

UNIVERSITY OF COIMBRA

DEPARTMENT OF INFORMATICS ENGINEERING

SYSTEMS INTEGRATION

Report for Project 3

Author:
António LIMA
2011166926

Author:
Pedro JANEIRO
2012143629

December 22, 2015

Contents

1	Introduction	2
2	Subscriptions	2
2.1	Subscribe	2
2.1.1	Web Subscription	2
2.1.2	SOAP Subscription	2
2.1.3	Shared Behaviour	2
2.2	Subscription Validation	3
2.3	Unsubscribe	4
2.3.1	Web Unsubscription	4
2.3.2	SOAP Unsubscription	5
2.3.3	Shared Behaviour	5
3	Adding Smartphones	5
3.1	Files Directory	6
3.2	SOAP	7
3.3	Shared Behaviour	7
4	Statistics	8
4.1	Database Handling	8
4.2	Tweeting	9
5	Sending Scheduled Emails	10
6	Error Handling/Logging	10

1 Introduction

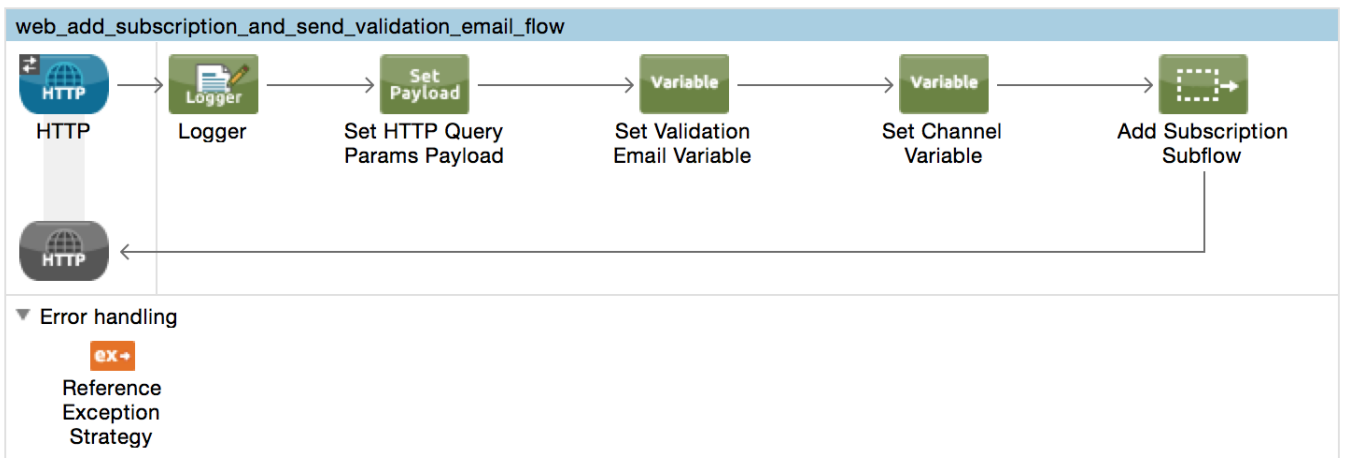
This assignment focused on orchestrating several information services with Mule Flows and AnypointStudio. These services include managing a periodic mailing list and its subscribers, tweeting about the system's statistics and processing new and relevant smartphones.

