Approach and Assumptions for the Entity Resolution Exercise

For the Sr. Data Engineer, MDM position at HubSpot

By Eric Wolfson

Tools / Languages Used in this project

- Python 3.11 [1]

- Pandas library for Python [2]

- MS SQL Server 2022 [3]
  (with pypyodbc library in Python for connecting)

- Dagster [4]

Reasons for Dagster

- Dagster allows stages of the pipeline to be recompiled and run independently of one another by storing and returning data at each asset.
- It works well with creating views in SQL so that only the SQL queries that are needed at specific stages can be run.
- Each step in the conversion process is treated as a Dagster asset, which makes it simple to add a step in the pipeline. All that needs to be added is the @asset annotation.
- A feature for running assets automatically can be easily implemented (slides 19-22 show an explanation):
  - For this project, when a csv file in the working directory is updated, the path in the pipeline for that data is run automatically. This is checked every 45 seconds.
  - We can run only the specific paths of the pipeline needed every time a file is updated.

Steps In Dagster Data Pipeline

1) Retrieve data for CSV files and load into DataFrames
   - crm_dataframe_from_csv asset
   - acme_dataframe_from_csv asset
   - rapid_data_dataframe_from_csv asset
2) Load dataframes in SQL table
   - load_crm_into_db asset
   - load_acme_into_db asset
   - rapid_data_dataframe_from_csv asset
3) Merge duplicates in RapidData (I chose to merge based on name, email_address and ip_address):
   - resolve_rapid_data_duplicates asset
4) Merge the three tables using SQL joins:
   - combine_exact_contacts asset
5) Pull data from SQL Server database back to Python Pandas:
   - extrapolate_data asset

Steps In Dagster Data Pipeline II

5) Fill in null values with placeholder values of correct type for the final CSV file (int, JSON, string…):

- clean_data asset

6) Fill in missing values (country, company_revenue, company_employees, company_industry):

- extrapolate_data asset

7) Combine common contacts one final time (I chose to merge based on name and phone number)

- Combine_post_merge asset

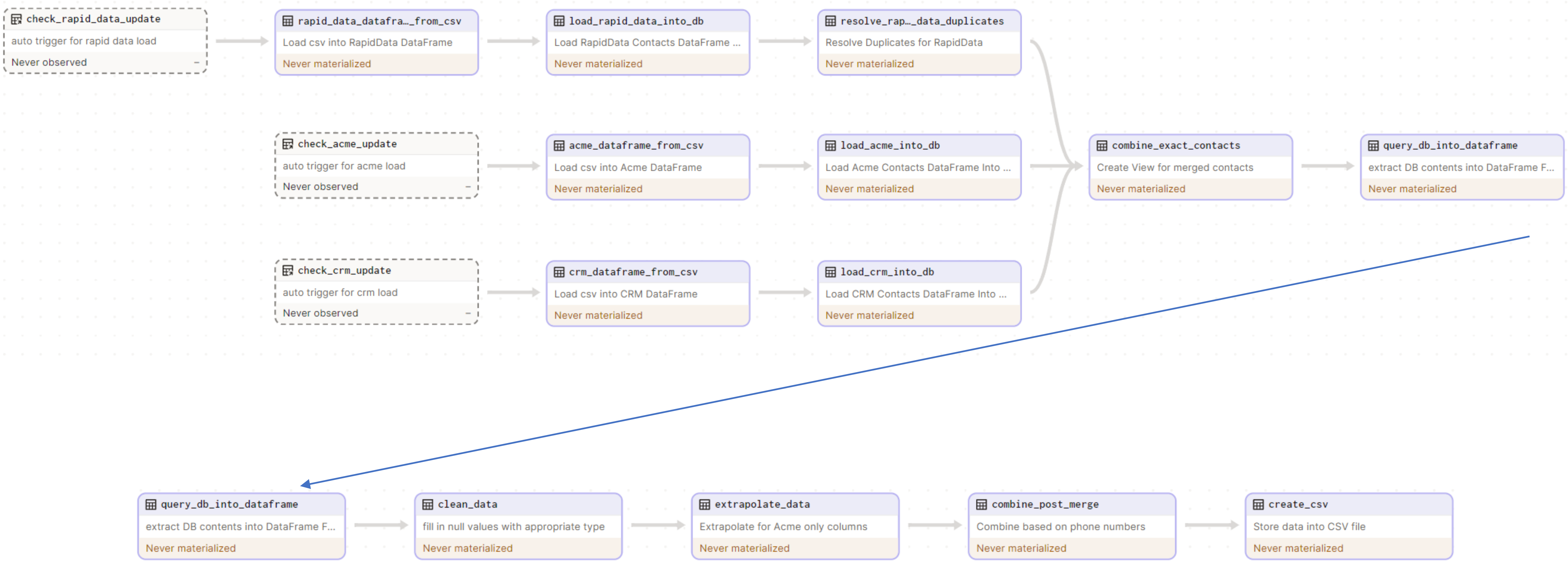
8) Store final table in csv result file

- Create_csv asset

Use of SQL and Python/Pandas

- For the merging of the 3 csv files, SQL (SQL Server) was used.
- Merging tables in SQL using an RDBMS is faster than merging DataFrames in Python, which will help significantly when using large datasets. [5]
- SQL Server Joins are multithreaded operations (using the ODBC driver), where merge in Pandas is single threaded. [6]
- In addition, data is stored in a tree-like data structure in order to perform operations more quickly. [7]
- Pandas is easier for cleaning data, filling in null values, and running operations that extrapolate data.

