Here is the response form HRD, Paul Reasor:

We agree to byte storage of the HWX and NAW reflectivity variables given proper metadata.

The metadata in the HWX and NAW CfRadial files *must* completely explain how to extract the correct reflectivity values. Given the byte format data, the metadata should contain all the necessary information to retrieve the float (correct) values. The data cannot be used by the research community otherwise. In the CfRadial 1.3 documentation, the following conversion between integer and float is given:

Float value = (integer value) * scale_factor + add_offset

Attributes "scale_factor" and "add_offset" are CfRadial 1.3 standards. L3 Harris should specify the scale_factor and add_offset attributes in the metadata *for each variable* where this is relevant, particularly the reflectivity variables DBZ and NAW_Range_Vector_Map.

Our guess (and L3 Harris can confirm) is that the relationship between the byte reflectivity data (idbz) and correct float values (dBZ) is:

dBZ = 48 + 2.6667*idbz

In this case, scale_factor should be set to 2.6667 and the add_offset should be set to 48 in the CfRadial attributes metadata. If the conversion from byte data to correct (float) data does not follow this standard, then L3 Harris needs to provide an explanation of how the byte data should be converted to correct (float) reflectivity values, and then determine how to convey this within the framework of the CfRadial 1.3 attributes metadata (we believe comments within attributes like "units" aren't an acceptable substitute, so the "resolution=0.375" should be removed as shown below).

Here are the present reflectivity attributes for HWX and NAW from an ncdump:

```
float DBZ(time, range) ;
        DBZ:long_name = "DBZ" ;
        DBZ:_fillValue = -9999.f ;
        DBZ:units = "dB (resolution=0.375)" ;

byte NAW_Range_Vector_Map(time, range) ;
        NAW_Range_Vector_Map:long_name = "Range_Vector_Map" ;
        NAW_Range_Vector_Map:_fillValue = 999.f ;
        NAW_Range_Vector_Map:sampling_ratio = 1.f ;
        NAW_Range_Vector_Map:coordinate = "time range" ;
        NAW_Range_Vector_Map:grid_mapping = "grid_mapping" ;
```

Here is our suggestion for what the attributes should look like if our current understanding is indeed correct:

```
byte DBZ(time, range) ;
        DBZ:long_name = "DBZ" ;
        DBZ:_fillValue = -9999.f ;
        DBZ:scale_factor = 2.6667f ;
        DBZ:add_offset = 48.f ;
        DBZ:units = "dB" ;
```

byte NAW_Range_Vector_Map(time, range) ;
NAW_Range_Vector_Map:long_name = "Range_Vector_Map" ;
NAW_Range_Vector_Map:_fillValue = -9999.f ;
NAW_Range_Vector_Map:sampling_ratio = 1.f ;
NAW_Range_Vector_Map:coordinate = "time range" ;
NAW_Range_Vector_Map:grid_mapping = "grid_mapping" ;
NAW_Range_Vector_Map: scale_factor = 2.6667f ;
NAW_Range_Vector_Map: add_offset = 48.f ;
NAW_Range_Vector_Map:units = "dB" ;

We once again want to reiterate our displeasure with using float 0 as an actual fill value for both DBZ and VEL (especially the latter, since 0 is guaranteed to be an actual physical value in the data). While it would make more sense to assign 0.f to _fillValue than the meaningless -9999.f that presently appears in the attributes, we fear this would lead to confusion to the user who might interpret ALL zeros (upon conversion to float) as fill values. So we reluctantly recommend keeping -9999.f as the _fillValue in the metadata. Note however, that we changed the _fillValue in NAW from the meaningless 999.f to the meaningless -9999.f for consistency with HWX. Again, if we have misunderstood the actual _fillValue present *in the data*, we would appreciate clarification from L3 Harris.