

Utils

1.9.0.12

by Teodor Krastev for Imperial College London

1 Namespace Index	1
1.1 Packages	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	7
4.1 File List	7
5 Namespace Documentation	9
5.1 PipesServerNS Namespace Reference	9
5.1.1 Function Documentation	9
5.1.1.1 DelegateMessage()	9
5.2 UtilsNS Namespace Reference	9
6 Class Documentation	11
6.1 UtilsNS.AutoFileLogger Class Reference	11
6.1.1 Detailed Description	12
6.1.2 Constructor & Destructor Documentation	12
6.1.2.1 AutoFileLogger()	12
6.1.3 Member Function Documentation	12
6.1.3.1 DropLastChar()	12
6.1.3.2 Flush()	12
6.1.3.3 log() [1/2]	12
6.1.3.4 log() [2/2]	13
6.1.4 Member Data Documentation	13
6.1.4.1 bufferLimit	13
6.1.4.2 defaultExt	13
6.1.4.3 header	13
6.1.4.4 stw	13
6.1.5 Property Documentation	13
6.1.5.1 AutoSaveFileName	14
6.1.5.2 bufferCharSize	14
6.1.5.3 bufferSize	14
6.1.5.4 Enabled	14
6.1.5.5 missingData	14
6.1.5.6 prefix	14
6.1.5.7 writing	15
6.2 UtilsNS.baseMMscan Class Reference	15
6.2.1 Detailed Description	15
6.2.2 Member Function Documentation	15

6.2.2.1 getAsString()	16
6.2.3 Property Documentation	16
6.2.3.1 sBy	16
6.2.3.2 sFrom	16
6.2.3.3 sParam	16
6.2.3.4 sTo	16
6.2.3.5 Value	16
6.3 UtilsNS.DictFileLogger Class Reference	17
6.3.1 Detailed Description	17
6.3.2 Constructor & Destructor Documentation	17
6.3.2.1 DictFileLogger()	17
6.3.3 Member Function Documentation	18
6.3.3.1 dictLog() [1/3]	18
6.3.3.2 dictLog() [2/3]	18
6.3.3.3 dictLog() [3/3]	18
6.3.3.4 setMMexecAsHeader()	19
6.3.3.5 writeHeader()	19
6.4 UtilsNS.DictFileReader Class Reference	19
6.4.1 Detailed Description	20
6.4.2 Constructor & Destructor Documentation	20
6.4.2.1 DictFileReader()	20
6.4.3 Member Function Documentation	20
6.4.3.1 doubleIterator()	21
6.4.3.2 stringIterator()	21
6.4.4 Member Data Documentation	21
6.4.4.1 counter	21
6.4.4.2 header	22
6.4.4.3 record	22
6.4.4.4 subheaders	22
6.5 UtilsNS.FileLogger Class Reference	22
6.5.1 Detailed Description	23
6.5.2 Constructor & Destructor Documentation	23
6.5.2.1 FileLogger()	23
6.5.3 Member Function Documentation	23
6.5.3.1 CreateLogger()	24
6.5.3.2 log() [1/2]	24
6.5.3.3 log() [2/2]	24
6.5.3.4 writeHeader()	24
6.5.4 Member Data Documentation	25
6.5.4.1 defaultExt	25
6.5.4.2 header	25
6.5.4.3 stw	25

6.5.4.4 subheaders	25
6.5.5 Property Documentation	25
6.5.5.1 Enabled	26
6.5.5.2 LogFilename	26
6.5.5.3 missingData	26
6.5.5.4 prefix	26
6.5.5.5 reqFilename	26
6.5.5.6 writing	26
6.6 UtilsNS.memLog Class Reference	27
6.6.1 Detailed Description	27
6.6.2 Constructor & Destructor Documentation	27
6.6.2.1 memLog()	27
6.6.3 Member Function Documentation	27
6.6.3.1 log()	28
6.6.4 Member Data Documentation	28
6.6.4.1 Enabled	28
6.7 UtilsNS.MMexec Class Reference	28
6.7.1 Detailed Description	29
6.7.2 Constructor & Destructor Documentation	29
6.7.2.1 MMexec()	29
6.7.3 Member Function Documentation	29
6.7.3.1 Abort()	29
6.7.3.2 Assign()	30
6.7.3.3 Clear()	30
6.7.3.4 Clone()	30
6.7.4 Member Data Documentation	31
6.7.4.1 prms	31
6.7.5 Property Documentation	31
6.7.5.1 cmd	31
6.7.5.2 id	31
6.7.5.3 mmexec	31
6.7.5.4 sender	31
6.8 UtilsNS.MMscan Class Reference	32
6.8.1 Detailed Description	32
6.8.2 Constructor & Destructor Documentation	33
6.8.2.1 MMscan()	33
6.8.3 Member Function Documentation	33
6.8.3.1 Assign()	33
6.8.3.2 Check()	33
6.8.3.3 Clone()	34
6.8.3.4 FromDictionary()	34
6.8.3.5 isFirstValue()	34

6.8.3.6 isLastValue()	34
6.8.3.7 Next()	35
6.8.3.8 TestInit()	35
6.8.3.9 ToDictionary()	35
6.8.4 Member Data Documentation	35
6.8.4.1 NextInChain	35
6.8.5 Property Documentation	36
6.8.5.1 AsString	36
6.8.5.2 groupID	36
6.9 PipesServerNS.PipeServer Class Reference	36
6.9.1 Detailed Description	36
6.9.2 Member Function Documentation	36
6.9.2.1 Listen()	37
6.9.3 Event Documentation	37
6.9.3.1 PipeMessage	37
6.10 UtilsNS.RemoteMessaging Class Reference	37
6.10.1 Detailed Description	38
6.10.2 Constructor & Destructor Documentation	38
6.10.2.1 RemoteMessaging()	38
6.10.3 Member Function Documentation	39
6.10.3.1 ActiveComm()	39
6.10.3.2 ActiveCommHandler()	39
6.10.3.3 AsyncSent()	39
6.10.3.4 AsyncSentHandler()	40
6.10.3.5 CheckConnection()	40
6.10.3.6 DoEvents()	40
6.10.3.7 elapsedTime()	40
6.10.3.8 ExitFrame()	41
6.10.3.9 Receive()	41
6.10.3.10 ReceiveHandler()	41
6.10.3.11 sendCommand()	41
6.10.3.12 synchroClock()	42
6.10.4 Member Data Documentation	42
6.10.4.1 dTimer	42
6.10.4.2 Enabled	42
6.10.4.3 Log	43
6.10.4.4 stopwatch	43
6.10.5 Property Documentation	43
6.10.5.1 autoCheckPeriod	43
6.10.5.2 Connected	43
6.10.5.3 keyID	43
6.10.5.4 lastRcvMsg	43

6.10.5.5 lastSndMsg	44
6.10.5.6 partner	44
6.10.5.7 partnerPresent	44
6.10.5.8 silentPartner	44
6.10.6 Event Documentation	44
6.10.6.1 OnActiveComm	44
6.10.6.2 OnAsyncSent	44
6.10.6.3 OnReceive	45
6.11 UtilsNS.WaitCursor Class Reference	45
6.11.1 Detailed Description	45
6.11.2 Constructor & Destructor Documentation	45
6.11.2.1 WaitCursor()	45
6.11.3 Member Function Documentation	45
6.11.3.1 Dispose()	45
7 File Documentation	47
7.1 E:/VSprojects/AxelSuite/Utils/PipeServer.cs File Reference	47
7.2 E:/VSprojects/AxelSuite/Utils/RemoteMessaging.cs File Reference	47
7.3 E:/VSprojects/AxelSuite/Utils/UtilsLib.cs File Reference	48
Index	49

