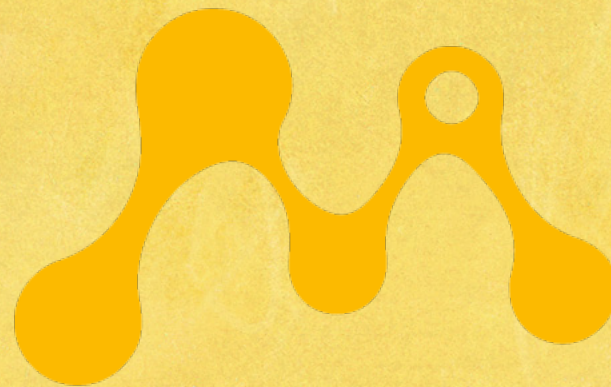


Getting Started with MicroProfile



Michael P. Redlich



Who's Mike?

- Bachelor of Science, Computer Science
- “Petrochemical Research Organization”
- Java Queue News Editor, InfoQ
- Leadership Council, Jakarta EE Ambassadors
- Amateur Computer Group of New Jersey

Objectives

- What is MicroProfile?
- Why MicroProfile?
- MicroProfile History
- Meet the MicroProfile APIs
- Get Started
- Live Demo (yea!)

What is MicroProfile?

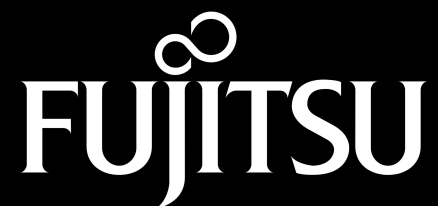
What is MicroProfile?

- A full suite of APIs for optimizing enterprise Java in a microservices architecture
- A collaboration of Java EE/Jakarta EE vendors

MicroProfile 2016



MicroProfile 2020



What is MicroProfile?

- An **open** standard platform that enables vendors to compete on implementation, price, or business model
- A **collaborative** standard and process that is driven by many vendors and individual developers rather than a single vendor

[MicroProfile - Collaborating to Bring Microservices to Enterprise Java](#)

What is MicroProfile?

- A **consistent** and **holistic** vision for all architectural tiers of the application
- A strong focus on adherence to the standard and **compatibility** between vendor implementations and versions of the specifications

[MicroProfile - Collaborating to Bring Microservices to Enterprise Java](#)

Why MicroProfile?

Why MicroProfile?

“Red Hat, IBM, Payara, Tomitribe and the London Java Community believe that enterprise Java is a solid foundation on which to build the next generation and the MicroProfile (which may ultimately become a submission for a standard specification) can make it easier and provide portability between vendor’s implementations.”

[MicroProfile - Collaborating to Bring Microservices to Enterprise Java](#)

Why MicroProfile?

“The initial goal is to provide developers who are most comfortable with enterprise Java (Java EE if you will) a starting point to work with microservices in a non-vendor specific way (which is what they’re used to from the Java EE world)...”

Martijn Verbug - Principal Engineering Group Manager at Microsoft

Why MicroProfile?

“...the aim is then for the developer community to actually drive what they feel they need in microservices API/runtime, so instead of the vendors guessing that you might want security, or logging or discovery or whatever. It will be up to the MicroProfile community to help define what might go in or what might stay out.”

Martijn Verbug - Principal Engineering Group Manager at Microsoft

MicroProfile History

A Concept is Born

- MicroProfile introduced June 27, 2016 at Red Hat's DevNation conference featuring:
 - Mark Little (Red Hat)
 - Alasdair Nottingham (IBM)
 - Theresa Nguyen (Tomitribe)
 - Mike Croft (Payara)
 - Martijn Verburg (London Java Community)

MicroProfile 1.0

- Released September 19, 2016 at JavaOne
- Introduced three initial APIs:



MicroProfile Joins Eclipse Foundation

- Announced December 14, 2016 by Mike Milinkovich, Executive Director at the Eclipse Foundation
- Ensured that MicroProfile remained vendor-neutral

