OSEHRA

Open API and Open Architecture Working Group (OA2-WG)

Progressing from Planning to Doing

April 24, 2019

Agenda for April 24, 2019

- Review February 20th Meeting Content (10 minute version)

  Review, because it’s been a while

  - The Goals for Healthcare IT Products
  - The Open API/Open Architecture Approach
  - Review “reused” system engineering terms/concepts discussed earlier
  - CBSA – A definition by analogy
- The OA2WG’s Proposed “Framework”, an Open Architecture for HIT
  - OA2WG, Progressing from Discussion to Action

Goals for Healthcare IT Products (skim)

- For the HIT application consumer and user, enable the industry to:
  - Reduce procurement cost
  - Increase quality of HIT by reducing risk of errors, known and unknown
  - Increase performance & usability

  …SIMULTANEOUSLY!
- Accomplish this by using Open API's, in an Open Architecture, employing Reusable HIT Components to:
  - Reduce the difficulty and complexity of learning how to build new HIT applications
  - Allow new developers, designers, maintainers, and the investors who develop the organizations that supply HIT products to succeed in the HIT industry

  … by opening the HIT market to new developers with existing IT skills, and reducing the risk in investing in HIT; while increasing competition to lower cost and increasing value

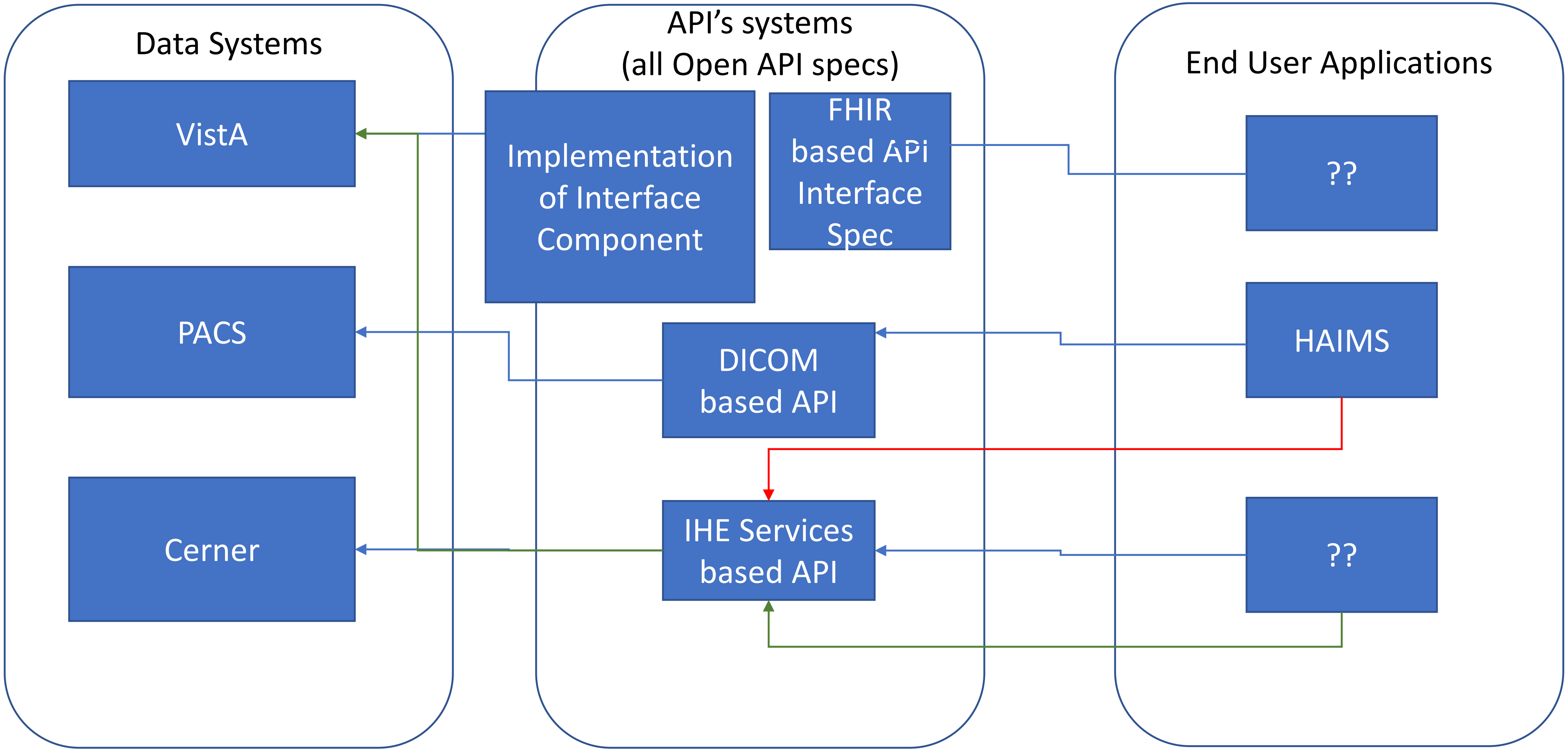
The Open API/Open Architecture Approach

- Open Architecture and Open API’s
  - Allowing all technology contributors to participate in HIT development
- Component Based Software Assembly (CBSA)
  - Reduce cost and increase quality by leveraging quality tested components that have already been developed; instead of re-inventing the wheel
