namespace java com.twitter.tweetypie.thriftjava

#@namespace scala com.twitter.tweetypie.thriftscala

#@namespace strato com.twitter.tweetypie

namespace py gen.twitter.tweetypie.media\_entity

namespace rb TweetyPie.media\_entity

namespace go tweetypie.media\_entity

include "com/twitter/mediaservices/commons/MediaInformation.thrift"

include "com/twitter/mediaservices/commons/MediaCommon.thrift"

include "com/twitter/mediaservices/commons/TweetMedia.thrift"

/\*\*

\* DEPRECATED

\* An RGB color.

\*

\* Each i8 should be interpreted as unsigned, ranging in value from 0 to

\* 255. Borrowed from gizmoduck/user.thrift.

\*

\* The way in which we use ColorValue here is as metadata for a media file,

\* so it needs to be annotated as having personal data. Fields that are of

\* structured types cannot be annotated, so we have to put the annotation

\* on the structure itself's fields even though it's more confusing to do so

\* and could introduce issues if someone else reuses ColorValue outside of

\* the context of a media file.

\*/

struct ColorValue {

1: i8 red (personalDataType = 'MediaFile')

2: i8 green (personalDataType = 'MediaFile')

3: i8 blue (personalDataType = 'MediaFile')

}(persisted = 'true', hasPersonalData = 'true')

struct MediaEntity {

1: i16 from\_index (personalDataType = 'MediaFile')

2: i16 to\_index (personalDataType = 'MediaFile')

/\*\*

\* The shortened t.co url found in the tweet text.

\*/

3: string url (personalDataType = 'ShortUrl')

/\*\*

\* The text to display in place of the shortened url.

\*/

4: string display\_url (personalDataType = 'LongUrl')

/\*\*

\* The url to the media asset (a preview image in the case of a video).

\*/

5: string media\_url (personalDataType = 'LongUrl')

/\*\*

\* The https version of media\_url.

\*/

6: string media\_url\_https (personalDataType = 'LongUrl')

/\*\*

\* The expanded media permalink.

\*/

7: string expanded\_url (personalDataType = 'LongUrl')

8: MediaCommon.MediaId media\_id (strato.space = "Media", strato.name = "media", personalDataType = 'MediaId')

9: bool nsfw

10: set<TweetMedia.MediaSize> sizes

11: string media\_path

12: optional bool is\_protected

/\*\*

\* The tweet that this MediaEntity was originally attached to. This value will be set if this

\* MediaEntity is either on a retweet or a tweet with pasted-pic.

\*/

13: optional i64 source\_status\_id (strato.space = "Tweet", strato.name = "sourceStatus", personalDataType = 'TweetId')

/\*\*

\* The user to attribute views of the media to.

\*

\* This field should be set when the media's attributableUserId field does not match the current

\* Tweet's owner. Retweets of a Tweet with media and "managed media" are some reasons this may

\* occur. When the value is None any views should be attributed to the tweet's owner.

\*\*/

14: optional i64 source\_user\_id (strato.space = "User", strato.name = "sourceUser", personalDataType = 'UserId')

/\*\*

\* Additional information specific to the media type.

\*

\* This field is optional with images (as the image information is in the

\* previous fields), but required for animated GIF and native video (as, in

\* this case, the previous fields only describe the preview image).

\*/

15: optional TweetMedia.MediaInfo media\_info

/\*\*

\* DEPRECATED

\* The dominant color for the entire image (or keyframe for video or GIF).

\*

\* This can be used for placeholders while the media downloads (either a

\* solid color or a gradient using the grid).

\*/

16: optional ColorValue dominant\_color\_overall

/\*\*

\* DEPRECATED

\* Dominant color of each quadrant of the image (keyframe for video or GIF).

\*

\* If present this list should have 4 elements, corresponding to

\* [top\_left, top\_right, bottom\_left, bottom\_right]

\*/

17: optional list<ColorValue> dominant\_color\_grid

// obsolete 18: optional map<string, binary> extensions

/\*\*

\* Stratostore extension points data encoded as a Strato record.

\*/

19: optional binary extensions\_reply

/\*\*

\* Holds metadata defined by the user for the tweet-asset relationship.

\*/

20: optional MediaInformation.UserDefinedProductMetadata metadata

/\*\*

\* Media key used to interact with the media systems.

\*/

21: optional MediaCommon.MediaKey media\_key

/\*\*

\* Flexible structure for additional media metadata. This field is only

\* included in a read-path request if specifically requested. It will

\* always be included, when applicable, in write-path responses.

\*/

22: optional MediaInformation.AdditionalMetadata additional\_metadata

}(persisted='true', hasPersonalData = 'true')