## 14-JavaJIT编译器(二): SeaofNodes为何如此强大?

你好,我是宫文学。这一讲,我们继续来研究Graal编译器,重点来了解一下它的IR的设计。

在上一讲中,我们发现Graal在执行过程中,创建了一个图的数据结构,这个数据结构就是Graal的IR。之后的很多处理和优化算法,都是基于这个IR的。可以说,这个IR是Graal编译器的核心特性之一。

# 那么,为什么这个IR采用的是图结构?它有什么特点和优点?编译器的优化算法又是如何基于这个IR来运行的呢?

今天,我就带你一起来攻破以上这些问题。在揭晓问题答案的过程中,你对真实编译器中IR的设计和优化处理过程,也就能获得直观的认识了。

### 基于图的IR

IR对于编译器非常重要,因为它填补了高级语言和机器语言在语义上的巨大差别。比如说,你在高级语言中是使用一个数组,而翻译成最高效的x86机器码,是用间接寻址的方式,去访问一块连续的内存。所以IR的设计必须有利于实现这种转换,并且还要有利于运行优化算法,使得生成的代码更加高效。

在上一讲中,通过跟踪Graal编译器的执行过程,我们会发现它在一开始,就把字节码翻译成了一种新的IR,这个IR是用图的结构来表示的。那这个图长什么样子呢?非常幸运的是,我们可以用工具来直观地看到它的结构。

你可以从Oracle的网站上,下载一个idealgraphvisualizer的工具。下载之后,解压缩,并运行它:

```
export PATH="/<上级目录>/idealgraphvisualizer/bin:$PATH"
idealgraphvisualizer &
```

这时,程序会启动一个图形界面,并在4445端口上等待GraalVM发送数据过来。

接着,还是运行Foo示例程序,不过这次你要增加一个参数"-Dgraal.Dump",这会让GraalVM输出编译过程的一些中间结果。并且在这个示例程序当中,我还增加了一个"-Xcomp"参数,它能让JIT编译器在第一次使用某个方法的时候,就去做编译工作。

```
mx vm \
   -XX:+UnlockExperimentalVMOptions \
   -XX:+EnableJVMCI \
   -XX:+UseJVMCICompiler \
   -XX:-TieredCompilation \
   -XX:CompileOnly=Foo \
   -Dgraal.Dump \
   -Xcomp \
Foo
```

idealgraphvisualizer。接着,在即时编译之后,idealgraphvisualizer就接收到了编译过程中生成的图,你可以点击显示它。

这里我展示了其中两个阶段的图,一个是刚解析完字节码之后(After parsing),一个是在处理完中间层之后(After mid tier)。

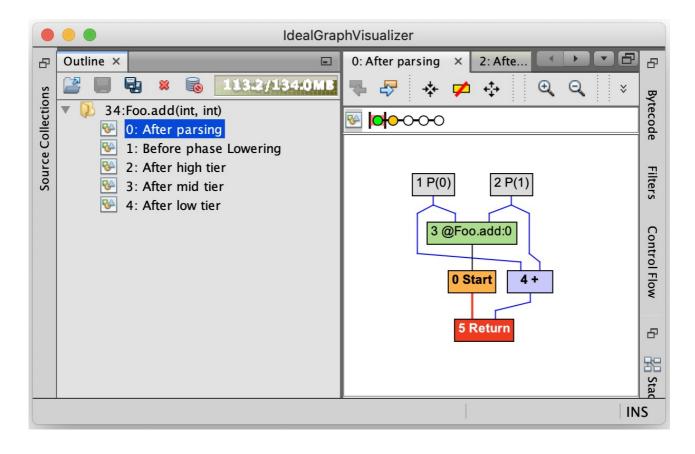


图1: After parsing

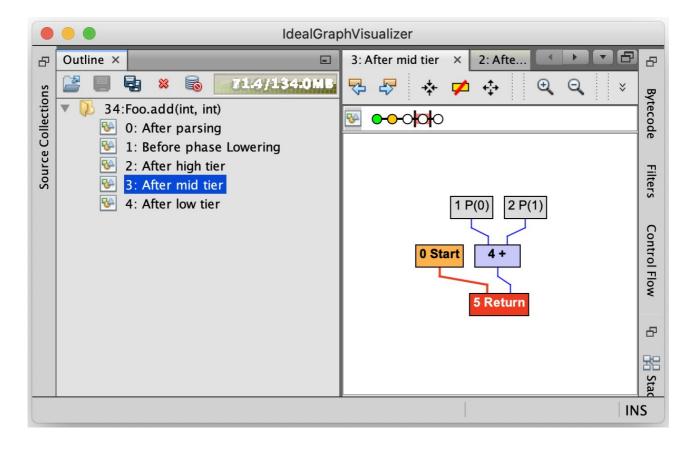


图2: After mid tier

Graal IR其实受到了"程序依赖图"的影响。我们在<mark>第6讲</mark>中提到过程序依赖图(PDG),它是用图来表示程序中的数据依赖和控制依赖。并且你也知道了,这种IR还有一个别名,叫做**节点之海(Sea of Nodes)**。因为当程序稍微复杂一点以后,图里的节点就会变得非常多,我们用肉眼很难看得清。

基于Sea of Nodes的IR呢,算是后起之秀。在HotSpot的编译器中,就采用了这种IR,而且现在Java的Graal编译器和JavaScript的V8编译器中的IR的设计,都是基于了Sea of Nodes结构,所以我们必须重视它。

这也不禁让我们感到好奇了: Sea of Nodes到底强在哪里?

我们都知道,数据结构的设计对于算法来说至关重要。IR的数据结构,会影响到算法的编写方式。好的IR的设计,会让优化算法的编写和维护都更加容易。

# 而Sea of Nodes最大的优点,就是能够用一个数据结构同时反映控制流和数据流,并且尽量减少它们之间的互相依赖。

怎么理解这个优点呢?在传统的编译器里,控制流和数据流是分开的。控制流是用控制流图(Control-flow Graph,CFG)来表示的,比如GNU的编译器、LLVM,都是基于控制流图的。而IR本身,则侧重于表达数据流。

以LLVM为例,它采用了SSA格式的IR,这种IR可以很好地体现值的定义和使用关系,从而很好地刻画了数据流。

而问题在于,采用这种比较传统的方式,控制流和数据流会耦合得比较紧,因为IR指令必须归属于某个基本块。

举个例子来说明一下吧。在下面的示例程序中,"int b = a\*2;"这个语句,会被放到循环体的基本块中。

可是,从数据流的角度看,变量b只依赖于a。所以这个语句没必要放在循环体内,而是可以提到外面。在 传统的编译器中,这一步是要分析出循环无关的变量,然后再把这条语句提出去。而如果采用Sea of Nodes 的数据结构,变量b一开始根本没有归属到特定的基本块,所以也就没有必要专门去做代码的移动了。

