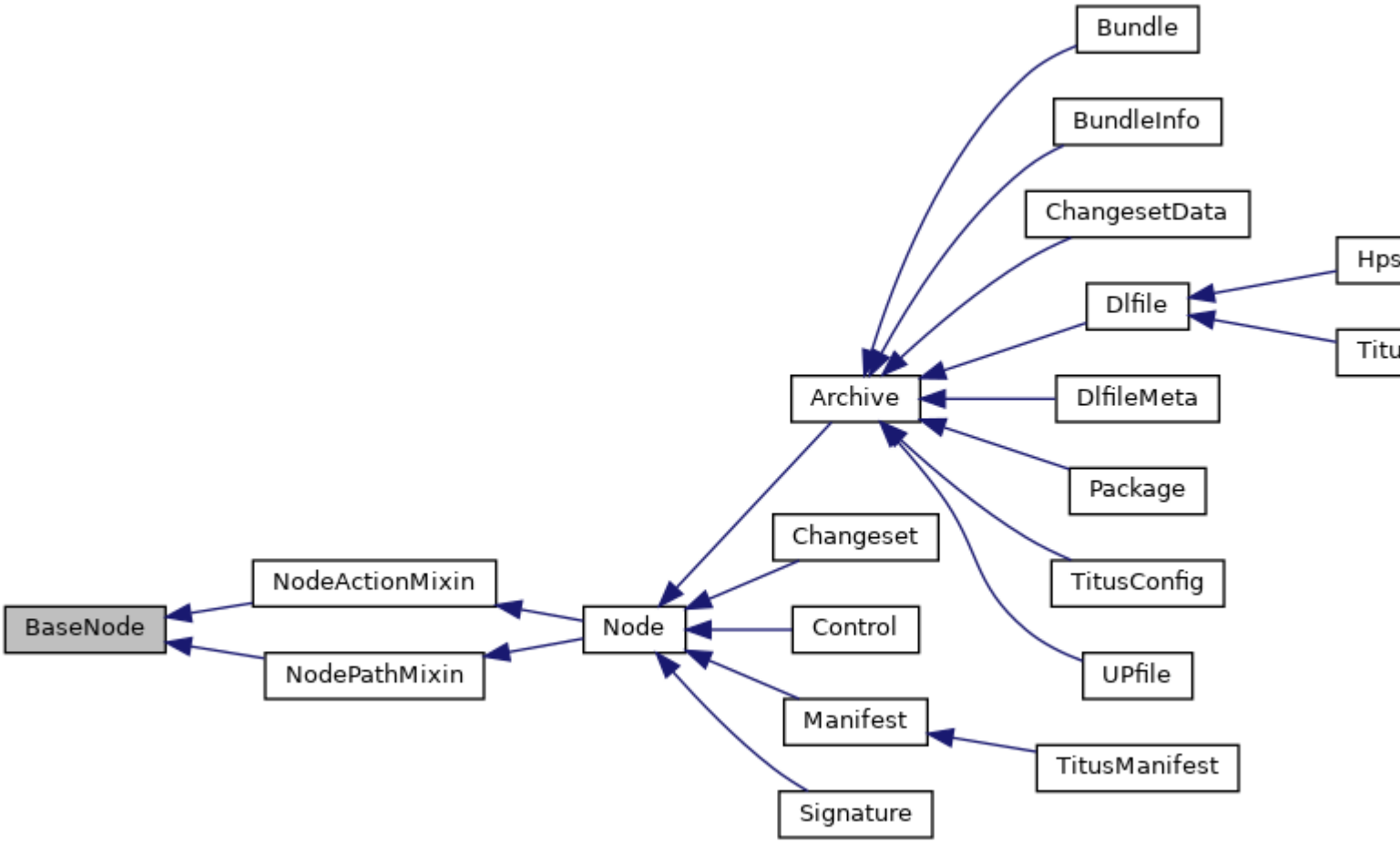
https://verifone.cloud/docs/application-development-kit-version-48/classpackmanlib_1_1basenode_1_1_base_node
Updated: 23-Dec-2025

BaseNode Class Reference

Inheritance diagram for BaseNode:



[legend]

Public Member Functions

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_titus_manifest (self)
Test if node is a titus 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_titusdl (self)
Test if node is a titusdl. More...
bool is_titusconfig (self)
Test if node is a titusconfig. 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)

Static Public Member Functions

tarfile.TarInfo	create_tarinfo (str tarinfo_name, entry_type)
	Tarinfo creation helper. More...
def	create_node (tarinfo_name, entry_type)

Data Fields

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...

