

```
"resourceType" : "Patient"
"text" : {
  "status" : "generated"
  "_status" : {
    "id" : "12344"
  },
  "div" : "<div
},
"identifier"
{
  "use"
  "ty"
```

FHIR FUNDAMENTALS COURSE

FAST HEALTHCARE INTEROPERABILITY RESOURCES



Curso FHIR Fundamentals, Unidad 4

Perfiles y Guías de Implementación

Material de Lectura

Course Overview

Module I: Introduction

Introduction to FHIR

Resources

Module II: Work with FHIR

RESTful FHIR

Searching with FHIR

Module III: FHIR advanced

Transactions

Paradigms

Messaging

Documents and CDA R2

Operations

Module IV: FHIR Conformance

Conformance Resources

Extensions

Profiles

Implementation

Tabla de Contenidos

Tabla de Contenidos	3
Contenido de la Unidad y Objetivos	5
¿Para qué necesitamos perfiles?	6
¿Porqué creamos perfiles?	8
¿Qué es lo que se puede restringir?	9
¿Cómo documentamos las restricciones?	10
Recursos de Conformidad FHIR	11
Operaciones	12
CapabilityStatement	12
OperationDefinition	13
SearchParameter	15
Contenido	16
StructureDefinition	16
DataElement	19
Extensiones FHIR	19
Extensiones Complejas	22
Extensiones modificantes ('Modifier')	22
Vinculación de Vocabulario - Binding	23
Terminología	24
CodeSystem	25
ValueSet	27
Operaciones sobre el recurso Valueset: \$expand and \$validate	28
ConceptMap	29
NamingSystem	29
Publicación	30
ImplementationGuide	30
Procesando el recurso ImplementationGuide	31
Validación de Recursos contra un Perfil	32
Declaración de Conformidad en Instancias de un Recurso	32
Validación de recursos	32
Herramientas para Crear Perfiles	34
Forge	35
Trifolia	36
Simplifier	37
clinFHIR	38
Snapper	39
Algunas guías de implementación y perfiles relevantes	40
Proyecto Argonaut	40

IHE-on-FHIR	42
Smart-On-FHIR.....	43
Guías de Implementación Oficiales de FHIR	44
Guías de Implementación Regionales Públicas	44
Más información sobre perfiles FHIR.....	45
La actividad de esta semana	46
Resumen de la Unidad y Conclusiones	47
Material de Lectura Adicional	48
Información sobre FHIR.....	48

Contenido de la Unidad y Objetivos

Esta unidad discute temas avanzados acerca de perfiles FHIR: cómo describir extensiones y restricciones sobre recursos FHIR, y cómo documentar estas restricciones.

Esta unidad complementa lo que ha aprendido en la unidad de la última semana: Ud. puede combinar cualquier recurso, y extender libremente FHIR, pero también debe **documentar estas decisiones**: permitir que el resto del mundo (o al menos, que los que participan en el proyecto de interoperabilidad, actuales y futuros), **qué se espera que cada aplicación reciba y/o envíe**, y **cómo validar los recursos** antes de (o inclusive en) producción.

La buena noticia es que el formato de la documentación también está en forma de recurso FHIR.

Nota: Esta unidad está basada libremente en el tutorial de Graham Grieve's 'Profiling Concepts', impartida durante FHIR DevDays 2014-2017, el tutorial de Michael Rutten's 'FHIR for Specifiers', y la experiencia e interpretación de los autores de la unidad.

Los recursos de conformidad fueron aumentados y re-estructurados por STU3, y constituyen una buena parte de lo que se transforma en normativo para FHIR R4 (Nivel de Madurez FHIR 5).

Nuestro objetivo para esta unidad **NO ES** que usted se transforme en un experto en Perfiles y Guías de Implementación FHIR, sino que al menos **conozca los recursos involucrados, su uso esperado, las herramientas disponibles, y también algunos ejemplos de su utilización**.

También queremos dirigir su atención a alguna de las guías de implementación más importantes.

¿Para qué necesitamos perfiles?

FHIR es una especificación internacional, utilizada en diversos contextos. Tal como ya habrá descubierto, cada servidor puede implementar un subconjunto de la especificación en términos de capacidades, contenido, formato y mecanismo de transportes.

De hecho, en realidad hay más desacuerdos que acuerdos. Las leyes, regulaciones, cultura, requerimientos, herramientas y diferencias en población pueden cambiar los objetivos para los sistemas y para el intercambio de información.

El mismo concepto puede ser implementado en sistemas diferentes utilizando distintos componentes, atributos, etc.

Ejemplo, como se ve en Figura 1:

Sistema 1: Alergia: Descripción (codificada) + Naturaleza de la Reacción

Sistema 2: Alergia: Descripción (codificada) + Texto + Naturaleza de la Reacción + Severidad

Sistema 3: Alergia: Descripción (codificada) + Texto + Tipo Genérico + Clase + Molécula

Sistema 4: Alergia: Descripción (codificada) + Comentarios + Estado + Fecha de Inicio+ Reacciones (codificada)

FIGURA 1 – DIFERENTES ENFOQUES PARA EL MISMO PROBLEMA

The figure displays four distinct user interface windows for managing allergies, illustrating different data models and input methods:

- Add Patient Allergy:** A form for a patient named Steven W. Carlson. It includes fields for 'Allergy' (Blackberries), 'Status' (Active), 'Onset Date' (12/13/1970), and a list of 'Reactions' (Skin Rashes/Hives, Nausea/Vomiting/Diarrhea, Shock/Unconsciousness, Anemia/Blood Disorder, Asthma/Shortness of Breath, Other). It also has a 'Comments' text area and 'Save'/'Cancel' buttons.
- Allergies:** A summary view for a patient. It shows 'Allergy' (Penicillin G Sodium), 'Details' (Nausea), 'Generic' (1282), 'Class' (0), 'Molecule', and 'Class'. It includes navigation arrows and 'Delete', 'Cancel', and 'Save' buttons.
- Add reaction:** A window for adding a reaction. It has a 'Search' field with a list of items (Acacia, Ant bites, Band Aids, Bee stings, Cats, Cockroach dander, Dogs, Eggs). It includes 'Nature of reaction' (Bronchospasm) and 'Severity' (Moderate) dropdowns, and 'Save'/'Cancel' buttons.
- Add Allergy:** A window for adding a new allergy. It has 'Enter allergy:' radio buttons for 'By drug', 'By class' (selected), and 'Other'. It features a list of drug classes (SHT RECEPTOR BLOCKERS, ACE INHIBITORS, ALPHA-ADRENERGIC BLOCKERS, ALUMINIUM HYDROXIDE AGENTS, AMINOGLYCOSIDES) and a 'Nature of reaction:' field. It includes 'Add' and 'Cancel' buttons.

Así que FHIR necesita ser implementado más allá de estas diferencias, pero también expresando de alguna manera las opciones y la implementación específica en cada sitio o región.

Históricamente, los perfiles en HL7 se desarrollaban con una sintaxis diferente que el estándar en sí (Schematron, grandes archivos Word, archivos especiales XML, Schemas, etc.). Y expresar los perfiles fue siempre pensado luego de pensar el estándar.

FHIR trae una nueva visión a los perfiles de conformidad y validación: los perfiles, declaraciones de conformidad, y las guías de implementación, pueden ser creados usando recursos FHIR, y un servidor FHIR puede validar instancias de los recursos contra estos perfiles como una de las operaciones definidas.

Así, FHIR incluye de fábrica la capacidad de desarrollar y verificar perfiles utilizando el mismo estándar y los mismos servidores.

Podemos ver a FHIR como una ‘especificación de plataforma’: siempre necesitamos acuerdos adicionales. La única diferencia con las especificaciones de HL7 anteriores es que en FHIR estos acuerdos son computables, y hasta ‘descubribles’: puedes preguntar a un servidor que servicios brinda, y actuar según la respuesta.

¿Porqué creamos perfiles?

- 1) Comunicar a sus colegas que se decidió o espera.
- 2) Permitir comparaciones y verificaciones automáticas
- 3) Soportar generación de código / descubrimiento en tiempo real
- 4) Crear un ecosistema de publicación y suscripción
- 5) Permitir validación de conformidad

Un servidor FHIR se convierte también en un repositorio de extensiones, perfiles, guías de implementación y una herramienta de validación.

¿Qué es lo que se puede restringir?

Casi todos los componentes en un servidor FHIR pueden ser restringidos.

Las preguntas básicas son “¿Qué se utiliza?”, “¿Qué es lo que no se utiliza?”, “¿Qué es lo que se extiende?”

Y la premisa básica es **“No se puede contrariar la especificación FHIR”**.

- 1) Si algo es obligatorio en FHIR, no puede ser opcional en su Guía de Implementación.
- 2) Si un código está ligado a un vocabulario requerido en FHIR, no podemos cambiar el vocabulario.

Una lista corta de ítems que se pueden restringir en un perfil:

Elementos en recursos:

¿Qué elementos **opcionales son obligatorios**?

¿Qué elementos necesitan soportar repeticiones en su implementación? ¿Cómo podemos diferenciar las repeticiones entre sí?

Códigos en Elementos Codificados:

¿Qué terminologías se utilizan? ¿Aplicados a qué elementos codificados?

Conducta:

¿Qué interacciones específicas de la API RESTful soporta nuestro servidor para cada recurso que manejamos?

¿Permitimos borrado de recursos? ¿Soportamos versionado de recursos?

Operaciones especiales: definición de operaciones NO-CRUD y sus parámetros.

Búsquedas:

¿Qué parámetros de búsqueda soportan nuestros servidores para cada recurso?