另外,我们之前讲<mark>本地优化和全局优化</mark>的时候,也提到过,它们的区别就是,在整个函数范围内,优化的范围是在基本块内还是会跨基本块。而Sea of Nodes没有过于受到基本块的束缚,因此也就更容易做全局优化了。

好,那在概要地理解了Graal IR的数据结构之后,接下来,我们就具体了解一下Graal IR,包括认识一下数据流与控制流的特点,了解两种不同的节点:浮动节点和固定节点,以及认识一种特殊的节点:FrameState。

#### 数据流和控制流

我们已经知道,Graal IR整合了两种图结构:数据流图和控制流图。

#### 首先,我们来看看它的数据流。

在下图中,蓝色的边代表的是数据流,也就是数据之间的依赖关系。参数1("P(0)"节点)和参数 2("P(1)"节点)的值流入到+号节点,再流入到Return节点。

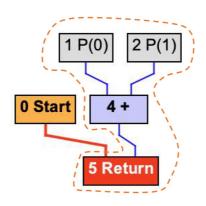


图3: Foo.add()的数据流

在Graal IR的设计中,Add节点有两个输入,分别是x和y,这两个输入是AddNode的两个属性。**注意**,这个 图中的箭头方向代表的是**数据依赖关系**,也就是Add节点保持着对它的两个输入节点的引用,这其实跟AST 是一致的。**而数据流向,则是反过来的**,从x和y流向Add节点。

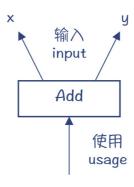


图4: Add节点的数据依赖关系

查看AddNode的设计,你会发现其父类中有两个成员变量,x和y。它们用@input做了注解,这就意味着, 这两个成员变量代表的是数据流图中的两条边。

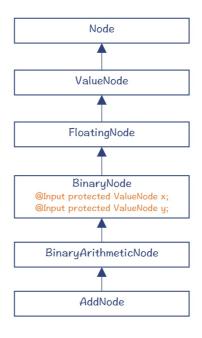


图5: Add节点及其各级父节点

另外,Graal IR的数据流图是符合SSA格式的。也就是说,每个节点代表了SSA中的一个值,它只被定义一次,也就相当于SSA中的每个变量只被赋值一次。

#### 我们再来看看控制流。

下图中,红色的边代表的是控制流,控制流图代表的是程序执行方向的改变。进入或退出一个函数、条件分支语句、循环语句等,都会导致程序的执行从一个地方跳到另一个地方。

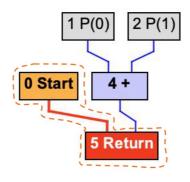


图6: Foo.add()的控制流

数据流加上控制流,就能完整表达程序的含义,它等价于字节码,也等价于更早期的AST。你可以从Start节点,沿着控制流遍历这个图。当到达Return节点之前,Return所依赖的数据(x+y)也需要计算出来。

add()方法的控制流很简单,只有Start和Return两个节点。我们做一个稍微复杂一点的例子,在Foo.add2()示例程序中,调用两个函数getX()和getY(),分别获取x和y成员变量。

```
public int add2(){
   return getX() + getY();
}
```

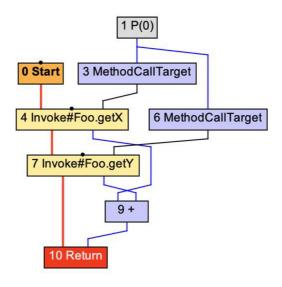


图7: Foo.add2()对应的IR

注意:对于这个例子,在使用GraalVM时,要使用-XX:-Inline选项,避免编译器做内联优化,否则Foo.getX()和Foo.getY()会被内联。我们在下一讲中就会探讨内联优化。

除了调用其他函数,if语句、循环语句等,也会导致控制流的变化。我们看看这个例子:

```
public int doif(int x, int y){
    int z;
    if (x < 2)
        z=x+y;
    else
        z=x*y;
    return z;
}</pre>
```

它对应的Graal图如下,if语句会让控制流产生分支,分别对应if块和else块,最后在Merge节点合并起来。

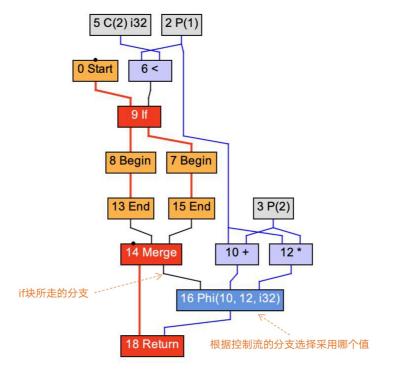


图8: doif()方法对应的IR

IfNode作为一种控制流节点,它保存着对下级节点的引用,并用@Successor注解来标注。这意味着 trueSuccessor和falseSuccessor两个成员变量,代表着控制流中的两条边。当然,你也会注意到,If节点有 一个数据流的输入,这就是If的判断条件。IR会基于这个判断条件,来决定控制流的走向。

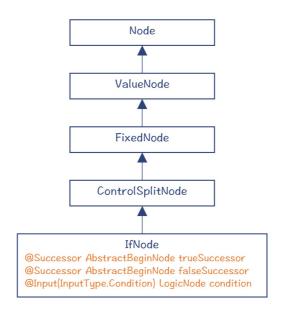


图9: IfNode及其各级父节点

跟控制流类似,数据流也产生了两个分支,分别是x+y和x\*y。最后用一个Phi节点合并到一起。

Phi节点是SSA的一个特性。在doif示例程序中,z可能有两个取值。如果控制流走的是if块,那么z=x+y;而如果走的是else块,则z=x\*y。Phi节点就起到这个作用,它根据控制流来选择值。

总结一下:控制流图表达的是控制的流转,而数据流图代表的是数据之间的依赖关系。二者双剑合璧,代表了源程序完整的语义。

接下来,我再给你介绍一下浮动节点和固定节点的概念。

#### 浮动节点和固定节点

注意,在Graal IR,数据流与控制流是相对独立的。你看看前面的doif示例程序,会发现x+y和x\*y的计算,与if语句的控制流没有直接关系。所以,你其实可以把这两个语句挪到if语句外面去执行,也不影响程序运行的结果(要引入两个临时变量z1和z2,分别代表z的两个取值)。

对于这些在执行时间上具有灵活性的节点,我们说它们是浮动的(Floating)。你在AddNode的继承层次中,可以看到一个父类: FloatingNode,这说明这个节点是浮动的。它可以在最后生成机器码(或LIR)的环节,再去确定到底归属哪个基本块。

除了浮动节点以外,还有一些节点是固定在控制流中的,前后顺序不能乱,这些节点叫做固定节点。除了那些流程控制类的节点(如IfNode)以外,还有一些节点是固定节点,比如内存访问的节点。当你访问一个对象的属性时,就需要访问内存。

内存是个共享资源,同一个内存地址(比如对象的属性),可以被多次读写。也就是说,内存位置不是SSA 中的值,所以也不受单赋值的约束。

对同一个内存地址的读写操作,顺序是不能乱的。比如下面代码中,第二行和第三行的顺序是不能变的,它 们被固定在了控制流中。

```
x := 10
store x to 地址a
y := load 地址a
z := y + 10
```

