

## 10.1 検査リソースの内容

---

<https://www.hl7.org/fhir/observation.html>

- 10.1 検査リソースの内容
  - 10.1.4 注釈:
    - 10.1.4.1 検査のプロファイリング
    - 10.1.4.2 検査の対象
    - 10.1.4.3 検査のグループ分け
      - 10.1.4.3.1 DiagnosticReport.result
      - 10.1.4.3.2 Observation.component
      - 10.1.4.3.3 Observation.hasMember of and Observation.derivedFrom
    - 10.1.4.4 Using codes in Observation
      - 10.1.4.4.1 Multiple Codings
      - 10.1.4.4.2 Text values for coded results:
      - 10.1.4.4.3 Interoperability Issues using code value pairs in FHIR
      - 10.1.4.4.4 Guidance:
    - 10.1.4.5 Refining the interpretation of an Observation using additional codes or Observations
    -  and Datatypes
    - 10.1.4.7 Physiologically Relevant Time of the Observation
    - 10.1.4.8 Reference Range
    - 10.1.4.9 Canceled or Aborted Observations
    - 10.1.4.10 Genetic Observations

### 10.1.4 注釈:

#### 10.1.4.1 検査のプロファイリング

最も単純な場合、リソースインスタンスはコード、値、および状態フラグだけで構成できます。他のプロパティの関連性は、検査の種類によって異なります。プロファイルは、与えられた事例に対する特定の種類の検査を記録する際の指針を提供するために作成されます。検査リソースは、大多数のシステムで記録された詳細さの度合いに焦点を当てています。ただし、特定の事例では、特定の状況に関連する追加の制約と追加の情報がある場合があります。こうした複雑さを追加で取り込めるように、他のリソースと同様、拡張機能が利用できます。

#### 10.1.4.2 検査の対象

通常、対象（患者もしくは患者グループ、場所、もしくは機器）に対して検査が行われ、その検査のために直接計測されたものと対象との間の区別は、検査コード（例えば、血液グルコース）によって特定され、別々に表現する必要はありません。ただし、対象そのものでない場合、検査の焦点を表現するために3つの属性を使用できます。`specimen` (検体) および `bodySite` (検査対応部位) の要素は、対象のサンプルもしくは解剖学的・形態学的な位置の測定値の表現に利用され、患者に埋め込まれた機器や別の検査などの注意点に相当するような対象の特定の観点に焦点を当てて表現しています。

#### 10.1.4.3 検査のグループ分け

Many observations have important relationships to other observations and need to be grouped together. These structures have been defined to do this: `DiagnosticReport` and `DiagnosticReport.result`, `Observation` and the elements: `Observation.component`, `Observation.hasMember` and `Observation.derivedFrom`. The sections below provides guidance around which structure to use. Because the idea of what to group together is often highly contextual and based upon the end user's point of view, the choice of which structure to use will be driven by jurisdiction, organizational practice and context. Profiling will normally be necessary for implementation.

#### 10.1.4.3.1 DiagnosticReport.result

`DiagnosticReport` relates directly to an order (`ServiceRequest`). The `DiagnosticReport.code` names the panel and serves as the grouping element, which is traditionally referred to as a "panel" or "battery" by laboratories. The `DiagnosticReport.result` element references the individual observations. Several examples demonstrate observation grouping using `DiagnosticReport` as the grouping structure.

#### 10.1.4.3.2 Observation.component

`Observation.component` is used for any supporting result that cannot reasonably be interpreted and used outside the scope of the `Observation` it is a component of. Component observations may make up the separate and individual parts of the observation or may provide qualifying information to `Observation.code` and may only be able to be understood in relation to the `Observation.code` (for example, see the `$stats` operation). Therefore **all** code-value and component.code-component.value pairs need to be taken into account to correctly understand the meaning of the observation. Components should only be used when there is only one method, one observation, one performer, one device, and one time. Some use cases for using this structure include:

1. Observations that are commonly produced and interpreted together. For example, systolic and diastolic blood pressure are represented as a single Blood pressure panel.
2. Assessment tool results that are commonly produced and interpreted together. For example, a newborn Apgar score that is a single `Observation` with five components.
3. Representing multiple answers to a question (relationship and boundaries between `Observation` and `Questionnaire/QuestionnaireResponse`). For example, reporting the types of alcohol consumed by a patient

On the other hand, any observations that are clinically relevant outside the context of being a component of another observation should be represented by separate `Observation` resources. For example a Body Mass Index (BMI) `Observation` should *not* contain components for height and weight because they are clinically relevant observations on their own and should be represented by separate `Observation` resources. See the section below on how to relate independent `Observations`.