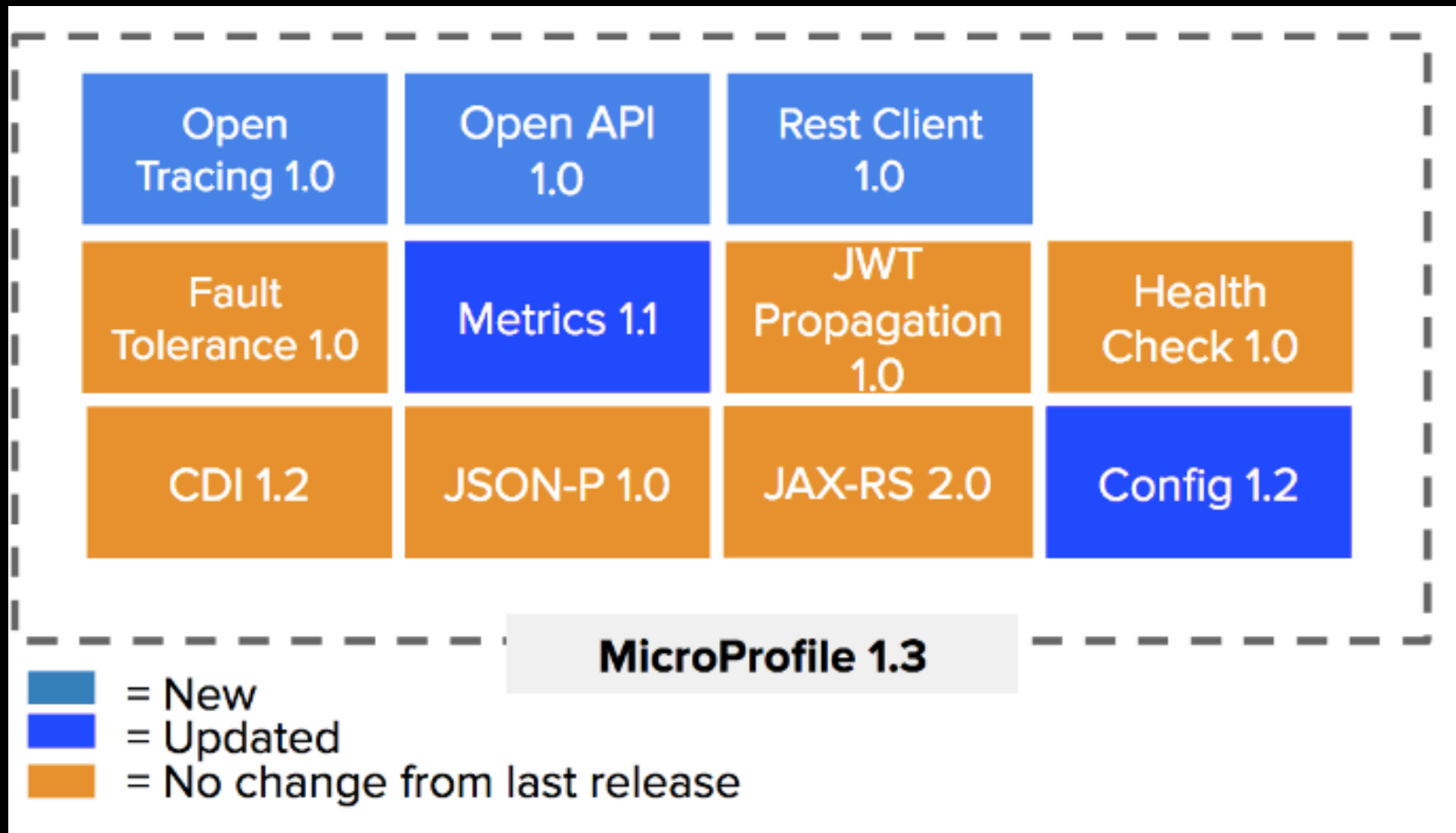
MicroProfile 1.1

- Released August 2017
- Introduced the `Config` API

MicroProfile 1.2

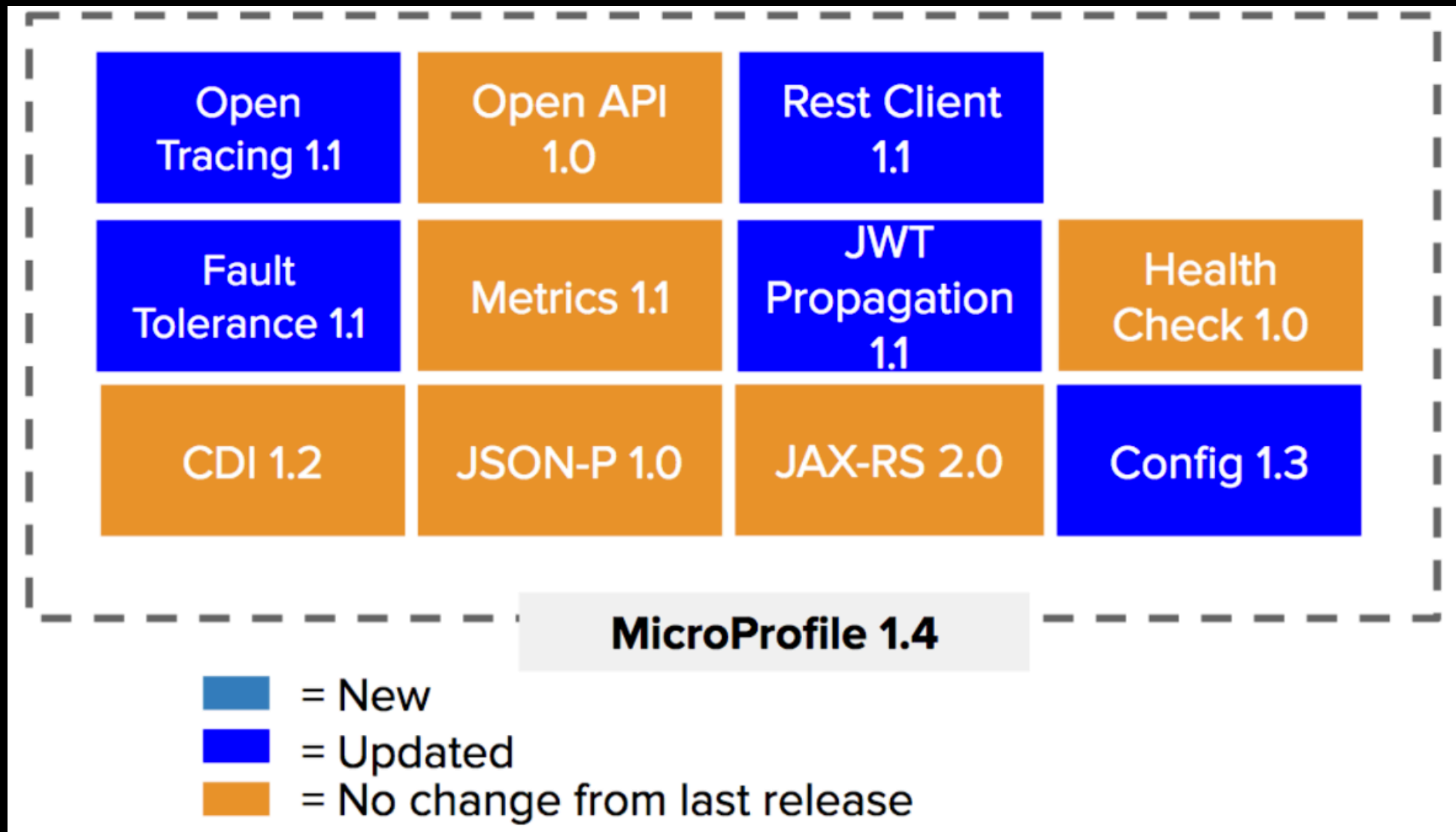
- Released September 2017
- Introduced the following APIs:
 - Fault Tolerance
 - Metrics
 - JWT Propagation
 - Health Check

