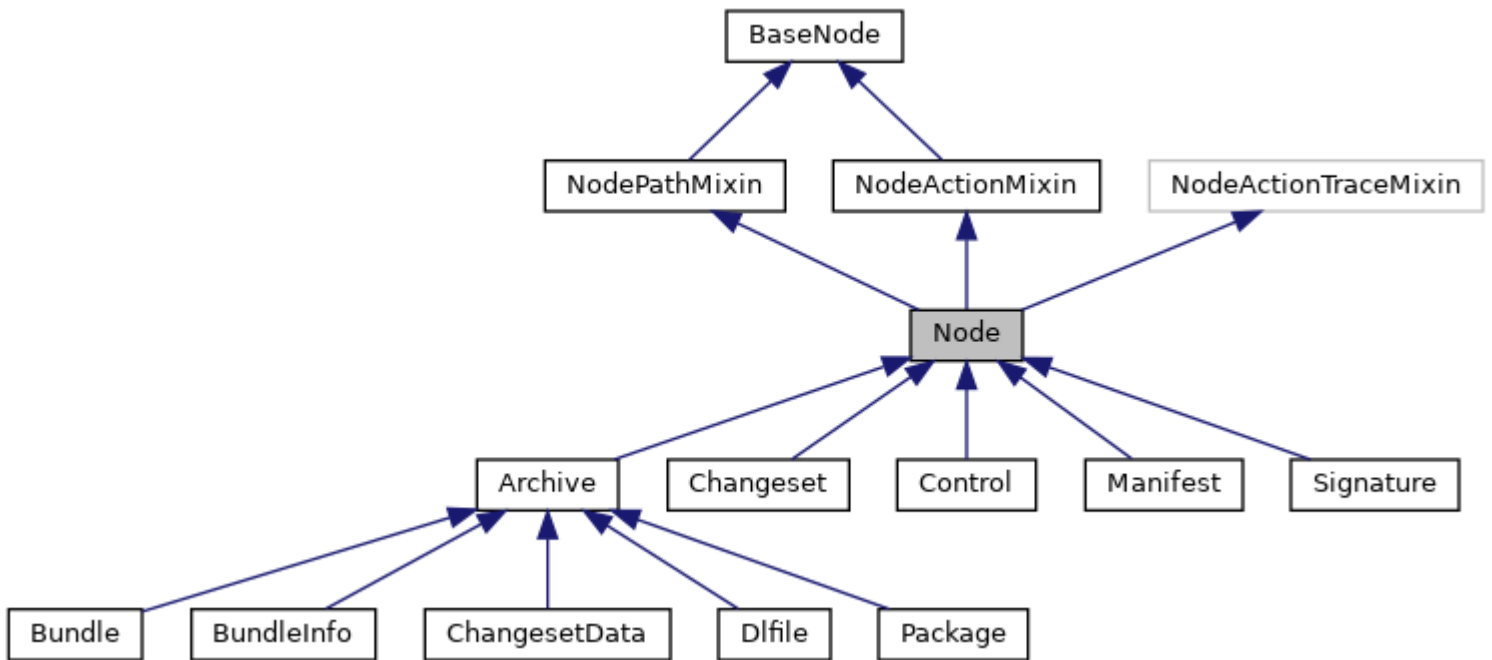


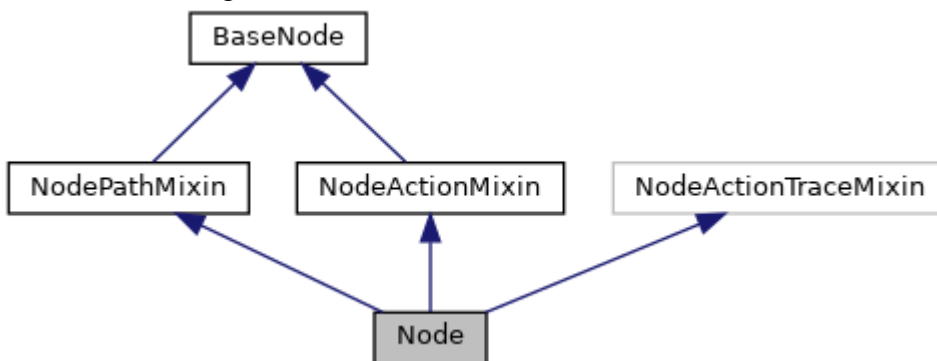
Node Class Reference

Inheritance diagram for Node:



[\[legend\]](#)

Collaboration diagram for Node:



[\[legend\]](#)

Public Member Functions

def [__init__](#) (self, [name](#)=None, [parent](#)=None, [tarinfo](#)=None)

▶ Public Member Functions inherited from [NodePathMixin](#)

def [get_path_nodes](#) (self, include_root=False, within_archive=False)

str [get_path](#) (self, include_root=False, within_archive=False)

Retrieve path of this node within the archive or withing whole tree of archives Separator '/' is used within archives and for recursion into archives. [More...](#)

int [get_depth](#) (self, within_archive=False)

