Performance Optimisations Angelo D' Agnano John Kostaras

## **Performance** activities

#### **Performance Monitoring**

- \* an activity of nonintrusively collecting or observing performance data from a running application
- \* Performance Profiling
  - an activity of collecting performance data from a running application that may be intrusive on application performance responsiveness or throughput
- \* Performance Tuning
  - an activity of changing tuneables, source code, or configuration attribute(s) for the purposes of improving application responsiveness or throughput

## OS Performance Monitoring

#### \* CPU

- \* Graphical
  - \* gnome-system-monitor, xosview (Unix/Linux)
  - \* cpubar (Solaris)
  - \* TaskManager, Performance Monitor (Windows)
- \* Text
  - \* vmstat, mpstat, pidstat, top
  - \* prstat (Solaris)
  - \* typeperf (Windows)

## OS Performance Monitoring (cont.)

- \* Memory
  - \* vmstat
  - \* cpubar (Solaris)
  - \* Performance Monitor, typeperf (Windows)
- \* Network
  - \* nicstat, typeperf (Windows)
- \* I/O
  - \* iostat, iosnoop.d (Solaris)
- \* Other
  - \* sar (sysstat), kstat (Solaris), cpustat, cputrack

## Performance Profiling -Types of Profilers

- Method profiler
  - Collects information about method execution times
  - Look for: internal/external method times, frequently called methods/classes etc.
- Memory profiler
  - Collects information about object creation and/or garbage collection
- Thread profiler
  - Looks for thread conflict situations

## Performance Profiling -Profilers

- Method profilers
  - Oracle Solaris Studio Performance Analyzer, HPJMeter, JVisualVM
  - <u>Free</u>: java -agentlib:hprof=[help][[<option>=<value>, ...], jvmstat, JRockit Runtime Analyzer
  - <u>Commercial</u>: JProbe, OptimizeIt!, JProfiler, YourKit
- Memory profilers
  - Oracle Solaris Studio Performance Analyzer
  - HPJMeter
  - JVisualVM
- Thread profilers
  - JVisualVM
  - Google's JChord, Java Race Finder, Mutability Detector, YourKit

A few things about the application to optimise

- \* A soft real-time Java application based on NetBeans RCP
- \* Displays plots and tracks on a digital map
- \* Requirement to run on SunBlade 1500 (1 CPU 1.062 GHz UltraSPARC, 1 GB RAM)

## Steps executed

- Research of root causes
- \* Analysis of GC impact
- Verification of lock contention
- \* Hotspots in application code
- 2) Profile main branch application
- 3) Identification of hotspots
- 4) Optimisation of hotspots
- 5) Performance measurement with & without changes
- 6) Integration of changes
- 7) Restart from point 1

