

# docker-composeを使用したマルチコンテナアプリケーションの管理

## 概要

docker-composeを使用すると、マルチコンテナアプリケーションを定義および管理できます。使いやすい定義ファイルを使用することにより、このようなアプリケーションスタックの展開が簡単になります。

## 目的

この記事では、docker-composeを使用してマルチコンテナアプリケーションを作成する方法について説明します。

## 所要時間

この記事の理解には20分ほどを要します。

## 目標

コンテナベースの独自のアプリケーションスタックを作成できるようになります。

## 要件

アプリケーションコンテナイメージ、またはアプリケーションの構築に使用する関連するソースファイル。

発行日: 11/12/2025

## 目次

- 1 マルチコンテナアプリケーションの管理に関するツール 3
- 2 マルチコンテナアプリケーションの作成 5
- 3 マルチコンテナアプリケーションの管理 9
- 4 法的事項 10
- A GNUフリー文書利用許諾契約書(GFDL) 11

# 1 マルチコンテナアプリケーションの管理に関するツール

docker-composeは、マルチコンテナアプリケーションを作成するためのツールです。デフォルトではDockerが使用されますが、DockerはSLE Microには存在しません。Dockerをバイパスして、代わりにPodmanを使用するには、`podman-docker`スクリプトを使用します。したがって、既存のスクリプトを、DockerではなくPodmanを使用するように変更する必要はありません。以下のセクションでは、ツールの詳細について説明します。

## 1.1 Podmanについて

PodmanはPod Manager Toolの略称です。Podmanはデーモンレスコンテナエンジンであり、コンテナとコンテナイメージを使用してアプリケーションを実行および展開できます。Podmanは、コンテナを管理するためのコマンドラインインタフェースを提供します。

Podmanはデーモンを持たないため、systemdとの統合が提供されています。これにより、systemdユニット経由でコンテナを制御できるようになります。このようなユニットを既存のコンテナ向けに作成できるほか、システムにコンテナが存在しない場合にコンテナを起動できるユニットを生成することもできます。Podmanはコンテナ内でsystemdを実行できます。

Podmanを使用すると、コンテナをポッドとして編成できます。ポッドは同じネットワークインタフェースとリソースを共有します。コンテナのグループをポッドに編成する代表的なユースケースは、データベースを実行するコンテナ、およびそのデータベースにアクセスするクライアントが含まれるコンテナです。

### 1.1.1 Podmanのインストール

PodmanはデフォルトでSLE Microに含まれます。ただし、Podmanが見つからない場合は、次の説明に従ってインストールできます。

1. 次のコマンドを実行します。

```
> sudo transactional-update pkg install podman*
```

2. システムを再起動して、新しいスナップショットで起動します。

## 1.2 podman-dockerについて

podman-dockerは、ユーザが実行するdockerコマンドを、渡された引数と同じ引数を使用して対応するpodmanコマンドに変更するbashスクリプトです。したがって、すべてのDockerスクリプトを変更せずに使用できます。

### 1.2.1 podman-dockerのインストール

podman-dockerは、デフォルトではSLE Microにインストールされていません。これをインストールするには、次の手順に従います。

1. 次のコマンドを実行して、podman-dockerパッケージをインストールします。

```
transactional-update pkg install podman-docker
```

2. システムを再起動して、最新のスナップショットに切り替えます。

## 1.3 docker-composeについて

docker-composeは、マルチコンテナアプリケーションを管理するためのツールです。docker-composeでは、単一ホスト上に複数の隔離された環境を設定しながら、環境間で変数を使用することができます。docker-composeを使用すると、変更されたコンテナのみを再作成することができ、マルチコンテナアプリケーション全体を破棄する必要はありません。

### ! 重要: dockerコマンドはpodmanコマンドに置き換え

SLE Microでは、docker-composeを実行すると常にpodman-dockerスクリプトを使用してPodmanを呼び出します。これは、SLE MicroにはDockerが存在しないためです。