不过,在运行某些优化算法的时候,某些固定节点会被转化成浮动节点,从而提供了更大的代码优化空间。 我们在下一讲的"内联和逃逸分析"中,会见到这样的例子。

#### FrameState节点

在看Graal IR的时候,你经常会遇到一个绿色的节点插在图中。为避免你产生困惑,接下来我就专门给你解释一下这个节点,我们一起来认识一下它。

在Foo.add()新生成的IR中,如果你不勾选"Remove State"选项,就会显示出一个绿色的节点。这个节点就是FrameState节点。

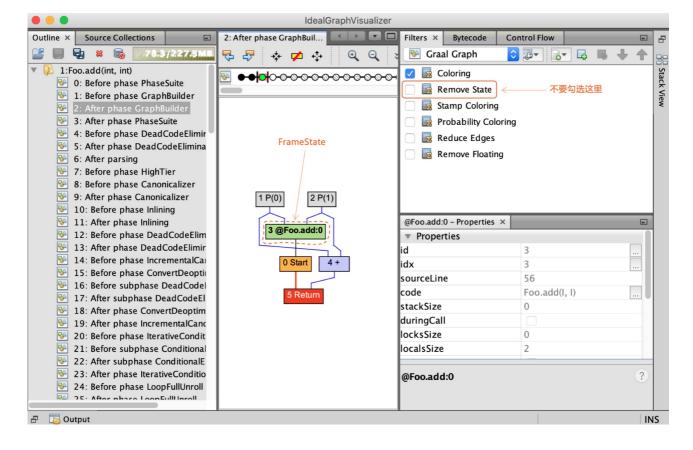


图10: Foo.add()中的FrameState节点

FrameState比较特殊。它保存了栈帧的状态,而且这里我指的是Java字节码解释器的栈帧的状态,包括了本地变量和操作数栈里的值。

#### 为什么要保存栈帧的状态呢?

**第一个用途,是用于逆优化。**上一讲我们说过,编译器有时候会基于推测做一些激进的优化,比如忽略掉某些分支。但如果推测依据的前提错了,那么就要做逆优化,重新回到解释器去执行。而FrameState的作用就是在代码中一些叫做安全点的地方,记录下栈帧的状态,便于逆优化以后,在解释器里去接着执行程序。

**第二个用途,是用于debug。** 编译器会用FrameState,来记录程序执行的一些中间状态值,以方便程序的调试。

对于Foo.add()方法的IR,通过后面的一些优化处理,你会发现Foo.add()并不需要逆优化,那么FrameState 节点就会被去掉。否则,FrameState就会转化成一个逆优化节点,生成与逆优化有关的代码。

如果你并不关心逆优化,那你在平常看IR的过程中,可以勾选"Remove State"选项,不用关注 FrameState节点就行了。

好了,我们已经大致了解了Graal IR。进一步,编译器要基于IR做各种处理和优化。

#### 对Graal IR的处理和优化

通过上一讲,我们已经知道在编译过程中要对图进行很多遍的处理。还是以Foo.add()示例程序为例,在运行GraalVM的时候,我们加上"-Dgraal.Dump=:5"选项,程序就会详细地dump出所做的处理步骤,你可以在idealgraphvisualizer中看到这些处理环节,点击每个环节可以看到相对应的IR图。