Data Pipeline For Entity Resolution In Dagster



check_rapid_data_update
auto trigger for rapid data load
Never observed –

rapid_data_datafra…_from_csv
Load csv into RapidData DataFrame
Never materialized

load_rapid_data_into_db
Load RapidData Contacts DataFrame ...
Never materialized

resolve_rap..._data_duplicates
Resolve Duplicates for RapidData
Never materialized

check_acme_update
auto trigger for acme load
Never observed –

acme_dataframe_from_csv
Load csv into Acme DataFrame
Never materialized

load_acme_into_db
Load Acme Contacts DataFrame Into ...
Never materialized

check_crm_update
auto trigger for crm load
Never observed –

crm_dataframe_from_csv
Load csv into CRM DataFrame
Never materialized

load_crm_into_db
Load CRM Contacts DataFrame Into ...
Never materialized

combine_exact_contacts
Create View for merged contacts
Never materialized

query_db_into_dataframe
extract DB contents into DataFrame F...
Never materialized

query_db_into_dataframe
extract DB contents into DataFrame F...
Never materialized

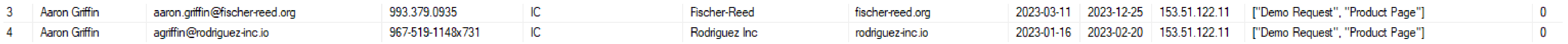
clean_data
fill in null values with appropriate type
Never materialized

extrapolate_data
Extrapolate for Acme only columns
Never materialized

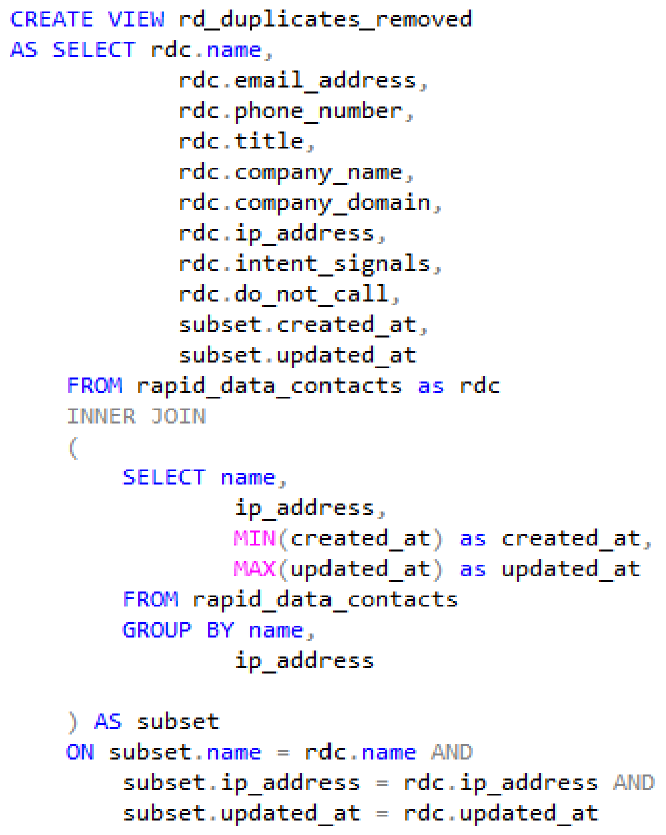
combine_post_merge
Combine based on phone numbers
Never materialized

create_csv
Store data into CSV file
Never materialized

Step for Condensing rapid_data_contacts table



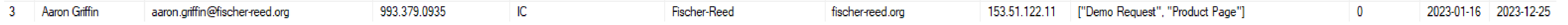
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	2023-03-11	2023-12-25	153.51.122.11	["Demo Request", "Product Page"]	0
4	Aaron Griffin	agriffin@rodriguez-inc.io	967-519-1148x731	IC	Rodriguez Inc	rodriguez-inc.io	2023-01-16	2023-02-20	153.51.122.11	["Demo Request", "Product Page"]	0



```sql
CREATE VIEW rd_duplicates_removed
AS SELECT rdc.name,
            rdc.email_address,
            rdc.phone_number,
            rdc.title,
            rdc.company_name,
            rdc.company_domain,
            rdc.ip_address,
            rdc.intent_signals,
            rdc.do_not_call,
            subset.created_at,
            subset.updated_at
    FROM rapid_data_contacts as rdc
    INNER JOIN
    (
        SELECT name,
                ip_address,
                MIN(created_at) as created_at,
                MAX(updated_at) as updated_at
        FROM rapid_data_contacts
        GROUP BY name,
                ip_address

    ) AS subset
    ON subset.name = rdc.name AND
        subset.ip_address = rdc.ip_address AND
        subset.updated_at = rdc.updated_at
```

rd_duplicates_removed



3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	153.51.122.11	["Demo Request", "Product Page"]	0	2023-01-16	2023-12-25

