DIGITAL ASSET RESEARCH

# Final Project: Inventory Analysis

UCLA Anderson School of Management
Data Management
**Group 2**

Ashwin Bharathwaj
Jiayi (Jeremy) Cao
Shirley Deng
Rick Kessels
Neville Lin
Charlotte Nouwen
Shreyeshi Somya
Xiaoxiao Ye

# List of Contents

# 1. Project Statement & Executive Summary

2009 marked the year in which the first cryptocurrency, Bitcoin, was established. A year later, it was the first cryptocurrency to be traded on the market. Rival cryptocurrencies, better known as “altcoins”, emerged as Bitcoin gained traction, aiming to offer either greater efficiency in terms of speed or anonymity. Currently, there are over 12,000 cryptocurrencies, and the market grew over 100% in the past year.

Digital Asset Research (DAR) was founded in 2017 with the aim to provide institutional quality research in the ever-evolving cryptomarket. DAR is a B2B company, selling their research reports and analyses on developments in the FinTech market to institutional clients, including but not limited to asset managers, hedge funds, tech firms and regulators.

DAR has been tracking all transactions in the cryptomarket since its establishment. This data is extremely valuable, not only in the sense that there is a record of the trades in history for ad hoc audit purposes, but perhaps more importantly, historical data allows for various econometric modeling techniques, which in turn can uncover the effect of a change in a variable (e.g. macroeconomic cycles, exogenous world events) on individual cryptocurrencies as well as the overall crypto market.

Recently, DAR’s marketing team has asked for clearer insights into the historical data at hand, so that they will be better able to sell DAR’s value proposition to potential clients. The problem is that the current data structure is only available to the Data Engineers and Data Scientists, and does not allow for a quick insight into the data inventory by the marketing team.

We have developed three dashboards that will provide the marketing team with the information they need to cater their marketing efforts to their intended audience. The first dashboard is a general overview of the market, and focuses on overall trends, including trading volume metrics, and top and bottom performers of the market for the chosen timeframe. The second dashboard is an asset drill down dashboard that allows the user to zoom in on a specific asset, and shows the key performance indicators of the chosen asset. The third dashboard shows the daily, weekly and monthly performance trends in terms of volume and price for a chosen asset and day.

The rest of the report is structured as follows. First, we will discuss DAR’s vetting process, as well as the structure of the S3 bucket that we were given access to. From here, we will introduce the two main tables that we used for the purposes of creating dashboards, and the variables that are included in these tables. We will then move onto the data handling, where we will discuss the data issues that we faced, the data cleaning that we did, and how we aggregated the data. Finally, we will discuss the dimension and fact tables that we created for our dashboards, introduce the newly created dashboards, and how we envision they can assist the marketing team in their daily operations.

## 2. Business context: DAR vetting process

Digital Asset Exchanges (DAX) are electronic platforms that expedite the process of trading digital assets or currencies. Over the past few years, some of these exchanges have been accused of inflating trading volume fraudulently. Since the market is relatively new, it is still rather loosely regulated, despite the fact that there are over 350 DAXs operating on a global level. This makes it difficult to determine which exchanges operate fairly and which prices rates are trustworthy.

DAR uses an objective exchange vetting process, built by FinTech professionals, to determine a so-called “clean” price. These prices are based on the actual exchanges between genuine buyers and sellers after filtering out the counterfeit exchanges and transactions.

There is a three-step procedure that DAR employs to determine which exchanges and players within the market can be trusted. The first step is intended to swiftly detect and delete any players within the market that are fake or inappropriate. This is done by looking at public reports and data to develop a sense of the liquidity, which represents the availability of liquid assets to the player or the market, domicile, which is the home-base of the player or market, and market and player behavior through data science modeling and screening. If an exchange clears these preliminary requirements, it is added to a so-called “watch list”.

This watch list is used for the second step, which comprises the comprehensive vetting process. Each exchange on the watch list is subjected to more data science assessments, which try to detect wash trading, which denotes buying and selling a certain asset in a short period of time to boost or decrease the price of the asset in order to mislead other market participants, spoofing, which denotes the act of criminals trying to fraudulently influence an asset’s price by placing counterfeit orders,  and other forms of price manipulation. DAR also investigates security requirements like regulatory compliance, adherence to anti-money laundering policies, and surveillance practices. The so-called “vetted list” includes only those exchanges that meet each requirement tested for in these two steps.

The third and final step includes an interaction of DAR with each exchange that is either on the vetted list or on the watch-list. Here, DAR promotes best practices, analyzes the exchanges regularly and tracks the requirements that might not be publicly visible. The goal of this final step is to agree to longer-term source contracts with the validated exchanges on the two lists. Every quarter, the selected exchanges are reviewed based on the previously mentioned requirements for watch-list and vetted exchanges. Only verified exchanges are then used for DAR’s pricing algorithm, which in turn results in the highly sought-after “clean” and reliable price that DAR is known for.