2 Subscriptions

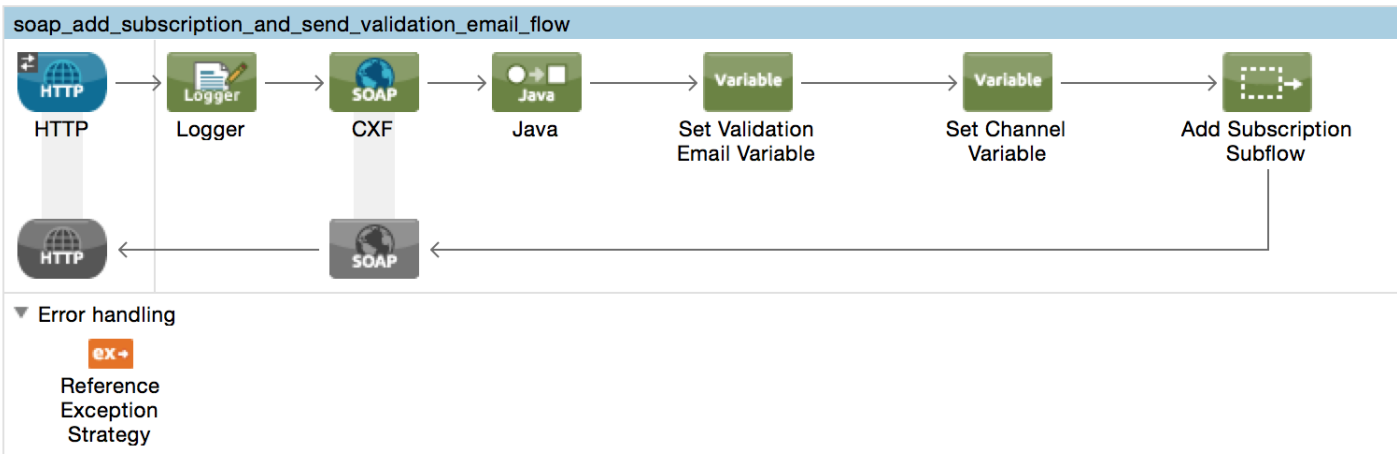
2.1 Subscribe

Subscriptions can be performed either via **Web** or **SOAP**.

2.1.1 Web Subscription

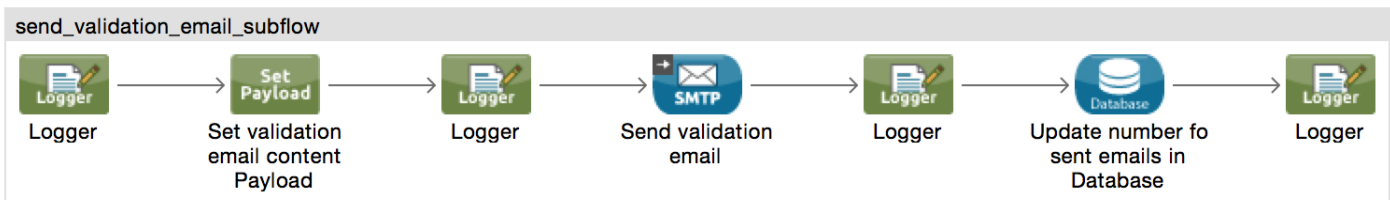
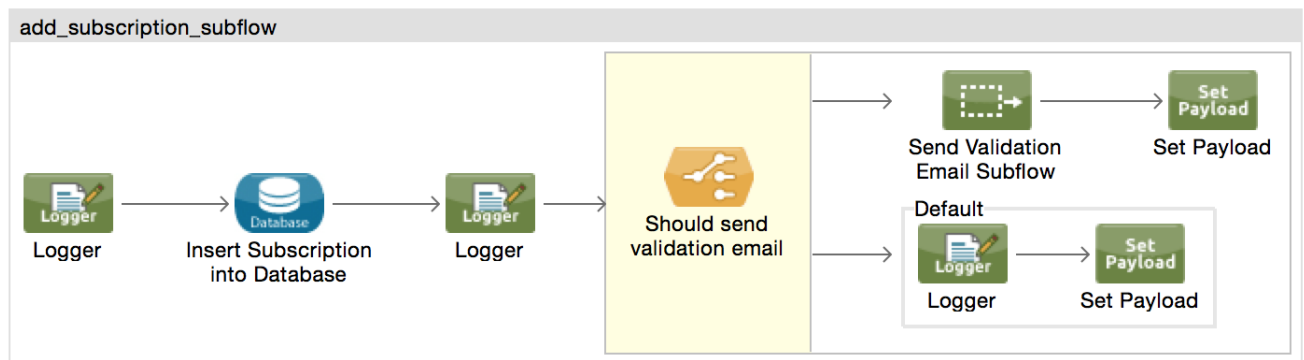


2.1.2 SOAP Subscription



2.1.3 Shared Behaviour

Both the SOAP and WEB flows reference the same subflow (`add_subscription_subflow`) thus their only concern is that of passing a like structure (a `HashMap` in this case).