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

PipesServerNS	9
UtilsNS	9

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

UtilsNS.AutoFileLogger	11
UtilsNS.baseMMscan	15
UtilsNS.MMscan	32
UtilsNS.DictFileReader	19
UtilsNS.FileLogger	22
UtilsNS.DictFileLogger	17
IDisposable	
UtilsNS.WaitCursor	45
List	
UtilsNS.memLog	27
UtilsNS.MMexec	28
PipesServerNS.PipeServer	36
UtilsNS.RemoteMessaging	37

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

UtilsNS.AutoFileLogger	This is an obsolete version, please use FileLogger below instead !!! Async data storage device first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)	11
UtilsNS.baseMMscan	Properties part of MMscan	15
UtilsNS.DictFileLogger	creates and logs in multicolumn table; record structure is defined in record List of string the column names must be set when created dictLog will extract only the keys with these names in that order	17
UtilsNS.DictFileReader	Read dictionary from multi-column text file (tab separated) format first line header (for conditions) - optional next line column names; header, subheaders & col names are read when instance is created if <code>_record = null</code> then read this row in record if <code>_record</code> has items the record will be the cross-section of <code>_record</code> and column names list (fileRecord)	19
UtilsNS.FileLogger	Async data storage device - new (dec.2018) optimized for speed (7x faster to AutoFilelogger) logger first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)	22
UtilsNS.memLog	In memory meassage log - mostly for debug	27
UtilsNS.MMexec	Class encapsulating one formatted by "Book of JaSON" message	28
UtilsNS.MMscan	Class encapsulating a scan of a parameter	32
PipesServerNS.PipeServer	36
UtilsNS.RemoteMessaging	Messaging service using Windows messages (quickest way possible)	37
UtilsNS.WaitCursor	Hour-glass cursor while waiting for Godot	45

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

E:/VSprojects/AxeISuite/Utils/ PipeServer.cs	47
E:/VSprojects/AxeISuite/Utils/ RemoteMessaging.cs	47
E:/VSprojects/AxeISuite/Utils/ UtilsLib.cs	48

Chapter 5

Namespace Documentation

5.1 PipesServerNS Namespace Reference

Classes

- class [PipeServer](#)

Functions

- delegate void [DelegateMessage](#) (string Reply)

5.1.1 Function Documentation

5.1.1.1 DelegateMessage()

```
delegate void PipesServerNS.DelegateMessage (  
    string Reply )
```

5.2 UtilsNS Namespace Reference

Classes

- class [AutoFileLogger](#)
This is an obsolete version, please use [FileLogger](#) below instead !!! Async data storage device first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)
- class [baseMMscan](#)
Properties part of [MMscan](#)
- class [DictFileLogger](#)

creates and logs in multicolumn table; record structure is defined in record List of string the column names must be set when created dictLog will extract only the keys with these names in that order

- class [DictFileReader](#)

Read dictionary from multi-column text file (tab separated) format first line [header](#) (for conditions) - optional next line column names; header, subheaders & col names are read when instance is created if `_record = null` then read this row in record if `_record` has items the record will be the cross-section of `_record` and column names list (fileRecord)

- class [FileLogger](#)

Async data storage device - new (dec.2018) optimized for speed (7x faster to AutoFilelogger) logger first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

- class [memLog](#)

In memory message log - mostly for debug

- class [MMexec](#)

Class encapsulating one formatted by "Book of JaSON" message

- class [MMscan](#)

Class encapsulating a scan of a parameter

- class [RemoteMessaging](#)

Messaging service using Windows messages (quickest way possible)

- class **Utils**

- class [WaitCursor](#)

Hour-glass cursor while waiting for Godot

Chapter 6

Class Documentation

6.1 UtilsNS.AutoFileLogger Class Reference

This is an obsolete version, please use [FileLogger](#) below instead !!! Async data storage device first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

Public Member Functions

- [AutoFileLogger](#) (string _prefix="", string Filename="")
- int [log](#) (List< string > newItems)
- int [log](#) (string newItem)
- void [DropLastChar](#) ()
- Task [Flush](#) ()

Public Attributes

- string [header](#) = ""
- string [defaultExt](#) = ".ahf"
- int [bufferLimit](#) = 256
- Stopwatch [stw](#)

Properties

- int [bufferSize](#) [get]
- int [bufferCharSize](#) [get]
- string [prefix](#) [get]
- bool [writing](#) [get]
- bool [missingData](#) [get]
- bool [Enabled](#) [get, set]
- string [AutoSaveFileName](#) [get, set]

6.1.1 Detailed Description

This is an obsolete version, please use [FileLogger](#) below instead !!! Async data storage device first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

Definition at line 434 of file UtilsLib.cs.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 AutoFileLogger()

```
UtilsNS.AutoFileLogger.AutoFileLogger (
    string _prefix = "",
    string Filename = "" )
```

Definition at line 449 of file UtilsLib.cs.

6.1.3 Member Function Documentation

6.1.3.1 DropLastChar()

```
void UtilsNS.AutoFileLogger.DropLastChar ( )
```

Definition at line 473 of file UtilsLib.cs.

6.1.3.2 Flush()

```
Task UtilsNS.AutoFileLogger.Flush ( )
```

Definition at line 485 of file UtilsLib.cs.

6.1.3.3 log() [1/2]

```
int UtilsNS.AutoFileLogger.log (
    List< string > newItem )
```

Definition at line 458 of file UtilsLib.cs.

6.1.3.4 log() [2/2]

```
int UtilsNS.AutoFileLogger.log (
    string newItem )
```

Definition at line 465 of file UtilsLib.cs.

6.1.4 Member Data Documentation

6.1.4.1 bufferLimit

```
int UtilsNS.AutoFileLogger.bufferLimit = 256
```

Definition at line 440 of file UtilsLib.cs.

6.1.4.2 defaultExt

```
string UtilsNS.AutoFileLogger.defaultExt = ".ahf"
```

Definition at line 438 of file UtilsLib.cs.

6.1.4.3 header

```
string UtilsNS.AutoFileLogger.header = ""
```

Definition at line 437 of file UtilsLib.cs.

6.1.4.4 stw

```
Stopwatch UtilsNS.AutoFileLogger.stw
```

Definition at line 447 of file UtilsLib.cs.

6.1.5 Property Documentation

6.1.5.1 AutoSaveFileName

```
string UtilsNS.AutoFileLogger.AutoSaveFileName [get], [set]
```

Definition at line 571 of file UtilsLib.cs.

6.1.5.2 bufferCharSize

```
int UtilsNS.AutoFileLogger.bufferCharSize [get]
```

Definition at line 443 of file UtilsLib.cs.

6.1.5.3 bufferSize

```
int UtilsNS.AutoFileLogger.bufferSize [get]
```

Definition at line 442 of file UtilsLib.cs.

6.1.5.4 Enabled

```
bool UtilsNS.AutoFileLogger.Enabled [get], [set]
```

Definition at line 529 of file UtilsLib.cs.

