syntax = "proto3";

package tensorflow;

option cc\_enable\_arenas = true;

option java\_outer\_classname = "VariableProtos";

option java\_multiple\_files = true;

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

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

// Indicates when a distributed variable will be synced.

enum VariableSynchronization {

// `AUTO`: Indicates that the synchronization will be determined by the

// current `DistributionStrategy` (eg. With `MirroredStrategy` this would be

// `ON\_WRITE`).

VARIABLE\_SYNCHRONIZATION\_AUTO = 0;

// `NONE`: Indicates that there will only be one copy of the variable, so

// there is no need to sync.

VARIABLE\_SYNCHRONIZATION\_NONE = 1;

// `ON\_WRITE`: Indicates that the variable will be updated across devices

// every time it is written.

VARIABLE\_SYNCHRONIZATION\_ON\_WRITE = 2;

// `ON\_READ`: Indicates that the variable will be aggregated across devices

// when it is read (eg. when checkpointing or when evaluating an op that uses

// the variable).

VARIABLE\_SYNCHRONIZATION\_ON\_READ = 3;

}

// Indicates how a distributed variable will be aggregated.

enum VariableAggregation {

// `NONE`: This is the default, giving an error if you use a

// variable-update operation with multiple replicas.

VARIABLE\_AGGREGATION\_NONE = 0;

// `SUM`: Add the updates across replicas.

VARIABLE\_AGGREGATION\_SUM = 1;

// `MEAN`: Take the arithmetic mean ("average") of the updates across

// replicas.

VARIABLE\_AGGREGATION\_MEAN = 2;

// `ONLY\_FIRST\_REPLICA`: This is for when every replica is performing the same

// update, but we only want to perform the update once. Used, e.g., for the

// global step counter.

VARIABLE\_AGGREGATION\_ONLY\_FIRST\_REPLICA = 3;

}

// Protocol buffer representing a Variable.

message VariableDef {

// Name of the variable tensor.

string variable\_name = 1;

// Name of the tensor holding the variable's initial value.

string initial\_value\_name = 6;

// Name of the initializer op.

string initializer\_name = 2;

// Name of the snapshot tensor.

string snapshot\_name = 3;

// Support for saving variables as slices of a larger variable.

SaveSliceInfoDef save\_slice\_info\_def = 4;

// Whether to represent this as a ResourceVariable.

bool is\_resource = 5;

// Whether this variable should be trained.

bool trainable = 7;

// Indicates when a distributed variable will be synced.

VariableSynchronization synchronization = 8;

// Indicates how a distributed variable will be aggregated.

VariableAggregation aggregation = 9;

}

message SaveSliceInfoDef {

// Name of the full variable of which this is a slice.

string full\_name = 1;

// Shape of the full variable.

repeated int64 full\_shape = 2;

// Offset of this variable into the full variable.

repeated int64 var\_offset = 3;

// Shape of this variable.

repeated int64 var\_shape = 4;

}