# 3. Data Literacy

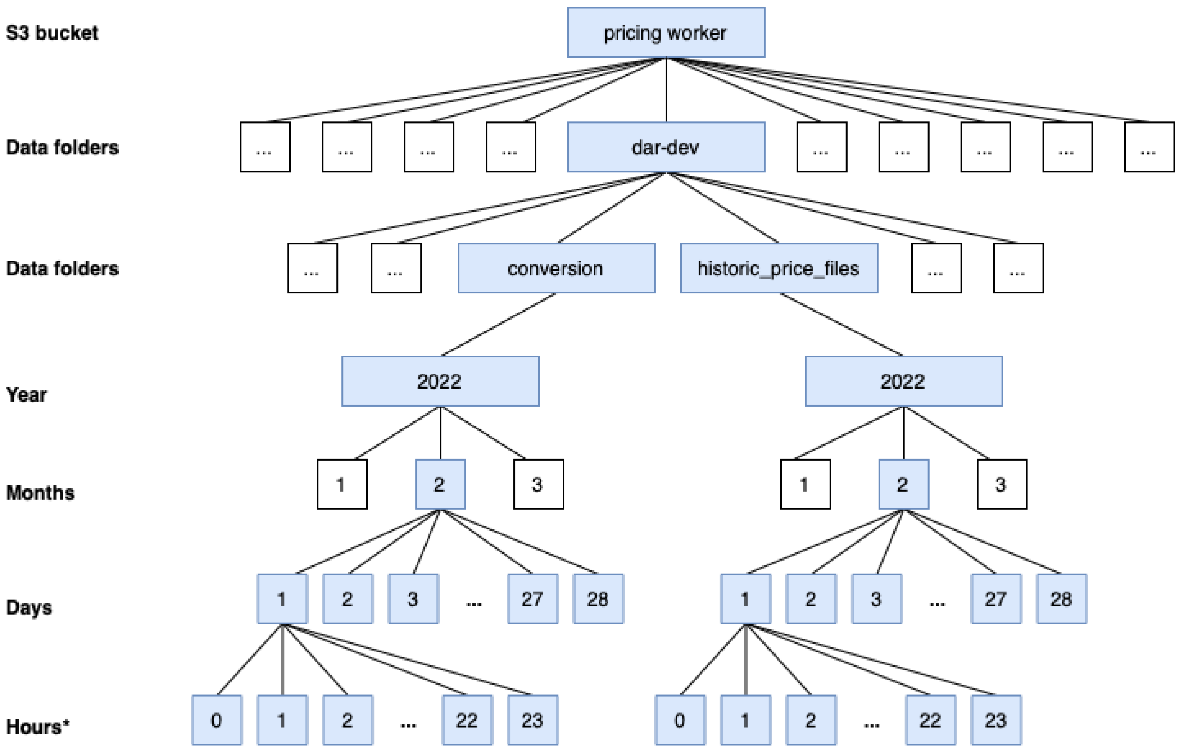
## 3.1 S3 Bucket Structure

We were given access to an S3 bucket with the name *pricing-worker*. The structure of this bucket is depicted in Figure 1.

The bucket contains ten folders and only one of these folders is relevant for us: *dar-dev*. Within the *dar-dev* folder, there are six folders that record different types of crypto-related events and developments. Two of these folders were relevant for us: *historic_price_file* and *conversion*. These two folders are both organized based on the date and time of data collection, as illustrated in Figure 1. The lowest level of folders represents the different days of data collection. Within these day-level folders, we can find one file for every 15 seconds of data collection.

The company explicitly instructed us to use the data from February 2022 only. The path towards this data is marked in blue in the image below.

**Figure 1: S3 Bucket Diagram**




* Every day-level folder contains 24 folders representing the different hours of data collection.

### 3.2 Data Structure

There are two main tables from the AWS S3 bucket that we used for the purpose of this project: (1) “full_window_price_v2” and (2) “conversion”.

**Figure 2: Data overview**

| full_window_price_v2 | |
|---|---|
| ticker | varchar(8) |
| methodology | varchar(12) |
| window_start | bigint(20) |
| window_end | bigint(20) |
| usd_price | double |
| usd_volume | double |
| effective_time | bigint(20) |
| price_id | varchar(20) |
| dar_identifier | varchar(20) |
| pricing_tier | varchar(5) |
| asset_name | varchar(50) |

| conversion | |
|---|---|
| ticker | varchar(8) |
| exchangeId | bigint(20) |
| pair | varchar(8) |
| currency | varchar(8) |
| price | decimal(10,0) |
| size | bigint(20) |
| rate | decimal(10,0) |
| usdPrice | decimal(10,0) |
| usdSize | decimal(10,0) |
| TStampTraded | decimal(10,0) |

*full_window_price_v2*

Each row in this table represents a 15-second window, that are consequently used to compute the definitive price (as denoted by usdPrice). Note that one row can contain zero, one or multiple transactions, conditional on the number of transactions that took place within the associated 15-second window. Each row contains information on the: (1) ticker, which is an identifier for the cryptocurrency, (2) methodology, which has the value “DAR” for each row, (3) window_start, which is the start of the 15-second window in unix time that DAR uses to determine the price, (4) window_end, which is the end of the previously mentioned window, (5) usd_price, which is the price calculated by DAR within the 15-second window, (6) usd_volume, which is the volume traded in the window, (7) effective_time, which denotes the time period in unix time in which the by DAR calculated price is applicable, (8) price_id, which is an identifier for the calculated price and consists of the ticker, the methodology in number format, and the effective time, (9) dar_identifier, which is the unique identifier used by DAR, (10) pricing_tier, which can take on values 1, 2, and 3, and (11) asset_name, which is the name of the cryptocurrency.

In the previous paragraph, we described that the pricing_tier variable can take on several values denoting the tier. The tier of an asset is determined by the exchanges used by DAR to calculate the price of the asset. As noted before, there are two main types of exchanges: (1) Watchlist exchanges and (2) Vetted exchanges. Both exchanges have to undergo four types of assessments: (1) A regulatory assessment, (2) a governance and institutional assessment, (3) a technical assessment, and (4) a comprehensive data science assessment. The difference between the two exchange classifications lies in the fact that an exchange needs to meet stricter requirements in the first three assessments in order to qualify for the Vetted exchange classification as compared to the Watchlist exchange classification.

Assets are allocated to Tier 1 or Tier 2. The price of Tier 2 assets are calculated from at least three Vetted or Watchlist exchanges, of which at least one is a Watchlist exchange. An asset can jump to Tier 1 classification if it was priced as a Tier 2 asset in the previous quarter, and can only have its price be determined by at least three Vetted exchanges. Tier 1 assets generally enjoy a high level of confidence in the quality of the calculated price. Conversely, there is a lower level of confidence in terms of the quality of its price for Tier 2 assets. Tier 3 assets are all assets that do not qualify for any of the first two tiers. For instance, all decentralized exchanges (DEXs) get the Tier 3 classification.

*conversion*

The conversion table contains all raw trades. Most of these trades went into the calculation of the price given in full_window_price_v2. Each row in conversion contains information on the: (1) ticker, which is an identifier for the cryptocurrency, (2) exchangeId, which is an identifier for the exchange that the trade took place on, (3) pair, which is the pair of assets traded, (4) currency, which denotes the currency of the trade, (5) price, which denotes the price of the trade, (6) size, which denotes the eligible volume of the trade, (7) rate, which denotes the exchange rate of the asset traded, (8) usdPrice, which is the price of the trade in US dollars, (9) usdSize, which is the size of the trade in US dollars, and (10) TStampTraded, which denotes the time stamp in unix time at which the trade took place.

The two tables, full_window_price_v2 and conversion, can be joined together conditional on the following requirements: (1) the tickers in the two tables have to be equal to each other, and (2) the TStampTraded in the conversion table needs to lie in the window denoted by [window_start, window_end] of the full_window_price_v2 table.

# 4. Data Handling

### 4.1 – Data Sources

We were given access to the S3 bucket on AWS. We connected the S3 bucket with our SingleStore accounts, and imported only the data from February 2022 by filtering the historic_price_file folder for the subfolders associated with this time frame. The query to import the data into one of our SingleStore warehouses took approximately 4 days. We conveniently shared the warehouse with the rest of the group members by adding them to the organization. This prevented everyone from having to run the same 4-day query. We provided the code used to connect SingleStore with the S3 bucket in the zip-file as well.