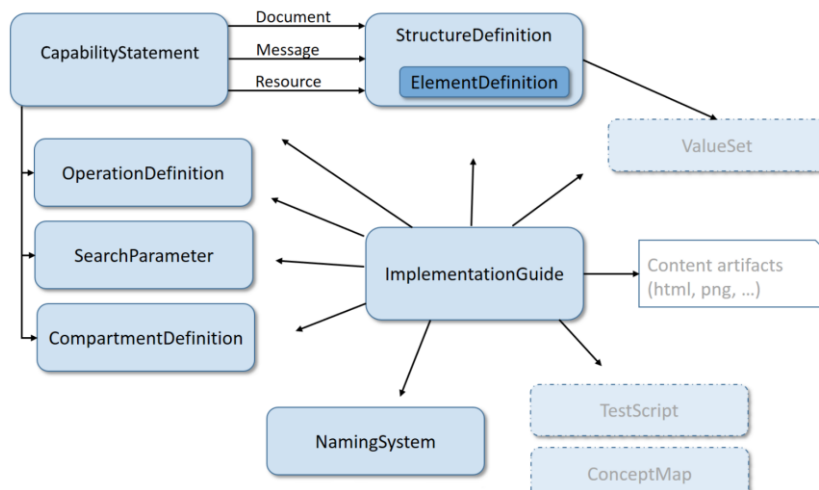
Extensiones:

¿Qué extensiones específicas debemos soportar o requerir? ¿Aplicados a qué recursos, elementos y tipos de datos?

¿Cómo documentamos las restricciones?

Las restricciones y sus enunciados de uso derivados necesitan ser:

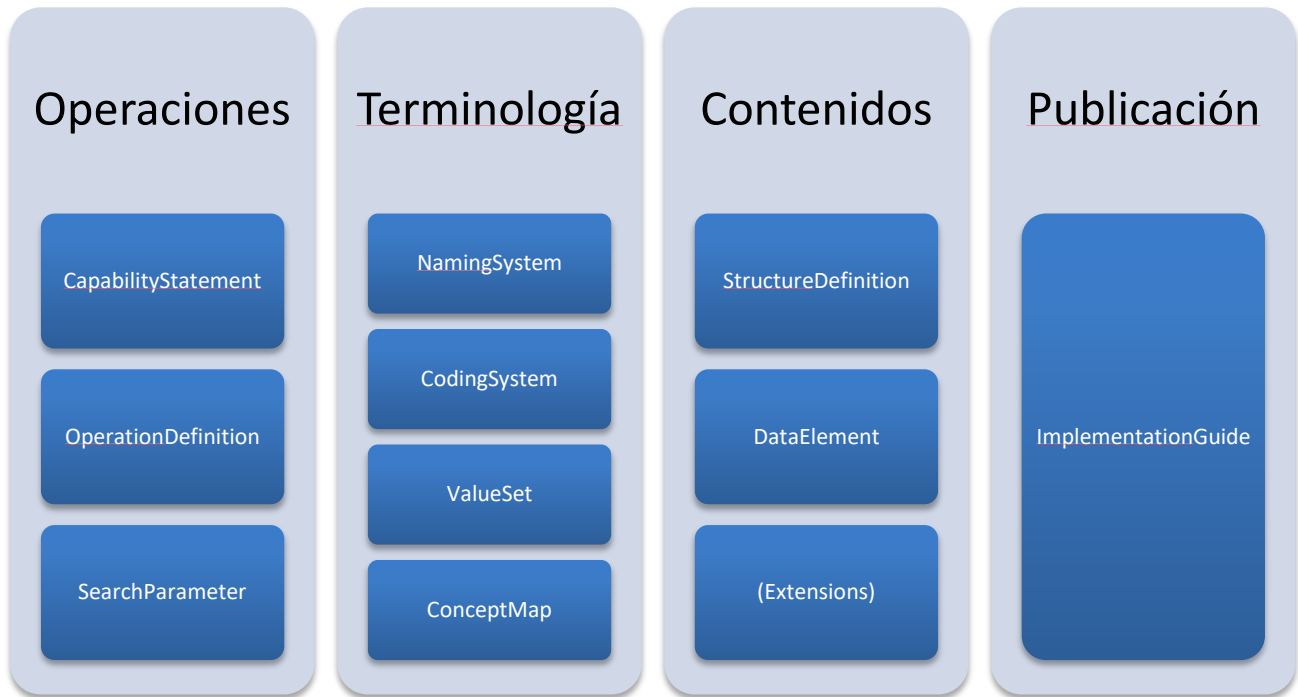
- 1) Creadas de manera estructurada
- 2) Publicadas en un repositorio
- 3) Usadas como base para la validación y generación de código y UI



Recursos de Conformidad FHIR

HL7 definió que se utilizarán recursos FHIR para expresar estas restricciones.

Aquí exploraremos el uso de cada recurso y sus combinaciones, dividiéndolos en cuatro áreas



Como ya habrá pensado, el recurso **ImplementationGuide** empaqueta todos los recursos de conformidad en un paquete para publicación disponible para los implementadores.

Nota: Hay más recursos de conformidad (MessageDefinition, GraphDefinition, CompartmentDefinition, etc.) pero nos enfocaremos en este subconjunto porque incluye los más utilizados.

Operaciones

Este grupo de recursos es acerca de la CONDUCTA del servidor, lo que el servidor puede o no hacer, y como esto se implementa.

CapabilityStatement

Este es el recurso más importante: permite entender que llamadas a API se aceptan, y sobre qué tipo de recursos.

Porque esto es lo primero que debe hacer cuando implementa o enfrenta (como cliente FHIR) un servidor FHIR.

Puede ver una rendición parcial de un recurso CapabilityStatement en la FIGURA 2.

Puede buscar la declaración de conformidad del servidor de este curso (o de cualquier otro) , pegando esto en su navegador de internet, o cliente REST (puede tomar un tiempo, sea paciente):

<http://fhir.hl7fundamentals.org/r4/metadata>

FIGURA 2 – DECLARACION DE CONFORMIDAD DE UN SERVIDOR

Mode	SERVER
Description	All the functionality defined in FHIR
Transaction	y
System History	y
System Search	y

Resource Type	Profile	Read	V-Read	Search	Update	Updates	Create	Delete	History
Account	http://hl7.org/fhir/StructureDefinition/Account	y	y	y	y	y	y	y	y
AllergyIntolerance	http://hl7.org/fhir/StructureDefinition/AllergyIntolerance	y	y	y	y	y	y	y	y
Appointment	http://hl7.org/fhir/StructureDefinition/Appointment	y	y	y	y	y	y	y	y
AppointmentResponse	http://hl7.org/fhir/StructureDefinition/AppointmentResponse	y	y	y	y	y	y	y	y
AuditEvent	http://hl7.org/fhir/StructureDefinition/AuditEvent	y	y	y	y	y	y	y	y
Basic	http://hl7.org/fhir/StructureDefinition/Basic	y	y	y	y	y	y	y	y
Binary	http://hl7.org/fhir/StructureDefinition/Binary	y	y	y	y	y	y	y	y

Información incluida en la declaración de conformidad:

Conformidad del Servidor Completo: Nombre y versión del Software, operaciones especiales soportadas, y sus parámetros, Versión de FHIR, Sintaxis soportadas: JSON, XML, otros.

Por cada clase de recurso: Interacciones CRUD soportado, Creación/Actualización/Borrado Condicional para cada tipo de recurso, Inclusiones de búsqueda, parámetros de búsqueda.

OperationDefinition

La API RESTful de FHIR define un conjunto de interacciones comunes (leer, actualizar, buscar, etc.) realizados en un repositorio de recursos tipados. Este enfoque resuelve muchos casos, pero a veces es mejor usar un paradigma RPC, complementando las operaciones CRUD regulares.

En este paradigma, hay una operación NOMINADA, invocada con parámetros de entrada (recursos, datos). Se invocan utilizando el signo \$ (pesos) ([fhirserver.com/base/\\$operation_name](http://fhirserver.com/base/$operation_name))

Las operaciones pueden ser realizadas en una instancia específica de recursos, un tipo de recurso, o un sistema completo.

El recurso OperationDefinition permite proveer una definición computable y legible de la operación especial definida (no-CRUD).

Puede ver una definición de operación en el archivo [OperationDefinitionExample.xml](#) (no incluida aquí por limitaciones de espacio).

Este recurso incluye la descripción de la operación, para cada parámetro, y para cada parámetro, el uso (entrada/salida), Nombre, Cardinalidad, Tipo, y Vinculación de Vocabulario.

En el ejemplo, la operación está definida en el recurso Composition (así que use se invoca [http://someserver.fhir/base/Composition/222/\\$document](http://someserver.fhir/base/Composition/222/$document))

Esta Operación se define así:

Context: sistema, tipo de recurso, o instancia de recurso.

Name: Un nombre para la Operación

Idempotent: La operación no cambia los recursos subyacentes, así que puede ser ejecutada a través de un GET.

Lista de Parámetros, con esta información para cada parámetro

Name: el nombre del parámetro.

Use: In | Out | Both

Type: un tipo de datos o tipo de recurso

Profile: un recurso StructureDefinition que aplica restricciones adicionales al recurso

Documentation – una descripción del uso del parámetro

Esta es una representación pequeña en JSON de la operación \$ihe_pix definida por la Guía de Implementación IHE PIX Mobile.

```
"resource": {
  "resourceType": "OperationDefinition",
  "name": "$ihe-pix",
  "status": "draft",
  "type": [
    "Patient"
  ],
  "parameter": [
    {
      "name": "sourceIdentifier",
      "use": "in"
    },
    {
      "name": "targetSystem",
      "use": "in"
    },
    {
      "name": "_format",
      "use": "in"
    }
  ]
},
```

GET [base]/Patient/\$ihe-pix?sourceIdentifier=[token]] {&targetSystem=[uri]} {&_format=[mime-type]}

Parámetro 1: sourceIdentifier (1..1) Token El parámetro de búsqueda identificador de paciente que será utilizado por el PIX Manager para encontrar identificadores asociados con el recurso Patient.

Parámetro 2: targetSystem 0..1 uri La autoridad de asignación que corresponde a los identificadores que se deben devolver

Parámetro 3: _format 0..1 mime-type El formato requerido de la respuesta

Ejemplo de la Llamada:

[base]/Patient/\$ihe-pix?sourceIdentifier=urn:oid:1.3.6.1.4.1.21367.2010.1.2.300|NA5404

Puede ver una descripción completa de la invocación aquí:

<http://hl7.org/fhir/STU3/operation-patient-match.html>

SearchParameter

Todos los parámetros de búsqueda soportados por los recursos standard FHIR se definen utilizando el recurso SearchParameter.