Retrieve depth of node within the archive or within whole tree of archives. [More...](#)

def [find](#) (self, str node_path)

Find node according to node path provided. [More...](#)

def [find_re](#) (self, str node_path_re)

Find node according to regular expression node path provided Search will look for regex matching node name match along the path. [More...](#)

bool [has_ancestor](#) (self, ancestor)

Find if node has another node as ancestor. [More...](#)

bool [has_ancestor_name](#) (self, ancestor_name)

Find if node has an ancestor with provided name (or in list of names) [More...](#)

▶ Public Member Functions inherited from [BaseNode](#)

def [__init__](#) (self, str [name](#)=None, [BaseNode parent](#)=None, tarfile.TarInfo [tarinfo](#)=None)

str [get_type_str](#) (self)

Returns node type as string Supported are: "Package", "Bundle", "Dlfile", "Signature", "Control", "Manifest", "Dir", "File", "Symlink". [More...](#)

bool [is_csd](#) (self)

Test if node is a CSD. [More...](#)

bool [is_inf](#) (self)

Test if node is an INF. [More...](#)

bool [is_meta](#) (self)

Test if node is a META. [More...](#)

bool [is_manifest](#) (self)

Test if node is a manifest. [More...](#)

bool [is_changeset](#) (self)

Test if node is a changeset. [More...](#)

bool [is_archive](#) (self)

Test if node is an archive. [More...](#)

bool [is_dlfile](#) (self)

Test if node is a dlfile. [More...](#)

bool [is_hpsfile](#) (self)

Test if node is a dlfile. [More...](#)

bool [is_upfile](#) (self)
Test if node is an upfile. [More...](#)

bool [is_bundle](#) (self)
Test if node is a bundle. [More...](#)

bool [is_package](#) (self)
Test if node is a package. [More...](#)

bool [is_signature](#) (self)
Test if node is a signature. [More...](#)

bool [is_control](#) (self)
Test if node is a control. [More...](#)

def [mark_modified](#) (self)

def [get_parent_archive](#) (self)
Retrieve parent archive node. [More...](#)

def [get_platform](#) (self)
Retrieve platform for this node. [More...](#)

def [get_root_archive](#) (self)
Retrieve root archive node Root archive node has no parent archive. [More...](#)

int [get_index](#) (self)
Retrieve index of this node in the list of its parent's children. [More...](#)

def [get_signer](#) (self)

int [get_mode](#) (self)
Retrieve file mode of this node. [More...](#)

def [__str__](#) (self)

► Public Member Functions inherited from [NodeActionMixin](#)

def [unlink](#) (self)
Remove this node from tree. [More...](#)

def [destroy](#) (self)
Remove and destroy this node and all its children. [More...](#)

def [add_child](#) (self, child_node, index=None)
Add a child node. [More...](#)

def [create_dir](#) (self, dir_name)
Add a child directory If same child name exists, it will be replaced. [More...](#)

def [create_file](#) (self, file_name, content=b'')
Add a child file. [More...](#)

def [set_content](#) (self, content)
Set the content of this file node. [More...](#)

def [set_mode](#) (self, mode)
Set the mode of this file/directory. [More...](#)

def [set_user_name](#) (self, name)
Set the user name of this file/directory. [More...](#)

def [set_group_name](#) (self, name)

Set the group name of this file/directory. [More...](#)

def [copy](#) (self)
Make a deep copy of this node and its content, children and attributes. [More...](#)

def [move](#) (self, [parent](#))
Move this node to another parent. [More...](#)

def [rename](#) (self, newname)
Rename this node and underlying tarinfo name. [More...](#)

Static Public Member Functions

def [create_node](#) ([name](#), entry_type)
▶ Static Public Member Functions inherited from [BaseNode](#)

tarfile.TarInfo [create_tarinfo](#) (str tarinfo_name, entry_type)
Tarinfo creation helper. [More...](#)

Additional Inherited Members

▶ Data Fields inherited from [BaseNode](#)

- [name](#)
node name (str) [More...](#)
- [parent](#)
parent node ([BaseNode](#)) [More...](#)
- [tarinfo](#)
children of this node in order. [More...](#)
- [file_content](#)
Object representing the content of the file. [More...](#)
- [signature_nodes](#)
Nodes holding signature of this node. [More...](#)
- [additional_signature_nodes](#)
Nodes holding additional signature of this node. [More...](#)

▶ Data Fields inherited from [NodeActionMixin](#)

- [parent](#)
- [children](#)
- [file_content](#)
- [name](#)
- [signature_nodes](#)

Detailed Description

Node class collecting mixins

Constructor & Destructor Documentation

? `__init__()`

```
def __init__( self,  
              name = None,  
              parent = None,  
              tarinfo = None  
            )
```

Reimplemented in [Signature](#), [Control](#), [Changeset](#), and [Manifest](#).

Member Function Documentation

? `create_node()`

```
def create_node( name,  
                entry_type static  
              )
```

Reimplemented from [BaseNode](#).

The documentation for this class was generated from the following file:

- [packman/packmanlib/node.py](#)