



# HL7<sup>®</sup> FHIR<sup>®</sup> DevDays

JUNE | 2022

## Introduction to FHIR Search

Gino Canessa



<http://aka.ms/devdays-gino>



# About Me

---

- Principal Software Engineer @ Microsoft
  - MSR - Healthcare Standards and Interop
- Focused on FHIR since 2019
  - Infrastructure / Tooling
  - Code generation
  - Topic-Based Subscriptions
- DICOM-centric before
  - Apologies for the CDs!
  - They were a good idea at the time =)
- Contact
  - [Gino.Canessa@microsoft.com](mailto:Gino.Canessa@microsoft.com)
  - [Zulip](https://chat.fhir.org) (chat.fhir.org)
  - [YouTube](#) (FHIR Educational Content)

<http://aka.ms/devdays-gino>





# Disclaimers

---

- Search is **massive**
  - Touches most of the spec
  - Touches many use-cases
- Not going to cover everything in 45 minutes
- Defined != Implemented
- Not covering GraphQL

# What Are We Going To Cover, Then?

- Basics, Use Cases, and Examples
- Broad view of what is available
- Current documentation
  - <http://hl7.org/fhir/search.html>
- Work In Progress changes
  - **\*\*NOT YET APPROVED\*\***
  - Hopefully in R5, CI-Build
  - <http://build.fhir.org/branches/searchPage/search.html>

# So... Search?

- HTTP is most common, so we are using it today
  - GET vs POST
- Passing in data – search parameters
  - Mapping from URLEncoding to FHIR datatypes
  - Changing search inputs - Modifiers and prefixes, oh my
- What is returned?
  - Hint: not an array of resources
- Relating resources – chaining and including

# What is a Search Request, Anyway?

- A URL, based on:
  - a “search context” and
  - HTTP method
- Search Parameters: Inputs / Filters / Etc.
- Links
  - R4 – [HTTP: Search](#)
  - R4 – [Search: Introduction](#)
  - WIP – Search: [Introduction](#), [Transport Protocols](#)

Context	GET URL	POST URL
Server root	[baseUrl] - https://fhirdemo5.azurewebsites.net	[baseUrl]/_search - https://fhirdemo5.azurewebsites.net/_search
Resource	[baseUrl]/[resource] - https://.../Patient	[baseUrl]/[resource]/_search - https://.../Patient/_search
Compartment* - Resource ID and type	[baseUrl]/[resource]/[id]/[resource2] - https://.../Patient/12345/Encounter	[baseUrl]/[resource]/[id]/[resource2]/_search - https://.../Patient/12345/Encounter/_search

Building a URL –  
Context and Method

# And the Search Response?

- FHIR [Bundle](#) – type: “[searchset](#)”
    - Either specified MIME type or server defaults
  - Metadata
    - Links: self, next, self, previous, self, last, and... self!
    - Total
  - Entries
    - Resources you asked for (match),
    - possibly some you didn't (include),
    - and zero or more [OperationOutcome](#) (outcome).
  - Security note: you may not be allowed everything you ask for!
-



# Response Bundle: Metadata

- Bundle type: searchset
- Links: Paging
  - Next / Last / Previous
- Link: Self
  - What parameters a server used
- Total
- Timestamp


# Demo: Searching via GET and POST

- [Visual Studio Code](#) + [REST Client Extension](#)
  - [Http file on GitHub](#)
- [Postman](#)
  - [Collection file on GitHub](#)

# Search Parameters

- Defined as Canonical resources
  - Scope: System or Resource (can be multiple)
  - Expression: Map from name to one (or more) FHIR elements
  - Type: Map from URL-Encoded to FHIR datatypes
  - Core or IG
  - Lower kebab cased (hyphenated)
  - Core system parameters are prefixed with underscore (“\_”)
- Defined != supported... but listed in CapabilityStatement does
  - Implementation ‘name’ may vary
- R4 – [Search Parameter Registry](#)
- Bottom of each resource page