MicroProfile 1.3



Released January 2018

MicroProfile 1.4



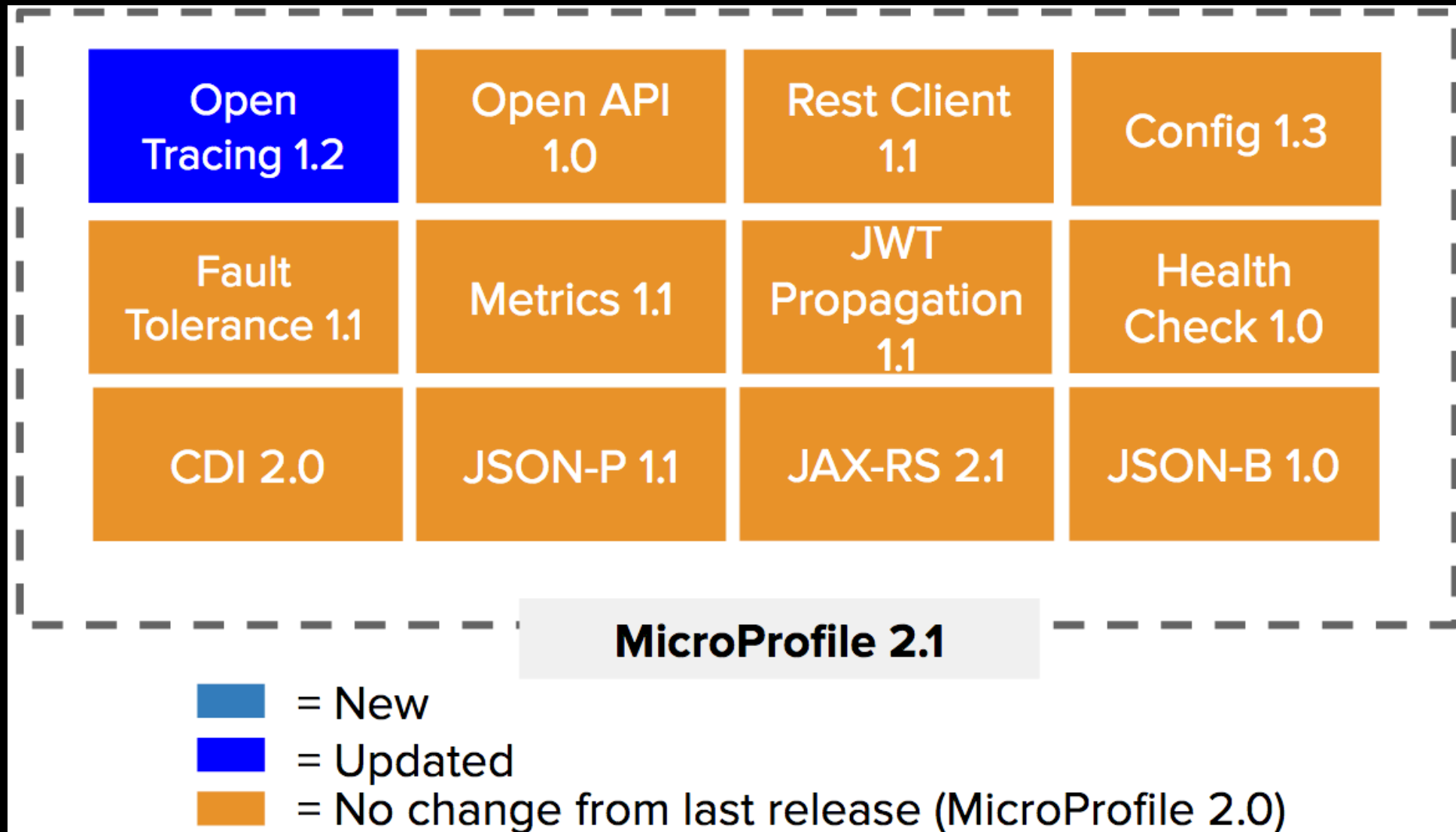
Released June 2018

MicroProfile 2.0



Released June 2018

MicroProfile 2.1






Released October 2018

MicroProfile 2.2

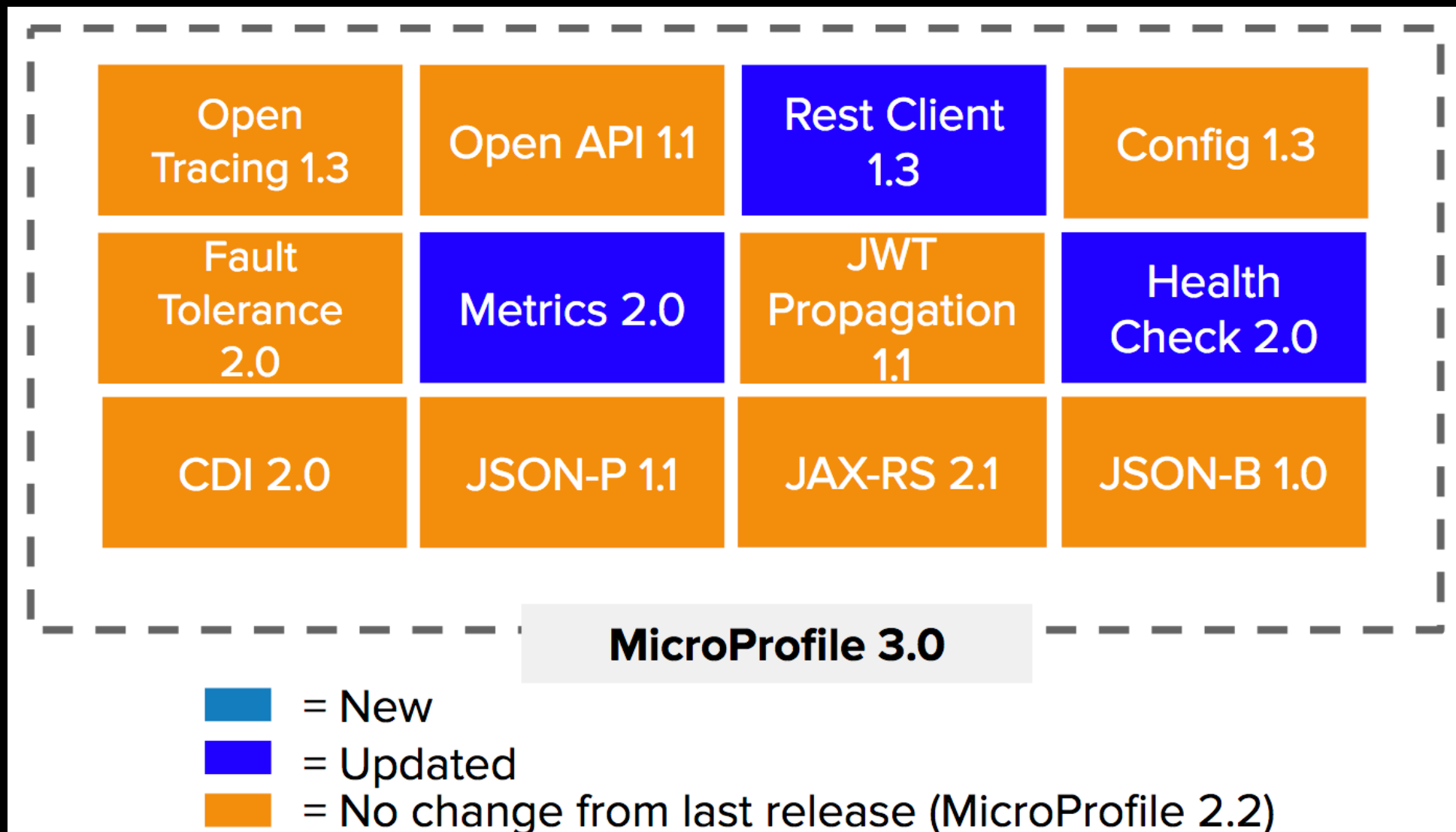
Open Tracing 1.3	Open API 1.1	Rest Client 1.2	Config 1.3
Fault Tolerance 2.0	Metrics 1.1	JWT Propagation 1.1	Health Check 1.0
CDI 2.0	JSON-P 1.1	JAX-RS 2.1	JSON-B 1.0

MicroProfile 2.2

 = New
 = Updated
 = No change from last release (MicroProfile 2.1)

