/\* ----------------------------------------------------------------------------

\* This file was automatically generated by SWIG (http://www.swig.org).

\* Version 4.0.2

\*

\* Do not make changes to this file unless you know what you are doing--modify

\* the SWIG interface file instead.

\* ----------------------------------------------------------------------------- \*/

package com.twitter.ann.faiss;

public class OnDiskInvertedLists extends InvertedLists {

private transient long swigCPtr;

protected OnDiskInvertedLists(long cPtr, boolean cMemoryOwn) {

super(swigfaissJNI.OnDiskInvertedLists\_SWIGUpcast(cPtr), cMemoryOwn);

swigCPtr = cPtr;

}

protected static long getCPtr(OnDiskInvertedLists obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_OnDiskInvertedLists(swigCPtr);

}

swigCPtr = 0;

}

super.delete();

}

public void setLists(SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_OnDiskOneList\_t value) {

swigfaissJNI.OnDiskInvertedLists\_lists\_set(swigCPtr, this, SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_OnDiskOneList\_t.getCPtr(value));

}

public SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_OnDiskOneList\_t getLists() {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_lists\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_std\_\_vectorT\_faiss\_\_OnDiskOneList\_t(cPtr, false);

}

static public class Slot {

private transient long swigCPtr;

protected transient boolean swigCMemOwn;

protected Slot(long cPtr, boolean cMemoryOwn) {

swigCMemOwn = cMemoryOwn;

swigCPtr = cPtr;

}

protected static long getCPtr(Slot obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_OnDiskInvertedLists\_Slot(swigCPtr);

}

swigCPtr = 0;

}

}

public void setOffset(long value) {

swigfaissJNI.OnDiskInvertedLists\_Slot\_offset\_set(swigCPtr, this, value);

}

public long getOffset() {

return swigfaissJNI.OnDiskInvertedLists\_Slot\_offset\_get(swigCPtr, this);

}

public void setCapacity(long value) {

swigfaissJNI.OnDiskInvertedLists\_Slot\_capacity\_set(swigCPtr, this, value);

}

public long getCapacity() {

return swigfaissJNI.OnDiskInvertedLists\_Slot\_capacity\_get(swigCPtr, this);

}

public Slot(long offset, long capacity) {

this(swigfaissJNI.new\_OnDiskInvertedLists\_Slot\_\_SWIG\_0(offset, capacity), true);

}

public Slot() {

this(swigfaissJNI.new\_OnDiskInvertedLists\_Slot\_\_SWIG\_1(), true);

}

}

public void setSlots(SWIGTYPE\_p\_std\_\_listT\_faiss\_\_OnDiskInvertedLists\_\_Slot\_t value) {

swigfaissJNI.OnDiskInvertedLists\_slots\_set(swigCPtr, this, SWIGTYPE\_p\_std\_\_listT\_faiss\_\_OnDiskInvertedLists\_\_Slot\_t.getCPtr(value));

}

public SWIGTYPE\_p\_std\_\_listT\_faiss\_\_OnDiskInvertedLists\_\_Slot\_t getSlots() {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_slots\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_std\_\_listT\_faiss\_\_OnDiskInvertedLists\_\_Slot\_t(cPtr, false);

}

public void setFilename(String value) {

swigfaissJNI.OnDiskInvertedLists\_filename\_set(swigCPtr, this, value);

}

public String getFilename() {

return swigfaissJNI.OnDiskInvertedLists\_filename\_get(swigCPtr, this);

}

public void setTotsize(long value) {

swigfaissJNI.OnDiskInvertedLists\_totsize\_set(swigCPtr, this, value);

}

public long getTotsize() {

return swigfaissJNI.OnDiskInvertedLists\_totsize\_get(swigCPtr, this);

}

public void setPtr(SWIGTYPE\_p\_unsigned\_char value) {

swigfaissJNI.OnDiskInvertedLists\_ptr\_set(swigCPtr, this, SWIGTYPE\_p\_unsigned\_char.getCPtr(value));

}

public SWIGTYPE\_p\_unsigned\_char getPtr() {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_ptr\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

public void setRead\_only(boolean value) {

swigfaissJNI.OnDiskInvertedLists\_read\_only\_set(swigCPtr, this, value);

}

public boolean getRead\_only() {

return swigfaissJNI.OnDiskInvertedLists\_read\_only\_get(swigCPtr, this);

}

public OnDiskInvertedLists(long nlist, long code\_size, String filename) {

this(swigfaissJNI.new\_OnDiskInvertedLists\_\_SWIG\_0(nlist, code\_size, filename), true);

}

public long list\_size(long list\_no) {

return swigfaissJNI.OnDiskInvertedLists\_list\_size(swigCPtr, this, list\_no);

}

public SWIGTYPE\_p\_unsigned\_char get\_codes(long list\_no) {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_get\_codes(swigCPtr, this, list\_no);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

public LongVector get\_ids(long list\_no) {

return new LongVector(swigfaissJNI.OnDiskInvertedLists\_get\_ids(swigCPtr, this, list\_no), false);

