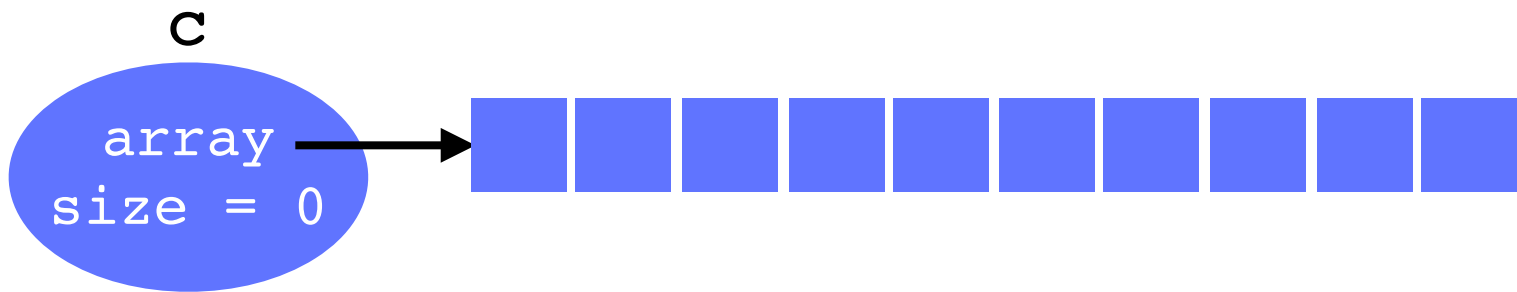


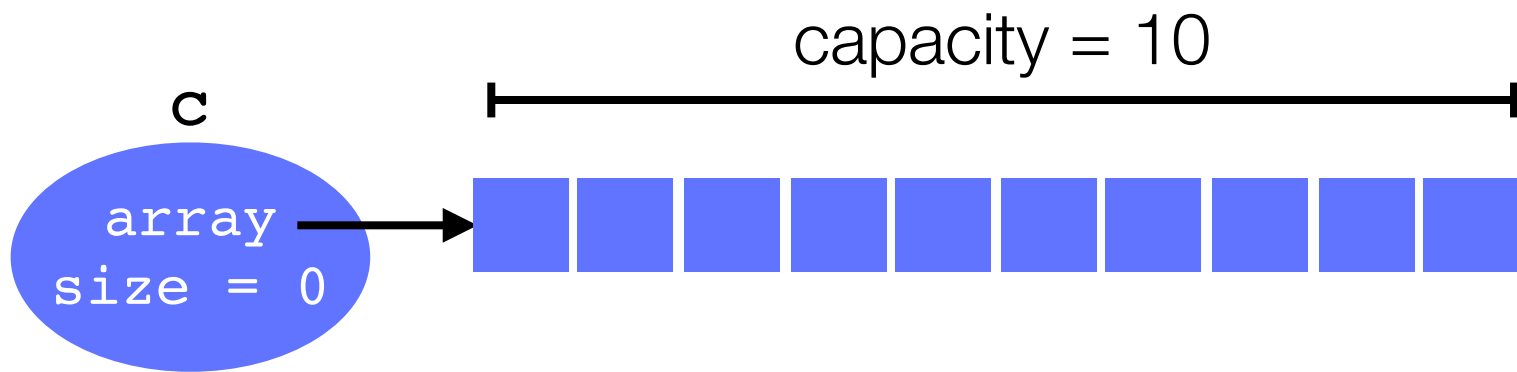
# Reducing Waste in Expandable Collections: The Pharo Case

