

File to Hadoop MySQL 設定表操作說明

註冊新設定表

UDF (CAST_OPERATION & MASK_OPERATION)

TABLE_CONFIG

COLUMN_CONFIG

執行順序

修改設定表內容

UDF (CAST_OPERATION & MASK_OPERATION)

TABLE_CONFIG

COLUMN_CONFIG

執行順序

刪除設定表內容

UDF (CAST_OPERATION & MASK_OPERATION)

TABLE_CONFIG

COLUMN_CONFIG

執行順序

Appendix

Oracle DB 與 Hive 資料型別對照表

設定表創建語法

註冊新設定表

UDF (CAST_OPERATION & MASK_OPERATION)

CAST_OPERATION

```
INSERT INTO CAST_OPERATION (CAST_OPERATION_NAME) VALUES ('<new UDF name>');
```

MASK_OPERATION

```
INSERT INTO MASK_OPERATION (MASK_OPERATION_NAME) VALUES ('<new UDF name>');
```



注意事項

1. ID 為自增主鍵值(auto_increment)，新增語法執行時會自動寫入
2. 此二張表僅在新增加 UDF 同時新增

TABLE_CONFIG

```
INSERT INTO `TABLE_CONFIG`
(JS_NAME, TWS_ACTION, HIVE_DATABASE_NAME, HIVE_TABLE_NAME, SOURCE_FILE_PREFIX, SOURCE_
FILE_TYPE, SOURCE_FILE_HEADER, SOURCE_FILE_SEP, SOURCE_FILE_QUOTE, SOURCE_FILE_ENCODING, `OWNER`, LOADING_METHOD, PARTITION_KEY, PARTITION_KEY_TYPE, FETCH_CONDITION, FETCH_
CONDITION_PARAMS, DROP_CONDITION, DROP_CONDITION_PARAMS, `ENABLE`, CREATED_AT, UPDATED_
_AT)
VALUES
('ACTLNKCUSTBIND_D', NULL, 'ODS_BANK', 'ACTLNKCUSTBIND', 'ACTLNKCUSTBIND', 'delimited',
0, '|', NULL, 'BIG5', 'ODS_BANK', 'truncate_insert', 'SNAP_DATE', 'STRING', "snap_date=to_date
('%s', 'yyyyMMdd')", 'snap_date', NULL, NULL, 1, '2022-07-12 12:12:12', now())
```

Column Name	Description
JS_NAME	對應的 TWS JS name
TWS_ACTION	填入 ALT (alter) 或 SUB (submit)，若無則填 NULL
HIVE_DATABASE_NAME	導入的 Hive database name，目前預設與 Oracle DB name 相同 (e.g. ODS_BANK)
HIVE_TABLE_NAME	導入的 Hive table name，與導入表名相同 (e.g. BS01)
SOURCE_FILE_PREFIX	檔案名稱前綴 (e.g. BS01 .20220101)
SOURCE_FILE_TYPE	導入的檔案格式，僅能填寫 delimited (以符號分隔)、 fixed_width (以欄位長度分隔)與 json
SOURCE_FILE_HEADER	檔案是否內含標題列，若有填 1 ，若無則填寫 0
SOURCE_FILE_SEP	檔案所使用的分隔符號，若為 delimited 以外格式檔案則填寫 NULL
SOURCE_FILE_QUOTE	Quotation
SOURCE_FILE_ENCODING	檔案的編碼格式，僅能填入 UTF_8 、 BIG5 與 BIG5_HKSCS
OWNER	導入的 table owner，目前填入值與 HIVE_DATABASE_NAME 相同
LOADING_METHOD	導入方式，僅能填寫 truncate_insert 與 delete_insert
PARTITION_KEY	Hive table 的 partition key，目前設定皆為 SNAP_DATE
PARTITION_KEY_TYPE	partition key 的資料型別，僅能填入 Hive 有的資料型別且必須為大寫 (e.g. STRING)