6.1.5.5 missingData

```
bool UtilsNS.AutoFileLogger.missingData [get]
```

Definition at line 446 of file UtilsLib.cs.

6.1.5.6 prefix

```
string UtilsNS.AutoFileLogger.prefix [get]
```

Definition at line 444 of file UtilsLib.cs.

6.1.5.7 writing

```
bool UtilsNS.AutoFileLogger.writing [get]
```

Definition at line 445 of file UtilsLib.cs.

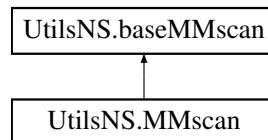
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[UtilsLib.cs](#)

6.2 UtilsNS.baseMMscan Class Reference

Properties part of [MMscan](#)

Inheritance diagram for UtilsNS.baseMMscan:



Public Member Functions

- string [getAsString](#) ()

Properties

- string [sParam](#) [get, set]
- double [sFrom](#) [get, set]
- double [sTo](#) [get, set]
- double [sBy](#) [get, set]
- double [Value](#) [get, set]

6.2.1 Detailed Description

Properties part of [MMscan](#)

Definition at line 527 of file RemoteMessaging.cs.

6.2.2 Member Function Documentation

6.2.2.1 getAsString()

```
string UtilsNS.baseMScan.getAsString ( )
```

Definition at line 535 of file RemoteMessaging.cs.

6.2.3 Property Documentation

6.2.3.1 sBy

```
double UtilsNS.baseMScan.sBy [get], [set]
```

Definition at line 532 of file RemoteMessaging.cs.

6.2.3.2 sFrom

```
double UtilsNS.baseMScan.sFrom [get], [set]
```

Definition at line 530 of file RemoteMessaging.cs.

6.2.3.3 sParam

```
string UtilsNS.baseMScan.sParam [get], [set]
```

Definition at line 529 of file RemoteMessaging.cs.

6.2.3.4 sTo

```
double UtilsNS.baseMScan.sTo [get], [set]
```

Definition at line 531 of file RemoteMessaging.cs.

6.2.3.5 Value

```
double UtilsNS.baseMScan.Value [get], [set]
```

Definition at line 533 of file RemoteMessaging.cs.

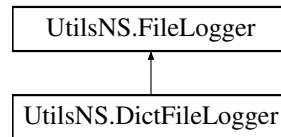
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[RemoteMessaging.cs](#)

6.3 UtilsNS.DictFileLogger Class Reference

creates and logs in multicolumn table; record structure is defined in record List of string the column names must be set when created dictLog will extract only the keys with these names in that order

Inheritance diagram for UtilsNS.DictFileLogger:



Public Member Functions

- [DictFileLogger](#) (string[] _record, string _prefix="", string _reqFilename="")
Class constructor
- void [setMMexecAsHeader](#) (MMexec mme)
When the group MMexec is known
- override void [writeHeader](#) ()
Write the header & subheaders and column names line
- void [dictLog](#) (Dictionary< string, string > dict)
This main methods in three variations, but that is the basic one
- void [dictLog](#) (Dictionary< string, object > dict)
log that way with undefined Values type
- void [dictLog](#) (Dictionary< string, double > dict, string format="")
log that way with double Values type with format

Additional Inherited Members

6.3.1 Detailed Description

creates and logs in multicolumn table; record structure is defined in record List of string the column names must be set when created dictLog will extract only the keys with these names in that order

Definition at line 742 of file UtilsLib.cs.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 DictFileLogger()

```

UtilsNS.DictFileLogger.DictFileLogger (
    string[] _record,
    string _prefix = "",
    string _reqFilename = "" )
  
```

Class constructor

Parameters

<i>_record</i>	Required and fixed column names
<i>_prefix</i>	Ending in case of timestamp name
<i>_reqFilename</i>	If empty timestamp name is generated

Definition at line 751 of file UtilsLib.cs.

6.3.3 Member Function Documentation

6.3.3.1 dictLog() [1/3]

```
void UtilsNS.DictFileLogger.dictLog (  
    Dictionary< string, double > dict,  
    string format = "" )
```

log that way with double Values type with format

Parameters

<i>dict</i>	
<i>format</i>	

Definition at line 828 of file UtilsLib.cs.

6.3.3.2 dictLog() [2/3]

```
void UtilsNS.DictFileLogger.dictLog (  
    Dictionary< string, object > dict )
```

log that way with undefined Values type

Parameters

<i>dict</i>	
-------------	--

Definition at line 813 of file UtilsLib.cs.

6.3.3.3 dictLog() [3/3]

```
void UtilsNS.DictFileLogger.dictLog (  
    Dictionary< string, string > dict )
```

This main methods in three variations, but that is the basic one

Parameters

<i>dict</i>	
-------------	--

Definition at line 796 of file UtilsLib.cs.

6.3.3.4 setMMexecAsHeader()

```
void UtilsNS.DictFileLogger.setMMexecAsHeader (
    MMexec mme )
```

When the group [MMexec](#) is known

Parameters

<i>mme</i>	
------------	--

Definition at line 762 of file UtilsLib.cs.

6.3.3.5 writeHeader()

```
override void UtilsNS.DictFileLogger.writeHeader ( ) [virtual]
```

Write the header & subheaders and column names line

Reimplemented from [UtilsNS.FileLogger](#).

Definition at line 780 of file UtilsLib.cs.

The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[UtilsLib.cs](#)

6.4 UtilsNS.DictFileReader Class Reference

Read dictionary from multi-column text file (tab separated) format first line [header](#) (for conditions) - optional next line column names; header, subheaders & col names are read when instance is created if `_record = null` then read this row in record if `_record` has items the record will be the cross-section of `_record` and column names list (fileRecord)

Public Member Functions

- [DictFileReader](#) (string Filename, string[] strArr=null)
Class constructor
- bool [stringIterator](#) (ref Dictionary< string, string > rslt)
returns one line (row) as (column.name , cell.value) dictionary
- bool [doubleIterator](#) (ref Dictionary< string, double > rslt)
same as above but Values are double

Public Attributes

- string [header](#)
- List< string > [subheaders](#)
- int [counter](#) = 0
- List< string > [record](#)

6.4.1 Detailed Description

Read dictionary from multi-column text file (tab separated) format first line [header](#) (for conditions) - optional next line column names; header, subheaders & col names are read when instance is created if `_record` = null then read this row in record if `_record` has items the record will be the cross-section of `_record` and column names list (fileRecord)

Definition at line 848 of file UtilsLib.cs.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 DictFileReader()

```
UtilsNS.DictFileReader.DictFileReader (
    string Filename,
    string[] strArr = null)
```

Class constructor

Parameters

<i>Filename</i>	File must exists
<i>strArr</i>	Array of column names

Definition at line 859 of file UtilsLib.cs.

6.4.3 Member Function Documentation

6.4.3.1 doubleIterator()

```
bool UtilsNS.DictFileReader.doubleIterator (
    ref Dictionary< string, double > rslt )
```

same as above but Values are double

Parameters

<i>rslt</i>	
-------------	--

Returns

Definition at line 930 of file UtilsLib.cs.

6.4.3.2 stringIterator()

```
bool UtilsNS.DictFileReader.stringIterator (
    ref Dictionary< string, string > rslt )
```

returns one line (row) as (column.name , cell.value) dictionary

Parameters

<i>rslt</i>	one table row
-------------	---------------

Returns

if we can go again

Definition at line 900 of file UtilsLib.cs.

