medianet

1.0.0

Generated by Doxygen 1.8.16

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

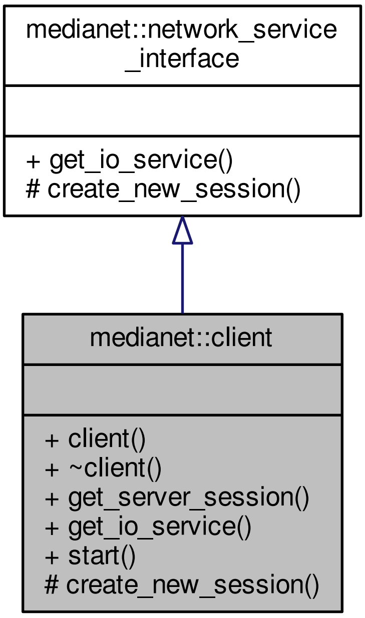
Class Documentation

3.1 medianet::client Class Reference

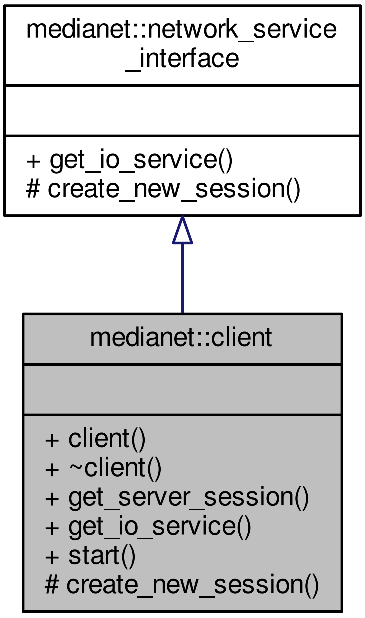
Can connect to only one host.

```
#include <client.h>
```

Inheritance diagram for medianet::client:



Collaboration diagram for medianet::client:



Public Member Functions

- session ∗ **get_server_session** ()
- io_service & **get_io_service** ()
- void start (std::string host, unsigned short port)

  Connect to given host and become client.

Protected Member Functions

- virtual session ∗ **create_new_session** (io_service &ios)
- virtual session ∗ **create_new_session** ()=0

3.1.1 Detailed Description

Can connect to only one host.

Author

leejm

3.1.2 Member Function Documentation

3.1.2.1 start()

```
void medianet::client::start (
             std::string host,
             unsigned short port )
```

Connect to given host and become client.

Parameters

host	IP or domain name.
port	Port.

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

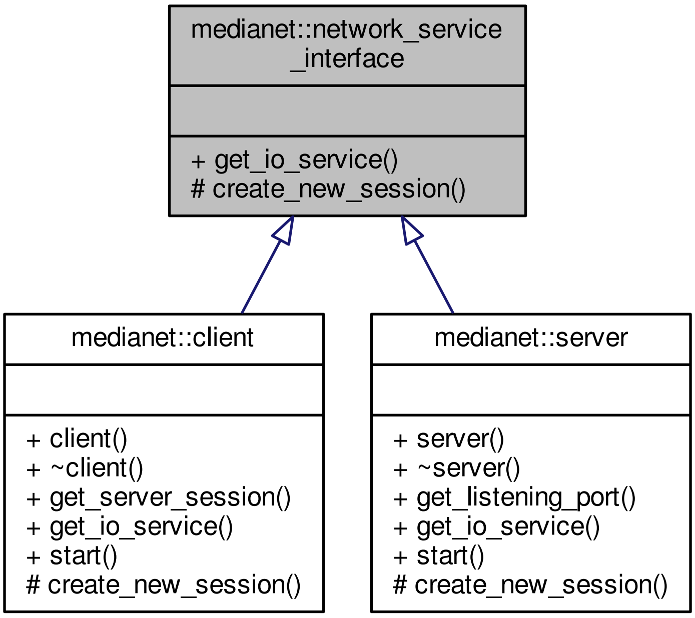
- /home/anz/workspace/lab/medianet/medianet/include/client.h
- /home/anz/workspace/lab/medianet/medianet/src/client.cpp

3.2 medianet::network_service_interface Class Reference

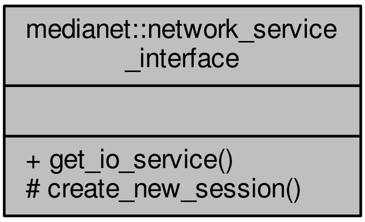
Network services are object that manages boost::asio::io_service and network session.

