C++ API

Packages

`google::protobuf` (#google.protobuf)
> *Core components of the Protocol Buffers runtime library.*

`google::protobuf::io` (#google.protobuf.io)
> *Auxiliary classes used for I/O.*

`google::protobuf::util` (#google.protobuf.util)
> *Utility classes.*

`google::protobuf::compiler` (#google.protobuf.compiler)
> *Implementation of the Protocol Buffer compiler.*

google::protobuf

Core components of the Protocol Buffers runtime library.

The files in this package represent the core of the Protocol Buffer system. All of them are part of the libprotobuf library.

A note on thread-safety:

Thread-safety in the Protocol Buffer library follows a simple rule: unless explicitly noted otherwise, it is always safe to use an object from multiple threads simultaneously as long as the object is declared const in all threads (or, it is only used in ways that would be allowed if it were declared const). However, if an object is accessed in one thread in a way that would not be allowed if it were const, then it is not safe to access that object in any other thread simultaneously.

Put simply, read-only access to an object can happen in multiple threads simultaneously, but write access can only happen in a single thread at a time.

The implementation does contain some "const" methods which actually modify the object behind the scenes – e.g., to cache results – but in these cases mutex locking is used to make the access thread-safe.

Files

---

google/protobuf/arena.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.arena)

*This file defines an _Arena_
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.arena#Arena)
allocator for better allocation performance.*

google/protobuf/descriptor.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.descriptor)

This file contains classes which describe a type of protocol message.

google/protobuf/descriptor.pb.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.descriptor.pb)

Protocol buffer representations of descriptors.

google/protobuf/descriptor_database.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.descriptor_database)

Interface for manipulating databases of descriptors.

google/protobuf/dynamic_message.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.dynamic_message)

*Defines an implementation of _Message_
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.message#Message)
which can emulate types which are not known at compile-time.*

google/protobuf/map.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.map)

This file defines the map container and its helpers to support protobuf maps.

google/protobuf/message.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.message)

*Defines _Message_
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.message#Message)
, the abstract interface implemented by non-lite protocol message objects.*

google/protobuf/message_lite.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.message_lite)

*Defines _MessageLite_
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.message_lite#MessageLite)
, the abstract interface implemented by all (lite and non-lite) protocol message objects.*

google/protobuf/repeated_field.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.repeated_field)

RepeatedField
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.repeated_field#RepeatedField)
and RepeatedPtrField
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.repeated_field#RepeatedPtrField)
are used by generated protocol message classes to manipulate repeated fields.

google/protobuf/service.h
 (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.service)
   DEPRECATED: This module declares the abstract interfaces underlying proto2 RPC services.

google/protobuf/text_format.h
 (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.text_format)
   Utilities for printing and parsing protocol messages in a human-readable, text-based format.

google/protobuf/unknown_field_set.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.unknown_field_set)
   Contains classes used to keep track of unrecognized fields seen while parsing a protocol message.

google/protobuf/stubs/common.h
 (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.common)
   Contains basic types and utilities used by the rest of the library.

google::protobuf::io

Auxiliary classes used for I/O.

The Protocol Buffer library uses the classes in this package to deal with I/O and
encoding/decoding raw bytes. Most users will not need to deal with this package. However,
users who want to adapt the system to work with their own I/O abstractions – e.g., to allow
Protocol Buffers to be read from a different kind of input stream without the need for a
temporary buffer – should take a closer look.

Files

google/protobuf/io/coded_stream.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.io.coded_stream)
   This file contains the CodedInputStream
    *(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.io.coded_stream#CodedInputStream)*
   and CodedOutputStream

(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.coded_stream#CodedOutputStream)
classes, which wrap a ZeroCopyInputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyInputStream)
or ZeroCopyOutputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyOutputStream)
, respectively, and allow you to read or write individual pieces of data in various formats.

google/protobuf/io/printer.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.printer)

Utility class for writing text to a ZeroCopyOutputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyOutputStream)
.

google/protobuf/io/tokenizer.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.tokenizer)

Class for parsing tokenized text from a ZeroCopyInputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyInputStream)
.

google/protobuf/io/zero_copy_stream.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream)

This file contains the ZeroCopyInputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyInputStream)
and ZeroCopyOutputStream
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream#ZeroCopyOutputStream)
interfaces, which represent abstract I/O streams to and from which protocol buffers can be read and written.

google/protobuf/io/zero_copy_stream_impl.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream_impl)

This file contains common implementations of the interfaces defined in zero_copy_stream.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream)
which are only included in the full (non-lite) protobuf library.

google/protobuf/io/zero_copy_stream_impl_lite.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream_impl_lite)

This file contains common implementations of the interfaces defined in zero_copy_stream.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.io.zero_copy_stream)
which are included in the "lite" protobuf library.

google::protobuf::util

Utility classes.

