Perforce Helixを用いた
分散バージョン管理環境（DVCS）の構築手順

2018年5月

株式会社東陽テクニカ
ソフトウェア・ソリューション

本書について

- 本書では、Perforce Helixを用いた
  分散バージョン管理のための環境を構築し、
  利用するための手順をご説明します
- 本書は、読まれる方にHelixの利用経験があることを
  前提としています
- 本書では、Helixサーバ 2017.2、GUIクライアント
  2017.3を使用することを前提とします

※ DVCS：Distributed Version Control System

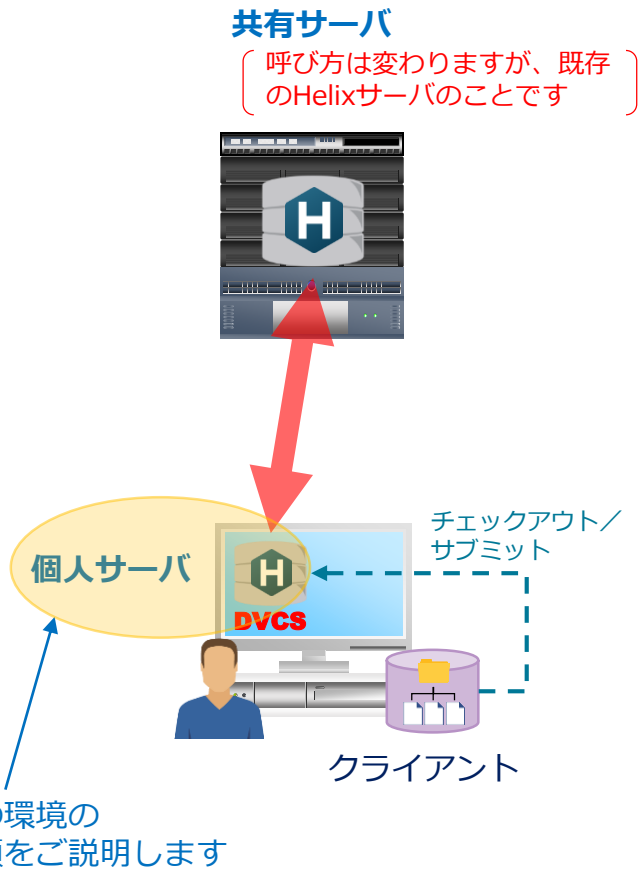
目次

- 概要
- 個人サーバの構築／利用手順
- 個人サーバ構築の応用例

概要

Helix DVCSとは

- Helix DVCSは、Gitと同様の開発環境を実現する機能です
- 既存のHelixサーバに保持しているファイルの一部、あるいはすべてをサブセットとして切り出し、個人用のサーバでバージョン管理します
  - ファイルの切り出し元にしたHelixサーバを、**共有サーバ**と呼びます
  - 新たに構築した個人用のHelixサーバを、**個人サーバ**と呼びます
- 個人サーバに接続したクライアントで、チェックアウトやサブミットを実行できます
- 共有サーバ上で更新された情報を個人サーバに取り込んだり、個人サーバで行った変更を共有サーバに登録したりできます



Helix DVCSがもたらすメリット

- 共有サーバから必要な情報（ファイル）を切り出し、個人サーバ上でバージョン管理します

  これにより、

  - 各ユーザは、試行錯誤の過程で作成されるファイルをローカルマシン内でバージョン管理しながら開発できます
  - 共有したい変更結果や成果物だけを共有サーバに戻すことにより、共有サーバは整理された状態を維持できます
  - 共有サーバが地理的に離れた拠点に存在する場合、共有サーバに接続することなく開発作業を進められます
  - すべてのユーザが共有サーバに常時接続する必要はないため、共有サーバの負荷を低減できます
  - 必要なファイルだけを保持することにより、個人サーバのディスク容量を最小限に抑えられます

