syntax = "proto3";

package tensorflow;

import "tensorflow/core/framework/tensor\_shape.proto";

import "tensorflow/core/framework/tensor\_slice.proto";

import "tensorflow/core/framework/types.proto";

import "tensorflow/core/framework/versions.proto";

option cc\_enable\_arenas = true;

option java\_outer\_classname = "TensorBundleProtos";

option java\_multiple\_files = true;

option java\_package = "org.tensorflow.util";

option go\_package = "github.com/tensorflow/tensorflow/tensorflow/go/core/protobuf/for\_core\_protos\_go\_proto";

// Protos used in the tensor bundle module (tf/core/util/tensor\_bundle/).

// Special header that is associated with a bundle.

//

// TODO(zongheng,zhifengc): maybe in the future, we can add information about

// which binary produced this checkpoint, timestamp, etc. Sometime, these can be

// valuable debugging information. And if needed, these can be used as defensive

// information ensuring reader (binary version) of the checkpoint and the writer

// (binary version) must match within certain range, etc.

message BundleHeaderProto {

// Number of data files in the bundle.

int32 num\_shards = 1;

// An enum indicating the endianness of the platform that produced this

// bundle. A bundle can only be read by a platform with matching endianness.

// Defaults to LITTLE, as most modern platforms are little-endian.

//

// Affects the binary tensor data bytes only, not the metadata in protobufs.

enum Endianness {

LITTLE = 0;

BIG = 1;

}

Endianness endianness = 2;

// Versioning of the tensor bundle format.

VersionDef version = 3;

}

// Describes the metadata related to a checkpointed tensor.

message BundleEntryProto {

// The tensor dtype and shape.

DataType dtype = 1;

TensorShapeProto shape = 2;

// The binary content of the tensor lies in:

// File "shard\_id": bytes [offset, offset + size).

int32 shard\_id = 3;

int64 offset = 4;

int64 size = 5;

// The CRC32C checksum of the tensor bytes.

fixed32 crc32c = 6;

// Iff present, this entry represents a partitioned tensor. The previous

// fields are interpreted as follows:

//

// "dtype", "shape": describe the full tensor.

// "shard\_id", "offset", "size", "crc32c": all IGNORED.

// These information for each slice can be looked up in their own

// BundleEntryProto, keyed by each "slice\_name".

repeated TensorSliceProto slices = 7;

}