



Kanta

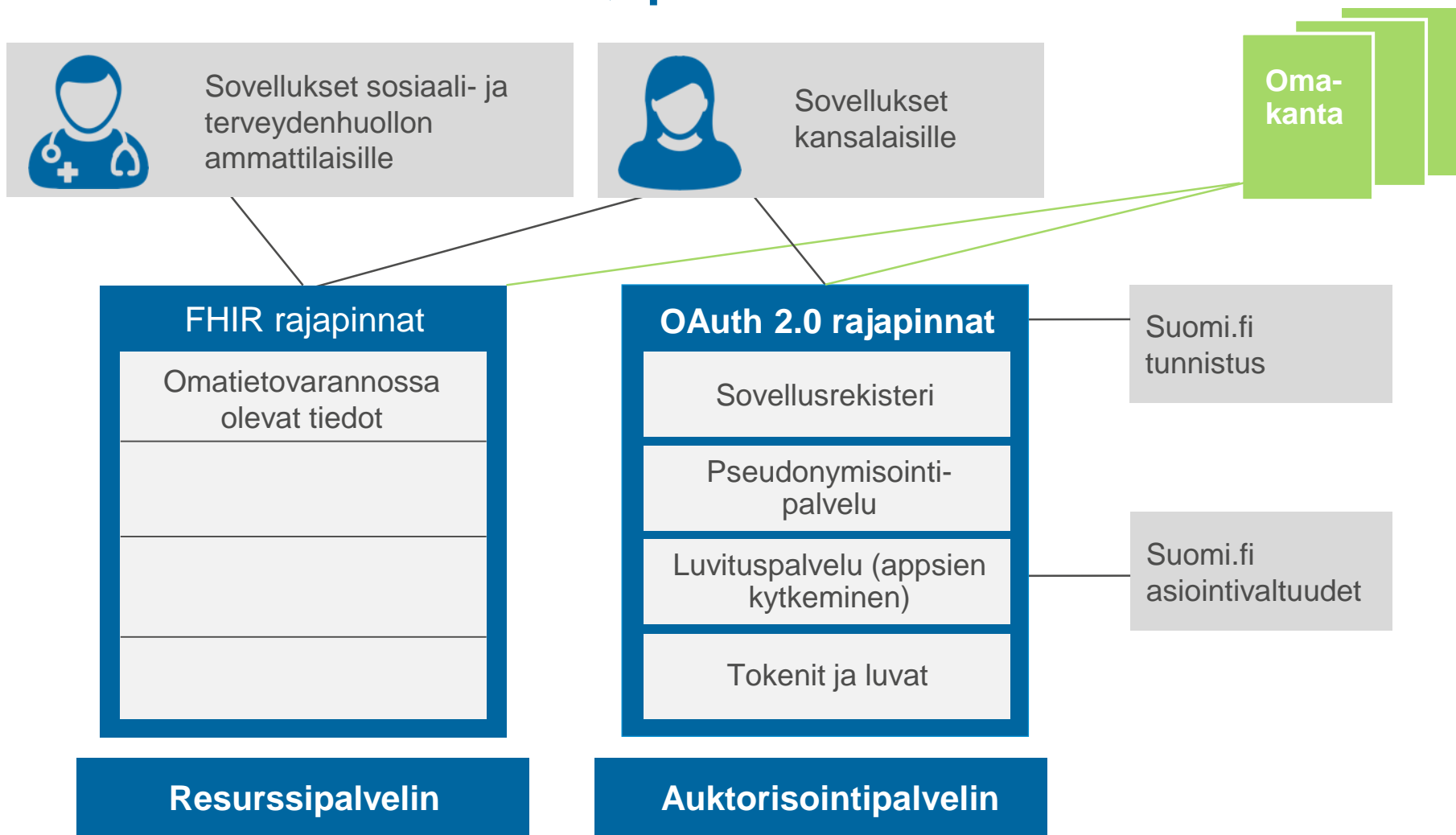
Kanta PHR:n CapabilityStatement ja REST-API

