1. **Data Display Algorithm**

General

After the data-coverages are selected and the associated data-sets are loaded the chart display will be generated by:

1. Create a set of drawing instructions for each dataset. This step is called portrayal and defined by the rules in the portrayal catalogue.
2. Render the drawing instructions as described below.

Notes:

* Datasets can only be portrayed entirely, there is no mechanism to only portray single data-coverages
* The algorithm assumes that the rendering is made by using a kind of the ‘Painters algorithm’. This means an opaque fill will completely obscure what has been rendered at this position before. This does not mean that any implementation must follow this approach, other techniques like Z-Buffer technique may be used. The algorithm will not give implementation details, any implementor has the freedom to reach the desired result in the most effective way.

The Rendering Algorithm

The first step is to group the datasets into subsets which we will denote ‘Layers’. The criteria for the separation will be the minimum display scale along with the drawing index of the dataset. Note that all data-coverages within a dataset will share the same minimum display scale and drawing index, and data-sets with the same minimum display scale or the same drawing index are not allowed to overlap. To be precise, the union of all data-coverages of one dataset must not overlap the union of the data-coverages of another dataset with the same minimum display scale or drawing index.

1. Datasets which share a common (non-null) drawing index are grouped together in single layers.
   1. The minimum display scale of these layers is the smallest minimum display scale (the largest scale denominator) of the component datasets.
2. From the remaining datasets, those which share a common minimum display scale are grouped together in single layers.
3. Layers from A and B which share a common minimum display scale are grouped together in single layers.

The ‘Layers’ are then sorted by their minimum display scale and sequentially rendered starting with the smallest minimum display scale.

**Algorithm**: *RenderChartImage*

**Input**: A set of datasets *dataSets*

A drawing device

1. Split the set dataSets into sub-sets denoted l*ayer0*, *layer1*, … *layern* such that the drawing index of each dataset in one *layerx* is not null but is otherwise the same.
   1. Assign a minimum display scale to each layer from the smallest minimum display scale (the largest scale denominator) of the component datasets.
2. Split the remaining dataSets (those where drawing index is null) into sub-sets denoted l*ayern+1*, *layern+2*, … such that the minimum display scale of each dataset in one *layerx* is the same.
3. Combine layers which share a common minimum display scale
4. Sort the *layer1 .. layern* by its associated minimum display scale
5. Clear the drawing device (e.g. by filling the drawing device with the NODTA colour or pattern.
6. Iterate over all *layerx* starting with the smallest minimum display scale
7. Render the layer with the algorithm *RenderLayer*

Note: For the sake of simplicity the concept of display planes (i.e. under and over radar) is not considered here. Without loss of generality the algorithm can be used multiple times to create the images for each display plane. One way of achieving it is to split the output of the portrayal into subsets one for each display plane and run the algorithm for each subset. However, the painters algorithm cannot be used to render data in the over radar display plane since there will not be skin of the earth objects present to obscure underlying layers.

Note: The algorithm as described here does not distinguish between official and non-official data. It could be achieved by taking this into account during the grouping of the input datasets.

The Algorithm RenderLayer

This algorithm describes how the datasets of one layer i.e. those that have the same minimum display scales and/or drawing indices are rendered.

Notes:

**Algorithm**: RenderLayer

**Input**: A set of datasets *dataSets* that have the same minimum display scale and/or drawing indices  
 A drawing device

1. **For** each display priority *displayPriority* starting with the smallest
   1. Collect the active drawing instructions from each dataset’s display instructions that are assigned to *displayPriority*
   2. *Note: while null instructions should not be rendered, they should show in the pick report below the other instructions from that collection*
   3. Render the area instructions from that collection
   4. Render the line instructions from that collection
   5. Render the point instructions from that collection
   6. Render the text instructions from that collection

1a:  
Rendering must take the *viewingGroup(s)*, *scaleMinimum*, *scaleMaximum, date dependency, and any other* properties of the display instruction which may affect the instructions visibility into account. (See S-100 Part 9)

1f:When rendering text, an implementation may take into account the guidance in S-100 Part 9 regarding text rendering to adjust this algorithm as needed to enhance the readability of text.