Alibaba Cloud E-MapReduce

FAQ

Document Version20200120

目次

1 Alibaba Cloud Elastic MapReduce (EMR) のよくある質問

Q : **ジョブと実行プランの値がいいは何ですか。**

A : EMR では、ジョブを実行するために **2** つのステップが必要です。

・ ジョブの作成

EMR では、"ジョブ" を作成すると直接実行できない "ジョブ実行設定" が作成されます。 **EMR** では "ジョブ" を作成することにより、"ジョブの実行方法に関する設定" を作成することになります。 ジョブの設定には、ジョブのために実行される **Job JAR**、データの入出力アドレス、および特定の実行パラメーターを指定する必要があります。 そのような設定を作成してそれに名前を付けると、ジョブが定義されます。 ただし、実行中のジョブをデバッグする場合は実行プランが必要となります。

・ 実行プランの作成

実行プランは、ジョブをクラスターに関連付けます。 実行プランを介して、複数のジョブを **1** つのジョブシーケンスに組み合わせ、そのジョブ用に実行中のクラスターを準備することができます。 実行によって一時的なクラスターが自動的に作成されたり、ジョブが既存のクラスターに関連付けられたりすることもあります。 また、実行プランはジョブシーケンス用の定期的な実行プランの設定に役立ち、クラスターはタスクが完了すると自動的に解放されます。 実行レコード一覧で実行ステータスとログエントリを閲覧することも可能です。

Q : **ジョブログエントリの閲覧方法を教えてください。**

A : EMR システムでは、**jobid** でジョブ実行中のログエントリを **OSS** にアップロードしました。 パスは、クラスター作成時にユーザーが設定します。 **Web** ページ上で直接ジョブログを表示することができます。 ジョブをサブミットしたりスクリプトを実行したりするためにマスターノードにログオンすれば、スクリプトでログを特定できます。

Q : **コアノードにログオン方法を教えてください。**.

A : 以下の手順に従います。

1. マスターノードの **hadoop** アカウントに切り替えます。

```
su hadoop
```

2. パスワードなしの **SSH** 認証を使用して対応するコアノードにログオンします。

```
ssh emr-worker-1
```

3. **sudo** コマンドで **root** 権限を取得します。

```
sudo vi /etc/hosts
```

Q : OSS **でログを閲覧する方法を教えてください。**

A : OSS から直接すべてのログファイルを見つけてダウンロードすることもできます。 ただし、**OSS** はログを直接表示しないため、使用しづらくなることもあります。 ロギングを有効にして **OSS** のログの場所を指定した場合、どうすればジョブログエントリを見つけられるでしょうか。 たとえば、*OSS://mybucket/emr/spark* のパスを指定したとします。

1. **1.** 実行プランページに移動し、対応する実行プランを見つけて、**[**レコードの表示**]** をクリックし実行ログページを入力します。
2. **2.** 実行ログページで、最後の実行ログエントリなど特定のログエントリを探します。**.** 対応する **[**実行クラスター**]** をクリックして、実行クラスターの **ID** を表示します。
3. **3.** 次に *OSS://mybucket/emr/spark ######*の下のクラスター **ID** ディレクトリ *OSS://mybucket/emr/spark/* を検索します。
4. **4.** ジョブの実行 **ID** に応じて *OSS://mybucket/emr/spark/cluster ID/jobs* の下にある複数のディレクトリが表示されます。各ディレクトリには、ジョブの実行ログファイルが保存されています。

Q : **クラスター、実行プラン、実行ジョブのスケジューリングポリシーとは何ですか。**

A : 以下のとおり利用可能なスケジューリングポリシーは **3** つあります。

・ クラスターのスケジューリングポリシー

クラスター一覧では、各クラスターの実行時間を確認できます。 実行時間の計算式は以下のとおりです。実行時間 = クラスターが解放される時間 - クラスターが作成された時間 クラスターの作成と同時にスケジューリングが開始され、クラスターのライフサイクルが終了するまで続きます。

・ 実行プランのスケジューリングポリシー

実行プランの実行ログ一覧では、すべての実行プランの実行時間を確認でき、タイミングポリシーは **2** つの状況に要約できます。

- **1.** 実行プランがオンデマンドで実行される場合、実行ログの実行プロセスには、クラスターの作成、ジョブのサブミット、ジョブの実行、およびクラスターの解放が含まれます。オンデマンド実行プランの計算式は、以下のとおりです。実行時間 = クラスターが作成された時間 + 実行計画内のすべてのジョブの実行に使用される合計時間 + クラスターが解放される時間
- **2.** 実行プランが既存のクラスターに関連付けられている場合、クラスターの作成と解放は全体の実行サイクルに含まれません。 この場合は、実行時間 = 実行プラン内のすべてのジョブの実行に使用される合計時間、となります。

