```
TreeNode< T, N >

    data

                        - level
                        - children
                        - parent
                        - child num

    split mode

                        + TreeNode()
                        + TreeNode()
                        + TreeNode()
                        + TreeNode()
                        + get_child_pointer()
                        + set_child_pointer()
+ get_child_num()
                        + set child num()
                        + increase_child_num()
                        + decrease_child_num()
                        + Parent()
                        + Parent()
                        + get_data_pointer()
                        + get_data_pointer()
                        + get_data_reference()
                        + get_data_reference()
                        + get_level()
                        + set_level()
                        + is_leaf()
                        + is root()
                        + get_split_mode()
                        + set_split_mode()
                        + operator==()
                         TreeNode()
                        * TreeNode()
                        * TreeNode()
                        * TreeNode()
                         get_child_pointer()
                         set_child_pointer()
                         get_child_num()
                         set_child_num()
                        * increase_child_num()
                        * decrease_child_num()
                         Parent()
                        * Parent()
                         get data pointer()
                         get_data_pointer()
                         get_data_reference()
                         get_data_reference()
                         get_level()
                         set_level()
                         is_leaf()
                         is_root()
                         get_split_mode()
                         set_split_mode()
                         operator==()
                < BlockCluster< spacedim,
                double >, BlockClusterTree
                < spacedim, double >::child_num > < spacedim, Number >::child_num >
TreeNode< BlockCluster
< spacedim, double >,
BlockClusterTree< spacedim,
```

double >::child num >

- data

 level children

- parent

child_num

split_mode

+ TreeNode()

+ TreeNode()

+ TreeNode()

+ TreeNode()

+ Parent()

+ get_level()

+ set_level()

+ is_leaf()

+ is_root()

+ get child pointer()

+ set_child_pointer()

+ increase_child_num()

+ decrease_child_num()

+ get_data_pointer()

+ get_data_pointer() + get_data_reference()

+ get_split_mode()

+ set_split_mode()

* get_child_pointer()

* set_child_pointer()

increase_child_num()

* decrease_child_num()

get_data_pointer()

get_data_pointer()

get_split_mode() * set_split_mode()

get_data_reference()

get_data_reference()

get_child_num() set_child_num()

+ operator==()

* TreeNode()

TreeNode()

TreeNode()

* TreeNode()

* Parent()

* Parent()

* get_level()

* set_level()

* operator==()

is_leaf() is_root()

+ get_data_reference()

+ get_child_num()

+ set child num()

TreeNode< BlockCluster < spacedim, Number >, BlockClusterTree< spacedim, Number >::child num > - data - level children - parent child_num split_mode + TreeNode() + TreeNode() + TreeNode() + TreeNode() + get child pointer() + set_child_pointer() + get_child_num() + set_child_num() + increase_child_num() + decrease child num() + Parent() + get_data_pointer() + get data pointer() + get_data_reference() + get_data_reference() + get_level() + set_level() + is_leaf() + is_root() + get_split_mode() + set_split_mode() + operator==() * TreeNode() TreeNode() TreeNode() * TreeNode() * get_child_pointer() * set_child_pointer() * get_child_num() set_child_num() increase_child_num() * decrease_child_num() * Parent() * Parent()

get_data_pointer()

get_data_pointer()

get_split_mode()

set_split_mode()

* get_level()

* set_level() * is_leaf()

is_root()

* operator==()

get_data_reference()

get_data_reference()

< BlockCluster< spacedim,</p>

Number >, BlockClusterTree