Alexandre Bergel, Alejandro Infante, Juan Pablo  
Sandoval Alcocer

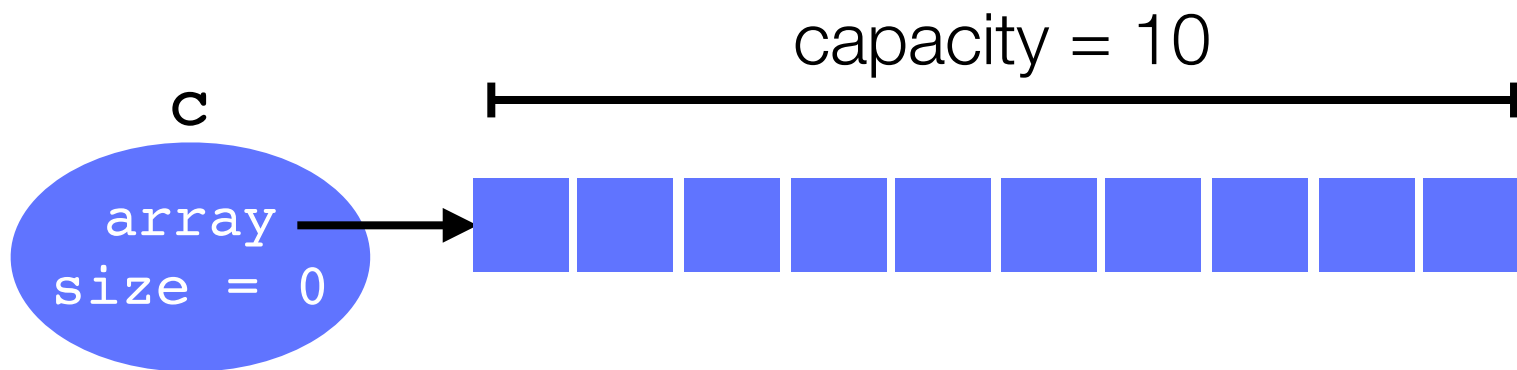
*University of Chile*



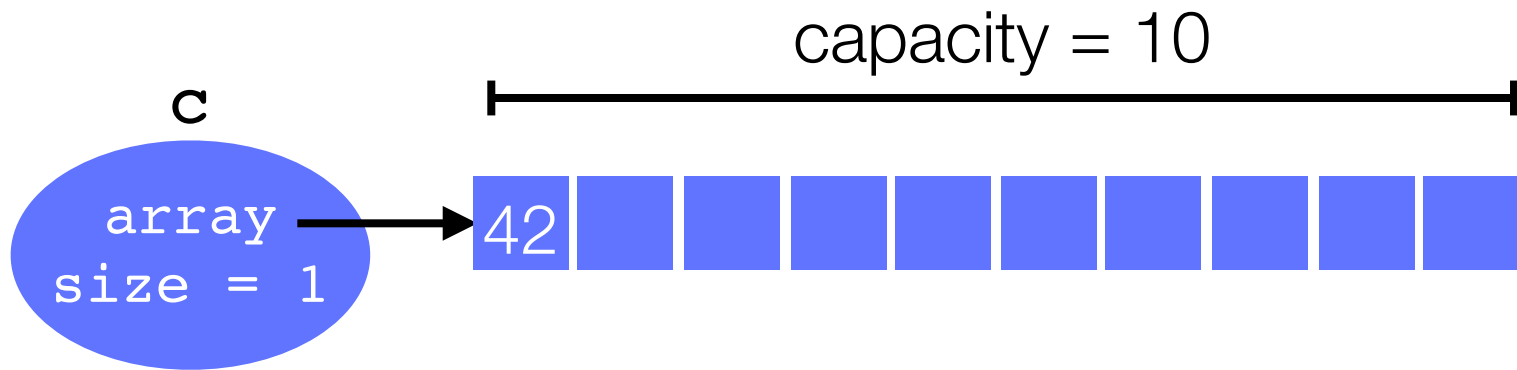
```
c := OrderedCollection new.
```



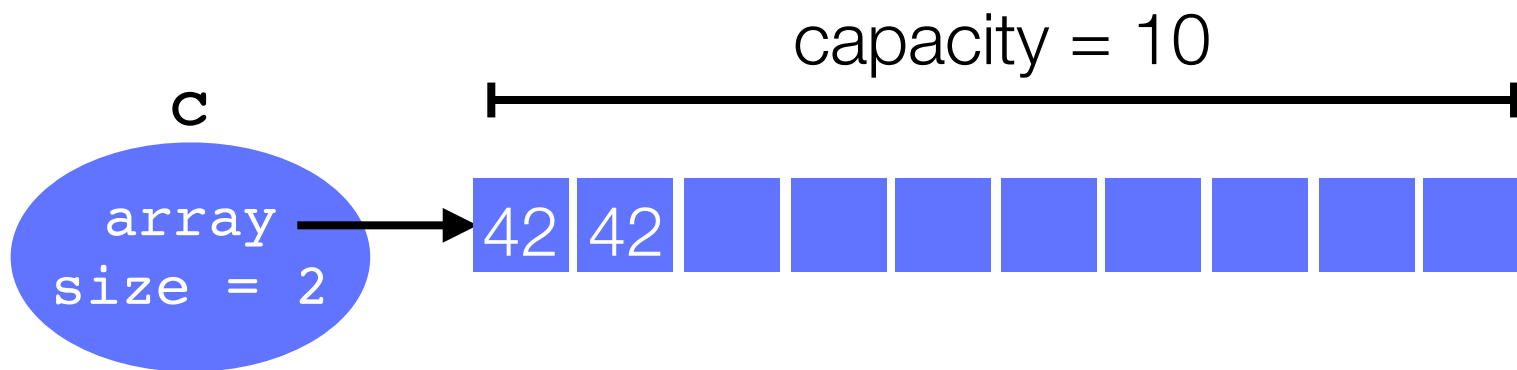
```
c := OrderedCollection new.
```



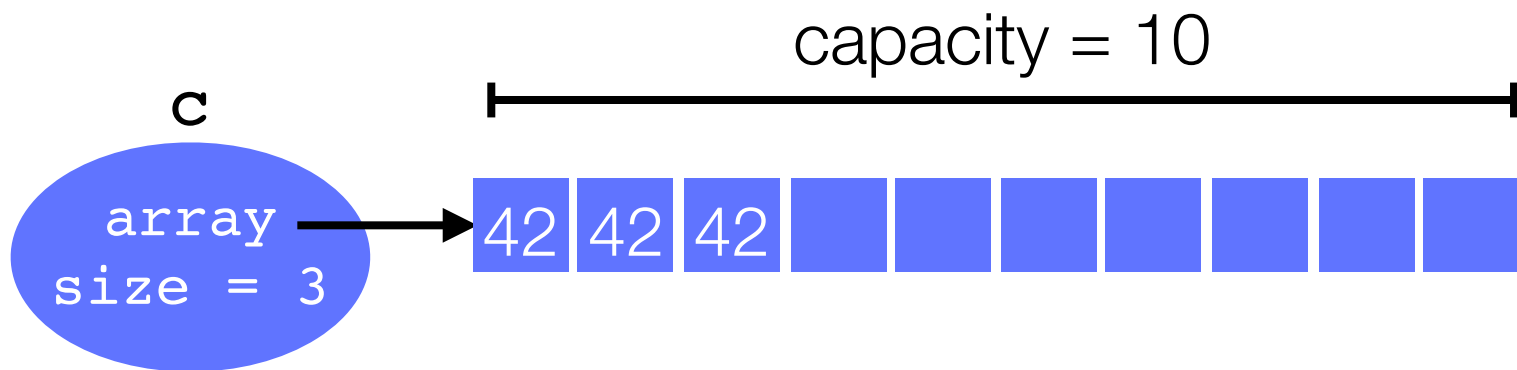
```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



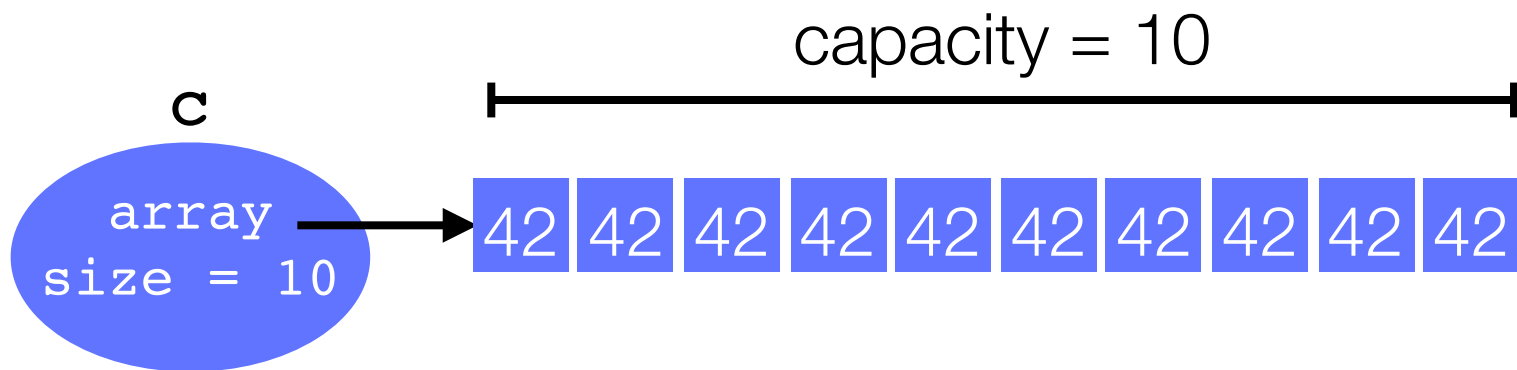
```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



```
c := new OrderedCollection.  
11 timesRepeat: [ c add: 42 ].
```

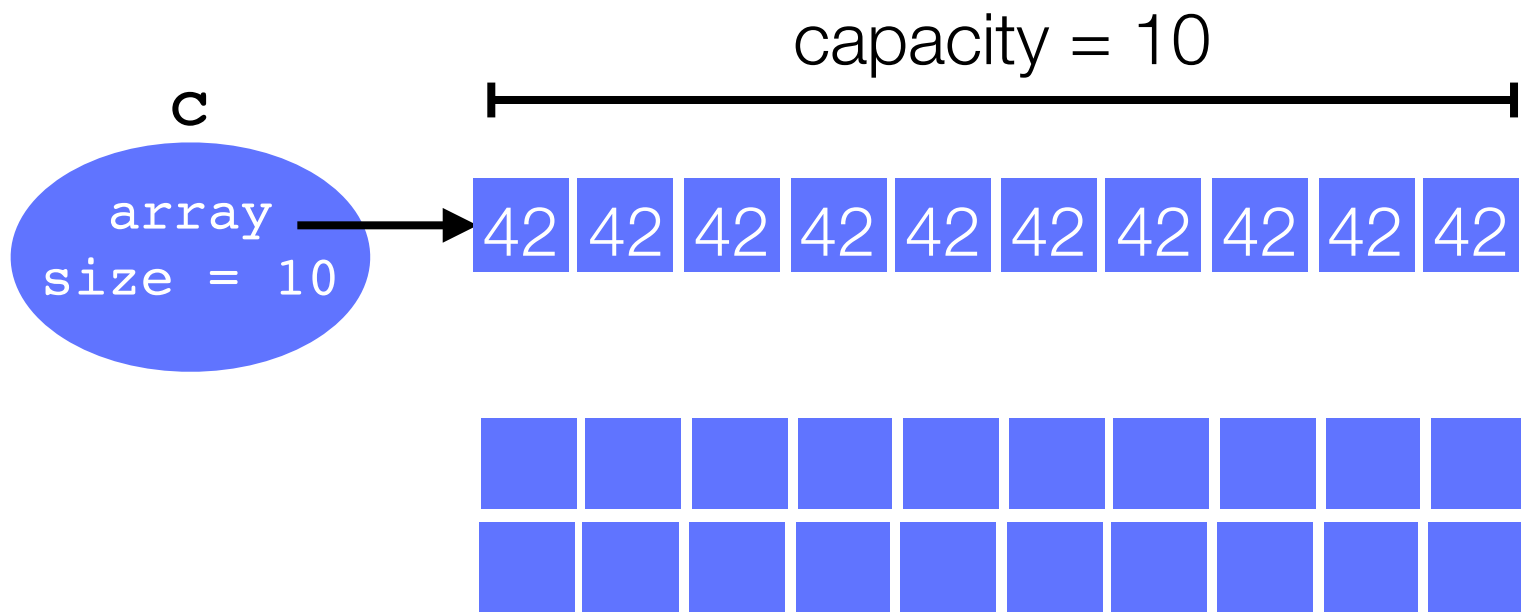


```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```

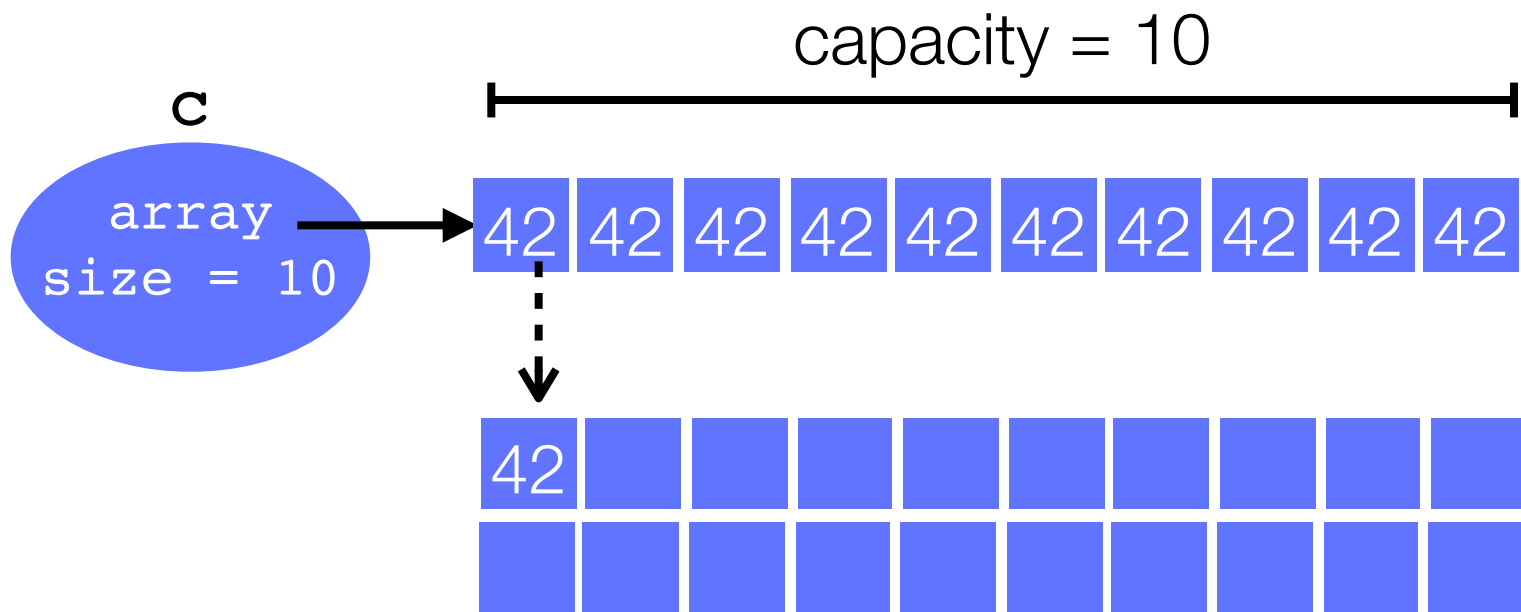