Eeva Turkka

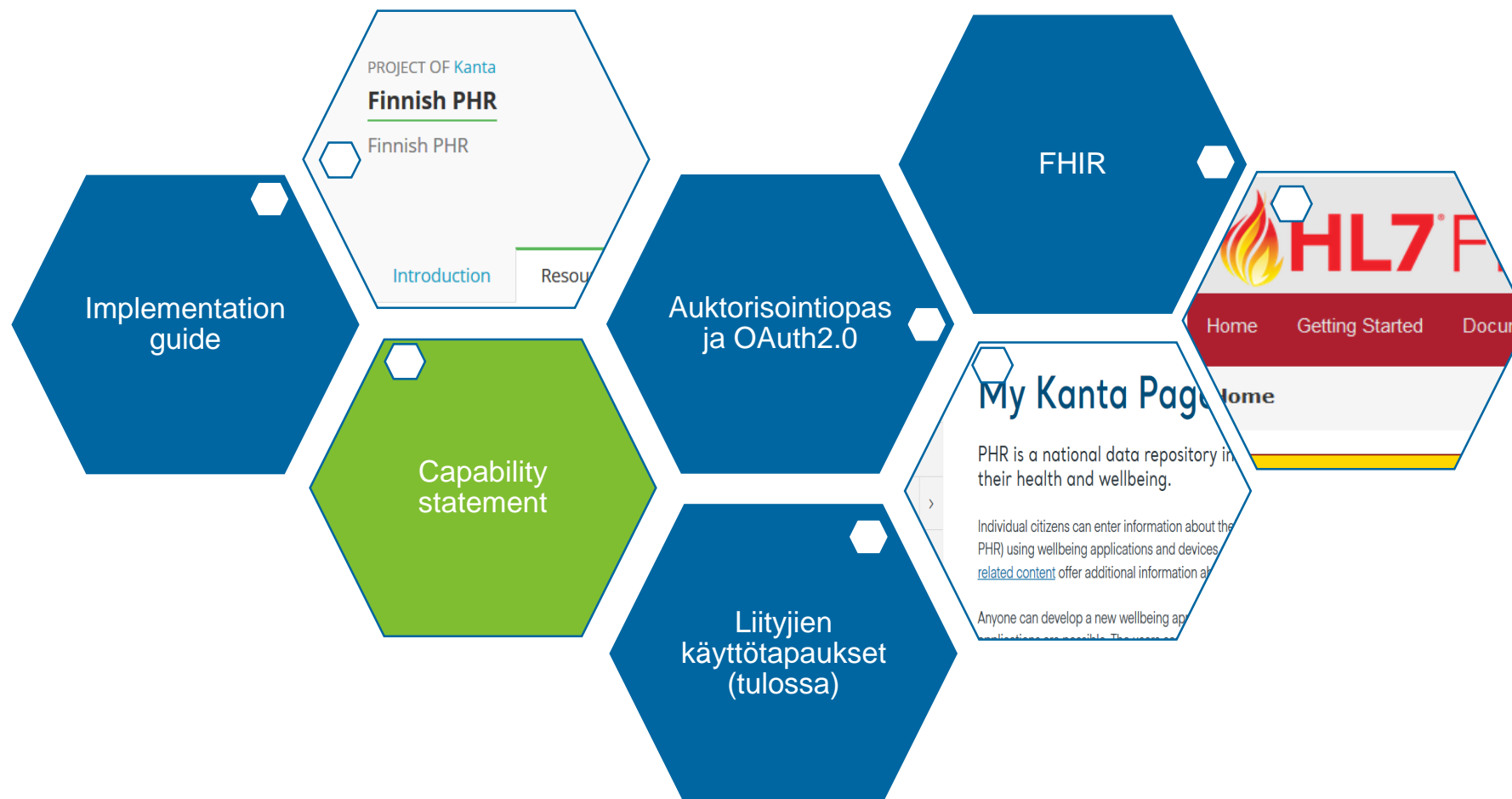


to

Omatietovaranto, pääelementit



Omatietovarannon dokumentaatio













FHIR-standardin tavat kuvata palvelu

- PHR:n toiminnot ja tietosisällöt on kuvattu FHIR-standardin keinoin
- Tietosisällöt on kuvattu implementointioppaaseen profiileina ja niistä on lisäksi esimerkit
- FHIR-palvelimen rajapinnat ja toiminnallisuudet on kuvattu Capability statement –tyyppisenä resurssina
- Lisäksi OAuth2.0 rajapinnat on kuvattu auktorisointioppaaseen

Mikä on CapabilityStatement

- ”CapabilityStatement dokumentoi joukon toimintoja, jotka kertovat mitä palvelin tekee tai määrittelee halutun toiminnallisuuden”
- Capability statement on FHIR-resurssi kuten mm mittausdatakin