Column Name	Description
FETCH_CONDITION	delete insert 時使用的擷取條件，目前與 <code>DROP_CONDITION</code> 相同，填入 <code>snap_date=to_date('%s', 'yyyyMMdd')</code> 即可
FETCH_CONDITION_PARAMS	delete insert 時使用的擷取參數，目前填入 <code>snap_date</code> 即可
DROP_CONDITION	刪除 partition 時使用的條件，填入 <code>snap_date=to_date('%s', 'yyyyMMdd')</code>
DROP_CONDITION_PARAMS	刪除 partition 時使用的參數，填入 <code>snap_date</code>
ENABLE	此表是否啟用， <code>1</code> 為啟用、 <code>0</code> 為停用
CREATED_AT	此筆資料建立時間
UPDATED_AT	此筆資料更新時間



ID 為自增主鍵值(auto_increment)，新增語法執行時會自動寫入

COLUMN_CONFIG

```
-- TABLE_ID = 2 ODS_BANK.ACTLNKCUSTBIND

INSERT INTO `COLUMN_CONFIG` (TABLE_ID, SOURCE_COLUMN_NAME, SOURCE_COLUMN_WIDTH, SOURCE_COLUMN_SEQUENCE, HIVE_COLUMN_NAME, HIVE_COLUMN_TYPE, HIVE_COLUMN_SEQ, CAST_OPERATION_ID, CAST_OPERATION_PARAMS, MASK_OPERATION_ID, CREATED_AT, UPDATED_AT)
VALUES (2, NULL, NULL, NULL, 'SNAP_DATE', 'STRING', 1, 12, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'CHANNELCODE', NULL, 1, 'CHANNELCODE', 'STRING', 2, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'CUSTEPACCOUNT', NULL, 2, 'CUSTEPACCOUNT', 'STRING', 3, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'CUSTID', NULL, 3, 'CUSTID', 'STRING', 4, 22, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'DUP', NULL, 4, 'DUP', 'STRING', 5, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'UUID', NULL, 5, 'UUID', 'STRING', 6, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'NAMECODE', NULL, 6, 'NAMECODE', 'STRING', 7, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'ACTIONTYPE', NULL, 7, 'ACTIONTYPE', 'STRING', 8, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'BANK', NULL, 8, 'BANK', 'STRING', 9, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'ACCOUNT', NULL, 9, 'ACCOUNT', 'STRING', 10, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'TXID', NULL, 10, 'TXID', 'STRING', 11, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'TXTIME', NULL, 11, 'TXTIME', 'STRING', 12, 3, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'CREATETIME', NULL, 12, 'CREATETIME', 'STRING', 13, 3, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, 'MODIFYTIME', NULL, 13, 'MODIFYTIME', 'STRING', 14, 3, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, NULL, NULL, NULL, 'TRANS_SN', 'STRING', 15, NULL, NULL, NULL, '2021-03-16 16:00:00', now())
, (2, NULL, NULL, NULL, 'EXEC_DATE', 'STRING', 16, 8, NULL, NULL, '2021-03-16 16:00:00', now());
```

Column Name	Description
TABLE_ID	<code>TABLE_CONFIG</code> 註冊 table 的 ID，為 <code>TABLE_CONFIG.ID</code> 之 foreign key
SOURCE_COLUMN_NAME	來源資料欄位的名稱
SOURCE_COLUMN_WIDTH	來源資料欄位的長度，若檔案格式非 <code>fixed_width</code> 則填 <code>NULL</code>
SOURCE_COLUMN_SEQUENCE	來源資料欄位的順序，從1開始
HIVE_COLUMN_NAME	Hive table 的欄位名稱
HIVE_COLUMN_TYPE	Hive table 的欄位資料型別 (相關對照見附件)
HIVE_COLUMN_SEQ	Hive table 的欄位順序，從1開始 欄位順序可能跟 <code>SOURCE_COLUMN_SEQUENCE</code> 不同，需進行盤點
CAST_OPERATION_ID	欲套用的資料轉換 UDF 的 ID，為 <code>CAST_OOPERATION.ID</code> 之 foreign key
CAST_OPERATION_PARAMS	欲套用的資料轉換 UDF 之參數，以下幾支 UDF 使用時須從命令行輸入參數 <code>execDateUdf</code> <code>snapDateUdf</code> <code>subStringUdf</code> <code>SubStringWithInputColumnAndParameterUdf</code> <code>toIntegerUdf</code> <code>trimPKUdf</code> <code>trimUdf</code>
MASK_OPERATION_ID	欲套用的資料遮罩 UDF 的 ID，為 <code>MASK_OOPERATION.ID</code> 之 foreign key
CREATED_AT	此筆資料建立時間
UPDATED_AT	此筆資料更新時間