- Integrated Development Environments (IDE) and Software PlugIns/Wizards (review)
  - Reuse design architecture in the same way we reuse individual components; for lower cost, higher quality with the added advantage of greater interoperability

“Reused” System Engineering Terms/Concepts

- Application Programming Interface (API)
  - An interface allowing integration of new software with an existing system
- Open API
  - An API that allows integration with multiple similar systems, which supply the same services and function with a common interface
  - Open API’s can be used to specify a component that is itself Open Source, or components that are proprietary
- Open Architecture
  - A system architecture that leverages multiple Open API’s (as well as proprietary API’s) to define an architecture of multiple similar systems
- Component Based Software assembly (skim?)
  - A approach to build systems from pre-built components using Open API’s & system development toolset, or Integrated Development Environment (IDE)
- IDE plugins aka Wizards (review)
  - A application domain (in our case HIT domain) specific customization of an IDE that leverages Open API’s, components that implement those API’s, an open Architecture, and assembly tools to create new systems in that domain

Open API/Open Architecture Example (skim)



Data Systems

VistA

PACS

Cerner

API's systems
(all Open API specs)

Implementation of Interface Component

FHIR based API Interface Spec

DICOM based API

IHE Services based API

End User Applications

??

HAIMS

??

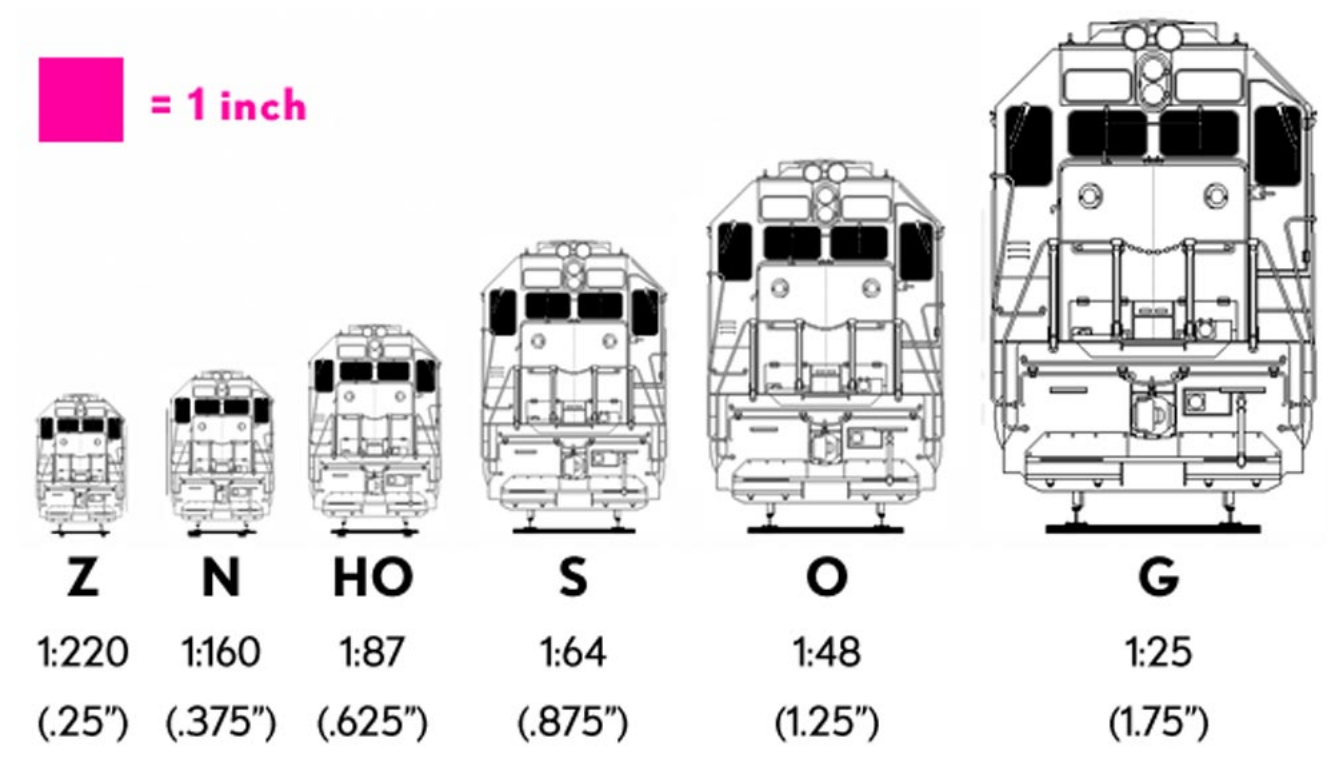
Component Based Software Assembly (Skim?)

