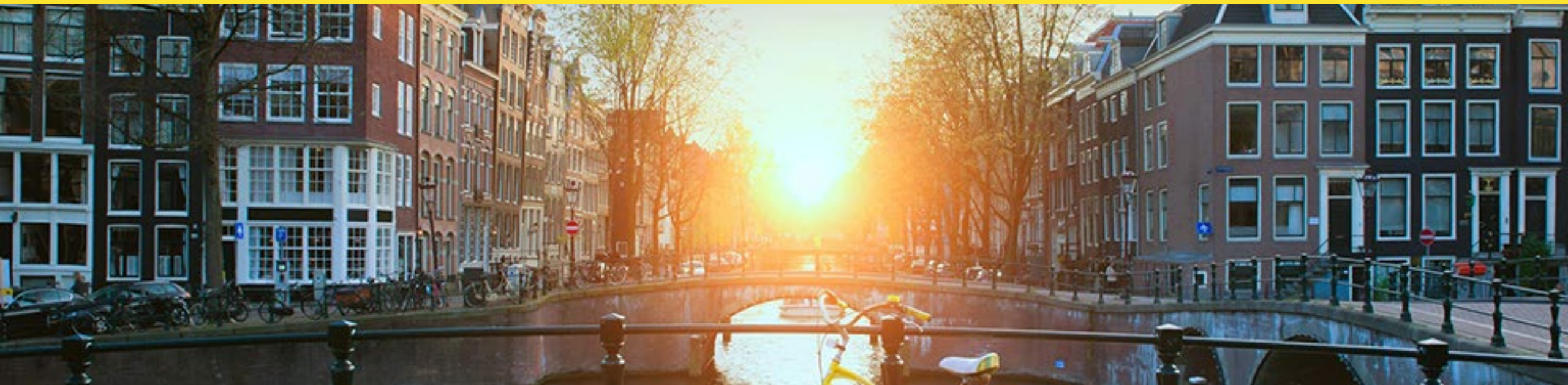


FHIR Testing and Certification

Richard Ettema



Amsterdam, 14-16 November | @HL7 @FirelyTeam | #fhirdevdays18 | www.fhirdevdays.com

Presented by

- **Name:** Richard Ettema
- **Position:**
 - Lead Consultant, AEGIS.net, Inc.
 - FHIR® Certified Implementer
- **Background:**
 - 35+ years IT industry experience
 - 15+ years leading HIT development/implementation efforts
 - 5+ years contributing to the HL7® FHIR® specification (focus on testing)
 - Sr. Architect / Lead Developer for the Touchstone Project
 - Author of the AEGIS WildFHIR public test server and client



Session Goals

- FHIR Testing Framework – Overview
- Program Testing and Certification Support
 - Implementation Guides
 - Validation
- Program Support – Touchstone Features
 - Organization Groups
 - Security and Privacy
 - FHIRSandbox, ...
- Nictiz MedMij Use Case (Quick) Review

FHIR Testing Framework

To ensure interoperability between applications claiming conformance to the FHIR specification, a testing framework has been established.

- This framework defines a TestScript resource as a natural language, computable format of a test case
<http://hl7.org/fhir/testscript.html>
- This framework also defines the test execution workflow that a FHIR Test Engine must follow
<http://hl7.org/fhir/testing.html#execution>

FHIR Testing Framework – Test Execution Workflow

<http://hl7.org/fhir/testing.html#execution>

★ Pre-Processing

- Determine whether the TestScript interactions match the capabilities of the system under test.
- Setup Execution
 - Optional operations that prepare the system under test for subsequent test execution.

★ Test Execution

- Execute and record each test's operations and assertions.
- Teardown Execution
 - Optional operations that revert the system under test to its pre-test state.
- Post-Processing
 - Cleanup of test execution data and collection of test results.

FHIR Testing Framework – Pre-processing

The TestScript execution workflow begins by determining if the TestScript is appropriate for the system under test (SUT).

- The TestScript interactions are evaluated against the SUT capabilities.
 - If the capabilities are supported, then the test engine proceeds with the TestScript execution.
 - Otherwise, the test engine will provide a comparison of the SUT capabilities and allow the TestScript to either be **skipped** or **executed**.

Why allow executed?

The user may wish to evaluate their test system's functional gaps.