Detailed Description

```
Base node class for all nodes
```

Constructor & Destructor Documentation

? __init__()

def __init__ (		*self*,
	str	*name* = None,
	BaseNode	*parent* = None,
	tarfile.TarInfo	*tarinfo* = None

)

Member Function Documentation

? __str__()

def __str__ (*self*)

? create_node()

def create_node (*tarinfo_name*,
entry_type static
)

Reimplemented in Node.

? create_tarinfo()

tarfile.TarInfo create_tarinfo (str *tarinfo_name*,
entry_type static
)

Tarinfo creation helper.

Parameters
tarinfo_name name for tarinfo
entry_type type of tar entry (see TarInfo)

Returns
TarInfo instance created

? get_index()

int get_index (*self*)

Retrieve index of this node in the list of its parent's children.

Returns
0-based index of this node

? get_mode()

int get_mode (*self*)

Retrieve file mode of this node.

Returns
mode value or None if error

? get_parent_archive()

def get_parent_archive (*self*)

Retrieve parent archive node.

Returns
Parent archive node, itself if no parent

? get_platform()

def get_platform (*self*)

Retrieve platform for this node.

Returns
platform instance

? get_root_archive()

def get_root_archive (*self*)

Retrieve root archive node Root archive node has no parent archive.

Returns
Root archive node

? get_signer()

def get_signer (*self*)

? get_type_str()

str get_type_str (*self*)

Returns node type as string Supported are: "Package", "Bundle", "Dlfile", "Signature", "Control", "Manifest", "Dir", "File", "Symlink".

Returns
node type string or 'Unknown'

? is_archive()

bool is_archive (*self*)

Test if node is an archive.

Returns
True if it is an archive, False otherwise

? is_bundle()

bool is_bundle (*self*)

Test if node is a bundle.

Returns
True if it is a bundle, False otherwise

? is_changeset()

bool is_changeset (*self*)

Test if node is a changeset.

Returns
True if it is a changeset, False otherwise

? is_control()

bool is_control (*self*)

Test if node is a control.

Returns
: True if it is a control file, False otherwise

? is_csd()

bool is_csd (*self*)

Test if node is a CSD.

Returns
: True if it is a CSD archive, False otherwise

? is_dlfile()

bool is_dlfile (*self*)

Test if node is a dlfile.

Returns
: True if it is a dlfile, False otherwise

? is_hpsfile()

bool is_hpsfile (*self*)

Test if node is a dlfile.

Returns
: True if it is a dlfile, False otherwise

? is_inf()

bool is_inf (*self*)

Test if node is an INF.

Returns
: True if it is an INF archive, False otherwise

? is_manifest()

bool is_manifest (*self*)

Test if node is a manifest.

Returns
: True if it is a manifest, False otherwise

? is_meta()

bool is_meta (*self*)

Test if node is a META.

Returns
: True if it is a META archive, False otherwise

? is_package()

bool is_package (*self*)

Test if node is a package.

Returns
: True if it is a package, False otherwise

? is_signature()

bool is_signature (*self*)

Test if node is a signature.

Returns
: True if it is a signature, False otherwise

? is_titus_manifest()

bool is_titus_manifest (*self*)

Test if node is a titus manifest.

Returns
True if it is a titus manifest, False otherwise

? is_titusconfig()

bool is_titusconfig (*self*)

Test if node is a titusconfig.

Returns
True if it is a titusconfig, False otherwise

? is_titusdl()

bool is_titusdl (*self*)

Test if node is a titusdl.

Returns
True if it is a titusdl, False otherwise

? is_upfile()

bool is_upfile (*self*)

Test if node is an upfile.

Returns
True if it is an upfile, False otherwise

? mark_modified()

def mark_modified (*self*)

Field Documentation

? additional_signature_nodes

additional_signature_nodes

Nodes holding additional signature of this node.

? file_content

file_content

Object representing the content of the file.

? name

name

node name (str)

? parent

parent

parent node (BaseNode)

? signature_nodes

signature_nodes

Nodes holding signature of this node.

? tarinfo

tarinfo

children of this node in order.

Keys are node name, values BaseNode instances Tarinfo object containing the type of archive entry, path, user, group, permission, etc. see TarInfo documentation

---

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

- packman/packmanlib/basenode.py