Structure

Name	Flags	Card.	Type	Description & Constraints
 CapabilityStatement	I N		DomainResource	A statement of system capabilities + Warning: Name should be usable as an identifier for the module applications such as code generation + Rule: A Capability Statement SHALL have at least one of REST element. + Rule: A Capability Statement SHALL have at least one of description implementation element. + Rule: Messaging end-point is required (and is only permitted) implementation. + Rule: The set of documents must be unique by the combination of name and title + Rule: If kind = instance, implementation must be present and + Rule: If kind = capability, implementation must be absent, so + Rule: If kind = requirements, implementation and software must be present Elements defined in Ancestors: id , meta , implicitRules , language , modifierExtension
 url	Σ	0..1	uri	Canonical identifier for this capability statement, represented as a URI
 version	Σ	0..1	string	Business version of the capability statement
 name	Σ I	0..1	string	Name for this capability statement (computer friendly)
 title	Σ	0..1	string	Name for this capability statement (human friendly)
 status	?! Σ	1..1	code	draft active retired unknown PublicationStatus (Required)
 experimental	Σ	0..1	boolean	For testing purposes, not real usage
 date	Σ	1..1	dateTime	Date last changed
 publisher	Σ	0..1	string	Name of the publisher (organization or individual)
 contact	Σ	0..*	ContactDetail	Contact details for the publisher

Kanta PHR:n CapabilityStatement

- CapabilityStatement –tyyppinen resurssi, jossa on kuvattu Kanta PHR:n sääntöjä
- Listaa tuetut toiminnot ja resurssit ja muita asioita, joiden toteuttamiseen on useita vaihtoehtoja standardissa
- CapabilityStatement pääosin listaa tuetut toiminnot, koska standardi on laaja, ja tukemattomat toiminnot vain jätetään pois
- Yksittäisten kenttien arvojen tulkinnasta ja arvojoukoista löytyy tarkempaa tietoa standardista:
<https://www.hl7.org/fhir/capabilitystatement.html>

Capability statement Simplifierissa

CapabilityStatement 'Finnish PHR's capability statement'

Canonical URL	http://phr.kanta.fi/capabilitystatement
Version	1.0
Published by	Kela
Status	Active (since 2018-06-15)

The capability statement for Kanta PHR service production environment

Contact Information

kantakehitys@kanta.fi

RESTful Interactions

Mode	Server
Description	See implementation guide for details

Resource Type	Profile	Read	Vread	Update	Delete	Create	SearchType	ConditionalCreate
CarePlan		✓	✓	✓	✓	✓	✓	✓
Observation		✓	✓	✓	✓	✓	✓	✓
Patient	Finnish PHR Patient profile	✓	✓	✓		✓	✓	✓
MedicationStatement		✓	✓	✓	✓	✓	✓	✓
MedicationAdministration		✓	✓	✓	✓	✓	✓	✓
Questionnaire		✓	✓	✓	✓	✓	✓	✓
QuestionnaireResponse		✓	✓	✓	✓	✓	✓	✓

Yhteenvedosta näkee silmäyksellä mitä toimintoja on, mutta tarkemmin rajapintoja on tulkittava käyttäen FHIR-standardia ja json-esitysmuotoa.

Simplifierissa on kaksi PHR:n CapabilityStatementtia, joista toinen kuvaa AT:ssa olevia toimintoja ja toinen tuotannossa olevia toimintoja.