El recurso incluye toda la información necesaria sobre la manera en la cual una clase de recurso específica puede ser buscado en un servidor FHIR.

- code (cómo se denomina el parámetro en las búsquedas)
- type (número, cadena, token, etc.)
- target (a qué recurso se aplica este parámetro)
- xpath o fhir path (dónde en el recurso se encuentra el contenido relacionado a este parámetro)

Nota: ¿Se está preguntando qué es **'fhir path'**? FHIR necesita un equivalente a XPath para JSON o, mejor aún, algo que pueda referir lógicamente al contenido de un recurso más allá de la sintaxis. De allí surge 'FHIRPath'. Si desea más información acerca de FHIRPath, por favor lea esto: <http://hl7.org/fhirpath/>

Este recurso también se utiliza si quiere soportar búsquedas que no están definidas por el estándar mismo. Para hacer esto, debe crear un nuevo parámetro de búsqueda (SearchParameter) e incluirlo tanto en la declaración de conformidad de servidor como en la guía de implementación del proyecto (además de por supuesto hacerlo funcionar en el servidor para el recurso especificado).

Aquí encontrará un ejemplo de un parámetro especial de búsqueda SearchParameter, definido por HL7 (en este caso es 'buscando por una extension')

<http://hl7.org/fhir/searchparameter-example-extension.html>

Contenido

StructureDefinition

Resumen

StructureDefinition permite crear tipos básicos y recursos, específicamente: todos los recursos FHIR se expresan como un recurso StructureDefinition.

Básicamente es una **cabecera** con información acerca de la estructura en sí: versión de FHIR, editor, recurso base, mapeos, y contexto, más una lista de definiciones de elementos de datos (**DataElement**).

Uso

StructureDefinition se usa para definir perfiles y restricciones sobre los recursos:

- 1) Limitar la cardinalidad
- 2) Cambiar la relación de ValueSet
- 3) Agregar restricciones adicionales
- 4) Agregar extensiones
- 5) Definir mapeos externos y lógicos

Recuerde el principio básico: Mientras hacemos todo esto, no podemos contradecir las definiciones básicas: elementos obligatorios, codificación, etc. Un elemento al cual aplicamos un perfil sigue siendo después de todo un elemento del mismo tipo.

Perfiles sobre StructureDefinition

La lista de definiciones para los elementos específicos puede tener dos formas:

Differential: una lista de elementos que cambian comparado con la definición general. Esto refleja la visión humana del perfil del recurso. Es poco densa (solo contiene la mínima cantidad de información que se modificó)

Snapshot: es el resultado de aplicar el 'differential' al recurso StructureDefinition base de la definición original. Lista todo lo que está permitido en el resutos. Esta vista es usada principalmente por las herramientas.

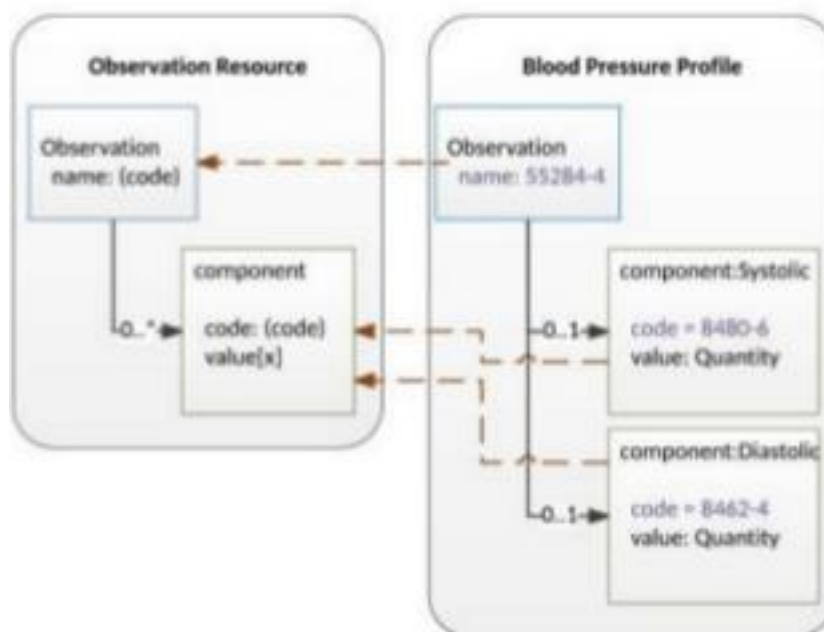
Slicing y Discriminators

Cualquier recurso que permite más de una repetición puede ser restringida con 'slicing'

¿Porqué necesitamos restringir las repeticiones? Algunos casos de uso:

- 1) Se necesitan exactamente dos identificadores de pacientes: identificador nacional de paciente, e identificadores locales de paciente.
- 2) Se necesitan dos observaciones con códigos específicos para cada uno. Ejemplo: Perfil de presión sanguínea: TA Sistólica+TA Diastolica, como se muestra en la figura 3. No necesitamos CUALQUIER recurso Observation, necesitamos dos observaciones específicas, identificadas por el valor de su código.

FIGURE 3 – EJEMPLO DE SLICING – DOS RESULTADOS ESPECIFICOS NECESARIOS COMO COMPONENTES



Ejemplo de 'slicing'

En un pequeño país imaginario, Berzerkistan, el Ministerio de Salud (MOH: Ministry of Health) necesita recibir para su Repositorio Unificado de Salud, una transacción con datos del paciente, organización, médico, encuentro, condición de salud cada vez que una enfermedad transmisible se diagnostique. Los pacientes de Berzerkistan se identifican por su BNI (Berzerkistan National Identifier), pero cómo hubo algunos problemas (repeticiones, asignación incorrecta, etc.), el MOH ha creado también el National Healthcare Identifier (NHIB). Para el recurso Patient, ambos identificadores son obligatorios.

Ahora, como se vería este 'slice' (note como hicimos este 'slice' para el elemento identifier, así que ahora tendremos dos elementos identifier requeridos). En este caso el 'slice' está basado en el atributo identifier.system. El primero de ellos debe estar valorizado como www.berzerkistan.gov/bni. El segundo de ellos debe estar valorizado como www.berzerkistan.gov/nhib

Name	Flags	Card.	Type	Description & Constraints
Patient		0..	Patient	
identifier	Σ			Slice: Unordered, Open At End, by system
identifier	Σ	1..1		
use	⌘Σ	0..0		
type	Σ	0..0		
system	Σ	1..		Fixed Value: www.berzerkistan.gov/bni
value	Σ	1..		Example: 123456
period	Σ	0..0		
assigner	Σ	0..0		
identifier	Σ	1..1		
use	⌘Σ	0..0		
type	Σ	0..0		
system	Σ	1..		Fixed Value: www.berzerkistan.gov/nhib
value	Σ	1..		Example: 123456
period	Σ	0..0		
assigner	Σ	0..0		
name	Σ	1..1		
gender	Σ	1..		
birthDate	Σ	1..		

Y así es como se vería una instancia válida (la parte correspondiente a Patient.identifier)

```

<identifier>
  <system value="www.berzerkistan.gov/bni"/>
  <value value="73928192"/>
</identifier>
<identifier>
  <system value="www.berzerkistan.gov/nhib"/>
  <value value="340323-20-09"/>
</identifier>

```

DataElement

DataElement describe las características que describen una parte de los datos – identificador, nombre, definición, tipos de datos, longitud, set de valores permitidos, notas de uso, etc. Se utilizan dentro de StructureDefinition para construir la lista de elementos.

Extensiones FHIR

En nuestra primera unidad, hemos aprendido como incluir una extension en su instancia de recurso. Recuerde: está compuesto por una url y un valor, la url apunta a la definición de extensiones.

Aquí aprenderemos como DEFINIR una extensión.

Una extensión también se define usando un recurso StructureDefinition.

Extensiones Simples

Una extensión simple es una extensión como hemos visto hasta ahora: con un solo valor.

Veamos ahora un ejemplo de una extension.

En un pequeño país inventado, Berzerkistán, donde el estado exige el pago de impuestos a la salud provista por prestadores privados. La gente sin cobertura ('particular' paga el 21% de impuesto, llamado PPH, sobre cualquiera de los servicios provistos. La gente que ha elegido su cobertura ('voluntarios' pagan el 10.5% de PPH, y la gente que utiliza los servicios de salud provistos por los sindicatos ('obligatorios') no pagan el impuesto. (esto fue decidido luego de una huelga nacional masiva que los sindicatos lideraron en Noviembre de 2010)

Esta es la tabla **PatientTaxSituation** del impuesto PPH de Berzerkistan

Código	Descripción	Impuesto PPH
U	Particular	21%
V	Voluntario	10.5%
M	Obligatorio	0%

Cada paciente tiene asignada una categoría, llamada “PatientTaxSituation”. Es un elemento codificado.

Como no hay ningún ítem en el recurso Patient tal como fue definido por la especificación de FHIR, la Agencia del Gobierno a cargo, decidió crear una extensión de FHIR.

Puede ver la definición de esta extensión en el archivo [SimpleExtensionExample.xml](#)

Algunos ítems a resaltar:

Metadatos acerca de la extensión: nombre, estado, descripción, requerimientos, versión de FHIR.

constrainedType: tipo de elemento. En este caso estamos restringiendo Extension: definiendo una nueva extensión

contextType: donde se utiliza este elemento: en un recurso, en un tipo de datos. En este caso: recurso.

context: ¿A qué contexto se aplica esta extensión? En nuestro caso: **patient**.

Finalmente, la definición incluye el diferencial para la extensión. Las extensiones simples tienen dos elementos: url y value[x]. Cualquier url, y una opción entre algunos tipos de valor (cantidad, cadena, etc.).