### 1.3.1 docker-composeのインストール

docker-composeがシステムに存在しない場合は、次の手順に従ってインストールできます。

1. 次のコマンドを実行します。

```
> sudo transactional-update pkg install docker-compose
```

2. インストールが完了したら、システムを再起動して新しいスナップショットで起動します。

## 2 マルチコンテナアプリケーションの作成

マルチコンテナアプリケーションを作成するには、次の手順に従います。

1. 設定ファイル `compose.yml` を作成します。詳細については、[2.1項「compose.ymlの作成」](#) を参照してください。
2. 必要なディレクトリ構造を準備します。`compose.yml` ファイルは、作業ディレクトリの最上位に置くことをお勧めします。
3. 必要に応じて、コンテナ化されたアプリケーションで使用されるサービスに固有の、独自のコンテナファイルを記述します。たとえば、Goアプリケーションを展開するには、必要な設定と依存関係を含む、Goアプリケーションのコンテナファイルを作成します。作業ディレクトリ内にサービスごとにサブディレクトリを作成し、そこにサービス固有のファイルを配置することを推奨します。
4. マルチコンテナアプリケーションを展開します。詳細については、[2.2項「マルチコンテナアプリケーションの展開」](#) を参照してください。

### 2.1 compose.ymlの作成

マルチコンテナアプリケーションを作成するには、`compose.yml` ファイルを作成し、できれば作業ディレクトリに配置する必要があります。このファイルは1つのファイルにすることも、フラグメントや拡張機能を利用して、よりきめ細かいアプローチを使用することもできます。複数の `docker-compose` ファイルをマージして、アプリケーションモデル全体を定義することもできます。

`compose.yml` ファイルはアプリケーションを定義します。以下の要素を含めることができます。

#### service

サービスとはアプリケーションのコンピューティングコンポーネントです。定義に関する詳細については、[2.1.1項「サービスの定義」](#) を参照してください。

## networks

`network`ステートメントを使用してカスタムネットワークを定義し、特定のサービスをカスタムネットワークに割り当てることができます。詳細については、[2.1.2項「ネットワークの定義」](#)を参照してください。

## volumes

コンテナエンジンによって管理されるディレクトリで、サービスはここにデータを保存および共有します。

## 環境変数

サービスに渡す環境変数のリストを使用しなければならない場合もあります。詳細については、[environment variables reference \(https://docs.docker.com/compose/environment-variables/\)](https://docs.docker.com/compose/environment-variables/) [🔗](#)を参照してください。

## configs

サービスに必要なすべての設定ファイルを`configs`セクションで宣言する必要があります。詳細については、[configs definition \(https://docs.docker.com/compose/compose-file/08-configs/\)](https://docs.docker.com/compose/compose-file/08-configs/) [🔗](#)を参照してください。

### 2.1.1 サービスの定義

サービスを定義する場合、使用するコンテナイメージを指定するか、サービスを構築する元になるソースファイルを用意する必要があります。

サービスコンテナをイメージから作成するには、`image`ステートメントを使用します。

```
services:
  db:
    image: database
```

Podmanは、`compose.yml`ファイルで宣言されたイメージ名がローカルコンテナストレージにあるかどうかをチェックします。ない場合、Podmanは、設定されたレジストリの1つからイメージを取得します。

サービスをソースファイルから構築するには、ソースファイルを用意してコンテナファイルを作成し、その両方を同じディレクトリに配置します。その後、`compose.yml`ファイルで次のように`build`ステートメントを使用します。

```
services:
  db:
    build: PATH_TO_SOURCE_FILES
```

特定のサービスを別のサービスの後に開始する必要がある場合は、`depends_on`ステートメントを使用します。

```
services:
  db:
    image: database
    depends_on:
      system:
        condition: SERVICE_CONDITION
```

`SERVICE_CONDITION`は、`service_started`、`service_healthy`、または`service_completed_successfully`のいずれかになります。

`services`の定義に関する詳細については、[services specification \(https://docs.docker.com/compose/compose-file/05-services/\)](https://docs.docker.com/compose/compose-file/05-services/) を参照してください。