Columns in the DataSets

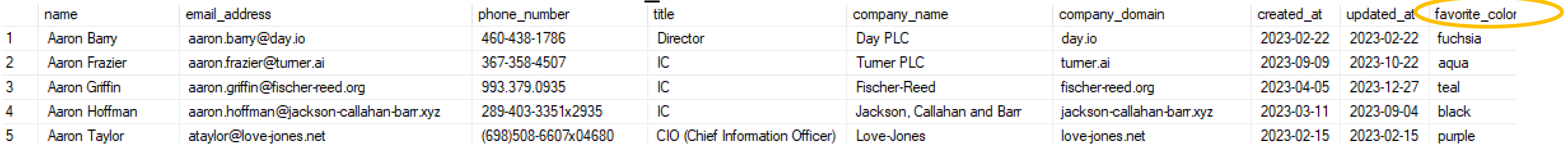
- There are common columns in all 3 datasets, and columns in each of the datasets that are unique to that dataset.
- The joining of the tables in one table attempts to combine the three datasets by checking for field similarity for the records for common columns.
- Afterwards, the non common columns are added from the other datasets for that record in the new table.
- The result after the SQL join operation is a table with all the columns that are common for the datasets, and each non common column in the new table for each record.

Merging using SQL (combine_exact_contents) Shown on next 3 slides

The following steps were taken in the SQL merging asset:

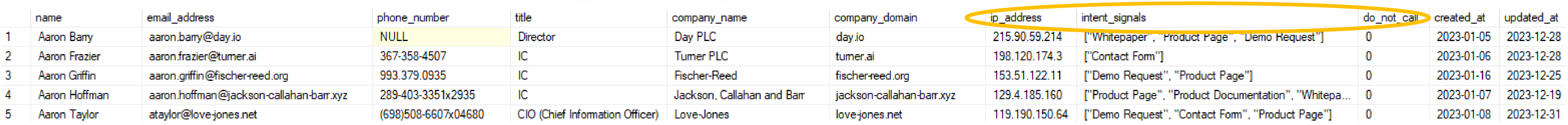
1. The CRM table was merged with the reduced RapidData table using a full join for merging and “Coalesce” function for taking only the non null values for the new columns.
2. The Acme data table was merged with the above result table using a full join for merging and Coalesce function for taking only the non null values for the new columns.
3. To preserve as much date information as possible, I decided to take the smallest created_at date from the three tables for a common record and largest updated_at date for those common records, to treat them as though they are all one contact record.

crm_contacts table

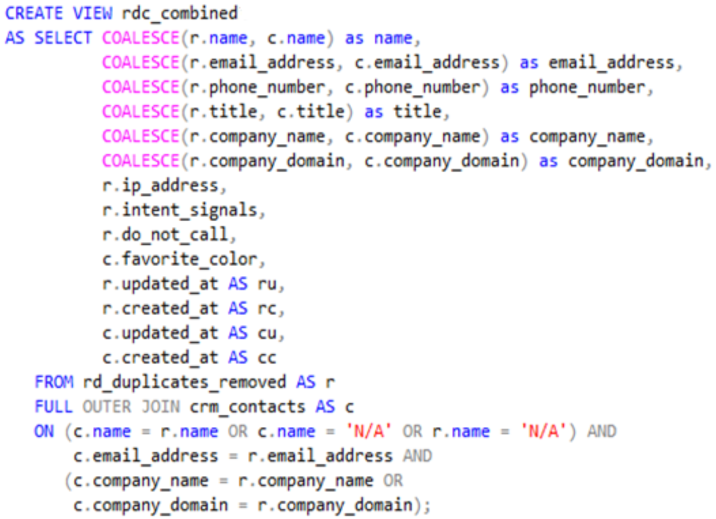


	name	email_address	phone_number	title	company_name	company_domain	created_at	updated_at	favorite_color
1	Aaron Barry	aaron.barry@day.io	460-438-1786	Director	Day PLC	day.io	2023-02-22	2023-02-22	fuchsia
2	Aaron Frazier	aaron.frazier@turner.ai	367-358-4507	IC	Turner PLC	turner.ai	2023-09-09	2023-10-22	aqua
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	2023-04-05	2023-12-27	teal
4	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	289-403-3351x2935	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	2023-03-11	2023-09-04	black
5	Aaron Taylor	ataylor@love-jones.net	(698)508-6607x04680	CIO (Chief Information Officer)	Love-Jones	love-jones.net	2023-02-15	2023-02-15	purple

