

EXPLORE

SPRB1969LV

API Versioning in Spring

Spencer Gibb
Spring Cloud Lead / Broadcom

#vmwareexplore #SPRB1969LV

Broadcom Proprietary and Confidential. Copyright © 2025 Broadcom.
All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.



Disclaimer

- Certain information in this presentation may outline Broadcom's general product direction.
- This presentation shall not serve to (i) affect the rights and/or obligations of Broadcom or its licensees under any existing or future license agreement or services agreement relating to any Broadcom software product; or (ii) amend any product documentation or specifications for any Broadcom software product.
- This presentation is based on current information and resource allocations and is subject to change or withdrawal by Broadcom at any time without notice.
- The development, release and timing of any features or functionality described in this presentation remain at Broadcom's sole discretion.
- Notwithstanding anything in this presentation to the contrary, upon the general availability of any future Broadcom product release referenced in this presentation, Broadcom may make such release available to new licensees in the form of a regularly scheduled major product release.
- Such release may be made available to licensees of the product who are active subscribers to Broadcom maintenance and support, on a when and if-available basis.
- The information in this presentation is not deemed to be incorporated into any contract.

Nice To Meet You!



Spencer Gibb

Spring Cloud Co-founder & Lead

X: [spencerbgibb](#)

LinkedIn: [/in/spencergibb](#)

Bluesky/blog: [spencer.gibb.us](#)

Rossen Stoyanchev

Spring engineering team

Spring Framework, Spring for GraphQL

Goals for Today

Brief intro to API versioning topic

Add some perspective

Tour of new features in Spring Framework 7 & Friends

Why is API versioning important?

Evolution

Iterative development

Manage change

The common advice

Not easy to come by

Typically, just a listing of common strategies

Roy Fielding says “DON’T”

Have you seen websites with versions?

Software engineering on the scale of decades

No need to anticipate breaking change

<https://www.infoq.com/articles/roy-fielding-on-versioning/>

“DON’T” version ?

Browser/human vs machine client

Favor compatibility (but mileage will vary)

Roy Fielding: “use new hostname”

Not a new API version, rather a new system/brand

New host or new paths => New site

Loose association between old and new

Deep change

Version in header or query param

Selective versioning by endpoint

Manage same data for multiple versions

Lightweight change

Favor
compatibility

Lightweight
change

Structural
change



Version by header,
query param, or
media type

Version by URL
(path or host)

Version from the start?

Not really

Add version header, or entity URLs later

Map request w/o version to unversioned endpoint

Selective Versioning by Endpoint

Selective update of request mappings

Endpoints without change support higher version

Examples

Microsoft [REST Guidelines](#)

Google guidelines on [API versioning](#)

GitHub [API versions](#)

StackExchange [Changelog](#)

API Versioning in Spring

Historically no explicit support

Possible, but not easy [spring-framework#33599](#)

First-class feature provides a common mechanism

Target nuances

<https://github.com/spring-projects/spring-framework/issues/33599>

Server Support

WebMvc and WebFlux config API versioning

@RequestMapping version attribute

Functional support with version RequestPredicate

Strategies to extract, parse, and validate versions

Demo server



Client Support

RestClient, WebClient support for ApiVersionInsertter

Default or per request version

Version in @HttpExchange interface

Demo clients



Test Support

MockMvc, WebTestClient support for ApiVersionInserter

API versioning setup for mock servers

Demo tests



Spring Cloud Support

Gateway Server WebMvc and WebFlux version predicate
Load Balancer to choose instance based on version

