

Alibaba Cloud Table Store

資料通道

檔案版本：20180929

目錄

1 MaxCompute	1
1.1 概述	1
1.2 增量同步處理（嚮導模式）	2
1.3 全量匯出（指令碼模式）	9
2 OSS	15
2.1 概述	15
2.2 全量匯出（指令碼模式）	16
2.3 增量同步處理（指令碼模式）	19

1 MaxCompute

1.1 概述

[Table Store](#) (Table Store) 是構建在阿里雲飛天分布式系統之上的分布式NoSQL資料存放區服務，根據99.99%的高可用以及11個9的資料可靠性的標準設計。Table Store通過資料分區和負載平衡技術，實現資料規模與訪問並發的無縫擴充，提供海量結構化資料的儲存和即時訪問。

[MaxCompute](#) (原ODPS) 是一項大資料計算服務，它能提供快速、完全託管的PB級資料倉儲解決方案，使您可以經濟並高效地分析處理海量資料。

使用情境

Table Store：提供專業的資料持久化儲存服務，以及面向使用者的即時高並發低延遲讀寫操作。

MaxCompute：提供計算服務，一般用來實現資料清洗、修正、計算等功能。

開通服務

開通Table Store

1. 登入[Table Store](#)。
2. 單擊立即開通。
3. 在[Table Store](#)建立[Table Store](#)和[Table Store](#)。



说明：

- 使用增量通道，資料表必須開啟[Stream](#)，有效時間可以選擇24小時。
- Table Store的讀寫CU分為預留CU和按量CU。如果您在建立表時將預留CU設定為0，表示只使用按量CU。在使用過程中，預留CU可以隨時進行修改。
- Table Store為您提供每月10 GB，1000萬按量讀和1000萬按量寫的免費額度。

開通MaxCompute

1. 登入[MaxCompute](#)。
2. 單擊立即購買。



说明：

有預付費和隨用隨付兩種收費模式。

資料通道

- 即時

直讀直寫

- 離線

- 增量同步處理到MaxCompute

- ####

- 全量匯出到MaxCompute

- #####

- 全量匯入到Table Store

指令碼模式

1.2 增量同步處理（嚮導模式）

Data Integration（Data Integration）產品提供資料同步服務，有嚮導模式和指令碼模式兩種方式。嚮導模式更簡單，指令碼模式更靈活。

本節介紹如何將Table Store中的增量資料（Put/Update/Delete）通過Table Store功能近即時地同步到MaxCompute中。



说明：

目前使用離線方式同步，因此同步延遲約10分鐘。

步驟1. 建立Table Store資料來源

1. 登入 #####。
2. 如果是首次使用Data Integration，需要##Data Integration##。
3. 在資料來源頁面，單擊新增資料來源。
4. 選擇Table Store作為資料來源。
5. 輸入參數並單擊測試連通性。

參數說明如下：

參數	說明
資料來源名稱	資料來源的名稱，例如，gps_data。
資料來源描述	資料來源的描述。
Endpoint	填入Table Store執行個體頁面的執行個體地址。 - 如果Table Store的執行個體和MaxCompute在同一個region，填入私網地址。 - 如果Table Store的執行個體和MaxCompute不在同一個region，填入公網地址。  说明： 不能填入VPC地址。
Table Store執行個體ID	Table Store的執行個體的名稱。
Access Id	登入帳戶的AccessKeyID。
Access Key	登入帳戶AccessKeyID對應的AccessKeySecret。



说明：

如果連通性測試失敗，檢查endpoint和執行個體名稱是否正確。如果仍然無法解決，請[####](#)。

6. 單擊完成，在資料來源頁面會出現該資料來源資訊。

步驟2. 建立MaxCompute資料來源

本操作與步驟1類似，只是選擇MaxCompute (ODPS)作為資料來源。

本樣本中，該資料來源名稱使用OTS2ODPS。

步驟3. 建立增量即時資料通道

1. 在[#####](#)頁面單擊同步任務。
2. 在頁面右側單擊第二步，建立同步任務。
3. 選擇嚮導模式。
4. 選擇步驟1建立的資料來源。

数据来源类型'. The form contains several fields: '* 数据源' with a dropdown menu showing 'gps_data (ots)', '* 表' with a dropdown menu showing 'sha...', '* 开始时间' with a text input '\$\${startTime}', '* 结束时间' with a text input '\$\${endTime}', '* 状态表' with a text input 'TableStoreStreamReaderStatusTable', '最大重试次数' with a text input '30', and '导出时序信息' with a checkbox and a help icon."/>