典型的なユースケース

Helix DVCS環境は、次のようなケースにおいて有効です

- 1つの修正タスクに対して、個人的には細かい変更ごとに差分を登録しておきたいが、Helixサーバ（共有サーバ）上には細かい中間作業の記録を残したくない
- Helixサーバ（共有サーバ）は地理的に離れた拠点に存在し、クライアントの拠点からアクセスするには低速のWANを経由する必要があり、パフォーマンス上の問題を抱えている
- 数百名のユーザがHelixサーバ（共有サーバ）に常時アクセスすることにより、Helixサーバの負荷が定常的に高くなり、パフォーマンスの低下が見られる
- Gitを利用しているが、Gitリポジトリでは全ファイルを保持したクローンしか作成できず、必要以上にローカルリポジトリのディスク領域を使用することに困っている

典型的なユースケース

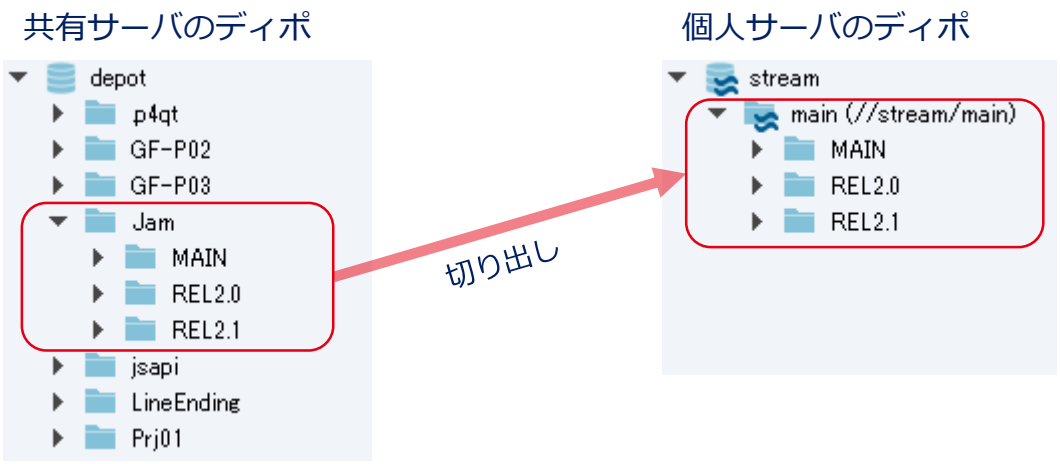
Helix DVCS環境は、次のようなケースにおいて有効です

- 1つの修正タスクに対して、個人的には細かい変更ごとに差分を登録しておきたいが、Helixサーバ（共有サーバ）上には細かい中間作業の記録を残したくない
- Helixサーバ（共有サーバ）は地理的に離れた拠点に存在し、クライアントの拠点からアクセスするには低速のWANを経由する必要があり、パフォーマンス上の問題を抱えている
- 数百名のユーザがHelixサーバ（共有サーバ）に同時アクセスすることにより、Helixサーバの負荷が定常的に高くなり、パフォーマンスの低下が見られる
- Gitを利用しているが、Gitリポジトリでは全ファイルを保持したクローンしか作成できず、必要以上にローカルリポジトリのディスク領域を使用することに困っている

本書では、このユースケースに基づいて個人サーバを構築し、利用する際の手順をご説明します

ファイルのサブセットを切り出し

- 個人サーバでバージョン管理したいファイルセットを、既存のHelixサーバから切り出します
- 複数箇所を選択することも可能です
- 切り出したファイルセットは、個人サーバ上のストリームとして構築されます