・ 実行中のジョブのタイミングポリシー

指定済みのジョブとは、実行プランに割り当てられているジョブのことです。 ジョブを参照するには、実行プランの実行ログの右側にある **[**ジョブリストの表示**]** をクリックします。 各ジョブの実行時間を計算する式は以下のとおりです。実行時間=ジョブが実行を停止する実際の時間 - ジョブが実行を開始する実際の時間。 ジョブが開始または停止する指定済みの時間とは、**Spark** または **Hadoop** クラスターによってジョブの実行または停止がスケジューリングされている時間のことです。

Q : **実行プランの初回実行時に利用可能なセキュリティグループがないのはなぜですか。**

A : セキュリティ上の理由で、既存のセキュリティグループを直接 **EMR** セキュリティグループとして使用することはできません。 そのため、**EMR** でセキュリティグループを作成していないと、利用可能なセキュリティグループを選択できません。テスト用のオンデマンドクラスターを作成するよう推奨します。 新しい **EMR** セキュリティグループはクラスターを作成するときに作成できます。 テストに合格後、実行プランをスケジューリングするスケジューリングサイクルを設定できます。 以前に作成したセキュリティグループも利用可能です。

Q : MaxCompute **の読み取り・書き込み時に** "java.lang.RuntimeException.Parse responsed failed: ‘<!DOCTYPE html>…’ **のエラーメッセージ が返されます。**DOCTYPE html>…’ " is returned when reading and writing MaxCompute .

A : odps トンネルエンドポイントが正しいかどうか確認します。 間違っているとこのエラーが発生します。

Q : **複数のコンシューマー** ID **が同じトピックを消費すると、**TPS**の競合が発生します。**

A : このトピックがベータテストその他の環境で作成されたことが原因で特定の消費者グループの消費データが競合する可能性があります。 チケットを起票して **ONS** に対応するトピックとコンシューマー **ID** をご報告・お問い合わせください。

Q : EMR **のワーカーノードのジョブログエントリを閲覧できますか。**

A : はい、できます。 前提条件 **:** クラスターを作成するときに **[**ログの保存**]** をクリックします。 ログエントリの閲覧方法 **: [**実行プラン一覧**] > [**もっと見る**] > [**ログの実行**] > [**ログの実行**] > [**ジョブ一覧の表示**] > [**ジョブ一覧**] > [**ワーカーログ**]** の順に選択します。

Q : Hive**で作成された外部テーブルにデータが含まれていないのはなぜですか。**

A : 例を挙げます。

```
CREATE EXTERNAL TABLE storage_log(content STRING) PARTITIONED BY (ds
STRING)
    ROW FORMAT DELIMITED
    FIELDS TERMINATED BY '\t'
    STORED AS TEXTFILE
    LOCATION 'oss://log-124531712/biz-logs/airtake/pro/storage';
    hive> select * from storage_log;
    OK
    所要時間: 0.3 秒
    作成した外部テーブルにはデータがありません。
```

Hive は指定されたディレクトリのパーティションディレクトリに自動的にバインドすることはありません。 手動でバインドする必要があります。たとえば以下のように記述します。

```
alter table storage_log add partition(ds=123);

                                                    OK
    所要時間 : 0.137 秒
    hive> select * from storage_log;
    OK
    abcd    123
    efgh    123
```

Q : Spark Streaming **ジョブが特に理由もなく停止するのはなぜですか。**

A : まず現在の **Spark** のバージョンが **1.6** より前かどうか確認します。 **Spark** バージョン **1.6** では、メモリリークのバグが修理されています。 このバグが原因でコンテナーのメモリ使用量が過剰に上昇し、ジョブが中断される恐れがあります。 このエラーが原因の **1** つである可能性があり、つまり **Spark 1.6** に問題がないわけではありません。 また、メモリ使用量を最適化するためコードを確認する必要があります。

Q : Spark Streaming **ジョブの終了時点で、ジョブのステータスが** EMR **コンソール上でまだ実行中になっているのはなぜですか。**

A：**Spark Streaming** ジョブの実行モードが **yarn-client** かどうか確認してください。 **Spark Streaming** ジョブの実行モードを **yarn-cluster** に設定します。 **yarn-client** モードにある **Spark Streaming** ジョブのステータスをモニタリングしていると **EMR** ではエラーが発生します。 この問題は早急に修理の対応をします。