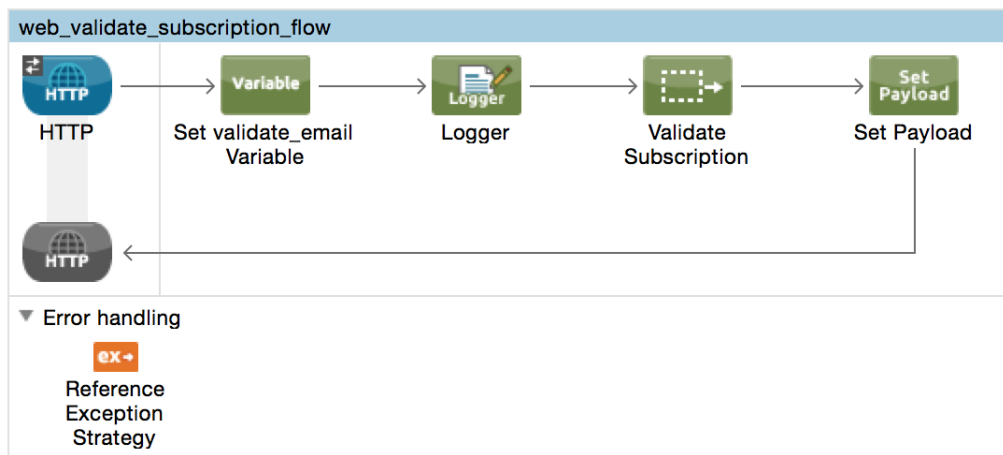


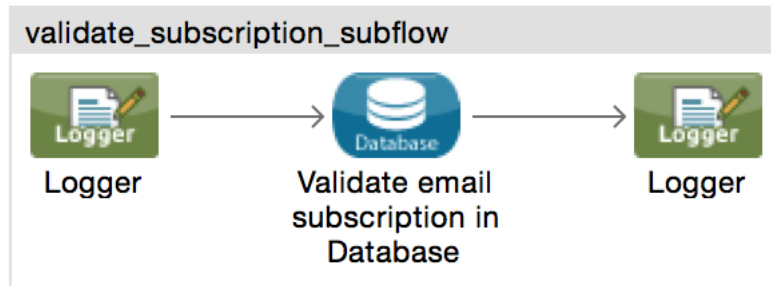
A subscription's details are processed by the server and, if there isn't already a subscription for the specified email address, a confirmation email will be sent to it containing a link to validate the user's subscription.

Example of a confirmation email:

1 Validate your subscription here: http://localhost:8081/smartphones_mailinglist/validate_subscription/?email=aclspam@hotmail.com

2.2 Subscription Validation



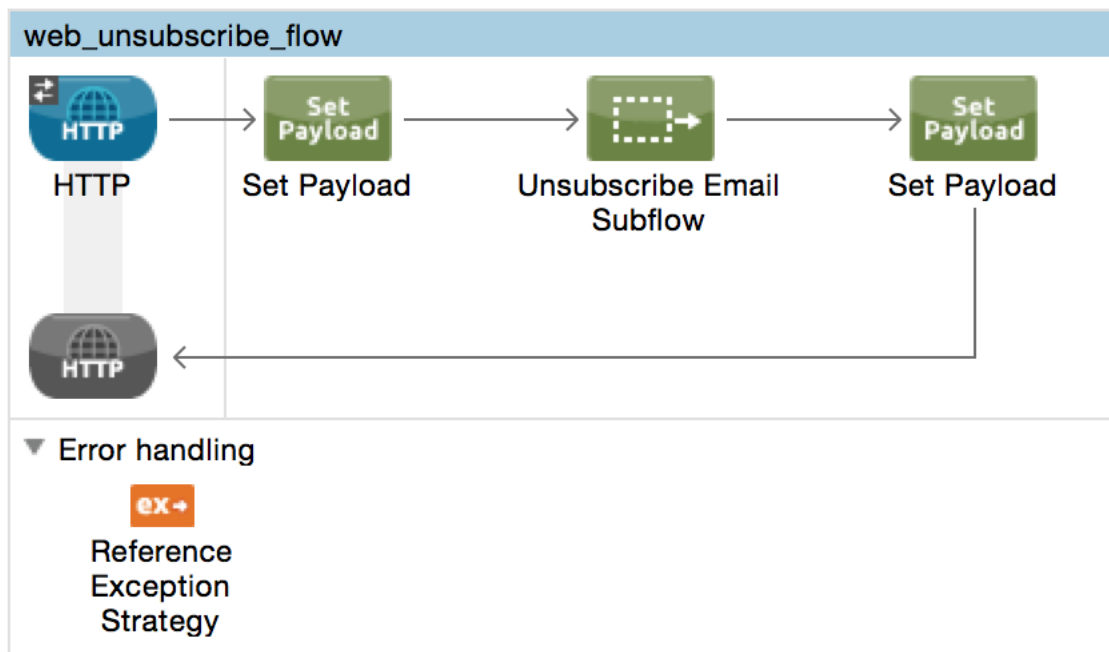


Upon clicking the validation Web link, the subscription is marked as having been validated.

