



EDB

Moving to New Backup Tool: Barman

リリース 8.1.0

EnterpriseDB

2021 年 08 月 31 日

目次

第 1 章	Choosing your Backup Tool	3
1.1	Barman	3
1.2	pgBackRest	3
1.3	ツールの比較	4
第 2 章	Configuring the New Tool	5
2.1	機能の設定方法	5
2.1.1	Postgres 接続の詳細	6
2.1.2	pg_basebackup の代わりに直接ファイルコピーモード (ssh または rsync) を使用	6
2.1.3	増分バックアップ	6
2.1.4	WAL アーカイブ圧縮**	7
2.1.5	バックアップ圧縮	7
2.1.6	並列バックアップと復元	7
2.1.7	保持ポリシー	8
2.1.8	リポジトリの場所を設定する	8
第 3 章	Planning the Migration	9
3.1	テストと検証	9
3.2	サイジングに関する考慮事項	9
3.3	リソースの可用性とタイムライン通信	10
第 4 章	Executing the Migration	13
第 5 章	Maintaining and Cleaning Up	15
第 6 章	Scheduling	17
6.1	保持管理	18

Barman と pgBackRest は現在、EDB の推奨バックアップおよびデータリカバリツールです。現在 BART をバックアップおよびリカバリツールとして使用している場合、EDB はこれらのツールのいずれかに移行することをお勧めします。通常、ラージのマルチテラバイトデータベースをバックアップするためのパフォーマンス要件がない限り、Barman は正しい選択です。EDB は現在も引き続き BART をサポートしていますが、新しい機能や拡張機能は追加しません。EDB は、バックアップツールとして BART から移行するのに十分なリードタイムで BART サポートが終了することを通知します。BART は、PostgreSQL または EDB Postgres Advanced Server バージョン v14 以降ではサポートされません。

EDB のバックアップおよびリカバリツールオプションの詳細については、EDB Web サイトの [Backup and Recovery](#) を参照してください。

この文書は、お好みのツールに BART から移行する方法について、ご使用の環境のために右であるどのツールを選択すると、どのようにあなたが BART で使用する設定オプションに基づいてお好みのツールを構成する方法について説明します。

どのツールが環境に適したツールであるかは、特定のニーズによって大きく異なります。この文書を読んでもまだ質問がある場合、またはさらにサポートが必要な場合は、次のチャンネルからご連絡ください。

- 「カスタマーサポートメール : techsupport @ enterprisedb.com」
- 「試用支援 : trial-help @ enterprisedb.com」
- 「電話 : 」
- 米国 + 1-732-331-1320 / 1-800-235-5891
 - 英国 + 44-2033719820
 - ブラジル + 55-2139581371
 - インド + 91-20-66449612

CHOOSING YOUR BACKUP TOOL

このセクションでは、2 つの代替バックアップツールに関する情報を提供します。

1.1 Barman

EDB は、BART ユーザーがバーマンに移行することを推奨しており、すぐに以下の機能を追加する予定です

- PEM との統合
- 他のバックアップソリューションとの統合
- 他の統合を実装するための API

EDB が投資しているツールを選択する利点に加えて、PostgreSQL ポートを使用してリモートバックアップを行い、パスワードなしの ssh を使用していない場合、Barman が正しい選択です。詳細については、[Barman documentation](#) を参照してください。

1.2 pgBackRest

pgBackRest の利点は次のとおりです。

- ラージ（マルチテラバイト）データベースのバックアップのパフォーマンス上の利点。デルタバックアップでファイルの変更を検出するための内部アルゴリズムによりオーバーヘッドが少なくなり、部分的な復元により、ラージマルチデータベースクラスタから 1 つのデータベースのみを復元できます。Barman で計画されている機能であるバックアップ圧縮を提供します。

詳細については、[pgBackRest documentation](#) を参照してください。

1.3 ツールの比較

次の選択マトリクスは、一般的なレベルでツールを区別します

Capability	Barman	pgBackRest
PostgreSQL protocol	Yes	•
RPO=0	Yes	•
Rate limiting	Yes	
Custom WAL sizes	Yes	Yes (v11+)
Backup compression	Coming soon	Yes
Partial restore (only selected databases)	•	Yes
PEM integration	Coming soon	•

CONFIGURING THE NEW TOOL

このセクションでは、3 つのツールの比較を行い、設定 (config ファイル) とスケジューリングの違いを示します。移行中は、これらの違いをすべて認識し、必要に応じて変更を適用することが重要です。