參數說明如下：

參數	說明
資料來源	已建立的Table Store資料來源。在本樣本中，選擇gps_data。

參數	說明
表	Data Integration自動從Table Store擷取最新資料表。 選擇的表必須開通Stream。如果沒開通Stream，單擊右側一鍵開通 Stream 即可開通。 增量資料有效期間最大為24小時。
開始時間	增量匯出的起始時間。 如果是周期性的任務，必須填入變數值，預設值為\${start_time}。
結束時間	增量匯出的結束時間。 如果是周期性的任務，必須填入變數值，預設值為\${end_time}。
狀態表	用於儲存增量匯出過程中的一些狀態值，推薦使用預設值。
最大重試次數	網路不穩定時的最大重試次數，預設為30次，可自訂配置。
匯出時序資訊	匯出的資料中是否包含時間資訊，預設不勾選。

5. 在選擇目標頁面，選擇步驟2建立的資料來源。

您要选择业务数据的目标，可以是您独立的数据库服务器，也可以是阿里云的RDS等，查看支持的[数据目标类型](#)

* 数据源：OTS2ODPS (odps) ?

* 表：ots_gps_data 一键生成目标表

* 分区信息：pt = \${bdp.system.bizdate} ?

清理规则：
☒ 写入前清理已有数据 Insert Overwrite
☐ 写入前保留已有数据 Insert Into

參數說明如下：

參數	說明
資料來源	已建立的MaxCompute資料來源。在本樣本中，選擇OTS2ODPS。

參數	說明
表	選擇該資料來源下的表。如果沒有表，可以單擊右側一鍵產生目標表建立。在彈出的對話方塊中，將your_table_name修改為要建立的表名，例如，ots_gps_data，如下圖所示。（由於timestamp是MaxCompute的保留欄位，此處不能使用，本樣本中，用ts代表timestamp。） http://docs.aliyun.cn-hangzhou.oss.aliyun-inc.com/assets/pic/61034/cn_zh/1510908157657/61034-6.png
分區資訊	預設值為 \${bdp.system.bizdate}，表示MaxCompute中的資料按日期分區。
清理規則	選擇寫入前清理已有資料 Insert Overwrite 。

- 在欄位對應頁面，確保Table Store表和MaxCompute表完成映射。欄位對應頁面左側是TableStore的列名，在多版本（預設）模式下不能修改，必須是自動產生的pk，colname，version，colvalue，optype和sequenceinfo，其中pk是使用者table中的pk列名。
- 在通道控制頁面，設定參數。

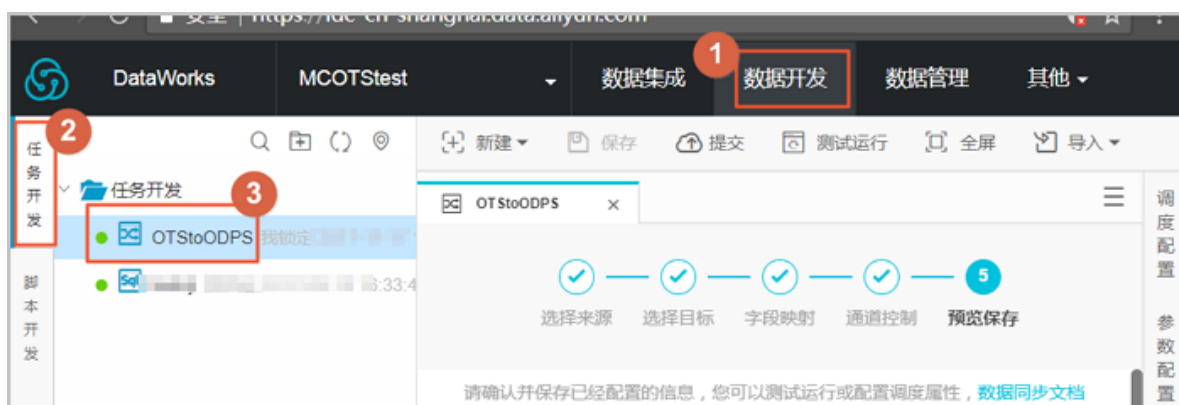
參數說明如下：

參數	說明
作業速率上限	範圍：1MB/s – 20MB/s。如果需要提高上限，請####。
作業並發數	最大可以設定為10。每個作業的最大速率 = 作業速率上限/作業並發數。
錯誤記錄數	出錯率的條數超過此值後任務會失敗，預設為0。

