package com.twitter.ann.common

import com.twitter.stitch.Stitch

/\*\*

\* This is a trait that allows you to query for nearest neighbors given an arbitrary type T1. This is

\* in contrast to a regular com.twitter.ann.common.Appendable, which takes an embedding as the input

\* argument.

\*

\* This interface uses the Stitch API for batching. See go/stitch for details on how to use it.

\*

\* @tparam T1 type of the query.

\* @tparam T2 type of the result.

\* @tparam P runtime parameters supported by the index.

\* @tparam D distance function used in the index.

\*/

trait QueryableById[T1, T2, P <: RuntimeParams, D <: Distance[D]] {

def queryById(

id: T1,

numOfNeighbors: Int,

runtimeParams: P

): Stitch[List[T2]]

def queryByIdWithDistance(

id: T1,

numOfNeighbors: Int,

runtimeParams: P

): Stitch[List[NeighborWithDistance[T2, D]]]

def batchQueryById(

ids: Seq[T1],

numOfNeighbors: Int,

runtimeParams: P

): Stitch[List[NeighborWithSeed[T1, T2]]]

def batchQueryWithDistanceById(

ids: Seq[T1],

numOfNeighbors: Int,

runtimeParams: P

): Stitch[List[NeighborWithDistanceWithSeed[T1, T2, D]]]

}