2.1 機能の設定方法

次のセクションでは、実装の詳細を簡単に比較できるように、さまざまなツールで機能を有効にする方法について説明します。

- Postgres connection details
- Use of direct file copy mode (ssh or rsync) instead of pg_basebackup
- Incremental backups
- WAL archive compression
- Backup compression
- Parallel backup and restore
- Retention Policy
- Configure repository location

</ div>

2.1.1 Postgres 接続の詳細

Tool	Configuration
BART	In the [SERVER] section of the config file: [example]... <code>host = localhost</code> <code>user = postgres</code> <code>port = 5432</code>
Barman	In the config file: [example]... <code>conn info = host=localhost user=postgres dbname=postgres</code>
pg Back- Rest[^]	In the config file: [example]... <code>pg1 -host = localhost</code> <code>pg1-path = /var/lib/postgres /data</code> <code>pg1-user = postgres</code> <code>port = 5432</code>

pgBackRest がローカルで実行されている場合は pg1- パスを指定し、pgBackRest がリモートで実行されている場合は pg1- ホストを指定します

2.1.2 pg_basebackup の代わりに直接ファイルコピーモード (ssh または rsync) を使用

Tool	Configuration
BART	Increase <code>threads > 1</code> in the [SERVER] or [BART] section of the config file: [example]... <code>thread_count = 2</code> Or, as a commandline option <code>--no-pg_basebackup</code>
Barman	In the config file: [example]... <code>backup_method = rsync</code>
pg- Back- Rest	pgBackRest does not support pg_basebackup option

2.1.3 増分バックアップ

Tool	Configuration
BART	Retrieve the <code>backup_id</code> or <code>backup_name</code> for the parent: <code>bart SHOW-BACKUPS example</code> Now supply the <code>backup_id</code> or <code>backup_name</code> as parent: <code>bart BACK UP -s example --parent { backup_id \ backup_name }</code>
Barman	In the config file: [example]... <code>reuse_backup = link</code> Or, as a command line option: <code>--reuse-backup=link</code>
pg- Back- Rest	<code>pgbackrest --stanza=example --type=incr backup</code>

2.1.4 WAL アーカイブ圧縮**

Tool	Configuration
BART	In the [SERVER] or [BART] section of the config file: [example]… wal_compression = enabled
Barman	In the config file:[example]…compression = gzip
pgBack- Rest	Global compress settings. Can be overloaded for [global:archive-push]

2.1.5 バックアップ圧縮

Tool	Configuration
BART	Command line option: Enable: --gzipLevel: --compress-level
Barman	N/A
pgBackRest	compress=ycompress-level=9compress-type=gzcompress-level-network=3

2.1.6 並列バックアップと復元

Tool	Configuration
BART	Set thread_cound in the [SERVER] or [BART] section of the config file: [example]… thread_count = 4Or, as a Command line option --thread-count=4
Bar- man	Set parallel_jobs in the configfile:[example]…parallel_jobs = 4Or, as a com- mand line option --jobs 4
pg- Back- Rest	Set process-max in the configfile:[example]…process-max = 4Or, as a command line option --process-max=4

2.1.7 保持ポリシー

Tool	Configuration
BART	Set <code>retention_policy</code> in the <code>[SERVER]</code> or <code>[BART]</code> section of the config file: file:[]example <code>...retention_policy = 2 BACKUPS # Or DAYS, WEEKS, or MONTHS</code>
Barman	Set <code>retention_policy</code> , and/or <code>wal_retention_policy</code> options in the config file:[]example <code>...retention_policy = REDUNDANCY 2 Or retention_policy = RECOVERY WINDOW OF 2 DAYS # WEEKS/MONTHS.</code>
pgBackRest	Can be set in the config file:[]example <code>repo -retention-full-type = count #can also be time`` repo-retention-full = 2```` repo-retention-diff = 6 ```` # Retain WAL archives for only 1 full backup:```` repo-retention-archive-type = full #can also be diff or incr`` repo-retention-archive = 1See Retention Policy in edb documentation for more information.</code>

2.1.8 リポジトリの場所を設定する

Tool	Configuration
BART	In the config file in the <code>[BART]</code> section: <code>[BART]...backup_path = /tmp/bart</code>
Barman	In the config file in the main chapter: <code>backup_directory = /tmp/barman</code>
pgBackRest	<code>[global]repo1-path=/var/lib/pgbackrest</code>