## 2.1.2 ネットワークの定義

デフォルトでは、`docker-compose`はデフォルトのネットワークを作成し、アプリケーションスタック内の各コンテナはそのネットワークに含まれます。デフォルトのネットワークは`docker-compose`によって自動的に作成されるため、`compose.yml`ファイルで宣言する必要はありません。

カスタムネットワークを定義し、特定のサービスをそのネットワークに割り当てることもできます。たとえば、2つのネットワーク`network1`と`network2`を作成するには、次のスニペットを追加します。

```
networks:
  network1:
    # Use a custom driver
    driver: custom-driver-1
  network2:
    # Use a custom driver and name the network
    driver: custom-driver-2
    name: custom_network
```

既存のネットワークを使用することもできます。この場合、ネットワークを`external`としてマークします。

```
networks:
  network1:
    name: network1
    external: true
```

`networks`の仕様の全詳細については、[networks specification \(https://docs.docker.com/compose/compose-file/06-networks/\)](https://docs.docker.com/compose/compose-file/06-networks/) を参照してください。

### 2.1.3 compose.ymlの例

次のcompose.ymlの例では、Prometheus監視システムとGrafana分析システムを使用するアプリケーションスタックを定義します。

```
services:
  prometheus:
    image: prom/prometheus
    container_name: prometheus
    command:
      - '--config.file=/etc/prometheus/prometheus.yml'
    ports:
      - 9090:9090
    restart: unless-stopped
    volumes:
      - ./prometheus:/etc/prometheus
      - prom_data:/prometheus
  grafana:
    image: grafana/grafana
    container_name: grafana
    ports:
      - 3000:3000
    restart: unless-stopped
    environment:
      - GF_SECURITY_ADMIN_USER=admin
      - GF_SECURITY_ADMIN_PASSWORD=grafana
    volumes:
      - ./grafana:/etc/grafana/provisioning/datasources
volumes:
  prom_data:
```

この例のプロジェクト構造は次のようになります。

```
.
├── compose.yml
├── grafana
│   └── datasource.yml
├── prometheus
│   └── prometheus.yml
└── README.md
```

## 2.2 マルチコンテナアプリケーションの展開

適切なディレクトリ構造とcompose.ymlファイルを作成したら、マルチコンテナアプリケーションを展開できます。

1. 実行しようとしているコンテナがまだ存在しないことを確認します。

```
> podman ps --all
```

必要に応じて、特定のコンテナを削除します。

```
> podman rm -fCONTAINER_ID
```

2. `compose.yml`が存在するディレクトリから次のコマンドを実行して、マルチコンテナアプリケーションを起動します。

```
> docker compose up -d
```

`docker-compose`により、そのマルチコンテナアプリケーション用に別のネットワークが作成されます。

3. 実行中のコンテナを一覧にすると、コンテナが実行中でポートがマップされていることを確認できます。

```
> podman ps
```

## 3 マルチコンテナアプリケーションの管理

マルチコンテナアプリケーションを作成した後に、`docker-compose`コマンドを使用して管理操作を実行できます。コマンドの構文は次のとおりです。

```
> docker compose[OPTIONS]COMMAND
```

このコマンドは、管理するマルチコンテナアプリケーションの`compose.yml`ファイルが存在するディレクトリと同じディレクトリから実行します。または、`-f`, `--file`オプションを使用して`compose.yml`ファイルへのパスを指定します。たとえば、マルチコンテナアプリケーションを終了して削除するには、次のコマンドを実行します。

```
> docker compose -f ./test/compose.yml down
```

その他の役立つコマンド:

### images

マルチコンテナアプリケーション内のコンテナが使用するすべてのイメージを一覧にします。

```
> docker compose images
```

## pause

すべてのコンテナを一時停止します。

```
> docker compose pause[SERVICE]
```

## ps

マルチコンテナアプリケーション内のコンテナを一覧にします。

```
> docker compose ps
```

## rm

停止したコンテナを削除します。

```
> docker compose rm
```

## start/stop

コンテナを開始または停止します。

```
> docker compose stop[SERVICE]
```

オプションとコマンドの完全なリストについては、次のコマンドを実行します。

```
> docker-compose --help
```

## 4 法的事項

Copyright © 2006–2025 SUSE LLC and contributors. All rights reserved.

