Open Source Conference 2024 Online / Spring

NTT DATA
Trusted Global Innovator

Apache Bigtop 3.3 (仮)

2024/03/01

岩崎 正剛
日本Hadoopユーザ会／株式会社NTTデータグループ

© 2024 NTT DATA Group Corporation

NTT DATA

Apache Bigtopの概要

© 2024 NTT DATA Group Corporation

Apache Bigtopとは何か

- 公式サイト (https://bigtop.apache.org/) より:
  - "Bigtop is an Apache Foundation project for Infrastructure Engineers and Data Scientists looking for comprehensive packaging, testing, and configuration of the leading open source big data components."
  - 『Bigtopは、先進的なビッグデータ関連OSSのパッケージングやテスト、設定を求めているインフラエンジニアやデータサイエンティストのためのApacheソフトウェア財団のプロジェクトです。』

- 一言で言うと「Hadoop・Spark を中心とした大規模データ処理基盤を容易に構築するための OSS」です。

© 2024 NTT DATA Group Corporation

データ基盤構築のためにBigtopが提供する機能

1. 要件を満たすソフトウェアの選定と、それらの間の相互運用性の確認
   →相互運用性を確認済みの幅広いビッグデータ関連ソフトウェアを、debやrpmとしてパッケージングするための資材および、ビルド済みパッケージを提供

2. ソフトウェアパッケージをサーバ群にインストールし、各製品を適切に設定
   →各ソフトウェアのデプロイを自動化するための Puppet マニフェストを提供

3. 設定を施したサーバ群が、クラスタとして正しく動作することを確認
   →各ソフトウェアの主要な機能の動作確認のための、スモークテストを提供

4. ローカル環境で、デプロイ資材の開発、各製品の試用を実現
   →Dockerコンテナ上でPuppetマニフェストの適用、スモークテストの実行や試用をするためのProvisionerを提供

© 2024 NTT DATA Group Corporation

Apache Bigtop の歴史

Components	1.0.0	1.1.0	1.2.1	1.3.0	1.4.0	1.5.0
Ambari			2.5.0	2.6.1	2.6.1	2.6.1
Flink			1.1.3	1.4.2	1.6.4	1.6.4
Hadoop	2.6.0	2.7.1	2.7.3	2.8.4	2.8.5	2.10.1
HBase	0.98.12	0.98.12	1.1.9	1.3.2	1.3.2	1.5.0
Hive	1.0.0	1.2.1	1.2.1	2.3.3	2.3.3	2.3.6
Kafka	0.8.1.1	0.8.1.1	0.10.1.1	0.10.2.2	0.10.2.2	2.4.0
Spark	1.3.1	1.5.1	2.1.1	2.2.1	2.2.3	2.4.5
Zeppelin		0.5.6	0.7.2	0.7.3	0.7.3	0.8.2
Zookeeper	3.4.6	3.4.6	3.4.6	3.4.6	3.4.6	3.4.13

© 2024 NTT DATA Group Corporation

NTT DATA

Apache Bigtop の歴史

Components	3.0.0	3.0.1	3.1.0	3.1.1	3.2.0	3.2.1	3.3.0
Ambari	2.7.5	2.7.5	2.7.5	2.7.5	2.7.5	2.8.0	
Flink	1.11.3	1.11.6	1.11.6	1.11.6	1.15.0	1.15.3	1.16.2
Hadoop	3.2.2	3.2.2	3.2.3	3.2.4	3.3.4	3.3.5	3.3.6
HBase	2.2.6	2.2.6	2.4.8	2.4.8	2.4.13	2.4.13	2.4.17
Hive	3.1.2	3.1.2	3.1.2	3.1.2	3.1.3	3.1.3	3.1.3
Kafka	2.4.1	2.4.1	2.8.1	2.8.1	2.8.1	2.8.1	2.8.2
Spark	3.0.1	3.0.1	3.1.2	3.1.2	3.2.1	3.2.1	3.3.4
Zeppelin	0.9.0	0.10.0	0.10.0	0.10.0	0.10.1	0.10.1	0.11.0
Zookeeper	3.4.14	3.4.14	3.5.9	3.5.9	3.5.9	3.5.9	3.7.2

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtop 3.2.1のソフトウェアスタック

コンポーネント	バージョン	説明
Alluxio	2.8.0	ストレージ仮想化
Ambari	2.7.5	クラスタ構築・管理
Flink	1.15.3	ストリーム処理系
GPDB	5.28.5	MPP
Hadoop	3.3.5	分散ファイルシステム・ 並列分散処理基盤
HBase	2.4.13	分散KVS
Hive	3.1.3	Hadoop用クエリ処理系
Kafka	2.8.1	分散メッセージブローカ
Livy	0.7.1	Spark用RESTゲートウェイ

コンポーネント	バージョン	説明
Oozie	5.2.1	ジョブスケジューラ
Phoenix	5.1.2	HBase用クエリ処理系
Solr	8.11.2	全文検索エンジン
Spark	3.2.1	並列分散処理エンジン・ ストリーム処理系
Tez	0.10.1	並列分散処理エンジン
YCSB	0.17.0	ベンチマークツール
Zeppelin	0.10.1	ノートブック
Zookeeper	3.5.9	分散コーディネーション

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtop 3.3のソフトウェアスタック(仮)

コンポーネント	バージョン	説明
Alluxio	2.**9.3**	ストレージ仮想化
~~Ambari~~	~~2.7.5~~	~~クラスタ構築・管理~~
Flink	1.**16.2**	ストリーム処理系
GPDB	**6.23.1**	MPP
Hadoop	3.3.**6**	分散ファイルシステム・ 並列分散処理基盤
HBase	2.4.**17**	分散KVS
Hive	3.1.3	Hadoop用クエリ処理系
Kafka	2.8.**2**	分散メッセージブローカ
Livy	0.**8.0**	Spark用RESTゲートウェイ

コンポーネント	バージョン	説明
~~Oozie~~	~~5.2.1~~	~~ジョブスケジューラ~~
Phoenix	5.1.**3**	HBase用クエリ処理系
Ranger	**2.4.0**	**認可**
Solr	**9.1.1**	全文検索エンジン
Spark	3.**3.4**	並列分散処理エンジン・ ストリーム処理系
Tez	0.10.**2**	並列分散処理エンジン
~~YCSB~~	~~0.17.0~~	~~ベンチマークツール~~
Zeppelin	0.**11.0**	ノートブック
Zookeeper	3.**7.2**	分散コーディネーション

※3.3 で削除されるコンポーネント

Ambari, Oozie, YCSB

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtop 3.3 のソフトウェアスタックにおける主な変更点

- Ambari削除
  - AmbariとBigtopとの間の依存関係が、循環するような状況を解消するため。
  - AmbariのBigtopスタック(mpack)定義はAmbari側に移動。
  - Ambariそれ自体のパッケージングはAmbari側で提供。
- Oozie削除
  - バージョン5.1.2(2021年)を最後にリリースされず、更新もほとんどないため
  - 現在アクティブに開発/利用されている代替プロダクトはApache Airflow?
- YCSB削除
  - 3年以上にわたり更新/リリースがないため
- Ambari、Oozie、YCSBは他のミドルウェアと連携する都合上、ビルド時間が長く、パッケージのバイナリが巨大になりがちだったため、これらの削除でBigtopの開発がすこし身軽になった。

- Rangerの追加 (予定)
  - HDFS, YARN, Hive, HBase, Kafka などに粒度の細かい認可ルールを設定・一元管理

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtop 3.3のサポートするOSディストリビューション

- 2024/3時点ではまだ未確定
- openEulerが追加(BIGTOP-3875)
  - Huawei社が開発したディストリビューションをオープンソース化。
  - Red Hat Enterprise Linuxベース
- 維持コストの削減のため同系列で複数LTSをサポートしない方向性
  - CentOS 7は(Ambariとセット)で利用ユーザが多いため継続中

Linux distro	Bigtop 3.2.1	Bigtop 3.3.0
CentOS	7	7
Rocky Linux	8	8 or **9**
Fedora	35, 36	**38**
openEuler		**22.03**
Debian	10, 11	11
Ubuntu	18.04, 20.04, 22.04	22.04

