

Header-to-Cell Reference Fix Analysis

Problem Summary

The validation system is failing because:

- `ColumnRef` now contains **header names** like "Heat Metered"
- But the code tries to build range addresses like `Range("Heat Metered59")`
- This fails because "Heat Metered59" is not a valid Excel range address

Debug Output Evidence

```
[DEBUG] AV_Validators :: Unable to build cell reference: Heat Metered59
[DEBUG] AV_Format :: Invalid column letter: Construction Date
[DEBUG] AV_Format :: WriteSystemTagToDropColumn ERROR: 1004 - Method 'Range' of object '_Worksheet'
failed
```

Files That Need Updating

1. AV_DataAccess.bas - ADD NEW HELPER

Need a function to get a cell from a table by header name and row:

```
vba

Public Function GetCellFromTableHeader(ws As Worksheet, _
    tableName As String, _
    headerName As String, _
    rowNum As Long) As Range
    ' Finds the column by header name and returns the cell at rowNum
End Function
```

2. AV_Engine.bas - UPDATE ValidateSingleRow

BEFORE (broken):

```
vba

Set TargetCell = wsData.Range(TargetColumnLet & rowNum)
```

AFTER (fixed):

```
vba

' Need to pass the target table ListObject to ValidateSingleRow
' Then use it to find the cell by header
Set TargetCell = GetCellFromTableHeader(wsData, tblName, TargetColumnLet, rowNum)
```

3. AV_Validators.bas - UPDATE GetSiblingCell

Same issue - trying to build "Heat Metered60" as a range address.

BEFORE (broken):

```
vba

Set GetSiblingCell = ws.Range(colLetter & cell.Row)
```

AFTER (fixed):

```
vba

' Use table-aware cell lookup
Set GetSiblingCell = AV_DataAccess.GetCellFromTableHeader(ws, tableName, headerName, cell.Row)
```

4. AV_Format.bas - UPDATE WriteSystemTagToDropColumn

Same issue with `dropColLetter` parameter.

BEFORE (broken):

```
vba  
  
Set cell = wsTarget.Range(dropColLetter & rowNum)
```

AFTER (fixed):

```
vba  
  
' Use table-aware lookup  
Set cell = AV_DataAccess.GetCellFromTableHeader(wsTarget, tableName, dropColHeader, rowNum)
```

Changes Cascade

The problem cascades because:

1. ValidateSingleRow needs the table reference
2. Validators need to pass table reference to GetSiblingCell
3. AddValidationFeedback needs table reference for WriteSystemTagToDropColumn
4. All functions need to know the table name

Recommended Fix Strategy

Option A: Pass ListObject/TableName Through Call Stack

- Requires updating function signatures everywhere
- Most complete solution
- Largest code change

Option B: Infer Table from Worksheet

- Add function to find the "main" ListObject on a worksheet
- Less invasive
- May fail if sheet has multiple tables

Option C: Hybrid Approach

- Store table name in config and access globally
- Use helper to get table from target sheet
- Moderate changes

Proposed Implementation

Use Option C - store table name globally after loading config:

1. In AV_Core, add: `Public CurrentTargetTableName As String`
2. In AV_Engine.ProcessValidationTarget: Set it before validation
3. In AV_DataAccess: Add `GetCellFromTableHeader` that uses global or passed table name
4. Update ValidateSingleRow, GetSiblingCell, WriteSystemTagToDropColumn

Quick Temporary Fix

For immediate testing, we could:

1. Update GetAutoValidationMap to read BOTH Letter AND Header
2. Store both in the dictionary
3. Use Letter when available, Header when in table mode

This maintains backward compatibility while the full fix is developed.