ID 為自增主鍵值(auto_increment)，新增語法執行時會自動寫入。

執行順序

```

MASK_OPERATION ↗
                |————> TABLE_CONFIG ———> COLUMN_CONFIG
CAST_OPERATION ↘

```

修改設定表內容

UDF (CAST_OPERATION & MASK_OPERATION)

CAST_OPERATION

```
UPDATE CAST_OPERATION SET CAST_OPERATION_NAME = 'newName' WHERE ID = '1';
```

MASK_OPERATION

```
UPDATE MASK_OPERATION SET MASK_OPERATION_NAME = 'newName' WHERE ID = '5';
```

TABLE_CONFIG

```
-- example 1 (where condition = ID)
UPDATE TABLE_CONFIG SET SOURCE_FILE_PREFIX = 'TEST' WHERE ID = 2;

-- example 2 (where condition = HIVE_TABLE_NAME)
UPDATE TABLE_CONFIG SET SOURCE_FILE_SEP = ',' WHERE HIVE_TABLE_NAME = 'ACTLNKCUSTBIN
D';
```

COLUMN_CONFIG

```
-- example 1 (where condition = ID)
UPDATE COLUMN_CONFIG SET HIVE_COLUMN_NAME = 'NEWNAME' WHERE ID = 1;

-- example 2 (where condition = SOURCE_COLUMN_NAME)
UPDATE COLUMN_CONFIG SET CAST_OPERATION_ID = 22 WHERE SOURCE_COLUMN_NAME = 'GOODS_CODE';
```

執行順序

MASK_OPERATION ↗

(僅能修改名稱)

CAST_OPERATION ↘

(僅能修改名稱)

————> TABLE_CONFIG ———> COLUMN_CONFIG

(除 ID 外皆可修改，需
注意 enum 類型欄位)

(需注意 `HIVE_COLUMN_TYPE`
為 enum 類型欄位)

刪除設定表內容



除非必要，否則不建議執行刪除動作

UDF (CAST_OPERATION & MASK_OPERATION)

CAST_OPERATION

```
DELETE FROM CAST_OPERATION WHERE ID = '20';
```

MASK_OPERATION

```
DELETE FROM MASK_OPERATION WHERE ID = '3';
```

TABLE_CONFIG

```
-- example (where condition = ID)
DELETE FROM TABLE_CONFIG WHERE ID = 5;
```

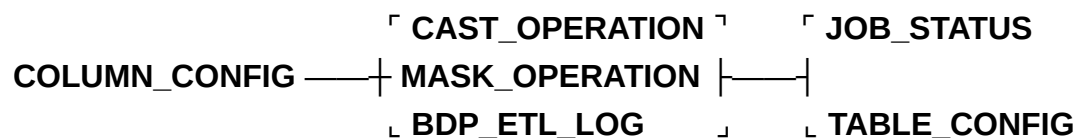
COLUMN_CONFIG

```
-- example (where condition = ID)
DELETE FROM COLUMN_CONFIG WHERE ID = 200;
```

執行順序



刪除時必須按下方順序，否則會由於表格間的鍵值依賴而無法刪除



Appendix

Oracle DB 與 Hive 資料型別對照表

Oracle	Hive
--------	------

Oracle	Hive
VARCHAR2	STRING
DATE	STRING (因應可能使用 Impala 之情形，因此轉為 STRING)
NUMBER	INT
NUMBER(10,2)	DECIMAL