### 4.2 – Data Cleaning and Issues

Luckily, most of the data that was provided by DAR was already in a clean format. However, we discovered a small problem with regards to the variables usdPrice and usdSize from the original conversion table. Since the variables price, size, and rate from the same conversion table are used as an input to calculate the values for the variables usdPrice and usdSize, the latter two variables are, to some extent, "derived attributes". This means that the values of these two variables are derived from other variables in the database (i.e. price, size and rate). The exact exchange rate that was used in this calculation is different for most rows in the table, and therefore should still be included in the table overview as we are unable to directly derive the values of the "derived attributes" from the remainder of the variables in the table.

**Figure 3: SQL for data issue (pt. 1)**


From the screenshot above, we can see that out of the total of 324,677,842 rows that are in the conversion table, 284,622,548 rows have the same value for the price variable as for the usdPrice variable. This means that 87.66% of the rows in the conversion table have the same value for price as for usdPrice. Now, let us add a condition where the usdSize should be equal to the size.

**Figure 4: SQL for data issue (pt. 2)**

```sql
SELECT COUNT(*)
FROM conversion
WHERE usdPrice = price AND usdSize = size
```

Message Logs | SELECT COU... ×

Save as CSV

| COUNT(*) |
|---|
| 283840913 |

From here, we can see that there is only 283,840,913 rows that have both the same value for usdPrice and price, as well as for usdSize and size. This is 99.73% of the rows in our previous screenshot. Now, let us investigate the 0.27% (781,635 rows) of cases where the price variables are equal, but the size variables have different values.

**Figure 5: SQL for data issue (pt. 3)**

```sql
SELECT *
FROM conversion
WHERE usdPrice = price AND usdSize != size
```

Message Logs | SELECT * FRO... × | SELECT COU... × | SELECT COU... × | SELECT COU... ×

Save as CSV

2.404 s Row limit reached. Showing 300 rows of many

| ticker | exchangeId | pair | currency | price | size | rate | usdPrice | usdSize | TStampTraded |
|---|---|---|---|---|---|---|---|---|---|
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643673672 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674157 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674233 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674235 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674311 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674424 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674484 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643674635 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643675854 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643675884 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643675939 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643676005 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643676021 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643676254 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643676294 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643676699 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643679424 |
| feg | 0 | FEG_USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643680111 |
| apenft | 102 | NFT-USDT | USDT | 0 | 10000000000 | 1 | 0 | 9999999999 | 1643680260 |

We can see that the price variables have value 0, but the size variables have a very large value (10000000000 and 9999999999 respectively). We check this hypothesis by means of the following code:

**Figure 6: SQL for data issue (pt. 4)**

```sql
SELECT price, usdPrice, size, usdSize, currency, COUNT(*) as "count"
FROM conversion
WHERE price = usdPrice and size != usdSize
group by price, usdPrice, size, usdSize, currency
ORDER BY count DESC
```

Message Logs | SELECT price, ...

Save as CSV

| price | usdPrice | size | usdSize | currency | count |
|---|---|---|---|---|---|
| 0 | 0 | 10000000000 | 9999999999 | USDT | 781620 |
| 0 | 0 | 10000000000 | 9999999999 | ETH | 9 |
| 0 | 0 | 10000000000 | 9999999999 | USD | 6 |

Only 15 out of the 781,635 do not have the value USDT for the currency variable, and hence it might be worth investigating the USDT transactions to figure out where this error comes from. For convenience, we have included the full table where the price variables have value 0, but the size variables have these large, but different values, in *Appendix A*. DAR asked us to flag these anomalies, but also specifically asked us not to remove these anomalous data.

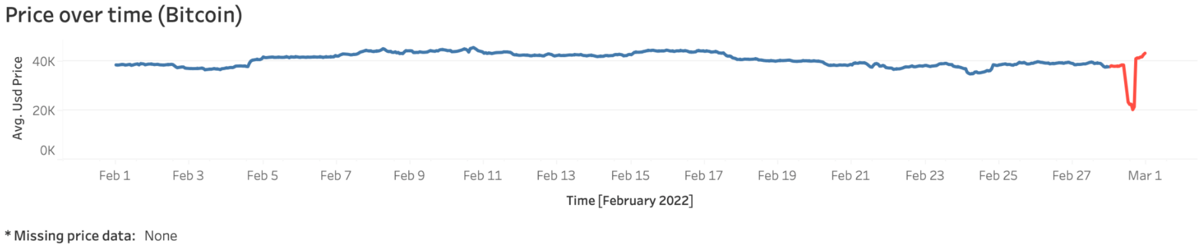
Besides cleaning the data and filtering out seemingly illegitimate transactions, we discovered the following three issues with regards to the data:

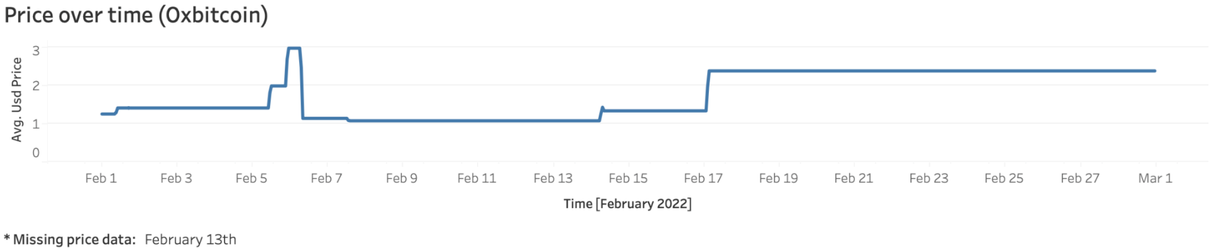
1. **Anomaly prices → Some assets have incorrect prices recorded on February 28th.** For instance, Bitcoin has some price below $1 recorded which is incorrect.
   ***Solution***: We created a calculated field that indicates whether or not a given asset has anomaly prices recorded for a certain day. We argue that an anomaly price is a price below 30% of the average monthly price. If an asset has anomaly prices for a given day, then data from this day will be marked in red as illustrated in *Figure 7*.

2. **Missing data → The prices for ~820 otherwise monitored assets were not recorded on February 13th between 10:45:30 and 11:09:30.**
   For instance, the price of the asset Oxbitcoin was not recorded during this period.
   ***Solution***: Our dashboard does not rely on the absence of missing values. If solely three prices are recorded in a given minute rather than four prices, then the average price for a given minute will be computed based on the three available prices. For instance, the average hourly price for Oxbitcoin would be computed based on the 45 minutes of available data between 10:00 and 11:00. Additionally, if there is missing data for a given asset, then our asset drill-down dashboard will flag this underneath the relevant graph as illustrated at the bottom of *Figure 7*. If there is no missing data for

February 13th, then the value of *missing price data* will be **none**. If there is missing data for that day, then the value for *missing price data* will equal **February 13th**.

