syntax = "proto3";

package tensorflow;

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

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

option cc\_enable\_arenas = true;

option java\_outer\_classname = "ResourceHandle";

option java\_multiple\_files = true;

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

option go\_package = "github.com/tensorflow/tensorflow/tensorflow/go/core/framework/resource\_handle\_go\_proto";

// Protocol buffer representing a handle to a tensorflow resource. Handles are

// not valid across executions, but can be serialized back and forth from within

// a single run.

message ResourceHandleProto {

// Unique name for the device containing the resource.

string device = 1;

// Container in which this resource is placed.

string container = 2;

// Unique name of this resource.

string name = 3;

// Hash code for the type of the resource. Is only valid in the same device

// and in the same execution.

uint64 hash\_code = 4;

// For debug-only, the name of the type pointed to by this handle, if

// available.

string maybe\_type\_name = 5;

// Protocol buffer representing a pair of (data type, tensor shape).

message DtypeAndShape {

DataType dtype = 1;

TensorShapeProto shape = 2;

}

// Data types and shapes for the underlying resource.

repeated DtypeAndShape dtypes\_and\_shapes = 6;

reserved 7;

}