rd_duplicates_removed view

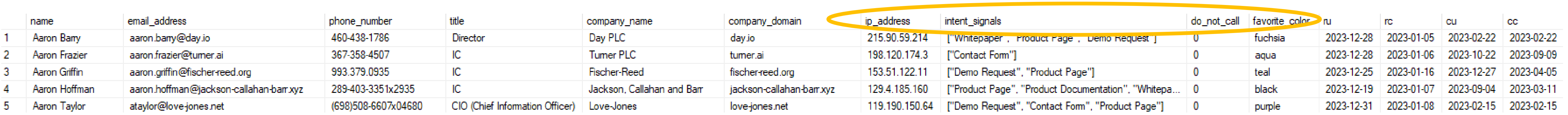


	name	email_address	phone_number	title	company_name	company_domain	ip_address	intent_signals	do_not_call	created_at	updated_at
1	Aaron Barry	aaron.barry@day.io	NULL	Director	Day PLC	day.io	215.90.59.214	["Whitepaper", "Product Page", "Demo Request"]	0	2023-01-05	2023-12-28
2	Aaron Frazier	aaron.frazier@turner.ai	367-358-4507	IC	Turner PLC	turner.ai	198.120.174.3	["Contact Form"]	0	2023-01-06	2023-12-28
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	153.51.122.11	["Demo Request", "Product Page"]	0	2023-01-16	2023-12-25
4	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	289-403-3351x2935	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	129.4.185.160	["Product Page", "Product Documentation", "Whitepa...	0	2023-01-07	2023-12-19
5	Aaron Taylor	ataylor@love-jones.net	(698)508-6607x04680	CIO (Chief Information Officer)	Love-Jones	love-jones.net	119.190.150.64	["Demo Request", "Contact Form", "Product Page"]	0	2023-01-08	2023-12-31



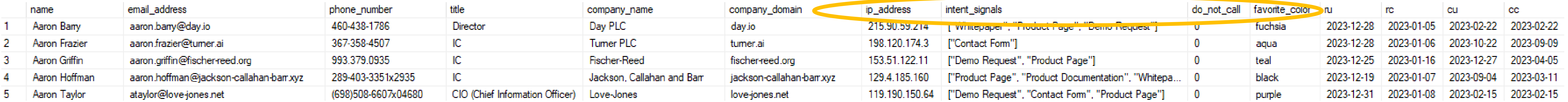
```sql
CREATE VIEW rdc_combined
AS SELECT COALESCE(r.name, c.name) as name,
          COALESCE(r.email_address, c.email_address) as email_address,
          COALESCE(r.phone_number, c.phone_number) as phone_number,
          COALESCE(r.title, c.title) as title,
          COALESCE(r.company_name, c.company_name) as company_name,
          COALESCE(r.company_domain, c.company_domain) as company_domain,
          r.ip_address,
          r.intent_signals,
          r.do_not_call,
          c.favorite_color,
          r.updated_at AS ru,
          r.created_at AS rc,
          c.updated_at AS cu,
          c.created_at AS cc
   FROM rd_duplicates_removed AS r
   FULL OUTER JOIN crm_contacts AS c
   ON (c.name = r.name OR c.name = 'N/A' OR r.name = 'N/A') AND
       c.email_address = r.email_address AND
      (c.company_name = r.company_name OR
       c.company_domain = r.company_domain);
```

rdc_combined view



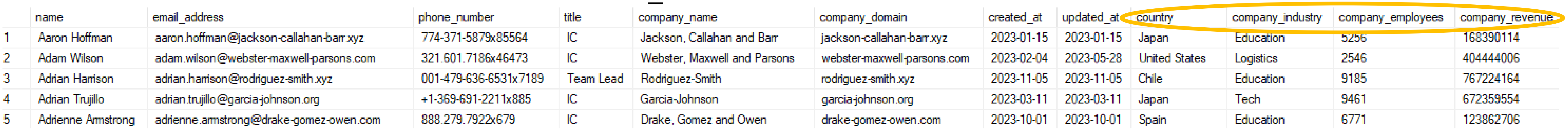
	name	email_address	phone_number	title	company_name	company_domain	ip_address	intent_signals	do_not_call	favorite_color	ru	rc	cu	cc
1	Aaron Barry	aaron.barry@day.io	460-438-1786	Director	Day PLC	day.io	215.90.59.214	["Whitepaper", "Product Page", "Demo Request"]	0	fuchsia	2023-12-28	2023-01-05	2023-02-22	2023-02-22
2	Aaron Frazier	aaron.frazier@turner.ai	367-358-4507	IC	Turner PLC	turner.ai	198.120.174.3	["Contact Form"]	0	aqua	2023-12-28	2023-01-06	2023-10-22	2023-09-09
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	153.51.122.11	["Demo Request", "Product Page"]	0	teal	2023-12-25	2023-01-16	2023-12-27	2023-04-05
4	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	289-403-3351x2935	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	129.4.185.160	["Product Page", "Product Documentation", "Whitepa...	0	black	2023-12-19	2023-01-07	2023-09-04	2023-03-11
5	Aaron Taylor	ataylor@love-jones.net	(698)508-6607x04680	CIO (Chief Information Officer)	Love-Jones	love-jones.net	119.190.150.64	["Demo Request", "Contact Form", "Product Page"]	0	purple	2023-12-31	2023-01-08	2023-02-15	2023-02-15

