

# BMA Server Framework

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	BMAAccount Class Reference . . . . .	5
3.2	BMAAuthenticate Class Reference . . . . .	6
3.3	BMACommand Class Reference . . . . .	7
3.4	BMAConsoleServer Class Reference . . . . .	8
3.5	BMAConsoleSession Class Reference . . . . .	10
3.5.1	Detailed Description . . . . .	11
3.5.2	Member Function Documentation . . . . .	11
3.5.2.1	output() . . . . .	12
3.6	BMAEPoll Class Reference . . . . .	12
3.6.1	Detailed Description . . . . .	14
3.6.2	Member Function Documentation . . . . .	14
3.6.2.1	registerSocket() . . . . .	14
3.6.2.2	unregisterSocket() . . . . .	14
3.7	BMAFile Class Reference . . . . .	15
3.7.1	Detailed Description . . . . .	15
3.8	BMAHeader Class Reference . . . . .	16
3.9	BMAHTTPRequestHandler Class Reference . . . . .	17

3.10	BMAHTTPServer Class Reference . . . . .	18
3.11	BMAHTTPSession Class Reference . . . . .	20
3.11.1	Member Function Documentation . . . . .	21
3.11.1.1	onDataReceived() . . . . .	22
3.12	BMALog Class Reference . . . . .	22
3.13	BMAMP3File Class Reference . . . . .	22
3.14	BMAMP3StreamContentProvider Class Reference . . . . .	23
3.15	BMAMP3StreamFrame Class Reference . . . . .	24
3.16	BMAObject Class Reference . . . . .	26
3.17	BMAParseHeader Class Reference . . . . .	26
3.18	BMAProperty< T > Class Template Reference . . . . .	26
3.19	BMASession Class Reference . . . . .	27
3.19.1	Detailed Description . . . . .	28
3.19.2	Member Function Documentation . . . . .	28
3.19.2.1	onConnected() . . . . .	28
3.19.2.2	onDataReceived() . . . . .	29
3.19.2.3	output() . . . . .	29
3.20	BMASIPINVITE Class Reference . . . . .	29
3.21	BMASIPREGISTER Class Reference . . . . .	30
3.22	BMASIPRequestHandler Class Reference . . . . .	31
3.23	BMASIPServer Class Reference . . . . .	33
3.24	BMASIPSession Class Reference . . . . .	35
3.24.1	Member Function Documentation . . . . .	36
3.24.1.1	onDataReceived() . . . . .	37
3.25	BMASocket Class Reference . . . . .	37
3.25.1	Constructor & Destructor Documentation . . . . .	39
3.25.1.1	~BMASocket() . . . . .	39
3.25.2	Member Function Documentation . . . . .	39
3.25.2.1	eventReceived() . . . . .	39
3.25.2.2	onConnected() . . . . .	39

3.25.2.3	onDataReceived()	40
3.25.2.4	onRegistered()	40
3.26	BMAStreamContentProvider Class Reference	40
3.27	BMAStreamFrame Class Reference	41
3.28	BMAStreamServer Class Reference	42
3.28.1	Detailed Description	43
3.29	BMAStreamSession Class Reference	44
3.29.1	Member Function Documentation	45
3.29.1.1	onDataReceived()	46
3.30	BMATCPServerSocket Class Reference	46
3.30.1	Detailed Description	47
3.30.2	Member Function Documentation	48
3.30.2.1	accept()	48
3.30.2.2	onDataReceived()	48
3.31	BMATCPSocket Class Reference	48
3.31.1	Detailed Description	49
3.31.2	Member Function Documentation	50
3.31.2.1	onConnected()	50
3.31.2.2	output()	50
3.32	BMAThread Class Reference	50
3.33	BMATimer Class Reference	51
3.33.1	Detailed Description	52
3.33.2	Member Function Documentation	52
3.33.2.1	onDataReceived()	52
3.34	BMAUDPServerSocket Class Reference	53
3.34.1	Detailed Description	54
3.34.2	Member Function Documentation	54
3.34.2.1	onDataReceived()	54
3.35	BMAUDPSocket Class Reference	55



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

BMAFile . . . . .	15
BMAMP3File . . . . .	22
BMALog . . . . .	22
BMAObject . . . . .	26
BMAAccount . . . . .	5
BMAAuthenticate . . . . .	6
BMACommand . . . . .	7
BMAEPoll . . . . .	12
BMATCPServerSocket . . . . .	46
BMAConsoleServer . . . . .	8
BMAHTTPServer . . . . .	18
BMASIPServer . . . . .	33
BMAStreamServer . . . . .	42
BMAUDPServerSocket . . . . .	53
BMAHeader . . . . .	16
BMAHTTPRequestHandler . . . . .	17
BMASIPRequestHandler . . . . .	31
BMASIPINVITE . . . . .	29
BMASIPREGISTER . . . . .	30
BMASocket . . . . .	37
BMATCPSocket . . . . .	48
BMASession . . . . .	27
BMAConsoleSession . . . . .	10
BMAHTTPSession . . . . .	20
BMASIPSession . . . . .	35
BMAStreamSession . . . . .	44
BMATCPServerSocket . . . . .	46
BMATimer . . . . .	51
BMAStreamServer . . . . .	42
BMAUDPSocket . . . . .	55
BMAUDPServerSocket . . . . .	53
BMAParseHeader . . . . .	26
BMAProperty< T > . . . . .	26
BMAStreamContentProvider . . . . .	40

BMAMP3File . . . . .	22
BMAMP3StreamContentProvider . . . . .	23
BMASStreamFrame . . . . .	41
BMAMP3StreamFrame . . . . .	24
BMAThread . . . . .	50
streambuf	
BMASocket . . . . .	37



## Chapter 2

# Class Index

### 2.1 Class List

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

BMAAccount	5
BMAAuthenticate	6
BMACommand	7
BMAConsoleServer	8
BMAConsoleSession	10
BMAEPoll	12
BMAFile	15
BMAHeader	16
BMAHTTPRequestHandler	17
BMAHTTPServer	18
BMAHTTPSession	20
BMALog	22
BMAMP3File	22
BMAMP3StreamContentProvider	23
BMAMP3StreamFrame	24
BMAObject	26
BMAParseHeader	26
BMAProperty< T >	26
BMASession	27
BMASIPINVITE	29
BMASIPREGISTER	30
BMASIPRequestHandler	31
BMASIPServer	33
BMASIPSession	35
BMASocket	37
BMAStreamContentProvider	40
BMAStreamFrame	41
BMAStreamServer	42
BMAStreamSession	44
BMATCPServerSocket	46
BMATCPSocket	48
BMAThread	50
BMATimer	51
BMAUDPServerSocket	53
BMAUDPSocket	55

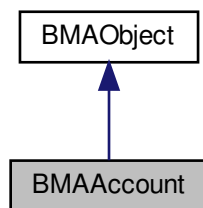


## Chapter 3

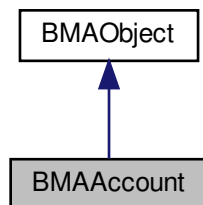
# Class Documentation

### 3.1 BMAAccount Class Reference

Inheritance diagram for BMAAccount:



Collaboration diagram for BMAAccount:



### Public Member Functions

- string **getName** ()
- void **setName** (string name)
- string **getAlias** ()
- void **setAlias** (string name)

### Public Attributes

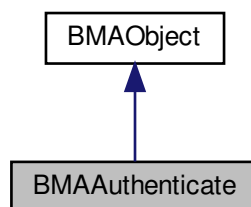
- vector< string > **contacts**

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

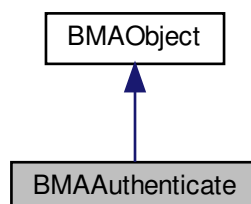
- /home/barant/Documents/Development/BMASockets/BMAAccount.h

## 3.2 BMAAuthenticate Class Reference

Inheritance diagram for BMAAuthenticate:



Collaboration diagram for BMAAuthenticate:



### Public Member Functions

- **BMAAuthenticate** ([BMASession](#) \*session)
- void **onStart** ()
- void **onDataReceived** (char \*data, int length)
- void **onEnd** ()

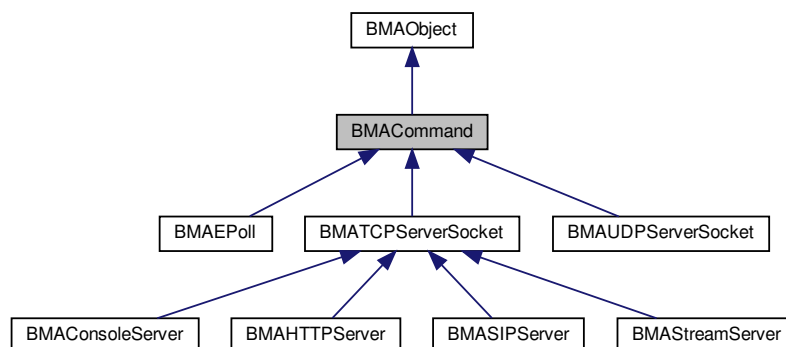
### Additional Inherited Members

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

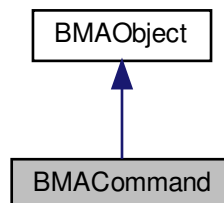
- /home/barant/Documents/Development/BMASockets/BMAAuthenticate.h
- /home/barant/Documents/Development/BMASockets/BMAAuthenticate.cpp