## **CPU Performance Monitoring**



### Monitoring Results of Main branch Plots & tracks with labels displayed CPU: 64%





າce 11

## **CPU Performance Profiling**

### Identification of Hotspot 1

#### Exclusive & inclusive user CPU utilisation metrics

Functions	Callers-Calle s Call Tree Source Disassembly Timeline Experiments	D Summary Timel	ine Details	
RUser	El User 182 ve		Selected Object	
CPU		Name: 20	altiple Selection	
100.00	100.00 - 77 (alb	- PC Address:		
8,92	1.92 c7 01-5ra (app.	Size: 1	13	
0.91	9,76 1a w.util.confurrent.ConcurrentLinkedOwne.minc()	Source File: 3	www/util/concurrent/Concurrent	ntLinkedQueue.java
5.64	5.64 ml.p ImmgeCopy na	Object File:		
1.96	8.04 java.lang.ttrictNath.pow(double, double)	Load Object	telasses>	
2,00	3.92isee754_pow	Mangled Name:		
2.01	2.01 mlub_v_ImmgeCiene_022_1	Alizzes		
1.94	15.06 com.lucimd.internal.transformation.i.a(com.lucied.shape.y, com.lucied.shape.shape3D.o)	Guaran.		
1,63	11.61 com.luciad.projection.WiodLambertConformal.a(com.luciad.shape.y, com.luciad.geodesy.d, com.luciad.shape.shapelD.o)		Metrics for selected Object	
1.30	1.30 mm more plats.AbstractPlot.getBounds()	three cases	AK Exclusive	Inclusive
1.29	4.43 sum.javald.loops.brandine.brandine(sum.javald.lunGraphics2b, sum.javald.lurfaceData, int, int, int, int, int)	gser cro.	0.216 (31.128)	9,099 134,121
1.20	26.57 generation of the second se	Mair	0, (0, 4)	0. (0. 1
0.98	0.98 java.lmag.Object.closm()	Total Thread:	131630 ( 0.97%)	14.600 ( 1.041
0.95	0.95 com.luriad.util.T.d(double)	System CPU:	0. (0. 1)	0. (0. *
0.91	0.90 generation of the second se	Wajt CPU:	4,953 (10,27%)	5.154 (19.014
0.00	0.00 java, Lang. String. charAt(int)	User Lock:	0.350 ( 0.07%)	0.350 ( 0.07#
0.84	0.86 com.luziad.view.gzy.arFa.a(druble, double, jwos.awt.Point)	Text Page Fault:	0. (0. 1)	0. (0. *
0,03	0.86 [ave.util.concurrent.ConcurrentLinkedQueue.macr(jave.util.concurrent.ConcurrentLinkedQueueIBode)	Data Page Fault:	0. (0. 4)	0. (0. )
0.76	15.70 com.luriad.transformation.TLodGeodeticJGeid.a(com.luriad.shape.y, com.luriad.shapelb.c)	Other Wait:	0. (0. 1)	Q. (Q. )
0.75	0.75 can Java callatesk seconded-			
0.75	0.73 com.luciad.shape.shape?p.n.getEinT()			
0.73	0.71 java.lang.Integer.equals(java.lang.Object)			
0.48	0,72 (]ava.ukii.concurrent.CoccurrentLiAAeQueuelitr.sdvance[]			
0.66	2.22 AMAGENP JANATG TOOR DEAVETING LEAVETING			
0.93	1.99 com.ucimo.view.s.a.(ava.ucilibint, java.lang.uogeot, 105, 107)			
0.40	arian org.met.un/ameriannice.compon.pputia.u.getAwaioontion(con.amian.view.gry.m, con.awian.eoge.y) 0.01 - on bored sectors i closels - deals			
0.50	U-SE COMPANEAREAREAREAREAREAREAREAREAREAREAREAREARE			
0.51	0.51 as savety end of the second seco			
0.44	26.02 rem Initial Visu mut and anniuch lang Chierth			
0.44	19.22 org.nato.usclassified.usc.mas.late.oil.cainters.d.maintipaya.ast.dranbirs.int. com.luctad.vieu.mv.wl			
0.46	0.46 sun.misc. Onsefe.serk/hooless. ison)			
0.44	3 43 more well more descent and black			