Esta extension específica ('el diferencial') tiene:

1. Una url definida:

<http://fhir.hl7fundamentals.org/berzerkistan/StructureDefinition/PatientTaxSituation>

2. Un tipo de valor específico: Coding.

¿Quizás piense que algo falta en esta explicación?

“¿Porqué el gobierno de Berzerkistan cobra impuestos a los servicios de salud?”

Esta no era la pregunta, aunque es una buena pregunta. Pero no la podemos responder aquí.

La pregunta es “¿Dónde está la tabla de códigos para el impuesto PPH? ¿Cómo sabemos que estamos utilizando el código correcto en nuestra instancia para la situación impositiva PPH del paciente?”

Hay una parte del recurso StructureDefinition dónde podemos incluir esto, y un recurso especial en FHIR donde guardar la tabla y apuntarla, llamada **ValueSet**.

La parte de la extension que define donde encontrar el set de valores apropiado para encontrar los códigos se denomina 'Binding'. Esto aplica también a cualquier elemento codificado restringido por la especificación FHIR o un perfil específico.

El ejemplo de ValueSet (la tabla en formato recurso FHIR) que extrae los valores del sistema de codificación original está en el archivo [SmallValueSetExample.xml](#)

Puede encontrar más información acerca de conjunto de valores (ValueSet) y Sistemas de Codificación (CodeSystem), y cómo utilizarlos y construirlos, en la próxima sección (Terminología).

Extensiones Complejas

Las extensiones complejas se definen a través de un url y una lista de elementos. De hecho, se pueden construir recursos enteros utilizando extensiones complejas. Si no hay ningún recurso en FHIR que resuelva sus necesidades, se puede generar extendiendo algo llamado el recurso 'Basic'. Puede extender el recurso 'Basic' con todas las extensiones que necesite (Ver <https://www.hl7.org/fhir/basic.html>) y crear sus propios recursos FHIR.

Puede encontrar un ejemplo de esto aquí: <http://standardhealthrecord.org/fhir/>

Lo único que podemos decirle es que hacer esto hace muy poco a favor de la interoperabilidad global. Por favor no lo haga.

Extensiones modificantes ('Modifier')

Estas extensiones permiten que en una implementación se definan extensiones que DEBEN ser procesados y entendidos por los sistemas que procesan los recursos.

Ejemplos de esto: registrar una instrucción acerca de NO tomar una medicación, usar un recurso Condition para transmitir el hecho que la condición pertenece a otra persona, no al paciente mismo, o declarar que el que un médico en especial NO ESTUVO PRESENTE en un procedimiento.

Así que si su extensión modifica el contenido del recurso, y el recurso NO PUEDE SER PROCESADO EN FORMA SEGURA PARA EL PACIENTE si se IGNORA, entonces su extension debe ser etiquetada como 'Modifier Extension'.

La declaración como 'Modifier Extension' es simple: sólo incluir isModifier=True en el recurso StructureDefinition.

La implementación de las extensiones modificantes es más difícil, porque necesita agregar la información a su sistema, y transferirla entre los sistemas también.

Se incluyen en una instancia usando el elemento modifierExtension en lugar del elemento Extension.

Vinculación de Vocabulario - Binding

Binding es la relación entre un elemento codificado y el set de valores para su validación.

Incluye **'strength'** (¿qué tan estricta es la validación?) y **'valueSetReference'** (¿dónde está la lista de códigos que podemos utilizar?).

Las opciones de vinculación son:

"required": no se pueden utilizar otros códigos en una instancia válida

"example": estos códigos son simplemente un ejemplo.

"preferred": preferimos el uso de estos códigos, pero se pueden utilizar otros

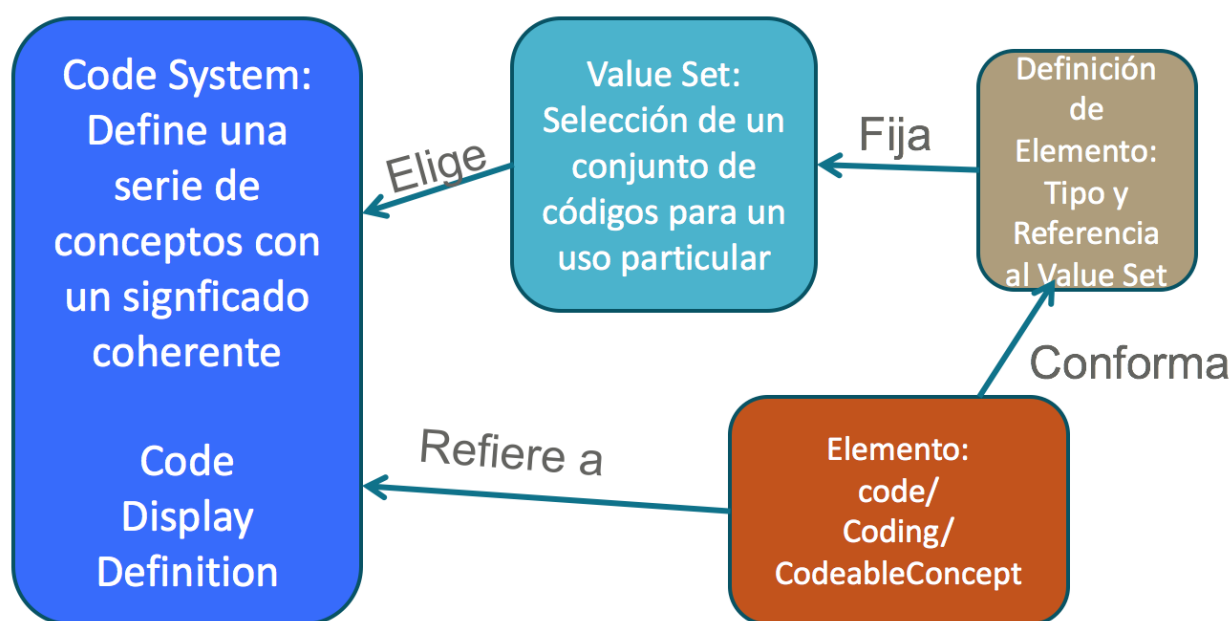
"extensible": puede agregar sus propios códigos a esta lista

```
<binding>
  <strength value="required"></strength>
  <description value="PPH Tax Table"></description>
  <valueSetReference>
    <reference
      value=
        "http://fhir.hl7fundamentals.org/berzerkistan/ValueSet/PatientTaxSituation">
    </reference>
  </valueSetReference>
</binding>
```

Terminología

Estos son los recursos relacionados a terminología y vocabularios. Lo más importante para una implementación de Perfiles es el recurso es ValueSet: puedes almacenar tus conjuntos de valores restringidos en un servidor FHIR, y referir las restricciones de vocabulario a este servidor.

Aquí están todos los recursos terminológicos, y sus relaciones.



La diferencia que tiene esto con las implementaciones anteriores de vocabulario para los estándares de HL7 es que las referencias al vocabulario pueden 'resolverse', así que puede realmente encontrar el vocabulario en el servidor que lo contiene. No es solamente una referencia a un archivo separado por comas, que hay que descargar en su sistema...

Los recursos CodeSystem, ValueSet y ConceptMap en FHIR tienen soporte para operaciones especiales que se utilizan para crear servicios terminológicos.

Nota: ¿Desea saber más acerca de que funciones puede cumplir un servicio terminológico? Por favor lea <https://www.hl7.org/fhir/terminology-service.html>

CodeSystem

Los sistemas de codificación definen un conjunto de códigos con significado, qué códigos existen, y también como se entienden. Los sistemas de codificación también son conocidos como Terminologías, Clasificaciones, Enumeraciones, u Ontologías.

El recurso CodeSystem tiene una URL que lo identifica (no importa donde esté guardado el recurso físicamente), versión y demás metadatos (editor, fecha, etc.)

Este recurso no está previsto para realizar mantenimiento de sistemas de codificación, sino para publicar los metadatos y (opcionalmente) el contenido (conjunto de códigos con su significado), para soportar expansión y validación. Los grandes sistemas de codificación (SNOMED CT, LOINC ICDx) tienen sus propios métodos de publicación y distribución.

Los sistemas de codificación publicados en forma externa a HL7 tienen una URL especial definida para su uso en artefactos FHIR. Puede encontrar la mayoría aquí: <https://www.hl7.org/fhir/terminologies-systems.html>

La publicación como recurso CodeSystem está prevista para los sistemas de codificación más pequeños, creados ad-hoc para proyectos específicos.

Este es un ejemplo de un CodeSystem:

En un pequeño país imaginario, Berzerkistan, el MOH (Ministerio de Salud) decretó que todas las organizaciones o médicos cobren a sus pacientes un porcentaje de sus honorarios por servicios, basada en una clasificación creada por el MOH denominada 'Patient Tax Situation' (Situación Impositiva del Paciente)

Puede encontrar una definición sencilla de un CodeSystem para el Ministerio de Salud de Berzerkistan, que aplica un impuesto a los servicios de salud, aquí: [CodeSystemSimpleExample.json](#)

También encontrará el ejemplo en nuestro servidor FHIR, aquí:

<http://fhir.hl7fundamentals.org/r4/CodeSystem/17582>

Esta es la operación terminológica más sencilla para la clase de recursos CodeSystem: **\$lookup** (no solo es una búsqueda de códigos, puede implicar recuperar y traer la información conceptual del sistema de codificación implicado)

[basefhir]/\$lookup?system=[system]&code=[code]

Puede probarla, buscando el código "M" en este recurso CodeSystem

[http://fhir.hl7fundamentals.org/r4/CodeSystem/\\$lookup?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=M](http://fhir.hl7fundamentals.org/r4/CodeSystem/$lookup?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=M)

ValueSet

Los conjuntos de valores (ValueSet) son la manera de expresar exactamente que valores se aceptan para el par system/value de un elemento codificado. En su guía de implementación o perfil, puede vincular cada elemento codificado con un set de valores específicos. Puede construir un ValueSet de las siguientes formas:

Enumerando los conceptos específicos provenientes de uno o más sistemas de codificación Por Ejemplo: <https://www.hl7.org/FHIR/valueset-doc-section-codes.html>

4.3.1.232.1 Content Logical Definition

This value set includes codes from the following code systems:

- Include these codes as defined in <http://loinc.org>

Code	Display
10154-3	Chief complaint Narrative - Reported
10157-6	History of family member diseases Narrative
10160-0	History of medication use Narrative
10164-2	History of present illness Narrative
10183-2	Hospital discharge medications Narrative
10184-0	Hospital discharge physical findings Narrative
10187-3	Review of systems Narrative - Reported
10210-3	Physical findings of General status Narrative
10216-0	Surgical operation note fluids Narrative
10218-6	Surgical operation note postoperative diagnosis Narrative
10218-6	Surgical operation note postoperative diagnosis Narrative
10223-6	Surgical operation note surgical procedure Narrative

- a) **Definiendo algún criterio** por el cual los conceptos se extraen de uno o más sistemas de codificación original. Ejemplo: <https://www.hl7.org/FHIR/valueset-activity-reason.html>. Este ValueSet extrae “todos los códigos de SNOMED CT donde el concepto es-un 404684003 (Hallazgo Clínico)”.