## 3.3 BMACommand Class Reference

Inheritance diagram for BMACommand:



Collaboration diagram for BMACommand:



## Public Member Functions

- **BMACommand** (std::string commandName)
- virtual int **processCommand** (BMASession \*session)

## Public Attributes

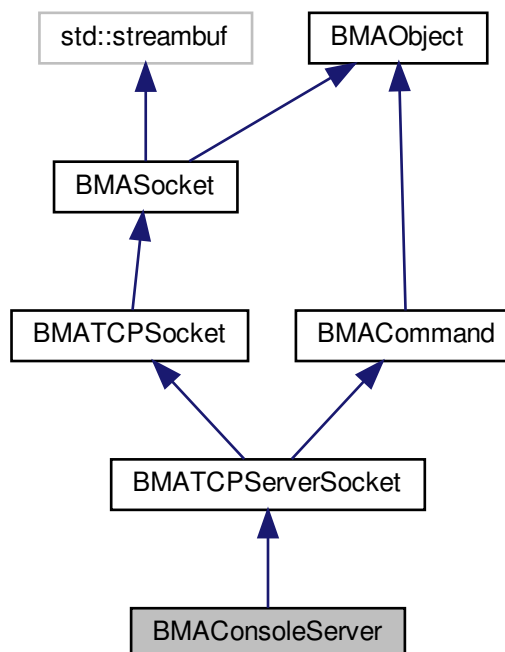
- std::string **commandName**

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

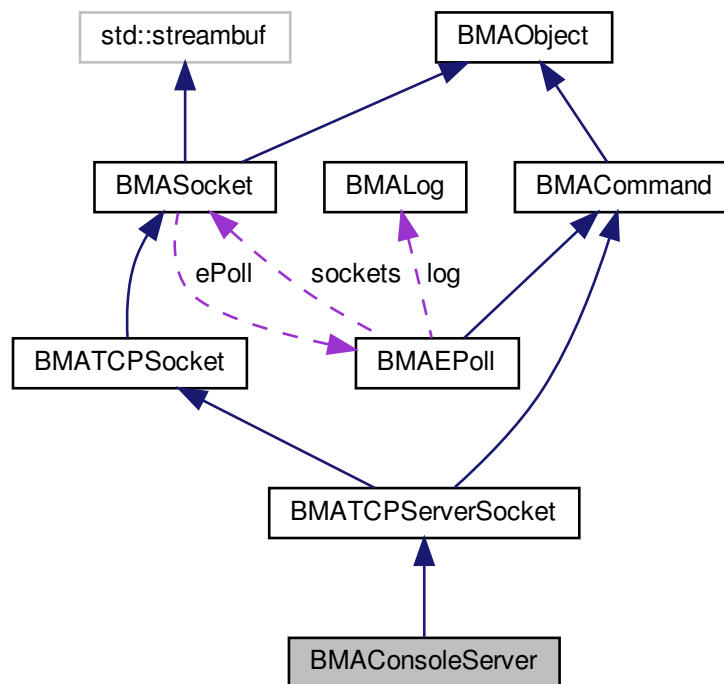
- /home/barant/Documents/Development/BMASockets/BMACommand.h
- /home/barant/Documents/Development/BMASockets/BMACommand.cpp

## 3.4 BMAConsoleServer Class Reference

Inheritance diagram for BMAConsoleServer:



Collaboration diagram for BMAConsoleServer:



### Public Member Functions

- **BMAConsoleServer** (**BMAEPoll** &ePoll, std::string url, short int port)
- **BMASession** \* **getSocketAccept** ()
- void **registerCommand** (**BMACommand** &command)
- int **processCommand** (**BMASession** \*session) override  
*Output the consoles array to the console.*

### Public Attributes

- std::vector< **BMACommand** \* > **commands**

### Additional Inherited Members

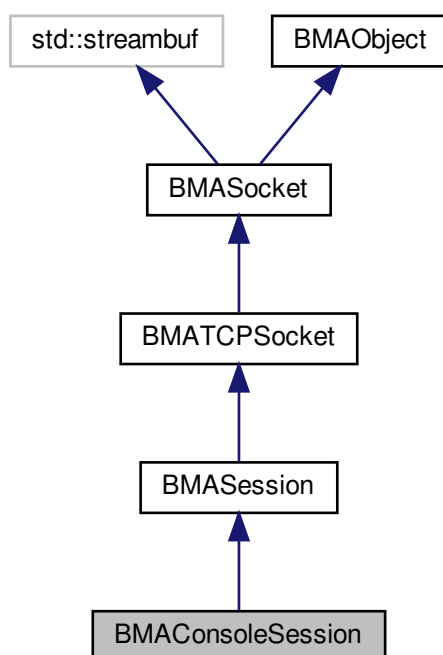
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMAConsoleServer.h
- /home/barant/Documents/Development/BMASockets/BMAConsoleServer.cpp

### 3.5 BMAConsoleSession Class Reference

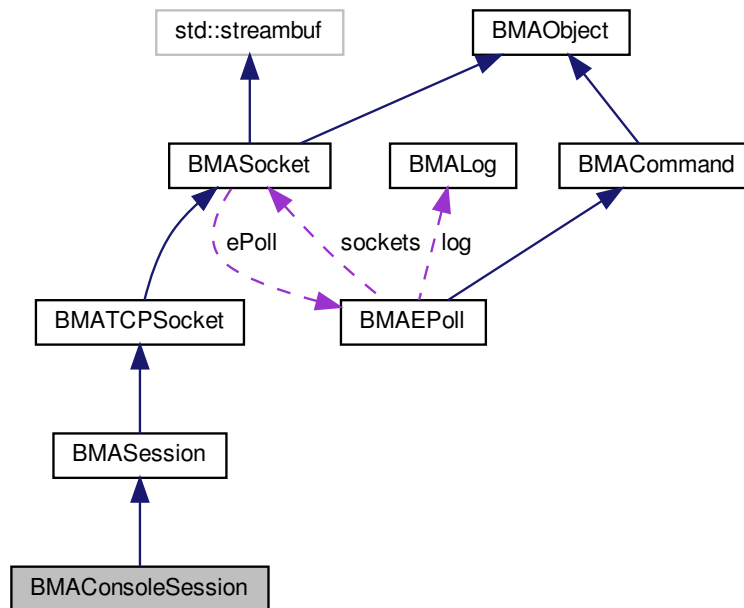
```
#include <BMAConsoleSession.h>
```

Inheritance diagram for BMAConsoleSession:





Collaboration diagram for BMAConsoleSession:



### Public Member Functions

- **BMAConsoleSession** ([BMAEPoll](#) &ePoll, [BMAConsoleServer](#) &server)
- virtual void [output](#) (std::stringstream &out)

### Protected Member Functions

- void **protocol** (char \*data, int length) override

### Additional Inherited Members

#### 3.5.1 Detailed Description

##### [BMAConsoleSession](#)

Extends the session parameters for this [BMATCPSocket](#) derived object. Extend the protocol() method in order to define the behavior and protocol interaction for this socket which is a console session.

#### 3.5.2 Member Function Documentation

### 3.5.2.1 output()

```
void BMAConsoleSession::output (
    std::stringstream & out ) [virtual]
```

The output method is called by a socket session ([BMASession](#)) and will output the detail information for the client socket. When extending [BMATCPSocket](#) or [BMASession](#) you can override the method to add attributes to the list.

Reimplemented from [BMASession](#).

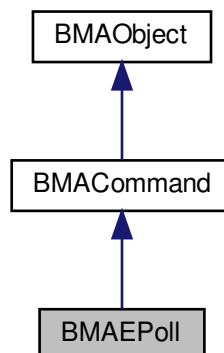
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMAConsoleSession.h
- /home/barant/Documents/Development/BMASockets/BMAConsoleSession.cpp

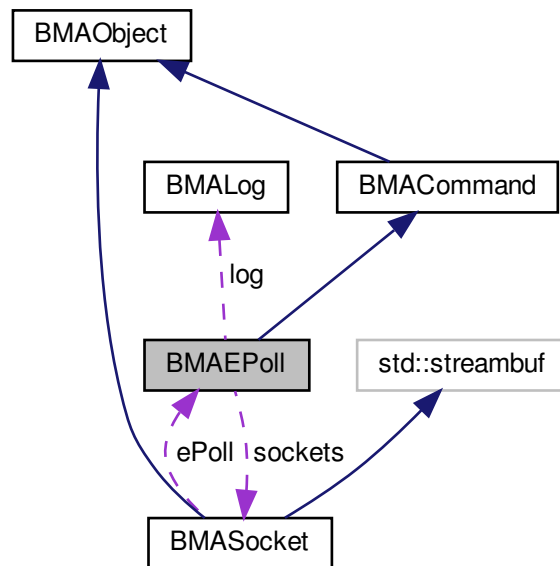
## 3.6 BMAEPoll Class Reference

```
#include <BMAEPoll.h>
```

Inheritance diagram for BMAEPoll:



Collaboration diagram for BMAEPoll:



## Public Member Functions

- `bool start (int numberOfThreads, int maxSockets)`  
Start the *BMAEPoll* processing.
- `bool stop ()`  
Stop and shut down the *BMAEPoll* processing.
- `bool isStopping ()`  
Returns a true if the stop command has been requested.
- `bool registerSocket (BMASocket *socket)`  
Register a *BMASocket* for monitoring by *BMAEPoll*.
- `bool unregisterSocket (BMASocket *socket)`  
Unregister a *BMASocket* from monitoring by *BMAEPoll*.
- `int getDescriptor ()`  
Return the descriptor for the ePoll socket.
- `int processCommand (BMASession *session) override`  
Output the threads array to the console.

