Title Filters

The **Title Filters** in Replyke allow developers to filter entities based on the content of their `title` property. This functionality is versatile and can be used to include or exclude entities with specific title characteristics. Let's explore how Title Filters work and how to use them effectively.

Overview of Title Filters

The `titleFilters` property can be passed into the `EntityListProvider` as an object with the following structure:

```
export interface TitleFilters {
  hasTitle?: boolean;
  includes?: string | string[];
  doesNotInclude?: string | string[];
}
```

- **`hasTitle`**: A boolean flag to include entities based on whether they have a title (`true`) or not (`false`).
- **`includes`**: A string or array of strings. Entities whose `title` contains any of these strings will be included.
- **`doesNotInclude`**: A string or array of strings. Entities whose `title` contains any of these strings will be excluded.

Dynamically Updating Title Filters

Developers can dynamically update the title filters using the `setTitleFilters` function provided by the `useEntityList` hook. This function expects a new `titleFilters` object or `null` to clear the filters.

```
const { setTitleFilters } = useEntityList();

// Set a new title filter
setTitleFilters({ includes: "React", doesNotInclude: "Angular" });

// Reset title filters
setTitleFilters(null);
```

How Title Filters Work

Title Filters are applied on the backend when querying entities. They allow for:

1. **Existence Filtering**: Using the `hasTitle` flag, you can include only entities that have a title (`true`) or exclude entities without a title (`false`).
2. **Text Matching**:
   - `includes` : Filters entities to include those whose title contains any of the specified strings.
   - `doesNotInclude` : Filters entities to exclude those whose title contains any of the specified strings.
3. **Combined Conditions**: Developers can combine these conditions to achieve granular control over the results.

Example Use Cases

1. Passing Static Filters to the EntityListProvider

To set static filters for titles at the initialization of the entity list, pass them directly to the `EntityListProvider` :

```
<EntityListProvider
  titleFilters={{ hasTitle: true, includes: ["React", "JavaScript"], doesNotInclude: "Ang
>
  <MyFeedComponent />
</EntityListProvider>
```

This setup ensures that only entities with a title are included, and those titles must contain either “React” or “JavaScript” while excluding “Angular.”

2. Dynamically Updating Filters Based on User Interaction

You can also dynamically adjust the title filters based on user actions, such as clicking a button:

```
const { setTitleFilters, entities, loadMore } = useEntityList();

const handleFilterUpdate = (newIncludes: string[], newExcludes: string[]) => {
  setTitleFilters({ includes: newIncludes, doesNotInclude: newExcludes });
};

return (
  <div>
```

```
      <h1>Filtered Titles</h1>
      <div>
        <button onClick={() => handleFilterUpdate(["React"], ["Angular"])}>
          Show React, Exclude Angular
        </button>
        <button onClick={() => handleFilterUpdate(null)}>Clear Filters</button>
      </div>
      <ul>
        {entities.map((entity) => (
          <li key={entity.id}>
            <h2>{entity.title}</h2>
          </li>
        ))}
      </ul>
      <button onClick={loadMore}>Load More</button>
    </div>
  );
```

In this example, clicking the button updates the title filters dynamically, allowing users to customize the entity list’s results.

Important Notes

- Title Filters are case-insensitive, making them flexible for user-defined queries.
- Filters are applied immediately, and the entity list resets to reflect the updated conditions.
- Passing `null` to `setTitleFilters` clears all title-based filtering.

Conclusion

Title Filters provide developers with powerful tools to fine-tune entity list results based on the content of entity titles. Whether setting static filters at initialization or dynamically updating them based on user input, Title Filters ensure your entity list presents the most relevant content. With options to filter by existence, inclusion, and exclusion, you can cater to diverse application needs effortlessly.

Last updated on May 7, 2025