4.3.1.137.1 Content Logical Definition

This value set includes codes from the following code systems:

- Include codes from <http://snomed.info/sct> where concept is-a 404684003 (Clinical finding)

4.3.1.137.2 Expansion

This expansion generated 19 Apr 2017

This value set has >1000 codes in it. In order to keep the publication size manageable, only a selection (1000 codes) of the whole set of codes is shown

All codes from system <http://snomed.info/sct>

Code	Display	Definition
109006	Anxiety disorder of childhood OR adolescence	
122003	Choroidal hemorrhage	
127009	Spontaneous abortion with laceration of cervix	
129007	Hemiothermia	
134006	Decreased hair growth	
140004	Chronic pharyngitis	
144008	Normal peripheral vision	
147001	Superficial foreign body of scrotum without major open wound but with infection	
150001	Abnormal bladder continence	

Operaciones sobre el recurso ValueSet: \$expand and \$validate

No discutiremos en profundidad todos los temas de terminología en este curso introductorio...pero si nos interesan tres operaciones básicas. La primera fue '\$lookup' para CodeSystem. Ahora revisaremos dos operaciones para ValueSet: \$expand, y \$validate.

Por supuesto estas operaciones son más complejas de lo que parecen aquí, pero las usaremos solamente para mostrarle el concepto de las operaciones, y el hecho que el perfil, las restricciones e inclusive el vocabulario para su guía de implementación, todo puede ser aprovechado a través de las operaciones nativas de su servidor FHIR, si es que las implementa, por supuesto.

Solo por curiosidad, ahora que conoce la tabla PPH del Ministerio de Salud de Berzerkistan realice estas operaciones en nuestro servidor, sin necesidad de usar un cliente FHIR o REST, directamente desde su navegador de internet:

[http://fhir.hl7fundamentals.org/r4/ValueSet/17583/\\$expand](http://fhir.hl7fundamentals.org/r4/ValueSet/17583/$expand)

Esto debería devolver la tabla PPH completa.

[http://fhir.hl7fundamentals.org/r4/ValueSet/17583/\\$expand?filter=MANDATED](http://fhir.hl7fundamentals.org/r4/ValueSet/17583/$expand?filter=MANDATED)

Esto solo debería incluir los códigos para la descripción "MANDATED"

Y finalmente estas dos:

[http://fhir.hl7fundamentals.org/r4/ValueSet/\\$validate-code?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=M&display=MANDATED](http://fhir.hl7fundamentals.org/r4/ValueSet/$validate-code?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=M&display=MANDATED)

Debería obtener un mensaje "**Found Code**" (Código encontrado)

[http://fhir.hl7fundamentals.org/r4/ValueSet/\\$validate-code?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=O&display=OTHER](http://fhir.hl7fundamentals.org/r4/ValueSet/$validate-code?system=http://fhir.hl7fundamentals.org/berzerkistan/CodeSystem/PatientTaxSituation&code=O&display=OTHER)

Debería obtener un mensaje "**Code Not Found**" (Código No Encontrado)

ConceptMap

El recurso ConceptMap permite definir con precisión las relaciones entre conceptos (y sus códigos relacionados) en un sistema de codificación a conceptos en otro sistema de codificación. Este mapeo no es 'absoluto' sino que depende del contexto y el caso de uso (es un mapeo con un objetivo definido).

Este recurso incluye los metadatos para el mapa (origen, destino, versión, autor) y una lista de códigos en el sistema de codificación de origen y su relación con uno o más códigos en el sistema de codificación destino.

Puede ver un ejemplo aquí, donde el tipo de muestra de la tabla 0487 de HL7 V2 (Specimen Type) se mapea a SNOMED CT.

<https://www.hl7.org/fhir/conceptmap-example-specimen-type.html>

Este recurso tiene una operación especial relacionada llamada **\$translate**, que permite a un servidor devolver la equivalencia en un sistema de codificación a códigos que tiene en un sistema de codificación de origen.

Este es un ejemplo de esta operación:

```
GET [base]/ConceptMap/$translate?system=http://hl7.org/fhir/composition-status
&code=preliminary&valueSet= http://hl7.org/fhir/ValueSet/composition-status
&target=http://hl7.org/fhir/ValueSet/v3-ActStatus
```

NamingSystem

Este recurso representa un espacio de nombres (Namespace) que genera identificadores únicos para conceptos (clasificaciones, sistemas de codificación, conjuntos de valores) o entidades (gente, dispositivos, ordenes, etc.)

Puede usar uno de los elementos `uniqueId.value` del recurso NamingSystem para su uso en 'system' en sus elementos codificados o de tipo `identifier` (identificadores de entidades).

NamingSystem incluye el nombre, tipo (CodeSystem o Identifier), responsable, editor, contexto de uso, y todos los identificadores únicos para el sistema.

Puede explorar un ejemplo de NamingSystem aquí:

<https://www.hl7.org/fhir/namingsystem-example-id.xml.html#DomainResource.text.div-end>

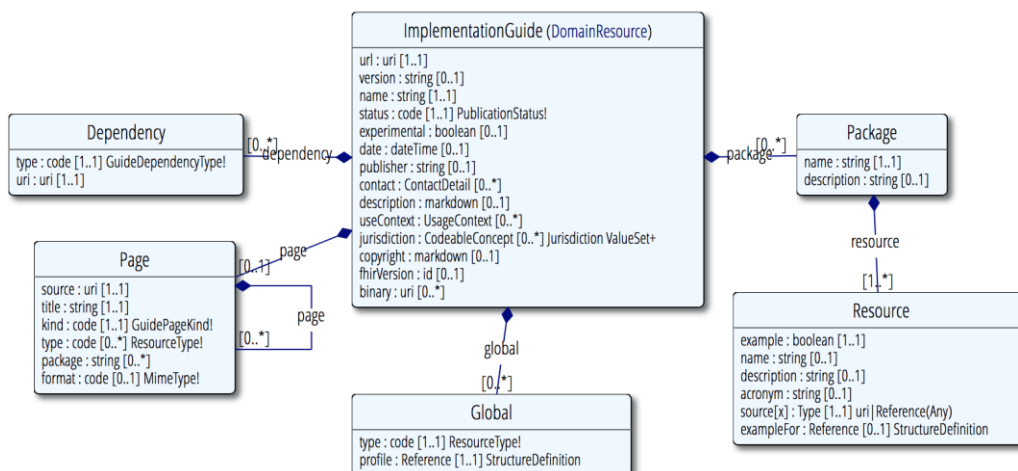
Publicación

ImplementationGuide

Un recurso ImplementationGuide es un paquete que incluye un conjunto de recursos StructureDefinition, NamingSystem, OperationDefinition, ValueSet y todo el resto de recursos de conformidad, previsto para su publicación. Contiene los metadatos de la guía: nombre, estado, versión de FHIR en la cual se base, y también todas las páginas en formato HTML o markdown con la parte ‘narrativa’ de la guía de implementación (escenarios, casos de uso, detalles de las interacciones, etc.). Esto es lo que contiene una guía de implementación típica:

- 1) Título
- 2) Tabla de Contenido
- 3) Información del Documento
- 4) Introducción a la guía
- 5) Principios y antecedentes
- 6) Casos de uso y requerimientos funcionales
- 7) Consideraciones de diseño
- 8) Contenido del paquete
- 9) Guía de seguridad y privacidad
- 10) Lista de todos los artefactos utilizados

Los ítems 1 a 9 son solo **Páginas**: html o markdown que ud. necesita crear, utilizando cualquier editor que prefiera, y los artefactos de formateo auxiliares (archivos css, js, img). El ítem 10 es una lista de todos los recursos sobre los que se realizaron restricciones, agrupados en **paquetes**. También puede incluir **dependencias** de otras guías de implementación. Puede construir su guía de implementación tomando como base recursos creados por otra guía de implementación (recuerde que las guías de implementación también se almacenan como recursos FHIR, así que esta dependencia es realmente computable).



Procesando el recurso ImplementationGuide

Para crear y publicar una guía de implementación una vez armado el recurso ImplementationGuide, el proceso es un poco complejo:

Algunas opciones: Guarde todos sus recursos y narrativa en Simplifier y pídale a Simplifier que construya su guía de implementación. Quedará como ésta:

The screenshot shows the top navigation bar of the eHealth Ontario website with the title "Provincial Client Registry (PCR) FHIR Implementation Guide - v1.0". The navigation menu includes: Introduction, Business Context, Implementation Guidance, Profiles & Operations, Conformance Requirements, Terminology, and Downloads. Below the navigation bar, the "Introduction" section is highlighted. The "Background" subsection explains that the PCR is a repository for patient demographics and identifiers in Ontario, enabling unique identification and merging of records. It also lists the standards supported: Pan-Canadian HL7 V3, HL7 V2, and IHE PIX/PDQ. A list of four operations is provided: 1. IHE PIXm - Patient Identifier Cross-reference, 2. Patient EMPI match, 3. Patient Read, and 4. Patient Search. The page concludes by stating that additional sections provide documentation on the use of the FHIR standard and how errors are handled.

