38.3. aepack — Conversion between Python variables and AppleEvent data containers¶

Platforms: Mac

The `aepack` module defines functions for converting (packing) Python variables to AppleEvent descriptors and back (unpacking). Within Python the AppleEvent descriptor is handled by Python objects of built-in type `AEDesc`, defined in module `Carbon.AE`.

Note

This module has been removed in Python 3.x.

The `aepack` module defines the following functions:

`aepack.pack`(*x*[, *forcetype*])¶

Returns an `AEDesc` object containing a conversion of Python value x. If *forcetype* is provided it specifies the descriptor type of the result. Otherwise, a default mapping of Python types to Apple Event descriptor types is used, as follows:

Python type	descriptor type
`FSSpec`	typeFSS
`FSRef`	typeFSRef
`Alias`	typeAlias
integer	typeLong (32 bit integer)
float	typeFloat (64 bit floating point)
string	typeText
unicode	typeUnicodeText
list	typeAEList
dictionary	typeAERecord
instance	*see below*

If *x* is a Python instance then this function attempts to call an `__aepack__()` method. This method should return an `AEDesc` object.

If the conversion *x* is not defined above, this function returns the Python string representation of a value (the repr() function) encoded as a text descriptor.

`aepack.unpack`(*x*[, *formodulename*])¶

x must be an object of type `AEDesc`. This function returns a Python object representation of the data in the Apple Event descriptor *x*. Simple AppleEvent data types (integer, text, float) are returned as their obvious Python counterparts. Apple Event lists are returned as Python lists, and the list elements are recursively unpacked. Object references (ex. `line 3 of document 1`) are returned as instances of `aetypes.ObjectSpecifier`, unless `formodulename` is specified. AppleEvent descriptors with descriptor type typeFSS are returned as `FSSpec` objects. AppleEvent record descriptors are returned as Python dictionaries, with 4-character string keys and elements recursively unpacked.

The optional `formodulename` argument is used by the stub packages generated by `gensuitemodule`, and ensures that the OSA classes for object specifiers are looked up in the correct module. This ensures that if, say, the Finder returns an object specifier for a window you get an instance of `Finder.Window` and not a generic `aetypes.Window`. The former knows about all the properties and elements a window has in the Finder, while the latter knows no such things.

See also

Module `Carbon.AE`
Built-in access to Apple Event Manager routines.
Module `aetypes`
Python definitions of codes for Apple Event descriptor types.

Previous topic

38.2. `aetools` — OSA client support

Next topic

38.4. `aetypes` — AppleEvent objects