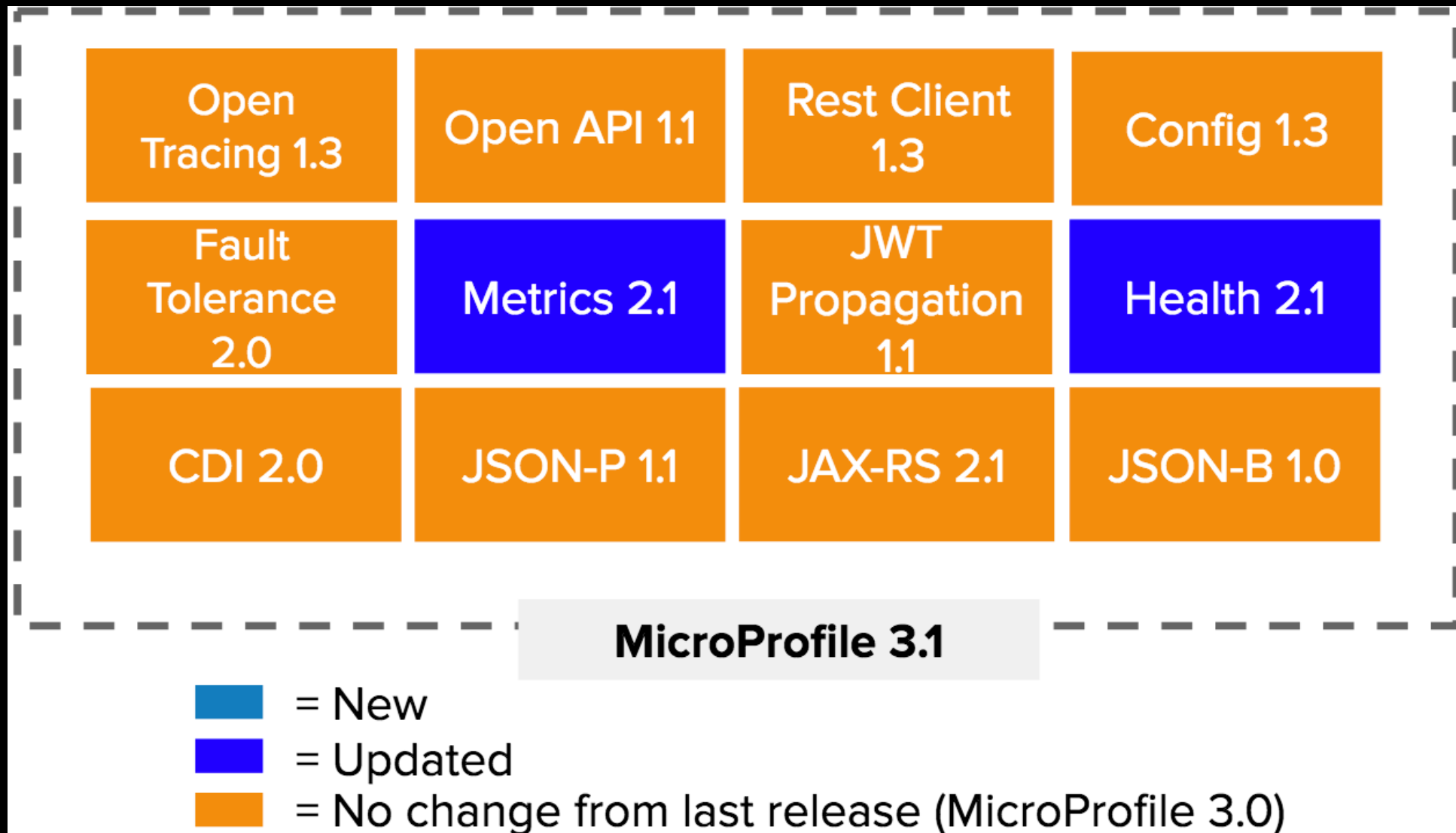
Released February 2019

MicroProfile 3.0



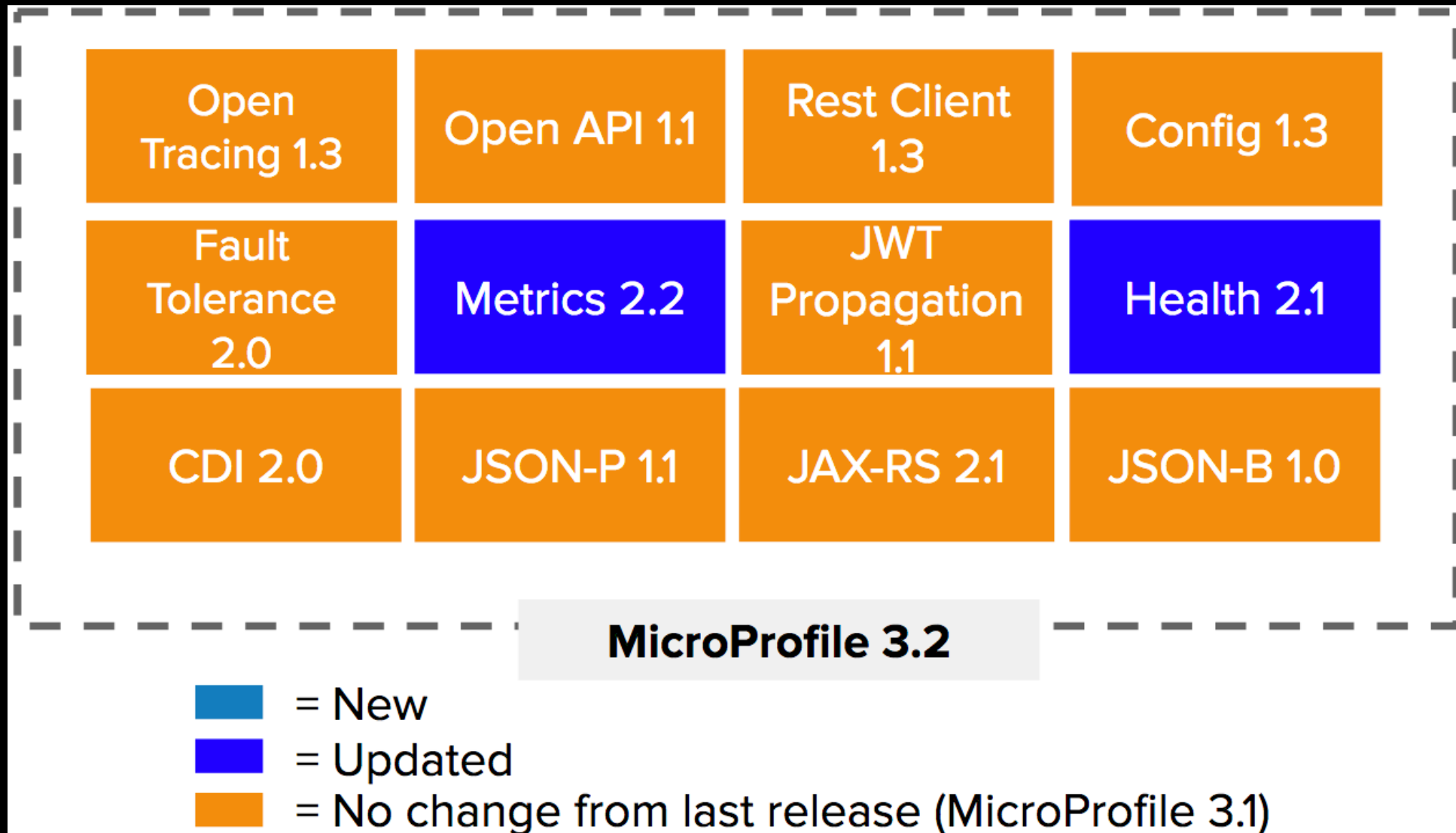
Released June 2019