rd_duplicates_removed view

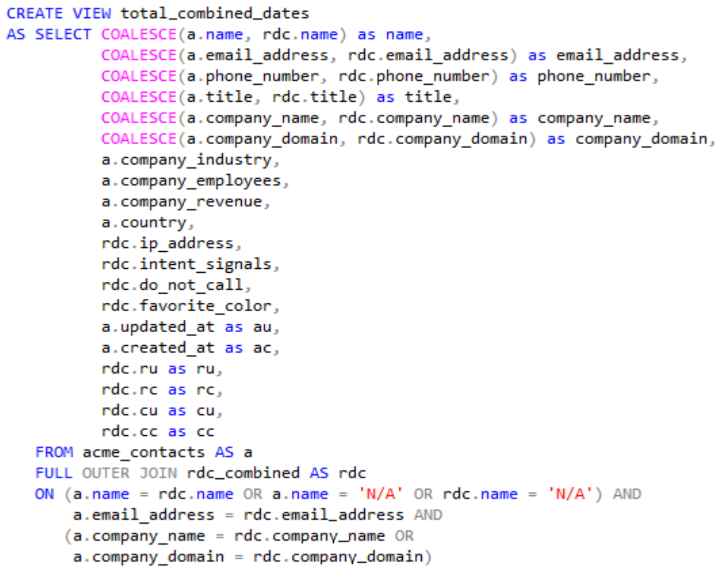


	name	email_address	phone_number	title	company_name	company_domain	ip_address	intent_signals	do_not_call	favorite_color	ru	rc	cu	cc
1	Aaron Barry	aaron.barry@day.io	460-438-1786	Director	Day PLC	day.io	215.90.59.214	["Whitepaper", "Product Page", "Demo Request"]	0	fuchsia	2023-12-28	2023-01-05	2023-02-22	2023-02-22
2	Aaron Frazier	aaron.frazier@turner.ai	367-358-4507	IC	Turner PLC	turner.ai	198.120.174.3	["Contact Form"]	0	aqua	2023-12-28	2023-01-06	2023-10-22	2023-09-09
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	153.51.122.11	["Demo Request", "Product Page"]	0	teal	2023-12-25	2023-01-16	2023-12-27	2023-04-05
4	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	289-403-3351x2935	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	129.4.185.160	["Product Page", "Product Documentation", "Whitepa...	0	black	2023-12-19	2023-01-07	2023-09-04	2023-03-11
5	Aaron Taylor	ataylor@love-jones.net	(698)508-6607x04680	CIO (Chief Information Officer)	Love-Jones	love-jones.net	119.190.150.64	["Demo Request", "Contact Form", "Product Page"]	0	purple	2023-12-31	2023-01-08	2023-02-15	2023-02-15

acme_contacts table

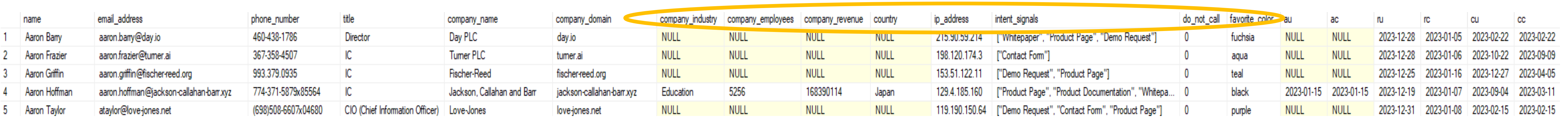


	name	email_address	phone_number	title	company_name	company_domain	created_at	updated_at	country	company_industry	company_employees	company_revenue
1	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	774-371-5879x85564	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	2023-01-15	2023-01-15	Japan	Education	5256	168390114
2	Adam Wilson	adam.wilson@webster-maxwell-parsons.com	321.601.7186x46473	IC	Webster, Maxwell and Parsons	webster-maxwell-parsons.com	2023-02-04	2023-05-28	United States	Logistics	2546	404444006
3	Adrian Harrison	adrian.harrison@rodriguez-smith.xyz	001-479-636-6531x7189	Team Lead	Rodriguez-Smith	rodriguez-smith.xyz	2023-11-05	2023-11-05	Chile	Education	9185	767224164
4	Adrian Trujillo	adrian.trujillo@garcia-johnson.org	+1-369-691-2211x885	IC	Garcia-Johnson	garcia-johnson.org	2023-03-11	2023-03-11	Japan	Tech	9461	672359554
5	Adrienne Armstrong	adrienne.armstrong@drake-gomez-owen.com	888.279.7922x679	IC	Drake, Gomez and Owen	drake-gomez-owen.com	2023-10-01	2023-10-01	Spain	Education	6771	123862706



```sql
CREATE VIEW total_combined_dates
AS SELECT COALESCE(a.name, rdc.name) as name,
          COALESCE(a.email_address, rdc.email_address) as email_address,
          COALESCE(a.phone_number, rdc.phone_number) as phone_number,
          COALESCE(a.title, rdc.title) as title,
          COALESCE(a.company_name, rdc.company_name) as company_name,
          COALESCE(a.company_domain, rdc.company_domain) as company_domain,
          a.company_industry,
          a.company_employees,
          a.company_revenue,
          a.country,
          rdc.ip_address,
          rdc.intent_signals,
          rdc.do_not_call,
          rdc.favorite_color,
          a.updated_at as au,
          a.created_at as ac,
          rdc.ru as ru,
          rdc.rc as rc,
          rdc.cu as cu,
          rdc.cc as cc
   FROM acme_contacts AS a
   FULL OUTER JOIN rdc_combined AS rdc
   ON (a.name = rdc.name OR a.name = 'N/A' OR rdc.name = 'N/A') AND
      a.email_address = rdc.email_address AND
     (a.company_name = rdc.company_name OR
      a.company_domain = rdc.company_domain)
```

