You're bumping into two separate issues:

What to change (frontend)

1) Ask the backend for a fallback to the latest snapshot

In `emissionsApiService.getEmissionSummary`, include `preferLatest=true` so that the server will return the most recent snapshot when the specified period has no data:

```
// build query params
params.set('periodType', periodType);
params.set('year', year.toString());
if (periodType === 'monthly' || periodType === 'daily') params.set('month', month.toString());

// NEW:
params.set('preferLatest', 'true');
```

Also, add an automatic retry if you still get a 404 or a 200 with 0 datapoints:

```
const response = await axios.get(`/api/summaries/${clientId}?${params}`, { ... });

if (response.status === 200 && response.data?.success) {
  const data = response.data.data;
  // If server gave you an empty snapshot, re-try without year/month to force "latest"
  if ((data?.metadata?.totalDataPoints ?? 0) === 0) {
    const latest = await axios.get(`/api/summaries/${clientId}?periodType=${periodType}&preferLatest=true`, { headers, timeout: 1500
    if (latest.status === 200 && latest.data?.success) return latest.data.data;
  }
  return data;
}

if (response.status === 404) {
  const latest = await axios.get(`/api/summaries/${clientId}?periodType=${periodType}&preferLatest=true`, { headers, timeout: 15000,
  if (latest.status === 200 && latest.data?.success) return latest.data.data;
  return null;
}
```

Do the same `preferLatest=true` in your `checkConnection` helper so your "LIVE API" status stops flipping to "OFFLINE" just because the current month has no entries yet.

2) Fix node filter name so node views actually work

Right now you pass `{ node: selectedNode }`, but the backend expects `nodeId` (see the filter handler).

- In your `useEffect`, build `filters` like this:

```
const filters = {};
if (selectedNode) filters.nodeId = selectedNode; // ← not "node"
if (selectedScope) filters.scope = selectedScope;
filters.period = activePeriod;
```

- Or, inside `getFilteredEmissionSummary` before building `params` add:

```
if (filter.node && !filter.nodeId) filter.nodeId = filter.node;
```

This makes node-specific filtered calls hit the right code path.

Why you were seeing "0 / not found" on initial load

- The UI always queried **the current month** (`periodType=monthly&year=<thisYear>&month=<thisMonth>`).
- If no current-month entries existed yet, the backend recalculated and returned **zeros** or **404** instead of returning the **latest available snapshot** from the previous month. (That's how the handler is currently structured.)

- When a new data entry arrives, your "update summaries on data change" hook recalculates and then you see data (that part is fine).

With the fallback + "don't save empty snapshot" + correct sorting + small frontend tweak, the dashboard will always display **the most recent computed summary** until fresh period data arrives.

If you want, I can also provide a tiny banner in the UI that says "Showing latest available snapshot (Aug 2025)" when `metadata.fallbackFor` is present—so users know it's a past period.

- When a new data entry arrives, your "update summaries on data change" hook recalculates and then you see data (that part is fine).

With the fallback + "don't save empty snapshot" + correct sorting + small frontend tweak, the dashboard will always display **the most recent computed summary** until fresh period data arrives.

If you want, I can also provide a tiny banner in the UI that says "Showing latest available snapshot (Aug 2025)" when `metadata.fallbackFor` is present—so users know it's a past period.