

OpcTypeDecodingFailedEventArgs

Members

Namespace: Opc.UaFx

Assemblies: Opc.UaFx.Advanced.dll, Opc.UaFx.Advanced.dll

The [OpcTypeDecodingFailedEventArgs](#) type exposes the following members.

Properties

Data

Gets the data which is tried to decode.

C#

```
public object Data { get; }
```

Property Value

[Object](#)

An [Object](#) instance which depends on the decoder used.

Encoding

Gets the type of encoding supported by the decoder used.

C#

```
public OpcEncodingType Encoding { get; }
```

Property Value

[OpcEncodingType](#)

One of the members defined by the [OpcEncodingType](#) enumeration.

Exception

Gets the [Exception](#) created or caught during the decoding process.

C#

```
public Exception Exception { get; }
```

Property Value

Exception

An instance of the [Exception](#) class or a null reference (Nothing in Visual Basic) in case there the conditions under which the decoding failed are in advance determined and reported.

FieldName

Gets the name of the field of an [Instance](#) which is about to get decoded.

C#

```
public string FieldName { get; }
```

Property Value

String

The name of the field to decode or a null reference (Nothing in Visual Basic) if there is no field name known by the decoder for the current decode failure.

FieldType

Gets the type of the field being decoded.

C#

```
public Type FieldType { get; }
```

Property Value

Type

The [Type](#) of the field to decode or a null reference (Nothing in Visual Basic) if there is no [Type](#) known by the decoder for the current decode failure.

Instance

Gets the [Object](#) instance its data is being decoded.

C#

```
public object Instance { get; }
```

Property Value

Object

An [Object](#) instance or a null reference (Nothing in Visual Basic) in case there the decoder is not about to decode an instance.

Message

Gets the description of the decoding failure.

C#

```
public string Message { get; }
```

Property Value

[String](#)

The description of the decoding failure or a null reference (Nothing in Visual Basic) in the case there the decoding failure indicates an exception which was handled using a generic exception handler.

Type

Gets the type information used to decode the data of the [Instance](#).

C#

```
public Type Type { get; }
```

Property Value

[Type](#)

The [Type](#) of the [Instance](#) being decoded or a null reference (Nothing in Visual Basic) if the decoding failure occurred in a context where an instance type is either not used or unknown.

TypeId

Gets the identifier of the type being decoded.

C#

```
public OpcNodeId TypeId { get; }
```

Property Value

[OpcNodeId](#)

An instance of the [OpcNodeId](#) class which identifies the type being decoded or a null reference (Nothing in Visual Basic) if the decoding failed in a context there a type identifier does not apply to the decoding operation performed.

TypeIdAbsolute

Gets the absolute [TypeId](#) (if available).

C#

```
public OpcNodeId TypeIdAbsolute { get; }
```

Property Value

OpcNodeId

The absolute [TypeId](#) in case there the [TypeId](#) has been resolved or a null reference (Nothing in Visual Basic) if the [TypeId](#) hasn't been resolved.

Methods

Dispose()

Releases all resources used by the [OpcTypeDecodingFailedEventArgs](#).

C#

```
public void Dispose()
```

Dispose(Boolean)

Releases the unmanaged resources used by the [OpcTypeDecodingFailedEventArgs](#) and optionally releases the managed resources.

C#

```
protected virtual void Dispose(bool disposing)
```

Parameters

disposing [Boolean](#)

The value true to release both managed and unmanaged resources; otherwise the value false to release only unmanaged resources.

GetData()

Retrieves a copy of the binary data in the [Stream](#) used by the decoder to decode the data.

C#

```
public byte[] GetData()
```

Returns

[Byte\[\]](#)

An array of [Byte](#) values with the data the decoder uses for decoding.

Remarks

In case there [GetStream](#) is used as well and the [Stream](#) instance is modified, the array offered by [GetData](#) is modified as well.

GetStackTrace()

Retrieves an instance which summarizes the stack within the decoder until the failure has been produced. The [OpcEncodingStackTrace](#) returned contains [OpcEncodingStackFrame](#) entries only in case there [IsEnabled](#) is equals true (this influences the overall performance while encoding / decoding).

C#

```
public OpcEncodingStackTrace GetStackTrace()
```

Returns

[OpcEncodingStackTrace](#)

An instance of the [OpcEncodingStackTrace](#) class which summarizes the stack within the decode until the failure has been produced. The instance returned provides an empty stack trace if [IsEnabled](#) is equals false (the default value).

GetStream()

Retrieves a copy of the [Stream](#) used by the decoder to decode the data.

C#

```
public Stream GetStream()
```

Returns

[Stream](#)

An instance of the [Stream](#) which represents a copy of the [Stream](#) used by the decoder to decode the data.

Remarks

The [Stream](#) provided allows reading, writing and seeking; but modifying or disposing the instance will influence the instance provided by subsequent calls to [GetStream](#) of the current [OpcTypeDecodingFailedEventArgs](#).

ToString()

Returns a [String](#) representing the [Message](#) or the [Message](#) of the [Exception](#) of the current [OpcTypeDecodingFailedEventArgs](#).

C#

```
public override string ToString()
```

Returns

String

A [String](#) representing the [Message](#) or the [Message](#) of the [Exception](#) of the current [OpcTypeDecodingFailedEventArgs](#).

Table of Contents

Properties	1
Data	1
Encoding	1
Exception	1
FieldName	2
FieldType	2
Instance	2
Message	3
Type	3
TypeId	3
TypeIdAbsolute	3
Methods	4
Dispose()	4
Dispose(Boolean)	4
GetData()	4
GetStackTrace()	5
GetStream()	5
ToString()	5