© 2024 NTT DATA Group Corporation

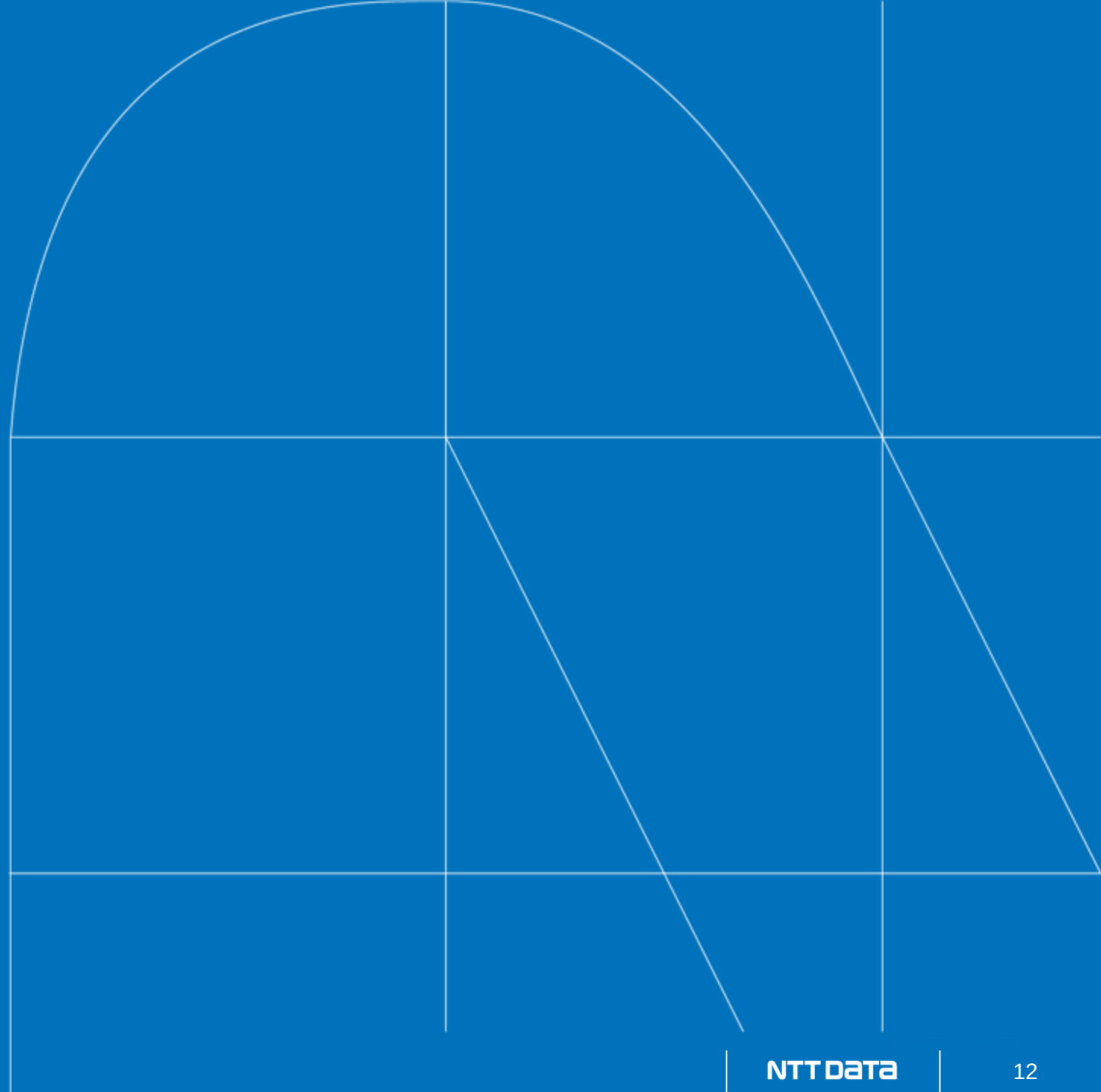
Bigtop 3.3のサポートするCPUアーキテクチャ

- x86_64とaarch64はAWS等を利用して開発、テストできる。
  - 中国系(企業)のcontributorはaarch64利用が多い印象。
- ppc64le環境でのビルドはベストエフォート的。
  - CI環境も乏しい(1ノード)ため、対応を落としがち。
- ネイティブコードを含むJavaライブラリで、x86_64だと、aarch64やppc64leの個別対処が必要になる。

CPU アーキテクチャ	Bigtop 3.2.1	Bigtop 3.3.0
x86_64	○	○
aarch64	○	○
ppc64le	○(Rocky8を除く)	○(Rocky8、???を除く)

© 2024 NTT DATA Group Corporation

プラットフォームと互換性



© 2024 NTT DATA Group Corporation

なぜLinuxなのか

- BigtopがサポートするのはLinuxの主要ディストリビューション
  - 主要プロダクトの多くがJavaベースではあるものの

- Hadoopのサーバプラットフォームとしては Linuxが前提
  - 主要部分はJava: 多くのプラットフォームで動く
  - CLIはbash (>=3.0): 多くのプラットフォームで動く
    - Windows用のhadoop.cmdも存在するがもうメンテされていない
  - Cで実装されたネイティブコード(libhadoop.so)はLinux依存
    - オプショナルな機能や性能向上を提供: 特にHDFSのサーバ側で重要
    - Linuxのシステムコール前提

- (Hadoopの)開発者はUbuntu on x86_64が多数派という印象
- HadoopのCI環境もUbuntu on x86_64
- それ以外のプラットフォーム(RHEL系、aarch、ppc64le)の問題が見落とされがち

© 2024 NTT DATA Group Corporation

nativeライブラリのポータビリティ

- Hadoopプロジェクト配布版binary tarballにもlibhadoop.soは入っているが、どのLinux環境でも動くとは限らない
  # binary tarballは特定の環境でビルドしたものをoptionalに提供
  # 例えばHadoop 3.2.2の場合はUbuntu 16.04でビルドしたもの

glibcのバージョンミスマッチでnativeライブラリが使えない状態:

```
$ hadoop checknative
Native library checking:
hadoop:  false
zlib:    false
...

$ ldd libhadoop.so
./libhadoop.so: /lib64/libc.so.6: version `GLIBC_2.14' not found (required by ./libhadoop.so)
         linux-vdso.so.1 =>  (0x00007ffc0959f000)
        ...
```

© 2024 NTT DATA Group Corporation

NTT DATA

Javaのバージョン

- パッケージング対象プロダクトの大半はJava/JVM言語で実装
- Bigtopは各種Linuxディストリビューション提供のOpenJDK **8**を(ビルドに)利用
- 主にHadoop開発環境のJava 11移行が難航しているため...
  - JerseyのupgradeがblockerHADOOP-15984)
    - 影響を受けるコードが非常に多い
    - Jersey 1とJersey 2(以降)がJAX-RSの異なるバージョンに基づき、両方を共存させてサブモジュールごとに段階的にアップデートできない。

- Hadoop 3.3(以降)は、実行環境としてはJava 11をサポート
- 他のJavaプロダクトもJava 11上で実行可能(なはず)

© 2024 NTT DATA Group Corporation

JavaのAPI

- 公開APIと内部APIの分離が不完全
  - サブモジュール間で参照するためにpublicにされたクラス/メソッドが多い
  - 下流のプロダクトがテストのために内部APIを利用できてしまう
  - 各プロダクトの開発者は、下流のプロダクトを意識しきれない
- プロダクトXの内部APIを変更すると、依存するプロダクトYのビルドが壊れたりしがち

- 公開APIを示すためのアノテーションを利用する例もあるが、強制力はない
  - @InterfaceAudience.Publicなものだけjavadocが出力される
  - @InterfaceAudience.Privateだとマイナーバージョンアップでも変更されうる

InterfaceAudienceアノテーションの例:

```
@InterfaceAudience.LimitedPrivate({ "MapReduce", "HBase" })
@InterfaceStability.Unstable
public class DistributedFileSystem extends FileSystem
    implements KeyProviderTokenIssuer, BatchListingOperations {
```