- 在預覽儲存頁面，檢查配置。
- 單擊儲存，儲存任務。本樣本中，儲存的任務名為OTStoODPS。

步驟4. 設定調度參數

- 單擊資料開發。
- 在任務開發頁簽中，雙擊開啟已建立的任務OTStoODPS。



3. 單擊調度配置，設定調度參數。

按照如下參數配置後，任務會從第二天開始執行。



參數說明如下：

參數	說明
調度狀態	預設不勾選，表示執行。
出錯重試	建議勾選，表示出錯後會重試。
生效日期	建議使用預設值。
調度周期	本樣本中，選擇分鐘。
開始時間	本樣本中，設為00時00分。
調度間隔	本樣本中，選擇5分鐘。
結束時間	本樣本中，設為23時59分。
相依性屬性	在相依性屬性地區，根據業務情況填寫，或者使用預設值。
跨周期依賴	在跨周期依賴地區，根據業務情況填寫，或者使用預設值。

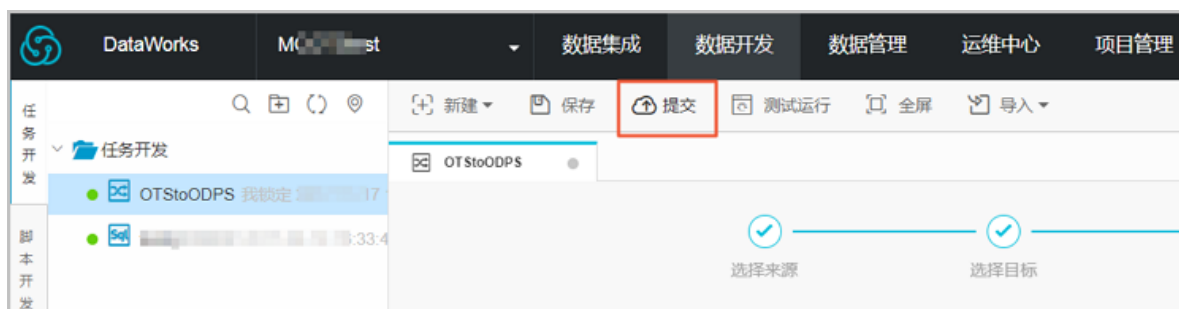
4. 單擊參數配置。

參數說明如下：

參數	說明
<code>\${bdp.system.bizdate}</code>	無需配置。
startTime	在調度配置中設定開始時間的變數。在本樣本中，設為 <code>[\$[yyyymmddhh24miss-10/24/60]</code> ，表示調度任務開始的時間減去10分鐘。
endTime	在調度配置中設定結束時間的變數。在本樣本中，設為 <code>[\$[yyyymmddhh24miss-5/24/60]</code> ，表示調度任務開始的時間減去5分鐘。

步驟5. 提交任務

1. 在任務詳情頁上方單擊提交。



2. 在彈出的對話方塊中，單擊確定提交。

提交成功後，會提示當前檔案為唯讀狀態。

步驟6. 查看任務

1. 單擊營運中心。



2. 在左側導覽列中，選擇工作清單 > 周期任務，可以查看建立的任務OTStoODPS。

3. 任務會從第二天的00點00分開始執行。

- 在左側導覽列中，選擇任務營運 > 周期執行個體，查看當天需要啟動並執行周期任務。單擊執行個體名稱，可以查看詳情。
- 當單個任務在運行中或運行結束後可以查看日誌。

步驟7. 查看匯入到MaxCompute中的資料

1. 單擊資料管理。



2. 在左側導覽列，單擊尋找資料，會列出MaxCompute中的所有表。

3. 找到剛剛匯入資料的表（ots_gps_data），單擊後進入該表的詳情頁。

4. 單擊資料預覽頁籤，查看匯入的資料。

1.3 全量匯出（指令碼模式）

Data Integration（Data Integration）產品提供資料同步服務，有嚮導模式和指令碼模式兩種方式。嚮導模式更簡單，指令碼模式更靈活。

本節介紹如何將Table Store中的全量資料（Put/Update/Delete）通過Data Integration全量匯出到MaxCompute中。

步驟1. 建立Table Store資料來源



說明：

- 如果已經建立了Table Store的資料來源，可以跳過此步驟。
- 如果您不希望建立資料來源，也可以在後續的配置頁面中配置相應的endpoint、instanceName、AccessKeyID和AccessKeySecret。