Type	Links	Description
composite	<a href="#">R4</a> , <a href="#">WIP</a>	A composite search parameter that combines a search on two values together.
date/dateTime	<a href="#">R4</a> , <a href="#">WIP</a>	Search parameter is on a date/time. The date format is the standard XML format, though other formats may be supported.
number	<a href="#">R4</a> , <a href="#">WIP</a>	Search parameter SHALL be a number (a whole number, or a decimal).
quantity	<a href="#">R4</a> , <a href="#">WIP</a>	A search parameter that searches on a quantity.
reference	<a href="#">R4</a> , <a href="#">WIP</a>	A reference to another resource (Reference or canonical).
special	<a href="#">R4</a> , <a href="#">WIP</a>	Special logic applies to this parameter per the description of the search parameter.
string	<a href="#">R4</a> , <a href="#">WIP</a>	Search parameter is a simple string, like a name part. Search is case-insensitive and accent-insensitive. May match just the start of a string. String parameters may contain spaces.
token	<a href="#">R4</a> , <a href="#">WIP</a>	Search parameter on a coded element or identifier. May be used to search through the text, display, code and code/codesystem (for codes) and label, system and key (for identifier). Its value is either a string or a pair of namespace and value, separated by a " ", depending on the modifier used.
uri	<a href="#">R4</a> , <a href="#">WIP</a>	A search parameter that searches on a URI (RFC 3986).



# Search Parameter Types



# Standard Search Parameters

- Common Use Cases
    - Global Scope
    - Prefixed with an underscore (prevent collisions)
    - `_id`, `_lastUpdated`, `_security`, `_tag`, etc.
  - Links
    - R4 – [Search: Standard Parameters](#)
    - WIP – [Search: Standard Parameters](#)
-

<b>Name</b>	_id
<b>Description</b>	Logical id of this artifact.
<b>Type</b>	token
<b>Expression</b>	Resource.id
<b>Use</b>	[base]/[Resource]?_id=[value]
<b>Example</b>	[base]/Patient?_id=1234
<b>Why?</b>	Return a search bundle instead of bare resource, etc.

## Demo: Searching by Id

---

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)

# Resource Search Parameters (Patient)

- Links
  - R4 – [Patient: Search Parameters](#)
  - R4 – [Search Parameter Registry: Patient](#)

Name	Type	Expression
active	token	Patient.active
address	string	Patient.address
address-city	string	Patient.address.city
		...

---

# Example: Searching a Patient by Family Name

<b>Name</b>	family
<b>Description</b>	A portion of the family name of the patient
<b>Type</b>	string
<b>Expression</b>	Patient.name.family
<b>Use</b>	[base]/Patient?family=[value]
<b>Example</b>	[base]/Patient?family=canessa

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)





# Search Result Parameters

- Not Search Parameters
  - Change the shape or form of results
  - Examples
    - Sorting (`_sort`)
    - Summary (`_summary`)
  - Links
    - R4 – [Search: Modifying Search Results](#)
    - WIP – [Search: Modifying Search Results](#)
-

# Demo: Sorting by Family Name

---

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)

<b>Name</b>	_sort
<b>Description</b>	Order results according to sorts on search parameters.
<b>Type</b>	token
<b>Expression</b>	n/a
<b>Use</b>	[base]/Patient?_sort=[value]
<b>Example</b>	[base]/Patient?_sort=family [base]/Patient?_sort=-family

# Modifiers and Prefixes

- Both are restricted by type
- Prefixes are limited to non-text (cannot delineate)
- Modifiers can change anything about the parameter
- Prefixes change *how* a match is determined
- One or the other – never both

# Modifier Details

- Attached to search parameter NAME with a colon
  - [parameter]:[modifier]=[value]
- Sample: missing
  - Modify the search parameter to a token (boolean - true|false) that tests whether resources have a value or not in the requested element.
  - Types that map to single elements (all but composite & special)
  - .../Patient?family:missing=true
- Links
  - R4 – [Search: Modifiers](#)
  - WIP – [Search: Modifiers](#)

<b>Name</b>	family
<b>Type</b>	string
<b>Default</b>	Case-insensitive, starts-with
<b>:contains</b>	Case-insensitive, contains
<b>:exact</b>	Case-sensitive, exact match
<b>:text</b>	Advanced text search...
<b>:missing</b>	Element has no value

## Demo: Patient Family Name Modifiers

---

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)