- Corra la herramienta de publicación de guías de implementación FHIR [FHIR IG Publishing Tool] (Basada en Java/Ruby): La ventaja de esta opción es que su guía quedará exactamente como las que publica HL7. Si está interesado en explorar esta opción, hay una pagina de la wiki que explica como instalar y utilizar la herramienta: http://wiki.hl7.org/index.php?title=IG_Publisher_Documentation y un powerpoint con una descripción del proceso, paso a paso, en inglés, aquí: <https://goo.gl/6hB1ZP>

Validación de Recursos contra un Perfil

Declaración de Conformidad en Instancias de un Recurso

Para declarar conformidad para un perfil específico en una instancia de un recurso FHIR, se debe incluir un tag en particular “**profile**”, en la sección de metadatos de la instancia del recurso, refiriendo a la guía de implementación o perfil. Se puede incluir más de un perfil para indicar conformidad con varios a la vez.

Ejemplo:

```
<meta>

  <profile

    value= "http://fhir.server.org/ImplementationGuide/HealthCareClaim"

  </profile>
</meta>
```

Validación de recursos

En principio, los recursos pueden ser validados contra la especificación original (el estándar FHIR), contra una guía de implementación (conformidad) o contra algún perfil específico (conformidad).

Se puede probar la conformidad de un recurso (validar) de varias maneras:

- 2) Usando el Esquema XML (solo para instancias expresadas en XML), básicamente lo que ha hecho en las primeras unidades.
- 3) Usando el validador de FHIR - es un archivo Java JAR que puede descargar desde aquí: <https://www.hl7.org/fhir/validator.zip>
- 4) Solicitando a un servidor FHIR a través de la operación especial \$validate. Es la manera más sencilla de ejecutar esta operación, si es que el servidor la soporta (no todos los servidores FHIR lo hacen).

Ejemplo de Operación \$validate

POST [base]/**Patient**/\$validate?profile=<http://hl7.org/fhir/StructureDefinition/daf-patient>

El recurso <Patient> a validar debe estar en el cuerpo de la operación POST.

Luego de validar el recurso, el servidor devolverá un recurso OperationOutcome con los problemas hallados.

Recuerde que no todos los servidores soportan la operación \$validate, aunque algunos de los servidores de prueba públicos la soportan, solo pueden validar contra perfiles registrados en el mismo servidor.

Todos estos métodos de validación son incompletos. Sólo pueden validar los aspectos ‘computables’ de la conformidad.

Algunas reglas narrativas solo pueden ser verificadas a través de una inspección manual de los recursos. El método a través de esquema/schematron es el menos completo, ya que no verifica la terminología.

La última palabra siempre la tiene la inspección manual, y la comparación con los requerimientos relevantes de las guías de implementación.

Herramientas para Crear Perfiles

Puede crear recursos de conformidad de la misma manera que crea cualquier otro recurso, editando XML o JSON, pero es un proceso complicado.

Se han desarrollado algunas herramientas específicas para crear recursos de conformidad FHIR (y continúan evolucionando).

Estas son algunas de las herramientas. Ninguna de ellas es 'la herramienta oficial de HL7 para crear perfiles'. Los grupos de trabajo solían hasta no hace mucho tiempo generar perfiles y artefactos destinados a las votaciones de los estándares utilizando una herramienta denominada Microsoft Excel (!)

Importante: Este curso no recomienda ningún producto, herramienta o compañía. Solo le mostramos como aprender más sobre este tema, e incluimos algunas capturas de pantalla para que vea como son las herramientas

Es recomendable que descargue alguna de las herramientas y las pruebe (o directamente las use si son basadas en web)

Forge

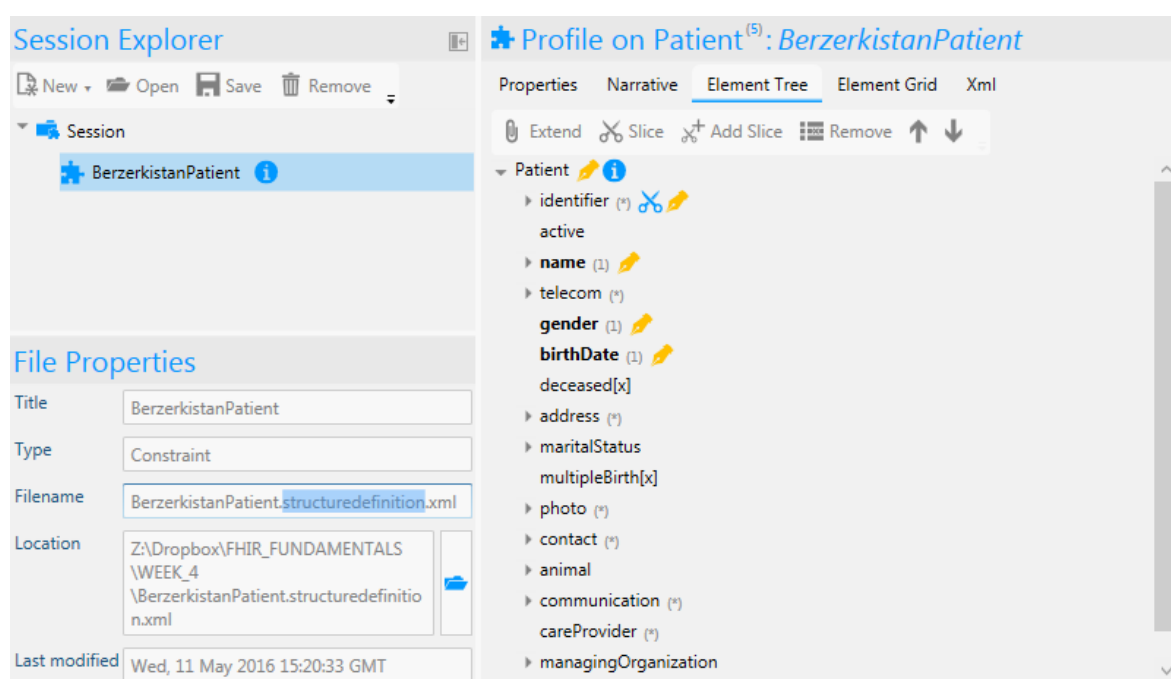
Sitio Web: <https://fire.ly/forge/>

Creada y Mantenido por: Firely, The Netherlands

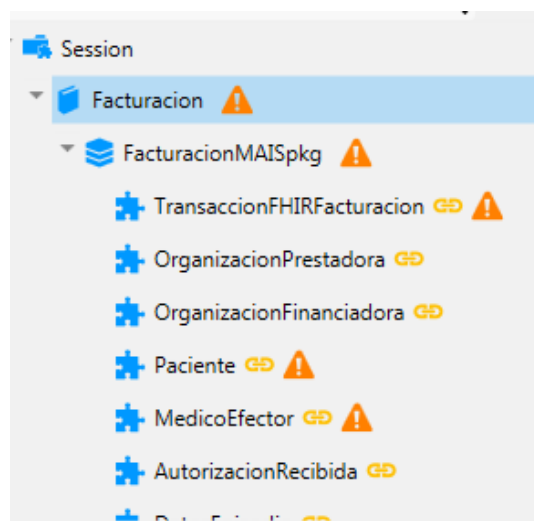
Condiciones: Free Download, requires registration. Runs only in Windows 7 +

Descripción: editor de recursos de conformidad para perfiles, restricciones, extensiones, slices, y vinculación de vocabulario. Puede ser conectada a Simplifier o cualquier otro servidor que almacene recursos de conformidad (recuerde que los perfiles pueden ser almacenados en cualquier servidor FHIR)

Explorador y Editor de Perfiles de Forge



Explorador de Guías de Implementación de Forge



Trifolia

Sitio Web: <https://trifolia.lantanagroup.com>

Creado y mantenido por: Lantana Group, US

Condiciones: Requiere registración, gratis para miembros de HL7, basado en web

Descripción: Originalmente pensado para crear guías de implementación HL7 CDA R2. La funcionalidad para FHIR fue agregada en 2015. Permite editar guías de implementación, perfiles y terminología (conjuntos de valores y sistemas de codificación)

EHO FHIR eConsult

FHIR DSTU2 HL7 Draft

Templates/Profiles Notes Primitives Audit Trail

Search

Bundle

EHO eConsult Bundle <https://ehealthontario.on.ca/eConsult/FHIR2/StructureDefinition/bundle>

Visor de Perfiles de Trifolia

EHO eConsult Bundle Back to List Copy Bulk Copy New Version

<https://ehealthontario.on.ca/eConsult/FHIR2/StructureDefinition/bundle>

EHO FHIR eConsult Draft Bundle Open HL7 borna.jafarpour (HL7)

Constraints Relationships StructureDefinition JSON StructureDefinition XML

1. **SHALL** contain exactly one [1..1] **type** (CONF:3248-191).
2. **SHALL** contain exactly one [1..1] **total** (CONF:3248-196).
3. **MAY** contain zero or more [0..*] **entry** (CONF:3248-190).
 1. The entry, if present, **SHALL** contain exactly one [1..1] **fullUrl** (CONF:3248-193).
 2. The entry, if present, **MAY** contain zero or one [0..1] **resource** (CONF:3248-194).

Visor y Editor de Restricciones de Trifolia. Una restricción sobre un recurso Bundle

Simplifier

Sitio Web: <https://www.simplifier.net/>

Creado y mantenido por: Firely, The Netherlands

Descripción: Pensado como registro global para perfiles FHIR. Almacena todo tipo de recursos de conformidad, y permite edición y búsqueda. Trabaja en conjunto con Forge (mismo proveedor). Es gratis para usar en proyectos personales. El registro global oficial <https://registry.fhir.org/> (mantenido por HL7 International y la FHIR Foundation) está basado (en su back-end) en Simplifier.