設定表創建語法



僅在資料庫內尚無相關設定表時使用

```
CREATE DATABASE IF NOT EXISTS BDP_PIPE;
USE BDP_PIPE;

-- TABLE_C. NFIG
CREATE TABLE IF NOT EXISTS BDP_PIPE.TABLE_CONFIG
(
    `ID` BIGINT NOT NULL AUTO_INCREMENT COMMENT 'ID',
    `JS_NAME` VARCHAR(30) COMMENT '對應的 TWS JS Name',
    `TWS_ACTION` VARCHAR(30) COMMENT 'ALT: Alter, SUB: Submit',
    `HIVE_DATABASE_NAME` VARCHAR(50) NOT NULL COMMENT '導入的 hive database',
    `HIVE_TABLE_NAME` VARCHAR(30) NOT NULL COMMENT '導入的 table name',
    `SOURCE_FILE_PREFIX` VARCHAR(100) NOT NULL COMMENT '在 FTP 的檔案 prefix',
    `SOURCE_FILE_TYPE` ENUM ('json', 'fixed_width', 'delimited') NOT NULL COMMENT '原始資料的格式',
    `SOURCE_FILE_HEADER` INT COMMENT 'header 佔了幾行, 1= skip 1 row, 0 = no header',
    `SOURCE_FILE_SEP` VARCHAR(10) COMMENT '分隔符號',
    `SOURCE_FILE_QUOTE` VARCHAR(10) COMMENT 'Quotation',
    `SOURCE_FILE_ENCODING` ENUM ('UTF_8', 'BIG5', 'BIG5_HKSCS') NOT NULL COMMENT '原始資料的 encoding',
    `OWNER` VARCHAR(50) NOT NULL COMMENT 'Table Owner',
    `LOADING_METHOD` ENUM ('truncate_insert', 'delete_insert') NOT NULL COMMENT '導入方式',
    `PARTITION_KEY` VARCHAR(20) NOT NULL COMMENT 'Hive 的 Partition Key',
    `PARTITION_KEY_TYPE` ENUM ('TINYINT', 'SMALLINT', 'INT', 'BIGINT', 'VARCHAR', 'CHAR', 'TIMESTAMP', 'DATE', 'STRING', 'BOOLEAN', 'BINARY', 'DECIMAL') NOT NULL COMMENT 'Partition Key 的型別',
    `FETCH_CONDITION` VARCHAR(200) COMMENT '資料擷取條件 for delete insert',
    `FETCH_CONDITION_PARAMS` VARCHAR(200) COMMENT '資料擷取參數 for delete_insert',
)
```

```

        `DROP_CONDITION`          VARCHAR(200) COMMENT 'drop partition',
        `DROP_CONDITION_PARAMS`    VARCHAR(200) COMMENT 'drop partition parameters',
        `ENABLE`                   BOOLEAN COMMENT '記錄此 Table 是否啟用',
        `CREATED_AT`               TIMESTAMP COMMENT '建立時間',
        `UPDATED_AT`               TIMESTAMP                                NOT NULL DEFAULT
T CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP COMMENT '更新時間',
        CONSTRAINT PK_TABLE_CONFIG PRIMARY KEY (ID)
    );

-- TABLE_CONFIG_INDEX
CREATE INDEX TABLE_CONFIG_INDEX ON TABLE_CONFIG (HIVE_DATABASE_NAME, HIVE_TABLE_NAME);

-- JOB_STATUS
CREATE TABLE IF NOT EXISTS BDP_PIPE.JOB_STATUS
(
    `ID`                BIGINT      NOT NULL AUTO_INCREMENT COMMENT 'ID',
    `TABLE_ID`          BIGINT COMMENT 'Table ID',
    `SNAP_DATE`         DATE COMMENT '資料日, 紀錄排程系統批次進入的批次時間',
    `SCHEDULE_DATE`     DATE COMMENT 'TWS 排程日',
    `EXECUTION_STATUS`  INT COMMENT '以此欄位監控批次執行狀態 0:success, 1: ingest executin
g, -1: execution failed',
    `ROW_COUNT`         INT COMMENT 'ETL 筆數',
    `TRANS_SN`          VARCHAR(20) COMMENT 'TWS trans_sn',
    `JS_NAME`           VARCHAR(20) COMMENT 'TWS Job Stream Name',
    `CREATED_AT`        TIMESTAMP COMMENT '建立時間',
    `UPDATED_AT`        TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_
TIMESTAMP COMMENT '更新時間',
    CONSTRAINT PK_JOB_STATUS PRIMARY KEY (ID),
    CONSTRAINT FK_JOB_STATUS FOREIGN KEY (TABLE_ID) REFERENCES TABLE_CONFIG (ID)
);

-- BDP_ETL_LOG
CREATE TABLE IF NOT EXISTS BDP_PIPE.BDP_ETL_LOG
(
    `ID`                BIGINT      NOT NULL AUTO_INCREMENT COMMENT 'ID',
    `JOB_ID`            BIGINT COMMENT 'TWS Job ID',
    `START_AT`          TIMESTAMP COMMENT '開始時間',
    `END_AT`            TIMESTAMP COMMENT '結束時間',
    `STAGE`             VARCHAR(20) COMMENT '階段',
    `STATUS`            ENUM ('success', 'running', 'failure') COMMENT '狀態',
    `CREATED_AT`        TIMESTAMP COMMENT '建立時間',
    `UPDATED_AT`        TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMEST
AMP COMMENT '更新時間',
    CONSTRAINT PK_BDP_ETL_LOG PRIMARY KEY (ID),
    CONSTRAINT FK_JOB_ID FOREIGN KEY (JOB_ID) REFERENCES JOB_STATUS (ID)
);

-- CAST_OPERATION
CREATE TABLE IF NOT EXISTS BDP_PIPE.CAST_OPERATION
(
    `ID`                BIGINT      NOT NULL AUTO_INCREMENT COMMENT 'ID',
    `CAST_OPERATION_NAME` VARCHAR(100) NOT NULL,
    CONSTRAINT PK_CAST_OPERATION PRIMARY KEY (ID)
);

-- MASK_OPERATION
CREATE TABLE IF NOT EXISTS BDP_PIPE.MASK_OPERATION
(