### Identification of Hotspot 2

Tile View	Oracle Sistais Performance Analyzer (MainBranch-Tracks.er (java))			9 - 9
885	A A A A A A A A A A A A A A A A A A A			
Functions	Callers-Callees Call Tree Source Disassembly Timeline Experiments	Summary Timelin	e Details	
Hitter	E User Name		Selected Object	
CPU	CPU	Name -		and the search .
(%)	V (%) 30.94 ] HWAR.BWING.UCORDONANT.PAINTOUTFRCT0001[NVK.BWR.VCRDNICS, int, int, int, int, int, int, int,	PC Address' 11	0×00000000	
0.	50.70 com.luciad.view.gxy.av.b(java.awt.Graphics)	Rita: 420	4667205	
0.	50.70 cost, Luciad.view.gxy.mw.paintComponent (java.awt.Graphice)	Reurse Eller		A A Barrow Brack and
0.	50.70 cos.luciad.view.gxy.aw.paint6XIView(java.awt.Graphics)	Source File.	CT NO	K/ADBCEBCCIEBUR. Jav
0.	56.59 com.luciad.vime.gxy.al.a(boolean, boolean)	Opject Pile:		CTOCK ADDIESOUTESC
0.01	56.59 com.luciad.view.gxy.al.a(com.luciad.view.gxy.alla, boolean)	Load Object: Od	:Lasses>	
ο.	56.59 com.luciad.view.gxy.bf.a(com.luciad.view.gxy.aIfa, boolean, boolean)	Mangled Name:		track.
0.	56.59 com.luciad.view.gxy.bf.b(com.luciad.view.gxy.mIfm, boolean)	Aliases:		
0.	55.97 com.luciad.view.gxy.al.a(com.luciad.view.gxy.0, java.mat.Rectangle, com.luciad.view.gxy.al%a, double, double)		Metrics for Selected Object	Comp.
0.	50.88 com.luciad.view.gxy.eV.a(jeva.aut.Graphics, jeva.util.Vector, ist, double, double)		🛱 Exclusive	🗮 Inclusive
0.06	50.88 com.luciad.view.gxy.aV.b(java.aut.Graphics, java.util.Vector, ist, double, double)	User CPU:	0. (0. %)	10.998 (13.75)
0.01	48.12 com.luciad.view.gxy.em.paint(java.swt.Graphics, int, com.luciad.view.gxy.6)	Wall:	0. (0. %)	0. (0. )
0.01	47.92 com.luciad.view.gxy.aa.paintFromOverlaps(java.aut.Graphics, int, com.luciad.model.c, com.luciad.view.gxy.m)	Total Thread:	0. (0. %)	24.527 [ 0.58%
0.	33:73 javax.swing.RepaintManagerTFaintManager.paint(javax.swing.JComponent, javax.swing.JComponent, java.swt.Graphics, int, int, int, int)	System CPU:	0. (0. %)	0.040 ( 1.72)
0.	33.73 javax.swing.RepaintManagerfPaintManager.paintDoubleBuffered(javax.swing.JComponent, java.swt.Image, java.swt.Graphics, int, int, int)	Wait CPU:	0. (0. 1)	12.339 (15.17)
0.01	30.85 java.lang.Thread.run()	User Lock:	0. (0. 1)	1.151 [ 0.009
0.10	27.01 common.ppi.model.h.appiyCelsteract2DBounds(ros.luriad.shape.h. com.luriad.util.s)	Text Page Fault	0. 10. 11	0. 10. 1
1.20	26.37 Common.ppi.model.8.a(int, com.luciad.util.s)	Data Page Fault	0. 10. 11	0, 10, 1
0.46	26.02 com.lucisd.view.gxy.astc.applyCh(java.lang.Gbject)	Other Walt	0. 1.0. 11	0. 1.0. 1
0.	25.99 com.luciad.view.gxy.asynchronous.g.paint(java.awf.oraphics. int. com.luciad.view.gxy.9)	offici time	0. ( 0. N	
U.,	25.12 common.ppl.iayer.s.paintreom/veriags(java.aws.orapiles, int, com.luciad.model.c, com.luciad.view,gxy.m)			
0.	common.ppi.layer.tta.appi/ointerectibounds/com.luind.shape.h, Doolean, com.luind.util.s, double, doubl			
W-	<pre>24.55 mm.com.com.com.com.com.com.com.com.com.c</pre>			
0.04	22.51 JWW.WILLOOKUIERC.INCOMPUTENCIANCOUCHERCE.EUN) 21.50 June will consumer "Availant Discontactive Tack (une land boundel)			
0.04	21.90 JWR. OLL/CONCURRENT INCOMPTON CONCURRENT FUNDAME,			
0.36	21.13 Common. ntl. Javre, Eth. avai(") javr. Jang. (bisch)			
0.05	20.81 two.util.computert.FutureTask.tin()			
0.03	20.73 hwws.util.concurrent.fucureTaskStync.innerBus()			
0.04	10.28 componetwork, dfa.run()			
0.46	19.22 plots.ppl.painters.d.paint(fava.aut.Graphics.int.com.luciad.view.gwy.m)			
0.61	16.28			
0.76	15.70 com, luciad.transformation.TLcdGeodetic2Grid.a(com.luciad.shape.y, com.luciad.shape.shapelD.o)			
0.30	15.54 com, luciad, transformation. ThedDefaultHoodilTHeoridTransformation.a(com, luciad, shape.y, com, luciad, shape.shapelD.o)			
1.94	15.06 com, luciad, internal.transformation.i.a(com, luciad, shape.y, com, luciad, shape.shapelb.o)			
0.08	14.09 town of the second			
0.	13.75 and 1. AbstractTrack.updateProperties(jawa.lang.Cbject)			
0.	12.24 org.nato.unclassified.mics.mase.plots.ppi.model.q.applyOnInteract2DBounds(com.luciad.shape.h, boolean, com.luciad.util.s, double, double, double, do			
1.63	11.61 com.luciad.projection.TLoLLambertConformal.a(com.luciad.shape.y, com.luciad.geodesy.d, com.luciad.shape.shapelD.o)			
0.01	10.51 awas, land.String.foreat/hava.land.String. hava.land.Chiect[1]			