Capability statement Simplifierissa

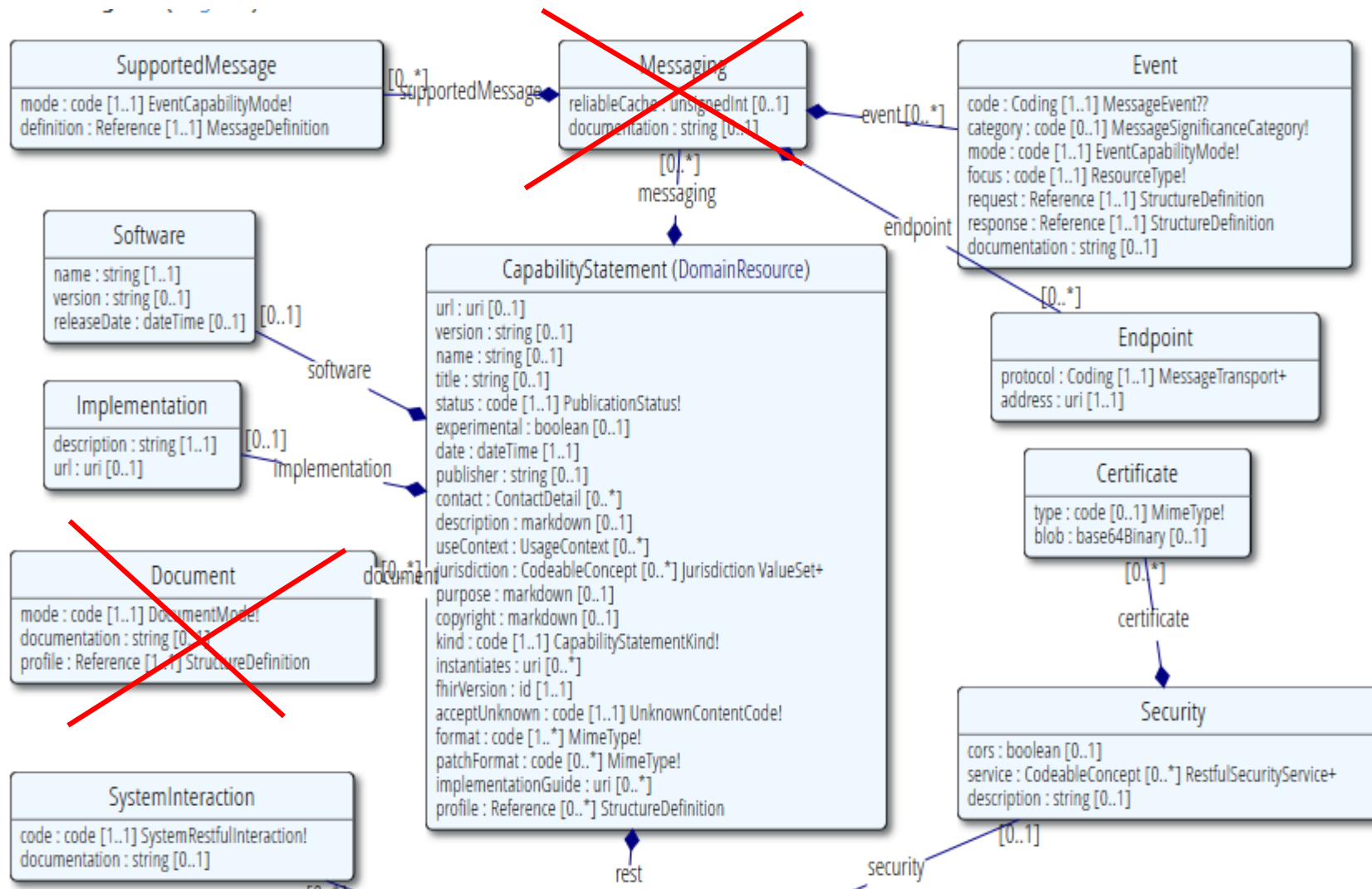
- Huom! Simplifierin json-näkymä ei näytä kaikkea tietoa vaan resurssi kannattaa downloadata!
- Tuotannon Capability statementin suora osoite on <https://simplifier.net/finnishphr/capabilitystatement3>

Rest?