1. Jefore phase CraphBuilder	- 5 45 110 . 1 . 2	GA CA Defense de la Cofensia Filippia di la
<ul> <li>□ 2: After phase CraphBuilder</li> <li>□ 3: After phase PhaseSuite</li> <li>□ 4: Before phase DeadCodeElimination</li> <li>□ 5: After parsing</li> <li>□ 7: After parsing</li> <li>□ 7: Before phase HighTier</li> <li>□ 8: Before phase HighTier</li> <li>□ 8: Before phase Canonicalizer</li> <li>□ 9: After phase HighTier</li> <li>□ 10: Before phase Incinnation</li> <li>□ 11: After phase Inlining</li> <li>□ 12: Before phase Canonicalizer</li> <li>□ 13: After phase DeadCodeElimination</li> <li>□ 14: Before phase IncrementalCanonicalizer</li> <li>□ 15: Before phase IncrementalCanonicalizer</li> <li>□ 16: Before phase IncrementalCanonicalizer</li> <li>□ 17: After phase Infining</li> <li>□ 18: After phase DeadCodeElimination</li> <li>□ 18: After phase LoopCodeElimination</li> <li>□ 18: After phase ConvertDeoptimizer Toduard</li> <li>□ 19: After phase ConvertDeoptimizer Toduard</li> <li>□ 10: After pha</li></ul>		
<ul> <li>№ 3: After phase CraphBuilder</li> <li>№ 3: After phase DeadCodeElimination</li> <li>№ 5: After phase DeadCodeElimination</li> <li>№ 6: After parsing</li> <li>№ 7: Before phase HighTer</li> <li>№ 8: Before phase Ednoricalizer</li> <li>№ 7: After phase Canonicalizer</li> <li>№ 7: After phase DeadCodeElimination</li> <li>№ 10: Before phase Infilming</li> <li>№ 11: After phase DeadCodeElimination</li> <li>№ 12: Before phase DeadCodeElimination</li> <li>№ 13: After phase DeadCodeElimination</li> <li>№ 15: Before phase Ednoricalizer</li> <li>№ 15: Before phase ConvertDeoptimizer FoGuard</li> <li>№ 16: Before subphase DeadCodeElimination</li> <li>№ 17: After phase ConvertDeoptimizer FoGuard</li> <li>№ 18: After phase IncrementalCanonicalizer</li> <li>№ 19: After phase IncrementalCanonicalizer</li> <li>№ 21: Before subphase ConditionalElimination</li> <li>№ 21: Before phase IncrementalCanonicalizer</li> <li>№ 22: After phase LoopPelling</li> <li>№ 23: After phase LoopPelling</li> <li>№ 26: After phase LoopPelling</li> <li>№ 27: After phase IncrementalCanonicalizer</li> <li>№ 28: After phase IncrementalCanonicalizer</li> <li>№ 29: After phase IncrementalCanonicalizer</li> <li>№ 20: Before phase IncrementalCanonicalizer</li> <li>№ 20: Before phase IncrementalCanonicalizer</li> <li>№ 21: After phase EdnopPelling</li> <li>№ 22: After phase IncrementalCanonicalizer</li> <li>№ 23: After phase EdnopPelling</li> <li>№ 24: After phase EdnopPelling</li> <li>№ 25: After phase LoopPelling</li> <li>№ 26: After phase EdnopPelling</li> <li>№ 27: After phase EdnopPelling</li> <li>№ 28: After phase EdnopPelli</li></ul>		
§ 3. After phase PhaseSuite		
<ul> <li>S. * After phase DeadCodeElimination</li> <li>S. * Stefore phase HighTifer</li> <li>G. * Stefore phase (HighTifer)</li> <li>S. * Before phase (HighTifer)</li> <li>S. * Before phase (HighTifer)</li> <li>J. * D. * Before phase Elimining</li> <li>11. * After phase Canonicalizer</li> <li>J. * D. * Before phase Inlining</li> <li>11. * After phase Canonicalizer</li> <li>T. * Stefore phase Inlining</li> <li>11. * After phase DeadCodeElimination</li> <li>13. * After phase DeadCodeElimination</li> <li>15. * Before subphase DeadCodeElimination</li> <li>15. * Before subphase DeadCodeElimination</li> <li>17. * After phase ConvertDeoptimizer FoCuard</li> <li>19. * After phase IncrementalCanonicalizer</li> <li>19. * After phase DeadCodeElimination</li> <li>11. * After phase DeadCodeElimination</li> <li>11. * After phase DeadCodeElimination</li> <li>11. * After phase DeadCodeElimination</li> <li>12. * After phase IncrementalCanonicalizer</li> <li>13. * After phase IncrementalCanonicalizer</li> <li>14. * Before subphase DeadCodElimination</li> <li>15. * Before subphase DeadCodElimination</li> <li>16. * Before subphase ConditionalElimination</li> <li>17. * After subphase ConditionalElimination</li> <li>18. * Before phase EloparationalCondicibler</li> <li>18. * Before subphase DeadCodeElimination</li> <li>18. * Before phase EloparationalCondicibler</li> <li>18. * After phase EloparationalCondicibler</li> <li>18. * After phase EloparationalCondicibler</li> <li>18</li></ul>		66: Before subphase Schedule
Si. After phase DeadCode Elimination	3: After phase PhaseSuite	67: After subphase Schedule
<ul> <li>S - After phase DeadCodeElimination</li> <li>S - Before phase HighTier</li> <li>S - Before phase Hornocializer</li> <li>S - Before phase Canonicalizer</li> <li>S - Before phase Encorrecializer</li> <li>S - Before phase Encorrecializer</li> <li>S - Before phase Incincilizer</li> <li>S - Before phase DeadCodeElimination</li> <li>S - Before phase IncrementalCanonicalizer</li> <li>S - Before phase Encorrect Decide Court</li> <li>S - Before phase ConvertDeoptimizer ToGuard</li> <li>S - Before phase Encorrect Decide Court</li> <li>S - Before phase Encorrect Decide Court</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - After phase EdeadCodeElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - ConditionalElimination</li> <li>S - Before S - Before S - S - Before S</li></ul>	4: Before phase DeadCodeElimination	68: After phase GuardLowering
□	5: After phase DeadCodeElimination	
Sefore phase Canonicalizer	6: After parsing	·
8. Before phase Canonicalizer 9. 10: Before phase Inlining 9. 11: After phase Inlining 9. 12: Before phase Inlining 9. 12: Before phase DeadCodeElimination 9. 13: After phase DeadCodeElimination 9. 14: Before subphase DeadCodeElimination 16: Before phase ConvertDeoptimize FoGuard 16: Before subphase DeadCodeElimination 17: After subphase DeadCodeElimination 18: After phase ConvertDeoptimize FoGuard 19: After phase IncrementalCanonicalizer 19: After phase IncrementalCanonicalizer 19: After phase IncrementalCanonicalizer 19: After subphase ConditionalElimination 19: After subphase ConditionalElimination 19: After phase IncrementalCanonicalizer 19: After phase IncrementalCanonicalizer 19: After phase LoopPelling 19: After p		
9 9. After phase Canonicalizer 11: After phase Inlining 11: After phase DeadCodeElimination 12: Before phase IncorrentalCanonicalizer 15: Before phase IncorrentalCanonicalizer 16: Before subphase DeadCodeElimination 17: After subphase DeadCodeElimination 18: After phase ComertDeoptimizer CoGurd 19: 16: Before subphase DeadCodeElimination 19: 17: After subphase DeadCodeElimination 19: 18: After phase ComertDeoptimizer CoGurd 19: 18: After phase ComertDeoptimizer CoGurd 19: 19: After phase ComertDeoptimizer CoGurd 19: 19: Before phase LoveringCound 19: 20: Before phase LeveringCound 19: 21: Before subphase ConditionalElimination 19: 22: After subphase ConditionalElimination 19: 23: After phase IncrementalCanonicalizer 19: 25: After phase LoopPelling 19: 26: After phase LoopPelling 19: 26: After phase LoopPelling 19: 27: Before phase LoopPelling 19: 28: After phase LoopPelling 19: 29: After phase LoopPelling 19: 31: After phase LoopPelling 19: 32: After phase LoopPelling 19: 33: After phase LoopPelling 19: 33: After phase LoopPelling 19: 33: After phase LoopPelling 19: 34: After phase LoopPelling 19: 35: After phase LoopPelling 19: 36: After phase LoopPelling 19: 36: After phase LoopPelling 19: 37: After phase LoopPelling 19: 38:		,
