

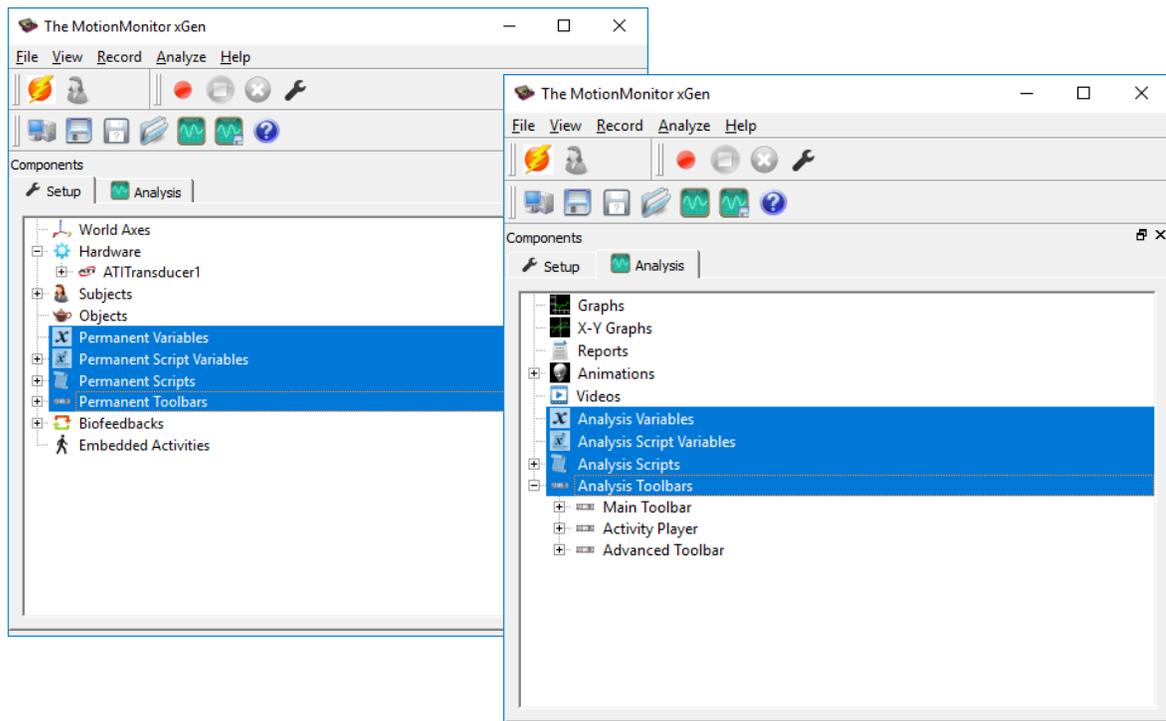
## The MotionMonitor xGen Software Guide: An Overview of Best Practices

This article attempts to identify approaches to setting up applications, protocols and experiments that will result in the best outcome and avoid pitfalls that might impair the performance of The MotionMonitor xGen.

As developers we tend to optimize software for the way we believe it will be used. And we know the way we expect it to be used is often parochial and based on biases that are not necessarily shared with others. So, when The MotionMonitor xGen does not behave in the way we have promised it is possible that you have found new ways to use it! Share with us when it does not live up to expectations and we will try and address the issue. In the meantime, you may want to skim this article before going deep into creation of an application or experiment. It tries to summarize things we have found work better than others. This article will soon be augmented with downloadable examples that you can use as benchmarks in your own work.

### Use of Setup vs Analysis Components

1. The first thing you will notice is that setup and analysis tabs share variables, script variables, scripts and toolbars. So which tab is the best to use? Well it depends. The setup side basically captures information at the time of a collection and is unchangeable thereafter. The analysis side is looking at stuff in the present and can be modified in post processing.
2. Analyses can be saved and opened and even merged with other analysis files. So, things you do on the analysis tab have a lot more flexibility than things done on the setup side...remember the setup side is fixed at the time of collection. Analysis files (.ian) can be created in either the Live Workspace or Activities and similarly shared between each.
3. If you are only doing Live data, ie. never recording an activity as might be done in a training device, it probably does not matter which tab you are using.