**Figure 7: Flagging Anomaly Prices and Missing Data**



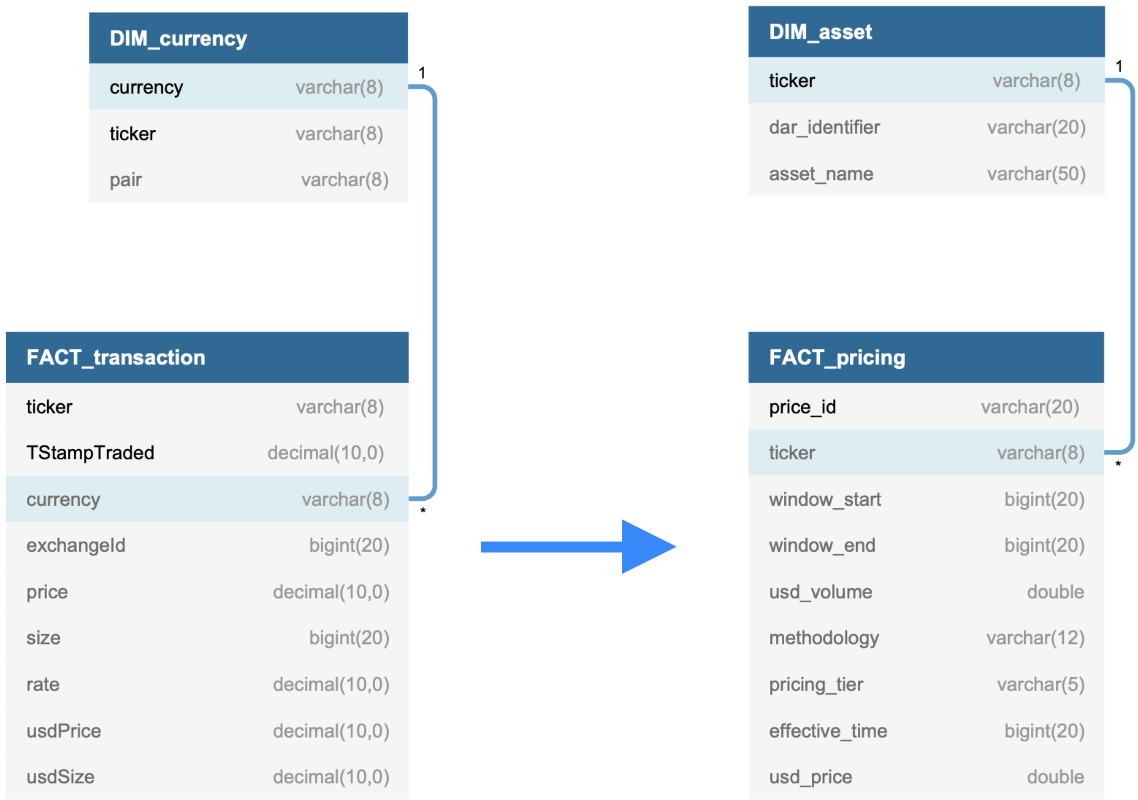

3. **Difficulties linking the two tables *Conversion* and *Full window price***
   Further, some issues were encountered when linking the two tables *Conversion* (table 1) and *Full window price* (table 2). Theoretically, the two tables could be joined by (A) ensuring that the ticker in table1 is equal to the ticker in table 2 and (B) the trading timestamp of ticker 1 is within the window [window_start, window_end] of table 2.
   However, when attempting this join in SingleStore, we encountered a memory error, and when attempting this join in Tableau, the query returned zero rows even though it was correct. An alternative solution involved *linking* the tables in Tableau rather than *joining* them. However, this method resulted in extremely slow dashboard updates when changing filter values. Thus, this solution was not feasible either.
   ***Solution***: In the end, we decided to join the tables but instead use two different sets of filters for (A) dashboard sheets based on *Conversion* and (B) dashboard sheets based on *Full window price* (See *Figure 10*).

4. **Inconsistent formatting in the Pair column of Conversion**
   Lastly, we encountered some issues when dealing with the *Pair* column of the *Conversion* table. The asset trading pairs in this table are not consistently formatted. For instance, a Bitcoin-USD trade is most frequently encountered as BTC-USD, but it could also be encountered as XBTUSD, BTC/USD, BTC.USD, btcusd, etc. This inconsistent formatting is present for not solely Bitcoin, but multiple assets.
   ***Solution:*** To minimize these issues, we cleaned the *Pair* column to be (1) uppercase for all assets and (2) to exclude any special characters (incl. / . – ). The only issue that we did not solve with this data cleaning is when an asset has multiple correct tickers (e.g. Bitcoin has two correct tickers: BTC and XBT). We could not solve this issue because this would require manually researching the correct tickers for each of DAR's ~1500 cryptocurrency assets.

