

Port Charlotte Update and Location Switching Fix

Date: January 5, 2026

Overview

This update addresses two key issues:

1. Port Charlotte data refresh with new customer data (57 accounts as of 12/30/2025)
 2. Fixed Revenue Analysis map not refreshing when switching between locations
-

Issue 1: Port Charlotte Data Update

Problem

- Port Charlotte had outdated data
- User provided updated Excel file with 57 current accounts
- ZIP boundaries were properly fetched but data files needed regeneration

Solution Implemented

1. Geocoded New Customer Data

Script: Python geocoding script using Google Maps API

- **Input:** Customer List Port Charlotte as of 12 30 2025.xlsx
- **Output:** /nextjs_space/public/portcharlotte-route-assignments.json
- **Results:**
 - Successfully geocoded all 57 accounts (100% success rate)
 - All addresses have precise latitude/longitude coordinates
 - Includes pricing data (monthly and yearly)

2. Fetched ZIP Boundaries

Script: fetch_portcharlotte_boundaries.js

- **Source:** Florida ZCTA GeoJSON from Census Bureau
- **Output:** /nextjs_space/public/portcharlotte-zip-boundaries.json
- **Results:**
 - Successfully fetched all 18 unique ZIP code boundaries
 - 1.11 MB file with complete polygon data
 - No missing ZIPs

3. Regenerated Data Files

Script: Python data aggregation script

Generated three data files:

a) portcharlotte-route-assignments.json

- 57 customer records

- All accounts geocoded successfully
- Fields: customerNumber, displayName, address, city, state, zipCode, territory, technician, dayOfWeek, latitude, longitude, monthlyPrice, yearlyPrice

b) portcharlotte-zip-revenue-data.json

- 18 ZIP codes
- Fields: zipCode, accountCount, monthlyRevenue, yearlyRevenue, averageMonthlyRate, averageYearlyRate, territory, technicians
- Total accounts: 57
- Total monthly revenue: \$7,960.00
- Total yearly revenue: \$95,520.00

c) portcharlotte-density-data.json

- 18 ZIP codes
- Fields: zipCode, area, territory, activeCount, terminatedCount, totalHistorical, churnRate
- All accounts marked as active (no historical churn data yet)

Port Charlotte Data Characteristics

- **Technicians:** 2 (Jovan Rodriguez, Max Carlini)
- **Territory:** All accounts in "All Pool - Outlying Maintenance"
- **Pricing:** Mostly \$140/month (with some at \$130 and \$150)
- **ZIP Codes:** 18 unique ZIPs spanning Port Charlotte, Punta Gorda, Englewood, North Port, Venice, Placida, and Rotonda West areas

Issue 2: Revenue Analysis Location Switching

Problem

When switching locations in the Revenue Analysis view (e.g., from Miami to Orlando), the previous location's territory filters and map state persisted on screen. Users had to manually refresh the browser to see the correct data.

Root Cause

The `useEffect` in `location-revenue-analysis.tsx` had an empty dependency array `[]`, meaning it only executed once when the component mounted. When the `location` prop changed, the component did not reload data or reset state.

Solution Implemented

File: `/nextjs_space/components/location-revenue-analysis.tsx`

Changes:

1. Added `location` and `config.name` to the `useEffect` dependency array
2. Added state reset logic at the beginning of the `useEffect`:

```
``typescript
useEffect(() => {
  // Reset map state when location changes
  setLoading(true);
  setError(null);
  setSelectedZip(null);
```

```
setHoveredZip(null);
setSortOrder('none');
```

```
// ... rest of data loading logic
```

```
}, [location, config.name]);
...

```

What Gets Reset:

- `loading` : Set to true during data fetch
- `error` : Cleared before new data load
- `selectedZip` : Clears any selected ZIP InfoWindow
- `hoveredZip` : Clears hover state
- `sortOrder` : Resets table sorting
- `territoryFilter` : Regenerated based on new location's territories
- `zipRevenue` : Loaded from new location's data file
- `boundaries` : Loaded from new location's boundary file

Benefits

- Seamless location switching without browser refresh
- Clean slate for each location's data visualization
- Prevents UI confusion from stale data
- Automatic map center and zoom adjustment per location

Files Modified

Data Files (Generated)

1. `/nextjs_space/public/portcharlotte-route-assignments.json` - 57 accounts with geocoded locations
2. `/nextjs_space/public/portcharlotte-zip-revenue-data.json` - Revenue aggregation by ZIP
3. `/nextjs_space/public/portcharlotte-density-data.json` - Density metrics by ZIP
4. `/nextjs_space/public/portcharlotte-zip-boundaries.json` - ZIP boundary polygons (18 ZIPs)

Code Files (Modified)

1. `/nextjs_space/components/location-revenue-analysis.tsx`
 - Line 116-159: Modified `useEffect` to include location dependency and state reset logic

Testing Results

TypeScript Compilation

✓ No type errors (exit_code=0)

Production Build

✓ Successful build

- All routes compiled successfully

- 10 static pages generated
- No build errors or warnings

Port Charlotte Functionality

- ✓ All data files verified:
- 57 accounts fully geocoded
- 18 ZIP codes with complete boundary data
- Revenue totals match expected values (\$7,960/month)

Location Switching

- ✓ Fixed - Revenue Analysis now automatically:
- Reloads data when location changes
- Resets all map state
- Updates territory filters for new location
- Prevents stale data display

Deployment Status

Build: ✓ Successful (exit_code=0)
Dev Server: ✓ Running on localhost:3000
Ready for: Production deployment

The application is now ready for use with:

- Updated Port Charlotte data (as of 12/30/2025)
- Automatic map refresh when switching locations
- All 5 locations fully functional (Phoenix/Tucson, Jacksonville, Dallas, Orlando, Miami, Port Charlotte)

Future Considerations

1. **Port Charlotte Historical Data:** As terminated accounts accumulate, update the density data to reflect churn rates
2. **Territory Expansion:** If Port Charlotte adds new territories beyond “All Pool - Outlying Maintenance”, update the territory color mapping
3. **Data Updates:** Process provided quarterly or as business needs require updated customer lists