```vhdl
package ABRO_data_type_pkg is
    constant bound : integer := 255;
    subtype substring is string (1 to bound);

    type data is record
        dataBody : substring;
    end record;

    constant dataInitVal : data;
    constant data_InitialValue : data;
    function format_substring ( param : in string) return substring;
end ABRO_data_type_pkg;

package body ABRO_data_type_pkg is

    function format_substring ( param : in string) return substring is
        variable aux : substring;
        variable index : natural;
    begin
        index := 1;

        while ( index <= param'length and param(index) /= NUL) loop
            aux(index) := param(index);
            index := index + 1;
        end loop;

        while ( index <= bound ) loop
            aux(index) := NUL;
            index := index + 1;
        end loop;

        return aux;
    end format_substring;

    constant dataInitVal : data := data'(dataBody => format_substring("init"));
    constant data_InitialValue : data := data'(dataBody => format_substring("init"
));
end ABRO_data_type_pkg;

use Work.ABRO_data_type_pkg.all;
package ABRO_data_pkg is
    function resolve(resolve_0:string; resolve_1:data) return data ;
  procedure assign_data(assign_data_0:in data; SIGNAL assign_data_1:out data);
end ABRO_data_pkg;

use Work.ABRO_data_type_pkg.all;
package body ABRO_data_pkg is

    function resolve(resolve_0:string; resolve_1:data) return data is
        variable buf : data;

        function endOfString(param : in substring) return positive is
            variable index : positive := 1;
        begin
            while (param(index) /= NUL) loop
                index := index + 1;
```

```vhdl
            end loop;

            return index;
        end endOfString;

    begin

        buf.dataBody := format_substring(resolve_1.dataBody(1 to endOfString(resol
ve_1.dataBody)) & " tirage: " & resolve_0);
        return buf;
    end resolve;

    procedure assign_data(assign_data_0:in data; SIGNAL assign_data_1:out data) i
s
    begin
        assign_data_1.dataBody <= assign_data_0.dataBody;
    end assign_data;
end ABRO_data_pkg;
```