and Goals*. 2015. http://opencimi.org/. Accessed June 22, 2015.
20. Sebastian G, Rong C, Heather L, *et al*. Archetype-based knowledge management for semantic interoperability of electronic health records. *Stud Health Technol Inform.* 2009;150:1007–1011.
21. Resources for Health. *Introduction*. 2011. http://web.archive.org/web/20111009093445/http://www.healthintersections.com.au/rfh/introduction.htm. Accessed June 22, 2015.
22. Health Intersections Pty Ltd. *FHIR Licensing Update*. 2015. http://www.healthintersections.com.au/?p=868. Accessed June 22, 2015.
23. Health Intersections Pty Ltd. *Resources For Health: A Fresh Look Proposal*. 2015. http://www.healthintersections.com.au/?p=502. Accessed June 22, 2015.
24. Health Level 7. *FHIR Specification Home Page*. 2015. http://hl7.org/fhir/. Accessed June 22, 2015.
25. Substitutable Medical Apps Reusable Technology. *SMART on FHIR Profiles*. 2015. http://docs.smarthealthit.org/profiles/. Accessed June 22, 2015.
26. Logical Observation Identifiers Names and Codes. *Search for "Serum Cholesterol"*. 2015. http://search.loinc.org/search.zul?query=serum+cholesterol. Accessed June 22, 2015.
27. Internet Engineering Task Force. RFC 6749. *The OAuth 2.0 Authorization Framework*. 2015. https://tools.ietf.org/html/rfc6749. Accessed June 22, 2015.
28. OpenID Foundation. *Final: OpenID Connect Core 1.0 Incorporating Errata set 1*. 2015. http://openid.net/specs/openid-connect-core-1_0.html. Accessed June 22, 2015.
29. Health Level 7. *Open Source FHIR Implementations*. 2015. http://wiki.hl7.org/index.php?title=Open_Source_FHIR_implementations. Accessed June 22, 2015.
30. MITREid Connect. http://mitreid-connect.github.io/. Accessed September 4, 2015.
31. McCandless D. *Information Is Beautiful*. 2015. http://www.informationisbeautiful.net/. Accessed June 22, 2015.
32. Leckart S. *The Blood Test Gets a Makeover*. WIRED. 2010. http://www.wired.com/2010/11/ff_bloodwork/3/. Accessed June 22, 2015.
33. Health Level 7. *FHIR MedicationPrescription*. 2015. https://www.hl7.org/fhir/medicationprescription.html. Accessed June 22, 2015.
34. Argonaut Project. http://argonautproject.org/. Accessed September 1, 2015.
35. Standards & Interoperability (S&I) Framework - Data Access Framework. http://wiki.siframework.org/Data+Access+Framework+Homepage. Accessed September 1, 2015.

AUTHOR AFFILIATIONS

[1]Computational Health Informatics Program at Harvard-MIT Health Sciences and Technology, Boston, MA, USA

[2]Department of Pediatrics, Harvard Medical School, Boston, MA, USA

[3]SMART Health IT Project, Center for Biomedical Informatics, Harvard Medical School, Boston, MA, USA

[4]Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA

[5]Department of Oral Health Policy and Epidemiology, Harvard School of Dental Medicine, Boston, MA, USA

RESEARCH AND APPLICATIONS