What’s Inside the Box?

Comparing Data Storage Across Frameworks

@ray_deck

CHAIN REA
ML
SPECIAL THANKS TO
SQUARESPACE
PRESENTED BY
INFINITE RED
CHAIN REACT
CHAIN REA
SPECIAL THANKS TO
SQUARESPACE
PRESENTED BY
INFINITE RED

Code as Data

PyTorch

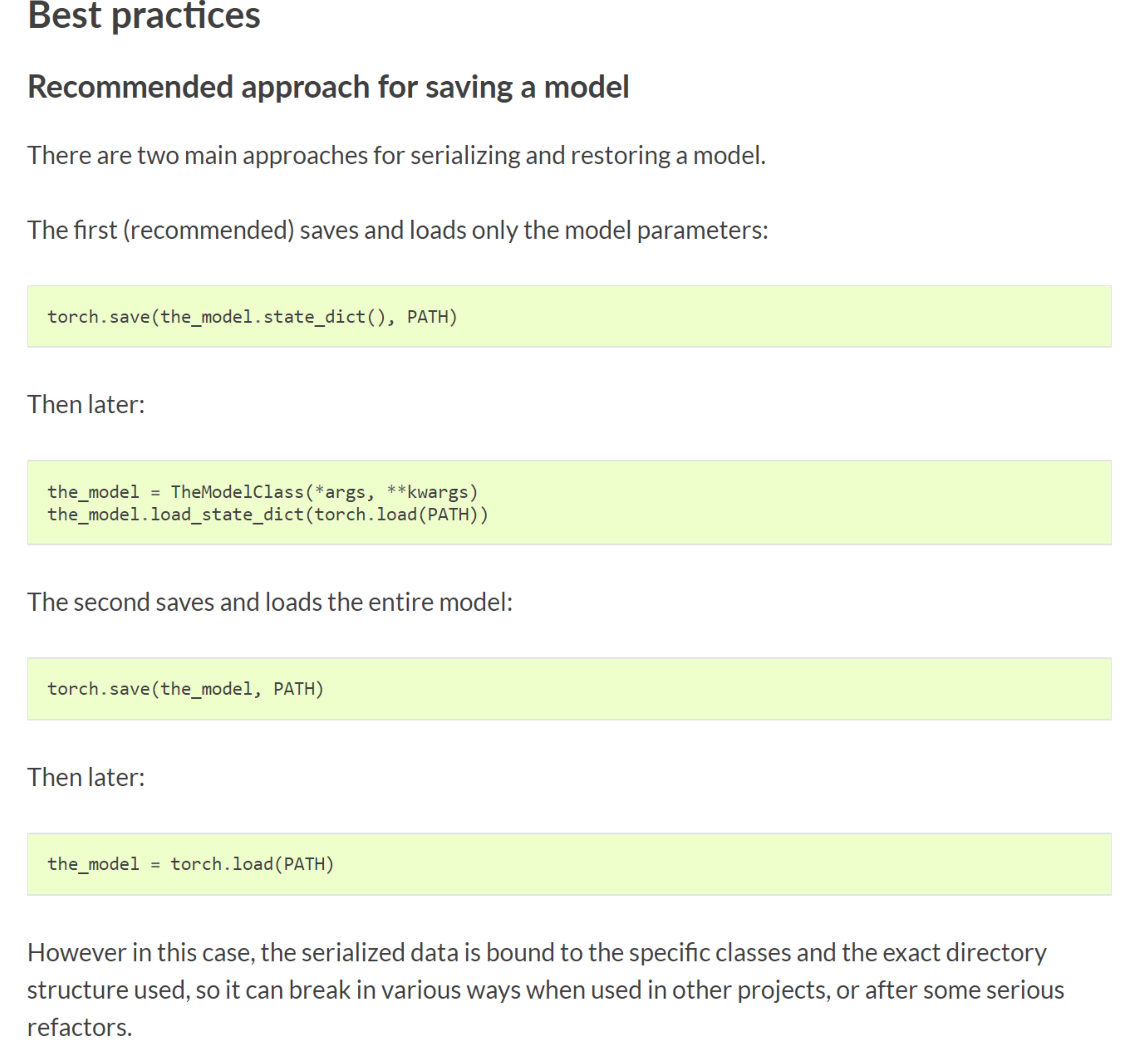
Keras

TensorFlow

TensorFlow Lite

CoreML

PYTORCH



Best practices

Recommended approach for saving a model

There are two main approaches for serializing and restoring a model.

The first (recommended) saves and loads only the model parameters:

```
torch.save(the_model.state_dict(), PATH)
```

Then later:

```
the_model = TheModelClass(*args, **kwargs)
the_model.load_state_dict(torch.load(PATH))
```

The second saves and loads the entire model:

```
torch.save(the_model, PATH)
```

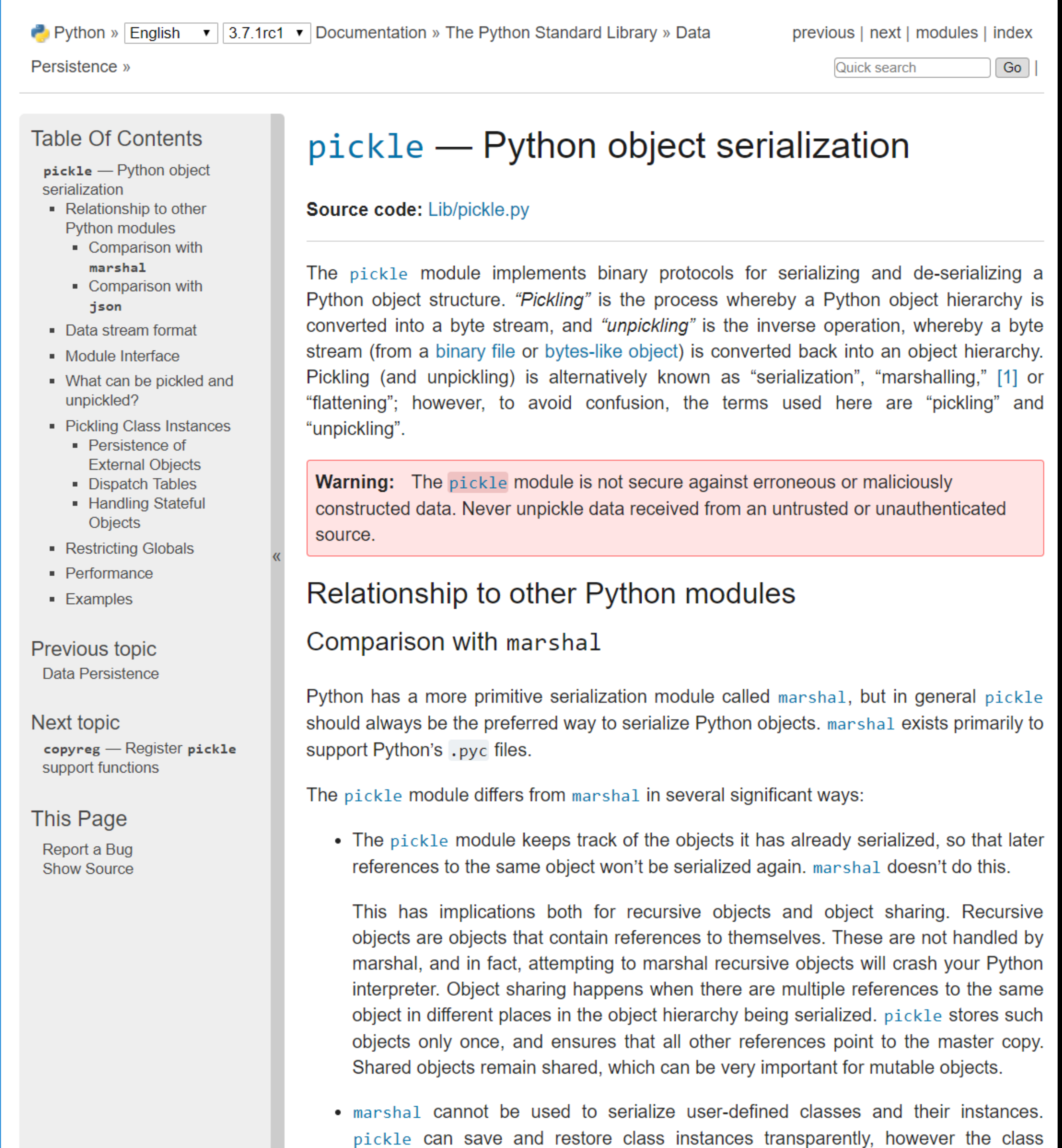
Then later:

```
the_model = torch.load(PATH)
```

However in this case, the serialized data is bound to the specific classes and the exact directory structure used, so it can break in various ways when used in other projects, or after some serious refactors.

Serialization

pytorch.org/
docs/
stable/
notes/
serialization.html



Python » English ▾ 3.7.1rc1 ▾ Documentation » The Python Standard Library » Data Persistence »
previous | next | modules | index
Quick search
Go

Table Of Contents

Previous topic

Data Persistence

Next topic

`copyreg` — Register `pickle` support functions

This Page

Report a Bug
Show Source

`pickle` — Python object serialization

Source code: Lib/pickle.py

The `pickle` module implements binary protocols for serializing and de-serializing a Python object structure. *"Pickling"* is the process whereby a Python object hierarchy is converted into a byte stream, and *"unpickling"* is the inverse operation, whereby a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy. Pickling (and unpickling) is alternatively known as "serialization", "marshalling," [1] or "flattening"; however, to avoid confusion, the terms used here are "pickling" and "unpickling".

Warning: The `pickle` module is not secure against erroneous or maliciously constructed data. Never unpickle data received from an untrusted or unauthenticated source.

Relationship to other Python modules

Comparison with `marshal`

Python has a more primitive serialization module called `marshal`, but in general `pickle` should always be the preferred way to serialize Python objects. `marshal` exists primarily to support Python's `.pyc` files.

The `pickle` module differs from `marshal` in several significant ways:

- The `pickle` module keeps track of the objects it has already serialized, so that later references to the same object won't be serialized again. `marshal` doesn't do this.

  This has implications both for recursive objects and object sharing. Recursive objects are objects that contain references to themselves. These are not handled by marshal, and in fact, attempting to marshal recursive objects will crash your Python interpreter. Object sharing happens when there are multiple references to the same object in different places in the object hierarchy being serialized. `pickle` stores such objects only once, and ensures that all other references point to the master copy. Shared objects remain shared, which can be very important for mutable objects.

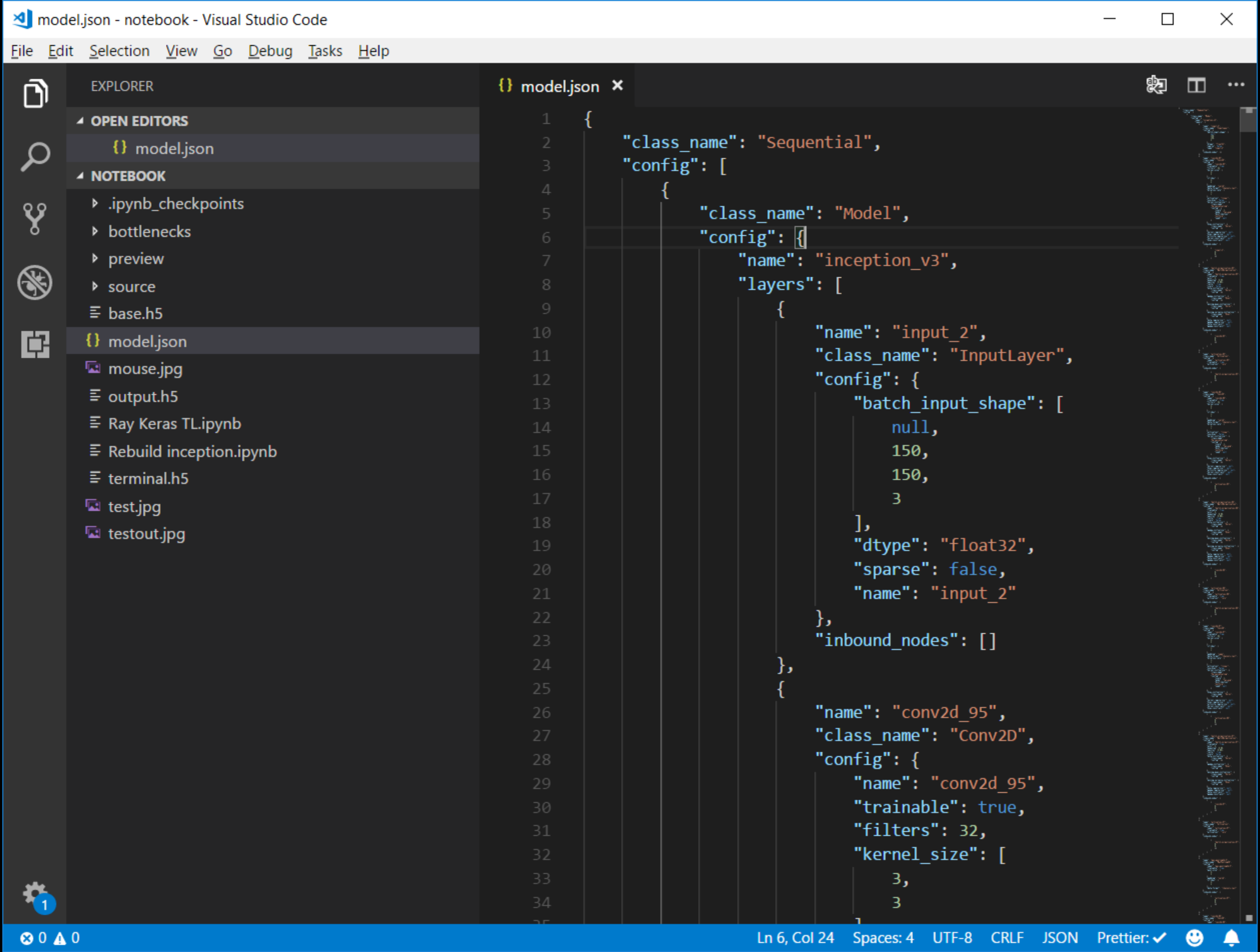
- `marshal` cannot be used to serialize user-defined classes and their instances. `pickle` can save and restore class instances transparently, however the class