Analysing heap dumps and optimising performance

#### Details of Hotspot 2



Analysing heap dumps and optimising performance

## Details of Hotspot 2 (cont.)

	<ul> <li>Cracto Satura Stutios Participante Many Set Statics and gravity in</li></ul>			- 07		
le Yiew						
100	Kolasi ⊂ Girie Vigw Mode User Y Fing Text ⊡ Ki Ki		Discrete internet			
unctions	Callers-Calless Call Tree Source Disassembly Timeline Experiments	Summary Timelin	ne Details			
部 User	Name	Selected Object				
CPU		Name: 34	va.lang.String.format(java.l	ang.String, java.las		
100,00	track, properties, MassAusTropertiesProvider, gstPropertyValue (java, lang, Chiect, Company, Component, att -	PC Address: 11	10x000039##			
		Size: 16				
		Source File: 30	va/lang/String.jeva			
t i		Object File: /t	mp/snalyzer.String.class.128	)		
1		Load Object	classes>			
		Mangled Name: 34	va.lang.String.format			
		Aliases:				
L			Metrics for Selected Object	2		
	Add Remove Bet Head Set Center Det Tall		Exclusive	Rinclusive		
		User CPU:	0.010 ( 0.014)	0.426 (10.534)		
0.03	B. Bodes, track. properties.MaseAuxPropertiesProvider.getPropertyValue(accalib.dataelesents.b.)	Walk	0. (0. 1)	0, (0, 1)		
		Total Thread	0.010 ( 0.008)	17,792 ( 0,424)		
		System CPU:	0. (0. 1)	0.030 ( 1.29%)		
		Wall CPU	0. (0. 1)	0.466 (10.414)		
		liner Lock:	0 (0 1)	0.071 / 0.068)		
		Taxt Bana Exult	0 10 11	0. (0.4)		
		Cata Page Fault	0. (0. 1)	0. 10. 4		
÷		Other Welt	0. 10. 17	0. 10. 4		
86.90	java.lang.ttring.format(java.lang.ttring.java.lang.cbject())	ordiet ware	- 1.0. H	0. 10. 1		
1,86	<pre>java.lang.StringBuilder.append(java.lang.Cbject)</pre>					
1.65	java.lang.String.valueOf(java.lang.Grject)					
1,45	accalib.segis.s.e()					
1.45	java.lang.StringBuilder.toString[]					
1.24	<pre>java.lang.BtringBuilder.clnit&gt;()</pre>					
1.24	config.lbray.wtl.d.get(int)					
0.03	Java lang Stringtulast java ang java ang ittang					
0.41	java ing stringeuiser appending)					
0.41	Controls Develop provide an Assachultronget (antropy) the January to date a least to be					
0.31	annalib.seenis.d.s()					
0.21	accilib.dataslements.b.c()					
0.31	java.lang.Boolean.valueOf(Doolean)					
0.21	java.lang.Byte.valueOf(byte)					
0.21	java.util.TreeMspiFrivsteEntryiterator.mextEntry()					
0.11	rack.properties.MuseAuxPropertiesFrowider.a(acoslib.dataelements.b)					
0.	java.lang.long.toString()					
0.	<pre>java.util.TreeMapfValueIterator.sext()</pre>					

Analysing heap dumps and optimising performance

### Details of Hotspot 2 (cont.)

ile Yiew Help S S S S A C J F & View Mode User . Find Text - 2 2 Functions Callers-Callees Call Tree Source Disassembly Timeline Experiments 73 Summary Timeline Details Selected Object 22 User Name CPU Name: jave.lang.String.formet(jave.lang.String, jave.las ¥ (%) PC Address: 11:0x00003977 50.00 track.properties.MaseAuxPropertiesProvider.getPropertyValue(accslib.dataelesents.b. mmon.pr Size: 16 15.91 con.util.m.e(double, jeve.lang.String) Source File: java/lang/ltring.java 12.59 amon.util.m.b(double, java.lang.String) 12.35 track.properties. TrackPropertiesProvider.getPropertyValue trock.link16. Object File: /tep/analyzer.String.class.1283 7.96 .plots.Plotter.a .plots.messages.gfa, int, java.lang.ftring) Load Object (jeinsses) 0.83 tdllib.link16.datafields.Mode\_III\_Code.b(int) Mangled Name: java.lang.String.format tdllib.link16.datafields.Node\_II\_Code.h(int) 0.83 Aliases: common.ppi.ststusher.b.run() 0.13 Metrics for Selected Object: R Exclusive Inclusive . User CPU: 0.010 ( 0.01%) 0.426 (10.53%) 1 0. 93 (0, 1) 0. 0. Walt Total Thread: 0.010 ( 0.00%) 17.792 ( 0.428) 4 D Add Bemove SetHand Set Center Set Tail System CPU: 1 0. 13 0.030 ( 1.29%) 0. 0.11 java.lang.String.format(java.lang.String, java.lang.Object[] Wait CPU: (0. 1) 8.466 (10.414) 0. 0.871 ( 0.06%) User Lock: 0. 10. 1) 10. 11 (0. 1) Text Page Fault 0. ٥. 10. 12 1 0. 17 **Data Page Fault** 0. 0. Other Wait: 0. 10. 11 0. 1.0. 10 . 91.21 java.util.Formatter.format(java.lang.String, java.lang.Object[]) 7.24 java.util.Formatter.<init>|) 1.43 java.util.Formatter.toString()