この文書は、GNUフリー文書ライセンスのバージョン1.2または(オプションとして)バージョン1.3の条項に従って、複製、頒布、および/または改変が許可されています。ただし、この著作権表示およびライセンスは変更せずに記載すること。ライセンスバージョン1.2のコピーは、「GNUフリー文書ライセンス」セクションに含まれています。

SUSEの商標については、<https://www.suse.com/company/legal/>を参照してください。その他の第三者のすべての商標は、各社の所有に帰属します。商標記号(®、™など)は、SUSEおよび関連会社の商標を示します。アスタリスク(\*)は、第三者の商標を示します。

本書のすべての情報は、細心の注意を払って編集されています。しかし、このことは正確性を完全に保証するものではありません。SUSE LLC、その関係者、著者、翻訳者のいずれも誤りまたはその結果に対して一切責任を負いかねます。

# A GNUフリー文書利用許諾契約書(GFDL)

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. この使用許諾書を一字一句そのままの複製および頒布することは許可されますが、変更は許可されません。

## 0. 序文

この利用許諾契約書の目的は、マニュアル、テキストブック、またはその他の機能的で有用な文書を、自由という意味で「フリー」にすることです。つまり、そのような文書を、変更の有無や商用非商用に関わらず、コピーまたは再配布する実効的な自由をすべての人々に保証することです。第二に、本利用許諾契約書は、作者または発行者が他者によって行われた変更について責任を負わないとともに、その著作物の功績が確保されるように意図されています。

本利用許諾契約書は、「コピーレフト」(著作物を自由に複製および改変できるようにすること)の一種であり、文書の派生著作物は、それ自体が同じ意味においてフリーでなければなりません。フリーソフトウェア向けに考慮されたコピーレフト利用許諾であるGNU一般公衆利用許諾契約書(GPL)を補足するものです。

弊社は、この利用許諾契約書をフリーソフトウェアのマニュアルに使用するために設計しました。それは、フリーソフトウェアにはフリーマニュアルが必要であるためです。つまり、フリープログラムには、そのソフトウェアと同じ自由を提供するマニュアルが付属しなければなりません。ただし、本利用許諾契約書は、ソフトウェアマニュアルに制限されるものではありません。主題であるか否か、または印刷された本として発行されるか否かに関わらず、任意のテキスト著作物に使用することができます。本利用許諾契約書は、その目的が指示または参照に置かれている著作物に主に使用することを推奨します。

## 1. 適用範囲と定義

本利用許諾契約書は、この利用許諾の条項に従って頒布できることを定めた著作権者の通告が記載されている任意のメディアにおけるマニュアルまたは他の著作物に適用されます。そのような通告は、その著作物をここに記載されている条件に従って使用するための世界的な無償の利用許諾を無期限で付与します。次に示す「文書」は、そのような任意のマニュアルまたは著作物を指します。その公衆ユーザはいずれも被許諾者であり、「利用者」と呼ばれます。利用者は、著作権法に従った許可が必要になるような方法で著作物を複製、変更または頒布する場合に、利用許諾を受け入れます。

文書の「変更された版」とは、そのまま複製されるか、変更または別の言語に翻訳された(またはその両方)文書あるいはその一部を含んだ著作物のことです。

「二次セクション」は、文書の発行者または作者と文書の全体的な主題(または関連事項)との関係のみを示す文書の名前付き付録または前付け部分です。総体的な主題に直接関わる内容は含まれていません。(したがって、文書が部分的に数学のテキストブックになっている場合、二次セクションでは数学について説明されない場合があります)。関係には、主題または関連事項との歴史的なつながり、あるいはそれらに関する法的、商的、哲学的、倫理的、政治的位置付けが含まれる場合があります。