## Public Attributes

- `int maxSockets`  
The maximum number of socket allowed.
- `BMASocket * sockets [1000]`  
List of sockets that have registered for monitoring.
- `BMALog log`

### 3.6.1 Detailed Description

#### BMAEPoll

Manage socket events from the epoll system call.

Use this object to establish a socket server using the epoll network structure of Linux.

Use this object to establish the basis of working with multiple sockets of all sorts using the epoll capabilities of the Linux platform. Socket objects can register with [BMAEPoll](#) which will establish a communication mechanism with that socket.

The maximum number of sockets to communicate with is specified on the start method.

Threads are used to establish a read queue for epoll. The desired number of threads (or queues) is established by a parameter on the start method.

### 3.6.2 Member Function Documentation

#### 3.6.2.1 registerSocket()

```
bool BMAEPoll::registerSocket (
    BMASocket * socket )
```

Register a [BMASocket](#) for monitoring by [BMAEPoll](#).

##### Parameters

<i>socket</i>	The <a href="#">BMASocket</a> to register.
---------------	--

#### 3.6.2.2 unregisterSocket()

```
bool BMAEPoll::unregisterSocket (
    BMASocket * socket )
```

Unregister a [BMASocket](#) from monitoring by [BMAEPoll](#).

##### Parameters

<i>socket</i>	The <a href="#">BMASocket</a> to unregister.
---------------	--

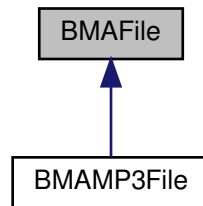
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMAEPoll.h
- /home/barant/Documents/Development/BMASockets/BMAEPoll.cpp

## 3.7 BMAFile Class Reference

```
#include <BMAFile.h>
```

Inheritance diagram for BMAFile:



### Public Member Functions

- **BMAFile** (std::string fileName, int mode=O\_RDONLY, int authority=0664)
- void **setBufferSize** (size\_t size)
- void **read** ()

### Public Attributes

- char \* **buffer**
- size\_t **size**

### 3.7.1 Detailed Description

#### BMAFile

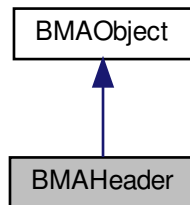
File abstraction class for accessing local file system files.

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

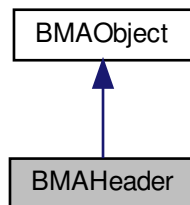
- /home/barant/Documents/Development/BMASockets/BMAFile.h
- /home/barant/Documents/Development/BMASockets/BMAFile.cpp

### 3.8 BMAHeader Class Reference

Inheritance diagram for BMAHeader:



Collaboration diagram for BMAHeader:



#### Public Member Functions

- **BMAHeader** (std::string data)
- std::string **requestMethod** ()
- std::string **requestPath** ()

#### Public Attributes

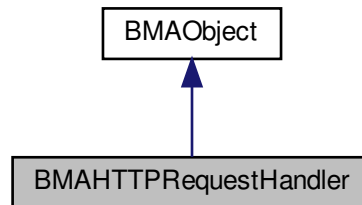
- std::string **data**

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

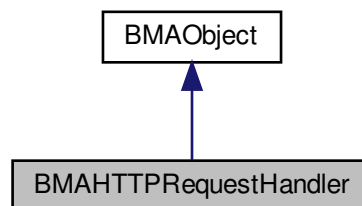
- /home/barant/Documents/Development/BMASockets/BMAHeader.h
- /home/barant/Documents/Development/BMASockets/BMAHeader.cpp

## 3.9 BMAHTTPRequestHandler Class Reference

Inheritance diagram for BMAHTTPRequestHandler:



Collaboration diagram for BMAHTTPRequestHandler:



### Public Member Functions

- **BMAHTTPRequestHandler** ([BMAHTTPServer](#) &server, std::string path)
- virtual int **response** (std::stringstream &sink)

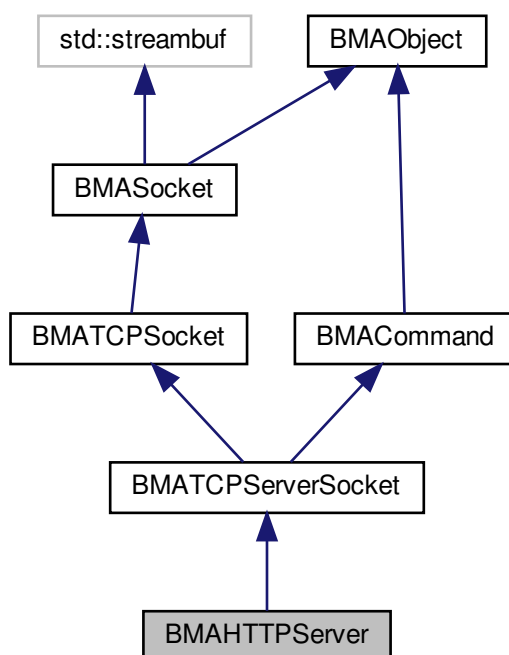
### Additional Inherited Members

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

- /home/barant/Documents/Development/BMASockets/BMAHTTPRequestHandler.h
- /home/barant/Documents/Development/BMASockets/BMAHTTPRequestHandler.cpp

### 3.10 BMAHTTPServer Class Reference

Inheritance diagram for BMAHTTPServer:

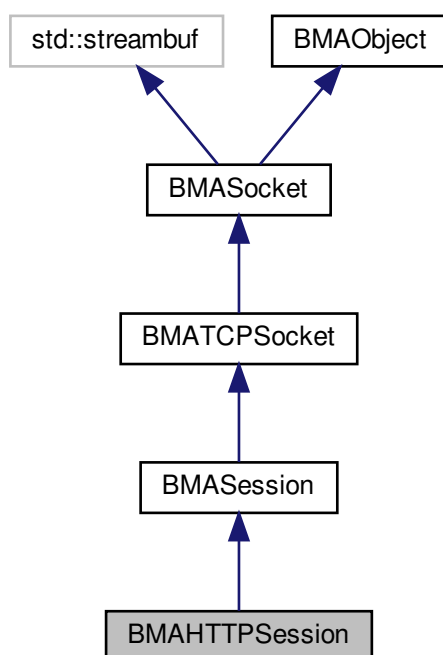






### 3.11 BMAHTTPSession Class Reference

Inheritance diagram for BMAHTTPSession:





### 3.11.1.1 onDataReceived()

```
void BMAHTTPSession::onDataReceived (
    char * data,
    int length ) [override], [protected], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

Reimplemented from [BMASocket](#).

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

- /home/barant/Documents/Development/BMASockets/BMAHTTPSession.h
- /home/barant/Documents/Development/BMASockets/BMAHTTPSession.cpp

## 3.12 BMALog Class Reference

### Public Member Functions

- std::ostringstream & **write** (int type)
- void **registerConsole** (BMAConsole &console)

### Protected Attributes

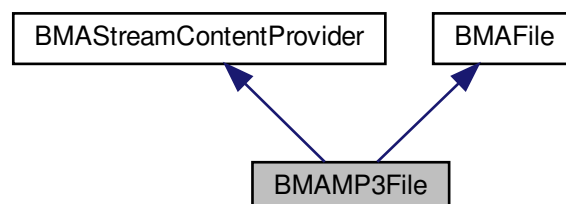
- std::ostringstream **os**

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

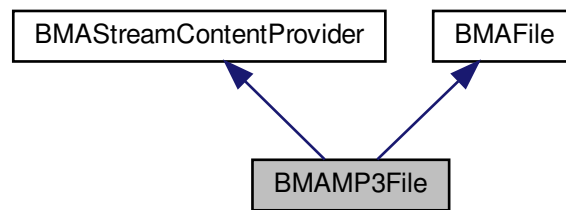
- /home/barant/Documents/Development/BMASockets/BMALog.h
- /home/barant/Documents/Development/BMASockets/BMALog.cpp

## 3.13 BMAMP3File Class Reference

Inheritance diagram for BMAMP3File:



Collaboration diagram for BMAMP3File:



### Public Member Functions

- **BMAMP3File** ([BMAServer](#) &server, std::string fileName)

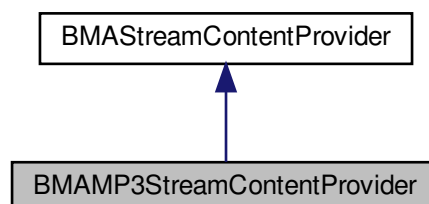
### Additional Inherited Members

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

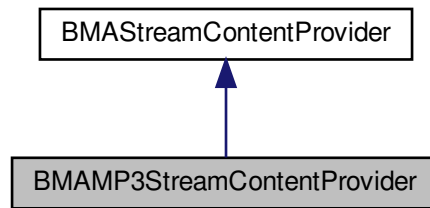
- /home/barant/Documents/Development/BMASockets/BMAMP3File.h
- /home/barant/Documents/Development/BMASockets/BMAMP3File.cpp

## 3.14 BMAMP3StreamContentProvider Class Reference

Inheritance diagram for BMAMP3StreamContentProvider:



Collaboration diagram for BMAMP3StreamContentProvider:



### Public Member Functions

- **BMAMP3StreamContentProvider** ([BMASStreamServer](#) &server)
- [BMASStreamFrame](#) \* **getStreamFrame** ()

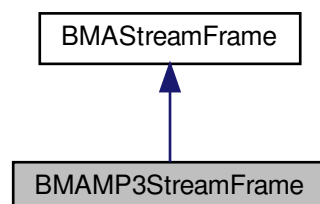
### Additional Inherited Members

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

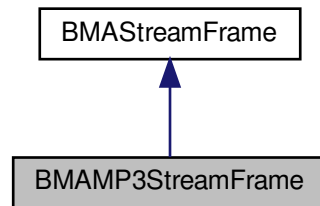
- `/home/barant/Documents/Development/BMASockets/BMAMP3StreamContentProvider.h`

## 3.15 BMAMP3StreamFrame Class Reference

Inheritance diagram for BMAMP3StreamFrame:



Collaboration diagram for BMAMP3StreamFrame:



### Public Member Functions

- **BMAMP3StreamFrame** (char \*stream)
- double **getDuration** ()
- int **getVersion** ()
- int **getLayer** ()
- int **getBitRate** ()
- int **getSampleRate** ()
- int **getPaddingSize** ()
- int **getFrameSampleSize** ()
- int **getFrameSize** ()

### Protected Attributes

- int **bit\_rates** [16] = { -1, 32, 40, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256, 320, -1 }
- int **sample\_rates** [4] = { 44100, 48000, 32000, -1 }

### Additional Inherited Members

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

- /home/barant/Documents/Development/BMASockets/BMAMP3StreamFrame.h
- /home/barant/Documents/Development/BMASockets/BMAMP3StreamFrame.cpp





### Protected Attributes

- **T value**

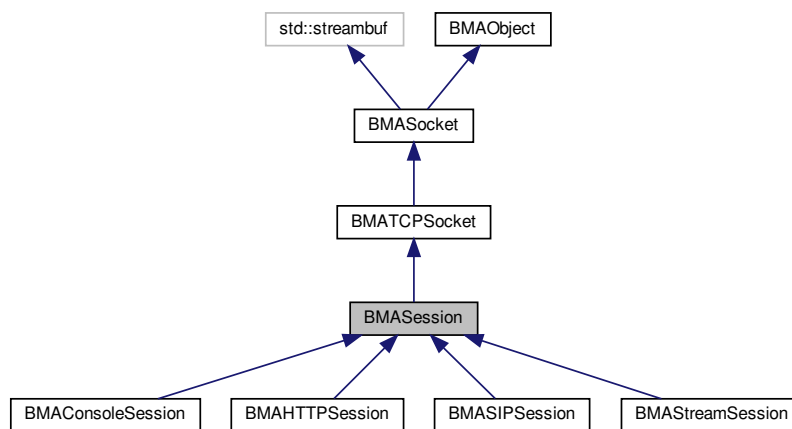
The documentation for this class was generated from the following file:

- /home/barant/Documents/Development/BMASockets/BMAProperty.h

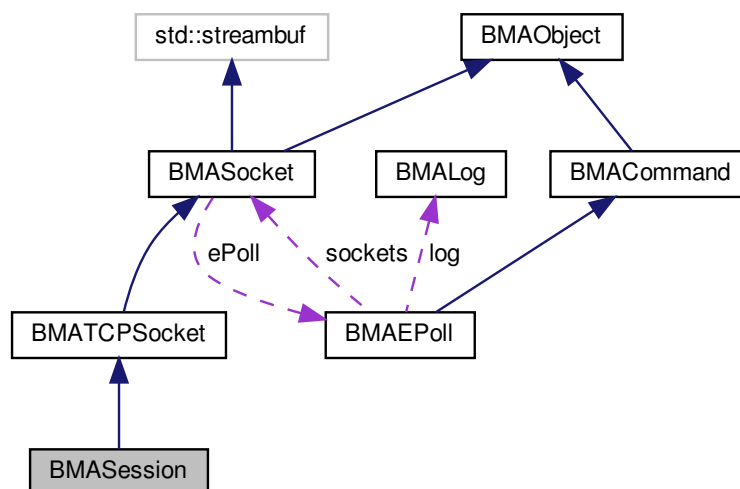
## 3.19 BMASession Class Reference

```
#include <BMASession.h>
```

Inheritance diagram for BMASession:



Collaboration diagram for BMASession:



## Public Member Functions

- **BMASession** ([BMAEPoll](#) &ePoll)
- virtual void [output](#) (std::stringstream &out)

## Public Attributes

- std::string **command**
- std::string **title**

## Protected Member Functions

- void [onConnected](#) () override  
*Called when socket is open and ready to communicate.*
- void [onDataReceived](#) (char \*data, int length) override  
*Called when data is received from the socket.*
- virtual void **protocol** (char \*data, int length)

## Additional Inherited Members

### 3.19.1 Detailed Description

#### [BMASession](#)

[BMASession](#) defines the nature of the interaction with the client and stores persistent data for a defined session. [BMASession](#) objects are not sockets but instead provide a communications control mechanism. Protocol conversations are provided through extensions from this object.

### 3.19.2 Member Function Documentation

#### 3.19.2.1 [onConnected\(\)](#)

```
void BMASession::onConnected ( ) [override], [protected], [virtual]
```

Called when socket is open and ready to communicate.

The onConnected method is called when the socket is ready to communicate. Writing to the socket can begin on this call to initiate a contact with the remote device.

Reimplemented from [BMASocket](#).

## 3.19.2.2 onDataReceived()

```
void BMASession::onDataReceived (
    char * data,
    int length ) [override], [protected], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

Reimplemented from [BMASocket](#).

Reimplemented in [BMASession](#), and [BMASIPSession](#).

## 3.19.2.3 output()

```
void BMASession::output (
    std::stringstream & out ) [virtual]
```

The output method is called by a socket session ([BMASession](#)) and will output the detail information for the client socket. When extending [BMATCPSocket](#) or [BMASession](#) you can override the method to add attributes to the list.

Reimplemented from [BMATCPSocket](#).

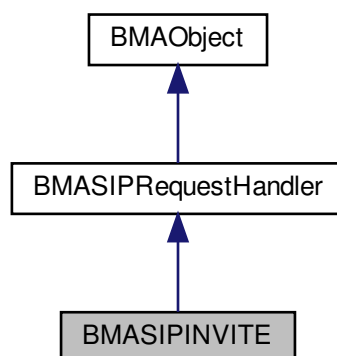
Reimplemented in [BMAConsoleSession](#).

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

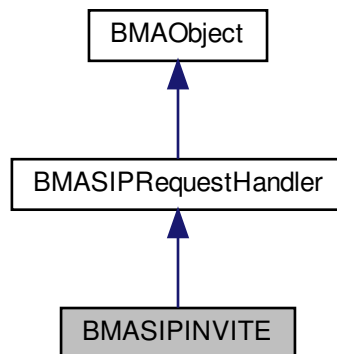
- /home/barant/Documents/Development/BMASockets/BMASession.h
- /home/barant/Documents/Development/BMASockets/BMASession.cpp

## 3.20 BMASIPINVITE Class Reference

Inheritance diagram for BMASIPINVITE:



Collaboration diagram for BMASIPINVITE:



#### Public Member Functions

- **BMASIPINVITE** ([BMASIPServer](#) &server, std::string path)
- int **response** (std::stringstream &sink) override

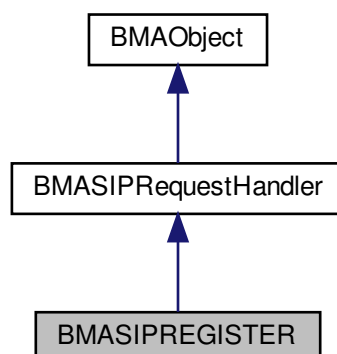
#### Additional Inherited Members

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

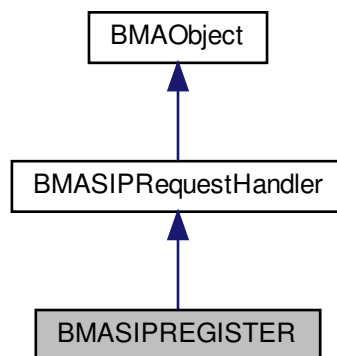
- /home/barant/Documents/Development/BMASockets/BMASIPINVITE.h

### 3.21 BMASIPREGISTER Class Reference

Inheritance diagram for BMASIPREGISTER:



Collaboration diagram for BMASIPREGISTER:



#### Public Member Functions

- **BMASIPREGISTER** ([BMASIPServer](#) &server, std::string path)
- int **response** (std::stringstream &sink) override

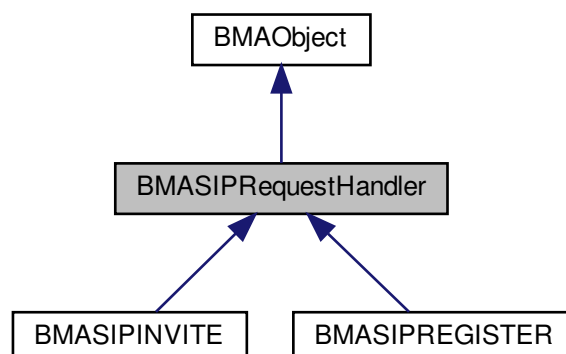
#### Additional Inherited Members

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

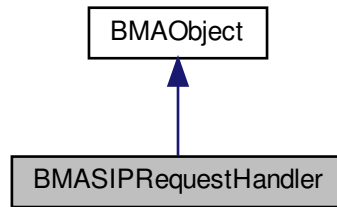
- /home/barant/Documents/Development/BMASockets/BMASIPREGISTER.h

