
udho

Neel Basu (Sunanda Bose)

Dec 04, 2020

CONTENTS:

1	Udho API	1
2	Quickstart	287
3	Building udho	289
4	URL Routing	291
5	Request Context	293
6	HTTP Server	297
7	Form	301
8	HTTP Session	305
9	Logging	307
10	Async Task Graph	309
11	Udho ()	321
12	Dependencies:	323
13	Features:	325
14	Stateful Example	327
15	Indices and tables	329
	Index	331

UDHO API

1.1 Class Hierarchy

1.2 File Hierarchy

1.3 Full API

1.3.1 Namespaces

Namespace udho

Contents

- *Namespaces*
- *Classes*
- *Enums*
- *Functions*
- *Typedefs*

Namespaces

- *Namespace udho::activities*
- *Namespace udho::cache*
- *Namespace udho::compositors*
- *Namespace udho::configs*
- *Namespace udho::contexts*
- *Namespace udho::defs*
- *Namespace udho::detail*
- *Namespace udho::exceptions*
- *Namespace udho::form*

- *Namespace udho::forms*
- *Namespace udho::internal*
- *Namespace udho::loggers*
- *Namespace udho::logging*
- *Namespace udho::servers*
- *Namespace udho::util*
- *Namespace udho::view*
- *Namespace udho::visitors*

Classes

- *Template Struct app_*
- *Template Struct app_< AppT, true >*
- *Struct assoc*
- *Template Struct attachment*
- *Template Struct attachment< AuxT, LoggerT, void >*
- *Template Struct attachment< AuxT, void, CacheT >*
- *Template Struct attachment< AuxT, void, void >*
- *Template Struct basic_router*
- *Template Struct bounded_str*
- *Template Struct bridge*
- *Template Struct client_options_*
- *Struct client_options_::follow_redirect_t*
- *Struct client_options_::http_version_t*
- *Struct client_options_::verify_certificate_t*
- *Template Struct config*
- *Template Struct config< configs::server_<> >*
- *Template Struct configuration*
- *Struct connection::send_lambda*
- *Template Struct content_wrapper0*
- *Template Struct content_wrapper1*
- *Template Struct context*
- *Template Struct context< AuxT, RequestT, void >*
- *Template Struct cookie_*
- *Template Struct cookies_*
- *Template Struct data_helper*
- *Template Struct data_helper< H, T, false >*

- *Template Struct data_helper< H, T, true >*
- *Struct declaration*
- *Template Struct field*
- *Template Struct field< T, false >*
- *Template Struct field< T, true >*
- *Struct field_common*
- *Template Struct field_value_extractor*
- *Template Struct field_value_extractor< std::chrono::time_point< std::chrono::system_clock, std::chrono::microseconds > >*
- *Template Struct form_*
- *Template Struct is_application*
- *Template Struct lookup_table*
- *Template Struct lookup_table< udho::prepared< DataT > >*
- *Template Struct lookup_table< udho::prepared_group< U, V > >*
- *Template Struct module_evaluator*
- *Template Struct module_evaluator< VisitorT, ModuleT, false >*
- *Template Struct module_evaluator< VisitorT, ModuleT, true >*
- *Struct module_info*
- *Template Struct module_overload*
- *Template Struct module_overload< Function, compositors::deferred >*
- *Template Struct multipart_form*
- *Struct multipart_form::form_part*
- *Template Struct overload_group*
- *Template Struct overload_group< U, app_< V, Ref > >*
- *Template Struct overload_group< U, overload_terminal< V > >*
- *Template Struct overload_group< U, overload_terminal< void > >*
- *Template Struct overload_group_helper*
- *Template Struct overload_group_helper< OverloadT, void >*
- *Template Struct overload_terminal*
- *Template Struct overload_terminal< void >*
- *Template Struct prepare*
- *Template Struct prepared*
- *Template Struct prepared< T, true >*
- *Template Struct prepared_group*
- *Template Struct prepared_group< H, void >*
- *Template Struct proxy*

- *Template Struct proxy< configs::server_<>::mimes_t, config< configs::server_<> > >*
- *Struct reroute*
- *Template Struct server*
- *Template Struct server< AuxT, void, CacheT >*
- *Template Struct session_*
- *Template Struct session_< RequestT, void >*
- *Template Struct start*
- *Template Struct start_*
- *Struct url*
- *Template Struct urlencoded_form*
- *Template Struct visitor*
- *Template Struct watch*
- *Template Struct watcher*
- *Template Class application*
- *Template Class connection*
- *Template Class listener*
- *Class url::param*

Enums

- *Enum form_type*

Functions

- *Template Function udho::_search*
- *Template Function udho::after(T&...)*
- *Function udho::after()*
- *Template Function udho::analyze*
- *Template Function udho::app(T...)*
- *Template Function udho::app(AppT&)*
- *Template Function udho::associate*
- *Template Function udho::collect*
- *Template Function udho::cookie*
- *Template Function udho::data(const H&, const T&)*
- *Template Function udho::data(const H&, const T&, const Rest&...)*
- *Template Function udho::data(const T&)*
- *Template Function udho::del(F)*

- *Template Function* `udho::del(F, AI&)`
- *Template Function* `udho::get(F)`
- *Template Function* `udho::get(F, AI&)`
- *Template Function* `udho::head(F)`
- *Template Function* `udho::head(F, AI&)`
- *Template Function* `udho::operator/=`
- *Template Function* `udho::operator<<(session_<RequestT, ShadowT>&, const T&)`
- *Template Function* `udho::operator<<(attachment<AuxT, LoggerT, CacheT>&, const udho::logging::message<Status>&)`
- *Template Function* `udho::operator<<(std::ostream&, const proxy<K, C>&)`
- *Template Function* `udho::operator<<(const overload_group<U, V>&, const F&)`
- *Template Function* `udho::operator<<(StreamT&, const overload_group<U, V>&)`
- *Template Function* `udho::operator<<(cookies_<RequestT>&, const udho::cookie_<V>&)`
- *Template Function* `udho::operator<<(udho::context<AuxT, RequestT, ShadowT>&, const exceptions::http_error&)`
- *Template Function* `udho::operator<<(module_overload<Function, CompositorT>, const char(&))`
- *Template Function* `udho::operator<<(const detail::context_common<AuxT, RequestT>&, const udho::logging::message<Status>&)`
- *Template Function* `udho::operator>>(const session_<RequestT, ShadowT>&, T&)`
- *Template Function* `udho::operator>>(const char(&), module_overload<Function, CompositorT>)`
- *Template Function* `udho::operator|(const overload_group<U, V>&, const F&)`
- *Template Function* `udho::operator|(const prepared_group<U, V>&, const udho::prepared<P>&)`
- *Template Function* `udho::operator|(const udho::prepared<U>&, const udho::prepared<V>&)`
- *Template Function* `udho::overload(boost::beast::http::verb, F, CompositorT<typename internal::function_signature<F>::return_type>)`
- *Template Function* `udho::overload(boost::beast::http::verb, F, AI&, CompositorT<typename internal::function_signature<F>::return_type>)`
- *Template Function* `udho::post(F)`
- *Template Function* `udho::post(F, AI&)`
- *Template Function* `udho::put(F)`
- *Template Function* `udho::put(F, AI&)`
- *Template Function* `udho::scope`
- *Template Function* `udho::visit`

Typedefs

- *Typedef udho::accessor*
- *Typedef udho::activity*
- *Typedef udho::client_options*
- *Typedef udho::configuration_type*
- *Typedef udho::perform*
- *Typedef udho::require*
- *Typedef udho::router*
- *Typedef udho::tcp*

Namespace udho::activities

Contents

- *Namespaces*
- *Classes*
- *Enums*
- *Functions*

Namespaces

- *Namespace udho::activities::detail*

Classes

- *Template Struct accessor*
- *Template Struct activity*
- *Template Struct after*
- *Template Struct after< HeadT >*
- *Struct after_none*
- *Template Struct apply_helper*
- *Template Struct collector*
- *Template Struct collector< ContextT, dataset< T... > >*
- *Template Struct combinator*
- *Template Struct combinator< NextT, start< ContextT, T... > >*
- *Template Struct dataset*
- *Template Struct fixed_key_accessor*

- *Template Struct joined*
- *Template Struct joined< CallbackT, activities::collector< ContextT, dataset< T... >>>*
- *Template Struct junction*
- *Template Struct perform*
- *Template Struct perform::require*
- *Template Struct require*
- *Template Struct result*
- *Template Struct result_data*
- *Template Struct start*
- *Template Struct subtask*
- *Template Struct subtask< ActivityT >*
- *Template Struct subtask< start< ContextT, T... >>*
- *Template Class analyzer*

Enums

- *Enum state*

Functions

- *Template Function udho::activities::collect*
- *Template Function udho::activities::operator<<(collector<ContextT, dataset<T...>>&, const U&)*
- *Template Function udho::activities::operator<<(accessor<T...>&, const U&)*
- *Function udho::activities::operator<<(std::ostream&, const state&)*
- *Template Function udho::activities::operator>>(const collector<ContextT, dataset<T...>>&, U&)*
- *Template Function udho::activities::operator>>(const accessor<T...>&, U&)*

Namespace udho::activities::detail

Contents

- *Classes*
- *Typedefs*

Classes

- *Template Struct after*
- *Template Struct after< subtask< ActivityT, DependenciesT... > >*
- *Template Struct after< udho::start_< ContextT, T... > >*
- *Template Struct apply_helper_*
- *Template Struct apply_helper_< false, FunctorT, TargetsT... >*
- *Template Struct final_intermediate*

Typedefs

- *Typedef udho::activities::detail::apply_helper*

Namespace udho::cache

Contents

- *Namespaces*
- *Classes*

Namespaces

- *Namespace udho::cache::storage*

Classes

- *Template Struct abstract_engine*
- *Template Struct content*
- *Template Struct content< void >*
- *Template Struct driver*
- *Template Struct engine*
- *Template Struct flake*
- *Template Struct generator*
- *Template Struct generator< boost::uuids::uuid >*
- *Template Struct master*
- *Template Struct registry*
- *Template Struct shadow*
- *Template Struct store*

Namespace udho::cache::storage

Contents

- *Classes*

Classes

- *Template Struct disk*
- *Template Struct memory*
- *Struct redis*

Namespace udho::compositors

Contents

- *Detailed Description*
- *Classes*

Detailed Description

compositors for content

Classes

- *Template Struct deferred*
- *Template Struct mimed*
- *Template Struct transparent*

Namespace udho::configs

Contents

- *Classes*
- *Typedefs*

Classes

- *Template Struct form_*
- *Template Struct logger_*
- *Template Struct router_*
- *Template Struct server_*
- *Struct server_::document_root_t*
- *Struct server_::mime_default_t*
- *Struct server_::mimes_t*
- *Struct server_::template_root_t*
- *Template Struct session_*
- *Struct session_::extension_t*
- *Struct session_::id_t*
- *Struct session_::path_t*
- *Struct session_::serialization_t*

Typedefs

- *Typedef udho::configs::form*
- *Typedef udho::configs::logger*
- *Typedef udho::configs::router*
- *Typedef udho::configs::server*
- *Typedef udho::configs::session*

Namespace udho::contexts

Contents

- *Typedefs*

Typedefs

- *Typedef udho::contexts::stateful*
- *Typedef udho::contexts::stateless*

Namespace udho::defs

Contents

- *Typedefs*

Typedefs

- *Typedef udho::defs::request_type*
- *Typedef udho::defs::response_type*
- *Typedef udho::defs::session_key_type*

Namespace udho::detail

Contents

- *Classes*
- *Functions*
- *Typedefs*

Classes

- *Template Struct association*
- *Template Struct association_group*
- *Template Struct association_group< U, void >*
- *Template Struct association_group_visitor*
- *Template Struct association_group_visitor< association_group< U, V >, false >*
- *Template Struct association_group_visitor< association_group< U, V >, true >*
- *Template Struct association_group_visitor< association_group< U, void >, false >*
- *Struct association_leaf*
- *Template Struct association_lexical_extractor*
- *Template Struct association_value_extractor*
- *Template Struct async_result*
- *Template Struct client_connection_wrapper*
- *Template Struct context_common*
- *Template Struct context_impl*
- *Template Struct http_client_connection*
- *Template Struct https_client_connection*

- *Struct interaction_*
- *Template Struct is_prepared*
- *Template Struct responder*
- *Template Struct responder< F, std::map< U, V > >*
- *Template Struct responder< F, std::vector< V > >*
- *Struct route*
- *Template Struct url_data_*
- *Struct url_data_::host_t*
- *Struct url_data_::path_t*
- *Struct url_data_::port_t*
- *Struct url_data_::protocol_t*
- *Struct url_data_::query_t*
- *Struct url_data_::target_t*
- *Template Class is_streamable*

Functions

- *Template Function udho::detail::operator|*
- *Template Function udho::detail::visit*

Typedefs

- *Typedef udho::detail::url_data*

Namespace udho::exceptions

Contents

- *Classes*
- *Functions*

Classes

- *Struct http_error*
- *Struct reroute*

Functions

- *Function `udho::exceptions::http_redirection`*

Namespace `udho::form`

Contents

- *Namespaces*
- *Classes*
- *Functions*
- *Typedefs*

Namespaces

- *Namespace `udho::form::validators`*

Classes

- *Template Struct `validated`*
- *Template Struct `validated< void >`*

Functions

- *Template Function `udho::form::operator<<(validated<udho::multipart_form<IteratorT>>&, field<T, Required>&)`*
- *Template Function `udho::form::operator<<(validated<>&, field<T, Required>&)`*
- *Template Function `udho::form::operator<<(validated<udho::form_<RequestT>>&, field<T, Required>&)`*
- *Function `udho::form::validate()`*
- *Template Function `udho::form::validate(udho::form_<RequestT>&)`*
- *Template Function `udho::form::validate(const udho::multipart_form<IteratorT>&)`*

Typedefs

- *Typedef `udho::form::optional`*
- *Typedef `udho::form::required`*

Namespace udho::form::validators

Contents

- *Classes*

Classes

- *Struct all_digit*
- *Struct date_time*
- *Struct exact_length*
- *Struct max_length*
- *Struct min_length*
- *Struct no_space*

Namespace udho::forms

Contents

- *Namespaces*
- *Classes*
- *Functions*
- *Typedefs*
- *Variables*

Namespaces

- *Namespace udho::forms::constraints*
- *Namespace udho::forms::detail*
- *Namespace udho::forms::drivers*

Classes

- *Struct accumulated*
- *Template Struct deserializer*
- *Template Struct deserializer< std::chrono::time_point< std::chrono::system_clock, DurationT >, std::string >*
- *Template Struct deserializer< std::string, std::chrono::time_point< std::chrono::system_clock, DurationT > >*
- *Template Struct deserializer< std::string, std::string >*
- *Template Struct deserializer< std::string, U >*

- *Template Struct deserializer< U, U >*
- *Template Struct deserializer< V, std::string >*
- *Template Struct field*
- *Template Struct field< std::chrono::time_point< std::chrono::system_clock, DurationT >, false >*
- *Template Struct field< std::chrono::time_point< std::chrono::system_clock, DurationT >, true >*
- *Template Struct field< T, false >*
- *Template Struct field< T, true >*
- *Template Struct form*
- *Template Struct parser*
- *Template Struct validated*
- *Template Struct validated< form< DriverT > >*

Functions

- *Template Function udho::forms::form_multipart*
- *Template Function udho::forms::operator<<(accumulated&, detail::constrained_field<ValidatorT, FieldT>&)*
- *Template Function udho::forms::operator<<(accumulated&, field<T, Required>&)*
- *Template Function udho::forms::operator>>(form<DriverT>&, detail::constrained_field<ValidatorT, FieldT>&)*
- *Template Function udho::forms::operator>>(form<DriverT>&, field<T, Required>&)*
- *Template Function udho::forms::operator>>(validated<form<DriverT>>&, detail::constrained_field<ValidatorT, FieldT>&)*
- *Template Function udho::forms::operator>>(validated<form<DriverT>>&, field<T, Required>&)*
- *Template Function udho::forms::validate*

Typedefs

- *Typedef udho::forms::form_*
- *Typedef udho::forms::form_multipart_*
- *Typedef udho::forms::optional*
- *Typedef udho::forms::query_*
- *Typedef udho::forms::required*

Variables

- Variable *udho::forms::default_datetime_format*

Namespace **udho::forms::constraints**

Contents

- *Classes*
- *Typedefs*

Classes

- *Struct all_digits*
- *Template Struct basic_constrain*
- *Template Struct eq*
- *Template Struct gt*
- *Template Struct gt< std::string >*
- *Template Struct gte*
- *Template Struct gte< std::string >*
- *Template Struct in*
- *Struct length_eq*
- *Template Struct lt*
- *Template Struct lt< std::string >*
- *Template Struct lte*
- *Template Struct lte< std::string >*
- *Template Struct neq*
- *Struct no_space*

Typedefs

- *Typedef udho::forms::constraints::length_gt*
- *Typedef udho::forms::constraints::length_gte*
- *Typedef udho::forms::constraints::length_lt*
- *Typedef udho::forms::constraints::length_lte*
- *Typedef udho::forms::constraints::length_max*
- *Typedef udho::forms::constraints::length_min*
- *Typedef udho::forms::constraints::max*

- *Typedef* `udho::forms::constraints::min`

Namespace `udho::forms::detail`

Contents

- *Classes*
- *Typedefs*

Classes

- *Template Struct* `basic_field`
- *Template Struct* `constrained_field`
- *Template Struct* `constrained_field< ValidatorT, constrained_field< T... > >`
- *Template Struct* `constraint_visitor_`
- *Template Struct* `constraint_visitor_< ConstrainedFieldT, 0 >`
- *Struct* `field_common`
- *Template Struct* `field_data`

Typedefs

- *Typedef* `udho::forms::detail::constraint_visitor`

Namespace `udho::forms::drivers`

Contents

- *Namespaces*
- *Classes*
- *Typedefs*

Namespaces

- *Namespace* `udho::forms::drivers::detail`

Classes

- *Template Struct combo*
- *Template Struct multipart*
- *Template Struct multipart_*
- *Struct multipart_::form_part*
- *Template Struct urlencoded*
- *Template Struct urlencoded_*

Typedefs

- *Typedef udho::forms::drivers::multipart_raw*
- *Typedef udho::forms::drivers::urlencoded_raw*

Namespace udho::forms::drivers::detail

Contents

- *Classes*

Classes

- *Template Struct extract_multipart_boundary*

Namespace udho::internal

Contents

- *Classes*
- *Functions*

Classes

- *Template Struct actual_callback_type*
- *Template Struct arg_to_tuple*
- *Template Struct arg_to_tuple< TupleT, 0 >*
- *Template Struct bind_first*
- *Template Struct bind_first< boost::function< R(Args...) > >*
- *Template Struct callable1*

- *Template Struct callable1< AI, R(*) (AI &, V...)>*
- *Template Struct cast_optionally*
- *Template Struct function_signature*
- *Template Struct function_signature< boost::function< R(Args...)> >*
- *Template Struct function_signature< R(*) (Args...)>*
- *Template Struct member_*
- *Template Struct member_< R C::* >*
- *Template Struct reduced_*
- *Template Struct reduced_< R(C::*)(Args...) const >*
- *Template Struct reduced_< R(C::*)(Args...)>*
- *Template Struct reducer*

Functions

- *Template Function udho::internal::arguments_to_tuple*
- *Template Function udho::internal::html_summary*
- *Template Function udho::internal::member*
- *Function udho::internal::path_cat*
- *Function udho::internal::path_inside*
- *Template Function udho::internal::reduced*
- *Template Function udho::internal::to_fusion(std::index_sequence<Is...>, T&&)*
- *Template Function udho::internal::to_fusion(boost::tuple<Ts...>)*
- *Template Function udho::internal::to_fusion(std::tuple<Ts...>)*

Namespace udho::loggers

Contents

- *Classes*
- *Typedefs*

Classes

- *Template Struct plain*

Typedefs

- *Typedef udho::loggers::ostream*

Namespace udho::logging

Contents

- *Namespaces*
- *Classes*
- *Enums*

Namespaces

- *Namespace udho::logging::features*
- *Namespace udho::logging::messages*

Classes

- *Template Struct message*

Enums

- *Enum segment*
- *Enum status*

Namespace udho::logging::features

Contents

- *Classes*

Classes

- *Struct colored*

Namespace udho::logging::messages

Contents

- *Namespaces*
- *Typedefs*

Namespaces

- *Namespace udho::logging::messages::formatted*

Typedefs

- *Typedef udho::logging::messages::debug*
- *Typedef udho::logging::messages::error*
- *Typedef udho::logging::messages::info*
- *Typedef udho::logging::messages::warning*

Namespace udho::logging::messages::formatted

Contents

- *Classes*
- *Functions*
- *Typedefs*

Classes

- *Template Struct message*

Functions

- *Template Function* `udho::logging::messages::formatted::operator<<`

Typedefs

- *Typedef* `udho::logging::messages::formatted::debug`
- *Typedef* `udho::logging::messages::formatted::error`
- *Typedef* `udho::logging::messages::formatted::info`
- *Typedef* `udho::logging::messages::formatted::warning`

Namespace `udho::servers`

Contents

- *Namespaces*
- *Classes*

Namespaces

- *Namespace* `udho::servers::ostreamed`
- *Namespace* `udho::servers::quiet`
- *Namespace* `udho::servers::stateless`

Classes

- *Template Struct* `logged`
- *Template Struct* `stateful`

Namespace `udho::servers::ostreamed`

Contents

- *Classes*
- *Typedefs*

Classes

- *Template Struct ostreamed_helper*

Typedefs

- *Typedef udho::servers::ostreamed::stateful*
- *Typedef udho::servers::ostreamed::stateless*

Namespace udho::servers::quiet

Contents

- *Typedefs*

Typedefs

- *Typedef udho::servers::quiet::stateful*
- *Typedef udho::servers::quiet::stateless*

Namespace udho::servers::stateless

Contents

- *Typedefs*

Typedefs

- *Typedef udho::servers::stateless::logged*
- *Typedef udho::servers::stateless::quiet*

Namespace udho::util

Contents

- *Classes*
- *Functions*

Classes

- *Template Struct is_invocable*

Functions

- *Function udho::util::charToHex*
- *Template Function udho::util::from_hex*
- *Function udho::util::hexToChar*
- *Template Function udho::util::to_hex*
- *Template Function udho::util::urldecode*
- *Template Function udho::util::urlencode*

Namespace udho::view

Contents

- *Namespaces*
- *Classes*
- *Functions*

Namespaces

- *Namespace udho::view::parsing*

Classes

- *Template Struct parser*

Functions

- *Template Function udho::view::expression*
- *Template Function udho::view::processor*

Namespace udho::view::parsing

Contents

- *Namespaces*
- *Classes*
- *Functions*

Namespaces

- *Namespace udho::view::parsing::attrs*
- *Namespace udho::view::parsing::tags*

Classes

- *Template Struct expression*
- *Struct token*
- *Template Struct xml*

Functions

- *Template Function udho::view::parsing::tokenize*

Namespace udho::view::parsing::attrs

Contents

- *Classes*

Classes

- *Template Struct add_class*
- *Template Struct condition*
- *Template Struct eval*
- *Template Struct target*

Namespace `udho::view::parsing::tags`

Contents

- *Classes*

Classes

- *Template Struct block*
- *Template Struct condition*
- *Template Struct eval*
- *Template Struct html*
- *Template Struct loop*
- *Template Struct skip*
- *Template Struct text*
- *Template Struct var*

Namespace `udho::visitors`

Contents

- *Classes*
- *Functions*
- *Typedefs*

Classes

- *Template Struct printing_visitor*
- *Template Struct printing_visitor_html*
- *Template Struct target_application*
- *Template Struct target_application< F, Visitables, 0 >*
- *Template Struct target_module*
- *Template Struct target_module< F, Visitables, 0 >*
- *Template Struct targetted_visitor*
- *Struct visitable*

Functions

- *Template Function* `udho::visitors::print`
- *Template Function* `udho::visitors::print_html`
- *Template Function* `udho::visitors::target`

Typedefs

- *Typedef* `udho::visitors::targeted`

1.3.2 Classes and Structs

Template Struct accessor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::fixed_key_accessor< udho::cache::shadow< std::string, T::result_type... > > (Template Struct fixed_key_accessor)`

Struct Documentation

template<typename ...**T**>

struct `udho::activities::accessor` : **public** `udho::activities::fixed_key_accessor<udho::cache::shadow<std::string, T::`

Access a subset of data from the collector

Public Types

typedef `fixed_key_accessor<udho::cache::shadow<std::string, typename T::result_type...>>` **base_type**

typedef `udho::cache::shadow<std::string, typename T::result_type...>` **shadow_type**

Public Functions

template<typename **Context****T**, typename ...**U**>

accessor (std::shared_ptr<*collector*<*Context***T**, *dataset*<**U**...>>> *collector*)

template<typename ...**U**>

accessor (*accessor*<**U**...> *accessor*)

std::string **name** () **const**

shadow_type &**shadow** ()

const *shadow_type* &**shadow** () **const**

template<typename **V**>

bool **exists** () const
Whether there exists any data for activity V

Template Parameters

- V: Activity Type

template<typename V>
const V::result_type &**get** () const
get data associated with activity V

Template Parameters

- V: activity type

template<typename V>
bool **completed** () const
Check whether activity V has completed.

Template Parameters

- V: activity type

template<typename V>
bool **canceled** () const
Check whether activity V has been canceled.

Template Parameters

- V: activity type

template<typename V>
bool **failed** () const
Check whether activity V has failed (only the failure data of V is valid).

Template Parameters

- V: activity type

template<typename V>
bool **okay** () const
Check whether activity V is okay.

Template Parameters

- V: activity type

template<typename V>
V::result_type::success_type **success** () const
get success data for activity V

Template Parameters

- V: activity type

template<typename V>
V::result_type::failure_type **failure** () const
get failure data for activity V

Template Parameters

- V: activity type

template<typename V>
void **set** (const typename V::result_type &value)

```
template<typename V, typename F>
void apply (F f) const
    Apply a callback on result of V
```

Template Parameters

- V: activity type

Parameters

- f: callback

Public Members

shadow_type **_shadow**

Template Struct activity

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Types

- `public std::enable_shared_from_this< DerivedT >`
- `public udho::activities::result< SuccessDataT, FailureDataT >` (*Template Struct result*)

Struct Documentation

```
template<typename DerivedT, typename SuccessDataT = void, typename FailureDataT = void>
```

```
struct udho::activities::activity : public std::enable_shared_from_this<DerivedT>, public udho::activities::result
```

An activity A must subclass from `activity<A, SuccessA, FailureA>` assuming `SuccessA` and `FailureA` are the types that contains the relevant information regarding its success or failure. The activity A must overload a no argument `operator() ()` which initiates the activity. After the activity is initiated either `success()` or `failure()` methods must be called in order to signal its completion. The activity A must take the collector as the first argument to its constructor, which is passed to the base class `activity<A, SuccessA, FailureA>`. Hence its preferred to take the first parameter to the constructor as template parameter.

Template Parameters

- `DerivedT`: Activity Class
- `SuccessDataT`: data associated to the activity if the activity succeeds
- `FailureDataT`: data associated to the activity if the activity fails

Public Types

```
typedef std::shared_ptr<DerivedT> derived_ptr_type
```

Public Functions

```
template<typename StoreT>  
activity (StoreT &store)  
  
derived_ptr_type self ()  
    shared_ptr to this
```

Template Struct after

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::detail::after< HeadT > (Template Struct after)`

Struct Documentation

```
template<typename HeadT, typename ...TailT>  
struct udho::activities::after : public udho::activities::detail::after<HeadT>
```

Public Functions

```
after (HeadT &head, TailT&... tail)
```

```
template<typename ActivityT, typename ...Args>  
subtask<ActivityT, typename HeadT::activity_type, typename TailT::activity_type...> perform (Args&&...  
args)
```

```
template<typename ActivityT, typename ...DependenciesT>  
void attach (subtask<ActivityT, DependenciesT...> &sub)
```

```
template<typename CallbackT, typename StartT>  
subtask<joined<CallbackT, typename StartT::collector_type>, typename HeadT::activity_type, typename TailT::activity_
```

Template Struct after< HeadT >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::detail::after< HeadT > (Template Struct after)`

Struct Documentation

```
template<typename HeadT>
struct udho::activities::after<HeadT> : public udho::activities::detail::after<HeadT>
```

Public Functions

```
after (HeadT &head)
```

```
template<typename ActivityT, typename ...Args>
subtask<ActivityT, typename HeadT::activity_type> perform (Args&&... args)
```

```
template<typename ActivityT, typename ...DependenciesT>
void attach (subtask<ActivityT, DependenciesT...> &sub)
```

```
template<typename CallbackT, typename StartT>
subtask<joined<CallbackT, typename StartT::collector_type>, typename HeadT::activity_type> finish (StartT
                                                                 &starter,
                                                                 Call-
                                                                 backT
                                                                 call-
                                                                 back)
```

Struct after_none

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
struct udho::activities::after_none
```

Public Functions

```
template<typename ActivityT, typename ...Args>
subtask<ActivityT> perform (Args&&... args)
```

Template Struct `apply_helper`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Types

- `public udho::activities::detail::apply_helper_< FunctorT > (Template Struct apply_helper_)`
- `public udho::activities::detail::apply_helper_< FunctorT, SuccessT > (Template Struct apply_helper_)`
- `public udho::activities::detail::apply_helper_< FunctorT, FailureT > (Template Struct apply_helper_)`
- `public udho::activities::detail::apply_helper_< FunctorT, SuccessT, FailureT > (Template Struct apply_helper_)`

Struct Documentation

```
template<typename FunctorT, typename SuccessT, typename FailureT>
```

```
struct udho::activities::apply_helper : public udho::activities::detail::apply_helper_<FunctorT>, public udho::
```

Public Functions

```
apply_helper (FunctorT &f)
```

```
void operator () ()
```

```
void operator () (const SuccessT &s)
```

```
void operator () (const FailureT &f)
```

```
void operator () (const SuccessT &s, const FailureT &f)
```

Public Members

```
FunctorT &_ftor
```

Template Struct collector

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ContextT, typename DatasetT>
struct collector
```

Template Struct collector< ContextT, dataset< T... > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Types

- `public udho::activities::dataset< T... >` (*Template Struct dataset*)
- `public std::enable_shared_from_this< collector< ContextT, dataset< T... > >`

Struct Documentation

```
template<typename ContextT, typename ...T>
```

```
struct udho::activities::collector<ContextT, dataset<T...>> : public udho::activities::dataset<T...>, public std::
    Collects data associated with all activities involved in the subtask graph
```

Public Types

```
typedef dataset<T...> base_type
```

```
typedef ContextT context_type
```

Public Functions

```
collector (context_type ctx, const std::string &name)
```

```
context_type &context ()
```

```
const context_type &context () const
```

Public Members

context_type **_context**

Template Struct combinator

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::junction< DependenciesT::result_type >` (*Template Struct junction*)

Struct Documentation

```
template<typename NextT, typename ...DependenciesT>
struct udho::activities::combinator : public udho::activities::junction<DependenciesT::result_type>
    A combinator combines multiple activities and proceeds towards the next activity
```

Template Parameters

- `NextT`: next activity
- `DependenciesT`: dependencies

Public Types

```
typedef std::shared_ptr<NextT> next_type
typedef boost::signals2::signal<void (NextT&)> signal_type
```

Public Functions

```
combinator (next_type &next)
```

```
template<typename U>
```

```
void operator() (const U &u)
```

whenever a subtask finishes the `operator()` of the combinator is called. which doesn't start the next subtask untill all the dependencies have completed. Before starting the next activity the next activity is prepared if any preparator is passed through the `prepare()` function

```
void cancel()
```

```
void propagate()
```

```
template<typename PreparatorT>
```

```
void prepare (PreparatorT prep)
```

set a preparator callback which will be called with a reference to teh next activity. The preparator callback is supposed to prepare the next activity by using the data collected till that time.

Public Members

next_type **_next**
 std::atomic<std::size_t> **_counter**
 std::mutex **_mutex**
signal_type **_preparators**
 std::atomic<bool> **_canceled**

Template Struct combinator< NextT, start< ContextT, T... > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename NextT, typename ContextT, typename ...T>
struct udho::activities::combinator<NextT, start<ContextT, T...>>
```

Public Types

```
typedef std::shared_ptr<NextT> next_type
```

Public Functions

```
combinator (next_type &next)  

void operator () ()  

void propagate ()
```

Public Members

next_type **_next**

Template Struct dataset

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- public udho::activities::fixed_key_accessor< udho::cache::shadow< std::string, T::result_type... > > (*Template Struct fixed_key_accessor*)

Struct Documentation

template<typename ...**T**>

struct udho::activities::dataset : public udho::activities::fixed_key_accessor<udho::cache::shadow<std::string, *T::result_type*...>>

Public Types

typedef *fixed_key_accessor*<udho::cache::shadow<std::string, **typename** *T::result_type*...>> **base_type**

typedef udho::cache::store<udho::cache::storage::memory, std::string, **typename** *T::result_type*...> **store_type**

typedef *store_type*::shadow_type **shadow_type**

typedef *accessor*<*T*...> **accessor_type**

Public Functions

dataset (const udho::configuration_type &config, const std::string &name)

std::string name () const

shadow_type &shadow ()

const *shadow_type* &shadow () const

Public Members

store_type _store

shadow_type _shadow

std::string _name

Template Struct after

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h

Inheritance Relationships

Derived Type

- public udho::activities::after< HeadT, TailT > (*Template Struct after*)

Struct Documentation

template<typename **T**>

struct after

Subclassed by *udho::activities::after*< HeadT, TailT >

Template Struct after< subtask< ActivityT, DependenciesT... > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ActivityT, typename ...DependenciesT>
struct udho::activities::detail::after<subtask<ActivityT, DependenciesT...>>
```

Public Functions

after (*subtask*<*ActivityT*, *DependenciesT*...> &*before*)

```
template<typename OtherActivityT, typename ...OtherDependenciesT>
void attach (subtask<OtherActivityT, OtherDependenciesT...> &sub)
```

Public Members

subtask<*ActivityT*, *DependenciesT*...> &**_before**

Template Struct after< udho::start_< ContextT, T... > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ContextT, typename ...T>
struct udho::activities::detail::after<udho::start_<ContextT, T...>>
```

Public Functions

after (udho::start_<*ContextT*, *T*...> &*before*)

```
template<typename OtherActivityT, typename ...OtherDependenciesT>
void attach (subtask<OtherActivityT, OtherDependenciesT...> &sub)
```

Public Members

udho::start_<*ContextT*, *T*...> &**_before**

Template Struct `apply_helper_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Derived Types

- `public udho::activities::apply_helper< FunctorT, SuccessT, FailureT > (Template Struct apply_helper)`
- `public udho::activities::apply_helper< FunctorT, SuccessT, FailureT > (Template Struct apply_helper)`
- `public udho::activities::apply_helper< FunctorT, SuccessT, FailureT > (Template Struct apply_helper)`
- `public udho::activities::apply_helper< FunctorT, SuccessT, FailureT > (Template Struct apply_helper)`

Struct Documentation

```
template<bool invocable, typename FunctorT, typename ...TargetsT>
struct udho::activities::detail::apply_helper_
    Subclassed by udho::activities::apply_helper< FunctorT, SuccessT, FailureT >,
    udho::activities::apply_helper< FunctorT, SuccessT, FailureT >, udho::activities::apply_helper< FunctorT, SuccessT, FailureT >
```

Public Functions

```
void operator () (FunctorT &f, const TargetsT&... t)
```

Template Struct `apply_helper_< false, FunctorT, TargetsT... >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename FunctorT, typename ...TargetsT>
struct udho::activities::detail::apply_helper_<false, FunctorT, TargetsT...>
```

Public Functions

void **operator** () (*FunctorT*&, const *TargetsT*&...)

Template Struct final_intermediate

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h

Struct Documentation

```
template<typename CollectorT, typename ...DependenciesT>
struct udho::activities::detail::final_intermediate
```

Public Functions

```
final_intermediate (std::shared_ptr<CollectorT> collector)
template<typename CallbackT>
subtask<joined<CallbackT, CollectorT>, DependenciesT...> exec (CallbackT callback)
```

Public Members

```
std::shared_ptr<CollectorT> _collector
```

Template Struct fixed_key_accessor

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h

Struct Documentation

```
template<typename StoreT>
struct udho::activities::fixed_key_accessor
```

Public Types

```
typedef StoreT::key_type key_type
```

Public Functions

```
fixed_key_accessor (StoreT &store, const key_type &key)
std::string key () const
template<typename V>
bool exists () const
template<typename V>
V get () const
template<typename V>
```

```
V at ()  
template<typename V>  
void set (const V &value)  
std::size_t size () const
```

Public Members

```
StoreT &_shadow  
key_type _key
```

Template Struct joined

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename CallbackT, typename CollectorT>  
struct joined
```

Template Struct joined< CallbackT, activities::collector< ContextT, dataset< T... > > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename CallbackT, typename ...T, typename ContextT>  
struct udho::activities::joined<CallbackT, activities::collector<ContextT, dataset<T...>>>
```

Public Types

```
typedef activities::collector<ContextT, dataset<T...>> collector_type  
typedef collector_type::accessor_type accessor_type  
typedef CallbackT callback_type  
typedef joined<callback_type, activities::collector<ContextT, dataset<T...>>> self_type  
typedef int cancel_if_ftor
```

Public Functions

```
joined (std::shared_ptr<collector_type> collector, CallbackT callback)
void operator () ()
void cancel ()
template<typename U>
void cancel_if (U&)
```

Template Struct junction

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename DependencyT>
struct udho::activities::junction
```

Public Functions

```
void operator () (const DependencyT&)
```

Template Struct perform

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Nested Relationships

Nested Types

- *Template Struct perform::require*

Struct Documentation

```
template<typename ActivityT>
struct udho::activities::perform
    create a subtask to perform activity ActivityT
```

Template Parameters

- ActivityT: the activity to perform

Public Static Functions

```
template<typename ...U>  
subtask<ActivityT> with (U&&... u)  
    arguments for the activity constructor
```

```
template<typename ...DependenciesT>  
struct require  
    mention the activities that has to be performed before executing this subtask.
```

Template Parameters

- DependenciesT: dependencies

Public Static Functions

```
template<typename ...U>  
subtask<ActivityT, DependenciesT...> with (U&&... u)  
    arguments for the activity constructor
```

Template Struct `perform::require`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Nested Relationships

This struct is a nested type of *Template Struct perform*.

Struct Documentation

```
template<typename ...DependenciesT>  
struct udho::activities::perform::require  
    mention the activities that has to be performed before executing this subtask.
```

Template Parameters

- DependenciesT: dependencies

Public Static Functions

```
template<typename ...U>  
subtask<ActivityT, DependenciesT...> with (U&&... u)  
    arguments for the activity constructor
```

Template Struct require

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ...DependenciesT>
struct udho::activities::require
```

Public Static Functions

```
template<typename CollectorT>
detail::final_intermediate<CollectorT, DependenciesT...> with (std::shared_ptr<CollectorT> collector)
```

Template Struct result

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::result_data< SuccessT, FailureT > (Template Struct result_data)`

Struct Documentation

```
template<typename SuccessT, typename FailureT>
struct udho::activities::result : public udho::activities::result_data<SuccessT, FailureT>
    Completion handler for an activity.
```

Template Parameters

- SuccessT: success data associated with the activity
- FailureT: failure data associated with the activity

Public Types

```
typedef result_data<SuccessT, FailureT> data_type
typedef accessor<data_type> accessor_type
typedef data_type::success_type success_type
typedef data_type::failure_type failure_type
typedef boost::signals2::signal<void (const data_type&) > signal_type
typedef boost::signals2::signal<void () > cancelation_signal_type
typedef boost::function<bool (const success_type&) > cancel_if_for
```

```
typedef boost::function<bool (const success_type&)> abort_error_ftor
typedef boost::function<bool (const failure_type&)> abort_failure_ftor
```

Public Functions

```
template<typename StoreT>
result (StoreT &store)
```

Parameters

- store: collector

```
template<typename CombinatorT>
void done (CombinatorT cmb)
```

attach another subtask as done callback which will be executed once this subtask finishes

Parameters

- cmb: next subtask

```
void required (bool flag)
```

mark the activity as required or optional

Parameters

- flag:

```
void cancel_if (cancel_if_ftor f)
```

Force cancelation of the activity even after it is successful to stop propagating to the next activities

Parameters

- f: callback which should return true to signal cancelation

```
void if_errored (abort_error_ftor ftor)
```

```
void if_failed (abort_failure_ftor ftor)
```

Public Members

```
accessor_type _shadow
```

```
signal_type _signal
```

```
bool _required
```

```
cancelation_signal_type _cancelation_signals
```

```
cancel_if_ftor _cancel_if
```

```
abort_error_ftor _abort_error
```

```
abort_failure_ftor _abort_failure
```

Protected Functions

void **success** (const *success_type* &data)

signal successful completion of the activity with success data of type SuccessT

Parameters

- data: success data

void **failure** (const *failure_type* &data)

signal failed completion of the activity with failure data of type FailureT

Parameters

- data: failure data

Friends

friend struct combinator

Template Struct *result_data*

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Derived Type

- `public udho::activities::result< SuccessT, FailureT >` (*Template Struct result*)

Struct Documentation

template<typename **SuccessT**, typename **FailureT**>

struct udho::activities::**result_data**

Contains **Copiable** Success or Failure data for an activity.

Template Parameters

- SuccessT: success data type
- FailureT: failure data type

Subclassed by `udho::activities::result< SuccessT, FailureT >`

Public Types

typedef *SuccessT* **success_type**

typedef *FailureT* **failure_type**

typedef *result_data<SuccessT, FailureT>* **self_type**

typedef *self_type* **result_type**

Public Functions

result_data()

bool **completed()** **const**
either success or failure data set

bool **failed()** **const**
whether the activity has failed

bool **canceled()** **const**
check whether the activity has been canceled

bool **okay()** **const**

const *success_type* &**success_data()** **const**
Success data

const *failure_type* &**failure_data()** **const**
Failure data

template<typename **CallableT**>

void **apply**(*CallableT* callback)

Apply a callable to the result data which will be invoked exactly once with appropriate arguments. The invocation of the callback depends on the state as shown below.

- Incomplete: callback()
- Canceled: callback(const SuccessT&, const FailureT&)
- Failed: callback(const FailureT&)
- Successful: callback(const SuccessT&)

The callback may not have all the overloads.

Parameters

- callback:

Public Members

bool **_completed**

bool **_success**

bool **_canceled**

success_type **_sdata**

failure_type **_fdata**

Protected Functions

```
void success (const success_type &data)
    Set Success Data

void failure (const failure_type &data)
    Set Failure Data

void cancel ()
    mark as canceled
```

Template Struct start

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ContextT, typename ...T>
struct udho::activities::start
```

Public Types

```
typedef activities::collector<ContextT, activities::dataset<T...>> collector_type
typedef activities::accessor<T...> accessor_type
typedef std::shared_ptr<collector_type> collector_ptr
typedef boost::signals2::signal<void () > signal_type
```

Public Functions

```
start (ContextT &ctx, const std::string &name)
collector_ptr collector () const
const accessor_type &accessor () const
accessor_type &accessor ()
void operator () ()
template<typename CombinatorT>
void done (CombinatorT cmb)
void success ()
```

Public Members

signal_type **_signal**

collector_ptr **_collector**

accessor_type **_accessor**

Template Struct subtask

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

template<typename **ActivityT**, typename ...**DependenciesT**>

struct udho::activities::subtask

A subtask is an instantiation of an activity. The subtask reuses an activity to model different use cases by attaching dependencies. A subtask contains two shared pointers, one to the activity and another one to the combinator. The subtask cannot be instantiated directly by calling the subtask constructor. Instead call the static `with` method to instantiate.

Template Parameters

- **ActivityT**: The activity
- **DependenciesT**: The activities that has to be performed before performing **ActivityT**

Public Types

typedef *ActivityT* **activity_type**

typedef *combinator<ActivityT, DependenciesT...>* **combinator_type**

typedef *subtask<ActivityT, DependenciesT...>* **self_type**

Public Functions

subtask (**const** *self_type* &*other*)

`std::shared_ptr<activity_type>` **activity** ()

shared pointer to the activity

template<typename **V**, typename ...**DependenciesV**>

self_type &**done** (*subtask<V, DependenciesV...>* &*next*)

execute task next after the current one

Parameters

- *next*: the next subtask

template<typename **V**, typename ...**DependenciesV**>

self_type &**after** (*subtask<V, DependenciesV...>* &*previous*)

t2.after(t1) is equivalent to *t1.done(t2)*

Parameters

- *previous*: the previous subtask

```
template<typename PreparatorT>
self_type &prepare (PreparatorT prep)
    attach a callback which will be called with a reference to the activity after it has been instantiated and all
    its dependencies have completed.

self_type &required (bool flag)
    Set required flag on or off. If a required subtask fails then all intermediate subtask that depend on it fails
    and the final callback is called immediately. By default all subtasks are required

self_type &cancel_if (typename activity_type::cancel_if_ftor cancelor)
    Force cancelation of the activity even after it is successful to stop propagating to the next activities
```

Parameters

- *f*: callback which should return true to signal cancelation

```
std::shared_ptr<activity_type> operator-- ()
    returns the shared pointer to the actiivity
```

```
template<typename FunctionT>
self_type &if_errored (FunctionT ffor)
    abort if canceled if ffor returns false. f will be called with the success if it has been canceled due to error
```

```
template<typename FunctionT>
self_type &if_failed (FunctionT ffor)
    abort if canceled if ffor returns false. f will be called with the failue data if it has been canceled due to
    failure
```

```
template<typename FunctionT>
self_type &if_canceled (FunctionT ffor)
```

Public Static Functions

```
template<typename CollectorT, typename ...U>
self_type with (CollectorT collector, U&&... u)
    Arguments for the constructor of the Activity
```

Friends

```
friend struct subtask
```

Template Struct subtask< ActivityT >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ActivityT>
struct udho::activities::subtask<ActivityT>
    Spetialization for the root subtask in the task graph
```

Template Parameters

- *ActivityT*: The activity

Public Types

```
typedef ActivityT activity_type  
typedef subtask<ActivityT> self_type
```

Public Functions

```
subtask (const self_type &other)  
  
std::shared_ptr<activity_type> activity ()  
  
template<typename V, typename ...DependenciesV>  
self_type &done (subtask<V, DependenciesV...> &next)  
    execute task next after the current one
```

Parameters

- *next*: the next subtask

```
template<typename ...U>  
void operator () (U&&... u)  
    calls the operator () () of the activity and starts executing the graph
```

```
self_type &required (bool flag)  
    Set required flag on or off. If a required subtask fails then all intermediate subtask that depend on it fails  
    and the final callback is called immediately. By default all subtasks are required
```

```
self_type &cancel_if (typename activity_type::cancel_if_ftor cancelor)  
    Force cancelation of the activity even after it is successful to stop propagating to the next activities
```

Parameters

- *f*: callback which should return true to signal cancelation

```
std::shared_ptr<activity_type> operator-- ()  
    returns the shared pointer to the actiivity
```

```
template<typename FunctionT>  
self_type &if_errored (FunctionT ftor)  
    abort if canceled if ftor returns false. f will be called with the success if it has been canceled due to error
```

```
template<typename FunctionT>  
self_type &if_failed (FunctionT ftor)  
    abort if canceled if ftor returns false. f will be called with the failue data if it has been canceled due to  
    failure
```

```
template<typename FunctionT>  
self_type &if_canceled (FunctionT ftor)
```

Public Static Functions

```
template<typename CollectorT, typename ...U>
self_type with (CollectorT collector, U&&... u)
    Arguments for the constructor of the Activity
```

Friends

```
friend struct subtask
```

Template Struct subtask< start< ContextT, T... > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ContextT, typename ...T>
struct udho::activities::subtask<start<ContextT, T...>>
```

Public Types

```
typedef start<ContextT, T...> activity_type
typedef subtask<start<ContextT, T...>> self_type
```

Public Functions

```
subtask (const self_type &other)

std::shared_ptr<activity_type> activity ()

template<typename V, typename ...DependenciesV>
self_type &done (subtask<V, DependenciesV...> &next)
    execute task next after the current one
```

Parameters

- *next*: the next subtask

```
void operator () ()
    calls the operator () () of the activity and starts executing the graph

std::shared_ptr<activity_type> operator-- ()
    returns the shared pointer to the activity
```

Public Static Functions

self_type **with** (*ContextT* *ctx*, **const** std::string &*name* = "")
Arguments for the constructor of the Activity

Protected Functions

subtask (*ContextT* *ctx*, **const** std::string &*name* = "")

Protected Attributes

std::shared_ptr<*activity_type*> **_activity**

Friends

friend struct subtask

Template Struct **app_**

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h

Struct Documentation

template<typename **AppT**, bool **Ref** = false>
struct udho::app_

Public Types

typedef app_<*AppT*, *Ref*> **self_type**
typedef *self_type* **application_type**

Public Functions

template<typename ...**T**>
app_ (*T*... *args*)
std::string **name** () **const**
self_type &**operator=** (**const** std::string &*path*)
template<typename **ContextT**, typename **Lambda**>
int **serve** (*ContextT* &*ctx*, boost::beast::http::verb *request_method*, **const** std::string &*subject*, *Lambda* *send*)
void **summary** (std::vector<*module_info*> &*stack*) **const**
template<typename **F**>
void **eval** (*F* &*fnc*)

Public Members

std::string **_path**

AppT **_app**

Template Struct `app_< AppT, true >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h`

Struct Documentation

```
template<typename AppT>
struct udho::app_<AppT, true>
```

Public Types

```
typedef app_<AppT, true> self_type
```

```
typedef self_type application_type
```

Public Functions

```
app_ (AppT &app)
```

```
std::string name () const
```

```
self_type &operator= (const std::string &path)
```

```
template<typename ContextT, typename Lambda>
```

```
int serve (ContextT &ctx, boost::beast::http::verb request_method, const std::string &subject, Lambda send)
```

```
void summary (std::vector<module_info> &stack) const
```

```
template<typename F>
```

```
void eval (F &fnc)
```

Public Members

std::string **_path**

AppT &**_app**

Struct `assoc`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Inheritance Relationships

Base Type

- `public udho::detail::association_group< detail::association_leaf, void >`
(*Template Struct association_group*)

Struct Documentation

```
struct udho::assoc : public udho::detail::association_group<detail::association_leaf, void>
```

Public Functions

```
assoc()
```

Template Struct attachment

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_attachment.h`

Inheritance Relationships

Base Types

- `public AuxT`
- `public CacheT`
- `public LoggerT`

Struct Documentation

```
template<typename AuxT, typename LoggerT = void, typename CacheT = void>  
struct udho::attachment : public AuxT, public CacheT, public LoggerT  
    logged stateful
```

Public Types

```
typedef attachment<AuxT, LoggerT, CacheT> self_type  
typedef AuxT auxiliary_type  
typedef LoggerT logger_type  
typedef CacheT cache_type  
typedef auxiliary_type::configuration_type configuration_type  
typedef cache_type::shadow_type shadow_type
```

Public Functions

attachment (boost::asio::io_service &*io*, *LoggerT* &*logger*)

shadow_type &**shadow** ()

AuxT &**aux** ()

boost::asio::io_service &**io** ()

Public Members

boost::asio::io_service &**_io**

shadow_type **_shadow**

Template Struct attachment< AuxT, LoggerT, void >

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_attachment.h

Inheritance Relationships

Base Types

- public AuxT
- public LoggerT

Struct Documentation

template<typename **AuxT**, typename **LoggerT**>

struct udho::attachment<*AuxT*, *LoggerT*, void> : **public** *AuxT*, **public** *LoggerT*
logged stateless

Public Types

typedef attachment<*AuxT*, *LoggerT*, void> **self_type**

typedef *AuxT* **auxiliary_type**

typedef *LoggerT* **logger_type**

typedef auxiliary_type::configuration_type **configuration_type**

typedef void **shadow_type**

Public Functions

attachment (boost::asio::io_service &*io*, *LoggerT* &*logger*)

int **shadow** ()

AuxT &**aux** ()

boost::asio::io_service &**io** ()

Public Members

boost::asio::io_service &**_io**

Template Struct attachment< AuxT, void, CacheT >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_attachment.h`

Inheritance Relationships

Base Types

- public AuxT
- public CacheT

Struct Documentation

template<typename **AuxT**, typename **CacheT**>

struct udho::attachment<*AuxT*, void, *CacheT*> : public *AuxT*, public *CacheT*
quiet stateful

Public Types

typedef attachment<*AuxT*, void, *CacheT*> **self_type**

typedef *AuxT* **auxiliary_type**

typedef void **logger_type**

typedef *CacheT* **cache_type**

typedef auxiliary_type::configuration_type **configuration_type**

typedef cache_type::shadow_type **shadow_type**

Public Functions