`#include <network_service_interface.h>`

Inheritance diagram for medianet::network_service_interface:



Collaboration diagram for medianet::network_service_interface:



Public Member Functions

- virtual io_service & **get_io_service** ()=0

Protected Member Functions

- virtual session ∗ **create_new_session** ()=0

3.2.1 Detailed Description

Network services are object that manages boost::asio::io_service and network session.

Author

leejm

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

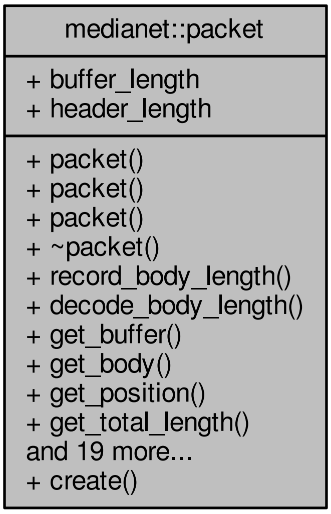
- /home/anz/workspace/lab/medianet/medianet/include/network_service_interface.h

3.3 medianet::packet Class Reference

A class for reading and writing byte stream.

```
#include <packet.h>
```

Collaboration diagram for medianet::packet:



Public Member Functions

- **packet** (char ∗buffer)
- **packet** (const packet &orig)
- void record_body_length ()
- void decode_body_length ()
- char ∗ **get_buffer** () const
- char ∗ **get_body** () const
- int **get_position** () const
- int get_total_length () const
- int get_body_length () const
- char **pop_byte** ()
- bool **pop_bool** ()
- int16_t **pop_int16** ()
- int32_t **pop_int32** ()
- int64_t **pop_int64** ()
- float **pop_single** ()
- double **pop_double** ()
- char ∗ **pop_byte_array** ()
- std::string **pop_string** ()
- void **push_byte** (char data)
- void **push_bool** (bool data)
- void **push_int16** (int16_t data)
- void **push_int32** (int32_t data)
- void **push_int64** (int64_t data)
- void **push_single** (float data)
- void **push_double** (double data)
- void **push_byte_array** (char ∗data, int16_t length)
- void **push_string** (std::string data)

Static Public Member Functions

- static boost::shared_ptr< packet > create ()

  Create new empty packet instance and wrap it in shared_ptr.

Static Public Attributes

- static const int buffer_length = 1440

  1440 is the maximum limit not to be segmented.
- static const int header_length = sizeof(int16_t)

  The header contains total body length.

3.3.1 Detailed Description

A class for reading and writing byte stream.

It doesn't care whether the structure is little endian or big endian.

Author

leejm

3.3.2 Member Function Documentation

3.3.2.1 create()

```
boost::shared_ptr< packet > medianet::packet::create ( )  [static]
```

Create new empty packet instance and wrap it in shared_ptr.

It is recommended to use this static method rather than constructors.

3.3.2.2 decode_body_length()

```
void medianet::packet::decode_body_length ( )
```

Read buffer and update m_body_length.

3.3.2.3 get_body_length()

```
int medianet::packet::get_body_length ( ) const
```

Returns the body length.

3.3.2.4 get_total_length()

```
int medianet::packet::get_total_length ( ) const
```

Returns the value of header length + body length.

3.3.2.5 record_body_length()

```
void medianet::packet::record_body_length ( )
```

Record final stream length on the header.

3.3.3 Member Data Documentation

3.3.3.1 buffer_length

```
const int medianet::packet::buffer_length = 1440  [static]
```

1440 is the maximum limit not to be segmented.

3.3.3.2 header_length

```
const int medianet::packet::header_length = sizeof(int16_t)  [static]
```

The header contains total body length.

