yasio Documentation

yasio is a multi-platform support c++11 library with focus on asio (asynchronous socket I/O) for any client application.

- Cross-platform:
  - Compiler:
    - Visual Studio 2013+
    - GCC4.7+
    - xcode9+
    - Other Compilers which support C++11+
  - Architecture: x86, x64, ARM and etc.
  - OS: Windows, macOS, Linux, FreeBSD, iOS, Android And etc.
- Open source location: 中国@深圳

Quick Start

This demo simply send http request to `tool.chinaz.com` and print resposne data.

io_service Class

Provides the functionality of `tcp, udp, kcp and ssl-client` communication with noblocking-io model.

Syntax

```
namespace yasio { namespace inet { class io_service; } }
```

Members

Public Constructors

Name	Description
io_service::io_service	Constructs a `io_service` object.

Public Methods

Name	Description
io_service::start	Start the network service thread.
io_service::stop	Stop the network service thread.
io_service::open	Open channel.
io_service::close	Close transport.
io_service::is_open	Tests whether channel or transport is open.
io_service::dispatch	Dispatch the network io events.
io_service::write	Sends data asynchronous.
io_service::write_to	Sends data to specific remote asynchronous.

io_channel Class

Provides the functionality of establishing tcp/udp/kcp connections.

Syntax

```
namespace yasio { namespace inet { class io_channel; } }
```

Public Methods

Name	Description
io_channel::get_service	Gets belong service of channel.
io_channel::index	Gets index of channel at service.
io_channel::remote_port	Gets remote port of channel.

Remarks

Once io_service initialized, the max count of channel can't be changed.
Retrieves through `io_service::channel_at`.

io_channel::get_service

Gets owner service.

```
io_service& get_service()
```

io_channel::index

Gets channel index at service.

```
int index() const
```

io_event Class

The event produced by io_service thread.

Syntax

```
namespace yasio { namespace inet { class io_event; } }
```

Public Methods

Name	Description
io_event::kind	Gets kind of event.
io_event::status	Gets status of event.
io_event::packet	Gets packet of event.
io_event::timestamp	Gets timestamp of event.
io_event::transport	Gets transport of event.
io_event::transport_id	Gets transport id of event.
io_event::transport_udata	Gets/Sets transport user data.

.. _kind:

io_event::kind

Gets kind of event.

```
int kind() const;
```

obstream Class

Provides the functionality of Binary Writer.

Syntax

```
namespace yasio {
using obstream = basic_obstream<endian::network_convert_tag>;

// The fast binary writer without byte order convertion.
using fast_obstream = basic_obstream<endian::host_convert_tag>;
}
```

Members

Public Constructors

Name	Description
obstream::obstream	Constructs a `obstream` object.

Public Methods

Name	Description
obstream::write	Function template, write number value.
obstream::write_ix	Function template, write **7bit Encoded Int/Int64**.
obstream::write_v	Write blob data with **7bit Encoded Int lenght field**.
obstream::write_byte	Write 1 byte.
obstream::write_bytes	Write blob data without length field.
obstream::empty	Check is stream empty.

ibstream_view Class

Provides the functionality of Binary Reader.

Syntax

```
namespace yasio {
using ibstream_view = basic_ibstream_view<endian::network_convert_tag>;
using fast_ibstream_view = basic_ibstream_view<endian::host_convert_tag>;
}
```

Members

Public Constructors

Name	Description
ibstream_view::ibstream_view	Constructs a `ibstream_view` object.

Public Methods

Name	Description
ibstream_view::reset	Reset input data, weak reference.
ibstream_view::read	Function template, read number value.
ibstream_view:read_ix	Function template,read **7bit Encoded Int/Int64**.
ibstream_view:read_v	Read blob data with **7bit Encoded Int/Int64 lenght field**.
ibstream_view:read_byte	Read 1 byte.
ibstream_view:read_bytes	Read blob data without length field.
ibstream_view::empty	Check is stream empty.

xxsocket Class

Provides the functionality of low-level socket based on POSIX socket APIs, support std::move

Syntax

```
namespace yasio { namespace inet { class xxsocket; } }
```

Members

Name	Description
xxsocket::xxsocket	Constructs a `xxsocket` object.

Public Methods

Name	Description
xxsocket::xpconnect	Cnnect remote via tcp.
xxsocket::xpconnect_n	Connect remote via tcp non-blocking.
xxsocket::pconnect	Connect remote via tcp.
xxsocket::pconnect_n	Connect remote via tcp non-blocking.
xxsocket::pserve	Create socket as tcp server.
xxsocket::swap	Swap socket handle.
xxsocket::open	Open a socket.
xxsocket::reopen	Reopen a socket.
xxsocket::is_open	Check whether socket opened.

io_service options

The following are the io_service options.

Name	Description
YOPT_S_DEFER_EVENT_CB	Set defer event callback params: callback:defer_event_cb_t remarks: a. User can do custom packet resolve at network thread, such as decompress and crc check. b. Return true, io_service will continue enque to event queue. c. Return false, io_service will drop the event.
YOPT_S_DEFERRED_EVENT	Set whether deferred dispatch event, default is: 1 params: deferred_event:int(1)
YOPT_S_RESOLV_FN	Set custom resolve function, native C++ ONLY params: func:resolv_fn_t*
YOPT_S_PRINT_FN	Set custom print function native C++ ONLY parmas: func:print_fn_t remarks: you must ensure thread safe of it
YOPT_S_PRINT_FN2	Set custom print function with log level parmas: func:print_fn2_t you must ensure thread safe of it
YOPT_S_EVENT_CB	Set event callback params: func:event_cb_t*
YOPT_S_TCP_KEEPALIVE	Set tcp keepalive in seconds, probes is tries. params: idle:int(7200), interal:int(75), probes:int(10)
YOPT_S_NO_NEW_THREAD	Don't start a new thread to run event loop. params: value:int(0)
YOPT_S_SSL_CACERT	Sets ssl verification cert, if empty, don't verify. params: path:const char*

yasio Macros

The macros listed in the table below may be used to control the interface, functionality, and behaviour of `yasio`.
You can define them at yasio/detail/config.hpp or compiler preprocessors.

Name	Description
YASIO_HAVE_KCP	Whether enable kcp, default: `off`
YASIO_HEADER_ONLY	Whether enable header only, default: `off`
YASIO_SSL_BACKEND	Choose ssl backend, since 3.36.0 `1`. Use OpenSSL `2`. Use mbedtls
YASIO_ENABLE_UDS	Whether enable unix domain socket support, current only unix-like system and win10 RS5 support this feature, default: `off`
YASIO_HAVE_CARES	Whether use c-ares to resolve domain name, default: `off`
YASIO_VERBOSE_LOG	Whether enable verbose log, default: `off`
YASIO_NT_COMPAT_GAI	Whether enable windows xp `getaddrinfo` API compatible, default: `off`
YASIO_USE_SPSC_QUEUE	Whether use SPSC queue, default: `off`
YASIO_HAVE_HALF_FLOAT	Whether enable half float, depends on half.hpp
YASIO_DISABLE_OBJECT_POOL	Whether disable object pool
YASIO_DISABLE_CONCURRENT_SINGLETON	Whether disable concurrent singleton

FAQ

FAQ

Can't load xlua bundle on macOS?

The file `xlua.bundle` needs change attr by command `sudo xattr -r -d com.apple.quarantine xlua.bundle`