建立資料來源的具體步驟，請參見[##Table Store####](#)。

步驟2. 建立MaxCompute資料來源

本操作與步驟1類似，只是選擇MaxCompute (ODPS)作為資料來源。

本樣本中，該資料來源名稱使用OTS2ODPS。

步驟3. 建立全量匯出通道

1. 在[#####](#)頁面單擊同步任務。

2. 選擇指令碼模式。

3. 在彈出的匯入模板對話方塊中，來源類型選擇OTS，目標類型選擇ODPS。
4. 單擊確認，進入配置頁面。
5. 設定配置參數。

```
{
  "type": "job",
  "version": "1.0",
  "configuration": {
    "setting": {
      "errorLimit": {
        "record": "0"      # 能夠允許的最大錯誤數
      },
      "speed": {
        "mbps": "1",      # 最大的流量，單位MB。
        "concurrent": "1" # 並發數。
      }
    },
    "reader": {
      "plugin": "ots",    # 讀取的外掛程式名稱
      "parameter": {
        "datasource": "", # 資料來源名稱
        "table": "",      # 表名
        "column": [      # 需要匯出到MaxCompute中去的Table Store中的列名
          {
            "name": "column1"
          },
          {
            "name": "column2"
          },
          {
            "name": "column3"
          },
          {
            "name": "column4"
          },
          {
            "name": "column5"
          }
        ],
        "range": {      # 需要匯出的資料範圍，如果是全量匯出，則需要從INF_MIN到
          INF_MAX
          "begin": [    # 需要匯出資料的起始位置，最小的位置是INF_MIN。begin中的配
            置項數目個數和Table Store中相應表的主鍵列個數一致。
            {
              "type": "INF_MIN"
            },
            {
              "type": "INF_MIN"
            },
            {
              "type": "STRING", # 這個配置項的意思是：第三列的起始位置是
begin1。
              "value": "begin1"
            },
            {
              "type": "INT",    # 這個配置項的意思是：第四列的起始位置是0。
              "value": "0"
            }
          ],
          "end": [      # 匯出資料的結束位置
```

```

    {
      "type": "INF_MAX"
    },
    {
      "type": "INF_MAX"
    },
    {
      "type": "STRING",
      "value": "end1"
    },
    {
      "type": "INT",
      "value": "100"
    }
  ],
  "split": [ # 這一項配置分區範圍，一般可以不配置，如果效能較差，可以提工
單或者加入釘釘群：11789671 找工作人員協助您設定
    {
      "type": "INF_MIN"
    },
    {
      "type": "STRING",
      "value": "splitPoint1"
    },
    {
      "type": "STRING",
      "value": "splitPoint2"
    },
    {
      "type": "STRING",
      "value": "splitPoint3"
    },
    {
      "type": "INF_MAX"
    }
  ]
}
},
"writer": {
  "plugin": "odps", # MaxCompute寫入的外掛程式名
  "parameter": {
    "datasource": "", # MaxCompute的資料來源名稱
    "column": [], # MaxCompute中的列名，這個列名順序對應TableStore中的列
名順序
    "table": "", # MaxCompute中的表名，需要提前建立好，否則任務會失敗。
    "partition": "", # 如果表為分區表，則必填。如果表為非分區表，則不能填
寫。需要寫入資料表的分區資訊，必須指定到最後一級分區。
    "truncate": false # 是否清空之前的資料
  }
}
}
}
}

