syntax = "proto3";

package tensorflow.serving;

// Config proto for FileSystemStoragePathSource.

message FileSystemStoragePathSourceConfig {

// A policy that dictates which version(s) of a servable should be served.

message ServableVersionPolicy {

// Serve the latest versions (i.e. the ones with the highest version

// numbers), among those found on disk.

//

// This is the default policy, with the default number of versions as 1.

message Latest {

// Number of latest versions to serve. (The default is 1.)

uint32 num\_versions = 1;

}

// Serve all versions found on disk.

message All {

}

// Serve a specific version (or set of versions).

//

// This policy is useful for rolling back to a specific version, or for

// canarying a specific version while still serving a separate stable

// version.

message Specific {

// The version numbers to serve.

repeated int64 versions = 1;

}

oneof policy\_choice {

Latest latest = 100;

All all = 101;

Specific specific = 102;

}

}

// A servable name and base path to look for versions of the servable.

message ServableToMonitor {

// The servable name to supply in aspired-versions callback calls. Child

// paths of 'base\_path' are considered to be versions of this servable.

string servable\_name = 1;

// The path to monitor, i.e. look for child paths of the form base\_path/123.

string base\_path = 2;

// The policy to determines the number of versions of the servable to be

// served at the same time.

tensorflow.serving.FileSystemStoragePathSourceConfig.ServableVersionPolicy

servable\_version\_policy = 4;

reserved 3; // Legacy version\_policy definition.

}

// The servables to monitor for new versions, and aspire.

repeated ServableToMonitor servables = 5;

// A single servable name/base\_path pair to monitor.

// DEPRECATED: Use 'servables' instead.

// TODO(b/30898016): Stop using these fields, and ultimately remove them here.

string servable\_name = 1 [deprecated = true];

string base\_path = 2 [deprecated = true];

// How long to wait between file-system polling to look for children of

// 'base\_path', in seconds.

//

// If set to zero, filesystem will be polled exactly once. If set to a

// negative value (for testing use only), polling will be entirely disabled.

int64 file\_system\_poll\_wait\_seconds = 3;

// If true, then FileSystemStoragePathSource::Create() and ::UpdateConfig()

// fail if, for any configured servables, the file system doesn't currently

// contain at least one version under the base path.

// (Otherwise, it will emit a warning and keep pinging the file system to

// check for a version to appear later.)

// DEPRECATED: Use 'servable\_versions\_always\_present' instead, which includes

// this behavior.

// TODO(b/30898016): Remove 2019-10-31 or later.

bool fail\_if\_zero\_versions\_at\_startup = 4 [deprecated = true];

// If true, the servable is always expected to exist on the underlying

// filesystem. FileSystemStoragePathSource::Create() and ::UpdateConfig() will

// fail if, for any configured servables, the file system doesn't currently

// contain at least one version under the base path. In addition, if a polling

// loop find the base path empty, it will not unload existing servables.

bool servable\_versions\_always\_present = 6;

}