□ 10. Before phase Inlining □ 11. After phase DeadCodeElimination □ 12. Before phase DeadCodeElimination □ 13. After phase DeadCodeElimination □ 14. Before phase ConvertDeoptimize FoGuard □ 15. Before phase ConvertDeoptimize FoGuard □ 16. Before subphase DeadCodeElimination □ 17. After subphase DeadCodeElimination □ 18. After phase IncrementalCanonicalizer □ 19. After phase IncrementalCanonicalizer □ 20. Before subphase DeadCodeElimination □ 21. After subphase DeadCodeElimination □ 22. After subphase ConditionalElimination □ 22. After subphase ConditionalElimination □ 23. After phase IncrementalCanonicalizer □ 24. Before phase LoopFullUrroll □ 25. After phase LoopPullUrroll □ 26. Before phase IncrementalCanonicalizer □ 27. Before phase LoopPullUrroll □ 28. After phase LoopPullUrroll □ 29. After phase ReassociateInvariant □ 29. After phase LoopPullurroll □ 29. After phase ReassociateInvariant □ 29. After phase Rea		=
11: After phase Inlining   13: After phase DeadCodeElimination   14: Before phase IncrementalCanonicalizer   15: Before phase IncrementalCanonicalizer   16: Before subphase DeadCodeElimination   17: After subphase DeadCodeElimination   18: After phase DeadCodElimination   18: After phase ConvertDeoptimizeToGuard   19: After phase ConvertDeoptimizeToGuard   19: After phase DeadCodElimination   19: After phase DeadCodElimination   19: After phase DeadCodElimination   19: After phase IncrementalCanonicalizer   19: After phase DeadCodElimination   19: After phase DeadCodElimination   19: After phase DeadCodElimination   19: After phase DeadCodElimination   19: After phase DeoptimizeToGuard   19: Af		73: Before phase LoopSafepointInsertion
31.2 Before phase DeadCodeElimination   31.3 After phase DeadCodeElimination   31.5 Before phase incrementalCanonicalizer   31.5 Before phase Converticepolimize ToGuard   31.6 Before subphase DeadCodeElimination   31.7 After subphase DeadCodeElimination   31.8 After phase LoweringPound   31.7 After subphase DeadCodeElimination   32.3 After phase IncrementalCanonicalizer   32.1 Before subphase ConditionalElimination   32.2 After subphase ConditionalElimination   32.3 After phase LoopFullUnroll   32.5 After phase LoopFullUnroll   32.5 After phase LoopFullUnroll   32.5 After phase LoopFullUnroll   32.6 Before phase IncrementalCanonicalizer   32.7 Before phase LoopFullUnroll   32.8 After phase LoopPullUnroll   32.9 After phase LoopPullUnroll   33.9 Before phase IncrementalCanonicalizer   33.1 Before phase IncrementalCanonicalizer   33.2 After phase LoopPullunroll   33.3 After phase LoopUnswitching   33.3 After phase IncrementalCanonicalizer   34.3 Before phase EarNeadElimination   35.4 After phase PartialEscape   35.5 After phase PartialEscape   36.5 After phase DeadCodeElimination   37.5 After aphyling effects   38.6 Before phase EarNeadElimination   39.6 Before phase EarNeadElimination   39.7 After aphyling effects   39.8 Before phase EarNeadElimination   39.9 Before phase DeadCodeElimination   39.9 Before phase DeadCodeElimination   39.9 Before phase Lowering   39.9 Before phase Lowering   30.8 Before phase DeadCodeElimination   30.9 After phase BincrementalCanonicalizer   30.9 After phase BincrementalCanonicalizer   30.9 After phase BincrementalCanonicalizer   30.9 After phase DeadCodeElimination   30.9 Before phase LoweringRound   31.8 Before phase Endemination   32.8 After phase Endemination   33.9 Before phase Endemination   34.9 After phase Endemination   35.9 Before phase IncrementalCanonicalizer   36.9 After phase Middlier   37.9 After phase BincrementalCanonicalizer   38.9 Before phase Endemination   39.9 Before phase Endemination   39.9 Before phase Endemination   39.9 Before phase Endemination   39.		74: After phase LoopSafepointInsertion
12: Before phase DeadCodeElimination   14: Before phase IncrementalCanonicalizer   15: Before phase IncrementalCanonicalizer   16: Before subphase DeadCodeElimination   17: After subphase DeadCodeElimination   18: After phase DeadCodeElimination   19: After phase IncrementalCanonicalizer   20: De Before phase IncrementalCanonicalizer   20: De Before phase IncrementalCanonicalizer   20: De Before phase IncrementalCanonicalizer   21: Before subphase ConditionalElimination   22: After subphase ConditionalElimination   23: After phase LoopFullUrroll   24: Before phase IncrementalCanonicalizer   27: Before phase IncrementalCanonicalizer   28: After phase LoopFullUrroll   29: After phase LoopFullUrroll   20: Before phase IncrementalCanonicalizer   20: Before phas		75: Before phase Lowering
13: After phase DeadCodeElmination     14: Before phase ConvertDeoptimizeToGuard     15: Before phase DeadCodeElmination     17: After subphase DeadCodeElmination     18: After phase ConvertDeoptimizeToGuard     19: After phase ConvertDeoptimizeToGuard     19: After phase ConvertDeoptimizeToGuard     19: After phase LorementalCanonicalizer     20: Before phase terativeConditionalElmination     21: Before subphase ConditionalElmination     22: After subphase ConditionalElmination     23: After phase LoopFullUrroll     25: After phase LoopFullUrroll     26: Before phase LoopPeeling     27: Before phase LoopPeeling     28: After phase DeopPeeling     29: After phase IncrementalCanonicalizer     31: Before phase LoopDeeling     29: After phase LoopDeeling		
14: Before phase incremental Canonicalizer		
1.5   Before subphase DeadCodeElimination	14: Before phase IncrementalCanonicalizer	
December 20 Supphase DeadCodeElimination December 21 Supphase ConvertDeoptimizeToGuard December 22 Subphase ConvertDeoptimizeToGuard December 22 Subphase Evalue ConditionalElimination December 23 Subphase ConditionalElimination December 24 Subphase Evalue ConditionalElimination December 25 Subphase Evalue ConditionalElimination December 26 Subphase Evalue ConditionalElimination December 27 Subphase Evalue Evalue ConditionalElimination December 28 Subphase Evalue Eva	15: Before phase ConvertDeoptimizeToGuard	
Wear	16: Before subphase DeadCodeElimination	•
<ul> <li>№ 18. After phase ConwertDeoptimizer of Cuard</li> <li>№ 19. After phase IncrementalCanonicalizer</li> <li>№ 20. Before phase IncramentalCanonicalizer</li> <li>№ 22. After subphase ConditionalElimination</li> <li>№ 23. After phase Incompetillumoil</li> <li>№ 24. Before phase LoopFullUmoil</li> <li>№ 25. After phase LoopFullUmoil</li> <li>№ 26. Before phase LoopFullUmoil</li> <li>№ 27. Before phase LoopFullUmoil</li> <li>№ 28. After phase LoopFullUmoil</li> <li>№ 29. After phase LoopFullUmoil</li> <li>№ 31. Before phase IncrementalCanonicalizer</li> <li>№ 31. After phase LoopDrawitching</li> <li>№ 32. After phase LoopUnswitching</li> <li>№ 33. After phase PartialEscape</li> <li>№ 34. Before phase FartialEscape</li> <li>№ 35. After phase PartialEscape</li> <li>№ 36. Before phase EarlyReadElimination</li> <li>№ 40. After subphase DeadCodeElimination</li> <li>№ 40. After subphase Canonicalizer</li> <li>№ 41. Before subphase Canonicalizer</li> <li>№ 42. After subphase Canonicalizer</li> <li>№ 43. After phase EarlyReadElimination</li> <li>№ 44. Before phase Elowering</li> <li>№ 51. After phase EuveringRound</li> <li>№ 48. After subphase IncrementalCanonicalizer</li> <li>№ 49. Sefore phase ExpandLogic</li> <li>№ 10. Before subphase Canonicalizer</li> <li>№ 11. Before phase ExpandLogic</li> <li>№ 10. After phase ExpandLogic</li> <li>№ 11. Before phase ExpandLogic</li> <li>№ 11. After phase ExpandLogic</li> <li>№ 11. Before phase ExpandLogic</li> <li>№ 11. After phase ExpandLogic</li> <li>№ 11. After phase ExpandLogic</li> <li>№ 11. After p</li></ul>	17: After subphase DeadCodeElimination	
Sefore phase IncrementalCanonicalizer		