「不変セクション」は、文書が本利用許諾契約書の条件の下でリリースされる旨を述べている通告において、そのタイトルが不変セクションのものとして指定されている、ある特定の二次セクションです。セクションが、すでに説明した二次セクションの定義に一致しない場合は、不変として指定することはできません。文書には、不変セクションが含まれない場合があります。文書で不変セクションを特定しない場合、不変セクションは含まれません。

「カバーテキスト」とは、文書が本利用許諾契約書の条件の下でリリースされる旨を述べている通告において、表カバーテキストまたは裏カバーテキストとして列挙されている、ある一定の短い文章のことです。表カバーテキストは、最大で5語、裏カバーテキストは、最大で25語によって構成できます。

文書の「透過的な複製」とは、その仕様が一般の利用者にとって入手可能で、一般的なテキストエディタまたは一般的な描画プログラム(画素で構成される画像用)、あるいは広く使用されている図面エディタ(図面用)で文書を直接改訂するのに適した形式で表される機械可読の複製のことです。テキストフォーマッタへの入力またはテキストフォーマッタへの入力に適したさまざまな形式への変換に適していることも前提になります。読者による以後の変更を阻止または妨げるようにマークアップまたはマークアップのない状態が調整されている、他の点では透過的なファイル形式で行われた複製は、透過的な複製ではありません。イメージ形式は、相当量のテキストに使用されている場合、透過的ではありません。「透過的」ではない複製は、「不透明」と呼ばれます。

透過的な複製に適した形式として、マークアップのないプレーンなASCII、Texinfo入力形式、LaTeX入力形式、一般に取得可能なDTDを使用するSGMLまたはXML、標準に準拠したHTML、人為的変更用のPostScriptまたはPDFがあります。透過的なイメージ形式には、PNG、XCF、JPGが含まれます。不透明な形式には、独自のワードプロセッサのみで読み取りおよび編集を行える独自の形式、DTDまたは処理(またはその両方)ツールを一般に取得できないSGMLまたはXML、機械生成HTML、出力のみを目的として一部のワードプロセッサによって作成されるPostScriptまたはPDFが含まれます。

「タイトルページ」とは、印刷された本の場合、タイトルページ自体、および本利用許諾契約書でタイトルページに表示することが要求される材料を読みやすいように保持するために必要な以降のページのことを指します。そのようなタイトルページがない形式の著作物の場合、「タイトルページ」は、本文の開始部分に先行する、著作物のタイトルを最も顕著に表している部分の近くにあるテキストのことを指します。

「XYZという表題の付いた」セクションとは、そのタイトルが正確にXYZになっているか、またはXYZを別の言語に翻訳しているテキストに続いてカッコ付きのXYZが含まれている文書の名前付きサブユニットのことです。(ここで、XYZは、次に示すように、「謝辞」、「献辞」、「推薦」、「履歴」などの特定のセクション名を表します)。文書を変更するときに、そのようなセクションの「タイトルを保存する」とは、この定義に従って「XYZという表題の付いた」セクションが残されることを表します。

文書では、本利用許諾契約書が文書に適用される旨を述べている通告の付近に保証の放棄を含めることができます。保証の放棄条項は、本利用許諾契約書内の参照によって、保証の放棄に関してのみ組み込まれると見なされます。つまり、これらの保証の放棄条項がもつ可能性のある他のいかなる含意も無効であり、本利用許諾契約書の意味にまったく影響を与えません。

## 2. そのままの複製

利用者は、商用か否かを問わず、任意のメディアにおいて文書を複製または頒布することができます。その際に、本利用許諾契約書、著作権表示、および本利用許諾契約書が文書に適用される旨を述べる利用許諾通告をすべての複製で再生し、本利用許諾契約書の条件に他のいかなる条件も追加しないことが前提条件になります。利用者は、技術的手段によって、作成または頒布する複製の読み込みまたはさらなる複製を妨げたり、制御したりすることはできません。ただし、複製と引き換えに対価を受け取ることができます。十分に大量の複製を頒布する場合は、セクション3の条件に従う必要もあります。