#### 10.1.4.3.3 Observation.hasMember of and Observation.derivedFrom

`Observation.hasMember` and `Observation.derivedFrom` and the core extensions: `Observation-sequelTo` and `Observation-replaces` are used for any supporting result that can be interpreted and used on its own and has one or more different values for method, observation, performer, device, time, and/or error conditions. Two common use cases for using this structure are:

For grouping related observations such as for a "panel" or "battery". In this case the `Observation.code` represents the "panel" code, typically `Observation.value[x]` is not present, and the set of member Observations are listed in `Observation.hasMember`. This structure permits nested grouping when used with DiagnosticReport (e.g. complex micro isolate and sensitivities report). When linking to other Observations from which an Observation is derived. In this case both `Observation.code` and `Observation.value[x]` are present, and the linked observations are listed in `Observation.derivedFrom`. An example of this would be a Body Mass Index (BMI) Observation where the height and weight measurements are referenced.

#### 10.1.4.4 Using codes in Observation

When a result value is represented as a predefined concept using a code, `valueCodeableConcept` is used. This element is bound to a value set comprised of a standard nomenclature such as SNOMED CT or a source system ("local") coded result values.

##### 10.1.4.4.1 Multiple Codings

Results may be coded in multiple value sets based on different code systems and these may be mapped using the ConceptMap resource and/or given as additional codings directly in the element as shown in the example below.

For example the LOINC 43304-5 *Chlamydia trachomatis rRNA [Presence] in Unspecified specimen by Probe and target amplification method* is typically associated with coded presence/absence concepts. Using the coded value for 'negative' with a standard code translation, `valueCodeableConcept` would be:

```
"valueCodeableConcept": {
  "coding": [
    {
      "system": "http://snomed.info/sct",
      "code": "260385009",
      "display": "Negative"
    }, {
      "system": "https://acme.lab/resultcodes",
      "code": "NEG",
      "display": "Negative"
    }
  ],
  "text": "Negative for Chlamydia Trachomatis rRNA"
}
```

##### 10.1.4.4.2 Text values for coded results:

When the data element is usually coded or the type associated with the `code` element defines a coded value, use `valueCodeableConcept` even if there is no appropriate code and only free text is available. For example using text only, the `valueCodeableConcept` element would be:

```
"valueCodeableConcept": {
  "text": "uncoded free text result"
}
```

When a coded answer list includes a concept code for "other" and there is a free text description of the concept, the `valueCodeableConcept.text` element should be used to capture the full meaning of the source. In the example below, the answer code "Other" is provided in the `valueCodeableConcept` element and the text value supplied value in the `CodeableConcept.text` element.

```
{
  "resourceType": "Observation",
  ... snip ...
  "code": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "74076-1",
        "display": "Medication or substance involved"
      }
    ]
  },
  .. snip ...
  "valueCodeableConcept": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "LA20343-2",
        "display": "Other substance: PLEASE SPECIFY"
      }
    ],
    "text": "Other: Blue pills I found under my couch"
  }
  .. snip ...
}
```

#### 10.1.4.4.3 Interoperability Issues using code value pairs in FHIR

A recurring issue for many observation events, regardless of the particular pattern, is determining how to populate `observation.code` and `observation.value`. While this is typically straight-forward for laboratory observations, it can get blurry for other types of observations, such as findings and disorders, family history observations, etc. This discussion focuses on the way in which the coded representation of such statements is expressed using the `Observation.code` and `Observation.value` elements.

There are two distinct facets that are central to a FHIR Observations:

- The action taken to make the finding and/or the property about which the property was observed. For example: measurement of blood hemoglobin.
- The result of the observation. For example: 14 g/dl.

