MBGuiApp\_ThingsToDo.docx

Samantha Zambo

June 2, 2017

1. Enable Canvas locks to enable/disable simultaneous map actions between canvases
2. Log window sizing
3. Raster color; raster properties dialog from double clicking canvas and saves to settings
4. Relative paths with local directory for third-party libs and includes required
5. Deletions (memory leaks)
6. Add canvas xz and button for toggling between yz and xz canvases
7. Look into ways to reduce memory consumption from both having to resave layers to edit them and using multiple canvases for different dimensional view of data (Currently, data is read in using the “delimitedtxt” provider, but this does not allow the layer to be edited. Thus, it is first saved as a shp file, the “delimitedtxt” layer is deleted, and the shp file is read back in as the layer. This is done once for canvas xy by setting the xfield=1 and yfield=2 in the URI, again for canvas yz by setting the yfield=2 and xfield=3, and will need to be done a third time for canvas xz by setting xfield=1 and yfield=3. This generates many intermediate files that swamp the CPU and memory. Potential remedying options to investigate included:
   1. Virtual tables and table joins
   2. QgsMapLayerCanvas inherits from QGraphicsView. Maybe we can great views into a single model table by copying how QgsMapLayerCanvas is created for the different dimensional views.
   3. Use 3D canvas instead
      1. Qt 4.8.6 – QwtPlot3D library
      2. Upgrade to Qt5.8.0 MSVC 2015
         1. Requires updating mergeBathy.exe c++ code and recompiling third-party libraries. This can be mitigated by disabling “.bag” file output, responsible for the inclusionary (and complex) third-party libraries.
8. QSetting settings vs QSetting settings(“NRL”, “MBGuiApp”)
9. QGIS vs Sams file actions (open, new, save, saveas)
10. Recompile libraries in default which will build both release and debug