# 5. Proposed Solution

## 5.1 Data modeling and dimensions

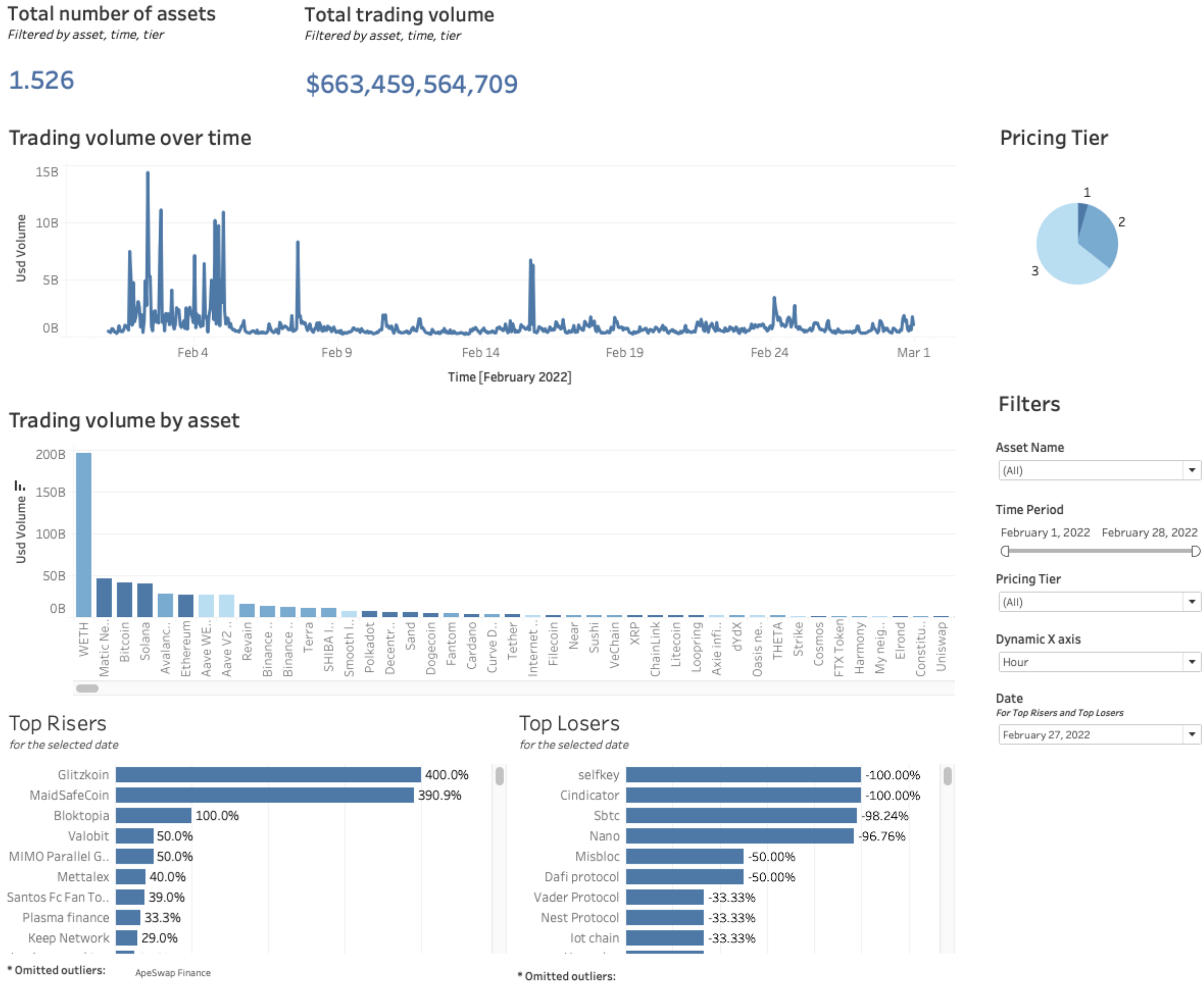
### Figure 8: Dimensional Modeling




Company representatives told us that the transactions in its associated facts table are used as an input for the pricing algorithm that determines the facts table for pricing. However, DAR uses sophisticated data science techniques and filtering to select a subgroup of the transactions, which consequently are used as an input for pricing the assets. Hence, not all transactions in the FACT_transaction facts table are used in determining the variables in the FACT_pricing facts table. Accordingly, there are no formal dimensional modeling links between FACT_transaction and FACT_pricing.

## 5.2 – Overall trading trends dashboard

**Figure 9: Overall trading trends dashboard**




### The overall goal of the dashboard

The overall goal of the overall trading trends dashboard is to provide a broad overview of the crypto pricing data. It gives information about the number of assets monitored, their popularity, their price trends and the reliability of the pricing data (i.e. pricing tiers). Further, the dashboard provides information about the overall trading volume that went into the calculation of asset prices. We can also use the filters on the right-hand side to zoom in on particular assets, time periods or pricing tiers.

**Dashboard Features**

| Feature | Description |
|---|---|
| Filters | All numbers portrayed in this dashboard will change based on which (combination of) asset(s) the user chooses, which time frame the user decides to investigate, and what tier the user wants to focus on. The user can also choose the unit that is on the X-axis in the graph “Trading volume over time”. This implies that the user can investigate aggregated daily trading volume, hourly trading volume, minutely trading volume, or trading volume for each 15-second window. Finally, the user is able to specify the date for the “Top risers” and “Top losers”. |
| Total number of assets | This number represents the total number of assets traded, conditional on the user-specified filters (i.e. asset(s), time frame and tier(s) chosen). |
| Total trading volume | This number represents the total trading volume in dollars, conditional on the user-specified filters (i.e. asset(s), time frame and tier(s) chosen). |
| Trading volume over time | This graph visualizes the total trading volume in dollars over time, conditional on the user-specified filters (i.e. asset(s), time frame and tier(s) chosen). |
| Trading volume by asset | This bar chart shows the total trading volume for the selected asset over the specified time frame in dollars. It is ranked from high to low and will be marked by different colors associated with the pricing tier of the asset. A darker blue indicates a higher pricing tier. |
| Top risers and Top losers | These bar charts represent the top and bottom assets in terms of their price development for the selected date. Assets with price changes exceeding 1000% are marked as outliers below the top risers/losers graph. If you hover over these outliers, a Tableau tooltip will indicate the price change associated with the outlier asset for the selected day. |