MicroProfile 3.1



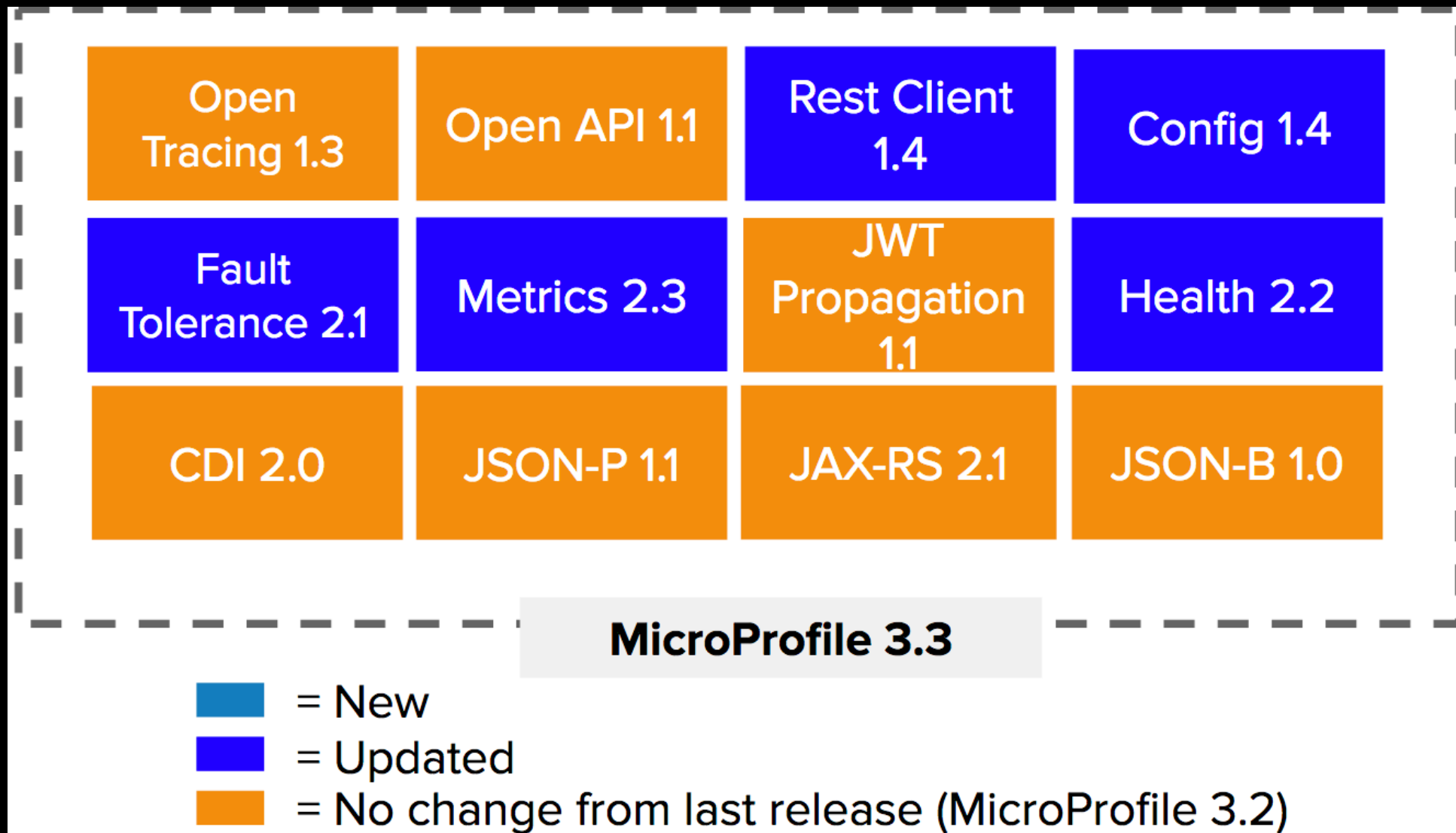
Released October 2019

MicroProfile 3.2



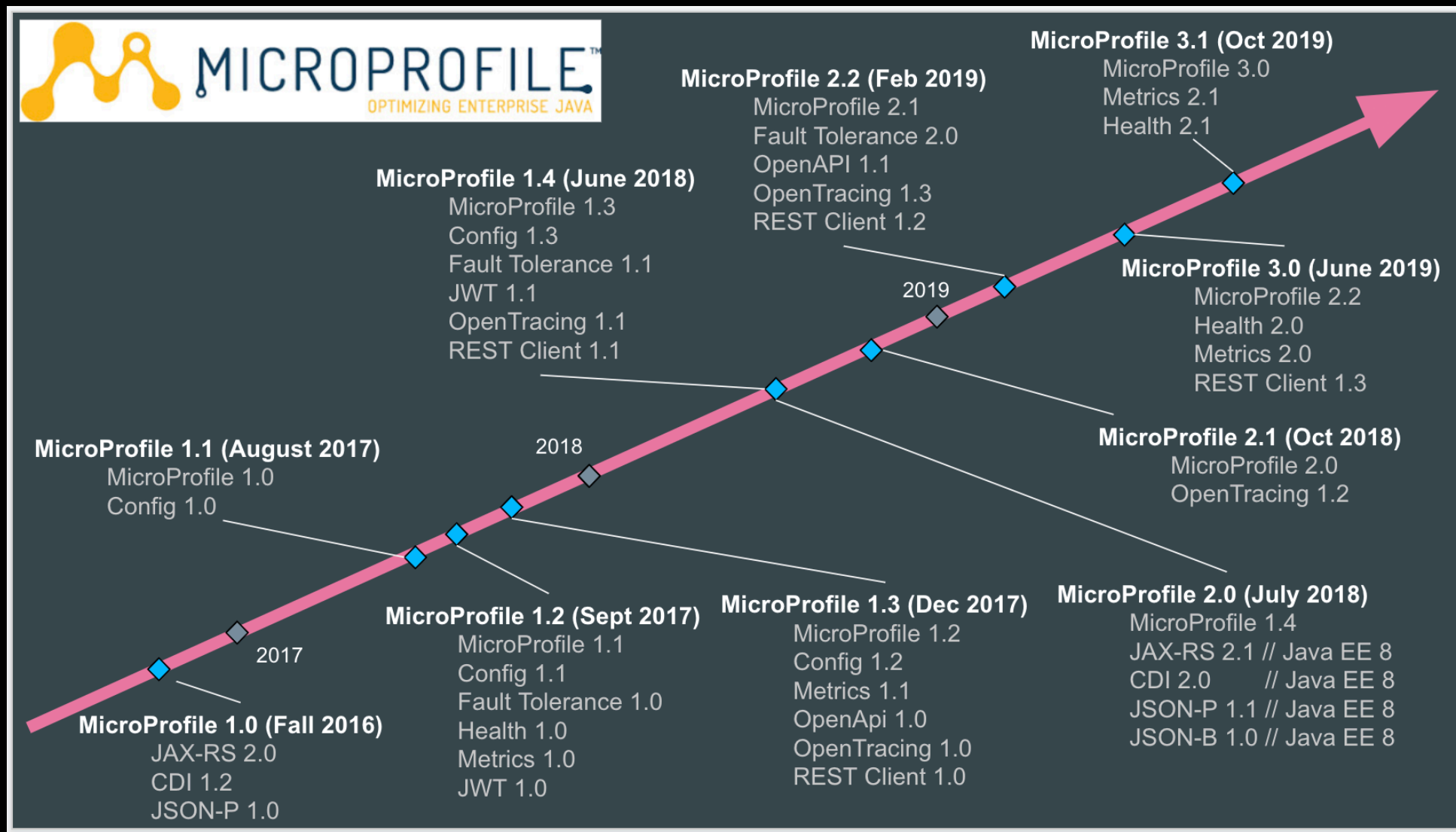
Released November 2019

MicroProfile 3.3



Released February 2020

Timeline



Meet the MicroProfile APIs

Context & Dependency Injection (CDI)

