# # AgentAPI

draft specification

Help develop national and global standard APIs for Third-Party Privacy Services!

* Expand integration between Consent Management/DSAR platform providers, Privacy Agents, Enforcement Authorities, Businesses, and individual consumers/data subjects.
* Automate all privacy rights requests to improve data quality and integrity.
* Make authentication and authorization services accessible to everyone.

This API, designed for individual consumers and their agents, will benefit all stakeholders.

Specifications for this API are derived from:

* HL7.org https://hl7.org/FHIR
* VGS API https://www.verygoodsecurity.com/docs/api/1

## ## FHIR IMPLEMENTATION GUIDELINE FOR DIGITAL RIGHTS

===================================================

### ### Topic: Architectual Overview

**\*\*FHIR implementation\*\***

At its core, FHIR contains two primary components:

1. Resources - a collection of information models that define the data elements, constraints and relationships for the business objects most relevant to healthcare. From a model-driven architecture perspective, FHIR resources are notionally equivalent to a physical model implemented in XML or JSON.
2. APIs - a collection of well-defined interfaces for interoperating between two applications.

Although not required, the FHIR specification targets RESTful interfaces for API implementation.

In the healthcare domain, there is a notional and ongoing evolutionary, consensus-based process for standardizing on a core set of common business objects including things like: a patient, a procedure, an observation, an order, etc. (see a list of defined resources).

The FHIR specification provides a framework for defining these healthcare business objects (resources), for relating them together in a compositional manner, for implementing them in a computable form, and for sharing them across well-defined interfaces.

The framework contains a verifiable and testable syntax, a set of rules and constraints, methods and interface signatures for FHIR-aware APIs, and specifications for the implementation of a server capable of requesting and delivering FHIR business objects.

**\*\*Digital Rights implementation\*\***

FHIR architecture's separation of content (resources) from process (APIs) makes possible adapting this framework for digital rights processes.

Process is not limited to the restful API: FHIR supports a wide variety of transport protocols in addition to HTTP, and its resources are designed to support without hard dependencies, a wide variety of implementation and storage options.

Likewise, Digital Rights can be composed as structured Requests and Responses which represent "a data subject", "a privacy right", "a vulnerability", and "a remediation".

============================

### ### Topic: Specification Overview

**\*\*FHIR implementation\*\***

The base FHIR specification (this specification) describes a set of base resources, frameworks and APIs that are used in healthcare.

Due to wide variability between jurisdictions and across the healthcare ecosystem around practices, requirements, regulations, education, not every action or resource is feasible and/or beneficial.

The FHIR specification, as a "platform specification", has particular contexts of use to support a variety of different adaptations. Typically, these adaptations specify:

* Rules about which resource elements are or are not used, and what additional elements are added that are not part of the base specification.
* Rules about which API features are used, and how they are used.
* Rules about which terminologies are used in particular elements.
* Descriptions of how the Resource elements and API features map to local requirements and/or implementations.

Because of the nature of the healthcare ecosystem, there may be multiple overlapping sets of adaptations - by healthcare domain, by country, by institution, and/or by vendor/implementation.

FHIR defines a cascade of artifacts for this purpose:

1. 1) Implementation Guide (IG): A coherent and bounded set of adaptations that are published as a single unit. This Guide is an example, although the Digital Rights adaptation is published adjacent to the FHIR Core specification for reference purposes.
2. 2) Package: A group of related adaptations that are published as a group within an Implementation Guide. US Core Capability Statements is an example of a Package.
3. 3) Conformance Resource: A single resource in a package that makes rules about how an implementation works. DAF Problem Value Set is an example of a Conformance Resource. It is broadly defined as a series of brief statements that catalog a patient's signs, symptoms, and defined conditions that are relevant to that patient's healthcare.
4. 4) Profile: A set of constraints on a resource represented as a structure definition with kind = constraint DAF Medication Request is an example of a Profile that defines constraints and extensions on the MedicationRequest resource for the minimal set of data to query and retrieve prescription information.

**\*\*Digital Rights implementation\*\***

Privacy laws and procedures, like healthcare, are also widely vary in content and process around the world. Even if this were not the case, each party's implementation of a digital rights specification would differ: not every party has the same technical capabilities, nor do they all support the same resources.

Fortunately for us, the FHIR architecture supports customized adaptations. For example:

THE FHIR specification accommodates these differences through artifacts such as a Server Capability Statement, and versioning of content resources and the FHIR framework itself.

Instead of a DAF Problem Value Set for healthcare adaptations, a Digital Rights Vulnerability Value Set could be a Conformance Resource that describes potential harms. Another Conformance Resource could be a Conformance/Violation List which catalogs a set of digital rights statues and regulations. Yet another Conformance Resource could be a list of remediation requirements.

A Profile is used as a set of constraints on a resource such a Person, Organization, or Practitioner.