© 20: Before phase IterativeConditionalElimination © 21: After subphase ConditionalElimination © 22: After subphase ConditionalElimination © 23: After phase IterativeConditionalElimination © 24: After subphase LoopFullUnroll © 25: After phase LoopFullUnroll © 25: After phase LoopFullUnroll © 26: Before phase LoopFullUnroll © 27: Before phase LoopFullUnroll © 28: After phase LoopFullUnroll © 29: After phase LoopFullunroll © 30: Before phase IncrementalCanonicalizer © 31: Before phase LoopFullunroll © 31: Before phase LoopFullunroll © 32: After phase LoopFullunroll © 33: After phase IncrementalCanonicalizer © 34: After phase PartialEscape © 35: After phase PartialEscape © 36: Before phase PartialEscape © 37: After phase DeadCodeElimination © 37: After aphyling effects © 38: Before phase Lowering © 39: Before subphase Canonicalizer © 39: Before subphase Canonicalizer © 39: Before phase Lowering © 31: Before subphase LoweringRound © 41: Before subphase LoweringRound © 42: After subphase LoweringRound © 43: After phase IncrementalCanonicalizer © 49: After phase IncrementalCanonicalizer © 50: Before phase IncrementalCanonicalizer © 51: After phase LowEllimination © 52: Before phase LowEllimination © 53: Before phase LowEllimination © 55: Before phase IncrementalCanonicalizer © 56: Before phase IncrementalCanonicalizer © 57: After phase IncrementalCanonicalizer © 58: Before phase Lowering © 110: After subphase ConditionalElimination © 51: After phase Lowering © 111:		82: After phase OptimizeDiv
© 21: Before subphase ConditionalElimination © 22: After subphase ConditionalElimination © 23: After phase letrativeConditionalElimination © 24: Before phase LoopFullUnroll © 25: Before phase loopFullUnroll © 26: Before phase IncrementalCanonicalizer © 27: Before phase IncrementalCanonicalizer © 30: Before phase IncrementalCanonicalizer © 30: Before phase IncrementalCanonicalizer © 31: Before phase LoopPulswitching © 32: After phase LoopPulswitching © 33: After phase IncrementalCanonicalizer © 34: Before phase LoopUnswitching © 33: After phase IncrementalCanonicalizer © 34: Before phase PartialEscape © 35: After phase PartialEscape © 36: Before phase PartialEscape © 37: After aphyling effects © 38: Before phase PartialEscape © 39: Before phase PartialEscape © 39: Before phase DeadCodeElimination © 40: After subphase DeadCodeElimination © 41: Before subphase Canonicalizer © 42: After subphase DeadCodeElimination © 43: Before subphase Canonicalizer © 43: After phase EarlyReadElimination © 44: Before subphase LoweringRound © 47: After subphase LoweringRound © 48: After subphase LoweringRound © 49: After phase HighTier © 51: After phase IncrementalCanonicalizer © 52: Before phase MidTier © 53: Before phase MidTier © 53: Before phase IncrementalCanonicalizer © 55: Before phase ConditionalElimination © 66: Before subphase ConditionalElimination © 66: After phase DeadCodeElimination © 66: After phase DeadCodeElimination © 66: After phase DeadCodeElimination © 67: After phase DeadCodeElimination © 68: After phase DeadCodeElimination © 75: After phase IncrementalCanonicalizer © 75: B		83: Before phase FrameStateAssignment
22: After subphase ConditionalElimination   22: After phase terrativeConditionalElimination   23: After phase toopFullUnroll   25: After phase LoopFullUnroll   25: After phase LoopFullUnroll   25: After phase LoopFullUnroll   26: Before phase LoopFullUnroll   27: Before phase LoopFullUnroll   28: After phase LoopPelling   29: After phase LoopFullorg   29: After phase LoopUnswitching   29: After phase LoopUnswitching   29: After phase LoopUnswitching   29: After phase LoopUnswitching   29: After phase LoopFullorg   29: After phase MidTier   29: After phase BarthalEscape   29: After phase PartialEscape   29: After phase PartialEscape   29: Before phase Lowering   29: After phase DeadCodeElimination   29: After subphase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase Expredated   29: After subphase DeadCodeElimination   29: After subphase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase IncrementalCanonicalizer   29: Before phase Lowering   29: After phase Expredated   29: After subphase ConditionalElimination   29: After subphase ConditionalElimination   29: After subphase ConditionalElimination   29: After phase IncrementalCanonicalizer   29: Before phase ConditionalEliminat		
23: After phase letrativeConditionalElimination   24: Before phase LoopFullUnroll   25: After phase LoopFullUnroll   26: Before phase LoopFullUnroll   27: Before phase LoopFullUnroll   28: After phase LoopFullunroll   29: After phase LoopFullunroll   29: After phase LoopFullunroll   29: After phase LoopFullinroll   29: After phase DoopFullinroll   29: After phase DoopFullinroll   29: After phase IncrementalCanonicalizer   29: After phase IncrementalCanonicalizer   29: After phase LoopFullinroll   29: After phase WriteBarrierAddition   29: After phase MidTier   29: After phase MidTier   29: After phase Lowering   29: After phase Lowering   29: After phase Lowering   29: After phase Lowering   29: After subphase IncrementalCanonicalizer   29: After subphase IncrementalCanonicalizer   29: After subphase Lowering   29: After phase Place Subphase IncrementalCanonicalizer   29: After phase ExpandLogic   20: After phase ExpandLogic   20: After phase ExpandLogic   20: After phase ExpandLogic   20: After phase		
2 4: Before phase LoopFullUnroll 2 25: After phase LoopFullUnroll 3 25: After phase LoopFullUnroll 3 26: Before phase lncrementalCanonicalizer 3 27: Before phase LoopPeeling 3 28: After phase LoopPeeling 3 28: After phase LoopPeeling 3 29: After phase LoopPeeling 3 30: Before phase LoopUnswitching 3 31: Before phase LoopUnswitching 3 32: After phase LoopUnswitching 3 33: After phase LoopUnswitching 3 34: Before phase LoopUnswitching 3 35: After phase PartialEscape 3 36: Before phase PartialEscape 3 36: Before phase EarlyReadElimination 3 37: After applying effects 3 38: EarlyReadElimination letration 3 39: Before subphase DeadCodeElimination 4 39: Before subphase DeadCodeElimination 4 41: Before subphase Canonicalizer 4 42: After subphase Canonicalizer 4 43: Before subphase Canonicalizer 4 44: Before phase EarlyReadElimination 4 45: Before subphase Canonicalizer 4 47: After subphase Lowering 4 48: After phase Lowering 4 49: After subphase LoweringRound 4 40: After subphase LoweringRound 4 50: After phase HighTier 5 50: After phase HighTier 5 51: After phase LockElimination 5 52: After phase FloatingRead 5 53: Before phase IncrementalCanonicalizer 5 59: Before phase IncrementalCanonicalizer 5 50: After phase FloatingRead 5 50: After phase CandicalElimination 5 60: Before subphase ConditionalElimination 6 60: Before		
25: After phase LoopFullUnroll 26: 26: Before phase IncrementalCanonicalizer 26: 27: Before phase LoopPeeling 28: After phase LoopPeeling 29: After phase LoopUnswitching 29: After phase PartialEscape 29: After phase Lowering 29: After subphase PartialEscape 29: After subphase Lowering 29: Before phase Lowering 29: After subphase Lowering 29: Before phase Explanation 29: After phase Canonicalizer 29: After pha		
Sefore phase Incremental   Sefore phase Incremental   Sefore phase LoopPeeling		
