1 TreeNode

Expression	Return Type	Description
X::value_type		The type of application specific data stored in a node.
X::index_type		Type used to represent global index to a node in the tree.
X::children_iterator		Model of ForwardIterator for children of a node.
x.value()	value_type	Provides access to application specific data.
x.parent()	X::index_type	Returns parent of the node.
x.children()	std::pair<c_begin,c_end>	Returns begin and end pair of children iterators of the node. c_begin and c_end are of type X::children_iterator.

2 BaseTree

Expression	Return Type	Description
X::tree_node		Model of TreeNode.
X::index_type		Same as X::tree_node::index_type.
X::value_type		Same as X::tree_node::value_type.
X::iterator		Iterator used to iterate through local nodes of the tree.
X::const_iterator		Const iterator used to iterate through local nodes of the tree.
x.size()	unsigned long	Returns local size of the tree.
x.begin()	iterator	Returns iterator pointing to first local node of the tree.
x.end()	iterator	Returns iterator pointing to local end of the tree.
x[i]	TreeNode	Returns copy of i-th node of the tree. i is of type X::index_type.

3 TreeSearchFunction

Expression	Return Type	Description
x(t,first,last,pu,pv,out)	bool	Write results of search on tree t to out. first and last are of type QueryIterator, which is a model of ForwardIterator to a sequence of predicates modeling QueryPredicate. pu and pv are of type SelectPredicate. out is a model of OutputIterator to a sequence of objects modeling BackInsertionSequence. t is a model of BaseTree. The expression returns true on success and false otherwise.

4 QueryPredicate

Expression	Return Type	Description
`x(u)`	`bool`	Model of `UnaryPredicate`. `u` is a model of `TreeNode`. The predicate returns true if `u` is searched item, false otherwise.

5 SelectPredicate

Expression	Return Type	Description
`x(u,p)`	`bool`	Model of `BinaryPredicate`. `u` is a model of `TreeNode`. `p` is a model of `QueryPredicate`. The predicate returns true if `u` should be included in the search path given `p`.

6 TreeComputeFunction

Expression	Return Type	Description
`x(t,generate,combine)`	`bool`	Invoked to perform computations at each node of tree `t`. `t` is of type `BaseTree`. `generate` is a model of `InteractionGenerateFunction`. `combine` is a model of `CombineFunction`. Returns true on success, false otherwise.

7 CombineFunction

Expression	Return Type	Description
`x(u,v)`	`TreeNode`	Operates on nodes `u` and `v` of type `TreeNode`.

8 InteractionGenerateFunction

Expression	Return Type	Description
`x(t,u,out)`	`bool`	Write *interaction set* of node `u` in the tree `t` to the output iterator `out`. The expression returns *dependency flag*. `t` is `BaseTree`, `u` is `NodeIndexType`.