#### Analysing heap dumps and optimising performance

## Identification of Hotspot 2 (cont.)

unctions	Callers-Callees Call Tree Source Disassembly Timeline Experiments	- Q2 ;	Summary Tim	teline Details	ř.				
R User	EUser Name				Selecte	d Obje	et:		
CPU	CPU DIANA		Name	java.lang.	String.	format	tijeve.l	ang.Steing	. jeve.l
0.	* (W)	-	PC Address	1110+000003	977				
0.	14.65 point inver.81a.applyContreractDDoundsloop.intiad.shape.h. bonlean.com.iudiad.util.s. double. double. double.	doub	Size	16					
D.	14.65 December 2010 and 10 appl.model.m.applyonInteractIBBounds/com.luciad.shape.h. boolsan. com.luciad.util.s. double, double	do	Source File	java/lang/	String.	389/8			
0.	22.34 inva.util.concurrent.ThreadPoolExecutorPacker.run()		Object File	/tno/analy	zer.Dtr	160.0	1449.120	1	
0.04	11.68 java.util.concurrent.ThreadFoolExecutorFWorker.runTask(java.lang.Funnable)		Load Object	Cirlanser)					
0.04	21.41 java.util.concurrent.ExecutorsfRunnableAdapter.call()		Manaled Name	taux lana	Thursday.	format			
0.36	21.13 ppi.layer.Hib.applyOn(java.lang.Object)		Alizza	Jararandi	a canada	a coa arres			
0.05	10.81 javw.util.concurrent.FutureTash.run()			1		576377	1200	_	_
0.03	20.73 java.util.concurrent.FutureTaskiSync.innerFun()			Metri	cs for S4	electer	d Object:		
0.04	20.20 .network.dfa.cun()		1120022	Jul Int	Exclusiv	10		Inclusi	IVe
0.46	19.22 plots.ppi.painters.d.paint (java.met.Oraphics, int. com.luciad.view.gvy.m)		User CPU	r	0.010	{ 0.0	18)	8.426	(10.534
0,61	16.28 ppi.util.d.getAWTLocation(com.luciad.view.gry.m. com.luciad.shape.y)		Wal	E .	0.	( 0,	8)	0.	(0. )
0,76	15.70 com.luciad.transformation.TLcdGeodetic2Grid.s(com.luciad.shape.y, com.luciad.shape.shape3D.o)		Total Thread	t	0.010	( 0.00	0.83	17.792	( 0.424
0.00	15.54 com.luciad.transformation.TLcdDefaultModelNTMorldTransformation.a(com.luciad.shape.y, com.luciad.shape.shapelD.o)		System CPL	t,	0.	( 0.	83	0.030	1 1 1.294
1.94	15.86 com.luciad.internal.transfocuation.i.alcom.luciad.sbape.y, com.luciad.shape.sbapelD.o)		Wajt CPU	J:	0.	( 0.	b)	0.455	(10.414
0.08	14.00 track.AbstractTrack.a(jwva.lang.Object, jwva.lang.Class,	#.Pr	User Loc	c	0.	( 0.	1)	0.871	( 0.061
Ο.	13.15		Text Page Faul	t	0.	(.0.	10	0.	(0. 1
0.	12.14 poly plots.ppi.model.q.applyOnInteract25Bounds(com.luciad.shape.h, boolean, com.luciad.util.s, double, d	, do	Data Page Faul		0.	( 0.	83	0.	10. 1
1.63	11.11 com.luciad.projection. TLodLambertConformal.a(com.luciad.shape.y, com.luciad.geodesy.d, com.luciad.shape.shapelD.o)		Other Wal		0.	10.	42	<u>0</u> .	1.0. 1
0.01	10.13 java.lang.String.formut(java.lang.String, java.lang.Object[])					-	~ 1		1000
D.	10.11 (marging and marging plots.model.m.removeRxpiredItems()								
0.	10.00 plots.ppl.model.q.removeElements(jeva.util.Vector, int)								
0.01	9.15 second and a second s								
8.91	9-16 _pave.util.concurrent ConcurrentLinkedQueue.sise()								
10.03	9.61 java.util.Formatter.format(java.lang.String, java.lang.Object[])								
0.	9.57 plots.ppi.model.m.addElement (prest/ plots.AbstractPlot, int)								