https://en.wikipedia.org/wiki/Representational_state_transfer

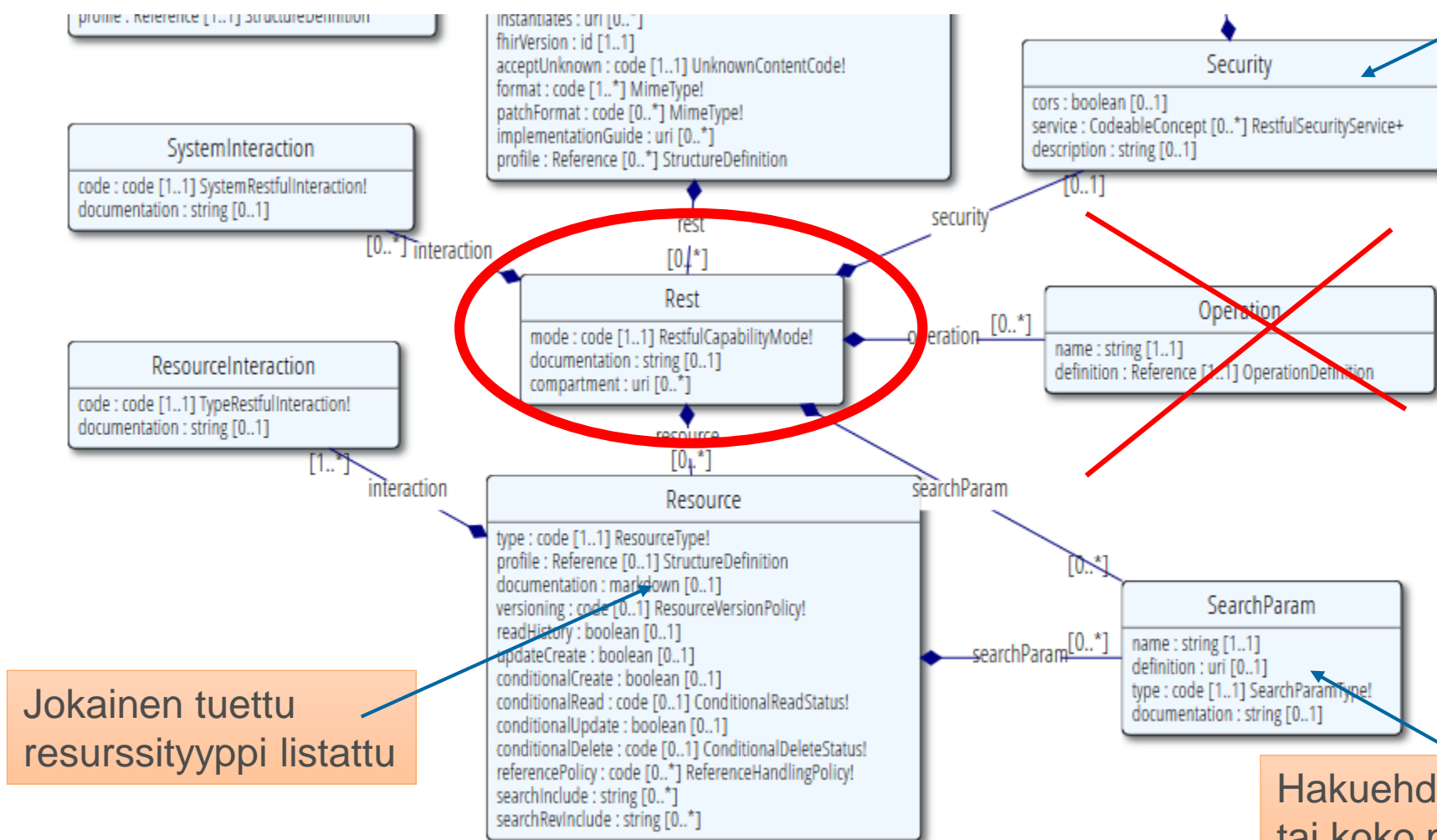
- HTTP-standardin tietty käyttötapa palveluiden toteuttamiseen
- Tilaton, eli palvelimelle ei ole istuntoa eikä palvelin ylläpidä tietoa edellisistä viesteistä
- Käyttää HTTP-metodeja toimintoihin:
 - GET, PUT, POST jne
- Laajalti käytössä, mutta ei itsessään kuvaa tarkasti toimintoja
- <https://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#relwwwrest>

CapabilityStatementin toteutetut osat



CapabilityStatementin toteutetut osat

OAuth2.0!



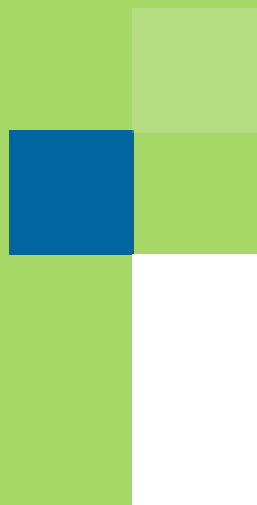
Jokainen tuettu resurssityyppi listattu

Hakuehdot joko resurssille tai koko palvelimella

Miten CapabilityStatementia hyödynnetään?

- Kanta PHR:n Simplifierissa oleva CapabilityStatement on tyyppiä "requirements" eli kuvaa miten palvelin itse toimii
- Esimerkiksi Sandboxin käyttöliittymä (joka pohjautuu Hapi-projektin FHIR-tester käyttöliittymään) lukee PHR-palvelimen CapabilityStatementista tuetut resurssit ja muodostaa näytölle listan sen pohjalta
- Eniten CapabilityStatementista on hyötyä (ainakin itselleni kehittäjänä!) tiiviinä yleiskuvana palvelimen toiminnoista

Omatietovarannon REST-apin toiminnot



Mitä tallentamisesta on CapabilityStatementissa?

