# BI Generate Notifications CSV file

Version: 1.3

**Change History**

| Version | Changes Summary | Author |
| --- | --- | --- |
| 0.6 | Modified the following sections:  1.7 Get generic Draw data  1.8 Create Notifications file in CSV format | Paul Cleary |
| 0.7 | Added Notification filtering report (1.4), changed filename definition | Paul Cleary |
| 1.0 | Added daily generation of CSV files and backup requirement for multiple generations of CSV file. Created final copy | Paul Cleary |
| 1.1 | Added extract checks for wallet balances and spend limits  Adjusted file name definition to include jackpot amounts of type $8.7M | Paul Cleary |
| 1.2 | Removed check account balance greater than $999 and revised order in which checks are to be done. Added new checks for users who exceed their topup limits in a month.  Added rules that monthly spending limit checks are not required on 1st day of each month | Paul Cleary |
| 1.3 | Corrected data column name for jackpot threshold, removed wrong format (integer) from Jackpot amount in section 1.7 and updated it to new format as requested by Chandra | Paul Cleary |

# Overview

This chapter describes the process for creating a CSV file containing data for all the notifications to be sent out via SMS to players for a Lotto game draw.

This CSV file will be uploaded to the MULE integration layer which will then process it and pass it across to the SMS transmission system

It is expected that the number of notifications will be small to begin with, and increase to a maximum of around 10,000 active notifications per draw.

# General Notes

The CSV file will be generated each day and not just on draw days. This is because occasionally the overnight BI warehouse Extract-Transform-Load process has problems and may not generate a CSV file in time to be sent out on the draw day – in this situation, the approved fallback is to use the CSV file from the previous day.

The MULE integration layer will carry out the following processing once it receives the CSV file:

* Simple data integrity checking of the CSV file contents
* Apply SMS message size constraints to the Dip name or Favourite name, and to the Player name
* Embed the notifications data into the SMS message text

A permanent extract of the data in the CSV file will be stored by MULE in the campaign\_entry table in the P2P database. This table supplies date for the MULE processing and for P2P reporting.

The Push 2 Play (P2P) database tables and their columns are described in the P2P design document “Push2Play Detailed Design” – the latest version can be obtained from Ferdinand Contreras at Lotto NZ.

**To the developer**: The process flow in this chapter is intended to be “descriptive” rather than “prescriptive”. If you have a more suitable approach to implementing these requirements, then feel free to use it so long as it will deliver the same behaviours and data results.

# Timeline for Lotto Game Draw day

For the first release of the Push2Play system, the start day for the process will always be restricted to the same date as the next impending Lotto draw.

The timeline below shows an example as to the positioning of the “BI Generate Notifications CSV file” process relative to other notifications events on the draw day.



Changes to a player’s notifications on the Lotto Game Draw Day will not be reflected in the notifications sent out on that day. This is a known risk that has been accepted by Lotto NZ   
(as confirmed with Mark Lee)

# Extract list of “active” players from Opted\_game table in P2P database

**Business Rules**

A Player’s notification is selected if their notifications data satisfies the following rules:

* opted\_game.status = 1 (Active)
* opted\_game.jackpot\_threshold >= BI warehouse.current jackpot value for the next Lotto draw

A Player will have either a Dip type or a Favourite – they cannot have both

If the Player has a Dip then the Dip name must be populated

**Input Data**

P2P database tables: opted\_game, dip\_type

BI warehouse: … unknown…

**Output Data: Notifications list**

| Data Element | Description | Source |
| --- | --- | --- |
| Notification Id | The unique id for a P2P notification | Opted\_game.id |
| Player User Id | ESI user id | opted\_game.user\_id |
| Mobile number |  | Opted\_game.mobile\_number |
| Dip type | ESI dip numeric code | Opted\_game.dip\_type |
| Dip name | Alphabetic name of dip code | Dip\_type.dip\_type |
| Favourite name |  | Opted\_game.favourite\_name |
| Wager amount | Currency value of Favourite or Dip in cents | Opted\_game.wager\_amount |

**Process report:**

Record the process name along with the start date and time

# Filter out “ineligible” players from Notifications list

**Business Rules**

A Player’s notification is removed from the list if ANY of the following criteria are met:

1. The Player has the Lotto game blocked for a date greater than or equal to today’s date
2. The Player has a status of “Login suspended”
3. The Player has a status of “Wallet suspended”
4. The Player has a status of “Email change pending”
5. The Player has reached their spending limit for the current week
6. The Player has reached their spending limit for the current month  
   (**Note**: this check is NOT to be run when the process is executing on the 1st day of a new month)
7. The sum of the Player’s current weekly spend and their wager amount will exceed their weekly spending limit
8. The sum of the Player’s current monthly spend and their wager amount will exceed their monthly spending limit  
   (**Note**: this check is NOT to be run when the process is executing on the 1st day of a new month)
9. The Player has insufficient funds and if they were to try to topup, they would exceed the $ value limit for topups in a month  
   (**Note**: this check is NOT to be run when the process is executing on the 1st day of a new month
10. The Player has insufficient funds and if they were to try to topup, they would exceed the limit for the number of topups in a month  
    (**Note**: this check is NOT to be run when the process is executing on the 1st day of a new month
11. The Player is not a “fully registered” myLotto user