```

attachment (boost::asio::io_service &io)
template<udho::logging::status Status>
self_type &operator () (const udho::logging::message<Status>&)
shadow_type &shadow ()
AuxT &aux ()
boost::asio::io_service &io ()

```

Public Members

```

boost::asio::io_service &_io
shadow_type _shadow

```

Template Struct attachment< AuxT, void, void >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_attachment.h`

Inheritance Relationships

Base Type

- `public AuxT`

Struct Documentation

```

template<typename AuxT>
struct udho::attachment<AuxT, void, void> : public AuxT
    quiet stateless

```

Public Types

```

typedef attachment<AuxT, void, void> self_type
typedef AuxT auxiliary_type
typedef void logger_type
typedef void cache_type
typedef void shadow_type
typedef auxiliary_type::configuration_type configuration_type

```

Public Functions

```
attachment (boost::asio::io_service &io)  
template<udho::logging::status Status>  
self_type &operator () (const udho::logging::message<Status>&)  
int shadow ()  
AuxT &aux ()  
boost::asio::io_service &io ()
```

Public Members

```
boost::asio::io_service &_io
```

Template Struct basic_router

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Inheritance Relationships

Base Type

- `public udho::overload_group< void, overload_terminal< AuxT > >` (*Template Struct overload_group*)

Struct Documentation

```
template<typename AuxT = void>  
struct udho::basic_router : public udho::overload_group<void, overload_terminal<AuxT>>  
    router maps HTTP requests with the callbacks
```

```
auto router = udho::router()  
| (udho::get(add).plain()    = "^/add/ (\\d+) / (\\d+) $")  
| (udho::get(hello).plain() = "^/hello$");
```

Public Types

```
typedef overload_group<void, overload_terminal<AuxT>> base_type
```

Public Functions

```
template<typename ...Args>
basic_router (Args... args)
    constructs a router
```

Friends

```
template<typename U, typename V, typename F>
friend overload_group<overload_group<U, V>, F> operator| (const overload_group<U, V>
                                                         &group, const F &method)
```

adds a callback url mapping to the router by conjugating an overload in the overload meta-chain. The url mapping is provided by content wrappers like *content_wrapper0* *content_wrapper1* which attaches feasibility criteria like http resource path, http verb with a callback function. The url mapping can be built using get post put head etc.. See router for an example

Parameters

- *group*: the overload group (which is actually the router or router attached with some url mappings)
- *method*: url mapping

Template Struct bounded_str

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename Iterator>
struct udho::bounded_str
```

Public Types

```
typedef bounded_str<Iterator> self_type
typedef std::pair<Iterator, Iterator> pair_type
```

Public Functions

```
bounded_str (Iterator begin, Iterator end)
bounded_str (const self_type &other)
self_type &operator= (const self_type &other)
Iterator begin () const
Iterator end () const
bool invalid () const
bool valid () const
template<typename Iterator>
```

```
void copy (Iterator it) const
template<typename ContainerT>
ContainerT copied () const
std::size_t size () const
```

Public Members

pair_type _pair

Template Struct bridge

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_bridge.h`

Struct Documentation

```
template<typename ConfigT>
struct udho::bridge
```

Todo:
write docs

Public Types

```
typedef ConfigT configuration_type
typedef bridge<ConfigT> self_type
```

Public Functions

```
bridge (boost::asio::io_service &io)
configuration_type &config ()
const configuration_type &config () const
boost::filesystem::path docroot () const
boost::filesystem::path tmproot () const
std::string contents (const boost::filesystem::path &local_path) const
boost::beast::http::response<boost::beast::http::file_body> file (const std::string &path, const
                                                                ::udho::defs::request_type &req,
                                                                std::string mime = "") const

template<typename RequestT, typename GroupT>
std::string render (const std::string &path, const udho::detail::context_common<self_type, Re-
                    questT> &ctx, ::udho::lookup_table<GroupT> &scope) const

template<typename RequestT, typename DataT>
std::string render (const std::string &path, const udho::detail::context_common<self_type, Re-
                    questT> &ctx, ::udho::prepared<DataT> &data) const

template<typename RequestT, typename U, typename V>
```

```

std::string render (const std::string &path, const udho::detail::context_common<self_type, Re-
    questT> &ctx, ::udho::prepared_group<U, V> &group) const
template<typename RequestT, typename ...DataT>
std::string render (const std::string &path, const udho::detail::context_common<self_type, Re-
    questT> &ctx, const DataT&... data) const
std::string render (const std::string &path) const
template<typename ContextT>
detail::client_connection_wrapper<ContextT> client (ContextT ctx, udho::config<udho::client_options>
    options)

```

Public Members

```

boost::asio::io_service &_io
configuration_type _config

```

Template Struct abstract_engine

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```

template<typename KeyT, typename ValueT = void>
struct udho::cache::abstract_engine

```

Public Types

```

typedef KeyT key_type
typedef ValueT value_type
typedef content<ValueT> content_type
typedef abstract_engine<KeyT, ValueT> self_type

```

Public Functions

```

std::size_t size () const = 0
bool exists (const key_type &key) const = 0
void create (const key_type &key, const content_type &content) = 0
content_type retrieve (const key_type &key) const = 0
bool update (const key_type &key, const content_type &content) = 0
bool remove (const key_type &key) = 0
boost::posix_time::ptime created (const key_type &key) const
boost::posix_time::ptime updated (const key_type &key) const
boost::posix_time::time_duration age (const key_type &key) const
boost::posix_time::time_duration idle (const key_type &key) const

```

```
template<typename U = value_type>
std::enable_if<!std::is_same<U, void>::value>::type insert (const key_type &key, const U
&value)

template<typename U = value_type>
std::enable_if<std::is_same<U, void>::value>::type insert (const key_type &key)

template<typename U = value_type>
std::enable_if<!std::is_same<U, void>::value, bool>::type update (const key_type &key, const U
&value)

bool update (const key_type &key)
```

Template Struct content

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<typename T = void>
struct udho::cache::content
```

Public Types

```
typedef T value_type
typedef content<T> self_type
```

Public Functions

```
content ()
content (const value_type &value)
boost::posix_time::ptime created () const
boost::posix_time::ptime updated () const
boost::posix_time::time_duration age () const
boost::posix_time::time_duration idle () const
const value_type &value () const
void update (const self_type &other)
void update (const value_type &value)
void update ()
template<class Archive>
void serialize (Archive &ar, const unsigned int version)
```

Public Members

```
boost::posix_time::ptime _created
boost::posix_time::ptime _updated
value_type _value
```

Friends

```
friend class boost::serialization::access
```

Template Struct content< void >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<>
struct udho::cache::content<void>
```

Public Types

```
typedef void value_type
typedef content self_type
```

Public Functions

```
content ()
template<typename U>
content (U)
boost::posix_time::ptime created () const
boost::posix_time::ptime updated () const
boost::posix_time::time_duration age () const
boost::posix_time::time_duration idle () const
void update ()
void update (const self_type&)
template<class Archive>
void serialize (Archive &ar, const unsigned int version)
```

Public Members

boost::posix_time::ptime **_created**

boost::posix_time::ptime **_updated**

Friends

friend class boost::serialization::access

Template Struct driver

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<typename KeyT, typename ValueT = void>
struct udho::cache::driver
```

Public Types

```
typedef KeyT key_type
```

```
typedef ValueT value_type
```

```
typedef content<ValueT> content_type
```

```
typedef abstract_engine<KeyT, ValueT> abstract_engine_type
```

Public Functions

```
driver (abstract_engine_type &e)
```

```
std::size_t size () const
```

```
bool exists (const key_type &key) const
```

```
template<typename U = value_type>
std::enable_if<!std::is_same<U, void>::value>::type insert (const key_type &key, const U
&value)
```

```
template<typename U = value_type>
std::enable_if<std::is_same<U, void>::value>::type insert (const key_type &key)
```

```
content_type retrieve (const key_type &key) const
```

```
bool update (const key_type &key, const content_type &content)
```

```
bool update (const key_type &key)
```

```
template<typename U = value_type>
std::enable_if<std::is_same<U, ValueT>::value, bool>::type update (const key_type &key, const U
&value)
```

```
bool remove (const key_type &key)
```

```
boost::posix_time::ptime created (const key_type &key) const
```

```
boost::posix_time::ptime updated (const key_type &key) const
boost::posix_time::time_duration age (const key_type &key) const
boost::posix_time::time_duration idle (const key_type &key) const
```

Public Members

```
abstract_engine_type &_engine
```

Template Struct engine

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Base Types

- private `StorageT`
- public `udho::cache::abstract_engine< StorageT::key_type, StorageT::value_type >` (*Template Struct abstract_engine*)

Struct Documentation

```
template<typename StorageT>
```

```
struct udho::cache::engine: private StorageT, public udho::cache::abstract_engine<StorageT::key_type, StorageT::
```

Public Types

```
typedef StorageT storage_type
```

```
typedef abstract_engine<typename StorageT::key_type, typename StorageT::value_type> abstract_engine_type
```

```
typedef StorageT::key_type key_type
```

```
typedef StorageT::value_type value_type
```

```
typedef StorageT::content_type content_type
```

```
typedef engine<StorageT> self_type
```

```
typedef StorageT::config_type config_type
```

Public Functions

```
engine (const config_type &config)
std::size_t size () const final
bool exists (const key_type &key) const final
void create (const key_type &key, const content_type &content) final
content_type retrieve (const key_type &key) const final
bool update (const key_type &key, const content_type &content) final
bool remove (const key_type &key) final
```

Template Struct flake

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Derived Type

- `public udho::cache::shadow< KeyT, T > (Template Struct shadow)`

Struct Documentation

```
template<typename KeyT, typename T>
struct udho::cache::flake
    copiable flake contains a reference to the actual registry object
    Subclassed by udho::cache::shadow< KeyT, T >
```

Public Types

```
typedef KeyT key_type
typedef T value_type
typedef registry<key_type, T> registry_type
typedef flake<KeyT, T> flake_type
typedef flake_type self_type
```

Public Functions

```
template<template<typename, typename> class StorageT, typename ...X>
flake (store<StorageT, KeyT, X...> &store)

bool exists (const key_type &key) const
value_type at (const key_type &key) const
void insert (const key_type &key, const value_type &value)
void update (const key_type &key, const value_type &value)
std::size_t size (const key_type &key) const
bool remove (const key_type &key)
boost::posix_time::ptime created (const key_type &key) const
boost::posix_time::ptime updated (const key_type &key) const
boost::posix_time::time_duration age (const key_type &key) const
boost::posix_time::time_duration idle (const key_type &key) const
```

Template Struct generator

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<typename KeyT>
struct generator
```

Template Struct generator< boost::uuids::uuid >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<>
struct udho::cache::generator<boost::uuids::uuid>
```

Public Functions

```
boost::uuids::uuid parse (const std::string &key)
boost::uuids::uuid generate ()
```

Template Struct master

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Base Type

- `public udho::cache::driver< KeyT >` (*Template Struct driver*)

Derived Type

- `protected udho::cache::store< StorageT, KeyT, T >` (*Template Struct store*)

Struct Documentation

```
template<typename KeyT>
struct udho::cache::master : public udho::cache::driver<KeyT>
    Subclassed by udho::cache::store< StorageT, KeyT, T >
```

Public Types

```
typedef KeyT key_type
typedef driver<KeyT> storage_type
typedef abstract_engine<KeyT> abstract_engine_type
typedef master<KeyT> self_type
```

Public Functions

```
master (abstract_engine_type &e)
master (const self_type&) = delete
master (self_type&&) = default
bool issued (const key_type &key) const
void issue (const key_type &key)
```

Template Struct registry

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Base Type

- `public udho::cache::driver< KeyT, T >` (*Template Struct driver*)

Derived Type

- `public udho::cache::store< StorageT, KeyT, T >` (*Template Struct store*)

Struct Documentation

```
template<typename KeyT, typename T>
struct udho::cache::registry : public udho::cache::driver<KeyT, T>
    Subclassed by udho::cache::store< StorageT, KeyT, T >
```

Public Types

```
typedef registry<KeyT, T> self_type
typedef KeyT key_type
typedef T value_type
typedef abstract_engine<KeyT, T> abstract_engine_type
```

Public Functions

```
registry (abstract_engine_type &e)
registry (const self_type&) = delete
registry (self_type&&) = default
value_type at (const key_type &key) const
```

Template Struct shadow

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Base Type

- `public udho::cache::flake< KeyT, T >` (*Template Struct flake*)

Struct Documentation

template<typename **KeyT**, typename ...**T**>
struct udho::cache::shadow: public udho::cache::flake<KeyT, T>
 copiable shadow of a store reference.

```

user u("Neel");
appearance a("red");
udho::cache::store<std::string, user, appearance> store;
udho::cache::shadow<std::string, user, appearance> shadow_ua(store);
shadow_ua.insert("x", u);
shadow_ua.insert("x", a);
std::cout << std::boolalpha << shadow_ua.exists<user>("x") << std::endl; // true
std::cout << std::boolalpha << shadow_ua.exists<appearance>("x") << std::endl; // 
↪true
std::cout << shadow_ua.get<user>("x").name << std::endl; // Neel
std::cout << shadow_ua.get<appearance>("x").color << std::endl; // red
udho::cache::shadow<std::string, user> shadow_u(store);
std::cout << std::boolalpha << shadow_u.exists<user>("x") << std::endl; // true
std::cout << std::boolalpha << shadow_u.get<user>("x").name << std::endl; // true
std::cout << shadow_u.exists<appearance>("x") << std::endl; // won't compile
udho::cache::shadow<std::string, user> shadow_u2(shadow_u); // copiable
udho::cache::shadow<std::string, user> shadow_u3(shadow_ua); // narrowable
udho::cache::shadow<std::string, appearance> shadow_a(store); // unordered
std::cout << shadow_a.exists<appearance>("x") << std::endl; // true
std::cout << shadow_a.get<appearance>("x").color << std::endl; // red
udho::cache::shadow<std::string> shadow_none(shadow_ua);

```

Public Types

```

typedef KeyT key_type
typedef shadow<key_type, T...> self_type
typedef self_type shadow_type
typedef master<key_type> master_type

```

Public Functions

```

template<template<typename, typename> class StorageT, typename ...X>
shadow (store<StorageT, key_type, X...> &store)

template<typename ...X>
shadow (shadow<key_type, X...> &other)

bool issued (const key_type &key) const

void issue (const key_type &key)

template<typename V>
bool exists (const key_type &key) const

template<typename V>
V at (const key_type &key) const

template<typename V>
V get (const key_type &key, const V &def = V()) const

```

```

template<typename V>
void set (const key_type &key, const V &value)

template<typename V>
void insert (const key_type &key, const V &value)

std::size_t size () const

template<typename V>
std::size_t size (const key_type &key) const

bool remove (const key_type &key)

template<typename V>
bool remove (const key_type &key)

boost::posix_time::ptime created (const key_type &key) const

boost::posix_time::ptime updated (const key_type &key) const

boost::posix_time::time_duration age (const key_type &key) const

boost::posix_time::time_duration idle (const key_type &key) const

template<typename V>
boost::posix_time::ptime created (const key_type &key) const

template<typename V>
boost::posix_time::ptime updated (const key_type &key) const

template<typename V>
boost::posix_time::time_duration age (const key_type &key) const

template<typename V>
boost::posix_time::time_duration idle (const key_type &key) const

```

Public Members

master_type & **_master**

Template Struct disk

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```

template<typename KeyT, typename ValueT = void>
struct udho::cache::storage::disk

```

Public Types

```
typedef KeyT key_type
typedef ValueT value_type
typedef content<ValueT> content_type
typedef std::map<key_type, content_type> map_type
typedef storage::disk<KeyT, ValueT> self_type
typedef udho::config<udho::configs::session> session_config_type
typedef session_config_type config_type
```

Public Functions

```
disk (const session_config_type &config)
disk (const self_type&) = delete
disk (self_type&&) = default
std::size_t size () const
bool exists (const key_type &key) const
void create (const key_type &key, const content_type &content)
content_type retrieve (const key_type &key) const
bool update (const key_type &key, const content_type &content)
bool remove (const key_type &key)
```

Protected Functions

```
boost::filesystem::path storage () const
udho::configs::session::format format () const
std::string extension () const
std::string filename (const key_type &key) const
boost::filesystem::path path (const key_type &key) const
void save (udho::configs::session::format format, std::ofstream &file, const content_type &content)
void load (udho::configs::session::format format, std::ifstream &file, content_type &content) const
void save (std::ofstream &file, const content_type &content)
void load (std::ifstream &file, content_type &content) const
```

Protected Attributes

```
const session_config_type &_config
std::string _name
boost::mutex _mutex
```

Template Struct memory

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
template<typename KeyT, typename ValueT = void>
struct udho::cache::storage::memory
```

Public Types

```
typedef KeyT key_type
typedef ValueT value_type
typedef content<ValueT> content_type
typedef std::map<key_type, content_type> map_type
typedef storage::memory<KeyT, ValueT> self_type
typedef udho::config<udho::configs::session> session_config_type
typedef session_config_type config_type
```

Public Functions

```
memory (const session_config_type &config)
memory (const self_type&) = delete
memory (self_type&&) = default
std::size_t size () const
bool exists (const key_type &key) const
void create (const key_type &key, const content_type &content)
content_type retrieve (const key_type &key) const
bool update (const key_type &key, const content_type &content)
bool remove (const key_type &key)
```

Protected Attributes

```
const session_config_type &_config  
map_type _storage  
boost::mutex _mutex
```

Struct redis

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Struct Documentation

```
struct redis
```

Template Struct store

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cache.h`

Inheritance Relationships

Base Types

- `private udho::cache::engine< StorageT< KeyT, void > >` (*Template Struct engine*)
- `private udho::cache::engine< StorageT< KeyT, T > >` (*Template Struct engine*)
- `protected udho::cache::master< KeyT >` (*Template Struct master*)
- `public udho::cache::registry< KeyT, T >` (*Template Struct registry*)

Struct Documentation

`template<template<typename, typename> class StorageT, typename KeyT, typename ...T>`

struct `udho::cache::store: private udho::cache::engine<StorageT<KeyT, void>>, private udho::cache::engine<StorageT<KeyT, T>>`

A type safe non-copiable storage

```
user u("Neel");  
appearance a("red");  
udho::cache::store<std::string, user, appearance> store;  
store.insert("x", u);  
store.insert("x", a);  
store.exists<user>();
```

Public Types

```
typedef KeyT key_type
typedef master<KeyT> master_type
typedef shadow<KeyT, T...> shadow_type
typedef store<StorageT, KeyT, T...> self_type
typedef engine<StorageT<KeyT, void>>::config_type config_type
```

Public Functions

```
store (const config_type &config)
store (const self_type&) = delete
template<typename V>
bool exists (const key_type &key) const
template<typename V>
V get (const key_type &key, const V &def = V()) const
template<typename V>
V at (const key_type &key)
template<typename V>
void insert (const key_type &key, const V &value)
template<typename V>
std::size_t size (const key_type &key) const
template<typename V>
bool remove (const key_type &key)
boost::posix_time::ptime created (const key_type &key) const
boost::posix_time::ptime updated (const key_type &key) const
boost::posix_time::time_duration age (const key_type &key) const
boost::posix_time::time_duration idle (const key_type &key) const
```

Friends

```
friend struct shadow< KeyT, T... >
```

Template Struct *client_options_*

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Nested Relationships

Nested Types

- *Struct client_options_::follow_redirect_t*
- *Struct client_options_::http_version_t*
- *Struct client_options_::verify_certificate_t*

Inheritance Relationships

Derived Type

- `public udho::config< udho::client_options >` (*Template Struct config*)

Struct Documentation

template<typename **T** = void>
struct udho::client_options_
 Subclassed by *udho::config< udho::client_options >*

Public Functions

```
client_options_()  
void set (verify_certificate_t, bool v)  
bool get (verify_certificate_t) const  
void set (follow_redirect_t, bool v)  
bool get (follow_redirect_t) const  
void set (http_version_t, int v)  
int get (http_version_t) const
```

Public Members

```
bool _verify_certificate  
bool _follow_redirect  
int _http_version
```

Public Static Attributes

```
const struct udho::client_options_::verify_certificate_t verify_certificate
const struct udho::client_options_::follow_redirect_t follow_redirect
const struct udho::client_options_::http_version_t http_version
struct follow_redirect_t
```

Public Types

```
typedef client_options_<T> component
struct http_version_t
```

Public Types

```
typedef client_options_<T> component
struct verify_certificate_t
```

Public Types

```
typedef client_options_<T> component
```

Struct **client_options_::follow_redirect_t**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Nested Relationships

This struct is a nested type of *Template Struct client_options_*.

Struct Documentation

```
struct udho::client_options_::follow_redirect_t
```

Public Types

```
typedef client_options_<T> component
```

Struct `client_options_::http_version_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Nested Relationships

This struct is a nested type of *Template Struct `client_options_`*.

Struct Documentation

```
struct udho::client_options_::http_version_t
```

Public Types

```
typedef client_options_<T> component
```

Struct `client_options_::verify_certificate_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Nested Relationships

This struct is a nested type of *Template Struct `client_options_`*.

Struct Documentation

```
struct udho::client_options_::verify_certificate_t
```

Public Types

```
typedef client_options_<T> component
```

Template Struct deferred

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_compositors.h`

Struct Documentation

```
template<typename OutputT>
```

```
struct udho::compositors::deferred
```

No output will be returned. `ctx.respond(...)` will be used instead

See *`udho::context::respond`*

Public Types

```
typedef OutputT response_type
```

Public Functions

```
template<typename ...T>
void operator() (T...)

std::string name() const
```

Template Struct mimed

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_compositors.h`

Struct Documentation

```
template<typename OutputT>
struct udho::compositors::mimed
    mimed content. The returned output will be sent with the given mime type
```

Public Types

```
typedef boost::beast::http::response<boost::beast::http::string_body> response_type
```

Public Functions

```
mimed(const std::string &mime)

template<typename ContextT>
response_type operator() (const ContextT &ctx, const OutputT &out)

std::string name() const
```

Public Members

```
std::string _mime
```

Template Struct transparent

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_compositors.h`

Struct Documentation

```
template<typename OutputT>
struct udho::compositors::transparent
    raw boost::beast HTTP response content
```

Public Types

```
typedef OutputT response_type
```

Public Functions

```
template<typename ContextT>
response_type &&operator() (const ContextT&, OutputT &&out)

std::string name() const
```

Template Struct config

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Inheritance Relationships

Base Type

- `public T`

Derived Types

- `public udho::configuration< detail::url_data > (Template Struct configuration)`
- `public udho::configuration< T > (Template Struct configuration)`

Struct Documentation

```
template<typename T>
struct udho::config: public T
    Subclassed by udho::configuration< detail::url_data >, udho::configuration< T >
```

Public Types

```
typedef config<T> self_type
```

Public Functions

```
template<typename K>
auto operator[] (const K&) const

template<typename K>
proxy<K, self_type> operator[] (const K&)
```

Template Struct `config< configs::server_<> >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Inheritance Relationships

Base Type

- `public udho::configs::server_<>` (*Template Struct server_*)

Struct Documentation

```
template<>
struct udho::config<configs::server_<>> : public udho::configs::server_<>
```

Public Types

```
typedef config<configs::server_<>> self_type
```

Public Functions

```
template<typename K>
auto operator[] (const K&) const

template<typename K>
proxy<K, self_type> operator[] (const K&)

void mime (const std::string &extension, const std::string &type)
```

Template Struct **form_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Struct Documentation

```
template<typename T = void>  
struct form_
```

Template Struct **logger_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Struct Documentation

```
template<typename T = void>  
struct logger_
```

Template Struct **router_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Struct Documentation

```
template<typename T = void>  
struct router_
```

Template Struct **server_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

Nested Types

- *Struct `server_::document_root_t`*
- *Struct `server_::mime_default_t`*
- *Struct `server_::mimes_t`*
- *Struct `server_::template_root_t`*

Struct Documentation

```
template<typename T = void>
struct udho::configs::server_
```

Public Types

```
typedef std::map<std::string, std::string> mime_map
```

Public Functions

```
server_()
void set (document_root_t, const boost::filesystem::path &v)
boost::filesystem::path get (document_root_t) const
void set (template_root_t, const boost::filesystem::path &v)
boost::filesystem::path get (template_root_t) const
void set (mime_default_t, std::string v)
std::string get (mime_default_t) const
const mime_map &get (mimes_t) const
std::string mime (const std::string &extension) const
void mime (const std::string &extension, const std::string &type)
```

Public Members

```
boost::filesystem::path _document_root
boost::filesystem::path _template_root
std::string _mime_default
mime_map _mimes
```

Public Static Attributes

```
const struct udho::configs::server_::document_root_t document_root
const struct udho::configs::server_::template_root_t template_root
const struct udho::configs::server_::mime_default_t mime_default
const struct udho::configs::server_::mimes_t mimes
struct document_root_t
```

Public Types

```
typedef server_<T> component
struct mime_default_t
```

Public Types

```
typedef server_<T> component
struct mimes_t
```

Public Types

```
typedef server_<T> component
struct template_root_t
```

Public Types

```
typedef server_<T> component
```

Struct **server_::document_root_t**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

This struct is a nested type of *Template Struct server_*.

Struct Documentation

```
struct udho::configs::server_::document_root_t
```

Public Types

```
typedef server_<T> component
```

Struct **server_::mime_default_t**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

This struct is a nested type of *Template Struct server_*.

Struct Documentation

```
struct udho::configs::server_::mime_default_t
```

Public Types

```
typedef server_<T> component
```

Struct **server_::mimes_t**

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h

Nested Relationships

This struct is a nested type of *Template Struct server_*.

Struct Documentation

```
struct udho::configs::server_::mimes_t
```

Public Types

```
typedef server_<T> component
```

Struct **server_::template_root_t**

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h

Nested Relationships

This struct is a nested type of *Template Struct server_*.

Struct Documentation

```
struct udho::configs::server_::template_root_t
```

Public Types

```
typedef server_<T> component
```

Template Struct `session_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

Nested Types

- *Struct `session_::extension_t`*
- *Struct `session_::id_t`*
- *Struct `session_::path_t`*
- *Struct `session_::serialization_t`*

Struct Documentation

```
template<typename T = void>  
struct udho::configs::session_
```

Public Types

```
enum format  
    Values:  
  
    enumerator text  
    enumerator binary  
    enumerator xml
```

Public Functions

```
session_()  
  
void set (extension_t, const std::string &e)  
std::string get (extension_t) const  
  
void set (serialization_t, format v)  
format get (serialization_t) const  
  
void set (path_t, const boost::filesystem::path &p)  
boost::filesystem::path get (path_t) const  
  
void set (id_t, std::string v)  
std::string get (id_t) const
```

Public Members

```
std::string _extension
format _serialization
boost::filesystem::path _path
std::string _id
```

Public Static Attributes

```
const struct udho::configs::session_::extension_t extension
const struct udho::configs::session_::serialization_t serialization
const struct udho::configs::session_::path_t path
const struct udho::configs::session_::id_t id
struct extension_t
```

Public Types

```
typedef session_<T> component
struct id_t
```

Public Types

```
typedef session_<T> component
struct path_t
```

Public Types

```
typedef session_<T> component
struct serialization_t
```

Public Types

```
typedef session_<T> component
```

Struct session_::extension_t

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h

Nested Relationships

This struct is a nested type of *Template Struct session_*.

Struct Documentation

```
struct udho::configs::session_::extension_t
```

Public Types

```
typedef session_<T> component
```

Struct **session_::id_t**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

This struct is a nested type of *Template Struct session_*.

Struct Documentation

```
struct udho::configs::session_::id_t
```

Public Types

```
typedef session_<T> component
```

Struct **session_::path_t**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

This struct is a nested type of *Template Struct session_*.

Struct Documentation

```
struct udho::configs::session_::path_t
```

Public Types

```
typedef session_<T> component
```

Struct `session_::serialization_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Nested Relationships

This struct is a nested type of *Template Struct session_*.

Struct Documentation

```
struct udho::configs::session_::serialization_t
```

Public Types

```
typedef session_<T> component
```

Template Struct configuration

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Inheritance Relationships

Base Type

- `public udho::config< T > (Template Struct config)`

Struct Documentation

```
template<typename ...T>
```

```
struct udho::configuration : public udho::config<T>
```

```
    typedef udho::configuration<udho::config::server,      udho::config::router,      udho::config::logging,  
    udho::config::view> configuration; configuration config; config[udho::configs::server::document_root] =  
    “/path/to/document/root”;
```

Public Functions

```
template<typename K>
auto operator[] (const K &k) const

template<typename K>
auto operator[] (const K &k)
```

Struct `connection::send_lambda`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_connection.h`

Nested Relationships

This struct is a nested type of *Template Class* `connection`.

Struct Documentation

```
struct udho::connection::send_lambda
```

Public Functions

```
send_lambda (self_type &self)

template<bool isRequest, class Body, class Fields>
void operator() (http::message<isRequest, Body, Fields> &msg) const

template<bool isRequest, class Body, class Fields>
void operator() (http::message<isRequest, Body, Fields> &&msg) const
```

Public Members

```
self_type &self_
```

Template Struct `content_wrapper0`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename F>
struct udho::content_wrapper0
    mapping of an url with a http request defined by http method and the url pattern
```

A content wrapper is defined by a HTTP verb, a callback and a url pattern. A content wrapper uses a compositor that prepares a HTTP response based on the callbacks return

Template Parameters

- F: callback type

Public Types

```
typedef content_wrapper0<F> self_type
```

Public Functions

```
content_wrapper0 (boost::beast::http::verb method, F ftor)
```

```
template<template<typename> class CompositorT = compositors::transparent>
auto unwrap (CompositorT<typename internal::function_signature<F>::return_type> compositor =
```

```
CompositorT<typename internal::function_signature<F>::return_type>())
```

```
auto deferred ()
```

```
auto raw ()
```

raw content delivery using transparent compositor

```
auto operator= (const std::string &pattern)
```

attach an url pattern

```
auto mimed (std::string mime)
```

applies a mimed compositor on the return

Parameters

- *mime*: returned mime type

```
auto html ()
```

shorthand for html mime type

```
auto plain ()
```

shorthand for plain text mime type

```
auto json ()
```

shorthand for json mime type

Public Members

```
F _ftor
```

```
boost::beast::http::verb _method
```

Template Struct *content_wrapper1*

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename F, typename A1>
```

```
struct udho::content_wrapper1
```

mapping of an url with a http request defined by http method and the url pattern

A content wrapper is defined by a HTTP verb, a callback and a url pattern. A content wrapper uses a compositor that prepares a HTTP response based on the callbacks return

Note binds a variable by reference with the mapping, which might be a database connection or some persistent stateful object

Template Parameters

- `F`: callback type
- `A1`: type of the passed value which will be passed by reference

Public Types

```
typedef content_wrapper1<F, A1> self_type
```

Public Functions

```
content_wrapper1 (boost::beast::http::verb method, F ftor, A1 &a1)
```

```
template<template<typename> class CompositorT = compositors::transparent>
```

```
auto unwrap (CompositorT<typename internal::function_signature<F>::return_type> compositor =  
            CompositorT<typename internal::function_signature<F>::return_type>())
```

```
auto raw ()
```

```
    raw content delivery using transparent compositor
```

```
auto operator= (const std::string &pattern)
```

```
    attach an url pattern
```

```
auto mimed (std::string mime)
```

```
    applies a mimed compositor on the return
```

Parameters

- `mime`: returned mime type

```
auto html ()
```

```
    shorthand for html mime type
```

```
auto plain ()
```

```
    shorthand for plain text mime type
```

```
auto json ()
```

```
    shorthand for json mime type
```

Public Members

```
F _ftor
```

```
A1 &_a1
```

```
boost::beast::http::verb _method
```

Template Struct context

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h`

Inheritance Relationships

Base Type

- `public udho::detail::context_common< AuxT, RequestT >` (*Template Struct context_common*)

Struct Documentation

```
template<typename AuxT, typename RequestT, typename ShadowT>
struct udho::context : public udho::detail::context_common<AuxT, RequestT>
```

A Stateful context passed to all callables along with the arguments. The context is always the first argument to the callable. Even if the callable takes no arguments, it must take the context as the first argument. A stateful context should be used in callables that need to use session states.

Note instead of instantiating this template directly use `udho::contexts::stateful`

Template Parameters

- `AuxT`: bridge between the server and the callable
- `RequestT`: HTTP request type
- `ShadowT`: the session data structure

Public Types

```
typedef detail::context_common<AuxT, RequestT> base_type
typedef RequestT request_type
typedef ShadowT shadow_type
typedef shadow_type::key_type key_type
typedef context<AuxT, request_type, shadow_type> self_type
typedef udho::session_<request_type, shadow_type> session_type
```

Public Functions

```
template<typename ...V>
context (AuxT &aux, const RequestT &request, udho::cache::shadow<key_type, V...> &shadow)

template<typename OtherShadowT>
context (context<AuxT, RequestT, OtherShadowT> &other)

session_type &session ()
    access the HTTP Session

See udho::session_
```

Public Members

session_type **_session**

Template Struct `context< AuxT, RequestT, void >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h`

Inheritance Relationships

Base Type

- `public udho::detail::context_common< AuxT, RequestT >` (*Template Struct context_common*)

Struct Documentation

`template<typename AuxT, typename RequestT>`

struct `udho::context<AuxT, RequestT, void> : public udho::detail::context_common<AuxT, RequestT>`

A Stateless context passed to all callables along with the arguments. he context is always the first argument to the callable. Even if the callable takes no arguments, it must take the context as the first argument. A stateless context should be used in callables that need not to use session states.

Note instead of instantiating this template directly use `udho::contexts::stateless`

Template Parameters

- `AuxT`: bridge between the server and the callable
- `RequestT`: HTTP request type

Public Types

typedef `void` **shadow_type**

typedef `context<AuxT, RequestT, void>` **self_type**

typedef `detail::context_common<AuxT, RequestT>` **base_type**

Public Functions

`template<typename OtherShadowT>`

context (`context<AuxT, RequestT, OtherShadowT> &other`)

Template Struct `cookie_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cookie.h`

Struct Documentation

```
template<typename ValueT>
struct udho::cookie_
```

Public Types

```
typedef cookie_<ValueT> self_type
```

Public Functions

```
cookie_ (const std::string &name)
cookie_ (const std::string &name, const ValueT &value)
cookie_ (const std::string &name, const ValueT &value, const std::string &path)
self_type &path (const std::string &p)
std::string path () const
self_type &domain (const std::string &d)
std::string domain () const
self_type &age (unsigned long age)
unsigned long age () const
template<typename StreamT>
StreamT &render (StreamT &stream) const
std::string to_string () const
```

Public Members

```
std::string _name
ValueT _value
bool _removed
boost::optional<std::string> _comment
boost::optional<std::string> _domain
boost::optional<std::string> _path
boost::optional<unsigned long> _age
boost::optional<bool> _secure
```

Template Struct `cookies_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cookie.h`

Struct Documentation

```
template<typename RequestT>
struct udho::cookies_
```

Public Types

```
typedef RequestT request_type
typedef boost::beast::http::header<true> headers_type
typedef std::map<std::string, std::string> cookie_jar_type
```

Public Functions

```
cookies_ (const request_type &request, headers_type &headers)
void collect ()
template<typename V>
void add (const cookie_<V> &c)
template<typename V>
void add (const std::string &key, const V &value)
bool exists (const std::string &key) const
template<typename V>
V get (const std::string &key) const
```

Public Members

```
const request_type &_request
headers_type &_headers
cookie_jar_type _jar
```

Template Struct `data_helper`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename H, typename T, bool IsPrepared = detail::is_prepared<H>::value>
struct data_helper
```

Template Struct data_helper< H, T, false >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename H, typename T>
struct udho::data_helper<H, T, false>
```

Public Types

```
typedef prepared_group<udho::prepared<T>, prepared_group<H, void>> result_type
```

Public Static Functions

```
result_type data (const H &head, const T &tail)
```

Template Struct data_helper< H, T, true >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename H, typename T>
struct udho::data_helper<H, T, true>
```

Public Types

```
typedef prepared_group<udho::prepared<T>, prepared_group<udho::prepared<H>, void>> result_type
```

Public Static Functions

```
result_type data (const H &head, const T &tail)
```

Struct declaration

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_scope.h`

Struct Documentation

struct `udho::declaration`

Public Functions

declaration (**const** `std::string &key`, `std::size_t depth`)

bool operator< (**const** *declaration* &decl) **const**

Public Members

`std::string _id`

`std::string _ref`

`std::size_t _depth`

Template Struct association

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Inheritance Relationships

Base Type

- `public udho::detail::responder< F, F::result_type >` (*Template Struct responder*)

Struct Documentation

`template<typename F>`

struct `udho::detail::association` : **public** `udho::detail::responder<F, F::result_type>`

Public Types

typedef *F* **callback_type**

typedef *responder*<*F*, `typename F::result_type`> **responder_type**

typedef *responder_type*::`result_type` **result_type**

Public Functions

association (**const** std::string &*key*, *callback_type* *callback*)

bool **matched** (**const** std::string &*key*) **const**

Public Members

std::string **_key**

Template Struct association_group

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

template<typename **U**, typename **V** = void>

struct udho::detail::association_group

```
association_group<X, void> | Y =>
↪ association_group<Y, association_group<X, void>>
association_group<Y, association_group<X, void>> | Z =>
↪ association_group<Z, association_group<Y, association_group<X, void>>>
association_group<Z, association_group<Y, association_group<X, void>>> | T =>
↪ association_group<T, association_group<Z, association_group<Y, association_group
↪ <X, void>>>>
```

Public Types

typedef *U* **head_type**

typedef *V* **tail_type**

typedef *head_type*::result_type **result_type**

typedef *association_group*<*U*, *V*> **self_type**

Public Functions

association_group (**const** *head_type* &*head*, **const** *tail_type* &*tail*)

bool **matched** (**const** std::string &*key*) **const**

bool **exists** (**const** std::string &*key*) **const**

auto **call** (**const** std::string &*key*) **const**

auto **call** (**const** std::string &*key*)

bool **iterable** () **const**

bool **leaf** () **const**

std::size_t **count** () **const**

std::vector<std::string> **keys** () **const**

Public Members

head_type **_head**

tail_type **_tail**

Template Struct association_group< U, void >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename U>
struct udho::detail::association_group<U, void>
```

Public Types

```
typedef U head_type
```

```
typedef void tail_type
```

```
typedef head_type::result_type result_type
```

```
typedef association_group<U, void> self_type
```

Public Functions

```
association_group (const head_type &head)
```

```
bool matched (const std::string &key) const
```

```
bool exists (const std::string &key) const
```

```
auto call (const std::string &key) const
```

```
auto call (const std::string &key)
```

```
bool iterable () const
```

```
bool leaf () const
```

```
std::size_t count () const
```

```
std::vector<std::string> keys () const
```

Public Members

head_type **_head**

Template Struct `association_group_visitor`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename GroupT, bool IsPrepared = is_prepared<typename GroupT::head_type::result_type>::value>
struct association_group_visitor
    An association group visitor visits the association group returned by T::index() where T is a subclass of
    udho::prepare
```

Template Struct `association_group_visitor< association_group< U, V >, false >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename U, typename V>
struct udho::detail::association_group_visitor<association_group<U, V>, false>
    not the last association, does not have nested objects
```

Public Types

```
typedef association_group<U, V> group_type
typedef V tail_type
typedef association_group_visitor<tail_type> tail_visitor_type
```

Public Functions

```
association_group_visitor (const group_type &group)
template<typename CallbackT>
bool find (CallbackT &callback, const std::string &key) const
std::size_t count (const std::string &key) const
std::vector<std::string> keys (const std::string &key) const
```

Public Members

```
const group_type &_group
tail_visitor_type _tail_visitor
```

Template Struct `association_group_visitor< association_group< U, V >, true >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename U, typename V>
struct udho::detail::association_group_visitor<association_group<U, V>, true>
    not the last association, have nested objects
```

Public Types

```
typedef association_group<U, V> group_type
typedef group_type::result_type result_type
typedef V tail_type
typedef association_group_visitor<tail_type> tail_visitor_type
```

Public Functions

```
association_group_visitor (const group_type &group)
template<typename CallbackT>
bool find (CallbackT &callback, const std::string &key) const
std::size_t count (const std::string &key) const
std::vector<std::string> keys (const std::string &key) const
```

Public Members

```
const group_type &_group
tail_visitor_type _tail_visitor
```

Template Struct `association_group_visitor< association_group< U, void >, false >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename U>
struct udho::detail::association_group_visitor<association_group<U, void>, false>
    last association, does not have nested object
```

Public Types

```
typedef association_group<U, void> group_type
```

Public Functions

```
association_group_visitor (const group_type &group)
```

```
template<typename CallbackT>  
bool find (CallbackT &callback, const std::string &key) const  
std::size_t count (const std::string&) const  
std::vector<std::string> keys (const std::string&) const
```

Public Members

```
const group_type &_group
```

Struct `association_leaf`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
struct udho::detail::association_leaf
```

Public Types

```
typedef void result_type
```

Public Functions

```
bool matched (const std::string&) const  
bool call (const std::string&) const  
template<typename C>  
bool invoke (const std::string&, C) const
```

Template Struct `association_lexical_extractor`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename ValueT>
struct udho::detail::association_lexical_extractor
```

Public Functions

```
association_lexical_extractor ()
```

```
void operator () (ValueT value)
```

```
template<typename T>
std::enable_if<!std::is_same<ValueT, T>::value && is_streamable<std::stringstream, T>::value>::type operator () (const
                                                                                                     T
                                                                                                     &value)
```

```
template<typename T>
std::enable_if<!std::is_same<ValueT, T>::value && !is_streamable<std::stringstream, T>::value>::type operator () (const
                                                                                                     T
                                                                                                     &value)
```

```
ValueT value () const
```

```
void clear ()
```

Public Members

```
ValueT _value
```

```
bool _success
```

Template Struct association_value_extractor

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h

Struct Documentation

```
template<typename ValueT>
struct udho::detail::association_value_extractor
```

Public Functions

```
association_value_extractor ()
```

```
void operator () (ValueT value)
```

```
template<typename T>
std::enable_if<!std::is_same<ValueT, T>::value>::type operator () (const T&)
```

```
ValueT value () const
```

```
void clear ()
```

Public Members

ValueT **_value**

bool **_success**

Template Struct `async_result`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Inheritance Relationships

Derived Types

- public `udho::detail::http_client_connection< ContextT >` (*Template* *Struct*
http_client_connection)
- public `udho::detail::https_client_connection< ContextT >` (*Template* *Struct*
https_client_connection)

Struct Documentation

template<typename **ContextT**>

struct `udho::detail::async_result`

Subclassed by `udho::detail::http_client_connection< ContextT >`, `udho::detail::https_client_connection< ContextT >`

Public Types

typedef *ContextT* **context_type**

typedef *async_result*<*ContextT*> **self_type**

typedef `udho::config<udho::client_options>` **options_type**

typedef `boost::function<void ( ContextT, const boost::beast::http::response<boost::beast::http::string_body>& ) >`
success_callback_type

typedef `boost::function<void ( ContextT, const boost::beast::error_code& ) >`
error_callback_type

typedef `boost::function<void ( const boost::beast::error_code& ) >`
error_callback_type_aux

typedef `boost::function<void ( const boost::beast::http::response<boost::beast::http::string_body>& ) >`
success_callback_type_aux_r

typedef `boost::function<void ( ContextT, boost::beast::http::status, const std::string& ) >`
success_callback_type_aux_xsc

typedef `boost::function<void ( ContextT, const std::string& ) >` **success_callback_type_aux_xc**

typedef `boost::function<void ( boost::beast::http::status, const std::string& ) >`
success_callback_type_aux_sc

typedef `boost::function<void ( const std::string& ) >` **success_callback_type_aux_c**

typedef `boost::function<void ( ) >` **finally_callback_type**

Public Functions

```
async_result () = delete
async_result (const context_type &ctx, options_type options)
async_result (const self_type &other) = delete
template<typename KeyT, typename ValueT>
self_type &option (KeyT key, ValueT value)
template<typename KeyT>
auto option (KeyT key) const
void success (const boost::beast::http::response<boost::beast::http::string_body> &res)
void failure (const boost::beast::error_code &ec)
void finish ()
self_type &then (success_callback_type cb)
self_type &failed (error_callback_type cb)
self_type &finally (finally_callback_type f)
self_type &error (error_callback_type_aux cb)
self_type &fetch (success_callback_type_aux_xsc cb)
self_type &body (success_callback_type_aux_xc cb)
self_type &after (success_callback_type_aux_r cb)
self_type &done (success_callback_type_aux_sc cb)
self_type &content (success_callback_type_aux_c cb)
context_type &context ()
```

Public Members

```
context_type _ctx
success_callback_type _callback
error_callback_type _ecallback
finally_callback_type _fcallback
options_type _options
```

Template Struct `client_connection_wrapper`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Struct Documentation

```
template<typename ContextT>
struct udho::detail::client_connection_wrapper
```

Public Types

```
typedef client_connection_wrapper<ContextT> self_type
typedef udho::detail::async_result<ContextT> result_type
typedef udho::config<udho::client_options> options_type
```

Public Functions

```
client_connection_wrapper (boost::asio::io_service &io, ContextT ctx, options_type options)
result_type &request (boost::beast::http::verb method, udho::url url)
result_type &get (udho::url url)
result_type &post (udho::url url)
result_type &put (udho::url url)
result_type &get (const std::string &url)
result_type &post (const std::string &url)
result_type &put (const std::string &url)
```

Public Members

```
boost::asio::io_service &_io
ContextT _context
options_type _options
```

Template Struct context_common

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h`

Inheritance Relationships

Derived Types

- `public udho::context< AuxT, RequestT, ShadowT >` (*Template Struct context*)
- `public udho::context< AuxT, RequestT, void >` (*Template Struct context< AuxT, RequestT, void >*)

Struct Documentation

template<typename **AuxT**, typename **RequestT**>

struct udho::detail::context_common

Subclassed by *udho::context< AuxT, RequestT, ShadowT >*, *udho::context< AuxT, RequestT, void >*

Public Types

typedef *RequestT* request_type

typedef *context_common<AuxT, request_type>* self_type

typedef detail::context_impl<request_type> impl_type

typedef boost::shared_ptr<impl_type> pimple_type

typedef udho::forms::form_<RequestT> form_type

typedef udho::cookies_<RequestT> cookies_type

typedef udho::forms::query_ query_parser_type

Public Functions

template<typename **C**>

context_common (*AuxT* &aux, **const** *RequestT* &request, **const** *C* &)

template<typename **ShadowT**>

context_common (*self_type* &other)

interaction_ &**interaction** ()

const *request_type* &**request** () **const**

returns the boost beast HTTP request

template<class **Body**, class **Fields**>

void **patch** (boost::beast::http::message<false, *Body*, *Fields*> &res) **const**

patches the response with the headers added in the context from inside the callable

form_type &**form** ()

returns the form handler object.