Several different ways of representing the same information exist using different combinations of the **Observation.code** and **Observation.value**. Unconstrained use of the alternatives presents a major challenge for computation of semantic equivalence and for safe interpretation of observations originating from different applications and users. The following four patterns could reasonably represent the same case. Considering that the Observation resource needs to support many use cases, the appropriate place to define the specific pattern is expected to be done through profiles and implementation guides as specified by the jurisdictions and/or organizations implementing FHIR:

1. **Observation.code** represents the nature of the observation and the **Observation.value** a code represents the non-numeric result value. These are two distinct facets that are central to a FHIR Observations. For example:
  - code=[Examination]
  - value=[Abdomen tender]
2. **Observation.code** is nearly identical to 1) above, but the level of granularity is shifted from the value to code. For example:
  - code=[Abdominal examination]
  - value=[Tenderness]
3. The **Observation.code** is also expressed in a way that does not specify the observation action but indicates a statement about findings reduced to a single name (or term), as in the above item. In this example, the **Observation.value** is present and "qualifies" the finding typically confirming or refuting it. For example:
  - code=[Abdominal tenderness]
  - value=[found/true]
4. In this example the **Observation.code** is expressed in a way that does not specify the observation action but indicates a statement about findings reduced to a single name (or term). In this particular example in that context, the **Observation.value** is omitted. For example:
  - code=[Abdominal tenderness]
  - value element is omitted

#### 10.1.4.4.4 Guidance:

1. Recommended rules for case 1 and 2 patterns:
  - The Observation.code is preferably a LOINC concept code.
    - If a SNOMED CT concept code is used, the expression SHOULD represent a 363787002 (Observable entity(Observable entity)) or 386053000 (Evaluation procedure(evaluation procedure))
  - For non-numeric values, the Observation.value is preferably a SNOMED CT concept code.
2. Recommended rules for case 3 pattern:
  - The Observation.code is preferably a LOINC or SNOMED CT concept code.

- If a SNOMED CT concept code is used, the expression SHOULD represent a 404684003 (Clinical finding (finding)) , 413350009 (Finding with explicit context(finding)), or 272379006 (Event(event)).
  - The Observation.value is represented by either
    - valueBoolean
    - valueCodeableConcept preferably using:
      - SNOMED CT where concept is-a 362981000 (Qualifier value (qualifier value))
      - v2 Yes/no Indicator
      - v2 Expanded Yes/no Indicator (unfortunately is missing 'not given')
3. Recommended rules for case 4 pattern:
- The Observation.code is preferably a SNOMED CT concept code where the concept is-a 404684003 (Clinical finding (finding)) , 413350009 (Finding with explicit context(finding)), or 272379006 (Event(event)).
  - The Observation.value is omitted. The default interpretation is the concept (single code or expression) represented in Observation.code is present in the patient. An Observation.dataAbsentReason value of 'clinical-finding' SHOULD be used to indicate why the expected value is missing.
4. SHOULD NOT use the Assertion pattern as described in HL7 Version 3 Implementation Guide: TermInfo - Using SNOMED CT in CDA R2 Models, Release 1 . ( The code is 'ASSERTION' and the value is a SNOMED CT concept or expression )

#### 10.1.4.5 Refining the interpretation of an Observation using additional codes or Observations

The following list provides guidance on using codes or other observations to provide additional context that may alter how an observation is interpreted.:

1. If possible, use the most specific code you can

e.g.:

```
{
  "resourceType": "Observation",
  ... snip ...
  "code": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "6689-4",
        "display": "Glucose [Mass/volume] in Blood --2 hours post meal"
      }
    ]
  },
  ... snip ...
}
```

2. Alternatively, use additional codes in Observation.code as described above.

e.g.: Observation.code = coding-1: 59408-5 Oxygen saturation in Arterial blood by Pulse oximetry,  
coding-2: 20564-1 Oxygen saturation in Blood

```
{
  "resourceType": "Observation",
  ... snip ...
  "code": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "59408-5",
        "display": "Oxygen saturation in Arterial blood by Pulse
oximetry"
      },
      {
        "system": "http://loinc.org",
        "code": "20564-1",
        "display": "Oxygen saturation in Blood"
      }
    ]
  },
  ... snip ...
}
```

As described above, observations are typically grouped together to provide additional information needed for correctly understanding and interpreting the observation. As an alternative to grouping observations, extensions may be used to provide references to other observations needed for understanding and interpreting an observation.