## 3.22 BMASIPRequestHandler Class Reference

Inheritance diagram for BMASIPRequestHandler:



Collaboration diagram for BMASIPRequestHandler:



### Public Member Functions

- **BMASIPRequestHandler** ([BMASIPServer](#) &server, std::string path)
- virtual int **response** (std::stringstream &sink)

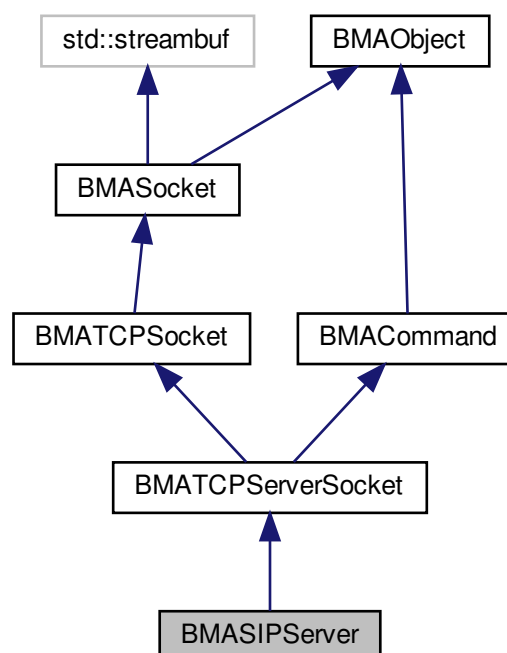
### Additional Inherited Members

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

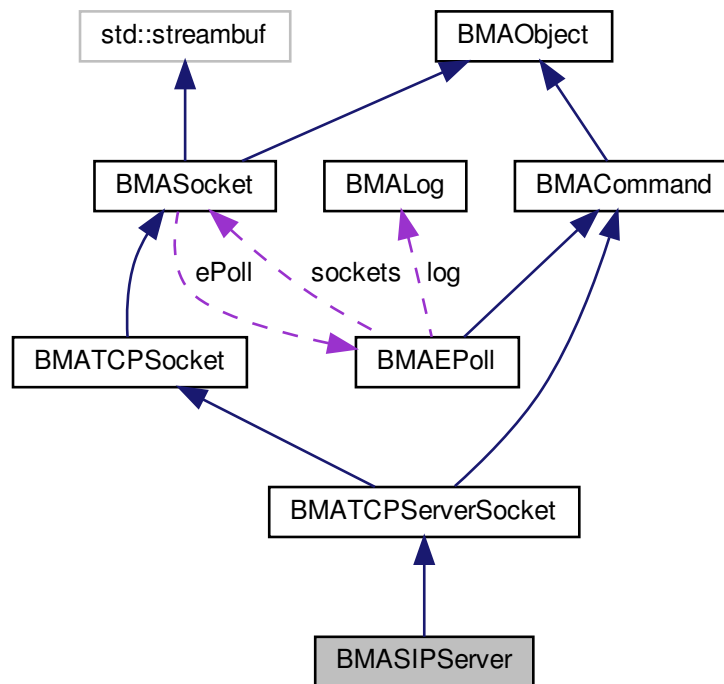
- /home/barant/Documents/Development/BMASockets/BMASIPRequestHandler.h
- /home/barant/Documents/Development/BMASockets/BMASIPRequestHandler.cpp

### 3.23 BMASIPServer Class Reference

Inheritance diagram for BMASIPServer:



Collaboration diagram for BMASIPServer:



## Public Member Functions

- **BMASIPServer** ([BMAEPoll](#) &ePoll, std::string url, short int port, std::string commandName)
- void **registerHandler** (std::string path, [BMASIPRequestHandler](#) &requestHandler)
- void **unregisterHandler** ([BMASIPRequestHandler](#) &requestHandler)
- [BMASIPRequestHandler](#) \* **getRequestHandler** (std::string request)

## Protected Member Functions

- [BMASession](#) \* **getSocketAccept** () override

## Additional Inherited Members

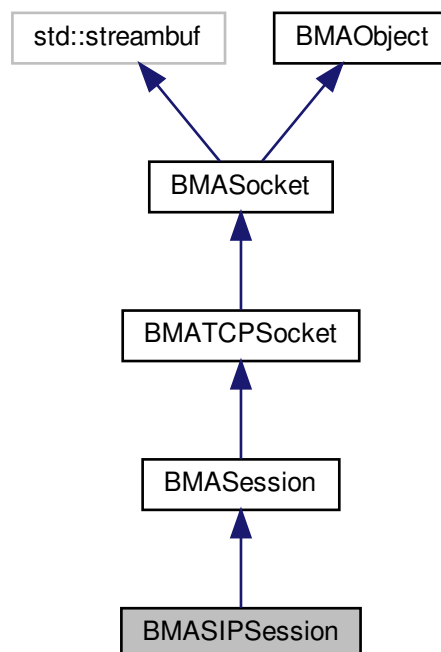
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMASIPServer.h
- /home/barant/Documents/Development/BMASockets/BMASIPServer.cpp

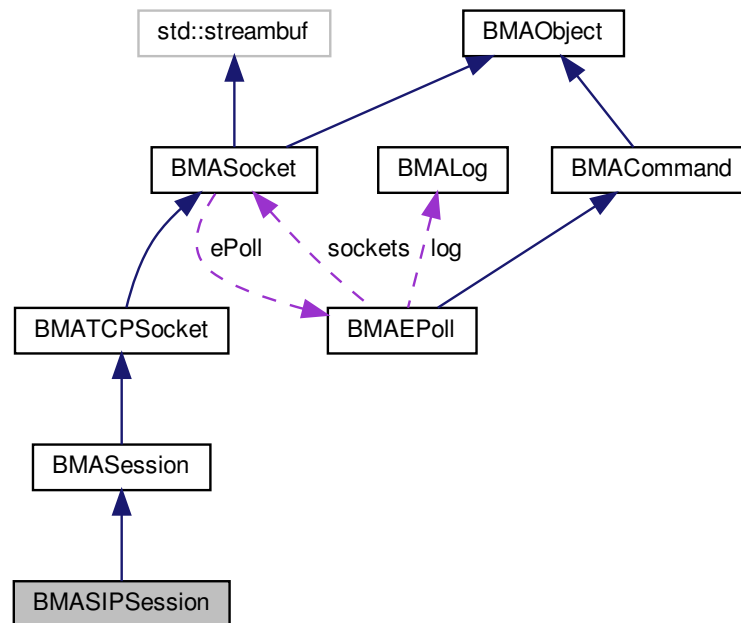


## 3.24 BMASIPSession Class Reference

Inheritance diagram for BMASIPSession:



Collaboration diagram for BMASIPSession:



## Public Member Functions

- **BMASIPSession** ([BMAEPoll](#) &ePoll, [BMASIPServer](#) &server)

## Protected Member Functions

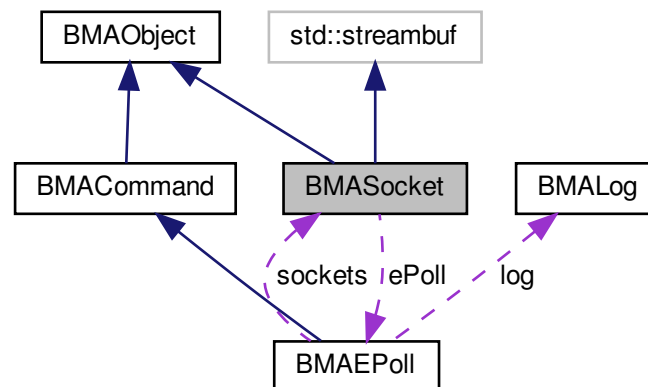
- void [onDataReceived](#) (char \*data, int length) override  
*Called when data is received from the socket.*

## Additional Inherited Members

### 3.24.1 Member Function Documentation



Collaboration diagram for BMASocket:



## Public Member Functions

- **BMASocket** ([BMAEPoll](#) &ePoll)
- [~BMASocket](#) ()
- void [setDescriptor](#) (int descriptor)  
*Set the descriptor for the socket.*
- int [getDescriptor](#) ()  
*Get the descriptor for the socket.*
- void [eventReceived](#) (struct epoll\_event event)  
*Parse epoll event and call specified callbacks.*
- void **write** (char \*buffer, int length)
- void **output** (std::stringstream &out)
- virtual void [onRegistered](#) ()  
*Called when the socket has finished registering with the epoll processing.*

## Public Attributes

- class {  
} **bufferSize**

## Protected Member Functions

- void **setBufferSize** (int length)
- virtual void [onConnected](#) ()  
*Called when socket is open and ready to communicate.*
- virtual void [onDataReceived](#) (char \*data, int length)  
*Called when data is received from the socket.*

## Protected Attributes

- [BMAEPoll](#) & [ePoll](#)
- bool **shutDown** = false

## 3.25.1 Constructor & Destructor Documentation

### 3.25.1.1 ~BMA Socket()

```
BMA Socket::~BMA Socket ( )
```

TODO: If the object is deleted then we need a super nice cleanup of the socket.

## 3.25.2 Member Function Documentation

### 3.25.2.1 eventReceived()

```
void BMA Socket::eventReceived (
    struct epoll_event event )
```

Parse epoll event and call specified callbacks.