See *udho::form_*

See *udho::urlencoded_form*

See *udho::multipart_form*

cookies_type &**cookies** ()

accesses the HTTP cookies

See *udho::cookies_*

operator request_type () **const**

conversion operator to convert to the corresponding beast request type

template<udho::logging::status **Status**>

void **log** (**const** udho::logging::message<*Status*> &msg) **const**

Logs a message to the logger attached with the server. A logging message can be passed to the `log` method or the `operator<<` can be used to log a message.

```

ctx << udho::logging::messages::formatted::debug("data", "testing log_
↳ functionality of %1% Hi %2%") % "Neel Basu" % 42;
ctx << udho::logging::messages::formatted::info("data", "testing log_
↳ functionality of %1% Hi %2%") % "Neel Basu" % 42;
ctx << udho::logging::messages::formatted::warning("data", "testing log_
↳ functionality of %1% Hi %2%") % "Neel Basu" % 42;
ctx << udho::logging::messages::formatted::error("data", "testing log_
↳ functionality of %1% Hi %2%") % "Neel Basu" % 42;

```

See *udho::logging::message*

template<typename **AttachmentT**>

void **attach** (*AttachmentT* &attachment)

attaches a context with its server counter part

AuxT &**aux** ()

returns a reference to the bridge between the callable and the server

void **respond** (udho::defs::response_type &response)

respond with a http status. The responded output will be put inside a beast HTTP response object and a content type header of type mime will be attached

template<typename **OutputT**>

void **respond** (const *OutputT* &output, const std::string &mime)

the responded output will be put inside a beast HTTP response object and a content type header of type mime will be attached

template<typename **OutputT**>

void **respond** (boost::beast::http::status s, const *OutputT* &output, const std::string &mime)

respond with a http status. The responded output will be put inside a beast HTTP response object and a content type header of type mime will be attached

void **status** (boost::beast::http::status s)

set a status code for the HTTP response

std::string **target** () const

target of the HTTP request including the ? if the request includes get parameters

```

// https://localhost/user/profile?id=245
ctx.target() // /user/profile?id=245

```

std::string **path** () const

path of the HTTP request before ? if any

```

// https://localhost/user/profile?id=245
ctx.path() // /user/profile

```

const *query_parser_type* &**query** () const

The get query of the HTTP request.

```

if(!ctx.query().has("type") || ctx.query().field<std::string>("type") == "json
↳ ") {
    // respond with JSON content
}else if (ctx.query().field<std::string>("type") == "xml")
    // respond with xml content
}

```

See *udho::urlencoded_form*

```
void clear ()

void reroute (const std::string &path)
    Internally reroute an HTTP request to another request

bool rerouted () const
    check whether this is a rerouted request or not

std::string alt_path () const

std::string render (const std::string &path) const
    render a file in path

template<typename ...DataT>
std::string render (const std::string &path, const DataT&... data) const
    render a template in path

void push (const udho::detail::route &r)

udho::detail::route top () const

udho::detail::route pop ()

std::size_t reroutes () const

detail::client_connection_wrapper<self_type> client (udho::config<udho::client_options> options =
    udho::config<udho::client_options>())

boost::asio::io_service &io ()
```

Public Members

pimpl_type **_pimpl**
AuxT &**_aux**

Template Struct context_impl

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h`

Inheritance Relationships

Base Type

- `public udho::detail::interaction_ (Struct interaction_)`

Struct Documentation

```
template<typename RequestT>
struct udho::detail::context_impl : public udho::detail::interaction_
```

Public Types

```
typedef RequestT request_type
typedef context_impl<request_type> self_type
typedef udho::forms::form_<request_type> form_type
typedef udho::cookies_<request_type> cookies_type
typedef boost::beast::http::header<true> headers_type
typedef udho::forms::query_ query_parser_type
typedef std::stack<udho::detail::route> route_stack_type
```

Public Functions

```
context_impl (const request_type &request)
context_impl (const self_type &other) = delete
interaction_ &interaction ()
const request_type &request () const
cookies_type &cookies ()
template<class Body, class Fields>
void patch (boost::beast::http::message<false, Body, Fields> &res) const
void status (boost::beast::http::status status)
void clear ()
void reroute (const std::string &path)
bool rerouted () const
std::string alt_path () const
std::string target () const
std::string path () const
std::string query_string () const
const query_parser_type &query () const
void push (const udho::detail::route &r)
udho::detail::route top () const
udho::detail::route pop ()
std::size_t reroutes () const
```

Public Members

```
const request_type &_request
form_type _form
std::string _query_string
query_parser_type _query
headers_type _headers
cookies_type _cookies
route_stack_type _routes
boost::beast::http::status _status
```

Template Struct http_client_connection

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Inheritance Relationships

Base Types

- public `std::enable_shared_from_this< http_client_connection< ContextT > >`
- public `udho::detail::async_result< ContextT >` (*Template Struct async_result*)

Struct Documentation

```
template<typename ContextT>
```

```
struct udho::detail::http_client_connection : public std::enable_shared_from_this<http_client_connection<ContextT>>
```

```
    Todo:  
        write docs
```

Public Types

```
typedef std::enable_shared_from_this<http_client_connection<ContextT>> base  
typedef async_result<ContextT> result_type  
typedef http_client_connection<ContextT> self_type  
typedef ContextT context_type  
typedef udho::config<udho::client_options> options_type  
typedef boost::function<void (const std::string&)> redirector_type
```

Public Functions

```

http_client_connection (ContextT context, const udho::url &u, boost::asio::executor ex, options_type options)
void start (boost::beast::http::verb method = boost::beast::http::verb::get)
void on_resolve (boost::beast::error_code ec, boost::asio::ip::tcp::resolver::results_type results)
void on_connect (boost::beast::error_code ec)
void on_write (boost::beast::error_code ec, std::size_t)
void on_read (boost::beast::error_code ec, std::size_t)
void on_shutdown (boost::beast::error_code ec)
result_type &result ()

```

Public Members

```

udho::url _url
boost::asio::ip::tcp::resolver resolver
boost::asio::ip::tcp::socket socket
boost::beast::flat_buffer buffer
boost::beast::http::request<boost::beast::http::empty_body> req
boost::beast::http::response<boost::beast::http::string_body> res

```

Public Static Functions

```

std::shared_ptr<self_type> create (boost::asio::io_service &io, ContextT ctx, udho::url url, options_type options)

```

Template Struct https_client_connection

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Inheritance Relationships

Base Types

- `public std::enable_shared_from_this< https_client_connection< ContextT > >`
- `public udho::detail::async_result< ContextT > (Template Struct async_result)`

Struct Documentation

template<typename **ContextT**>

struct udho::detail::https_client_connection : public std::enable_shared_from_this<https_client_connection<ContextT>>

Todo:

write docs

Public Types

typedef std::enable_shared_from_this<https_client_connection<ContextT>> **base**

typedef async_result<ContextT> **result_type**

typedef https_client_connection<ContextT> **self_type**

typedef ContextT **context_type**

typedef udho::config<udho::client_options> **options_type**

typedef boost::function<void (const std::string&)> **redirector_type**

Public Functions

https_client_connection (ContextT context, const udho::url &u, boost::asio::executor ex, boost::asio::ssl::context &ssl_ctx, options_type options)

void **start** (boost::beast::http::verb method = boost::beast::http::verb::get)

void **on_resolve** (boost::beast::error_code ec, boost::asio::ip::tcp::resolver::results_type results)

void **on_connect** (boost::beast::error_code ec)

void **on_handshake** (boost::beast::error_code ec)

void **on_write** (boost::beast::error_code ec, std::size_t)

void **on_read** (boost::beast::error_code ec, std::size_t)

void **on_shutdown** (boost::beast::error_code ec)

result_type &**result** ()

Public Members

udho::url **_url**

boost::asio::ip::tcp::resolver **resolver**

boost::asio::ssl::context &**_ssl_ctx**

boost::asio::ssl::stream<boost::asio::ip::tcp::socket> **stream**

boost::beast::flat_buffer **buffer**

boost::beast::http::request<boost::beast::http::empty_body> **req**

boost::beast::http::response<boost::beast::http::string_body> **res**

Public Static Functions

`std::shared_ptr<self_type> create` (`boost::asio::io_service &io`, *ContextT* `ctx`, `udho::url url`, *options_type* `options`)

Struct `interaction_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h`

Inheritance Relationships

Derived Type

- `public udho::detail::context_impl< RequestT >` (*Template Struct context_impl*)

Struct Documentation

struct `udho::detail::interaction_`
Subclassed by `udho::detail::context_impl< RequestT >`

Public Functions

```
template<typename AuxT, typename LoggerT, typename CacheT>
void attach (udho::attachment<AuxT, LoggerT, CacheT> &attachment)

void log (const udho::logging::messages::error &msg) const
void log (const udho::logging::messages::warning &msg) const
void log (const udho::logging::messages::info &msg) const
void log (const udho::logging::messages::debug &msg) const
void respond (udho::defs::response_type &response)
```

Public Members

```
boost::signals2::signal<void (const udho::logging::messages::error&) > _error
boost::signals2::signal<void (const udho::logging::messages::warning&) > _warning
boost::signals2::signal<void (const udho::logging::messages::info&) > _info
boost::signals2::signal<void (const udho::logging::messages::debug&) > _debug
boost::signals2::signal<void (udho::defs::response_type&) > _respond
```

Template Struct `is_prepared`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename U>
struct udho::detail::is_prepared
```

Public Types

```
enum [anonymous]
    Values:
    enumerator value
```

Public Static Functions

```
template<typename V>
    V::prepared_type test (int)

template<typename>
    void test (...)
```

Template Struct `responder`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename F, typename R>
struct udho::detail::responder
```

Public Types

```
typedef F callback_type
typedef R result_type
typedef responder<F, R> self_type
```

Public Functions

```

responder () = delete
responder (callback_type callback)
responder (const self_type &other)
result_type call (const std::string&) const
result_type call (const std::string&)
bool iterable () const
bool leaf () const
std::size_t count () const
std::vector<std::string> keys () const

```

Public Members

```
callback_type _callback
```

Template Struct responder< F, std::map< U, V > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```

template<typename F, typename U, typename V>
struct udho::detail::responder<F, std::map<U, V>>

```

Public Types

```

typedef F callback_type
typedef V value_type
typedef U key_type
typedef std::map<U, V> container_type
typedef V result_type
typedef container_type::size_type size_type

```

Public Functions

```
responder (callback_type callback)  
result_type call (const std::string &key) const  
result_type call (const std::string &key)  
bool iterable () const  
bool leaf () const  
std::size_t count () const  
std::vector<std::string> keys () const
```

Public Members

```
callback_type _callback
```

Template Struct responder< F, std::vector< V > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename F, typename V>  
struct udho::detail::responder<F, std::vector<V>>
```

Public Types

```
typedef F callback_type  
typedef V value_type  
typedef std::vector<V> container_type  
typedef V result_type  
typedef container_type::size_type size_type
```

Public Functions

```
responder (callback_type callback)  
result_type call (const std::string &key) const  
result_type call (const std::string &key)  
bool iterable () const  
bool leaf () const  
std::size_t count () const  
std::vector<std::string> keys () const
```

Public Members

callback_type **_callback**

Struct route

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

struct udho::detail::route

Public Functions

void **reroute** (const std::string &path)

bool **rerouted** () const

std::string **rerouted_path** () const

Public Members

std::string **_path**

std::string **_subject**

std::string **_pattern**

std::string **_rerouted**

Template Struct **url_data_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

Nested Types

- *Struct* `url_data_::host_t`
- *Struct* `url_data_::path_t`
- *Struct* `url_data_::port_t`
- *Struct* `url_data_::protocol_t`
- *Struct* `url_data_::query_t`
- *Struct* `url_data_::target_t`

Struct Documentation

```
template<typename T = void>
struct udho::detail::url_data_
```

Public Functions

```
url_data_()
void set (protocol_t, const std::string &v)
std::string get (protocol_t) const
void set (host_t, const std::string &v)
std::string get (host_t) const
void set (port_t, std::size_t v)
std::size_t get (port_t) const
void set (target_t, const std::string &v)
std::string get (target_t) const
void set (path_t, const std::string &v)
std::string get (path_t) const
void set (query_t, const std::string &v)
const std::string &get (query_t) const
```

Public Members

```
std::string _protocol
std::string _host
std::size_t _port
std::string _path
std::string _target
std::string _query
```

Public Static Attributes

```
const struct udho::detail::url_data_::protocol_t protocol
const struct udho::detail::url_data_::host_t host
const struct udho::detail::url_data_::port_t port
const struct udho::detail::url_data_::target_t target
const struct udho::detail::url_data_::path_t path
const struct udho::detail::url_data_::query_t query
struct host_t
```

Public Types

```
typedef url_data_<T> component
struct path_t
```

Public Types

```
typedef url_data_<T> component
struct port_t
```

Public Types

```
typedef url_data_<T> component
struct protocol_t
```

Public Types

```
typedef url_data_<T> component
struct query_t
```

Public Types

```
typedef url_data_<T> component
struct target_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::host_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct url_data_*.

Struct Documentation

```
struct udho::detail::url_data_::host_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::path_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct url_data_*.

Struct Documentation

```
struct udho::detail::url_data_::path_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::port_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct url_data_*.

Struct Documentation

```
struct udho::detail::url_data_::port_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::protocol_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct `url_data_`*.

Struct Documentation

```
struct udho::detail::url_data_::protocol_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::query_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct `url_data_`*.

Struct Documentation

```
struct udho::detail::url_data_::query_t
```

Public Types

```
typedef url_data_<T> component
```

Struct `url_data_::target_t`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This struct is a nested type of *Template Struct `url_data_`*.

Struct Documentation

```
struct udho::detail::url_data::target_t
```

Public Types

```
typedef url_data<T> component
```

Struct http_error

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_page.h`

Inheritance Relationships

Base Type

- `public exception`

Struct Documentation

```
struct udho::exceptions::http_error : public exception
    throw HTTP error exception in order to send a HTTP response other than 200 OK. Uncaught exceptions are
    presented as html page with routing summary
```

```
http_error(boost::beast::http::status::notfound)
```

Public Types

```
typedef boost::beast::http::header<false> headers_type
```

Public Functions

```
http_error (boost::beast::http::status status, const std::string &message = "")
```

```
void add_header (boost::beast::http::field key, const std::string &value)
```

```
void redirect (const std::string &url)
```

```
template<typename T>
```

```
boost::beast::http::response<boost::beast::http::string_body> response (const
                                                                    boost::beast::http::request<T>
                                                                    &request) const
```

```
template<typename AuxT, typename U, typename V>
```

```
boost::beast::http::response<boost::beast::http::string_body> response (const udho::context<AuxT,
                                                                    U, V> &ctx) const
```

```
template<typename T, typename RouterT>
```

```
boost::beast::http::response<boost::beast::http::string_body> response (const
                                                                    boost::beast::http::request<T>
                                                                    &request, RouterT &router)
                                                                    const
```

```

template<typename AuxT, typename U, typename V, typename RouterT>
boost::beast::http::response<boost::beast::http::string_body> response (const udho::context<AuxT,
                                                                    U, V> &ctx, RouterT
                                                                    &router) const

std::string page (const std::string &target, std::string content = "") const

template<typename RouterT>
std::string page (const std::string &target, RouterT &router) const

const char *what () const noexcept

boost::beast::http::status result () const

```

Public Members

```

boost::beast::http::status _status

std::string _message

headers_type _headers

```

Struct reroute

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_page.h

Inheritance Relationships

Base Type

- public exception

Struct Documentation

```
struct udho::exceptions::reroute : public exception
```

Public Functions

```

reroute (const std::string &alt_path)

std::string alt_path () const

```

Public Members

```
std::string _alt_path
```

Template Struct field

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
template<typename T, bool Required>
struct field
```

Template Struct field< T, false >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Inheritance Relationships

Base Types

- `public udho::field_common` (*Struct field_common*)
- `public udho::field_value_extractor< T >` (*Template Struct field_value_extractor*)

Struct Documentation

```
template<typename T>
struct udho::field<T, false> : public udho::field_common, public udho::field_value_extractor<T>
```

Public Types

```
typedef field<T, false> self_type
typedef field_common common_type
typedef boost::function<bool (const std::string&, std::string&) > function_type
typedef field_value_extractor<T> extractor_type
typedef std::vector<function_type> validators_collection_type
```

Public Functions

```
template<typename ...U>
field (const std::string &name, U... args)

T value () const

template<typename F>
self_type &constrain (F ffor)

template<typename FormT>
void validate (const FormT &form)

void check (const std::string &input)
```

self_type &operator() (const std::string &input)
 validate the field when using a field without a form (e.g. JSON or XML document) if a value if provided in the document

self_type &unchecked (const T &input)

self_type &operator() ()
 validate the field when using a field without a form (e.g. JSON or XML document) if no value if provided in the document

Public Members

T_value

validators_collection_type _validators

Template Struct field< T, true >

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx

Inheritance Relationships

Base Types

- public udho::field_common (*Struct field_common*)
- public udho::field_value_extractor< T > (*Template Struct field_value_extractor*)

Struct Documentation

```
template<typename T>
struct udho::field<T, true> : public udho::field_common, public udho::field_value_extractor<T>
```

Public Types

```
typedef field<T, true> self_type
```

```
typedef field_common common_type
```

```
typedef boost::function<bool (const std::string&, std::string&) > function_type
```

```
typedef field_value_extractor<T> extractor_type
```

```
typedef std::vector<function_type> validators_collection_type
```

Public Functions

```
template<typename ...U>
field (const std::string &name, const std::string &message, U... args)

T value () const

template<typename F>
self_type &constrain (F ffor)

template<typename FormT>
void validate (const FormT &form)

void check (const std::string &input)

self_type &operator () (const std::string &input)
    validate the field when using a field without a form (e.g. JSON or XML document) if a value if provided
    in the document

self_type &unchecked (const T &input)

self_type &operator () ()
    validate the field when using a field without a form (e.g. JSON or XML document) if no value if provided
    in the document
```

Public Members

```
T _value

validators_collection_type _validators

std::string _message_required
```

Struct field_common

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Inheritance Relationships

Base Type

- `public udho::prepare< field_common > (Template Struct prepare)`

Derived Types

- `public udho::field< T, false > (Template Struct field< T, false >)`
- `public udho::field< T, true > (Template Struct field< T, true >)`

Struct Documentation

struct `udho::field_common` : **public** `udho::prepare<field_common>`
 common meta parameters of a form field

Subclassed by `udho::field< T, false >`, `udho::field< T, true >`

Public Functions

field_common (**const** `std::string &name`)

bool validated () **const**

std::string name () **const**

bool valid () **const**

std::string error () **const**

void clear ()

template<**typename DictT**>

auto dict (*DictT assoc*) **const**

Public Members

std::string _name

bool _is_valid

bool _validated

std::string _err

std::string _value

Template Struct `field_value_extractor`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Inheritance Relationships

Derived Types

- `public udho::field< T, false >` (*Template Struct `field< T, false >`*)
- `public udho::field< T, true >` (*Template Struct `field< T, true >`*)

Struct Documentation

```
template<typename T>
struct udho::field_value_extractor
    Subclassed by udho::field< T, false >, udho::field< T, true >
```

Public Types

```
typedef T value_type
```

Public Functions

```
field_value_extractor (const std::string &message = "Extraction Failed")
bool operator () (const std::string &input, value_type &value) const
std::string message () const
```

Public Members

```
std::string _message
```

```
Template Struct field_value_extractor< std::chrono::time_point< std::chrono::system_clock,
std::chrono::microseconds > >
```

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
template<>
struct udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock, std::chrono::microseconds>>
```

Public Types

```
typedef std::chrono::time_point<std::chrono::system_clock, std::chrono::microseconds> value_type
```

Public Functions

```
field_value_extractor (const std::string &format, const std::string &message = "Extraction
Failed")
bool operator () (const std::string &input, value_type &value) const
std::string message () const
```

Public Members

std::string **_format**
std::string **_message**

Template Struct validated

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Inheritance Relationships

Base Type

- public udho::prepare< validated< FormT > > (*Template Struct prepare*)

Struct Documentation

```
template<typename FormT = void>
struct udho::form::validated: public udho::prepare<validated<FormT>>
```

Public Types

```
typedef udho::prepare<validated<FormT>> base
typedef FormT form_type
typedef validated<FormT> self_type
```

Public Functions

```
validated (const form_type &form)
void add (const field_common &fld)
bool valid () const
const std::vector<std::string> &errors () const
const field_common &operator [] (const std::string &name) const
template<typename DictT>
auto dict (DictT assoc) const
```

Public Members

```
const form_type &_form
bool _submitted
bool _valid
std::vector<std::string> _errors
std::map<std::string, field_common> _fields
```

Template Struct `validated< void >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Inheritance Relationships

Base Type

- public `udho::prepare< validated< void > > (Template Struct prepare)`

Struct Documentation

```
template<>
struct udho::form::validated<void>: public udho::prepare<validated<void>>
```

Public Types

```
typedef udho::prepare<validated<>> base
typedef validated<void> self_type
```

Public Functions

```
validated()
void add(const field_common &fld)
bool valid() const
const std::vector<std::string> &errors() const
const field_common &operator [] (const std::string &name) const
template<typename DictT>
auto dict (DictT assoc) const
```

Public Members

```
bool _submitted  
bool _valid  
std::vector<std::string> _errors  
std::map<std::string, field_common> _fields
```

Struct all_digit

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::all_digit
```

Public Functions

```
all_digit (std::string custom = "")  
bool operator () (const std::string &value, std::string &error) const
```

Public Members

```
std::string _custom
```

Struct date_time

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::date_time
```

Public Functions

```
date_time (std::string format, std::string custom = "")  
bool operator () (const std::string &value, std::string &error) const
```

Public Members

std::string **_format**
std::string **_custom**

Struct `exact_length`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::exact_length
```

Public Functions

```
exact_length (std::size_t length, std::string custom = "")  
bool operator () (const std::string &value, std::string &error) const
```

Public Members

std::size_t **_length**
std::string **_custom**

Struct `max_length`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::max_length
```

Public Functions

```
max_length (std::size_t length, std::string custom = "")  
bool operator () (const std::string &value, std::string &error) const
```

Public Members

std::size_t **_length**
std::string **_custom**

Struct `min_length`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::min_length
```

Public Functions

```
min_length (std::size_t length, std::string custom = "")
```

```
bool operator () (const std::string &value, std::string &error) const
```

Public Members

```
std::size_t _length
```

```
std::string _custom
```

Struct `no_space`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
struct udho::form::validators::no_space
```

Public Functions

```
no_space (std::string custom = "")
```

```
bool operator () (const std::string &value, std::string &error) const
```

Public Members

```
std::string _custom
```

Template Struct `form_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

template<typename **RequestT**>

struct udho::form_

unified form accessor that can extract field from both urlencoded as well as multipart forms.

Public Types

typedef *RequestT* **request_type**

typedef *request_type::body_type::value_type* **body_type**

typedef std::map<std::string, std::string> **fields_map_type**

Public Functions

form_ (**const** *request_type* &*request*)

void **parse_urlencoded** ()

parse the beast request body as urlencoded form data

void **parse_multipart** ()

parse the beast request body as multipart form data

bool **is_urlencoded** () **const**

check whether the submitted form is urlencoded

bool **is_multipart** () **const**

check whether the submitted form is multipart

const *urlencoded_form*<std::string::const_iterator> &**urlencoded** () **const**

return the urlencoded specific form accessor

const *multipart_form*<std::string::const_iterator> &**multipart** () **const**

return the multipart specific form accessor

bool **parsed** () **const**

returns true once the form is parsed

bool **has** (**const** std::string &*name*) **const**

checks whether the form contains any field with matching name

Parameters

- *name*: name of the form field

template<typename **V**>

V **field** (**const** std::string &*name*) **const**

returns a form field object for a given field

fields_map_type::size_type **count** () **const**

returns number of fields in the form

Public Members

```
const request_type &_request
form_type _type
urlencoded_form<std::string::const_iterator> _urlencoded
multipart_form<std::string::const_iterator> _multipart
```

Struct accumulated

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- public `udho::prepare< accumulated >` (*Template Struct prepare*)

Derived Type

- public `udho::forms::validated< form< DriverT > >` (*Template Struct validated< form< DriverT > >*)

Struct Documentation

```
struct udho::forms::accumulated: public udho::prepare<accumulated>
    Subclassed by udho::forms::validated<form< DriverT > >
```

Public Types

```
typedef udho::prepare<accumulated> base
typedef accumulated self_type
```

Public Functions

```
accumulated()
template<typename ValidatorT, typename TailT>
self_type &add(const detail::constrained_field<ValidatorT, TailT> &cf)
self_type &add(const detail::field_common &common)
bool valid() const
void add_error(const std::string &err)
const std::vector<std::string> &errors() const
const detail::field_common &operator[] (const std::string &name) const
template<typename DictT>
```

```
auto dict (DictT assoc) const
```

Struct `all_digits`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< all_digits >` (*Template Struct `basic_constrain`*)

Struct Documentation

```
struct udho::forms::constraints::all_digits : public udho::forms::constraints::basic_constrain<all_digits>
```

Public Types

```
typedef basic_constrain<all_digits> base
```

Public Functions

```
all_digits (std::string message = "")  
bool operator () (const std::string &value) const
```

Template Struct `basic_constrain`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename DerivedT>  
struct udho::forms::constraints::basic_constrain
```

Public Types

```
typedef DerivedT derived_type
```

Public Functions

```

basic_constrain (const std::string &m)
derived_type &self ()
derived_type &error (const std::string &m)
derived_type &message (const std::string &m)
const std::string &message () const

```

Template Struct eq

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< eq< T > >` (*Template Struct basic_constrain*)

Struct Documentation

```

template<typename T>
struct udho::forms::constraints::eq: public udho::forms::constraints::basic_constrain<eq<T>>

```

Public Types

```

typedef basic_constrain<eq<T>> base

```

Public Functions

```

eq (const T &value, const std::string &message = "")
bool operator () (const T &input) const

```

Public Members

```

T _value

```

Template Struct gt

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< gt< T > >` (*Template Struct basic_constrain*)

Struct Documentation

`template<typename T>`

struct `udho::forms::constraints::gt` : **public** `udho::forms::constraints::basic_constrain<gt<T>>`

Public Types

typedef `basic_constrain<gt<T>>` **base**

Public Functions

gt (**const** `T` &*value*, **const** `std::string` &*message* = "")

bool operator () (**const** `T` &*input*) **const**

Public Members

`T` *_value*

Template Struct gt< std::string >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< gt< std::string > >` (*Template Struct basic_constrain*)

Struct Documentation

template<>

struct udho::forms::constraints::gt<std::string> : **public** udho::forms::constraints::basic_constrain<gt<std::string>>

Public Types

typedef basic_constrain<gt<std::string>> **base**

Public Functions

gt (const std::size_t &value, const std::string &message = "")

bool operator () (const std::string &input) const

Public Members

std::size_t _value

Template Struct gte

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Inheritance Relationships

Base Type

- public udho::forms::constraints::basic_constrain< gte< T > > (*Template Struct basic_constrain*)

Struct Documentation

template<typename T>

struct udho::forms::constraints::gte : **public** udho::forms::constraints::basic_constrain<gte<T>>

Public Types

typedef basic_constrain<gte<T>> **base**

Public Functions

```
gte (const T &value, const std::string &message = "")  
bool operator () (const T &input) const
```

Public Members

T _**value**

Template Struct gte< std::string >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< gte< std::string > >`
(*Template Struct basic_constrain*)

Struct Documentation

template<>

```
struct udho::forms::constraints::gte<std::string> : public udho::forms::constraints::basic_constrain<gte<std::string>
```

Public Types

```
typedef basic_constrain<gte<std::string>> base
```

Public Functions

```
gte (const std::size_t &value, const std::string &message = "")  
bool operator () (const std::string &input) const
```

Public Members

std::size_t _**value**

Template Struct in

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< in< T > >` (*Template Struct basic_constrain*)

Struct Documentation

```
template<typename T>
struct udho::forms::constraints::in : public udho::forms::constraints::basic_constrain<in<T>>
```

Public Types

```
typedef basic_constrain<in<T>> base
```

Public Functions

```
template<typename It>
in (const std::string &message, It begin, It end)

template<typename ...U>
in (const std::string &message, const U&... args)

template<typename ...U>
in (const U&... args)

bool operator () (const T &input) const
```

Public Members

```
std::set<T> _values
```

Struct length_eq

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< length_eq >` (*Template Struct basic_constrain*)

Struct Documentation

struct udho::forms::constraints::length_eq: **public** udho::forms::constraints::basic_constrain<length_eq>

Public Types

typedef basic_constrain<length_eq> **base**

Public Functions

length_eq (std::size_t *length*, std::string *message* = "")
bool **operator** () (const std::string &*value*) **const**

Public Members

std::size_t **_length**

Template Struct It

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< lt< T > >` (*Template Struct basic_constrain*)

Struct Documentation

template<typename **T**>
struct udho::forms::constraints::lt: **public** udho::forms::constraints::basic_constrain<lt<**T**>>

Public Types

```
typedef basic_constrain<lt<T>> base
```

Public Functions

```
lt (const T &value, const std::string &message = "")  
bool operator () (const T &input) const
```

Public Members

```
T _value
```

Template Struct lt< std::string >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< lt< std::string > > (Template Struct basic_constrain)`

Struct Documentation

```
template<>
```

```
struct udho::forms::constraints::lt<std::string> : public udho::forms::constraints::basic_constrain<lt<std::string>>
```

Public Types

```
typedef basic_constrain<lt<std::string>> base
```

Public Functions

```
lt (const std::size_t &value, const std::string &message = "")  
bool operator () (const std::string &input) const
```

Public Members

`std::size_t _value`

Template Struct lte

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< lte< T > >` (*Template Struct basic_constrain*)

Struct Documentation

```
template<typename T>
struct udho::forms::constraints::lte: public udho::forms::constraints::basic_constrain<lte<T>>
```

Public Types

```
typedef basic_constrain<lte<T>> base
```

Public Functions

```
lte (const T &value, const std::string &message = "")
```

```
bool operator () (const T &input) const
```

Public Members

```
T _value
```

Template Struct lte< std::string >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< lte< std::string > >` (*Template Struct basic_constrain*)

Struct Documentation

```
template<>
```

```
struct udho::forms::constraints::lte<std::string> : public udho::forms::constraints::basic_constrain<lte<std::string>>
```

Public Types

```
typedef basic_constrain<lte<std::string>> base
```

Public Functions

```
lte (const std::size_t &value, const std::string &message = "")
```

```
bool operator () (const std::string &input) const
```

Public Members

```
std::size_t _value
```

Template Struct neq

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< neq< T > >` (*Template Struct basic_constrain*)

Struct Documentation

```
template<typename T>
```

```
struct udho::forms::constraints::neq : public udho::forms::constraints::basic_constrain<neq<T>>
```

Public Types

```
typedef basic_constrain<neq<T>> base
```

Public Functions

```
neq (const T &value, const std::string &message = "")  
bool operator () (const T &input) const
```

Public Members

T_value

Struct no_space

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::constraints::basic_constrain< no_space > (Template Struct basic_constrain)`

Struct Documentation

```
struct udho::forms::constraints::no_space : public udho::forms::constraints::basic_constrain<no_space>
```

Public Types

```
typedef basic_constrain<no_space> base
```

Public Functions

```
no_space (std::string message = "")  
bool operator () (const std::string &value) const
```

Template Struct deserializer

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename V, typename U>
struct deserializer
```

Template Struct deserializer< std::chrono::time_point< std::chrono::system_clock, DurationT >, std::string >

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Struct Documentation

```
template<typename DurationT>
struct udho::forms::deserializer<std::chrono::time_point<std::chrono::system_clock, DurationT>, std::string>
```

Public Types

```
typedef std::chrono::time_point<std::chrono::system_clock, DurationT> time_type
```

Public Static Functions

```
bool check (const std::string &input, const std::string &format = default_datetime_format)
time_type deserialize (const std::string &input, const std::string &format = default_datetime_format)
```

Template Struct deserializer< std::string, std::chrono::time_point< std::chrono::system_clock, DurationT > >

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Struct Documentation

```
template<typename DurationT>
struct udho::forms::deserializer<std::string, std::chrono::time_point<std::chrono::system_clock, DurationT>>
```

Public Types

```
typedef std::chrono::time_point<std::chrono::system_clock, DurationT> time_type
```

Public Static Functions

```
bool check (const std::chrono::time_point<std::chrono::system_clock, DurationT> &input, const  
std::string &format = default_datetime_format)
```

```
std::string deserialize (const std::chrono::time_point<std::chrono::system_clock, DurationT>  
&input, const std::string &format = default_datetime_format)
```

Template Struct `deserializer< std::string, std::string >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<>  
struct udho::forms::deserializer<std::string, std::string>
```

Public Static Functions

```
bool check (const std::string&)
```

```
std::string deserialize (const std::string &input)
```

Template Struct `deserializer< std::string, U >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename U>  
struct udho::forms::deserializer<std::string, U>
```

Public Static Functions

```
bool check (const U &input)
```

```
std::string deserialize (const U &input)
```

Template Struct `deserializer< U, U >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename U>
struct udho::forms::deserializer<U, U>
```

Public Static Functions

```
bool check (const U&)
std::string deserialize (const U &input)
```

Template Struct deserializer< V, std::string >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename V>
struct udho::forms::deserializer<V, std::string>
```

Public Static Functions

```
bool check (const std::string &input)
V deserialize (const std::string &input)
```

Template Struct basic_field

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::field_data< ValueT >` (*Template Struct field_data*)

Struct Documentation

```
template<typename ValueT, typename DerivedT>
struct udho::forms::detail::basic_field: public udho::forms::detail::field_data<ValueT>
```

Public Types

```
typedef ValueT value_type
typedef DerivedT derived_type
typedef field_data<ValueT> data_type
```

Public Functions

```
basic_field (const std::string &name)
derived_type &absent (const std::string &message)
derived_type &unparsable (const std::string &message)
data_type &data ()
const data_type &data () const
template<typename ValidatorT>
detail::constrained_field<ValidatorT, derived_type> constrain (const ValidatorT &validator)
template<template<typename> class ValidatorT, typename ...ArgsT>
detail::constrained_field<ValidatorT<value_type>, derived_type> constrain (ArgsT&&... args)
```

Protected Functions

```
template<typename ValidatorT>
bool validate (const ValidatorT &validator)
derived_type &self ()
const derived_type &self () const
```

Protected Attributes

```
std::string _message_absent
std::string _message_unparsable
```

Template Struct constrained_field

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename ValidatorT, typename FieldT>
struct udho::forms::detail::constrained_field
```

Public Types

```
enum [anonymous]
    Values:
        enumerator depth
typedef FieldT field_type
typedef field_type::value_type value_type
typedef ValidatorT validator_type
typedef constrained_field<validator_type, field_type> self_type
```

Public Functions

```
constrained_field(const validator_type &validator, const FieldT &field)
template<typename FormT>
bool validate(const FormT &form)
template<typename FormT>
bool operator() (const FormT &form)
const field_type &field() const
field_type &field()
bool valid() const
value_type value() const
std::string message() const
value_type operator*() const
bool operator!() const
std::string name() const
template<typename ValidatorU>
detail::constrained_field<ValidatorU, self_type> constrain(const ValidatorU &validator)
template<template<typename> class ValidatorU, typename ...ArgsT>
detail::constrained_field<ValidatorU<value_type>, self_type> constrain(ArgsT&&... args)
template<unsigned Depth>
udho::forms::detail::constraint_visitor<self_type, Depth>::validator_type &constrain()
```

Public Members

```
validator_type _validator
FieldT _field
```

Template Struct `constrained_field< ValidatorT, constrained_field< T... > >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename ValidatorT, typename ...T>
struct udho::forms::detail::constrained_field<ValidatorT, constrained_field<T...>>
```

Public Types

```
enum [anonymous]
    Values:

    enumerator depth

typedef constrained_field<T...> head_type
typedef head_type::field_type field_type
typedef field_type::value_type value_type
typedef ValidatorT validator_type
typedef constrained_field<validator_type, head_type> self_type
```

Public Functions

```
constrained_field(const validator_type &validator, const head_type &head)

template<typename FormT>
bool validate(const FormT &form)

template<typename FormT>
bool operator()(const FormT &form)

field_type &field()

const field_type &field() const

bool valid() const

value_type value() const

std::string message() const

value_type operator*() const

bool operator!() const

std::string name() const

template<typename ValidatorU>
detail::constrained_field<ValidatorU, self_type> constrain(const ValidatorU &validator)

template<template<typename> class ValidatorU, typename ...ArgsT>
detail::constrained_field<ValidatorU<value_type>, self_type> constrain(ArgsT&&... args)

template<unsigned Depth>
udho::forms::detail::constraint_visitor<self_type, Depth>::validator_type &constrain()
```

Public Members

validator_type **_validator**

head_type **_head**

Template Struct **constraint_visitor_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename ConstrainedFieldT, unsigned Depth>
struct udho::forms::detail::constraint_visitor_
```

Public Types

```
typedef constraint_visitor_<typename ConstrainedFieldT::head_type, Depth - 1>::validator_type validator_type
```

Public Static Functions

```
validator_type &validator (ConstrainedFieldT &cf)
```

Template Struct **constraint_visitor_< ConstrainedFieldT, 0 >**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename ConstrainedFieldT>
struct udho::forms::detail::constraint_visitor_<ConstrainedFieldT, 0>
```

Public Types

```
typedef ConstrainedFieldT::validator_type validator_type
```

Public Static Functions

```
validator_type &validator (ConstrainedFieldT &cf)
```

Struct field_common

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::prepare< field_common > (Template Struct prepare)`

Derived Types

- `public udho::forms::detail::field_data< std::chrono::time_point< std::chrono::system_clock, DurationT > > (Template Struct field_data)`
- `public udho::forms::detail::field_data< T > (Template Struct field_data)`
- `public udho::forms::detail::field_data< ValueT > (Template Struct field_data)`

Struct Documentation

struct `udho::forms::detail::field_common` : **public** `udho::prepare<field_common>`
Subclassed by `udho::forms::detail::field_data< std::chrono::time_point< std::chrono::system_clock, DurationT > >`, `udho::forms::detail::field_data< T >`, `udho::forms::detail::field_data< ValueT >`

Public Functions

```
field_common (const std::string &name)
bool absent () const
bool unparsable () const
bool fetched () const
const std::string &name () const
std::string value_serialized () const
bool valid () const
const std::string &message () const
const std::string &error () const
bool operator! () const
template<typename DictT>
auto dict (DictT assoc) const
```

Protected Attributes

```
std::string _name
std::string _message
bool _absent
bool _unparsable
bool _fetched
bool _valid
std::string _value_str
```

Template Struct field_data

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::field_common` (*Struct field_common*)

Derived Type

- `public udho::forms::detail::basic_field< ValueT, DerivedT >` (*Template Struct basic_field*)

Struct Documentation

```
template<typename ValueT>
struct udho::forms::detail::field_data : public udho::forms::detail::field_common
    Subclassed by udho::forms::detail::basic_field< ValueT, DerivedT >
```

Public Types

```
typedef ValueT value_type
typedef field_data<ValueT> self_type
```

Public Functions

```
const value_type &value () const
const value_type &operator* () const
template<typename ...ArgsT>
void value (const value_type &y, const ArgsT&... args)
field_common &common ()
const field_common &common () const
```

Template Struct combo

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Types

- `private udho::forms::drivers::urlencoded_< RequestT::body_type::value_type::const_iterator> (Template Struct urlencoded_)`
- `private udho::forms::drivers::multipart_< RequestT::body_type::value_type::const_iterator> (Template Struct multipart_)`

Struct Documentation

```
template<typename RequestT>
struct udho::forms::drivers::combo : private udho::forms::drivers::urlencoded_<RequestT::body_type::value_type::const_iterator>
```

Public Types

```
enum types
    Values:
        enumerator unparsed
        enumerator urlencoded
        enumerator multipart
typedef RequestT request_type
typedef urlencoded_<typename request_type::body_type::value_type::const_iterator> urlencoded_type
typedef multipart_<typename request_type::body_type::value_type::const_iterator> multipart_type
typedef request_type::body_type::value_type body_type
```

Public Functions

```

combo (const request_type &request)
void parse_urlencoded ()
    parse the beast request body as urlencoded form data
void parse_multipart ()
    parse the beast request body as multipart form data
bool is_urlencoded () const
    check whether the submitted form is urlencoded
bool is_multipart () const
    check whether the submitted form is multipart
const urlencoded_type &urlencoded () const
const multipart_type &multipart () const
    return the multipart specific form accessor
bool empty (const std::string name) const
    checks whether the value for the field is empty
bool exists (const std::string name) const
    checks whether there exists any field with the name provided
template<typename T, typename ParserT = udho::forms::parser<T>>
bool parsable (const std::string &name) const
template<typename T, typename ParserT = udho::forms::parser<T>>
const T parsed (const std::string &name) const
    returns the value of the field with the name provided lexically casted to type T

```

Public Members

```

const request_type &_request
types _type

```

Template Struct `extract_multipart_boundary`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```

template<typename RequestT>
struct udho::forms::drivers::detail::extract_multipart_boundary

```

Public Types

```
typedef RequestT request_type
```

Public Functions

```
extract_multipart_boundary (const request_type &request)  
const std::string &boundary () const  
bool extract (const std::string &boundary_key = "boundary=")
```

Public Members

```
request_type _request  
std::string _boundary
```

Template Struct multipart

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::drivers::multipart_< RequestT::body_type::value_type::const_iterator
> (Template Struct multipart_)`

Struct Documentation

```
template<typename RequestT>  
struct udho::forms::drivers::multipart : public udho::forms::drivers::multipart_<RequestT::body_type::value_type
```

Public Types

```
typedef RequestT request_type  
typedef request_type::body_type::value_type body_type  
typedef multipart_<typename request_type::body_type::value_type::const_iterator> multipart_type
```

Public Functions

multipart (**const** *request_type* &*request*)

Public Members

const *request_type* &**_request**

Template Struct **multipart_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Nested Relationships

Nested Types

- *Struct* `multipart_::form_part`

Struct Documentation

```
template<typename Iterator = std::string::const_iterator>
struct udho::forms::drivers::multipart_
    Form accessor for multipart forms
```

Public Types

```
typedef Iterator iterator_type
typedef std::iterator_traits<iterator_type>::value_type value_type
typedef std::basic_string<value_type> string_type
typedef bounded_str<iterator_type> bounded_string_type
typedef std::map<string_type, bounded_string_type> header_map_type
typedef bounded_string_type bounded_string
```

Public Functions

```
void parse (const std::string &boundary, iterator_type begin, iterator_type end)
void parse_part (iterator_type begin, iterator_type end)
std::size_t count () const
    number of fields in the form
bool exists (const std::string name) const
    checks whether the form has any field with the given name
bool empty (const std::string name) const
const form_part &part (const std::string &name) const
    returns the part associated with the given name
```

```
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
bool parsable (const std::string &name, const ArgsT&... args) const

template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
const T parsed (const std::string &name, const ArgsT&... args) const
    returns the value of the field with the name provided lexically casted to type T
```

Public Members

```
std::string _boundary
std::map<string_type, form_part> _parts

struct form_part
    A part in the multipart form data
```

Public Functions

```
form_part (iterator_type begin, iterator_type end)

void set_header (const header_map_type &headers)

const bounded_string_type &header (const std::string &key) const
    returns the value associated with the key in the header of the part
```

```
header ("Content-Disposition").copied<std::string>();
```

```
bounded_string_type header (const std::string &key, const std::string &sub) const
    returns the value associated with the key and sub key in the header of the part
```

```
header ("Content-Disposition", "name").copied<std::string>();
```

```
bounded_string_type name () const
    name of the part
```

```
name().copied<std::string>();
```

```
bounded_string_type filename () const
    filename of the part
```

```
name().copied<std::string>();
```

```
const bounded_string_type &body () const
    body of the part returned as a pair of string iterators
```

```
body().copied<std::string>();
```

```
template<typename StrT>
StrT copied () const
```

```
std::string str () const
    returns the body of the part as string
```

```
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
bool parsable (const ArgsT&... args) const
```

```
bool empty () const
```

```
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
const T parsed(const ArgsT&... args) const
    returns the value of the field with the name provided lexically casted to type T
```

Public Members

```
header_map_type _headers
bounded_string_type _body
```

Struct `multipart_::form_part`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Nested Relationships

This struct is a nested type of *Template Struct multipart_*.

Struct Documentation

```
struct udho::forms::drivers::multipart_::form_part
    A part in the multipart form data
```

Public Functions

```
form_part (iterator_type begin, iterator_type end)
```

```
void set_header (const header_map_type &headers)
```

```
const bounded_string_type &header (const std::string &key) const
    returns the value associated with the key in the header of the part
```

```
header("Content-Disposition").copied<std::string>();
```

```
bounded_string_type header (const std::string &key, const std::string &sub) const
    returns the value associated with the key and sub key in the header of the part
```

```
header("Content-Disposition", "name").copied<std::string>();
```

```
bounded_string_type name () const
    name of the part
```

```
name().copied<std::string>();
```

```
bounded_string_type filename () const
    filename of the part
```

```
name().copied<std::string>();
```

```
const bounded_string_type &body () const
    body of the part returned as a pair of string iterators
```

```
body().copied<std::string>();
```

```
template<typename StrT>
```

```
StrT copied() const
```

```
std::string str() const
```

```
returns the body of the part as string
```

```
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
```

```
bool parsable(const ArgsT&... args) const
```

```
bool empty() const
```

```
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>
```

```
const T parsed(const ArgsT&... args) const
```

```
returns the value of the field with the name provided lexically casted to type T
```

Public Members

```
header_map_type _headers
```

```
bounded_string_type _body
```

Template Struct urlencoded

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::drivers::urlencoded_< RequestT::body_type::value_type::const_iterator> (Template Struct urlencoded_)`

Struct Documentation

```
template<typename RequestT>
```

```
struct udho::forms::drivers::urlencoded: public udho::forms::drivers::urlencoded_<RequestT::body_type::value_
```

Public Types

```
typedef RequestT request_type
```

```
typedef request_type::body_type::value_type body_type
```

```
typedef urlencoded_<typename request_type::body_type::value_type::const_iterator> urlencoded_type
```

Public Functions

`urlencoded` (`const` *request_type* &*request*)

Public Members

`const` *request_type* &`_request`

Template Struct `urlencoded_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Derived Type

- `public` `udho::url` (*Struct* *url*)

Struct Documentation

template<typename **Iterator** = `std::string::const_iterator`>

struct `udho::forms::drivers::urlencoded_`

Form driver for urlencoded forms

Subclassed by *udho::url*

Public Types

typedef *Iterator* **iterator_type**

typedef `std::iterator_traits<iterator_type>::value_type` **value_type**

typedef `std::basic_string<value_type>` **string_type**

typedef *bounded_str*<*iterator_type*> **bounded_string_type**

typedef `std::map<string_type, bounded_string_type>` **header_map_type**

typedef *bounded_string_type* **bounded_string**

Public Functions

`void` **parse** (*iterator_type* *begin*, *iterator_type* *end*)

`bool` **empty** (`const` `std::string` &*name*) `const`
checks whether the value for the field is empty

`bool` **exists** (`const` `std::string` &*name*) `const`
checks whether there exists any field with the name provided

template<typename **T**, typename **ParserT** = `udho::forms::parser<T>`, typename ...**ArgsT**>

`bool` **parsable** (`const` `std::string` &*name*, `const` *ArgsT*&... *args*) `const`

template<typename **T**, typename **ParserT** = `udho::forms::parser<T>`, typename ...**ArgsT**>

```
const T parsed (const std::string &name, const ArgsT&... args) const
    returns the value of the field with the name provided lexically casted to type T
```

Public Members

```
header_map_type _fields
std::string _query
```

Template Struct field

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename T, bool Required = false>
struct field
```

Template Struct field< std::chrono::time_point< std::chrono::system_clock, DurationT >, false >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::basic_field< std::chrono::time_point< std::chrono::system_clock, DurationT >, field< std::chrono::time_point< std::chrono::system_clock, DurationT >, false > > (Template Struct basic_field)`

Struct Documentation

```
template<typename DurationT>
struct udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false> : public udho::forms::
```

Public Types

```
enum [anonymous]
    Values:
```

```
    enumerator is_required
```

```
typedef std::chrono::time_point<std::chrono::system_clock, DurationT> value_type
```

```
typedef field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false> self_type
```

```
typedef detail::basic_field<std::chrono::time_point<std::chrono::system_clock, DurationT>, field<std::chrono::time_point<std::
```

Public Functions

```
field(const std::string &name, const std::string &format = default_datetime_format, const
      value_type &def = value_type())
const value_type &def() const
template<typename FormT>
bool fetch(const FormT &form)
template<typename FormT, typename ValidatorT>
bool validate(const FormT &form, const ValidatorT &validator)
```

Public Members

```
std::string _format
```

Protected Attributes

```
value_type _def
```

Template Struct field< std::chrono::time_point< std::chrono::system_clock, DurationT >, true >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::basic_field< std::chrono::time_point< std::chrono::system_clock, DurationT >, field< std::chrono::time_point< std::chrono::system_clock, DurationT >, true > > (Template Struct basic_field)`

Struct Documentation

```
template<typename DurationT>
struct udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true> : public udho::forms::
```

Public Types

```
enum [anonymous]
```

Values:

```
enumerator is_required
```

```
typedef std::chrono::time_point<std::chrono::system_clock, DurationT> value_type
```

```
typedef field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true> self_type
```

```
typedef detail::basic_field<std::chrono::time_point<std::chrono::system_clock, DurationT>, field<std::chrono::time_point<st
```

Public Functions

```
field (const std::string &name, const std::string &format = default_datetime_format)  
  
template<typename FormT>  
bool fetch (const FormT &form)  
  
template<typename FormT, typename ValidatorT>  
bool validate (const FormT &form, const ValidatorT &validator)
```

Public Members

```
std::string _format
```

Template Struct `field< T, false >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::basic_field< T, field< T, false > >` (*Template Struct basic_field*)

Struct Documentation

```
template<typename T>  
struct udho::forms::field<T, false> : public udho::forms::detail::basic_field<T, field<T, false>>
```

Public Types

```
enum [anonymous]  
    Values:  
    enumerator is_required  
  
typedef T value_type  
  
typedef field<T, false> self_type  
  
typedef detail::basic_field<T, field<T, false>> base
```

Public Functions

```
field(const std::string &name, const value_type &def = value_type())
const value_type &def() const
template<typename FormT>
bool fetch(const FormT &form)
template<typename FormT, typename ValidatorT>
bool validate(const FormT &form, const ValidatorT &validator)
```

Protected Attributes

```
value_type _def
```

Template Struct field< T, true >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::detail::basic_field< T, field< T, true > >` (*Template Struct basic_field*)

Struct Documentation

```
template<typename T>
struct udho::forms::field<T, true> : public udho::forms::detail::basic_field<T, field<T, true>>
```

Public Types

```
enum [anonymous]
    Values:
        enumerator is_required
typedef T value_type
typedef field<T, true> self_type
typedef detail::basic_field<T, field<T, true>> base
```

Public Functions

```
field (const std::string &name)  
  
template<typename FormT>  
bool fetch (const FormT &form)  
  
template<typename FormT, typename ValidatorT>  
bool validate (const FormT &form, const ValidatorT &validator)
```

Template Struct form

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public DriverT`

Struct Documentation

```
template<typename DriverT>  
struct udho::forms::form: public DriverT
```

Public Functions

```
bool has (const std::string &name) const  
  
template<typename T, typename ParserT = udho::forms::parser<T>>  
T field (const std::string &name, bool *ok = 0x0) const  
  
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>  
T field (const std::string &name, bool *ok, const ArgsT&... args) const  
  
template<typename T, typename ParserT = udho::forms::parser<T>, typename ...ArgsT>  
T field (const std::string &name, const ArgsT&... args) const
```

Template Struct parser

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename T>
struct udho::forms::parser
```

Public Static Functions

```
template<typename U, typename ...ArgsT>
bool parsable (const U &input, const ArgsT&... args)

template<typename U, typename ...ArgsT>
T parse (const U &input, const ArgsT&... args)
```

Template Struct validated

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Struct Documentation

```
template<typename FormT>
struct validated
```

Template Struct validated< form< DriverT > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Inheritance Relationships

Base Type

- `public udho::forms::accumulated` (*Struct accumulated*)

Struct Documentation

```
template<typename DriverT>
struct udho::forms::validated<form<DriverT>> : public udho::forms::accumulated
```

Public Types

```
typedef form<DriverT> form_type
typedef validated<form<DriverT>> self_type
```

Public Functions

validated (*form_type* &*f*)

```
template<typename ValidatorT, typename TailT>  
self_type &add(detail::constrained_field<ValidatorT, TailT> &cf)
```

```
template<typename T, bool Required>  
self_type &add(field<T, Required> &f)
```

Template Struct `actual_callback_type`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename C>  
struct udho::internal::actual_callback_type
```

Public Types

```
typedef C callback_type
```

Template Struct `arg_to_tuple`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename TupleT, int Index = boost::tuples::length<TupleT>::value - 1>  
struct udho::internal::arg_to_tuple
```

Public Types

```
typedef arg_to_tuple<TupleT, Index - 1> base_type
```

Public Static Functions

```
void convert (TupleT &tuple, const std::vector<std::string> &args)
```

Template Struct `arg_to_tuple< TupleT, 0 >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename TupleT>
struct udho::internal::arg_to_tuple<TupleT, 0>
```

Public Static Functions

```
void convert (TupleT&, const std::vector<std::string>&)
```

Template Struct `bind_first`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename T>
struct bind_first
```

Template Struct `bind_first< boost::function< R(Args...) > >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename R, typename ...Args>
struct udho::internal::bind_first<boost::function<R (Args...) >>
```

Public Types

```
typedef reducer<R, Args...> reducer_type
typedef boost::function<R (Args...) > base_function_type
typedef reducer_type::object_type object_type
typedef reducer_type::function_type reduced_function_type
```

Public Functions

bind_first (*object_type* *that*, *base_function_type* *base*)
reduced_function_type **reduced** ()

Public Members

object_type **_that**
reducer_type **_reducer**

Template Struct callable1

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename A1, typename R, typename ...V>  
struct callable1
```

Template Struct callable1< A1, R(*) (A1 &, V...)>

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename A1, typename R, typename ...V>  
struct udho::internal::callable1<A1, R (*) (A1&, V...)>
```

Public Types

```
typedef actual_callback_type<R (*) (A1&, V...)>::callback_type callback_type  
typedef boost::function<R (V...)> function_type  
typedef R return_type
```

Public Functions

```
callable1 (callback_type cb, A1 &a1)  
return_type operator () (V... args)
```

Public Members

callback_type **_callback**

A1 & **_a1**

Template Struct `cast_optionally`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename T, int Index>
struct udho::internal::cast_optionally
```

Public Static Functions

T **cast** (**const** std::vector<std::string> &*args*)

Template Struct `function_signature`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename T>
struct function_signature
    extract the function signature
```

Template Struct `function_signature< boost::function< R(Args...) > >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename R, typename ...Args>
struct udho::internal::function_signature<boost::function<R (Args...) >>
```

Public Types

typedef *R* **return_type**

typedef boost::tuple<std::decay_t<*Args*>...> **tuple_type**

typedef boost::fusion::tuple<std::decay_t<*Args*>...> **arguments_type**

Template Struct `function_signature< R(*) (Args...) >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename R, typename ...Args>
struct udho::internal::function_signature<R (*) (Args...) >
```

Public Types

```
typedef R return_type
typedef boost::tuple<std::decay_t<Args>...> tuple_type
typedef boost::fusion::tuple<std::decay_t<Args>...> arguments_type
```

Template Struct `member_`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename T>
struct member_
```

Template Struct `member_< R C::* >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename R, typename C>
struct udho::internal::member_<R C::*>
```

Public Types

```
typedef R result_type
typedef C that_type
RC::* actual_function_type
typedef boost::function<R (const C*)> const_base_function_type
typedef bind_first<const_base_function_type> const_binder_type
typedef const_binder_type::object_type object_type
typedef const_binder_type::reduced_function_type const_reduced_function_type
```

Public Functions

member_ (actual_function_type *function*)
const_reduced_function_type **reduced** (const *C* **that*)

Public Members

const_base_function_type **_function**

Template Struct **reduced_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename T>
struct reduced_
```

Template Struct **reduced_<R(C::*)(Args...) const >**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename R, typename C, typename ...Args>
struct udho::internal::reduced_<R (C::*) (Args...) const>
```

Public Types

```
typedef R result_type
typedef C that_type
typedef R (C::* actual_function_type) (Args...) const
typedef boost::function<R (const C*, Args...) > base_function_type
typedef bind_first<base_function_type> binder_type
typedef const binder_type::object_type object_type
typedef binder_type::reduced_function_type reduced_function_type
```

Public Functions

reduced_ (*actual_function_type* *function*)
reduced_function_type **reduced** (**const** *C* **that*)

Public Members

base_function_type **_function**

Template Struct **reduced_< R(C::*)(Args...)>**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename R, typename C, typename ...Args>  
struct udho::internal::reduced_<R (C::*) (Args...) >
```