## **Live Window and the Number of Variables**

1. All hardware data streams are being captured in a separate thread from what's being used to display data in the Live Workspace window. This separate thread helps maintain the integrity of the underlying hardware data at the measurement rate specified in their respective hardware parameter panels.
2. In the Live window, data are updated on an "as needed" basis. Meaning that data are computed when they're used in a graph or an animation window, or are needed to compute the variable being plotted or displayed, and are updated at the intervals specified in their formulas or displays. Hence, the number of variables in the system during collection can have an impact on performance. However, data that are in a graph or animation window that are tabbed behind another window are not updated until their window is brought to the top. Essentially, having a window tabbed behind another would be equivalent to having the window closed, in terms of their impact on performance.
3. If simply collecting data with all analyses and/or feedback being done in post-processing, it would be best to have the absolute minimum number of variables necessary to complete collection in the Live window and then use multiple analysis files (.ian) to perform processing and provide feedback to the user.
4. While it's true that any graphs and animation windows that are closed or tabbed behind another window do not affect the Live window performance, they will have an impact on the time for saving and opening an activity. Each graph, animation and variable needs to be copied and saved into the Activity file. This is another reason to use analysis files for any data that isn't essential in the Live window.

## **Live Window and the Live Period**

1. The Live period specified in the Setup Hardware parameters panel can have a significant impact on performance. As the period is lengthened, more memory is required and depending on the variables, more computations are required.
2. The Live period does not affect the Recording period (Record|Edit Recording Parameters menu) or the length of a collected activity. So, it is important that the live period be specified consistent with the needs of the Live application while the collection be consistent with the needs of post processing.

## **Variable Types & Resolutions**

1. Variable/operator types and their use in a Live window can have a huge effect on performance. Int(), tmax() and related variables evaluate over time intervals and at resolutions specified in graphs. So, they can have an extreme effect on performance. At one time these were going to be restricted to use only in post processing because of their impact on performance, however their judicious use does have potential benefit.
2. Graphs specify the interval over which data points are plotted. Displaying intervals smaller than 10 ms can cause performance issues and should be reserved for post processing.
3. There is no upper limit to number of variables that can be created within a Workspace and Analysis, so drop-lists and cascading right-click menus can quickly become cluttered or take up the entire screen. A Variable Group can be added to any of the Setup or Analysis Variable nodes in order to create sub-nodes for custom variable organization. Variables can be cut and pasted into a Variable Group.

## **Hardware**

1. The type and number of hardware sources can influence performance. The number of physical cores on a computer should be at least one higher than the number of data sources for optimal performance. Each data source is collected in its own thread and performs best when it has a dedicated core.
2. Digital video currently imposes a 2 second hold at the starting of a collection to ensure its hardware has started successfully.
3. Most hardware devices have the option to 'Suspend live data' under their parameters panel Advanced menu. Suspending live data reduces CPU requirements by making the data only available in post-processing. This feature is commonly enabled for hardware devices with a large number of channels and high measurement rates.
4. Most hardware devices also have the option to enable a "Live data decimation" and to specify the decimation factor through their parameters panel Advanced menu. When enabled, the full measurement rate for the hardware device is available in post processing, but only the decimated measurement rate is available for Live computations. Many devices will have the Live data decimation feature enabled, by default.

## **Subject**

1. The Subject Height and Mass are editable formula fields and may be modified in post-processing. Only static scalar/integer numbers or Script variables should be used here. Specifying a Permanent or Analysis variable or value directly from a hardware component, such as the Stylus or a force plate, will result in the values continuously updating and impact Live performance.
2. The Advanced menu in the Subject parameters panel will have a "Use spline fit to forces and moments" check box and spline interval. By default, joint forces and moments are typically calculated at a 1 msec interval, which can impact Live performance, analyses and exporting. Enabling this checkbox and specifying the spline interval, such as one equivalent to the kinematic tracking hardware, reduces the computational requirements for calculating joint forces and moments. The spline interval is saved with Analysis files and can be loaded with Analysis files when generating reports or exporting data.

## **Component Sets**

1. Component Sets (.ics) are files which allow for sharing of selective components between different workspaces or activities. Common usages of Component Sets may be to copy a stylus or force plate between two workspaces without copying the entire workspace. Components can be added to a Component Set by highlighting one or multiple items from the same parent node and right-clicking and selecting the Export option. Component Sets can be loaded through the File|Import Component Set menu or through running the *ImportComponentSet()*; script function.