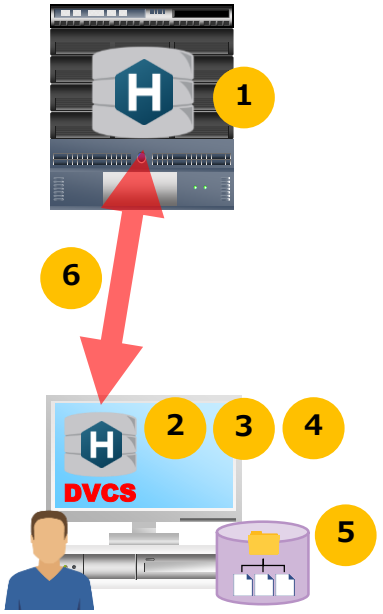


個人サーバの構築／利用手順

構築手順の流れ

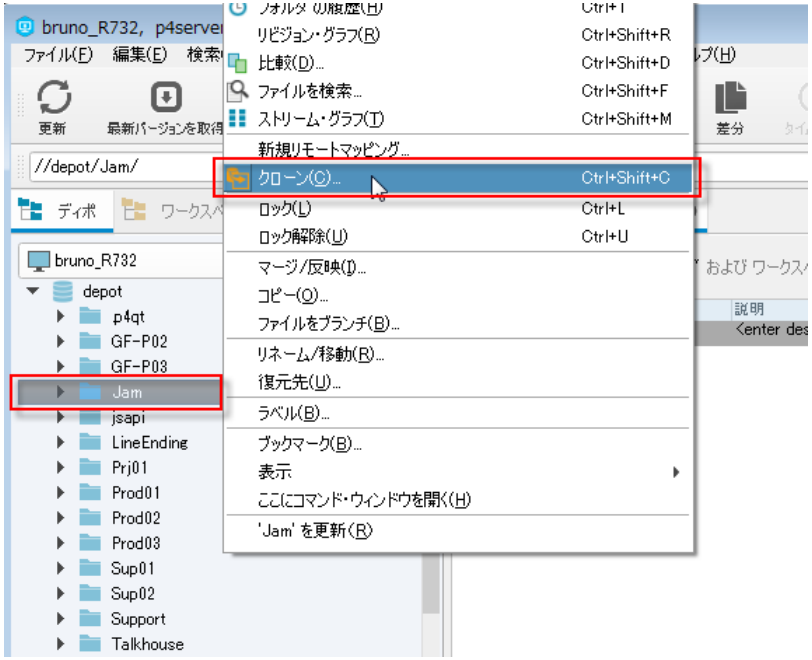
個人サーバを構築する手順をご説明します

① 切り出すファイルセットを選択
② 個人サーバの構成を決定
③ 個人サーバの配置場所を決定
④ 個人サーバを構築
⑤ 個人サーバへの接続を確認
⑥ 個人サーバ構築後の利用手順

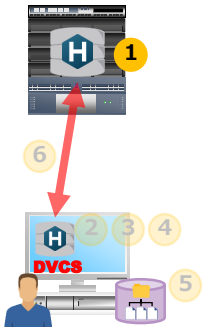


① 切り出すファイルセットを選択

- 共有サーバに接続して、操作します
- 共有サーバから切り出して、個人サーバ上に保持したいディレクトリを選択して右クリックし、[クローン]を選択します

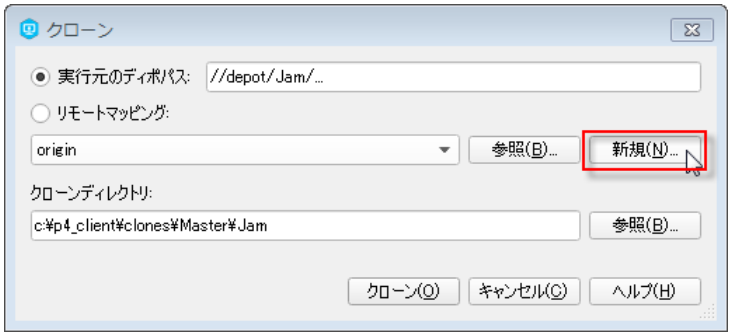


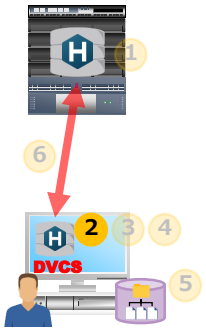
共有サーバ上の、//depot/Jam/... を切り出す例



② 個人サーバの構成を決定 (1/2)

- [クローン]ダイアログで、リモートマッピングの[新規]ボタンをクリックします
- 個人サーバの構成を決定するための、[リモートマッピング]フォームが表示されます





② 個人サーバの構成を決定 (2/2)