Person and Organization are nouns representing two different things. A Practitioner is an adjective used to describe a Resource, such as a Person or an Organization. Profiles are well-suited to define the contextual roles of these resources, and the constraints and extensions on their structure and use. A Profile could also govern a selected resource such as a RemediationRequest, represented as a structure definition with kind = constraint.

============================

### ### Topic: Framework Overview

**\*\*FHIR implementation\*\***

Foundation Resources: Foundation resources are the most rudimentary, foundational resources. They are often used for infrastructural tasks. Although not prohibited, they are not always referenced by other resources.

Base Resources: Layer 2 consists of base resources. These are often the leaf nodes of a resource graph. In other words, they are often referenced by other resources, but don't typically reference other resources themselves.

Clinical Resources: Layer 3 includes the resources that are clinical in nature but are also very common across many use cases. This includes resources for clinical observations, clinical treatment, care provision, and medications.

Financial Resources: Layer 4 is dedicated to financial resources. Logically, financial resources build on clinical and base resources. For example, a billing resource will reference clinical events and activities as well as base resources like a patient.

Specialized Resources: In layer 5, we find more specialized resources for less common use cases. These resources almost always reference resources in lower layers.

Resource Contextualization: Layer 6 does not contain resources. However, it does extend the composition framework made up by the first five layers of resources. Layer 6 includes profiles and graphs.

**\*\*Digital Rights implementation\*\***

Foundation Resources in Layer 1 SHALL NOT BE MODIFIED.

Base Resources in Layer 2 MAY BE MODIFIED, but SHOULD BE extended by reference whenever feasible.

Instead of Clinical Resources, Layer 3 includes MODIFIED OR ADDITIONAL resources that are used for Digital Rights. Examples include Remediation in place of Medication, etc.

For Digital Rights, Layer 4 represents the enforcement aspect of implementing policies and controls. Instead of supporting the costing, financial transactions and billing which occur within and between healthcare providers, insurers and patients in FHIR, the Digital Rights implementation supports filing complaints and/or lawsuits, terminating relationships, testing for violations and effectiveness of remediation controls, and publishing of evidence in a privacy-preserving manner for the benefit of investigators, auditors, and any other relevant party.

Specialized Resources in layer 5, and Resource Contextualization in Layer 6 is where MOST MODIFICATIONS should be made for Digital Rights implementations. For instance, although the following resources are defined in Layer 4 of FHIR, their modified versions should be defined or constrained by Layer 5 or 6.

Consumer PII-Vault

 This is a proxy for a consumer's repository of personal information. Initially it is used to store test data. Ultimately, it is used to store or access real personal information from one or more data sources. This is intended to share mostly non-sensitive personal data typically used for identity verification, contact information, and account creation.

Consumer Auditor

 This component registers event-listeners for a variety of data sources, which can be configured to generate log entries and notifications on specific events.

HTTP API Server

This is a proxy for the server hosting the API for submitting privacy rights requests and receiving correlated responses.

Identity Provider

This is a proxy for integrating with a number of identity providers using SAML, and/or oAuth.

Vendor Risk Manager

This is a proxy for collecting evidence about vendors, running tests and publishing test results. It also provides risk assessments and tools for vendor life-cycle management.

HTTP API Client SDKs

These SDKs will assist developers in consuming API services and orchestrating them to support custom business processes and integration with local resources.

eDiscovery Tools

A suite of localized tools for discovery of vendors, accounts, services on personal devices and resources, or online resources.

Supporting Environment

The AgentAPI design includes support for integration with other APIs and external security and compliance controls. I chose the Vault API and Control API from VGS, (Very Good Security), a firm which provides data protection solutions for PCI-DSS.

Like FHIR, the VGS Vault also protects sensitive information, such as payment data, personally identifiable information (PII), passwords, private keys, environment variables, and SIEM data. VGS claims their APIs can reduce your compliance certification tasks by over 90% for PCI and SOC2, as well as compliance with HIPAA and GDPR. The VGS Vault is an external Consumer PII-Vault used to test AgentAPI integration with other vendors.

VGS also provides a Control API, which contains everything you need to either build a security program from scratch or achieve any compliance your business needs. Control fundamentally accelerates all aspects of achieving compliance by providing:

* A complete set of security controls for your organization to adopt and modify as you need
* Up to date and prescriptive tasks translated into english for non-specialists and engineers
* Automated evidence collection, saving time from screenshots and deciphering what to show to meet a control
* Ongoing real security monitoring, above and beyond compliance minimums
* The ability to assign tasks and notifications for completion
* Automated compliance headaches like vendor risk questionnaires, onboarding, and gap assessments.
* Real-time auditor interaction and evidence feedback

Control is fundamentally designed to save your team two hires: a compliance expert who has to function as a project manager, and a security engineer who has to function as an IT operations manager. By providing automated evidence and the complete audit in the tool, Control keeps your audit on track while monitoring your evidence and providing guidance.

The Control API is needed on the assumption that no system can reliably monitor and regulate itself.