

# LogManager

Erzeugt von Doxygen 1.8.9.1

Die Jun 30 2015 08:35:56



# Inhaltsverzeichnis

<b>1</b>	<b>Verzeichnis der Namensbereiche</b>	<b>1</b>
1.1	Pakete . . . . .	1
<b>2</b>	<b>Klassen-Verzeichnis</b>	<b>3</b>
2.1	Auflistung der Klassen . . . . .	3
<b>3</b>	<b>Datei-Verzeichnis</b>	<b>5</b>
3.1	Auflistung der Dateien . . . . .	5
<b>4</b>	<b>Dokumentation der Namensbereiche</b>	<b>7</b>
4.1	Paket LogManager . . . . .	7
4.1.1	Dokumentation der Aufzählungstypen . . . . .	7
4.1.1.1	LogfileNameType . . . . .	7
4.1.1.2	LogLevel . . . . .	7
4.1.1.3	LogTarget . . . . .	8
<b>5</b>	<b>Klassen-Dokumentation</b>	<b>9</b>
5.1	LogManager.LoggingFunctions Klassenreferenz . . . . .	9
5.1.1	Ausführliche Beschreibung . . . . .	9
5.1.2	Dokumentation der Elementfunktionen . . . . .	10
5.1.2.1	Debug . . . . .	10
5.1.2.2	Error . . . . .	10
5.1.2.3	Exception . . . . .	10
5.1.2.4	Log . . . . .	11
5.1.2.5	Status . . . . .	11
5.1.2.6	Trace . . . . .	12
5.1.3	Dokumentation der Propertys . . . . .	12
5.1.3.1	FileName . . . . .	12
5.1.3.2	FilePath . . . . .	12
5.1.3.3	LogfileNameType . . . . .	12
5.1.3.4	LogTarget . . . . .	12
5.1.3.5	SetLogLevel . . . . .	12
5.2	LogManager.LogManager Klassenreferenz . . . . .	13

5.2.1	Beschreibung der Konstruktoren und Destruktoren . . . . .	13
5.2.1.1	LogManager . . . . .	13
5.2.2	Dokumentation der Elementfunktionen . . . . .	13
5.2.2.1	Log . . . . .	13
5.2.3	Dokumentation der Propertys . . . . .	13
5.2.3.1	FileName . . . . .	13
5.2.3.2	FileNameType . . . . .	14
5.2.3.3	FilePath . . . . .	15
5.2.3.4	LogLevel . . . . .	15
5.2.3.5	Target . . . . .	15
<b>6</b>	<b>Datei-Dokumentation</b>	<b>17</b>
6.1	Internal/Logger.cs-Dateireferenz . . . . .	17
6.2	LoggingFunctions.cs-Dateireferenz . . . . .	17
6.3	Properties/AssemblyInfo.cs-Dateireferenz . . . . .	18
<b>Index</b>		<b>19</b>

# Kapitel 1

## Verzeichnis der Namensbereiche

### 1.1 Pakete

Hier folgen die Pakete mit einer Kurzbeschreibung (wenn verfügbar):

<a href="#">LogManager</a> . . . . .	7
--------------------------------------	---



## Kapitel 2

# Klassen-Verzeichnis

### 2.1 Auflistung der Klassen

Hier folgt die Aufzählung aller Klassen, Strukturen, Varianten und Schnittstellen mit einer Kurzbeschreibung:

#### [LogManager.LoggingFunctions](#)

This is the public static access class for logging. You can log Arrays, Lists and Dictionarys and other objects. The objects will be logged with their `.ToString()` functions. See `README.txt` for more details and implementation. . . . .