total_combined_dates view



	name	email_address	phone_number	title	company_name	company_domain	company_industry	company_employees	company_revenue	country	ip_address	intent_signals	do_not_call	favorite_color	au	ac	ru	rc	cu	cc
1	Aaron Barry	aaron.barry@day.io	460-438-1786	Director	Day PLC	day.io	NULL	NULL	NULL	NULL	215.90.59.214	["Whitepaper", "Product Page", "Demo Request"]	0	fuchsia	NULL	NULL	2023-12-28	2023-01-05	2023-02-22	2023-02-22
2	Aaron Frazier	aaron.frazier@turner.ai	367-358-4507	IC	Turner PLC	turner.ai	NULL	NULL	NULL	NULL	198.120.174.3	["Contact Form"]	0	aqua	NULL	NULL	2023-12-28	2023-01-06	2023-10-22	2023-09-09
3	Aaron Griffin	aaron.griffin@fischer-reed.org	993.379.0935	IC	Fischer-Reed	fischer-reed.org	NULL	NULL	NULL	NULL	153.51.122.11	["Demo Request", "Product Page"]	0	teal	NULL	NULL	2023-12-25	2023-01-16	2023-12-27	2023-04-05
4	Aaron Hoffman	aaron.hoffman@jackson-callahan-barr.xyz	774-371-5879x85564	IC	Jackson, Callahan and Barr	jackson-callahan-barr.xyz	Education	5256	168390114	Japan	129.4.185.160	["Product Page", "Product Documentation", "Whitepa...	0	black	2023-01-15	2023-01-15	2023-12-19	2023-01-07	2023-09-04	2023-03-11
5	Aaron Taylor	ataylor@love-jones.net	(698)508-6607x04680	CIO (Chief Information Officer)	Love-Jones	love-jones.net	NULL	NULL	NULL	NULL	119.190.150.64	["Demo Request", "Contact Form", "Product Page"]	0	purple	NULL	NULL	2023-12-31	2023-01-08	2023-02-15	2023-02-15

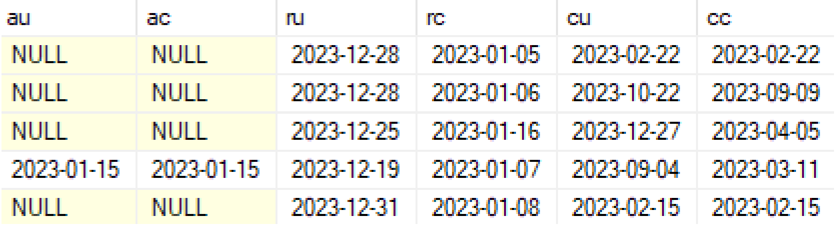
Merging of date information

- crm_contacts, acme_contacts and rapid_data_contacts each have updated_at and created_at dates for each record

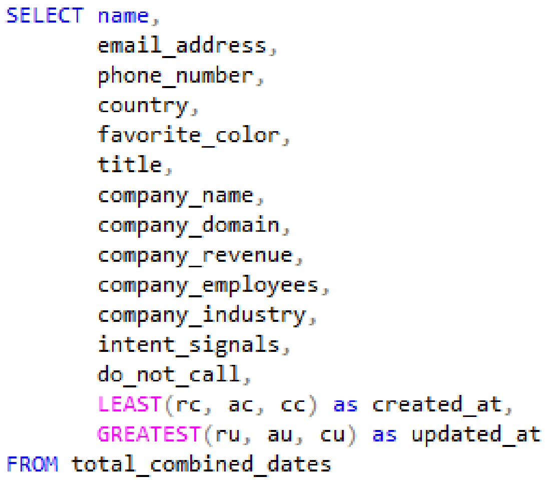
- We take the greatest of the three updated_at fields for the combined contact

- We take the least of the three created_at fields for the combined contact

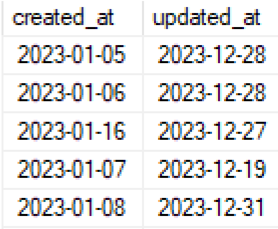
- This is done to preserve the most date information across equivalent contacts by treating them as one record



au	ac	ru	rc	cu	cc
NULL	NULL	2023-12-28	2023-01-05	2023-02-22	2023-02-22
NULL	NULL	2023-12-28	2023-01-06	2023-10-22	2023-09-09
NULL	NULL	2023-12-25	2023-01-16	2023-12-27	2023-04-05
2023-01-15	2023-01-15	2023-12-19	2023-01-07	2023-09-04	2023-03-11
NULL	NULL	2023-12-31	2023-01-08	2023-02-15	2023-02-15



```
SELECT name,
       email_address,
       phone_number,
       country,
       favorite_color,
       title,
       company_name,
       company_domain,
       company_revenue,
       company_employees,
       company_industry,
       intent_signals,
       do_not_call,
       LEAST(rc, ac, cc) as created_at,
       GREATEST(ru, au, cu) as updated_at
FROM total_combined_dates
```



created_at	updated_at
2023-01-05	2023-12-28
2023-01-06	2023-12-28
2023-01-16	2023-12-27
2023-01-07	2023-12-19
2023-01-08	2023-12-31