This package contains various utilities for message comprasion, JSON conversion, well known types, etc.

Files

google/protobuf/util/field_comparator.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.util.field_comparator)

Defines classes for field comparison.

google/protobuf/util/field_mask_util.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.util.field_mask_util)

Defines utilities for the FieldMask well known type.

google/protobuf/util/json_util.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.util.json_util)

Utility functions to convert between protobuf binary format and proto3 JSON format.

google/protobuf/util/message_differencer.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.util.message_differencer)

This file defines static methods and classes for comparing Protocol Messages.

google/protobuf/util/time_util.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.util.time_util)

Defines utilities for the Timestamp and Duration well known types.

google/protobuf/util/type_resolver.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.util.type_resolver)

Defines a TypeResolver for the Any message.

google/protobuf/util/type_resolver_util.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.util.type_resolver_util)

Defines utilities for the TypeResolver.

google::protobuf::compiler

Implementation of the Protocol Buffer compiler.

This package contains code for parsing .proto files and generating code based on them. There are two reasons you might be interested in this package:

- You want to parse .proto files at runtime. In this case, you should look at importer.h (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.importer) . Since this functionality is widely useful, it is included in the libprotobuf base library; you do not have to link against libprotoc.

- You want to write a custom protocol compiler which generates different kinds of code, e.g. code in a different language which is not supported by the official compiler. For this purpose, command_line_interface.h (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.command_line_interface) provides you with a complete compiler front-end, so all you need to do is write a custom implementation of CodeGenerator (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.code_generator#CodeGenerator) and a trivial main() function. You can even make your compiler support the official languages in addition to your own. Since this functionality is only useful to those writing custom compilers, it is in a separate library called "libprotoc" which you will have to link against.

Files

google/protobuf/compiler/code_generator.h (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.code_generator)

Defines the abstract interface implemented by each of the language-specific code generators.

google/protobuf/compiler/command_line_interface.h (https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.command_line_interface)

Implements the Protocol Compiler front-end such that it may be reused by custom compilers written to support other languages.

google/protobuf/compiler/importer.h

(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.importer)
　　This file is the public interface to the .proto file parser.

google/protobuf/compiler/parser.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.parser)
　　Implements parsing of .proto files to FileDescriptorProtos.

google/protobuf/compiler/plugin.h
(https://developers.google.com/protocol-buffers/docs/reference/cpp/google.protobuf.compiler.plugin)
　　Front-end for protoc code generator plugins written in C++.

google/protobuf/compiler/plugin.pb.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.plugin.pb)
　　API for protoc plugins.

google/protobuf/compiler/cpp/cpp_generator.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.cpp_generator)
　　Generates C++ code for a given .proto file.

google/protobuf/compiler/csharp/csharp_generator.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.csharp_generator)
　　Generates C# code for a given .proto file.

google/protobuf/compiler/csharp/csharp_names.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.csharp_names)
　　Provides a mechanism for mapping a descriptor to the fully-qualified name of the corresponding C#
　　class.

google/protobuf/compiler/java/java_generator.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.java_generator)
　　Generates Java code for a given .proto file.

google/protobuf/compiler/java/java_names.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.java_names)
　　Provides a mechanism for mapping a descriptor to the fully-qualified name of the corresponding Java
　　class.

google/protobuf/compiler/javanano/javanano_generator.h
(https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.javanano_generator)
　　Generates Java nano code for a given .proto file.

google/protobuf/compiler/js/js_generator.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.js_generator)

   Generates JavaScript code for a given .proto file.

---

google/protobuf/compiler/objectivec/objectivec_generator.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.objectivec_generator)

   Generates ObjectiveC code for a given .proto file.

---

google/protobuf/compiler/objectivec/objectivec_helpers.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.objectivec_helpers)

   Helper functions for generating ObjectiveC code.

---

google/protobuf/compiler/python/python_generator.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.python_generator)

   Generates Python code for a given .proto file.

---

google/protobuf/compiler/ruby/ruby_generator.h
 (https://developers.google.com/protocol-
buffers/docs/reference/cpp/google.protobuf.compiler.ruby_generator)

   Generates Ruby code for a given .proto file.

Downloads
Protocol buffers downloads
and instructions

GitHub
The latest protocol buffers
code and releases