9

#### [LogManager.LogManager](#) . . . . .

13





## Kapitel 3

# Datei-Verzeichnis

### 3.1 Auflistung der Dateien

Hier folgt die Aufzählung aller Dateien mit einer Kurzbeschreibung:

<a href="#">LoggingFunctions.cs</a> . . . . .	17
Internal/ <a href="#">Logger.cs</a> . . . . .	17
Properties/ <a href="#">AssemblyInfo.cs</a> . . . . .	18



# Kapitel 4

## Dokumentation der Namensbereiche

### 4.1 Paket LogManager

#### Klassen

- class [LoggingFunctions](#)  
*This is the public static access class for logging. You can log Arrays, Lists and Dictionarys and other objects. The objects will be logged with their .ToString() functions. See README.txt for more details and implementation.*
- class [LogManager](#)

#### Aufzählungen

- enum [LogLevel](#) {  
    [LogLevel.Trace](#), [LogLevel.Debug](#), [LogLevel.Error](#), [LogLevel.Exception](#),  
    [LogLevel.Status](#) }  
*The LogLevel Enum*
- enum [LogTarget](#) { [LogTarget.Console](#) = 1, [LogTarget.File](#) = 2 }  
*The LogTarget Enum*
- enum [LogfileNameType](#) { [LogfileNameType.Date](#), [LogfileNameType.Rolling](#) }  
*The LogfileNameType enum*

#### 4.1.1 Dokumentation der Aufzählungstypen

##### 4.1.1.1 enum LogManager.LogfileNameType

The LogfileNameType enum

##### Aufzählungswerte

- Date** A date as logfile name.  
**Rolling** A rolling logfile name

##### 4.1.1.2 enum LogManager.LogLevel

The LogLevel Enum

##### Aufzählungswerte

- Trace** The trace LogLevel

***Debug*** The debug LogLevel

***Error*** The error LogLevel

***Exception*** The exception LogLevel

***Status*** The status LogLevel

#### 4.1.1.3 enum **LogManager.LogTarget**

The LogTarget Enum

Aufzählungswerte

***Console*** The console LogTarget

***File*** The file LogTarget

# Kapitel 5

## Klassen-Dokumentation

### 5.1 LogManager.LoggingFunctions Klassenreferenz

This is the public static access class for logging. You can log Arrays, Lists and Dictionarys and other objects. The objects will be logged with their `.ToString()` functions. See README.txt for more details and implementation.

#### Öffentliche, statische Methoden

- static void `Trace` (object toLog)  
*Writes the object as Trace to the log.*
- static void `Debug` (object toLog)  
*Writes the object as Debug to the log.*
- static void `Error` (object toLog)  
*Writes the object as Error to the log.*
- static void `Exception` (object toLog)  
*Writes the object as Exception to the log.*
- static void `Status` (object toLog)  
*Writes the object as Status to the log.*
- static void `Log` (object toLog, `LogLevel` level)  
*Writes the object with a defined LogLevel to the log.*

#### Propertys

- static `LogLevel SetLogLevel` [get, set]  
*Gets or sets the minimum LogLevel. Everything passed to Log(...) with a LogLevel below will be ignored.*
- static string `FileName` [get, set]  
*Gets or sets the name of the LogFile.*
- static string `FilePath` [get, set]  
*Gets or sets the path of the LogFile. Can be a relative or absolute path.*
- static `LogfileNameType LogfileNameType` [get, set]  
*Gets or sets the type of the logfile name. Can be LogfileNameType.Date or LogfileNameType.Rolling*
- static `LogTarget LogTarget` [get, set]  
*Gets or sets the Logtarget. Can be LogTarget.Console and/or LogTarget.File*

#### 5.1.1 Ausführliche Beschreibung

This is the public static access class for logging. You can log Arrays, Lists and Dictionarys and other objects. The objects will be logged with their `.ToString()` functions. See README.txt for more details and implementation.

## 5.1.2 Dokumentation der Elementfunktionen

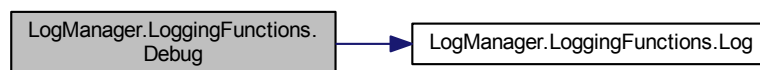
### 5.1.2.1 `static void LogManager.LoggingFunctions.Debug ( object toLog ) [static]`

Writes the object as Debug to the log.