- Examples of building from kits, versus building from “frameworks”
  - BTW, we need a better term then “frameworks”
- CBSA build news products from:
  - A set of reusable software components, each of which implements a part of a HIT application which multiple systems have in common (e.g.ID patient, or create appointment, or list available radiology studies for a specific patient)
  - A set of “Open API’s”, each of which allows us to control a set of reusable software components which perform similar functions
    - Each individual component which supplies the functions defined in the Open API is an implementation, or instantiation of that Open AP
  - An Open Architecture, which is an architecture of a HIT application which uses Open API’s to integrate multiple reusable components together to build MULTIPLE applications
  - Tools and guidance in how to use CBSA resources to build applications

Component Based Software Assembly Defined by Analogy (review)

- Model Trains Example
- Electronics Example

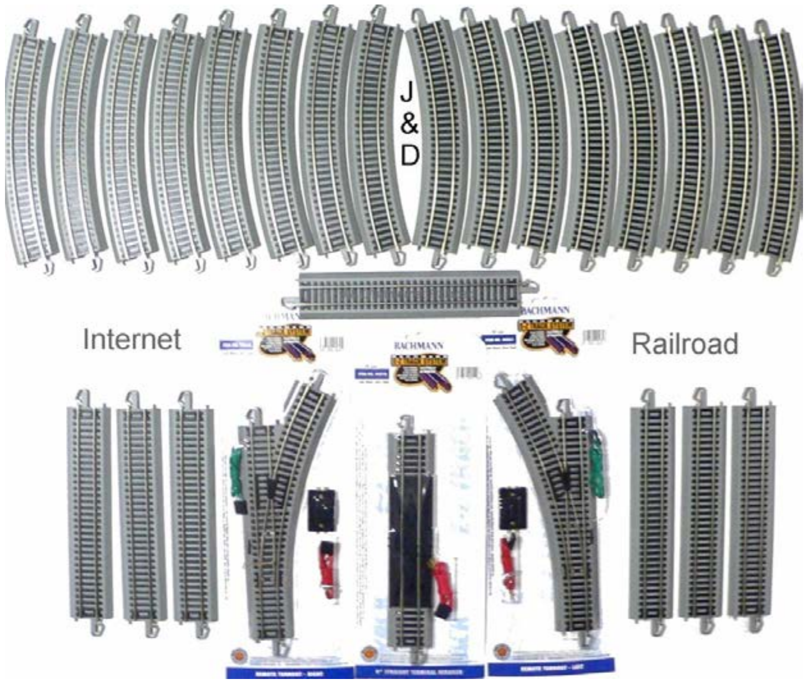
Model Train Components (review)