```



说明：

詳細的配置參數可參考[##Table Store Reader](#)和[##MaxCompute Writer](#)。

6. 單擊儲存，儲存任務。

步驟4. 運行任務（測試）

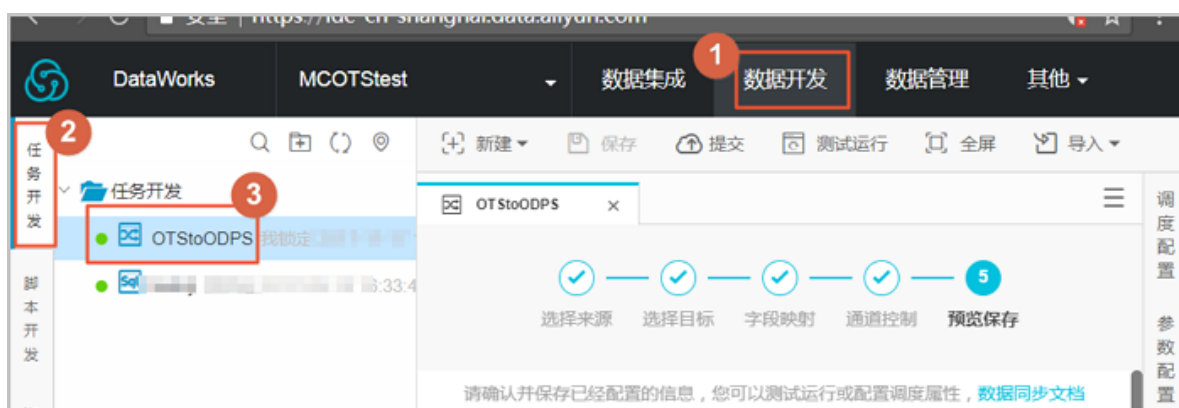
1. 單擊頁面上方的運行，開始執行任務。

如果配置裡面沒有變數，則會立即執行。如果有變數，則會要求填入變數的真實值，單擊確認後開始運行。

2. 運行結束後，在日誌中可以查看任務是否成功和匯出的資料行數。

步驟5. 設定調度參數

1. 單擊資料開發。
2. 在任務開發標籤中，雙擊開啟已建立的任務OTStoODPS。



3. 單擊調度配置，設定調度參數。

按照如下參數配置後，任務會從第二天開始執行。



參數說明如下：

參數	說明
調度狀態	預設不勾選，表示執行。
出錯重試	建議勾選，表示出錯後會重試。

參數	說明
生效日期	建議使用預設值。
調度周期	本樣本中，選擇分鐘。
開始時間	本樣本中，設為00時00分。
調度間隔	本樣本中，選擇5分鐘。
結束時間	本樣本中，設為23時59分。
相依性屬性	在相依性屬性地區，根據業務情況填寫，或者使用預設值。
跨周期依賴	選擇自依賴，等待上一調度周期結束，才能繼續運行。

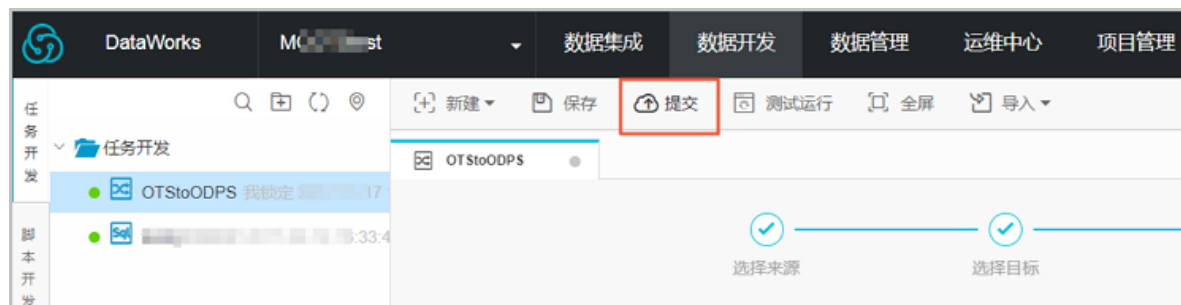
4. 單擊參數配置，配置參數。

參數說明如下：

參數	說明
<code>bdp.system.bizdate</code>	無需配置。
startTime	在調度配置中設定開始時間的變數。在本樣本中，設為 <code>yyyyymmddhh24miss-10/24/60</code> ，表示調度任務開始的時間減去10分鐘。
endTime	在調度配置中設定結束時間的變數。在本樣本中，設為 <code>yyyyymmddhh24miss-5/24/60</code> ，表示調度任務開始的時間減去5分鐘。

步驟6. 提交任務

1. 在頁面上方單擊提交。



2. 在彈出的對話方塊中，單擊確定提交。

提交成功後，顯示當前檔案為唯讀狀態。

步驟7. 查看任務

1. 單擊營運中心。



2. 在左側導覽列中，選擇工作清單 > 周期任務，可以查看建立的任務OTStoODPS。

3. 任務會從第二天的00點00分開始執行。

- 在左側導覽列中，選擇任務營運 > 周期執行個體，查看當天需要啟動並執行周期任務。單擊執行個體名稱，可以查看詳情。
- 當單個任務在運行中或運行結束後可以查看日誌。

步驟8. 查看匯入到MaxCompute中的資料

1. 單擊資料管理。



2. 在左側導覽列，單擊尋找資料，會列出MaxCompute中的所有表。

3. 找到剛剛匯入資料的表（ots_gps_data），單擊後進入該表的詳情頁。

4. 單擊資料預覽頁籤，查看匯入的資料。



2 OSS

2.1 概述

[Table Store](#)是構建在阿里雲飛天分布式系統之上的分布式NoSQL資料存放區服務，根據99.99%的高可用以及11個9的資料可靠性的標準設計。Table Store通過資料分區和負載平衡技術，實現資料規模與訪問並發的無縫擴充，提供海量結構化資料的儲存和即時訪問。