Parameter

<i>toLog</i>	Object to log.
--------------	----------------

Hier ist ein Graph, der zeigt, was diese Funktion aufruft:



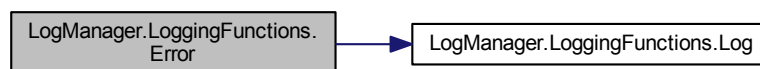
### 5.1.2.2 `static void LogManager.LoggingFunctions.Error ( object toLog ) [static]`

Writes the object as Error to the log.

Parameter

<i>toLog</i>	Object to log.
--------------	----------------

Hier ist ein Graph, der zeigt, was diese Funktion aufruft:



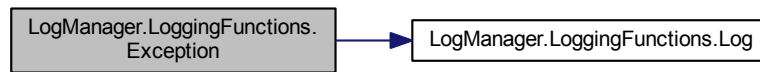
### 5.1.2.3 `static void LogManager.LoggingFunctions.Exception ( object toLog ) [static]`

Writes the object as Exception to the log.

Parameter

<i>toLog</i>	Object to log.
--------------	----------------

Hier ist ein Graph, der zeigt, was diese Funktion aufruft:



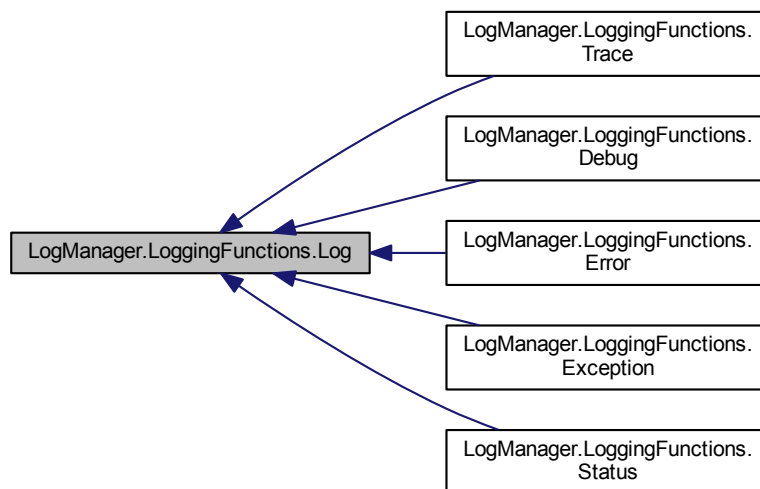
#### 5.1.2.4 static void LogManager.LoggingFunctions.Log ( object *toLog*, LogLevel *level* ) [static]

Writes the object with a defined LogLevel to the log.

Parameter

<i>toLog</i>	Object to log.
<i>level</i>	The LogLevel.

Hier ist ein Graph der zeigt, wo diese Funktion aufgerufen wird:



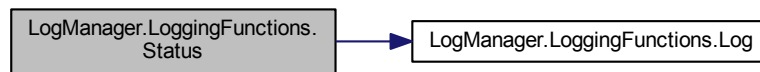
#### 5.1.2.5 static void LogManager.LoggingFunctions.Status ( object *toLog* ) [static]

Writes the object as Status to the log.

Parameter

<i>toLog</i>	Object to log.
--------------	----------------

Hier ist ein Graph, der zeigt, was diese Funktion aufruft:



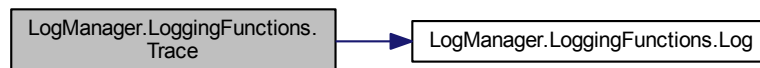
#### 5.1.2.6 `static void LogManager.LoggingFunctions.Trace ( object toLog ) [static]`

Writes the object as Trace to the log.

Parameter

<i>toLog</i>	Object to log.
--------------	----------------

Hier ist ein Graph, der zeigt, was diese Funktion aufruft:



### 5.1.3 Dokumentation der Propertys

#### 5.1.3.1 `string LogManager.LoggingFunctions.FileName [static], [get], [set]`

Gets or sets the name of the LogFile.

