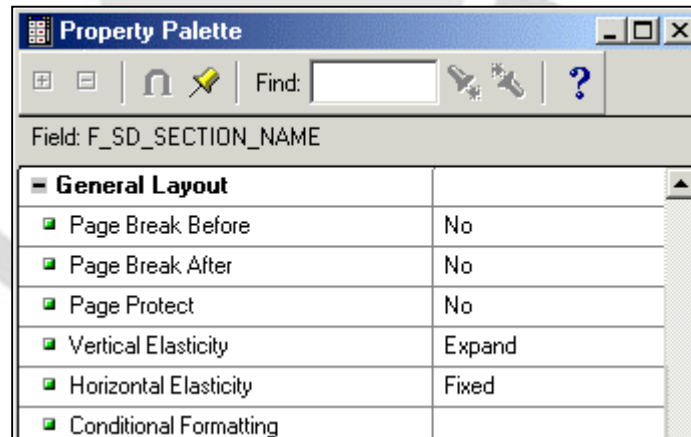# Conditional Formatting

The Developer will already be aware that all objects have properties and that these properties can be used to influence the content and appearance of the object when printed.

There is another property however, which can be used to suppress or change the output of an object completely, Conditional Formatting.

Conditional Formatting

In previous versions clicking on the property edit Format Trigger would bring up the Program Unit Editor. However, Reports 6i displays a wizard for the Developer to dictate what conditions will effect an object's output, this wizard can be run more than once and also creates a Program unit which can be edited. This can be useful to identify which objects have formatting on them.

This property can also be used to set colours/ fonts for an object, to red for example when negative.

In the following example, a frame will be suppressed if a value in a parameter isn't set to 'Y'. Please note that due to the layering aspect of Reports, all children of this frame will also be suppressed, therefore a large number of objects can be affected by one conditional format.

Conditional Formatting

Conditional Formatting

Conditional formatting can also be used to suppress boilerplates when no data has been retrieved, as well as being very useful when overlaying objects on top of each other according to some value in another column, for example, number of axles when a vehicle is HGV, number of seats when it is PSV.

Conditional Formatting
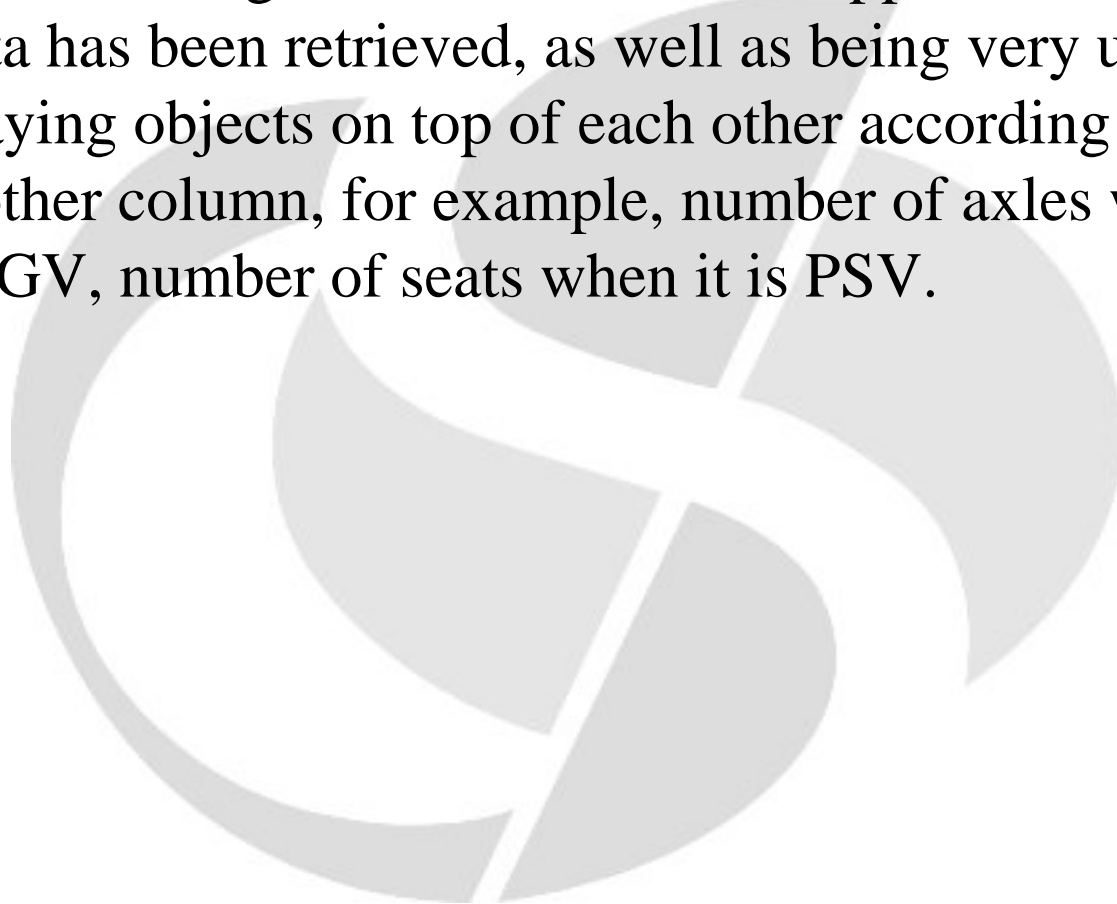
You can communicate to users via the Report using Conditional formatting or the Format trigger behind it.

By returning False instead of the normal True the object will not display. Reversing that round, if an object should only be displayed if a criteria hasn't been met then this coding will suffice …

```
function F_1FormatTrigger return boolean is
begin
   if :section_number is null then
           return (TRUE);
   else
           return(FALSE);
   end if;
end;
```

Conditional Formatting

A similar technique can be applied to restrict certain users from viewing certain report data.

For example, report objects are displayed depending on the username entered. An error message could also be displayed if the user doesn't have the necessary privileges to view any data the report returns.

```
function M_G_SECTION_GRPFRFor return boolean is
begin
  if :p_1 = 'low_user' then
          return(FALSE);
  else
          return (TRUE);
  end if;
end;
```