[OSS](#) (Object Storage Service) 是海量、安全、低成本、高可靠的雲端儲存體服務，提供99.99999999%的資料可靠性。使用RESTful API可以在互連網任何位置儲存和訪問，容量和處理能力彈性擴充，多種儲存類型供選擇全面最佳化儲存成本。

使用情境

Table Store：提供專業的資料持久化儲存服務，以及面向使用者的即時高並發低延遲讀寫操作。

OSS：提供極低成本的備份功能。

使用方式

- 寫

直接寫入Table Store。

- 讀

直接讀取Table Store。

- 備份

自動備份。

- 恢複

使用Data Integration (OSSreader + OTSwriter) 重新寫回Table Store

限制

- 整行寫入

使用Table Store Stream功能，要求每次寫入Table Store的資料必須是整行資料。目前類似物聯網資料的時序資料寫入方式都是整行寫入，後續基本無修改。

- 同步延時

目前使用的是周期調度，每隔5分鐘調度一次，並且外掛程式有5分鐘延遲，同步總延遲為5~10分鐘。

開通服務

- 開通Table Store
 1. 登入Table Store#####。
 2. 單擊立即開通。
 3. 在Table Store###建立####和###。



说明：

- 使用增量通道，資料表必須開啟Stream##，有效時間可以選擇24小時。
- Table Store的讀寫CU分為預留CU和按量CU。如果您在建立表時將預留CU設定為0，表示只使用按量CU。在使用過程中，預留CU可以隨時進行修改。
- Table Store為您提供每月10 GB，1000萬按量讀和1000萬按量寫的免費額度。

- 開通OSS
 1. 登入OSS#####。
 2. 單擊立即開通。

資料通道

離線

- 全量匯出到OSS
 - #####
- 增量同步處理到OSS
 - #####
- 全量匯入到Table Store
 - 指令碼模式

2.2 全量匯出 (指令碼模式)

Data Integration (Data Integration) 產品提供資料同步服務，有指令碼模式和嚮導模式兩種方式。指令碼模式更靈活，嚮導模式更簡單。

本章介紹如何將Table Store中的全量資料通過Data Integration的指令碼模式匯出到OSS中，從而使用者可以自由下載檔案自用，或者作為Table Store的備份存於OSS中。

途徑

Data Integration指令碼模式

- Reader : OTSReader
- Writer : OSSWriter

步驟1. 建立Table Store資料來源



说明：

如果已經建立了Table Store的資料來源，可以跳過這一步。

建立資料來源的具體步驟，請參見[##Table Store####](#)。

步驟2. 建立OSS資料來源

本操作與步驟1類似，區別只是選擇OSS作為資料來源。



说明：

配置OSS資料來源的參數時，注意Endpoint不包括bucketName。

步驟3. 建立匯出任務

1. 登入[Data Integration###](#)。
2. 在同步任務頁面，選擇指令碼模式。
3. 在彈出的匯入模板對話方塊中，來源類型選擇OTS，目標類型選擇OSS。
4. 單擊確認，進入配置頁面。

步驟4. 完善配置項

1. 在配置介面，已經提前嵌入了otsreader和osswriter的模板，請參考以下解釋完成配置。

```
{
  "type": "job",      # 不能修改
  "version": "1.0",   # 不能修改
  "configuration": {
    "setting": {
      "errorLimit": {
        "record": "0" # 超過record個錯誤後，匯入任務會失敗。
      },
      "speed": {
        "mbps": "1", # 匯入速率，單位是MB。
        "concurrent": "1" # 並發度。
      }
    },
    "reader": {
      "plugin": "ots", # 不能修改
      "parameter": {
        "datasource": "", # Data Integration中的資料來源名稱，需要提前設定好，這裡有兩種選擇，一種是配置datasource資料來源，一種寫明文的AccessKeyID等鑒權資訊，鼓勵使用資料來源。
        "table": "", # Table Store中的表名
        "column": [ # 需要匯出到OSS的列名，不能設定為空白。
          {
```

```

        "name": "column1"      # Table Store中列名，此列需要匯入到OSS
    },
    {
        "name": "column2"      # Table Store中列名，此列需要匯入到OSS
    }
],
"range": {
    "begin": [
        {
            "type": "INF_MIN"    # Table Store中第一列主鍵的起始位置。如果
            要匯出全量，這裡需要配置為INF_MIN，如何匯出部分，則按需要配置。如果有幾列主鍵
            列，這裡begin中就需要幾項配置。
        }
    ],
    "end": [
        {
            "type": "INF_MAX"    # Table Store中第一列主鍵的結束位置。如果
            要匯出全量，這裡需要配置為INF_MAX，如何匯出部分，則按需要配置。
        }
    ],
    "split": [    # 用來配置Table Store的表的分區資訊，可以加速匯出，下一
    個版本會自動處理。
    ]
}
},
"writer": {
    "plugin": "oss",
    "parameter": {
        "datasource": "",    # 配置OSS的資料來源
        "object": "",    # object 首碼，不需要包括bucket名字，比如：tablestore
        /20171111/，如果是定時匯出，則這裡需要使用變數，比如tablestore/${date}，然後
        在配置調度參數的時候配置${date}的值。
        "writeMode": "truncate",    # 支援truncate|append|nonConflict
        , truncate會清理已存在的同名檔案；append會加到已存在的同名檔案內容後面；
        nonConflict會報錯當同名檔案存在時。全量匯出使用truncate。
        "fileFormat": "csv",    # 支援csv和txt兩種格式
        "encoding": "UTF-8",    # 編碼
        "nullFormat": "null",    # 定義null值的字串標識符方式，也可以是Null 字元
        串
        "dateFormat": "yyyy-MM-dd HH:mm:ss",    # 時間格式
        "fieldDelimiter": ",",    # 每一列的分隔字元
    }
}
}
}

