



IMoraeEvent interface

Purpose of interface:

The plug-in must implement this interface to provide basic information about an individual event. Events are accessed via an IEnumMoraeEvent interface. This enumerator can come from a call to IMoraeEventStream::EnumEvents or IMoraeEventSearch::RunSearch.

This interface derives from IUnknown.

Interface ID:

IID_IMoraeEvent = 975F418C-AA02-468E-A398-A1BC1381B3FE

Methods:

1. HRESULT GetTime(

VARIANT* pvarElapsedTime, [out]

VARIANT* pvarTimeDate) [out]

This call requests the time at which the event occurred in two different formats.

*pvarElapsedTime should be set to a VT_I8 type. The lVal of the VARIANT should be set to the elapsed time into the recording in 100-nanosecond units. (10000 units is equivalent to 1 millisecond.) The *pvarTimeDate should be set to a VT_DATE type. The date element of the VARIANT should be set to the DATE value that reflects the time and date at which the event occurred.

2. HRESULT GetTimeRange(

VARIANT* pVarStartTimes, [out]

VARIANT* pVarEndTimes) [out]

Plug-ins are not required to support this call. It is not used by Manager at this time (version 3.2). It is intended for plug-ins whose events represent one or more blocks of time. If the plug-in does implement this call, the variants should be set to SAFEARRAYs of type VT_I8. Each element of the array should indicate an elapsed time into the recording in 100-nanosecond units.

If the event represents a single time point, the pVarEndTimes should not be set. The pVarStartTimes should be set to a SAFEARRAY holding a single value.

If the event represents one or more blocks of time, the pVarStartTimes and pVarEndTimes should be set to SAFEARRAYs with equal numbers of elements.

3. HRESULT GetAppTag(

int* pTag) [out]



Plug-ins are not required to support this call. If implemented, it should set the value of pTag to indicate the application that this event occurred in. For information on how to translate an application name to a tag, see the documentation of the IMoraeRecording interface.

4. HRESULT GetWndClassTag(

int* pTag) [out]

Plug-ins are not required to support this call. It is not used by Manager at this time (version 3.2). If implemented, pTag should be set to a value that indicates the class of window in which the event occurred. For information on how to translate a window class name to a tag, see the documentation of the IMoraeRecording interface.

5. HRESULT GetHWND(

VARIANT* pVar) [out]

Plug-ins are not required to support this call. If implemented, the pVar should be set to a VT_UI4 or VT_UI8 value that gives the HWND of the window in which this event occurred.

6. HRESULT GetEventTypeId(

IID* pID) [out]

The pID should be set to an IID value that uniquely identifies the type of event. It should be the same value as provided by the IMoraeEventStream::GetEventStreamID call. Manager uses these two calls to associate an individual event with its event stream source.

(For Morae's native event streams, the survey event is an exception to this. For this call, it outputs a unique IID value for the particular survey that it refers to.)