# Prefix Details

- Attached to search parameter VALUE as a prefix
  - [parameter]=[prefix][value]
- Non-string types (number, date/dateTime, quantity)
- Sample: ge
  - The value for the parameter in the resource is greater than or equal to the provided value.
  - .../Patient?birthdate=ge2022-01-01
- Links
  - R4 – [Search: Prefixes](#)
  - WIP – [Search: Prefixes](#)

# Demo: Patient Birth Date Prefixes

---

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)

<b>Name</b>	birthdate
<b>Type</b>	date
<b>Default</b>	Equality (eq)
<b>eq</b>	Equals
<b>ne</b>	Not Equals
<b>gt</b>	Greater Than
<b>lt</b>	Less Than
<b>ge</b>	Greater Than or Equals
<b>le</b>	Less Than or Equals

# Multiple Parameters or Values

- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)

Join Type	Process
AND	Parameters joined with ampersand (&)
OR (single parameter)	Values joined with comma (,)
OR (multiple parameters)	Use two queries =) Really – consider a batch
Special	Composite Search Parameters



# Linking Resources: References

- Establish relationships between resources
- Subtle differences in resolutions
- Links
  - GitHub – [Http file](#)
  - GitHub – [Postman collection](#)



# Chaining and Including

- Follow references in elements
  - Chaining is to *search* between resources (FHIR 'dot' notation)
  - Including is to *retrieve* additional resources (`_include`)
  - Both have a 'reverse' direction (`_has` and `_revinclude`)
  - Links
    - GitHub – [Http file](#)
    - GitHub – [Postman collection](#)
-



# And More!

- Batching Searches
- Transactions
- `_query`
- Messaging
- Contained Resources
- External References
- (See the search pages)



Questions?

---

Monday June 6	
3:00 PM - 3:45 PM - Room: 201	<b>Introduction to FHIR search</b> - Gino Canessa

Tuesday, June 7	
10:45 AM - 11:30 AM - Room: Ballroom A	<b>SMART on FHIR: Introduction</b> - Josh Mandel
1:20 PM - 1:50 PM - Room: 203	<b>Azure Health Data Services</b> - Doug Seven
2:55 PM - 3:40 PM - Room: 221	<b>Topic-based subscriptions</b> - Gino Canessa
4:10 PM - 4:55 PM - Room: 201	<b>SMART on FHIR: hands on</b> - Josh Mandel
6:00 PM - 8:00 PM - Room: 203	<b>Generating actionable insights from real world clinical and genomic data in FHIR</b> - Geralyn Miller
6:00 PM - 6:45 PM - Room: SpatialChat	<b>Pubtalk – JavaScript is for Uis</b> - Brian Postlethwaite - Gino Canessa
6:00 PM - 6:45 PM - Room: SpatialChat	<b>Pubtalk – Are we making this too complicated?</b> - Josh Mandel

Wednesday, June 8	
11:45 AM - 12:05 PM - Room: 304	<b>SMART Health Cards - standard, testing &amp; validation tools</b> - Christian Paquin
1:20 PM - 1:50 PM - Room: 203	<b>React to Events of Data Changes on FHIR Service</b> - Teng Li
4:10 PM - 4:30 PM - Room: Ballroom A	<b>TypeScript FHIR library</b> - Gino Canessa
5:00 PM - 5:45 PM - Room: 201	<b>FHIR Questionnaires and structured data capture with examples</b> - Brian Postlethwaite

Thursday, June 9	
11:10 AM - 11:30 AM - Room: 221	<b>FHIR Data Ingestion Tool: convert, anonymize, and publish</b> - Mr Ibrahim Kivanc
11:45 AM - 12:30 PM - Room: ABC	<b>Mapping FHIR data to parquet for analytics and machine learning</b> - Quan Wan
11:45 AM - 12:30 PM - Room: 201	<b>SMART on FHIR: what's new, what's next?</b> - Josh Mandel
2:00 PM - 2:45 PM - Room: 201	<b>FHIRPath by example</b> - Brian Postlethwaite
2:55 PM - 3:40 PM - Room: ABC	<b>SMART Health Links for verifiable clinical information - design discussion</b> - Josh Mandel





# HL7<sup>®</sup> FHIR<sup>®</sup> DevDays

JUNE | 2022

THANK YOU!

---

Gino Canessa



<http://aka.ms/devdays-gino>