Handling Null Values

- For strings that have missing values and can not be determined based on other data, “N/A” was filled in.
- For JSON strings that have missing values, [] was used
- For the do_not_call boolean, False was filled in for missing values to maintain consistent data types
- -1 was filled in to missing integer types

Final Cleanup In Python
(See comments in code for explanation of Steps)

- In the last few assets, cleanup was done to get the correct types in each cell for each null value. This is done so that the final csv can be pulled in from elsewhere (some other system into another database), and not need to be processed to fit the datatypes of the new table.’
- In the extrapolate data asset, missing values were filled in that could be inferred from other data. For example if we have 2 records:
  - 1) Name: “Full Name”, Company Name: **“SemiConductors Inc.”**Country: **“USA”,** Company Employees: **1000**, Company Revenue: **1000000**, Company Industry: **“Semi Conductors”**
  - 2) Name: “Full Name 2”, Company Name: **“SemiConductors Inc.”**, Country: **“N/A”**, Company Employees: **-1**, Company Revenue: **-1**, Company Industry: **“N/A”**
  - We can fill in Country, Company Employees, Company Revenue and Company Industry for the second record since we can infer them from the first record that has the same company name
- Lastly, we merge records based on common name and phone number (limitations discussed in next slide)

Assumptions and Limitations I

- One assumption I made was that if a record with a name, email_address and ip_address were all the same, it represented the same individual.
- It is most likely the case that it represents the same individual, but a downside is that it is possible that it doesn’t and may result in a small amount of contacts being lost (This is shown on the next slide).
- Another assumption in the final post merge at the end of the pipeline is that two records with same name and phone number are the same contact. While likely, there may be a small amount of contacts lost.

Assumptions and Limitations II

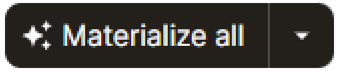
- Integer fields (like company revenue and company employees) was filled in with -1. The reason is to be consistent across data types, but anyone processing the final result will have to code into their system a conditional that processes -1 as a missing value
- Boolean values that are missing are filled in with “False”. This is inaccurate data but the tradeoff was used to have consistent data types (as opposed to putting “N\A” for a boolean field).

Assumptions and Limitations III

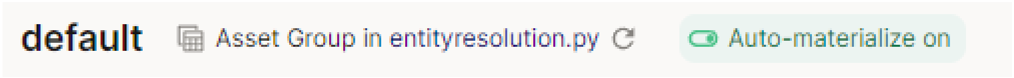
- Despite the fact that SQL join is faster than Pandas merge, there are limitations with using an RDBMS to begin with.
  - One limitation is the need to incorporate more tools and make the system more complex (by incorporating SQL Server, for example)
  - The other is that there is an overhead to make a call to the database from Pandas and vice verse. Moving data from the RDBMS and back is expensive.
  - The benefits of using SQL would be most apparent when using very large datasets to offset the overhead of moving data back and forth to and from the RDBMS.

Automatic running of data paths

- To run the pipeline you can either click “Materialize All” in the top right corner or wait until 45 seconds have passed and the pipeline will run automatically.



- For the first automatic run, the entire pipeline with all paths will materialize. Auto-materialize must be set to on (in the top right corner of the window

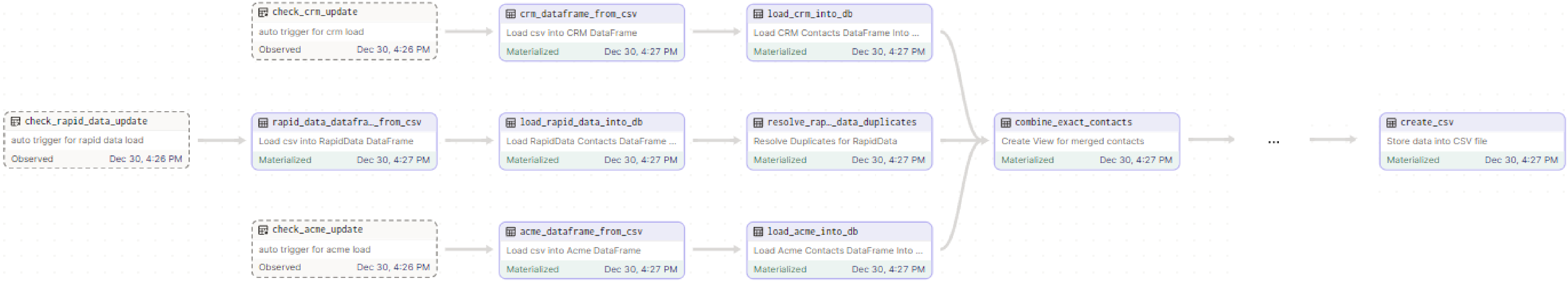


Automatic running of data paths II

- Each of the paths representing the input csv files are triggered using an @observable_source_asset annotation.
- For each subsequent trigger, only the paths in the pipeline that have data that needs to be modified will be materialized.
- For instance, if the acme__contacts.csv file was updated only, at the next trigger only acme_dataframe_from_csv and load_acme_into_db will be run before the SQL merging.
- The tables that were already loaded for the last run for RapidData and CRM will be used.
- There is no need to run those other assets in the pipeline again since they will just end up producing the same result regardless.