表示されたリモートマッピングにおいて、
次のフィールドで構成を決定します

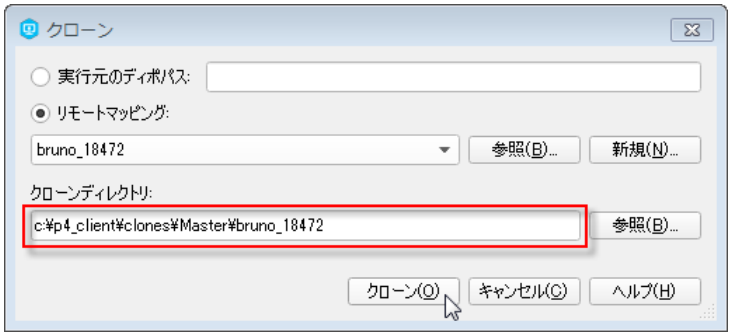
- ディポマッピング：
  - 共有サーバのパス（右）を記述します

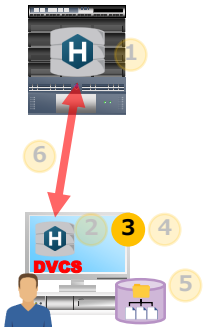
    //depot/Jam/...
  - 個人サーバのパス（左）を記述します

    //stream/dev/...

③ 個人サーバの配置場所を指定

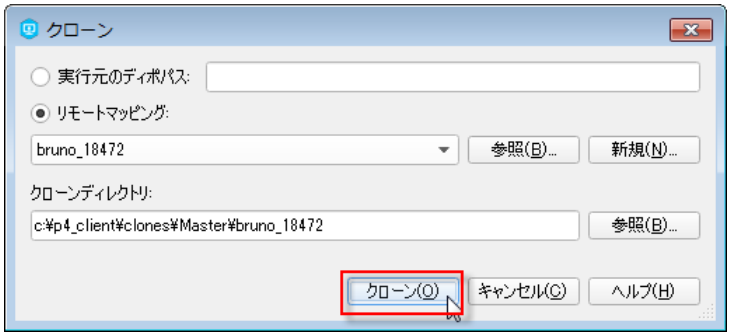
- [クローン]ダイアログに戻り、[クローンディレクトリ]フィールドに配置場所を入力します
- 指定したクローンディレクトリに、個人サーバは次の情報を保持します
  - 個人サーバのディポとデータベース
  - ワークスペースのルート
- パフォーマンスの観点から、ローカルのディスク領域を指定することをお勧めします

