

【MicroProfile開発ガイド】 MicroProfile Health

2018/06

日本アイ・ビー・エム システムズ・エンジニアリング株式会社



Disclaimer

- この資料は日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社の正式なレビューを受けておりません。
- 当資料は、資料内で説明されている製品の仕様を保証するものではありません。
- 資料の内容には正確を期するよう注意しておりますが、この資料の内容は2018年6月現在の情報であり、製品の新しいリリース、PTFなどによって動作、仕様が変わる可能性があるのでご注意ください。
- 今後国内で提供されるリリース情報は、対応する発表レターなどでご確認ください。
- IBM、IBMロゴおよびibm.comは、世界の多くの国で登録されたInternational Business Machines Corporationの商標です。他の製品名およびサービス名等は、それぞれIBMまたは各社の商標である場合があります。現時点でのIBMの商標リストについては、www.ibm.com/legal/copytrade.shtmlをご覧ください。
- 当資料をコピー等で複製することは、日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社の承諾なしではできません。
- 当資料に記載された製品名または会社名はそれぞれの各社の商標または登録商標です。
- JavaおよびすべてのJava関連の商標およびロゴはOracleやその関連会社の米国およびその他の国における商標または登録商標です。
- Microsoft, WindowsおよびWindowsロゴは、Microsoft Corporationの米国およびその他の国における商標です。
- Linuxは、Linus Torvaldsの米国およびその他の国における登録商標です。
- UNIXはThe Open Groupの米国およびその他の国における登録商標です。

目次

- MicroProfile Health概要
 - MicroProfile Healthとは
 - 利用ケース
 - フィーチャーの有効化
- MicroProfile Healthの使用方法
- 参考リンク

MicroProfile Health概要

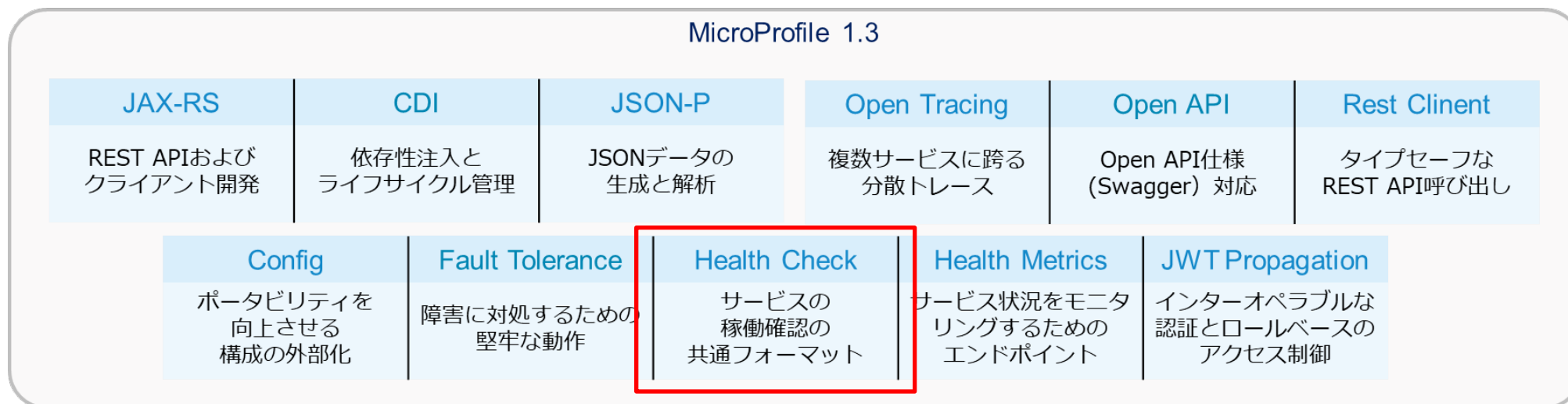
MicroProfile Healthとは

MicroProfile Healthは、サービスの正常性を検査するための機能を提供します。

サービスが自分の正常性を報告し、全体的な正常性ステータスを定義済みのエンドポイントに公開することができます。

ヘルスチェックは、依存関係、エンドポイントへの正常な接続、システムプロパティ、データベース接続、または必要なリソースの可用性など、サービスが必要とするものをチェックすることができます。

WAS LibertyではmpHealth-1.0フィーチャーが提供されています。



利用ケース

MicroProfile Healthは、Kubernetesのヘルスチェックなどマシンツーマシンのメカニズムとしての利用が意図されています。

(参考情報)

Kubernetesでは2種類のヘルスチェックがあります。

- LivenessProbe

コンテナが実行中かどうかを示します。Liveness Probeが失敗した場合、コンテナを再起動します。

- ReadinessProbe

コンテナがリクエストを処理できる状態かどうかを示します。Readiness Probeが失敗した場合、エンドポイントコントローラは、Podに一致するすべてのサービスのエンドポイントからPodのIPアドレスを削除します。