```

2. 單擊儲存，儲存任務。

步驟5. 運行任務

1. 單擊運行，開始執行任務。

如果在配置裡面有變數，比如存在`${date}`，則會出現變數設定介面，只能設定具體值，如下圖所示。



2. 查看頁面下方的日誌。

如果日誌中沒有報錯，則說明執行成功，您可以到目標OSS中去檢查資料了。



说明：

因為全量匯出一般是一次性的，所以不需要配置自動調度參數。如果需要配置調度參數，請參考#####中的調度參數配置。

步驟6. 檢查匯出到OSS中的資料

1. 登入 [OSS###](#)。
2. 選擇相應bucket和檔案名稱，下載後查看內容是否符合預期。

2.3 增量同步處理（指令碼模式）

Data Integration (Data Integration) 產品提供資料同步服務，有嚮導模式和指令碼模式兩種方式。嚮導模式更簡單，指令碼模式更靈活。

本章介紹如何將Table Store中的增量資料通過Data Integration的指令碼模式同步到OpenSearch中。

途徑

Data Integration指令碼模式

- Reader：OTSSStreamReader
- Writer：OSSWriter

配置Table Store

無需配置。

配置OSS

無需配置。

配置Data Integration

1. 建立Table Store資料來源。



说明：

- 如果已經建立了Table Store的資料來源，可以跳過這一步。
- 如果您不希望建立資料來源，也可以在後續的配置頁面中配置相應的endpoint、instanceName、AccessKeyId和AccessKeySecret。

建立資料來源的具體步驟，請參見[##Table Store####](#)。

2. 建立OSS資料來源。

本操作與上一個步驟類似，只是選擇OSS作為資料來源。



说明：

配置OSS資料來源的參數時，注意Endpoint不包括bucketName。

3. 建立同步任務。

- 登入[Data Integration###](#)。
- 在同步任務頁面，選擇指令碼模式。
- 在彈出的匯入模板對話方塊中，來源類型選擇OTS Stream，目標類型選擇OSS。
- 單擊確認，進入配置頁面。