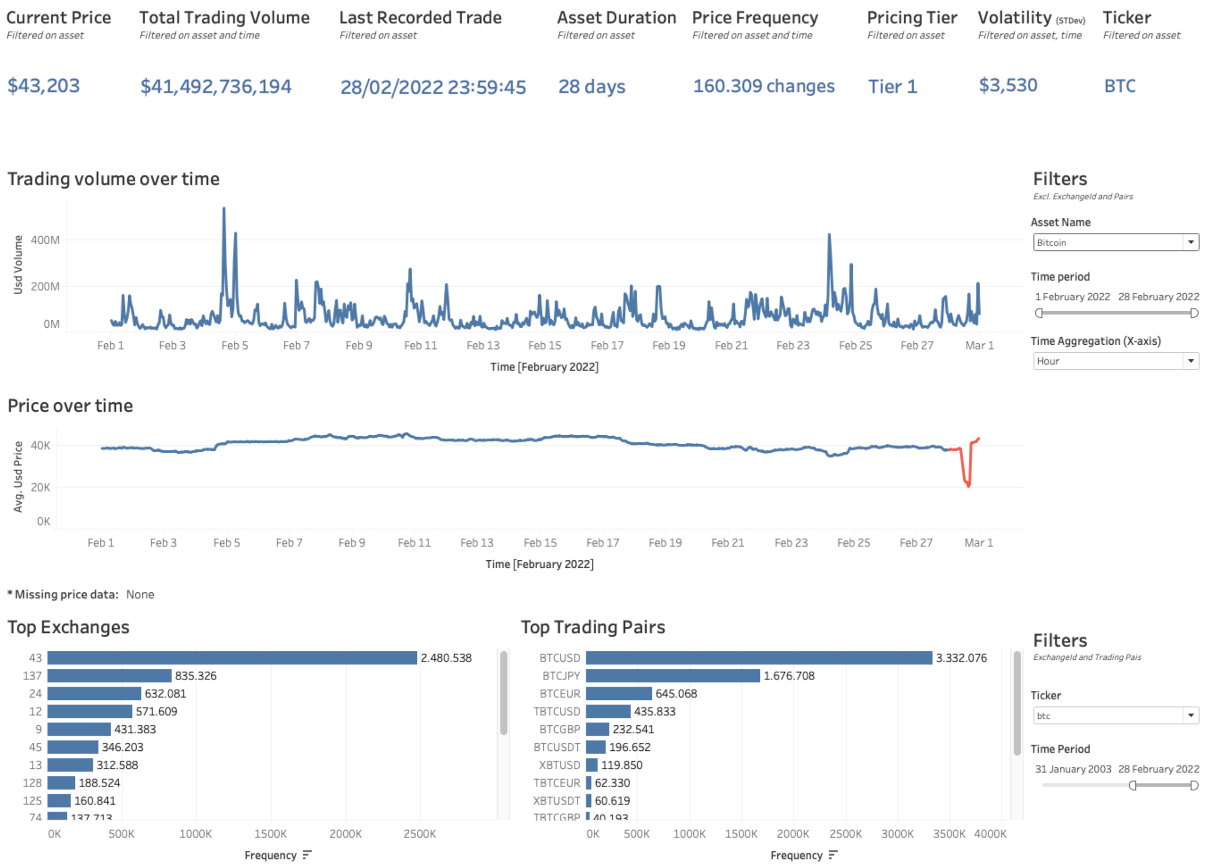
**Usefulness for the marketing team**

The filters allow the user to manually adjust the asset(s), monitoring period, and pricing tier(s) they want to focus on. This feature will grant flexibility to the marketing team to access different market data based on their current needs. The ease with which filters can be applied on this dashboard allows the non-technical marketing team to swiftly investigate the filtered data. This prioritization for simplicity from the development team made them develop single clicks and drags to activate a drop-down menu to select the asset or pricing tier and to choose the window of time to be investigated, respectively.

Another great feature of this dashboard is the date aggregation for the X-axis of the "Trading volume over time" and "Trading volume by asset". The options for the aggregation unit include days, hours, minutes or even seconds. This specific function can fulfill users' needs in ample ways: The user can view the trading volume and price development for every 15-second window, they can view the total trading volume for each hour, or they can view the average hourly price of the cryptocurrency over time.

### 5.3 – Asset Drill-down Dashboard

**Figure 10: Asset drill-down dashboard**




### The overall goal of the dashboard

The overall goal of the asset drill-down dashboard is to help the marketing team to have an overall view of a specific crypto asset's pricing data. It gives the most important information about the cryptocurrency in a clean overview. The current price of the asset, as well as the last recorded trade and pricing tier, are listed. Furthermore, the volatility of the price, which is a very important metric to DAR's clients, is provided on the dashboard. The time series shows the trading volume and price of the chosen asset over the time period chosen by the user in the associated filter menu. The bar charts at the bottom of the dashboard show the top exchanges on which this asset is traded, as well as the top trading assets that the chosen asset is traded against. Again, the asset and time period are user-specified in the filter menu on the right of the charts.

**Dashboard Features**

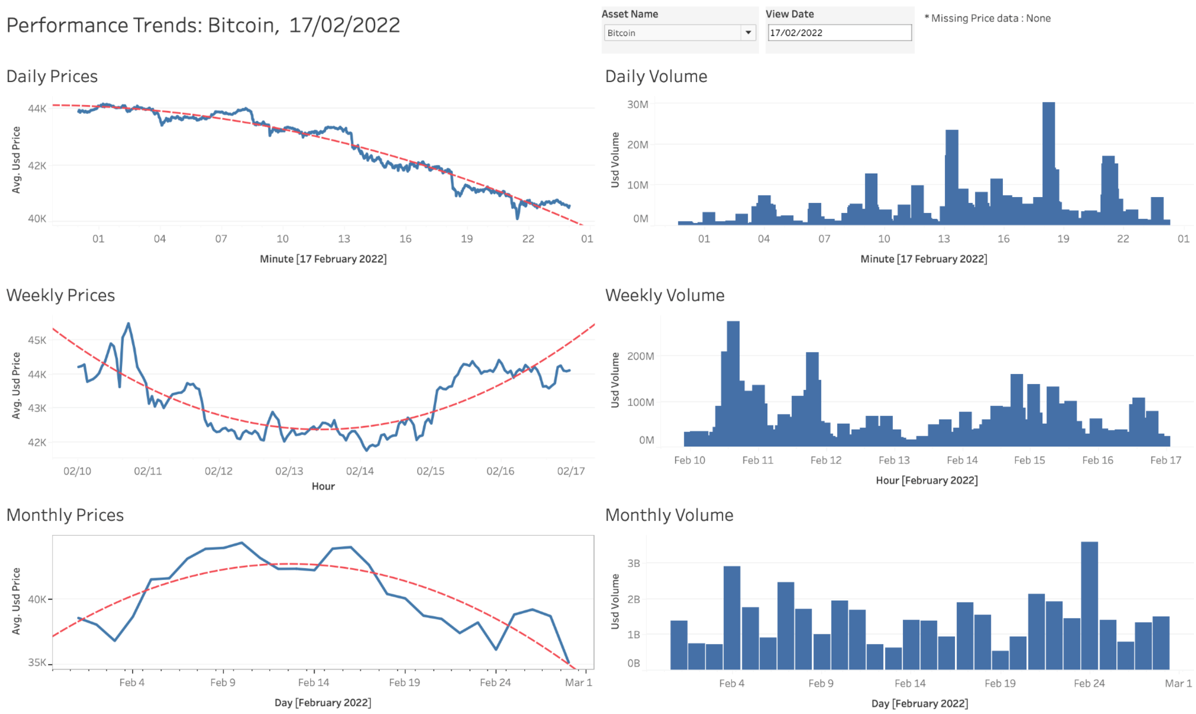
| Feature | Description |
|---|---|
| Key metrics and stats | ● The current price denotes the last recorded price of the chosen asset.<br>● The total trading volume metric shows the total trading volume that went into the calculation of the price of the asset, based on the time period chosen by the user.<br>● The last recorded trade denotes the time stamp in the format “DD/MM/YYYY HH:MM:SS” of the chosen asset.<br>● The asset duration represents the length of time (in days) that the selected asset was monitored by DAR in the given data set. If the asset was newly added at some point during February, then this value will be less than 28 days.<br>● The price frequency denotes the number of times the price of the chosen asset changed in the time window specified by the user.<br>● The pricing tier denotes the chosen asset’s pricing tier.<br>● The volatility represents the standard deviation in dollars of the asset’s price over the specified time window.<br>● The ticker is the identifying ticker symbol used by DAR to denote the asset. |
| Trading volume over time | This time series shows the total trading volume over the time unit chosen in USD volume, conditional on the user-specified filters. |
| Price over time | This time series shows the development of the average price over the time unit chosen in dollar units, conditional on the user-specified filters. In the section “Data issues”, we noted that DAR asked us to flag anomalies in the data, and we argued that an anomaly price is a price below 30% of the average monthly price. Here, we flagged anomalous prices by a red color, that denotes the day(s) on which anomalous prices were seen for the chosen asset. Additionally, we included a feature that shows whether we detected any missing pricing data. If there is any, then we denote the date(s) of the missing price data below the graph. |
| Top exchanges | The top exchanges is a bar chart that ranks those exchanges that had the highest number of trades (i.e. frequency) of the asset over the specified time period. Note that the exchanges are denoted by identifying numbers for DAR’s convenience, as these exchange IDs are also used by DAR internally. |
| Top trading pairs | The top trading pairs also is a bar chart and represents the other assets that the chosen asset is most frequently traded against. Note that the trading pairs are denoted by identifying strings of letters for DAR’s convenience, as these trading pair IDs are also used by DAR internally. |