Q : "**エラー: メインクラスが見つからない、または読み込めません" のエラーメッセージが返されるのはなぜですか。**

A : ジョブ設定で **Job Jar** のパスプロトコルヘッダーが `ossref` かどうか確認します。 **ossref** でない場合は `ossref` に変更します。

Q : **クラスター内のマシンの役割分担はどうなっていますか。**

A : EMR には、マスターノードと複数のスレーブまたはワーカーノードがあります。 マスターノードはデータ保存またはコンピューティングのタスクを担当しません。 データ保存またはコンピューティングのタスクはスレーブノードが担当します。 たとえば、**4** コア **8 GB** のマシンが **3** 台あるクラスターでは、**1** 台がマスタノードとして、他の **2** 台がスレーブノードとして機能します。 この場合、このクラスターで利用可能なコンピューティングリソースは、**2** 台の **4** コア **8 GB** マシンです。

Q : **ローカルの共有ライブラリを** MR **ジョブで使用する方法を教えてください。**

A : 以下に例を示しますが使用できる方法は複数あります。たとえば、*mapred-site.xml* ファイルを変更します。

```
<property>
    <name>mapred.child.java.opts</name>
    <value>-Xmx1024m -Djava.library.path=/usr/local/share/</value>
  </property>
  <property>
    <name>mapreduce.admin.user.env</name>
    <value>LD_LIBRARY_PATH=$HADOOP_COMMON_HOME/lib/native:/usr/local/
lib</value>
  </property>
```

必要に応じてライブラリを追加してローカル共有ライブラリファイルを使用できます。

Q : MR **または** Spark **ジョブで** OSS **データソースファイルパスを指定する方法を教えてください。**

A : 以下の `OSS URL: oss://[accessKeyId:accessKeySecret@]bucket[.endpoint]/object/path` を使用できます。

ジョブ内で入出力データソースを指定するために使用される **URI** で、`hdfs://` と類似しています。 **OSS** データの操作時はこの手順に従います。

- **(**推奨される方法**) EMR** は、**AssessKey** なしで **OSS** データへのアクセス、*// bucket/Object/path*パス上のファイルへの直接書き込みが可能になる **MetaService** を提供しています。
- **(**推奨されない方法**) AccessKeyId**、**AccessKeySecret**、および設定 **(Spark** ジョブの **SparkConf**、**MR** ジョブの設定**)** へのエンドポイントを設定、または **AccessKeyId**、**AccessKeySecret**、および **URL** のエンドポイントを直接指定できます。 詳細は、「開発の準備 *(Development preparation)*」セクションをご参照ください。

Q : Spark SQL **から "スレッド "メイン"** java.sql.SQLException **で例外が発生しました :** jdbc:mysql:xxx **の適切なドライバが見つかりません" のエラーメッセージが返ってくるのはなぜでしょうか。**

A :

- **1. mysql-connector-java** の以前のバージョンではこのような問題が発生する場合があります。 **mysql-connector-java** を最新のバージョンにアップデートしましょう。
- **2.** ジョブパラメーターで *–driver-class-path ossref://bucket/…/mysql-connector-java-[version].jar* を使用して *mysql-connector-java* パッケージをロードします。 また、*mysql-connector-java* パッケージを直接 **Job Jar** にインストールするとこの問題が発生することもあります。

Q : Spark SQL **を** RDS **に接続すると "サーバーのメッセージです。権限付与が正しく指定されていません。**IP **がホワイトリストに登録されていません" のエラーメッセージが返ってくるのはなぜですか。**

A : RDS ホワイトリストの設定を確認し、クラスターマシンの内部ネットワーク **IP** アドレスを **RDS** ホワイトリストに追加します。

Q : **設定が低構成の複数台のマシンにクラスターを作成する場合の注意事項を教えてください。**

A :

- マスターノードに **2** コア **4 GB** のマシンを選択する場合、マスターノードはメモリが高負荷状態になります。 これはメモリ不足の原因になりかねません。 マスターノードのメモリキャパシティを増設するよう推奨します。
- スレーブノードに **2** コア **4 GB** マシンを選択する場合、**MR** または **Hive** のジョブを実行するときはパラメーターを調整します。 **MR** ジョブの場合は `-D yarn.app.mapreduce.am.resource.mb=1024` パラメーターを追加します。 **Hive** ジョブの場合は `set yarn.app.mapreduce.am.resource.mb=1024` パラメーターを追加します。 この調整によりジョブの中断を防止できます。