Pickle

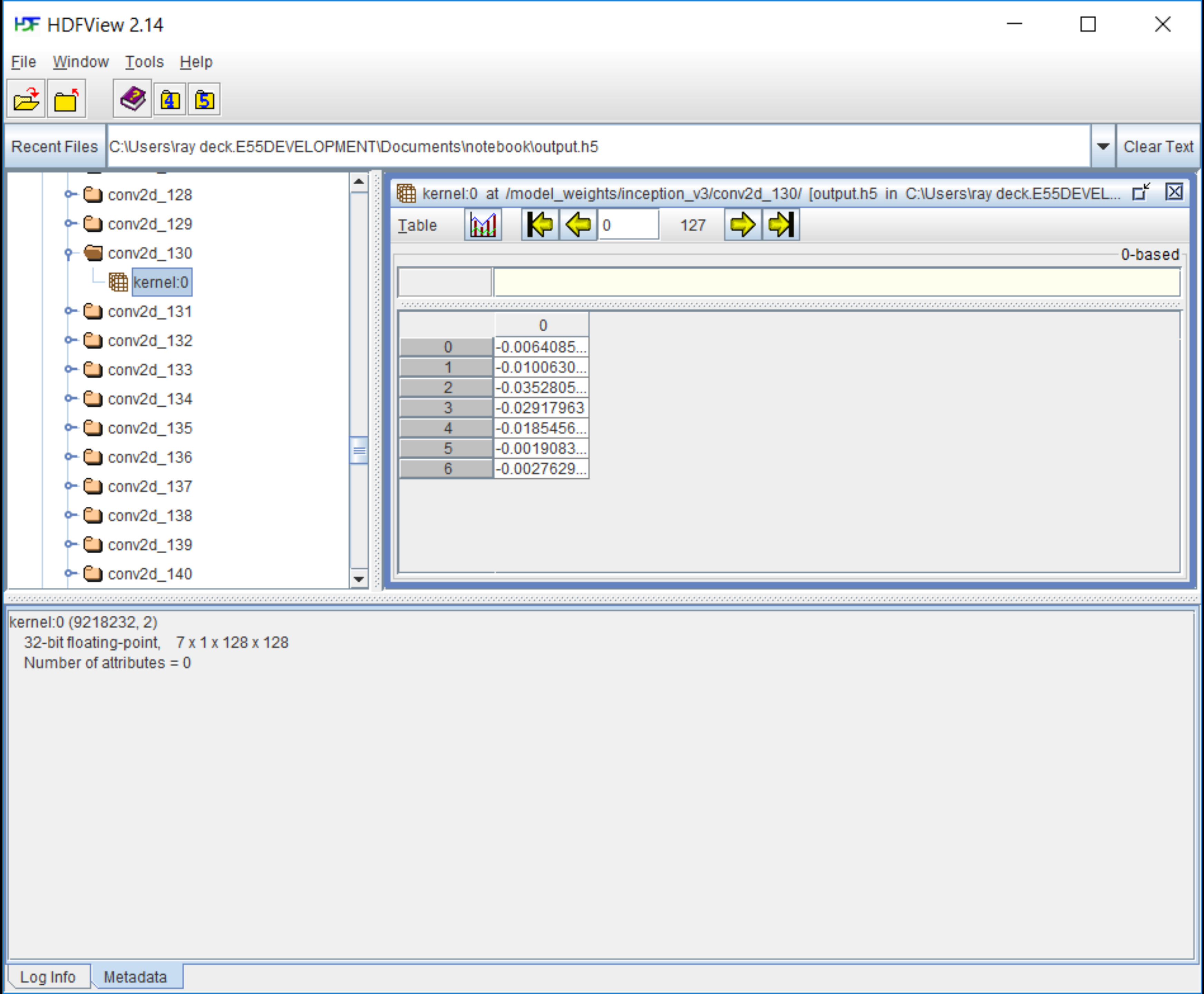
docs.python.org/3/library/pickle.html

K
Keras

JSON + HDF



```
{
    "class_name": "Sequential",
    "config": [
        {
            "class_name": "Model",
            "config": {
                "name": "inception_v3",
                "layers": [
                    {
                        "name": "input_2",
                        "class_name": "InputLayer",
                        "config": {
                            "batch_input_shape": [
                                null,
                                150,
                                150,
                                3
                            ],
                            "dtype": "float32",
                            "sparse": false,
                            "name": "input_2"
                        },
                        "inbound_nodes": []
                    },
                    {
                        "name": "conv2d_95",
                        "class_name": "Conv2D",
                        "config": {
                            "name": "conv2d_95",
                            "trainable": true,
                            "filters": 32,
                            "kernel_size": [
                                3,
                                3
```



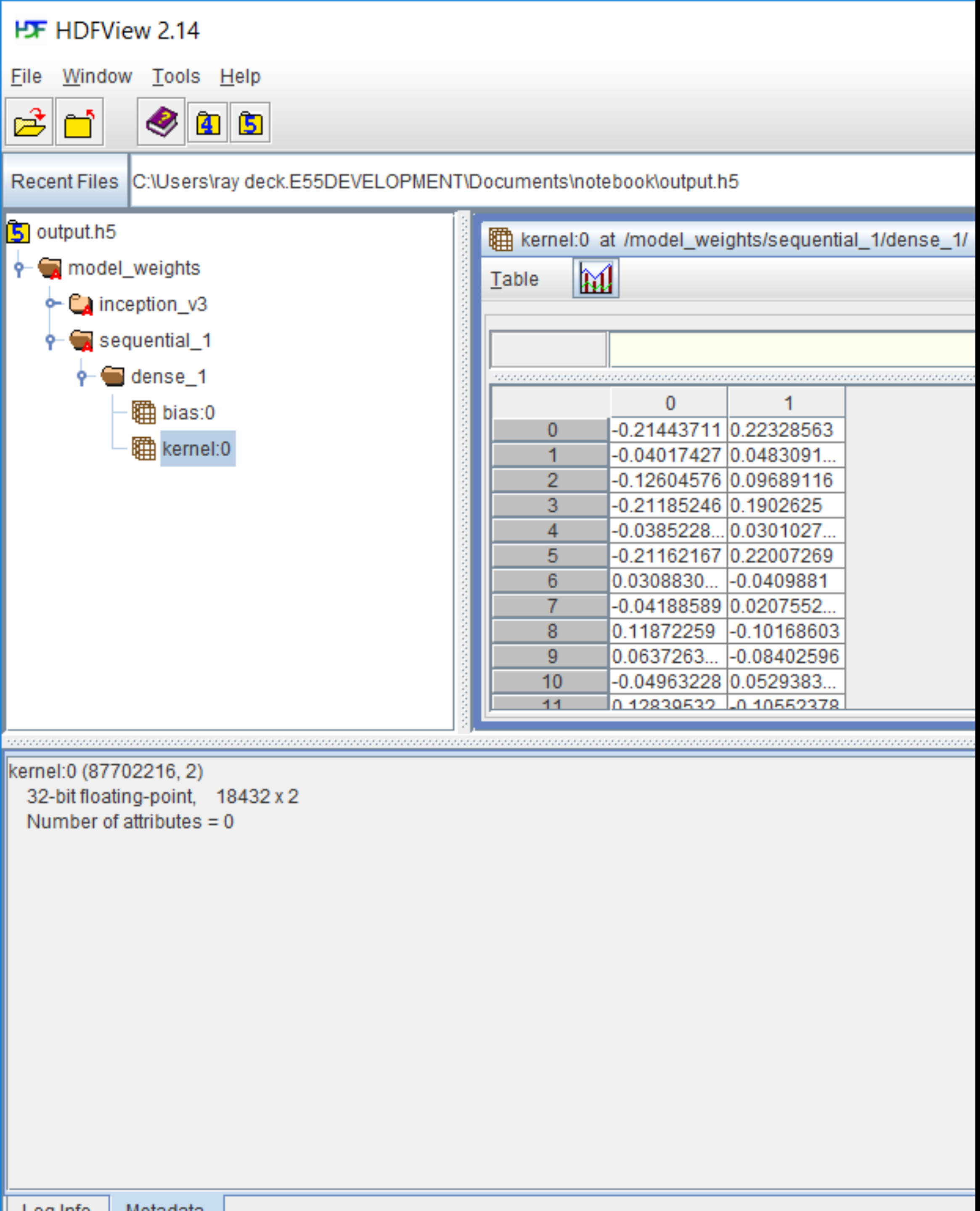
C:\Users\ray deck.E55DEVELOPMENT\Documents\notebook\output.h5

- conv2d_128
- conv2d_129
- conv2d_130
  - kernel:0
- conv2d_131
- conv2d_132
- conv2d_133
- conv2d_134
- conv2d_135
- conv2d_136
- conv2d_137
- conv2d_138
- conv2d_139
- conv2d_140

kernel:0 at /model_weights/inception_v3/conv2d_130/ [output.h5 in C:\Users\ray deck.E55DEVEL...

	0
0	-0.0064085...
1	-0.0100630...
2	-0.0352805...
3	-0.02917963
4	-0.0185456...
5	-0.0019083...
6	-0.0027629...

```
kernel:0 (9218232, 2)
   32-bit floating-point,    7 x 1 x 128 x 128
   Number of attributes = 0
```



C:\Users\ray deck.E55DEVELOPMENT\Documents\notebook\output.h5

output.h5
- model_weights
  - inception_v3
  - sequential_1
    - dense_1
      - bias:0
      - kernel:0

kernel:0 at /model_weights/sequential_1/dense_1/

	0	1
0	-0.21443711	0.22328563
1	-0.04017427	0.0483091...
2	-0.12604576	0.09689116
3	-0.21185246	0.1902625
4	-0.0385228...	0.0301027...
5	-0.21162167	0.22007269
6	0.0308830...	-0.0409881
7	-0.04188589	0.0207552...
8	0.11872259	-0.10168603
9	0.0637263...	-0.08402596
10	-0.04963228	0.0529383...
11	0.12839532	-0.10552378