#### 5.1.3.2 `string LogManager.LoggingFunctions.FilePath [static], [get], [set]`

Gets or sets the path of the LogFile. Can be a relative or absolute path.

#### 5.1.3.3 `LogfileNameType LogManager.LoggingFunctions.LogfileNameType [static], [get], [set]`

Gets or sets the type of the logfile name. Can be `LogfileNameType.Date` or `LogfileNameType.Rolling`

#### 5.1.3.4 `LogTarget LogManager.LoggingFunctions.LogTarget [static], [get], [set]`

Gets or sets the Logtarget. Can be `LogTarget.Console` and/or `LogTarget.File`

#### 5.1.3.5 `LogLevel LogManager.LoggingFunctions.SetLogLevel [static], [get], [set]`

Gets or sets the minimum LogLevel. Everything passed to `Log(...)` with a LogLevel below will be ignored.

Die Dokumentation für diese Klasse wurde erzeugt aufgrund der Datei:



- [LoggingFunctions.cs](#)

## 5.2 LogManager.LogManager Klassenreferenz

### Paketfunktionen

- [LogManager](#) ()  
*Initializes a new instance of the [LogManager](#) class.*
- void [Log](#) (string message, [LogLevel](#) level)  
*Logs the specified message with the loglevel in the file and/or the console.*

### Propertys

- [LogTarget Target](#) [get, set]  
*Gets or sets the Logtarget.*
- [LogfileNameType FileNameType](#) [get, set]  
*Gets or sets the type of the file name.*
- [LogLevel LogLevel](#) [get, set]  
*Gets or sets the log level.*
- string [FileName](#) [get, set]  
*Gets or sets the name of the file.*
- string [FilePath](#) [get, set]  
*Gets or sets the file path.*

### 5.2.1 Beschreibung der Konstruktoren und Destruktoren

#### 5.2.1.1 LogManager.LogManager.LogManager ( ) [package]

Initializes a new instance of the [LogManager](#) class.

### 5.2.2 Dokumentation der Elementfunktionen

#### 5.2.2.1 void LogManager.LogManager.Log ( string message, LogLevel level ) [package]

Logs the specified message with the loglevel in the file and/or the console.

##### Parameter

<i>message</i>	The message to write into the log.
<i>level</i>	The loglevel.

### 5.2.3 Dokumentation der Propertys

#### 5.2.3.1 string LogManager.LogManager.FileName [get],[set],[package]

Gets or sets the name of the file.

The name of the file.

**5.2.3.2 LogfileNameType** `LogManager.LogManager.FileNameType` `[get], [set], [package]`

Gets or sets the type of the file name.

The type of the file name.

## Ausnahmebehandlung

<i>System.NotSupportedException</i>	More than one flag is not allowed for FileNameType
-------------------------------------	--

**5.2.3.3 string LogManager.LogManager.FilePath** [get], [set], [package]

Gets or sets the file path.

The file path.

**5.2.3.4 LogLevel LogManager.LogManager.LogLevel** [get], [set], [package]

Gets or sets the log level.

The log level.

## Ausnahmebehandlung

<i>System.NotSupportedException</i>	More that one flag is not allowed for LogLevel
-------------------------------------	--

**5.2.3.5 LogTarget LogManager.LogManager.Target** [get], [set], [package]

Gets or sets the Logtarget.

The logtarget. Console and/or File.