Q : Spark SQL でインポートした Parquet テーブルを Hive または Impala のジョブで読み込むと "例外 java.io.IOException:org.apache.parquet.io.ParquetDecodingException で障害が発生しました。hdfs ファイルのブロック -1 に 0 の値を読み込めません// ... /…/part-00000-xxx.snappy.parquet" のエラーメッセージが返されるのはなぜでしょうか。/…/part-00000-xxx.snappy.parquet" returned when Hive or Impala jobs reads Parquet tables that are imported by Spark SQL ?

A : 10 進型で書き込む場合、Hive および Spark SQL は異なる変換方法を使用します。 この違いが原因で Hive が Spark SQL によってインポートされるデータを正しく読み取れない場合があります。 Hive または Impala で Spark SQL を使用してインポートされたデータを使用する必要がある場合は、`spark.sql.parquet.writeLegacyFormat=true` パラメーターを追加してからデータを再度インポートするよう推奨します。

Q : Beeline で Kerberos セキュリティクラスターへアクセスする方法について教えてください。

A :

- HA クラスター (ディスカバリーモード)

```
! connect jdbc:hive2://emr-header-1:2181,emr-header-2:2181,emr-
header-3:2181/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=
hiveserver2;principal=hive/_HOST@EMR.${clusterId). COM
```

- HA cluster (directly connected to a machine)

  Connect to emr-header-1.

```
! connect jdbc:hive2://emr-header-1:10000/;principal=hive/emr-header
-1@EMR.${clusterId}. COM
```

  Connect to emr-header-2.

```
! connect jdbc:hive2://emr-header-2:10000/;principal=hive/emr-header
-2@EMR.${clusterId}. COM
```

- 非 HA クラスター

```
! connect jdbc:hive2://emr-header-1:10000/;principal=hive/emr-header
-1@EMR.${clusterId}. COM
```

2 エラーメッセージ

エラーメッセージ : このリージョンでは従量課金インスタンスは利用できません。

クラスターを作成するリージョンで従量課金 **ECS** インスタンスを購入できない場合に返されるエラーメッセージです。 インスタンスを購入するには、別のリージョンに切り替えるよう推奨します。

エラーメッセージ : 不明なエラー、例外、または障害のため リクエスト処理が失敗しました。

これは **ECS** 管理システムで発生する不明なエラーです。 **EMR** は **Alibaba Cloud Elastic Compute Service (ECS)** に構築されており、このエラーの影響も受けます。 後で試すか、チケットを起票して問題のトラブルシューティングを実行できます。

エラーメッセージ : ノードコントローラは一時的に利用できません。

EMR は **ECS** に構築されています。 **ECS** 管理システムに一時的な問題がある場合に返されるエラーメッセージです。 後でクラスターを作成してみてください。

エラーメッセージ : 利用可能なクォータまたはゾーンがありません。

指定されたゾーンに使用可能な **ECS** クォータがない場合に返されるエラーメッセージです。 手動で別のゾーンに切り替え、またはシステムにより自動でゾーンを選択することもできます。

エラーメッセージ : 指定された InstanceType **は使用を許可されていません。**

従量課金の高構成インスタンス **(8** 個を上回る数のコアを持つインスタンス**)** を使用するには、申請が必要です。 申請するには、をクリックします。**.** アプリケーションが承認されたら高構成インスタンスを作成できます。 **8** コアの **16 GB**、**8** コアの **32 GB**、および **16** コアの **64 GB** のタイプなど、必ず **EMR** でサポートされているインスタンスを申請します。

3 ジョブ例外

Q : Spark のジョブレポート "コンテナーがメモリ制限を超えたため YARN によって無効にされました" または MapReduce のジョブレポート "コンテナーが物理的なメモリ制限を超えて実行中です" が発生するのはなぜですか。

A : アプリケーションがサブミットされた時点で割り当てられているメモリの量が少ないためです。 JVM は割り当てられた量を超えて起動時にメモリを大量に消費します。 この消費が原因となりジョブは NodeManager によって中断されます。 この現象は、より多くのオフヒープメモリを消費する可能性がある Spark ジョブにも影響します。ます。 Spark ジョブの対応としては、spark.yarn.driver.memoryOverhead または spark.yarn.executor.memoryOverhead の値を大きくします。 MapReduce ジョブの場合は、mapreduce.map.memory.mb および mapreduce.reduce.memory.mb の値を大きくします。

