NetVecCad code documentaton Paul Schrum

Essay on Features

Date: 6/26/2014. If you are reading this after 6/27/2014, it might be out of date.

1. A Feature determines the appearance of graphic elements on the screen and page. It is the same as Level in Microstation and Layer in Autocad. Not only does Feature determine appearance, it is also used as a display filter (and other kinds of filters).

2. Internally Features are collected in the FeatureList class, which holds all Features in a

public SortedList<String, Feature> Children;

3. A file may have a Feature Library attached, or have its own FeatureList, or both, in which case the two are zipped into one. Name collisions of features are revolved in favor of the one in the local file. The loser is ignored.

4. Features are automatically put into a hierarchy structure by parsing the Feature name to form a tree relationship. It is possible to display all Features in a Features List as a WPF Tree Control, or they can be displayed like in Microstation – flat (unparsed) in a Combo Box. The delimiter is customizable, but the default delimiters are "- " (minus and space). *(Not implemented as of June 2014).*

5. Color. There are three ways to define Color: Color Table (from .dxf import), RGB values, or System.Windows.Media.SolidColorBrush (from WPF). As of 6/26/2014, only SolidColorBrush is implemented. *(Will implement RGB by 30 June)*

6. Weight and Thickness. These two both influence how a Path element is drawn at different view scales. (Influence on Text elements may be considered later.) Weight is defined as the pixel width of an element in screen-pixels with this formula: PixWidth = (Weight / 2) + 0.5; if (PixWidth > 9) PixWidth = 9; For Thickness, PixWidth is the actual Thickness at-scale. If PixWidth from Weight is greater than PixWdith from Thickness, we use PixWidth from Weight. Thus a Path will never be so thin that it disappears. Also, there is no maximum on PixWidth from Thickness.

7. For the standard styles, dashes are defined by length in pixels. Long dash is 6,3 (Solid is 6 pixels, skip is 3 pixels). Medium Dash is 4,2. Dot is 1,2. *This will be adjusted as needed after seeing how the implementation looks.*

8. Not now, but eventually FeatureOveride must be implemented. FeatureOveride, where not-null, takes precedence over the Feature of the item. In this way, for example, an item could be on a level where Color is Red, but the FeatureOveride Color is Blue, so the item is drawn as Blue.