© 2024 NTT DATA Group Corporation

Javaライブラリの依存関係(とdependency hell)

- 同一パッケージ名、同一クラス名のライブラリを複数バージョン混在させることは不可能
- ダメなパターンの例:
  - プロダクトXがプロダクトYに依存
  - プロダクトXがhoge-1.0に依存し、hoge-1.5に無いAPIを利用
  - プロダクトYがhoge-1.5に依存し、hoge-1.0に無いAPIを利用
- 汎用的なライブラリは特にプロダクト間で競合しがち
  - SLF4J, Log4j
  - commons-logging, commons-cli, commons-httpclient
  - Jackson
  - Guava
  - Netty, Jetty, Jersey
  - protobuf-java
  - ZooKeeper, Curator
- 脆弱性対応のために依存ライブラリをアップグレードすると、非互換な変更が入ることも

© 2024 NTT DATA Group Corporation

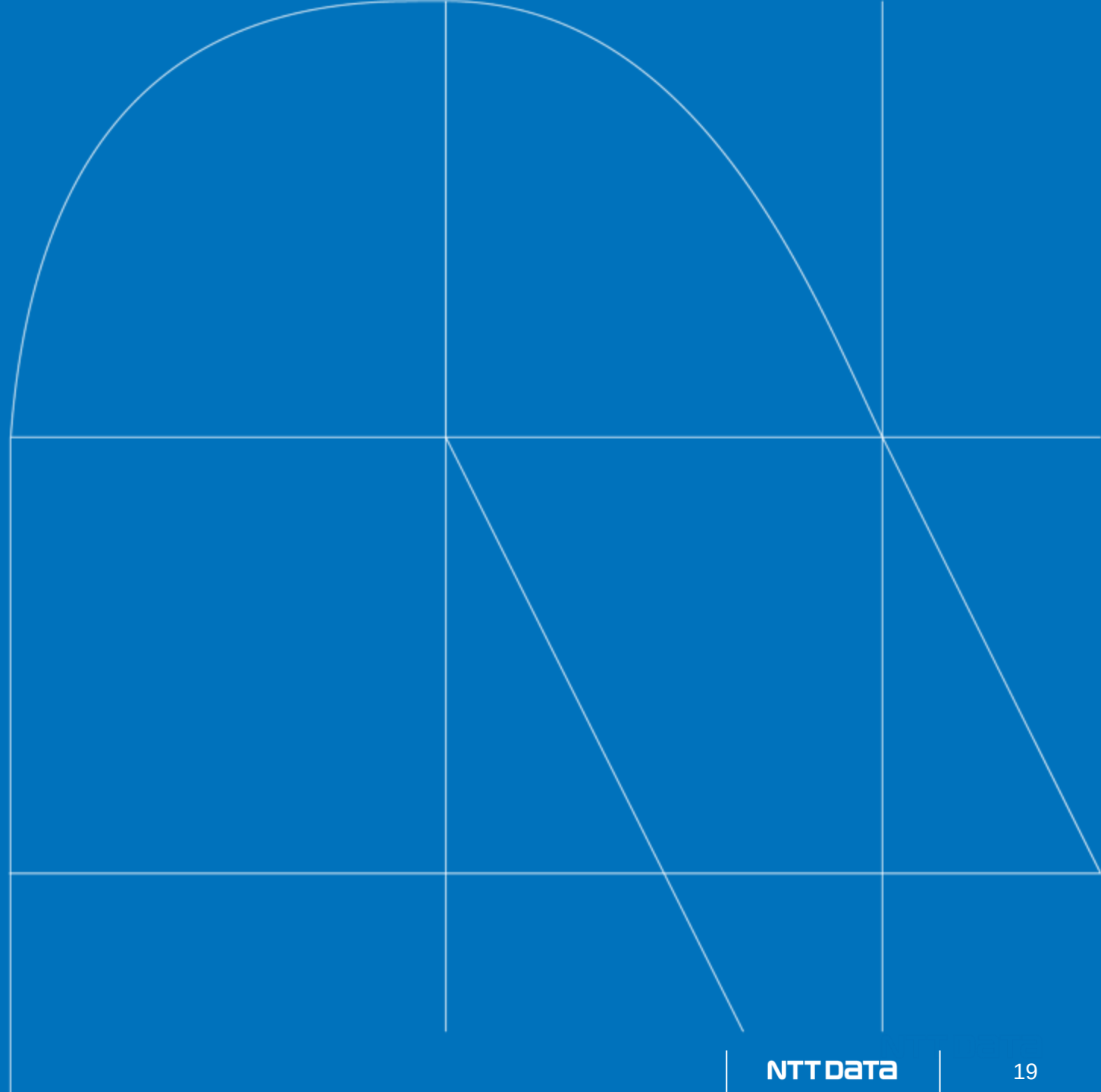
Bigtopの役割

- ちゃんと機能するプロダクトのバージョンの組み合わせは自明ではない
  - Bigtopは「機能するバージョンの組み合わせ」を選定
  - 場合によってはパッチをあてて辻褄を合わせる

- CPUアーキテクチャ、OSディストリビューションごとの調整も必要
  - Bigtopはそれぞれの環境でビルドしてパッケージングしてテストする仕組みを提供

© 2024 NTT DATA Group Corporation

Bigtopの基本素材



© 2024 NTT DATA Group Corporation

Bigtopのソースコード

- ソースコードはGitHubにある
  - # 2019年4月からASF(Apache Software Foundation)全体として
    # https://blogs.apache.org/foundation/entry/the-apache-software-foundation-expands
- 開発の管理はJIRA
  - https://issues.apache.org/jira/projects/BIGTOP
  - issueを作ってパッチを添付するかpull requestのリンクを貼る

Bigtopのソースコードの取得:

```
$ git clone https://github.com/apache/bigtop
$ cd bigtop
```

© 2024 NTT DATA Group Corporation

Bigtopのタスク

- Gradleを利用して各種処理を実行する

タスク一覧の表示:

```
$ ./gradlew tasks

> Task :tasks

------------------------------------------------------------
Tasks runnable from root project - Bigtop
------------------------------------------------------------

Apache Creadur tasks
--------------------
rat - Runs Apache Rat checks

Build tasks
-----------
assemble - Assembles the outputs of this project.
...
```

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtop BOM

- 部品表: 対象プロダクトとそのバージョンなどが記述されている
- GroovyのConfigSlurperの書式

```
$ cat bigtop.bom
  bigtop { // *the name should be change: the parsing code depends on it*
    version = "STACK-VERSION" // *required*
    stack { // *required* Fundamental properties of the Stack: JDK, SDK, GDK, etc
      'jdk' { version = '1.8'; version_base = version }
      'scala' { version = '2.10.4'; version_base = version }
    }
    ...
  components {
    'hadoop' {
      name    = 'hadoop'
      relNotes = 'Apache Hadoop'
      version { base = '3.2.2'; pkg = base; release = 1 }
      tarball { destination = "${name}-${version.base}.tar.gz"
                source      = "${name}-${version.base}-src.tar.gz" }
      url     { download_path = "/$name/common/$name-${version.base}"
                site = "${apache.APACHE_MIRROR}/${download_path}"
                archive = "${apache.APACHE_ARCHIVE}/${download_path}" }
    }
```

© 2024 NTT DATA Group Corporation

bigtop_toolchain

© 2024 NTT DATA Group Corporation

bigtop_toolchain

- プロダクト群をビルドするための環境をセットアップする資材
  - Puppetをインストールするスクリプト
  - ビルドツール群をインストールPuppetマニフェスト
    - JDK, Maven, GCC, CMake, Protocol Buffers, lib*-devel, ...
- OS提供パッケージだとバージョンが合わないものは個別にセットアップ
- プラットフォーム差分に対応するためのワークアラウンドも提供
  - 例) aarch64対応のバックポートパッチを当ててprotobuf-2.5.0をビルド

bigtop_toolchainによるビルド環境のセットアップ:

```
$ sudo bigtop_toolchain/bin/puppetize.sh
$ ./gradlew toolchain
```

