Update Loop Considerations

A primary use of kst is in looking at real time data – which requires a powerful and effective update loop system. The following issues must be addressed in the update system:

Multithreading: the presence of multiple processors should be capitalized on by the update system. This includes parallel computation of data objects when interdependencies don't prevent it, and, if possible, parallel generation of plots. While there are times when the CPU usage is dominated by data object computation, a very common use case is for the drawing of plots (eg, 16 plots, each with 1E6 samples, real time updating) to dominate. In my experience, because of the optimizations in the renderers, it is the pre-drawing optimizations which dominate this part, in this case, not the actual line draws.

UI responsiveness: The UI should remain responsive during updates. This should include dialogs and menus, but could also (if possible) include the zoom modes. (The latter implies that axis and label are drawn first, and quickly, and that the curves are drawn later, in a separate thread, and that their drawing can be interrupted by a new zoom command – but not by new data or it might never get drawn.... this may be harder than we can do now and is not a requirement.)

Plot Geometry Interdependencies: The size and arrangement of each plot in a tab is dependent on the size and arrangement of all other plots in the window (See Plot Layout document). Consequently, the updating of all plots in a window must start together – in a coordinated way, and must not start again until all have finished drawing.

Efficient update triggers: Updates can be triggered by the following events. Everything dependent on the change must be updated, but things which are not dependent must not be updated.

- New data in one or more data sources.
- Data object changed in the UI (eg, via a dialog)
- View object changes (eg, a plot zoom requires re-layout of all plots in a window).

Data Source self-syncronization: If a data source acquires new data, then **all** data objects dependent on new data from that data source must be updated (not just the first one which kst notices has changed). Otherwise, you could have the X axis of a plot out of sync with the Y axis.

Update Throttling: It must be possible to reduce the rate at which updates happen – eg – data is coming in at 10Hz, but updates of 1 Hz are sufficient for the user.