kernel:0 (87702216, 2)
32-bit floating-point, 18432 x 2
Number of attributes = 0

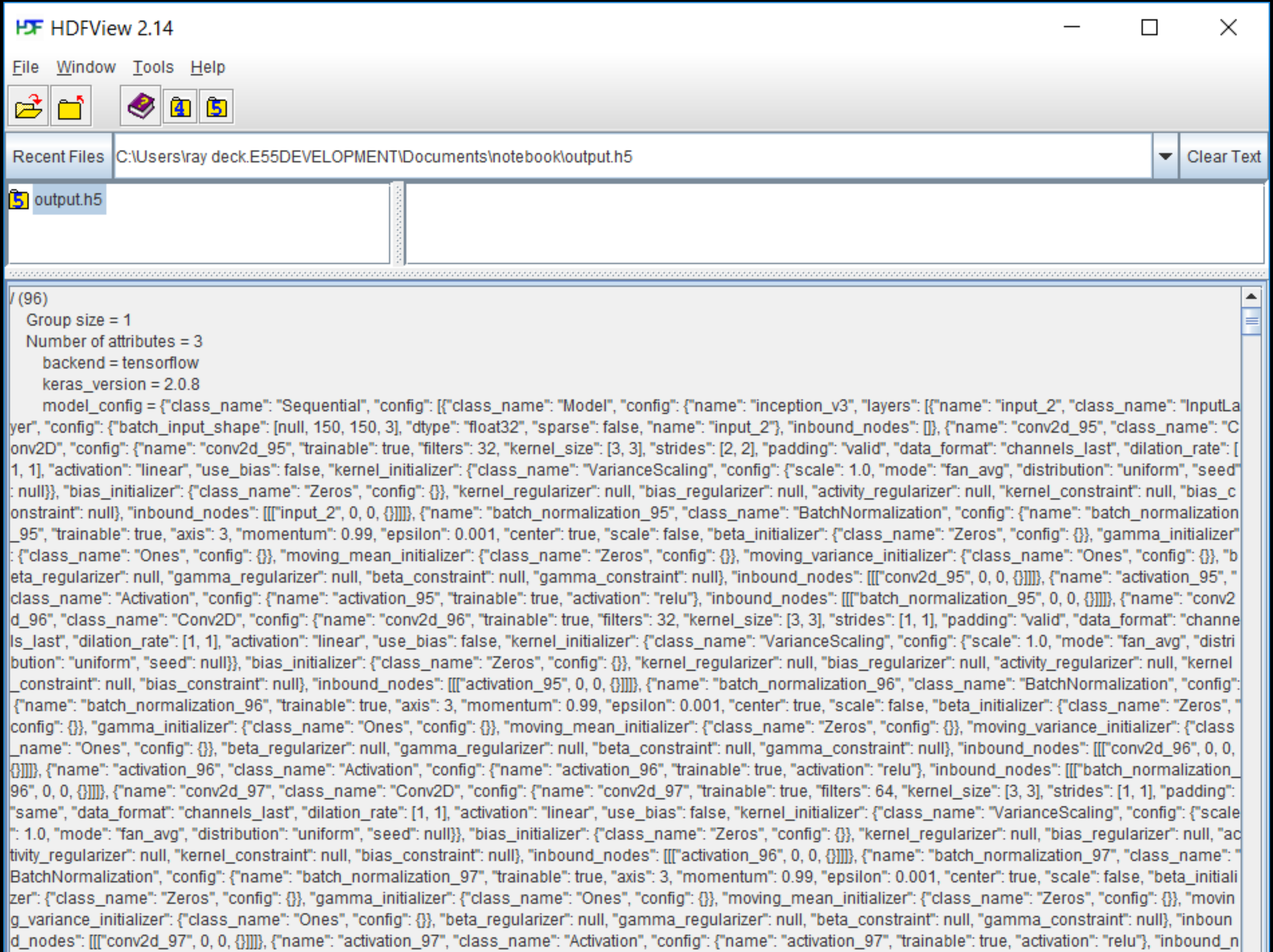
HDFView

portal.hdfgroup.org/display/support/Download+HDFView

Panoply

www.giss.nasa.gov/tools/panoply/

JSON **in** HDF5



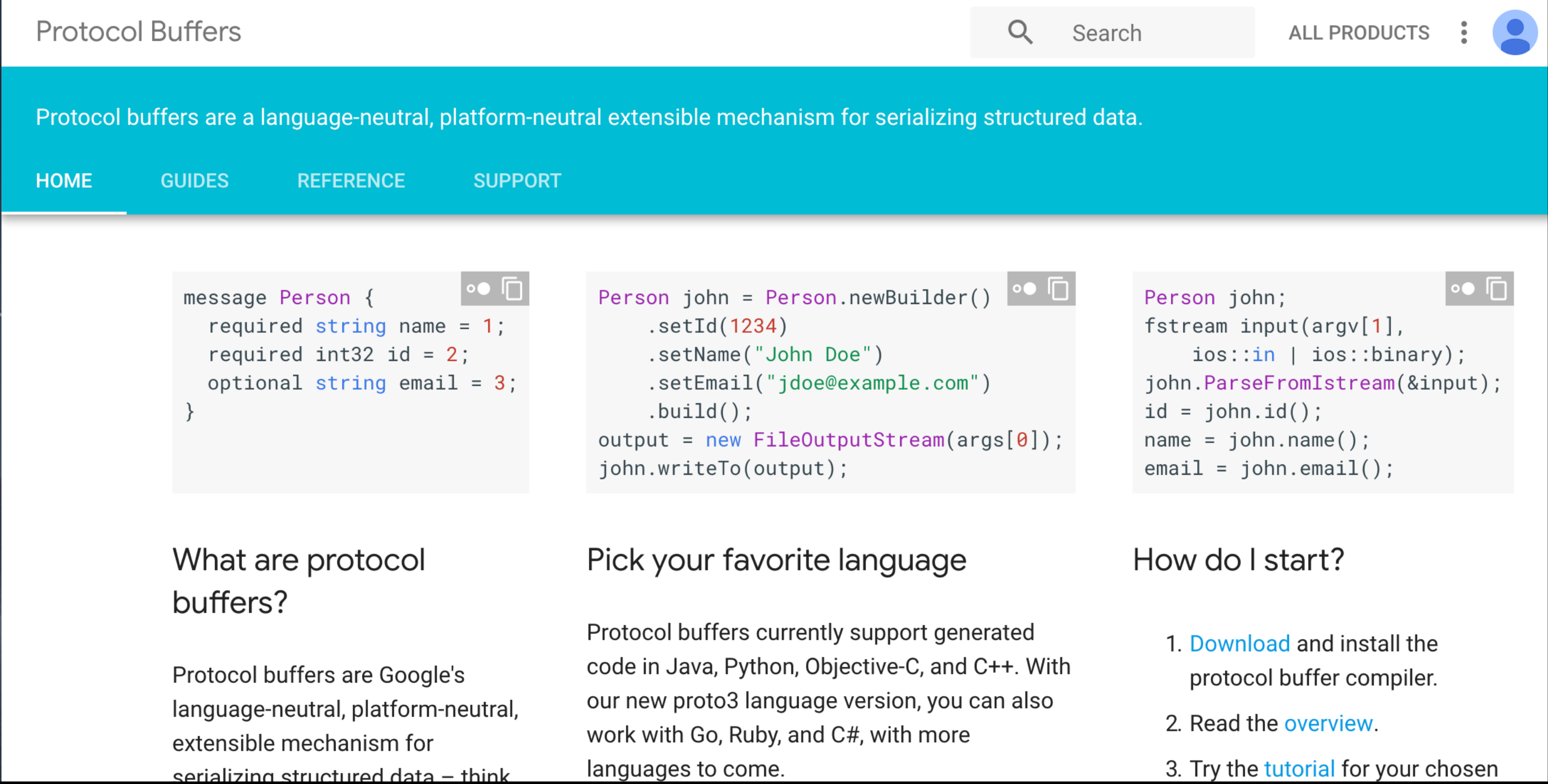
C:\Users\ray deck.E55DEVELOPMENT\Documents\notebook\output.h5

```
/ (96)
   Group size = 1
   Number of attributes = 3
      backend = tensorflow
      keras_version = 2.0.8
```

```
      model_config = {"class_name": "Sequential", "config": [{"class_name": "Model", "config": {"name": "inception_v3", "layers": [{"name": "input_2", "class_name": "InputLa
yer", "config": {"batch_input_shape": [null, 150, 150, 3], "dtype": "float32", "sparse": false, "name": "input_2"}, "inbound_nodes": []}, {"name": "conv2d_95", "class_name": "C
onv2D", "config": {"name": "conv2d_95", "trainable": true, "filters": 32, "kernel_size": [3, 3], "strides": [2, 2], "padding": "valid", "data_format": "channels_last", "dilation_rate": [
1, 1], "activation": "linear", "use_bias": false, "kernel_initializer": {"class_name": "VarianceScaling", "config": {"scale": 1.0, "mode": "fan_avg", "distribution": "uniform", "seed"
: null}}, "bias_initializer": {"class_name": "Zeros", "config": {}}, "kernel_regularizer": null, "bias_regularizer": null, "activity_regularizer": null, "kernel_constraint": null, "bias_c
onstraint": null}, "inbound_nodes": [[["input_2", 0, 0, {}]]]}, {"name": "batch_normalization_95", "class_name": "BatchNormalization", "config": {"name": "batch_normalization
_95", "trainable": true, "axis": 3, "momentum": 0.99, "epsilon": 0.001, "center": true, "scale": false, "beta_initializer": {"class_name": "Zeros", "config": {}}, "gamma_initializer"
: {"class_name": "Ones", "config": {}}, "moving_mean_initializer": {"class_name": "Zeros", "config": {}}, "moving_variance_initializer": {"class_name": "Ones", "config": {}}, "b
eta_regularizer": null, "gamma_regularizer": null, "beta_constraint": null, "gamma_constraint": null}, "inbound_nodes": [[["conv2d_95", 0, 0, {}]]]}, {"name": "activation_95", "
class_name": "Activation", "config": {"name": "activation_95", "trainable": true, "activation": "relu"}, "inbound_nodes": [[["batch_normalization_95", 0, 0, {}]]]}, {"name": "conv2
d_96", "class_name": "Conv2D", "config": {"name": "conv2d_96", "trainable": true, "filters": 32, "kernel_size": [3, 3], "strides": [1, 1], "padding": "valid", "data_format": "channe
ls_last", "dilation_rate": [1, 1], "activation": "linear", "use_bias": false, "kernel_initializer": {"class_name": "VarianceScaling", "config": {"scale": 1.0, "mode": "fan_avg", "distri
bution": "uniform", "seed": null}}, "bias_initializer": {"class_name": "Zeros", "config": {}}, "kernel_regularizer": null, "bias_regularizer": null, "activity_regularizer": null, "kernel
_constraint": null, "bias_constraint": null}, "inbound_nodes": [[["activation_95", 0, 0, {}]]]}, {"name": "batch_normalization_96", "class_name": "BatchNormalization", "config":
 {"name": "batch_normalization_96", "trainable": true, "axis": 3, "momentum": 0.99, "epsilon": 0.001, "center": true, "scale": false, "beta_initializer": {"class_name": "Zeros", "
config": {}}, "gamma_initializer": {"class_name": "Ones", "config": {}}, "moving_mean_initializer": {"class_name": "Zeros", "config": {}}, "moving_variance_initializer": {"class
_name": "Ones", "config": {}}, "beta_regularizer": null, "gamma_regularizer": null, "beta_constraint": null, "gamma_constraint": null}, "inbound_nodes": [[["conv2d_96", 0, 0,
{}]]]}, {"name": "activation_96", "class_name": "Activation", "config": {"name": "activation_96", "trainable": true, "activation": "relu"}, "inbound_nodes": [[["batch_normalization_
96", 0, 0, {}]]]}, {"name": "conv2d_97", "class_name": "Conv2D", "config": {"name": "conv2d_97", "trainable": true, "filters": 64, "kernel_size": [3, 3], "strides": [1, 1], "padding":
"same", "data_format": "channels_last", "dilation_rate": [1, 1], "activation": "linear", "use_bias": false, "kernel_initializer": {"class_name": "VarianceScaling", "config": {"scale
": 1.0, "mode": "fan_avg", "distribution": "uniform", "seed": null}}, "bias_initializer": {"class_name": "Zeros", "config": {}}, "kernel_regularizer": null, "bias_regularizer": null, "ac
tivity_regularizer": null, "kernel_constraint": null, "bias_constraint": null}, "inbound_nodes": [[["activation_96", 0, 0, {}]]]}, {"name": "batch_normalization_97", "class_name": "
BatchNormalization", "config": {"name": "batch_normalization_97", "trainable": true, "axis": 3, "momentum": 0.99, "epsilon": 0.001, "center": true, "scale": false, "beta_initiali
zer": {"class_name": "Zeros", "config": {}}, "gamma_initializer": {"class_name": "Ones", "config": {}}, "moving_mean_initializer": {"class_name": "Zeros", "config": {}}, "movin
g_variance_initializer": {"class_name": "Ones", "config": {}}, "beta_regularizer": null, "gamma_regularizer": null, "beta_constraint": null, "gamma_constraint": null}, "inboun
d_nodes": [[["conv2d_97", 0, 0, {}]]]}, {"name": "activation_97", "class_name": "Activation", "config": {"name": "activation_97", "trainable": true, "activation": "relu"}, "inbound_n
```

protobuf

Protocol Buffers



Protocol buffers are a language-neutral, platform-neutral extensible mechanism for serializing structured data.

```
message Person {
  required string name = 1;
  required int32 id = 2;
  optional string email = 3;
}
```

```
Person john = Person.newBuilder()
    .setId(1234)
    .setName("John Doe")
    .setEmail("jdoe@example.com")
    .build();
output = new FileOutputStream(args[0]);
john.writeTo(output);
```

```
Person john;
fstream input(argv[1],
    ios::in | ios::binary);
john.ParseFromIstream(&input);
id = john.id();
name = john.name();
email = john.email();
```

What are protocol buffers?

Protocol buffers are Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data – think

Pick your favorite language

Protocol buffers currently support generated code in Java, Python, Objective-C, and C++. With our new proto3 language version, you can also work with Go, Ruby, and C#, with more languages to come.

How do I start?