Automatic running of data paths III

In this example, acme__contacts.csv has been modified, and at the next check this path will be materialized



check_crm_update
auto trigger for crm load
Observed Dec 30, 4:26 PM

crm_dataframe_from_csv
Load csv into CRM DataFrame
Materialized Dec 30, 4:27 PM

load_crm_into_db
Load CRM Contacts DataFrame Into ...
Materialized Dec 30, 4:27 PM

check_rapid_data_update
auto trigger for rapid data load
Observed Dec 30, 4:26 PM

rapid_data_datafra..._from_csv
Load csv into RapidData DataFrame
Materialized Dec 30, 4:27 PM

load_rapid_data_into_db
Load RapidData Contacts DataFrame ...
Materialized Dec 30, 4:27 PM

resolve_rap..._data_duplicates
Resolve Duplicates for RapidData
Materialized Dec 30, 4:27 PM

combine_exact_contacts
Create View for merged contacts
Materialized Dec 30, 4:27 PM

...

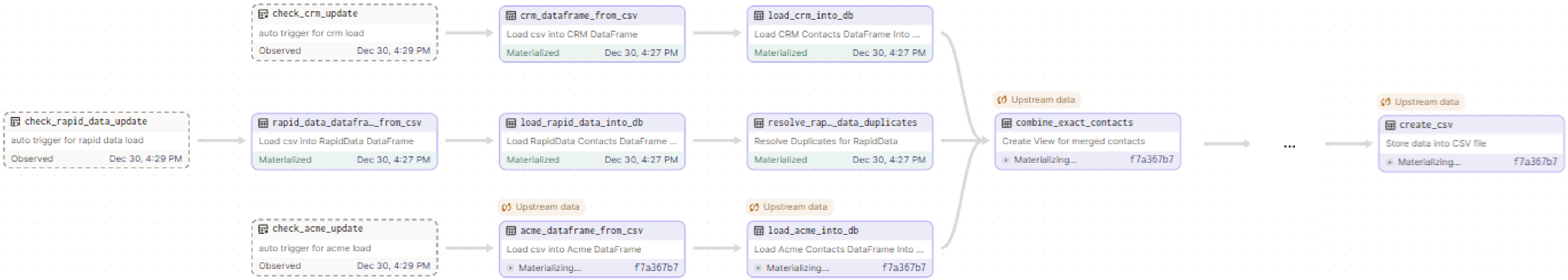
create_csv
Store data into CSV file
Materialized Dec 30, 4:27 PM

check_acme_update
auto trigger for acme load
Observed Dec 30, 4:26 PM

acme_dataframe_from_csv
Load csv into Acme DataFrame
Materialized Dec 30, 4:27 PM

load_acme_into_db
Load Acme Contacts DataFrame Into ...
Materialized Dec 30, 4:27 PM

The trigger occurs and only the path that has assets that need to be modified are re-materialized:



check_crm_update
auto trigger for crm load
Observed Dec 30, 4:29 PM

crm_dataframe_from_csv
Load csv into CRM DataFrame
Materialized Dec 30, 4:27 PM

load_crm_into_db
Load CRM Contacts DataFrame Into ...
Materialized Dec 30, 4:27 PM

check_rapid_data_update
auto trigger for rapid data load
Observed Dec 30, 4:29 PM

rapid_data_datafra..._from_csv
Load csv into RapidData DataFrame
Materialized Dec 30, 4:27 PM

load_rapid_data_into_db
Load RapidData Contacts DataFrame ...
Materialized Dec 30, 4:27 PM

resolve_rap..._data_duplicates
Resolve Duplicates for RapidData
Materialized Dec 30, 4:27 PM

Upstream data
combine_exact_contacts
Create View for merged contacts
Materializing... f7a367b7

...

Upstream data
create_csv
Store data into CSV file
Materializing... f7a367b7

check_acme_update
auto trigger for acme load
Observed Dec 30, 4:29 PM

Upstream data
acme_dataframe_from_csv
Load csv into Acme DataFrame
Materializing... f7a367b7

Upstream data
load_acme_into_db
Load Acme Contacts DataFrame Into ...
Materializing... f7a367b7

References

- [1]: https://www.python.org/
- [2]: https://pandas.pydata.org/
- [3]: https://www.microsoft.com/en-us/sql-server/
- [4]: https://dagster.io/
- [5]: https://www.scaler.com/topics/pandas/sql-vs-pandas/
- [6]: https://learn.microsoft.com/en-us/sql/relational-databases/native-client/odbc/creating-a-driver-application-multithreaded-applications?view=sql-server-ver16
- [7]: https://www.pragimtech.com/blog/sql-optimization/how-is-data-stored-in-sql-database/