© 2024 NTT DATA Group Corporation

NTT DATA

Protocol Buffers

- 最近のHadoopやHBaseはprotobuf-maven-pluginを利用
  - protocのバイナリをビルド時に自動的にダウンロードして利用
- バイナリが未提供のプラットフォーム/バージョンではうまくいかない
  - com.google.protobuf:protoc < 3.5.0 ではaarch64用バイナリが未提供
  - com.google.protobuf:protoc < 3.7.0 ではppcle64バイナリが未提供
- bigtop-toochainはワークアラウンドとしてProtocol Buffersにパッチを当ててビルド
  - ビルド時にmvn install:install-fileでローカルアーティファクトをインストールして利用

- (FYI)Hadoop 3.3.0でprotobuf-2.5.0からprotobuf-3.7.1に移行
  - だがしかしTezはまだprotobuf-2.5.0を利用

do-component-build内でローカルビルドしたprotocからアーティファクトをインストール:

```
if [ $HOSTTYPE = "aarch64" ] ; then
  mvn install:install-file -DgroupId=com.google.protobuf -DartifactId=protoc -Dversion=2.5.0 ¥
      -Dclassifier=linux-aarch_64 -Dpackaging=exe -Dfile=/usr/local/bin/protoc
fi
```

© 2024 NTT DATA Group Corporation

bigtop-packages

© 2024 NTT DATA Group Corporation

bigtop-packages

- プロダクトをビルドしてパッケージングする
- 「プロダクト名-pkg」というGradleタスクで実行
- .rpmと.debの2種類に対応
  - ビルド環境に応じたパッケージが作られる
    - Red Hat系なら.rpm
    - Debian系なら.deb

パッケージングタスクの実行例:

```
$ ./gradlew hadoop-pkg
$ ./gradlew zookeeper-pkg
$ ./gradlew bigtop-jsvc-pkg bigtop-utils-pkg bigtop-groovy-pkg
```

© 2024 NTT DATA Group Corporation

パッケージング資材

- 共通のビルド手順とインストール手順を.debと.rpm用の資材から呼び出して使う
  - ビルド手順: do-component-build
  - インストール手順: install_プロダクト名.sh

パッケージング資材の配置:

```
bigtop-packages
├── src
│   ├── common
│   │   ├── hadoop
│   │   │   ├── do-component-build
│   │   │   ├── install_hadoop.sh
│   │
│   ├── deb
│   │   ├── hadoop
│   │   │   ├── control
│   │   │   ├── rules
│   │
│   ├── rpm
│   │   ├── hadoop
│   │   │   ├── SPECS
│   │   │   │   ├── hadoop.spec
```

© 2024 NTT DATA Group Corporation

NTT DATA

Bigtopパッケージングにおけるパッチ適用

- パッケージング時にpatch.*diffという名前のパッチファイルを適用
- 各プロダクトのリリース版そのままでは解決できない問題に対応
  - 特定プラットフォームでビルドが失敗する問題への対処
  - パッケージング対象バージョンに含まれない修正のバックポート
  - 互換性を壊す変更のrevert
  - プロダクト間のdependenciesのつじつま合わせ
- patchを書いたらなるべく本家にfeedback

Bigtopが適用するpatchの例:

```
$ find bigtop-packages/src/common/hadoop -name 'patch*.diff'  | sort
bigtop-packages/src/common/hadoop/patch0-revert-HADOOP-16598.diff
bigtop-packages/src/common/hadoop/patch1-HADOOP-15939.diff
bigtop-packages/src/common/hadoop/patch2-exclude-spotbugs-annotations.diff
bigtop-packages/src/common/hadoop/patch3-fix-broken-dir-detection.diff
```

© 2024 NTT DATA Group Corporation

パッケージリポジトリの作成

- $BIGTOP_HOME/output 下に出力されるパッケージファイルでリポジトリを作成する
- repoタスクで実行
- その環境に応じたリポジトリが作られる (Red Hat系ならYumの、Debian系ならAptの)

リポジトリの作成(CentOS 8で実行した場合の例):

```
$ ./gradlew repo
$ tree output
...
├── hadoop
│   ├── hadoop-3.2.2-1.el8.src.rpm
│   └── x86_64
│       ├── hadoop-3.2.2-1.el8.x86_64.rpm
│       ├── hadoop-client-3.2.2-1.el8.x86_64.rpm
│       ├── hadoop-conf-pseudo-3.2.2-1.el8.x86_64.rpm
...
│       ├── hadoop-hdfs-3.2.2-1.el8.x86_64.rpm
...
├── repodata
...
│   ├── aa0ff69b812187315d6825fcf7d862c326ac1ba42bf9625e57b826cd6462a68c-filelists.xml.gz
│   ├── cb8d35ada5c22a89e1f65a62229acde45b4175d5281a513fc6e1190d0ce4544c-filelists.sqlite.bz2
│   ├── f2077b7afd203a11428c21ca3c41f02a786fc2f514888274f12f514673d4f88f-primary.xml.gz
│   └── repomd.xml
```

© 2024 NTT DATA Group Corporation

NTT DATA

Dockerコンテナを利用したパッケージング

- 「プロダクト名-pkg-ind」というGradleタスクで実行(in Docker container)
- ひとつの環境で全OSディストリビューションのパッケージをビルドできる
- Docker Hubのbigtop/slavesのイメージを利用
  - -POS=centos-7 -Pprefix=trunkだとtrunk-centos-7タグになる(x86_64の場合)
- CPUアーキテクチャはホスト環境に応じて決まる
  - aarch64のマシン上でやるとtrunk-centos-7-aarch64タグ
- 毎回ライブラリをすべてダウンロードすると遅い(BIGTOP-3801)
  - -Pmvn-cache-volume=trueでローカルリポジトリをvolumeに保存して再利用

- repo-indタスクは、repoタスクをDockerコンテナで実行
  - Docker provisioner(後述)と組み合わせてテストに利用

パッケージングタスクの実行例:

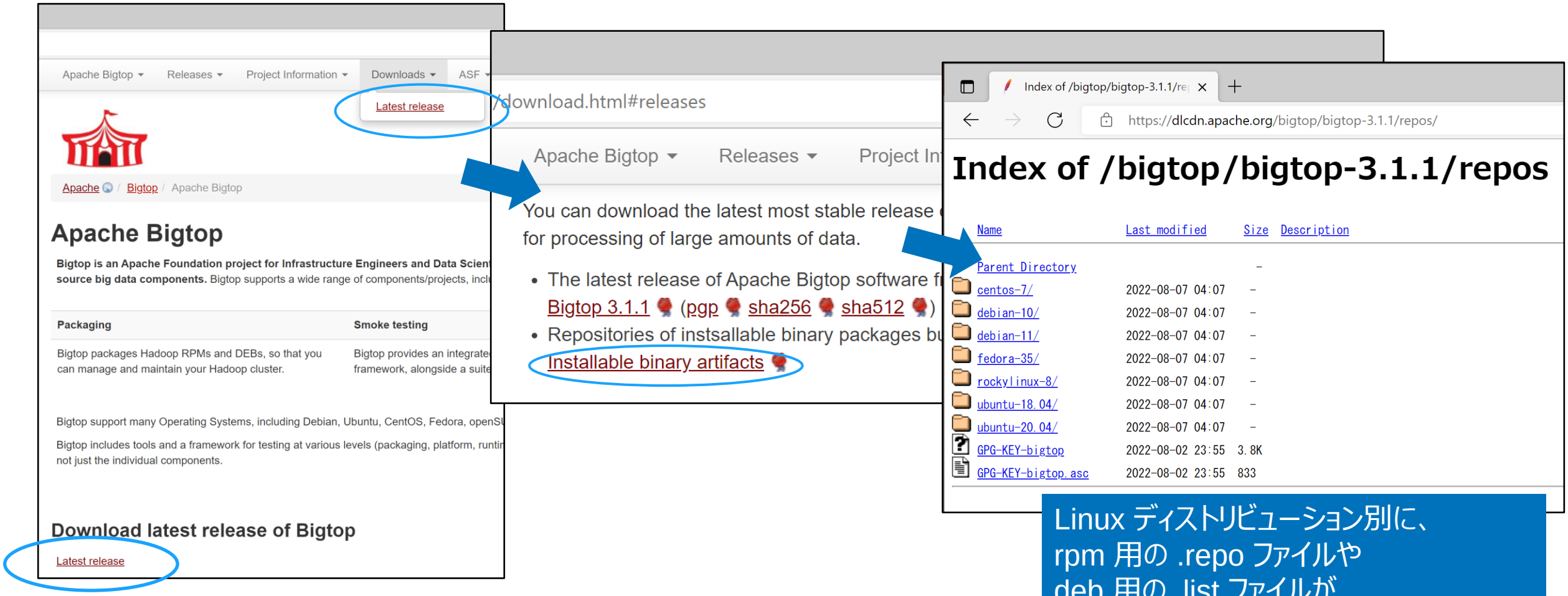
```
$ ./gradlew hadoop-pkg-ind -POS=rockylinux-8 -Pprefix=trunk -Pmvn-cache-volume=true
$ ./gradlew repo-ind -POS=rockylinux-8 -Pprefix=trunk
```