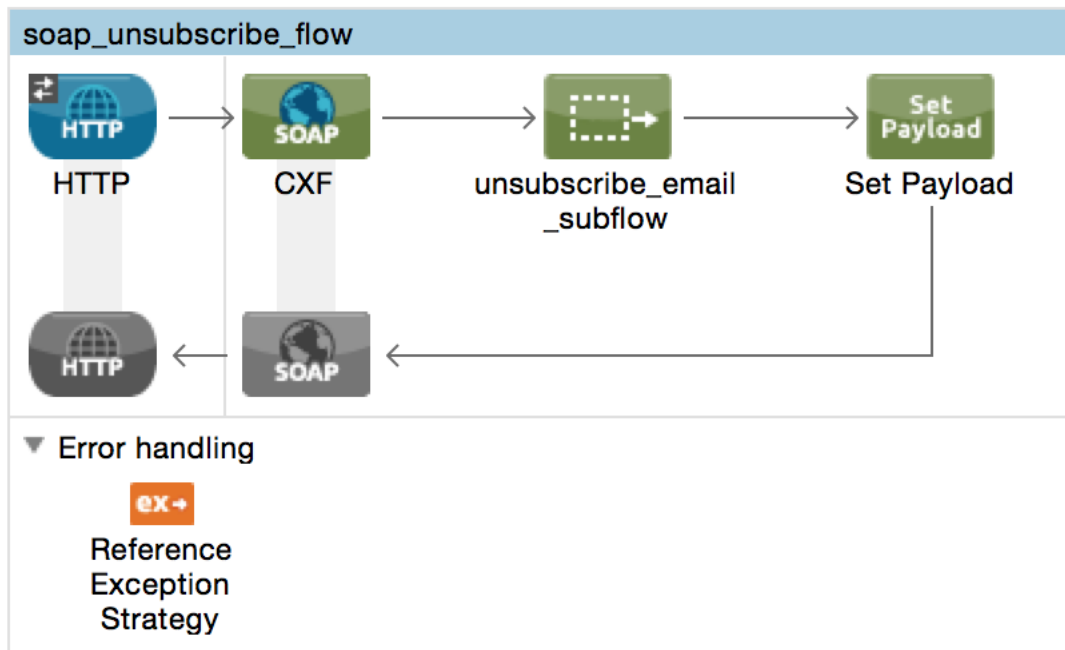
2.3 Unsubscribe

Unsubscription can be performed via either Web or SOAP.

