# Splunk Leaflet Maps - Debug Fix Summary

## ✅ Changes Completed

### 1. Comprehensive Console Logging Added

The visualization.js file now includes extensive console logging throughout the data processing pipeline:

#### In `formatData()` function:

- ✅ Logs the complete data object received from Splunk
- ✅ Logs all available field names
- ✅ Shows which field indices were found for lat, lon, description, category, and color
- ✅ Displays the first 3 rows in detail with coordinate parsing
- ✅ Tracks valid vs invalid rows
- ✅ Shows processing summary with category counts
- ✅ Provides detailed error messages when data is missing or invalid

#### In `updateView()` function:

- ✅ Logs when the function is called
- ✅ Shows the data and config received
- ✅ Tracks whether map initialization is needed
- ✅ Logs each step of the update process

#### In `_initializeMap()` function:

- ✅ Logs map container detection
- ✅ Shows when Leaflet map is created
- ✅ Tracks tile layer addition
- ✅ Confirms layer control setup

#### In `_addDataToMap()` function:

- ✅ Logs categories being processed
- ✅ Shows point counts for each category
- ✅ Displays first 2 points of each category with their colors
- ✅ Confirms when layers are added to the map

### 2. Enhanced Field Detection

Expanded the field name matching to handle more variations:

| Purpose | Supported Field Names |
|---|---|
| **Latitude** | `latitude`, `lat`, `Latitude`, `_geo_lat` |
| **Longitude** | `longitude`, `lon`, `lng`, `Longitude`, `_geo_lon` |
| **Description** | `description`, `desc`, `Description`, `name`, `Name` |
| **Category/Layer** | `category`, `type`, `Category`, `Type`, `layer`, `Layer` |
| **Color** | `color`, `Color`, `colour`, `Colour` |

### 3. Better Error Messages

Instead of generic “No data available”, you now get specific errors:
- ✅ “No data available - data object is null”
- ✅ “No data available - fields are missing”
- ✅ “No data available - rows are missing”
- ✅ “No data available - no rows returned from search”
- ✅ “Required fields not found. Available fields: [list of fields]”
- ✅ “No valid data points found. All rows had invalid coordinates.”

### 4. Custom Color Support

- ✅ Added support for `color` field in data
- ✅ Points can now have individual colors
- ✅ Falls back to category default color if no custom color provided
- ✅ Color changes are logged in the console

### 5. Fixed OpenStreetMap Tile Layer

- ✅ Corrected tile server URL to use proper OpenStreetMap tiles
- ✅ Changed from static image to dynamic tile server

### 6. Created Comprehensive Testing Guide

- ✅ Added TESTING_GUIDE.md with detailed instructions
- ✅ Includes multiple test scenarios
- ✅ Provides troubleshooting steps
- ✅ Explains how to interpret console logs

## 📊 What the Console Logs Will Show

When you run a search with the visualization, you’ll now see output like this:

```
=== updateView called ===
Received data: {fields: Array(4), rows: Array(3)}
Received config: {...}
Calling formatData...
=== formatData called ===
Data object: {fields: Array(4), rows: Array(3)}
Data keys: ["fields", "rows"]
data.fields: [{name: "latitude"}, {name: "longitude"}, {name: "description"}, {name: "
layer"}]
data.rows: [[40.7128, -74.006, "New York City", "city"], ...]
Number of fields: 4
Number of rows: 3
Field names: ["latitude", "longitude", "description", "layer"]
Field indices found:
  latIndex: 0 (field: latitude)
  lonIndex: 1 (field: longitude)
  descIndex: 2 (field: description)
  categoryIndex: 3 (field: layer)
  colorIndex: -1 (field: NOT FOUND)
Sample row (first row): [40.7128, -74.006, "New York City", "city"]
Row 0: [40.7128, -74.006, "New York City", "city"]
  lat (latitude): 40.7128 -> parsed: 40.7128
  lon (longitude): -74.006 -> parsed: -74.006
Processing summary:
  Valid rows: 3
  Invalid rows: 0
  Categories found: ["city"]
    city: 3 points
=== formatData completed successfully ===
formatData returned: {city: Array(3)}
Map not initialized, initializing now...
_initializeMap called
Map container found: true
Creating Leaflet map...
Map created: {...}
Adding tile layer...
Tile layer added
_initializeMap completed
Adding data to map...
_addDataToMap called with data: {city: Array(3)}
Categories to process: ["city"]
Processing category 'city' with 3 points
  Default color for 'city': #FF6B6B
    Point 0: {lat: 40.7128, lon: -74.006, ...} color: #FF6B6B
    Point 1: {lat: 34.0522, lon: -118.2437, ...} color: #FF6B6B
  Added 3 markers to layer group for 'city'
  Layer 'city' added to map (visible)
_addDataToMap completed
Updating layer controls...
=== updateView completed ===
```