Public Types

```
typedef R result_type  
typedef C that_type  
typedef R (C::* actual_function_type) (Args...)  
typedef boost::function<R (C*, Args...) > base_function_type  
typedef bind_first<base_function_type> binder_type  
typedef binder_type::object_type object_type  
typedef binder_type::reduced_function_type reduced_function_type
```

Public Functions

reduced_ (*actual_function_type* *function*)
reduced_function_type **reduced** (*C* **that*)

Public Members

base_function_type **_function**

Template Struct reducer

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
template<typename R, typename First, typename ...Rest>
struct udho::internal::reducer
```

Public Types

```
typedef First object_type
typedef boost::function<R (Rest...)> function_type
typedef boost::function<R (First, Rest...)> actual_function_type
```

Public Functions

```
reducer (object_type that, actual_function_type actual)
R operator() (const Rest&... rest)
```

Public Members

```
object_type _that
actual_function_type _actual
```

Template Struct is_application

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename U>
struct udho::is_application
```

Public Types

```
enum [anonymous]
  Values:
  enumerator value
```

Public Static Functions

```
template<typename V>  
V::application_type test (int)  
  
template<typename>  
void test (...)
```

Template Struct plain

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Struct Documentation

```
template<typename StreamT>  
struct udho::loggers::plain
```

Public Types

```
typedef StreamT stream_type  
typedef plain<StreamT> self_type
```

Public Functions

```
plain (StreamT &stream)  
  
void log (const std::chrono::system_clock::time_point &time, udho::logging::status status, const  
        std::string &segment, int level, const std::string &message)  
  
self_type &operator+= (const logging::features::colored &colored)  
  
template<udho::logging::status Status>  
self_type &operator () (const udho::logging::message<Status> &msg)
```

Public Members

```
StreamT &_stream  
  
logging::features::colored _feature_color
```

Struct colored

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Struct Documentation

struct udho::logging::features::colored

Public Functions

colored()

colored(bool *c*)

Public Members

bool **_colored**

Template Struct message

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Inheritance Relationships

Derived Type

- `public udho::logging::messages::formatted::message< Status >` (*Template Struct message*)

Struct Documentation

template<udho::logging::*status* Status>

struct udho::logging::message

A logging message

Template Parameters

- Status:

Subclassed by `udho::logging::messages::formatted::message< Status >`

Public Types

typedef std::chrono::system_clock::time_point **time_type**

Public Functions

```
message (const std::string &segment, unsigned level = 0)
message (const std::string &segment, const std::string &content, unsigned level = 0)
std::string what () const
```

Public Members

```
const time_type time = std::chrono::system_clock::now()
const std::string _segment
const unsigned _level
std::string _content
```

Public Static Attributes

```
constexpr const udho::logging::status status = Status
```

Template Struct message

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Inheritance Relationships

Base Type

- `public udho::logging::message< Status >` (*Template Struct message*)

Struct Documentation

```
template<udho::logging::status Status>
struct udho::logging::messages::formatted::message : public udho::logging::message<Status>
```

Public Types

```
typedef messages::formatted::message<Status> self_type
typedef udho::logging::message<Status> base_type
```

Public Functions

```
message (const std::string &segment, const std::string &format, unsigned level = 0)
template<typename T>
self_type &prepare (const T &value)
template<typename T>
self_type &operator% (const T &value)
std::string what () const
```

Public Members

```
boost::format _format
```

Template Struct lookup_table

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_scope.h`

Struct Documentation

```
template<typename DataT>
struct lookup_table
```

Accepts a prepared object that can be queried through string keys. Maintains a lookup table which contains aliases of such keys. So that the keys can be with both the real name as well as through the alias names. An unsigned integer depth value associated with every aliases. Based on which all aliases of some depth can be cleared. The up and down increases and decreases the value of depth. add will add an alias with the current depth. It can be used in a scoped environment where the entering to and leaving from an inner scope will call the down and up functions respectively. Calling up will clear the aliases set in the depth below

Template Struct lookup_table< udho::prepared< DataT > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_scope.h`

Struct Documentation

```
template<typename DataT>
struct udho::lookup_table<udho::prepared<DataT>>
```

Public Types

```
typedef udho::prepared<DataT> prepared_type
typedef boost::multi_index_container<declaration, boost::multi_index::indexed_by<boost::multi_index::ordered_unique<bo
typedef storage_type::nth_index<1>::type named_storage_type
typedef storage_type::nth_index<2>::type depth_storage_type
```

Public Functions

```
lookup_table (const prepared_type &p)
bool valid (const std::string &key) const
std::string lookup (const std::string &key) const
bool add (const std::string &key, const std::string &ref, std::size_t depth)
std::size_t clear (std::size_t depth)
std::string eval (const std::string &key) const
std::string operator [] (const std::string &key) const
template<typename T>
T extract (const std::string &key) const
template<typename T>
T parse (const std::string &key) const
std::vector<std::string> keys (const std::string &key) const
std::size_t count (const std::string &key) const
std::size_t down ()
std::size_t up ()
bool add (const std::string &key, const std::string &ref)
```

Public Members

```
const prepared_type &_prepared
storage_type _storage
named_storage_type &_storage_named
depth_storage_type &_storage_depth
std::size_t _level
```

Template Struct lookup_table< udho::prepared_group< U, V > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_scope.h`

Struct Documentation

```
template<typename U, typename V>
struct udho::lookup_table<udho::prepared_group<U, V>>
```

Public Types

```

typedef udho::prepared_group<U, V> prepared_type
typedef boost::multi_index_container<declaration, boost::multi_index::indexed_by<boost::multi_index::ordered_unique<bo
typedef storage_type::nth_index<1>::type named_storage_type
typedef storage_type::nth_index<2>::type depth_storage_type

```

Public Functions

```

lookup_table (const prepared_type &p)
bool valid (const std::string &key) const
std::string lookup (const std::string &key) const
bool add (const std::string &key, const std::string &ref, std::size_t depth)
std::size_t clear (std::size_t depth)
std::string eval (const std::string &key) const
std::string operator[] (const std::string &key) const
template<typename T>
T extract (const std::string &key) const
template<typename T>
T parse (const std::string &key) const
std::vector<std::string> keys (const std::string &key) const
std::size_t count (const std::string &key) const
std::size_t down ()
std::size_t up ()
bool add (const std::string &key, const std::string &ref)

```

Public Members

```

const prepared_type &_prepared
storage_type _storage
named_storage_type &_storage_named
depth_storage_type &_storage_depth
std::size_t _level

```

Template Struct `module_evaluator`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename VisitorT, typename ModuleT, bool IsApplication = is_application<ModuleT>::value>
struct module_evaluator
```

Template Struct `module_evaluator< VisitorT, ModuleT, false >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename VisitorT, typename ModuleT>
struct udho::module_evaluator<VisitorT, ModuleT, false>
```

Public Functions

```
module_evaluator (VisitorT &visitor)
void operator() (ModuleT &modul)
```

Public Members

```
VisitorT &_visitor
```

Template Struct `module_evaluator< VisitorT, ModuleT, true >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename VisitorT, typename ModuleT>
struct udho::module_evaluator<VisitorT, ModuleT, true>
```

Public Functions

```
module_evaluator (VisitorT &visitor)
void operator() (ModuleT &app)
```

Public Members

VisitorT & **_visitor**

Struct module_info

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Struct Documentation

```
struct udho::module_info
```

Public Members

```
boost::beast::http::verb _method
std::string _pattern
std::string _compositor
const void *_fptr
std::vector<module_info> _children
```

Template Struct module_overload

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename Function, template<typename> class CompositorT = compositors::transparent>
struct udho::module_overload
```

Public Types

```
typedef Function function_type
typedef module_overload<Function, CompositorT> self_type
typedef internal::function_signature<Function>::return_type return_type
typedef internal::function_signature<Function>::tuple_type tuple_type
typedef internal::function_signature<Function>::arguments_type arguments_type
typedef CompositorT<return_type> compositor_type
typedef compositor_type::response_type response_type
```

Public Functions

```
module_overload (boost::beast::http::verb request_method, function_type f, compositor_type compositor = compositor_type())  
module_overload (const self_type &other)  
std::string pattern () const  
self_type &operator= (const std::string &pattern)  
bool feasible (boost::beast::http::verb request_method, const std::string &subject) const  
    check number of arguments supplied on runtime and number of arguments with which this overload has  
    been prepared at compile time.  
  
template<typename T>  
return_type call (T &value, const std::vector<std::string> &args)  
  
template<typename T>  
response_type operator () (T &value, const std::vector<std::string> &args)  
  
template<typename T>  
response_type operator () (T &value, const std::string &subject)  
  
module_info info () const
```

Public Members

```
boost::beast::http::verb _request_method  
std::string _pattern  
function_type _function  
compositor_type _compositor
```

Template Struct `module_overload< Function, compositors::deferred >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename Function>  
struct udho::module_overload<Function, compositors::deferred>
```

Public Types

```
typedef Function function_type  
typedef module_overload<Function, compositors::deferred> self_type  
typedef internal::function_signature<Function>::return_type return_type  
typedef internal::function_signature<Function>::tuple_type tuple_type  
typedef internal::function_signature<Function>::arguments_type arguments_type  
typedef compositors::deferred<return_type> compositor_type  
typedef compositor_type::response_type response_type
```

Public Functions

```

module_overload (boost::beast::http::verb request_method, function_type f, compositor_type compositor = compositor_type())
module_overload (const self_type &other)
std::string pattern () const
self_type &operator= (const std::string &pattern)
bool feasible (boost::beast::http::verb request_method, const std::string &subject) const
    check number of arguments supplied on runtime and number of arguments with which this overload has
    been prepared at compile time.

template<typename T>
void call (T &value, const std::vector<std::string> &args)

template<typename T>
void operator () (T &value, const std::vector<std::string> &args)

template<typename T>
void operator () (T &value, const std::string &subject)

module_info info () const

```

Public Members

```

boost::beast::http::verb _request_method
std::string _pattern
function_type _function
compositor_type _compositor

```

Template Struct multipart_form

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Nested Relationships

Nested Types

- *Struct multipart_form::form_part*

Struct Documentation

```

template<typename Iterator>
struct udho::multipart_form
    Form accessor for multipart forms

```

Public Types

```
typedef Iterator iterator_type
typedef std::iterator_traits<iterator_type>::value_type value_type
typedef std::basic_string<value_type> string_type
typedef bounded_str<iterator_type> bounded_string_type
typedef std::map<string_type, bounded_string_type> header_map_type
typedef bounded_string_type bounded_string
```

Public Functions

```
void parse (const std::string &boundary, iterator_type begin, iterator_type end)
void parse_part (iterator_type begin, iterator_type end)

std::size_t count () const
    number of fields in the form

bool has (const std::string name) const
    checks whether the form has any field with the given name

const form_part &part (const std::string &name) const
    returns the part associated with the given name

template<typename T>
const T field (const std::string &name) const
    returns the value of the field lexically casted with the desired type
```

```
form.field<int> ("age");
```

Public Members

```
std::string _boundary
std::map<string_type, form_part> _parts

struct form_part
    A part in the multipart form data
```

Public Functions

```
form_part (iterator_type begin, iterator_type end)
void set_header (const header_map_type &headers)

const bounded_string_type &header (const std::string &key) const
    returns the value associated with the key in the header of the part
```

```
header ("Content-Disposition").copied<std::string>();
```

```
bounded_string_type header (const std::string &key, const std::string &sub) const
    returns the value associated with the key and sub key in the header of the part
```

```
header("Content-Disposition", "name").copied<std::string>();
```

bounded_string_type **name()** **const**

name of the part

```
name().copied<std::string>();
```

bounded_string_type **filename()** **const**

filename of the part

```
name().copied<std::string>();
```

const *bounded_string_type* &**body()** **const**

body of the part returned as a pair of string iterators

```
body().copied<std::string>();
```

template<typename **StrT**>

StrT **copied()** **const**

std::string **str()** **const**

returns the body of the part as string

template<typename **T**>

T **value()** **const**

lexically convert the contents of the body to the requested type

Public Members

header_map_type **_headers**

bounded_string_type **_body**

Struct multipart_form::form_part

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Nested Relationships

This struct is a nested type of *Template Struct multipart_form*.

Struct Documentation

struct udho::multipart_form::form_part

A part in the multipart form data

Public Functions

form_part (iterator_type *begin*, iterator_type *end*)

void **set_header** (const header_map_type &*headers*)

const bounded_string_type &**header** (const std::string &*key*) const

returns the value associated with the key in the header of the part

```
header("Content-Disposition").copied<std::string>();
```

bounded_string_type **header** (const std::string &*key*, const std::string &*sub*) const

returns the value associated with the key and sub key in the header of the part

```
header("Content-Disposition", "name").copied<std::string>();
```

bounded_string_type **name** () const

name of the part

```
name().copied<std::string>();
```

bounded_string_type **filename** () const

filename of the part

```
name().copied<std::string>();
```

const bounded_string_type &**body** () const

body of the part returned as a pair of string iterators

```
body().copied<std::string>();
```

template<typename **StrT**>

StrT **copied** () const

std::string **str** () const

returns the body of the part as string

template<typename **T**>

T **value** () const

lexically convert the contents of the body to the requested type

Public Members

header_map_type **_headers**

bounded_string_type **_body**

Template Struct overload_group

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename U, typename V>
struct udho::overload_group
    compile time chain of url mappings
```

See [*content_wrapper0*](#)

See [*content_wrapper1*](#)

Public Types

```
typedef overload_group<U, V> self_type
    type of this overload
```

```
typedef U parent_type
    type of the parient in the meta-chain of overloads
```

```
typedef V overload_type
    type of the next child in the overload chain
```

```
typedef parent_type::terminal_type terminal_type
```

Public Functions

```
overload_group (const parent_type &parent, const overload_type &overload)
```

```
template<typename ContextT, typename Lambda>
int serve (ContextT &ctx, boost::beast::http::verb request_method, const std::string &subject, Lambda
    send)
    serves the content if the http request matches with the content's request method and path. Calls the callback
    in _overload if it is feasible for the current request. Otherwise bubbles the request meta-recursively if
    some other overload group is feasible.
```

Parameters

- req: the http request to serve
- request_method: http vmethod get post put head etc ...
- subject: http resource path
- send: the write callback

```
void summary (std::vector<module_info> &stack) const
```

```
template<typename AttachmentT>
self_type &listen (boost::asio::io_service &io, AttachmentT &attachment, int port = 9198)
```

```
template<typename F>
void eval (F &fnc)
```

```
const terminal_type &terminal () const
```

Public Members

parent_type **_parent**
overload_type **_overload**

Template Struct overload_group< U, app_< V, Ref > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h`

Struct Documentation

```
template<typename U, typename V, bool Ref>  
struct udho::overload_group<U, app_<V, Ref>>
```

Public Types

```
typedef overload_group<U, app_<V, Ref>> self_type  
typedef U parent_type  
typedef app_<V, Ref> overload_type  
typedef parent_type::terminal_type terminal_type
```

Public Functions

```
overload_group (const parent_type &parent, const overload_type &overload)  
  
template<typename ContextT, typename Lambda>  
int serve (ContextT &ctx, boost::beast::http::verb request_method, const std::string &subject, Lambda  
    send)  
  
void summary (std::vector<module_info> &stack) const  
  
template<typename AttachmentT>  
self_type &listen (boost::asio::io_service &io, AttachmentT &attachment, int port = 9198)  
  
template<typename F>  
void eval (F &fnc)  
  
const terminal_type &terminal () const
```

Public Members

parent_type **_parent**
overload_type **_overload**

Template Struct `overload_group< U, overload_terminal< V > >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename U, typename V>
struct udho::overload_group<U, overload_terminal<V>>
    terminal node of the compile time chain of url mappings

    See content_wrapper0
    See content_wrapper1
```

Public Types

```
typedef overload_group<U, overload_terminal<V>> self_type
    type of this overload

typedef V overload_type
    type of parent in the overload meta-chain

typedef U parent_type
    type of the next child in the overload chain

typedef overload_terminal<V> terminal_type
```

Public Functions

```
template<typename ...Args>
overload_group (Args... args)

template<typename ContextT, typename Lambda>
int serve (ContextT &ctx, boost::beast::http::verb request_method, const std::string &subject, Lambda
    send)

void summary (std::vector<module_info>&) const

template<typename F>
void eval (F &fnc)

const terminal_type &terminal () const
```

Public Members

```
terminal_type _terminal
```

Template Struct `overload_group< U, overload_terminal< void > >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename U>
struct udho::overload_group<U, overload_terminal<void>>
```

Public Types

```
typedef overload_group<U, overload_terminal<void>> self_type
    type of this overload
```

```
typedef void overload_type
    type of parent in the overload meta-chain
```

```
typedef U parent_type
    type of the next child in the overload chain
```

```
typedef overload_terminal<void> terminal_type
```

Public Functions

```
template<typename ...Args>
overload_group (Args...)

template<typename ContextT, typename Lambda>
int serve (ContextT&, boost::beast::http::verb, const std::string&, Lambda)

void summary (std::vector<module_info>&) const

template<typename F>
void eval (F&)

const terminal_type &terminal () const
```

Public Members

```
terminal_type _terminal
```

Template Struct `overload_group_helper`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename OverloadT, typename ResponseT = typename OverloadT::response_type>
struct udho::overload_group_helper
```

Public Types

```
typedef OverloadT overload_type
typedef overload_type::response_type response_type
```

Public Functions

```
overload_group_helper (OverloadT &overload)
template<typename ContextT, typename Lambda>
int resolve (ContextT &ctx, Lambda send, const std::string &subject)
```

Public Members

OverloadT &_overload

Template Struct overload_group_helper< OverloadT, void >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<typename OverloadT>
struct udho::overload_group_helper<OverloadT, void>
```

Public Types

```
typedef OverloadT overload_type
typedef void response_type
```

Public Functions

```
overload_group_helper (OverloadT &overload)
template<typename ContextT, typename Lambda>
int resolve (ContextT &ctx, Lambda send, const std::string &subject)
```

Public Members

OverloadT & **_overload**

Template Struct `overload_terminal`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Inheritance Relationships

Base Type

- `public AuxT`

Struct Documentation

```
template<typename AuxT = void>
struct udho::overload_terminal : public AuxT
```

Public Types

```
typedef AuxT aux_type
```

Template Struct `overload_terminal< void >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
template<>
struct udho::overload_terminal<void>
```

Public Types

```
typedef void aux_type
```

Public Functions

```
template<typename ...Args>
overload_terminal (Args...)
```

Template Struct prepare

- Defined in file `__home_docs_checkout_readthedocs.org_user_builds_udho_checkout_latest_includes_udho_access.h`

Struct Documentation

```
template<typename DerivedT>
struct udho::prepare
```

Public Types

```
typedef DerivedT prepared_type
```

Public Functions

```
template<typename F>
auto var (const std::string &key, F f)

template<typename F>
auto var (const std::string &key, F f) const

template<typename F>
auto fn (const std::string &key, F f)

template<typename F>
auto fn (const std::string &key, F f) const

auto index () const
```

Template Struct prepared

- Defined in file `__home_docs_checkout_readthedocs.org_user_builds_udho_checkout_latest_includes_udho_access.h`

Struct Documentation

```
template<typename T, bool IsPrepared = detail::is_prepared<T>::value>
struct prepared
```

`prepared<T>` keeps a copy of the string based index returned by the `T::index()` function and const references the actual data object of type `T`. It also instantiates a visitor for traversing that data object and extracting the values.

Template Struct prepared< T, true >

- Defined in file `__home_docs_checkout_readthedocs.org_user_builds_udho_checkout_latest_includes_udho_access.h`

Struct Documentation

```
template<typename T>
struct udho::prepared<T, true>
```

Public Types

```
typedef T derived_type
typedef std::result_of<decltype(&derived_type::index) (const derived_type*) >::type
                                     index_type
typedef detail::association_group_visitor<index_type> visitor_type
```

Public Functions

```
prepared (const derived_type &data)
prepared (const prepared<T, true> &other)
    a visitor keeps a reference to the index, So while copy constructing if the other object is going to be destruc-
    ted then the copied visitor will have a dangling reference to the other._index hence the copy constructor
    instantiates a visitor again from the copied index

template<typename V>
V extract (const std::string &key, bool *okay = 0x0) const

template<typename V>
V parse (const std::string &key, bool *okay = 0x0) const

std::string stringify (const std::string &key, bool *okay = 0x0) const

template<typename V>
V at (const std::string &key, bool *okay = 0x0) const

std::string operator[] (const std::string &key) const

std::size_t count (const std::string &key) const

std::vector<std::string> keys (const std::string &key) const
```

Public Members

```
const derived_type &_data
index_type _index
visitor_type _visitor
```

Template Struct prepared_group

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

template<typename **H**, typename **T** = void>

struct udho::prepared_group

Fold expression containing multiple prepared objects. Head is always type of `udho::prepared<X>` where X subclasses from `udho::prepare<X>` Tail is another *prepared_group* The actual prepared object is copied so it works even if the actual prepared object is an rvalue

Public Types

typedef *H* head_type

typedef *T* tail_type

typedef *prepared_group*<*H*, *T*> self_type

Public Functions

prepared_group (**const** *head_type* &head, **const** *tail_type* &tail)

template<typename **V**>

V **extract** (**const** std::string &key) **const**

template<typename **V**>

V **parse** (**const** std::string &key) **const**

std::string **stringify** (**const** std::string &key) **const**

template<typename **V**>

V **at** (**const** std::string &key) **const**

std::string **operator[]** (**const** std::string &key) **const**

std::size_t **count** (**const** std::string &key) **const**

std::vector<std::string> **keys** (**const** std::string &key) **const**

Public Members

head_type _head

tail_type _tail

Template Struct prepared_group< H, void >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Struct Documentation

```
template<typename H>
struct udho::prepared_group<H, void>
```

Public Types

```
typedef H head_type
typedef prepared_group<H, void> self_type
```

Public Functions

```
prepared_group (const head_type &head)
template<typename V>
V extract (const std::string &key) const
template<typename V>
V parse (const std::string &key) const
std::string stringify (const std::string &key) const
template<typename V>
V at (const std::string &key) const
std::string operator[] (const std::string &key) const
std::size_t count (const std::string &key) const
std::vector<std::string> keys (const std::string &key) const
```

Public Members

```
head_type _head
```

Template Struct proxy

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Struct Documentation

```
template<typename K, typename C>
struct udho::proxy
```

Public Types

```
typedef K key_type
typedef C config_type
typedef proxy<K, C> proxy_type
```

Public Functions

```
proxy (config_type &conf)
value_type value () const
operator value_type () const
template<typename V>
proxy_type &operator= (V value)
```

Public Members

```
decltype(C().get(K())) typedef value_type
config_type &_config
```

Template Struct proxy< configs::server_<>::mimes_t, config< configs::server_<> > >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Struct Documentation

```
template<>
struct udho::proxy<configs::server_<>::mimes_t, config<configs::server_<>>>
```

Public Types

```
typedef configs::server_<>::mimes_t key_type
typedef config<configs::server_<>> config_type
typedef proxy<key_type, config_type> proxy_type
```

Public Functions

```
proxy (config_type &conf)  
value_type value () const  
operator value_type () const  
template<typename V>  
proxy_type &operator= (V value)  
std::string at (const std::string &key) const  
std::string of (const std::string &key) const  
void set (const std::string &ext, const std::string &mime)
```

Public Members

```
decltype(config_type() .get(key_type())) typedef value_type  
config_type &_config
```

Struct reroute

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Struct Documentation

```
struct udho::reroute
```

Public Types

```
boost::function< boost::beast::http::response< boost::beast::http::string_body >udho::
```

Public Functions

```
reroute (const std::string &alt_path)  
std::string alt_path () const  
boost::beast::http::response<boost::beast::http::string_body> operator () (udho::contexts::stateless  
ctx)
```

Public Members

```
std::string _alt_path
```

Template Struct server

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Struct Documentation

```
template<typename AuxT, typename LoggerT = void, typename CacheT = void>
struct udho::server
```

Todo:
write docs

Public Types

```
typedef LoggerT logger_type
typedef CacheT cache_type
typedef AuxT auxiliary_type
typedef udho::attachment<auxiliary_type, logger_type, cache_type> attachment_type
typedef server<auxiliary_type, logger_type, cache_type> self_type
typedef attachment_type::configuration_type configuration_type
using context_type = udho::context<auxiliary_type, RequestT, attachment_type>
typedef udho::defs::request_type http_request_type
typedef http_request_type request_type
```

Public Functions

```
server (boost::asio::io_service &io, logger_type &logger)
server (const self_type&) = delete
server (self_type &&other) = default

template<typename RouterT>
void serve (RouterT &&router, int port = 9198)

template<typename FeatureT>
auto operator += (const FeatureT &feature)
configuration_type &config()
const configuration_type &config() const

template<typename ConfigKeyT>
auto operator [] (const ConfigKeyT &k)

template<typename ConfigKeyT>
auto operator [] (const ConfigKeyT &k) const
```

Public Members

```
boost::asio::io_service &_io  
attachment_type _attachment
```

Template Struct server< AuxT, void, CacheT >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Struct Documentation

```
template<typename AuxT, typename CacheT>  
struct udho::server<AuxT, void, CacheT>
```

Public Types

```
typedef CacheT cache_type  
typedef AuxT auxiliary_type  
typedef udho::attachment<auxiliary_type, void, cache_type> attachment_type  
typedef server<auxiliary_type, void, cache_type> self_type  
typedef attachment_type::configuration_type configuration_type  
using context_type = udho::context<auxiliary_type, RequestT, attachment_type>  
typedef udho::defs::request_type http_request_type  
typedef context_type<http_request_type> context  
typedef http_request_type request_type
```

Public Functions

```
server (boost::asio::io_service &io)  
server (const self_type&) = delete  
server (self_type &&other) = default  
  
template<typename RouterT>  
void serve (RouterT &&router, int port = 9198)  
  
template<typename FeatureT>  
auto operator += (const FeatureT &feature)  
configuration_type &config ()  
  
const configuration_type &config () const  
  
template<typename ConfigKeyT>  
auto operator [] (const ConfigKeyT &k)  
  
template<typename ConfigKeyT>  
auto operator [] (const ConfigKeyT &k) const
```

Public Members

boost::asio::io_service &**_io**
attachment_type **_attachment**

Template Struct logged

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Struct Documentation

template<typename **T**>
struct udho::servers::logged

Public Types

typedef *T* **logger_type**
using **stateful** = *server*<udho::bridge<udho::configuration_type>, *logger_type*, udho::cache::store<StorageT, boost::uuids::uuid>>
using **stateless** = *server*<udho::bridge<udho::configuration_type>, *logger_type*, void>

Template Struct ostreamed_helper

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Struct Documentation

template<typename **CacheT**>
struct udho::servers::ostreamed::ostreamed_helper

Public Types

typedef *server*<udho::bridge<udho::configuration_type>, udho::loggers::ostream, *CacheT*> **server_type**
typedef *server_type*::logger_type **logger_type**
typedef *server_type*::cache_type **cache_type**
typedef *server_type*::attachment_type **attachment_type**
typedef *server_type*::http_request_type **request_type**

Public Functions

```
ostreamed_helper (boost::asio::io_service &io, udho::loggers::ostream::stream_type &stream)  
  
template<typename RouterT>  
void serve (RouterT &&router, int port = 9198)  
  
template<typename FeatureT>  
auto operator+= (const FeatureT &feature)  
  
server_type::configuration_type &config ()  
  
const server_type::configuration_type &config () const  
  
template<typename ConfigKeyT>  
auto operator[] (const ConfigKeyT &k)  
  
template<typename ConfigKeyT>  
auto operator[] (const ConfigKeyT &k) const
```

Public Members

```
udho::loggers::ostream _logger  
  
server_type _server
```

Template Struct stateful

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Struct Documentation

```
template<template<typename, typename> class StorageT, typename ...T>  
struct udho::servers::stateful
```

Public Types

```
typedef udho::cache::store<StorageT, boost::uuids::uuid, T...> cache_type  
  
using logged = server<udho::bridge<udho::configuration_type>, LoggerT, cache_type>  
  
using ostreamed = logged<udho::loggers::ostream>  
  
using quiet = server<udho::bridge<udho::configuration_type>, void, cache_type>
```

Template Struct session_

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_session.h`

Struct Documentation

```
template<typename RequestT, typename ShadowT>
struct udho::session_
```

Public Types

```
typedef RequestT request_type
typedef udho::cookies_<request_type> cookies_type
typedef ShadowT shadow_type
typedef shadow_type::key_type key_type
typedef session_<request_type, shadow_type> self_type
typedef udho::cache::generator<key_type> generator_type
typedef udho::config<udho::configs::session> session_config_type
```

Public Functions

```
session_ (cookies_type &cookies, shadow_type &shadow, const session_config_type &config)
template<typename ...T>
session_ (session_<request_type, udho::cache::shadow<key_type, T...>> &other)
void identify ()
const session_config_type &config () const
const key_type &id () const
bool returning () const
template<typename V>
bool exists () const
template<typename V>
V get () const
template<typename V>
V at ()
template<typename V>
void set (const V &value)
std::size_t size () const
bool remove ()
template<typename V>
bool remove ()
boost::posix_time::ptime created () const
boost::posix_time::ptime updated () const
boost::posix_time::time_duration age () const
boost::posix_time::time_duration idle () const
template<typename V>
```

```
boost::posix_time::ptime created() const  
template<typename V>  
boost::posix_time::ptime updated() const  
template<typename V>  
boost::posix_time::time_duration age() const  
template<typename V>  
boost::posix_time::time_duration idle() const  
std::string sessid() const
```

Public Members

```
const session_config_type &_config  
cookies_type &_cookies  
shadow_type _shadow  
bool _returning  
bool _identified  
key_type _id  
generator_type _generator
```

Template Struct `session_< RequestT, void >`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_session.h`

Struct Documentation

```
template<typename RequestT>  
struct udho::session_<RequestT, void>
```

Public Functions

```
template<typename ...U>  
session_(U...)
```

Template Struct `start`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Struct Documentation

```
template<typename ...T>
struct udho::start
```

Public Functions

```
start () = delete
start (const start<T...>&) = delete
```

Public Static Functions

```
template<typename ContextT>
start_<ContextT, T...> with (ContextT ctx, const std::string &name = "activity")
```

Template Struct **start_**

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Inheritance Relationships

Base Type

- `public udho::activities::subtask< activities::start< ContextT, T... > >`
(*Template Struct subtask*)

Struct Documentation

```
template<typename ContextT, typename ...T>
struct udho::start_ : public udho::activities::subtask<activities::start<ContextT, T...>>
```

Public Types

```
typedef activities::start<ContextT, T...> activity_type
typedef activities::subtask<activity_type> base
typedef activity_type::collector_type collector_type
typedef activity_type::accessor_type accessor_type
```

Public Functions

```
start_ (ContextT ctx, const std::string &name = "activity")
auto collector ()
auto data () const
auto data ()
template<typename ActivityT>
auto success () const
template<typename ActivityT>
auto failure () const
```

Struct url

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

Nested Types

- *Class `url::param`*

Inheritance Relationships

Base Types

- `public udho::configuration< detail::url_data >` (*Template Struct `configuration`*)
- `public udho::forms::drivers::urlencoded_< Iterator >` (*Template Struct `urlencoded_`*)

Struct Documentation

```
struct udho::url : public udho::configuration<detail::url_data>, public udho::forms::drivers::urlencoded_<Iterator>
```

Public Types

```
typedef udho::forms::drivers::urlencoded_raw urlencoded_type
```

Public Functions

```
url () = delete
url (const url &other)
std::string stringify () const
```

Public Static Functions

```
url build (const std::string &base, const std::initializer_list<param> &params)
url parse (const std::string &str)
class param
```

Public Functions

```
param (const std::string &k, const std::string &v)
std::string key () const
std::string value () const
std::string to_string () const
```

Template Struct urlencoded_form

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Struct Documentation

```
template<typename Iterator>
struct udho::urlencoded_form
    Form accessor for urlencoded forms
```

Public Types

```
typedef Iterator iterator_type
typedef std::iterator_traits<iterator_type>::value_type value_type
typedef std::basic_string<value_type> string_type
typedef bounded_str<iterator_type> bounded_string_type
typedef std::map<string_type, bounded_string_type> header_map_type
typedef bounded_string_type bounded_string
```

Public Functions

void **parse** (*iterator_type begin, iterator_type end*)
std::size_t **count** () **const**
 number of fields in the form
bool **has** (**const** std::string *name*) **const**
 checks whether there exists any field with the name provided
template<typename **T**>
const T field (**const** std::string &*name*) **const**
 returns the value of the field with the name provided lexically casted to type **T**

Public Members

header_map_type **_fields**

Template Struct is_invocable

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Inheritance Relationships

Base Type

- public std::is_constructible< std::function< void(Args ...)>,
 std::reference_wrapper< std::remove_reference< F >::type > >

Struct Documentation

```
template<typename F, typename ...Args>  
struct is_invocable : public std::is_constructible<std::function<void (Args...)>,  
                                                    std::reference_wrapper<std::remove_reference<F>::ty
```

Template Struct parser

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>  
struct udho::view::parser
```

Public Types

```

typedef ScopeT table_type
typedef parsing::expression<table_type> evaluator_type
typedef parsing::xml<ContextT> xml_parser_type
typedef parsing::tags::block<table_type, ContextT> block_directive_type
typedef parsing::tags::var<table_type, ContextT> var_directive_type
typedef parsing::tags::loop<table_type, ContextT> loop_directive_type
typedef parsing::tags::condition<table_type, ContextT> condition_directive_type
typedef parsing::tags::text<table_type, ContextT> text_directive_type
typedef parsing::tags::html<table_type, ContextT> html_directive_type
typedef parsing::tags::eval<table_type, ContextT> eval_directive_type
typedef parsing::tags::skip<table_type, ContextT> skip_directive_type
typedef parsing::attrs::target<table_type, ContextT> target_directive_type
typedef parsing::attrs::eval<table_type, ContextT> eval_attr_directive_type
typedef parsing::attrs::add_class<table_type, ContextT> add_class_directive_type
typedef parsing::attrs::condition<table_type, ContextT> condition_attr_directive_type

```

Public Functions

```

parser (table_type &scope, const ContextT &ctx)
void prepare ()
std::string process (const std::string &contents)

```

Public Members

```

table_type &_table
evaluator_type _evaluator
xml_parser_type _xml_parser
block_directive_type _block
var_directive_type _var
loop_directive_type _loop
condition_directive_type _condition
text_directive_type _text
html_directive_type _html
eval_directive_type _eval
skip_directive_type _skip
target_directive_type _target

```

```
eval_attr_directive_type _eval_attr
add_class_directive_type _add_class
condition_attr_directive_type _condition_attr
```

Template Struct add_class

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::attrs::add_class
    udho::add-class:NAME = EXPR will add a class named NAME if the expression EXPR is true.
```

```
<div udho:add-class:visible="count (books) > 1"></div>
```

is transformed to

```
<div class="visible"></div>
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
add_class (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node, pugi::xml_attribute attr, const ContextT
                  &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Template Struct condition

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::attrs::condition
    works like the udho:if tag but can be used in attribute
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
condition (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node, pugi::xml_attribute attr, const ContextT
    &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Template Struct eval

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::attrs::eval
    udho::eval:NAME = EXPR will add an attribute named NAME with value retrieved by evaluating the expression
    EXPR. It can have a conditional form too where the condition expr is first evaluated and if expr is true then the
    attribute is added, otherwise not
```

```
<input type="text" udho:eval:value="count (books) "></input>
<input type="text" udho:eval:title="expr ? count (books) "></input>
```

is transformed to

```
<input type="text" value="2"></input>
<input type="text" value="2"></input>
```

Public Types

```
typedef ScopeT table_type
```

```
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
eval (parsing::xml<ContextT> &ctrl, table_type &table)
```

```
bool operator () (pugi::xml_node node, pugi::xml_node, pugi::xml_attribute attr, const ContextT &ctx)
```

```
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
```

```
table_type &_table
```

```
evaluator_type _evaluator
```

Template Struct target

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
```

```
struct udho::view::parsing::attrs::target
```

udho::target:NAME = KEY will add an attribute named NAME with value retrieved by resolving KEY through scope table. It can have a conditional form too where the condition `expr` is first evaluated and if `expr` is true then the attribute is added, otherwise not

```
<div class="book" udho:target:title="book.title"></div>
<div class="book" udho:target:title="expr ? book.title"></div>
```

is transformed to

```
<div class="book" title="Some title"></div>
<div class="book" title="Some title"></div>
```

Public Types

```
typedef ScopeT table_type
```

```
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
target (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node, pugi::xml_attribute attr, const ContextT
                  &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Template Struct expression

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h

Struct Documentation

```
template<typename ScopeT>
struct udho::view::parsing::expression
```

Public Types

```
typedef ScopeT scope_type
```

Public Functions

```
expression (scope_type &table)
std::vector<token> rpn (const std::string &infix)
template<typename T>
T fetch (const token &t)
std::size_t count (const token &t) const
template<typename T>
T evaluate (const std::string &infix)
```

Public Members

```
scope_type &_table
```

Template Struct block

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::block
    udho::block
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
block (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Template Struct condition

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::condition
    udho::if
```

Public Types

```
typedef ScopeT table_type
```

```
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
condition (parsing::xml<ContextT> &ctrl, table_type &table)
```

```
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
```

```
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
```

```
table_type &_table
```

```
evaluator_type _evaluator
```

Template Struct eval

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::eval
    udho::eval
```

Public Types

```
typedef ScopeT table_type
```

```
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
eval (parsing::xml<ContextT> &ctrl, table_type &table)
```

```
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
```

```
std::string prefix () const
```

Public Members

`parsing::xml<ContextT> &_ctrl`
`table_type &_table`
`evaluator_type _evaluator`

Template Struct html

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::html
    udho::html
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
html (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
std::string prefix () const
```

Public Members

`parsing::xml<ContextT> &_ctrl`
`table_type &_table`
`evaluator_type _evaluator`

Template Struct loop

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::loop
    udho::for
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
loop (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Template Struct skip

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::skip
    udho::template
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
skip (parsing::xml<ContextT> &ctrl, table_type &table)  
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)  
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl  
table_type &_table  
evaluator_type _evaluator
```

Template Struct text

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>  
struct udho::view::parsing::tags::text  
    udho::text
```

Public Types

```
typedef ScopeT table_type  
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
text (parsing::xml<ContextT> &ctrl, table_type &table)  
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)  
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl  
table_type &_table  
evaluator_type _evaluator
```

Template Struct var

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ScopeT, typename ContextT>
struct udho::view::parsing::tags::var
    udho::var
```

Public Types

```
typedef ScopeT table_type
typedef parsing::expression<ScopeT> evaluator_type
```

Public Functions

```
var (parsing::xml<ContextT> &ctrl, table_type &table)
bool operator () (pugi::xml_node node, pugi::xml_node target, const ContextT &ctx)
std::string prefix () const
```

Public Members

```
parsing::xml<ContextT> &_ctrl
table_type &_table
evaluator_type _evaluator
```

Struct token

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
struct udho::view::parsing::token
```

Public Types

```
enum type
    Values:
        enumerator op
        enumerator function
        enumerator call
        enumerator comma
```

```
    enumerator parenthesis_open
    enumerator parenthesis_close
    enumerator real
    enumerator id
enum category
    Values:
    enumerator none
    enumerator arithmetic
    enumerator logical
    enumerator junction
typedef std::map<std::string, std::size_t> operator_map_type
typedef std::map<std::string, int> function_map_type
```

Public Functions

```
bool is (const type &t) const
bool is (const category &c) const
int precedence () const
std::string value () const
```

Public Static Functions

```
token create (const std::string &buff)
template<typename T>
T arithmetic (const token &op_token, const T &l, const T &r)
template<typename T>
bool logical (const token &op_token, const T &l, const T &r)
```

Template Struct xml

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Struct Documentation

```
template<typename ContextT>
struct udho::view::parsing::xml
```

Public Types

```
typedef ContextT context_type
typedef boost::function<bool (pugi::xml_node,    pugi::xml_node,    const context_type&)>
    tag_directive
typedef boost::function<bool (pugi::xml_node, pugi::xml_node, pugi::xml_attribute, const con-
    text_type&)> attr_directive
typedef std::map<std::string, tag_directive> tag_directive_map
typedef std::map<std::string, attr_directive> attr_directive_map
```

Public Functions

```
xml (const context_type &ctx)
void open (const std::string &contents)
void clear ()
void parse (pugi::xml_node node, pugi::xml_node target)
void step_in (pugi::xml_node node, pugi::xml_node target)
void traverse (pugi::xml_node node, pugi::xml_node target)
void parse ()
std::string output ()
template<typename DirectiveT>
xml &add_directive_tag (DirectiveT directive)
template<typename DirectiveT>
xml &add_directive_attr (DirectiveT directive)
```

Public Members

```
const context_type &_ctx
pugi::xml_document _source
pugi::xml_document _transformed
tag_directive_map _directives_tag
attr_directive_map _directives_attr
```

Template Struct visitor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Inheritance Relationships

Base Type

- `public VisitorT`

Struct Documentation

```
template<typename VisitorT>
struct udho::visitor: public VisitorT
```

Public Functions

```
template<typename ModuleT>
void operator() (ModuleT &overload)
void operator() ()
```

Template Struct `printing_visitor`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<std::uint8_t Visitables, typename StreamT>
struct udho::visitors::printing_visitor
```

Public Functions

```
printing_visitor (StreamT &stream)
template<typename V>
StreamT &column (const V &v, int width)
template<typename ModuleT>
void _module (const ModuleT &overload)
template<typename AppT>
void _application (const AppT &app)
void operator() ()
```

Public Members

StreamT &**_stream**

std::size_t **_indent**

Template Struct printing_visitor_html

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h

Struct Documentation

```
template<std::uint8_t Visitables, typename StreamT>
struct udho::visitors::printing_visitor_html
```

Public Functions

printing_visitor_html (*StreamT* &*stream*)

template<typename **ModuleT**>
void **_module** (const *ModuleT* &*overload*)

template<typename **AppT**>
void **_application** (const *AppT* &*app*)

void **operator** () ()

Public Members

StreamT &**_stream**

std::size_t **_indent**

Template Struct target_application

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h

Struct Documentation

```
template<typename F, std::uint8_t Visitables, std::uint8_t HasApplication = (Visitables & visitable::application)>
struct udho::visitors::target_application
```

Public Functions

```
target_application (F callback)  
template<typename AppT>  
void _application (AppT &overload)
```

Public Members

```
F _callback
```

Template Struct target_application< F, Visitables, 0 >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename F, std::uint8_t Visitables>  
struct udho::visitors::target_application<F, Visitables, 0>
```

Public Functions

```
target_application (F callback)  
template<typename AppT>  
void _application (AppT&)
```

Public Members

```
F _callback
```

Template Struct target_module

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename F, std::uint8_t Visitables, std::uint8_t HasModule = (Visitables & visitable::callable)>  
struct udho::visitors::target_module
```

Public Functions

```
target_module (F callback)
template<typename ModuleT>
void _module (ModuleT &overload)
```

Public Members

F _callback

Template Struct target_module< F, Visitables, 0 >

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
template<typename F, std::uint8_t Visitables>
struct udho::visitors::target_module<F, Visitables, 0>
```

Public Functions

```
target_module (F callback)
template<typename ModuleT>
void _module (ModuleT&)
```

Public Members

F _callback

Template Struct targetted_visitor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Inheritance Relationships

Base Types

- public udho::visitors::target_module< F, Visitables > (*Template Struct target_module*)
- public udho::visitors::target_application< F, Visitables > (*Template Struct target_application*)

Struct Documentation

```
template<std::uint8_t Visitables, typename F>
struct udho::visitors::targetted_visitor : public udho::visitors::target_module<F, Visitables>, public udho::v
```

Public Functions

```
targetted_visitor(F callback)
void operator() ()
```

Struct visitable

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Struct Documentation

```
struct udho::visitors::visitable
```

Public Static Attributes

```
const std::uint8_t callable = (1 << 0)
const std::uint8_t application = (1 << 1)
const std::uint8_t both = callable | application
```

Template Struct watch

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_watcher.h`

Struct Documentation

```
template<typename DerivedT, typename KeyT>
struct udho::watch
```

Public Types

```
typedef KeyT key_type
typedef watch<DerivedT, KeyT> self_type
typedef boost::posix_time::ptime time_point_type
typedef boost::posix_time::time_duration time_duration_type
```

Public Functions

```

watch (const key_type &key)
void release (const boost::system::error_code &e = boost::asio::error::operation_aborted)
time_point_type expiry () const
bool released () const
key_type key () const
bool valid () const
void expire_at (boost::posix_time::ptime time)

```

Public Members

```

key_type _key
time_point_type _expiry
bool _released

```

Template Struct watcher

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_watcher.h`

Struct Documentation

```

template<typename WatchT>
struct udho::watcher

```

Public Types

```

typedef WatchT watch_type
typedef watch_type::key_type key_type
typedef watcher<WatchT> self_type
typedef std::list<watch_type> container_type
typedef container_type::iterator iterator_type
typedef std::multimap<key_type, iterator_type> index_type
typedef boost::posix_time::ptime time_point_type
typedef boost::posix_time::time_duration time_duration_type

```

Public Functions

```
watcher (boost::asio::io_service &io, const time_duration_type &duration)  
bool insert (watch_type watch)  
void expired (const boost::system::error_code &err)  
std::size_t notify (const key_type &key)  
std::size_t notify_all ()  
void async_notify (const key_type &key)  
void async_notify_all ()
```

Public Members

```
boost::asio::io_service &_io  
time_duration_type _duration  
boost::asio::deadline_timer _timer  
container_type _watchers  
index_type _index  
boost::mutex _mutex
```

Template Class analyzer

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Class Documentation

```
template<typename ActivityT>  
class udho::activities::analyzer
```

```
const udho::accessor<A1, A2i, A3i>& d;  
d.apply<A1>(udho::activities::analyzer<A1>())  
    .canceled([](const A1SData& s, const A1FData& f) {  
        std::cout << "CANCELED" << std::endl;  
    })  
    .incomplete([]() {  
        std::cout << "INCOMPLETE" << std::endl;  
    })  
    .failure([](const A1FData& f) {  
        std::cout << "FAILED" << std::endl;  
    })  
    .success([](const A1SData& s) {  
        std::cout << "SUCCESS" << std::endl;  
    })  
);
```

An analyzer can be applied on an activity to analyze its result data. The analyzer can be equipped with callbackss to handle different result states.

See `state`

Public Functions

void **operator**() ()

void **operator**() (const success_type &s, const failure_type &f)

void **operator**() (const failure_type &f)

void **operator**() (const success_type &s)

template<typename **AccessorT**>

analyzer (*AccessorT* accessor)

construct an analyzer

self_type &**incomplete** (incomplete_ftor_type *ftor*)

If the activity is incomplete and an incomplete callback is set then the incomplete callback is invoked if any of the following holds

- there is no error callback
- there is an error callback but the task has not been canceled
- there is an error callback and the task has been canceled (

See **operator**()(const success_type&, const failure_type&))

self_type &**canceled** (cancelation_ftor_type *ftor*)

If the activity is canceled and a canceled callback is set then the canceled callback is invoked if the following holds

- there is no error callback

self_type &**error** (success_ftor_type *ftor*)

The error callback is called if the subtask has not failed but has been canceled (through **cancel_if**) Once the error callback is set failure and incomplete callbacks are invoked differently The error callback is called if the following holds

- there is an error callback set and the activity is canceled and the activity has not failed

self_type &**failure** (failure_ftor_type *ftor*)

If the activity has failed and a failure callback is set then the failure callback is called if any of the following holds

- there is no error callback
- there is an error callback set and the activity is canceled
- there is an error callback set and the activity is not canceled

self_type &**success** (success_ftor_type *ftor*)

If the activity is complete and neither failed nor canceled then the success callback is called if a success callback is set

activity_state **apply**()

bool **successful**()

Template Class application

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h`

Class Documentation

template<typename **DerivedT**>

class udho::application

all applications derive from this class

applications are a class having a set of methods bound with urls

```
struct my_app: public udho::application<my_app>{
    my_app();
    int add(udho::request_type req, int a, int b);
    int mul(udho::request_type req, int a, int b);
    template <typename RouterT>
    auto route(RouterT& router){
        return router | (get(&my_app::add).plain() = "^/add/(\\d+)/ (\\d+)$")
                       | (get(&my_app::mul).plain() = "^/mul/(\\d+)/ (\\d+)$");
    }
};
```

Template Parameters

- **DerivedT**: The derived application class

Public Functions

application (**const** std::string &name)

std::string **name** () **const**

template<typename **F**>

auto **get** (*F* *f*)

add a get method

Template Parameters

- **F**: callback type

Parameters

- *f*: callback

template<typename **F**>

auto **post** (*F* *f*)

add a post method

Template Parameters

- **F**: callback type

Parameters

- *f*: callback

template<typename **F**>

auto **head** (*F* *f*)

add a head method

Template Parameters

- F: callback type

Parameters

- f: callback

```
template<typename F>
auto put (F f)
    add a put method
```

Template Parameters

- F: callback type

Parameters

- f: callback

```
template<typename F>
auto del (F f)
    add a delete method
```

Template Parameters

- F: callback type

Parameters

- f: callback

Template Class connection

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_connection.h`

Nested Relationships**Nested Types**

- *Struct `connection::send_lambda`*

Inheritance Relationships**Base Type**

- `public std::enable_shared_from_this< connection< RouterT, AttachmentT > >`

Class Documentation

```
template<typename RouterT, typename AttachmentT>
class udho::connection : public std::enable_shared_from_this<connection<RouterT, AttachmentT>>
    Stateful HTTP Session
```

Public Functions

connection (*RouterT* &router, attachment_type &attachment, socket_type socket)
session constructor

Parameters

- router: router
- socket: TCP socket

~connection ()

void **run** ()
start the read loop

void **do_read** ()

void **on_read** (boost::system::error_code ec, std::size_t bytes_transferred)

void **on_write** (boost::system::error_code, std::size_t bytes_transferred, bool close)

void **do_close** ()

void **respond** (udho::defs::response_type &msg)

Template Class is_streamable

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h

Class Documentation

```
template<typename S, typename T>
class udho::detail::is_streamable
```

Public Static Attributes

const bool **value** = decltype(test<S,T>(0))::value

Template Class listener

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_listener.h

Inheritance Relationships

Base Type

- `public std::enable_shared_from_this< listener< RouterT, AttachmentT > >`

Class Documentation

```
template<typename RouterT, typename AttachmentT>
class udho::listener: public std::enable_shared_from_this<listener<RouterT, AttachmentT>>
    listener runs accept loop for HTTP sockets
```

Public Functions

```
listener (RouterT &router, boost::asio::io_service &service, attachment_type &attachment, const
          boost::asio::ip::tcp::endpoint &endpoint)
```

Parameters

- router: HTTP url mapping router
- service: I/O service
- endpoint: HTTP server endpoint to listen on

```
void stop ()
    stops accepting incoming connections
```

```
~listener ()
```

```
void run ()
    starts the async accept loop
```

```
void accept ()
    accept an incoming connection
```

```
void on_accept (boost::system::error_code ec)
```

Class url::param

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Nested Relationships

This class is a nested type of *Struct url*.