27: Before phase LoopPeeling   28: After phase LoopPeeling   29: After phase IncrementalCanonicalizer   30: Before phase IncrementalCanonicalizer   31: Before phase IncrementalCanonicalizer   32: After phase LoopUnswitching   32: After phase LoopUnswitching   33: After phase LoopUnswitching   34: Before phase PartialEscape   35: After phase PartialEscape   36: Before phase PartialEscape   37: After applying effects   38: EarlyReadElimination   37: After applying effects   38: EarlyReadElimination   37: After applying effects   38: EarlyReadElimination   39: Before subphase DeadCodeElimination   39: Before subphase DeadCodeElimination   39: After phase EarlyReadElimination   39: Before subphase Canonicalizer   39: Before subphase LoweringRound   39: After phase Lowering   39: After phase Lowering   39: After phase Lowering   39: After phase MidTier   39: Before phase WriteBarrierAddition   39: After phase Lowering   39: After phase LowTier   39: Before phase LowTier   39: Befo		
28: After phase LoopPeeling   92: Before phase Canonicalizer   93: After phase IncrementalCanonicalizer   93: After phase Canonicalizer   93: After phase LoopUnswitching   95: After phase LoopUnswitching   96: After phase WriteBarrierAddition   96: After phase MidTier   97: After phase LoopUnswitching   96: After phase MidTier   97: After phase Lowering   97: After phase Schedule   97: After phase Lowering   97: After p	26: Before phase IncrementalCanonicalizer	
29: After phase IncrementalCanonicalizer   30: Before phase IncrementalCanonicalizer   30: Before phase LoopUnswitching   32: After phase LoopUnswitching   33: After phase LoopUnswitching   33: After phase IncrementalCanonicalizer   34: Before phase PartialEscape   35: After phase PartialEscape   36: Before phase PartialEscape   36: Before phase EarlyReadEllmination   37: After applying effects   38: EarlyReadEllmination   39: Before subphase DeadCodeEllmination   39: Before subphase DeadCodeEllmination   39: Before subphase DeadCodeEllmination   30: After subphase Canonicalizer   30: After phase EarlyReadEllmination   30: After subphase Canonicalizer   30: Before subphase IncrementalCanonicalizer   30: Before subphase IncrementalCanonicalizer   30: Before subphase ExpandLogic   30: After phase ExpandL	27: Before phase LoopPeeling	91: After phase DeoptimizationGrouping
№ 30: Before phase IncrementalCanonicalizer       9 31: Before phase LoopUnswitching       95: After phase WriteBarrierAddition         № 31: After phase LoopUnswitching       95: After phase MidTier         № 33: After phase IncrementalCanonicalizer       97: After mid tier         № 35: After phase PartialEscape       98: Before phase Lowering         № 36: Before phase PartialEscape       98: Before phase Lowering         № 37: After aphying effects       99: Before phase Lowering         № 38: EarlyReadElimination iteration       101: Before subphase LoweringRound         № 37: After subphase DeadCodeElimination       102: After subphase LoweringRound         № 41: Before subphase Canonicalizer       104: After subphase Lowering         № 45: Before phase Lowering       105: Before phase ExpandLogic         № 45: Before subphase LoweringRound       107: Before subphase Schedule         № 47: After subphase LoweringRound       108: Before subphase Schedule         № 48: After phase IncrementalCanonicalizer       109: After phase FixReads         № 51: After phase IncrementalCanonicalizer       110: After phase Canonicalizer         № 52: Before phase MidTier       111: Before subphase Canonicalizer         № 52: Before phase MidTier       111: Before phase Canonicalizer         № 53: Before phase IncrementalCanonicalizer       111: Before phase Canonicalizer         № 55: Before phase	28: After phase LoopPeeling	92: Before phase Canonicalizer
93 (Sefore phase IncrementalCanonicalizer 3) 31: Before phase LoopUnswitching 32: After phase LoopUnswitching 33: After phase LoopUnswitching 33: After phase IncrementalCanonicalizer 34: Before phase PartialEscape 36: Before phase PartialEscape 37: After phase PartialEscape 37: After phase PartialEscape 38: Before phase EarlyReadElimination 37: After applying effects 39: Before subphase LoweringRound 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase LoweringRound 44: Before phase LoweringRound 45: After subphase LoweringRound 46: After subphase LorementalCanonicalizer 46: Before subphase LoweringRound 47: After phase LoweringRound 48: After subphase LoweringRound 49: After phase HighTier 49: After phase LoweringRound 49: After phase HighTier 49: After phase Lowering 49: After phase HighTier 49: After phase Lowering 49: After phase MidTier 49: After phase LockElimination 49: 53: Before phase LockElimination 49: 55: Before phase IncrementalCanonicalizer 49: After phase DeadCodeElimination 49: 55: Before phase IrcrementalCanonicalizer 49: After phase PropagateDeoptimizeProbability 49: After phase PropagateDeoptimizeProbability 49: After phase IncrementalCanonicalizer 49: After phase IrcrementalCanonicalizer 49: After phase IrcrementalCanonicalizer 49: After phase PropagateDeoptimizeProbability 49: After phase PropagateDeoptimizeProbability 49: After phase PropagateDeoptimizeProbability 49: After phase PropagateDeoptimizeProbability 49: After phase Schedul		93: After phase Canonicalizer
95: After phase WriteBarrierAddition   96: After phase WriteBarrierAddition   97: After phase MidTier   97: After phase IncrementalCanonicalizer   97: After mid tier   97: After mid tier   98: Before phase PartialEscape   98: Before phase Lowering   99: Before phase Lowering   100: Before subphase Lowering   100: Before subphase Lowering   103: After subphase Lowering   104: After phase Lowering   105: Before phase ExpandLogic   106: After phase ExpandLogic   106: After phase ExpandLogic   107: Before phase ExpandLogic   108: Before phase Schedule   109: After phase Schedule   109: After subphase Schedule   109: After subphase Schedule   109: After phase ExpandLogic   110: After phase FixReads   110: After phase ExpandLogic   110: After phase FixReads   110: After phase ExpandLogic   110: After phase FixReads   110: After phase	30: Before phase IncrementalCanonicalizer	
32: After phase LoopUnswitching 33: After phase IncrementalCanonicalizer 33: After phase PartialEscape 35: After phase PartialEscape 36: Before phase EarlyReadElimination 37: After applying effects 38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before subphase Canonicalizer 45: Before subphase IncrementalCanonicalizer 46: Before subphase IncrementalCanonicalizer 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After phase LoweringRound 49: After phase LoweringRound 50: After phase LoweringRound 50: After phase LoweringRound 51: After phase LoweringRound 52: Before phase IncrementalCanonicalizer 53: Before phase Lokelimination 54: After phase Lokelimination 55: Before phase Lokelimination 55: Before phase Lokelimination 55: Before phase Lokelimination 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase FloatingRead 57: After phase PropagateDeoptimizeProbability 58: After phase IncrementalCanonicalizer 59: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase Lokelimination 50: Sefore phase Lokelimination 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase Lokelimination 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase DeadCodelimination 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase DeadCodelimination 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase DeadCodelimination 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase Increment		