The event received from epoll is sent through the eventReceived method which will parse the event and call the read and write callbacks on the socket.

This method is called by the [BMAEPoll](#) object and should not be called from any user extended classes unless an epoll event is being simulated.

### 3.25.2.2 onConnected()

```
void BMA Socket::onConnected ( ) [protected], [virtual]
```

Called when socket is open and ready to communicate.

The onConnected method is called when the socket is ready to communicate. Writing to the socket can begin on this call to initiate a contact with the remote device.

Reimplemented in [BMATCPSocket](#), and [BMASession](#).

### 3.25.2.3 onDataReceived()

```
void BMASocket::onDataReceived (
    char * data,
    int length ) [protected], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

Reimplemented in [BMATCPServerSocket](#), [BMAUDPServerSocket](#), [BMASession](#), [BMATimer](#), [BMASessionStreamSession](#), [BMAHTTPSession](#), and [BMASIPSession](#).

### 3.25.2.4 onRegistered()

```
void BMASocket::onRegistered ( ) [virtual]
```

Called when the socket has finished registering with the epoll processing.

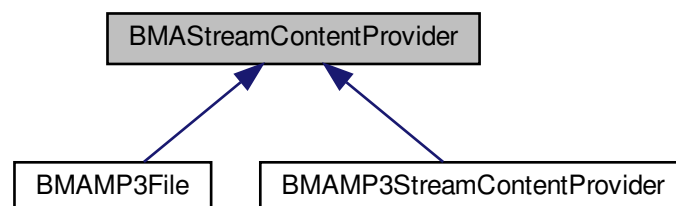
The onRegistered method is called whenever the socket is registered with ePoll and socket communication events can be started.

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

- /home/barant/Documents/Development/BMASockets/BMASocket.h
- /home/barant/Documents/Development/BMASockets/BMASocket.cpp

## 3.26 BMASessionStreamContentProvider Class Reference

Inheritance diagram for BMASessionStreamContentProvider:



### Public Member Functions

- **BMASessionStreamContentProvider** ([BMASessionStreamServer](#) &server)
- virtual [BMASessionStreamFrame](#) \* getNextStreamFrame ()
- int getFrameCount ()

### Public Attributes

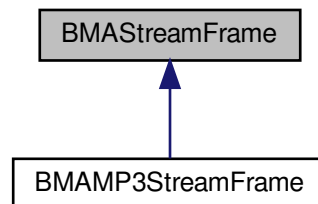
- bool **ready** = false
- std::vector< [BMASStreamFrame](#) \* > **frames**
- int **cursor**

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

- /home/barant/Documents/Development/BMASockets/BMASStreamContentProvider.h
- /home/barant/Documents/Development/BMASockets/BMASStreamContentProvider.cpp

## 3.27 BMASStreamFrame Class Reference

Inheritance diagram for BMASStreamFrame:



### Public Member Functions

- **BMASStreamFrame** (char \*streamData)
- virtual double **getDuration** ()=0
- virtual int **getFrameSize** ()=0

### Public Attributes

- char \* **streamData**
- bool **lastFrame**

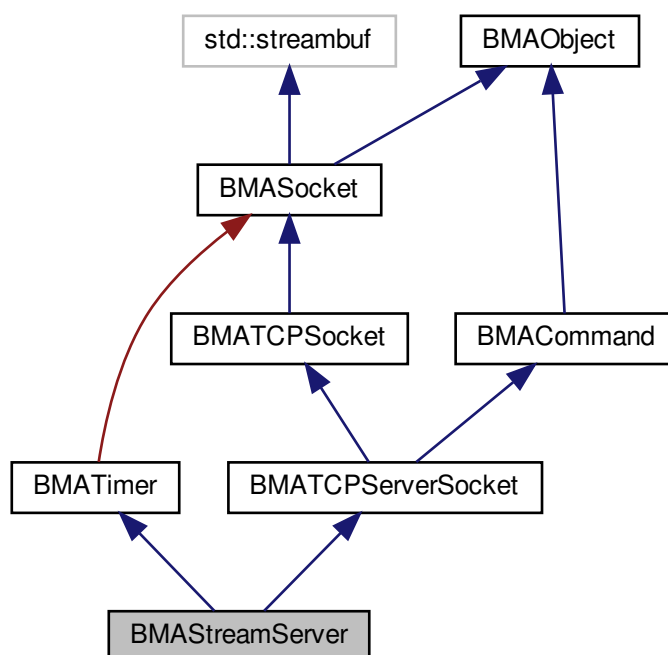
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMASStreamFrame.h
- /home/barant/Documents/Development/BMASockets/BMASStreamFrame.cpp

### 3.28 BMAServer Class Reference

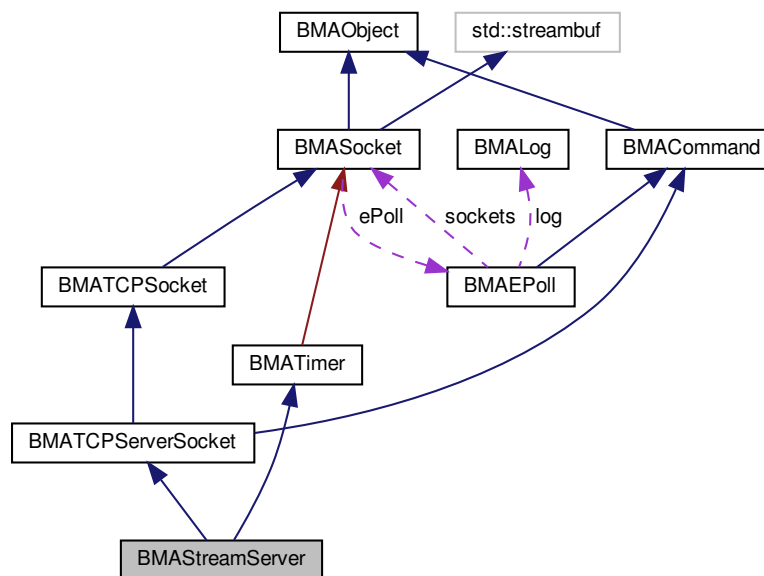
```
#include <BMAServer.h>
```

Inheritance diagram for BMAServer:





Collaboration diagram for BMAServer:



## Public Member Functions

- **BMAServer** ([BMAEPoll](#) &ePoll, std::string url, short int port, std::string commandName)
- void **startStreaming** ()
- void **setContentProvider** ([BMAServerContentProvider](#) &contentProvider)
- void **sendFrameToClients** ([BMAServerFrame](#) \*streamFrame)
- [BMASession](#) \* **getSocketAccept** () override
- void **onTimeout** () override

## Additional Inherited Members

### 3.28.1 Detailed Description

#### [BMAServer](#)

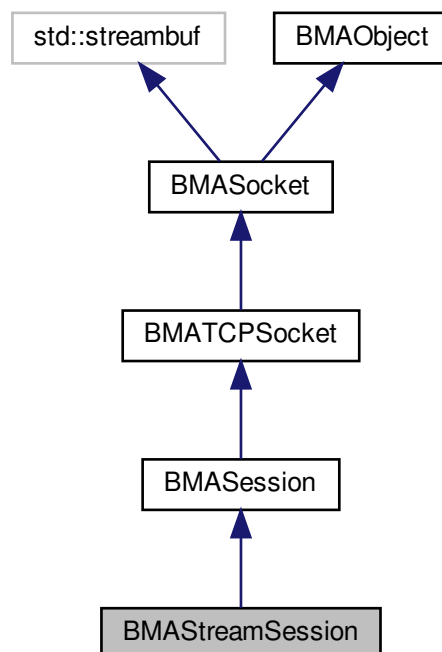
Extends the socket to create a frame based streaming media streamer.

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

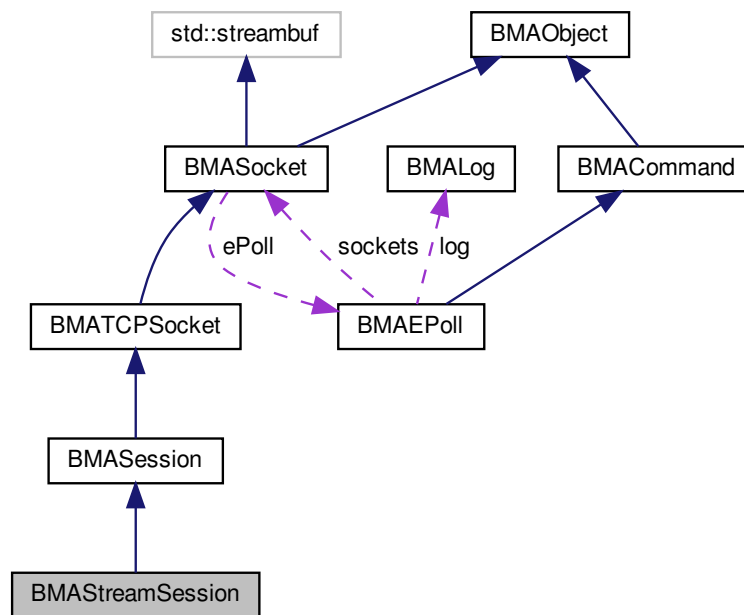
- /home/barant/Documents/Development/BMASockets/BMAServer.h
- /home/barant/Documents/Development/BMASockets/BMAServer.cpp

### 3.29 BMASession Class Reference

Inheritance diagram for BMASession:



Collaboration diagram for BMAStreamSession:



## Public Member Functions

- **BMAStreamSession** ([BMAEPoll](#) &ePoll)
- int **writeFrame** ([BMAStreamFrame](#) \*frame)

## Protected Member Functions

- void [onDataReceived](#) (char \*data, int length)  
*Called when data is received from the socket.*
- void **onStreamDataReceived** ([BMAStreamFrame](#) \*frame)

## Additional Inherited Members

### 3.29.1 Member Function Documentation

### 3.29.1.1 onDataReceived()

```
void BMASession::onDataReceived (
    char * data,
    int length ) [protected], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

Reimplemented from [BMASession](#).

