**SUMMARY**

A concurrency bug in Eclipse 4.5 JDT Core.

**DETAILS**

Some details can also be found at: <https://bugs.eclipse.org/bugs/show_bug.cgi?id=455882>

This bug is due to a data race.

|  |  |
| --- | --- |
| Thread1 (JavaElementDelta.java) | Thread2 (JavaElementDelta.java) |
| protected void removeExistingChild(Key key, int index) {  this.affectedChildren = removeAndShrinkArray(this.affectedChildren, index);  if (this.childIndex != null) {  int length = this.affectedChildren.length;  if (length < NEED\_CHILD\_INDEX)  this.childIndex = null;  else {  this.childIndex.remove(key);  for (int i = index; i < length; i++) {  this.childIndex.put(new Key(this.affectedChildren[i].getElement()), i);  }  }  }  } | protected void removeExistingChild(Key key, int index) {  this.affectedChildren = removeAndShrinkArray(this.affectedChildren, index);  if (this.childIndex != null) {  int length = this.affectedChildren.length;  if (length < NEED\_CHILD\_INDEX)  this.childIndex = null;  else {  this.childIndex.remove(key);  for (int i = index; i < length; i++) {  this.childIndex.put(new Key(this.affectedChildren[i].getElement()), i);  }  }  }  } |
| protected IJavaElementDelta[] removeAndShrinkArray(IJavaElementDelta[] old, int index) {  IJavaElementDelta[] array = new IJavaElementDelta[old.length - 1];  if (index > 0)  System.arraycopy(old, 0, array, 0, index);  int rest = old.length - index - 1;  if (rest > 0)  System.arraycopy(old, index + 1, array, index, rest);  return array;  } |  |
| 0-lock | 0-lock |