すでに述べた同じ条件に従って複製を貸与したり、複製を公開したりすることもできます。

## 3. 大量の複製

発行する文書の印刷した複製(または、通常、印刷したカバーをもつメディアに含まれた複製)が100部を超え、文書の利用許諾通告でカバーテキストを必要とする場合は、すべてのカバーテキスト(表カバーの表カバーテキスト、裏カバーの裏カバーテキスト)を明瞭かつ読みやすく記載したカバーに文書の複製を同封する必要があります。また、両方のカバーでは、これらの複製の発行者として、利用者を読みやすい状態で明確に識別しなければなりません。表カバーには、フルタイトルを記述し、タイトルのすべての語が同等に目立つようにする必要があります。カバーには他のマテリアルを追加することもできます。カバーに限って変更を行った場合の複製は、文書のタイトルが保持されていて、これらの条件を満たしている限り、他の点に関してそのままの複製と見なすことができます。

いずれかのカバーで、必要なテキストが多すぎて、読みやすい状態に収まらない場合は、列挙されている最初の部分(問題なく収まる分)を実際のカバーに記載し、残りの部分を隣接ページに入れます。

文書の不透明な複製を100部以上公開または頒布する場合は、それぞれの不透明な複製とともに機械可読の透過的な複製を含めるか、それぞれの不透明な複製内あるいはその複製とともに、ネットワークの一般利用者が標準的な一般ネットワークプロトコルを使用して、追加マテリアルのない文書の完全な透過的複製をダウンロードするときにアクセスできるコンピュータネットワークの場所を明記する必要があります。後者のオプションを使用する場合は、不透明な複製の大量頒布を開始するときに十分慎重な手順を取り、この透過的な複製が、その版の不透明な複製を最後に一般頒布した後(直接またはエージェントや小売業者を通じて)少なくとも1年間、指定した場所で継続的にアクセス可能となるように配慮する必要があります。

大量の複製を再頒布する時点よりもかなり前に、文書の作者に連絡して、文書の更新版を提供する機会を与えることが要求されますが、必須ではありません。

#### 4. 変更

文書の変更された版を、すでに述べた第2項および第3項の条件に従って複製および頒布することができます。その際は、本利用許諾契約書に確実に従って、変更された版をリリースし、変更された版が文書の役割を担うようにして、その複製を所要する任意の利用者に変更された版の頒布および変更の利用許諾を与えることが前提になります。また、変更された版で次のことを行う必要があります。

- A. タイトルページ(カバーがある場合はカバー上も含める)で、文書、および以前の版の文書(以前の版がある場合は、その旨、文書の履歴セクションに列挙する)と識別されるタイトルを使用します。前の版と同じタイトルは、その版の元の発行者が許可を与えた場合に、使用することができます。
- B. タイトルページ上に、この要件から解放されない限り、変更された版において変更の著者としての責任を担う1人以上の人またはエンティティとともに、文書の筆頭著者を少なくとも5人、作者として列挙します(5人に満たない場合は、その筆頭著者のすべて)。
- C. タイトルページ上に、変更された版の発行者の名前を、発行者として記載します。
- D. 文書のすべての著作権表示を保持します。
- E. 変更に関する適切な著作権表示を、他の著作権表示の隣に追加します。
- F. 著作権表示の直後に、本利用許諾契約書の条項に従って変更された版を利用するための許可を一般利用者にする利用許諾通告を、次の補遺に示す形式で含めます。
- G. その利用許諾通告に、不変セクションの全リスト、および文書の利用許諾通告で指定されている必須カバーテキストを保持します。
- H. 本利用許諾契約書の変更されていない複製を含めます。