Class Documentation

class udho::url::param

Public Functions

param (const std::string &k, const std::string &v)

std::string **key** () const

std::string **value** () const

std::string **to_string** () const

1.3.3 Enums

Enum state

- Defined in file __home_docs_checkout_readthedocs.org_user_builds_udho_checkout_latest_includes_udho_activities.h

Enum Documentation

enum udho::activities::state

activity result states

Values:

enumerator unknown

enumerator incomplete

The activity is incomplete

enumerator failure

The activity has failed

enumerator success

The activity completed successfully

enumerator canceled

The activity is canceled

enumerator error

The activity is canceled but has not failed. The error state is only used when an error callback is passed to the analyzer

Enum form_type

- Defined in file __home_docs_checkout_readthedocs.org_user_builds_udho_checkout_latest_includes_udho_form.hxx

Enum Documentation

enum udho::form_type

Values:

enumerator unparsed
enumerator urlencoded
enumerator multipart

Enum segment

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Enum Documentation

enum udho::logging::segment

Values:

enumerator unknown
enumerator router
enumerator server
enumerator parser
enumerator user

Enum status

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Enum Documentation

enum udho::logging::status

Values:

enumerator error
enumerator warning
enumerator info
enumerator debug

1.3.4 Functions

Template Function `udho::_search`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
template<class ForwardIt1, class ForwardIt2>
constexpr ForwardIt1 udho::_search(ForwardIt1 first, ForwardIt1 last, ForwardIt2 s_first, ForwardIt2
                                     s_last)
```

Template Function `udho::activities::collect`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

```
template<typename ...T, typename ContextT>
std::shared_ptr<collector<ContextT, dataset<T...>>> udho::activities::collect(ContextT &ctx,
                                                                              const std::string
                                                                              &name)
```

Template Function `udho::activities::operator<<(collector<ContextT, dataset<T...>>&, const U&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::activities::operator<<`” with arguments `(collector<ContextT, dataset<T...>>&, const U&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- std::ostream &operator<<(std::ostream &os, const state &s)
- template<typename U, typename ...T> accessor<T...> &operator<<(accessor<T...> &h,
↪ const U &data)
- template<typename U, typename ContextT, typename ...T> collector<ContextT, dataset
↪ <T...>> &operator<<(collector<ContextT, dataset<T...>> &h, const U &data)
```

Template Function `udho::activities::operator<<(accessor<T...>&, const U&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::activities::operator<<`” with arguments `(accessor<T...>&, const U&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- std::ostream &operator<<(std::ostream &os, const state &s)
- template<typename U, typename ...T> accessor<T...> &operator<<(accessor<T...> &h,
  ↳const U &data)
- template<typename U, typename ContextT, typename ...T> collector<ContextT, dataset
  ↳<T...>> &operator<<(collector<ContextT, dataset<T...>> &h, const U &data)
```

Function `udho::activities::operator<<(std::ostream&, const state&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::activities::operator<<`” with arguments `(std::ostream&, const state&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- std::ostream &operator<<(std::ostream &os, const state &s)
- template<typename U, typename ...T> accessor<T...> &operator<<(accessor<T...> &h,
  ↳const U &data)
- template<typename U, typename ContextT, typename ...T> collector<ContextT, dataset
  ↳<T...>> &operator<<(collector<ContextT, dataset<T...>> &h, const U &data)
```

Template Function `udho::activities::operator>>(const collector<ContextT, dataset<T...>>&, U&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::activities::operator>>`” with arguments `(const collector<ContextT, dataset<T...>>&, U&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename U, typename ...T> const accessor<T...> &operator>>(const
  ↳accessor<T...> &h, U &data)
- template<typename U, typename ContextT, typename ...T> const collector<ContextT,
  ↳dataset<T...>> &operator>>(const collector<ContextT, dataset<T...>> &h, U &data)
```

Template Function `udho::activities::operator>>(const accessor<T...>&, U&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::activities::operator>>`” with arguments `(const accessor<T...>&, U&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename U, typename ...T> const accessor<T...> &operator>>(const ↵  
↵accessor<T...> &h, U &data)  
- template<typename U, typename ContextT, typename ...T> const collector<ContextT, ↵  
↵dataset<T...>> &operator>>(const collector<ContextT, dataset<T...>> &h, U &data)
```

Template Function `udho::after(T&...)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::after`” with arguments `(T&...)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename ...T> udho::activities::after<T...> after(T&... dependencies)  
- udho::activities::after_none after()
```

Function `udho::after()`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

`udho::activities::after_none` `udho::after()`

Template Function `udho::analyze`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

template<typename **ActivityT**, typename **AccessorT**>
 udho::activities::*analyzer*<*ActivityT*> udho:::analyze (*AccessorT* &accessor)

See *udho::activities::analyzer*

Template Function udho::app(T...)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::app” with arguments (T...) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename AppT, typename ...T> app_<AppT, false> app(T... args)
- template<typename AppT> app_<AppT, true> app(AppT &a)
```

Template Function udho::app(AppT&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_application.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::app” with arguments (AppT&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename AppT, typename ...T> app_<AppT, false> app(T... args)
- template<typename AppT> app_<AppT, true> app(AppT &a)
```

Template Function udho::associate

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

template<typename **F**>
 detail::association<*F*> udho:::associate (const std::string &key, *F* callback)

Template Function `udho::collect`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Function Documentation

```
template<typename ...T, typename ContextT>
auto udho::collect (ContextT &ctx, const std::string &name)
    shorthand for udho::activities::collect
```

Template Function `udho::cookie`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cookie.h`

Function Documentation

```
template<typename ValueT>
udho::cookie_<ValueT> udho::cookie (const std::string &name, const ValueT &v)
```

Template Function `udho::data(const T&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::data`” with arguments `(const T&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename H, typename T, typename ...Rest> auto data(const H &head, const_
↪T &tail, const Rest&... rest)
- template<typename H, typename T> data_helper<H, T>::result_type data(const H &
↪head, const T &tail)
- template<typename T> prepared<T> data(const T &obj)
```

Template Function `udho::data(const H&, const T&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::data” with arguments (const H&, const T&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename H, typename T, typename ...Rest> auto data(const H &head, const_
↪T &tail, const Rest&... rest)
- template<typename H, typename T> data_helper<H, T>::result_type data(const H &
↪head, const T &tail)
- template<typename T> prepared<T> data(const T &obj)
```

Template Function udho::data(const H&, const T&, const Rest&...)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::data” with arguments (const H&, const T&, const Rest&...) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename H, typename T, typename ...Rest> auto data(const H &head, const_
↪T &tail, const Rest&... rest)
- template<typename H, typename T> data_helper<H, T>::result_type data(const H &
↪head, const T &tail)
- template<typename T> prepared<T> data(const T &obj)
```

Template Function udho::del(F)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::del” with arguments (F) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto del(F ftor, A1 &a1)
- template<typename F> auto del(F ftor)
```

Template Function `udho::del(F, A1&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::del`” with arguments (F, A1&) in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename F, typename A1> auto del(F ftor, A1 &a1)
- template<typename F> auto del(F ftor)
```

Template Function `udho::detail::operator|`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

template<typename **U**, typename **V**, typename **F**>

association_group<*association*<*F*>, *association_group*<*U*, *V*>> `udho::detail::operator|` (**const** *associa-*
tion_group<*U*,
V>
&*non-*
terminal,
const
associa-
tion<*F*>
&*termi-*
nal)

Template Function `udho::detail::visit`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

template<typename **U**, typename **V**>

association_group_visitor<*association_group*<*U*, *V*>> `udho::detail::visit` (**const** *associa-*
tion_group<*U*, *V*>
&*nonterminal*)

Function udho::exceptions::http_redirection

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_page.h`

Function Documentation

```
http_error udho::exceptions::http_redirection (const      std::string      &location,
                                                boost::beast::http::status      status      =
                                                boost::beast::http::status::temporary_redirect)
```

Template Function udho::form::operator<<(validated<udho::form_<RequestT>>&, field<T, Required>&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::form::operator<<” with arguments (validated<udho::form_<RequestT>>&, field<T, Required>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename T, bool Required, typename IteratorT> validated<udho::multipart_
  ↳form<IteratorT>> &operator<<(validated<udho::multipart_form<IteratorT>> &
  ↳validator, field<T, Required> &field_)
- template<typename T, bool Required, typename RequestT> validated<udho::form_
  ↳<RequestT>> &operator<<(validated<udho::form_<RequestT>> &validator, field<T,
  ↳Required> &field_)
- template<typename T, bool Required> validated &operator<<(validated<> &validator,
  ↳field<T, Required> &field_)
```

Template Function udho::form::operator<<(validated<udho::multipart_form<IteratorT>>&, field<T, Required>&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::form::operator<<” with arguments (validated<udho::multipart_form<IteratorT>>&, field<T, Required>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename T, bool Required, typename IteratorT> validated<udho::multipart_
  ↳form<IteratorT>> &operator<<(validated<udho::multipart_form<IteratorT>> &
  ↳validator, field<T, Required> &field_)
- template<typename T, bool Required, typename RequestT> validated<udho::form_
  ↳<RequestT>> &operator<<(validated<udho::form_<RequestT>> &validator, field<T,
  ↳Required> &field_)
- template<typename T, bool Required> validated &operator<<(validated<> &validator,
  ↳field<T, Required> &field_)
```

Template Function `udho::form::operator<<(validated<>&, field<T, Required>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::form::operator<<`” with arguments (`validated<>&`, `field<T, Required>&`) in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename T, bool Required, typename IteratorT> validated<udho::multipart_
↪form<IteratorT>> &operator<<(validated<udho::multipart_form<IteratorT>> &
↪validator, field<T, Required> &field_)
- template<typename T, bool Required, typename RequestT> validated<udho::form_
↪<RequestT>> &operator<<(validated<udho::form_<RequestT>> &validator, field<T,
↪Required> &field_)
- template<typename T, bool Required> validated &operator<<(validated<> &validator,
↪field<T, Required> &field_)
```

Function `udho::form::validate()`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

validated<void> `udho::form::validate()`

Template Function `udho::form::validate(udho::form_<RequestT>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::form::validate`” with arguments (`udho::form_<RequestT>&`) in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename IteratorT> validated<udho::multipart_form<IteratorT>>
↪validate(const udho::multipart_form<IteratorT> &form)
- template<typename RequestT> validated<udho::form_<RequestT>> validate(udho::form_
↪<RequestT> &form)
- validated<void> validate()
```

Template Function `udho::form::validate(const udho::multipart_form<IteratorT>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::form::validate`” with arguments `(const udho::multipart_form<IteratorT>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename IteratorT> validated<udho::multipart_form<IteratorT>>_
  ↳ validate(const udho::multipart_form<IteratorT> &form)
- template<typename RequestT> validated<udho::form<RequestT>> validate(udho::form_
  ↳ <RequestT> &form)
- validated<void> validate()
```

Template Function `udho::forms::form_multipart`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

```
template<typename RequestT>
form_multipart<RequestT> udho::forms::form_multipart(const RequestT &req)
```

Template Function `udho::forms::operator<<(accumulated&, detail::constrained_field<ValidatorT, FieldT>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::forms::operator<<`” with arguments `(accumulated&, detail::constrained_field<ValidatorT, FieldT>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename T, bool Required> accumulated &operator<<(accumulated &acc, _
  ↳ field<T, Required> &field)
- template<typename ValidatorT, typename FieldT> accumulated &operator<
  ↳ <(accumulated &acc, detail::constrained_field<ValidatorT, FieldT> &field)
```

Template Function `udho::forms::operator<<(accumulated&, field<T, Required>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::forms::operator<<`” with arguments `(accumulated&, field<T, Required>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename T, bool Required> accumulated &operator<<(accumulated &acc,
↪field<T, Required> &field)
- template<typename ValidatorT, typename FieldT> accumulated &operator<
↪<(accumulated &acc, detail::constrained_field<ValidatorT, FieldT> &field)
```

Template Function `udho::forms::operator>>(form<DriverT>&, detail::constrained_field<ValidatorT, FieldT>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::forms::operator>>`” with arguments `(form<DriverT>&, detail::constrained_field<ValidatorT, FieldT>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename DriverT, typename T, bool Required> form<DriverT> &operator>>
↪(form<DriverT> &form, field<T, Required> &field)
- template<typename DriverT, typename T, bool Required> validated<form<DriverT>> &
↪operator>>(validated<form<DriverT>> &vf, field<T, Required> &field)
- template<typename DriverT, typename ValidatorT, typename FieldT> form<DriverT> &
↪operator>>(form<DriverT> &form, detail::constrained_field<ValidatorT, FieldT> &
↪field)
- template<typename DriverT, typename ValidatorT, typename FieldT> validated<form
↪<DriverT>> &operator>>(validated<form<DriverT>> &vf, detail::constrained_field
↪<ValidatorT, FieldT> &field)
```

Template Function `udho::forms::operator>>(form<DriverT>&, field<T, Required>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::forms::operator>>” with arguments (form<DriverT>&, field<T, Required>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename DriverT, typename T, bool Required> form<DriverT> &operator>>
  ↳(form<DriverT> &form, field<T, Required> &field)
- template<typename DriverT, typename T, bool Required> validated<form<DriverT>> &
  ↳operator>>(validated<form<DriverT>> &vf, field<T, Required> &field)
- template<typename DriverT, typename ValidatorT, typename FieldT> form<DriverT> &
  ↳operator>>(form<DriverT> &form, detail::constrained_field<ValidatorT, FieldT> &
  ↳field)
- template<typename DriverT, typename ValidatorT, typename FieldT> validated<form
  ↳<DriverT>> &operator>>(validated<form<DriverT>> &vf, detail::constrained_field
  ↳<ValidatorT, FieldT> &field)
```

Template **Function** **udho::forms::operator>>(validated<form<DriverT>>&, detail::constrained_field<ValidatorT, FieldT>&)** **de-**

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::forms::operator>>” with arguments (validated<form<DriverT>>&, detail::constrained_field<ValidatorT, FieldT>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename DriverT, typename T, bool Required> form<DriverT> &operator>>
  ↳(form<DriverT> &form, field<T, Required> &field)
- template<typename DriverT, typename T, bool Required> validated<form<DriverT>> &
  ↳operator>>(validated<form<DriverT>> &vf, field<T, Required> &field)
- template<typename DriverT, typename ValidatorT, typename FieldT> form<DriverT> &
  ↳operator>>(form<DriverT> &form, detail::constrained_field<ValidatorT, FieldT> &
  ↳field)
- template<typename DriverT, typename ValidatorT, typename FieldT> validated<form
  ↳<DriverT>> &operator>>(validated<form<DriverT>> &vf, detail::constrained_field
  ↳<ValidatorT, FieldT> &field)
```

Template Function **udho::forms::operator>>(validated<form<DriverT>>&, field<T, Required>&)**

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::forms::operator>>” with arguments (validated<form<DriverT>>&, field<T, Required>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename DriverT, typename T, bool Required> form<DriverT> &operator>>
  ↪(form<DriverT> &form, field<T, Required> &field)
- template<typename DriverT, typename T, bool Required> validated<form<DriverT>> &
  ↪operator>>(validated<form<DriverT>> &vf, field<T, Required> &field)
- template<typename DriverT, typename ValidatorT, typename FieldT> form<DriverT> &
  ↪operator>>(form<DriverT> &form, detail::constrained_field<ValidatorT, FieldT> &
  ↪field)
- template<typename DriverT, typename ValidatorT, typename FieldT> validated<form
  ↪<DriverT>> &operator>>(validated<form<DriverT>> &vf, detail::constrained_field
  ↪<ValidatorT, FieldT> &field)
```

Template Function udho::forms::validate

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Function Documentation

template<typename **DriverT**>
validated<form<DriverT>> udho::forms::validate (*form<DriverT>* &f)

Template Function udho::get(F)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::get” with arguments (F) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto get(F ftor, A1 &a1)
- template<typename F> auto get(F ftor)
```

Template Function udho::get(F, A1&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::get” with arguments (F, A1&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto get(F ftor, A1 &a1)
- template<typename F> auto get(F ftor)
```

Template Function udho::head(F)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::head” with arguments (F) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto head(F ftor, A1 &a1)
- template<typename F> auto head(F ftor)
```

Template Function udho::head(F, A1&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::head” with arguments (F, A1&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto head(F ftor, A1 &a1)
- template<typename F> auto head(F ftor)
```

Template Function `udho::internal::arguments_to_tuple`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

```
template<typename TupleT>
void udho::internal::arguments_to_tuple(TupleT &tuple, const std::vector<std::string>
                                         &args)
```

Template Function `udho::internal::html_summary`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_page.h`

Function Documentation

```
template<typename RouterT>
std::string udho::internal::html_summary(RouterT &router)
```

Template Function `udho::internal::member`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
template<typename T>
member_<T>::const_reduced_function_type udho::internal::member(T function, const typename
                                                                member_<T>::that_type
                                                                *that)
```

Function `udho::internal::path_cat`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
boost::filesystem::path udho::internal::path_cat(const boost::filesystem::path &base, const
                                                  std::string &path)
```

Function `udho::internal::path_inside`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
bool udho::internal::path_inside(const boost::filesystem::path &base, const boost::filesystem::path &path)
```

Template Function `udho::internal::reduced`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
template<typename T>
reduced_<T>::reduced_function_type udho::internal::reduced(T function, typename reduced_<T>::object_type that)
```

Template Function `udho::internal::to_fusion(std::index_sequence<Is...>, T&&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::internal::to_fusion`” with arguments `(std::index_sequence<Is...>, T&&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<class ...Ts> auto to_fusion(boost::tuple<Ts...> in)
- template<class ...Ts> boost::fusion::tuple<Ts...> to_fusion(std::tuple<Ts...> in)
- template<std::size_t... Is, class T> auto to_fusion(std::index_sequence<Is...>, T_
  ↳ &&in)
```

Template Function `udho::internal::to_fusion(boost::tuple<Ts...>)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::internal::to_fusion” with arguments (boost::tuple<Ts...>) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<class ...Ts> auto to_fusion(boost::tuple<Ts...> in)
- template<class ...Ts> boost::fusion::tuple<Ts...> to_fusion(std::tuple<Ts...> in)
- template<std::size_t... Is, class T> auto to_fusion(std::index_sequence<Is...>, T_
  ↳&&in)
```

Template Function udho::internal::to_fusion(std::tuple<Ts...>)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::internal::to_fusion” with arguments (std::tuple<Ts...>) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<class ...Ts> auto to_fusion(boost::tuple<Ts...> in)
- template<class ...Ts> boost::fusion::tuple<Ts...> to_fusion(std::tuple<Ts...> in)
- template<std::size_t... Is, class T> auto to_fusion(std::index_sequence<Is...>, T_
  ↳&&in)
```

Template Function udho::logging::messages::formatted::operator<<

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Function Documentation

```
template<udho::logging::status Status, typename T>
message<Status> &udho::logging::messages::formatted::operator<< (message<Status>
                                                                &msg, const T
                                                                &value)
```

Template Function udho::operator/=

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

template<typename **U**, typename **V**, typename **F**>

udho::overload_group<U, V> &udho::operator/=(udho::overload_group<U, V> &group, F fixture)

Template Function udho::operator<<(attachment<AuxT, LoggerT, CacheT>&, const udho::logging::message<Status>&)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_attachment.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::operator<<” with arguments (attachment<AuxT, LoggerT, CacheT>&, const udho::logging::message<Status>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const_
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator<<(const overload_group<U, V> &group, const F &method)
```

Template Function udho::operator<<(std::ostream&, const proxy<K, C>&)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::operator<<” with arguments (std::ostream&, const proxy<K, C>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const_
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator<<(const overload_group<U, V> &group, const F &method)
```

Template Function udho::operator<<(const detail::context_common<AuxT, RequestT>&, const udho::logging::message<Status>&)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_context.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::operator<<” with arguments (const detail::context_common<AuxT, RequestT>&, const udho::logging::message<Status>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const_
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
↳overload, const char (&expr) [N])
```

```

- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator<<(const overload_group<U, V> &group, const F &method)

```

Template Function udho::operator<<(cookies_<RequestT>&, const udho::cookie_<V>&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_cookie.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::operator<<” with arguments (cookies_<RequestT>&, const udho::cookie_<V>&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```

- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator<<(const overload_group<U, V> &group, const F &method)

```

Template Function `udho::operator<<(udho::context<AuxT, RequestT, ShadowT>&, const exceptions::http_error&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_page.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator<<`” with arguments (`udho::context<AuxT, RequestT, ShadowT>&`, `const exceptions::http_error&`) in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
  ↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
  ↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
  ↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
  ↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const
  ↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
  ↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
  ↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
  ↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
  ↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
  ↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
  ↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
  ↳F> operator<<(const overload_group<U, V> &group, const F &method)
```

Template Function `udho::operator<<(module_overload<Function, CompositorT>, const char(&))`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator<<`” with arguments (`module_overload<Function, CompositorT>`, `const char(&)`) in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
  ↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
  ↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
  ↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
  ↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const
  ↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
  ↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
```

```

- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
  ↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
  ↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
  ↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
  ↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
  ↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
  ↳F> operator<<(const overload_group<U, V> &group, const F &method)

```

Template Function `udho::operator<<(const overload_group<U, V>&, const F&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator<<`” with arguments `(const overload_group<U, V>&, const F&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```

- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
  ↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
  ↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
  ↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
  ↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const
  ↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
  ↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
  ↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
  ↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
  ↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
  ↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
  ↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
  ↳F> operator<<(const overload_group<U, V> &group, const F &method)

```

Template Function `udho::operator<<(StreamT&, const overload_group<U, V>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator<<`” with arguments `(StreamT&, const overload_group<U, V>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const_
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
- template<typename Function, template<typename> class CompositorT, int N> module_
↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT>
↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream,
↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT,
↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator<<(const overload_group<U, V> &group, const F &method)
```

Template Function `udho::operator<<(session_<RequestT, ShadowT>&, const T&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_session.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator<<`” with arguments `(session_<RequestT, ShadowT>&, const T&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename AuxT, typename LoggerT, typename CacheT, udho::logging::status_
↳Status> attachment<AuxT, LoggerT, CacheT> &operator<<(attachment<AuxT, LoggerT,
↳CacheT> &attachment, const udho::logging::message<Status> &msg)
- template<typename AuxT, typename RequestT, typename ShadowT> udho::context<AuxT,
↳RequestT, ShadowT> &operator<<(udho::context<AuxT, RequestT, ShadowT> &context,
↳const exceptions::http_error &error)
- template<typename AuxT, typename RequestT, udho::logging::status Status> const_
↳detail::context_common<AuxT, RequestT> &operator<<(const detail::context_common
↳<AuxT, RequestT> &ctx, const udho::logging::message<Status> &msg)
```

```

- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator<<(module_overload<Function, CompositorT> &
  ↳overload, const char (&expr) [N])
- template<typename K, typename C> std::ostream &operator<<(std::ostream &stream, &
  ↳const proxy<K, C> &p)
- template<typename RequestT, typename ShadowT, typename T> session_<RequestT, &
  ↳ShadowT> &operator<<(session_<RequestT, ShadowT> &session, const T &data)
- template<typename RequestT, typename V> cookies_<RequestT> &operator<<(cookies_
  ↳<RequestT> &cookies, const udho::cookie_<V> &cookie)
- template<typename StreamT, typename U, typename V> StreamT &operator<<(StreamT &
  ↳stream, const overload_group<U, V> &router)
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>, &
  ↳F> operator<<(const overload_group<U, V> &group, const F &method)

```

Template Function `udho::operator>>(const char(&), module_overload<Function, CompositorT>)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator>>`” with arguments `(const char(&), module_overload<Function, CompositorT>)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```

- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator>>(const char (&expr) [N], module_overload
  ↳<Function, CompositorT> overload)
- template<typename RequestT, typename ShadowT, typename T> const session_<RequestT,
  ↳ShadowT> &operator>>(const session_<RequestT, ShadowT> &session, T &data)

```

Template Function `udho::operator>>(const session_<RequestT, ShadowT>&, T&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_session.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator>>`” with arguments `(const session_<RequestT, ShadowT>&, T&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```

- template<typename Function, template<typename> class CompositorT, int N> module_
  ↳overload<Function, CompositorT> operator>>(const char (&expr) [N], module_overload
  ↳<Function, CompositorT> overload)
- template<typename RequestT, typename ShadowT, typename T> const session_<RequestT,
  ↳ShadowT> &operator>>(const session_<RequestT, ShadowT> &session, T &data)

```

Template Function `udho::operator|(const prepared_group<U, V>&, const udho::prepared<P>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator|`” with arguments `(const prepared_group<U, V>&, const udho::prepared<P>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator|(const overload_group<U, V> &group, const F &method)
- template<typename U, typename V, typename P> prepared_group<udho::prepared<P>,
↳prepared_group<U, V>> operator|(const prepared_group<U, V> &nonterminal, const
↳udho::prepared<P> &terminal)
- template<typename U, typename V> prepared_group<udho::prepared<V>, prepared_group
↳<udho::prepared<U>, void>> operator|(const udho::prepared<U> &left, const
↳udho::prepared<V> &right)
```

Template Function `udho::operator|(const udho::prepared<U>&, const udho::prepared<V>&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_access.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “`udho::operator|`” with arguments `(const udho::prepared<U>&, const udho::prepared<V>&)` in doxygen xml output for project “udho” from directory: `dox/xml`. Potential matches:

```
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator|(const overload_group<U, V> &group, const F &method)
- template<typename U, typename V, typename P> prepared_group<udho::prepared<P>,
↳prepared_group<U, V>> operator|(const prepared_group<U, V> &nonterminal, const
↳udho::prepared<P> &terminal)
- template<typename U, typename V> prepared_group<udho::prepared<V>, prepared_group
↳<udho::prepared<U>, void>> operator|(const udho::prepared<U> &left, const
↳udho::prepared<V> &right)
```

Template Function `udho::operator|(const overload_group<U, V>&, const F&)`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::operator!” with arguments (const overload_group<U, V>&, const F&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename U, typename V, typename F> overload_group<overload_group<U, V>,
↳F> operator|(const overload_group<U, V> &group, const F &method)
- template<typename U, typename V, typename P> prepared_group<udho::prepared<P>,
↳prepared_group<U, V>> operator|(const prepared_group<U, V> &nonterminal, const
↳udho::prepared<P> &terminal)
- template<typename U, typename V> prepared_group<udho::prepared<V>, prepared_group
↳<udho::prepared<U>, void>> operator|(const udho::prepared<U> &left, const
↳udho::prepared<V> &right)
```

Template Function udho::overload(boost::beast::http::verb, F, CompositorT<typename internal::function_signature<F>::return_type>)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::overload” with arguments (boost::beast::http::verb, F, CompositorT<typename internal::function_signature<F>::return_type>) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, template<typename> class CompositorT =
↳compositors::transparent> module_overload<F, CompositorT>
↳overload(boost::beast::http::verb request_method, F ftor, CompositorT<typename
↳internal::function_signature<F>::return_type> compositor = CompositorT<typename
↳internal::function_signature<F>::return_type>())
- template<typename F, typename A1, template<typename> class CompositorT =
↳compositors::transparent> auto overload(boost::beast::http::verb request_method,
↳F ftor, A1 &a1, CompositorT<typename internal::function_signature<F>::return_type>
↳compositor = CompositorT<typename internal::function_signature<F>::return_type>
↳())
```

Template Function udho::overload(boost::beast::http::verb, F, A1&, CompositorT<typename internal::function_signature<F>::return_type>)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::overload” with arguments (boost::beast::http::verb, F, A1&, CompositorT<typename internal::function_signature<F>::return_type>) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, template<typename> class CompositorT =
↳compositors::transparent> module_overload<F, CompositorT>
↳overload(boost::beast::http::verb request_method, F ftor, CompositorT<typename
↳internal::function_signature<F>::return_type> compositor = CompositorT<typename
↳internal::function_signature<F>::return_type>())
- template<typename F, typename A1, template<typename> class CompositorT =
↳compositors::transparent> auto overload(boost::beast::http::verb request_method,
↳F ftor, A1 &a1, CompositorT<typename internal::function_signature<F>::return_type>
↳compositor = CompositorT<typename internal::function_signature<F>::return_type>
↳())
```

Template Function udho::post(F)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::post” with arguments (F) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto post(F ftor, A1 &a1)
- template<typename F> auto post(F ftor)
```

Template Function udho::post(F, A1&)

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::post” with arguments (F, A1&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto post(F ftor, A1 &a1)
- template<typename F> auto post(F ftor)
```

Template Function udho::put(F)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::put” with arguments (F) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto put(F ftor, A1 &a1)
- template<typename F> auto put(F ftor)
```

Template Function udho::put(F, A1&)

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Function Documentation

Warning: doxygenfunction: Unable to resolve multiple matches for function “udho::put” with arguments (F, A1&) in doxygen xml output for project “udho” from directory: dox/xml. Potential matches:

```
- template<typename F, typename A1> auto put(F ftor, A1 &a1)
- template<typename F> auto put(F ftor)
```

Template Function udho::scope

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_scope.h`

Function Documentation

`template<typename GroupT>`

`lookup_table<GroupT> udho::scope(const GroupT &p)`

returns a table of strings mapped with value returning objects GroupT must be an instance of udho::prepare<T> which can be obtained by using udho::data(const T&)

Function udho::util::charToHex

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

std::string udho::util::charToHex (char *c*)

Template Function udho::util::from_hex

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

template<typename **CharT**>
CharT udho::util::from_hex (*CharT ch*)

Function udho::util::hexToChar

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

char udho::util::hexToChar (char *first*, char *second*)

Template Function udho::util::to_hex

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

template<typename **CharT**>
CharT udho::util::to_hex (*CharT x*)

Template Function udho::util::urldecode

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

template<typename **CharT**>
std::basic_string<*CharT*> udho::util::urldecode (const std::basic_string<*CharT*> &*src*)

Template Function udho::util::urlencode

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_util.h`

Function Documentation

```
template<typename CharT>
std::basic_string<CharT> udho::util::urlencode (const std::basic_string<CharT> &src)
```

Template Function udho::view::expression

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Function Documentation

```
template<typename ScopeT>
parsing::expression<ScopeT> udho::view::expression (ScopeT &scope)
```

Template Function udho::view::parsing::tokenize

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Function Documentation

```
template<typename InputIt, typename OutputIt>
OutputIt udho::view::parsing::tokenize (InputIt begin, InputIt end, OutputIt output)
```

Template Function udho::view::processor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_parser.h`

Function Documentation

```
template<typename ScopeT, typename ContextT>
parser<ScopeT, ContextT> udho::view::processor (ScopeT &scope, const ContextT &ctx)
    returns a parser interface using which an xml based view can be parsed ScopeT must be a lookup table which
    can be obtained by using udho::scope
```

Template Function `udho::visit`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Function Documentation

```
template<typename ActorT, typename ModuleT>
void udho::visit (ActorT &actor, ModuleT &mod)
```

Template Function `udho::visitors::print`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Function Documentation

```
template<std::uint8_t Visitables, typename F>
visitor<printing_visitor<Visitables, F>> udho::visitors::print (F &callback)
prints the selected routes in the router
```

```
router /= budho::visitors::print<budho::visitors::visitable::callable |  
↳ budho::visitors::visitable::application, std::ostream>(std::cout);
```

Template Function `udho::visitors::print_html`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Function Documentation

```
template<std::uint8_t Visitables, typename F>
visitor<printing_visitor_html<Visitables, F>> udho::visitors::print_html (F &callback)
```

Template Function `udho::visitors::target`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h`

Function Documentation

```
template<std::uint8_t Visitables, typename F>
visitor<targetted_visitor<Visitables, F>> udho::visitors::target (F callback)
```

1.3.5 Variables

Variable `udho::forms::default_datetime_format`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Variable Documentation

```
constexpr const char *udho::forms::default_datetime_format = "%Y-%m-%d %H:%M:%S"
```

1.3.6 Defines

Define `ROUTING_DEFERRED`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Define Documentation

`ROUTING_DEFERRED`

Define `UDHO_SESSION_COOKIE_NAME`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Define Documentation

`UDHO_SESSION_COOKIE_NAME`

Define `UDHO_SESSION_FILE_EXTENSION`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Define Documentation

`UDHO_SESSION_FILE_EXTENSION`

Define `UDHO_VERSION`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_defs.h`

Define Documentation

UDHO_VERSION

Define UDHO_VERSION_STRING

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_defs.h`

Define Documentation

UDHO_VERSION_STRING

1.3.7 Typedefs

Typedef udho::accessor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Typedef Documentation

using `udho::accessor` = `activities::accessor<T...>`
shorthand for *udho::activities::accessor*

See *udho::activities::accessor*

Typedef udho::activities::detail::apply_helper

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Typedef Documentation

using `udho::activities::detail::apply_helper` = *apply_helper_<udho::util::is_invocable<FunctorT, TargetsT...>::va*

Typedef udho::activity

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Typedef Documentation

using `udho::activity` = `activities::activity<DerivedT, SuccessDataT, FailureDataT>`
shorthand for *udho::activities::activity*

See *udho::activities::activity*

Typedef udho::client_options

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_client.h`

Typedef Documentation

typedef *client_options_* udho::client_options

Typedef udho::configs::form

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef *form_* udho::configs::form

Typedef udho::configs::logger

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef *logger_* udho::configs::logger

Typedef udho::configs::router

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef *router_* udho::configs::router

Typedef udho::configs::server

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef *server_* udho::configs::server

Typedef udho::configs::session

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef *session* udho::configs::session

Typedef udho::configuration_type

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_configuration.h`

Typedef Documentation

typedef udho::configuration<udho::configs::server, udho::configs::session, udho::configs::router, udho::configs::logger, udho::confi

Typedef udho::contexts::stateful

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_contexts.h`

Typedef Documentation

using udho::contexts::stateful = *context*<udho::bridge<udho::configuration_type>, udho::defs::request_type, udho::cach
A stateful context for stateful callable using the default bridge and the default HTTP request type.

See *udho::context*< AuxT, RequestT, void >

Template Parameters

- T . . . : Session states

Typedef udho::contexts::stateless

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_contexts.h`

Typedef Documentation

using udho::contexts::stateless = *context*<udho::bridge<udho::configuration_type>, udho::defs::request_type, void>
A stateless context for stateless callable using the default bridge and the default HTTP request type.

See *udho::context*

Typedef udho::defs::request_type

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_defs.h`

Typedef Documentation

typedef boost::beast::http::request<boost::beast::http::string_body> udho::defs::request_type

Typedef udho::defs::response_type

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_defs.h`

Typedef Documentation

typedef boost::beast::http::response<boost::beast::http::string_body> udho::defs::response_type

Typedef udho::defs::session_key_type

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_defs.h`

Typedef Documentation

typedef boost::uuids::uuid udho::defs::session_key_type

Typedef udho::detail::url_data

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_url.h`

Typedef Documentation

typedef url_data udho::detail::url_data

Typedef udho::form::optional

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Typedef Documentation

using udho::form::optional = *field*<T, false>

Typedef udho::form::required

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_form.hxx`

Typedef Documentation

```
using udho::form::required = field<T, true>
```

Typedef udho::forms::constraints::length_gt

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef gt<std::string> udho::forms::constraints::length_gt
```

Typedef udho::forms::constraints::length_gte

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef gte<std::string> udho::forms::constraints::length_gte
```

Typedef udho::forms::constraints::length_lt

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef lt<std::string> udho::forms::constraints::length_lt
```

Typedef udho::forms::constraints::length_lte

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef lte<std::string> udho::forms::constraints::length_lte
```

Typedef udho::forms::constraints::length_max

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef length_lte udho::forms::constraints::length_max
```

Typedef udho::forms::constraints::length_min

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef length_gte udho::forms::constraints::length_min
```

Typedef udho::forms::constraints::max

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::constraints::max = lte<T>
```

Typedef udho::forms::constraints::min

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::constraints::min = gte<T>
```

Typedef udho::forms::detail::constraint_visitor

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::detail::constraint_visitor = constraint_visitor<ConstrainedFieldT, ConstrainedFieldT::depth>
```

Typedef udho::forms::drivers::multipart_raw

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef multipart<std::string::const_iterator> udho::forms::drivers::multipart_raw
```

Typedef udho::forms::drivers::urlencoded_raw

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
typedef urlencoded<std::string::const_iterator> udho::forms::drivers::urlencoded_raw
```

Typedef udho::forms::form_

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::form_ = form<drivers::combo<RequestT>>
```

Typedef udho::forms::form_multipart_

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::form_multipart_ = form<drivers::multipart<RequestT>>
```

Typedef udho::forms::optional

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::optional = field<T, false>
```

Typedef udho::forms::query_

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::query_ = form<drivers::urlencoded_raw>
```

Typedef udho::forms::required

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_forms.h`

Typedef Documentation

```
using udho::forms::required = field<T, true>
```

Typedef udho::loggers::ostream

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Typedef Documentation

```
typedef plain<std::ostream> udho::loggers::ostream
```

Typedef udho::logging::messages::debug

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Typedef Documentation

```
typedef message<udho::logging::status::debug> udho::logging::messages::debug
```

Typedef udho::logging::messages::error

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Typedef Documentation

```
typedef message<udho::logging::status::error> udho::logging::messages::error
```

Typedef udho::logging::messages::formatted::debug

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Typedef Documentation

typedef *message*<udho::logging::status::debug> udho::logging::messages::formatted::debug

Typedef udho::logging::messages::formatted::error

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Typedef Documentation

typedef *message*<udho::logging::status::error> udho::logging::messages::formatted::error

Typedef udho::logging::messages::formatted::info

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Typedef Documentation

typedef *message*<udho::logging::status::info> udho::logging::messages::formatted::info

Typedef udho::logging::messages::formatted::warning

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Typedef Documentation

typedef *message*<udho::logging::status::warning> udho::logging::messages::formatted::warning

Typedef udho::logging::messages::info

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h

Typedef Documentation

typedef *message*<udho::logging::status::info> udho::logging::messages::info

Typedef udho::logging::messages::warning

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_logging.h`

Typedef Documentation

typedef *message*<udho::logging::status::warning> udho::logging::messages::warning

Typedef udho::perform

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Typedef Documentation

using udho::perform = activities::perform<ActivityT>
shorthand for *udho::activities::perform*

See *udho::activities::perform*

Typedef udho::require

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_activities.h`

Typedef Documentation

using udho::require = activities::require<DependenciesT...>
shorthand for *udho::activities::require*

See *udho::activities::require*

Typedef udho::router

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_router.h`

Typedef Documentation

typedef *basic_router* udho::router

Typedef udho::servers::ostreamed::stateful

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using `udho::servers::ostreamed::stateful` = *ostreamed_helper*<`udho::cache::store`<StorageT, boost::uuids::uuid, U...>

Typedef `udho::servers::ostreamed::stateless`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using `udho::servers::ostreamed::stateless` = *ostreamed_helper*<void>

Typedef `udho::servers::quiet::stateful`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using `udho::servers::quiet::stateful` = *server*<`udho::bridge`<`udho::configuration_type`>, void, `udho::cache::store`<StorageT, boost::uuids::uuid, U...>

Typedef `udho::servers::quiet::stateless`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using `udho::servers::quiet::stateless` = *server*<`udho::bridge`<`udho::configuration_type`>, void, void>

Typedef `udho::servers::stateless::logged`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using `udho::servers::stateless::logged` = *server*<`udho::bridge`<`udho::configuration_type`>, LoggerT, void>

Typedef `udho::servers::stateless::quiet`

- Defined in file `__home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_server.h`

Typedef Documentation

using udho::servers::stateless::**quiet** = *server*<udho::*bridge*<udho::*configuration_type*>, void, void>

Typedef udho::tcp

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_connection.h

Typedef Documentation

using udho::**tcp** = boost::asio::ip::tcp

Typedef udho::visitors::targeted

- Defined in file __home_docs_checkouts_readthedocs.org_user_builds_udho_checkouts_latest_includes_udho_visitor.h

Typedef Documentation

using udho::visitors::**targeted** = *visitor*<*targetted_visitor*<Visitables, F>>

QUICKSTART

BUILDING UDHO

udho can be built like a regular cmake library. The requirements of udho library are available in most *nix repositories.

3.1 Requirements

- boost >= 1.66
- icu [optional]
- git
- cmake
- c++ compiler

3.1.1 Ubuntu

```
sudo apt-get install git cmake build-essential boost-all-dev libicu-dev pugixml-dev
```

3.1.2 Fedora

```
sudo dnf install @development-tools  
sudo dnf install git cmake boost-devel libicu-devel pugixml-devel
```

3.1.3 Arch Linux

```
sudo pacman -S base-devel git cmake boost icu pugixml
```

3.1.4 Mac

```
brew install git cmake boost icu pugixml
```

3.2 Build

After the dependencies are installed udho has to be cloned from the git repository and compiled as a regular C++ library. use `make tests` to execute the unit tests. The examples are inside `examples` directory.

```
git clone https://gitlab.com/neel.basu/udho.git
cd udho
mkdir build
cd build
cmake ..
make
```

URL ROUTING

REQUEST CONTEXT

A context wraps a raw HTTP request and is passed to all callables once its corresponding url pattern is matched. The context also associates the request with the underlying server. So using the server the server common functions that are not specific any particular callable like logging, template rendering etc. can be performed. The context can be `stateful` or `stateless`. With a `stateless` context session informations can not be accessed even if the server is `stateful`. A `stateful` context is instantiated with a set of `states` that can be accessed from the callable. The set of states with which the `stateful` context is created should be the subset of the states of a `stateful` server. As the `stateful` context contains a subset of server states only that subset of state information is accessible through the `stateful` context.

Note: The context templates require the bridge and HTTP request type to be passed as the template parameters while instantiation. Its advised to use the convenience types instead of directly instiating these templates.

```
str::string profile(udho::contexts::stateful<user> ctx){
    if(ctx.session().exists<user>()){
        user data;
        ctx.session() >> data;
        // user logged in
    }else{
        // user not logged in
    }
}
```

`udho::contexts::stateful` is actually a typedef of `udho::context` with default bridge and request type

```
std::string profile(udho::contexts::stateless ctx, std::size_t uid){
    // view public profile of user uid
}
```

`udho::contexts::stateless` is actually a typedef of `udho::context` with default bridge and request type

5.1 Request

The request parameters can be accessed either through the convenience methods or through the raw boost-beast request object. Use `ctx.request()` to get the underlying request object from the context. However for common tasks there are convenience methods that extracts information from the request object and returns.

Warning: doxygenfunction: Cannot find function “udho::context::request” in doxygen xml output for project “udho” from directory: dox/xml

Warning: doxygenfunction: Cannot find function “udho::context::target” in doxygen xml output for project “udho” from directory: dox/xml

Warning: doxygenfunction: Cannot find function “udho::context::path” in doxygen xml output for project “udho” from directory: dox/xml

Warning: doxygenfunction: Cannot find function “udho::context::query” in doxygen xml output for project “udho” from directory: dox/xml

5.2 Logging

The context passes a logging message to the logger attached with the server.

Warning: doxygenfunction: Cannot find function “udho::context::log” in doxygen xml output for project “udho” from directory: dox/xml

5.3 API

5.3.1 Stateful Context Template

```
template<typename AuxT, typename RequestT, typename ShadowT>
struct udho::context : public udho::detail::context_common<AuxT, RequestT>
```

A Stateful context passed to all callables along with the arguments. The context is always the first argument to the callable. Even if the callable takes no arguments, it must take the context as the first argument. A stateful context should be used in callables that need to use session states.

Note instead of instantiating this template directly use `udho::contexts::stateful`

Template Parameters

- `AuxT`: bridge between the server and the callable
- `RequestT`: HTTP request type
- `ShadowT`: the session data structure

Public Functions

session_type &**session** ()
access the HTTP Session

See *udho::session_*

5.3.2 Stateless Context Template

template<typename **AuxT**, typename **RequestT**>

struct context<*AuxT*, *RequestT*, void> : **public** udho::detail::context_common<*AuxT*, *RequestT*>

A Stateless context passed to all callables along with the arguments. The context is always the first argument to the callable. Even if the callable takes no arguments, it must take the context as the first argument. A stateless context should be used in callables that need not to use session states.

Note instead of instantiating this template directly use `udho::contexts::stateless`

Template Parameters

- *AuxT*: bridge between the server and the callable
- *RequestT*: HTTP request type

HTTP SERVER

udho provides varying types of servers depending on the logging and session facilities. A server is called stateful if it provides HTTP session, otherwise it is called stateless. If the server provides APIs that do not require any session then use stateless server, otherwise use a stateful server. If no logging is required then use a quiet server that uses a void logger, otherwise use a logging server. The servers `udho::servers::ostreamed::stateless` and `udho::servers::ostreamed::stateful<>` used in the examples are stateless and stateful servers respectively that uses ostreamed logger present inside udho. All servers inside `udho::servers` namespace are typedef'ed `udho::server`.

6.1 Logged Stateless Server

```
udho::servers::stateless::logged<LoggerT>
udho::servers::logged<LoggerT>::stateless
```

6.2 Quiet Stateless Server

```
udho::servers::stateless::quiet
udho::servers::quiet::stateless
```

6.3 Logged Stateful Server

```
udho::servers::stateful<States...>::logged<LoggerT>
udho::servers::logged<LoggerT>::stateful<States...>
```

6.4 Quiet Stateful Server

```
udho::servers::stateful<States...>::quiet
udho::servers::quiet::stateful<States...>
```

udho comes with a logger `udho::loggers::ostream` that prints logginf messages to `std::ostream`. This logger is used in all examples. So there is a typedef `servers::stateful<States...>::ostreamed` that actually is `servers::stateful<States...>::logged<loggers::ostream>`

A Logging server takes reference to a logger in constructor. So the Logger has to be instantiated by the caller, as shown in the following example.

```
udho::loggers::ostream logger(std::cout);
udho::servers::stateful<StateA, StateB>::ostreamed server(logger);
```

For ease of access there is an `ostreamed` server is provided which owns the `ostream` logger inside. To use this server, no logger has to be instantiated. Only the stream (`std::cout`) has to be provided. through the constructor.

```
udho::servers::ostreamed::stateless
udho::servers::ostreamed::stateful<StateA, StateB>
```

6.4.1 States

The stateful server uses session cookie for remembering a returning user. The session cookie is named `UDHOSESSID`. For each session some information can be stored at server side. This type of features are useful for login, captcha, CSRF token etc.. `udho` takes advantage of stateful nation of C++ and provides session functionality around a set of states defined at compile time. A state can be any copiable C++ type. Once a stateful server is instantiated with `N` number of states, callables that takes stateful contexts with `M <= N` states can be used.

```
struct student{
    std::string first_name;
    std::string last_name;
    unsigned int registration_id;
};
struct appearance{
    std::string theme;
};
```

Using these two states a server can be defined as following.

```
udho::servers::ostreamed::stateful<student, appearance> server(std::cout);
```

A function taking a stateful context around zero or more of these two states can be used in the router when the above server is used. So all four (`func1`, `func2`, `func3`, and `func4`) are valid functions that can be mapped with the router when the above mentioned server is used.

```
std::string func1(udho::contexts::stateful<student, appearance> ctx){
    // ...
}
std::string func2(udho::contexts::stateful<student> ctx){
    // ...
}
std::string func3(udho::contexts::stateful<appearance> ctx){
    // ...
}
std::string func4(udho::contexts::stateless ctx){
    // ...
}
```

In these functions the corresponding values of the current session of the requested type can be accessed like the following.

```
std::string func1(udho::contexts::stateful<student, appearance> ctx){
    bool student_data_exists = ctx.session().exists<student>();
    if(!student_data_exists){
        student data;
        data.first_name = "Jane";
    }
}
```

(continues on next page)

(continued from previous page)

```
data.last_name = "Doe";  
data.registration_id = 1;  
  
ctx.session() << data;  
} else {  
    student data;  
    ctx.session() >> data;  
    std::cout << data.first_name << std::endl;  
}  
}
```


Forms are accessed through the context using `ctx.form()` method. Forms submitted in POST request can be url encoded or multipart. Different form parsing techniques are used by `udho::urlencoded_form` and `udho::multipart_form`.

7.1 API

7.1.1 `udho::form`

template<typename **RequestT**>

struct `udho::form_`

unified form accessor that can extract field from both urlencoded as well as multipart forms.

Public Functions

void **parse_urlencoded** ()

parse the beast request body as urlencoded form data

void **parse_multipart** ()

parse the beast request body as multipart form data

bool **is_urlencoded** () **const**

check whether the submitted form is urlencoded

bool **is_multipart** () **const**

check whether the submitted form is multipart

const *urlencoded_form*<std::string::const_iterator> &**urlencoded** () **const**

return the urlencoded specific form accessor

const *multipart_form*<std::string::const_iterator> &**multipart** () **const**

return the multipart specific form accessor

bool **parsed** () **const**

returns true once the form is parsed

bool **has** (**const** std::string &*name*) **const**

checks whether the form contains any field with matching name

Parameters

- *name*: name of the form field

template<typename **V**>

V field(**const** std::string &*name*) **const**
returns a form field object for a given field

fields_map_type::size_type **count** () **const**
returns number of fields in the form

7.1.2 udho::urlencoded_form

template<typename **Iterator**>
struct udho::urlencoded_form
Form accessor for urlencoded forms

Public Functions

std::size_t **count** () **const**
number of fields in the form

bool **has** (**const** std::string *name*) **const**
checks whether there exists any field with the name provided

template<typename **T**>
const T field(**const** std::string &*name*) **const**
returns the value of the field with the name provided lexically casted to type **T**

7.1.3 udho::multipart_form

template<typename **Iterator**>
struct udho::multipart_form
Form accessor for multipart forms

Public Functions

std::size_t **count** () **const**
number of fields in the form

bool **has** (**const** std::string *name*) **const**
checks whether the form has any field with the given name

const form_part &part (**const** std::string &*name*) **const**
returns the part associated with the given name

template<typename **T**>
const T field(**const** std::string &*name*) **const**
returns the value of the field lexically casted with the desired type

`form.field<int>("age");`

struct form_part
A part in the multipart form data

Public Functions

const *bounded_string_type* &**header** (**const** std::string &*key*) **const**
 returns the value associated with the key in the header of the part

```
header("Content-Disposition").copied<std::string>();
```

bounded_string_type **header** (**const** std::string &*key*, **const** std::string &*sub*) **const**
 returns the value associated with the key and sub key in the header of the part

```
header("Content-Disposition", "name").copied<std::string>();
```

bounded_string_type **name** () **const**
 name of the part

```
name().copied<std::string>();
```

bounded_string_type **filename** () **const**
 filename of the part

```
name().copied<std::string>();
```

const *bounded_string_type* &**body** () **const**
 body of the part returned as a pair of string iterators

```
body().copied<std::string>();
```

std::string **str** () **const**
 returns the body of the part as string

template<typename T>

T **value** () **const**
 lexically convert the contents of the body to the requested type

HTTP SESSION

udho server takes a `LoggerT` and a `CacheT` as template parameters. The `LoggerT` provides logging facilities and the `CacheT` serves as the HTTP session. the first parameter `udho::bridge` bridges a server with the contexts. It provides utilities like document root, that are configured at the server to the callable's context. The reference to the bridge can be accessed through `aux()` method in the context.

```
typedef server<udho::bridge, LoggerT, CacheT> server_type;
```

The cache is implemented by the template `udho::cache::store<KeyT, States...>`. The `cache::store` serves as a key-value store where the set of possible values are limited by the variadic parameter `States...`.

```
struct X{};
struct Y{};
struct Z{};
udho::cache::store<int, X, X, Z> store;
```

In the example above an integer indexed `store` is created that can store only three types of values (`X`, `Y`, `Z`) for each key. Any read or write operation using any type other than these three will be restricted at compile time. The operations that can be performed on the store are

- checks for existence of any value of type `V` against the `key`

- gets the value of type `V` associated with the `key`. If no such value is inserted returns the `default_value`

- returns the value of type `V` associated with `key` and throws exception if that doesn't exist.

- associates a value of type `V` with `key`

udho uses `boost::uuids::uuid` as `KeyT` to provide session id. So each session id is 128 bit uuid generated by boost uuid library.

LOGGING

ASYNC TASK GRAPH

Execution of Asynchronous Tasks may be required for executing SQL or any other data queries including HTTP queries to fetch external data. `udho` does not including a database library with itself. However any asynchronous database library using `boost asio` may benefit from an async activity framework. `udho::activities` provides an way to describe a task graph that gather data into an heterogenous collector. A task may be associated with its dependencies of other tasks and task related data.

10.1 Collector

Different activities using different success and failure data will be involved in the subtask graph. A subtask or even an activity may need to use the data fetched by its dependencies in order to execute its action. Hence a heterogenous collector collects the data submitted by the `success` or `failure` methods, which can be accessed through an *accessor*. In the following example `data` is a collector that collects success or failure data of three activities `A1`, `A2`, and `A3` respectively. Usually all activities involved in the subtask graph should be passed as a template parameter of `collect`

```
auto data = udho::activities::collect<A1, A2, A3>(ctx, "Any string name for the_
↪collector");
```

However under many circumstances the entire collector is not required to be accessed. Instead a subset of what has been collected are accessed. So an `accessor<...>` is created from that collector.

