Alert Management with HTML Data Tables In Outlook Emails

**Deloitte**

**5/1/2015**

Jerome Rajan

Data transparency and traceability are critical to the success of any ETL flow. The business or the technical support team needs to stay updated about potential data outliers. Besides data transparency, the stakeholders also need to be on top of the process executions on a day to day basis. This accelerator can be plugged in to any flow to create email alerts for the business users in the most intuitive and streamlined manner

Alert Management with HTML Data Tables in Outlook Emails

# Abstract:

Data transparency and traceability are critical to the success of any ETL flow. The business or the technical support team needs to stay updated about potential data outliers. Outliers could be in the form of:

1. Data semantic errors
2. Data not meeting functional constraints
3. Unexpected data.
4. New data not matching internal master data

Besides data transparency, the stakeholders also need to be on top of the process executions on a day to day basis. For e.g., if a process aborts due to whatever reason, the support team needs to be notified immediately about the situation through an email.

Many ETL flows incorporate such email alerts in an ad-hoc manner such that the process would send an email as soon as it encounters an issue. But these alerts are in no way regulated or archived for future reference. A data issue could have been alerted via email to a certain Mr. X. But tomorrow, if Mr. X were replaced with Mr. Y, Mr. Y would have no way to know if the data issue is present or not.

Hundreds of alerts can go out within a small timeframe and it is essential that such alerts be treated with the same care as the actual data. This reusable asset looks to do exactly that by ensuring that alerts are logged, audited and transparent. The framework also ensures complete management of email metadata by allowing the recipient list, email subject and body to be driven through tables.

A typical ETL flow



What this accelerator aims to do



# Data Model

These tables will be central to the alert management.

The reference table will define the email metadata and what kind of data should go out in each email. Each entry into the reference table will correspond to a different alert type.

The alert log table is where every alert that is logged before it is sent out. The Alert key will tell which type of alert the log corresponds to.

|  |  |
| --- | --- |
| Alert Reference | Alert Log |
|  |  |

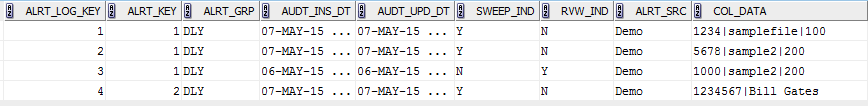
The developer’s job is to only make entries into this alert log table at whichever point he needs to send out an alert in the ETL process. The COL\_DATA field would basically be a pipe delimited string of all the business values that need to go into the alert. At the end of the process the Alert UNIX script should be invoked which will read the entire latest alert log and create a HTML table and send it out in an email to the defined recipient list.

# Sample Data

ALRT\_REF\_TBL



ALRT\_LOG\_TBL



# Code

|  |  |  |
| --- | --- | --- |
| ALRT\_LOG\_TBL | ALRT\_REF\_TBL | UNIX Driver |
|  |  |  |

# Business Case

The supply chain division of a global organization has adopted the strategy of Vendor Managed Inventory to ensure that all its customers are well stocked and that its products are able to meet the seasonal ups and downs of demand and supply. To achieve optimum supply chain balance, the company’s customers send point of sale data about their inventory on a daily basis. This information consists of new customers, new products, legacy products, threshold values of critical quantity metrics, etc.

The business typically uses all these metrics through BI reports that tell them about certain outliers in regular patterns. It gives them information on stock levels, order levels, how a product is performing, seasonal peaks and slides, etc.

The source data comes from more than 1000 customers who could be retailers, wholesalers or distribution centers. These customers are spread across different geographies. The supply chain division needs to keep a tab on the activities of each of these customers at a product level. The business wanted to be kept informed about

1. New products that have entered the market – This could be new products or a new way in which a certain product has been packaged.
2. New customers
3. Legacy products – If a customer is sending inventory data for a product that is considered legacy/inactive
4. Invalid data – Data that doesn’t sync up with the organization’s internal master
5. Missing information – The data sent by the customers to this company is part of an agreement that involves a cost. The organization needs to ensure that its customers are sending all the information as agreed upon though the agreement.
6. Duplicate information
7. Stale data

And many more…

The process supporting the needs of the business was near real time and executed in intervals of an hour. The business wanted these alerts every one hour so that appropriate action could be taken. The ability to view all the required information directly in their emails without having to open attachments and dig through CSV data was a big time saver and value add for the client.

# Impact

1. The ability to view the required data directly in an email through a structured HTML table was a big value-add to the business which until then was dealing with csv attachments.
2. The business started to quote the alert emails verbatim for all discussions amongst the senior management
3. The accelerator saved 100 man hours of effort by because of its reusable nature. It ensured that re-work and re-design to the ETL flow did not cause any impact to the way alert notifications were being generated.
4. The alert framework has given far greater visibility to the business and has made the reports more trustable and up to date.