© 2024 NTT DATA Group Corporation

NTT DATA

ビルド済パッケージのリポジトリ定義ファイル

Bigtop の公式サイト 以下のリンクを辿って取得



Linux ディストリビューション別に、
rpm 用の .repo ファイルや
deb 用の .list ファイルが
ダウンロード可能

© 2024 NTT DATA Group Corporation

NTT DATA

ビルド済パッケージのインストール手順(RPM)

- RPMパッケージを用いたコンポーネントのインストールは以下の3ステップ

1. リポジトリ設定ファイルをダウンロード・配置

```
$ sudo wget -P /etc/yum.repos.d -q ¥
  https://dlcdn.apache.org/bigtop/bigtop-3.1.1/repos/centos-7/bigtop.repo
```

2. GPG キーをインポート

```
$ sudo rpm --import ¥
  https://dlcdn.apache.org/bigtop/bigtop-3.1.1/repos/GPG-KEY-bigtop
```

3. リポジトリ情報のアップデート・コンポーネントのインストール

```
$ sudo yum install hadoop

$ hadoop version
Hadoop 3.2.4
```

© 2024 NTT DATA Group Corporation

ビルド済パッケージのインストール手順(DEB)

- debパッケージを用いたコンポーネントのインストール手順も同様

1. リポジトリ設定ファイルをダウンロード・配置

```
$ sudo wget -P /etc/apt/sources.list.d -q ¥
  https://dlcdn.apache.org/bigtop/bigtop-3.1.1/repos/ubuntu-20.04/bigtop.list

$ curl -s https://dlcdn.apache.org/bigtop/bigtop-3.1.1/repos/GPG-KEY-bigtop | ¥
  sudo apt-key add -
```

2. GPG キーをインポート

```
$ sudo apt-get update
```

3. リポジトリアップデート・コンポーネントインストール

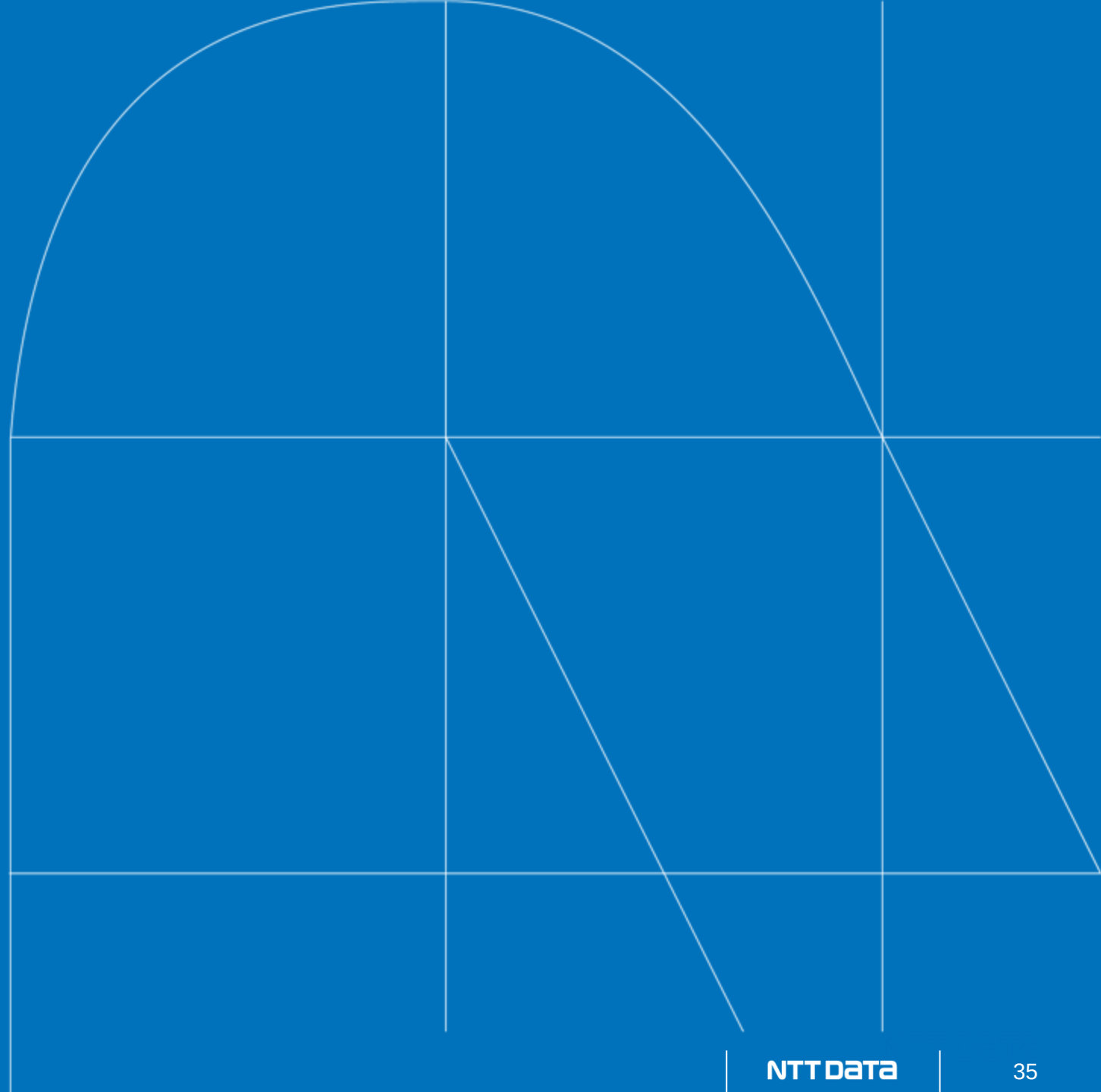
```
$ sudo apt-get install hadoop

$ hadoop version
Hadoop 3.2.4
```

© 2024 NTT DATA Group Corporation

NTT DATA

bigtop-deploy



© 2024 NTT DATA Group Corporation

bigtop-deploy

- パッケージの導入でインストール作業は省力化できたが、以下の問題は解決されていない
  - サーバの台数が多い場合、1台1台インストールのコマンドを投入するのは大変
  - パッケージのインストール後も、各ソフトウェアに適切な設定を施したり、サービスを起動したりといった作業が必要

- Bigtop はこれらの作業を自動化するためにPuppet マニフェストを提供
  - Puppet はRubyで実装された構成管理ツール
  - Puppet manifest はシステムの「あるべき姿」を宣言的に記述したDSL
    - 記述に従って Puppet がパッケージのインストールや設定、サービス起動などを行う
  - Hieraはキーバリュー型の設定管理システム
    - ユーザが環境に応じて変更したいパラメータをmanifestと独立に指定できる

© 2024 NTT DATA Group Corporation

bigtop-deploy/puppet

- HadoopクラスタをPuppetマニフェスト:

