## Short note about HL7 FHIR (for people from the FAIR community)

FHIR is a data exchange standard for medical data. It is developed and maintained by HL7. The current release 4 (December 2019) describes the first normative version. In terms of content, FHIR is a further development of the existing HL7 standards V2 (text-oriented messages) and V3 (XML) and borrows conceptually from the HL7 Reference Information Model (RIM). FHIR uses universal web technologies such as HTTP, ReST, JSON, XML, TLS and OAuth and has APIs for all common programming languages.

The primary building block in FHIR is the resource. Resources model delimitable artifacts of the real world such as [patients](http://hl7.org/fhir/patient.html), [diagnoses](http://hl7.org/fhir/condition.html), [laboratory tests](https://www.hl7.org/fhir/observation.html), or [medications](https://www.hl7.org/fhir/medication.html). Even an [ImplementationGuide](https://www.hl7.org/fhir/implementationguide.html) is defined as a FHIR resource.

FHIR resources have a variety of coordinated, explanatory attributes so that common use cases can be covered without the need for extensions. FHIR resources reference each other so that complex medical "stories" can be represented.

## Relation between FHIR and FAIR

FHIR is a technical standard that provides a rich set of realization options for complex medical data models, but does not mandate a unique realization in every case. The FAIR principles are also vague on many points to ensure broad validity.

The goal of this Implementation Guide is to develop best practices for implementing the FAIR principles when exposing medical data using FHIR.

A simple example would be FAIR's requirement for a unique, universally valid and stable identifier for individual instances of resources, such as a single patient. The FHIR standard provides for various identifier references, some of which are technical and some of which are business identifiers. At the same time, FHIR supports arbitrary systems for identifiers. In practice, there are local standards (patient number in hospital), standards with a wide distribution in the field of medicine (OID) or standards from the field of technical implementation (URI).

Since the FHIR model allows many possibilities, it makes sense to mark preferred variants.

## Findability

### F1: (Meta) data are assigned globally unique and persistent identifiers

The first and arguably most important FAIR principles is that [F1](https://www.go-fair.org/fair-principles/f1-meta-data-assigned-globally-unique-persistent-identifiers/): "(Meta) data are assigned globally unique and persistent identifiers". In practice, often another property of identifiers is added, namely resolvability, meaning that the identifier doubles as a link to a location which can be used to retrieve the (meta)data.

When medical data is exposed as FHIR Resources, then automatically, per the base [Resource](https://www.hl7.org/fhir/resource.html) type, the resource will have a 'logical' [id](https://www.hl7.org/fhir/resource-definitions.html#Resource.id) which serves as a URL for the resource. The description also states that "Once assigned, this value never changes", so we can consider this identifier to be persistent at least at specification level. When combined with the URL prefix for the FHIR server, a globally unique and also resolvable identifier can be constructed, e.g. here are two GUPRIs for example Patient records from FHIR test servers: <http://test.fhir.org/r3/Patient/10> and <https://server.fire.ly/Patient/57bb2db9-1044-45f1-94c5-7ec549ae76f3>. Both resolve to a description of the resource, the first to a web page describing the resource, the second leads to a direct download of an XML document which describes this resource. However, if the URL at which the FHIR resource server is changed, all the identifiers thus constructed would change as well. It is therefore advisable to designate a permanent namespace for prefixing the resources. This URL can also be referenced in the [fullUrl](https://www.hl7.org/fhir/bundle-definitions.html" \l "Bundle.entry.fullUrl) topic of a Bundle that wraps the resource, as in the Firely server example.