

# Object-oriented graphics for quantitative programming environments: design and implementation in the Quail system.

R.W. Oldford<sup>1</sup>

## Abstract

In this talk, I will give an overview and demonstration of the *Views* visual system of *Quail*. Besides being object-oriented, the design of Views differs from that of other systems in many ways.

A fundamental design principle is that any meaningful graphic element is really a visual representation of some underlying data structure. The data structure might be a dataset, a single case, a batch, a mathematical function, etc. As its visual representation, the view is a conduit to the data structure and interaction with the display should permit this.

A second principle is that a single view can be displayed in multiple locations. Wherever it appears, in however many windows, it looks the same; changing it in one place changes it everywhere else.

A third is that all but the simplest views are hierarchical in that they contain subviews. The principal responsibility of such views is the layout of their subviews. Drawing etc, is the responsibility of each subview; their location is the responsibility of the containing view. Such views layout their subviews to display relations between between them — a scatterplot being a case in point. Subviews of views can in turn have subviews and so on to arbitrary depth.

Spatial layout is can also be used to demonstrate relations between two variables mediated by a third. General layouts are themselves views and so can be subviews of other views. In this way, arbitrary interfaces can be made.

---

<sup>1</sup>This is joint work with Catherine Hurley of the George Washington University