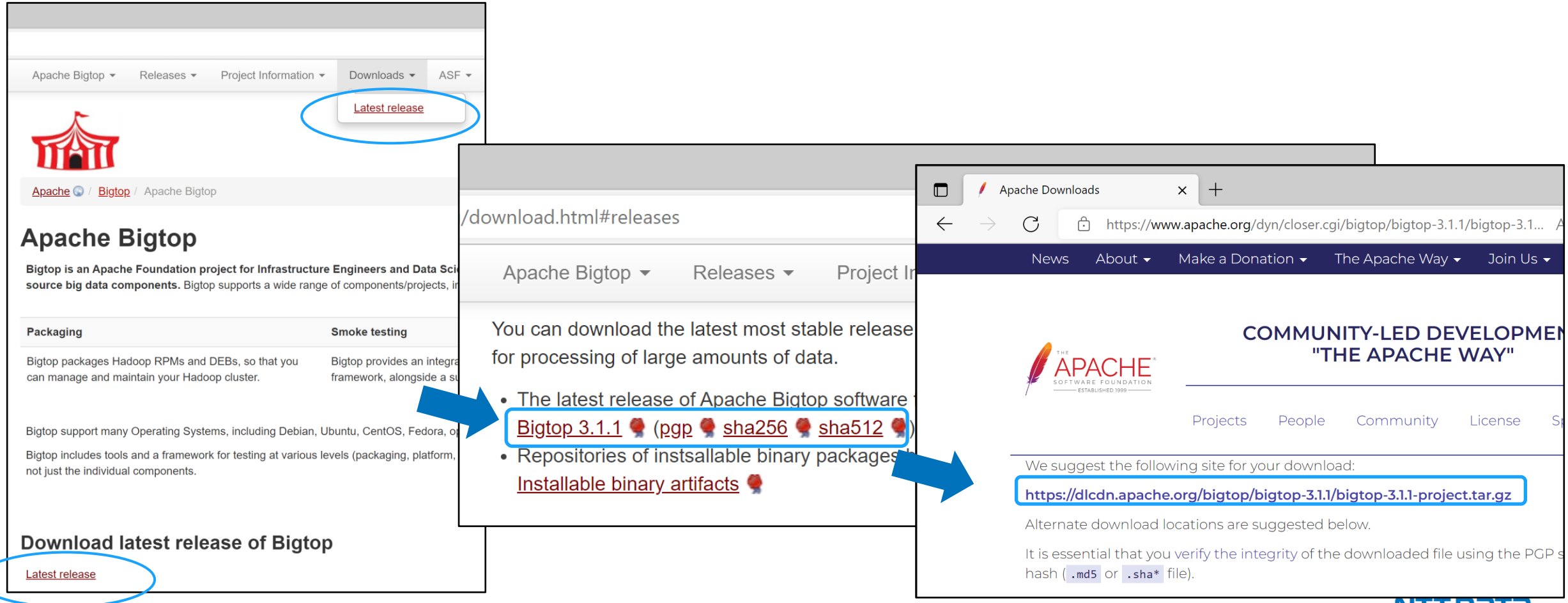
```
$ tree bigtop-deploy/puppet
bigtop-deploy/puppet
├── hieradata
│   ├── bigtop
│   │   ├── cluster.yaml
...
│   └── site.yaml
├── hiera.yaml
├── manifests
│   ├── bigtop_repo.pp
│   ├── cluster.pp
...
│   ├── hadoop
...
│   │   ├── manifests
│   │   │   └── init.pp
│   │   └── templates
│   │       ├── container-executor.cfg
│   │       ├── core-site.xml
...
```

© 2024 NTT DATA Group Corporation

NTT DATA

Puppet マニフェストのダウンロード

- Bigtop のサイトからsource tarballをダウンロード
- またはhttps://github.com/apache/bigtopからソースコードを取得:



© 2024 NTT DATA Group Corporation

Puppet マニフェストを用いたコンポーネントのデプロイ手順

- Puppet マニフェストを用いたコンポーネントのデプロイは、以下の4ステップ (次ページに続く)

```
$ curl -sLO https://dlcdn.apache.org/bigtop/bigtop-3.1.1/bigtop-3.1.1-project.tar.gz
$ tar xf bigtop-3.1.1-project.tar.gz
$ cd bigtop-3.1.1
$ sudo bigtop_toolchain/bin/puppetize.sh
$ vi bigtop-deploy/puppet/hieradata/site.yaml
$ cat bigtop-deploy/puppet/hieradata/site.yaml
---
bigtop::hadoop_head_node: "fqdn.of.head.node"

hadoop::hadoop_storage_dirs:
  - /data

hadoop_cluster_node::cluster_components:
  - hdfs
  - yarn
  - mapreduce
```

1. Bigtop の tarball をダウンロードし展開

2. 中に含まれるスクリプトを実行して Puppet をインストール

3. Hiera ファイルを編集。最低限必要な設定は左記の3つだが、インストールするコンポーネントに応じて必要な設定を記述する。
   - bigtop::hadoop_head_node: Hadoop のマスターノードのFQDN
   - hadoop::hadoop_storage_dirs: Hadoop のデータ格納先ファイルパス
   - hadoop_cluster_node::cluster_components:
     インストールするコンポーネントのリスト。例では HDFS, YARN, MapReduceをインストールしている。

© 2024 NTT DATA Group Corporation

NTT DATA

Puppet マニフェストを用いたコンポーネントのデプロイ手順 (続き)

```
$ sudo cp -r bigtop-deploy/puppet/hiera* /etc/puppet
$ sudo puppet apply --parser future ¥
  --modulepath=bigtop-deploy/puppet/modules:/etc/puppet/modules ¥
  bigtop-deploy/puppet/manifests

(略)


Notice: Finished catalog run in 65.99 seconds
$ echo $?
0
$ sudo jps
7873 ResourceManager
8722 DataNode
8486 NodeManager
8922 Jps
8251 NameNode
7775 WebAppProxyServer
8159 JobHistoryServer
```

4. Hiera ファイル群を所定のパスにコピーし、Puppet を実行
※"--parser future"オプションはCentOS 7にのみ付与
※--modulepathオプションに指定している"/etc/puppet/modules"の部分は
ディストリビューションによって異なる(puppet moduleのインストール先を指定する)

※一連の手順を parallel-ssh (http://code.google.com/p/parallel-ssh/)
などを用いて実行することで、クラスタ内の全サーバでデプロイを行うことが可能

© 2024 NTT DATA Group Corporation

NTT DATA

provisioner

© 2024 NTT DATA Group Corporation

Docker provisioner

- ローカルやステージング環境で確認したいケースは多々ある
  - 業務上・システム上の要件を満たせるか
  - 構築手順が正しいこと
  - ソフトウェアをバージョンアップで問題が生じないか

- Bigtop は コンテナ上でコンポーネントをデプロイできる Provisioner を提供
- 事前にDockerとdocker-composeのセットアップが必要

© 2024 NTT DATA Group Corporation

Docker provisioner の使用方法

```
$ ./gradlew docker-provisioner -Pnum_instances=3 -POS=centos-7 -Pprefix=3.1.1 -Pstack=hdfs,yarn

...

Pulling bigtop (bigtop/puppet:3.1.1-centos-7)...
3.1.1-centos-7: Pulling from bigtop/puppet
Digest: sha256:42f20509d1b9460b753b70f6e21882699de3a488f93af28375c466181bc8727a
Status: Downloaded newer image for bigtop/puppet:3.1.1-centos-7
Creating 20220929_115113_r20838_bigtop_1 ... done
Creating 20220929_115113_r20838_bigtop_2 ... done
Creating 20220929_115113_r20838_bigtop_3 ... done

...

Notice: Roles to deploy: [resourcemanager, nodemanager, mapred-app, hadoop-client, namenode, datanode]
Notice: Roles to deploy: [nodemanager, mapred-app, datanode]
Notice: Roles to deploy: [nodemanager, mapred-app, datanode]

...

Notice: Finished catalog run in 231.67 seconds

BUILD SUCCESSFUL in 4m 39s
```