18

## **CPU Performance Tuning**

## Performance before the changes

#### CPU: 60%



15/9/2012

20

## Performance after the changes CPU: 50%



15/9/2012

21

## Profiling comparison 1

1226920	Gre v	iew Mode U	ser 💌		Find Text:	7	<u>R</u> <u>R</u>
nctions Callers-C	allees Call Tr	e Source	Disassembly	Timeline	Experiments		
inBranch-Tracks.er	NewQueue N	ewMap-Tracks er CPU	er MainBran	ch-Tracks er ier CPU	NewQueue N	NewMap-Tracks er Iser CPU	Name
♡ (%)		(%)		(%)		(%)	
100.00	1	00.00	1	00.00		100.00	<total></total>
8.92		8.86		8.92		8.86	<jvm-system></jvm-system>
8.91		0.	- 11 D	9.76		0.	java.util.concurrent.ConcurrentLinkedQueue.size()
5.64		8.20		5.64		8.20	mlib_ImageCopy_na
3.92		2.94		8.04		6.81	java.lang.StrictMath.pow(double, double)
3.80		3.74		3.92		3.79	ieee754_pow
2.01		2.54		2.01		2.54	mlib_v_ImageClear_S32_1
1.94		2.30		15.06		14.24	com.luciad.internal.transformation.i.a(com.luciad.shape.y, com.luc
1.63		1.78		11.61		10.48	com.luciad.projection.TLcdLambertConformal.a(com.luciad.shape.y, co
1.30		1.27		1.30		1.27	plots.AbstractPlot.getBounds()
1.28		1.14		4.63		5.44	sun.java2d.loops.DrawLine.DrawLine(sun.java2d.SunGraphics2D, sun.java2d.SunGraphics2D, sun.javaSunGraphics2D, sun.javaSunGr
1.20		1.01		26.57		26.50	common.ppi.model.h.a(int, com.luciad.ut.
0.98		0.93		0.98		0.93	java.lang.Object.clone()
0.95		0.71		0.95		0.71	com.luciad.util.Y.d(double)
0.91		0.87		0.98		0.94	common.ppi.layer.E%c.accept(java.lang.G
0.88		1.06		0.88		1.05	java.lang.String.charAt(int)
0.84		0.74		0.86		0.75	com.luciad.view.gxy.az\$a.a(double, double, java.awt.Point)
0.83		0.04		0.86		0.06	java.util.concurrent.ConcurrentLinkedQueue.succ(java.util.concurrent
0.76		0.93		15.70		15.01	com.luciad.transformation.TLcdGeodetic2Grid.a(com.luciad.shape.y, o
0.75		1.07		0.75		1.07	<no callstack="" java="" recorded=""></no>
0.75		0.98		0.75		0.98	com.luciad.shape.shape3D.n.getSinY()
0.73		0.68		0.73		0.72	java.lang.Integer.equals(java.lang.Object)
0.68		0.52		0.75		0.56	java.util.concurrent.ConcurrentLinkedQueue\$Itr.advance()
0.66		0.77		3.35		4.29	Java_sun_java2d_loops_DrawLine_DrawLine
0.63		0.65		1.39		1.53	com.luciad.view.s.a(java.util.List, java.lang.Object, int, int)