1. Download and install the protocol buffer compiler.
2. Read the overview.
3. Try the tutorial for your chosen

developers.google.com/protocol-buffers

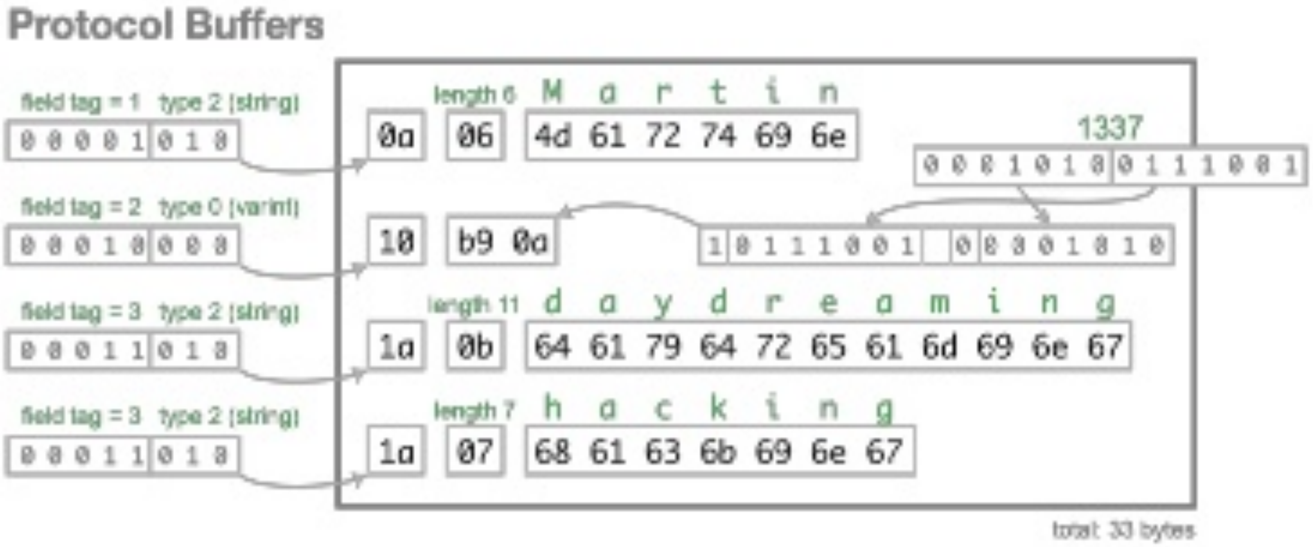
Message

Person.json

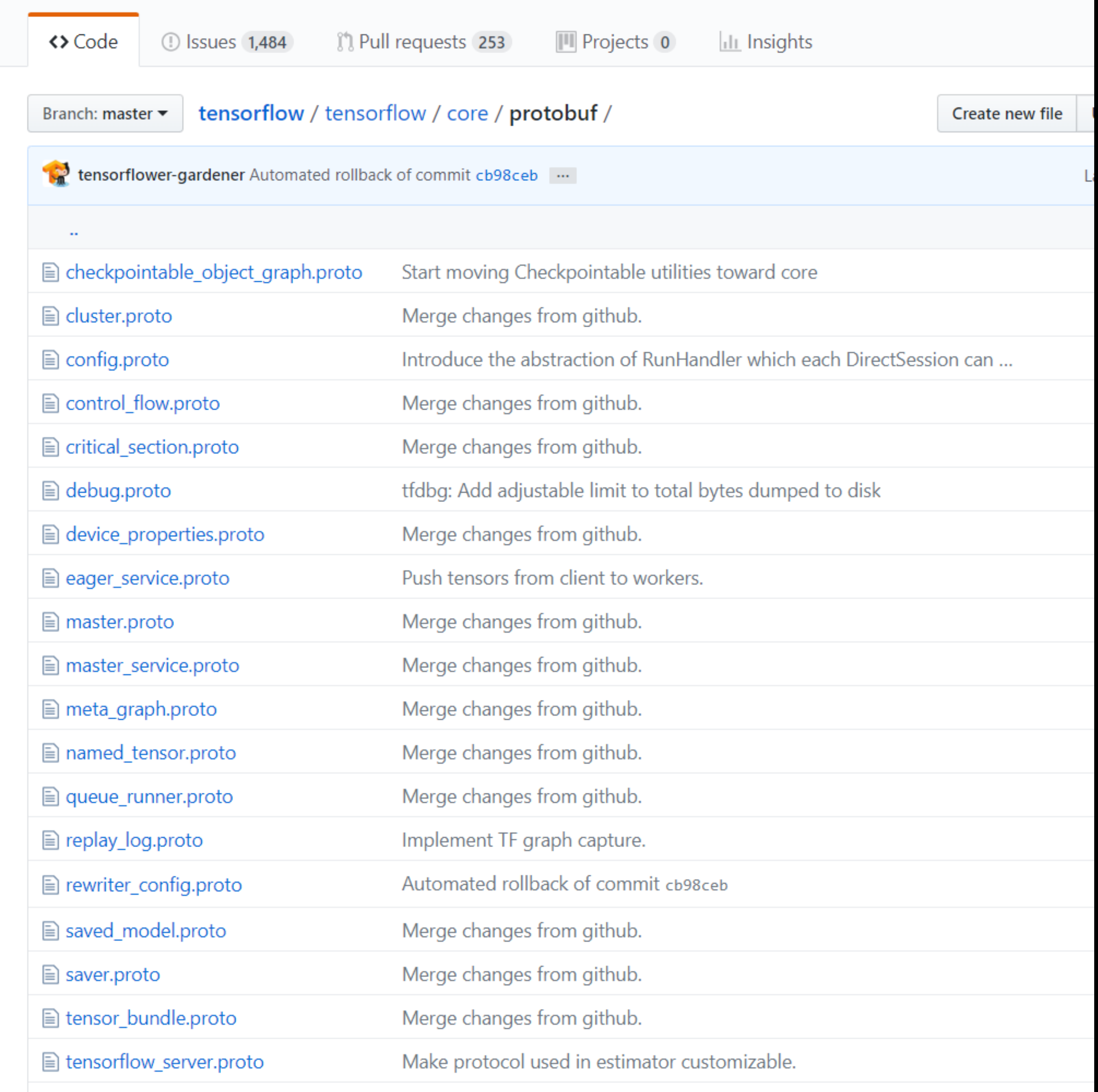
```
{
    "userName": "Martin",
    "favouriteNumber": 1337,
    "interests": ["daydreaming", "hacking"]
}
```

Person.proto

```
message Person {
    required string user_name        = 1;
    optional int64  favourite_number = 2;
    repeated string interests        = 3;
}
```



https://www.slideshare.net/SergeyPodolsky/google-protocol-buffers-56085699

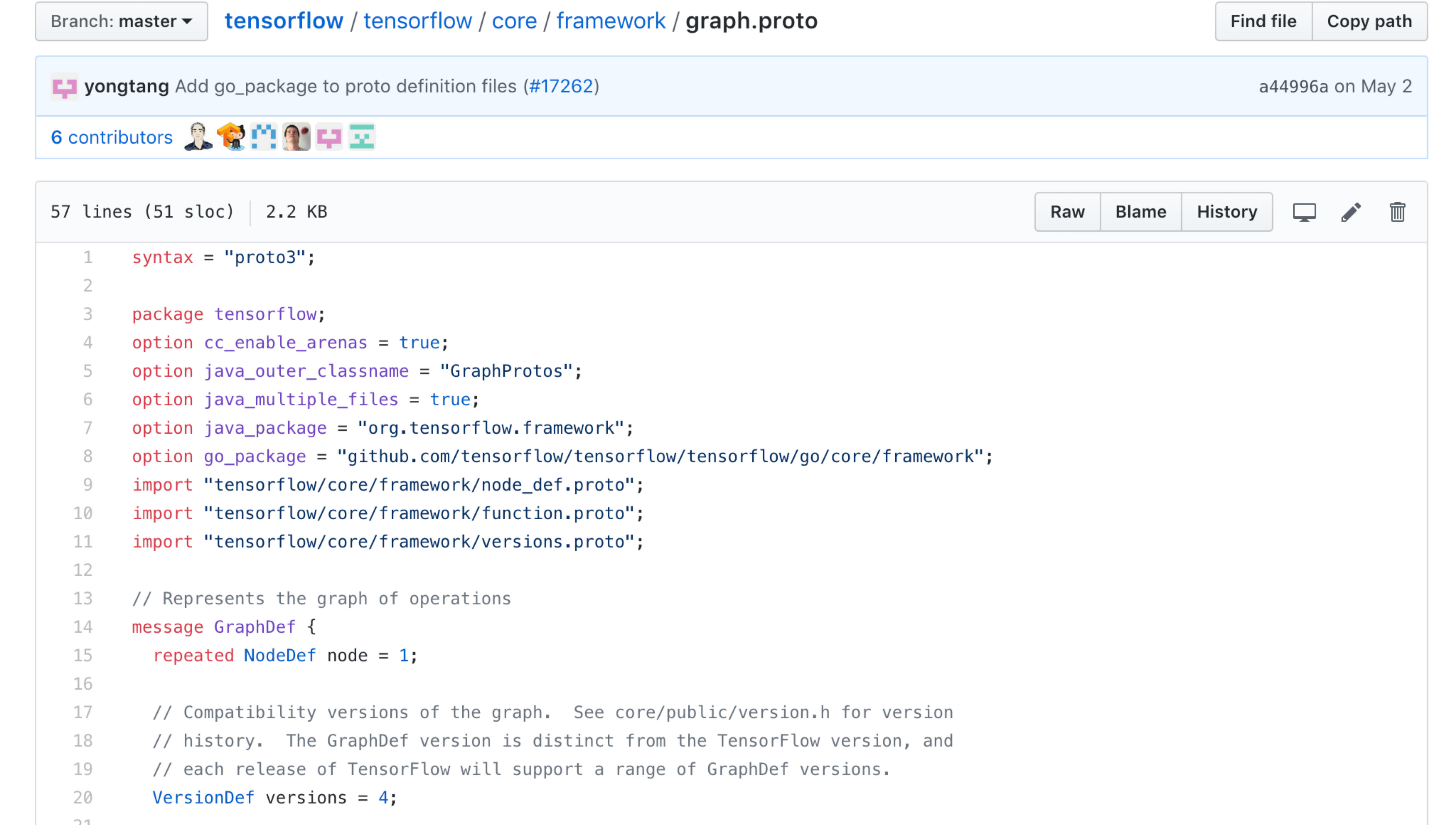


tensorflow / tensorflow / core / protobuf /

tensorflower-gardener Automated rollback of commit cb98ceb ...

..	
checkpointable_object_graph.proto	Start moving Checkpointable utilities toward core
cluster.proto	Merge changes from github.
config.proto	Introduce the abstraction of RunHandler which each DirectSession can ...
control_flow.proto	Merge changes from github.
critical_section.proto	Merge changes from github.
debug.proto	tfdbg: Add adjustable limit to total bytes dumped to disk
device_properties.proto	Merge changes from github.
eager_service.proto	Push tensors from client to workers.
master.proto	Merge changes from github.
master_service.proto	Merge changes from github.
meta_graph.proto	Merge changes from github.
named_tensor.proto	Merge changes from github.
queue_runner.proto	Merge changes from github.
replay_log.proto	Implement TF graph capture.
rewriter_config.proto	Automated rollback of commit cb98ceb
saved_model.proto	Merge changes from github.
saver.proto	Merge changes from github.
tensor_bundle.proto	Merge changes from github.
tensorflow_server.proto	Make protocol used in estimator customizable.

github.com/
tensorflow/
tensorflow/
tree/
master/
tensorflow/
core/
protobuf



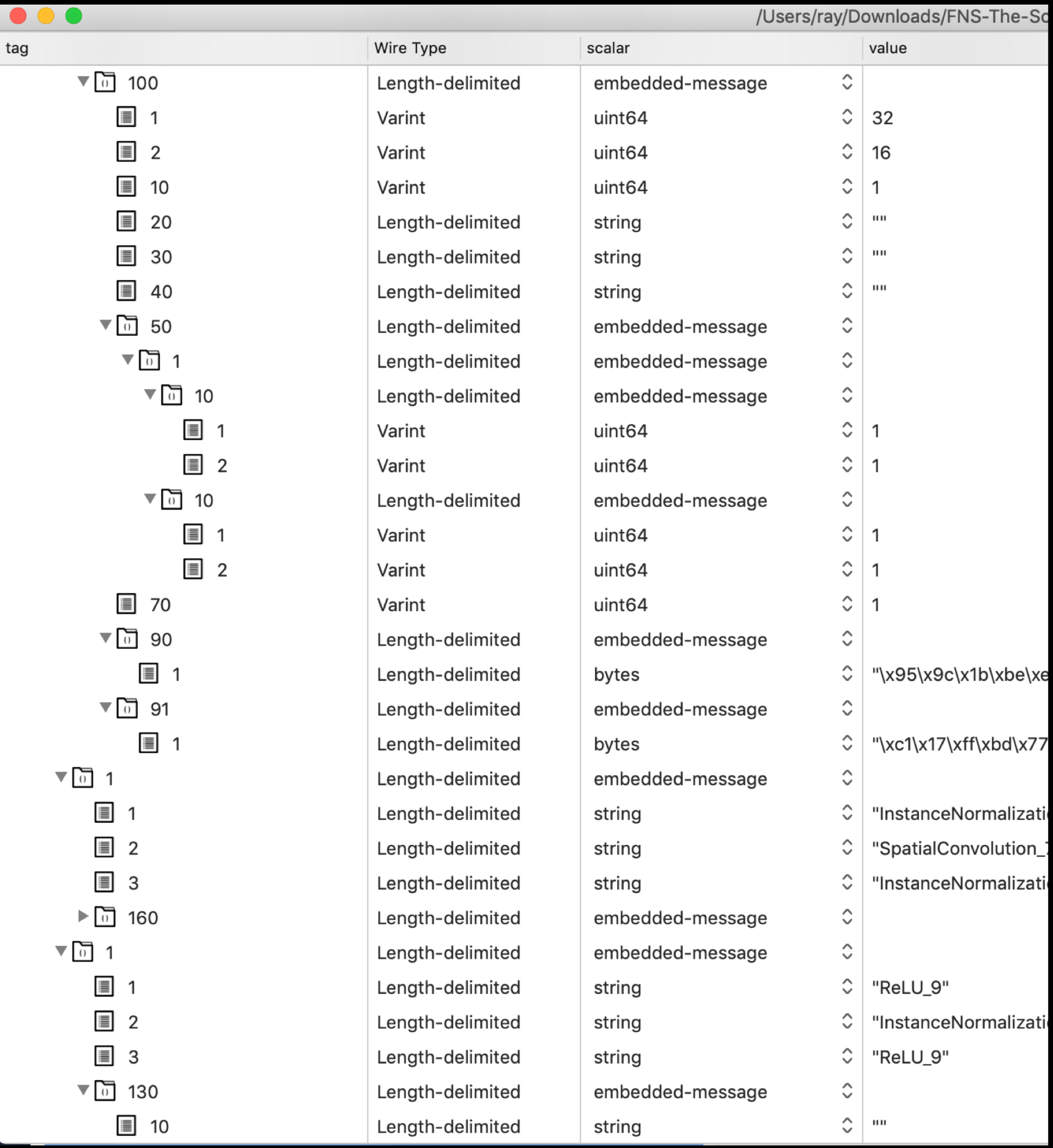
tensorflow / tensorflow / core / framework / **graph.proto**

yongtang Add go_package to proto definition files (#17262) a44996a on May 2

6 contributors

57 lines (51 sloc) | 2.2 KB

```
syntax = "proto3";

package tensorflow;
option cc_enable_arenas = true;
option java_outer_classname = "GraphProtos";
option java_multiple_files = true;
option java_package = "org.tensorflow.framework";
option go_package = "github.com/tensorflow/tensorflow/tensorflow/go/core/framework";
import "tensorflow/core/framework/node_def.proto";
import "tensorflow/core/framework/function.proto";
import "tensorflow/core/framework/versions.proto";

// Represents the graph of operations
message GraphDef {
  repeated NodeDef node = 1;

  // Compatibility versions of the graph.  See core/public/version.h for version
  // history.  The GraphDef version is distinct from the TensorFlow version, and
  // each release of TensorFlow will support a range of GraphDef versions.
  VersionDef versions = 4;
```