Demo Spring Cloud



Resources

<https://github.com/spencergibb/springone2025-versioning>

<https://github.com/spencergibb/spring-cloud-gateway-mvc-sample>

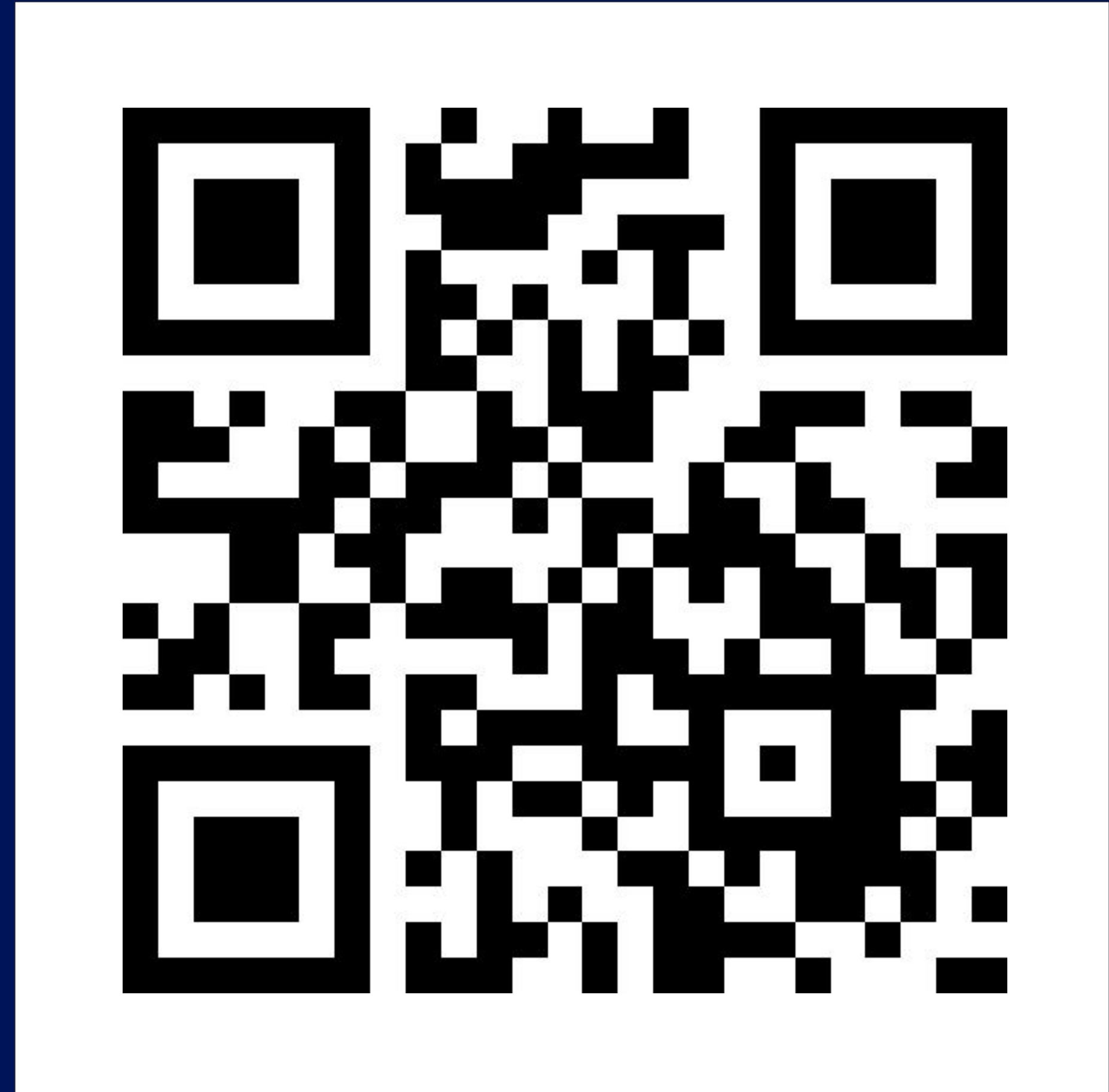
Route21VersionPredicate

Spring Framework 7 docs

<https://docs.spring.io/spring-framework/reference/7.0-SNAPSHOT/web/webmvc-versioning.html>

EXPLORE

Please take
your survey.



EXPLORE

Stay Connected

Let's continue the conversation

Follow me on X @spencerbgibb and 🦋 spencer.gibb.us

EXPLORE

<https://github.com/spencergibb/springone2025-versioning>

<https://github.com/spencergibb/spring-cloud-gateway-mvc-sample>

Route21VersionPredicate

<https://docs.spring.io/spring-framework/reference/7.0-SNAPSHOT/web/webmvc-versioning.html>

Spencer Gibb

X: spencerbgibb

LinkedIn: /in/spencergibb

Bluesky/blog: spencer.gibb.us