6.4.4 Member Data Documentation

6.4.4.1 counter

```
int UtilsNS.DictFileReader.counter = 0
```

Definition at line 852 of file UtilsLib.cs.

6.4.4.2 header

```
string UtilsNS.DictFileReader.header
```

Definition at line 850 of file UtilsLib.cs.

6.4.4.3 record

```
List<string> UtilsNS.DictFileReader.record
```

Definition at line 853 of file UtilsLib.cs.

6.4.4.4 subheaders

```
List<string> UtilsNS.DictFileReader.subheaders
```

Definition at line 851 of file UtilsLib.cs.

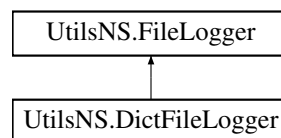
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[UtilsLib.cs](#)

6.5 UtilsNS.FileLogger Class Reference

Async data storage device - new (dec.2018) optimized for speed (7x faster to AutoFilelogger) logger first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

Inheritance diagram for UtilsNS.FileLogger:



Public Member Functions

- [FileLogger](#) (string _prefix="", string _reqFilename="")
Class constructor
- void [log](#) (List< string > newItems)
The main call if you have List of strings
- void [log](#) (string newItem)
That's the main method
- void [CreateLogger](#) (string filePath)
Create actual asynchronous logger
- virtual void [writeHeader](#) ()
Write a header and subheaders with #

Public Attributes

- string `header` = ""
- List< string > `subheaders`
- string `defaultExt` = ".ahf"
- Stopwatch `stw`

Properties

- string `prefix` [get]
- string `reqFilename` [get]
- bool `writing` [get]
- bool `missingData` [get]
- string `LogFilename` [get]
- bool `Enabled` [get, set]

Switch this on to create the file and start to accept logs switch it off to flush the buffer and close the file

6.5.1 Detailed Description

Async data storage device - new (dec.2018) optimized for speed (7x faster to AutoFilelogger) logger first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

Definition at line 590 of file UtilsLib.cs.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 FileLogger()

```
UtilsNS.FileLogger.FileLogger (
    string _prefix = "",
    string _reqFilename = "" )
```

Class constructor

Parameters

<code>_prefix</code>	
<code>_reqFilename</code>	

Definition at line 608 of file UtilsLib.cs.

6.5.3 Member Function Documentation

6.5.3.1 CreateLogger()

```
void UtilsNS.FileLogger.CreateLogger (
    string filePath )
```

Create actual asynchronous logger

Parameters

<i>filePath</i>	
-----------------	--

Definition at line 658 of file UtilsLib.cs.

6.5.3.2 log() [1/2]

```
void UtilsNS.FileLogger.log (
    List< string > newItems )
```

The main call if you have List of strings

Parameters

<i>newItems</i>	
-----------------	--

Definition at line 621 of file UtilsLib.cs.

6.5.3.3 log() [2/2]

```
void UtilsNS.FileLogger.log (
    string newItem )
```

That's the main method

Parameters

<i>newItem</i>	
----------------	--

Definition at line 632 of file UtilsLib.cs.

6.5.3.4 writeHeader()

```
virtual void UtilsNS.FileLogger.writeHeader ( ) [virtual]
```


Write a header and subheaders with #

Reimplemented in [UtilsNS.DictFileLogger](#).

Definition at line 687 of file UtilsLib.cs.

6.5.4 Member Data Documentation

6.5.4.1 defaultExt

```
string UtilsNS.FileLogger.defaultExt = ".ahf"
```

Definition at line 595 of file UtilsLib.cs.

6.5.4.2 header

```
string UtilsNS.FileLogger.header = ""
```

Definition at line 593 of file UtilsLib.cs.

6.5.4.3 stw

```
Stopwatch UtilsNS.FileLogger.stw
```

Definition at line 601 of file UtilsLib.cs.

6.5.4.4 subheaders

```
List<string> UtilsNS.FileLogger.subheaders
```

Definition at line 594 of file UtilsLib.cs.

6.5.5 Property Documentation

6.5.5.1 Enabled

```
bool UtilsNS.FileLogger.Enabled [get], [set]
```

Switch this on to create the file and start to accept logs switch it off to flush the buffer and close the file

Definition at line 701 of file UtilsLib.cs.

6.5.5.2 LogFilename

```
string UtilsNS.FileLogger.LogFilename [get]
```

Definition at line 675 of file UtilsLib.cs.

6.5.5.3 missingData

```
bool UtilsNS.FileLogger.missingData [get]
```

Definition at line 600 of file UtilsLib.cs.

6.5.5.4 prefix

```
string UtilsNS.FileLogger.prefix [get]
```

Definition at line 597 of file UtilsLib.cs.

6.5.5.5 reqFilename

```
string UtilsNS.FileLogger.reqFilename [get]
```

Definition at line 598 of file UtilsLib.cs.

6.5.5.6 writing

```
bool UtilsNS.FileLogger.writing [get]
```

Definition at line 599 of file UtilsLib.cs.

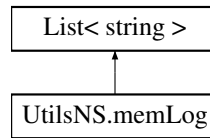
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[UtilsLib.cs](#)

6.6 UtilsNS.memLog Class Reference

In memory message log - mostly for debug

Inheritance diagram for UtilsNS.memLog:



Public Member Functions

- `memLog` (int depth=32)
- void `log` (string txt)

Public Attributes

- bool `Enabled` = true

6.6.1 Detailed Description

In memory message log - mostly for debug

Definition at line 20 of file RemoteMessaging.cs.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 memLog()

```
UtilsNS.memLog.memLog (  
    int depth = 32 )
```

Definition at line 24 of file RemoteMessaging.cs.

6.6.3 Member Function Documentation

6.6.3.1 log()

```
void UtilsNS.memLog.log (
    string txt )
```

Definition at line 28 of file RemoteMessaging.cs.

6.6.4 Member Data Documentation

6.6.4.1 Enabled

```
bool UtilsNS.memLog.Enabled = true
```

Definition at line 22 of file RemoteMessaging.cs.

The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[RemoteMessaging.cs](#)

6.7 UtilsNS.MMexec Class Reference

Class encapsulating one formatted by "Book of JaSON" message

Public Member Functions

- [MMexec](#) (string Caption="", string Sender="", string Command="", int ID=-1)
Class constructor
- void [Clear](#) ()
Clean all up
- void [Assign](#) ([MMexec](#) src)
Assign src to this
- [MMexec Clone](#) ()
Clone this - copy all the props to a new instance
- string [Abort](#) (string Sender="")
Standard Abort (Cancel) message

Public Attributes

- Dictionary< string, object > [prms](#)

Properties

- string `mmexec` [get, set]
- string `sender` [get, set]
- string `cmd` [get, set]
- int `id` [get, set]

6.7.1 Detailed Description

Class encapsulating one formatted by "Book of JaSON" message

Definition at line 446 of file RemoteMessaging.cs.

6.7.2 Constructor & Destructor Documentation

6.7.2.1 MMexec()

```
UtilsNS.MMexec.MMexec (
    string Caption = "",
    string Sender = "",
    string Command = "",
    int ID = -1 )
```

Class constructor

Parameters

<i>Caption</i>	
<i>Sender</i>	
<i>Command</i>	
<i>ID</i>	

Definition at line 462 of file RemoteMessaging.cs.