/Users/ray/Downloads/FNS-The-Sc

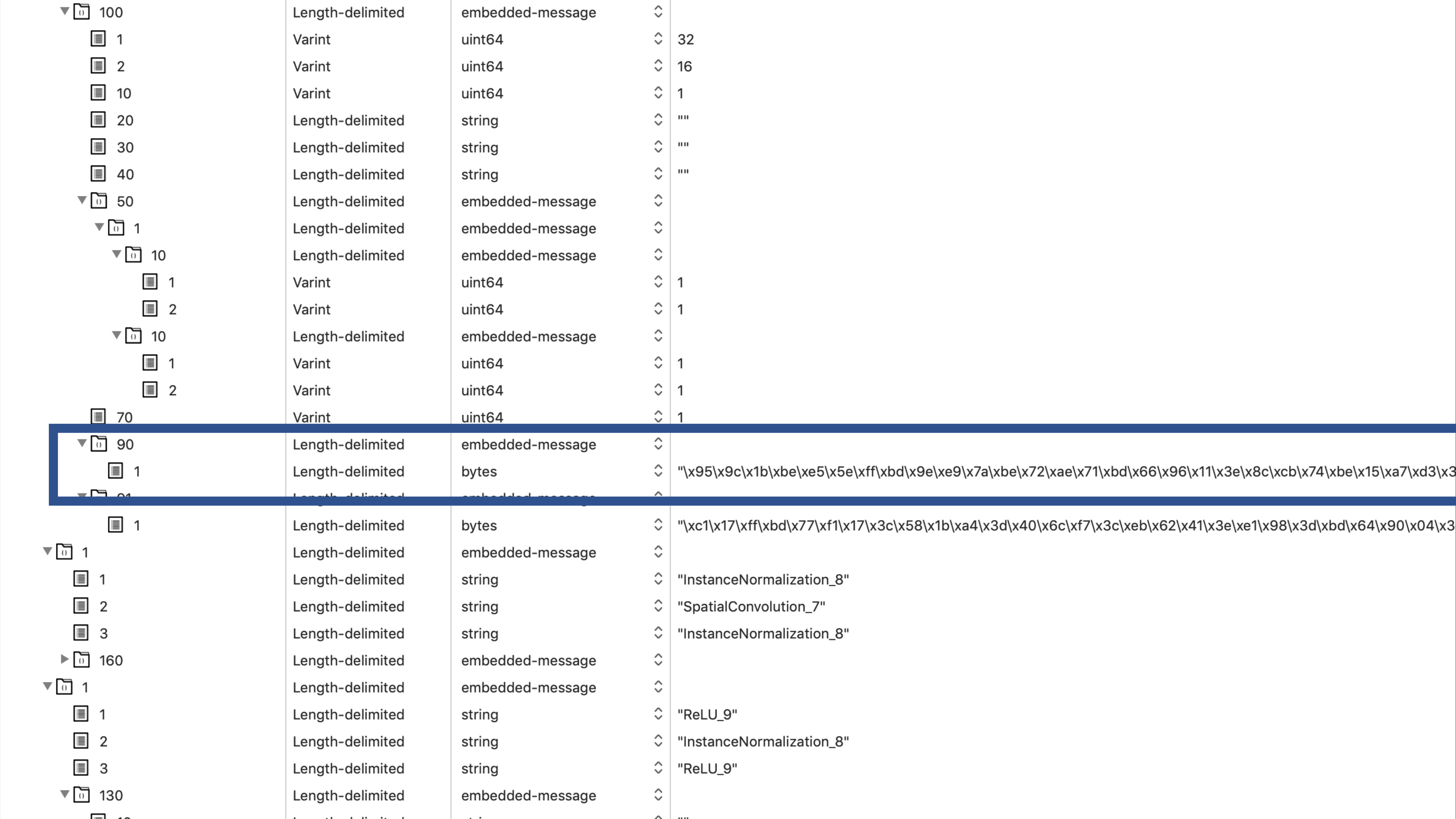
```
tag                  Wire Type          scalar               value
    ▼ 100            Length-delimited   embedded-message
        1            Varint             uint64               32
        2            Varint             uint64               16
        10           Varint             uint64               1
        20           Length-delimited   string               ""
        30           Length-delimited   string               ""
        40           Length-delimited   string               ""
      ▼ 50           Length-delimited   embedded-message
        ▼ 1          Length-delimited   embedded-message
          ▼ 10       Length-delimited   embedded-message
              1      Varint             uint64               1
              2      Varint             uint64               1
          ▼ 10       Length-delimited   embedded-message
              1      Varint             uint64               1
              2      Varint             uint64               1
        70           Varint             uint64               1
      ▼ 90           Length-delimited   embedded-message
          1          Length-delimited   bytes                "\x95\x9c\x1b\xbe\xe
      ▼ 91           Length-delimited   embedded-message
          1          Length-delimited   bytes                "\xc1\x17\xff\xbd\x77
  ▼ 1                Length-delimited   embedded-message
      1              Length-delimited   string               "InstanceNormalizati
      2              Length-delimited   string               "SpatialConvolution_
      3              Length-delimited   string               "InstanceNormalizati
    ▶ 160            Length-delimited   embedded-message
  ▼ 1                Length-delimited   embedded-message
      1              Length-delimited   string               "ReLU_9"
      2              Length-delimited   string               "InstanceNormalizati
      3              Length-delimited   string               "ReLU_9"
    ▼ 130            Length-delimited   embedded-message
        10           Length-delimited   string               ""
```

Protobuf Viewer

MacOS App Store

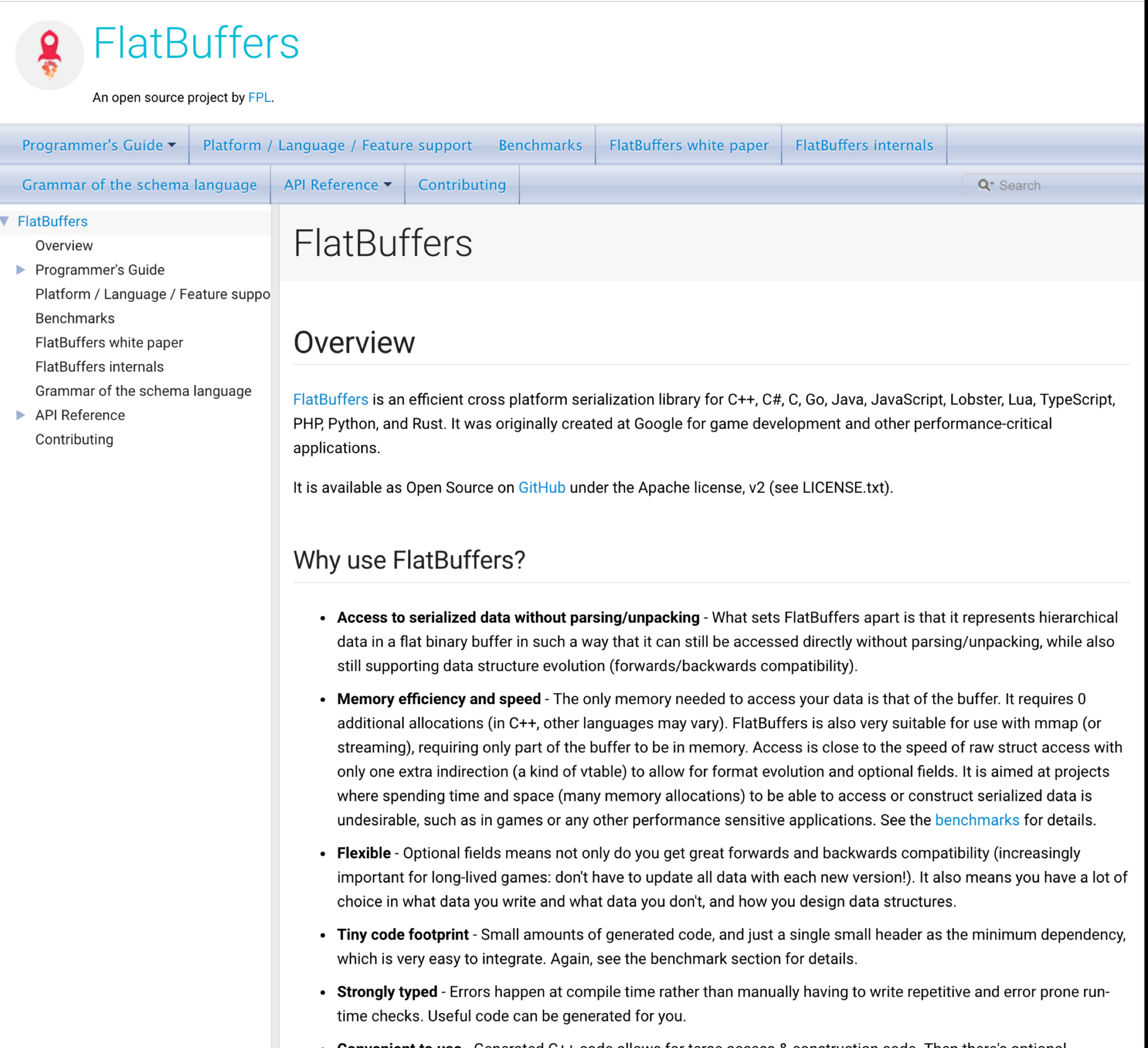
Protobuf Editor

sourceforge.net/projects/protobufeditor/



```
▼ 100                    Length-delimited   embedded-message
    1                    Varint             uint64              32
    2                    Varint             uint64              16
    10                   Varint             uint64              1
    20                   Length-delimited   string              ""
    30                   Length-delimited   string              ""
    40                   Length-delimited   string              ""
  ▼ 50                   Length-delimited   embedded-message
    ▼ 1                  Length-delimited   embedded-message
      ▼ 10               Length-delimited   embedded-message
          1              Varint             uint64              1
          2              Varint             uint64              1
      ▼ 10               Length-delimited   embedded-message
          1              Varint             uint64              1
          2              Varint             uint64              1
    70                   Varint             uint64              1
  ▼ 90                   Length-delimited   embedded-message
      1                  Length-delimited   bytes               "\x95\x9c\x1b\xbe\xe5\x5e\xff\xbd\x9e\xe9\x7a\xbe\x72\xae\x71\xbd\x66\x96\x11\x3e\x8c\xcb\x74\xbe\x15\xa7\xd3\x3
  ▼ 91                   Length-delimited   embedded-message
      1                  Length-delimited   bytes               "\xc1\x17\xff\xbd\x77\xf1\x17\x3c\x58\x1b\xa4\x3d\x40\x6c\xf7\x3c\xeb\x62\x41\x3e\xe1\x98\x3d\xbd\x64\x90\x04\x3
▼ 1                      Length-delimited   embedded-message
    1                    Length-delimited   string              "InstanceNormalization_8"
    2                    Length-delimited   string              "SpatialConvolution_7"
    3                    Length-delimited   string              "InstanceNormalization_8"
  ▶ 160                  Length-delimited   embedded-message
▼ 1                      Length-delimited   embedded-message
    1                    Length-delimited   string              "ReLU_9"
    2                    Length-delimited   string              "InstanceNormalization_8"
    3                    Length-delimited   string              "ReLU_9"
  ▼ 130                  Length-delimited   embedded-message
      10                 Length-delimited   string              ""
```

TensorFlow Lite



FlatBuffers

Overview

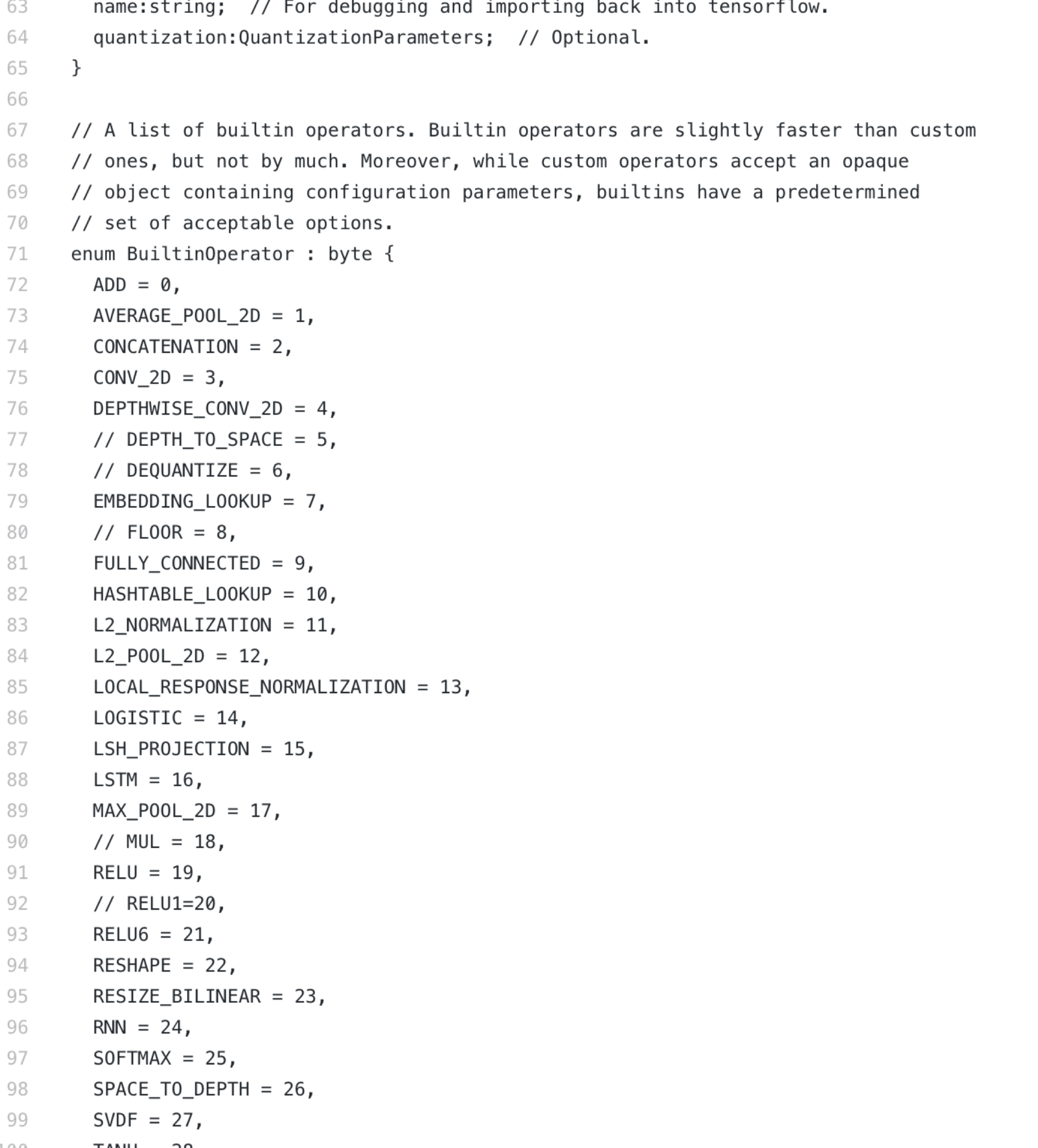
FlatBuffers is an efficient cross platform serialization library for C++, C#, C, Go, Java, JavaScript, Lobster, Lua, TypeScript, PHP, Python, and Rust. It was originally created at Google for game development and other performance-critical applications.