```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```

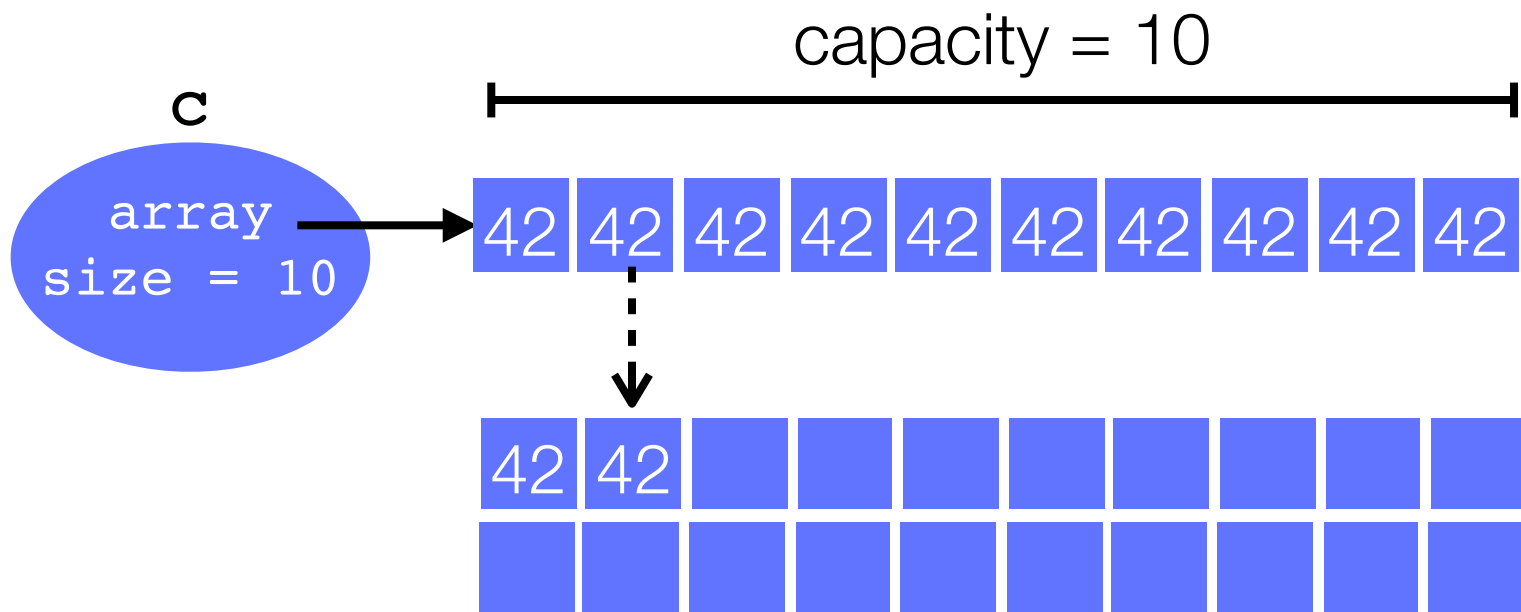




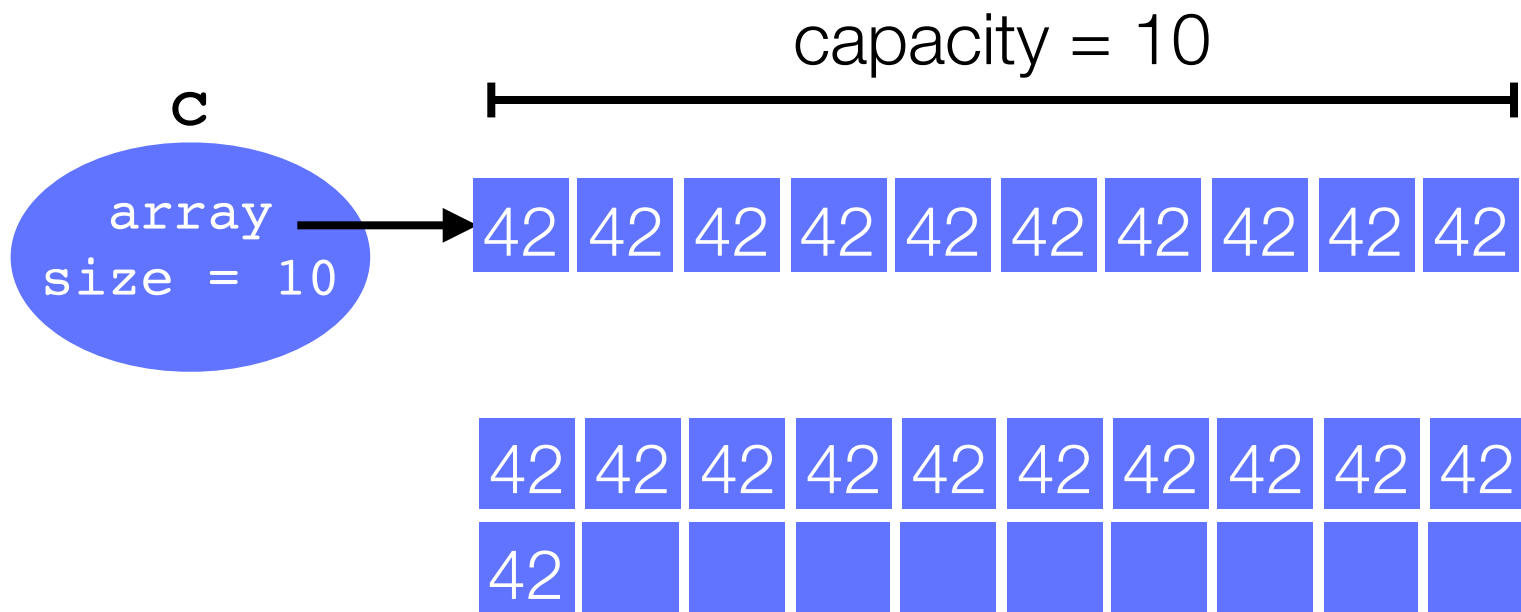
```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



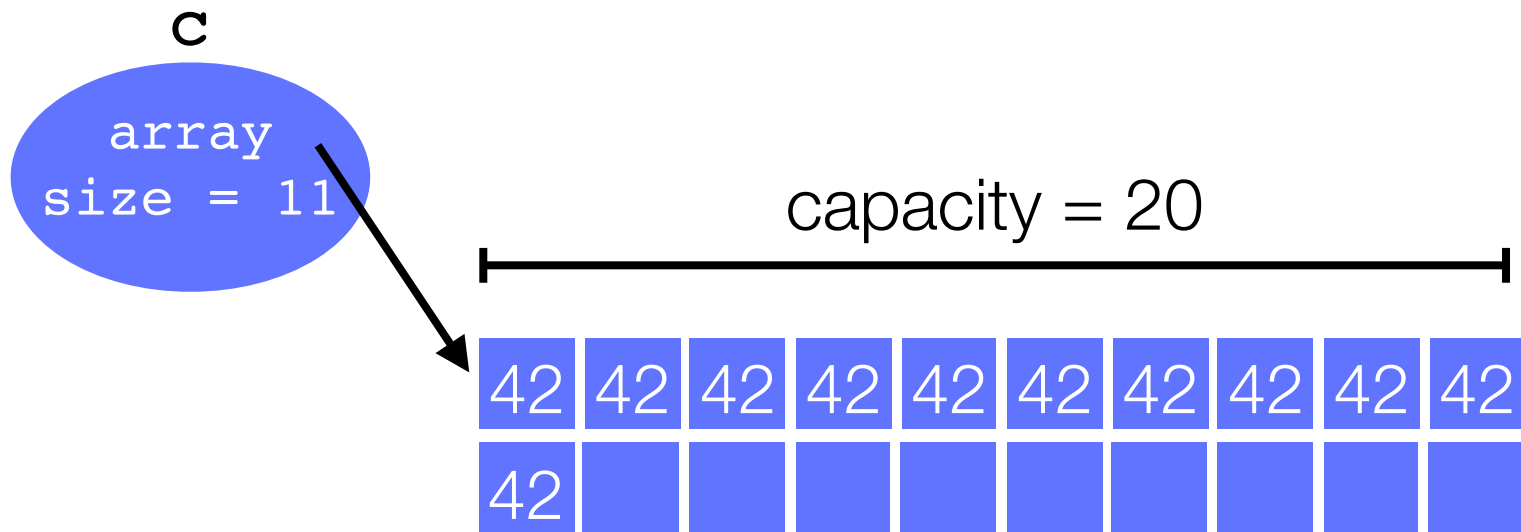
```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```

```
c := OrderedCollection new.  
[ 30000000 timesRepeat: [ c add: 42 ] ] timeToRun  
=> 3375 milliseconds
```

```
c := OrderedCollection new.  
[ 30000000 timesRepeat: [ c add: 42 ] ] timeToRun  
=> 3375 milliseconds
```

22 expansions, 164 Mb copied

```
c := OrderedCollection new.  
[ 30000000 timesRepeat: [ c add: 42 ] ] timeToRun  
=> 3375 milliseconds
```

22 expansions, 164 Mb copied

```
c := OrderedCollection new: 30000000.  
[ 30000000 timesRepeat: [ c add: 42 ] ] timeToRun  
=> 1356 milliseconds
```

0 expansion, 0 Mb copied



# Our Paper

---

Characterize the use of expandable collections

Measure the overhead associated with expandable collections

Reducing the overhead associated with expandable collections

<b>index</b>	<b>Application</b>	<b>LOC</b>	<b>#Ref</b>
*1	AST	8,091	57
2	Arki	627	6
3	Glamour	17,525	105
4	GraphET	2,757	10
5	Magritte	5,884	29
6	Manifest	2,864	11
7	NECompletion	3,446	33
*8	Nautilus	1,566	9
*9	Petit	14,919	95
10	Refactoring	21,328	125
*11	Regex	5,055	16
12	Ring	3,378	50
*13	Roassal	19,844	133
14	RoelTyper	2,003	85
15	Shout	3,290	10
16	SmallDude	3,805	66
17	Spec	10,212	37