Q : ジョブをサブミットすると "エラー : Java ヒープスペース" が返されるのはなぜですか。

A : タスクのプロセス内に大量のデータがありますが、JVM のメモリが不足しています。 その結果、OutOfMemoryError エラーが返されます。 Tez ジョブの場合は、hive.tez.java.opts の値を大きくします。 Spark ジョブの場合は、spark.executor.memory または park.driver.memory の値を大きくします。 MapReduce ジョブの場合は、mapreduce.map.java.opts または mapreduce.reduce.java.opts の値を大きくします。

Q : ジョブをサブミットすると "デバイスに空き容量がありません" と返されるのはなぜですか。

A : マスターノードまたはワーカーノードのストレージ容量が不足しているため、ジョブのサブミットが失敗します。 ディスクがいっぱいになると、MySQL Server などのローカル Hive メタデータベースでの例外、または Hive Metastore 接続エラーが発生する場合があります。 システムディスクおよび HDFS 容量を含め、マスターノードの十分なディスク容量を空けておくよう推奨します。

Q : OSS または Log Service を使用すると、"ConnectTimeoutException" または "ConnectionException" が返されるのはなぜですか。

A : OSS エンドポイントはパブリックネットワークアドレスですが、EMR ワーカーノードにはパブリック IP アドレスがありません。 そのため、OSS や Log Service にはアクセスできません。 たとえば、`select * from tbl limit 10` の文は正常に実行できますが、`Hive SQL: select count(1) from tbl` は失敗します。

OSS エンドポイントを *oss-cn-hangzhou-internal.aliyuncs.com* など内部ネットワークアドレスに設定するか、**EMR** が提供する **MetaService** を使用します。 **MetaService** を使用する場合は、エンドポイントを指定する必要はありません。

```
alter table tbl set location "oss://bucket.oss-cn-hangzhou-internal.
aliyuncs.com/xxx"
alter table tbl partition (pt = 'xxxx-xx-xx') set location "oss://
bucket.oss-cn-hangzhou-internal.aliyuncs.com/xxx"
```

Q : Snappy **ファイルを読み込むと** "OutOfMemoryError" **が返されるのはなぜですか。**

A : Log Service で記述される標準の **Snappy** ファイルの形式は、**Hadoop Snappy** ファイルの形式と異なります。 **EMR** はデフォルトでは **Hadoop Snappy** ファイルを処理します。 標準の **Snappy** ファイルを処理すると、**OutOfMemoryError** エラーが返されます。 トラブルシューティングのため、対応するパラメーターの値を **true** に設定しておくことができます。 **Hive** ジョブの場合は、`set io.compression.codec.snappy.native=true` に設定します。 **MapReduce** ジョブの場合は、`Dio.compression.codec.snappy.native=true` に設定します。 **Spark** ジョブの場合は、`spark.hadoop.io.compression.codec.snappy.native=true` に設定します。

Q : EMR **クラスターを** RDS **インスタンスに接続すると、"無効な権限付与仕様が無効です。サーバーからのメッセージ** : "IP **がホワイトリストにもブラックリストにも登録されていません。クライアントの** IP **アドレスが** xxx **です" と返されるのはなぜですか。**

A : EMR クラスターを **RDS** インスタンスに接続する場合は、**RDS** インスタンスのホワイトリストを設定する必要があります。 ホワイトリストにクラスタノードの **IP** アドレスを追加しないと、特にクラスターの拡張後にこのエラーが発生します。

Q : OSS **データの読み取りまたは書き込み時に "スレッド "メイン" 例外**java.lang.RuntimeException: java.lang.ClassNotFoundException: Class com.aliyun.fs.oss.nat.NativeOssFileSystem **が見つかりません" と返されるのはなぜですか。**

A : Spark ジョブで **OSS** データを読み書きするときは、**EMR SDK** をジョブ **JAR** にパッケージ化する必要があります。 詳細は、「前提条件 *(Prerequisites)*」をご参照ください。

Q : Spark **が** Flume **に接続されていると、**Spark **ノードの使用可能なメモリを超えるのはなぜですか。**

A : データ受信モードがプッシュベースかどうかを確認します。 この設定になっていない場合は、モードをプッシュベースに設定します。 詳細は、「ドキュメント *(Documentation)*」をご参照ください。