2.3.1 Web Unsubscription

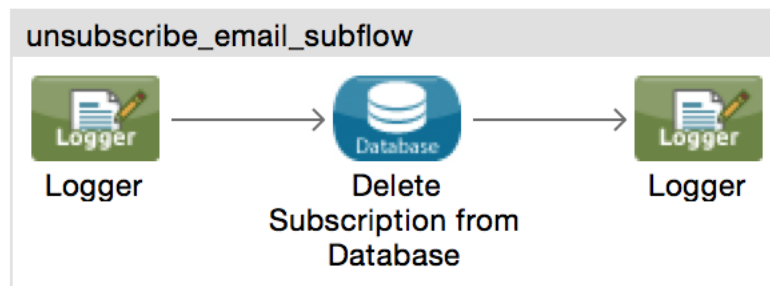


2.3.2 SOAP Unsubscription



TODO

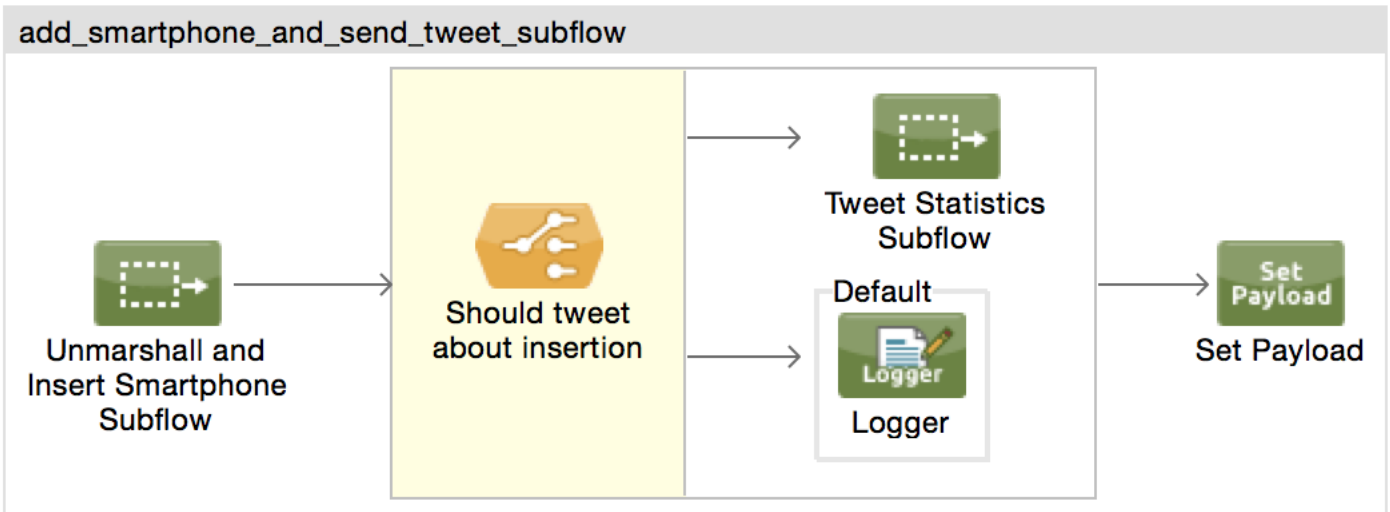
2.3.3 Shared Behaviour



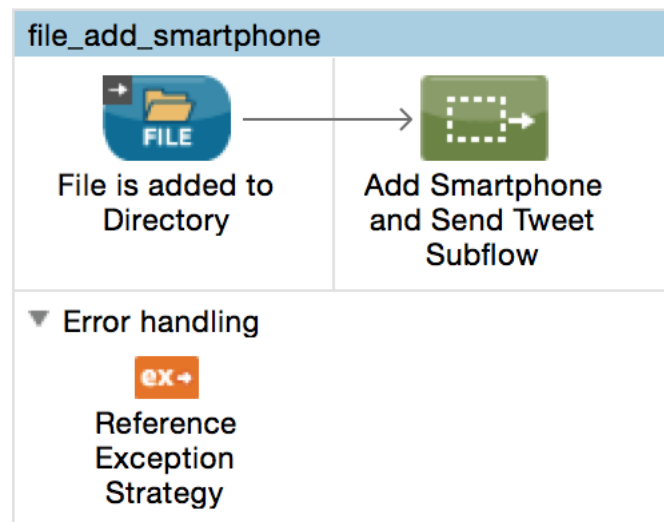
Both the SOAP and Web flows reference the same subflow for performing an unsubscription, which is just a deletion in the database.

3 Adding Smartphones

Smartphones can be added either by putting the XML files under the appropriate directory (`/smartphones/`) or via SOAP.