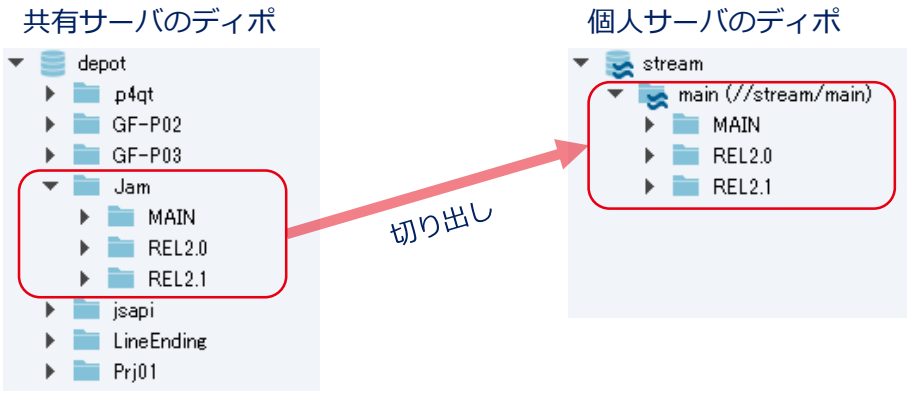




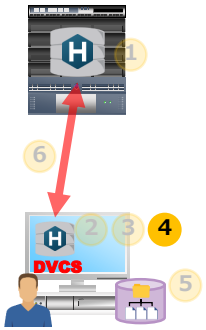
④ 個人サーバを構築

- [クローン]ボタンをクリックして、個人サーバを構築します



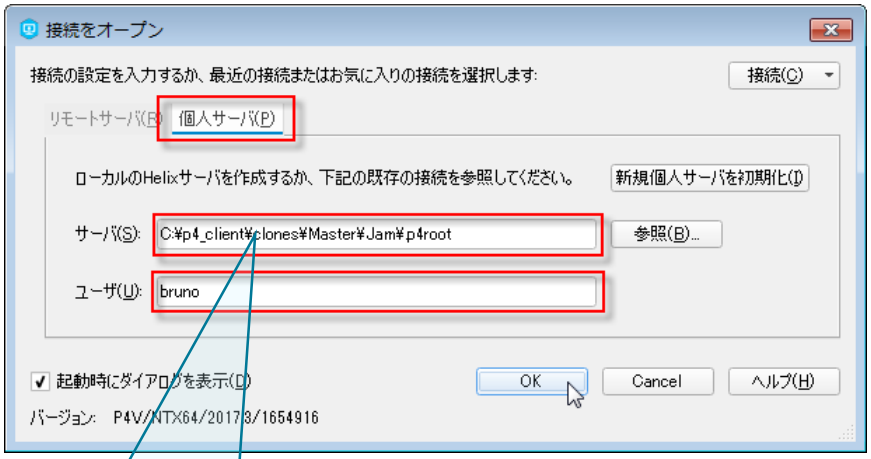


構築した個人サーバの //stream/dev/... には、
共有サーバの //depot/Jam/... が切り出され、保持されます

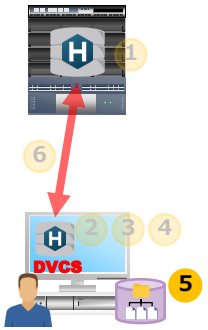


⑤ 個人サーバへの接続を確認

- P4Vを起動します
- P4Vの[接続をオープン]ダイアログにおいて、[個人サーバ]タブを開きます
- [サーバ]フィールドにクローンディレクトリを指定します
- [ユーザ]フィールドに自分のユーザ名を指定します
- [OK]ボタンをクリックして、P4Vのメイン画面が立ち上がることを確認します

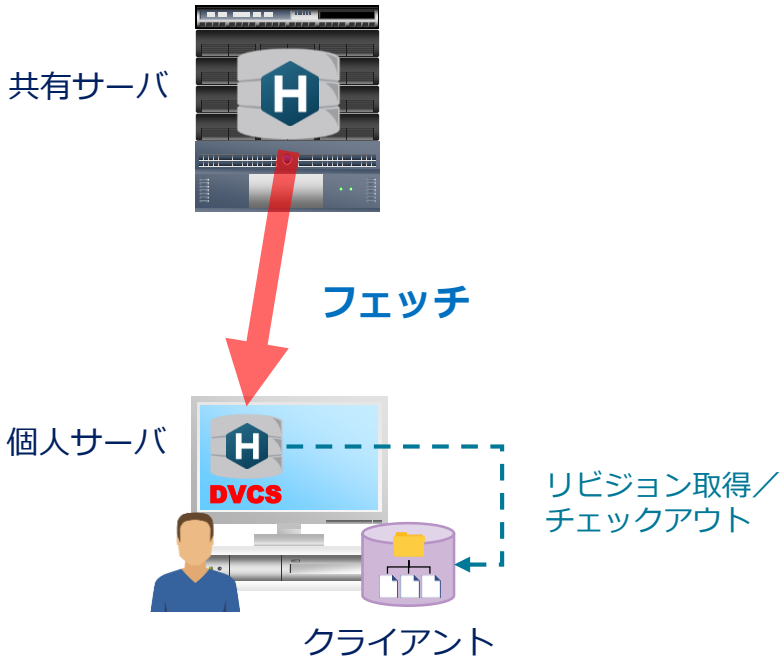


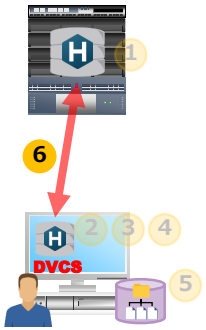
ご注意:
通常のHelixサーバに接続するときのような、
<サーバ名>:<ポート番号>
という形式ではありません



⑥ 個人サーバ構築後の利用手順 (1)