PLANNING THE MIGRATION

EDB が移行計画に含めることを推奨している項目については、このセクションを確認してください。

3.1 テストと検証

各組織には、どの環境で何をすべきかを定義するさまざまなポリシーと機能があります。一部の組織は、テスト環境で一時システムに移行した後に稼動データベースを復元します。他の組織では、テスト環境でプロセスをテストし、稼動環境がテスト環境と同じように動作することを期待します。組織に最適なものが何であれ、以下のすべてが更新および検証されていることを確認してください

- 展開と管理の自動化
- データベースを復元するプロセス
- バックアップインフラストラクチャとの統合などの統合

EDB は、これらのすべてのアクションが既存の環境に影響を与えることなく実行できるように、特に新しいツールをオンボーディングするための一時的なテストシステムをセットアップすることをお勧めします。テスト計画には、テストシステムの立ち上げ、このテストシステムで行われるすべてのアクションの計画、および移行後のシステムの整理していることを確認してください。

一部の組織では、新しいツールに移行されたすべてのデータベースの正常な復元を確認する必要があります。

3.2 サイジングに関する考慮事項

実際の移行に応じて、さまざまなサイジングの考慮事項が適用されます。2 つの異なる移行経路が存在します

1. 新しいバックアップ用に新しいストレージが接続されます。

この場合、現在のサイズ設定要件に合わせてストレージのサイズを設定することから始めるのが適切です。これは、特大のストレージ場所を縮小する良い機会になるかもしれないことに注意してください。必要に応じて、両方のバックアップツールを同時に実行することを検討してください。インパクトが許容できる場合、バックアップを成功させるためのロールバックと追加のセキュリティのオプションになります。EDB は、最も簡単でエラーが発生しにくい、このアプローチを推奨しています。

2. 既存のバックアップのマウントポイントは、バックアップに再利用されます。

make への重要なノートは、さまざまなツールのためのリポジトリは異なるレイアウトを持っている、と互換性がないということです。つまり、正確な場所は明確にニーズがあります。ただし、それらは同じマウントポイントに存在できます。利点は、追加の必要なストレージが、新しいストレージを接続する場合より小さいと予想されることです。そうは言っても、移行中には追加のストレージが必要であり、ダウンスケーリングはオプションではないかもしれません。さらに、このオプションでは、両方のバックアップツールを同時に実行する余地がありません。追加のサイジング要件は、正確なバックアップスキーム、メンテナンススキーム、および差分バックアップのサイズの違いに依存します。適切な開始点は、追加のバックアップセット（1つの完全バックアップとすべての差分）、および追加のフルバックアップを準備することです。モニタリングしきい値も必要に応じて適切に調整されていることを確認してください。

古いツールと新しいツールの間で圧縮オプションが同様に構成されていることを確認 make ことが重要であることに注意してください。または、バックアップや WAL サイズを補うために追加のストレージスペースが必要です。

計画段階では、次のことに注意する必要があります。

- アプローチを選択します。
- 予想される追加ストレージのサイズを設定し、可用性を確認します。
- 移行計画のパートとして、追加のストレージを追加または増やすことを計画します。
- 次の余分なストレージを期待し、また必ず移行余分なストレージ>の間に直接的に利用可能であること、それは不測の事態のために必要とされなければならない make。
- 最初のアプローチ（追加のマウントポイント）を使用する場合は、移行後に古いストレージを消去することを計画し移行。古いリポジトリが組織のロールバックポリシーを満たすのに十分な期間保存されることを確認してください。

3.3 リソースの可用性とタイムライン通信

環境のサイズ、移行を実行できる利用可能な DBA の数、および自動化の量によっては、移行にかなりの時間がかかる場合があります。移行中、環境の一部が新しいツールで既に実行され、追加のストレージ要件が特定され、DBA リソースが移行に重点を置いており、他のタスクでは利用できないため、組織にストレスがかかる場合があります。

したがって、環境ごとに移行を計画し、その計画から移行の期間を導き出すことが重要です。これらのタイムラインを組織の残りの部分に伝えて、DBA チームがこの期間中は余分に占有されていることを認識させます。さらに、必要に応じて追加のストレージを準備して接続するためにストレージチームが利用できる make を確認します。