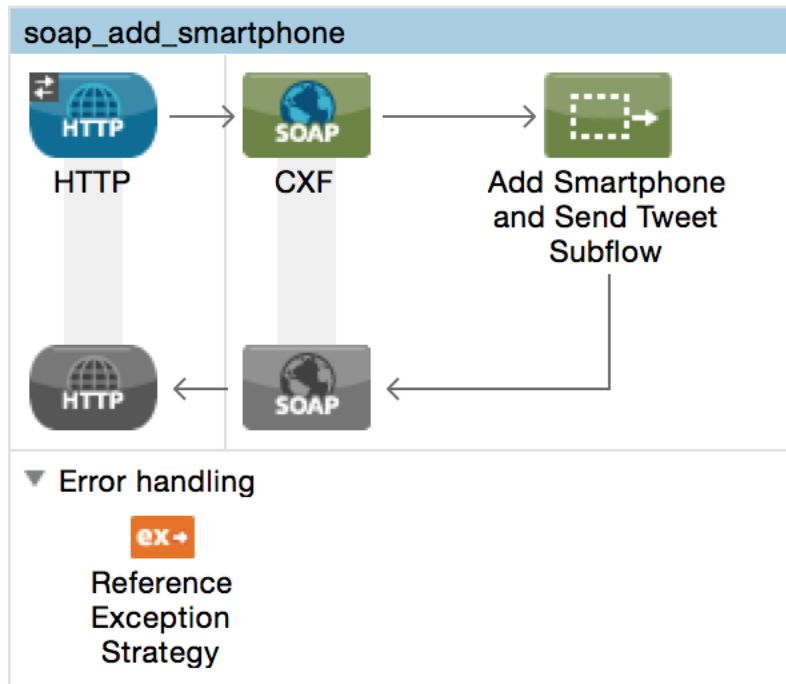


3.1 Files Directory



The File connector automatically deletes all processed files, this in turn minimizes duplicate entries in the database.

