package com.twitter.tweetypie.core

sealed trait StoredTweetResult {

def canHydrate: Boolean

}

object StoredTweetResult {

sealed trait Error

object Error {

case object Corrupt extends Error

case object ScrubbedFieldsPresent extends Error

case object FieldsMissingOrInvalid extends Error

case object ShouldBeHardDeleted extends Error

}

case class Present(errors: Seq[Error], canHydrate: Boolean) extends StoredTweetResult

case class HardDeleted(softDeletedAtMsec: Long, hardDeletedAtMsec: Long)

extends StoredTweetResult {

override def canHydrate: Boolean = false

}

case class SoftDeleted(softDeletedAtMsec: Long, errors: Seq[Error], canHydrate: Boolean)

extends StoredTweetResult

case class BounceDeleted(deletedAtMsec: Long, errors: Seq[Error], canHydrate: Boolean)

extends StoredTweetResult

case class Undeleted(undeletedAtMsec: Long, errors: Seq[Error], canHydrate: Boolean)

extends StoredTweetResult

case class ForceAdded(addedAtMsec: Long, errors: Seq[Error], canHydrate: Boolean)

extends StoredTweetResult

case class Failed(errors: Seq[Error]) extends StoredTweetResult {

override def canHydrate: Boolean = false

}

object NotFound extends StoredTweetResult {

override def canHydrate: Boolean = false

}

}