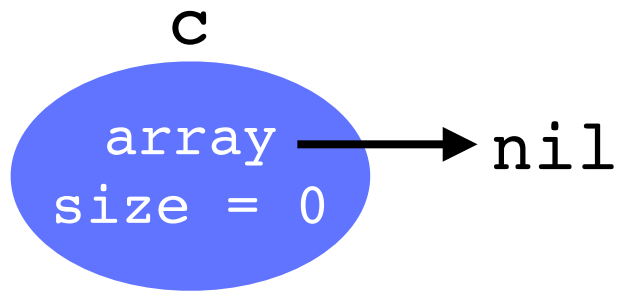
Table 2: Description of the benchmark (the #Ref column indicates the number of references to expandable collection in source code)

bench.	<i>NC</i>	<i>NNEC</i>	<i>NEC</i>	<i>NCE</i>	<i>NCB</i>	<i>NAC</i>	<i>NOSM</i>	<i>NSM</i>	<i>NAB</i>	<i>NUB</i>
1	643,603	151,665(23%)	491,938(76%)	2,636	35,940	646,239	112,308(17%)	112,439(17%)	18,819,920	17,907,204
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	78,976	34,912(44%)	44,064(55%)	322	2,272	79,298	10,604(13%)	10,681(13%)	2,374,464	2,213,384
4	1,690	512(30%)	1,178(69%)	30	1,720	1,720	245(14%)	245(14%)	65,760	55,420
5	3,191	2,009(62%)	1,182(37%)	30	1,208	3,193	200(6%)	251(7%)	77,808	67,724
6	96	44(45%)	52(54%)	0	0	96	40(41%)	40(41%)	3,740	3,464
7	612	218(35%)	394(64%)	614	2,257,960	1,231	48(7%)	48(7%)	4,657,696	837,948
8	2	0(0%)	2(100%)	0	0	2	0(0%)	0(0%)	40	40
9	158,589	58,371(36%)	100,218(63%)	5,663	280,876	164,252	57,644(36%)	57,728(36%)	6,833,532	5,534,280
10	1,432,306	14,891(1%)	1,417,415(98%)	6,074	12,258,424	1,438,380	8,248(0%)	8,344(0%)	46,958,632	34,241,204
11	6,839	2,058(30%)	4,781(69%)	1,280	78,712	6,967	471(6%)	480(7%)	291,052	256,852
12	8,363	3,530(42%)	4,833(57%)	1,103	47,852	9,466	73(0%)	73(0%)	279,236	165,460
13	108,571	57,590(53%)	50,981(46%)	1,739	369,336	109,990	8,151(7%)	8,810(8%)	5,778,016	4,845,764
14	10,305	586(5%)	9,719(94%)	145	14,044	10,448	82(0%)	110(1%)	443,828	420,352
15	20,815	14,886(71%)	5,929(28%)	255	125,900	21,070	5,736(27%)	5,740(27%)	2,692,404	1,984,240
16	766	172(22%)	594(77%)	17	496	783	123(16%)	123(16%)	126,368	122,532
17	1,203	880(73%)	323(26%)	1,512	48,384	2,715	764(63%)	764(63%)	127,356	35,988
<b>total</b>	<b>2,475,940</b>	<b>342,336(13%)</b>	<b>2,133,604(86%)</b>	<b>21,420</b>	<b>15,523,124</b>	<b>2,495,863</b>	<b>204,738(8%)</b>	<b>205,877(8%)</b>	<b>89,530,320</b>	<b>68,692,248</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	210,000	38,000(18%)	38,000(18%)	6,752,000	6,468,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,000	183,000	41,000(22%)	41,000(22%)	5,928,000	5,580,000
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,200	431,220	87,570(20%)	87,570(20%)	13,795,440	13,212,720
bN1	150	0(0%)	150(100%)	0	0	150	0(0%)	0(0%)	3,000	3,000
bN2	180	150(83%)	30(16%)	60	9,000	240	120(66%)	120(66%)	22,440	7,680
bN3	240	240(100%)	0(0%)	60	9,000	300	180(75%)	180(75%)	22,680	7,560
bPP1	90,600	46,200(50%)	44,400(49%)	5,600	436,800	96,200	46,200(50%)	46,200(50%)	4,214,400	3,033,600
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	84,600	44,800(57%)	44,800(57%)	3,790,400	2,571,200
bPP3	546,710	398,420(72%)	148,290(27%)	52,860	6,475,120	599,570	398,420(72%)	398,420(72%)	29,103,720	17,192,120
bReg1	1,000	200(20%)	800(80%)	0	0	1,000	100(10%)	100(10%)	34,400	33,600
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,080	2,162,860	10(0%)	10(0%)	86,513,920	84,799,800
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,680	1,950,010	10(0%)	10(0%)	78,001,720	76,093,720
bR1	400,011	7(0%)	400,004(99%)	46	2,631,600	400,055	0(0%)	3(0%)	17,263,480	13,023,236
bR2	2,530	1,583(62%)	947(37%)	117	15,608	2,642	289(11%)	299(11%)	141,056	99,404
bR3	79,456	53,259(67%)	26,197(32%)	4,809	686,196	84,073	13,365(16%)	13,454(16%)	7,701,916	6,045,808
<b>total</b>	<b>6,129,207</b>	<b>1,637,669(26%)</b>	<b>4,491,538(73%)</b>	<b>980,792</b>	<b>46,953,084</b>	<b>6,205,920</b>	<b>670,064(10%)</b>	<b>670,166(10%)</b>	<b>253,288,572</b>	<b>228,171,448</b>
<b>Total</b>	<b>8,605,147</b>	<b>1,980,005(23%)</b>	<b>6,625,142(76%)</b>	<b>1,002,212</b>	<b>62,476,208</b>	<b>8,701,783</b>	<b>874,802(10%)</b>	<b>876,043(10%)</b>	<b>342,818,892</b>	<b>296,863,696</b>

Table 3: Original benchmark (baseline for all the other measurements)

# Lazy Array creation

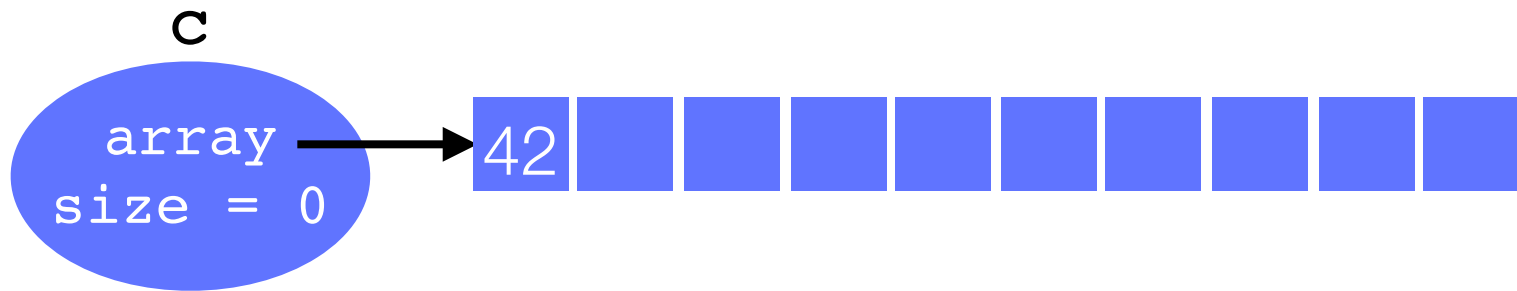
---



```
c := OrderedCollection new.
```