Notes:

* The check for these criteria must be executed as per the numbered order above
* The process must check all criteria regardless of whether they are met or not

**Input Data**

Notifications list

BI warehouse: … unknown…

**Output Data: Notifications list**

| Data Element | Description | Source |
| --- | --- | --- |
| Notification Id | The unique id for a P2P notification | Opted\_game.id |
| Player User Id | ESI user id | opted\_game.user\_id |
| Mobile number |  | Opted\_game.mobile\_number |
| Dip type | ESI dip code | Opted\_game.dip\_type |
| Dip name | Alphabetic name of dip code | Dip\_type.dip\_type |
| Favourite name |  | Opted\_game.favourite\_name |
| Wager amount | Currency value of Favourite or Dip in cents | Opted\_game.wager\_amount |

**Notification filtering report:**

This report is required for the Lotto NZ Help desk to answers questions about why a user’s Notification was not sent out on a specific draw date.

The report will contain the following data elements

* Player user id (= ESi user id)
* Mobile phone number
* Date of process run
* Reason for removing user from the Notifications list

Note that this report may require a summary section at the end which contains the following:

* List of removal reasons used, with number of rejections per reason, for this instance of the process execution

# Enrich players in Notifications list with additional detail from their myLotto account

**Business rules**

* None applicable

**Input Data**

* Notifications list
* BI warehouse: … unknown…

**Output Data: Notifications list**

| Data Element | Description | Source |
| --- | --- | --- |
| Notification Id | The unique id for a P2P notification | Opted\_game.id |
| Player User Id | ESI user id | opted\_game.user\_id |
| Player name | Player first name | BI warehouse |
| Mobile number |  | Opted\_game.mobile\_number |
| Dip type | ESI dip code | Opted\_game.dip\_type |
| Dip name | Alphabetic name of dip code | Dip\_type.dip\_type |
| Favourite name |  | Opted\_game.favourite\_name |
| Wager amount | Currency value of Favourite or Dip in cents | Opted\_game.wager\_amount |

# Get generic Draw data for Notifications filename

This data is retrieved and used to create the filename of the Notification CSV file

**Business rules**

* None applicable

**Input Data**

* BI warehouse: … unknown…

**Output Data:**

| Data Element | Description | Source |
| --- | --- | --- |
| Draw Game prefix | Short form of name for Game | Always set to ‘LottoPB’ for the initial release |
| Draw date | Date of next Lotto draw | BI warehouse  Format is: YYYYMMDD |
| Draw number | Identifier of next Lotto draw | BI warehouse  Format is: nnnn (no leading zeros) |
| Jackpot amount | Current jackpot value for the next Lotto draw | BI warehouse  Format is: nnnnn  e.g. 12,867,543 would be 12900  4,567,899 would be 4600  4,345,000 would be 4000 |
| Draw cutoff date and time | Purchase cutoff date time for SMS message purchase requests. It includes the draw day date | Format is:  YYYYMMDD-HHMM  YYYYMMDD part is set to the draw date.  HHMM part is hardcoded as “1930” |

*Note: Campaign id was removed as assumed this is not relevant as it is not available until after the Notifications CSV file upload to MULE*

# Create Notifications file in CSV format

**Business rules**

CSV file column separator is a comma (,)

Values in Character-format columns must have “ at start and end, e.g. “LuckyDip”

The End of Line is indicated by either standard Windows or Unix EOL characters – *developer to decide which*

The name of the CSV file is formatted as follows:

LottoPB\_<*draw number*>\_<*draw date*>\_<*jackpot amount*>\_<*cutoff date and time*>.CSV

Where the values in the < …> are as per section “1.7 Get generic Draw data for Notifications filename” above

Example: Lotto draw 1532 with jackpot of $15.8 million on draw date Sat 16 April 2016 at 7:30PM would result in a CSV filename of:

LottoPB\_1532\_20160416\_15M\_20160416-1930.csv

Lotto NZ IT have requested that a “trigger CSV” file also be generated to indicate that the process has finished and the CSV file is ready for pickup.

**Input Data**

* Notifications list

**Output data: Notifications CSV file**

| Order | CSV Column Name | Format | Source in Notification List |
| --- | --- | --- | --- |
| 1 | Mobile\_phone | Char | Mobile number |
| 2 | First\_name | Char | Player name |
| 3 | Wager\_amount | Numeric - integer | Wager\_amount |
| 4 | Dip\_name | Char | Dip name (may be blank) |
| 5 | Favourite\_name | Char | Favourite name (may be blank) |
| 6 | Opted\_game\_id | Char | Notification Id |

**Process Report:**

Record the process name with the end date and time

# Backup Notifications CSV file for multiple generations

If there is a problem with the quality of the data in the CSV file, then Lotto IT Operations need to be able to inspect recent CSV files to help diagnose the cause of the problem. Hence copies of the CSV file as sent to the Send SMS Messages process in MULE are to be retained in a specific location to meet this requirement.

**Business rules**

* 8 generations of the Notifications CSV file are to be retained

**Input Data**

* Notifications CSV file

**Output Data:**

* Renamed copy of the Notifications CSV file with the date of generation appended to the file name