- Osa CapabilityStatementin kentistä koskee kaikkia toimintoja
- acceptUnknown kertoo että tunnistamattomia tietotyypppejä, resursseja tai extensioineja ei hyväksytä
- Format kertoo, että palvelimen tukee json-muotoisia resursseja – muita muotoja ei ole

```
"acceptUnknown": "no",  
"format": [  
  "json"  
],
```

Mitä tallentamisesta on CapabilityStatementissa?

- REST-noodin alla löytyy lista resursseista, joita tuetaan
- Toiminnoista "create" ja "update" on tuettu
- Jokaiselle resurssille on listattu sen toiminnot
- muista listatuista toiminnoista lisää myöhemmin

```
{  
  "type": "Observation",  
  "interaction": [  
    {  
      "code": "read"  
    },  
    {  
      "code": "vread"  
    },  
    {  
      "code": "update"  
    },  
    {  
      "code": "create",  
      "documentation": "Conditional create allows only the  
use of identifier as condition"  
    },...  
  ]  
}
```

Mitä tallentamisesta on CapabilityStatementissa?

- Resurssit täytyy luoda ilman id:tä (updateCreate)
- Resurssille voi antaa luontiehdon:
 - ”jos tällä identifiaerilla löytyy jos resurssi, älä tee uutta”
- Päivitystä ei voi tehdä ehdollisesti
- Viittaukset tarkastetaan ja viittauksia Kanta PHR:n ulkopuolisille palvelimille ei sallita (esimerkiksi mittauksetulosten ”subject” –kenttä) – nämä ovat kenttäkohtaisesti jokaisessa profiilissa vielä

```
"updateCreate": false,  
"conditionalCreate": true,  
"conditionalUpdate": false,  
"referencePolicy": [  
  "literal",  
  "enforced",  
  "local"  
],
```


Tallenna-pyyntö

POST <http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation>

- ”Tallenna verenpainemittaus Kanta PHR:ään” ?
 - Mittaustulos on Observation-tyypin resurssi
 - Resurssi esitetään json-objektina
 - Tallennus PHR:ään tehdään lähettämällä resurssi PHR:n FHIR-palvelimelle
 - Pyyntö on restin mukaan POST ja varsinainen resurssi jsonina bodyssa
 - Paluusanomassa saa tiedon resurssin id:stä:
http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation/435b5586-7f65-439e-9d6d-9ca2ae07e11f/_history/1

Päivitä

PUT <http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation/435b5586-7f65-439e-9d6d-9ca2ae07e11f>

- ”Tämä on uusin versio mittaustuloksesta X”
- Päivitys on kuten tallennus, mutta sen REST-pyyntötyyppi on PUT postin sijaan
- FHIRin update
- Osoitteessa on myös päivitettävän resurssin tunniste ja bodyssa kokonaan uusi resurssi, joka korvaa aikaisemman
- Tunniste ei muutu päivittäessä!
- Vastauksena päivityksestä palvelu kertoo mikä uusi versio oli Location:
http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation/435b5586-7f65-439e-9d6d-9ca2ae07e11f/_history/2
- Vastauksen osoitteen osoittama resurssi tekee versiohaun

Mitä CapabilityStatement sanoo hauista?

- Hakuparametrit on listattu resurssikohtaisesti
- Palvelintasolla toimivia hakuparametreja ei ole CapabilityStatementissa, ja haut onkin aina kohdistettava tiettyyn tyyppiin
- Tuetut hakuparametrit on listattu CapabilityStatementissa, valitettavasti niiden modifiereita ja kaikkia tarkennoksia ei saa rakenteisesti kuvattua

```
"searchParam": [  
  {  
    "name": "_id",  
    "definition": "Observation.id",  
    "type": "token",  
    "documentation": "Logical id of this artifact"  
  },  
  {  
    "name": "_lastUpdated",  
    "definition": "Observation.meta.lastUpdated",  
    "type": "date",  
    "documentation": "When the resource version last changed"  
  },  
  {  
    "name": "_security",  
    "definition": "Observation.meta.security",  
    "type": "token",  
    "documentation": "If the viewing the resource is restricted. The following  
syntaxes are supported: [parameter]=[code] and [parameter]=[system]][code]."  
  },  
  {  
    "name": "date",  
    "definition": "Observation.effective[x]",  
    "type": "date",  
    "documentation": "Obtained date/time. If the obtained element is a  
period, a date that falls in the period."  
  }, ...  
]
```

