

5.2. STAR FILE UTILITIES

scope of the query, or FALSE if the condition is not met otherwise.

The branching request must have a condition and a branch request that is made if the condition returns TRUE. The other possible branches are optional. The **unknown_** branch, if present, is executed when the condition returns a value of UNKNOWN. If there is no **unknown_** branch, then the default truth value is set as FALSE (*i.e.* a return value of UNKNOWN is treated as equivalent to FALSE) and the **else_** branch is executed if present. It is possible to override this behaviour by using the special operator **assume_true_** before a condition. This operator forces the default truth value to TRUE when a condition returns an UNKNOWN value. It is a useful shorthand when the same branch request is applied against a condition that is either TRUE or UNKNOWN. The syntax is

assume_true_ (<condition>).

A <branch request> (that is, the set of actual individual requests within a branching request construct) has three possible forms:

- (i) a conditional request,
- (ii) a branching request,
- (iii) **scope_<scope setting>** <branch request> **endscope_**.

Note carefully the different contexts in which branch requests and nested branching requests may occur.

scope_<scope setting> specifies the range of data to be searched in the input file. The effect of the setting is closed by the **endscope_** statement. The permitted values of <scope setting> are:

- (i) **data_item_** restricts the branch request to the data items in the condition,
- (ii) **loop_packet_** restricts the branch request to the contents of the loop packet in which data match the condition,
- (iii) **loop_structure_** restricts the branch request to the loop structure in which data match the condition,
- (iv) **save_frame_** restricts the branch request to the contents of the save frame in which data match the condition,
- (v) **data_block_** restricts the branch request to the contents of the data block in which data match the condition,
- (vi) **file_** specifies that the branch request applies to the contents of the file containing data matching the condition (the default setting).

The default scope is invoked when a **scope_<scope setting>** is *not* specified; in such a case the scope of the branch request is the same as that of the condition.

Fig. 5.2.3.1 demonstrates the construction of a branching request that restricts the scope of the query. The two requests in this figure are applied to the STAR File example of Fig. 5.2.2.1. In Figure 5.2.3.1(a), the query is targeted to retrieve all data items in a loop packet where the value of **_basis_set_contraction_scheme** includes the substring (3), provided that the value of **_basis_set_atomic_name** identically matches the string value **hydrogen** in the next outer nested loop packet. The data relevant to the contraction scheme labelled (3) -> [2] are returned. Note how the wildcard data request ***_** retrieves the data items from the next outer loop structure in which the requested data lie.

In the example of Fig. 5.2.3.1(c), the request for an unknown data name cannot be matched within the input file, and the **unknown_** branch of the request is executed. In this case, the secondary request is more specific (only data names including the substring **contraction** are matched) and hence only a few items from the second-level loop are returned. Scopes can be expanded or contracted. If the example of Fig. 5.2.3.1(a) were to be modified by replacing the innermost

```

if _basis_set_atomic_name = hydrogen
  scope_loop_packet_
  if _basis_set_contraction_scheme ?= (3)
    scope_loop_packet_
    *
  endscope_
endif_
endscope_
endif_
(a)

data_Gaussian
loop_
  _basis_set_atomic_name
  _basis_set_atomic_symbol
  _basis_set_atomic_number
  _basis_set_atomic_mass
loop_
  _basis_set_contraction_scheme
  _basis_set_funct_per_contraction
  _basis_set_primary_reference
  _basis_set_source_exponent
  _basis_set_source_coefficient
  _basis_set_comments_index
  _basis_set_atomic_energy
  loop_
  _basis_set_function_exponent
  _basis_set_function_coefficient
  stop_
stop_

hydrogen H 1 1.0079
(3) -> [2] 2:1 PKC1.23.1 R75 R75 C13,C19 -0.496979
4.5018000E+00 1.5628500E-01
6.8144400E-01 9.0469100E-01
1.5139800E-01 1.0000000E+01
stop_
stop_
(b)

if _basis_set_atomic_name = hydrogen
  scope_loop_packet_
  if _basis_set_contraction_XXXXXX ?= (3)
    scope_loop_packet_
    *
  endscope_
  unknown_
  *contraction*
endif_
endscope_
endif_
(c)

data_Gaussian
loop_
  loop_
  _basis_set_contraction_scheme
  _basis_set_funct_per_contraction
  stop_

(2) -> [2] 1: (2) -> [2] 1:
(2) -> [1] 2 (3) -> [2] 2:1
stop_
(d)

```

Fig. 5.2.3.1. Examples of branching requests and the results returned by *Star_Base* from the example file of Fig. 5.2.2.1. (a) A query designed to extract a data structure relevant to one contraction scheme and one atom type. (b) The results of that request. (c) A similar request, but with a branch followed when the condition cannot be matched against a requested data item in the current scope, and (d) the resulting output. See text for discussion.

scope_loop_packet_ declaration with **scope_loop_structure_**, the query would proceed by testing for the existence of a contraction scheme value including the string (3) in the loop packet relevant to the hydrogen results (as before). Finding that this condition was satisfied, the result returned would be all data names in