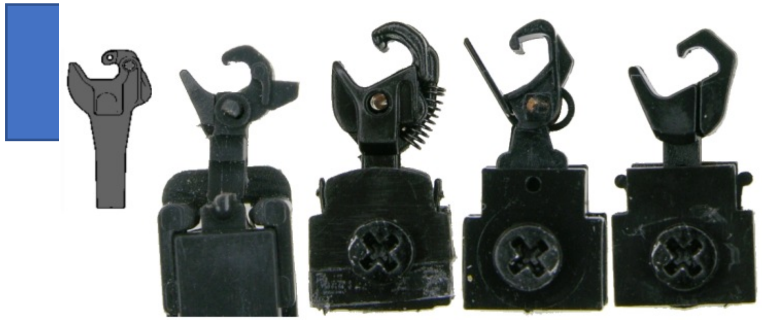


Z	N	HO	S	O	G
1:220	1:160	1:87	1:64	1:48	1:25
(.25")	(.375")	(.625")	(.875")	(1.25")	(1.75")

COMMON MODEL RAILROAD SCALES (AND GAUGES)

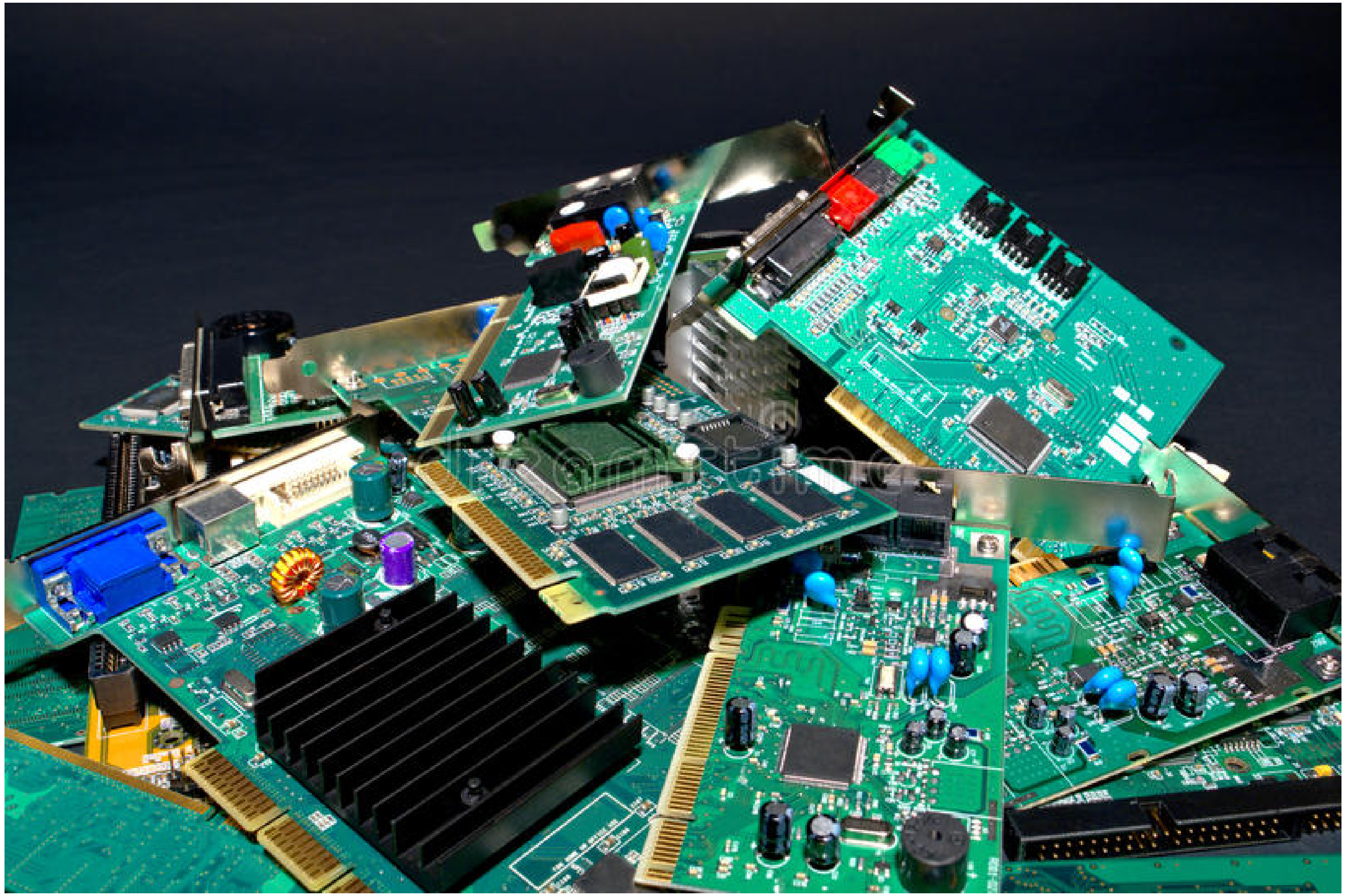
Model Train Open Architecture(s) – one per gauge (review)





