

Location Filters

Overview

The Location Filters feature in Replyke allows you or your users to filter entities within a specified radius of a given geographic coordinate. This is particularly useful for applications that involve location-based content, such as local events, businesses, or nearby user posts. With Location Filters, users can narrow down their entity list (content feed) to display only entities relevant to a specific location.

How It Works

The `locationFilters` prop is used to define the geographic filter parameters. The filter expects an object of type `LocationFilters`, which contains the latitude, longitude, and radius (in meters) to define the area of interest.

```
interface LocationFilters {  
  latitude: number;  
  longitude: number;  
  radius: number;  
}
```

This object can be provided directly to the `EntityListProvider` component to apply static location filtering during initialization. To dynamically adjust the location filters after initialization, the `useEntityList` hook provides the `setLocationFilters` method.

```
setLocationFilters: (location: LocationFilters | null) => void;
```

- Setting the filters to `null` removes the location-based filtering.
- Providing valid `latitude`, `longitude`, and `radius` values updates the filter dynamically.

Example Usage

Static Location Filters

To initialize the entity list with location filters:

```
const locationFilters = {
  latitude: 37.7749,
  longitude: -122.4194,
  radius: 5000, // 5 km
};

<EntityListProvider locationFilters={locationFilters}>
  {/* Your content feed here */}
</EntityListProvider>;
```

Dynamic Location Filters

To change the location filter dynamically, use the `useEntityList` hook:

```
const MyComponent = () => {
  const { setLocationFilters } = useEntityList();

  const updateLocation = () => {
    setLocationFilters({
      latitude: 40.7128,
      longitude: -74.0060,
      radius: 10000, // 10 km
    });
  };

  const clearLocation = () => {
    setLocationFilters(null); // Removes location filters
  };

  return (
    <div>
      <button onClick={updateLocation}>Set New Location</button>
      <button onClick={clearLocation}>Clear Location Filters</button>
    </div>
  );
};
```

Notes

1. **Precision:** Ensure the latitude and longitude values are precise enough for the desired filtering. Slight inaccuracies in coordinates may result in unexpected results.
2. **Radius Limitations:** The radius value determines the size of the filtering area in meters. Be mindful of performance implications when using very large radii.
3. **No Location Filter:** If `setLocationFilters` is set to `null`, the system will ignore location as a filtering criterion when fetching entities.

Conclusion

Location Filters provide a powerful way to enhance entity lists' relevance by focusing on geographic proximity. Whether used statically during initialization or dynamically via the `useEntityList` hook, this feature enables users to tailor their content feeds for a highly localized experience. Combine it with other filters like Keywords or Title Filters for even more precise content curation.

Last updated on May 7, 2025