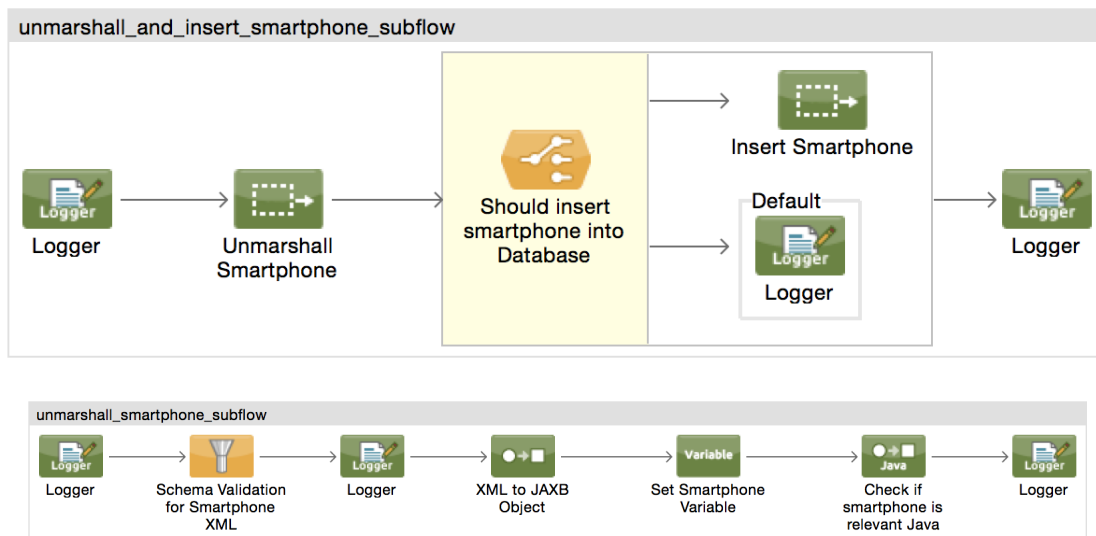
3.2 SOAP

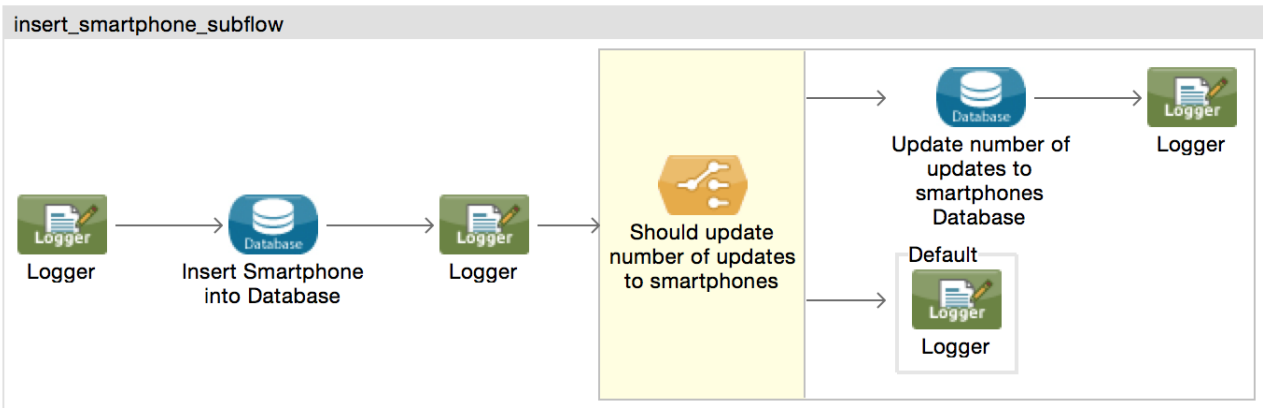


TODO

3.3 Shared Behaviour

Both the File and SOAP flows reference the same subflow which will unmarshal the given XML file, check its validity with a schema, and check if it has a relevant screen size (greater or equal to 10).





4 Statistics

Whenever the number of updates done to the list of smartphones in the database, the number of smartphones in the database, the number of emails sent, or the total number of tweets changes a tweet is posted to the target account **is.spammer.2015** (https://twitter.com/is_spammer_2015).

4.1 Database Handling

In order to track these changes, the database keeps a table named *statistics* and acts as a basic global counter.

```

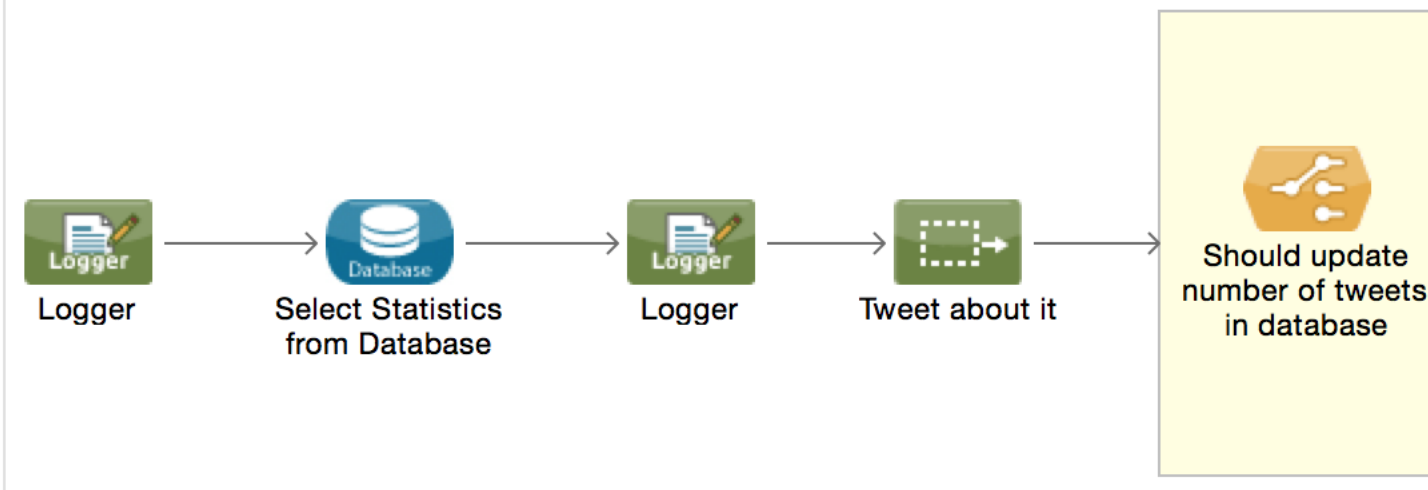
1 /* Table for System Statistics */
2 CREATE TABLE statistics (
3   num_updates_to_smartphones INT DEFAULT 0,
4   num_cur_smartphones INT DEFAULT 0,
5   num_sent_emails INT DEFAULT 0,
6   num_sent_tweets INT DEFAULT 0
7 );

```

We tried to use triggers to manage the statistics "accounting" for us, but unfortunately, mule would more often than not generate deadlock states which we just couldn't seem to control, thusly, we opted for updating each of the tracked statistics target counter (number of updates, number of sent emails, etc.) upon success of the associated actions (adding smartphones, sending and email, etc.). It's not the best solution, but given that this assignment is focused on correct usage of Mule and ESB technologies, we feel as this should not compromise our grade.