Q : OSS をインターネットに接続すると、"原因 : java.io.IOException: 5242880 バイトが書き込まれたため、使用可能なバッファーサイズが 524288 を超え入力ストリームをリセットできません" と返されるのはなぜですか。

A : ネットワーク接続の再試行中にキャッシュ保存用の十分な容量がないために発生するバグです。 EMR SDK を V1.1.0 以降のバージョンで使用するよう推奨します。

Q : Spark 実行中に "メタアクセスに失敗しました。runtime.org.apache.hadoop.hive.ql.metadata.HiveException: java.lang.RuntimeException でこのクラスにアクセスしないでください。org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient のインスタンスを作成できません" と返されるのはなぜですか。 This class should not accessed in runtime.org.apache.hadoop.hive.ql.metadata.HiveException: java.lang.RuntimeException: Unable to instantiate org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient” returned when Spark is running ?

A : Spark が Hive データを処理するときは、Spark の実行モードを yarn-client または local に設定する必要があります。 モードを yarn-cluster に設定しないようにします。 yarn-cluster にするとエラーが発生します。 ジョブの JAR パッケージにサードパーティのファイルが含まれていると、Spark の実行中にこのエラーが発生する場合があります。

Q : Spark で OSS SDK を使用すると、"java.lang.NoSuchMethodError:org.apache.http.conn.ssl.SSLConnetionSocketFactory.init(Ljavax/net/ssl/SSLContext;Ljavax/net/ssl/HostnameVerifier)" が返されるのはなぜですか。

A : OSS SDK が依存している http-core および http-client パッケージは、Spark および Hadoop の実行環境とバージョン依存関係が競合しています。 コード内で OSS SDK を使用しないよう推奨します。 使用する場合は、手動でこの問題を解決する必要があります。 オブジェクトの一覧表示など、OSS ファイル処理の基本的な操作を実行する必要がある場合は、ここをクリックして OSS ファイルの処理方法に関する詳細情報を閲覧します。

Q : OSS を使用すると、"java.lang.IllegalArgumentException: 不正な FS : oss:// xxxxx、予期された状態 : hdfs://ip:9000" が返されるのはなぜですか。

A : OSS データ処理時は HDFS のデフォルトのファイルシステムが使用されます。 ファイルシステムを以下のステップで OSS 上のデータ処理に使用できるよう初期化するには、OSS パスを使用する必要があります。

```
Path outputPath = new Path(EMapReduceOSSUtil.buildOSSCompleteUri("oss
://bucket/path", conf));        org.apache.hadoop.fs.FileSystem fs =
org.apache.hadoop.fs.FileSystem.get(outputPath.toUri(), conf);
if (fs.exists(outputPath)) {
  fs.delete(outputPath, true);
```

```
}
```

Q : **ガベージコレクションに時間がかかり、ジョブの実行が遅くなるのはなぜですか。**

A : ジョブを実行する **JVM** 上のヒープメモリのサイズが小さすぎると、ガベージコレクションに時間がかかり、ジョブのパフォーマンスに影響が出る場合があります。 Java ヒープサイズを大きくするよう推奨します。 **Tez** ジョブの場合は、`hive.tez.java.opts` **Hive** パラメーターの値を大きくします。 **Spark** ジョブの場合は、`spark.executor.memory` または `spark.driver.memory` の値を大きくします。 **MapReduce** ジョブの場合は、`mapreduce.map.java.opts` または `mapreduce.reduce.java.opts` の値を大きくします。

Q : AppMaster **がタスクの開始に時間がかかるのはなぜですか。**

A : ジョブタスクや **Spark** エグゼキュータの数が多すぎると、**AppMaster** がタスクの開始に時間がかかることがあります。 単一タスクの実行時間は短く、ジョブをスケジューリングするためのオーバーヘッドは大きくなります。 **CombinedInputFormat** を使用してタスク数を減らすよう推奨します。 以前のジョブで作成されたデータのブロックサイズ (`dfs.blocksize`)、または `mapreduce.input.fileinputformat.split.maxsize` の値を大きくすることもできます。 **Spark** ジョブの場合は、エグゼキュータ (`spark.executor.instances`) の数、または同時ジョブ (`spark.default.parallelism`) の数を減らすことができます。