It is available as Open Source on GitHub under the Apache license, v2 (see LICENSE.txt).

Why use FlatBuffers?

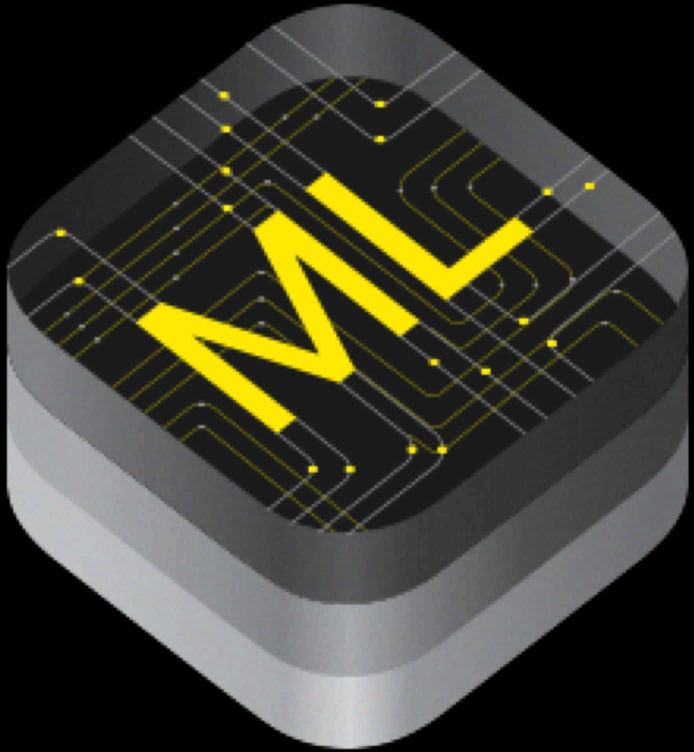
- **Access to serialized data without parsing/unpacking** - What sets FlatBuffers apart is that it represents hierarchical data in a flat binary buffer in such a way that it can still be accessed directly without parsing/unpacking, while also still supporting data structure evolution (forwards/backwards compatibility).
- **Memory efficiency and speed** - The only memory needed to access your data is that of the buffer. It requires 0 additional allocations (in C++, other languages may vary). FlatBuffers is also very suitable for use with mmap (or streaming), requiring only part of the buffer to be in memory. Access is close to the speed of raw struct access with only one extra indirection (a kind of vtable) to allow for format evolution and optional fields. It is aimed at projects where spending time and space (many memory allocations) to be able to access or construct serialized data is undesirable, such as in games or any other performance sensitive applications. See the benchmarks for details.
- **Flexible** - Optional fields means not only do you get great forwards and backwards compatibility (increasingly important for long-lived games: don't have to update all data with each new version!). It also means you have a lot of choice in what data you write and what data you don't, and how you design data structures.
- **Tiny code footprint** - Small amounts of generated code, and just a single small header as the minimum dependency, which is very easy to integrate. Again, see the benchmark section for details.
- **Strongly typed** - Errors happen at compile time rather than manually having to write repetitive and error prone run-time checks. Useful code can be generated for you.
- **Convenient to use** - Generated C++ code allows for terse access & construction code. Then there's optional

google.github.io/
flatbuffers/

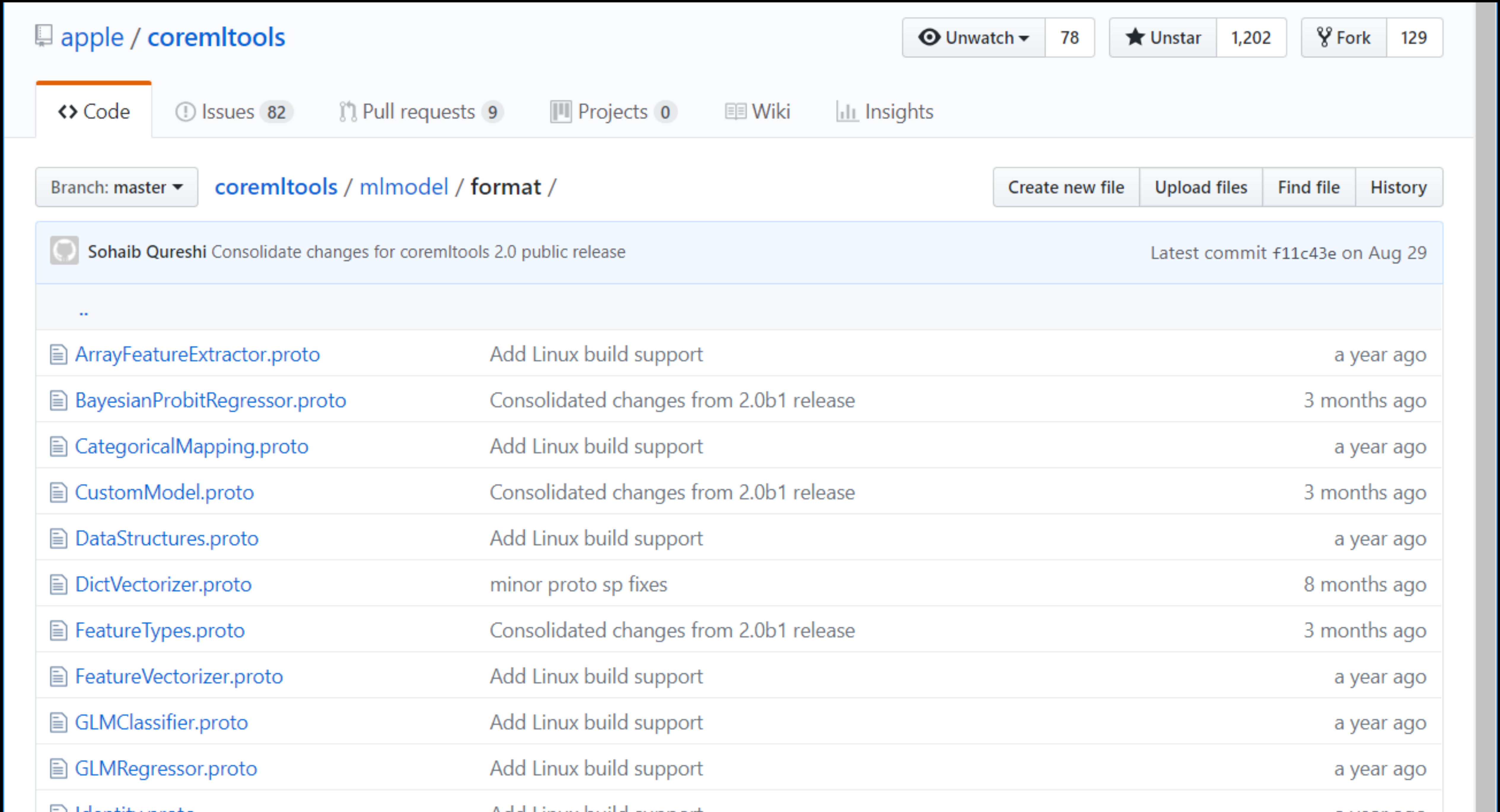


```
  name:string;  // For debugging and importing back into tensorflow.
  quantization:QuantizationParameters;  // Optional.
}

// A list of builtin operators. Builtin operators are slightly faster than custom
// ones, but not by much. Moreover, while custom operators accept an opaque
// object containing configuration parameters, builtins have a predetermined
// set of acceptable options.
enum BuiltinOperator : byte {
  ADD = 0,
  AVERAGE_POOL_2D = 1,
  CONCATENATION = 2,
  CONV_2D = 3,
  DEPTHWISE_CONV_2D = 4,
  // DEPTH_TO_SPACE = 5,
  // DEQUANTIZE = 6,
  EMBEDDING_LOOKUP = 7,
  // FLOOR = 8,
  FULLY_CONNECTED = 9,
  HASHTABLE_LOOKUP = 10,
  L2_NORMALIZATION = 11,
  L2_POOL_2D = 12,
  LOCAL_RESPONSE_NORMALIZATION = 13,
  LOGISTIC = 14,
  LSH_PROJECTION = 15,
  LSTM = 16,
  MAX_POOL_2D = 17,
  // MUL = 18,
  RELU = 19,
  // RELU1=20,
  RELU6 = 21,
  RESHAPE = 22,
  RESIZE_BILINEAR = 23,
  RNN = 24,
  SOFTMAX = 25,
  SPACE_TO_DEPTH = 26,
  SVDF = 27,
  TANH = 28
```

github.com/
tensorflow/
tensorflow/
blob/
master/
tensorflow/
contrib/
lite/
schema/
schema_v3.fbs


ML

github.com/
apple/
coremltools/
tree/
master/
mlmodel/
format

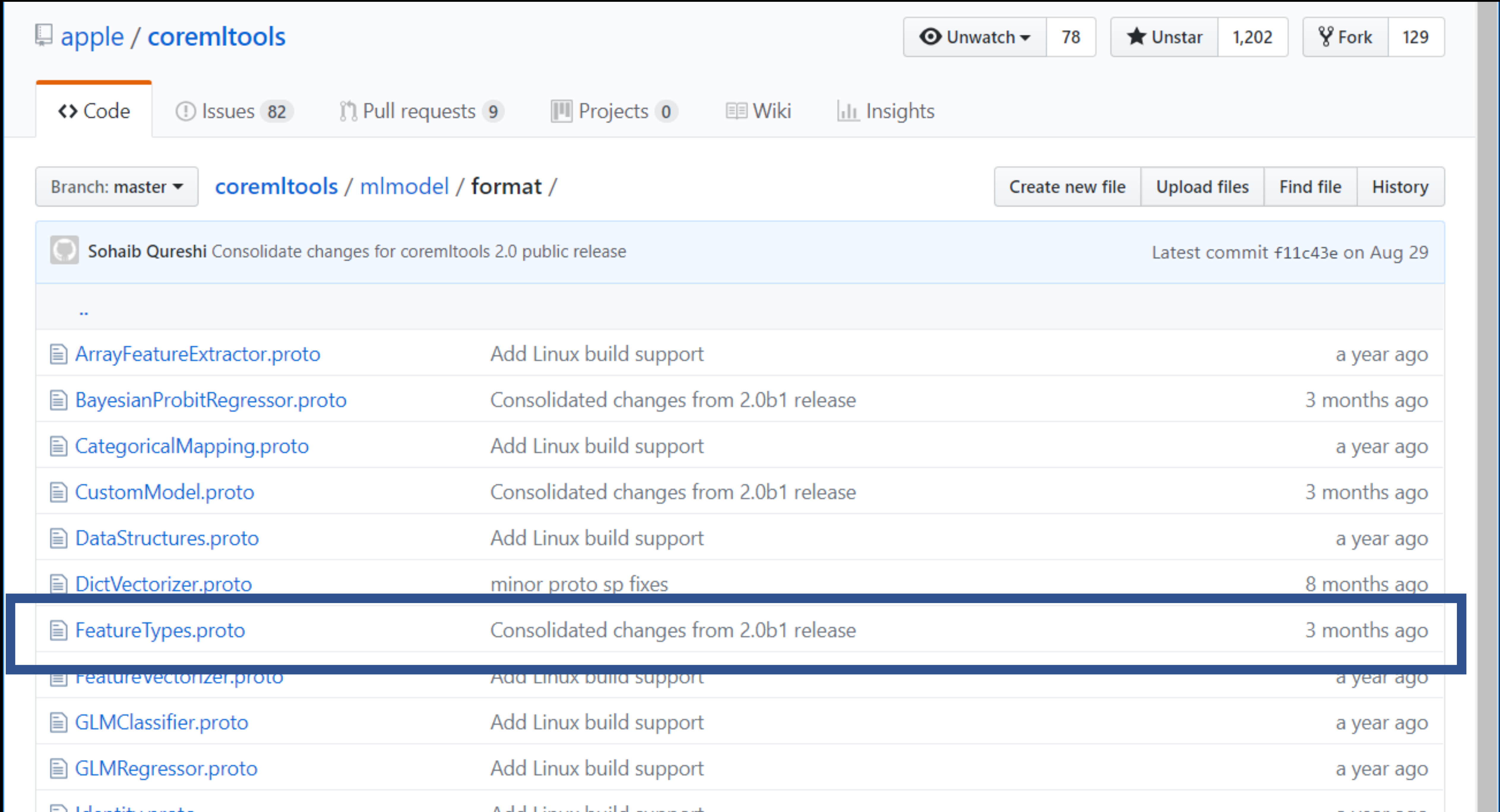


apple / coremltools

coremltools / mlmodel / format /

Sohaib Qureshi Consolidate changes for coremltools 2.0 public release   Latest commit f11c43e on Aug 29