4. 完善配置項。

- 在配置介面，已經提前嵌入了OTSSStreamReader和OSSWriter的模板，請參考以下解釋完成配置。

```
{
  "type": "job",
  "version": "1.0",
  "configuration": {
    "setting": {
      "errorLimit": {
        "record": "0" # 允許出錯的個數，當錯誤超過這個數目的時候同步任務會失敗。
      },
      "speed": {
        "mbps": "1", # 每次同步任務的最大流量。
        "concurrent": "1" # 每次同步任務的並發度。
      }
    },
    "reader": {
      "plugin": "otsstream", # Reader外掛程式的名稱。
      "parameter": {
        "datasource": "", # Table Store的資料來源名稱，如果有此項則不再需要配置endpoint, accessId, accessKey和instanceName。
        "dataTable": "", # TableStore中的表名。
      }
    }
  }
}
```

```

"statusTable": "TableStoreStreamReaderStatusTable", # 儲存
TableStore Stream狀態的表，一般不需要修改。
"startTimestampMillis": "", # 開始匯出的時間點，由於是增量匯出，需要迴
圈啟動此任務，則這裡每次啟動的時候的時間都不一樣，這裡需要設定一個變數，比如${
start_time}。
"endTimestampMillis": "", # 結束匯出的時間點。這裡也需要設定一個變數，
比如${end_time}。
"date": "yyyyMMdd", # 匯出哪一天的資料，功能和startTimestampMillis、
endTimestampMillis重複，這一項需要刪除。
"mode": "single_version_and_update_only", # TableStore Stream匯出
資料的格式，目前需要設定成：single_version_and_update_only。如果配置模板
中沒有則需要增加。
"column": [ # 需要匯出TableStore中的哪些列到OSS中去，如果配置模板中沒有
則需要增加，具體配置個數由使用者自訂設定
{
    "name": "uid" # 列名，這個是Table Store中的主鍵
},
{
    "name": "name" # 列名，這個是Table Store中的屬性列。
},
],
"isExportSequenceInfo": false, # single_version_and_update_only
模式下只能是false。
"maxRetries": 30 # 最大重試次數。
},
},
"writer": {
"plugin": "oss", # Writer外掛程式的名稱
"parameter": {
    "datasource": "", # OSS的資料來源名稱
    "object": "", # 最後備份到OSS的檔案名稱的首碼，建議Table Store執行個體
名/表名/date。比如"instance/table/{date}"
    "writeMode": "truncate", # 支援truncate|append|nonConflict，
truncate會清理已存在的同名檔案；append會加到已存在的同名檔案內容後面；
nonConflict會報錯當同名檔案存在時。
    "fileFormat": "csv", # 檔案類型
    "encoding": "UTF-8", # 編碼類別型
    "nullFormat": "null", # 當遇到控制時，在文本中如何表示
    "dateFormat": "yyyy-MM-dd HH:mm:ss", # 時間格式
    "fieldDelimiter": ",", # 每一列的分隔字元
}
}
}
}
}

```



说明：

詳細的配置項解釋請參見[##OTSStreamReader](#)和[##OSSWriter](#)。

b. 單擊儲存，儲存任務。

5. 運行任務。

a. 單擊頁面上方的運行。

b. 在彈出的配置框中，組態變數參數。

c. 單擊確認後開始運行任務。

d. 運行結束後登入[OSS](#)檢查是否成功備份檔案。

6. 配置調度。

a. 單擊提交。

b. 在彈出的對話方塊中，配置各項參數。

提交

周期属性

- * 调度类型: ☒ 周期调度 ☐ 一次性调度
- * 自动重跑: ☒ 自动重跑 ?
- * 生效日期: 1970-01-01 - 2116-11-22
- * 调度周期: **分钟** 小时 天 周 月
- * 起始时间: 00:00 至 23:59
- * 时间间隔: 5分钟

自定义变量参数 ?

start_time: \${yyyymmddhh24miss-10/24/60}

end_time: \${yyyymmddhh24miss-5/24/60}

date: \${bdp.system.bizdate}

依赖属性

* 添加依赖: test12244 请选择上游任务

项目名称	任务名称	操作

參數說明如下：

參數	描述
調度類型	選擇周期調度。
自動重跑	如果勾選，則當失敗的時候會預設重試3次，每次間隔2分鐘。
生效日期	使用預設值。預設從1970-01-01到一百年後。
調度周期	選擇分鐘。
起始時間	選擇00:00至23:59，表示全天24小時都需要調度。
時間間隔	選擇5分鐘。

參數	描述
start_time	輸入\${yyyyymmddhh24miss-10/24/60}，表示調度時間減去10分鐘。
end_time	輸入\${yyyyymmddhh24miss-5/24/60}，表示調度時間減去5分鐘。
date	輸入\${bdp.system.bizdate}，表示調度日期。
相依性屬性	如果有依賴則填寫，沒有則不用填。
跨周期依賴	選擇自依賴，等待上一調度周期結束，才能繼續運行。

c. 單擊確認。

周期性的同步任務配置完成，當前設定檔顯示為唯讀狀態。

7. 查看任務。

a. 單擊頁面上方的營運中心。

b. 在左側導覽列，選擇工作清單 > 周期任務，可以查看新建立的同步任務。

c. 建立的任務會從第二天00點開始執行。

- 在左側導覽列中，選擇任務營運 > 周期執行個體，查看每一個預建立的當天同步任務，每個任務相隔5分鐘，每個任務處理過去10~5分鐘的資料。
- 單擊執行個體名稱，可以查看詳情。

d. 單個任務在運行中或運行結束後，可以查看日誌。

8. 檢查匯出到OSS中的資料。

登入[OSS###](#)，查看是否產生了新的檔案，檔案內容是否正確。

至此，Table Store資料可以在延遲5~10分鐘的基礎上自動同步到OSS中了。