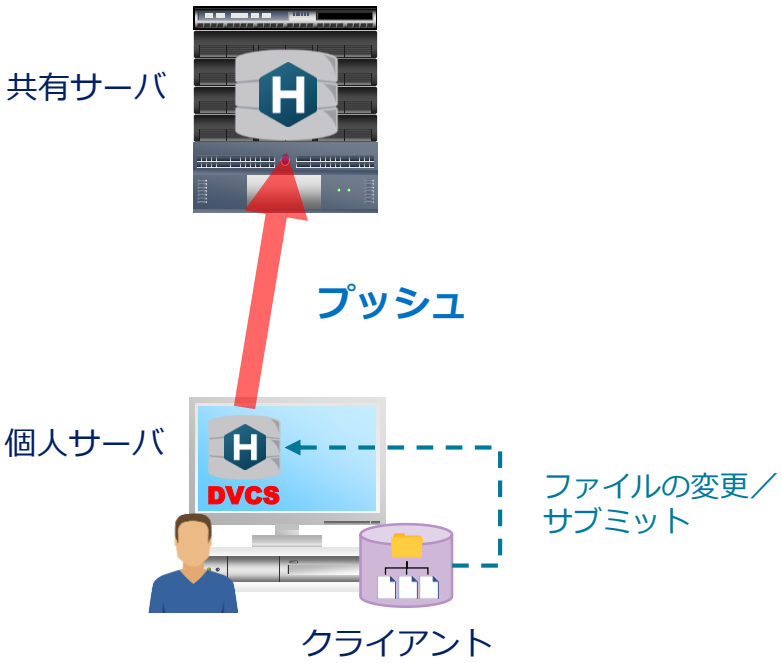
- 他のユーザが変更した共有サーバ上のファイルを、個人サーバに取り込みます（この操作をフェッチと呼びます）
- 取り込んだファイルを、個人サーバ上で変更（チェックアウト／サブミット）します

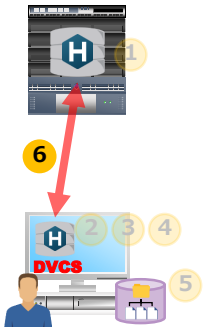




⑥ 個人サーバ構築後の利用手順 (2)

- 個人サーバ上で変更したファイルを、共有サーバに登録します（この操作をプッシュと呼びます）
- 共有サーバに登録した情報は、他のユーザから参照できるようになります





個人サーバ構築の応用例

応用例(1)：特定のディレクトリを除外

- 個人サーバ上に保持するファイルセットから、特定のディレクトリやファイルを除外することができます

- ディポマッピングにおける除外マッピング
  - 個人サーバのパスの行頭にマイナス記号「－」を付けます
  - ワークスペースビューのように、指定したディレクトリやファイルを除外できます

- 記述例：

//stream/dev/...	//depot/Jam/...
-//stream/dev/test/...	//depot/Jam/test/...
-//stream/dev/....iso	//depot/Jam/....iso

意味：
//depot/Jam/... 配下のファイルを保持
そのうち、
//depot/Jam/test/... については、保持の対象から除外
すべての .iso ファイルについても、保持の対象から除外

応用例(2)：保持するリビジョン数を制限

- 個人サーバ上に保持するファイルにおいて、保持するリビジョン数を制限することができます

- アーカイブ制限でリビジョン数を制限
  - パスとリビジョン数を指定します
  - ファイルタイプ修飾子 +S のように、保持するリビジョン数を制限できます

- 記述例：

```
//stream/dev/bin/...          10
//stream/dev/....zip          1
```

意味：
//depot/Jam/bin/... については、最新の10リビジョンのみを保持
すべての .zip ファイルについては、最新リビジョンのみを保持

お問い合わせ先

より複雑なHelix DVCS環境の構築や、応用操作の具体的な手順については、
マニュアル「Using Helix Server for Distributed Versioning」
（https://www.toyo.co.jp/files/user/img/product/ss/help/perforce/r17.2/manuals/dvcs/dvcs.pdf）
をご参照ください（近日中に、日本語版を公開予定）

その他、ご不明な点があればご遠慮なくお問い合わせください

株式会社東陽テクニカ
ソフトウェア・ソリューション
テクニカルサポート: ss_support@toyo.co.jp

東陽テクニカ　PERFORCE