Note: We are seeking input from the implementer community in evaluating existing Observation Extensions for this purpose [Feedback here](#) .

## ☑ and Datatypes

- The element, Observation.value[x], has a variable name depending on the type as follows:
  - valueQuantity
  - valueCodeableConcept
  - valueString
  - valueBoolean
  - valueInteger
  - valueRange
  - valueRatio
  - valueSampledData
  - valueTime
  - valueDateTime
  - valuePeriod

- See above section on Using codes for result values
- The Boolean data type is rarely used for `value[x]` because most observations result values are never truly Boolean due to exceptional values such as "unknown", therefore they should use the CodeableConcept data type instead and select codes from <http://terminology.hl7.org/ValueSet/v2-0136> (these "yes/no" concepts can be mapped to the display name "true/false" or other mutually exclusive terms that may be needed")
- The special values "E" (error), "L" (below detection limit) and "U" (above detection limit) can be used are in the SampledData data type. However, when using valueQuantity in an observation for above and below detection limit values, valueQuantity should be used by stating the limit along with the comparator. In addition, when there is an error the dataAbsentReason element should be used with the appropriate value ('error' or 'NaN'). For example if the value was below the lower limit of detection of <2.0 mmol/L the `valueQuantity` would be:

```
"valueQuantity": {
  "value": 2.0,
  "comparator": "<",
  "unit": "mmol/l",
  "system": "http://unitsofmeasure.org",
  "code": "mmol/L"
}
```

If the value was "NaN" (i.e. an error) the `valueCodeableConcept` element would be absent and `dataAbsentReason` element would be:

```
"dataAbsentReason": {
  "coding": [
    {
      "system":
"http://terminology.hl7.org/CodeSystem/data-absent-reason",
      "code": "NaN",
      "display": "Not a Number"
    }
  ]
}
```

- Because there are multiple types allowed for the `value` element, multiple value search parameters are defined. There is no standard parameter for searching values of type Ratio

#### 10.1.4.7 Physiologically Relevant Time of the Observation

The effectiveDateTime or effectivePeriod is the time that the observation is most relevant as an observation of the subject. For a biological subject (e.g. a human patient), this is the physiologically relevant time of the observation. In the case of an observation using a specimen, this represents the start and end of the specimen collection (e.g. 24-hour Urine Sodium), but if the



collection time is sufficiently short, this is reported as a point in time value (e.g. normal venipuncture). In the case of an observation obtained directly from a subject (e.g. BP, Chest X-ray), this is the start and end time of the observation process, which again, is often reported as a single point in time.

#### 10.1.4.8 Reference Range

Most common observations will only have one generic reference range. Reference ranges may be useful for laboratory tests and other measures like systolic blood pressure but will have little relevance for something like "pregnancy status". Systems MAY choose to restrict to only supplying the relevant reference range based on knowledge about the patient (e.g. specific to the patient's age, gender, weight and other factors), but this might not be possible or appropriate. Whenever more than one reference range is supplied, the differences between them SHOULD be provided in the reference range and/or age properties.

#### 10.1.4.9 Canceled or Aborted Observations

If a measurement or test could not be completed (for example if the specimen is unsatisfactory or the provider cancelled the order), then the status value should be updated to "cancelled" and the specific details given - preferably as coded values in the `dataAbsentReason` or `valueCodeableConcept` element. Additional information may be provided in the `note` element as well. The specimen reject example demonstrates this using a coded value for unsatisfactory specimen in `dataAbsentReason`.

#### 10.1.4.10 Genetic Observations

Genetic reporting makes heavy use of the `DiagnosticReport` and `Observation` resources. An implementation guide describing how to represent genetic results can be found [here](#).