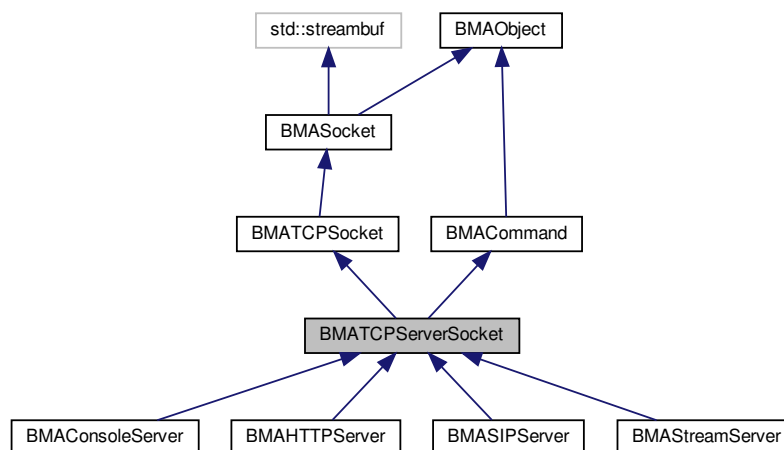
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMASession.h
- /home/barant/Documents/Development/BMASockets/BMASession.cpp

## 3.30 BMATCPServerSocket Class Reference

```
#include <BMATCPServerSocket.h>
```

Inheritance diagram for BMATCPServerSocket:





### 3.30.2 Member Function Documentation

#### 3.30.2.1 accept()

```
BMASession * BMATCPServerSocket::accept ( ) [virtual]
```

Allows for the overriding of the accept process from the newly contacted socket. Most likely the default handler will work fine since it will call `getSocketAccept` to retrieve the extended object to manage the new connection.

#### 3.30.2.2 onDataReceived()

```
void BMATCPServerSocket::onDataReceived (
    char * data,
    int length ) [override], [protected], [virtual]
```

Called when data is received from the socket.

The `onDataReceived` method is called when the socket has received an event from `epoll` and there is data ready to be read from the socket. The default handler will pull the data and put it into the `streambuf` for the socket. `EPOLLIN`

Reimplemented from [BMASocket](#).

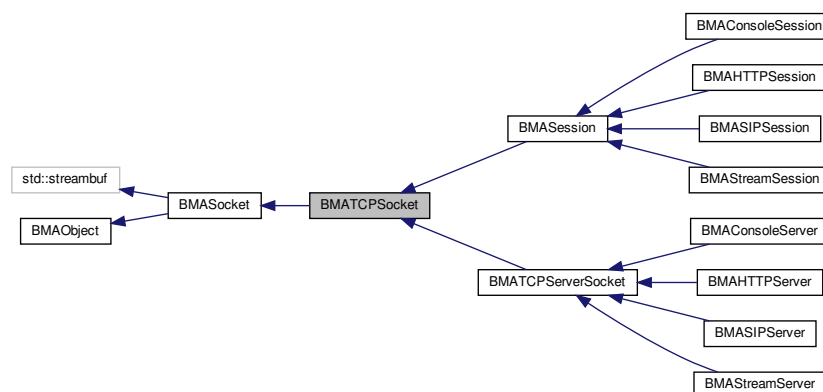
The documentation for this class was generated from the following files:

- `/home/barant/Documents/Development/BMASockets/BMATCPServerSocket.h`
- `/home/barant/Documents/Development/BMASockets/BMATCPServerSocket.cpp`

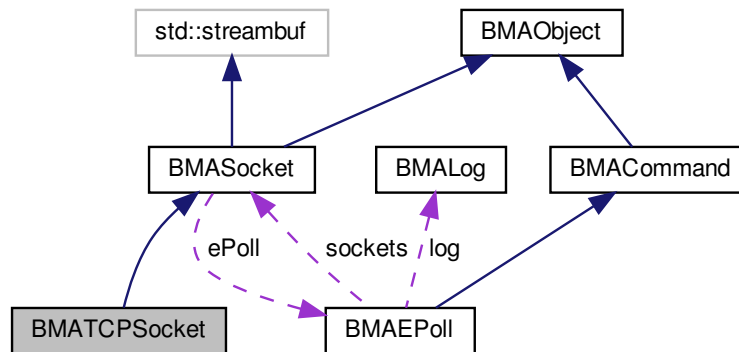
## 3.31 BMATCPSocket Class Reference

```
#include <BMATCPSocket.h>
```

Inheritance diagram for `BMATCPSocket`:



Collaboration diagram for BMATCPSocket:



## Public Member Functions

- **BMATCPSocket** (BMAEPoll &ePoll)
- std::string [getClientAddress](#) ()  
*Get the client network address as xxx.xxx.xxx.xxx.*
- std::string [getClientAddressAndPort](#) ()  
*Get the client network address and port as xxx.xxx.xxx.xxx:ppppp.*
- int [getClientPort](#) ()  
*Get the client network port number.*
- virtual void [output](#) (std::stringstream &out)
- void [onConnected](#) () override  
*Called when socket is open and ready to communicate.*

## Public Attributes

- struct sockaddr\_in **client\_addr**
- socklen\_t **client\_addr\_len**

## Additional Inherited Members

### 3.31.1 Detailed Description

#### BMATCPSocket

Provides a network TCP socket.

For accessing TCP network functions use this object. The connection oriented nature of TCP provides a single client persistent connection with data error correction and a durable synchronous data connection.

### 3.31.2 Member Function Documentation

#### 3.31.2.1 onConnected()

```
void BMATCPSocket::onConnected ( ) [override], [virtual]
```

Called when socket is open and ready to communicate.

The onConnected method is called when the socket is ready to communicate. Writing to the socket can begin on this call to initiate a contact with the remote device.

Reimplemented from [BMASocket](#).

#### 3.31.2.2 output()

```
void BMATCPSocket::output (
    std::stringstream & out ) [virtual]
```

The output method is called by a socket session ([BMASession](#)) and will output the detail information for the client socket. When extending [BMATCPSocket](#) or [BMASession](#) you can override the method to add attributes to the list.

Reimplemented in [BMASession](#), and [BMAConsoleSession](#).

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

- /home/barant/Documents/Development/BMASockets/BMATCPSocket.h
- /home/barant/Documents/Development/BMASockets/BMATCPSocket.cpp

## 3.32 BMAThread Class Reference

### Public Member Functions

- **BMAThread** ([BMAEPoll](#) &ePoll)
- void **start** ()
- void **join** ()
- std::string **getStatus** ()
- pid\_t **getThreadId** ()
- int **getCount** ()
- void **run** ()

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

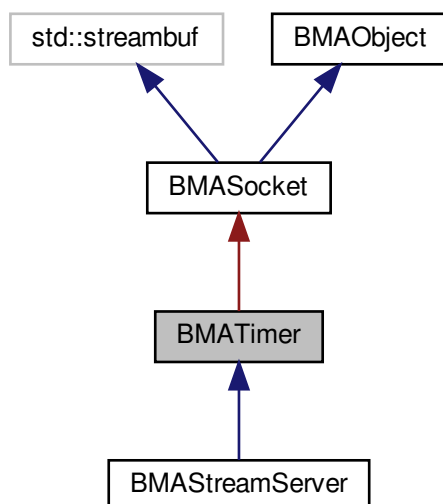
- /home/barant/Documents/Development/BMASockets/BMAThread.h
- /home/barant/Documents/Development/BMASockets/BMAThread.cpp



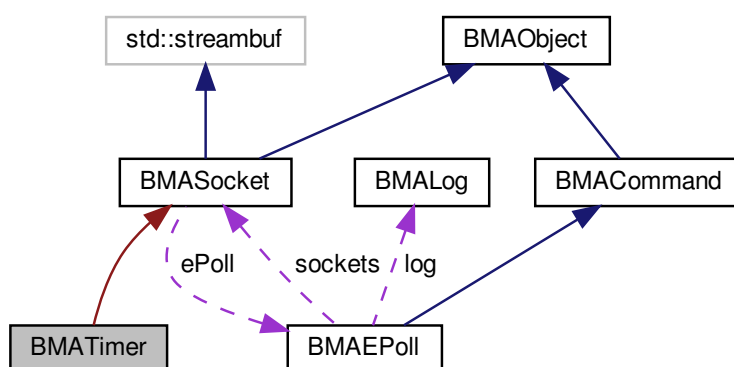
### 3.33 BMATimer Class Reference

```
#include <BMATimer.h>
```

Inheritance diagram for BMATimer:



Collaboration diagram for BMATimer:



#### Public Member Functions

- **BMATimer** ([BMAEPoll](#) &ePoll)

- **BMATimer** ([BMAEPoll](#) &ePoll, double delay)
- void **setTimer** (double delay)
- void **clearTimer** ()
- void **onDataReceived** (char \*data, int length) override

*Called when data is received from the socket.*

## Protected Member Functions

- virtual void **onTimeout** ()

### 3.33.1 Detailed Description

#### [BMATimer](#)

Set and trigger callback upon specified timeout.