Q : **リソースの申請に長い時間がかかり、ジョブ保留問題の原因となるのはなぜですか。**

A : ジョブがサブミットされたら、**AppMaster** はタスクを開始するためのリソースを申請する必要があります。 この期間中にクラスターが占有されており、リソースの申請に長い時間がかかるため、ジョブ保留の問題が発生する場合があります。 リソースグループの設定が不適切かどうか、および現在のリソースグループは占有されているがクラスターにまだ利用可能なリソースがあるかどうかを確認するよう推奨します。 設定が不適でクラスターに余裕がある場合は、主要なリソースグループの設定を調整したり、リソースを最大限活用できるようクラスターのサイズを変更したりできます。

Q : **少数のタスクの実行に時間がかかり、ジョブの全体的な実行時間が長くなるのはなぜですか (データスキューの問題)。**

A : タスクの特定の段階では、データは不均等に分散されています。 この状況では、ほとんどのタスクは迅速に実行されますが、データが大量なため少数のタスクの実行に長い時間がかかります。 これにより、ジョブ全体の実行時間が長くなります。 **Hive** の **mapjoin** 機能と `set hive.optimize.skewjoin = true` の使用を推奨します。

Q : **失敗したタスクの試行によってジョブの実行時間が長くなるのはなぜですか。**

A : ジョブには失敗したタスクまたは失敗したジョブの試行があります。 ジョブは正常に終了する可能性がありますが、失敗した試行によってジョブの実行時間が長くなる可能性があります。このセクションでタスクの失敗の原因を突き止めるよう推奨します。

Q : Spark **ジョブの実行中に** "java.lang.IllegalArgumentException: Size **が** Integer.MAX_VALUE **を超えています" と返されるのはなぜですか。**

A : パーティション数が少なすぎると、ブロックサイズが大きくなりすぎる場合があります。データシャッフルを実行すると、**Integer.MAX_VALUE (2 GB)** の最大値を超えることがあります。 データシャッフルの実行前に、パーティション数を増やして `spark.default.parallelism`、**spark.sql.shuffle.partitions** の値を大きくするか、パーティション再作成の操作を実行するよう推奨します。

4 実行プランの使用

高構成インスタンスに適用

高構成インスタンスを有効化していない場合、高構成インスタンスを使用してクラスタを作成するとエラーが発生し、以下のエラーメッセージが表示されます。

指定された **InstanceType** の使用は許可されていません。

Click ここ をクリックしてチケットを起票し高構成インスタンスを有効化します。

セキュリティグループの使用

EMR でクラスターを作成する場合は、**EMR** で作成されるセキュリティグループを使用する必要があります。 アクセス可能な **EMR** のクラスターのポートが **22** だけなのでセキュリティグループが必要になります。 既存のインスタンスを機能に基づいてさまざまなセキュリティグループにソートするよう推奨します。 たとえば、**EMR** のセキュリティグループは **"EMR-security group"** で、既存のセキュリティグループの名前は **"User-security group"** とすることができます。 各セキュリティグループは、ニーズに基づいて独自のアクセス制御を適用します。 作成されたクラスターにセキュリティグループをバインドする必要がある場合は、以下のステップに従います。

- **EMR** クラスターを既存のセキュリティグループに追加します。

  [詳細] をクリックします。 すべての **ECS** インスタンスに関連するセキュリティグループが表示されます。 **ECS** コンソールで左下角の **[セキュリティグループ]** タブをクリックしてセキュリティグループ **"EMR-security group"** を見つけます。 **[インスタンスの管理]** をクリックします。 **emr-xxx** で始まる **ECS** インスタンス名が表示されます。 これらは、**EMR** クラスター内の対応する **ECS** インスタンスです。 これらのインスタンスをすべて選択し、右上角の **[セキュリティグループへ移動]** をクリックしてこれらのインスタンスを別のセキュリティグループに移動します。
- 既存のクラスターを **"EMR** セキュリティグループ**"** に追加します。

  既存のクラスターが配置されているセキュリティグループを探します。 上記の操作を繰り返して、クラスターを **"EMR -security group"** に移動します。 **ECS** コンソールでクラスターが使用していないインスタンスを選択し、バッチ操作を使用してそれらのインスタンスを **" EMR -security group"** に移動します。

- セキュリティグループのルール

  ECS インスタンスが複数の異なるセキュリティグループに属している場合、セキュリティグループのルールは **OR** 関係に従います。 たとえば、 **"User-security group"** はすべてのポートがアクセス可能ですが、**EMR security** はポート **22** だけがアクセス可能です。 **EMR** クラスターを **"User-security group"** に追加すると、**EMR** オープンのインスタンスのすべてのポートがアクセス可能になります。 なお、以下次の点に注意してください。

   :