EXECUTING THE MIGRATION

移行の実行は非常に簡単です。手順には以下が含まれます

1. バックアップサーバにストレージ容量を追加する-新しいマウントポイント（または Cloud Storage エンドポイント）に移行するときの新しいマウントポイントとして。-同じマウントポイントを再利用する場合、既存のマウントポイントの追加ストレージとして。
2. バックアップサーバに新しいツールをインストールします。
3. 新しいツールの構成ファイルを構成します。あるツールの構成ファイルを別のツールに変更する方法の詳細については、How features are configured を参照してください。
4. スケジューリングを再構成します。スケジューリングを 1 つのツールから別のツールに変更する詳細については、Scheduling を参照してください。
5. 構成の確認-`barman check <example>-pgbackrest --stanza=<example> --log-level-console=info`
`check` 問題を修正します。
6. スケジューリングに依存する前に、初期バックアップを手動で実行します（推奨）。チェックコマンドは、最も一般的な問題を回復しますが、常に、まだそれが失敗したリストアのリスクを増加させるため、スケジュールされたバックアップが失敗した場合などストレージ>の問題、IO やネットワークレイテンシの問題、のようにそれは不幸になり、不測の問題がある可能性があります。チームがより多くの移行を実行するにつれて、必要に応じてこのステップをスキップ自信が増します。
7. 一部の組織では、バックアップを（別のシステムで）実際に復元および復元し、予想されるデータを確認することを好む場合があります。これはオプションの追加手順であり、組織のポリシーに大きく依存します。

MAINTAINING AND CLEANING UP

移行期間は、移行後に必要とされます。これは

- 空のリポジトリでは、完全なバックアップが作成され、通常のシチュエーションでは差分バックアップが取られます。
- 古いバックアップのクリーニングのメンテナンスプロセスは、その後にのみ開始されます保持期間を超えています。

この移行期間中、バックアップの問題を特定して修正するために、バックアップに特別な注意を払うニーズがあります。オプション

- モニタリング通知レベルを増やし、フリースペースのアラートしきい値を減らします。
- 移行中のバックアップのマニュアルチェックの計画。

EDB は、この移行期間中はオリジナルのバックアップソリューションを保持することをお勧めします。追加の安全対策または追加のロールバックオプションとして、システムを長期間保持することができます。新しいバックアップシステムが完全に動作し、余分なロールバック時間が超過した場合、古いシステムは次の方法でプロビジョニングを解除できます。

- システムからオリジナルのツールを削除します。
- 新しいマウントポイントが新しいストレージリポジトリにアタッチされた後、古いマウントポイントとストレージのプロビジョニングを解除します。
- 新しいツールが新しいサーバーで実行される場合、古いサーバーのプロビジョニングを解除します。

SCHEDULING

BART、Barman、および pgBackRest すべてのスケジュールので、スケジューリングを変更し、cron のを使用してバックアップが簡単に BART、Barman、および pgBackRest に予定 commands.Example コマンドを変更する必要があります

1. クラスタのフルバックアップを実行します ' 例'

Tool	Command
BART	bart BACKUP -s example
Barman	barman backup example
pgBackRest	pgbackrest backup --type=full --stanza=example

2. すべての構成済みサーバーのフルバックアップを実行します

Tool	Command
BART	bart BACKUP -s all
Barman	barman backup all
pgBack- Rest	pgbackrest does not have an option to run for all stanzas with one command

3. 増分バックアップを実行する

Tool	Command
BART	Retrieve the backup_id/backup_name for the parent (select a full for an incremental, select another inc/diff for a differential backup plan): bart SHOW-BACKUPS example Now supply the backup_id/backup_name as parent: bart BACKUP - s example --parent { backup_id \ Barman barman backup example - reuse-backup =link >This can also be set in the configuration file at the global/server level with reuse_backup = link pgBackRest Incremental : pgbackrest backup - type=incr - stanza=example >Differential pgbackrest backup - type=diff - stanza=example`

6.1 保持管理

- BART では通常、cron ジョブは、上のメンテナンスを実行するように設定されています BART リポジトリ (bart MANAGE)。-d オプションを指定すると、廃止されたバックアップは自動的にクリーンアップされます WAL アーカイブ)。
- 同様に、Barman は、上でメンテナンスを実行するために cron コマンドを使用しています Barman リポジトリ barman cron\。barman cron\コマンドより多くのことを処理します (ストリームされた WAL ファイルを WAL アーカイブディレクトリ)、およびすべてを実行するようにスケジュールする必要があります分。Barman rpm または debian パッケージは自動的に barman ユーザとして毎分実行される cron エントリー。

pgBackRest はエキスパイアコマンド (pgbackrest でメンテナンスを実行しますエキスパイア)、ただしエキスパイアコマンドはそれぞれの後に自動的に実行されます正常にバックアップ、個別にスケジュールする必要はありません。

新しいツールへの移行中に、適切な保存管理コマンドを実行 make ように cron が再構成されていることを確認してください。