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

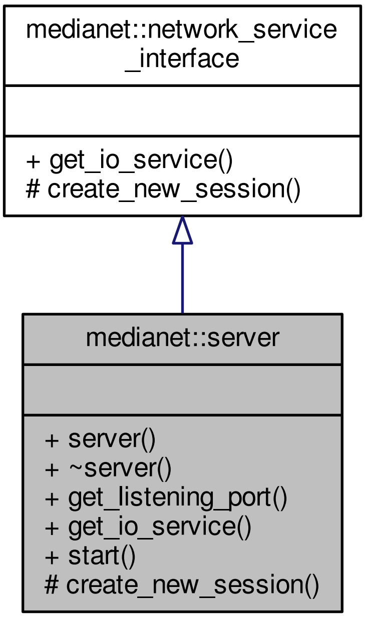
- /home/anz/workspace/lab/medianet/medianet/include/packet.h
- /home/anz/workspace/lab/medianet/medianet/src/packet.cpp

3.4 medianet::server Class Reference

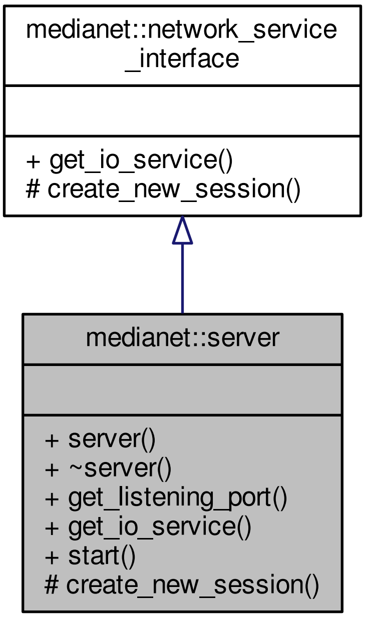
Can accept multiple connections and create one session per each connection.

```
#include <server.h>
```

Inheritance diagram for medianet::server:



Collaboration diagram for medianet::server:



Public Member Functions

- unsigned short **get_listening_port** () const
- io_service & **get_io_service** ()
- void start (unsigned short port=0)

  Start client listening and become host.

Protected Member Functions

- virtual session ∗ **create_new_session** (io_service &ios)
- virtual session ∗ **create_new_session** ()=0

3.4.1 Detailed Description

Can accept multiple connections and create one session per each connection.

Author

leejm

3.4.2 Member Function Documentation

3.4.2.1 start()

```
void medianet::server::start (
            unsigned short port = 0 )
```

Start client listening and become host.

Parameters

port	Assign listening port. If 0(default), than random port in the range of dynamic port domain will be assigned.

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

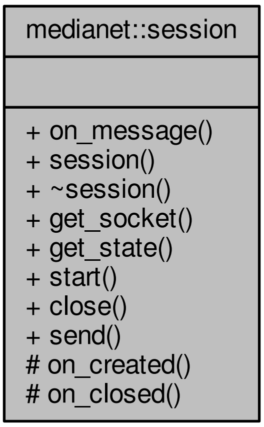
- /home/anz/workspace/lab/medianet/medianet/include/server.h
- /home/anz/workspace/lab/medianet/medianet/src/server.cpp

3.5 medianet::session Class Reference

Represents each connection.

```
#include <session.h>
```

Collaboration diagram for medianet::session:



Public Types

- enum state { **idle**, **connected**, reserve_closing, **closed** }

Public Member Functions

- virtual void on_message (packet msg)
- **session** (io_service &ios)
- tcp::socket & **get_socket** ()
- state **get_state** () const
- void **start** ()
- void **close** ()
- void send (boost::shared_ptr< packet > msg)

  Send packet.

Protected Member Functions

- virtual void on_created ()
- virtual void on_closed ()

3.5.1 Detailed Description

Represents each connection.

You can send message, handle received message, and observe socket status.

Author

leejm

3.5.2 Member Enumeration Documentation

3.5.2.1 state

```
enum medianet::session::state
```

Enumerator

reserve_closing	Disconnection reserved. If close() method is called while waiting for sending packets, it will be disconnected after all remaining packets are sent.

3.5.3 Member Function Documentation

3.5.3.1 on_closed()

```
void medianet::session::on_closed ( )  [protected], [virtual]
```

Be called right after the socket is closed.

3.5.3.2 on_created()

```
void medianet::session::on_created ( )  [protected], [virtual]
```

Be called right after the start() is called.

3.5.3.3 on_message()

```
void medianet::session::on_message (
             packet msg )  [virtual]
```

Be called when new packet has been received.

3.5.3.4 send()

```
void medianet::session::send (
             boost::shared_ptr< packet > msg )
```

Send packet.

So many copy operations may occurs if packet is passed as stack object. And parameter packet can't be a raw pointer object since sending operation is done asynchronously. Therefor, it takes packet as form of shared_ptr.

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

- /home/anz/workspace/lab/medianet/medianet/include/session.h
- /home/anz/workspace/lab/medianet/medianet/src/session.cpp

Index