# Lazy Array creation

---



```
c := OrderedCollection new.  
c add: 42
```

bench.	<i>NC</i>	<i>NNEC</i>	<i>NEC</i>	<i>NCE</i>	<i>NCB</i>	<i>NAC</i>	<i>NOSM</i>	<i>NSM</i>	<i>NAB</i>	<i>NUB</i>
1	643,603	151,665(23%)	491,938(76%)	2,636	35,940	229,379	112,308(17%)	112,439(17%)	3,956,792	3,044,076
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	79,035	34,970(44%)	44,065(55%)	45	2,272	44,294	10,612(13%)	10,689(13%)	1,397,668	1,236,260
4	1,690	512(30%)	1,178(69%)	30	1,720	905	245(14%)	245(14%)	33,160	22,820
5	3,551	2,009(56%)	1,542(43%)	30	1,208	2,685	200(5%)	251(7%)	65,208	55,124
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	927	48(7%)	48(7%)	4,509,216	689,468
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,395	58,331(36%)	100,064(63%)	5,663	280,876	123,703	57,637(36%)	57,718(36%)	5,220,152	3,921,204
10	1,432,306	14,891(1%)	1,417,415(98%)	6,074	12,258,424	68,132	8,248(0%)	8,344(0%)	19,547,672	6,830,248
11	6,839	2,058(30%)	4,781(69%)	1,280	78,712	2,186	471(6%)	480(7%)	109,212	75,012
12	7,870	3,472(44%)	4,398(55%)	1,094	44,384	5,253	20(0%)	20(0%)	194,732	84,716
13	108,571	57,590(53%)	50,981(46%)	1,739	369,336	62,510	8,151(7%)	8,810(8%)	4,218,656	3,286,404
14	10,305	586(5%)	9,719(94%)	145	14,044	963	82(0%)	110(1%)	136,008	112,532
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,489	5,736(27%)	5,740(27%)	2,640,784	1,932,632
16	766	172(22%)	594(77%)	17	496	221	123(16%)	123(16%)	14,288	10,452
17	1,203	880(73%)	323(26%)	1,512	48,384	2,392	764(63%)	764(63%)	120,856	29,488
<b>total</b>	<b>2,475,704</b>	<b>342,296(13%)</b>	<b>2,133,408(86%)</b>	<b>21,134</b>	<b>15,519,656</b>	<b>562,143</b>	<b>204,686(8%)</b>	<b>205,822(8%)</b>	<b>42,168,432</b>	<b>21,334,112</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	536,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,000	53,000	41,000(22%)	41,000(22%)	1,016,000	668,000
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,200	113,040	87,570(20%)	87,570(20%)	2,389,680	1,806,960
bN1	150	0(0%)	150(100%)	0	0	0	0(0%)	0(0%)	0	0
bN2	180	150(83%)	30(16%)	60	9,000	210	120(66%)	120(66%)	21,840	7,080
bN3	240	240(100%)	0(0%)	60	9,000	300	180(75%)	180(75%)	22,680	7,560
bPP1	90,600	46,200(50%)	44,400(49%)	5,600	436,800	78,000	46,200(50%)	46,200(50%)	3,490,400	2,309,600
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	70,200	44,800(57%)	44,800(57%)	3,218,400	1,999,200
bPP3	546,710	398,420(72%)	148,290(27%)	52,860	6,475,120	543,770	398,420(72%)	398,420(72%)	26,952,320	15,040,720
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,080	428,000	10(0%)	10(0%)	17,120,000	15,405,880
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,680	476,070	10(0%)	10(0%)	19,044,600	17,136,600
bR1	400,011	7(0%)	400,004(99%)	46	2,631,600	52	0(0%)	3(0%)	5,263,360	1,223,116
bR2	2,422	1,583(65%)	839(34%)	117	15,608	1,698	289(11%)	299(12%)	109,356	67,704
bR3	78,145	53,259(68%)	24,886(31%)	4,809	686,196	63,400	13,365(17%)	13,454(17%)	7,034,296	5,378,188
<b>total</b>	<b>6,127,788</b>	<b>1,637,669(26%)</b>	<b>4,490,119(73%)</b>	<b>980,792</b>	<b>46,953,084</b>	<b>1,874,940</b>	<b>670,064(10%)</b>	<b>670,166(10%)</b>	<b>86,510,132</b>	<b>61,393,008</b>
<b>Total</b>	<b>8,603,492</b>	<b>1,979,965(23%)</b>	<b>6,623,527(76%)</b>	<b>1,001,926</b>	<b>62,472,740</b>	<b>2,437,083</b>	<b>874,750(10%)</b>	<b>875,988(10%)</b>	<b>128,678,564</b>	<b>82,727,120</b>

Table 4: Lazy internal array creation

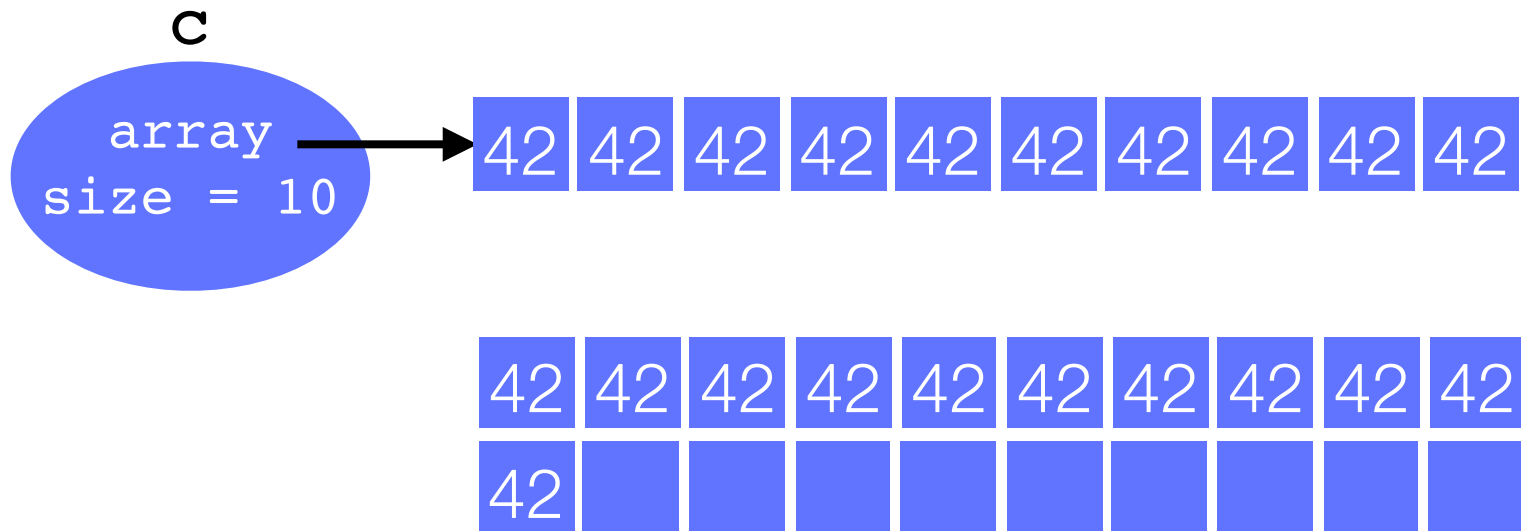
bench.	NC	NNEC	NEC	NCE	NCB	NAC	NOSM	NSM	NAB	NUB
1	643,603	151,665(23%)	491,938(76%)	2,636	35,940	229,379	112,308(17%)	112,439(17%)	3,956,792	3,044,076
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	79,035	34,970(44%)	44,065(55%)	45	2,272	44,294	10,612(13%)	10,689(13%)	1,397,668	1,236,260
4	1,690	512(30%)	1,178(69%)	30	1,720	905	245(14%)	245(14%)	33,160	22,820
5	3,551	2,009(56%)	1,542(43%)	30	1,208	2,685	200(5%)	251(7%)	65,208	55,124
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	927	48(7%)	48(7%)	4,509,216	689,468
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,395	58,331(36%)	100,064(63%)	5,663	280,876	123,703	57,637(36%)	57,718(36%)	5,220,152	3,921,204
10	1,432,306	14,891(1%)	1,417,415(98%)	6,074	12,258,424	68,132	8,248(0%)	8,344(0%)	19,547,672	6,830,248
11	6,839	2,058(30%)	4,781(69%)	1,280	78,712	2,186	471(6%)	480(7%)	109,212	75,012
12	7,870	3,472(44%)	4,398(55%)	1,094	44,384	5,253	20(0%)	20(0%)	194,732	84,716
13	108,571	57,590(53%)	50,981(46%)	1,739	369,336	62,510	8,151(7%)	8,810(8%)	4,218,656	3,286,404
14	10,305	586(5%)	9,719(94%)	145	14,044	963	82(0%)	110(1%)	136,008	112,532
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,489	5,736(27%)	5,740(27%)	2,640,784	1,932,632
16	766	172(22%)	594(77%)	17	496	221	123(16%)	123(16%)	14,288	10,452
17	1,203	880(73%)	323(26%)	1,512	48,384	2,392	764(63%)	764(63%)	120,856	29,488
<b>total</b>	<b>2,475,704</b>	<b>342,296(13%)</b>	<b>2,133,408(86%)</b>	<b>21,134</b>	<b>15,519,656</b>	<b>562,143</b>	<b>204,686(8%)</b>	<b>205,822(8%)</b>	<b>42,168,432</b>	<b>21,334,112</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	526,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,000	52,000	41,000(22%)	41,000(22%)	1,016,000	660,000
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,200	123,000	87,570(20%)	87,570(20%)	2,389,680	1,518,610
bN1	150	0(0%)	150(100%)	0	0	0	0(0%)	0(0%)	0	0
bN2	180	150(83%)	30(16%)	60	9,000	9,000	120(66%)	120(66%)	21,840	21,840
bN3	240	240(100%)	0(0%)	60	9,000	300	180(75%)	180(75%)	22,680	7,560
bPP1	90,600	46,200(50%)	44,400(49%)	5,600	436,800	78,000	46,200(50%)	46,200(50%)	3,490,400	2,309,600
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	70,200	44,800(57%)	44,800(57%)	3,218,400	1,999,200
bPP3	546,710	398,420(72%)	148,290(27%)	52,860	6,475,120	543,770	398,420(72%)	398,420(72%)	26,952,320	15,040,720
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,080	17,118,080	10(0%)	10(0%)	17,120,000	17,120,000
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,680	19,042,680	10(0%)	10(0%)	19,044,600	19,044,600
bR1	400,011	7(0%)	400,004(99%)	46	2,631,600	2,631,600	0(0%)	3(0%)	5,263,360	5,263,360
bR2	2,422	1,583(65%)	839(34%)	117	15,608	15,608	289(11%)	299(12%)	109,356	67,704
bR3	78,145	53,259(68%)	24,886(31%)	4,809	686,196	686,196	13,365(17%)	13,454(17%)	7,034,296	5,998,728
<b>total</b>	<b>6,127,788</b>	<b>1,637,669(26%)</b>	<b>4,490,119(73%)</b>	<b>980,792</b>	<b>46,953,084</b>	<b>1,874,940</b>	<b>670,064(10%)</b>	<b>670,166(10%)</b>	<b>86,510,132</b>	<b>61,393,008</b>
<b>Total</b>	<b>8,603,492</b>	<b>1,979,965(23%)</b>	<b>6,623,527(76%)</b>	<b>1,001,926</b>	<b>62,472,740</b>	<b>2,437,083</b>	<b>874,750(10%)</b>	<b>875,988(10%)</b>	<b>128,678,564</b>	<b>82,727,120</b>