FHIR Testing Framework – Test Execution

Test execution consists of two action types:

- **operation**: The FHIR RESTful API interaction to be executed.
 - All FHIR operations including the extended operations framework are supported
 - ⚡ Always executed regardless of whether an assert follows
- **assert**: A rule that is evaluated against the results of an operation.
 - Acts on the current, most recently executed operation context
 - Is a condition, rule or expression evaluated against the executed operation context
 - If true, the assert passes

Touchstone Demo – Test Execution



TOUCHSTONE

AEGIS.net

Program Testing and Certification Support

- Program level support for testing and certification is defined and based on a FHIR Implementation Guide
- A FHIR Implementation Guide:
 - Provides human-readable information (narrative)
 - Contains formatted text, pictures, attachments, ...
 - Explains to developers how to implement specific use cases
 - Is **published** to a local/public website or PDF
 - Provides conformance artifacts for **validation** of data contents

FHIR Implementation Guides – Publication

A FHIR Implementation Guide (IG) can be published:

- Using the FHIR publication tool, **igpublisher**, from the FHIR specification
 - [http://wiki.hl7.org/index.php?title=IG Publisher Documentation](http://wiki.hl7.org/index.php?title=IG_Publisher_Documentation)
 - Examples – **US Core**, AU Base, etc.; see <http://www.fhir.org/guides/registry>
- Using SIMPLIFIER.NET FHIR Registry
 - <https://simplifier.net/guides>
 - Examples – Canadian eReferral, **Nictiz MedMij** (with separate wiki)

FHIR Implementation Guides – US Core

<http://hl7.org/fhir/us/core/index.html>



US Core Implementation Guide (Release 1.0.1)



[Home](#)
[General Guidance](#)
[US Core Profiles](#)
[Extensions](#)
[Terminology](#)
[Search Parameters](#)
[Capability Statements](#)
[Security](#)
[Downloads](#)

US Core Implementation Guide

This is the Current officially released version of the US Core FHIR Implementation Guide US Core Implementation Guide, based on [FHIR Version 3.0.1](#). See the [Directory of published versions](#) ↗

This Implementation Guide is published by the [HL7 US Realm Steering Committee](#). To submit formal requests for changes use [GForge](#). When submitting a tracker, select 'US Core' for the Specification.

Introduction

The US Core Implementation Guide is based on [FHIR Version 3.0.1](#) and defines the minimum conformance requirements for accessing patient data as defined by the [Argonaut](#) pilot implementations and the [ONC 2015 Edition Common Clinical Data Set \(CCDS\)](#). These profiles are intended to be the foundation for future US Realm FHIR implementation guides. In addition to Argonaut, they are used by [DAF-Research](#), [QI-Core](#), and [CIMI](#). Under the guidance of HL7 and the HL7 US Realm Steering Committee, the content will expand in future versions to meet the needs specific to the US Realm.


These requirements were originally developed, balloted, and published in FHIR DSTU2 as part of the [Office of the National Coordinator for Health Information Technology \(ONC\)](#) sponsored [Data Access Framework \(DAF\)](#) project. For more information on how DAF became US Core see the [US Core change notes](#).

Contents

- [Introduction](#)
- [US Core Actors](#)
- [US Core Profiles](#)
- [US Core Conformance Requirements](#)

US Core Actors

FHIR Implementation Guides – Nictiz MedMij




SIMPLIFIER.NET

<https://simplifier.net/nictizstu3-zib2017>

SNIPPET FEEDBACK


PROJECT OF Nictiz



Nictiz STU3 Zib 2017

Nictiz repository of FHIR STU3 conformance for HCIM Release 2017.
Includes MedMij and HL7 NL.

- Introduction
- Resources
- Profiles
- ValueSets
- Extensions







- Hoofdpagina
- Recente wijzigingen
- Willekeurige pagina
- Support
 - Wiki handleiding
- Hulpmiddelen
 - Koppelingen naar deze pagina
 - Verwante wijzigingen
 - Speciale pagina's
 - Printvriendelijke versie

https://informatiestandaarden.nictiz.nl/wiki/MedMij:V2018.03_FHIR_MedicationProcess

MedMij Overleg Lezen Brontekst bekijken Geschiedenis weergeven Zoeken

MedMij FHIR use case Medication Process


 MEDICATION
  AFSPRAKENSTELSEL
  FUNCTIONEEL
  TECHNISCH

Inhoud [\[weergeven\]](#)

Introduction

Nictiz is the centre of expertise for standardisation and eHealth in The Netherlands

This page describes patient's use cases, in a personal health record (PHR) context, in relation to the information standard Medication Process (MP). These use cases are based on version 9.x of this standard. Functional



FHIR Implementation Guides – Validation

Regardless of the publication tooling and repository, the IG must define and provide the necessary FHIR conformance resources for validation:

- **Profiles** (StructureDefinition)
 - Data structures, cardinality, constraints/invariants, terminology bindings
- **Operations** (OperationDefinition)
 - Custom operations/functionality
- **Search parameters** (SearchParameter)
- **Terminology** (ValueSet, CodeSystem, ConceptMap, NamingSystem)