```
udho::activities::accessor<A1, A2> access(data);
if(!access.failed<A1>()){
    A1SData pre = access.success<A1>();
}
```

The collector has to be passed to the constructor of all activities.

10.2 Activity

An *activity* is the definition of the job that has to be performed. A subtask is an instantiation of an activity. Definition of an activity consists of two types `SuccessDataT`, `FailureDataT` and a no argument function call `operator()()` overload. A subtask on the other hand instantiates an activity by specifying its data and execution dependencies thus establishing an order of invocation. The philosophy is that the activity is reusable while the subtask reuses that to model different use cases. The activity is invoked through its `operator()()` overload after all its prior activities have completed (dependencies are specified by the subtask). The `operator()()` must call either `success()` or `failure()` methods either synchronously or asynchronously with `SuccessDataT` or `FailureDataT` in order to signal its completion. An activity `A` subclasses

from `udho::activity<A, SuccessA, FailureA>` which defines the `success()` and `failure()` methods.

In the following example the success and failure data for activity *A1* are *A1SData* and *A1FData* respectively. *A1* starts a timer for 5 seconds. Once the timer finishes the value 42 is set as the success value and the activity finishes.

```
struct A1SData{
    int value;
};

struct A1FData{
    int reason;
};

struct A1: udho::activities::activity<A1, A1SData, A1FData>{
    typedef udho::activities::activity<A1, A1SData, A1FData> base;

    boost::asio::deadline_timer _timer;

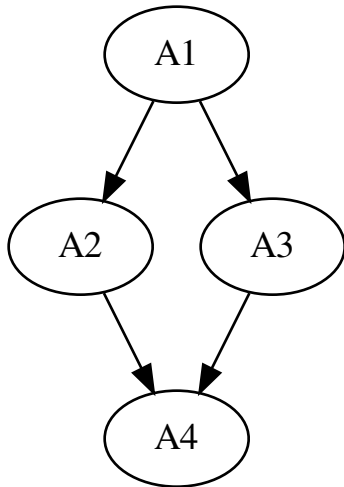
    template <typename CollectorT>
    A1(CollectorT c, boost::asio::io_context& io): base(c), _timer(io){}

    void operator() () {
        _timer.expires_from_now(boost::posix_time::seconds(5));
        _timer.async_wait(boost::bind(&A1::finished, self(),
        ↪boost::asio::placeholders::error));
    }

    void finished(const boost::system::error_code& e){
        A1SData data;
        data.value = 42; // Fetched 42 from somewhere
        success(data); // successful
    }
};
```

10.3 Subtask

Activities serves the purpose of defining an action. however to action is supposed to fit in some use case depending on the application needs. The application needs may require a specific order of invocation of these activities. So that sufficient data is collected before some activity is invoked.



In the above mentioned task graph, Both A2 and A3 depends on A1. Hence A2 and A3 may start as soon as A1 completes. A4 depends on both A2 and A3 and cannot start until both A2 and A3 completes. In this situation A2 and A3 can access the data fetched by A1 activity because that has already been completed (succeeded or failed) before A2 or A3 is invoked. Similarly A4 can access the data collected by A1, A2, A3 that has completed before A4 has been invoked.

`udho::activities::perform` describes the above subtask graph as shown in the following code block.

```

auto data = udho::activities::collect<A1, A2i, A3i>(ctx, "A");

auto t1 = udho::activities::perform<A1>::with(data, io);
auto t2 = udho::activities::perform<A2>::require<A1>::with(data, io).after(t1);
auto t3 = udho::activities::perform<A3>::require<A1>::with(data, io).after(t1);
auto t4 = udho::activities::perform<A4>::require<A2, A3>::with(data, io).after(t2).
    ↪after(t3);

```

The above statements using `perform` are equivalent to the following.

```

auto data = udho::activities::collect<A1, A2i, A3i>(ctx, "A");

auto t1 = udho::activities::subtask<A1>::with(data, io);
auto t2 = udho::activities::subtask<A2, A1>::with(data, io).after(t1);
auto t3 = udho::activities::subtask<A3, A1>::with(data, io).after(t1);
auto t4 = udho::activities::subtask<A4, A2, A3>::with(data, io).after(t2).after(t3);

```

With this arrangement Success/Failure data of A1 is available to A2 and A3 before it starts execution. To access that data the constructors on A2 and A3 may create an accessor as shown below.

```

struct A2: udho::activities::activity<A2, A2SData, A2FData>{
    typedef udho::activities::activity<A2, A2SData, A2FData> base;

    boost::asio::deadline_timer _timer;
    udho::activities::accessor<A1> _accessor;           // An accessor to access success/
    ↪failure data of A1
}

```

(continues on next page)

(continued from previous page)

```

template <typename CollectorT>
A2(CollectorT c, boost::asio::io_context& io): base(c), _timer(io), _accessor(c){}

void operator() () {
    _timer.expires_from_now(boost::posix_time::seconds(10));
    _timer.async_wait(boost::bind(&A2::finished, self(),
↳boost::asio::placeholders::error));
}

void finished(const boost::system::error_code& err){
    if(!err && !_accessor.failed<A1>()){           // check whether A1 has failed or
↳not
        A1SData pre = _accessor.success<A1>();    // access the success data
↳collected from A1
        A2SData data;
        data.value = pre.value + 2;               // creates its own success data
        success(data);                           // signal completion with success
    }
}
};

```

However in the above mentioned implementation A2 always depends on A1. In an use case where A2 depends on some other activity or even executed independently won't be feasible because A2 will still have an accessor to A1 and it will try to extract the success data of A1 irrespective of the graph. On the other hand, A2 may still need some data that has to be set in order to perform the activity. that data may be derived from A1 or from some other activity or through some constructor argument.

```

struct A2i: udho::activities::activity<A2i, A2SData, A2FData>{
    typedef udho::activities::activity<A2i, A2SData, A2FData> base;

    int prevalue;
    boost::asio::deadline_timer _timer;

    template <typename CollectorT>
    A2i(CollectorT c, boost::asio::io_context& io, int p): base(c), prevalue(p),
↳_timer(io){}

    template <typename CollectorT>
    A2i(CollectorT c, boost::asio::io_context& io): base(c), _timer(io){}

    void operator() () {
        _timer.expires_from_now(boost::posix_time::seconds(10));
        _timer.async_wait(boost::bind(&A2i::finished, self(),
↳boost::asio::placeholders::error));
    }

    void finished(const boost::system::error_code& err){
        if(!err){
            A2SData data;
            data.value = prevalue + 2;
            success(data);
        }
    }
};

```

In the above example prevalue is that piece of information that is required by A2i in order to perform its activity.

It doesn't care who provides that information, whether its A1, or some other activity. A2i has a constructor that takes that prevalue as an argument which can be used when prevalue is known at the time of construction, or when A2i is the starting activity in the graph.

```
auto t2 = udho::activities::perform<A2i>::require<A1>::with(data, io).after(t1).
↳prepare([data](A2i& a2i){
    udho::activities::accessor<A1> a1_access(data);
    A1SData pre = a1_access.success<A1>();
    a2i.prevalue = pre.value;
});
```

In the above code block that activity A2 is prepared through its reference passed as the argument right after all its dependencies (only A1 in this case) completes by the callback provided in the prepare function. The collected data is captured inside the lambda function from which the A1 success data is accessed.

10.4 Final

Finally we would like to do something once the entire subtask graph has completed in a similar fashion. In the following example we do not perform A4. Instead we execute the following piece of code once both A2i and A3i completes.

```
udho::activities::require<A2i, A3i>::with(data).exec([ctx, name](const_
↳udho::activities::accessor<A1, A2i, A3i>& d) mutable{
    std::cout << "Final begin" << std::endl;

    int sum = 0;

    if(!d.failed<A2i>()){
        A2SData pre = d.success<A2i>();
        sum += pre.value;
        std::cout << "A2i " << pre.value << std::endl;
    }

    if(!d.failed<A3i>()){
        A3SData pre = d.success<A3i>();
        sum += pre.value;
        std::cout << "A3i " << pre.value << std::endl;
    }

    ctx.respond(boost::lexical_cast<std::string>(sum), "text/plain");

    std::cout << "Final end" << std::endl;
}).after(t2).after(t3);
```

Once everything is set up we start the initial task t1()

```
t1();
```

Note: By default if one subtask fails then all subtasks that depend on it will be cancelled and the final callback will be called immediately. However the sibling subtasks of the failing subtask will continue executing. To change this behavior use `required(false)` on the subtask.

```
auto t1 = udho::activities::perform<A1>::with(data, io).required(false);
```

In the above example all other subtasks will execute even if t1 fails.

Note: The following shorthands may be used for convenience. `udho::collect` `udho::accessor`
`udho::activity` `udho::perform` `udho::require`

10.5 Example

10.6 API

10.6.1 Data

```
template<typename ContextT, typename DatasetT>
struct collector
```

```
template<typename ...T>
```

```
struct udho::activities::accessor : public udho::activities::fixed_key_accessor<udho::cache::shadow<std::string, T::
    Access a subset of data from the collector
```

Public Functions

```
template<typename V>
bool exists () const
    Whether there exists any data for activity V
```

Template Parameters

- V: Activity Type

```
template<typename V>
const V::result_type &get () const
    get data associated with activity V
```

Template Parameters

- V: activity type

```
template<typename V>
bool completed () const
    Check whether activity V has completed.
```

Template Parameters

- V: activity type

```
template<typename V>
bool canceled () const
    Check whether activity V has been canceled.
```

Template Parameters

- V: activity type

```
template<typename V>
bool failed () const
    Check whether activity V has failed (only the failure data of V is valid).
```

Template Parameters

- V: activity type

```
template<typename V>
bool okay () const
    Check whether activity V is okay.
```

Template Parameters

- V: activity type

```
template<typename V>
V::result_type::success_type success () const
    get success data for activity V
```

Template Parameters

- V: activity type

```
template<typename V>
V::result_type::failure_type failure () const
    get failure data for activity V
```

Template Parameters

- V: activity type

```
template<typename V, typename F>
void apply (F f) const
    Apply a callback on result of V
```

Template Parameters

- V: activity type

Parameters

- f: callback

10.6.2 Activity

```
template<typename DerivedT, typename SuccessDataT = void, typename FailureDataT = void>
```

```
struct udho::activities::activity: public std::enable_shared_from_this<DerivedT>, public udho::activities::result {
```

An activity A must subclass from `activity<A, SuccessA, FailureA>` assuming `SuccessA` and `FailureA` are the types that contains the relevant information regarding its success or failure. The activity A must overload a no argument operator `()` which initiates the activity. After the activity is initiated either `success()` or `failure()` methods must be called in order to signal its completion. The activity A must take the collector as the first argument to its constructor, which is passed to the base class `activity<A, SuccessA, FailureA>`. Hence its preferred to take the first parameter to the constructor as template parameter.

Template Parameters

- DerivedT: Activity Class
- SuccessDataT: data associated to the activity if the activity succeeds
- FailureDataT: data associated to the activity if the activity fails

Public Functions

derived_ptr_type **self** ()
shared_ptr to this

template<typename **SuccessT**, typename **FailureT**>
struct udho::activities::result : public udho::activities::result_data<*SuccessT*, *FailureT*>
Completion handler for an activity.

Template Parameters

- **SuccessT**: success data associated with the activity
- **FailureT**: failure data associated with the activity

Public Functions

template<typename **StoreT**>
result (*StoreT* &store)

Parameters

- store: collector

template<typename **CombinatorT**>
void **done** (*CombinatorT* cmb)
attach another subtask as done callback which will be executed once this subtask finishes

Parameters

- cmb: next subtask

void **required** (bool *flag*)
mark the activity as required or optional

Parameters

- flag:

void **cancel_if** (*cancel_if_for* f)
Force cancelation of the activity even after it is successful to stop propagating to the next activities

Parameters

- f: callback which should return true to signal cancelation

template<typename **SuccessT**, typename **FailureT**>
struct udho::activities::result_data
Contains **Copiable** Success or Failure data for an activity.

Template Parameters

- **SuccessT**: success data type
- **FailureT**: failure data type

Subclassed by *udho::activities::result*< *SuccessT*, *FailureT* >

Public Functions

bool **completed()** **const**
either success or failure data set

bool **failed()** **const**
whether the activity has failed

bool **canceled()** **const**
check whether the activity has been canceled

const *success_type* &**success_data()** **const**
Success data

const *failure_type* &**failure_data()** **const**
Failure data

template<typename **CallableT**>

void **apply**(*CallableT* *callback*)

Apply a callable to the result data which will be invoked exactly once with appropriate arguments. The invocation of the callback depends on the state as shown below.

- Incomplete: `callback()`
- Canceled: `callback(const SuccessT&, const FailureT&)`
- Failed: `callback(const FailureT&)`
- Successful: `callback(const SuccessT&)`

The callback may not have all the overloads.

Parameters

- `callback`:

10.6.3 Subtask

template<typename **ActivityT**, typename ...**DependenciesT**>

struct udho::activities::subtask

A subtask is an instantiation of an activity. The subtask reuses an activity to model different use cases by attaching dependencies. A subtask contains two shared pointers, one to the activity and another one to the combinator. The subtask cannot be instantiated directly by calling the subtask constructor. Instead call the static `with` method to instantiate.

Template Parameters

- `ActivityT`: The activity
- `DependenciesT`: The activities that has to be performed before performing `ActivityT`

Public Functions

`std::shared_ptr<activity_type> activity ()`
shared pointer to the activity

`template<typename V, typename ...DependenciesV>`
`self_type &done (subtask<V, DependenciesV...> &next)`
execute task next after the current one

Parameters

- `next`: the next subtask

`template<typename V, typename ...DependenciesV>`
`self_type &after (subtask<V, DependenciesV...> &previous)`
`t2.after(t1)` is equivalent to `t1.done(t2)`

Parameters

- `previous`: the previous subtask

`template<typename PreparatorT>`
`self_type &prepare (PreparatorT prep)`
attach a callback which will be called with a reference to the activity after it has been instantiated and all its dependencies have completed.

`self_type &required (bool flag)`
Set required flag on or off. If a required subtask fails then all intermediate subtask that depend on it fails and the final callback is called immediately. By default all subtasks are required

`self_type &cancel_if (typename activity_type::cancel_if_ftor cancelor)`
Force cancelation of the activity even after it is successful to stop propagating to the next activities

Parameters

- `f`: callback which should return true to signal cancelation

`std::shared_ptr<activity_type> operator-> ()`
returns the shared pointer to the activity

`template<typename FunctionT>`
`self_type &if_errored (FunctionT ftor)`
abort if canceled if ftor returns false. f will be called with the success if it has been canceled due to error

`template<typename FunctionT>`
`self_type &if_failed (FunctionT ftor)`
abort if canceled if ftor returns false. f will be called with the failure data if it has been canceled due to failure

Public Static Functions

`template<typename CollectorT, typename ...U>`
`self_type with (CollectorT collector, U&&... u)`
Arguments for the constructor of the Activity

`template<typename ActivityT>`
struct `udho::activities::perform`
create a subtask to perform activity ActivityT

Template Parameters

- `ActivityT`: the activity to perform

Public Static Functions

```
template<typename ...U>  
subtask<ActivityT> with (U&&... u)  
    arguments for the activity constructor
```

```
template<typename ...DependenciesT>  
struct require  
    mention the activities that has to be performed before executing this subtask.
```

Template Parameters

- DependenciesT: dependencies

Public Static Functions

```
template<typename ...U>  
subtask<ActivityT, DependenciesT...> with (U&&... u)  
    arguments for the activity constructor
```


UDHO ()

udho is a minimalistic featureful HTTP framework based on [Boost.Beast](#).

[Code](#) | [Issues](#)

```
std::string world(udho::contexts::stateless ctx){
    return "{\"planet': 'Earth'}";
}
std::string planet(udho::contexts::stateless ctx, std::string name){
    return "Hello "+name;
}
int main(){
    boost::asio::io_service io;
    udho::servers::ostreamed::stateless server(io, std::cout);

    auto urls = udho::router() | "/world"          >> udho::get(&world).json()
                               | "/planet/(\\w+)"    >> udho::get(&planet).plain();

    server.serve(urls, 9198);

    io.run();
    return 0;
}
```

11.1 Philosophy

`udho::router` comprises of mapping between url patterns (as [regex](#)) and the corresponding callables (function pointers, function objects). Multiple such mappings are combined at compile time using pipe (`|`) operator. The url mappings are passed to the server along with the listening port and document root. The [request methods](#) (`udho::get`, `udho::post`, `udho::head`, `udho::put`, etc..) and the response content type (`json()`, `plain()` etc..) are attached with the callable on compile time. Whenever an [HTTP request](#) appears to the server its path is matched against the url patterns and the matching callable is called. The values captured from the url patterns are lexically converted (using `boost::lexical_cast` <https://theboostcpplibraries.com/boost.lexical_cast>``) and passed as arguments to the callables. The server can be logging or quiet. The example above uses an ostreamed logger that logs on `std::cout`.

11.2 States

HTTP is stateless in general. But Session makes it stateful using a [session cookie](#) (e.g. `PHPSESSID` for PHP). A server can be `stateless` or `stateful` depending on the choice of `session`. If the server uses HTTP session then it has to be stateful, and the states comprising the session has to be declared at the compile time. The example above uses a stateless server (no HTTP session [cookie](#)). Hence all the callables take a statless context. The *example* shown below uses stateful session, where `user` is the only state. There can be multiple states passed as template parameters of `stateful<StateA, StateB, StateC>` while constructing the server. Callables should have a matching context, that takes all `contexts::stateful<StateA, StateB, StateC>` or lesser number of states `contexts::stateful<StateB, StateC>`. Callables in a stateful server can have a `contexts::stateless context` too.

11.3 Context

All the callables must take a compatible context as the first parameter. The context includes the [HTTP request object](#), logger reference, cookies, session information and additional accessories that might be required for serving the request.

DEPENDENCIES:

udho depend on boost. As boost-beast is only available on boost ≥ 1.66 , udho requires a boost version at least 1.66. udho may optionally use ICU library for unicode regex functionality. In that case ICU library may be required.

- boost > 1.66
- pugixml
- icu [optional]

FEATURES:

- [x] regular expression based url routing to callables (functions / function objects)
- [x] compile time binding of routing rule with callables
- [x] compile time traversable url router
- [x] response mime type specification in routing rule
- [x] any default constructible can be used as callable argument type that can be parsed from `std::string`
- [x] any ostreamable can be used as return type of callables
- [x] automatic type coercion for url based method calling
- [x] throwable http error messages
- [x] throwable HTTP redirection
- [x] serving static content from disk document root if no rule matched
- [x] urlencoded/multipart form parsing
- [x] form field validation
- [x] on memory / on disk session for stateful web applications (no globals) and no session for stateless applications
- [x] strictly typed on memory / on disk session storage
- [x] compile time pluggable & customizable logging
- [] server-sent events
- [x] deferred response and long polling
- [x] aync task graph
- [x] xml based template parsing (working but need refactoring)
- [x] arithmetic and logical expression evaluation in xml template (working but need refactoring)

STATEFUL EXAMPLE

```
struct user{
    std::string name;
    user() {}
    user(const std::string& nm) : name(nm) {}
};

std::string login(udho::contexts::stateful<user> ctx){ /// < strictly typed stateful
↳context
    const static username = "derp";
    const static password = "derp123";

    if(ctx.session().exists<user>()){
        user data;
        ctx.session() >> data; /// < extract session data
        return "already logged in";
    }else{
        if(ctx.form().has("user") && ctx.form().has("pass")){
            std::string usr = ctx.form().field<std::string>("user"); /// < form field
↳value from post request
            std::string psw = ctx.form().field<std::string>("pass"); /// < form field
↳value from post request
            if(usr == username && psw == password){
                ctx.session() << user(usr); /// < put data in session
                return "successful";
            }
        }
        return "failed";
    }
}

std::string echo(udho::contexts::stateful<user> ctx, int num){
    if(ctx.session().exists<user>()){
        user data;
        ctx.session() >> data;
        return boost::format("{ 'name': '%1%', 'num': %2%}") % data.name % num;
    }
    return "{}";
}

int main(){
    std::string doc_root("/path/to/static/document/root");

    boost::asio::io_service io;
    udho::servers::ostreamed::stateful<user> server(io, std::cout);
```

(continues on next page)

(continued from previous page)

```
auto router = udho::router()
| (udho::post(&login).plain() = "^/login$")
| (udho::get(&echo).json()    = "^/echo/(\\d+)$");

server.serve(router, 9198, doc_root);

io.run();

return 0;
}
```

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

R

ROUTING_DEFERRED (*C macro*), 273

U

udho::_search (*C++ function*), 242

udho::accessor (*C++ type*), 274

udho::activities::accessor (*C++ struct*), 27, 314

udho::activities::accessor::_shadow (*C++ member*), 29

udho::activities::accessor::accessor (*C++ function*), 27

udho::activities::accessor::apply (*C++ function*), 28, 315

udho::activities::accessor::base_type (*C++ type*), 27

udho::activities::accessor::canceled (*C++ function*), 28, 314

udho::activities::accessor::completed (*C++ function*), 28, 314

udho::activities::accessor::exists (*C++ function*), 27, 314

udho::activities::accessor::failed (*C++ function*), 28, 314

udho::activities::accessor::failure (*C++ function*), 28, 315

udho::activities::accessor::get (*C++ function*), 28, 314

udho::activities::accessor::name (*C++ function*), 27

udho::activities::accessor::okay (*C++ function*), 28, 315

udho::activities::accessor::set (*C++ function*), 28

udho::activities::accessor::shadow (*C++ function*), 27

udho::activities::accessor::shadow_type (*C++ type*), 27

udho::activities::accessor::success (*C++ function*), 28, 315

udho::activities::activity (*C++ struct*), 29, 315

udho::activities::activity::activity (*C++ function*), 30

udho::activities::activity::derived_ptr_type (*C++ type*), 30

udho::activities::activity::self (*C++ function*), 30, 316

udho::activities::after (*C++ struct*), 30

udho::activities::after::after (*C++ function*), 30

udho::activities::after::attach (*C++ function*), 30

udho::activities::after::finish (*C++ function*), 30

udho::activities::after::perform (*C++ function*), 30

udho::activities::after_none (*C++ struct*), 31

udho::activities::after_none::perform (*C++ function*), 32

udho::activities::after<HeadT> (*C++ struct*), 31

udho::activities::after<HeadT>::after (*C++ function*), 31

udho::activities::after<HeadT>::attach (*C++ function*), 31

udho::activities::after<HeadT>::finish (*C++ function*), 31

udho::activities::after<HeadT>::perform (*C++ function*), 31

udho::activities::analyzer (*C++ class*), 234

udho::activities::analyzer::analyzer (*C++ function*), 235

udho::activities::analyzer::apply (*C++ function*), 235

udho::activities::analyzer::canceled (*C++ function*), 235

udho::activities::analyzer::error (*C++ function*), 235

udho::activities::analyzer::failure (*C++ function*), 235

udho::activities::analyzer::incomplete (*C++ function*), 235

```

udho::activities::analyzer::operator()
    (C++ function), 235
udho::activities::analyzer::success
    (C++ function), 235
udho::activities::analyzer::successful
    (C++ function), 235
udho::activities::apply_helper (C++
    struct), 32
udho::activities::apply_helper::_ftor
    (C++ member), 32
udho::activities::apply_helper::apply_helper
    (C++ function), 32
udho::activities::apply_helper::operator()
    (C++ function), 32
udho::activities::collect (C++ function),
    242
udho::activities::collector (C++ struct),
    33, 314
udho::activities::collector<ContextT,
    dataset<T...>> (C++ struct), 33
udho::activities::collector<ContextT,
    dataset<T...>::_context (C++
    member), 34
udho::activities::collector<ContextT,
    dataset<T...>::base_type (C++
    type), 33
udho::activities::collector<ContextT,
    dataset<T...>::collector (C++
    function), 33
udho::activities::collector<ContextT,
    dataset<T...>::context (C++
    function), 33
udho::activities::collector<ContextT,
    dataset<T...>::context_type
    (C++ type), 33
udho::activities::combinator (C++ struct),
    34
udho::activities::combinator::_canceled
    (C++ member), 35
udho::activities::combinator::_counter
    (C++ member), 35
udho::activities::combinator::_mutex
    (C++ member), 35
udho::activities::combinator::_next
    (C++ member), 35
udho::activities::combinator::_preparator
    (C++ member), 35
udho::activities::combinator::cancel
    (C++ function), 34
udho::activities::combinator::combinator
    (C++ function), 34
udho::activities::combinator::next_type
    (C++ type), 34
udho::activities::combinator::operator()
    (C++ function), 34
udho::activities::combinator::prepare
    (C++ function), 34
udho::activities::combinator::propagate
    (C++ function), 34
udho::activities::combinator::signal_type
    (C++ type), 34
udho::activities::combinator<NextT,
    start<ContextT, T...>> (C++ struct),
    35
udho::activities::combinator<NextT,
    start<ContextT, T...>::_next
    (C++ member), 35
udho::activities::combinator<NextT,
    start<ContextT,
    T...>::combinator (C++ function), 35
udho::activities::combinator<NextT,
    start<ContextT,
    T...>::next_type (C++ type), 35
udho::activities::combinator<NextT,
    start<ContextT,
    T...>::operator() (C++ function), 35
udho::activities::combinator<NextT,
    start<ContextT,
    T...>::propagate (C++ function),
    35
udho::activities::dataset (C++ struct), 36
udho::activities::dataset::_name (C++
    member), 36
udho::activities::dataset::_shadow (C++
    member), 36
udho::activities::dataset::_store (C++
    member), 36
udho::activities::dataset::accessor_type
    (C++ type), 36
udho::activities::dataset::base_type
    (C++ type), 36
udho::activities::dataset::dataset (C++
    function), 36
udho::activities::dataset::name (C++
    function), 36
udho::activities::dataset::shadow (C++
    function), 36
udho::activities::dataset::shadow_type
    (C++ type), 36
udho::activities::dataset::store_type
    (C++ type), 36
udho::activities::detail::after (C++
    struct), 36
udho::activities::detail::after<subtask<ActivityT,
    DependenciesT...>> (C++ struct), 37
udho::activities::detail::after<subtask<ActivityT,
    DependenciesT...>::_before (C++
    member), 37

```

```

udho::activities::detail::after<subtask<AdhovingActivities::fixed_key_accessor::size
    DependenciesT...>>::after    (C++          (C++ function), 40
    function), 37                udho::activities::joined (C++ struct), 40
udho::activities::detail::after<subtask<AdhovingActivities::joined<CallbackT,
    DependenciesT...>>::attach  (C++          activities::collector<ContextT,
    function), 37                dataset<T...>>> (C++ struct), 40
udho::activities::detail::after<udho::start_<ContextT,
    T...>> (C++ struct), 37      activities::collector<ContextT,
udho::activities::detail::after<udho::start_<ContextT,
    T...>>::before (C++ member), 37      (C++ type), 40
udho::activities::detail::after<udho::start_<ContextT,
    T...>>::after (C++ function), 37      activities::collector<ContextT,
udho::activities::detail::after<udho::start_<ContextT,
    T...>>::attach (C++ function), 37      (C++ type), 40
udho::activities::detail::apply_helper  udho::activities::joined<CallbackT,
    (C++ type), 274              activities::collector<ContextT,
udho::activities::detail::apply_helper_  dataset<T...>>>::cancel    (C++
    (C++ struct), 38              function), 41
udho::activities::detail::apply_helper_::operator() activities::joined<CallbackT,
    (C++ function), 38              activities::collector<ContextT,
udho::activities::detail::apply_helper_<false, dataset<T...>>>::cancel_if (C++
    FunctorT, TargetsT...> (C++ struct),  function), 41
    38                          udho::activities::joined<CallbackT,
udho::activities::detail::apply_helper_<false, activities::collector<ContextT,
    FunctorT, TargetsT...>::operator()  dataset<T...>>>::cancel_if_ftor
    (C++ function), 39              (C++ type), 40
udho::activities::detail::final_intermediate udho::activities::joined<CallbackT,
    (C++ struct), 39              activities::collector<ContextT,
udho::activities::detail::final_intermediate:: dataset<T...>>>::collector_type
    (C++ member), 39              (C++ type), 40
udho::activities::detail::final_intermediate udho::activities::joined<CallbackT,
    (C++ function), 39              activities::collector<ContextT,
udho::activities::detail::final_intermediate:: dataset<T...>>>::final_intermediate
    (C++ function), 39              function), 41
udho::activities::fixed_key_accessor  udho::activities::joined<CallbackT,
    (C++ struct), 39              activities::collector<ContextT,
udho::activities::fixed_key_accessor::key  dataset<T...>>>::operator() (C++
    (C++ member), 40              function), 41
udho::activities::fixed_key_accessor::shadow udho::activities::joined<CallbackT,
    (C++ member), 40              activities::collector<ContextT,
udho::activities::fixed_key_accessor::at  dataset<T...>>>::self_type (C++
    (C++ function), 39              type), 40
udho::activities::fixed_key_accessor::execute udho::activities::junction (C++ struct), 41
    (C++ function), 39              udho::activities::junction::operator()
udho::activities::fixed_key_accessor::fixed_key (C++ function), 41
    (C++ function), 39              udho::activities::perform (C++ struct), 41,
udho::activities::fixed_key_accessor::get  318
    (C++ function), 39              udho::activities::perform::require (C++
udho::activities::fixed_key_accessor::key  struct), 42, 319
    (C++ function), 39              udho::activities::perform::require::with
udho::activities::fixed_key_accessor::key_type (C++ function), 42, 319
    (C++ type), 39              udho::activities::perform::with    (C++
udho::activities::fixed_key_accessor::set  function), 42, 319
    (C++ function), 40              udho::activities::require (C++ struct), 43

```

udho::activities::require::with (C++ function), 43

udho::activities::result (C++ struct), 43, 316

udho::activities::result::_abort_error (C++ member), 44

udho::activities::result::_abort_failure (C++ member), 44

udho::activities::result::_cancel_if (C++ member), 44

udho::activities::result::_cancellation_signal (C++ member), 44

udho::activities::result::_required (C++ member), 44

udho::activities::result::_shadow (C++ member), 44

udho::activities::result::_signal (C++ member), 44

udho::activities::result::_abort_error_ftor (C++ type), 43

udho::activities::result::_abort_failure_ftor (C++ type), 44

udho::activities::result::_accessor_type (C++ type), 43

udho::activities::result::_cancel_if (C++ function), 44, 316

udho::activities::result::_cancel_if_ftor (C++ type), 43

udho::activities::result::_cancellation_signal (C++ type), 43

udho::activities::result::_data_type (C++ type), 43

udho::activities::result::_done (C++ function), 44, 316

udho::activities::result::_failure (C++ function), 45

udho::activities::result::_failure_type (C++ type), 43

udho::activities::result::_if_errored (C++ function), 44

udho::activities::result::_if_failed (C++ function), 44

udho::activities::result::_required (C++ function), 44, 316

udho::activities::result::_result (C++ function), 44, 316

udho::activities::result::_signal_type (C++ type), 43

udho::activities::result::_success (C++ function), 45

udho::activities::result::_success_type (C++ type), 43

udho::activities::result_data (C++ struct), 45, 316

udho::activities::result_data::_canceled (C++ member), 46

udho::activities::result_data::_completed (C++ member), 46

udho::activities::result_data::_fdata (C++ member), 46

udho::activities::result_data::_sdata (C++ member), 46

udho::activities::result_data::_success (C++ member), 46

udho::activities::result_data::_apply (C++ function), 46, 317

udho::activities::result_data::_cancel (C++ function), 47

udho::activities::result_data::_canceled (C++ function), 46, 317

udho::activities::result_data::_completed (C++ function), 46, 317

udho::activities::result_data::_failed (C++ function), 46, 317

udho::activities::result_data::_failure (C++ function), 47

udho::activities::result_data::_failure_data (C++ function), 46, 317

udho::activities::result_data::_failure_type (C++ type), 45

udho::activities::result_data::_okay (C++ function), 46

udho::activities::result_data::_result_data (C++ function), 46

udho::activities::result_data::_result_type (C++ type), 45

udho::activities::result_data::_self_type (C++ type), 45

udho::activities::result_data::_success (C++ function), 47

udho::activities::result_data::_success_data (C++ function), 46, 317

udho::activities::result_data::_success_type (C++ type), 45

udho::activities::start (C++ struct), 47

udho::activities::start::_accessor (C++ member), 48

udho::activities::start::_collector (C++ member), 48

udho::activities::start::_signal (C++ member), 48

udho::activities::start::_accessor (C++ function), 47

udho::activities::start::_accessor_type (C++ type), 47

udho::activities::start::_collector (C++ function), 47

udho::activities::start::_collector_ptr

(C++ type), 47
 udho::activities::start::collector_type
 (C++ type), 47
 udho::activities::start::done (C++ func-
 tion), 47
 udho::activities::start::operator()
 (C++ function), 47
 udho::activities::start::signal_type
 (C++ type), 47
 udho::activities::start::start (C++ func-
 tion), 47
 udho::activities::start::success (C++
 function), 47
 udho::activities::state (C++ enum), 240
 udho::activities::state::canceled (C++
 enumerator), 240
 udho::activities::state::error (C++ enu-
 merator), 240
 udho::activities::state::failure (C++
 enumerator), 240
 udho::activities::state::incomplete
 (C++ enumerator), 240
 udho::activities::state::success (C++
 enumerator), 240
 udho::activities::state::unknown (C++
 enumerator), 240
 udho::activities::subtask (C++ struct), 48,
 317
 udho::activities::subtask::activity
 (C++ function), 48, 318
 udho::activities::subtask::activity_type
 (C++ type), 48
 udho::activities::subtask::after (C++
 function), 48, 318
 udho::activities::subtask::cancel_if
 (C++ function), 49, 318
 udho::activities::subtask::combinator_type
 (C++ type), 48
 udho::activities::subtask::done (C++
 function), 48, 318
 udho::activities::subtask::if_canceled
 (C++ function), 49
 udho::activities::subtask::if_errored
 (C++ function), 49, 318
 udho::activities::subtask::if_failed
 (C++ function), 49, 318
 udho::activities::subtask::operator->
 (C++ function), 49, 318
 udho::activities::subtask::prepare (C++
 function), 48, 318
 udho::activities::subtask::required
 (C++ function), 49, 318
 udho::activities::subtask::self_type
 (C++ type), 48
 udho::activities::subtask::subtask (C++
 function), 48
 udho::activities::subtask::with (C++
 function), 49, 318
 udho::activities::subtask<ActivityT>
 (C++ struct), 49
 udho::activities::subtask<ActivityT>::activity
 (C++ function), 50
 udho::activities::subtask<ActivityT>::activity_type
 (C++ type), 50
 udho::activities::subtask<ActivityT>::cancel_if
 (C++ function), 50
 udho::activities::subtask<ActivityT>::done
 (C++ function), 50
 udho::activities::subtask<ActivityT>::if_canceled
 (C++ function), 50
 udho::activities::subtask<ActivityT>::if_errored
 (C++ function), 50
 udho::activities::subtask<ActivityT>::if_failed
 (C++ function), 50
 udho::activities::subtask<ActivityT>::operator()
 (C++ function), 50
 udho::activities::subtask<ActivityT>::operator->
 (C++ function), 50
 udho::activities::subtask<ActivityT>::required
 (C++ function), 50
 udho::activities::subtask<ActivityT>::self_type
 (C++ type), 50
 udho::activities::subtask<ActivityT>::subtask
 (C++ function), 50
 udho::activities::subtask<ActivityT>::with
 (C++ function), 51
 udho::activities::subtask<start<ContextT,
 T...>> (C++ struct), 51
 udho::activities::subtask<start<ContextT,
 T...>>::_activity (C++ member), 52
 udho::activities::subtask<start<ContextT,
 T...>>::activity (C++ function), 51
 udho::activities::subtask<start<ContextT,
 T...>>::activity_type (C++ type), 51
 udho::activities::subtask<start<ContextT,
 T...>>::done (C++ function), 51
 udho::activities::subtask<start<ContextT,
 T...>>::operator() (C++ function), 51
 udho::activities::subtask<start<ContextT,
 T...>>::operator-> (C++ function), 51
 udho::activities::subtask<start<ContextT,
 T...>>::self_type (C++ type), 51
 udho::activities::subtask<start<ContextT,
 T...>>::subtask (C++ function), 51, 52
 udho::activities::subtask<start<ContextT,
 T...>>::with (C++ function), 52
 udho::activity (C++ type), 274
 udho::after (C++ function), 244

udho::analyze (C++ function), 245
 udho::app_ (C++ struct), 52
 udho::app_::_app (C++ member), 53
 udho::app_::_path (C++ member), 53
 udho::app_::app_ (C++ function), 52
 udho::app_::application_type (C++ type), 52
 udho::app_::eval (C++ function), 52
 udho::app_::name (C++ function), 52
 udho::app_::operator= (C++ function), 52
 udho::app_::self_type (C++ type), 52
 udho::app_::serve (C++ function), 52
 udho::app_::summary (C++ function), 52
 udho::app_<AppT, true> (C++ struct), 53
 udho::app_<AppT, true>::_app (C++ member), 53
 udho::app_<AppT, true>::_path (C++ member), 53
 udho::app_<AppT, true>::app_ (C++ function), 53
 udho::app_<AppT, true>::application_type (C++ type), 53
 udho::app_<AppT, true>::eval (C++ function), 53
 udho::app_<AppT, true>::name (C++ function), 53
 udho::app_<AppT, true>::operator= (C++ function), 53
 udho::app_<AppT, true>::self_type (C++ type), 53
 udho::app_<AppT, true>::serve (C++ function), 53
 udho::app_<AppT, true>::summary (C++ function), 53
 udho::application (C++ class), 236
 udho::application::application (C++ function), 236
 udho::application::del (C++ function), 237
 udho::application::get (C++ function), 236
 udho::application::head (C++ function), 236
 udho::application::name (C++ function), 236
 udho::application::post (C++ function), 236
 udho::application::put (C++ function), 237
 udho::assoc (C++ struct), 54
 udho::assoc::assoc (C++ function), 54
 udho::associate (C++ function), 245
 udho::attachment (C++ struct), 54
 udho::attachment::_io (C++ member), 55
 udho::attachment::_shadow (C++ member), 55
 udho::attachment::attachment (C++ function), 55
 udho::attachment::aux (C++ function), 55
 udho::attachment::auxiliary_type (C++ type), 54
 udho::attachment::cache_type (C++ type), 54
 udho::attachment::configuration_type (C++ type), 54
 udho::attachment::io (C++ function), 55
 udho::attachment::logger_type (C++ type), 54
 udho::attachment::self_type (C++ type), 54
 udho::attachment::shadow (C++ function), 55
 udho::attachment::shadow_type (C++ type), 54
 udho::attachment<AuxT, LoggerT, void> (C++ struct), 55
 udho::attachment<AuxT, LoggerT, void>::_io (C++ member), 56
 udho::attachment<AuxT, LoggerT, void>::attachment (C++ function), 56
 udho::attachment<AuxT, LoggerT, void>::aux (C++ function), 56
 udho::attachment<AuxT, LoggerT, void>::auxiliary_type (C++ type), 55
 udho::attachment<AuxT, LoggerT, void>::configuration_type (C++ type), 55
 udho::attachment<AuxT, LoggerT, void>::io (C++ function), 56
 udho::attachment<AuxT, LoggerT, void>::logger_type (C++ type), 55
 udho::attachment<AuxT, LoggerT, void>::self_type (C++ type), 55
 udho::attachment<AuxT, LoggerT, void>::shadow (C++ function), 56
 udho::attachment<AuxT, LoggerT, void>::shadow_type (C++ type), 55
 udho::attachment<AuxT, void, CacheT> (C++ struct), 56
 udho::attachment<AuxT, void, CacheT>::_io (C++ member), 57
 udho::attachment<AuxT, void, CacheT>::_shadow (C++ member), 57
 udho::attachment<AuxT, void, CacheT>::attachment (C++ function), 57
 udho::attachment<AuxT, void, CacheT>::aux (C++ function), 57
 udho::attachment<AuxT, void, CacheT>::auxiliary_type (C++ type), 56
 udho::attachment<AuxT, void, CacheT>::cache_type (C++ type), 56
 udho::attachment<AuxT, void,

```

    CacheT>::configuration_type
    (C++ type), 56
udho::attachment<AuxT, void,
    CacheT>::io (C++ function), 57
udho::attachment<AuxT, void,
    CacheT>::logger_type (C++ type),
    56
udho::attachment<AuxT, void,
    CacheT>::operator() (C++ function),
    57
udho::attachment<AuxT, void,
    CacheT>::self_type (C++ type),
    56
udho::attachment<AuxT, void,
    CacheT>::shadow (C++ function), 57
udho::attachment<AuxT, void,
    CacheT>::shadow_type (C++ type),
    56
udho::attachment<AuxT, void, void> (C++
    struct), 57
udho::attachment<AuxT, void,
    void>::_io (C++ member), 58
udho::attachment<AuxT, void,
    void>::attachment (C++ function),
    58
udho::attachment<AuxT, void,
    void>::aux (C++ function), 58
udho::attachment<AuxT, void,
    void>::auxiliary_type (C++ type), 57
udho::attachment<AuxT, void,
    void>::cache_type (C++ type), 57
udho::attachment<AuxT, void,
    void>::configuration_type (C++
    type), 57
udho::attachment<AuxT, void, void>::io
    (C++ function), 58
udho::attachment<AuxT, void,
    void>::logger_type (C++ type),
    57
udho::attachment<AuxT, void,
    void>::operator() (C++ function),
    58
udho::attachment<AuxT, void,
    void>::self_type (C++ type), 57
udho::attachment<AuxT, void,
    void>::shadow (C++ function), 58
udho::attachment<AuxT, void,
    void>::shadow_type (C++ type),
    57
udho::basic_router (C++ struct), 58
udho::basic_router::base_type (C++ type),
    58
udho::basic_router::basic_router (C++
    function), 59
udho::basic_router::operator| (C++ func-
    tion), 59
udho::bounded_str (C++ struct), 59
udho::bounded_str::_pair (C++ member), 60
udho::bounded_str::begin (C++ function), 59
udho::bounded_str::bounded_str (C++ func-
    tion), 59
udho::bounded_str::copied (C++ function), 60
udho::bounded_str::copy (C++ function), 59
udho::bounded_str::end (C++ function), 59
udho::bounded_str::invalid (C++ function),
    59
udho::bounded_str::operator= (C++ func-
    tion), 59
udho::bounded_str::pair_type (C++ type), 59
udho::bounded_str::self_type (C++ type), 59
udho::bounded_str::size (C++ function), 60
udho::bounded_str::valid (C++ function), 59
udho::bridge (C++ struct), 60
udho::bridge::_config (C++ member), 61
udho::bridge::_io (C++ member), 61
udho::bridge::bridge (C++ function), 60
udho::bridge::client (C++ function), 61
udho::bridge::config (C++ function), 60
udho::bridge::configuration_type (C++
    type), 60
udho::bridge::contents (C++ function), 60
udho::bridge::docroot (C++ function), 60
udho::bridge::file (C++ function), 60
udho::bridge::render (C++ function), 60, 61
udho::bridge::self_type (C++ type), 60
udho::bridge::tmplroot (C++ function), 60
udho::cache::abstract_engine (C++ struct),
    61
udho::cache::abstract_engine::age (C++
    function), 61
udho::cache::abstract_engine::content_type
    (C++ type), 61
udho::cache::abstract_engine::create
    (C++ function), 61
udho::cache::abstract_engine::created
    (C++ function), 61
udho::cache::abstract_engine::exists
    (C++ function), 61
udho::cache::abstract_engine::idle (C++
    function), 61
udho::cache::abstract_engine::insert
    (C++ function), 61, 62
udho::cache::abstract_engine::key_type
    (C++ type), 61
udho::cache::abstract_engine::remove
    (C++ function), 61
udho::cache::abstract_engine::retrieve
    (C++ function), 61

```

```

udho::cache::abstract_engine::self_type
    (C++ type), 61
udho::cache::abstract_engine::size (C++
    function), 61
udho::cache::abstract_engine::update
    (C++ function), 61, 62
udho::cache::abstract_engine::updated
    (C++ function), 61
udho::cache::abstract_engine::value_typeudho::cache::driver::abstract_engine_type
    (C++ type), 61
udho::cache::content (C++ struct), 62
udho::cache::content::_created (C++ mem-
    ber), 63
udho::cache::content::_updated (C++ mem-
    ber), 63
udho::cache::content::_value (C++ mem-
    ber), 63
udho::cache::content::age (C++ function), 62
udho::cache::content::content (C++ func-
    tion), 62
udho::cache::content::created (C++ func-
    tion), 62
udho::cache::content::idle (C++ function),
    62
udho::cache::content::self_type (C++
    type), 62
udho::cache::content::serialize (C++
    function), 62
udho::cache::content::update (C++ func-
    tion), 62
udho::cache::content::updated (C++ func-
    tion), 62
udho::cache::content::value (C++ function),
    62
udho::cache::content::value_type (C++
    type), 62
udho::cache::content<void> (C++ struct), 63
udho::cache::content<void>::_created
    (C++ member), 64
udho::cache::content<void>::_updated
    (C++ member), 64
udho::cache::content<void>::age (C++
    function), 63
udho::cache::content<void>::content
    (C++ function), 63
udho::cache::content<void>::created
    (C++ function), 63
udho::cache::content<void>::idle (C++
    function), 63
udho::cache::content<void>::self_type
    (C++ type), 63
udho::cache::content<void>::serialize
    (C++ function), 63
udho::cache::content<void>::update (C++
    function), 63
udho::cache::content<void>::updated
    (C++ function), 63
udho::cache::content<void>::value_type
    (C++ type), 63
udho::cache::driver (C++ struct), 64
udho::cache::driver::_engine (C++ mem-
    ber), 65
udho::cache::driver::abstract_engine_type
    (C++ type), 64
udho::cache::driver::age (C++ function), 65
udho::cache::driver::content_type (C++
    type), 64
udho::cache::driver::created (C++ func-
    tion), 64
udho::cache::driver::driver (C++ function),
    64
udho::cache::driver::exists (C++ function),
    64
udho::cache::driver::idle (C++ function), 65
udho::cache::driver::insert (C++ function),
    64
udho::cache::driver::key_type (C++ type),
    64
udho::cache::driver::remove (C++ function),
    64
udho::cache::driver::retrieve (C++ func-
    tion), 64
udho::cache::driver::size (C++ function), 64
udho::cache::driver::update (C++ function),
    64
udho::cache::driver::updated (C++ func-
    tion), 64
udho::cache::driver::value_type (C++
    type), 64
udho::cache::engine (C++ struct), 65
udho::cache::engine::abstract_engine_type
    (C++ type), 65
udho::cache::engine::config_type (C++
    type), 65
udho::cache::engine::content_type (C++
    type), 65
udho::cache::engine::create (C++ function),
    66
udho::cache::engine::engine (C++ function),
    66
udho::cache::engine::exists (C++ function),
    66
udho::cache::engine::key_type (C++ type),
    65
udho::cache::engine::remove (C++ function),
    66
udho::cache::engine::retrieve (C++ func-
    tion), 66

```

```

udho::cache::engine::self_type(C++ type), 65
udho::cache::engine::size(C++ function), 66
udho::cache::engine::storage_type(C++ type), 65
udho::cache::engine::update(C++ function), 66
udho::cache::engine::value_type(C++ type), 65
udho::cache::flake(C++ struct), 66
udho::cache::flake::age(C++ function), 67
udho::cache::flake::at(C++ function), 67
udho::cache::flake::created(C++ function), 67
udho::cache::flake::exists(C++ function), 67
udho::cache::flake::flake(C++ function), 67
udho::cache::flake::flake_type(C++ type), 66
udho::cache::flake::idle(C++ function), 67
udho::cache::flake::insert(C++ function), 67
udho::cache::flake::key_type(C++ type), 66
udho::cache::flake::registry_type(C++ type), 66
udho::cache::flake::remove(C++ function), 67
udho::cache::flake::self_type(C++ type), 66
udho::cache::flake::size(C++ function), 67
udho::cache::flake::update(C++ function), 67
udho::cache::flake::updated(C++ function), 67
udho::cache::flake::value_type(C++ type), 66
udho::cache::generator(C++ struct), 67
udho::cache::generator<boost::uuids::uuid(C++ struct), 67
udho::cache::generator<boost::uuids::uuid(C++ function), 67
udho::cache::generator<boost::uuids::uuid(C++ function), 67
udho::cache::master(C++ struct), 68
udho::cache::master::abstract_engine_type(C++ type), 68
udho::cache::master::issue(C++ function), 68
udho::cache::master::issued(C++ function), 68
udho::cache::master::key_type(C++ type), 68
udho::cache::master::master(C++ function), 68
udho::cache::master::self_type(C++ type), 68
udho::cache::master::storage_type(C++ type), 68
udho::cache::registry(C++ struct), 69
udho::cache::registry::abstract_engine_type(C++ type), 69
udho::cache::registry::at(C++ function), 69
udho::cache::registry::key_type(C++ type), 69
udho::cache::registry::registry(C++ function), 69
udho::cache::registry::self_type(C++ type), 69
udho::cache::registry::value_type(C++ type), 69
udho::cache::shadow(C++ struct), 70
udho::cache::shadow::_master(C++ member), 71
udho::cache::shadow::age(C++ function), 71
udho::cache::shadow::at(C++ function), 70
udho::cache::shadow::created(C++ function), 71
udho::cache::shadow::exists(C++ function), 70
udho::cache::shadow::get(C++ function), 70
udho::cache::shadow::idle(C++ function), 71
udho::cache::shadow::insert(C++ function), 71
udho::cache::shadow::issue(C++ function), 70
udho::cache::shadow::issued(C++ function), 70
udho::cache::shadow::key_type(C++ type), 70
udho::cache::shadow::master_type(C++ type), 70
udho::cache::shadow::remove(C++ function), 71
udho::cache::shadow::self_type(C++ type), 70
udho::cache::shadow::set(C++ function), 70
udho::cache::shadow::shadow(C++ function), 70
udho::cache::shadow::shadow_type(C++ type), 70
udho::cache::shadow::size(C++ function), 71
udho::cache::shadow::updated(C++ function), 71
udho::cache::storage::disk(C++ struct), 71
udho::cache::storage::disk::_config(C++ member), 73
udho::cache::storage::disk::_mutex(C++ member), 73

```

udho::cache::storage::disk::_name (C++ member), 73	udho::cache::storage::memory::content_type (C++ type), 73
udho::cache::storage::disk::config_type (C++ type), 72	udho::cache::storage::memory::create (C++ function), 73
udho::cache::storage::disk::content_type (C++ type), 72	udho::cache::storage::memory::exists (C++ function), 73
udho::cache::storage::disk::create (C++ function), 72	udho::cache::storage::memory::key_type (C++ type), 73
udho::cache::storage::disk::disk (C++ function), 72	udho::cache::storage::memory::map_type (C++ type), 73
udho::cache::storage::disk::exists (C++ function), 72	udho::cache::storage::memory::memory (C++ function), 73
udho::cache::storage::disk::extension (C++ function), 72	udho::cache::storage::memory::remove (C++ function), 73
udho::cache::storage::disk::filename (C++ function), 72	udho::cache::storage::memory::retrieve (C++ function), 73
udho::cache::storage::disk::format (C++ function), 72	udho::cache::storage::memory::self_type (C++ type), 73
udho::cache::storage::disk::key_type (C++ type), 72	udho::cache::storage::memory::session_config_type (C++ type), 73
udho::cache::storage::disk::load (C++ function), 72	udho::cache::storage::memory::size (C++ function), 73
udho::cache::storage::disk::map_type (C++ type), 72	udho::cache::storage::memory::update (C++ function), 73
udho::cache::storage::disk::path (C++ function), 72	udho::cache::storage::memory::value_type (C++ type), 73
udho::cache::storage::disk::remove (C++ function), 72	udho::cache::storage::redis (C++ struct), 74
udho::cache::storage::disk::retrieve (C++ function), 72	udho::cache::store (C++ struct), 74
udho::cache::storage::disk::save (C++ function), 72	udho::cache::store::age (C++ function), 75
udho::cache::storage::disk::self_type (C++ type), 72	udho::cache::store::at (C++ function), 75
udho::cache::storage::disk::session_config_type (C++ type), 72	udho::cache::store::config_type (C++ type), 75
udho::cache::storage::disk::size (C++ function), 72	udho::cache::store::created (C++ function), 75
udho::cache::storage::disk::storage (C++ function), 72	udho::cache::store::exists (C++ function), 75
udho::cache::storage::disk::update (C++ function), 72	udho::cache::store::get (C++ function), 75
udho::cache::storage::disk::value_type (C++ type), 72	udho::cache::store::idle (C++ function), 75
udho::cache::storage::memory (C++ struct), 73	udho::cache::store::insert (C++ function), 75
udho::cache::storage::memory::_config (C++ member), 74	udho::cache::store::key_type (C++ type), 75
udho::cache::storage::memory::_mutex (C++ member), 74	udho::cache::store::master_type (C++ type), 75
udho::cache::storage::memory::_storage (C++ member), 74	udho::cache::store::remove (C++ function), 75
udho::cache::storage::memory::config_type (C++ type), 73	udho::cache::store::self_type (C++ type), 75
	udho::cache::store::shadow_type (C++ type), 75
	udho::cache::store::size (C++ function), 75
	udho::cache::store::store (C++ function), 75
	udho::cache::store::updated (C++ function), 75