6.7.3 Member Function Documentation

6.7.3.1 Abort()

```
string UtilsNS.MMexec.Abort (
    string Sender = "" )
```

Standard Abort (Cancel) message

Parameters

<i>Sender</i>	
---------------	--

Returns

Definition at line 514 of file RemoteMessaging.cs.

6.7.3.2 Assign()

```
void UtilsNS.MMexec.Assign (
    MMexec src )
```

Assign src to this

Parameters

<i>src</i>	
------------	--

Definition at line 486 of file RemoteMessaging.cs.

6.7.3.3 Clear()

```
void UtilsNS.MMexec.Clear ( )
```

Clean all up

Definition at line 474 of file RemoteMessaging.cs.

6.7.3.4 Clone()

```
MMexec UtilsNS.MMexec.Clone ( )
```

Clone this - copy all the props to a new instance

Returns

Definition at line 498 of file RemoteMessaging.cs.

6.7.4 Member Data Documentation

6.7.4.1 prms

`Dictionary<string, object> UtilsNS.MMexec.prms`

Definition at line 453 of file RemoteMessaging.cs.

6.7.5 Property Documentation

6.7.5.1 cmd

`string UtilsNS.MMexec.cmd [get], [set]`

Definition at line 451 of file RemoteMessaging.cs.

6.7.5.2 id

`int UtilsNS.MMexec.id [get], [set]`

Definition at line 452 of file RemoteMessaging.cs.

6.7.5.3 mmexec

`string UtilsNS.MMexec.mmexec [get], [set]`

Definition at line 449 of file RemoteMessaging.cs.

6.7.5.4 sender

`string UtilsNS.MMexec.sender [get], [set]`

Definition at line 450 of file RemoteMessaging.cs.

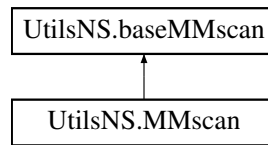
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[RemoteMessaging.cs](#)

6.8 UtilsNS.MMscan Class Reference

Class encapsulating a scan of a parameter

Inheritance diagram for UtilsNS.MMscan:



Public Member Functions

- [MMscan](#) (string _groupId="", string _sParam="", double _sFrom=double.NaN, double _sTo=double.NaN, double _sBy=double.NaN, double _Value=double.NaN)
- bool [Check](#) ()
- bool [isFirstValue](#) ()
- bool [isLastValue](#) ()
- bool [Next](#) ()
 - The main call when scan, it works multiscan (chain by NextInChain of scans) mode The latter uses recursive (on class level) call the same Next down the chain*
- void [Assign](#) ([MMscan](#) src)
 - Assign src to this*
- [MMscan Clone](#) ()
 - Clone this - copy all the props to a new instance*
- void [TestInit](#) ()
 - Fill in something - only for tests*
- void [ToDictionary](#) (ref Dictionary< string, object > dict)
 - Fill in dictionary with props*
- bool [FromDictionary](#) (Dictionary< string, object > dict)
 - Update props from a dictionary*

Public Attributes

- [MMscan NextInChain](#) = null

Properties

- string [groupId](#) [get, set]
- string [AsString](#) [get, set]
 - Package / unpackage the scan params as string*

6.8.1 Detailed Description

Class encapsulating a scan of a parameter

Definition at line 544 of file RemoteMessaging.cs.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 MMscan()

```
UtilsNS.MMscan.MMscan (
    string _groupID = "",
    string _sParam = "",
    double _sFrom = double.NaN,
    double _sTo = double.NaN,
    double _sBy = double.NaN,
    double _Value = double.NaN )
```

Definition at line 548 of file RemoteMessaging.cs.

6.8.3 Member Function Documentation

6.8.3.1 Assign()

```
void UtilsNS.MMscan.Assign (
    MMscan src )
```

Assign src to this

Parameters

<i>src</i>	
------------	--

Definition at line 622 of file RemoteMessaging.cs.

6.8.3.2 Check()

```
bool UtilsNS.MMscan.Check ( )
```

Definition at line 552 of file RemoteMessaging.cs.

6.8.3.3 Clone()

```
MMscan UtilsNS.MMscan.Clone ( )
```

Clone this - copy all the props to a new instance

Returns

Definition at line 630 of file RemoteMessaging.cs.

6.8.3.4 FromDictionary()

```
bool UtilsNS.MMscan.FromDictionary (
    Dictionary< string, object > dict )
```

Update props from a dictionary

Parameters

<i>dict</i>	
-------------	--

Returns

Definition at line 664 of file RemoteMessaging.cs.

6.8.3.5 isFirstValue()

```
bool UtilsNS.MMscan.isFirstValue ( )
```

Definition at line 560 of file RemoteMessaging.cs.

6.8.3.6 isLastValue()

```
bool UtilsNS.MMscan.isLastValue ( )
```

Definition at line 565 of file RemoteMessaging.cs.

6.8.3.7 Next()

```
bool UtilsNS.MMscan.Next ( )
```

The main call when scan, it works multiscan (chain by NextInChain of scans) mode The latter uses recursive (on class level) call the same Next down the chain

Returns

Definition at line 577 of file RemoteMessaging.cs.

6.8.3.8 TestInit()

```
void UtilsNS.MMscan.TestInit ( )
```

Fill in something - only for tests

Definition at line 638 of file RemoteMessaging.cs.

6.8.3.9 ToDictionary()

```
void UtilsNS.MMscan.ToDictionary (
    ref Dictionary< string, object > dict )
```

Fill in dictionary with props

Parameters

<i>dict</i>	
-------------	--

Definition at line 650 of file RemoteMessaging.cs.

6.8.4 Member Data Documentation

6.8.4.1 NextInChain

```
MMscan UtilsNS.MMscan.NextInChain = null
```

Definition at line 570 of file RemoteMessaging.cs.

6.8.5 Property Documentation

6.8.5.1 AsString

```
string UtilsNS.MMscan.AsString [get], [set]
```

Package / unpackage the scan params as string

Definition at line 601 of file RemoteMessaging.cs.

6.8.5.2 groupID

```
string UtilsNS.MMscan.groupID [get], [set]
```

Definition at line 546 of file RemoteMessaging.cs.

The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Utils/[RemoteMessaging.cs](#)

6.9 PipesServerNS.PipeServer Class Reference

Public Member Functions

- void [Listen](#) (string PipeName)

Events

- [DelegateMessage](#) PipeMessage

6.9.1 Detailed Description

Definition at line 14 of file PipeServer.cs.

6.9.2 Member Function Documentation

6.9.2.1 Listen()

```
void PipesServerNS.PipeServer.Listen (
    string PipeName )
```

Definition at line 19 of file PipeServer.cs.

6.9.3 Event Documentation

6.9.3.1 PipeMessage

```
DelegateMessage PipesServerNS.PipeServer.PipeMessage
```

Definition at line 16 of file PipeServer.cs.

