NAME

Net::DBus::Binding::Value − Strongly typed data value

SYNOPSIS

```
  # Import the convenience functions
  use Net::DBus qw(:typing);

  # Call a method with passing an int32
  $object->doit(dint32("3"));
```

DESCRIPTION

This module provides a simple wrapper around a raw Perl value, associating an explicit DBus type with the value. This is used in cases where a client is communicating with a server which does not provide introspection data, but for which the basic data types are not sufficient. This class should not be used directly, rather the convenience functions in Net::DBus be called.

METHODS

my `$value` = Net::DBus::Binding::Value−>new($type, `$value`);

Creates a wrapper for the perl value `$value` marking it as having the dbus data type `$type`. It is not necessary to call this method directly, instead the data typing methods in the Net::DBus object should be used.

my `$raw` = `$value`−>value

Returns the raw perl value wrapped by this object

my `$type` = `$value`−>type

Returns the dbus data type this value is marked as having

AUTHOR

Daniel P. Berrange

COPYRIGHT

SEE ALSO

Net::DBus, Net::DBus::Binding::Introspector, Net::DBus::Binding::Iterator