```

udho::client_options (C++ type), 275
udho::client_options_ (C++ struct), 76
udho::client_options::_follow_redirect (C++ member), 76
udho::client_options::_http_version (C++ member), 76
udho::client_options::_verify_certificate (C++ member), 76
udho::client_options::client_options_ (C++ function), 76
udho::client_options::follow_redirect (C++ member), 77
udho::client_options::follow_redirect_t (C++ struct), 77
udho::client_options::follow_redirect_t::component (C++ function), 81
udho::client_options::follow_redirect_t::type (C++ type), 77
udho::client_options::get (C++ function), 76
udho::client_options::http_version (C++ member), 77
udho::client_options::http_version_t (C++ struct), 77, 78
udho::client_options::http_version_t::component (C++ type), 77, 78
udho::client_options::set (C++ function), 76
udho::client_options::verify_certificate (C++ member), 77
udho::client_options::verify_certificate_t (C++ struct), 77, 78
udho::client_options::verify_certificate_t::component (C++ type), 77, 78
udho::collect (C++ function), 246
udho::compositors::deferred (C++ struct), 78
udho::compositors::deferred::name (C++ function), 79
udho::compositors::deferred::operator() (C++ function), 79
udho::compositors::deferred::response_type (C++ type), 79
udho::compositors::mimed (C++ struct), 79
udho::compositors::mimed::_mime (C++ member), 79
udho::compositors::mimed::mimed (C++ function), 79
udho::compositors::mimed::name (C++ function), 79
udho::compositors::mimed::operator() (C++ function), 79
udho::compositors::mimed::response_type (C++ type), 79
udho::compositors::transparent (C++ struct), 80
udho::compositors::transparent::name (C++ function), 80
udho::compositors::transparent::operator() (C++ function), 80
udho::compositors::transparent::response_type (C++ type), 80
udho::config (C++ struct), 80
udho::config::operator[] (C++ function), 81
udho::config::self_type (C++ type), 81
udho::config<configs::server_>> (C++ struct), 81
udho::config<configs::server_>::mime (C++ function), 81
udho::config<configs::server_>::operator[] (C++ function), 81
udho::config<configs::server_>::self_type (C++ type), 81
udho::configs::form (C++ type), 275
udho::configs::form_ (C++ struct), 82
udho::configs::logger (C++ type), 275
udho::configs::logger_ (C++ struct), 82
udho::configs::router (C++ type), 275
udho::configs::router_ (C++ struct), 82
udho::configs::server (C++ type), 275
udho::configs::server_ (C++ struct), 83
udho::configs::server::_document_root (C++ member), 83
udho::configs::server::_mime_default (C++ member), 83
udho::configs::server::_mimes (C++ member), 83
udho::configs::server::_template_root (C++ member), 83
udho::configs::server::_document_root (C++ member), 83
udho::configs::server::_document_root_t (C++ struct), 83, 84
udho::configs::server::_document_root_t::component (C++ type), 84
udho::configs::server::get (C++ function), 83
udho::configs::server::mime (C++ function), 83
udho::configs::server::mime_default (C++ member), 83
udho::configs::server::mime_default_t (C++ struct), 84, 85
udho::configs::server::mime_default_t::component (C++ type), 84, 85
udho::configs::server::mime_map (C++ type), 83
udho::configs::server::mimes (C++ member), 83
udho::configs::server::mimes_t (C++

```

```

    struct), 84, 85
udho::configs::server_::mimes_t::component
    (C++ type), 84, 85
udho::configs::server_::server_ (C++
    function), 83
udho::configs::server_::set (C++ function),
    83
udho::configs::server_::template_root
    (C++ member), 83
udho::configs::server_::template_root_t
    (C++ struct), 84, 85
udho::configs::server_::template_root_t:~
    (C++ type), 84, 86
udho::configs::session (C++ type), 276
udho::configs::session_ (C++ struct), 86
udho::configs::session_::_extension
    (C++ member), 87
udho::configs::session_::_id (C++ mem-
    ber), 87
udho::configs::session_::_path (C++ mem-
    ber), 87
udho::configs::session_::_serialization
    (C++ member), 87
udho::configs::session_::extension (C++
    member), 87
udho::configs::session_::extension_t
    (C++ struct), 87, 88
udho::configs::session_::extension_t::com-
    ponent (C++ type), 87, 88
udho::configs::session_::format (C++
    enum), 86
udho::configs::session_::format::binary
    (C++ enumerator), 86
udho::configs::session_::format::text
    (C++ enumerator), 86
udho::configs::session_::format::xml
    (C++ enumerator), 86
udho::configs::session_::get (C++ func-
    tion), 86
udho::configs::session_::id (C++ member),
    87
udho::configs::session_::id_t (C++
    struct), 87, 88
udho::configs::session_::id_t::component
    (C++ type), 87, 88
udho::configs::session_::path (C++ mem-
    ber), 87
udho::configs::session_::path_t (C++
    struct), 87, 88
udho::configs::session_::path_t::component
    (C++ type), 87, 89
udho::configs::session_::serialization
    (C++ member), 87
udho::configs::session_::serialization_t
    (C++ struct), 87, 89
udho::configs::session_::serialization_t::component
    (C++ type), 87, 89
udho::configs::session_::session_ (C++
    function), 86
udho::configs::session_::set (C++ func-
    tion), 86
udho::configuration (C++ struct), 89
udho::configuration::operator[] (C++
    function), 90
udho::configuration_type (C++ type), 276
udho::connection (C++ class), 238
udho::connection::~~connection (C++ func-
    tion), 238
udho::connection::connection (C++ func-
    tion), 238
udho::connection::do_close (C++ func-
    tion), 238
udho::connection::do_read (C++ func-
    tion), 238
udho::connection::on_read (C++ func-
    tion), 238
udho::connection::on_write (C++ func-
    tion), 238
udho::connection::respond (C++ func-
    tion), 238
udho::connection::run (C++ function), 238
udho::connection::send_lambda (C++
    struct), 90
udho::connection::send_lambda::operator()
    (C++ function), 90
udho::connection::send_lambda::self_
    (C++ member), 90
udho::connection::send_lambda::send_lambda
    (C++ function), 90
udho::content_wrapper0 (C++ struct), 90
udho::content_wrapper0::_ftor (C++ mem-
    ber), 91
udho::content_wrapper0::_method (C++
    member), 91
udho::content_wrapper0::content_wrapper0
    (C++ function), 91
udho::content_wrapper0::deferred (C++
    function), 91
udho::content_wrapper0::html (C++ func-
    tion), 91
udho::content_wrapper0::json (C++ func-
    tion), 91
udho::content_wrapper0::mimed (C++ func-
    tion), 91
udho::content_wrapper0::operator= (C++
    function), 91
udho::content_wrapper0::plain (C++ func-
    tion), 91

```

udho::content_wrapper0::raw (C++ function), 91
 udho::content_wrapper0::self_type (C++ type), 91
 udho::content_wrapper0::unwrap (C++ function), 91
 udho::content_wrapper1 (C++ struct), 91
 udho::content_wrapper1::_a1 (C++ member), 92
 udho::content_wrapper1::_ftor (C++ member), 92
 udho::content_wrapper1::_method (C++ member), 92
 udho::content_wrapper1::content_wrapper1 (C++ function), 92
 udho::content_wrapper1::html (C++ function), 92
 udho::content_wrapper1::json (C++ function), 92
 udho::content_wrapper1::mimed (C++ function), 92
 udho::content_wrapper1::operator= (C++ function), 92
 udho::content_wrapper1::plain (C++ function), 92
 udho::content_wrapper1::raw (C++ function), 92
 udho::content_wrapper1::self_type (C++ type), 92
 udho::content_wrapper1::unwrap (C++ function), 92
 udho::context (C++ struct), 93, 294
 udho::context::_session (C++ member), 94
 udho::context::base_type (C++ type), 93
 udho::context::context (C++ function), 93
 udho::context::key_type (C++ type), 93
 udho::context::request_type (C++ type), 93
 udho::context::self_type (C++ type), 93
 udho::context::session (C++ function), 93, 295
 udho::context::session_type (C++ type), 93
 udho::context::shadow_type (C++ type), 93
 udho::context<AuxT, RequestT, void> (C++ struct), 94, 295
 udho::context<AuxT, RequestT, void>::base_type (C++ type), 94
 udho::context<AuxT, RequestT, void>::context (C++ function), 94
 udho::context<AuxT, RequestT, void>::self_type (C++ type), 94
 udho::context<AuxT, RequestT, void>::shadow_type (C++ type), 94
 udho::contexts::stateful (C++ type), 276
 udho::contexts::stateless (C++ type), 276
 udho::cookie (C++ function), 246
 udho::cookie_ (C++ struct), 95
 udho::cookie::_age (C++ member), 95
 udho::cookie::_comment (C++ member), 95
 udho::cookie::_domain (C++ member), 95
 udho::cookie::_name (C++ member), 95
 udho::cookie::_path (C++ member), 95
 udho::cookie::_removed (C++ member), 95
 udho::cookie::_secure (C++ member), 95
 udho::cookie::_value (C++ member), 95
 udho::cookie::age (C++ function), 95
 udho::cookie::cookie_ (C++ function), 95
 udho::cookie::domain (C++ function), 95
 udho::cookie::path (C++ function), 95
 udho::cookie::render (C++ function), 95
 udho::cookie::self_type (C++ type), 95
 udho::cookie::to_string (C++ function), 95
 udho::cookies_ (C++ struct), 96
 udho::cookies::_headers (C++ member), 96
 udho::cookies::_jar (C++ member), 96
 udho::cookies::_request (C++ member), 96
 udho::cookies::add (C++ function), 96
 udho::cookies::collect (C++ function), 96
 udho::cookies::cookie_jar_type (C++ type), 96
 udho::cookies::cookies_ (C++ function), 96
 udho::cookies::exists (C++ function), 96
 udho::cookies::get (C++ function), 96
 udho::cookies::headers_type (C++ type), 96
 udho::cookies::request_type (C++ type), 96
 udho::data_helper (C++ struct), 97
 udho::data_helper<H, T, false> (C++ struct), 97
 udho::data_helper<H, T, false>::data (C++ function), 97
 udho::data_helper<H, T, false>::result_type (C++ type), 97
 udho::data_helper<H, T, true> (C++ struct), 97
 udho::data_helper<H, T, true>::data (C++ function), 97
 udho::data_helper<H, T, true>::result_type (C++ type), 97
 udho::declaration (C++ struct), 98
 udho::declaration::_depth (C++ member), 98
 udho::declaration::_id (C++ member), 98
 udho::declaration::_ref (C++ member), 98
 udho::declaration::declaration (C++ function), 98
 udho::declaration::operator< (C++ function), 98
 udho::defs::request_type (C++ type), 277

```

udho::defs::response_type (C++ type), 277
udho::defs::session_key_type (C++ type),
    277
udho::detail::association (C++ struct), 98
udho::detail::association::_key (C++
    member), 99
udho::detail::association::association
    (C++ function), 99
udho::detail::association::callback_type
    (C++ type), 98
udho::detail::association::matched (C++
    function), 99
udho::detail::association::responder_type
    (C++ type), 98
udho::detail::association::result_type
    (C++ type), 98
udho::detail::association_group (C++
    struct), 99
udho::detail::association_group::_head
    (C++ member), 100
udho::detail::association_group::_tail
    (C++ member), 100
udho::detail::association_group::association_group
    (C++ function), 99
udho::detail::association_group::call
    (C++ function), 99
udho::detail::association_group::count
    (C++ function), 99
udho::detail::association_group::exists
    (C++ function), 99
udho::detail::association_group::head_type
    (C++ type), 99
udho::detail::association_group::iterable
    (C++ function), 99
udho::detail::association_group::keys
    (C++ function), 99
udho::detail::association_group::leaf
    (C++ function), 99
udho::detail::association_group::matched
    (C++ function), 99
udho::detail::association_group::result_type
    (C++ type), 99
udho::detail::association_group::self_type
    (C++ type), 99
udho::detail::association_group::tail_type
    (C++ type), 99
udho::detail::association_group_visitor
    (C++ struct), 101
udho::detail::association_group_visitor<association_group,
    V>, false> (C++ struct), 101
udho::detail::association_group_visitor<association_group,
    V>, false>::_group (C++ member),
    101
udho::detail::association_group_visitor<association_group,
    V>, false>::_tail_visitor (C++
    member), 101
udho::detail::association_group_visitor<association_group,
    V>, false>::count (C++ function), 101
udho::detail::association_group_visitor<association_group,
    V>, false>::find (C++ function), 101
udho::detail::association_group_visitor<association_group,
    V>, false>::group_type (C++ type),
    101
udho::detail::association_group_visitor<association_group,
    V>, false>::keys (C++ function), 101
udho::detail::association_group_visitor<association_group,
    V>, false>::tail_type (C++ type),
    101
udho::detail::association_group_visitor<association_group,
    V>, false>::tail_visitor_type
    (C++ type), 101
udho::detail::association_group_visitor<association_group,
    V>, true> (C++ struct), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::_group (C++ member), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::_tail_visitor (C++
    member), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::association_group_visitor
    (C++ function), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::count (C++ function), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::find (C++ function), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::group_type (C++ type),
    102
udho::detail::association_group_visitor<association_group,
    V>, true>::keys (C++ function), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::result_type (C++ type),
    102
udho::detail::association_group_visitor<association_group,
    V>, true>::tail_type (C++ type), 102
udho::detail::association_group_visitor<association_group,
    V>, true>::tail_visitor_type
    (C++ type), 102
udho::detail::association_group_visitor<association_group,
    V>, false> (C++ struct), 102
udho::detail::association_group_visitor<association_group,
    V>, false>::_group (C++ mem-
    ber), 103
udho::detail::association_group_visitor<association_group,
    V>, false>::_tail_visitor (C++
    member), 101

```

(C++ function), 103
 udho::detail::association_group_visitor<association_group_visitor<C++ member>, 104
 void>, false>::count (C++ function), udho::detail::association_lexical_extractor::_value
 103 (C++ member), 104
 udho::detail::association_group_visitor<association_group_visitor<C++ member>, 104
 void>, false>::find (C++ function), (C++ function), 104
 103 udho::detail::association_lexical_extractor::clear
 udho::detail::association_group_visitor<association_group_visitor<C++ member>, 104
 void>, false>::group_type (C++ udho::detail::association_lexical_extractor::operator
 type), 103 (C++ function), 104
 udho::detail::association_group_visitor<association_group_visitor<C++ member>, 104
 void>, false>::keys (C++ function), (C++ function), 104
 103 udho::detail::association_value_extractor
 udho::detail::association_group<U, (C++ struct), 104
 void> (C++ struct), 100 udho::detail::association_value_extractor::_success
 udho::detail::association_group<U, (C++ member), 105
 void>::_head (C++ member), 100 udho::detail::association_value_extractor::_value
 udho::detail::association_group<U, (C++ member), 105
 void>::association_group (C++ udho::detail::association_value_extractor::association
 function), 100 (C++ function), 104
 udho::detail::association_group<U, udho::detail::association_value_extractor::clear
 void>::call (C++ function), 100 (C++ function), 104
 udho::detail::association_group<U, udho::detail::association_value_extractor::operator
 void>::count (C++ function), 100 (C++ function), 104
 udho::detail::association_group<U, udho::detail::association_value_extractor::value
 void>::exists (C++ function), 100 (C++ function), 104
 udho::detail::association_group<U, udho::detail::async_result (C++ struct), 105
 void>::head_type (C++ type), 100 udho::detail::async_result::_callback
 udho::detail::association_group<U, (C++ member), 106
 void>::iterable (C++ function), 100 udho::detail::async_result::_ctx (C++
 member), 106
 udho::detail::association_group<U, udho::detail::async_result::_ecallback
 void>::keys (C++ function), 100 (C++ member), 106
 udho::detail::association_group<U, udho::detail::async_result::_fcallback
 void>::leaf (C++ function), 100 (C++ member), 106
 udho::detail::association_group<U, udho::detail::async_result::_options
 void>::matched (C++ function), 100 (C++ member), 106
 udho::detail::association_group<U, udho::detail::async_result::after (C++
 void>::result_type (C++ type), 100 function), 106
 udho::detail::association_group<U, udho::detail::async_result::async_result
 void>::self_type (C++ type), 100 (C++ function), 106
 udho::detail::association_group<U, udho::detail::async_result::body (C++
 void>::tail_type (C++ type), 100 function), 106
 udho::detail::association_leaf (C++ udho::detail::async_result::content
 struct), 103 (C++ function), 106
 udho::detail::association_leaf::call (C++ function), 103 udho::detail::async_result::context
 (C++ function), 106 udho::detail::async_result::context_type
 udho::detail::association_leaf::invoke (C++ function), 105
 (C++ function), 103 udho::detail::async_result::done (C++
 udho::detail::association_leaf::matched (C++ function), 106
 (C++ function), 103 udho::detail::async_result::error (C++
 udho::detail::association_leaf::result_type function), 106
 (C++ type), 103 udho::detail::async_result::error_callback_type
 udho::detail::association_lexical_extractor function), 106
 (C++ struct), 104 udho::detail::async_result::error_callback_type

(C++ function), 109
 udho::detail::context_common::query_parser_type (C++ type), 111
 (C++ type), 108
 udho::detail::context_common::render (C++ function), 110
 udho::detail::context_common::request (C++ function), 108
 udho::detail::context_common::request_type (C++ type), 108
 udho::detail::context_common::reroute (C++ function), 110
 udho::detail::context_common::rerouted (C++ function), 110
 udho::detail::context_common::reroutes (C++ function), 110
 udho::detail::context_common::respond (C++ function), 109
 udho::detail::context_common::self_type (C++ type), 108
 udho::detail::context_common::status (C++ function), 109
 udho::detail::context_common::target (C++ function), 109
 udho::detail::context_common::top (C++ function), 110
 udho::detail::context_impl (C++ struct), 110
 udho::detail::context_impl::_cookies (C++ member), 112
 udho::detail::context_impl::_form (C++ member), 112
 udho::detail::context_impl::_headers (C++ member), 112
 udho::detail::context_impl::_query (C++ member), 112
 udho::detail::context_impl::_query_string (C++ member), 112
 udho::detail::context_impl::_request (C++ member), 112
 udho::detail::context_impl::_routes (C++ member), 112
 udho::detail::context_impl::_status (C++ member), 112
 udho::detail::context_impl::alt_path (C++ function), 111
 udho::detail::context_impl::clear (C++ function), 111
 udho::detail::context_impl::context_impl (C++ function), 111
 udho::detail::context_impl::cookies (C++ function), 111
 udho::detail::context_impl::cookies_type (C++ type), 111
 udho::detail::context_impl::form_type (C++ type), 111
 udho::detail::context_impl::headers_type (C++ type), 111
 udho::detail::context_impl::interaction (C++ function), 111
 udho::detail::context_impl::patch (C++ function), 111
 udho::detail::context_impl::path (C++ function), 111
 udho::detail::context_impl::pop (C++ function), 111
 udho::detail::context_impl::push (C++ function), 111
 udho::detail::context_impl::query (C++ function), 111
 udho::detail::context_impl::query_parser_type (C++ type), 111
 udho::detail::context_impl::query_string (C++ function), 111
 udho::detail::context_impl::request (C++ function), 111
 udho::detail::context_impl::request_type (C++ type), 111
 udho::detail::context_impl::reroute (C++ function), 111
 udho::detail::context_impl::rerouted (C++ function), 111
 udho::detail::context_impl::reroutes (C++ function), 111
 udho::detail::context_impl::route_stack_type (C++ type), 111
 udho::detail::context_impl::self_type (C++ type), 111
 udho::detail::context_impl::status (C++ function), 111
 udho::detail::context_impl::target (C++ function), 111
 udho::detail::context_impl::top (C++ function), 111
 udho::detail::http_client_connection (C++ struct), 112
 udho::detail::http_client_connection::_url (C++ member), 113
 udho::detail::http_client_connection::base (C++ type), 112
 udho::detail::http_client_connection::buffer (C++ member), 113
 udho::detail::http_client_connection::context_type (C++ type), 112
 udho::detail::http_client_connection::create (C++ function), 113
 udho::detail::http_client_connection::http_client_ (C++ function), 113
 udho::detail::http_client_connection::on_connect (C++ function), 113

```

udho::detail::http_client_connection::on_write
    (C++ function), 113
udho::detail::http_client_connection::on_write_closed
    (C++ function), 113
udho::detail::http_client_connection::on_write_down
    (C++ function), 113
udho::detail::http_client_connection::on_write_error
    (C++ function), 113
udho::detail::http_client_connection::options_type
    (C++ type), 112
udho::detail::http_client_connection::redirect_type
    (C++ type), 112
udho::detail::http_client_connection::request
    (C++ member), 113
udho::detail::http_client_connection::result
    (C++ member), 113
udho::detail::http_client_connection::result_type
    (C++ member), 113
udho::detail::http_client_connection::resolver
    (C++ member), 113
udho::detail::http_client_connection::resolver_type
    (C++ member), 113
udho::detail::http_client_connection::start
    (C++ function), 113
udho::detail::http_client_connection::stream
    (C++ type), 112
udho::detail::http_client_connection::self_type
    (C++ type), 112
udho::detail::http_client_connection::socket
    (C++ member), 113
udho::detail::http_client_connection::start
    (C++ function), 113
udho::detail::https_client_connection
    (C++ struct), 114
udho::detail::https_client_connection::_ssl_ctx
    (C++ member), 114
udho::detail::https_client_connection::_url
    (C++ member), 114
udho::detail::https_client_connection::base
    (C++ type), 114
udho::detail::https_client_connection::buffer
    (C++ member), 114
udho::detail::https_client_connection::context
    (C++ type), 114
udho::detail::https_client_connection::create
    (C++ function), 115
udho::detail::https_client_connection::detail::connection_prepared
    (C++ function), 114
udho::detail::https_client_connection::on_handshake
    (C++ function), 114
udho::detail::https_client_connection::on_read
    (C++ function), 114
udho::detail::https_client_connection::on_resolve
    (C++ function), 114
udho::detail::https_client_connection::on_shutdown
    (C++ function), 114
udho::detail::https_client_connection::on_write
    (C++ function), 114
udho::detail::https_client_connection::options_type
    (C++ type), 114
udho::detail::https_client_connection::redirector_type
    (C++ type), 114
udho::detail::https_client_connection::request
    (C++ member), 114
udho::detail::https_client_connection::result
    (C++ member), 114
udho::detail::https_client_connection::result_type
    (C++ type), 114
udho::detail::https_client_connection::self_type
    (C++ type), 114
udho::detail::https_client_connection::start
    (C++ function), 114
udho::detail::https_client_connection::stream
    (C++ member), 114
udho::detail::interaction_ (C++ struct), 115
udho::detail::interaction_::_debug (C++
    member), 115
udho::detail::interaction_::_error (C++
    member), 115
udho::detail::interaction_::_info (C++
    member), 115
udho::detail::interaction_::_respond
    (C++ member), 115
udho::detail::interaction_::_warning
    (C++ member), 115
udho::detail::interaction_::_attach (C++
    function), 115
udho::detail::interaction_::_log (C++
    function), 115
udho::detail::interaction_::_respond
    (C++ function), 115
udho::detail::is_prepared (C++ struct), 116
udho::detail::is_prepared::test (C++
    function), 116
udho::detail::is_prepared::[anonymous]
    (C++ enum), 116
udho::detail::is_prepared::[anonymous]::value
    (C++ enumerator), 116
udho::detail::is_streamable (C++ class),
    238
udho::detail::is_streamable::value (C++
    member), 238
udho::detail::operator| (C++ function), 248
udho::detail::responder (C++ struct), 116
udho::detail::responder::_callback (C++
    member), 117

```

udho::detail::responder::call (C++ *function*), 117
 udho::detail::responder::callback_type (C++ *type*), 116
 udho::detail::responder::count (C++ *function*), 117
 udho::detail::responder::iterable (C++ *function*), 117
 udho::detail::responder::keys (C++ *function*), 117
 udho::detail::responder::leaf (C++ *function*), 117
 udho::detail::responder::responder (C++ *function*), 117
 udho::detail::responder::result_type (C++ *type*), 116
 udho::detail::responder::self_type (C++ *type*), 116
 udho::detail::responder<F, std::map<U, V>> (C++ *struct*), 117
 udho::detail::responder<F, std::map<U, V>>::_callback (C++ *member*), 118
 udho::detail::responder<F, std::map<U, V>>::call (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::callback_type (C++ *type*), 117
 udho::detail::responder<F, std::map<U, V>>::container_type (C++ *type*), 117
 udho::detail::responder<F, std::map<U, V>>::count (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::iterable (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::key_type (C++ *type*), 117
 udho::detail::responder<F, std::map<U, V>>::keys (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::leaf (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::responder (C++ *function*), 118
 udho::detail::responder<F, std::map<U, V>>::result_type (C++ *type*), 117
 udho::detail::responder<F, std::map<U, V>>::size_type (C++ *type*), 117
 udho::detail::responder<F, std::map<U, V>>::value_type (C++ *type*), 117
 udho::detail::responder<F, std::vector<V>> (C++ *struct*), 118
 udho::detail::responder<F, std::vector<V>>::_callback (C++ *member*), 119
 udho::detail::responder<F, std::vector<V>>::call (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::callback_type (C++ *type*), 118
 udho::detail::responder<F, std::vector<V>>::container_type (C++ *type*), 118
 udho::detail::responder<F, std::vector<V>>::count (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::iterable (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::keys (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::leaf (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::responder (C++ *function*), 118
 udho::detail::responder<F, std::vector<V>>::result_type (C++ *type*), 118
 udho::detail::responder<F, std::vector<V>>::size_type (C++ *type*), 118
 udho::detail::responder<F, std::vector<V>>::value_type (C++ *type*), 118
 udho::detail::route (C++ *struct*), 119
 udho::detail::route::_path (C++ *member*), 119
 udho::detail::route::_pattern (C++ *member*), 119
 udho::detail::route::_rerouted (C++ *member*), 119
 udho::detail::route::_subject (C++ *member*), 119
 udho::detail::route::reroute (C++ *function*), 119
 udho::detail::route::rerouted (C++ *function*), 119
 udho::detail::route::rerouted_path (C++ *function*), 119
 udho::detail::url_data (C++ *type*), 277
 udho::detail::url_data_ (C++ *struct*), 120
 udho::detail::url_data::_host (C++ *member*), 120
 udho::detail::url_data::_path (C++ *member*), 120
 udho::detail::url_data::_port (C++ *member*), 120
 udho::detail::url_data::_protocol (C++

member), 120

udho::detail::url_data::_query (C++ *member*), 120

udho::detail::url_data::_target (C++ *member*), 120

udho::detail::url_data::_get (C++ *function*), 120

udho::detail::url_data::_host (C++ *member*), 120

udho::detail::url_data::_host_t (C++ *struct*), 120, 122

udho::detail::url_data::_host_t::component (C++ *type*), 121, 122

udho::detail::url_data::_path (C++ *member*), 120

udho::detail::url_data::_path_t (C++ *struct*), 121, 122

udho::detail::url_data::_path_t::component (C++ *type*), 121, 122

udho::detail::url_data::_port (C++ *member*), 120

udho::detail::url_data::_port_t (C++ *struct*), 121, 122

udho::detail::url_data::_port_t::component (C++ *type*), 121, 122

udho::detail::url_data::_protocol (C++ *member*), 120

udho::detail::url_data::_protocol_t (C++ *struct*), 121, 123

udho::detail::url_data::_protocol_t::component (C++ *type*), 121, 123

udho::detail::url_data::_query (C++ *member*), 120

udho::detail::url_data::_query_t (C++ *struct*), 121, 123

udho::detail::url_data::_query_t::component (C++ *type*), 121, 123

udho::detail::url_data::_set (C++ *function*), 120

udho::detail::url_data::_target (C++ *member*), 120

udho::detail::url_data::_target_t (C++ *struct*), 121, 124

udho::detail::url_data::_target_t::component (C++ *type*), 121, 124

udho::detail::url_data::_url_data_ (C++ *function*), 120

udho::detail::visit (C++ *function*), 248

udho::exceptions::http_error (C++ *struct*), 124

udho::exceptions::http_error::_headers (C++ *member*), 125

udho::exceptions::http_error::_message (C++ *member*), 125

udho::exceptions::http_error::_status (C++ *member*), 125

udho::exceptions::http_error::add_header (C++ *function*), 124

udho::exceptions::http_error::headers_type (C++ *type*), 124

udho::exceptions::http_error::http_error (C++ *function*), 124

udho::exceptions::http_error::page (C++ *function*), 125

udho::exceptions::http_error::redirect (C++ *function*), 124

udho::exceptions::http_error::response (C++ *function*), 124

udho::exceptions::http_error::result (C++ *function*), 125

udho::exceptions::http_error::what (C++ *function*), 125

udho::exceptions::http_redirection (C++ *function*), 249

udho::exceptions::reroute (C++ *struct*), 125

udho::exceptions::reroute::_alt_path (C++ *member*), 125

udho::exceptions::reroute::alt_path (C++ *function*), 125

udho::exceptions::reroute::reroute (C++ *function*), 125

udho::field (C++ *struct*), 126

udho::field_common (C++ *struct*), 129

udho::field_common::_err (C++ *member*), 129

udho::field_common::_is_valid (C++ *member*), 129

udho::field_common::_name (C++ *member*), 129

udho::field_common::_validated (C++ *member*), 129

udho::field_common::_value (C++ *member*), 129

udho::field_common::clear (C++ *function*), 129

udho::field_common::dict (C++ *function*), 129

udho::field_common::error (C++ *function*), 129

udho::field_common::field_common (C++ *function*), 129

udho::field_common::name (C++ *function*), 129

udho::field_common::valid (C++ *function*), 129

udho::field_common::validated (C++ *function*), 129

udho::field_value_extractor (C++ *struct*), 130

udho::field_value_extractor::_message (C++ *member*), 130

```

udho::field_value_extractor::field_value_extractor<T, false>::validators_collection_type
    (C++ function), 130
udho::field_value_extractor::message      udho::field<T, false>::value (C++ func-
    (C++ function), 130                    tion), 126
udho::field_value_extractor::operator()   udho::field<T, true> (C++ struct), 127
    (C++ function), 130                   udho::field<T, true>::_message_required
udho::field_value_extractor::value_type   (C++ member), 128
    (C++ type), 130                       udho::field<T, true>::_validators (C++
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>> (C++ udho::field<T, true>::_value (C++ mem-
    struct), 130                          ber), 128
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::_format udho::field<T, true>::check (C++ function),
    (C++ member), 131                    128
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::_message udho::field<T, true>::common_type (C++
    (C++ member), 131                    type), 127
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::field_value (C++ type), 127
    (C++ function), 130                  udho::field<T, true>::constrain (C++
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::message udho::field<T, true>::function_type
    (C++ function), 130                  (C++ type), 127
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::operator() udho::field<T, true>::next_system_type (C++
    (C++ function), 130                  function), 128
udho::field_value_extractor<std::chrono::time_point<std::chrono::system_clock,
    std::chrono::microseconds>>::value_type udho::field<T, true>::self_type (C++
    (C++ type), 130                      type), 127
udho::field<T, false> (C++ struct), 126 udho::field<T, true>::unchecked (C++
udho::field<T, false>::_validators (C++ function), 128
    member), 127                         udho::field<T, true>::validate (C++ func-
udho::field<T, false>::_value (C++ mem-   tion), 128
    ber), 127                           udho::field<T, true>::validators_collection_type
udho::field<T, false>::check (C++ func-   (C++ type), 127
    tion), 126                           udho::field<T, true>::value (C++ function),
udho::field<T, false>::common_type (C++   128
    type), 126                           udho::form::optional (C++ type), 277
udho::field<T, false>::constrain (C++     udho::form::required (C++ type), 278
    function), 126                       udho::form::validate (C++ function), 250
udho::field<T, false>::extractor_type     udho::form::validated (C++ struct), 131
    (C++ type), 126                     udho::form::validated::_errors (C++ mem-
udho::field<T, false>::field (C++ func-   ber), 132
    tion), 126                           udho::form::validated::_fields (C++ mem-
udho::field<T, false>::function_type       ber), 132
    (C++ type), 126                     udho::form::validated::_form (C++ mem-
udho::field<T, false>::operator() (C++     ber), 132
    function), 126, 127                 udho::form::validated::_submitted (C++
udho::field<T, false>::self_type (C++      member), 132
    type), 126                           udho::form::validated::_valid (C++ mem-
udho::field<T, false>::unchecked (C++      ber), 132
    function), 127                       udho::form::validated::add (C++ func-
udho::field<T, false>::validate (C++       tion), 131
    function), 126                       udho::form::validated::base (C++ type), 131
                                         udho::form::validated::dict (C++ function),
                                         131

```

udho::form::validated::errors (C++ *function*), 131

udho::form::validated::form_type (C++ *type*), 131

udho::form::validated::operator[] (C++ *function*), 131

udho::form::validated::self_type (C++ *type*), 131

udho::form::validated::valid (C++ *function*), 131

udho::form::validated::validated (C++ *function*), 131

udho::form::validated<void> (C++ *struct*), 132

udho::form::validated<void>::_errors (C++ *member*), 133

udho::form::validated<void>::_fields (C++ *member*), 133

udho::form::validated<void>::_submitted (C++ *member*), 133

udho::form::validated<void>::_valid (C++ *member*), 133

udho::form::validated<void>::add (C++ *function*), 132

udho::form::validated<void>::base (C++ *type*), 132

udho::form::validated<void>::dict (C++ *function*), 132

udho::form::validated<void>::errors (C++ *function*), 132

udho::form::validated<void>::operator[] (C++ *function*), 132

udho::form::validated<void>::self_type (C++ *type*), 132

udho::form::validated<void>::valid (C++ *function*), 132

udho::form::validated<void>::validated (C++ *function*), 132

udho::form::validators::all_digit (C++ *struct*), 133

udho::form::validators::all_digit::_custom (C++ *member*), 133

udho::form::validators::all_digit::all_digit (C++ *function*), 133

udho::form::validators::all_digit::operator() (C++ *function*), 133

udho::form::validators::date_time (C++ *struct*), 133

udho::form::validators::date_time::_custom (C++ *member*), 134

udho::form::validators::date_time::_format (C++ *member*), 134

udho::form::validators::date_time::date_time (C++ *function*), 133

udho::form::validators::date_time::operator() (C++ *function*), 133

udho::form::validators::exact_length (C++ *struct*), 134

udho::form::validators::exact_length::_custom (C++ *member*), 134

udho::form::validators::exact_length::_length (C++ *member*), 134

udho::form::validators::exact_length::exact_length (C++ *function*), 134

udho::form::validators::exact_length::operator() (C++ *function*), 134

udho::form::validators::max_length (C++ *struct*), 134

udho::form::validators::max_length::_custom (C++ *member*), 134

udho::form::validators::max_length::_length (C++ *member*), 134

udho::form::validators::max_length::max_length (C++ *function*), 134

udho::form::validators::max_length::operator() (C++ *function*), 134

udho::form::validators::min_length (C++ *struct*), 135

udho::form::validators::min_length::_custom (C++ *member*), 135

udho::form::validators::min_length::_length (C++ *member*), 135

udho::form::validators::min_length::min_length (C++ *function*), 135

udho::form::validators::min_length::operator() (C++ *function*), 135

udho::form::validators::no_space (C++ *struct*), 135

udho::form::validators::no_space::_custom (C++ *member*), 135

udho::form::validators::no_space::no_space (C++ *function*), 135

udho::form::validators::no_space::operator() (C++ *function*), 135

udho::form_ (C++ *struct*), 136, 301

udho::form::_::_multipart (C++ *member*), 137

udho::form::_::_request (C++ *member*), 137

udho::form::_::_type (C++ *member*), 137

udho::form::_::_urlencoded (C++ *member*), 137

udho::form::_::body_type (C++ *type*), 136

udho::form::_::count (C++ *function*), 136, 302

udho::form::_::field (C++ *function*), 136, 301

udho::form::_::fields_map_type (C++ *type*), 136

udho::form::_::form_ (C++ *function*), 136

udho::form::_::has (C++ *function*), 136, 301

udho::form::_::is_multipart (C++ *function*), 136, 301

```

udho::form::is_urlencoded (C++ function), 136, 301
udho::form::multipart (C++ function), 136, 301
udho::form::parse_multipart (C++ function), 136, 301
udho::form::parse_urlencoded (C++ function), 136, 301
udho::form::parsed (C++ function), 136, 301
udho::form::request_type (C++ type), 136
udho::form::urlencoded (C++ function), 136, 301
udho::form_type (C++ enum), 241
udho::form_type::multipart (C++ enumerator), 241
udho::form_type::unparsed (C++ enumerator), 241
udho::form_type::urlencoded (C++ enumerator), 241
udho::forms::accumulated (C++ struct), 137
udho::forms::accumulated::accumulated (C++ function), 137
udho::forms::accumulated::add (C++ function), 137
udho::forms::accumulated::add_error (C++ function), 137
udho::forms::accumulated::base (C++ type), 137
udho::forms::accumulated::dict (C++ function), 137
udho::forms::accumulated::errors (C++ function), 137
udho::forms::accumulated::operator[] (C++ function), 137
udho::forms::accumulated::self_type (C++ type), 137
udho::forms::accumulated::valid (C++ function), 137
udho::forms::constraints::all_digits (C++ struct), 138
udho::forms::constraints::all_digits::all_digits (C++ function), 138
udho::forms::constraints::all_digits::base (C++ type), 138
udho::forms::constraints::all_digits::operator() (C++ function), 138
udho::forms::constraints::basic_constraint (C++ struct), 138
udho::forms::constraints::basic_constraint::base (C++ function), 139
udho::forms::constraints::basic_constraint::derived_type (C++ type), 138
udho::forms::constraints::basic_constraint::error (C++ function), 139
udho::forms::constraints::basic_constraint::message (C++ function), 139
udho::forms::constraints::basic_constraint::self (C++ function), 139
udho::forms::constraints::eq (C++ struct), 139
udho::forms::constraints::eq::_value (C++ member), 139
udho::forms::constraints::eq::base (C++ type), 139
udho::forms::constraints::eq::eq (C++ function), 139
udho::forms::constraints::eq::operator() (C++ function), 139
udho::forms::constraints::gt (C++ struct), 140
udho::forms::constraints::gt::_value (C++ member), 140
udho::forms::constraints::gt::base (C++ type), 140
udho::forms::constraints::gt::gt (C++ function), 140
udho::forms::constraints::gt::operator() (C++ function), 140
udho::forms::constraints::gt<std::string> (C++ struct), 141
udho::forms::constraints::gt<std::string>::_value (C++ member), 141
udho::forms::constraints::gt<std::string>::base (C++ type), 141
udho::forms::constraints::gt<std::string>::gt (C++ function), 141
udho::forms::constraints::gt<std::string>::operator() (C++ function), 141
udho::forms::constraints::gte (C++ struct), 141
udho::forms::constraints::gte::_value (C++ member), 142
udho::forms::constraints::gte::base (C++ type), 141
udho::forms::constraints::gte::gte (C++ function), 142
udho::forms::constraints::gte::operator() (C++ function), 142
udho::forms::constraints::gte<std::string> (C++ struct), 142
udho::forms::constraints::gte<std::string>::_value (C++ member), 142
udho::forms::constraints::gte<std::string>::base (C++ type), 142
udho::forms::constraints::gte<std::string>::gte (C++ function), 142
udho::forms::constraints::gte<std::string>::operator() (C++ function), 142

```

```

udho::forms::constraints::in (C++ struct), 143
udho::forms::constraints::in::_values (C++ member), 143
udho::forms::constraints::in::base (C++ type), 143
udho::forms::constraints::in::in (C++ function), 143
udho::forms::constraints::in::operator() (C++ function), 143
udho::forms::constraints::length_eq (C++ struct), 144
udho::forms::constraints::length_eq::_length (C++ member), 144
udho::forms::constraints::length_eq::base (C++ type), 144
udho::forms::constraints::length_eq::length_eq (C++ function), 144
udho::forms::constraints::length_eq::operator() (C++ function), 144
udho::forms::constraints::length_gt (C++ type), 278
udho::forms::constraints::length_gte (C++ type), 278
udho::forms::constraints::length_lt (C++ type), 278
udho::forms::constraints::length_lte (C++ type), 278
udho::forms::constraints::length_max (C++ type), 279
udho::forms::constraints::length_min (C++ type), 279
udho::forms::constraints::lt (C++ struct), 144
udho::forms::constraints::lt::_value (C++ member), 145
udho::forms::constraints::lt::base (C++ type), 145
udho::forms::constraints::lt::lt (C++ function), 145
udho::forms::constraints::lt::operator() (C++ function), 145
udho::forms::constraints::lt<std::string> (C++ struct), 145
udho::forms::constraints::lt<std::string>::_value (C++ member), 146
udho::forms::constraints::lt<std::string>::base (C++ type), 145
udho::forms::constraints::lt<std::string>::lt (C++ function), 145
udho::forms::constraints::lt<std::string>::operator() (C++ function), 145
udho::forms::constraints::lte (C++ struct), 146
udho::forms::constraints::lte::_value (C++ member), 146
udho::forms::constraints::lte::base (C++ type), 146
udho::forms::constraints::lte::lte (C++ function), 146
udho::forms::constraints::lte::operator() (C++ function), 146
udho::forms::constraints::lte<std::string> (C++ struct), 147
udho::forms::constraints::lte<std::string>::_value (C++ member), 147
udho::forms::constraints::lte<std::string>::base (C++ type), 147
udho::forms::constraints::lte<std::string>::lte (C++ function), 147
udho::forms::constraints::lte<std::string>::operator() (C++ function), 147
udho::forms::constraints::max (C++ type), 279
udho::forms::constraints::min (C++ type), 279
udho::forms::constraints::neq (C++ struct), 147
udho::forms::constraints::neq::_value (C++ member), 148
udho::forms::constraints::neq::base (C++ type), 147
udho::forms::constraints::neq::neq (C++ function), 148
udho::forms::constraints::neq::operator() (C++ function), 148
udho::forms::constraints::no_space (C++ struct), 148
udho::forms::constraints::no_space::base (C++ type), 148
udho::forms::constraints::no_space::no_space (C++ function), 148
udho::forms::constraints::no_space::operator() (C++ function), 148
udho::forms::default_datetime_format (C++ member), 273
udho::forms::deserializer (C++ struct), 149
udho::forms::deserializer<std::chrono::time_point<std::chrono::DurationT>, std::string> (C++ struct), 149
udho::forms::deserializer<std::chrono::time_point<std::chrono::DurationT>, std::string>::check (C++ function), 149
udho::forms::deserializer<std::chrono::time_point<std::chrono::DurationT>, std::string>::deserialize (C++ function), 149
udho::forms::deserializer<std::chrono::time_point<std::chrono::DurationT>, std::string>::operator() (C++ function), 149

```

```

    DurationT>,
    std::string>::time_type      (C++
    type), 149
udho::forms::deserializer<std::string,
    std::chrono::time_point<std::chrono::system_clock,
    DurationT>> (C++ struct), 149
udho::forms::deserializer<std::string,
    std::chrono::time_point<std::chrono::system_clock,
    DurationT>>::check (C++ function), 150
udho::forms::deserializer<std::string,
    std::chrono::time_point<std::chrono::system_clock,
    DurationT>>::deserialize (C++ function), 150
udho::forms::deserializer<std::string,
    std::chrono::time_point<std::chrono::system_clock,
    DurationT>>::time_type (C++ type), 149
udho::forms::deserializer<std::string,
    std::string> (C++ struct), 150
udho::forms::deserializer<std::string,
    std::string>::check (C++ function), 150
udho::forms::deserializer<std::string,
    std::string>::deserialize (C++ function), 150
udho::forms::deserializer<std::string,
    U> (C++ struct), 150
udho::forms::deserializer<std::string,
    U>::check (C++ function), 150
udho::forms::deserializer<std::string,
    U>::deserialize (C++ function), 150
udho::forms::deserializer<U, U> (C++ struct), 151
udho::forms::deserializer<U, U>::check (C++ function), 151
udho::forms::deserializer<U, U>::deserialize (C++ function), 151
udho::forms::deserializer<V,
    std::string> (C++ struct), 151
udho::forms::deserializer<V,
    std::string>::check (C++ function), 151
udho::forms::deserializer<V,
    std::string>::deserialize (C++ function), 151
udho::forms::detail::basic_field (C++ struct), 151
udho::forms::detail::basic_field::_message_absent (C++ member), 152
udho::forms::detail::basic_field::_message_unparsable (C++ member), 152
udho::forms::detail::basic_field::absent (C++ function), 152
udho::forms::detail::basic_field::basic_field (C++ function), 152
udho::forms::detail::basic_field::constrain (C++ function), 152
udho::forms::detail::basic_field::data (C++ function), 152
udho::forms::detail::basic_field::data_type (C++ type), 152
udho::forms::detail::basic_field::derived_type (C++ type), 152
udho::forms::detail::basic_field::self (C++ function), 152
udho::forms::detail::basic_field::unparsable (C++ function), 152
udho::forms::detail::basic_field::validate (C++ function), 152
udho::forms::detail::basic_field::value_type (C++ type), 152
udho::forms::detail::constrained_field (C++ struct), 152
udho::forms::detail::constrained_field::_field (C++ member), 153
udho::forms::detail::constrained_field::_validator (C++ member), 153
udho::forms::detail::constrained_field::constrain (C++ function), 153
udho::forms::detail::constrained_field::constrained (C++ function), 153
udho::forms::detail::constrained_field::field (C++ function), 153
udho::forms::detail::constrained_field::field_type (C++ type), 153
udho::forms::detail::constrained_field::message (C++ function), 153
udho::forms::detail::constrained_field::name (C++ function), 153
udho::forms::detail::constrained_field::operator! (C++ function), 153
udho::forms::detail::constrained_field::operator() (C++ function), 153
udho::forms::detail::constrained_field::operator* (C++ function), 153
udho::forms::detail::constrained_field::self_type (C++ type), 153
udho::forms::detail::constrained_field::valid (C++ function), 153
udho::forms::detail::constrained_field::validate (C++ function), 153
udho::forms::detail::constrained_field::validator_t (C++ type), 153
udho::forms::detail::constrained_field::value (C++ function), 153
udho::forms::detail::constrained_field::value_type (C++ type), 153

```

