Diamond Data – Consolidation (Matching) Rules

# Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Comment** | **Date** |
| 0.1 | Steph Voss | First draft – version passed to Cloudstream for POC | 19/03/2018 |
| 0.2 | Chris Dunigan | Basis for POC and will stay similar for full build – removed rules 6-10 as matching too vague, added CD Notes for clarification | 27/03/2018 |
| 0.3 | Joe SP | Clarified ‘Initial’ rule | 9/4/2018 |
| 0.4 | Joe SP | Added rules 8 and 9 back in, added rules for single email. Signed off version for full build. | 20/4/2018 |
| 0.5 | Tina Wood | Minor review of structure and terminology – removed Design Principles & Suggestions for Implementation section for BU stakeholders, added info on current matching rules after confirming with JSP | 01/06/2018 |
| 0.6 | Joe SP | Added requirement that an email should only belong on one contact | 18/09/2018 |

# Background

This document describes the changes needed to enhance the existing consolidation (matching) rules in DiaMonD in order to minimise the risk of over consolidation. These changes have been implemented as part of the DiaMonD Set Analyser Replacement solution (June 2018).

The existing matching rules (pre June 2018) are as follows:

Source contact data is matched then consolidated if the source contact data records:

* Share the same email address
* Share the same telephone number, first name and last name
* Share the same first address line, postcode, first name and last name

The set of matching rules in Diamond (pre June 2018) operate mostly correctly, however, approximately 10% of contact records are over consolidated when the matching rules bringing them together rely on email address alone. The enhanced rules take into account other name elements thereby preventing the over consolidation of contact records which are deemed to be the same person.

Another key difference between consolidation (matching) rules pre June 2018 and enhanced consolidation (matching) rules is that previously new records were considered for matching against existing records in the Consolidated database. However, in the new solution, this does not happen as consolidation (matching) rules are run daily, taking into account and processing all data every day.

# Matching Rules

The table below shows the attributes and combinations of each which should be considered for consolidation (matching) purposes:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **INITIAL** | **FIRST** | **LAST** | **ADDRESSLINE 1** | **POSTCODE** | **EMAIL** | **PHONE** | **COMPANY** | **CONSTRAINT** |
| 1 |  | X | X |  |  | X |  |  | TITLE |
| 2 | X |  | X |  |  | X |  |  | TITLE, FIRST |
| 3 |  |  | X |  |  | X |  |  | TITLE, FIRST |
| 4 |  |  |  |  |  | X |  |  | TITLE, FIRST, LAST |
| 5 |  | X | X |  |  |  |  | X | TITLE |
| 8 | X |  | X | X | X |  |  |  | TITLE, FIRST |
| 9 |  | X | X | X | X |  |  |  | TITLE |
| 11 |  | X | X |  |  |  | X |  | TITLE |

Where an email can match to two distinct contacts then match to one of these. Use the following criteria:

* Use the contact where ContactRole = ‘Prime Contact’
* Then use minimum ‘FirstName’ value
* Then use minimum ‘LastName’ value

Example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FirstName | LastName | EmailAddress | ContactRole | Contact Grouping |
| Bob | Perks | [bob@perkins.com](mailto:bob@perkins.com) | Prime Contact | 1 |
| Joseph | Perks | [bob@perkins.com](mailto:bob@perkins.com) | Prime Contact | 2 |
| Mary | Perks | [bob@perkins.com](mailto:bob@perkins.com) |  | 3 |
|  |  | [bob@perkins.com](mailto:bob@perkins.com) |  | 1 (this joins to Contact Grouping 1 because of above criteria.) |

The principle applied in the matching rules is that at least 3 pieces of information are required to identify an individual:

* Surname
* Initial or Firstname
* One other contact point identifier (emailaddress, address or telephone) or Company Name

The exception to this is rule 4 – emailaddress only. This can be used as a single match attribute if there are no other conflicting elements of the contact. That is:

* Emailaddress matches across both records
* Firstname is NULL or populated on one side of the match only
* Surname is NULL or populated on one side of the match only
* If emailaddress matches and there is a conflict on the firstname or surname then the records should be prevented from coming together.