}

public long add\_entries(long list\_no, long n\_entry, LongVector ids, SWIGTYPE\_p\_unsigned\_char code) {

return swigfaissJNI.OnDiskInvertedLists\_add\_entries(swigCPtr, this, list\_no, n\_entry, SWIGTYPE\_p\_long\_long.getCPtr(ids.data()), ids, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public void update\_entries(long list\_no, long offset, long n\_entry, LongVector ids, SWIGTYPE\_p\_unsigned\_char code) {

swigfaissJNI.OnDiskInvertedLists\_update\_entries(swigCPtr, this, list\_no, offset, n\_entry, SWIGTYPE\_p\_long\_long.getCPtr(ids.data()), ids, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public void resize(long list\_no, long new\_size) {

swigfaissJNI.OnDiskInvertedLists\_resize(swigCPtr, this, list\_no, new\_size);

}

public long merge\_from(SWIGTYPE\_p\_p\_faiss\_\_InvertedLists ils, int n\_il, boolean verbose) {

return swigfaissJNI.OnDiskInvertedLists\_merge\_from\_\_SWIG\_0(swigCPtr, this, SWIGTYPE\_p\_p\_faiss\_\_InvertedLists.getCPtr(ils), n\_il, verbose);

}

public long merge\_from(SWIGTYPE\_p\_p\_faiss\_\_InvertedLists ils, int n\_il) {

return swigfaissJNI.OnDiskInvertedLists\_merge\_from\_\_SWIG\_1(swigCPtr, this, SWIGTYPE\_p\_p\_faiss\_\_InvertedLists.getCPtr(ils), n\_il);

}

public long merge\_from\_1(InvertedLists il, boolean verbose) {

return swigfaissJNI.OnDiskInvertedLists\_merge\_from\_1\_\_SWIG\_0(swigCPtr, this, InvertedLists.getCPtr(il), il, verbose);

}

public long merge\_from\_1(InvertedLists il) {

return swigfaissJNI.OnDiskInvertedLists\_merge\_from\_1\_\_SWIG\_1(swigCPtr, this, InvertedLists.getCPtr(il), il);

}

public void crop\_invlists(long l0, long l1) {

swigfaissJNI.OnDiskInvertedLists\_crop\_invlists(swigCPtr, this, l0, l1);

}

public void prefetch\_lists(LongVector list\_nos, int nlist) {

swigfaissJNI.OnDiskInvertedLists\_prefetch\_lists(swigCPtr, this, SWIGTYPE\_p\_long\_long.getCPtr(list\_nos.data()), list\_nos, nlist);

}

public void setLocks(SWIGTYPE\_p\_faiss\_\_LockLevels value) {

swigfaissJNI.OnDiskInvertedLists\_locks\_set(swigCPtr, this, SWIGTYPE\_p\_faiss\_\_LockLevels.getCPtr(value));

}

public SWIGTYPE\_p\_faiss\_\_LockLevels getLocks() {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_locks\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_faiss\_\_LockLevels(cPtr, false);

}

public void setPf(SWIGTYPE\_p\_faiss\_\_OnDiskInvertedLists\_\_OngoingPrefetch value) {

swigfaissJNI.OnDiskInvertedLists\_pf\_set(swigCPtr, this, SWIGTYPE\_p\_faiss\_\_OnDiskInvertedLists\_\_OngoingPrefetch.getCPtr(value));

}

public SWIGTYPE\_p\_faiss\_\_OnDiskInvertedLists\_\_OngoingPrefetch getPf() {

long cPtr = swigfaissJNI.OnDiskInvertedLists\_pf\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_faiss\_\_OnDiskInvertedLists\_\_OngoingPrefetch(cPtr, false);

}

public void setPrefetch\_nthread(int value) {

swigfaissJNI.OnDiskInvertedLists\_prefetch\_nthread\_set(swigCPtr, this, value);

}

public int getPrefetch\_nthread() {

return swigfaissJNI.OnDiskInvertedLists\_prefetch\_nthread\_get(swigCPtr, this);

}

public void do\_mmap() {

swigfaissJNI.OnDiskInvertedLists\_do\_mmap(swigCPtr, this);

}

public void update\_totsize(long new\_totsize) {

swigfaissJNI.OnDiskInvertedLists\_update\_totsize(swigCPtr, this, new\_totsize);

}

public void resize\_locked(long list\_no, long new\_size) {

swigfaissJNI.OnDiskInvertedLists\_resize\_locked(swigCPtr, this, list\_no, new\_size);

}

public long allocate\_slot(long capacity) {

return swigfaissJNI.OnDiskInvertedLists\_allocate\_slot(swigCPtr, this, capacity);

}

public void free\_slot(long offset, long capacity) {

swigfaissJNI.OnDiskInvertedLists\_free\_slot(swigCPtr, this, offset, capacity);

}

public void set\_all\_lists\_sizes(SWIGTYPE\_p\_unsigned\_long sizes) {

swigfaissJNI.OnDiskInvertedLists\_set\_all\_lists\_sizes(swigCPtr, this, SWIGTYPE\_p\_unsigned\_long.getCPtr(sizes));

}

public OnDiskInvertedLists() {

this(swigfaissJNI.new\_OnDiskInvertedLists\_\_SWIG\_1(), true);

}

}