### Profiling comparison 2

ile Yiew H		G Vigw Mode	User 💌	_	Find Text:			
Functions	Callers-Callees	Call Tree Source	ce Disassem	bly Timeline	Experiments D ; Sum	mary Timelin	e Details	
MamBranch	Stringittis.er	MainBranch.er	StringLitils er	Name			Selected Object	
RUser CPU	AUser CPU	MUSER CPU	盖User CPU			Name: Hul	tiple Selection	
(%)	(%)	♥ (%)	(%)		8	PC Address:		
0.06	0.	47.01	44.22	com.luciad.vi	ew.gxy.as.paint(java.awt.Gcaphics, int, com.luciad.view.gxy.6)	Size: #58	9934606	
0.02	ο.	46.75	44.07	com.luciad.vie	ew.gxy.ma.paintFromOverlaps(java.mot.Graphics, int, com.luciad.model.c, com.luciad.view.gxy.m) 🚽 🍙	Source File:		
0.	0.	31.62	25,59	java.lang.Thr	end. run ()	Object File:		
0.14	0.27	25.94	25.05	6	common.ppi.model.h.applyOnInteract298ounds(com.luciad.shape.h, com.luciad.util.s)	and Object cir	lasses)	
1.36	0.93	25.66	24.76		.common.ppi.model.h.s(int, com.luciad.util.s)	aniad Name:	empera-	
0.	0.	25.56	23.79	com.luciad.vid	m.gxy.asynchronous.g.paint(java.set.Graphics, int, com.luciad.visu.gxy.0)	igied Name.		
0.02	0.	25.20	23.59	وكالأسر عدم	common.ppi.layer.W.paintfromOverlaps(java.aut.Graphics, int, com.luciad.model.c, com.	Bilases:		
0.46	0.27	24.68	27.30	com.luciad.vi	nw.gxy.male.applyOn(java.lang.Object)		Metrics for Selected Object	_
0.02	Ο.	24.20	22.64	( <b>1</b>	.comeon.ppi.layer.Efs.applyOnInteract2DBounds(com.luciad.shape.h, boolean, com.luciad.		R Exclusive	a Inclusive
0.	0.	24.10	22.64		.plots.ppi.model.m.applyOnInteractIDBounds(com.luciad.shaps.h, boolean, com.lucia	User CPU:	0.030 ( 0.10%)	4.563 (15.07
0.02	0.	22.15	14.66	java.util.com	current.ThreadFoolExecutorfWorker.run()	Wall;	0. (0. *)	0, 10,
0.06	0.05	21.73	14.01	java.util.com	rurrent.ThreadPoolEsecutor(Worker.cunTask(java.lang.Punnable) Tg	otal Thread:	0.080 ( 0.00%)	12.029 ( 0.56
0.05	ο.	21.49	13.01	java.util.com	current.ExecutorsfRunnableAdapter.csl1() S	System CPU:	0. (0. 1)	0.020 ( 1.01
0.06	0.	20.91	13.44	java.util.com	surrent.FutureTask\$Sync.innerFun()	Wait CPU:	0.050 ( 0.15%)	6.595 (20.27
0.02	0.02	20.91	13.47	java.util.com	current.FutureTask.run()	User Lock:	0, (0, 1)	1.651 ( 0.14
0.40	0.49	20.53	21.10	-	common.ppi.layer.E(b.applyOn(java.lang.Object)	Pane Exult	0 (0 1)	0. 10.
0.12	0.05	20.05	12.34		.common.network.dfa.run()	Reas Frank	0. 10. 1	0. 10.
0.32	0.46	18.62	19.48		plots.ppi.psinters.d.paint(java.mst.Graphics, int, com.luciad.view.gsy.m)	a rage rause	0. (0. 1)	0. 10.
0.62	0.64	15.34	16.74		.ppi.util.d.getAWTLocation(com.luciad.vise.gxy.m, com.luciad.shape.y)	Other wait:	0. (0. %)	0. 10.
0.94	1.32	14.51	16.50	com.luciad.tr	msformation.WicdGeodetic2Grid.a(com.luciad.shape.y, com.luciad.shape.shape2D.o)			
0.32	0.24	14.21	16.04	com.luciad.tr	ansformation.TLcdDefmaltModelXTMorIdTransformation.s(com.luciad.shape.y, com.luciad.shape.shape20.0)			
1.02	2.37	13.71	15.40	com.lucied.in	eroal.transformation.i.s(com.luciad.shape.y, com.luciad.shape.shape20.0)			
0.14	0.05	12.57	5.60	a second second	track.AbstractTrack.s(jwvs.lang.Chject, jwvs.lang.Class, org.nato.uncl			
0.	0,02	13.11	5.10	Constanting of the local division of the loc	track.AbstractTrack.updateProperties(java.lang.Object)			
0.	0.	11.21	17.93	com.luciad.vi	w.gxy.al.b(com.luciad.view.gxy.0, java.met.Rectangle, com.luciad.view.gxy.alta, double, double)			
1,72	2.40	10.67	12.10	com.lucisd.pr	ojection.TLodLasbertConformal.a(com.luciad.shape.y, com.luciad.geodesy.d, com.luciad.shape.shapelD.d			
0.04	0.	9.64	0,37	Java.lang.ptr.	ing.format(java.lang.String, java.lang.Object[))			
0.04	0.05	9.50	6.60	(Contraction)	mice.common.picture.mervices.lessageRandler.processNessage(accslib.a)			
9.40	14.40	9.40	14.40	<jvh-system></jvh-system>				

