Data Table - Search PRD

Purpose

Enhance the modern data table with a search feature to allow users to quickly and easily find data rows of interest.

Background and Strategic Fit

One of the main pieces of functionality that is preventing the depreciation of the classic data table a functional equivalent of the classic search feature. Account Management has identified the need to quickly find labels & IDs without knowing their exact name or location as a major critical gap in modern data table functionality. Not only is quickly finding labels/IDs a major use case, but tracing the hierarchy of attributes attached to a label or ID is also a common use case. Closing this gap in functionality will allow us to move closer to sunsetting the classic data table. This initiative is important because supporting two data tables is a drag on infrastructure, development, and QA resources.

Links

Data Table Roadmap

Target Users

- [PRODUCT] Clients
- Account Managers (AM)
- Technical Account Managers (TAM)

Assumptions

- A full search of all attribute facets and archived values is technically infeasible.
- Searching by one attribute level at a time will be much more performant than searching all attributes every time.
- Data table filters will continue to persist when we change the attribute level.

Feature Usage

Data table search is ranked as the 5th most utilized feature in the past year of classic data table feature analytic.

The search functionality is often sited as one of three main driving forces of traffic the classic data table. The others being benchmarks comparisons and benchmarks download.

Use Cases

Use Case	Priority	Details
As any Moat Analytics user, I need to quickly find an ID or label in the data table without entering the specific hierarchy of attributes, so that I may quickly view the associated measurements.	Must Have	I usually know the attribute level of my label or ID, so searching all facets isn't critical, especially if it reduced performance. Sometimes I know the exact ID or label name that I am looking for. Sometimes I do not know the exact ID or label name that I am looking for, I would like to be provided a list of suggested results to choose from in this case.
As any Moat Analytics user, I need to adjust the table attribute view level after I've searched for an ID so that I can view measurements at different levels that are associated with my original ID or label.	Must Have	Once I have successfully searched for an ID or label, I can change the attribute view of the data table to trace the hierarchy of attributes for this ID or label.
As any Moat Analytics user, I need to have the ability to adjust the filter settings after I've surfaced an ID or label, so that I can expand or alter my data table view as needed for my research.	Must Have	Once I select a search result from the type ahead drop down, the result should be returned in the data table. The data table filters should reflect the selections that are required for this data table view. This way, I can expand or change my filter selection immediately after a search has been returned.
As any Moat Analytics user, I need to continue to be prompted by a drop down of filter selection options if I am unsure of what I am looking for, that way I may still use the filters in a traditional fashion.	Nice to Have	If I do not know the formatting of filter entries, such as advertisement size, the drop down will inform me.

Use Case	Priority	Details
As any Moat Analytics user, when I search for an ID or label that is not currently active, I should be informed that the selection is outside of my date range.	Nice to Have	If the ID is active within my currently selected time period, the measurements for those dates are returned when the search is completed. If the ID is active outside of my currently selected time period, a screen is returned prompting me to expand my date selection. - I am told the estimated campaign start and stop dates - I am also told the time period within which we hold data for this campaign.

Success Criteria

Success criteria	Priority	Details
Filter selections at any attribute or slicer level can be applied at any time.	Must Have	All attribute filters have been "unlocked" so the user may focus on any attribute filter in any order. Example Test Cases: - As a user I am able to filter the data table to a single level 4 attribute, even if the hierarchy of level 1, level 2, and level 3 items have not been specified. - As a user I am able to filter the data table to a multiple unrelated level 4 attributes without specifying hierarchy attributes. - As a user I can continue to filter the data table by any slicer, without regard to which attribute filters have or have not been selected. - As a user I can view the 4th attribute level of the data table then proceed to select a multiple filter options for the 2nd attribute level. - As a user I can filter the data table by both a level 2 selection and a level 4 selection then I may click apply.