## 🔍 How This Helps Debug

### Before (Old Behavior):

- Shows “Error: No data available”
- No information about what went wrong
- No visibility into the data Splunk is sending
- Hard to diagnose field name mismatches

### After (New Behavior):

- **See exactly what data Splunk sends** - You can verify the data structure
- **Know which fields are detected** - Shows if latitude/longitude fields are found
- **Track data processing** - See how many valid vs invalid rows
- **Get specific error messages** - Know exactly what's missing or wrong
- **Verify categories** - See what categories were created and how many points
- **Confirm map creation** - Know if the map initialized successfully

## 🚀 Next Steps for Testing

1. **Pull the latest changes in Splunk:**
   `bash`
   `cd $SPLUNK_HOME/etc/apps/leaflet_maps_app`
   `git pull origin main`

2. **Restart Splunk:**
   `bash`
   `$SPLUNK_HOME/bin/splunk restart`

3. **Open browser developer console** (F12)

4. **Run a test search:**
   `spl`
   `| makeresults`
   `| eval latitude=40.7128, longitude=-74.0060, description="New York City", layer="city"`
   `| append [| makeresults | eval latitude=34.0522, longitude=-118.2437, description="Los Angeles", layer="city"]`
   `| table latitude longitude description layer`

5. **Check the console logs** to see:
   - What data is being received
   - Which fields are found
   - If any errors occur and why
   - How many markers are created

6. **Review the TESTING_GUIDE.md** for more test scenarios and troubleshooting

## 📝 Commit Information

**Commit**: `5c947da`
**Branch**: `main`
**Status**: ✅ Pushed to GitHub

**Files Changed:**
- `appserver/static/visualizations/leaflet_map/visualization.js` (352 insertions, 15 deletions)
- `TESTING_GUIDE.md` (new file)

## 🎯 Expected Outcomes

After applying this fix:

1. **If data is being received but not displayed:**
   - Console logs will show the exact data structure
   - You'll see which fields are being matched
   - You'll know if coordinates are valid
   - You can adjust field names accordingly

2. **If no data is being received:**
   - Console will clearly state "data is null or undefined"
   - You'll know the issue is with the search or Splunk configuration

3. **If fields don't match:**
   - Console will list all available fields
   - Console will show which field indices were found
   - You can see what field names Splunk is actually using

4. **If coordinates are invalid:**
   - Console will show how many valid vs invalid rows
   - First 3 rows will be logged with coordinate parsing
   - You'll see which rows are being skipped and why

## 💡 Key Improvements

| Issue | Solution |
|---|---|
| No visibility into data | Comprehensive console logging |
| Generic error messages | Specific, actionable error messages |
| Limited field name support | Expanded field name matching |
| No color customization | Added per-point color support |
| Hard to debug | Step-by-step process logging |
| Wrong map tiles | Fixed OpenStreetMap URL |

## ✨ Additional Features

- **Valid/Invalid row tracking** - Know how many points were successfully processed
- **Category summary** - See all categories found and point counts
- **Sample row logging** - Inspect first few rows in detail
- **Field availability** - See all fields in your data
- **Processing confirmation** - "formatData completed successfully" message

---

**Status**: ✅ All changes committed and pushed to GitHub
**Ready for**: Testing in Splunk environment
**Documentation**: See TESTING_GUIDE.md for detailed testing procedures