The constraint column in the table above defines the prevention condition for each of the match rules in the table.

* Title can be ignored in each of the match rules *except* for where it conflicts with another match candidate e.g. Mr versus Mrs. For this reason it should be considered as a match attribute for each rule where it is populated.
* Where First is listed as a match attribute constraint (rules 2-4,6,8,10) it means if there is a firstname conflict (as it is not part of the match attribute for that rule) then match candidates should be prevented from matching
* Where Last is listed as a match attribute constraint (rule 4 – emailaddress only) it means match candidates should not be brought together where there is an emailaddress match but a lastname mismatch.

The table below presents the different combinations of values that could be found for these constraint attributes and what the outcome of the match would be in each case. These are considered as Constraint Rules.

|  |  |  |  |
| --- | --- | --- | --- |
| Constraint Rule # | Match Candidate 1 Attribute Value | Match Candidate 2 Attribute Value | Result |
| 1 | Null | Null | Match |
| 2 | Not null | Null | Match |
| 3 | Null | Not Null | Match |
| 4 | Not null & the same | Not null & the same | Match |
| 5 | Not null & different | Not null & different | No Match |

An example illustrates this point:

For Rule 4 which is emailaddress only, one of the constraint attributes is surname. The table shows the match outcome for each constraint rule example.

**Constraint Rule 1: NULL vs NULL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | Surname | Surname Constraint Rule | Outcome |
| 1 | Bob.smith@hotmail.com | Null | Null | Match |
| 2 | Bob.smith@hotmail.com | Null | Null |

**Constraint Rule 2: NULL vs NOT NULL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | Surname | Surname Constraint Rule | Outcome |
| 1 | Bob.smith@hotmail.com | Null | Null | Match |
| 2 | Bob.smith@hotmail.com | Smith | Not Null |

**Constraint Rule 3: NOT NULL vs NULL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | Surname | Surname Constraint Rule | Outcome |
| 1 | Bob.smith@hotmail.com | Jones | Not Null | Match |
| 2 | Bob.smith@hotmail.com | Null | Null |

**Constraint Rule 4: NOT NULL vs NOT NULL (and the same)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | Surname | Surname Constraint Rule | Outcome |
| 1 | Bob.smith@hotmail.com | Smith | Not Null Same | Match |
| 2 | Bob.smith@hotmail.com | Smith | Not Null Same |

**Constraint Rule 4: NOT NULL vs NOT NULL (and different)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | Surname | Surname Constraint Rule | Outcome |
| 1 | Bob.smith@hotmail.com | Smith | Not Null Different | No Match |
| 2 | Bob.smith@hotmail.com | Jones | Not Null Different |

Note:

* For matching: NULLS can’t match with other NULLS – except if in the title field. (Title can be removed as a matching criteria and only acts as a constraint)
* For constraints: NULLS can match with other NULLS

Note: ‘Initial’ is defined as consisting of a single character in the First Name. This prevents over-matching eg

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | FirstName | Surname | Outcome |
| 1 | Bob.smith@hotmail.com | Steve | Smith | No Match |
| 2 | Bob.smith@hotmail.com | Sarah | Smith |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Record # | Email | FirstName | Surname | Outcome |
| 1 | Bob.smith@hotmail.com | S | Smith | Match |
| 2 | Bob.smith@hotmail.com | Sarah | Smith |

**Email should only relate to one Contact**

If consolidation has resulted in an email belonging to more than one contact then a single contact should be chosen for the email to relate to. The rule for deciding which contact the email should be is the same for deciding which contact an email on its own should match to.

* Use the contact where ContactRole = ‘Prime Contact’
* Then use minimum ‘FirstName’ value
* Then use minimum ‘LastName’ value

For example, to continue with the above example, ContactID 1 should have the EmailAddress. It should be made null for the other two contacts.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FirstName | LastName | EmailAddress | ContactRole | ContactID |
| Bob | Perks | [bob@perkins.com](mailto:bob@perkins.com) | Prime Contact | 1 |
| Joseph | Perks |  | Prime Contact | 2 |
| Mary | Perks |  |  | 3 |
|  |  | [bob@perkins.com](mailto:bob@perkins.com) |  | 1 |