- I. 「履歴」という表題のセクションを保持して、そのタイトルを保持し、タイトルページに記載されているとおりに、変更された版のタイトル、年度、新しい作者、発行者を少なくとも示す項目を追加します。文書に履歴というセクションがない場合は、そのタイトルページに記載されているとおりに文書のタイトル、年度、作者、発行者を示すセクションを作成し、前の文章に記載されているとおりに変更された版を示す項目を追加します。
- J. 文書の透過的な複製に一般利用者がアクセスできるように文書で指定されている場合は、そのネットワークの場所、およびその文書の基盤となった前の版に対応して文書で指定されているネットワークの場所を保持します。これらは、「履歴」セクションに配置することができます。文書自体よりも4年以上前に発行された著作物の場合、または参照されているその版の元の発行者が許可を与えている場合は、そのネットワークの場所を省略することができます。
- K. 「謝辞」または「献辞」という表題のセクションの場合は、そのセクションのタイトルを保持し、セクション内に、それぞれの貢献者謝辞またはその中の献辞(またはその両方)のすべての内容と意味合いを保持します。
- L. 文書のすべての不変セクションを保持し、そのテキストおよびタイトルを未変更のままにします。セクション番号またはそれと同等の要素は、セクションタイトルの一部と見なされません。
- M. 「推薦」という表題の任意のセクションを削除します。そのようなセクションは、変更された版に含めることはできません。
- N. 既存のセクションのタイトルを変更して、「推薦」という表題にしたり、タイトルが不変セクションと矛盾したりしないようにします。
- O. 保証の放棄を保持します。

変更された版に、二次セクションと見なされ、文書から複製された材料を含まない新しい前付けセクションまたは付録が含まれる場合は、これらの一部またはすべてを任意に「不変」として指定することができます。これを行うには、変更された版の利用許諾表示内で列挙されている不変セクションにそのタイトルを追加します。これらのタイトルは、他のすべてのセクションタイトルと異なっている必要があります。

「推薦」という表題のセクションを追加することができますが、その際は、変更された版のさまざまな当事者による推薦以外の要素が含まれていないことが前提になります。たとえば、校正者によるコメント、または文が標準的な信頼できる定義として組織によって承認されていることを示すという宣言文などが相当します。

表カバーテキストとしての最大5語の短い文、および裏カバーテキストとしての最大25語の短い文を、変更された版のカバーテキストのリストの終わりに追加できます。表カバーテキストの短い1文および裏カバーテキストの短い1文のみを、1つのエンティティが追加できます(また

はエンティティによって行われた調整を通じて)。文書に、利用者または利用者が関わる同じエンティティによって行われた調整を通じて前に追加された同じカバーのカバーテキストがすでに含まれている場合は、別のものを追加することはできませんが、古いものを置き換えることができます。ただし、その古いものを追加した前の発行者から明示的な許可を得る必要があります。

文書の作者および発行者は、本利用許諾契約書により、その名前を得るために使用したり、変更された版の推薦を主張または暗示したりする許可を与えるものではありません。

## 5. 文書の結合

文書は、すでに述べた変更された版に関するセクション4の条件に従って、本利用許諾契約書の下でリリースされた他の文書と結合することができます。その際は、その組み合わせの中に、元の全文書のすべての不変セクションを未変更のまま含めて、そのすべてを結合された著作物の不変セクションとしてその利用許諾表示に列挙し、そのすべての保証の放棄を保持することが前提となります。

結合された著作物には、本利用許諾契約書の複製を1つのみ含める必要があります。複数の同一の不変セクションは、単一の複製で置き換えることができます。同じ名前だが内容の異なる複数の不変セクションがある場合は、そのような各セクションのタイトルを固有なものにします。その際は、その終わりに、カッコ付きで、そのセクションの元の作者または発行者の名前(既知の場合)、あるいは固有の番号を追加します。不変セクションのリスト内のセクションタイトルには、結合された著作物の利用許諾表示の場合と同じ調整を加えます。

組み合わせでは、さまざまな元の文書の「履歴」という表題のセクションを結合して、1つの「履歴」というセクションを構築する必要があります。同じように、「謝辞」という表題のセクション、および「献辞」という表題のセクションも結合します。「推薦」という表題のすべてのセクションを削除する必要があります。

## 6. 文書のコレクション

文書および本利用許諾契約書の下でリリースされた他の文書から成るコレクションを作成して、さまざまな文書に含まれる本利用許諾契約書の個々の複製を、コレクションに含まれる単一の複製で置き換えることができますが、他のすべての点での各文書のそのままの複製に関する本利用許諾契約書の規則に従うことが前提になります。