```



```

        `ID` BIGINT NOT NULL AUTO_INCREMENT COMMENT 'ID',
        `MASK_OPERATION_NAME` VARCHAR(100) NOT NULL,
        CONSTRAINT PK_MASK_OPERATION PRIMARY KEY (ID)
    );

-- COLUMN_CONFIG
CREATE TABLE IF NOT EXISTS BDP_PIPE.COLUMN_CONFIG
(
    `ID` BIGINT NOT NULL AUTO_INCREMENT COMMENT 'ID',
    `TABLE_ID` BIGINT NOT NULL COMMENT 'TABLE_CONFIG 之 ID',
    `SOURCE_COLUMN_NAME` VARCHAR(50) COMMENT '欄位名稱',
    `SOURCE_COLUMN_WIDTH` INT COMMENT '僅用在 table.source_file_type = fixed_width
時',
    `SOURCE_COLUMN_SEQUENCE` INT COMMENT '原始資料順序',
    `HIVE_COLUMN_NAME` VARCHAR(100) COMMENT 'Hive table 的欄位名稱',
    `HIVE_COLUMN_TYPE` ENUM ('TINYINT', 'SMALLINT', 'INT', 'BIGINT', 'VARCHAR',
'CHAR',
'TIMESTAMP', 'DATE', 'STRING', 'BOOLEAN', 'BINARY', 'DECIMAL') COMMENT 'Hive t
able 的欄位類型',
    `HIVE_COLUMN_SEQUENCE` INT COMMENT 'Hive table 的欄位順序',
    `CAST_OPERATION_ID` BIGINT COMMENT '欄位轉出規則',
    `CAST_OPERATION_PARAMS` VARCHAR(20) COMMENT '給 CastUDF 的參數',
    `MASK_OPERATION_ID` BIGINT COMMENT '欄位遮罩規則',
    `CREATED_AT` TIMESTAMP COMMENT '建立時間',
    `UPDATED_AT` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CU
RRENT_TIMESTAMP COMMENT '更新時間',
    CONSTRAINT PK_COLUMN_CONFIG PRIMARY KEY (ID),
    CONSTRAINT FK_COLUMN_CONFIG FOREIGN KEY (TABLE_ID) REFERENCES TABLE_CONFIG (ID),
    CONSTRAINT FK_CAST_OPERATION FOREIGN KEY (CAST_OPERATION_ID) REFERENCES CAST_OPERA
TION (ID),
    CONSTRAINT FK_MASK_OPERATION FOREIGN KEY (MASK_OPERATION_ID) REFERENCES M
ASK_OPERATION (ID)
);

-- COLUMN_CONFIG_INDEX
CREATE INDEX COLUMN_CONFIG_INDEX ON COLUMN_CONFIG (TABLE_ID, HIVE_COLUMN_NAME);

```