Data Review

The data is definitely in a great form. However, some inherent issues exist, in my opinion.

Event Data

- Subscription Created
  - Does not contain details of what subscription was purchased
    - This can be used to analyse trends in our types of used (car make, subscription type seeing the most growth, etc.)
    - Can be inferred from generated revenue, but it would be better to be more explicit
    - ***SOLUTION***: Add a field for the purchased subscription type
  - No way to distinguish between once-off hardware vs. software
    - The hardware and software values are in different files and in different forms
    - ***SOLUTION:*** Hardware to be included as an event with all the corresponding details, saving time with having to load and transform this data separately, and providing a more unified data structure
- Subscription cancelled (greatest area of gain)
  - We do not have a reason for the cancellation
  - This could have good business value add, i.e. we can analyse the possible reasons that users are cancelling their services and use this to improve our service offering - this could also be utilised to identify future features of note to implement/include and upcoming competition
  - ***SOLUTION:*** Initially we could come up with a list of possible cancellation reasons and extend as time progresses/as we learn (could be a static list with

the option to provide a custom reason) - implemented as basic questionnaire/survey when users are in the process of cancellation, to gain insight into their decision

- NOTE: Custom reason could pollute our data, so this should ideally be avoided, as we will need to tightly control the user input “Cost” vs “cost” vs. “product cost”
    - Free-text is always tricky, static list is better

```
Possible cancellation list:


Cost
Lack of Value
Limited Usage
Prefer competitor’s product
Duplicate Services
Change in Interests or Needs
Unsatisfactory Customer Service
Limited Features or Updates
Temporary Need
Life Changes
Dissatisfaction with Product/Service
```

- Subscription renewed
    - The subscription is renewed, but we have no way to independently link it to the subscription associated with the renewal
    - If the order ID is used to reference the renewal to the previous subscription, this would be fine, however, it may prove beneficial to have the data in the event
    - It would be beneficial to include the subscription type here as well, as we can then generate insights related to which subscriptions are being renewed, to show us which parts of the business are seeing consistent growth
        - ***SOLUTION***: Add a field for the purchased subscription type
        - NOTE: Again, this may be inferred from the generated revenue, however, explicit is better than implicit
    - There is no customer ID related to the renewed subscription, so the order_id would need to be traversed

        - ***SOLUTION:*** Include customer_id on all events, as this would improve data traversal
- Overall
    - Timestamp data seems misleading
        - Initially seems like a date, but is in epoch format
            - ***SOLUTION:*** Rename the “timestamp” to rather reference epoch, i.e. rename to “epoch_timestamp” and include a new field in date time format “timestamp” or “date_created”
                - NOTE: This could be ignored, but extra work done to convert the timestamp data to a consumable format for processing - I would recommend including the date for both brevity and to decrease computational work required for transformation step
    - Time events out of chronological order
        - Timestamp data is not in chronological order
        - ***SOLUTION:*** order by epoch
    - No way to distinguish between main item and add-ons
        - The subscription can be comprised of various add-ons and
        - ***SOLUTION:*** Amend the event data structure to include more details about the purchased subscription
            - NOTE: this would require some cross-functional team interaction, and the feasibility is highly dependent on the current system architecture
    - Inconsistent event data structure
        - The schemas don’t differ substantially, but I believe that it would be easier for data processing/traversal, if the schemas were more aligned, specifically referring to the omission of customer_id and revenue from renewal and cancellation events - this would make it easier to calculate revenue values as well
        - ***SOLUTION:*** Include the customer_id and revenue to all the events

```
Unified event schema with additonal fields

{
  "customer_id": "e9f0dbb6-90b5-4497-932a-57021895ce48",
  "event_type": "subscription_created",
  "order_id": "00006C1ED6CEA59D4FA7",
  "subscription_type": "premium"
  "subscription_items": {
    "make": "volkswagen",
    "tier": "premium",
    "inclusions": ["odb", "smart_mechanic", ...]
}
  "revenue": 33.33,
  "timestamp": 1642016441
}
```

Customer Data

- I don’t see much room for improvement here
- The provided data is not in the same format as our event data
  - NOTE: this is normal in data settings (that’s what data connectors are for), but it would make it easier to process the data if the data and file type were consistent across the areas (events, hardware, customer data)
  - ***SOLUTION:*** provide the data exports in same format (I vote for JSON)

Hardware Data

- I don’t see much room for improvement here
- The provided data is not in the same format as our event data
  - NOTE: this is normal in data settings (that’s what data connectors are for), but it would make it easier to process the data if the data and file type were consistent across the areas (events, hardware, customer data)
  - ***SOLUTION:*** provide the data exports in same format (I vote for JSON)