package com.twitter.search.earlybird.common;

import java.util.Optional;

import com.twitter.common.optional.Optionals;

import com.twitter.search.common.util.FinagleUtil;

import com.twitter.search.earlybird.thrift.EarlybirdRequest;

import com.twitter.strato.opcontext.Attribution;

import com.twitter.strato.opcontext.HttpEndpoint;

public final class ClientIdUtil {

// Blenders should always set the EarlybirdRequest.clientId field. It should be set to the Finagle

// client ID of the client that caused the blender to send this request to the roots. If the

// Finagle ID of the blender's client cannot be determined, it will be set to "unknown" (see

// com.twitter.search.common.util.FinagleUtil.UNKNOWN\_CLIENT\_NAME). However, other services that

// send requests to roots might not set EarlybirdRequest.clientId.

//

// So an "unset" clientId means: EarlybirdRequest.clientId was null.

// An "unknown" clientId means: the client that sent us the request

// tried setting EarlybirdRequest.clientId, but couldn't figure out a good value for it.

public static final String UNSET\_CLIENT\_ID = "unset";

private static final String CLIENT\_ID\_FOR\_UNKNOWN\_CLIENTS = "unknown\_client\_id";

private static final String CLIENT\_ID\_PREFIX = "client\_id\_";

private static final String FINAGLE\_CLIENT\_ID\_AND\_CLIENT\_ID\_PATTERN =

"finagle\_id\_%s\_and\_client\_id\_%s";

private static final String CLIENT\_ID\_AND\_REQUEST\_TYPE = "client\_id\_%s\_and\_type\_%s";

private ClientIdUtil() {

}

/\*\* Returns the ID of the client that initiated this request or UNSET\_CLIENT\_ID if not set. \*/

public static String getClientIdFromRequest(EarlybirdRequest request) {

return Optional

.ofNullable(request.getClientId())

.map(String::toLowerCase)

.orElse(UNSET\_CLIENT\_ID);

}

/\*\*

\* Returns the Strato http endpoint attribution as an Optional.

\*/

public static Optional<String> getClientIdFromHttpEndpointAttribution() {

return Optionals

.optional(Attribution.httpEndpoint())

.map(HttpEndpoint::name)

.map(String::toLowerCase);

}

/\*\* Formats the given clientId into a string that can be used for stats. \*/

public static String formatClientId(String clientId) {

return CLIENT\_ID\_PREFIX + clientId;

}

/\*\*

\* Formats the given Finagle clientId and the given clientId into a single string that can be used

\* for stats, or other purposes where the two IDs need to be combined.

\*/

public static String formatFinagleClientIdAndClientId(String finagleClientId, String clientId) {

return String.format(FINAGLE\_CLIENT\_ID\_AND\_CLIENT\_ID\_PATTERN, finagleClientId, clientId);

}

/\*\*

\* Formats the given clientId and requestType into a single string that can be used

\* for stats or other purposes.

\*/

public static String formatClientIdAndRequestType(

String clientId, String requestType) {

return String.format(CLIENT\_ID\_AND\_REQUEST\_TYPE, clientId, requestType);

}

/\*\*

\* Format the quota client id

\*/

public static String getQuotaClientId(String clientId) {

if (FinagleUtil.UNKNOWN\_CLIENT\_NAME.equals(clientId) || UNSET\_CLIENT\_ID.equals(clientId)) {

return CLIENT\_ID\_FOR\_UNKNOWN\_CLIENTS;

}

return clientId;

}

}