  セキュリティグループのルールを設定するときは、必ず **IP** アドレス範囲でアクセスを制限します。 攻撃を回避する **IP** 範囲は **0.0.0.0** に設定しないようにしましょう。

実行プランに関するよくある質問

- 実行プランの編集

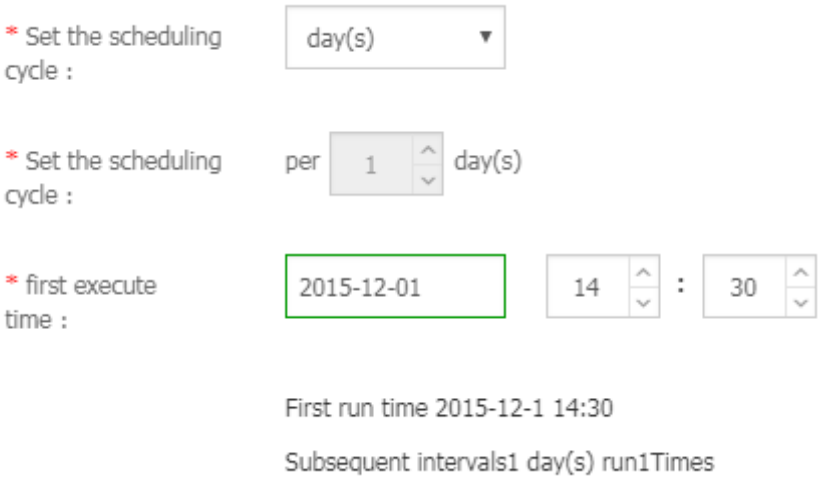
  ステータスが実行中またはスケジューリング中でない実行プランを編集できます。 編集ボタンがクリックできない場合は、実行プランのステータスを確認してから再試行します。

- 実行プランの実行

  実行プラン作成時にスケジューリングモードを即時実行に設定すると、プランは作成後に自動で実行されます。 既存の実行プランである場合は、手動で実行プランを実行する必要があります。 実行プランは作成後すぐには実行されません。

- 定期実行時間

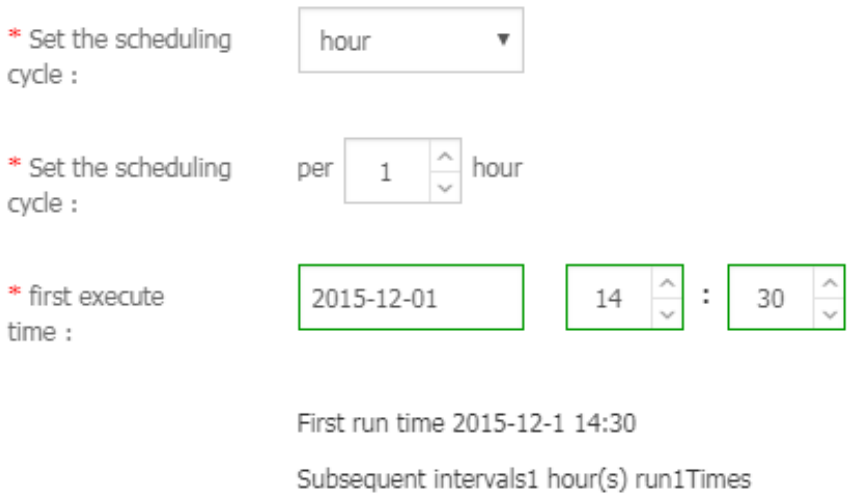
  定期実行クラスターの開始時刻は、実行プランが実行を開始する時刻を示しています。 時間の精度は分単位です。 スケジュールのサイクルは、開始時刻からの 2 つの実行の間の間隔を示します。 以下に例を示します。



  初回の実行は **2015** 年 **12** 月 **1** 日の **14:30:00** であり、**2** 回目の実行は **2015** 年 **12** 月 **2** 日の **14:30:00** です。 実行プランは **1** 日に **1** 回実行されます。

  現在の時刻がスケジューリングした時刻より遅い場合、スケジューリングの最新時刻は **2015** 年 **12** 月 **1** 日の **14** 時 **30** 分 **00** 秒 です。

  例 **:**



  現在時刻が **2015** 年 **12** 月 **2** 日 **9** 時 **30** 分である場合、スケジューリングの最新時刻は、スケジューリング規則に基づく **2015** 年 **12** 月 **2** 日 **10** 時 **00** 分 **00** 秒です。 この時点で初回の実行が開始します。