The screenshot shows the Simplifier.NET website. On the left, there is a 'FILTER' sidebar with sections for 'Search' (a search bar), 'Published' (radio buttons for 'All', 'Yes', 'No'), and 'Publication Categories' (checkboxes for 'StructureDefinitions', 'ValueSets', 'Extensions', 'SearchParameters', 'Conformances', 'OperationDefinitions'). The main content area displays a list of 'ValueSet' resources. The first entry is 'C-CDA Problems Value Set' with a 'Published' status and a last update of 10/2/2015 1:56:55 PM. The second entry is 'C-CDA Vital Sign Result' with a 'Published' status and a last update of 10/2/2015 1:57:00 PM. The third entry is another 'C-CDA Vital Sign Result' with a last update of 10/2/2015 1:56:07 PM. The fourth entry is 'Common UCUM units' with a last update of 10/2/2015 1:56:50 PM. At the top, there is a search bar and 'SIGN UP' and 'LOG IN' buttons.

Formulario de búsqueda de recursos de Simplifier

The screenshot shows the search form for the 'Encounter' resource. At the top, there are filters for 'type' (Profile on Encounter), 'status' (published), 'version' (none), and 'FHIR' (1.0.1). Below these is a 'Canonical' field with the URL 'http://hl7.org/fhir/StructureDefinition/daf-encounter'. The main content area is divided into tabs: 'Overview', 'Table', 'XML', 'JSON', and 'History'. The 'Table' tab is selected, showing a table with columns: 'Name', 'Flags', 'Card. Type', and 'Description & Constraints'. The table lists various components of the Encounter resource, including 'Encounter', 'meta', 'implicitRules', 'language', 'text', 'contained', 'extension', 'modifierExtension', 'identifier', 'status', 'statusHistory', and 'extension'. Each row provides details on the component's flags, cardinality, type, and description.

Visor de Recursos de Conformidad de Simplifier

clinFHIR

Sitio Web: <http://clinfhir.com/>

Creado y Mantenido por: Dr. David Hay, HL7 New Zealand

Descripción: Editor de recursos para usuario no técnicos, editor para perfiles y extensiones, herramienta para comentar perfiles.

Puede seleccionar su servidor de pacientes, perfiles y terminologías y crear sus instancias basados en ellos. clinFHIR tiene muchas otras funcionalidades que puede probar Ud. mismo.

Resource Creator

Patient: John Smith (461) [Details](#)
Profile: StructureDefinition/Observation

[Validate](#) [Park](#) [Save](#) [Cancel](#)

Resource Navigator

- Observation
 - status
 - subject

Element Details: Observation

Child Elements

identifier	0..*	Identifier
status	1..1	code
category	0..1	CodeableConcept
code	1..1	CodeableConcept
subject	0..1	-> Patient <ul style="list-style-type: none">-> Group-> Device-> Location-> Encounter
encounter	0..1	
effective[x]	0..1	dateTime

Resource: Json

Sorry, there were some issues

Severity	Code	Details/Diagnostics
error		Error in component.where(code = \$context.code).empty() at 32: Found , expecting a valid token name (0000000000BCE0E1){FHIRServer.exe} [0000000000FCF0E FHIRPath.TFHIRExpressionEngine.parseExpression + \$381 (0000000000040E9B){FHIRServer.exe} [00000000000441E9E System.SysUtils.RaiseExceptObject + \$2B (000000000000FD66){FHIRServer.exe} [0000000000410D66

Editor de Recursos de clinfhir

Find Extensions for Patient

[Cancel](#) [Select](#)

[Query Parameters](#) [Results](#) [results-table](#)

Place of Birth for patient
The registered place of birth of the patient. A system may use the address.text if they don't store the birthPlace address in discrete elements.

Target anatomic location or structure
Record details about the anatomical location of a specimen or body part. This resource may be used when a coded concept does not provide the necessary detail needed for the use case.

Conformance expectation
Defines the level of expectation associated with a given system capability.

Url <http://hl7.org/fhir/StructureDefinition/birthPlace>
Name Place of Birth for patient
Description: The registered place of birth of the patient. A system may use the address.text if they don't store the birthPlace address in discrete elements.
Context Patient
Publisher Health Level Seven International (FHIR Infrastructure)
DataType [address](#)

```
{
  "resourceType": "StructureDefinition",
  "id": "birthPlace",
  "meta": {
    "versionId": "1",
    "lastUpdated": "2016-03-28T06:16:54Z"
  },
  "url": "http://hl7.org/fhir/StructureDefinition/birthPlace",
  "name": "Place of Birth for patient",
  "status": "draft",
  "publisher": "Health Level Seven International (FHIR Infrastructure)",
  "contact": [
    {

```

Visor/Constructor de extensiones de clinfhir

©2016-18 HL7 Argentina & HL7 International

V3.

Page 38

Snapper

Sitio Web: <https://ontoserver.csiro.au/snapper2/licence.html?source=author>

Condiciones: Uso gratuito, requiere aceptar las condiciones de uso (para Author, Mapper y SNO-MED CT).

Creado y mantenido por: The Australian e-Health Research Centre, CSIRO

Descripción: Editor y visor de terminología FHIR (recursos ValueSet, CodeSystem y ConceptMap).

Code	Display	Definition
U	Uncovered	Patient pays 21%
V	Voluntary	Patient pays 10.5%
M	Mandated	Patient pays 0%
Code that identifies the c...	Text to display to the user	Formal definition

Ud. puede importar, editar y exportar o publicar sus sistemas de codificación, conjuntos de valores y mapas de conceptos usando Snapper.

Algunas guías de implementación y perfiles relevantes.

En esta sección listaremos algunos perfiles que son importantes y que Ud. necesita tener en cuenta si piensa implementar FHIR (y realmente conectarse con alguien). No todos son guías de implementación ‘tradicionales’ como describimos en esta unidad, pero todos son relevantes.

De nuevo, no recomendamos ningún producto o empresa comercial, solo tratamos que Ud. sepa lo que está ocurriendo y donde encontrar los materiales que necesita para participar.

Proyecto Argonaut

Sitio Web: http://argonautwiki.hl7.org/index.php?title=Main_Page

Descripción: El proyecto Argonaut es una iniciativa del sector privado en USA para desarrollar una API basada en FHIR para facilitar que se compartan los registros electrónicos de salud.

Sponsors: Accenture, Athenahealth, Beth Israel Deaconess Medical Center, Cerner, Epic, Intermountain Healthcare, Mayo Clinic, MEDITECH, McKesson, Partners Healthcare System, Boston Children’s Hospital, Surescripts, The Advisory Board Company

Componentes: Acceso seguro a través de Smart App -> Consulta de Datos / Documentos

Guía de Implementación “Argonaut Data Query” – Consulta de datos

(<http://hl7.org/fhir/us/core/index.html>) para acceso seguro de pacientes y médicos a información de salud. Tiene perfiles para acceder a registros de Paciente en cualquier hospital (que haya adoptado la guía), y luego información granular acerca de Alergias, Plan y Equipo de Cuidado, Condiciones, Dispositivos, Pruebas y Reportes de Laboratorio Clínico, Vacunas, Medicación, Procedimientos, Signos Vitales, Tabaquismo y Documentos.

Guía de Implementación “Argonaut Provider Directory”

(<http://www.fhir.org/guides/argonaut/pd/>) para directorio de prestadores (incluye buscar médicos e instituciones)

Hay más guías de implementación Argonaut siendo creadas: Agendamiento de Turnos y CDS-Hooks - permitir soporte a la decisión clínica a través del intercambio de recursos FHIR.

Adopción:

Quizas esta sea la guía más relevante en términos de adopción. Fue adoptada no solo por sus sponsors, sino que también por Apple, que adoptó la versión basada en FHIR R2 de esta guía para su módulo 'Salud' (embebido en iOS 11.3). Apple declara tener 40 hospitales / grupos conectados (hasta el 20 de marzo de 2018)



Ver más: <http://www.healthcareitnews.com/news/how-apples-health-records-could-reshape-patient-engagement>

También fue silenciosamente adoptado por Google a través de su subsidiaria Apigee, en un producto llamado apigee-health-apix (Managed API for Healthcare).

Ver más: <https://apigee.com/about/tags/argonaut>

El objetivo de Apigee-health-apix es permitir a su propia aplicación que brinde servicios FHIR/Argonaut pero manejados a través de un portal Apigee.

apigeeHealth APIx

APIS BLOGS COMMUNITY EVENTS APP GALLERY REGISTER LOGIN

Patients Search

GET Patients Search

Returns Patients based on filter criteria OR Provides provision for pagination (stateid, page, page_size, _pretty & _bundletype are query parameters used for Paging).
FHIR documentation for Patient.

OAuth 2.0 Set...

Send Reset

Request Response

Make a request and see the response.

Resource URL	Header Parameters	Query Parameters
Environment	URL	
Live	https://fhirsandbox-prod.apigee.net/fhir/v1/Patient	

FHIR APIs

- Authorization API
- Patient API
- AllergyIntolerance API
- CarePlan API
- Condition API
- Coverage API
- DiagnosticOrder API
- DiagnosticReport API
- DocumentReference API
- Encounter API
- Immunization API
- MedicationAdministration API
- MedicationDispense API
- MedicationOrder API
- MedicationStatement API

IHE-on-FHIR

A pesar que IHE todavía publica sus guías de implementación como archivos PDF gigantes, finalmente está adoptando el estándar FHIR para alguno de sus perfiles, para alinearse con las necesidades de la industria y porque finalmente FHIR está lo suficientemente maduro para ser considerado.