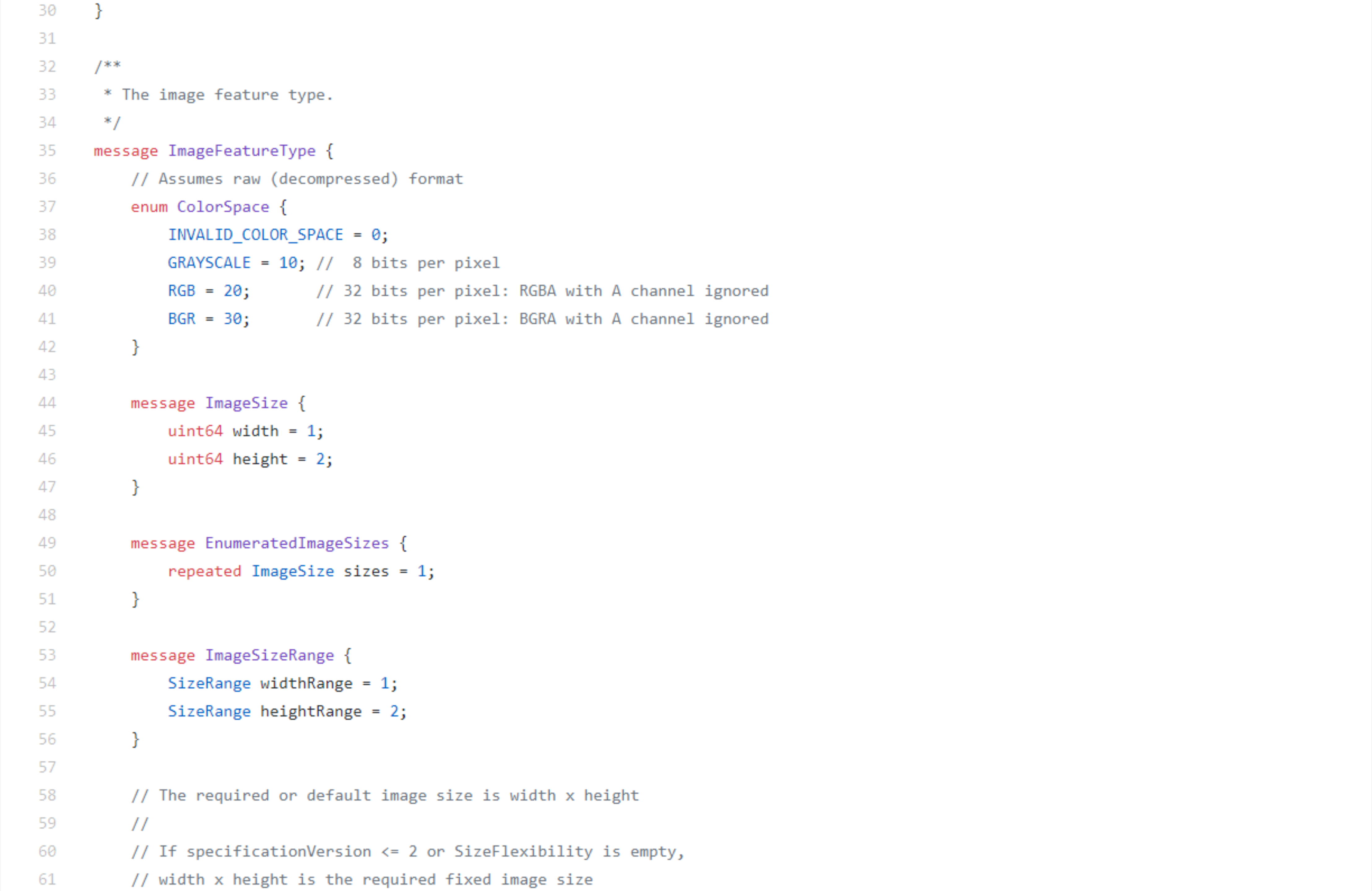
..		
ArrayFeatureExtractor.proto	Add Linux build support	a year ago
BayesianProbitRegressor.proto	Consolidated changes from 2.0b1 release	3 months ago
CategoricalMapping.proto	Add Linux build support	a year ago
CustomModel.proto	Consolidated changes from 2.0b1 release	3 months ago
DataStructures.proto	Add Linux build support	a year ago
DictVectorizer.proto	minor proto sp fixes	8 months ago
FeatureTypes.proto	Consolidated changes from 2.0b1 release	3 months ago
FeatureVectorizer.proto	Add Linux build support	a year ago
GLMClassifier.proto	Add Linux build support	a year ago
GLMRegressor.proto	Add Linux build support	a year ago
Identity.proto	Add Linux build support	a year ago



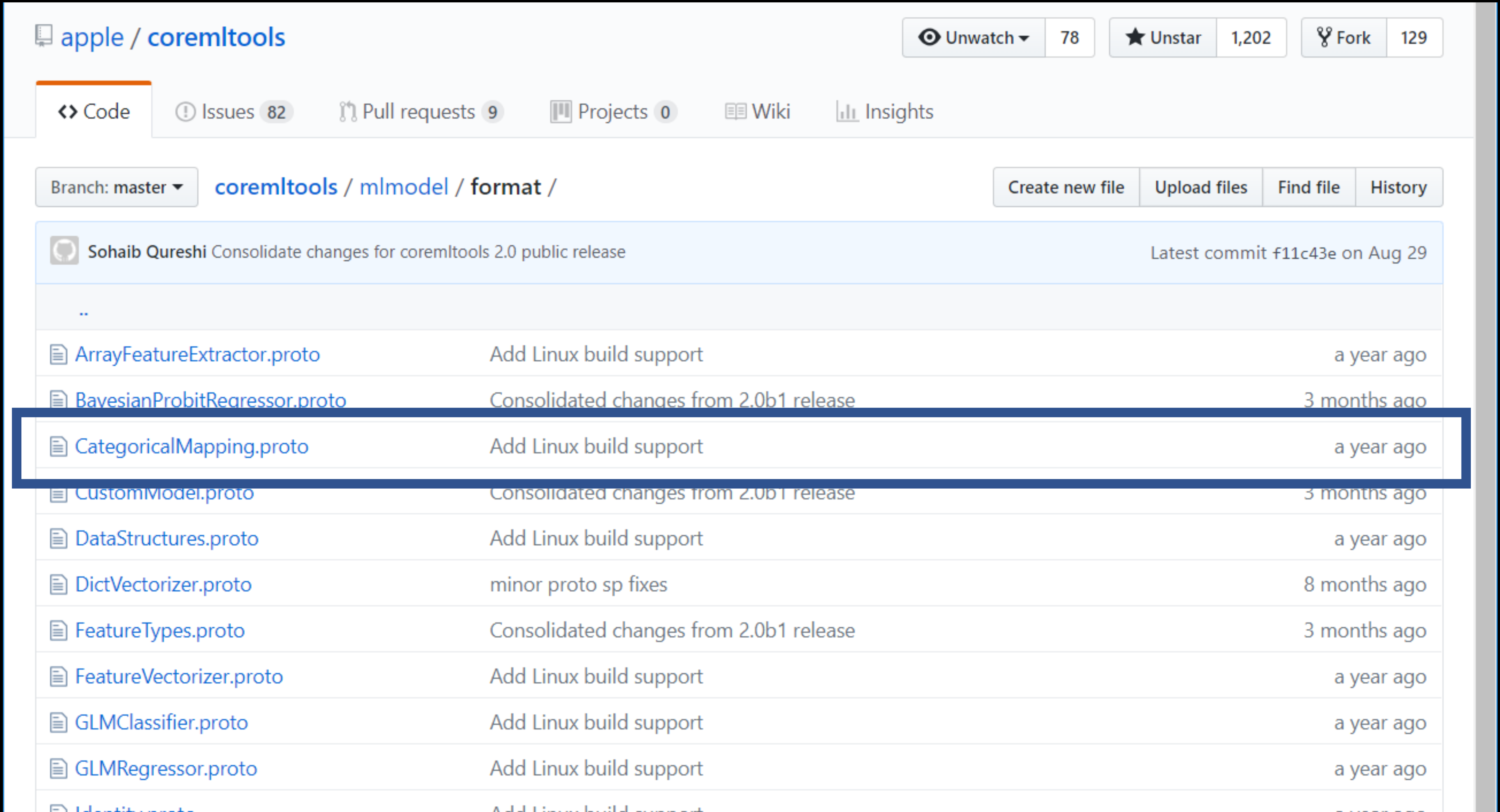
coremltools / mlmodel / format /

Sohaib Qureshi Consolidate changes for coremltools 2.0 public release — Latest commit f11c43e on Aug 29

..		
ArrayFeatureExtractor.proto	Add Linux build support	a year ago
BayesianProbitRegressor.proto	Consolidated changes from 2.0b1 release	3 months ago
CategoricalMapping.proto	Add Linux build support	a year ago
CustomModel.proto	Consolidated changes from 2.0b1 release	3 months ago
DataStructures.proto	Add Linux build support	a year ago
DictVectorizer.proto	minor proto sp fixes	8 months ago
FeatureTypes.proto	Consolidated changes from 2.0b1 release	3 months ago
FeatureVectorizer.proto	Add Linux build support	a year ago
GLMClassifier.proto	Add Linux build support	a year ago
GLMRegressor.proto	Add Linux build support	a year ago
Identity.proto	Add Linux build support	a year ago



```
}

/**
 * The image feature type.
 */
message ImageFeatureType {
    // Assumes raw (decompressed) format
    enum ColorSpace {
        INVALID_COLOR_SPACE = 0;
        GRAYSCALE = 10; //  8 bits per pixel
        RGB = 20;       // 32 bits per pixel: RGBA with A channel ignored
        BGR = 30;       // 32 bits per pixel: BGRA with A channel ignored
    }

    message ImageSize {
        uint64 width = 1;
        uint64 height = 2;
    }

    message EnumeratedImageSizes {
        repeated ImageSize sizes = 1;
    }

    message ImageSizeRange {
        SizeRange widthRange = 1;
        SizeRange heightRange = 2;
    }

    // The required or default image size is width x height
    //
    // If specificationVersion <= 2 or SizeFlexibility is empty,
    // width x height is the required fixed image size
```

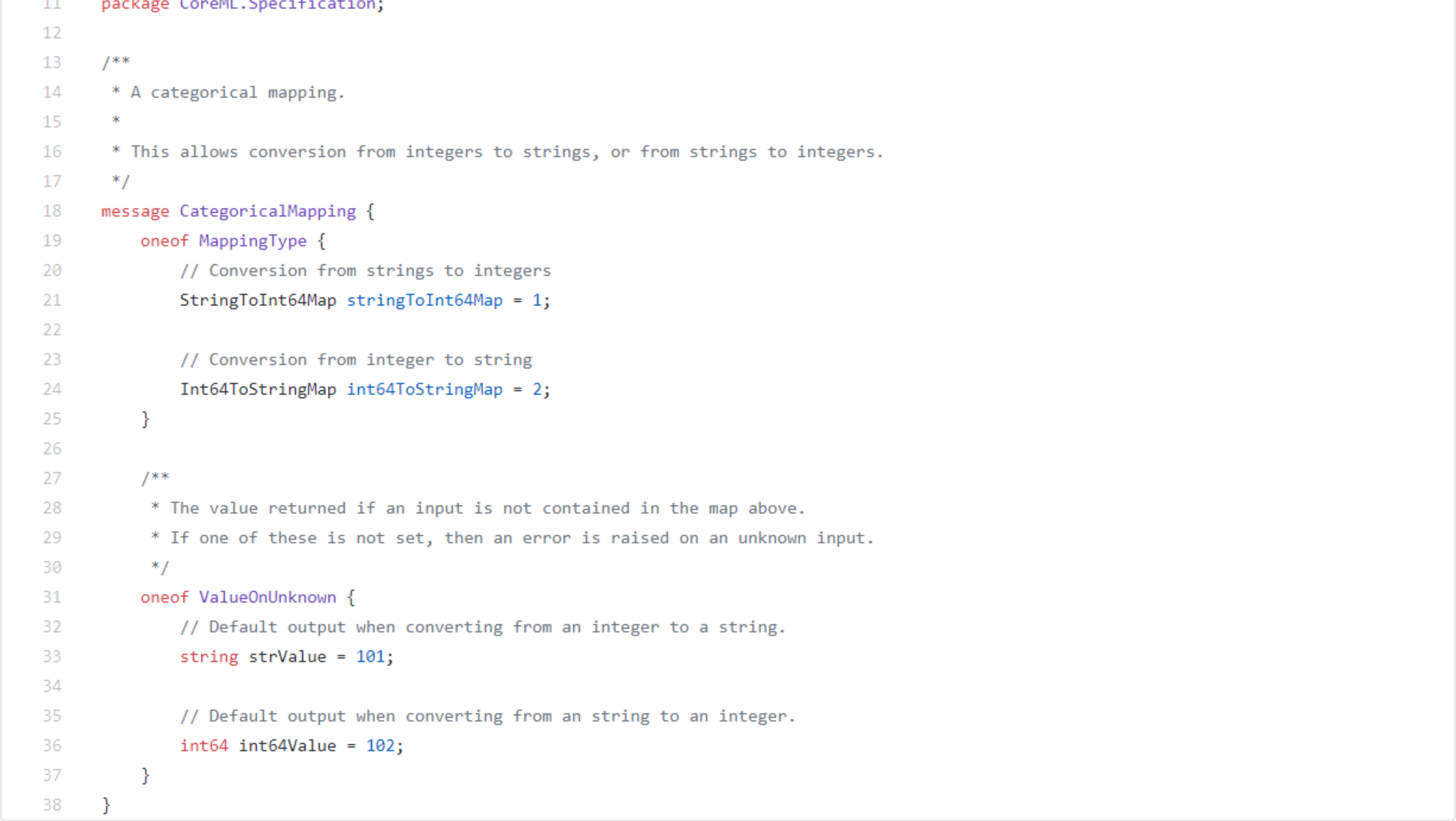


coremltools / mlmodel / format /

Sohaib Qureshi Consolidate changes for coremltools 2.0 public release — Latest commit f11c43e on Aug 29

..