- Manages the lifecycle of stateful components through type-safe injection into client objects
- Introduced in MicroProfile 1.0
- Based on JSR 365
- Current version: 2.0

Java API for JSON Processing (JSON-P)

- Processes JSON messages that produce and consume streamed JSON text in binding with Java objects
- Introduced in MicroProfile 1.0
- Based on JSR 374
- Current version: 1.1

Java API for RESTful Web Services (JAX-RS)

- Provides support for creating web services as defined by the Representational State Transfer (REST) architectural pattern
- Introduced in MicroProfile 1.0
- Based on JSR 370
- Current version: 2.1

Config

- Externalizes configuration from microservices
- Introduced in MicroProfile 1.1
- Based on JSR 382
- Current version: 1.4

Fault Tolerance

- Leverages strategies to guide the execution and result of business logic upon some failure within the application
- Introduced in MicroProfile 1.2
- Current version: 2.1

Metrics

- Provides a unified way to export monitoring data (telemetry)
- Introduced in MicroProfile 1.2
- Current version: 2.3

JWT Propagation

- Provides application callers with the ability to authenticate themselves using a JWT token
- Introduced in MicroProfile 1.2
- Current version: 1.1

Health Check/Health

- Determines the health of a computing node and replace it, if necessary
- Introduced in MicroProfile 1.2
- Current version: 2.2

Open Tracing

- Allows easier tracing of the flow of a request across service boundaries in a microservices environment
- Introduced in MicroProfile 1.3
- Based on the Open Tracing Specification
- Current version: 1.3

Open API

- Provides a unified API for the Open API specification
- Introduced in MicroProfile 1.3
- Current version: 1.1

Rest Client

- A type-safe approach to invoke RESTful services over HTTP
- Introduced in MicroProfile 1.3
- Current version: 1.4

Java API for JSON Binding (JSON-B)

- A standard binding layer for converting Java objects to/from JSON messages
- Introduced in MicroProfile 2.0
- Based on JSR 367
- Current version: 1.0

Reactive Streams Operators

- Allows two different asynchronous libraries the ability to stream data to/from each other
- Standalone API
- Introduced in February 2019
 - MicroProfile 2.2
- Current version: 1.01

Reactive Context Propagation

- Obtains `CompletableFuture` objects that are backed by managed threads with the ability to capture context from threads
- Standalone API
- Introduced in July 2019
 - MicroProfile 3.0
- Current version: 1.02

Reactive Messaging

- Provides asynchronous messaging support for Reactive Streams Operators
- Introduced in July 2019
 - MicroProfile 3.0
- Current version: 1.0x

GraphQL

- Enables developers to build portable GraphQL-based applications
- Introduced in February 2020
 - MicroProfile 3.3
- Current version: 1.01

Let's Get Started!



MicroProfile Starter

- Generates a basic MicroProfile application with your choice of APIs
- Same concept as Spring Initializr

Live Demo

Demo Application

CDI and Config APIs

- Add dynamic configuration to an Apache TomEE microservice using CDI
- Based on a tutorial by David Salter, author, blogger, founder of the West Yorkshire Java Users Group and MicroProfile committer
 - MicroProfile Config on TomEE
- Config property: `beer.type`

CDI and Config APIs

- ConfigSource
 - an interface for building custom configuration
- System Properties (ordinal = 400)
- Environment Variables (ordinal = 300)
- Property Files (ordinal = 100)

System Properties

```
<configuration>  
  <args>-Dbeer.type=stout</args>  
  <context>ROOT</context>  
  <version>3.3</version>  
  <tomeeClassifier>pom</tomeeClassifier>  
</configuration>
```

Environment Variables

```
$ export BEER_TYPE=IPA
```

Property Files

```
beer.type=porter
```

Live Demo

MicroProfile Resources

- <https://microprofile.io>
- <https://microprofile.io/blog>
- <https://microprofile.io/projects>
- <https://start.microprofile.io>

Further Reading

The screenshot shows a search interface with a search bar containing 'microprofile'. A dropdown menu displays the following results: 'micronaut', 'microprofile' (highlighted), 'micropro 1.0', 'microprofile 1.0', 'microprofile 1.3', and 'microservices'. The interface includes a sidebar with 'Culture & Methods' and 'Dev' sections, and a top navigation bar with a user profile 'MICHAEL' and a notification bell. A 'QCon' banner for the Software Dev Conference is also visible, listing events in New York, Munich, and SF for 2020.

microprofile

micronaut

microprofile

micropro 1.0

microprofile 1.0

microprofile 1.3

microservices

Culture & Methods

Dev

Observability

APM

QCon

Software Dev Conference

New York Jun 15-19, 2020

Munich Oct 19-21, 2020

SF Nov 16-20, 2020

Software Trends 2020

MICHAEL

Acknowledgements

- MicroProfile Community
- Emily Jiang
- Amelia Eiras
- #shareMPknowledge

Contact Info

`mike@redlich.net`

`@mpredli`

`redlich.net/portfolio`

`github.com/mpredli01`

Thanks!