15/9/2012

## Memory Performance Profiling/Tuning

Goal is not to improve the CPU usage but to mitigate the impact of the GC on the real-time behaviour of the application

Generational Spaces & Garbage Collectors									
Eden	S1	S2	Tenured	Permane	nt				
New Gener	eration ration GCs	/	Old Generation Old Generation GCs						
Copying collector (< . -XX:+UseSerialGC	Java 5)		Mark-Sweep collector (< Java 5) -XX:+UseSerialGC						
Parallel copying colle -XX:+UseParNewGC	ctor (≥ Jav	ra 5)	Parallel Scavenge MarkSweep collector -XX:+UseParallelOldGC						
Parallel scavenge col >10GB heap space) -XX:+UseParallelGC	lector (≥ Ja	ava 5,	Concurrent Mark Sweep collector (≥ Java 6) -XX:+UseConcMarkSweepGC						
G1 young generation -XX:+UseG1GC 15/9/2012 Anal	(≥ Java 7) ysing hear	o dumps a	G1 mixed genera -XX:+UseG1GC nd optimising pe	tion (≥ Java 7) rformance	25				

## **Recommended** Settings

http://randomlyrr.blogspot.be/2012/03/java-tuning-in-nutshell-part-1.html

-Xmx = -Xms

- > -XX:PermSize = -XX:MaxPermSize
- Use -Xmn instead of -XX:NewSize and -XX:MaxNewSize
- Disable adaptive sizing of generations -XX:-UseAdaptiveSizePolicy if -XX:SurvivorRatio is specified
- Use -XX:+UseConcMarkSweepGC
- > 40<-XX:CMSInitiatingOccupancyFraction<70
  and always use</pre>

-XX:+UseCMSInitiatingOccupancyOnly with it



3gc_SurvivorRatio3_m	256_mx768_Xincgc_NEW_	Versio <mark>n.vg</mark> c			0	j x
File View Help						
5 🖹 2						
Summary Heap U	Isage After GC DL	ration Cumula	tive Allocation Creation F	Rate User-Defined N	Aultiple User-Defined	
Heap Capacity			196 - Carlos Car			
	Eden	Surv	vor Old	Perm	Total	
Initial Capacity	N/A	N/A	N/A	N/A	252,812 (MB)	^
Final Capacity	N/A	N/A	N/A	N/A	755,25 (MB)	E
Peak Capacity	N/A	N/A	N/A	N/A	755,25 (MB)	
Peak Usage of Capacity GC Activity Summary	N/A	N/A	N/A	N/A	98.852%	~
	Last occurrence (s)	Count	Average interval (s)	Average duration (s)	Average rate of collection	on
CMS	59,497.293 (s)	225	265.605 (s)	0.012 (s)	N/A	^
Scavenge	59,496.296 (s)	61,914	0.961 (s)	0.011 (s)	3,214 (GB/s)	=
Old full (with Perm Gen)	45,789.978 (s)	3	15,271.854 (s)	1.165 (s)	401,407 (MB/s)	
Other full GC Overall Statistics	59,462.632 (s)	9	7,431.736 (s)	0.779 (s)	111,082 (MB/s)	<b>~</b>
Name		Value	Name		Value	
Duration of the measurement	6	0,687.209 (s)	Time spent in GC		667.204 (s)	~
Total bytes allocated	2	,053( TB)	Percentage of time	in GC	1.099%	
Number of GC events	6	2,151	Time spent in Full G	ic .	10.502 (s)	
Average bytes allocated per GC	3	4,631 (MB)	Percentage of time	in Full GC	0.017%	E
Avg. ideal allocation rate	3	5,861 (MB/s)	Avg. allocation rate		35,466 (MB/s)	
Residual bytes	2	77,839 (MB)				
1						
	Time in	GC: 1.1%	Time in Full GC: 0.02%	Full GC to GC: 1.57%		

15/9/2012









15/9/2012 Analysing heap dumps and optimising performance

32

# Thread Profiling/Tuning ?

## References

- Hunt C. & John B. (2011), Java Performance, Prentice Hall
- \* Kabutz H. (2011), Java Master's course slides
- \* <u>http://www.youtube.com/watch?v=VGQAL9aUKfs</u>
- \* <u>http://randomlyrr.blogspot.be/2012/03/java-tuning-</u> <u>in-nutshell-part-1.html</u>
- \* <u>http://www.fasterj.com/articles/oraclecollectors1.sh</u> <u>tml</u>
- \* <u>http://www.oracle.com/technetwork/java/javase/te</u> <u>ch/vmoptions-jsp-140102.html</u>

## Questions