3台の Docker コンテナ上 (OS は CentOS 7) に Bigtop 3.1.1 の HDFS, YARN をインストールするコマンドを実行

(ローカルに存在しなければ) Bigtop 3.1.1 の CentOS 7 イメージが pull され、そのイメージからコンテナが作成される

指定したコンポーネントの Puppet マニフェストを使って、コンテナ上へのデプロイが実行される

© 2024 NTT DATA Group Corporation

Docker provisioner の使用方法（続き）

プロビジョニングしたコンテナは、通常通り
`docker ps` コマンドで確認できる

```
$ docker ps
CONTAINER ID   IMAGE                              COMMAND        CREATED          STATUS          PORTS     NAMES
6df7279c442e   bigtop/puppet:3.1.1-centos-7   "/sbin/init"   4 minutes ago   Up 4 minutes
20220929_115113_r20838_bigtop_3
a37ba6399273   bigtop/puppet:3.1.1-centos-7   "/sbin/init"   4 minutes ago   Up 4 minutes
20220929_115113_r20838_bigtop_2
850c8ce4785a   bigtop/puppet:3.1.1-centos-7   "/sbin/init"   4 minutes ago   Up 4 minutes
20220929_115113_r20838_bigtop_1

$ docker exec -it 20220929_115113_r20838_bigtop_1 bash
[root@850c8ce4785a /]# jps
2416 Jps
1528 JobHistoryServer
1676 NodeManager
1212 ResourceManager
2156 DataNode
1086 WebAppProxyServer
1951 NameNode
[root@850c8ce4785a /]# exit

$ ./gradlew docker-provision-destroy
...
BUILD SUCCESSFUL in 14s
1 actionable task: 1 executed
```

`docker exec` で任意のコマンドをコンテナ上で実行可能

別のクラスタを起動する前には、既存のクラスタを破棄する必要がある

© 2024 NTT DATA Group Corporation

NTT DATA

Docker provisioner の使用方法 (続き)

```
$ ./gradlew tasks

...

Deployment tasks
----------------
docker-provisioner -
Provision a Bigtop stack cluster on Docker container(s). Default to CentOS and 1 node.
Properties:
  -Pconfig=[CONFIG_FILE] (located under provisioner/docker/)
  -Penable_local_repo
  -Pimage=[DOCKER_IMAGE] (overwrites -POS and -Pprefix)
  -Pmemory=[4g|8g|...]
  -Pnum_instances=[NUM_INSTANCES]
  -Pnexus=[NEXUS_URL] (NEXUS_URL is optional)
  -POS=[centos-7|fedora-35|debian-10|ubuntu-18.04|opensuse-42.3]
  -Pprefix=[trunk|1.2.1|1.2.0|1.1.0|...]
  -Prepository=[REPO_URL]
  -Prun_smoke_tests (run test components defined in config file)
  -Psmoke_tests=[COMPONENTS]
  -Pstack=[COMPONENTS]
```

Docker provisioner はオプションが多いため
`./gradlew tasks` コマンドで一覧を確認

© 2024 NTT DATA Group Corporation

NTT DATA

Docker provisionerの実装

- 中身はDockerコンテナを起動してPuppetでクラスタを構築するスクリプト
- スクリプト(provisioner/docker/docker-hadoop.sh)を直接使ってもよい
- cgroup v2のホストでは`--docker-compose-yml docker-compose-cgroupv2.yml`を指定

Dockerコンテナを利用したデプロイの例:

```
$ cd provisioner/docker
$ ./docker-hadoop.sh ¥
    --create 3 ¥
    --image bigtop/puppet:trunk-ubuntu-22.04 ¥
    --docker-compose-yml docker-compose-cgroupv2.yml ¥
    --docker-compose-plugin ¥
    --memory 8g ¥
    --repo file:///bigtop-home/output/apt ¥
    --disable-gpg-check ¥
    --stack hdfs,yarn,mapreduce,spark

$ ./docker-hadoop.sh --list
             Name                   Command     State   Ports
-------------------------------------------------------------
20210124_122554_r12819_bigtop_1   /sbin/init   Up
20210124_122554_r12819_bigtop_2   /sbin/init   Up
20210124_122554_r12819_bigtop_3   /sbin/init   Up

$ ./docker-hadoop.sh --exec 1 /bin/bash
```

© 2024 NTT DATA Group Corporation

Docker provisionerの仕組み

- docker-hadoop.sh --createで出力されたdocker-compose.ymlを見ると分かる
- Bigtopのソースツリーを/bigtop-homeにマウント
- Hieraの設定を生成してpuppet apply

Docker provisionerの設定ファイル(の一部):

```
$ cat docker-compose.yml
bigtop:
    ...
    volumes:
    - ../../:/bigtop-home
    - ./config/hiera.yaml:/etc/puppet/hiera.yaml
    - ./config/hieradata:/etc/puppet/hieradata
    - ./config/hosts:/etc/hosts
    - /sys/fs/cgroup:/sys/fs/cgroup:ro

$ cat config/hieradata/site.yaml
bigtop::hadoop_head_node: b2d614be6039.bigtop.apache.org
hadoop::hadoop_storage_dirs: [/data/1, /data/2]
bigtop::bigtop_repo_uri: file:///bigtop-home/output
bigtop::bigtop_repo_gpg_check: false
hadoop_cluster_node::cluster_components: [hdfs,yarn,mapreduce]
hadoop_cluster_node::cluster_nodes: [172.17.0.3, 172.17.0.4, 172.17.0.2]
...
```

© 2024 NTT DATA Group Corporation

Dockerクラスタの終了

- スクリプト経由でコンテナ停止と、一時ファイル等の削除
- クラスタ起動中の問題などでやり直す場合も、まず一度--destroyしてから

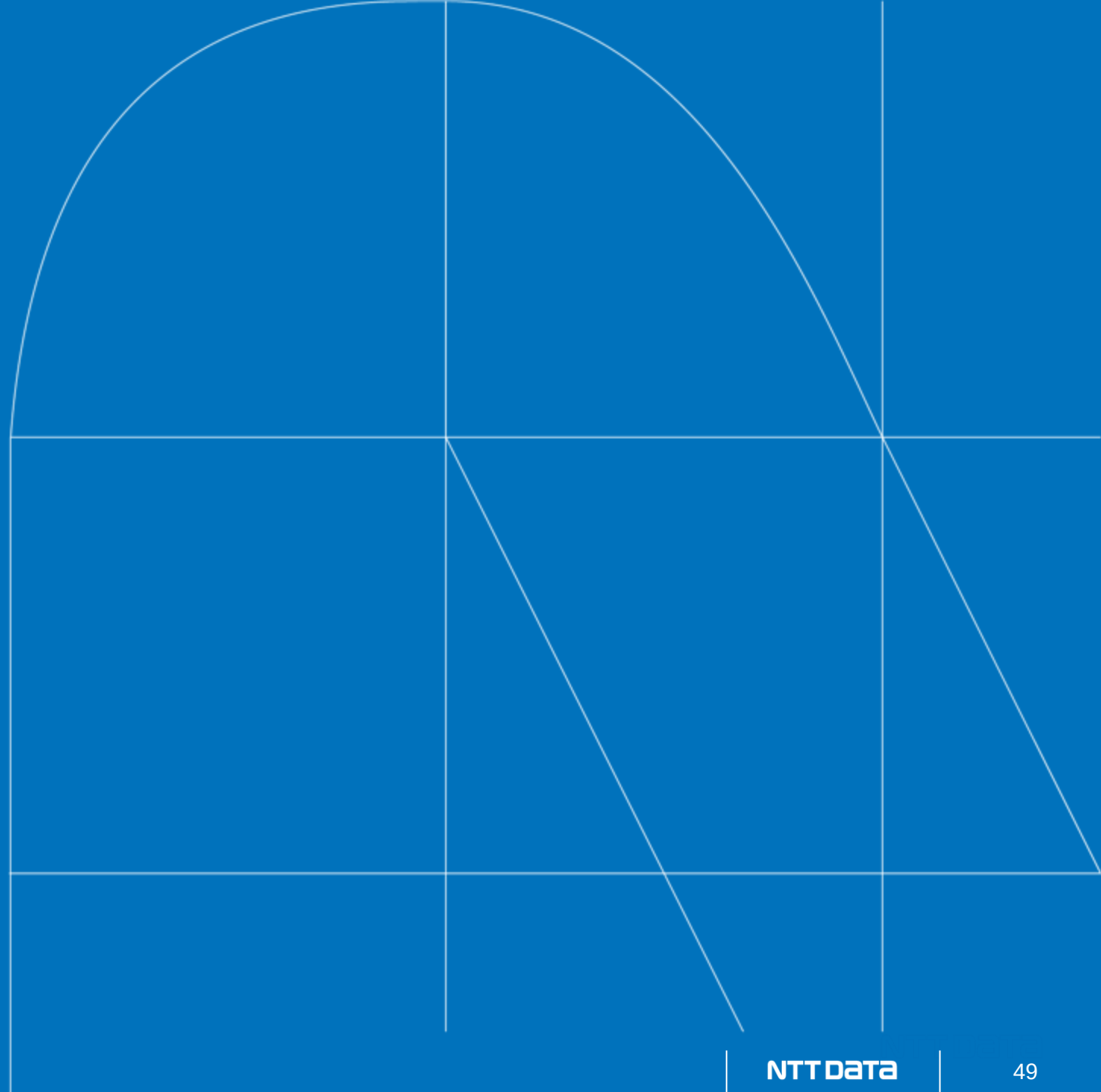
Dockerコンテナクラスタの停止:

```
$ ./docker-hadoop.sh --destroy
```