- Configure Liveness and Readiness Probes

<https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-probes/>

フィーチャーの有効化

WAS Liberty上でMicroProfile Healthの機能を有効化するためには、該当サーバーのserver.xmlにmpHealth-1.0フィーチャーを設定します。

```
<featureManager>  
    <feature>mpHealth-1.0</feature>  
</featureManager>
```

mpHealth-1.0フィーチャーは、MicroProfile-1.3を設定することでも有効化（暗黙的ロード）されます。

また、mpHealth-1.0フィーチャーを設定した場合、以下にリストされたフィーチャーも有効化されます。

- cdi-1.2
- jndi-1.0
- json-1.0
- servlet-3.0
- servlet-3.1

MicroProfile Healthの使用方法

MicroProfile Healthの使用方法（1/4）

ヘルスチェックを提供するためには、org.eclipse.microprofile.health.HealthCheckインターフェースを実装します。

フィーチャーが提供するエンドポイント（/health）を呼び出すと、@Healthアノテーションを付与されたカスタムHealthCheckクラスを全て自動検出してそのチェックロジックを実行、各々の結果を統合して返します。

■ヘルスチェック用インターフェース

クラス	
org.eclipse.microprofile.health.HealthCheck	
メソッド	説明
HealthCheckResponse call()	ノードの健全性の検証ロジックを実行し、結果をHealthCheckResponseに格納して返すように実装する

MicroProfile Healthの使用方法 (2/4)

以下はヘルスチェックのサンプルです。

実際にはヘルスチェックの結果を受けて、"UP"または"DOWN"を返すことになります。

```
@Health
@ApplicationScoped
public class MyCheck implements HealthCheck {

    @Override
    public HealthCheckResponse call() {

        // ここでヘルスチェックを実行する

        // ヘルスチェックの結果が正常だった場合、HealthCheckResponseにUPを設定して返す
        return HealthCheckResponse.named("MyCheck")
            .withData("key1", "val1")
            .withData("key2", "val2")
            .up()
            .build();

    }
}
```

MicroProfile Healthの使用方法（3/4）

/health エンドポイントへの HTTP 要求は、以下のいずれかの状況コードを含む応答を返します。

状況コード	説明
200	ヘルスチェックの結果が正常であることを示します。 つまり"outcome": "UP"のケースです。
503	ヘルスチェックの結果が異常であることを示します。 つまり"outcome": "DOWN"のケースです。
500	ヘルスチェックの一部または全体が処理できなかったことを示します。 プロシージャラーにエラーがある可能性も考えられます。

HTTP応答の形式（JSON）は以下の通りです。

JSON要素	説明
outcome	ヘルスチェック全体の結果が全て正常の場合"UP"、それ以外の場合"DOWN"。
checks	個々のヘルスチェックの結果が含まれます。
name	ヘルスチェックの名前です。named()メソッドで設定します。
state	ヘルスチェックの結果が正常の場合"UP"、異常の場合"DOWN"。 チェック結果に応じてup()/down()のメソッドで設定します。
data	withData()メソッドでヘルスチェックに付与したい情報を設定できます。

MicroProfile Healthの使用方法 (4/4)

以下は応答に含まれるjsonのサンプルです。

```
{
  "outcome": "UP",
  "checks": [
    {
      "name": "MyCheck",
      "state": "UP",
      "data": {
        "key1": "val1",
        "key2": "val2"
      }
    }
  ]
}
```

"checks"には複数のヘルスチェックが含まれ、全ての"state"が"UP"だった場合、"outcome"は"UP"となります。1つでも"state"が"DOWN"だった場合、"outcome"は"DOWN"となります。

参考リンク

- IBM Knowledge Center 「MicroProfile ヘルス・チェックの実行」

https://www.ibm.com/support/knowledgecenter/ja/SS7K4U_liberty/com.ibm.websphere.wlp.zseries.doc/ae/twlp_microprofile_healthcheck.html

- Adding health reports to microservices

<https://openliberty.io/guides/microprofile-health.html>

- Configure Liveness and Readiness Probes

<https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-probes/>

- Package org.eclipse.microprofile.health Class HealthCheckResponse

<https://openliberty.io/docs/ref/microprofile/1.3/#package=org/eclipse/microprofile/health/package-frame.html&class=org/eclipse/microprofile/health/HealthCheckResponse.html>

- Package org.eclipse.microprofile.health Class HealthCheckResponseBuilder

<https://openliberty.io/docs/ref/microprofile/1.3/#package=org/eclipse/microprofile/health/package-frame.html&class=org/eclipse/microprofile/health/HealthCheckResponseBuilder.html>

End of File