そのようなコレクションから単一の文書を抽出して、その文書を本利用許諾契約書に従って個々に頒布することができますが、その際は、本利用許諾契約書の複製を抽出した文書に挿入して、その文書のそのままの複製に関するその他のすべての点で、本利用許諾契約書に従う必要があります。

## 7. 独立した著作物の集積

文書またはその派生物を他の個別および独立した文書または著作物とともに、ストレージまたは頒布メディア内またはそのボリューム上に蓄積することを「集積」と呼びます。その場合は、個々の著作物の許可を超えてその蓄積の利用者の法的権限を制限することに、蓄積による著作権を使用しないことが前提になります。文書が集積に含まれる場合、本利用許諾契約書は、それ自体が文書の派生著作物ではない集積内の他の著作物に適用されません。

セクション3のカバーテキスト要件が文書のこれらの複製に適用可能であり、文書が集積全体の半分に満たない場合は、文書のカバーテキストを、集積内の文書のカバー、または文書が電子形式の場合は、電子的な同等のカバーに配置することができます。それ以外の場合は、集積全体の印刷されたカバー上に表示する必要があります。

## 8. 翻訳

翻訳は一種の変更と見なされるため、セクション4の条件に従って文書の翻訳を配布することができます。不変セクションを翻訳で置き換えるには、著作権者からの特別な許可が必要ですが、これらの不変セクションの元のバージョンのほかに、一部またはすべての不変セクションの翻訳を含めることができます。本利用許諾契約書、文書内のすべての利用許諾表示、および保証の放棄の翻訳を含めることができますが、その際は、本利用許諾契約書の元の英語版、およびそれらの利用許諾表示と保証の放棄の元の版も含めることが前提になります。本利用許諾契約書、利用許諾表示、または保証の放棄の翻訳と元の版との間に不一致がある場合は、元の版が優先されます。

文書内のセクションに、「謝辞」、「献辞」、または「履歴」という表題が付いている場合は、そのタイトルを保持する(セクション1)ための要件(セクション4)により、通常は実際のタイトルを変更する必要があります。

## 9. 終了

利用者は、本利用許諾契約書に明示的に記載されている形態を除き、文書を複製、改変、二次利用許諾、および頒布してはなりません。別の方法で文書を複製、改変、二次利用許諾、または頒布しようとするのは無効であり、本利用許諾契約書の下で利用者の権利は自動的に消滅します。ただし、本使用許諾の下で利用者から複製物または権利を受領した関係者は、条項を遵守している限り、権利が消滅することはありません。

## 10. 本利用許諾契約書の将来の改訂

フリーソフトウェア財団は、GNUフリー文書利用許諾契約書(GFDL)の新版または改訂版を随時公表することがあります。そのような新版は、性格的には現行版と似たものになりますが、新たな問題や懸案事項に対応するために細部が異なる可能性があります。 <https://www.gnu.org/copyleft/> を参照してください。

本利用許諾契約書の各版には、区別するための版番号が設定されます。文書に、それに適用される本利用許諾契約書の特定の版番号と「後継版」が指定されている場合、利用者は、選択によって、その指定された版の条項またはフリーソフトウェア財団から公開される後継版の条項(ドラフトではない)に従うことになります。文書に、本利用許諾契約書の版番号が指定されていない場合、利用者は、フリーソフトウェア財団からこれまでに公開された任意の版(ドラフトではない)を選択することができます。

### 補遺: 本利用許諾書をご使用の文書に使用する方法

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".
```

不変セクション、表カバーテキスト、および裏カバーテキストがある場合は、「with...Texts」の行を次のように置き換えます。

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

カバーテキストのない不変セクションが含まれている場合、またはこの3つの他の組み合わせの場合は、その2つの代替要素をマージして状況に合わせます。

文書にプログラムコードの重要な例が含まれている場合は、GNU一般公衆利用許諾契約書(GPL)などの選択したフリーソフトウェアの利用許諾に従って、これらの例を平行してリリースし、フリーソフトウェアでのその利用を許可することを推奨します。