Success criteria	Priority	Details
Filter drop down loads up to n (20?) initial results for the user to select from.	Must Have	When the user initially clicks on the filter box, a loading wheel is presented and *n* options should populate the filter drop down for selection. Attribute filter drop down options will be sorted descending by unfiltered impressions. Slicer filter drop down options will continue to be sorted alphabetically. Example Test Cases: - As a user I click on the level 1 filter box. I expect to see a loading icon while the the level 1 filter options with the most applicable impressions are returned to me. - Once the API call has returned with filter options, the loading icon will disappear and the drop down will be populated with up to 20 options - Initial attribute options should be sorted such that the result with the most unfiltered views is presented first - Initial slicer options should be sorted alphabetically.

Success criteria	Priority	Details
Filters that are already being applied continue to appear as "checked" in the filter drop down.	Must Have	On initial filter box focus, items that are currently selected appear above the n initial results. On type ahead filter, items that are currently selected that match the type ahead query will appear above the type ahead results. The user may un-check the box to remove this filter once the selection is applied via the apply button. Example Test Cases: - As a user I have already filtered the data table for the level 1 attribute "123" - this attribute has a high number of unfiltered views. I expect to see this option selected when I re-open the level 1 attribute filter. The drop down options that follow should be the 20 highest impression results. - As a user I have already filtered the data table for the level 1 attribute "123" - this attribute has a very low number of unfiltered views. I expect to see this option selected when I re-open the level 1 attribute filter. The drop down options that follow should be the 20 highest impression results. - As a user I have already filtered the data table for the level 1 attribute "123". When I type into the level 1 attribute filter "abc" I do not expect to see the "123" attribute returned as a selected option in the drop down. - As a user I have already filtered the data table for the level 1 attribute "123". When I type into the level 1 attribute filter "12" I expect to see the "123" attribute returned as a selected option in the drop down.

Success criteria	Priority	Details
Filter drop down options are no longer affected by existing filters.	Must Have	Filters that are currently applied to the data table have no effect on subsequent filter drop down results, no matter the attribute level. Filters that are not yet applied to the data table also have no effect on subsequent filter results, example: - filter for advertiser "123" - filter for creative starting with "A" - the creative list returns all creatives starting with "A", not the list that intersects advertiser 123 and creative "A"
Filter selections are implemented with "and" logic between attribute levels.	Must Have	For a row to be returned to the data table, the associated attributes must satisfy a filter selection made at each attribute level in which a selection is in place.
Filter selections are implemented with "or" logic within an attribute level.		For a row to be returned to the data table, the associated attribute must satisfy any of the selected filters for a specific attribute level.
If no results match filter selection, users are presented with a "*No Results Found"* screen.	Must Have	The no results found screen is detailed in the attached mockups
The view level of the data table is automatically adjusted to reflect the most granular filter selection made when the user clicks apply.	Nice to Have	Only applies to attributes Example Test Cases: - The data table is defaulted to be view level 1, A user successfully chooses and applies a filter for attribute level 3. The view level of the data table automatically changes to level 3 and the matching filter results are returned in the table. - The data table is being viewed at attribute level 4, no filters are applied. A filter is applied for attribute level 1, upon hitting apply, the view level of the data table automatically changes to attribute level 1.
Filters load n (20?) results for user selection on initial focus.	Must Have	Attribute results are sorted by descending unfiltered views, while slicer results are sorted alphabetically.

Success criteria	Priority	Details
Initially loaded attribute results are sorted by descending unfiltered views.	Must Have	
Initially loaded slicer results are sorted alphabetically.	Must Have	
The user may select a load next 100 option at the bottom of the suggested results drop down.	Nice to Have	Currently filter results are limited to 100 options. These 100 options are loaded when a user clicks on the filter box for the first time, there appears to be no method to determine exactly which 100 results are returned. The challenge presented is we cannot load all results from a large filter list without introducing performance issues. We would, however, like to automatically provide the user a list of potential options without requiring them to enter text for two reasons - 1. The user may select from this drop down list (currently the list is alphabetically sorted) 2. The user will understand how the data is formatted, informing their search to be more fruitful The proposed solution is to continue loading the initial result set of N options. - on first click load 6 results and number of results total, show 6 results and a "show me more button” - if user clicks show me more, load remainder of options (possible limitation for large datasets)