Table 4: Lazy interna -73% ation

-72%

# When adding the 11th element

---

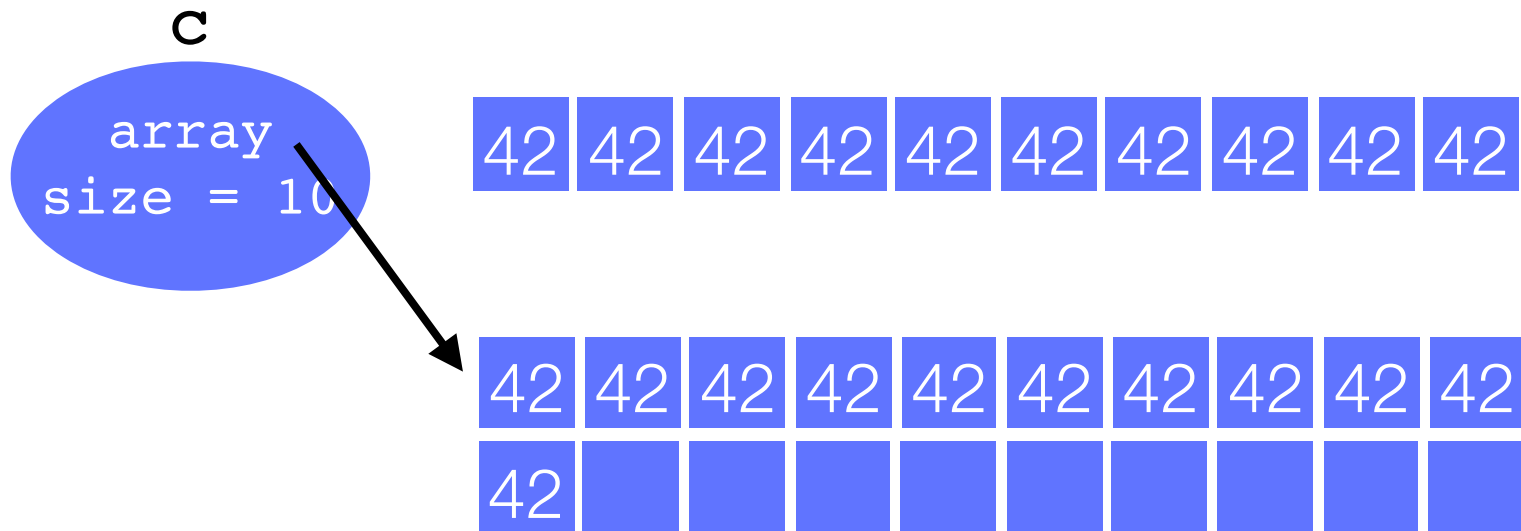


```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```



# When adding the 11th element

---