The documentation for this class was generated from the following file:

- [E:/VSprojects/AxelSuite/Utils/PipeServer.cs](#)

6.10 UtilsNS.RemoteMessaging Class Reference

Messaging service using Windows messages (quickest way possible)

Classes

- struct **COPYDATASTRUCT**
The COPYDATASTRUCT structure contains data to be passed to another application by the WM_COPYDATA message.
- struct **MyStruct**
- class **NativeMethod**

Public Member Functions

- [RemoteMessaging](#) (string Partner, int _keyID=666)
Establish communication channel
- double [elapsedTime](#) ()
Precise timer for accurate time stamps
- void [DoEvents](#) ()
- object [ExitFrame](#) (object f)
- bool [synchroClock](#) (bool force=false)
Synchronizing stopwatches of two apps with precision less than 15us
- delegate bool [ReceiveHandler](#) (string msg)
- delegate void [ActiveCommHandler](#) (bool active, bool forced)
- delegate void [AsyncSentHandler](#) (bool OK, string json2send)
- bool [sendCommand](#) (string msg, int delay=0)
The wrapper around the Windows messaging send part
- bool [CheckConnection](#) (bool forced=false)
ping<->pong to check the connection

Public Attributes

- [memLog](#) [Log](#)
- Stopwatch [stopwatch](#)
- DispatcherTimer [dTimer](#)
- bool [Enabled](#) = true

Protected Member Functions

- bool [Receive](#) (string msg)
Receive message
- void [ActiveComm](#) (bool active, bool forced)
When the channel opens/closes
- void [AsyncSent](#) (bool OK, string json2send)

Properties

- string [partner](#) [get]
- bool [silentPartner](#) [get, set]
- string [lastRcvMsg](#) [get]
- string [lastSndMsg](#) [get]
- int [autoCheckPeriod](#) [get, set]
- int [keyID](#) [get]
- bool [Connected](#) [get]
- bool [partnerPresent](#) [get]

Events

- [ReceiveHandler](#) [OnReceive](#)
- [ActiveCommHandler](#) [OnActiveComm](#)
- [AsyncSentHandler](#) [OnAsyncSent](#)

6.10.1 Detailed Description

Messaging service using Windows messages (quickest way possible)

Definition at line 39 of file RemoteMessaging.cs.

6.10.2 Constructor & Destructor Documentation

6.10.2.1 RemoteMessaging()

```
UtilsNS.RemoteMessaging.RemoteMessaging (
    string Partner,
    int _keyID = 666 )
```

Establish communication channel

Parameters

<i>Partner</i>	The title (caption) of the application
<i>_keyID</i>	Similar to port - one app can have more than one channel with diff. keyID

Definition at line 74 of file RemoteMessaging.cs.

6.10.3 Member Function Documentation

6.10.3.1 ActiveComm()

```
void UtilsNS.RemoteMessaging.ActiveComm (
    bool active,
    bool forced ) [protected]
```

When the channel opens/closes

Parameters

<i>active</i>	
<i>forced</i>	

Definition at line 185 of file RemoteMessaging.cs.

6.10.3.2 ActiveCommHandler()

```
delegate void UtilsNS.RemoteMessaging.ActiveCommHandler (
    bool active,
    bool forced )
```

6.10.3.3 AsyncSent()

```
void UtilsNS.RemoteMessaging.AsyncSent (
    bool OK,
    string json2send ) [protected]
```

Definition at line 269 of file RemoteMessaging.cs.

6.10.3.4 AsyncSentHandler()

```
delegate void UtilsNS.RemoteMessaging.AsyncSentHandler (
    bool OK,
    string json2send )
```

6.10.3.5 CheckConnection()

```
bool UtilsNS.RemoteMessaging.CheckConnection (
    bool forced = false )
```

ping<->pong to check the connection

Parameters

<i>forced</i>	
---------------	--

Returns

Definition at line 352 of file RemoteMessaging.cs.

6.10.3.6 DoEvents()

```
void UtilsNS.RemoteMessaging.DoEvents ( )
```

Definition at line 121 of file RemoteMessaging.cs.

6.10.3.7 elapsedTime()

```
double UtilsNS.RemoteMessaging.elapsedTime ( )
```

Precise timer for accurate time stamps

Returns

Definition at line 115 of file RemoteMessaging.cs.

6.10.3.8 ExitFrame()

```
object UtilsNS.RemoteMessaging.ExitFrame (
    object f )
```

Definition at line 129 of file RemoteMessaging.cs.

6.10.3.9 Receive()

```
bool UtilsNS.RemoteMessaging.Receive (
    string msg ) [protected]
```

Receive message

Parameters

<i>msg</i>	
------------	--

Returns

Definition at line 172 of file RemoteMessaging.cs.

6.10.3.10 ReceiveHandler()

```
delegate bool UtilsNS.RemoteMessaging.ReceiveHandler (
    string msg )
```

6.10.3.11 sendCommand()

```
bool UtilsNS.RemoteMessaging.sendCommand (
    string msg,
    int delay = 0 )
```

The wrapper around the Windows messaging send part

Parameters

<i>msg</i>	
<i>delay</i>	

Returns

Definition at line 280 of file RemoteMessaging.cs.

6.10.3.12 synchroClock()

```
bool UtilsNS.RemoteMessaging.synchroClock (
    bool force = false )
```

Synchronizing stopwatches of two apps with precision less than 15us

Parameters

<i>force</i>	
--------------	--

Returns

Definition at line 140 of file RemoteMessaging.cs.

6.10.4 Member Data Documentation

6.10.4.1 dTimer

```
DispatcherTimer UtilsNS.RemoteMessaging.dTimer
```

Definition at line 49 of file RemoteMessaging.cs.

6.10.4.2 Enabled

```
bool UtilsNS.RemoteMessaging.Enabled = true
```

Definition at line 57 of file RemoteMessaging.cs.

6.10.4.3 Log

`memLog` UtilsNS.RemoteMessaging.Log

Definition at line 46 of file RemoteMessaging.cs.

6.10.4.4 stopwatch

`Stopwatch` UtilsNS.RemoteMessaging.stopwatch

Definition at line 47 of file RemoteMessaging.cs.

6.10.5 Property Documentation

6.10.5.1 autoCheckPeriod

`int` UtilsNS.RemoteMessaging.autoCheckPeriod [get], [set]

Definition at line 52 of file RemoteMessaging.cs.

6.10.5.2 Connected

`bool` UtilsNS.RemoteMessaging.Connected [get]

Definition at line 59 of file RemoteMessaging.cs.

6.10.5.3 keyID

`int` UtilsNS.RemoteMessaging.keyID [get]

Definition at line 58 of file RemoteMessaging.cs.

6.10.5.4 lastRcvMsg

`string` UtilsNS.RemoteMessaging.lastRcvMsg [get]

Definition at line 44 of file RemoteMessaging.cs.

6.10.5.5 lastSndMsg

```
string UtilsNS.RemoteMessaging.lastSndMsg [get]
```

Definition at line 45 of file RemoteMessaging.cs.

6.10.5.6 partner

```
string UtilsNS.RemoteMessaging.partner [get]
```

Definition at line 41 of file RemoteMessaging.cs.

6.10.5.7 partnerPresent

```
bool UtilsNS.RemoteMessaging.partnerPresent [get]
```

Definition at line 61 of file RemoteMessaging.cs.

6.10.5.8 silentPartner

```
bool UtilsNS.RemoteMessaging.silentPartner [get], [set]
```

Definition at line 42 of file RemoteMessaging.cs.

6.10.6 Event Documentation

6.10.6.1 OnActiveComm

```
ActiveCommHandler UtilsNS.RemoteMessaging.OnActiveComm
```

Definition at line 179 of file RemoteMessaging.cs.

6.10.6.2 OnAsyncSent

```
AsyncSentHandler UtilsNS.RemoteMessaging.OnAsyncSent
```

Definition at line 268 of file RemoteMessaging.cs.