Success criteria	Priority	Details
Type-ahead searching implemented within will allow users to return relevant results beyond the initially loaded drop down.	Must Have	If a user types in a filter parameter in the attribute filter, query the database to match this parameter and return drop down of all results (see real time search success criteria) Examples: - I click the filter for advertiser in my data table filter modal, N results are found and displayed as when I click the filter box - I type "A" into the advertiser filter box - I am informed more data is being retrieved - All attribute filters that match "A" are returned (partial string matching) - I may select filter parameters from this drop down, or continue narrowing down my search by typing “App" - The advertiser database is queried again, the drop down list is narrowed to results that match the "App" search
Enable partial string matching	Nice To Have	Currently, searching filters only matches results from the start of a query, i.e. searching the filter for "kat" will not return the result "kit kat", but searching "kit k" will return the desired "kit kat". If partial string matching is enabled if the query matches any part of a string it is returned i.e. typing "kat" into the filter will return the selectable result "kit kat”.
Enable fuzzy matching	Nice To Have	Example - when searching for "Sam's Club" I should be able to type in "sams" and get a hit with the result
Fix multi-select filter results after selection	Nice To Have	Appears to be a bug in filter result selection, to reproduce: 1. Enter a filter search that has multiple results 2. Select one of the results 3. Notice that your result list has reset Expected behavior: Result list does not reset upon single selection, only resets once a new search is initiated or the current one is cancelled

Questions

Question	Outcome
Are there any other important use cases for the search feature outside of those listed in the PRD	The other major use case for search that we are choosing to not confront with this feature implementation is an archive search. Previously the search feature in classic would be used to access historical advertisement campaign measurements. This was accomplished by providing a modal containing relevant search results, estimated campaign activity dates, and date picker adjustments upon result selection. We are aiming to re-frame data table usage as a tool used to quickly check in on current measurements and troubleshoot campaign setup. Historical data should be accessed via API or export.
Are clients actually analyzing the activity timeline of specific data?	No, this is used in the selection process described above.
Is a client likely to search for a creative label or ID in hopes of returning information on the advertiser / campaign / line item?	This is not typical - when a user is searching for a specific ID they are looking for measurements pertaining to that attribute level. It may be considered a secondary, ad-hoc use case that a user would search for an ID then explore the related attributes.
Would type ahead filter be useful to the user, or is the ability to browse filter options more useful? Can we do both?	Implementing both a type ahead search and returning initial drop down options will provide the most value to the user. When the user is using the filters in a traditional sense in order to manipulate their current data table view, an initial drop down with relevant options is most useful. When the user knows what they're looking for very specifically, the type ahead filter is best suited.
Should level 1 filters affect the results returned in the level 2 drop down?	No, it would be difficult to engineer and QA the downstream, chained effects of filtering. For the MVP product filters should not affect selection, leaving it up to the user to make reasonable filtering decisions.
What should we do when there are more than 100 results in a filter drop down?	We show the initial list, and prompt the user to load the next 100 options that have been paginated in the API call. If the load more options feature is out of scope, we prompt the user to narrow their selection by typing in what they would like to filter.

Question	Outcome
Will we continue to persist filters once the data table view has been changed?	Yes, this is essential
How many options should be shown in the filter drop down on initial focus?	

User Interaction and Design

See mockups attached.

Not Doing

- Filters return results outside of currently selected timeline
- The activity timeline of a filter result is surfaced to the user
- Ability to select all and unselect all filter results
- It is clearly indicated that there are no relevant results in a filter query
- Current data table filter functionality will not be adjusted
- There will be no additional interfaces for users to interact with after a search has been prompted or initiated beyond the suggested results
    - No recent searches will be displayed
    - No hierarch will be called out
    - No activity timeline will be returned with the result

- Fully featured partial string matching will not be developed - you must search for more than "K" if you would like to return the brand "Kit Kat"