package com.twitter.search.core.earlybird.index.inverted;

/\*\*

\* Encodes and decodes term pointers.

\*/

public abstract class TermPointerEncoding {

/\*\*

\* Returns the start of the text stored in a {@link BaseByteBlockPool} of the given term.

\*/

public abstract int getTextStart(int termPointer);

/\*\*

\* Returns true, if the given term stores a per-term payload.

\*/

public abstract boolean hasPayload(int termPointer);

/\*\*

\* Encodes and returns a pointer for a term stored at the given textStart in a

\* {@link BaseByteBlockPool}.

\*/

public abstract int encodeTermPointer(int textStart, boolean hasPayload);

public static final TermPointerEncoding DEFAULT\_ENCODING = new TermPointerEncoding() {

@Override public int getTextStart(int termPointer) {

return termPointer >>> 1;

}

@Override public boolean hasPayload(int termPointer) {

return (termPointer & 1) != 0;

}

@Override

public int encodeTermPointer(int textStart, boolean hasPayload) {

int code = textStart << 1;

return hasPayload ? (code | 1) : code;

}

};

}