6.10.6.3 OnReceive

[ReceiveHandler](#) `UtilsNS.RemoteMessaging.OnReceive`

Definition at line 166 of file `RemoteMessaging.cs`.

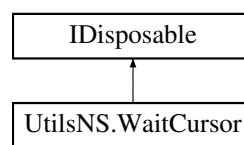
The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Utils/RemoteMessaging.cs`

6.11 UtilsNS.WaitCursor Class Reference

Hour-glass cursor while waiting for Godot

Inheritance diagram for `UtilsNS.WaitCursor`:



Public Member Functions

- [WaitCursor](#) ()
- void [Dispose](#) ()

6.11.1 Detailed Description

Hour-glass cursor while waiting for Godot

Definition at line 955 of file `UtilsLib.cs`.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 WaitCursor()

`UtilsNS.WaitCursor.WaitCursor ()`

Definition at line 958 of file `UtilsLib.cs`.

6.11.3 Member Function Documentation

6.11.3.1 Dispose()

`void UtilsNS.WaitCursor.Dispose ()`

Definition at line 966 of file `UtilsLib.cs`.

The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Utils/UtilsLib.cs`

Chapter 7

File Documentation

7.1 E:/VSprojects/AxelSuite/Utils/PipeServer.cs File Reference

Classes

- class [PipesServerNS.PipeServer](#)

Namespaces

- namespace [PipesServerNS](#)

Functions

- delegate void [PipesServerNS.DelegateMessage](#) (string Reply)

7.2 E:/VSprojects/AxelSuite/Utils/RemoteMessaging.cs File Reference

Classes

- class [UtilsNS.memLog](#)
In memory message log - mostly for debug
- class [UtilsNS.RemoteMessaging](#)
Messaging service using Windows messages (quickest way possible)
- struct **UtilsNS.RemoteMessaging.MyStruct**
- struct **UtilsNS.RemoteMessaging.COPYDATASTRUCT**
The COPYDATASTRUCT structure contains data to be passed to another application by the WM_COPYDATA message.
- class **UtilsNS.RemoteMessaging.NativeMethod**
- class [UtilsNS.MMexec](#)
Class encapsulating one formatted by "Book of JaSON" message
- class [UtilsNS.baseMMscan](#)
Properties part of MMscan
- class [UtilsNS.MMscan](#)
Class encapsulating a scan of a parameter

Namespaces

- namespace [UtilsNS](#)

7.3 E:/VSprojects/AxelSuite/Utils/UtilsLib.cs File Reference

Classes

- class **UtilsNS.Utils**
- class [UtilsNS.AutoFileLogger](#)

This is an obsolete version, please use [FileLogger](#) below instead !!! Async data storage device first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

- class [UtilsNS.FileLogger](#)

Async data storage device - new (dec.2018) optimized for speed (7x faster to AutoFilelogger) logger first you set the full path of the file, otherwise it will save in data dir under date-time file name when you want the logging to start you set Enabled to true at the end you set Enabled to false (that will flush the buffer to HD)

- class [UtilsNS.DictFileLogger](#)

creates and logs in multicolumn table; record structure is defined in record List of string the column names must be set when created dictLog will extract only the keys with these names in that order

- class [UtilsNS.DictFileReader](#)

Read dictionary from multi-column text file (tab separated) format first line [header](#) (for conditions) - optional next line column names; header, subheaders & col names are read when instance is created if `_record = null` then read this row in record if `_record` has items the record will be the cross-section of `_record` and column names list (fileRecord)

- class [UtilsNS.WaitCursor](#)

Hour-glass cursor while waiting for Godot

Namespaces

- namespace [UtilsNS](#)

Index

- Abort
 - UtilsNS.MMexec, [29](#)
- ActiveComm
 - UtilsNS.RemoteMessaging, [39](#)
- ActiveCommHandler
 - UtilsNS.RemoteMessaging, [39](#)
- Assign
 - UtilsNS.MMexec, [30](#)
 - UtilsNS.MMscan, [33](#)
- AsString
 - UtilsNS.MMscan, [36](#)
- AsyncSent
 - UtilsNS.RemoteMessaging, [39](#)
- AsyncSentHandler
 - UtilsNS.RemoteMessaging, [39](#)
- autoCheckPeriod
 - UtilsNS.RemoteMessaging, [43](#)
- AutoFileLogger
 - UtilsNS.AutoFileLogger, [12](#)
- AutoSaveFileName
 - UtilsNS.AutoFileLogger, [13](#)
- bufferCharSize
 - UtilsNS.AutoFileLogger, [14](#)
- bufferLimit
 - UtilsNS.AutoFileLogger, [13](#)
- bufferSize
 - UtilsNS.AutoFileLogger, [14](#)
- Check
 - UtilsNS.MMscan, [33](#)
- CheckConnection
 - UtilsNS.RemoteMessaging, [40](#)
- Clear
 - UtilsNS.MMexec, [30](#)
- Clone
 - UtilsNS.MMexec, [30](#)
 - UtilsNS.MMscan, [33](#)
- cmd
 - UtilsNS.MMexec, [31](#)
- Connected
 - UtilsNS.RemoteMessaging, [43](#)
- counter
 - UtilsNS.DictFileReader, [21](#)
- CreateLogger
 - UtilsNS.FileLogger, [23](#)
- defaultExt
 - UtilsNS.AutoFileLogger, [13](#)
 - UtilsNS.FileLogger, [25](#)
- DelegateMessage
 - PipesServerNS, [9](#)
- DictFileLogger
 - UtilsNS.DictFileLogger, [17](#)
- DictFileReader
 - UtilsNS.DictFileReader, [20](#)
- dictLog
 - UtilsNS.DictFileLogger, [18](#)
- Dispose
 - UtilsNS.WaitCursor, [45](#)
- DoEvents
 - UtilsNS.RemoteMessaging, [40](#)
- doubleIterator
 - UtilsNS.DictFileReader, [20](#)
- DropLastChar
 - UtilsNS.AutoFileLogger, [12](#)
- dTimer
 - UtilsNS.RemoteMessaging, [42](#)
- E:/VSprojects/AxelSuite/Utils/PipeServer.cs, [47](#)
- E:/VSprojects/AxelSuite/Utils/RemoteMessaging.cs, [47](#)
- E:/VSprojects/AxelSuite/Utils/UtilsLib.cs, [48](#)
- elapsedTime
 - UtilsNS.RemoteMessaging, [40](#)
- Enabled
 - UtilsNS.AutoFileLogger, [14](#)
 - UtilsNS.FileLogger, [25](#)
 - UtilsNS.memLog, [28](#)
 - UtilsNS.RemoteMessaging, [42](#)
- ExitFrame
 - UtilsNS.RemoteMessaging, [40](#)
- FileLogger
 - UtilsNS.FileLogger, [23](#)
- Flush
 - UtilsNS.AutoFileLogger, [12](#)
- FromDictionary
 - UtilsNS.MMscan, [34](#)
- getAsString
 - UtilsNS.baseMMscan, [15](#)
- groupID
 - UtilsNS.MMscan, [36](#)
- header
 - UtilsNS.AutoFileLogger, [13](#)
 - UtilsNS.DictFileReader, [21](#)
 - UtilsNS.FileLogger, [25](#)
- id
 - UtilsNS.MMexec, [31](#)

