

Property Types

Article • 07/23/2022 • 3 minutes to read

Applies to: Outlook 2013 | Outlook 2016

MAPI supports both single-value and multiple-value properties. With a single-value property, there is one value of the base type for the property. With a multiple-value property, there are multiple values of the base type.

The single-value and multiple-value property types that MAPI supports are described in the following table. For each single-value type that has a corresponding multiple-value type, the multiple-value type appears in parentheses after the single-value type.

Property Type	Hex Value	Description
PT_UNSPECIFIED	0000	Indicates that the property type is unknown. This property type is reserved for use with interface methods.
PT_NULL	0001	Indicates no property value. This property type is reserved for use with interface methods and is the same as the OLE type VT_NULL.
PT_I2 (PT_MV_I2)	0002	Signed 16-bit (2-byte) integer. This property type is the same as PT_SHORT (PT_MV_SHORT) and the OLE type VT_I2.
PT_I4 (PT_MV_I4)	0003	Signed 32-bit (4-byte) integer. This property type is the same as PT_LONG (PT_MV_LONG) and the OLE type VT_I4.
PT_FLOAT (PT_MV_FLOAT)	0004	32-bit (8-byte) floating point value. This property type is the same as PT_R4 (PT_MV_R4) and the OLE type VT_R4.
PT_DOUBLE (PT_MV_DOUBLE)	0005	64-bit (8-byte) floating point value. This property type is the same as PT_R8 and the OLE types VT_R8 and VT_DOUBLE.
PT_CURRENCY (PT_MV_CURRENCY)	0006	64-bit (8-byte) integer interpreted as decimal. This property type is compatible with the Microsoft Visual Basic CURRENCY type and is the same as the OLE type VT_CY.
PT_APPTIME (PT_MV_APPTIME)	0007	Double value that is interpreted as date and time. The integer part is the date and the fraction part is the time. This property type is the same as the OLE type VT_DATE and is compatible with the Microsoft Visual Basic time representation.
PT_ERROR	000A	SCODE value; 32-bit (4-byte) unsigned integer. This property type is the same as the OLE type VT_ERROR.

Property Type	Hex Value	Description
PT_BOOLEAN (PT_MV_12)	000B	16-bit (2-byte) Boolean value where zero equals false and non-zero equals true . This property type is the same as the OLE type VT_BOOL.
PT_OBJECT	000D	Pointer to an object that implements the IUnknown interface. This property type is similar to several OLE types such as VT_UNKNOWN.
PT_I8 (PT_MV_I8)	0014	Signed 64-bit (8-byte) integer that uses the LARGE_INTEGER structure. This property type is the same as PT_I8 and the OLE type VT_I8.
PT_STRING8 (PT_MV_STRING8)	001E	Null-terminated 8-bit (2-byte) character string. This property type is the same as the OLE type VT_LPSTR.
PT_TSTRING (PT_MV_TSTRING)	001F	Null-terminated 16-bit (2-byte) character string. Properties with this type have the property type reset to PT_UNICODE when compiling with the UNICODE symbol and to PT_STRING8 when not compiling with the UNICODE symbol. This property type is the same as the OLE type VT_LPSTR for resulting PT_STRING8 properties and VT_LPWSTR for PT_UNICODE properties
PT_SYSTIME (PT_MV_SYSTIME)	0040	64-bit (8-byte) integer data and time value in the form of a FILETIME structure. This property type is the same as the OLE type VT_FILETIME.
PT_CLSID (PT_MV_CLSID)	0048	CLSID structure value. This property type is the same as the OLE type VT_CLSID.
PT_SVREID	00FB	Variable size, a 16-bit (2-byte) COUNT followed by a structure.
PT_SRESTRICT	00FD	Variable size, a byte array representing one or more Restriction structures.
PT_ACTIONS	00FE	Variable size, a 16-bit (2-byte) COUNT of actions (not bytes) followed by that many Rule Action structures.
PT_BINARY (PT_MV_BINARY)	0102	SBinary structure value, a counted byte array.

ⓘ Note

To determine the Hex value for the multi-valued property type, OR the PT_MV flag (0x00001000) to the Hex value for the property type. For example, the Hex value for PT_MV_UNICODE is 0x101F and the Hex value for PT_MV_BINARY is 0x1102.

MAPI shares the value type numbers with [\[OLE variants\]](#). However, not all OLE types are specified for MAPI. In particular, unsigned types such as VT_UI4 have no counterpart in MAPI. Comparison of PT_I2/I4/I8 property values, e.g. during the evaluation of [\[restrictions\]](#) (filters), are performed as a signed comparison.