```
c := OrderedCollection new.  
11 timesRepeat: [ c add: 42 ].
```

# Recycling the array ...

---

42 42 42 42 42 42 42 42 42 42

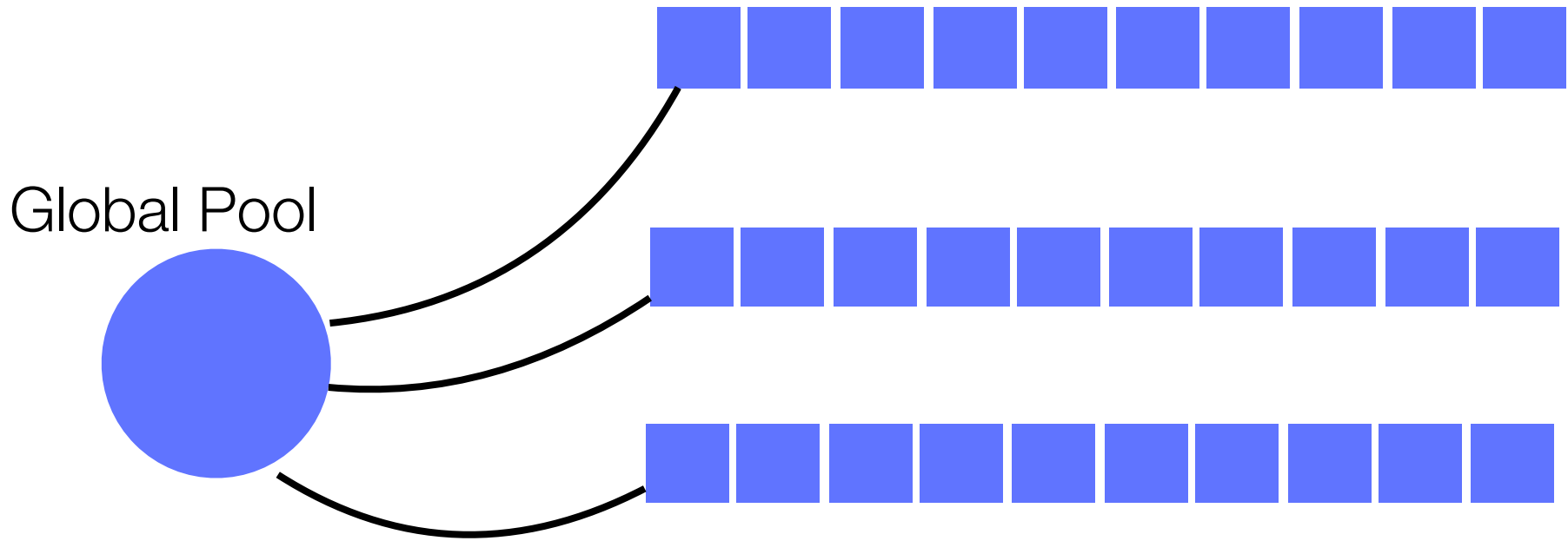
# Emptying it

---



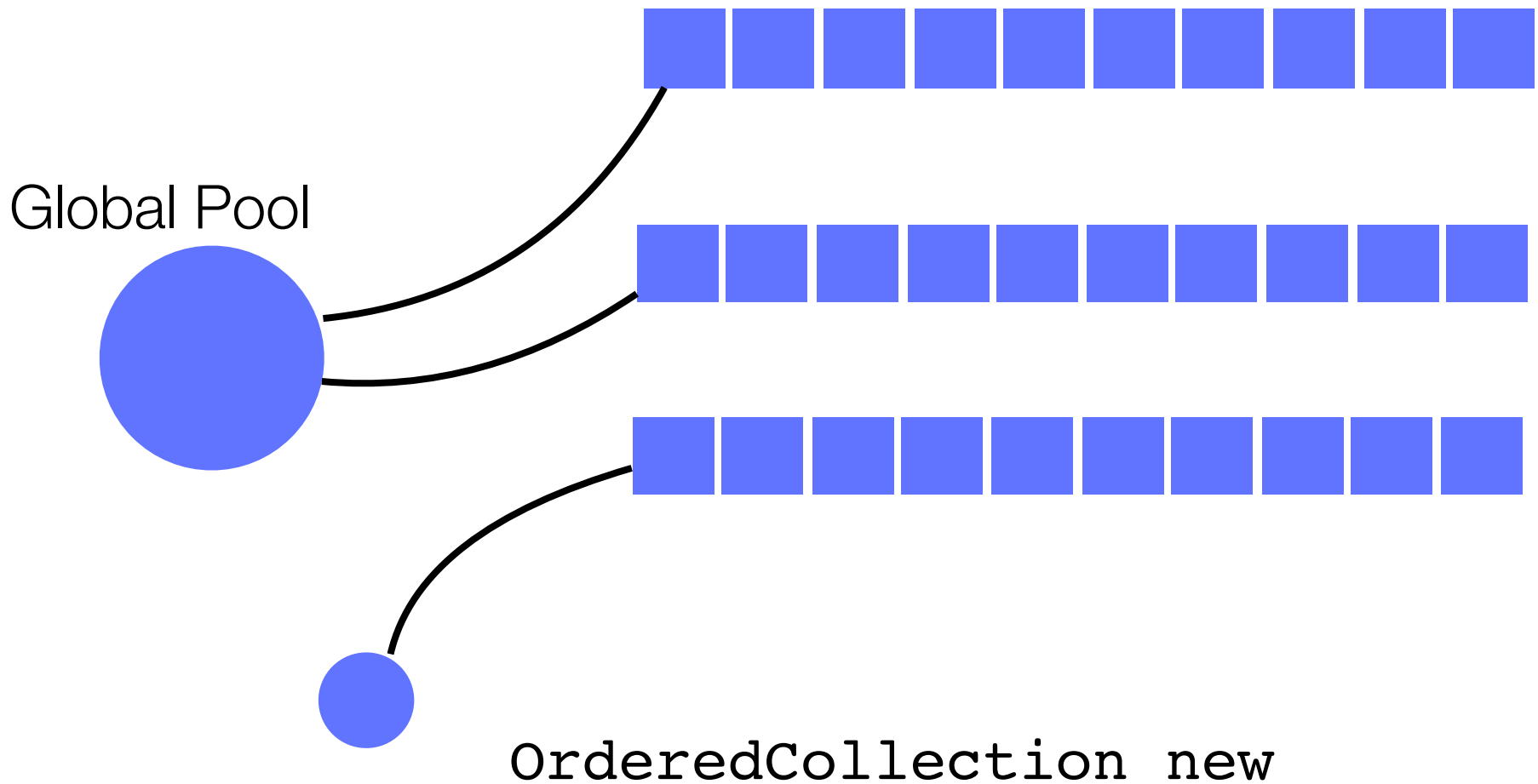
# Inserting the array in a pool

---



# Inserting the array in a pool

---



bench.	<i>NC</i>	<i>NNEC</i>	<i>NEC</i>	<i>NCE</i>	<i>NCB</i>	<i>NAC</i>	<i>NOSM</i>	<i>NSM</i>	<i>NAB</i>	<i>NUB</i>
1	643,603	151,665(23%)	491,938(76%)	2,636	35,940	226,736	112,308(17%)	112,439(17%)	3,921,136	3,044,076
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	78,977	34,912(44%)	44,065(55%)	45	2,272	44,193	10,604(13%)	10,681(13%)	1,393,396	1,234,432
4	1,690	512(30%)	1,178(69%)	30	1,720	873	245(14%)	245(14%)	31,200	22,820
5	3,551	2,009(56%)	1,542(43%)	30	1,208	2,683	200(5%)	251(7%)	65,168	55,124
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	235	48(7%)	48(7%)	2,372,616	689,468
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,399	58,333(36%)	100,066(63%)	5,663	280,876	118,044	57,639(36%)	57,720(36%)	4,938,224	3,921,356
10	1,432,306	14,891(1%)	1,417,415(98%)	6,074	12,258,424	62,065	8,248(0%)	8,344(0%)	11,877,480	6,830,244
11	6,839	2,058(30%)	4,781(69%)	1,280	78,712	2,056	471(6%)	480(7%)	91,804	75,012
12	7,870	3,472(44%)	4,398(55%)	1,094	44,384	4,163	20(0%)	20(0%)	144,144	84,716
13	108,571	57,590(53%)	50,981(46%)	1,739	369,336	61,326	8,151(7%)	8,810(8%)	3,907,184	3,286,404
14	10,305	586(5%)	9,719(94%)	145	14,044	817	82(0%)	110(1%)	119,512	112,532
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,233	5,736(27%)	5,740(27%)	2,514,840	1,932,620
16	766	172(22%)	594(77%)	17	496	203	123(16%)	123(16%)	13,700	10,452
17	1,203	880(73%)	323(26%)	1,512	48,384	882	764(63%)	764(63%)	72,536	29,488
<b>total</b>	<b>2,475,650</b>	<b>342,240(13%)</b>	<b>2,133,410(86%)</b>	<b>21,134</b>	<b>15,519,656</b>	<b>542,613</b>	<b>204,680(8%)</b>	<b>205,816(8%)</b>	<b>31,466,968</b>	<b>21,332,420</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	536,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,000	49,002	41,000(22%)	41,000(22%)	992,012	668,000
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,200	110,370	87,570(20%)	87,570(20%)	2,361,480	1,806,960
bN1	150	0(0%)	150(100%)	0	0	0	0(0%)	0(0%)	0	0
bN2	180	150(83%)	30(16%)	60	9,000	153	120(66%)	120(66%)	13,400	7,080
bN3	240	240(100%)	0(0%)	60	9,000	243	180(75%)	180(75%)	14,240	7,560
bPP1	91,000	46,400(50%)	44,600(49%)	5,600	437,600	72,603	46,400(50%)	46,400(50%)	3,058,196	2,312,800
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	63,604	44,800(57%)	44,800(57%)	2,743,088	2,000,000
bPP3	546,710	398,420(72%)	148,290(27%)	52,170	6,449,480	490,915	398,420(72%)	398,420(72%)	20,488,808	15,051,560
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,080	427,970	10(0%)	10(0%)	17,119,200	15,405,880
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,720	476,011	10(0%)	10(0%)	19,042,492	17,136,640
bR1	400,011	7(0%)	400,004(99%)	46	2,631,600	38	0(0%)	3(0%)	5,243,040	1,023,116
bR2	2,422	1,583(65%)	839(34%)	117	15,608	1,597	289(11%)	299(12%)	94,656	67,712
bR3	78,145	53,259(68%)	24,886(31%)	4,872	699,036	58,872	13,365(17%)	13,454(17%)	6,375,776	5,390,776
<b>total</b>	<b>6,128,188</b>	<b>1,637,869(26%)</b>	<b>4,490,319(73%)</b>	<b>980,165</b>	<b>46,941,124</b>	<b>1,798,578</b>	<b>670,264(10%)</b>	<b>670,366(10%)</b>	<b>78,373,588</b>	<b>61,420,484</b>
<b>Total</b>	<b>8,603,838</b>	<b>1,980,109(13%)</b>	<b>6,623,729(86%)</b>	<b>1,001,299</b>	<b>62,460,780</b>	<b>2,341,191</b>	<b>874,944(8%)</b>	<b>876,182(8%)</b>	<b>109,840,556</b>	<b>82,752,904</b>

Table 5: Lazy internal array creation + reuse of array

bench.	NC	NNEC	NEC	NCE	NCB	NAC	NOSM	NSM	NAB	NUB
1	643,603	151,665(23%)	491,938(76%)	2,636	35,940	226,736	112,308(17%)	112,439(17%)	3,921,136	3,044,076
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	78,977	34,912(44%)	44,065(55%)	45	2,272	44,193	10,604(13%)	10,681(13%)	1,393,396	1,234,432
4	1,690	512(30%)	1,178(69%)	30	1,720	873	245(14%)	245(14%)	31,200	22,820
5	3,551	2,009(56%)	1,542(43%)	30	1,208	2,683	200(5%)	251(7%)	65,168	55,124
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	235	48(7%)	48(7%)	2,372,616	689,468
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,399	58,333(36%)	100,066(63%)	5,663	280,876	118,044	57,639(36%)	57,720(36%)	4,938,224	3,921,356
10	1,432,306	14,891(1%)	1,417,415(98%)	6,074	12,258,424	62,065	8,248(0%)	8,344(0%)	11,877,480	6,830,244
11	6,839	2,058(30%)	4,781(69%)	1,280	78,712	2,056	471(6%)	480(7%)	91,804	75,012
12	7,870	3,472(44%)	4,398(55%)	1,094	44,384	4,163	20(0%)	20(0%)	144,144	84,716
13	108,571	57,590(53%)	50,981(46%)	1,739	369,336	61,326	8,151(7%)	8,810(8%)	3,907,184	3,286,404
14	10,305	586(5%)	9,719(94%)	145	14,044	817	82(0%)	110(1%)	119,512	112,532
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,233	5,736(27%)	5,740(27%)	2,514,840	1,932,620
16	766	172(22%)	594(77%)	17	496	303	123(16%)	123(16%)	13,700	10,450
17	1,203	880(73%)	323(26%)	1,512	48,384	882	764(63%)	764(63%)	72,536	29,488
<b>total</b>	<b>2,475,650</b>	<b>342,240(13%)</b>	<b>2,133,410(86%)</b>	<b>21,134</b>	<b>15,519,656</b>	<b>542,613</b>	<b>204,680(8%)</b>	<b>205,816(8%)</b>	<b>31,466,968</b>	<b>21,332,420</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	550,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,0	17,000	11,000(22%)	41,000(22%)	5	5
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,2	27,000	37,570(20%)	87,570(20%)	2,3	2,3
bN1	150	0(0%)	150(100%)	0			0(0%)	0(0%)		
bN2	180	150(83%)	30(16%)	60	9,000	8,940	120(66%)	120(66%)	1,100	1,040
bN3	240	240(100%)	0(0%)	60	9,000	243	180(75%)	180(75%)	14,240	7,560
bPP1	91,000	46,400(50%)	44,600(49%)	5,600	437,600	72,603	46,400(50%)	46,400(50%)	3,058,196	2,312,800
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	63,604	44,800(57%)	44,800(57%)	2,743,088	2,000,000
bPP3	546,710	398,420(72%)	148,290(27%)	52,170	6,449,480	490,915	398,420(72%)	398,420(72%)	20,488,808	15,051,560
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,000	16,680	10(0%)	10(0%)	17,	17,
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,7	18,560	10(0%)	10(0%)	19,	19,
bR1	400,011	7(0%)	400,004(99%)	46	2,631,6	2,585	0(0%)	3(0%)	5,	5,
bR2	2,422	1,583(65%)	839(34%)	117	15,6	14,417	289(11%)	299(12%)	1,100	1,100
bR3	78,145	53,259(68%)	24,886(31%)	4,872	699,036	53,872	13,365(17%)	13,454(17%)	6,375,776	5,300,576
<b>total</b>	<b>6,128,188</b>	<b>1,637,869(26%)</b>	<b>4,490,319(73%)</b>	<b>980,165</b>	<b>46,941,124</b>	<b>1,798,578</b>	<b>670,264(10%)</b>	<b>670,366(10%)</b>	<b>78,373,588</b>	<b>61,420,484</b>
<b>Total</b>	<b>8,603,838</b>	<b>1,980,109(13%)</b>	<b>6,623,729(86%)</b>	<b>1,001,299</b>	<b>62,460,780</b>	<b>2,341,191</b>	<b>874,944(8%)</b>	<b>876,182(8%)</b>	<b>109,840,556</b>	<b>82,752,904</b>

**-0.03%**

**-0.0005%**

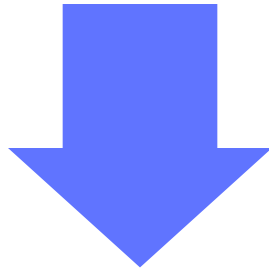
**-0.04%**

**+0.0004%**

Table 5: Lazy internal at **-0.00003%** of array

**0%**

```
ROView>>elementsToRender
| answer |
answer := OrderedCollection new.
self elementsToRenderDo:
    [ :el | answer add: el ].
^ answer
```



```
ROView>>elementsToRender
| answer |
answer := OrderedCollection new:
           (self elements size).

