package com.twitter.frigate.pushservice.store

import com.twitter.livevideo.common.ids.EventId

import com.twitter.livevideo.timeline.client.v2.LiveVideoTimelineClient

import com.twitter.livevideo.timeline.domain.v2.Event

import com.twitter.livevideo.timeline.domain.v2.LookupContext

import com.twitter.stitch.storehaus.ReadableStoreOfStitch

import com.twitter.stitch.NotFound

import com.twitter.stitch.Stitch

import com.twitter.storehaus.ReadableStore

case class EventRequest(eventId: Long, lookupContext: LookupContext = LookupContext.default)

object LexServiceStore {

def apply(

liveVideoTimelineClient: LiveVideoTimelineClient

): ReadableStore[EventRequest, Event] = {

ReadableStoreOfStitch { eventRequest =>

liveVideoTimelineClient.getEvent(

EventId(eventRequest.eventId),

eventRequest.lookupContext) rescue {

case NotFound => Stitch.NotFound

}

}

}

}