**Usefulness for the marketing team**

This dashboard will allow the marketing team to zoom in on a specific asset and investigate the available pricing information. The metrics at the top of the dashboard answer DAR's provided business questions that are most relevant for the marketing team. Further, the marketing team can investigate any trends in trading volume and price development of the selected asset. Any anomalies in price are easily seen from the red coloring in the time series data, and the user is made aware of any missing price data as well. Furthermore, the top exchanges and top trading pairs allow the marketing team to quickly gain insight into what forms and where the asset is most frequently traded.

## 5.4 Performance Trends Dashboard

**Figure 11: Performance trends dashboard**



**The overall goal of the dashboard**

The overall goal of this dashboard is to show the performance trends of a chosen asset for a chosen date. It shows the daily, weekly and monthly price development and volume, and therefore allows for a quick overview of the asset's performance in recent times.

**Dashboard Features**

| Feature | Description |
|---|---|
| Daily prices | This time series shows the minute-specific development and daily trend of the chosen asset's price on the user-specified day. Note that we again included a feature that shows whether we detected any missing pricing data. If there is any, then we denote the date(s) of the missing price data in the top right corner of the dashboard. Additionally, for all pricing time series, days with pricing outliers are marked with a red-colored line segment. |
| Weekly prices | This time series shows the hour-specific development and weekly trend of the chosen asset's price in the seven days prior up until the user-specified day. |
| Monthly prices | This time series shows the day-specific development and monthly trend of the chosen asset's price in the month of the user-specified day. |
| Daily volume | This bar chart shows the minute-specific development and daily trend of the chosen asset's trading volume on the user-specified day. |
| Weekly volume | This bar chart shows the hour-specific development and weekly trend of the chosen asset's trading volume in the seven days prior up until the user-specified day. |
| Monthly volume | This bar chart shows the day-specific development and monthly trend of the chosen asset's trading volume in the month of the user-specified day. |

**Time trends**

The time trends as can be seen from the pricing time series are polynomial trend lines with degrees of freedom equal to 3, which allow for the flexibility of non-linear trends. These non-linear trends are appropriate given the general volatility of stock trends, and a polynomial with 3 degrees of freedom can follow the general non-linear stock without overfitting.

**Usefulness for the marketing team**

This dashboard will allow the marketing team to investigate a specific asset price and trading volume development over time. The trend lines in the time series allow for fast insights into the price development in the past day, week and month. For instance, in the example, we can see that there is a negative trend today, a general positive trend in the past week, but a negative trend in the past month. The red-colored line segments in the time series allow for quick notice of any anomalies in the price development, and the user will be able to see whether there is any missing data as well.

## 5.5. Grafana versus Tableau

Grafana and Tableau both are applications that mainly are used for data visualization. We will first discuss what Grafana and Tableau exactly are, and will then compare both applications in terms of how they best cater the efforts of the marketing team at DAR.

Grafana is an open-source application that is mainly used for data visualization. For metric monitoring specifically, Grafana offers an interactive dashboard, which allows for the users to tweak and adjust the dashboard in ways that offer them the most value. It is able to visualize time series metrics through graphs and other types of visualizations, and has decent support for time series database systems like OpenTSDB, InfluxDB, Prometheus and Graphite.

Tableau was created to aid people when it comes to understanding their data. Whereas Grafana might be more of a monitoring tool, Tableau offers its users Business Intelligence (BI). From Tableau, users can connect to almost any database (including SingleStore), create visualization by simple drag-and-drop actions, and share their charts and graphs swiftly with the rest of the organization.

*The User Interface*

Tableau Desktop has the advantage of having an easy-to-understand user interface, which allows even the less-skilled Tableau users to create meaningful data visualizations. Grafana is open-source, and in order for certain types of dashboards to work properly, plugins need to be installed. This makes the user interface of Grafana less accessible for the less experienced users. However, the online documentation allows users to gain skills with regards to Grafana quickly, and due to the open-source nature of Grafana, the community of the application allows for relatively easy troubleshooting.

*Connection to data sources & executing SQL queries*

Both Grafana, as well as Tableau, are able to connect to external data sources to query and retrieve data. Their servers query the data source for data, and the data source in turn provides the applications with the requested data in the query, and is consequently portrayed on the dashboard of the application. Additionally, both data visualization tools are able to execute SQL queries.

However, there is one big difference between Tableau and Grafana when it comes to connecting external data sources. To connect Grafana with SingleStore, there needs to be an export of SingleStore metrics to Prometheus. The latter can then be connected with Grafana to visualize the exported metrics. In Tableau, users can directly connect to SingleStore and load all the data that is in a SingleStore data warehouse into Tableau.

*Statistical analyses*

Tableau offers methods for statistical analyses within its application. Users gain the ability to better comprehend the data at hand by creating data summaries (e.g. histograms), trend analyses (including residual analyses), investigating the standard deviation of the data, investigating the moving averages of the data, comparing data by means of the percentile, and executing window- and reference calculations through functions like window_avg(), lookup() and previous_value(). Finally, Tableau offers the ability to integrate the statistical program R as well as Python. Grafana, on the other hand, was mainly designed to create proper and neat dashboards instead of doing statistical analyses, and therefore should not be used in case the firm values doing quick statistical analyses. However, there is a Grafana Machine Learning plug-in that can be bought in the "Pro" version that requires a monthly subscription.

*Our recommendation*

Based on the above considerations, we would recommend DAR to choose Tableau. Namely, with the easy user interface, the marketing team is able to immediately start investigating the data with our newly created dashboards, and perhaps create any additional visualizations to explore any anomalies or interesting facts portrayed in the dashboards. Furthermore, since Tableau is able to connect to SingleStore easily, there is no need to connect SingleStore to Prometheus first, after which only then Prometheus can be connected to the Grafana interface. Finally, the statistical analyses that can be performed in Tableau are very useful for the marketing team of DAR, since they will need to possibly be able to investigate and explain outliers to DAR's clients, but will not need to do any sophisticated machine learning analyses that are offered by Grafana.