- isFirstValue
 - UtilsNS.MMscan, 34
- isLastValue
 - UtilsNS.MMscan, 34
- keyID
 - UtilsNS.RemoteMessaging, 43
- lastRcvMsg
 - UtilsNS.RemoteMessaging, 43
- lastSndMsg
 - UtilsNS.RemoteMessaging, 43
- Listen
 - PipesServerNS.PipeServer, 36
- Log
 - UtilsNS.RemoteMessaging, 42
- log
 - UtilsNS.AutoFileLogger, 12
 - UtilsNS.FileLogger, 24
 - UtilsNS.memLog, 27
- LogFilename
 - UtilsNS.FileLogger, 26
- memLog
 - UtilsNS.memLog, 27
- missingData
 - UtilsNS.AutoFileLogger, 14
 - UtilsNS.FileLogger, 26
- MMexec
 - UtilsNS.MMexec, 29
- mmexec
 - UtilsNS.MMexec, 31
- MMscan
 - UtilsNS.MMscan, 33
- Next
 - UtilsNS.MMscan, 34
- NextInChain
 - UtilsNS.MMscan, 35
- OnActiveComm
 - UtilsNS.RemoteMessaging, 44
- OnAsyncSent
 - UtilsNS.RemoteMessaging, 44
- OnReceive
 - UtilsNS.RemoteMessaging, 44
- partner
 - UtilsNS.RemoteMessaging, 44
- partnerPresent
 - UtilsNS.RemoteMessaging, 44
- PipeMessage
 - PipesServerNS.PipeServer, 37
- PipesServerNS, 9
 - DelegateMessage, 9
- PipesServerNS.PipeServer, 36
 - Listen, 36
 - PipeMessage, 37
- prefix
 - UtilsNS.AutoFileLogger, 14
- UtilsNS.FileLogger, 26
- prms
 - UtilsNS.MMexec, 31
- Receive
 - UtilsNS.RemoteMessaging, 41
- ReceiveHandler
 - UtilsNS.RemoteMessaging, 41
- record
 - UtilsNS.DictFileReader, 22
- RemoteMessaging
 - UtilsNS.RemoteMessaging, 38
- reqFilename
 - UtilsNS.FileLogger, 26
- sBy
 - UtilsNS.baseMMscan, 16
- sendCommand
 - UtilsNS.RemoteMessaging, 41
- sender
 - UtilsNS.MMexec, 31
- setMMexecAsHeader
 - UtilsNS.DictFileLogger, 19
- sFrom
 - UtilsNS.baseMMscan, 16
- silentPartner
 - UtilsNS.RemoteMessaging, 44
- sParam
 - UtilsNS.baseMMscan, 16
- sTo
 - UtilsNS.baseMMscan, 16
- stopwatch
 - UtilsNS.RemoteMessaging, 43
- stringIterator
 - UtilsNS.DictFileReader, 21
- stw
 - UtilsNS.AutoFileLogger, 13
 - UtilsNS.FileLogger, 25
- subheaders
 - UtilsNS.DictFileReader, 22
 - UtilsNS.FileLogger, 25
- synchroClock
 - UtilsNS.RemoteMessaging, 42
- TestInit
 - UtilsNS.MMscan, 35
- ToDictionary
 - UtilsNS.MMscan, 35
- UtilsNS, 9
 - AutoFileLogger, 11
 - AutoSaveFileName, 13
 - bufferCharSize, 14
 - bufferLimit, 13
 - bufferSize, 14
 - defaultExt, 13
 - DropLastChar, 12
 - Enabled, 14

- Flush, [12](#)
- header, [13](#)
- log, [12](#)
- missingData, [14](#)
- prefix, [14](#)
- stw, [13](#)
- writing, [14](#)
- UtilsNS.baseMMscan, [15](#)
 - getAsString, [15](#)
 - sBy, [16](#)
 - sFrom, [16](#)
 - sParam, [16](#)
 - sTo, [16](#)
 - Value, [16](#)
- UtilsNS.DictFileLogger, [17](#)
 - DictFileLogger, [17](#)
 - dictLog, [18](#)
 - setMMexecAsHeader, [19](#)
 - writeHeader, [19](#)
- UtilsNS.DictFileReader, [19](#)
 - counter, [21](#)
 - DictFileReader, [20](#)
 - doubleIterator, [20](#)
 - header, [21](#)
 - record, [22](#)
 - stringIterator, [21](#)
 - subheaders, [22](#)
- UtilsNS.FileLogger, [22](#)
 - CreateLogger, [23](#)
 - defaultExt, [25](#)
 - Enabled, [25](#)
 - FileLogger, [23](#)
 - header, [25](#)
 - log, [24](#)
 - LogFilename, [26](#)
 - missingData, [26](#)
 - prefix, [26](#)
 - reqFilename, [26](#)
 - stw, [25](#)
 - subheaders, [25](#)
 - writeHeader, [24](#)
 - writing, [26](#)
- UtilsNS.memLog, [27](#)
 - Enabled, [28](#)
 - log, [27](#)
 - memLog, [27](#)
- UtilsNS.MMexec, [28](#)
 - Abort, [29](#)
 - Assign, [30](#)
 - Clear, [30](#)
 - Clone, [30](#)
 - cmd, [31](#)
 - id, [31](#)
 - MMexec, [29](#)
 - mmexec, [31](#)
 - prms, [31](#)
 - sender, [31](#)
- UtilsNS.MMscan, [32](#)
 - Assign, [33](#)
 - AsString, [36](#)
 - Check, [33](#)
 - Clone, [33](#)
 - FromDictionary, [34](#)
 - groupID, [36](#)
 - isFirstValue, [34](#)
 - isLastValue, [34](#)
 - MMscan, [33](#)
 - Next, [34](#)
 - NextInChain, [35](#)
 - TestInit, [35](#)
 - ToDictionary, [35](#)
- UtilsNS.RemoteMessaging, [37](#)
 - ActiveComm, [39](#)
 - ActiveCommHandler, [39](#)
 - AsyncSent, [39](#)
 - AsyncSentHandler, [39](#)
 - autoCheckPeriod, [43](#)
 - CheckConnection, [40](#)
 - Connected, [43](#)
 - DoEvents, [40](#)
 - dTimer, [42](#)
 - elapsedTime, [40](#)
 - Enabled, [42](#)
 - ExitFrame, [40](#)
 - keyID, [43](#)
 - lastRcvMsg, [43](#)
 - lastSndMsg, [43](#)
 - Log, [42](#)
 - OnActiveComm, [44](#)
 - OnAsyncSent, [44](#)
 - OnReceive, [44](#)
 - partner, [44](#)
 - partnerPresent, [44](#)
 - Receive, [41](#)
 - ReceiveHandler, [41](#)
 - RemoteMessaging, [38](#)
 - sendCommand, [41](#)
 - silentPartner, [44](#)
 - stopwatch, [43](#)
 - synchroClock, [42](#)
- UtilsNS.WaitCursor, [45](#)
 - Dispose, [45](#)
 - WaitCursor, [45](#)
- Value
 - UtilsNS.baseMMscan, [16](#)
- WaitCursor
 - UtilsNS.WaitCursor, [45](#)
- writeHeader
 - UtilsNS.DictFileLogger, [19](#)
 - UtilsNS.FileLogger, [24](#)
- writing
 - UtilsNS.AutoFileLogger, [14](#)
 - UtilsNS.FileLogger, [26](#)