4.2 Tweeting

tweet_statistics_subflow



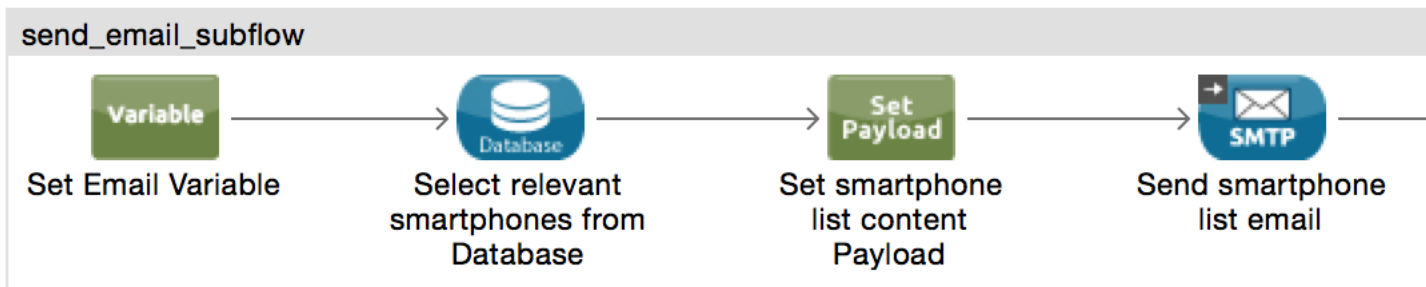
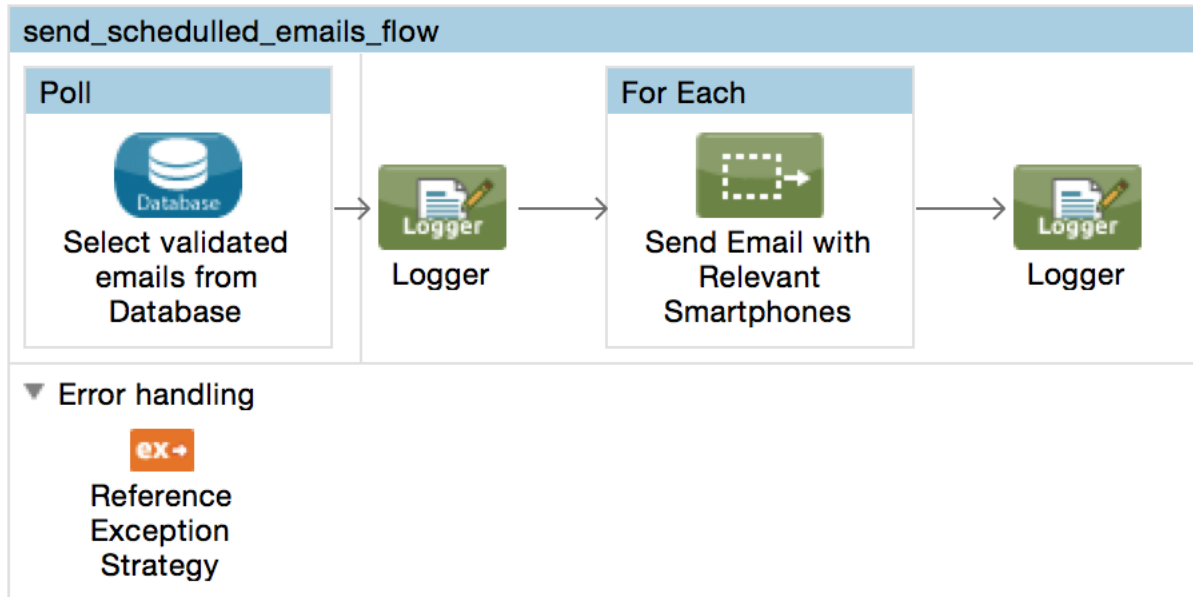
tweet_subflow



As the image shows, we first increment the number of posted tweets, proceeding to fetch the current statistics from the database, parsing the queries result into a nicer format such as

```
1 2015-12-21T18:02:38.768Z Statistics: 9 updates, 6 smartphones, 0 emails, 9 tweets
```

5 Sending Scheduled Emails



6 Error Handling/Logging



AnypointStudio provides a simple asynchronous logging mechanism, automatically updating a file at `.../.mule/logs/assignment3.log`. However, since it doesn't provide the most user-friendly debugging tools, we decided to log almost every input and output for each component. As an added bonus, we also provide a **global catch exception strategy** which is referenced in every flow, logging any exception which might occur.