FHIR Implementation Guides – Validation

Validation testing for conformance to a FHIR IG involves:

- Verifying proper behavior based on the IG Guidelines, Use Cases and Scenarios, and Security Requirements
 - *Read the IG carefully!*
- Use of a FHIR Validation Engine to verify the resource instances
 - Java RI Validation Engine from the FHIR specification (Grahame, James and Community)
 - .NET Validation Engine available as NuGet packages (Ewout, Brian and Community)

FHIR Implementation Guides – FHIR Test Engine

- IG Guidelines, Use Cases and Scenarios, and Security Requirements are used to define TestScripts:
 - Simple tests for individual operations
 - Complex tests with multiple operations simulating workflow patterns
 - Asserts to verify specific requirements

Executed by a FHIR Test Engine; e.g. Touchstone

FHIR Implementation Guides – FHIR Test Engine

- Validating a resource means checking the following aspects:
 - **Structure:** Check that all the content in the resource is described by the specification, and nothing extra is present
 - **Cardinality:** Check that the cardinality of all properties is correct (min & max)
 - **Value Domains:** Check that the values of all properties conform to the rules for the specified types
 - **Code bindings:** Check that codes/displays provided in the Coding/CodeableConcept types are valid
 - **Invariants:** Check that the invariants have been followed correctly
 - **Profiles:** Check that any rules in profiles have been followed

Supported by a FHIR Test Engine; e.g. Touchstone

Touchstone Demo – Test Execution with Validation



TOUCHSTONE

AEGIS.net

Program Support – Touchstone Features

- Organization Groups – Public/Private Access to Test Results
- Security and Privacy Settings for Test Definitions
- FHIRSandbox Test Definitions
- Touchstone RESTful API
 - Remote launch and monitor of test executions
 - Integration with continuous build environments (Jenkins, Travis, ...)
- Touchstone IDE
 - Eclipse-based IDE for TestScript development and upload to Touchstone
 - Integration with Simplifier
- WildFHIR Test Server
- Java Validation Engine (incorporated as an internal service)

Program Support – Nictiz MedMij

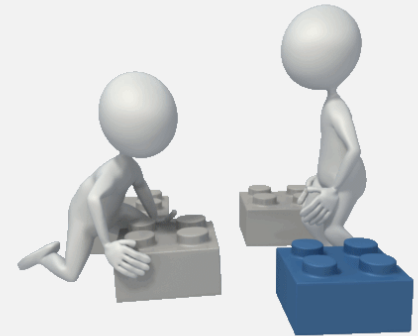
- Nictiz engaged with AEGIS earlier this year for FHIR Testing Support with Touchstone
- Training on use of Touchstone and TestScript Development delivered
- Nictiz staff leading development of TestScripts against MedMij Implementation Guide
- AEGIS supporting TestScript Development, Use of Touchstone and Validation of Nictiz MedMij FHIR Profiles
- To-date nine (9) different program areas under development for testing and certification: eAppointments to Medication Process
- Nictiz WildFHIR Test Server instance available as reference implementation for various MedMij use cases

Touchstone Demo – Program Support



Hands on Exercises

- Getting Started – registration and test system setup
- Test Setup and Execution - FHIR3-0-1-DevDays18-11
 - *Capabilities*: capabilities (metadata) operation
 - *Basic Operations*: read, search, create, update, delete
 - *Workflow*: basic operations in a single TestScript illustrating a simple workflow
 - *Validation*: show use of the integrated validation engine
- Nictiz MedMij Use Case Review – Medication Process
 - https://informatistandaarden.nictiz.nl/wiki/MedMij:V2018.03_FHIR_MedicationProcess
 - Demonstration TestScript for MedMij Medication Process Use Case



Check out our FHIR Testing and Validation Paper

<https://medinform.jmir.org/2018/4/e10870>

Original Paper

Validation and Testing of Fast Healthcare Interoperability Resources Standards Compliance: Data Analysis

Jason Walonoski¹, MS  ; Robert Scanlon¹, MS  ; Conor Dowling¹  ; Mario Hyland²  ; Richard Ettema², BS  ;
Steven Posnack³, MS, MHS 

¹The MITRE Corporation, Bedford, MA, United States

²AEGIS.net, Inc, Rockville, MD, United States

³The Office of the National Coordinator for Health Information Technology, US Department of Health and Human Services, Washington, DC, United States

Discussion (Q & A)



See you at the Hands-on FHIR Testing table.
Thank you!

ORGANIZED BY

firely

PARTNERS

 HL7® FHIR®
FOUNDATION



gefyra

richard.ettema@aegis.net