```

udho::forms::detail::constrained_field::[anonymous] (C++ type), 154
(C++ enum), 153 udho::forms::detail::constrained_field<ValidatorT,
udho::forms::detail::constrained_field::[anonymous] constrained_field<T...>>::value
(C++ enumerator), 153 (C++ function), 154
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>> (C++ constrained_field<T...>>::value_type
struct), 154 (C++ type), 154
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::_head constrained_field<T...>>::[anonymous]
(C++ member), 155 (C++ enum), 154
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::_validator constrained_field<T...>>::[anonymous]::depth
(C++ member), 155 (C++ enumerator), 154
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::constrain (C++ type), 279
(C++ function), 154 udho::forms::detail::constraint_visitor_
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::constrained_field udho::forms::detail::constraint_visitor_::validator
(C++ function), 154 (C++ function), 155
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::field (C++ type), 155
(C++ function), 154 udho::forms::detail::constraint_visitor_<Constrained
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::field_type udho::forms::detail::constraint_visitor_<Constrained
(C++ type), 154 0>>::validator (C++ function), 155
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::head_type 0>>::validator_type (C++ type), 155
(C++ type), 154 udho::forms::detail::field_common (C++
struct), 156
constrained_field<T...>>::message udho::forms::detail::field_common::_absent
(C++ function), 154 (C++ member), 157
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::name (C++ member), 157
(C++ function), 154 udho::forms::detail::field_common::_message
(C++ member), 157
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::operator udho::forms::detail::field_common::_name
(C++ function), 154 (C++ member), 157
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::operator() (C++ member), 157
(C++ function), 154 udho::forms::detail::field_common::_unparsable
(C++ member), 157
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::operator udho::forms::detail::field_common::_valid
(C++ function), 154 (C++ member), 157
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::self_type (C++ function), 156
(C++ type), 154 udho::forms::detail::field_common::_absent
(C++ function), 156
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::valid udho::forms::detail::field_common::error
(C++ function), 154 (C++ function), 156
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::validate (C++ function), 156
(C++ function), 154 udho::forms::detail::field_common::_field_common
(C++ function), 156
udho::forms::detail::constrained_field<ValidatorT,
constrained_field<T...>>::validator udho::forms::detail::field_common::message

```

(C++ function), 156
 udho::forms::detail::field_common::name (C++ function), 156
 udho::forms::detail::field_common::operator (C++ function), 156
 udho::forms::detail::field_common::unparsable (C++ function), 156
 udho::forms::detail::field_common::valid (C++ function), 156
 udho::forms::detail::field_common::value (C++ function), 156
 udho::forms::detail::field_data (C++ struct), 157
 udho::forms::detail::field_data::common (C++ function), 158
 udho::forms::detail::field_data::operator (C++ function), 158
 udho::forms::detail::field_data::self_type (C++ type), 157
 udho::forms::detail::field_data::value (C++ function), 158
 udho::forms::detail::field_data::value_type (C++ type), 157
 udho::forms::drivers::combo (C++ struct), 158
 udho::forms::drivers::combo::_request (C++ member), 159
 udho::forms::drivers::combo::_type (C++ member), 159
 udho::forms::drivers::combo::body_type (C++ type), 158
 udho::forms::drivers::combo::combo (C++ function), 159
 udho::forms::drivers::combo::empty (C++ function), 159
 udho::forms::drivers::combo::exists (C++ function), 159
 udho::forms::drivers::combo::is_multipart (C++ function), 159
 udho::forms::drivers::combo::is_urlencoded (C++ function), 159
 udho::forms::drivers::combo::multipart (C++ function), 159
 udho::forms::drivers::combo::multipart_type (C++ type), 158
 udho::forms::drivers::combo::parsable (C++ function), 159
 udho::forms::drivers::combo::parse_multipart (C++ function), 159
 udho::forms::drivers::combo::parse_urlencoded (C++ function), 159
 udho::forms::drivers::combo::parsed (C++ function), 159
 udho::forms::drivers::combo::request_type (C++ type), 158
 udho::forms::drivers::combo::types (C++ enum), 158
 udho::forms::drivers::combo::types::multipart (C++ enumerator), 158
 udho::forms::drivers::combo::types::unparsed (C++ enumerator), 158
 udho::forms::drivers::combo::types::urlencoded (C++ enumerator), 158
 udho::forms::drivers::combo::urlencoded (C++ function), 159
 udho::forms::drivers::combo::urlencoded_type (C++ type), 158
 udho::forms::drivers::detail::extract_multipart_body (C++ struct), 159
 udho::forms::drivers::detail::extract_multipart_body (C++ member), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ member), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ function), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ function), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ function), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ function), 160
 udho::forms::drivers::detail::extract_multipart_body (C++ type), 160
 udho::forms::drivers::multipart (C++ struct), 160
 udho::forms::drivers::multipart::_request (C++ member), 161
 udho::forms::drivers::multipart::body_type (C++ type), 160
 udho::forms::drivers::multipart::multipart (C++ function), 161
 udho::forms::drivers::multipart::multipart_type (C++ type), 160
 udho::forms::drivers::multipart::request_type (C++ type), 160
 udho::forms::drivers::multipart_ (C++ struct), 161
 udho::forms::drivers::multipart_::_boundary (C++ member), 162
 udho::forms::drivers::multipart_::_parts (C++ member), 162
 udho::forms::drivers::multipart_::bounded_string (C++ type), 161
 udho::forms::drivers::multipart_::bounded_string_type (C++ type), 161
 udho::forms::drivers::multipart_::count (C++ function), 161
 udho::forms::drivers::multipart_::empty (C++ function), 161
 udho::forms::drivers::multipart_::exists (C++ function), 161

(C++ function), 161
 udho::forms::drivers::multipart_::form_path (C++ struct), 162, 163
 udho::forms::drivers::multipart_::form_path::body (C++ member), 163, 164
 udho::forms::drivers::multipart_::form_path::headers (C++ member), 163, 164
 udho::forms::drivers::multipart_::form_path::body::udho::forms::drivers::urlencoded_ (C++ struct), 165
 udho::forms::drivers::multipart_::form_path::copy::udho::forms::drivers::urlencoded_::_fields (C++ member), 166
 udho::forms::drivers::multipart_::form_path::empty::udho::forms::drivers::urlencoded_::_query (C++ member), 166
 udho::forms::drivers::multipart_::form_path::filename::udho::forms::drivers::urlencoded_::bounded_string (C++ type), 165
 udho::forms::drivers::multipart_::form_path::formpart::udho::forms::drivers::urlencoded_::bounded_string (C++ type), 165
 udho::forms::drivers::multipart_::form_path::headers::udho::forms::drivers::urlencoded_::empty (C++ function), 165
 udho::forms::drivers::multipart_::form_path::names::udho::forms::drivers::urlencoded_::exists (C++ function), 165
 udho::forms::drivers::multipart_::form_path::parsable::udho::forms::drivers::urlencoded_::header_map_type (C++ type), 165
 udho::forms::drivers::multipart_::form_path::parsed::udho::forms::drivers::urlencoded_::iterator_type (C++ type), 165
 udho::forms::drivers::multipart_::form_path::set::udho::forms::drivers::urlencoded_::parsable (C++ function), 165
 udho::forms::drivers::multipart_::form_path::sforms::udho::forms::drivers::urlencoded_::parse (C++ function), 165
 udho::forms::drivers::multipart_::header::udho::forms::drivers::urlencoded_::parsed (C++ function), 165
 udho::forms::drivers::multipart_::iterator::udho::forms::drivers::urlencoded_::string_type (C++ type), 165
 udho::forms::drivers::multipart_::parsable::udho::forms::drivers::urlencoded_::value_type (C++ type), 165
 udho::forms::drivers::multipart_::parse udho::forms::drivers::urlencoded_raw (C++ type), 280
 udho::forms::drivers::multipart_::parse_path::udho::forms::field (C++ struct), 166
 udho::forms::drivers::multipart_::parsed DurationT>, false> (C++ struct), 166
 udho::forms::drivers::multipart_::part DurationT>, false>::_def (C++ member), 167
 udho::forms::drivers::multipart_::string::udho::forms::field<std::chrono::time_point<std::chrono::DurationT>, false>::_format (C++ member), 167
 udho::forms::drivers::multipart_::value_type udho::forms::field<std::chrono::time_point<std::chrono::DurationT>, false>::base (C++ type), 166
 udho::forms::drivers::multipart_raw DurationT>, false>::base (C++ type), 166
 udho::forms::drivers::urlencoded (C++ struct), 164
 udho::forms::drivers::urlencoded::_request udho::forms::field<std::chrono::time_point<std::chrono::DurationT>, false>::def (C++ function), 167
 udho::forms::drivers::urlencoded::body_type udho::forms::field<std::chrono::time_point<std::chrono::DurationT>, false>::fetch (C++

function), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::field (C++ *udho::forms::field<T, false>::def* (C++ *function*), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::self_type (C++ *function*), 169
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::validate (C++ *function*), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::value_type (C++ *function*), 169
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::[anonymous] (C++ *enum*), 166
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, false>::[anonymous]::is_required (C++ *enumerator*), 166
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true> (C++ *struct*), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::_format (C++ *udho::forms::field<T, true>::base* (C++ *member*), 168
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::base (C++ *type*), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::fetch (C++ *udho::forms::field<T, true>::self_type* (C++ *type*), 169
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::field (C++ *function*), 170
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::self_type (C++ *function*), 170
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::[anonymous] (C++ *enum*), 169
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::[anonymous]::is_required (C++ *enumerator*), 169
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::value_type (C++ *struct*), 170
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::[anonymous] (C++ *enum*), 167
 udho::forms::field<std::chrono::time_point<std::chrono::system_clock, DurationT>, true>::[anonymous]::is_required (C++ *enumerator*), 167
 udho::forms::field<T, false> (C++ *struct*), 168
 udho::forms::field<T, false>::_def (C++ *member*), 169
 udho::forms::field<T, false>::base (C++ *type*), 168
 udho::forms::field<T, false>::field (C++ *function*), 169
 udho::forms::field<T, false>::self_type (C++ *type*), 168
 udho::forms::field<T, false>::[anonymous] (C++ *enum*), 168
 udho::forms::field<T, false>::[anonymous]::is_required (C++ *enumerator*), 168
 udho::forms::field<T, true> (C++ *struct*), 169
 udho::forms::field<T, true>::base (C++ *type*), 169
 udho::forms::field<T, true>::field (C++ *function*), 170
 udho::forms::field<T, true>::self_type (C++ *type*), 169
 udho::forms::field<T, true>::[anonymous] (C++ *enum*), 169
 udho::forms::field<T, true>::[anonymous]::is_required (C++ *enumerator*), 169
 udho::forms::form (C++ *struct*), 170
 udho::forms::form::field (C++ *function*), 170
 udho::forms::form::form_ (C++ *type*), 280
 udho::forms::form::form_multipart (C++ *function*), 280
 udho::forms::form::form_multipart_ (C++ *type*), 280
 udho::forms::optional (C++ *type*), 280
 udho::forms::parser (C++ *struct*), 171
 udho::forms::parser::parsable (C++ *function*), 171
 udho::forms::parser::parse (C++ *function*), 171

171	(*) (A1&, V...)>::_a1 (C++ member), 175
udho::forms::query_ (C++ type), 281	udho::internal::callable1<A1, R (*) (A1&, V...)>::_callback (C++ member), 175
udho::forms::required (C++ type), 281	udho::internal::callable1<A1, R (*) (A1&, V...)>::callable1 (C++ function), 174
udho::forms::validate (C++ function), 254	udho::internal::callable1<A1, R (*) (A1&, V...)>::callback_type (C++ type), 174
udho::forms::validated (C++ struct), 171	udho::internal::callable1<A1, R (*) (A1&, V...)>::function_type (C++ type), 174
udho::forms::validated<form<DriverT>> (C++ struct), 171	udho::internal::callable1<A1, R (*) (A1&, V...)>::operator() (C++ function), 174
udho::forms::validated<form<DriverT>>::add (C++ function), 172	udho::internal::callable1<A1, R (*) (A1&, V...)>::return_type (C++ type), 174
udho::forms::validated<form<DriverT>>::form_type (C++ type), 171	udho::internal::cast_optionally (C++ struct), 175
udho::forms::validated<form<DriverT>>::set_function_type (C++ type), 171	udho::internal::cast_optionally::cast (C++ function), 175
udho::forms::validated<form<DriverT>>::validate (C++ function), 172	udho::internal::function_signature (C++ struct), 175
udho::internal::actual_callback_type (C++ struct), 172	udho::internal::function_signature<boost::function<R (*) (Args...)> (C++ struct), 175
udho::internal::actual_callback_type::callback_type (C++ type), 172	udho::internal::function_signature<boost::function<R (*) (Args...)> (C++ type), 175
udho::internal::arg_to_tuple (C++ struct), 172	udho::internal::function_signature<boost::function<R (*) (Args...)> (C++ type), 175
udho::internal::arg_to_tuple::base_type (C++ type), 172	udho::internal::function_signature<R (*) (Args...)>::arguments_type (C++ type), 176
udho::internal::arg_to_tuple::convert (C++ function), 172	udho::internal::function_signature<R (*) (Args...)>::base_function_type (C++ type), 176
udho::internal::arg_to_tuple<TupleT, 0> (C++ struct), 173	udho::internal::function_signature<R (*) (Args...)>::binding_type (C++ type), 176
udho::internal::arg_to_tuple<TupleT, 0>::convert (C++ function), 173	udho::internal::function_signature<R (*) (Args...)>::object_type (C++ type), 176
udho::internal::arguments_to_tuple (C++ function), 256	udho::internal::html_summary (C++ function), 256
udho::internal::bind_first (C++ struct), 173	udho::internal::member (C++ function), 256
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ struct), 173	udho::internal::member<C++ struct> (C++ type), 176
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ member), 174	udho::internal::member_<R C::*> (C++ struct), 176
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ member), 174	udho::internal::member_<R C::*>::function (C++ member), 177
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ type), 173	
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ type), 173	
udho::internal::bind_first<boost::function<R (*) (Args...)> (C++ type), 173	
udho::internal::callable1 (C++ struct), 174	
udho::internal::callable1<A1, R (*) (A1&, V...)> (C++ struct), 174	
udho::internal::callable1<A1, R (*) (A1&, V...)> (C++ struct), 174	

```

udho::internal::member_<R
    C::*>::const_base_function_type
    (C++ type), 176
udho::internal::member_<R
    C::*>::const_binder_type    (C++
    type), 176
udho::internal::member_<R
    C::*>::const_reduced_function_type
    (C++ type), 176
udho::internal::member_<R
    C::*>::member_ (C++ function), 177
udho::internal::member_<R
    C::*>::object_type    (C++    type),
    176
udho::internal::member_<R
    C::*>::reduced (C++ function), 177
udho::internal::member_<R
    C::*>::result_type    (C++    type),
    176
udho::internal::member_<R
    C::*>::that_type (C++ type), 176
udho::internal::path_cat (C++ function), 256
udho::internal::path_inside (C++ function),
    257
udho::internal::reduced (C++ function), 257
udho::internal::reduced_ (C++ struct), 177
udho::internal::reduced_<R
    (C::*) (Args...)    const>    (C++
    struct), 177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::_function    (C++    member),
    178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::actual_function_type
    (C++ type), 177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::base_function_type    (C++
    type), 177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::binder_type    (C++    type),
    177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::object_type    (C++    type),
    177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced (C++ function), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced_    (C++
    function), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced_function_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::result_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::that_type
    (C++ type), 178
udho::internal::reducer (C++ struct), 179
udho::internal::reducer::_actual    (C++
    member), 179
udho::internal::reducer::_that (C++ mem-
    ber), 179
udho::internal::reducer::actual_function_type
    (C++ type), 179
udho::internal::reducer::function_type
    (C++ type), 179
udho::internal::reduced_    (C++    function),
    178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced_function_type
    (C++ type), 177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::result_type    (C++    type),
    177
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::actual_function_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::base_function_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::binder_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::object_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced    (C++
    function), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced_    (C++
    function), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::reduced_function_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::result_type
    (C++ type), 178
udho::internal::reduced_<R
    (C::*) (Args...)
    const>::that_type
    (C++ type), 178
udho::internal::reducer (C++ struct), 179
udho::internal::reducer::_actual    (C++
    member), 179
udho::internal::reducer::_that (C++ mem-
    ber), 179
udho::internal::reducer::actual_function_type
    (C++ type), 179
udho::internal::reducer::function_type
    (C++ type), 179

```

udho::internal::reducer::object_type (C++ type), 179
 udho::internal::reducer::operator() (C++ function), 179
 udho::internal::reducer::reducer (C++ function), 179
 udho::is_application (C++ struct), 179
 udho::is_application::test (C++ function), 180
 udho::is_application::[anonymous] (C++ enum), 179
 udho::is_application::[anonymous]::value (C++ enumerator), 179
 udho::listener (C++ class), 239
 udho::listener::~~listener (C++ function), 239
 udho::listener::accept (C++ function), 239
 udho::listener::listener (C++ function), 239
 udho::listener::on_accept (C++ function), 239
 udho::listener::run (C++ function), 239
 udho::listener::stop (C++ function), 239
 udho::loggers::ostream (C++ type), 281
 udho::loggers::plain (C++ struct), 180
 udho::loggers::plain::_feature_color (C++ member), 180
 udho::loggers::plain::_stream (C++ member), 180
 udho::loggers::plain::log (C++ function), 180
 udho::loggers::plain::operator() (C++ function), 180
 udho::loggers::plain::operator+= (C++ function), 180
 udho::loggers::plain::plain (C++ function), 180
 udho::loggers::plain::self_type (C++ type), 180
 udho::loggers::plain::stream_type (C++ type), 180
 udho::logging::features::colored (C++ struct), 181
 udho::logging::features::colored::_colored (C++ member), 181
 udho::logging::features::colored::colored (C++ function), 181
 udho::logging::message (C++ struct), 181
 udho::logging::message::_content (C++ member), 182
 udho::logging::message::_level (C++ member), 182
 udho::logging::message::_segment (C++ member), 182
 udho::logging::message::message (C++ function), 182
 udho::logging::message::status (C++ member), 182
 udho::logging::message::time (C++ member), 182
 udho::logging::message::time_type (C++ type), 181
 udho::logging::message::what (C++ function), 182
 udho::logging::messages::debug (C++ type), 281
 udho::logging::messages::error (C++ type), 281
 udho::logging::messages::formatted::debug (C++ type), 282
 udho::logging::messages::formatted::error (C++ type), 282
 udho::logging::messages::formatted::info (C++ type), 282
 udho::logging::messages::formatted::message (C++ struct), 182
 udho::logging::messages::formatted::message::_formatted (C++ member), 183
 udho::logging::messages::formatted::message::base_type (C++ type), 182
 udho::logging::messages::formatted::message::message (C++ function), 183
 udho::logging::messages::formatted::message::operator (C++ function), 183
 udho::logging::messages::formatted::message::prepare (C++ function), 183
 udho::logging::messages::formatted::message::self_type (C++ type), 182
 udho::logging::messages::formatted::message::what (C++ function), 183
 udho::logging::messages::formatted::operator<< (C++ function), 258
 udho::logging::messages::formatted::warning (C++ type), 282
 udho::logging::messages::info (C++ type), 282
 udho::logging::messages::warning (C++ type), 283
 udho::logging::segment (C++ enum), 241
 udho::logging::segment::parser (C++ enumerator), 241
 udho::logging::segment::router (C++ enumerator), 241
 udho::logging::segment::server (C++ enumerator), 241
 udho::logging::segment::unknown (C++ enumerator), 241
 udho::logging::segment::user (C++ enumerator), 241

```

udho::logging::status (C++ enum), 241
udho::logging::status::debug (C++ enumerator), 241
udho::logging::status::error (C++ enumerator), 241
udho::logging::status::info (C++ enumerator), 241
udho::logging::status::warning (C++ enumerator), 241
udho::lookup_table (C++ struct), 183
udho::lookup_table<udho::prepared_group<U,
    V>> (C++ struct), 184
udho::lookup_table<udho::prepared_group<U,
    V>>::_level (C++ member), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::_prepared (C++ member), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::_storage (C++ member), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::_storage_depth (C++ member),
    185
udho::lookup_table<udho::prepared_group<U,
    V>>::_storage_named (C++ member),
    185
udho::lookup_table<udho::prepared_group<U,
    V>>::add (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::clear (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::count (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::depth_storage_type (C++ type),
    185
udho::lookup_table<udho::prepared_group<U,
    V>>::down (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::eval (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::extract (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::keys (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::lookup (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::lookup_table (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::named_storage_type (C++ type),
    185
udho::lookup_table<udho::prepared_group<U,
    V>>::operator[] (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::parse (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::prepared_type (C++ type), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::storage_type (C++ type), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::up (C++ function), 185
udho::lookup_table<udho::prepared_group<U,
    V>>::valid (C++ function), 185
udho::lookup_table<udho::prepared<DataT>>
    (C++ struct), 183
udho::lookup_table<udho::prepared<DataT>>::_level
    (C++ member), 184
udho::lookup_table<udho::prepared<DataT>>::_prepared
    (C++ member), 184
udho::lookup_table<udho::prepared<DataT>>::_storage
    (C++ member), 184
udho::lookup_table<udho::prepared<DataT>>::_storage
    (C++ member), 184
udho::lookup_table<udho::prepared<DataT>>::_storage
    (C++ member), 184
udho::lookup_table<udho::prepared<DataT>>::add
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::clear
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::count
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::depth_st
    (C++ type), 183
udho::lookup_table<udho::prepared<DataT>>::down
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::eval
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::extract
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::keys
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::lookup
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::lookup_t
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::named_st
    (C++ type), 183
udho::lookup_table<udho::prepared<DataT>>::operator
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::parse
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::prepared
    (C++ type), 183
udho::lookup_table<udho::prepared<DataT>>::storage
    (C++ type), 183
udho::lookup_table<udho::prepared<DataT>>::up
    (C++ function), 184
udho::lookup_table<udho::prepared<DataT>>::valid
    (C++ function), 184
udho::module_evaluator (C++ struct), 186
udho::module_evaluator<VisitorT,

```

```

    ModuleT, false> (C++ struct), 186
udho::module_evaluator<VisitorT,
    ModuleT, false>::_visitor (C++
    member), 186
udho::module_evaluator<VisitorT,
    ModuleT, false>::module_evaluator
    (C++ function), 186
udho::module_evaluator<VisitorT,
    ModuleT, false>::operator()
    (C++ function), 186
udho::module_evaluator<VisitorT,
    ModuleT, true> (C++ struct), 186
udho::module_evaluator<VisitorT,
    ModuleT, true>::_visitor (C++
    member), 187
udho::module_evaluator<VisitorT,
    ModuleT, true>::module_evaluator
    (C++ function), 186
udho::module_evaluator<VisitorT,
    ModuleT, true>::operator() (C++
    function), 186
udho::module_info (C++ struct), 187
udho::module_info::_children (C++ mem-
    ber), 187
udho::module_info::_compositor (C++ mem-
    ber), 187
udho::module_info::_fptr (C++ member), 187
udho::module_info::_method (C++ member),
    187
udho::module_info::_pattern (C++ member),
    187
udho::module_overload (C++ struct), 187
udho::module_overload::_compositor (C++
    member), 188
udho::module_overload::_function (C++
    member), 188
udho::module_overload::_pattern (C++
    member), 188
udho::module_overload::_request_method
    (C++ member), 188
udho::module_overload::arguments_type
    (C++ type), 187
udho::module_overload::call (C++ function),
    188
udho::module_overload::compositor_type
    (C++ type), 187
udho::module_overload::feasible (C++
    function), 188
udho::module_overload::function_type
    (C++ type), 187
udho::module_overload::info (C++ function),
    188
udho::module_overload::module_overload
    (C++ function), 188
udho::module_overload::operator() (C++
    function), 188
udho::module_overload::operator= (C++
    function), 188
udho::module_overload::pattern (C++ func-
    tion), 188
udho::module_overload::response_type
    (C++ type), 187
udho::module_overload::return_type (C++
    type), 187
udho::module_overload::self_type (C++
    type), 187
udho::module_overload::tuple_type (C++
    type), 187
udho::module_overload<Function,
    compositors::deferred> (C++ struct),
    188
udho::module_overload<Function,
    compositors::deferred>::_compositor
    (C++ member), 189
udho::module_overload<Function,
    compositors::deferred>::_function
    (C++ member), 189
udho::module_overload<Function,
    compositors::deferred>::_pattern
    (C++ member), 189
udho::module_overload<Function,
    compositors::deferred>::_request_method
    (C++ member), 189
udho::module_overload<Function,
    compositors::deferred>::arguments_type
    (C++ type), 188
udho::module_overload<Function,
    compositors::deferred>::call
    (C++ function), 189
udho::module_overload<Function,
    compositors::deferred>::compositor_type
    (C++ type), 188
udho::module_overload<Function,
    compositors::deferred>::feasible
    (C++ function), 189
udho::module_overload<Function,
    compositors::deferred>::function_type
    (C++ type), 188
udho::module_overload<Function,
    compositors::deferred>::info
    (C++ function), 189
udho::module_overload<Function,
    compositors::deferred>::module_overload
    (C++ function), 189
udho::module_overload<Function,
    compositors::deferred>::operator()
    (C++ function), 189
udho::module_overload<Function,

```

compositors::deferred>::operator=	udho::multipart_form::has (C++ function), 189	190, 302
udho::module_overload<Function, compositors::deferred>::pattern	(C++ function), 189	udho::multipart_form::header_map_type (C++ type), 190
udho::module_overload<Function, compositors::deferred>::response_type	(C++ type), 188	udho::multipart_form::iterator_type (C++ type), 190
udho::module_overload<Function, compositors::deferred>::return_type	(C++ type), 188	udho::multipart_form::parse (C++ function), 190
udho::module_overload<Function, compositors::deferred>::self_type	(C++ type), 188	udho::multipart_form::parse_part (C++ function), 190
udho::module_overload<Function, compositors::deferred>::tuple_type	(C++ type), 188	udho::multipart_form::part (C++ function), 190, 302
udho::multipart_form (C++ struct), 189, 302		udho::multipart_form::string_type (C++ type), 190
udho::multipart_form::_boundary	(C++ member), 190	udho::multipart_form::value_type (C++ type), 190
udho::multipart_form::_parts	(C++ member), 190	udho::operator/= (C++ function), 259
udho::multipart_form::bounded_string	(C++ type), 190	udho::overload_group (C++ struct), 193
udho::multipart_form::bounded_string_type	(C++ type), 190	udho::overload_group::_overload
udho::multipart_form::count (C++ function), 190, 302		(C++ member), 194
udho::multipart_form::field (C++ function), 190, 302		udho::overload_group::_parent
udho::multipart_form::form_part	(C++ struct), 190, 191, 302	(C++ member), 194
udho::multipart_form::form_part::_body	(C++ member), 191, 192	udho::overload_group::eval
udho::multipart_form::form_part::_headers	(C++ member), 191, 192	(C++ function), 193
udho::multipart_form::form_part::body	(C++ function), 191, 192, 303	udho::overload_group::listen
udho::multipart_form::form_part::copied	(C++ function), 191, 192	(C++ function), 193
udho::multipart_form::form_part::filename	(C++ function), 191, 192, 303	udho::overload_group::overload_group
udho::multipart_form::form_part::form_part	(C++ function), 190, 192	(C++ function), 193
udho::multipart_form::form_part::header	(C++ function), 190, 192, 303	udho::overload_group::overload_type
udho::multipart_form::form_part::name	(C++ function), 191, 192, 303	(C++ type), 193
udho::multipart_form::form_part::set_header	(C++ function), 190, 192	udho::overload_group::parent_type
udho::multipart_form::form_part::str	(C++ function), 191, 192, 303	(C++ type), 193
udho::multipart_form::form_part::value	(C++ function), 191, 192, 303	udho::overload_group::self_type
		(C++ type), 193
		udho::overload_group::serve
		(C++ function), 193
		udho::overload_group::summary
		(C++ function), 193
		udho::overload_group::terminal
		(C++ function), 193
		udho::overload_group::terminal_type
		(C++ type), 193
		udho::overload_group_helper
		(C++ struct), 197
		udho::overload_group_helper::_overload
		(C++ member), 197
		udho::overload_group_helper::overload_group_helper
		(C++ function), 197
		udho::overload_group_helper::overload_type
		(C++ type), 197
		udho::overload_group_helper::resolve
		(C++ function), 197
		udho::overload_group_helper::response_type
		(C++ type), 197

```
udho::overload_group_helper<OverloadT,  
    void>(C++ struct), 197  
udho::overload_group_helper<OverloadT,  
    void>::_overload(C++ member), 198  
udho::overload_group_helper<OverloadT,  
    void>::overload_group_helper  
        (C++ function), 197  
udho::overload_group_helper<OverloadT,  
    void>::overload_type(C++ type), 197  
udho::overload_group_helper<OverloadT,  
    void>::resolve(C++ function), 197  
udho::overload_group_helper<OverloadT,  
    void>::response_type(C++ type), 197  
udho::overload_group<U, app_<V, Ref>>  
    (C++ struct), 194  
udho::overload_group<U, app_<V,  
    Ref>>::_overload(C++ member),  
    194  
udho::overload_group<U, app_<V,  
    Ref>>::_parent(C++ member), 194  
udho::overload_group<U, app_<V,  
    Ref>>::eval(C++ function), 194  
udho::overload_group<U, app_<V,  
    Ref>>::listen(C++ function), 194  
udho::overload_group<U, app_<V,  
    Ref>>::overload_group(C++ func-  
tion), 194  
udho::overload_group<U, app_<V,  
    Ref>>::overload_type(C++ type),  
    194  
udho::overload_group<U, app_<V,  
    Ref>>::parent_type(C++ type),  
    194  
udho::overload_group<U, app_<V,  
    Ref>>::self_type(C++ type), 194  
udho::overload_group<U, app_<V,  
    Ref>>::serve(C++ function), 194  
udho::overload_group<U, app_<V,  
    Ref>>::summary(C++ function), 194  
udho::overload_group<U, app_<V,  
    Ref>>::terminal(C++ function), 194  
udho::overload_group<U, app_<V,  
    Ref>>::terminal_type(C++ type),  
    194  
udho::overload_group<U,  
    overload_terminal<V>>(C++ struct),  
    195  
udho::overload_group<U,  
    overload_terminal<V>>::_terminal  
        (C++ member), 195  
udho::overload_group<U,  
    overload_terminal<V>>::eval  
        (C++ function), 195  
udho::overload_group<U,  
    overload_terminal<V>>::parent_type  
        (C++ type), 195  
udho::overload_group<U,  
    overload_terminal<V>>::self_type  
        (C++ type), 195  
udho::overload_group<U,  
    overload_terminal<V>>::serve  
        (C++ function), 195  
udho::overload_group<U,  
    overload_terminal<V>>::summary  
        (C++ function), 195  
udho::overload_group<U,  
    overload_terminal<V>>::terminal  
        (C++ function), 195  
udho::overload_group<U,  
    overload_terminal<V>>::terminal_type  
        (C++ type), 195  
udho::overload_group<U,  
    overload_terminal<void>>(C++  
struct), 196  
udho::overload_group<U,  
    overload_terminal<void>>::_terminal  
        (C++ member), 196  
udho::overload_group<U,  
    overload_terminal<void>>::eval  
        (C++ function), 196  
udho::overload_group<U,  
    overload_terminal<void>>::overload_group  
        (C++ function), 196  
udho::overload_group<U,  
    overload_terminal<void>>::overload_type  
        (C++ type), 196  
udho::overload_group<U,  
    overload_terminal<void>>::parent_type  
        (C++ type), 196  
udho::overload_group<U,  
    overload_terminal<void>>::self_type  
        (C++ type), 196  
udho::overload_group<U,  
    overload_terminal<void>>::serve  
        (C++ function), 196  
udho::overload_group<U,  
    overload_terminal<void>>::summary  
        (C++ function), 196  
udho::overload_group<U,  
    overload_terminal<void>>::terminal  
        (C++ function), 196
```

```

    overload_terminal<void>::terminal_type (C++ function), 202
    (C++ type), 196
udho::overload_terminal (C++ struct), 198
udho::overload_terminal::aux_type (C++ type), 198
udho::overload_terminal<void> (C++ struct), 198
udho::overload_terminal<void>::aux_type (C++ type), 198
udho::overload_terminal<void>::overload_terminal (C++ function), 202
    (C++ function), 198
udho::perform (C++ type), 283
udho::prepare (C++ struct), 199
udho::prepare::fn (C++ function), 199
udho::prepare::index (C++ function), 199
udho::prepare::prepared_type (C++ type), 199
udho::prepare::var (C++ function), 199
udho::prepared (C++ struct), 199
udho::prepared_group (C++ struct), 201
udho::prepared_group::_head (C++ member), 201
udho::prepared_group::_tail (C++ member), 201
udho::prepared_group::at (C++ function), 201
udho::prepared_group::count (C++ function), 201
udho::prepared_group::extract (C++ function), 201
udho::prepared_group::head_type (C++ type), 201
udho::prepared_group::keys (C++ function), 201
udho::prepared_group::operator[] (C++ function), 201
udho::prepared_group::parse (C++ function), 201
udho::prepared_group::prepared_group (C++ function), 201
udho::prepared_group::self_type (C++ type), 201
udho::prepared_group::stringify (C++ function), 201
udho::prepared_group::tail_type (C++ type), 201
udho::prepared_group<H, void> (C++ struct), 202
udho::prepared_group<H, void>::_head (C++ member), 202
udho::prepared_group<H, void>::at (C++ function), 202
udho::prepared_group<H, void>::count (C++ function), 202
udho::prepared_group<H, void>::extract
    overload_terminal<void>::terminal_type (C++ function), 202
    (C++ type), 196
    void>::head_type (C++ type), 202
    udho::prepared_group<H, void>::keys
        (C++ function), 202
    udho::prepared_group<H, void>::operator[] (C++ function), 202
    udho::prepared_group<H, void>::parse
        (C++ function), 202
    udho::prepared_group<H, void>::prepared_group
        (C++ function), 202
    void>::prepared_group (C++ function), 202
    udho::prepared_group<H, void>::self_type (C++ type), 202
    udho::prepared_group<H, void>::stringify (C++ function), 202
    udho::prepared<T, true> (C++ struct), 200
    udho::prepared<T, true>::_data (C++ member), 200
    udho::prepared<T, true>::_index (C++ member), 200
    udho::prepared<T, true>::_visitor (C++ member), 200
    udho::prepared<T, true>::at (C++ function), 200
    udho::prepared<T, true>::count (C++ function), 200
    udho::prepared<T, true>::derived_type (C++ type), 200
    udho::prepared<T, true>::extract (C++ function), 200
    udho::prepared<T, true>::index_type (C++ type), 200
    udho::prepared<T, true>::keys (C++ function), 200
    udho::prepared<T, true>::operator[] (C++ function), 200
    udho::prepared<T, true>::parse (C++ function), 200
    udho::prepared<T, true>::prepared (C++ function), 200
    udho::prepared<T, true>::stringify (C++ function), 200
    udho::prepared<T, true>::visitor_type (C++ type), 200
    udho::proxy (C++ struct), 203
    udho::proxy::_config (C++ member), 203
    udho::proxy::config_type (C++ type), 203
    udho::proxy::key_type (C++ type), 203
    udho::proxy::operator value_type (C++ function), 203
    udho::proxy::operator= (C++ function), 203

```

```

udho::proxy::proxy (C++ function), 203
udho::proxy::proxy_type (C++ type), 203
udho::proxy::value (C++ function), 203
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>> (C++ struct), 203
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::_confid (C++ member), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::at (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::confid (C++ type), 203
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::key_type (C++ type), 203
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::of (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::operator value_type (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::operator= (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::proxy (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::proxy_type (C++ type), 203
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::set (C++ function), 204
udho::proxy<configs::server_<>::mimes_t,
    config<configs::server_<>>>::value (C++ function), 204
udho::require (C++ type), 283
udho::reroute (C++ struct), 204
udho::reroute::_alt_path (C++ member), 204
udho::reroute::alt_path (C++ function), 204
udho::reroute::operator() (C++ function), 204
udho::reroute::reroute (C++ function), 204
udho::router (C++ type), 283
udho::scope (C++ function), 269
udho::server (C++ struct), 205
udho::server::_attachment (C++ member), 206
udho::server::_io (C++ member), 206
udho::server::attachment_type (C++ type), 205
udho::server::auxiliary_type (C++ type), 205
udho::server::cache_type (C++ type), 205
udho::server::config (C++ function), 205
udho::server::configuration_type (C++ type), 205
udho::server::context_type (C++ type), 205
udho::server::http_request_type (C++ type), 205
udho::server::logger_type (C++ type), 205
udho::server::operator+= (C++ function), 205
udho::server::operator[] (C++ function), 205
udho::server::request_type (C++ type), 205
udho::server::self_type (C++ type), 205
udho::server::serve (C++ function), 205
udho::server::server (C++ function), 205
udho::server::server<AuxT, void, CacheT> (C++ struct), 206
udho::server<AuxT, void, CacheT>::_attachment (C++ member), 207
udho::server<AuxT, void, CacheT>::_io (C++ member), 207
udho::server<AuxT, void, CacheT>::attachment_type (C++ type), 206
udho::server<AuxT, void, CacheT>::auxiliary_type (C++ type), 206
udho::server<AuxT, void, CacheT>::cache_type (C++ type), 206
udho::server<AuxT, void, CacheT>::config (C++ function), 206
udho::server<AuxT, void, CacheT>::configuration_type (C++ type), 206
udho::server<AuxT, void, CacheT>::context (C++ type), 206
udho::server<AuxT, void, CacheT>::context_type (C++ type), 206
udho::server<AuxT, void, CacheT>::http_request_type (C++ type), 206
udho::server<AuxT, void, CacheT>::operator+= (C++ function), 206
udho::server<AuxT, void, CacheT>::operator[] (C++ function), 206
udho::server<AuxT, void, CacheT>::request_type (C++ type), 206

```


udho::start_ (C++ struct), 211
udho::start_::accessor_type (C++ type), 211
udho::start_::activity_type (C++ type), 211
udho::start_::base (C++ type), 211
udho::start_::collector (C++ function), 212
udho::start_::collector_type (C++ type), 211
udho::start_::data (C++ function), 212
udho::start_::failure (C++ function), 212
udho::start_::start_ (C++ function), 212
udho::start_::success (C++ function), 212
udho::tcp (C++ type), 285
udho::url (C++ struct), 212
udho::url::build (C++ function), 213
udho::url::param (C++ class), 213, 240
udho::url::param::key (C++ function), 213, 240
udho::url::param::param (C++ function), 213, 240
udho::url::param::to_string (C++ function), 213, 240
udho::url::param::value (C++ function), 213, 240
udho::url::parse (C++ function), 213
udho::url::stringify (C++ function), 213
udho::url::url (C++ function), 213
udho::url::urlencoded_type (C++ type), 212
udho::urlencoded_form (C++ struct), 213, 302
udho::urlencoded_form::_fields (C++ member), 214
udho::urlencoded_form::bounded_string (C++ type), 213
udho::urlencoded_form::bounded_string_type (C++ type), 213
udho::urlencoded_form::count (C++ function), 214, 302
udho::urlencoded_form::field (C++ function), 214, 302
udho::urlencoded_form::has (C++ function), 214, 302
udho::urlencoded_form::header_map_type (C++ type), 213
udho::urlencoded_form::iterator_type (C++ type), 213
udho::urlencoded_form::parse (C++ function), 214
udho::urlencoded_form::string_type (C++ type), 213
udho::urlencoded_form::value_type (C++ type), 213
udho::util::charToHex (C++ function), 270
udho::util::from_hex (C++ function), 270
udho::util::hexToChar (C++ function), 270
udho::util::is_invocable (C++ struct), 214
udho::util::to_hex (C++ function), 270
udho::util::urldecode (C++ function), 270
udho::util::urlencode (C++ function), 271
udho::view::expression (C++ function), 271
udho::view::parser (C++ struct), 214
udho::view::parser::_add_class (C++ member), 216
udho::view::parser::_block (C++ member), 215
udho::view::parser::_condition (C++ member), 215
udho::view::parser::_condition_attr (C++ member), 216
udho::view::parser::_eval (C++ member), 215
udho::view::parser::_eval_attr (C++ member), 215
udho::view::parser::_evaluator (C++ member), 215
udho::view::parser::_html (C++ member), 215
udho::view::parser::_loop (C++ member), 215
udho::view::parser::_skip (C++ member), 215
udho::view::parser::_table (C++ member), 215
udho::view::parser::_target (C++ member), 215
udho::view::parser::_text (C++ member), 215
udho::view::parser::_var (C++ member), 215
udho::view::parser::_xml_parser (C++ member), 215
udho::view::parser::add_class_directive_type (C++ type), 215
udho::view::parser::block_directive_type (C++ type), 215
udho::view::parser::condition_attr_directive_type (C++ type), 215
udho::view::parser::condition_directive_type (C++ type), 215
udho::view::parser::eval_attr_directive_type (C++ type), 215
udho::view::parser::eval_directive_type (C++ type), 215
udho::view::parser::evaluator_type (C++ type), 215
udho::view::parser::html_directive_type (C++ type), 215
udho::view::parser::loop_directive_type (C++ type), 215
udho::view::parser::parser (C++ function), 215
udho::view::parser::prepare (C++ function),

215 (C++ member), 218
 udho::view::parser::process (C++ function), udho::view::parsing::attrs::eval::_evaluator
 215 (C++ member), 218
 udho::view::parser::skip_directive_type udho::view::parsing::attrs::eval::_table
 (C++ type), 215 (C++ member), 218
 udho::view::parser::table_type (C++ type), udho::view::parsing::attrs::eval::eval
 215 (C++ function), 218
 udho::view::parser::target_directive_type udho::view::parsing::attrs::eval::evaluator_type
 (C++ type), 215 (C++ type), 218
 udho::view::parser::text_directive_type udho::view::parsing::attrs::eval::operator()
 (C++ type), 215 (C++ function), 218
 udho::view::parser::var_directive_type udho::view::parsing::attrs::eval::prefix
 (C++ type), 215 (C++ function), 218
 udho::view::parser::xml_parser_type udho::view::parsing::attrs::eval::table_type
 (C++ type), 215 (C++ type), 218
 udho::view::parsing::attrs::add_class udho::view::parsing::attrs::target (C++
 (C++ struct), 216 struct), 218
 udho::view::parsing::attrs::add_class::_ctrl udho::view::parsing::attrs::target::_ctrl
 (C++ member), 216 (C++ member), 219
 udho::view::parsing::attrs::add_class::_evaluator udho::view::parsing::attrs::target::_evaluator
 (C++ member), 216 (C++ member), 219
 udho::view::parsing::attrs::add_class::_table udho::view::parsing::attrs::target::_table
 (C++ member), 216 (C++ member), 219
 udho::view::parsing::attrs::add_class::add_class udho::view::parsing::attrs::target::evaluator_type
 (C++ function), 216 (C++ type), 218
 udho::view::parsing::attrs::add_class::evaluator_type udho::view::parsing::attrs::target::operator()
 (C++ type), 216 (C++ function), 219
 udho::view::parsing::attrs::add_class::operator_view udho::view::parsing::attrs::target::prefix
 (C++ function), 216 (C++ function), 219
 udho::view::parsing::attrs::add_class::prefix udho::view::parsing::attrs::target::table_type
 (C++ function), 216 (C++ type), 218
 udho::view::parsing::attrs::add_class::table_type udho::view::parsing::attrs::target::target
 (C++ type), 216 (C++ function), 219
 udho::view::parsing::attrs::condition udho::view::parsing::expression (C++
 (C++ struct), 217 struct), 219
 udho::view::parsing::attrs::condition::_ctrl udho::view::parsing::expression::_table
 (C++ member), 217 (C++ member), 219
 udho::view::parsing::attrs::condition::_evaluator udho::view::parsing::expression::count
 (C++ member), 217 (C++ function), 219
 udho::view::parsing::attrs::condition::_table udho::view::parsing::expression::evaluate
 (C++ member), 217 (C++ function), 219
 udho::view::parsing::attrs::condition::condition udho::view::parsing::expression::expression
 (C++ function), 217 (C++ function), 219
 udho::view::parsing::attrs::condition::evaluator_type udho::view::parsing::expression::fetch
 (C++ type), 217 (C++ function), 219
 udho::view::parsing::attrs::condition::operator_view udho::view::parsing::expression::rpn
 (C++ function), 217 (C++ function), 219
 udho::view::parsing::attrs::condition::prefix udho::view::parsing::expression::scope_type
 (C++ function), 217 (C++ type), 219
 udho::view::parsing::attrs::condition::table_type udho::view::parsing::tags::block (C++
 (C++ type), 217 struct), 220
 udho::view::parsing::attrs::eval (C++ udho::view::parsing::tags::block::_ctrl
 struct), 217 (C++ member), 220
 udho::view::parsing::attrs::eval::_ctrl udho::view::parsing::tags::block::_evaluator

(C++ member), 220 (C++ member), 222
udho::view::parsing::tags::block::_tableudho::view::parsing::tags::html::_table
(C++ member), 220 (C++ member), 222
udho::view::parsing::tags::block::block udho::view::parsing::tags::html::evaluator_type
(C++ function), 220 (C++ type), 222
udho::view::parsing::tags::block::evaluator_typeudho::view::parsing::tags::html::html
(C++ type), 220 (C++ function), 222
udho::view::parsing::tags::block::operator()udho::view::parsing::tags::html::operator()
(C++ function), 220 (C++ function), 222
udho::view::parsing::tags::block::prefixudho::view::parsing::tags::html::prefix
(C++ function), 220 (C++ function), 222
udho::view::parsing::tags::block::table_typeudho::view::parsing::tags::html::table_type
(C++ type), 220 (C++ type), 222
udho::view::parsing::tags::condition udho::view::parsing::tags::loop (C++
(C++ struct), 220 struct), 223
udho::view::parsing::tags::condition::_ctrludho::view::parsing::tags::loop::_ctrl
(C++ member), 221 (C++ member), 223
udho::view::parsing::tags::condition::_evaluatorudho::view::parsing::tags::loop::_evaluator
(C++ member), 221 (C++ member), 223
udho::view::parsing::tags::condition::_tableudho::view::parsing::tags::loop::_table
(C++ member), 221 (C++ member), 223
udho::view::parsing::tags::condition::conditionudho::view::parsing::tags::loop::evaluator_type
(C++ function), 221 (C++ type), 223
udho::view::parsing::tags::condition::evaluator_typeudho::view::parsing::tags::loop::loop
(C++ type), 221 (C++ function), 223
udho::view::parsing::tags::condition::operator()udho::view::parsing::tags::loop::operator()
(C++ function), 221 (C++ function), 223
udho::view::parsing::tags::condition::prefixudho::view::parsing::tags::loop::prefix
(C++ function), 221 (C++ function), 223
udho::view::parsing::tags::condition::table_typeudho::view::parsing::tags::loop::table_type
(C++ type), 221 (C++ type), 223
udho::view::parsing::tags::eval (C++ udho::view::parsing::tags::skip (C++
struct), 221 struct), 223
udho::view::parsing::tags::eval::_ctrl udho::view::parsing::tags::skip::_ctrl
(C++ member), 222 (C++ member), 224
udho::view::parsing::tags::eval::_evaluatorudho::view::parsing::tags::skip::_evaluator
(C++ member), 222 (C++ member), 224
udho::view::parsing::tags::eval::_table udho::view::parsing::tags::skip::_table
(C++ member), 222 (C++ member), 224
udho::view::parsing::tags::eval::eval udho::view::parsing::tags::skip::evaluator_type
(C++ function), 221 (C++ type), 223
udho::view::parsing::tags::eval::evaluator_typeudho::view::parsing::tags::skip::operator()
(C++ type), 221 (C++ function), 224
udho::view::parsing::tags::eval::operator()udho::view::parsing::tags::skip::prefix
(C++ function), 221 (C++ function), 224
udho::view::parsing::tags::eval::prefix udho::view::parsing::tags::skip::skip
(C++ function), 221 (C++ function), 224
udho::view::parsing::tags::eval::table_typeudho::view::parsing::tags::skip::table_type
(C++ type), 221 (C++ type), 223
udho::view::parsing::tags::html (C++ udho::view::parsing::tags::text (C++
struct), 222 struct), 224
udho::view::parsing::tags::html::_ctrl udho::view::parsing::tags::text::_ctrl
(C++ member), 222 (C++ member), 224
udho::view::parsing::tags::html::_evaluatorudho::view::parsing::tags::text::_evaluator

(C++ member), 224
 udho::view::parsing::tags::text::_table (C++ member), 224
 udho::view::parsing::tags::text::evaluator_type (C++ type), 224
 udho::view::parsing::tags::text::operator() (C++ function), 224
 udho::view::parsing::tags::text::prefix (C++ function), 224
 udho::view::parsing::tags::text::table_type (C++ type), 224
 udho::view::parsing::tags::text::text (C++ function), 224
 udho::view::parsing::tags::var (C++ struct), 225
 udho::view::parsing::tags::var::_ctrl (C++ member), 225
 udho::view::parsing::tags::var::_evaluator (C++ member), 225
 udho::view::parsing::tags::var::_table (C++ member), 225
 udho::view::parsing::tags::var::evaluator_type (C++ type), 225
 udho::view::parsing::tags::var::operator() (C++ function), 225
 udho::view::parsing::tags::var::prefix (C++ function), 225
 udho::view::parsing::tags::var::table_type (C++ type), 225
 udho::view::parsing::tags::var::var (C++ function), 225
 udho::view::parsing::token (C++ struct), 225
 udho::view::parsing::token::arithmetic (C++ function), 226
 udho::view::parsing::token::category (C++ enum), 226
 udho::view::parsing::token::category::arithmetic (C++ enumerator), 226
 udho::view::parsing::token::category::junction (C++ function), 227
 udho::view::parsing::token::category::junction (C++ enumerator), 226
 udho::view::parsing::token::category::logical (C++ type), 227
 udho::view::parsing::token::category::logical (C++ enumerator), 226
 udho::view::parsing::token::category::none (C++ type), 227
 udho::view::parsing::token::category::none (C++ enumerator), 226
 udho::view::parsing::token::create (C++ function), 226
 udho::view::parsing::token::function_map_type (C++ type), 226
 udho::view::parsing::token::is (C++ function), 226
 udho::view::parsing::token::logical (C++ function), 226
 udho::view::parsing::token::operator_map_type (C++ type), 226
 udho::view::parsing::token::precedence (C++ function), 226
 udho::view::parsing::token::type (C++ enumerator), 225
 udho::view::parsing::token::type::call (C++ enumerator), 225
 udho::view::parsing::token::type::comma (C++ enumerator), 225
 udho::view::parsing::token::type::function (C++ enumerator), 225
 udho::view::parsing::token::type::id (C++ enumerator), 226
 udho::view::parsing::token::type::op (C++ enumerator), 225
 udho::view::parsing::token::type::parenthesis_close (C++ enumerator), 226
 udho::view::parsing::token::type::parenthesis_open (C++ enumerator), 225
 udho::view::parsing::token::type::real (C++ enumerator), 226
 udho::view::parsing::token::value (C++ function), 226
 udho::view::parsing::tokenize (C++ function), 271
 udho::view::parsing::xml (C++ struct), 226
 udho::view::parsing::xml::_ctx (C++ member), 227
 udho::view::parsing::xml::_directives_attr (C++ member), 227
 udho::view::parsing::xml::_directives_tag (C++ member), 227
 udho::view::parsing::xml::_source (C++ member), 227
 udho::view::parsing::xml::_transformed (C++ member), 227
 udho::view::parsing::xml::add_directive_attr (C++ function), 227
 udho::view::parsing::xml::add_directive_tag (C++ function), 227
 udho::view::parsing::xml::attr_directive (C++ type), 227
 udho::view::parsing::xml::attr_directive_map (C++ type), 227
 udho::view::parsing::xml::clear (C++ function), 227
 udho::view::parsing::xml::context_type (C++ type), 227
 udho::view::parsing::xml::open (C++ function), 227
 udho::view::parsing::xml::output (C++ function), 227
 udho::view::parsing::xml::parse (C++ function), 227
 udho::view::parsing::xml::step_in (C++

[function\), 227](#)
[udho::view::parsing::xml::tag_directive \(C++ type\), 227](#)
[udho::view::parsing::xml::tag_directive_map \(C++ type\), 227](#)
[udho::view::parsing::xml::traverse \(C++ function\), 227](#)
[udho::view::parsing::xml::xml \(C++ function\), 227](#)
[udho::view::processor \(C++ function\), 271](#)
[udho::visit \(C++ function\), 272](#)
[udho::visitor \(C++ struct\), 228](#)
[udho::visitor::operator\(\) \(C++ function\), 228](#)
[udho::visitors::print \(C++ function\), 272](#)
[udho::visitors::print_html \(C++ function\), 272](#)
[udho::visitors::printing_visitor \(C++ struct\), 228](#)
[udho::visitors::printing_visitor::_application \(C++ function\), 228](#)
[udho::visitors::printing_visitor::_indent \(C++ member\), 229](#)
[udho::visitors::printing_visitor::_module \(C++ function\), 228](#)
[udho::visitors::printing_visitor::_stream \(C++ member\), 229](#)
[udho::visitors::printing_visitor::column \(C++ function\), 228](#)
[udho::visitors::printing_visitor::operator\(\) \(C++ function\), 228](#)
[udho::visitors::printing_visitor::printing_visitor \(C++ function\), 228](#)
[udho::visitors::printing_visitor_html \(C++ struct\), 229](#)
[udho::visitors::printing_visitor_html::_application \(C++ function\), 229](#)
[udho::visitors::printing_visitor_html::_indent \(C++ member\), 229](#)
[udho::visitors::printing_visitor_html::_module \(C++ function\), 229](#)
[udho::visitors::printing_visitor_html::_stream \(C++ member\), 229](#)
[udho::visitors::printing_visitor_html::operator\(\) \(C++ function\), 229](#)
[udho::visitors::printing_visitor_html::printing_visitor_html \(C++ function\), 229](#)
[udho::visitors::target \(C++ function\), 272](#)
[udho::visitors::target_application \(C++ struct\), 229](#)
[udho::visitors::target_application::_application \(C++ function\), 230](#)
[udho::visitors::target_application::_callback \(C++ member\), 230](#)
[udho::visitors::target_application::target_application \(C++ function\), 230](#)
[udho::visitors::target_application::target_application<F, Visitables, 0> \(C++ struct\), 230](#)
[udho::visitors::target_application<F, Visitables, 0>::_application \(C++ function\), 230](#)
[udho::visitors::target_application<F, Visitables, 0>::_callback \(C++ member\), 230](#)
[udho::visitors::target_application<F, Visitables, 0>::target_application \(C++ function\), 230](#)
[udho::visitors::target_module \(C++ struct\), 230](#)
[udho::visitors::target_module::_callback \(C++ member\), 231](#)
[udho::visitors::target_module::_module \(C++ function\), 231](#)
[udho::visitors::target_module::target_module \(C++ function\), 231](#)
[udho::visitors::target_module<F, Visitables, 0> \(C++ struct\), 231](#)
[udho::visitors::target_module<F, Visitables, 0>::_callback \(C++ member\), 231](#)
[udho::visitors::target_module<F, Visitables, 0>::_module \(C++ function\), 231](#)
[udho::visitors::target_module<F, Visitables, 0>::target_module \(C++ function\), 231](#)
[udho::visitors::targeted \(C++ type\), 285](#)
[udho::visitors::targetted_visitor \(C++ struct\), 232](#)
[udho::visitors::targetted_visitor::operator\(\) \(C++ function\), 232](#)
[udho::visitors::targetted_visitor::targetted_visitor \(C++ function\), 232](#)
[udho::visitors::visitable \(C++ struct\), 232](#)
[udho::visitors::visitable::application \(C++ member\), 232](#)
[udho::visitors::visitable::both \(C++ member\), 232](#)
[udho::visitors::visitable::callable \(C++ member\), 232](#)
[udho::watch \(C++ struct\), 232](#)
[udho::watch::_expiry \(C++ member\), 233](#)
[udho::watch::_key \(C++ member\), 233](#)
[udho::watch::_released \(C++ member\), 233](#)
[udho::watch::expire_at \(C++ function\), 233](#)
[udho::watch::expiry \(C++ function\), 233](#)
[udho::watch::key \(C++ function\), 233](#)

```

udho::watch::key_type (C++ type), 232
udho::watch::release (C++ function), 233
udho::watch::released (C++ function), 233
udho::watch::self_type (C++ type), 232
udho::watch::time_duration_type (C++
    type), 232
udho::watch::time_point_type (C++ type),
    232
udho::watch::valid (C++ function), 233
udho::watch::watch (C++ function), 233
udho::watcher (C++ struct), 233
udho::watcher::_duration (C++ member), 234
udho::watcher::_index (C++ member), 234
udho::watcher::_io (C++ member), 234
udho::watcher::_mutex (C++ member), 234
udho::watcher::_timer (C++ member), 234
udho::watcher::_watchers (C++ member), 234
udho::watcher::async_notify (C++ function),
    234
udho::watcher::async_notify_all (C++
    function), 234
udho::watcher::container_type (C++ type),
    233
udho::watcher::expired (C++ function), 234
udho::watcher::index_type (C++ type), 233
udho::watcher::insert (C++ function), 234
udho::watcher::iterator_type (C++ type),
    233
udho::watcher::key_type (C++ type), 233
udho::watcher::notify (C++ function), 234
udho::watcher::notify_all (C++ function),
    234
udho::watcher::self_type (C++ type), 233
udho::watcher::time_duration_type (C++
    type), 233
udho::watcher::time_point_type (C++ type),
    233
udho::watcher::watch_type (C++ type), 233
udho::watcher::watcher (C++ function), 234
UDHO_SESSION_COOKIE_NAME (C macro), 273
UDHO_SESSION_FILE_EXTENSION (C macro), 273
UDHO_VERSION (C macro), 274
UDHO_VERSION_STRING (C macro), 274

```