# Appendix

### Appendix A: Conversion Outliers

| ticker | exchangeId | pair | currency | rate | count |
|---|---|---|---|---|---|
| **babydoge** | 0 | BABYDOGE | USDT | 1 | 153203 |
| **kishu** | 0 | KISHU_US | USDT | 1 | 123651 |
| **quack** | 0 | QUACK_US | USDT | 1 | 123004 |
| **kishu** | 6 | KISHU-US | USDT | 1 | 86969 |
| **luffy** | 0 | LUFFY_US | USDT | 1 | 49219 |
| **quack** | 124 | QUACK/US | USDT | 1 | 35154 |
| **luffy** | 144 | luffy_us | USDT | 1 | 33247 |
| **babydoge** | 71 | BABYDOGE | USDT | 1 | 30627 |
| **babydoge** | 48 | BABYDOGE | USDT | 1 | 22364 |
| **babydoge** | 102 | BABYDOGE | USDT | 1 | 22005 |
| **quack** | 71 | QUACK_US | USDT | 1 | 15371 |
| **kishu** | 71 | KISHU_US | USDT | 1 | 14714 |
| **kishu** | 102 | KISHU-US | USDT | 1 | 12976 |
| **babydoge** | 92 | babydoge | USDT | 1 | 10977 |
| **luffy** | 71 | LUFFY_US | USDT | 1 | 8338 |
| **apenft** | 102 | NFT-USDT | USDT | 1 | 7142 |
| **bezoge** | 0 | BEZOGE_U | USDT | 1 | 5755 |
| **feg** | 0 | FEG_USDT | USDT | 1 | 5238 |
| **luffy** | 92 | luffy_us | USDT | 1 | 5033 |
| **babydoge** | 22 | BABYDOGE | USDT | 1 | 3868 |
| **luffy** | 20 | LUFFY_US | USDT | 1 | 2396 |
| **babydoge** | 20 | BABYDOGE | USDT | 1 | 2356 |
| **bezoge** | 92 | bezoge_u | USDT | 1 | 2212 |
| **ghc** | 71 | GHC_USDT | USDT | 1 | 1096 |
| **kishu** | 22 | KISHUUSD | USDT | 1 | 801 |
| **talk** | 144 | talk_usd | USDT | 1 | 664 |
| **polydoge** | 102 | POLYDOGE | USDT | 1 | 457 |
| **safemars** | 0 | SAFEMARS | USDT | 1 | 321 |
| **babydoge** | 201 | BABYDOGE | USDT | 1 | 311 |

| | | | | | |
|---|---|---|---|---|---|
| **kishu** | 92 | kishu_us | USDT | 1 | 310 |
| **babydoge** | 6 | BABYDOGE | USDT | 1 | 237 |
| **ham** | 107 | HAM-USDT | USDT | 1 | 203 |
| **safemars** | 71 | SAFEMARS | USDT | 1 | 187 |
| **kishu** | 20 | KISHU_US | USDT | 1 | 155 |
| **polydoge** | 92 | polydoge | USDT | 1 | 155 |
| **ghc** | 14 | coin-usd | USDT | 1 | 108 |
| **elon** | 14 | coin-usd | USDT | 1 | 96 |
| **babydoge** | 144 | babydoge | USDT | 1 | 93 |
| **polydoge** | 71 | POLYDOGE | USDT | 1 | 90 |
| **babydoge** | 124 | BABYDOGE | USDT | 1 | 90 |
| **polydoge** | 106 | USDT_POL | USDT | 1 | 77 |
| **ham** | 92 | ham_usdt | USDT | 1 | 72 |
| **metapets** | 124 | METAPETS | USDT | 1 | 66 |
| **ghc** | 144 | ghc_usdt | USDT | 1 | 31 |
| **ham** | 144 | ham_usdt | USDT | 1 | 27 |
| **kuma** | 0 | KUMA_USD | USDT | 1 | 23 |
| **luffy** | 124 | LUFFY/US | USDT | 1 | 19 |
| **ghc** | 22 | GHCUSDT | USDT | 1 | 16 |
| **kishu** | 14 | coin-usd | USDT | 1 | 13 |
| **lnr** | 20 | LNR_USDT | USDT | 1 | 9 |
| **zinu** | 20 | ZINU_USD | USDT | 1 | 9 |
| **sos** | 0 | SOS_USDT | USDT | 1 | 8 |
| **happy** | 20 | HAPPY_US | USDT | 1 | 8 |
| **yooshi** | 0 | YOOSHI_U | USDT | 1 | 7 |
| **zinu** | 92 | zinu_usd | USDT | 1 | 6 |
| **apenft** | 76 | NFTUSD | USD | 1 | 6 |
| **ghc** | 16 | GHC-USDT | USDT | 1 | 6 |
| **quack** | 92 | quack_us | USDT | 1 | 5 |
| **ham** | 124 | HAM/USDT | USDT | 1 | 4 |
| **mc** | 144 | mc_usdt | USDT | 1 | 3 |
| **wsg** | 20 | WSG_USDT | USDT | 1 | 3 |
| **ghc** | 20 | GHC_USDT | USDT | 1 | 2 |
| **elon** | 106 | USDT_ELO | USDT | 1 | 2 |

| | | | | | |
|---|---|---|---|---|---|
| **quack** | 20 | QUACK_US | USDT | 1 | 2 |
| **elon** | 71 | ELON_USD | USDT | 1 | 1 |
| **kishu** | 7 | KISHU_ET | ETH | 2610 | 1 |
| **ghc** | 105 | GHC_USDT | USDT | 1 | 1 |
| **kishu** | 7 | KISHU_ET | ETH | 2780 | 1 |
| **gnt** | 92 | gnt_usdt | USDT | 1 | 1 |
| **ghc** | 126 | GHCUSDT | USDT | 1 | 1 |
| **luffy** | 71 | LUFFY_ET | ETH | 3159 | 1 |
| **fox** | 20 | FOX_USDT | USDT | 1 | 1 |
| **luffy** | 71 | LUFFY_ET | ETH | 2742 | 1 |
| **elon** | 90 | ELON-USD | USDT | 1 | 1 |
| **kishu** | 7 | KISHU_ET | ETH | 2724 | 1 |
| **elon** | 0 | ELON_USD | USDT | 1 | 1 |
| **luffy** | 71 | LUFFY_ET | ETH | 2636 | 1 |
| **babydoge** | 134 | babydoge | USDT | 1 | 1 |
| **kishu** | 7 | KISHU_ET | ETH | 2779 | 1 |
| **luffy** | 71 | LUFFY_ET | ETH | 2807 | 1 |
| **tonic** | 188 | TONIC_US | USDT | 1 | 1 |
| **kishu** | 7 | KISHU_ET | ETH | 2639 | 1 |