ArrayFeatureExtractor.proto	Add Linux build support	a year ago
BayesianProbitRegressor.proto	Consolidated changes from 2.0b1 release	3 months ago
CategoricalMapping.proto	Add Linux build support	a year ago
CustomModel.proto	Consolidated changes from 2.0b1 release	3 months ago
DataStructures.proto	Add Linux build support	a year ago
DictVectorizer.proto	minor proto sp fixes	8 months ago
FeatureTypes.proto	Consolidated changes from 2.0b1 release	3 months ago
FeatureVectorizer.proto	Add Linux build support	a year ago
GLMClassifier.proto	Add Linux build support	a year ago
GLMRegressor.proto	Add Linux build support	a year ago
Identity.proto	Add Linux build support	a year ago



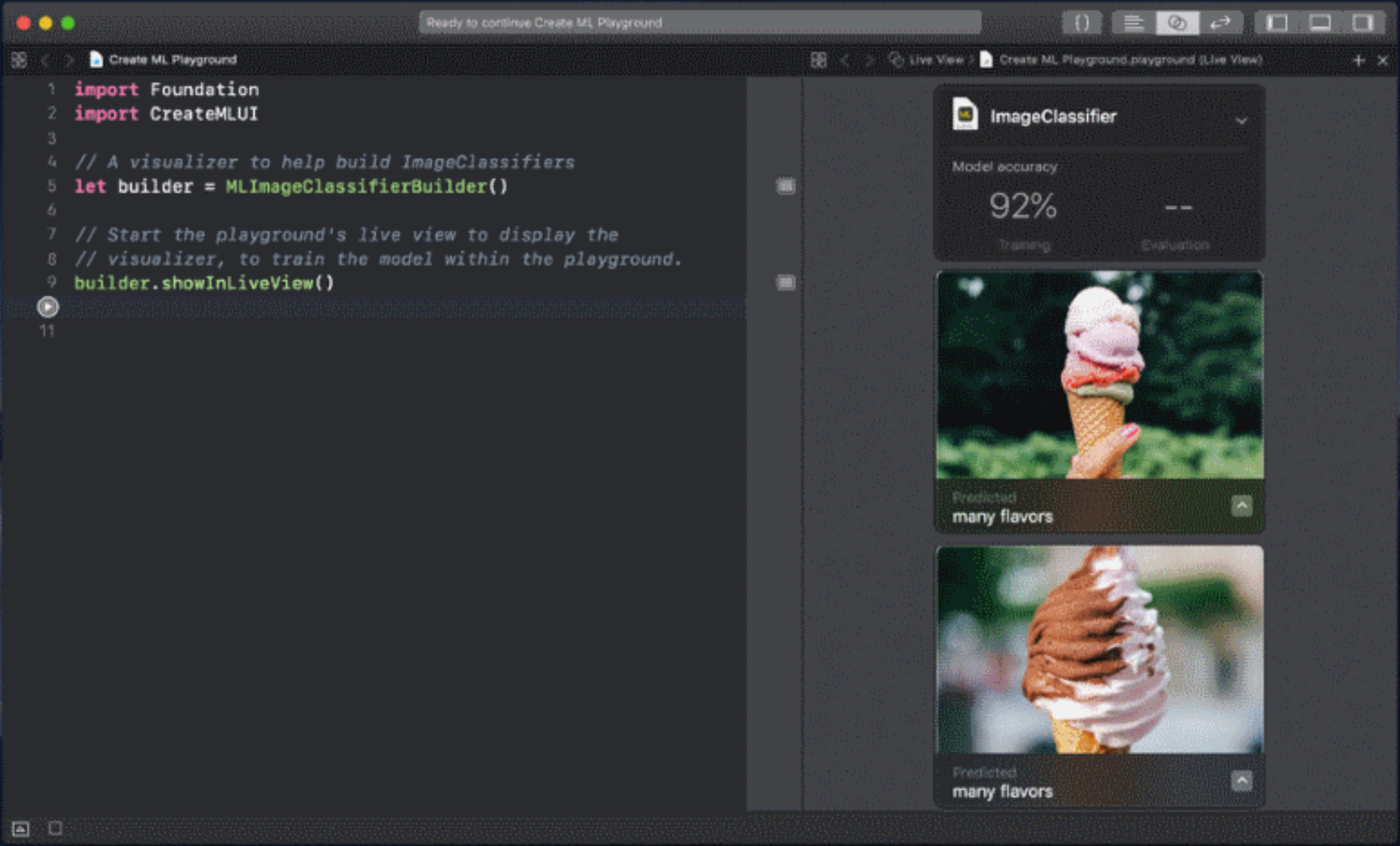
```
package CoreML.Specification;

/**
 * A categorical mapping.
 *
 * This allows conversion from integers to strings, or from strings to integers.
 */
message CategoricalMapping {
    oneof MappingType {
        // Conversion from strings to integers
        StringToInt64Map stringToInt64Map = 1;

        // Conversion from integer to string
        Int64ToStringMap int64ToStringMap = 2;
    }

    /**
     * The value returned if an input is not contained in the map above.
     * If one of these is not set, then an error is raised on an unknown input.
     */
    oneof ValueOnUnknown {
        // Default output when converting from an integer to a string.
        string strValue = 101;

        // Default output when converting from an string to an integer.
        int64 int64Value = 102;
    }
}
```

```swift
let c = MLModel.compile(url: u)
let m = MLModel(url: c)
let d:[String:Any] = ["input": image]
let dfpin = MLDictionaryFeatureProvider(dictionary: d)
let fp = m.predict(dfp)
let dfpout = fp.featureValue(for: "labels")
let dic = dfpout.dictionaryValue
let topResult = dic.first()
```

Vision and natural language

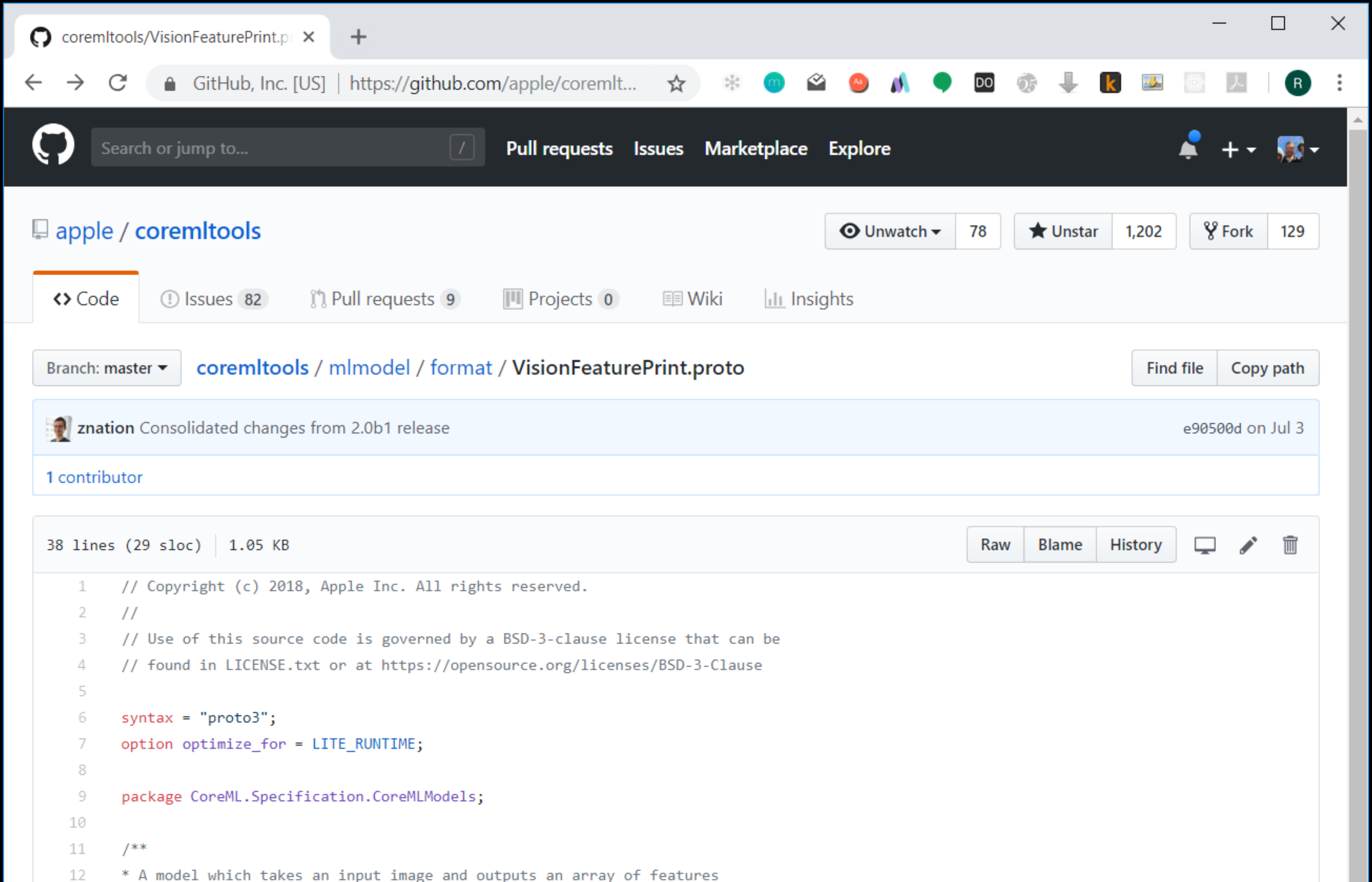
Custom data

Built in Swift

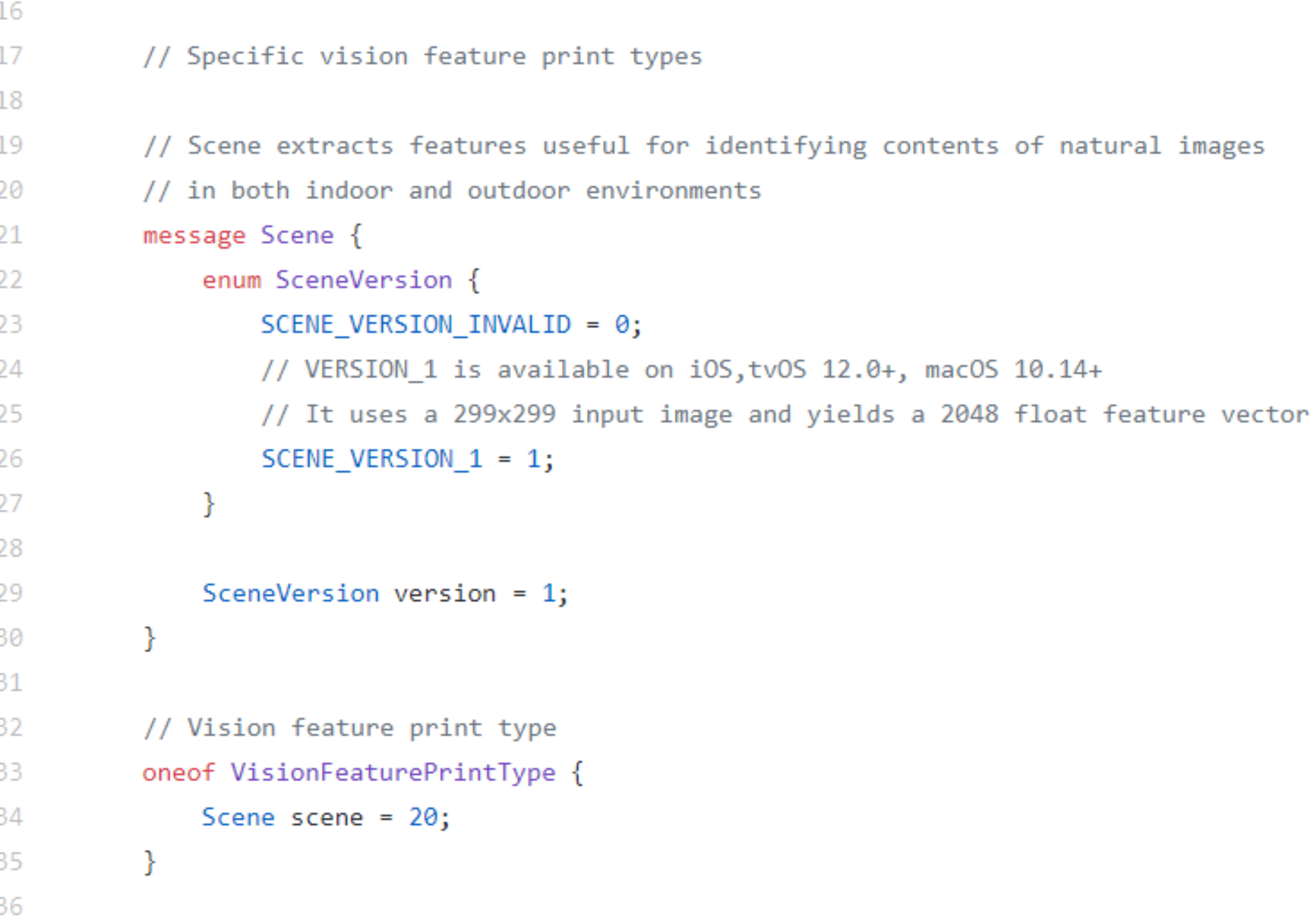


```
import Foundation
import CreateMLUI

// A visualizer to help build ImageClassifiers
let builder = MLImageClassifierBuilder()

// Start the playground's live view to display the
// visualizer, to train the model within the playground.
builder.showInLiveView()

```

ML



```
// Copyright (c) 2018, Apple Inc. All rights reserved.
//
// Use of this source code is governed by a BSD-3-clause license that can be
// found in LICENSE.txt or at https://opensource.org/licenses/BSD-3-Clause

syntax = "proto3";
option optimize_for = LITE_RUNTIME;

package CoreML.Specification.CoreMLModels;

/**
* A model which takes an input image and outputs an array of features
```



```
    // Specific vision feature print types

    // Scene extracts features useful for identifying contents of natural images
    // in both indoor and outdoor environments
    message Scene {
        enum SceneVersion {
            SCENE_VERSION_INVALID = 0;
            // VERSION_1 is available on iOS,tvOS 12.0+, macOS 10.14+
            // It uses a 299x299 input image and yields a 2048 float feature vector
            SCENE_VERSION_1 = 1;
        }

        SceneVersion version = 1;
    }

    // Vision feature print type
    oneof VisionFeaturePrintType {
        Scene scene = 20;
    }

```

Quo Vadis?

Conference React Native EU 2018

/reactnativeeu

@react_native_eu

Coming Soon

nicornot

.com

Powered by {callstack}

Thank You

@ray_deck

github.com/rhdeck/papis-2018