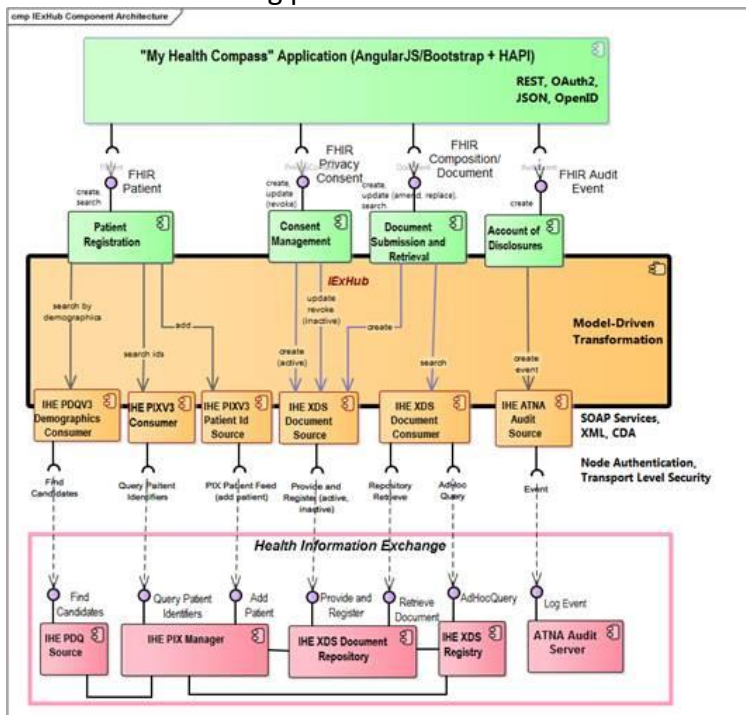
Todos los perfiles IHE basados en FHIR figuran aquí: <https://wiki.ihe.net/index.php/Category:FHIR>, así que no incluiremos un vínculo a cada uno de ellos.

El primer conjunto de perfiles fue un paso para incorporar al mundo FHIR los perfiles ‘legados’ de IHE (los más importantes):

- Mobile Access to Healthcare Documents (MHD) – XDS pero sobre FHIR
- ATNA / Restful Query for ATNA – Para registros y consulta de auditoría
- Patient Demographics Query for Mobile / Patient Identifier Cross Ref for Mobile - PIX/PDQ pero sobre FHIR
- Query for Existing Data for Mobile (QEDm) – QED sobre FHIR

Si su empresa tiene base instalada sobre perfiles IHE y quiere abrir una ‘puerta’ FHIR o ‘fachada FHIR’ (que así es como se denomina), estas son las especificaciones que estaba buscando.

Foto tomada del blog post de John Moehrke sobre IHE-on-FHIR



Smart-On-FHIR

Sitio Web: <https://smarthealthit.org>

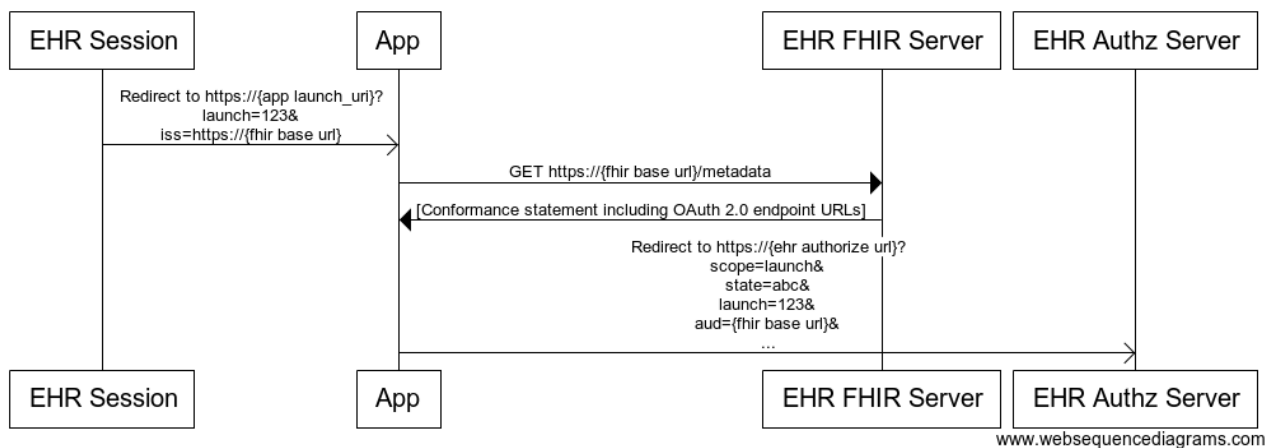
Sponsors: Boston Children's Hospital Computational Health Informatics Program y Harvard Medical School Department for Biomedical Informatics.

Componentes: Smart-On-FHIR fue al principio denominado "Smart Healthcare", y su objetivo fue crear 'una plataforma de aplicaciones para salud'. Cuando estaban en el medio del desarrollo de la API, emergió FHIR, y lo adoptaron de todo corazón. La idea principal es conectar un mercado de apps para salud con cualquier sistema de historia clínica de cualquier organización (Puede ver la galería de aplicaciones aquí: <https://apps.smarthealthit.org/>)

Es un conjunto de especificaciones para: **Datos Estructurados:** básicamente, FHIR +Argonaut , **Permisos y Alcances:** OAuth2 (qué paciente/médico/encuentro) está siendo accedido 'en contexto' en la HCE, y **Autenticación:** Login Simplificado a través de OpenID e **Integración a nivel de interface de Usuario:** Corre dentro de widgets de navegador o marcos in-line.

Pero lo que proveen es mucho más que una guía de implementación: crearon un conjunto de bibliotecas de código abierto que permite a los creadores de apps (HTML5/Javascript/iOS/Python) conectarse a servidores FHIR. Estas bibliotecas cubren las funcionalidades mencionadas en el párrafo anterior.

Adopción: Boston Children's Hospital, DukeMedicine, CoxHealth, Geisinger, Hospital Corporation of America, Intermountain Healthcare, Ochsner, Allscripts, Cerner, Epic, Apple.



Guías de Implementación Oficiales de FHIR

Todas las guías de implementación votadas y publicadas por HL7 International y la FHIR Foundation pueden hallarse aquí: <http://www.fhir.org/guides/registry>

Todos los perfiles (restricciones sobre recursos, extensiones, conjuntos de valores) votados y publicados por HL7 International pueden hallarse aquí: <https://registry.fhir.org/>

Guías de Implementación Regionales Públicas

Las implementaciones regionales públicas (creadas por afiliados a HL7 de Francia, Finlandia, Noruega, UK, etc.) pueden ser encontradas en Simplifier, así: <https://simplifier.net/search?category=ImplementationGuide>

Puede haber más guías de implementación FHIR, pero no fueron publicadas o son privadas, así que no podemos brindar más información sobre ellas.

Guías de Implementación de Salud Digital en Argentina

Esta es una serie de guías desarrolladas en Argentina para la estrategia de salud digital

<https://simplifier.net/saluddigital.ar>

Más información sobre perfiles FHIR

La creación de perfiles FHIR es un tema complejo. Hay un sitio con más información acerca del tema si le interesa, publicado por la compañía holandesa Firely, que también publica Simplifier y Forge.

Se denomina 'FHIR Profiling Academy'

<https://simplifier.net/guide/ProfilingAcademy/Modules>

Alguno de los artículos está enfocado en cómo usar Forge para generar perfiles, así que si no está usando Forge, puede que no lo encuentre interesante. De todas maneras, son una buena fuente de información acerca de que puede ser restringido en FHIR, y cómo.

La actividad de esta semana

Trataremos de

Parte 1: validar instancias de pacientes de Berzerkistan contra un servidor FHIR.

Parte 2: crear una extensión con información sobre pacientes para los dispositivos UWearMe

Si llegó hasta aquí, quiere decir que está llegando al final del curso. ¡Adelante!

Resumen de la Unidad y Conclusiones

En esta unidad hemos discutido los conceptos básicos de conformidad y guías de implementación utilizando FHIR.

Creamos perfiles de conformidad para **restringir los estándares a nuestros casos de uso o escenarios específicos**, y documentar nuestras decisiones, permitiendo la validación automatizada de conformidad en caso de ser posible.

En FHIR puedes restringir: **elementos dentro de recursos** (opcionalidad, repeticiones), **vocabulario** – códigos en elementos codificados - (vocabulario específico, incluyendo la representación física de los sistemas de codificación y conjuntos de valores) , **conducta** (que métodos CRUD o no-CRUD y que búsquedas soporta nuestro servidor FHIR), y también que **extensiones** son necesarias en cada nivel (tipo de datos, recursos), y si son obligatorias o no.

Las restricciones en FHIR se implementan creando recursos FHIR. Estos son algunos de los recursos FHIR utilizados para documentar restricciones: ImplementationGuide, CapabilityStatement, OperationDefinition, SearchParameter, StructureDefinition, ConceptMap, NamingSystem, DataElement, ValueSet, CodeSystem.

Algunas de las herramientas disponibles para crear y probar perfiles son Trifolia, Forge, Simplifier (publicación) y clinFHIR. No todas las herramientas soportan edición de todos los tipos de recursos de conformidad.

La validación de conformidad puede realizarse utilizando el XML Schema (validación simple) o utilizando la función de validación de algunos servidores FHIR. Hemos explorado también algunas guías de implementación relevantes o muy utilizadas.

Todavía queda mucho por hacer en FHIR acerca de la conformidad, pero podemos empezar a soñar con guías de implementación computables y herramientas adecuadas para generarlas.

Material de Lectura Adicional

Información sobre FHIR

Hay muchos lugares donde obtener información sobre FHIR.

- La especificación está disponible en www.hl7.org/FHIR. Con todos los vínculos y muy fácil de seguir. Es muy recomendable tener acceso a la especificación mientras lee esta unidad, ya que hay muchas referencias cruzadas, particularmente para los detalles acerca de los aspectos más complejos de FHIR.
- Todos los temas de esta unidad están documentados profundamente en la especificación de FHIR. Si hay algo que no está explicado completamente, la fuente de verdad primaria es la especificación.
- La página principal de la wiki de HL7 para FHIR está aquí: <http://wiki.hl7.org/index.php?title=FHIR>. La información que contiene es más para aquellos que desarrollan recursos, pero es igual muy interesante. Parte de la información es histórica y puede no reflejar el estado actual de la especificación
- El equipo usar 'stack overflow' como un lugar para responder preguntas sobre la implementación. También puede utilizar la mesa de ayuda de HL7 si es miembro, o chat.fhir.org (un chat donde más de 2000 participantes realizan y contestan preguntas sobre FHIR)