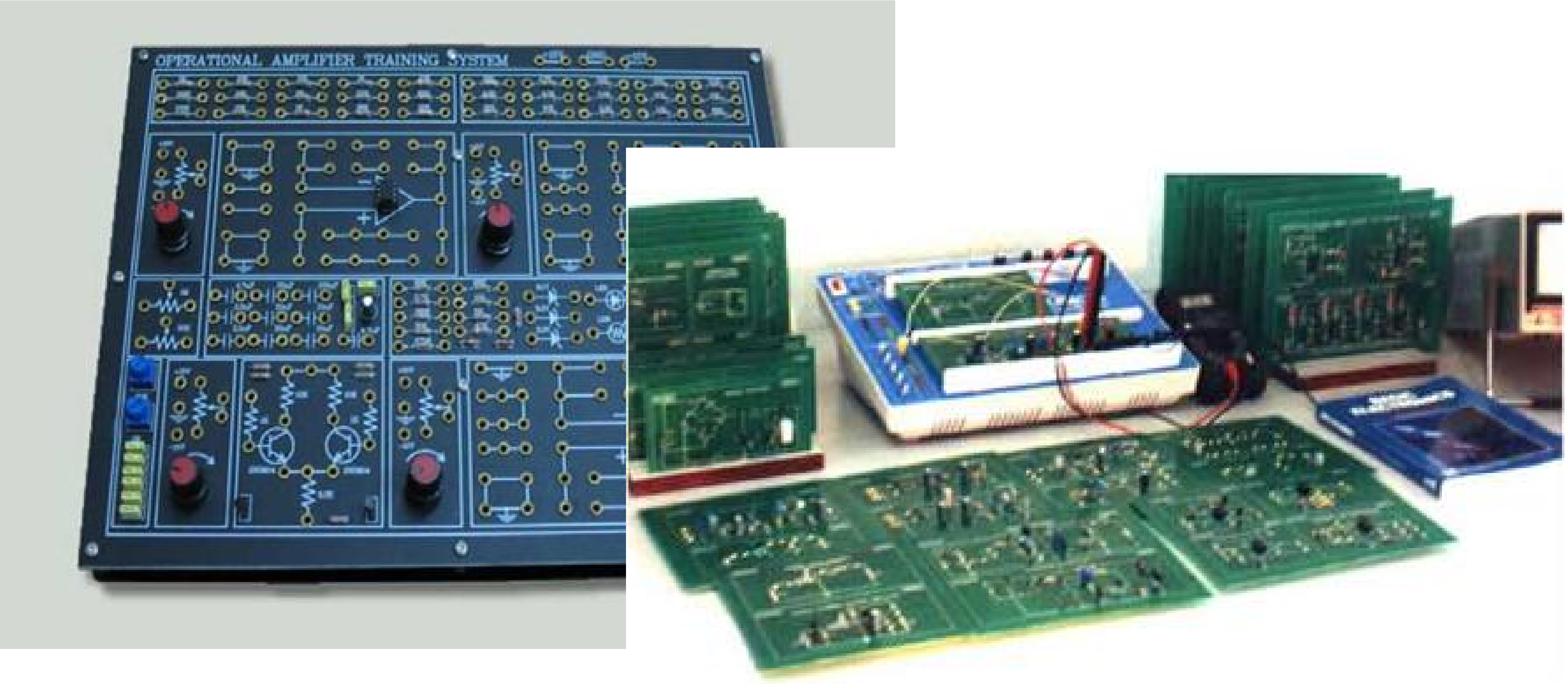
This is the interface

Electronic Components (review)



Categorized Electronic Components (review)

An Open Architecture for Electronics (review)



Time to get to work, ready?

