

Saxon - Bug #4966

Detecting circular dependencies among accumulators

2021-04-13 19:33 - Michael Kay

Status:	Closed	Start date:	2021-04-13
Priority:	Low	Due date:	
Assignee:	Michael Kay	% Done:	0%
Category:		Estimated time:	0:00 hour
Sprint/Milestone:		Spent time:	0:00 hour
Legacy ID:		Fix Committed on Branch:	trunk
Applies to branch:	10, trunk	Fixed in Maintenance Release:	

Description

The logic for detecting circularities among accumulators appears to be flawed.

Test accumulator-080 tests this condition, and we pass the test by catching a `StackOverflowError` in `AccumulatorData.visit()`. But it's not possible to catch a `StackOverflow` in C#, and this reveals that the other mechanisms which should catch the circularity earlier (long before the `StackOverflow` occurs) are failing to do so.

Specifically, there are two mechanisms

(a) `AccumulatorData.buildIndex()` keeps a local flag "building" which should catch any recursive entry into `buildIndex()` for the same `AccumulatorData` object. But this achieves nothing because we're creating a new `AccumulatorData` object on each recursive entry.

(b) `AccumulatorManager.getAccumulatorData()` keeps (for each document) a map from accumulator declarations to accumulator data objects, and the logic is designed to place a `MARKER` in this map while the accumulator data is under construction. But the code to place this `MARKER` in the map doesn't get executed on the path where the map exists, but doesn't hold an entry for the relevant accumulator: that is, the `MARKER` is only set up for the first accumulator that's built for each document.

History

#1 - 2021-04-13 19:35 - Michael Kay

Fixed by moving the line `map.put(acc, MARKER)`; out of the else branch of the conditional (`AccumulatorManager.getAccumulatorData()`)

#2 - 2021-04-13 19:39 - Michael Kay

- Fix Committed on Branch trunk added

Fixed on the 11.x branch; I'll hold back on 10.x until I'm more confident there's no regression.

#3 - 2021-06-09 12:16 - Michael Kay

- Status changed from New to Closed

Decided to make no change to the 10.x as the problem is asymptomatic in the issued product.