Die Dokumentation für diese Klasse wurde erzeugt aufgrund der Datei:

- Internal/[Logger.cs](#)



# Kapitel 6

## Datei-Dokumentation

### 6.1 Internal/Logger.cs-Dateireferenz

#### Klassen

- class [LogManager.LogManager](#)

#### Namensbereiche

- package [LogManager](#)

#### Aufzählungen

- enum [LogManager.LogLevel](#) {  
    [LogManager.LogLevel.Trace](#), [LogManager.LogLevel.Debug](#), [LogManager.LogLevel.Error](#), [LogManager.LogLevel.Exception](#),  
    [LogManager.LogLevel.Status](#) }

*The LogLevel Enum*

- enum [LogManager.LogTarget](#) { [LogManager.LogTarget.Console](#) = 1, [LogManager.LogTarget.File](#) = 2 }

*The LogTarget Enum*

- enum [LogManager.LogfileNameType](#) { [LogManager.LogfileNameType.Date](#), [LogManager.LogfileNameType.Rolling](#) }

*The LogfileNameType enum*

### 6.2 LoggingFunctions.cs-Dateireferenz

#### Klassen

- class [LogManager.LoggingFunctions](#)

*This is the public static access class for logging. You can log Arrays, Lists and Dictionarys and other objects. The objects will be logged with their .ToString() functions. See README.txt for more details and implementation.*

#### Namensbereiche

- package [LogManager](#)

### 6.3 Properties/AssemblyInfo.cs-Dateireferenz

# Index

- Console
  - LogManager, [8](#)
- Date
  - LogManager, [7](#)
- Debug
  - LogManager, [7](#)
  - LogManager::LoggingFunctions, [10](#)
- Error
  - LogManager, [8](#)
  - LogManager::LoggingFunctions, [10](#)
- Exception
  - LogManager, [8](#)
  - LogManager::LoggingFunctions, [10](#)
- File
  - LogManager, [8](#)
- FileName
  - LogManager::LogManager, [13](#)
  - LogManager::LoggingFunctions, [12](#)
- FileNameType
  - LogManager::LogManager, [13](#)
- FilePath
  - LogManager::LogManager, [15](#)
  - LogManager::LoggingFunctions, [12](#)
- Internal/Logger.cs, [17](#)
- Log
  - LogManager::LogManager, [13](#)
  - LogManager::LoggingFunctions, [11](#)
- LogLevel
  - LogManager, [7](#)
  - LogManager::LogManager, [15](#)
- LogManager, [7](#)
  - Console, [8](#)
  - Date, [7](#)
  - Debug, [7](#)
  - Error, [8](#)
  - Exception, [8](#)
  - File, [8](#)
  - LogLevel, [7](#)
  - LogManager::LogManager, [13](#)
  - LogTarget, [8](#)
  - LogfileNameType, [7](#)
  - Rolling, [7](#)
  - Status, [8](#)
  - Trace, [7](#)
- LogManager.LogManager, [13](#)
- LogManager.LoggingFunctions, [9](#)

- LogManager::LogManager
  - FileName, [13](#)
  - FileNameType, [13](#)
  - FilePath, [15](#)
  - Log, [13](#)
  - LogLevel, [15](#)
  - LogManager, [13](#)
  - Target, [15](#)
- LogManager::LoggingFunctions
  - Debug, [10](#)
  - Error, [10](#)
  - Exception, [10](#)
  - FileName, [12](#)
  - FilePath, [12](#)
  - Log, [11](#)
  - LogTarget, [12](#)
  - LogfileNameType, [12](#)
  - SetLogLevel, [12](#)
  - Status, [11](#)
  - Trace, [12](#)
- LogTarget
  - LogManager, [8](#)
  - LogManager::LoggingFunctions, [12](#)
- LogfileNameType
  - LogManager, [7](#)
  - LogManager::LoggingFunctions, [12](#)
- LoggingFunctions.cs, [17](#)
- Properties/AssemblyInfo.cs, [18](#)
- Rolling
  - LogManager, [7](#)
- SetLogLevel
  - LogManager::LoggingFunctions, [12](#)
- Status
  - LogManager, [8](#)
  - LogManager::LoggingFunctions, [11](#)
- Target
  - LogManager::LogManager, [15](#)
- Trace
  - LogManager, [7](#)
  - LogManager::LoggingFunctions, [12](#)