The [BMATimer](#) is used to establish a timer using the timer socket interface.

### 3.33.2 Member Function Documentation

#### 3.33.2.1 [onDataReceived\(\)](#)

```
void BMATimer::onDataReceived (  
    char * data,  
    int length ) [override], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

Reimplemented from [BMASocket](#).

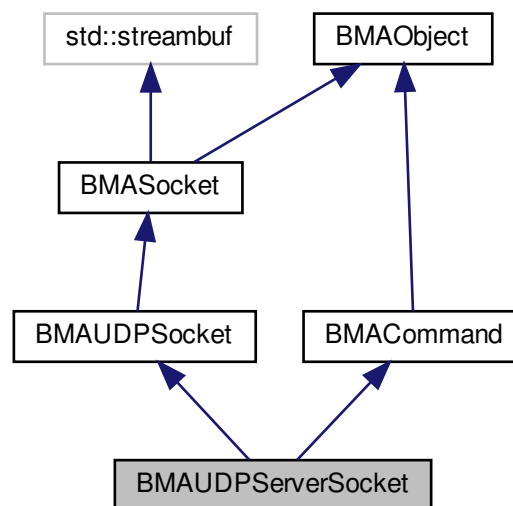
The documentation for this class was generated from the following files:

- /home/barant/Documents/Development/BMASockets/BMATimer.h
- /home/barant/Documents/Development/BMASockets/BMATimer.cpp

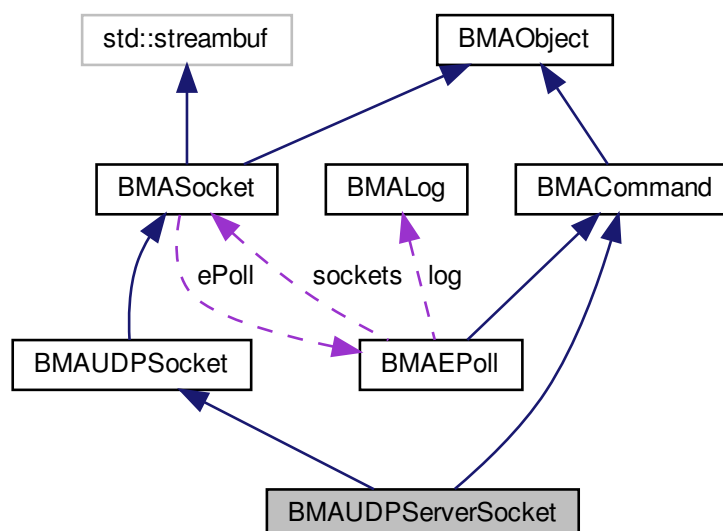
### 3.34 BMAUDPServerSocket Class Reference

```
#include <BMAUDPServerSocket.h>
```

Inheritance diagram for BMAUDPServerSocket:



Collaboration diagram for BMAUDPServerSocket:



## Public Member Functions

- **BMAUDPServerSocket** ([BMAEPoll](#) &ePoll, std::string url, short int port, std::string commandName)

## Protected Member Functions

- void [onDataReceived](#) (char \*data, int length) override  
*Called when data is received from the socket.*
- int **processCommand** ([BMASession](#) \*session)

## Protected Attributes

- std::vector< [BMASession](#) \* > **sessions**

## Additional Inherited Members

### 3.34.1 Detailed Description

#### [BMAUDPSocket](#)

Manage a socket connection as a UDP server type. Connections to the socket are processed through the session list functionality. A list of sessions is maintained in a vector object.

### 3.34.2 Member Function Documentation

#### 3.34.2.1 [onDataReceived\(\)](#)

```
void BMAUDPServerSocket::onDataReceived (
    char * data,
    int length ) [override], [protected], [virtual]
```

Called when data is received from the socket.

The onDataReceived method is called when the socket has received an event from epoll and there is data ready to be read from the socket. The default handler will pull the data and put it into the streambuf for the socket. EPOLLIN

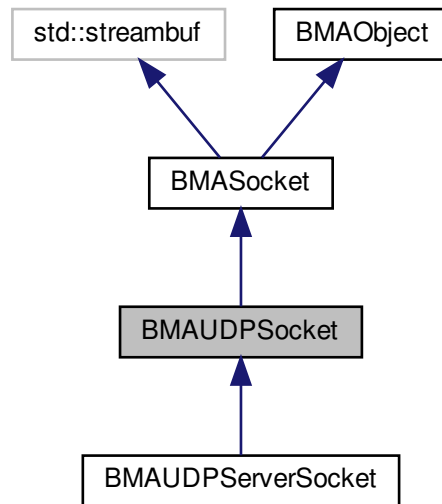
Reimplemented from [BMASocket](#).

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

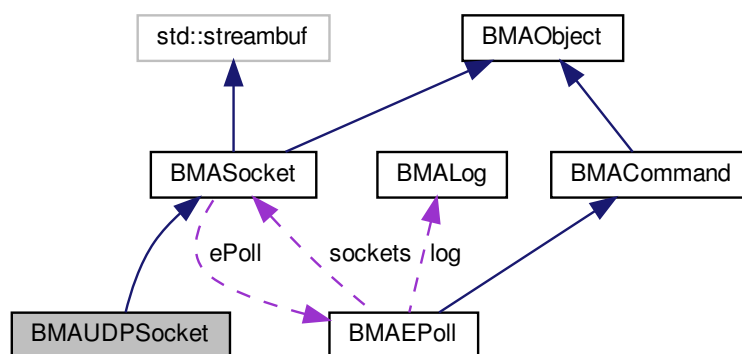
- /home/barant/Documents/Development/BMASockets/BMAUDPServerSocket.h
- /home/barant/Documents/Development/BMASockets/BMAUDPServerSocket.cpp

### 3.35 BMAUDPSocket Class Reference

Inheritance diagram for BMAUDPSocket:



Collaboration diagram for BMAUDPSocket:



#### Public Member Functions

- **BMAUDPSocket** ([BMAEPoll](#) &ePoll)

### Additional Inherited Members

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

- `/home/barant/Documents/Development/BMASockets/BMAUDPSocket.h`
- `/home/barant/Documents/Development/BMASockets/BMAUDPSocket.cpp`

# Index

- ~BMA Socket
  - BMA Socket, [39](#)
- accept
  - BMATCP ServerSocket, [48](#)
- BMA Account, [5](#)
- BMA Authenticate, [6](#)
- BMA Command, [7](#)
- BMA ConsoleServer, [8](#)
- BMA ConsoleSession, [10](#)
  - output, [11](#)
- BMAEPoll, [12](#)
  - registerSocket, [14](#)
  - unregisterSocket, [14](#)
- BMAFile, [15](#)
- BMAHTTPRequestHandler, [17](#)
- BMAHTTPServer, [18](#)
- BMAHTTPSession, [20](#)
  - onDataReceived, [21](#)
- BMAHeader, [16](#)
- BMA Log, [22](#)
- BMAMP3File, [22](#)
- BMAMP3StreamContentProvider, [23](#)
- BMAMP3StreamFrame, [24](#)
- BMAObject, [26](#)
- BMA ParseHeader, [26](#)
- BMAProperty< T >, [26](#)
- BMASIPINVITE, [29](#)
- BMASIPREGISTER, [30](#)
- BMASIPRequestHandler, [31](#)
- BMASIPServer, [33](#)
- BMASIPSession, [35](#)
  - onDataReceived, [36](#)
- BMA Session, [27](#)
  - onConnected, [28](#)
  - onDataReceived, [28](#)
  - output, [29](#)
- BMA Socket, [37](#)
  - ~BMA Socket, [39](#)
  - eventReceived, [39](#)
  - onConnected, [39](#)
  - onDataReceived, [39](#)
  - onRegistered, [40](#)
- BMAStreamContentProvider, [40](#)
- BMAStreamFrame, [41](#)
- BMAStreamServer, [42](#)
- BMAStreamSession, [44](#)
  - onDataReceived, [45](#)
- BMATCP ServerSocket, [46](#)
  - accept, [48](#)
  - onDataReceived, [48](#)
- BMATCP Socket, [48](#)
  - onConnected, [50](#)
  - output, [50](#)
- BMA Thread, [50](#)
- BMA Timer, [51](#)
  - onDataReceived, [52](#)
- BMAUDPServerSocket, [53](#)
  - onDataReceived, [54](#)
- BMAUDP Socket, [55](#)
- eventReceived
  - BMA Socket, [39](#)
- onConnected
  - BMA Session, [28](#)
  - BMA Socket, [39](#)
  - BMATCP Socket, [50](#)
- onDataReceived
  - BMAHTTPSession, [21](#)
  - BMASIPSession, [36](#)
  - BMA Session, [28](#)
  - BMA Socket, [39](#)
  - BMAStreamSession, [45](#)
  - BMATCP ServerSocket, [48](#)
  - BMA Timer, [52](#)
  - BMAUDPServerSocket, [54](#)
- onRegistered
  - BMA Socket, [40](#)
- output
  - BMA ConsoleSession, [11](#)
  - BMA Session, [29](#)
  - BMATCP Socket, [50](#)
- registerSocket
  - BMAEPoll, [14](#)
- unregisterSocket
  - BMAEPoll, [14](#)