© 2024 NTT DATA Group Corporation

NTT DATA

bigtop-tests



© 2024 NTT DATA Group Corporation

smoke-tests

- パッケージをデプロイした状態で動くかざっくり確認するためのテスト
- おもにGroovyで実装されている
- CLIを実行して期待する出力が得られるかを確認というパターンが多い

smoke-testsのテストケースの例:

```
$ cat bigtop-tests/test-artifacts/hadoop/src/main/groovy/.../hdfs/TestTextSnappy.groovy
...
  @Test
  void testTextSnappy() {
    String cmd = "hadoop fs -text ${testDir}/${snappyFileName}"
    System.out.println(cmd)
    sh.exec(cmd)
    String output = sh.getOut().join("¥n")
    logError(sh)
    String expected = "1¥trafferty¥t31¥n2¥tjones¥t33¥n3¥tsteinberg¥t33"
    System.out.println("Expected output:¥n${expected}")
    System.out.println("Actual output:¥n${output}")
    assertEquals("Incorrect output", expected, output)
  }
```

© 2024 NTT DATA Group Corporation

NTT DATA

スモークテストの実行方法

- スモークテストの実行は以下の3ステップ。

```
$ curl -sLO https://downloads.apache.org/bigtop/bigtop-3.1.1/bigtop-3.1.1-project.tar.gz
$ tar xf bigtop-3.1.1-project.tar.gz
$ cd bigtop-3.1.1
```

1. Bigtop の tarball をダウンロードし展開

```
$ . /usr/lib/bigtop-utils/bigtop-detect-javahome
$ export HADOOP_CONF_DIR=/etc/hadoop/conf
```

2. JAVA_HOME, およびテストの実行に必要な環境変数の定義. (コンポーネントごとに必要な環境変数は異なる。確認するには展開したファイルの bigtop-tests/smoke-tests/コンポーネント名/build.gradle の doFirst メソッドを参照)

```
$ ./gradlew bigtop-tests:smoke-tests:yarn:test -Psmoke.tests --info

org.apache.bigtop.itest.hadoop.yarn.TestNode > testNodeBasic STANDARD_OUT
    -list
    -status

org.apache.bigtop.itest.hadoop.yarn.TestRmAdmin > testRmAdminBasic STANDARD_OUT
    -help
    -getGroups
...
```

3. この例ではYARNのテストを実行

© 2024 NTT DATA Group Corporation

NTT DATA

スモークテストの実行方法（続き）

テストの詳細な結果を知りたい場合は
これらのファイルを参照

```
Gradle Test Executor 2 finished executing tests.

> Task :bigtop-tests:smoke-tests:yarn:test
Finished generating test XML results (0.009 secs) into: /home/vagrant/bigtop-3.1.1/bigtop-tests/smoke-tests/yarn/build/test-results/test
Generating HTML test report...
Finished generating test html results (0.02 secs) into: /home/vagrant/bigtop-3.1.1/bigtop-tests/smoke-tests/yarn/build/reports/tests/test
Now testing...
:bigtop-tests:smoke-tests:yarn:test (Thread[Execution worker for ':',5,main]) completed. Took 7.453 secs.

BUILD SUCCESSFUL in 53s
```

© 2024 NTT DATA Group Corporation

Docker provisionerでtestを実行

- Docker provisionerで起動したクラスタ上でテストを実行することもできる

起動済みのDockerコンテナを利用してsmoke testsを実行:

```
$ cd provisioner/docker
$ ./docker-hadoop.sh --smoke-tests hdfs,yarn,mapreduce
```

© 2024 NTT DATA Group Corporation

NTT DATA

開発コミュニティの動向

© 2024 NTT DATA Group Corporation

Bigtopの開発コミュニティ

- Bigtop 3.0のリリース以降、Contributorは増えた印象
- Bigtop 3.1のリリース後に深刻な問題の報告(BIGTOP-3714)があり、3.1.1をリリースした
- Bigtop 3.2のリリース後にHadoop 3.3.5へのアップグレードのため、3.2.1をリリースした
- Committer/PMC memberは微増
- (Bigtopに限らず)中国系企業の技術者からのcontributionのボリュームが増えた印象
  - openEuler対応はその表れ

© 2024 NTT DATA Group Corporation

Apache Ambari

- Hadoopエコシステムのプロビジョニング/運用管理ツール
- 旧Hortonworks社が開発を主導
  - 旧Hortonworks社のHadoopディストリビューション(HDP)が前提
- BigtopはAmbariをパッケージングしているが、mpackは(ほとんど)未提供

- Cloudera社とHortonworks社との合併(2019)以降、開発は停止
  - Cloudera Managerと競合/重複するプロダクトのため
  - HDPのパッケージやソースコードも非公開化 (有償サブスクリプションが必須)
- ASFの開発プロジェクトはAtticに移動(2022/01): https://attic.apache.org
  - 活動休止したプロジェクトの情報を提供
  - 過去のリリース等の資材やWebサイトは残り、ソースコードはリードオンリー化

- Ambari利用企業の開発者が参加し活動再開(2022/09)
- デフォルトスタックをHDPからBigtopに変更
- Bigtopのmpack定義はAmbari側に移行

© 2024 NTT DATA Group Corporation

Hadoopの開発コミュニティ

- 3.4系(branch-3.4)、3.3系(branch-3.3) を維持
  - 3.4.0、3.3.6が最新版
    3.2系は2023年12にEOL宣言された
  - 2.10系は明示的にはEOL宣言されていない
  - もうリリースしないがセキュリティfixはbranch-2.10にもバックポートする可能性あり

- 3.4.0は、このスライド作成時点でリリース作業中
  - 1000以上の(fixed in 3.4.0 onlyな)修正がある
  - transitive dependenciesの変更が大きい
    - downstream productsに影響する可能性が高い
  - Java 11への(完全)移行はいまだに難航中

© 2024 NTT DATA Group Corporation

おわりに

© 2024 NTT DATA Group Corporation

おわりに

BigtopはHadoopエコシステムのプロダクトの「使える組み合わせ」を提供
Bigtop 3.3の大きなトピックはOpenEuler対応とRanger追加
主要プロダクトのバージョンアップ作業は進行中
Hadoop 3.4系対応はBigtop 3.4でやるかも

© 2024 NTT DATA Group Corporation

NTT DATA

Trusted Global Innovator

本資料に記載されている会社名、商品名、又はサービス名は、各社の登録商標又は商標です。

NTT DATA