Hakuparametrit: Observation-resurssin patient

- Definition-kentässä kerrotaan mihin resurssin jsonpath-kohtaan parametri kohdistuu, esim. "patient" kohdistuu subject-kenttään
- Hakuparametrin tyyppi kertoo millaisia arvoja hakuun voi antaa, tässä hakuparametrin tyyppi on kentässä oleva referenssi viittaus
- <https://www.hl7.org/fhir/search.html#reference>

```
{
    "name": "patient",
    "definition":
"Observation.subject",
    "type": "reference",
    "documentation": "The patient
that the observation is about"
},
```

Hakuparametrit

- Muita hakuparametrityyppejä ovat esimerkiksi päivämäärät ja koodit
- Näiden osalta noudatetaan standardin käytöstä, mutta osin suppeammin esimerkiksi modifierien osalta
- CapabilityStatementiin ei pysty kuvaamaan kaikkea hakuparametreista, mutta documentation-osassa on kerrottu jos jokin parametri on esimerkiksi pakollinen
- searchInclude ja searchRevInclude puuttuvat resurssien alta, eli niitä ei tueta. Osa kuvauksesta on kenttien puuttumista CapabilityStatementista, koska kuvaus tehdään listaamalla tuetut vaihtoehdot!

Hae

GET <http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation?patient=91db836a-b62a-49af-8e7a-a94ed90a41d5>

- ”Hae henkilön Y mittaustulokset”
- FHIRin search / search-type eli resurssityyppikohtainen haku
- Palauttaa tulosjoukon parametreja vastaavia mittaustuloksia
- Haun tulos palautetaan FHIRin bundlana, jossa on sisällä resurssit
- CapabilityStatementin interaktio ”search-type” on listattuna muiden interaktioiden tasolla

Mitä resurssin lukemisesta on CapabilityStatementissa?

- Lukemiseen on interaktiot "read" ja "vread"
- Read-toiminto palauttaa aina resurssin uusimman version
- Vread-toiminnolla, jos jostain syystä on tarve, voi hakea vanhoja versioita
- Historiasta on myös tieto: "readHistory": true, joka kertoo palauttaako vread vanhoja versioita resurssista

```
"interaction": [  
  {  
    "code": "read"  
  },  
  {  
    "code": "vread"  
  },  
]
```

Lue

GET <http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation/435b5586-7f65-439e-9d6d-9ca2ae07e11f>

- ”Anna mittautulos X”
- Viittaa suoraan resurssiin
- Fhirin read tai vread
- Osoite on [base][objektityyppi][id]
- GET-pyyntö
- Palauttaa vastauksessa sanoman bodyssa resurssin json-muodossa

Poista

DELETE <http://fhirsandbox.kanta.fi/phr-resourceserver/baseStu3/Observation/fcd8b4c9-a0c0-43d6-9774-7dd916137fab>

- Poistamisesta on varsin vähän CapabilityStatementissa
- FHIRin delete
- Toiminto on kuvattuna interaktiona resursseilla joilla se on tuettu
 - Patient ja Consent -tyyppien resursseja ei voi poistaa
- Lisäksi ehdollista poistamista ei tueta vaan resurssi on poistettava käyttämällä sen yksilöivää loogista tunnistetta eli id-kentän arvoa
 - "conditionalDelete": "not-supported",

Miten luvat ja FHIR-toiminnot liittyvät toisiinsa?

- Kanta PHR:n FHIR-toimintojen luvitus on toteutettu OAuth2.0 standardilla
- Luvat ovat resurssityyppikohtaisia, esim. lupa voi olla "Observation"-resurssiin
- Koska mittauksilla on omistaja, lupa on aina omistajakohtainen
- Resurssiin on erikseen lukuoikeus ja kirjoitusoikeus
 - Kirjoitusoikeudella myös päivitetään ja poistetaan

The logo for Kanta, featuring the word "Kanta" in a white, sans-serif font on a blue background. A small green plus sign is positioned above the letter 'a'.

Kanta

Kiitos!

Eeva Turkka
Software Architect
eeva.turkka@citrus.fi
@kantapalvelut