self elementsToRenderDo:
    [ :el | answer add: el ].
^ answer
```



bench.	<i>NC</i>	<i>NNEC</i>	<i>NEC</i>	<i>NCE</i>	<i>NCB</i>	<i>NAC</i>	<i>NOSM</i>	<i>NSM</i>	<i>NAB</i>	<i>NUB</i>
1	643,626	151,687(23%)	491,939(76%)	2,648	35,956	226,964	112,329(17%)	112,460(17%)	3,906,608	3,029,456
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	79,043	34,974(44%)	44,069(55%)	65	2,108	44,259	10,616(13%)	10,693(13%)	1,384,428	1,225,288
4	1,691	513(30%)	1,178(69%)	30	1,760	874	246(14%)	246(14%)	30,952	22,532
5	3,551	2,009(56%)	1,542(43%)	28	1,120	2,683	200(5%)	251(7%)	65,352	55,308
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	235	48(7%)	48(7%)	2,372,608	689,492
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,399	58,333(36%)	100,066(63%)	5,651	280,636	118,043	57,639(36%)	57,720(36%)	4,938,340	3,921,516
10	1,432,306	14,891(1%)	1,417,415(98%)	6,069	12,280,124	62,070	8,248(0%)	8,344(0%)	11,852,428	6,828,504
11	6,839	2,058(30%)	4,781(69%)	1,280	78,760	2,055	471(6%)	480(7%)	91,812	75,064
12	7,870	3,472(44%)	4,398(55%)	1,095	46,236	4,163	20(0%)	20(0%)	145,360	86,528
13	108,571	57,589(53%)	50,982(46%)	1,683	367,164	61,330	8,150(7%)	8,809(8%)	3,865,908	3,245,964
14	10,305	586(5%)	9,719(94%)	145	14,044	819	82(0%)	110(1%)	120,872	112,660
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,233	5,736(27%)	5,740(27%)	2,514,084	1,932,656
16	766	172(22%)	594(77%)	14	376	202	123(16%)	123(16%)	12,488	9,428
17	1,203	880(73%)	323(26%)	0	0	880	764(63%)	764(63%)	72,472	29,488
<b>total</b>	<b>2,475,740</b>	<b>342,324(13%)</b>	<b>2,133,416(86%)</b>	<b>19,577</b>	<b>15,492,144</b>	<b>542,914</b>	<b>204,713(8%)</b>	<b>205,849(8%)</b>	<b>31,377,740</b>	<b>21,267,560</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	536,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	24,000	49,002	41,000(22%)	41,000(22%)	992,012	668,000
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	28,200	110,370	87,570(20%)	87,570(20%)	2,329,080	1,774,560
bN1	150	0(0%)	150(100%)	0	0	0	0(0%)	0(0%)	0	0
bN2	180	150(83%)	30(16%)	60	9,000	154	120(66%)	120(66%)	11,280	4,920
bN3	240	240(100%)	0(0%)	60	9,000	244	180(75%)	180(75%)	12,120	5,400
bPP1	90,600	46,200(50%)	44,400(49%)	5,600	437,600	72,403	46,200(50%)	46,200(50%)	3,057,396	2,312,800
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	63,604	44,800(57%)	44,800(57%)	2,743,088	2,000,000
bPP3	546,710	398,420(72%)	148,290(27%)	52,170	6,449,480	490,915	398,420(72%)	398,420(72%)	20,488,808	15,051,560
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,118,080	427,970	10(0%)	10(0%)	17,119,200	15,405,880
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,042,720	476,011	10(0%)	10(0%)	19,042,492	17,136,640
bR1	400,011	7(0%)	400,004(99%)	46	2,631,600	38	0(0%)	3(0%)	5,243,040	1,023,116
bR2	2,422	1,583(65%)	839(34%)	117	15,608	1,597	289(11%)	299(12%)	94,616	67,672
bR3	78,145	53,259(68%)	24,886(31%)	4,872	699,036	58,872	13,365(17%)	13,454(17%)	6,375,736	5,390,736
<b>total</b>	<b>6,127,788</b>	<b>1,637,669(26%)</b>	<b>4,490,119(73%)</b>	<b>980,165</b>	<b>46,941,124</b>	<b>1,798,380</b>	<b>670,064(10%)</b>	<b>670,166(10%)</b>	<b>78,336,068</b>	<b>61,383,684</b>
<b>Total</b>	<b>8,603,528</b>	<b>1,979,993(23%)</b>	<b>6,623,535(76%)</b>	<b>999,742</b>	<b>62,433,268</b>	<b>2,341,294</b>	<b>874,777(10%)</b>	<b>876,015(10%)</b>	<b>109,713,808</b>	<b>82,651,244</b>

Table 6: Lazy internal array creation + reuse of array + code refactoring

bench.	NC	NNEC	NEC	NCE	NCB	NAC	NOSM	NSM	NAB	NUB
1	643,626	151,687(23%)	491,939(76%)	2,648	35,956	226,964	112,329(17%)	112,460(17%)	3,906,608	3,029,456
2	13	12(92%)	1(7%)	0	0	13	1(7%)	1(7%)	468	392
3	79,043	34,974(44%)	44,069(55%)	65	2,108	44,259	10,616(13%)	10,693(13%)	1,384,428	1,225,288
4	1,691	513(30%)	1,178(69%)	30	1,760	874	246(14%)	246(14%)	30,952	22,532
5	3,551	2,009(56%)	1,542(43%)	28	1,120	2,683	200(5%)	251(7%)	65,352	55,308
6	96	44(45%)	52(54%)	0	0	91	40(41%)	40(41%)	3,560	3,284
7	644	218(33%)	426(66%)	614	2,257,960	235	48(7%)	48(7%)	2,372,608	689,492
8	2	0(0%)	2(100%)	0	0	0	0(0%)	0(0%)	0	0
9	158,399	58,333(36%)	100,066(63%)	5,651	280,636	118,043	57,639(36%)	57,720(36%)	4,938,340	3,921,516
10	1,432,306	14,891(1%)	1,417,415(98%)	6,069	12,280,124	62,070	8,248(0%)	8,344(0%)	11,852,428	6,828,504
11	6,839	2,058(30%)	4,781(69%)	1,280	78,760	2,055	471(6%)	480(7%)	91,812	75,064
12	7,870	3,472(44%)	4,398(55%)	1,095	46,236	4,163	20(0%)	20(0%)	145,360	86,528
13	108,571	57,589(53%)	50,982(46%)	1,683	367,164	61,330	8,150(7%)	8,809(8%)	3,865,908	3,245,964
14	10,305	586(5%)	9,719(94%)	145	14,044	819	82(0%)	110(1%)	120,872	112,660
15	20,815	14,886(71%)	5,929(28%)	255	125,900	18,233	5,736(27%)	5,740(27%)	2,514,084	1,932,656
16	766	172(22%)	594(77%)	14	376	222	123(16%)	123(16%)	12,488	7,720
17	1,203	880(73%)	323(26%)	0	0	880	764(63%)	764(63%)	72,472	29,488
<b>total</b>	<b>2,475,740</b>	<b>342,324(13%)</b>	<b>2,133,416(86%)</b>	<b>19,577</b>	<b>15,492,144</b>	<b>542,914</b>	<b>204,713(8%)</b>	<b>205,849(8%)</b>	<b>31,377,740</b>	<b>21,267,560</b>
bAST1	210,000	38,000(18%)	172,000(81%)	0	0	47,000	38,000(18%)	38,000(18%)	820,000	556,000
bAST2	179,000	47,000(26%)	132,000(73%)	4,000	0	0	0(0%)	41,000(22%)	9	0
bAST3	428,550	103,830(24%)	324,720(75%)	2,670	0	0	570(20%)	87,570(20%)	2,3	0
bN1	150	0(0%)	150(100%)	0	0	0	0(0%)	0(0%)	0	0
bN2	180	150(83%)	30(16%)	60	0	0	120(66%)	120(66%)	11,200	7,200
bN3	240	240(100%)	0(0%)	60	9,000	244	180(75%)	180(75%)	12,120	5,400
bPP1	90,600	46,200(50%)	44,400(49%)	5,600	437,600	72,403	46,200(50%)	46,200(50%)	3,057,396	2,312,800
bPP2	78,000	44,800(57%)	33,200(42%)	6,600	476,800	63,604	44,800(57%)	44,800(57%)	2,743,088	2,000,000
bPP3	546,710	398,420(72%)	148,290(27%)	52,170	6,449,480	490,915	398,420(72%)	398,420(72%)	20,488,808	15,051,560
bReg1	1,000	200(20%)	800(80%)	0	0	200	100(10%)	100(10%)	7,200	6,400
bReg2	2,162,830	427,970(19%)	1,734,860(80%)	427,950	17,000	0	10(0%)	10(0%)	17,1	0
bReg3	1,949,950	476,010(24%)	1,473,940(75%)	476,020	19,000	0	10(0%)	10(0%)	19,0	0
bR1	400,011	7(0%)	400,004(99%)	46	2,000	0	0(0%)	3(0%)	5,2	0
bR2	2,422	1,583(65%)	839(34%)	117	0	0	289(11%)	299(12%)	0	0
bR3	78,145	53,259(68%)	24,886(31%)	4,872	699,036	13,372	13,365(17%)	13,454(17%)	6,375,736	5,999,720
<b>total</b>	<b>6,127,788</b>	<b>1,637,669(26%)</b>	<b>4,490,119(73%)</b>	<b>980,165</b>	<b>46,941,124</b>	<b>1,798,380</b>	<b>670,064(10%)</b>	<b>670,166(10%)</b>	<b>78,336,068</b>	<b>61,383,684</b>
<b>Total</b>	<b>8,603,528</b>	<b>1,979,993(23%)</b>	<b>6,623,535(76%)</b>	<b>999,742</b>	<b>62,433,268</b>	<b>2,341,294</b>	<b>874,777(10%)</b>	<b>876,015(10%)</b>	<b>109,713,808</b>	<b>82,651,244</b>

**+0.0005%**

**-0.003%**

**-0.0001%**

**-0.001%**

Table 6: Lazy internal array creation **-0.03%** rray + code refactoring **-0.001%**

# Conclusion

---

Expandable collections are an important piece of the runtime

3 optimizations

Lazy array initialization

Recycling arrays

Refactoring applications

# Conclusion

---

reduced the number of created intermediary internal array storage by 73%

reduced the number of allocated bytes by 67%

reduced the number of unused bytes by 72%

*Lazy array creation should be put in the main release*