- First, change our name, to “Healthcare IT Open Architecture”, or simply the Open Architecture working group
    - The name change has been raised in earlier meetings, and its worth considering
    - People outside the working group need to have a title that has semantic meaning, and acronyms generally don’t do that
- We reviewed these next slides in February but did not have time to discuss them, now we will

Life cycle of a Open Architecture Framework

- To build a usable CBSA based Open Architecture, we'll need
  - A scope of what kind of systems we want to be able to build with the Open Architecture
    - A mission statement and a set of usage scenarios that define the functions we will be able to implement
  - A set of Open API interfaces
  - A set of components that implement those interfaces
  - The Open Architecture Model itself
    - A structure, a design pattern that shows how components which implement those API's work together to achieve a larger goal than one component can achieve
  - Tools to build new components and build system from components
  - Instructional material on how to use all these new resources

Our Core Goal a HIT Open Architecture Model

- To define our CBSA based Open Architecture model, we must define, identify or create:
  - What are our standards interfaces, aka HIT Open API's
  - What are our HIT CBSA reusable components, which are implementations of those Open API's?
    - Remember, there is a one to many relationships between a Open API, and the set of components which implements EACH of those Open API's
  - What is our Open Architecture which guides how to integrate multiple reusable components, through their respective Open API's to build new HIT applications?
  - What tools and guidance will we collect and/or create to help developers build new HIT applications from our HIT CBSA "framework"?

Using CBSA to Build New Apps from Open Architecture

- Define a single design pattern, which can be instantiated through a family of applications
    - Be careful to define SEI “family of apps” and “UML/Gang of 4” design patterns
- Design that pattern, its commonalities and variations
- Identify the common components that can be used to build applications that are variants of that pattern
- Collect existing reusable components (those that we can acquire and use)
- Build new components to be reusable, where feasible
- Assemble a set of system development tools that will streamline the process of implementing a member of that design pattern “family”
- Build new applications using the pattern, assembled (in part) from those reusable components (and new ones) assembled using that toolset

Mission Statement: What HIT Services can be Built Using HIT Open Architecture 1.0?

- We’ve previously discussed this and talked about a version 1.0 scope that isn’t already well served by HIT industry, and also is manageable in terms of size and complexity
- Approach like a Reengineering project: as is state, to be state, and the transition plan
- Two functional scopes have come up more than once:
  - Information aggregators – client software that leverages interoperability services to aggregate and normalize patient information
  - Patient facing apps – client software that directly serves patients: e.g. managing medical information from multiple sources, storing it under the control of the patient rather than individual providers or payer organizations

What is the scope of Open Architecture 1.0?

What are our standards interfaces, aka HIT Open API’s that Fit that Scope?

- … and what common services will be needed that have yet to be defined as Open API’s?

What HIT CBSA reusable components implement those Open API’s?

- …and which ones don’t exist that will need to be build?

What are our HIT CBSA reusable components, the implementations of those Open API’s?

What Development Platforms are supported?

- We may consume services from many platforms:
  - M-Code/Cache
  - Proprietary systems such as Cerner
  - Etc.
- We need to maximize the number of potential developers that will be able to move into HIT, by picking the most popular development platforms that can support HIT development
  - Potential Candidates: Visual Studio (C#, VB and C++), Java, and scripting language stacks such as HTML5
- We need to limit the number of platforms in Open Architecture 1.0 to reduce the complexity and level of effort to get 1.0 completed.

Work to be done 1st - Volunteers

- 1st Draft of Scope Statement and List of Usage Scenarios
  - Just the list not the complete scenarios (yet)
- 1st Draft or Open Architecture
  - The common components that will be required to satisfy the those usage scenarios
- 1st Draft List of available Open API's, candidates for inclusion in Open Architecture 1.0

Obviously whatever each team for each task produces may and probably will change what other teams do; so this process will have to be iterative.