33: After phase IncrementalCanonicalizer 34: Before phase PartialEscape 35: After phase EarlyReadElimination 37: After applying effects 38: EarlyReadElimination iteration 39: 38: EarlyReadElimination iteration 40: After subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After subphase Lowering 44: Before subphase Lowering 45: Before subphase LoweringRound 47: After subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase LoweringRound 49: After subphase LoweringRound 40: After phase LoweringRound 40: After phase LoweringRound 40: After phase LoweringRound 41: Before subphase LoweringRound 42: Before subphase LoweringRound 43: After subphase LoweringRound 44: Before subphase LoweringRound 45: Before subphase LoweringRound 46: Before subphase LoweringRound 47: After subphase IncrementalCanonicalizer 49: After phase Lowering 49: After phase Lowering 49: After phase Lowering 40: After phase Lowerin		
34: Before phase PartialEscape 35: After phase PartialEscape 36: Before phase EarlyReadElimination 37: After applying effects 38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase Canonicalizer 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before subphase Lowering 44: Before subphase Lowering 45: Before subphase Lowering 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After subphase IncrementalCanonicalizer 49: After subphase IncrementalCanonicalizer 49: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After phase HighTier 52: Before phase MotTier 53: Before phase MotTier 53: Before phase IncrementalCanonicalizer 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 57: After phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase RetrativeConditionalElimination 50: Before subphase ConditionalElimination 50: Before phase RetrativeConditionalElimination 50		The state of the s
35: After phase PartialEscape 36: Before phase EarlyReadElimination 37: After applying effects 38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase IncrementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 50: S1: After phase HighTier 51: After phase LockElimination 52: Before phase MidTier 53: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 56: S6: Before phase IncrementalCanonicalizer 57: After phase IncrementalCanonicalizer 58: S8: Before phase IncrementalCanonicalizer 59: S8: Before phase IncrementalCanonicalizer 50: S8: Before phase Incrementa		
36: Before phase EarlyReadElimination 37: After applying effects 38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase Lowering 46: Before subphase Lowering 47: After subphase Lowering 48: After subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase LoweringRound 50: After phase EarlyReadElimination 50: After phase HighTier 51: After phase HighTier 52: Before phase MidTier 53: Before phase MidTier 55: Before phase LockElimination 55: After phase LockElimination 56: Before phase IncrementalCanonicalizer 57: After phase LockElimination 58: After phase FloatingRead 59: Sefore phase IncrementalCanonicalizer 59: Sefore phase IncrementalCanonicalizer 50: After phase PloatingRead 50: Before phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: After phase PloatingRead 50: After phase PloatingRead 51: Before phase IncrementalCanonicalizer 52: After phase PloatingRead 53: After phase PloatingRead 54: After phase PloatingRead 55: After phase PloatingRead 56: After phase ConditionalElimination 56: Before phase IterativeConditionalElimination 56: After phase ConditionalElimination 56: After phase ConditionalElimination 57: After phase ConditionalElimination 58: After phase ConditionalElimination 59: After phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 50		
37: After applying effects 38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase Canonicalizer 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before subphase Lowering 45: Before subphase IncrementalCanonicalizer 46: Before subphase Lowering 47: After subphase LoweringRound 47: After subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After phase Lowering 50: After phase Lowering 50: After phase HighTier 51: After phase HighTier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase IncrementalCanonicalizer 59: Sefore phase IncrementalCanonicalizer 59: Sefore phase IncrementalCanonicalizer 59: Sefore phase IncrementalCanonicalizer 59: Sefore phase FloatingRead 50: After phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: After phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: Sefore phase FloatingRead 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase FloatingRead 50: S		
38: EarlyReadElimination iteration 39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase Lowering 45: Before subphase LoweringRound 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase Lowering 50: After phase IncrementalCanonicalizer 49: After phase Lowering 50: After phase MidTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 57: After phase FloatingRead 57: After phase FloatingRead 59: Sefore phase IncrementalCanonicalizer 50: Sefore phase Sendule 50: Sefore phase IncrementalCanonicalizer 50: Sefore phase IncrementalCanonic		
39: Before subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase DeadCodeElimination 40: After subphase Lowering 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 42: After phase EarlyReadElimination 43: After phase ExpandLogic 43: After phase ExpandLogic 43: After phase ExpandLogic 44: Before phase Lowering 45: Before subphase IncrementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase Lowering 50: After phase IncrementalCanonicalizer 49: After subphase Schedule 100: After phase ExpandLogic 106: After phase ExpandLogic 107: Before phase Schedule 109: After subphase IncrementalCanonicalizer 100: After phase ExpandLogic 100: After subphase Canonicalizer 109: After phase ExpandLogic 100: After phase ExpandLogic 100: After phase ExpandLogic 100: After subphase Canonicalizer 109: After phase ExpandLogic 100: After phase ExpandLogic 100: After subphase Canonicalizer 109: After phase ExpandLogic 100: After phase ExpandLogic 100: After subphase Canonicalizer 109: After subphase IncrementalCanonicalizer 109: After subphase IncrementalCanonicalizer 109: After phase ExpandLogic 100: After subphase ExpandLogic 100: After subphase Canonicalizer 109: After subphase AddressLowering 110: After subphase AddressLowering 111: Before phase AddressLowering 111: Before phase UseTrappingNullChecks 111: After phase DeadCodeElimination 112: After phase DeadCodeElimination 113: After phase DeadCodeElimination 114: After phase DeadCodeElimination 115: Before phase IncrementalCanonicalizer 116: After phase DeadCodeElimination 117: Before phase IncrementalCanonicalizer 118: After phase Propagat		101: Before subphase LoweringRound
40: After subphase DeadCodeElimination 41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After subphase Canonicalizer 44: Before phase EarlyReadElimination 44: Before phase Lowering 45: Before phase Lowering 46: Before subphase IncrementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase MidTier 53: Before phase MidTier 53: Before phase MidTier 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 55: Before phase IncrementalCanonicaliz		102: After subphase LoweringRound
41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase Lowering 45: Before subphase LoweringRound 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase IncrementalCanonicalizer 49: After subphase IncrementalCanonicalizer 49: After subphase IncrementalCanonicalizer 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 57: After phase FloatingRead 58: After phase FloatingRead 59: Before phase IncrementalCanonicalizer 59: Before phase FloatingRead 50: After phase PropagateDeoptimizeProbability 50: After phase IncrementalCanonicalizer 59: Before phase IterativeConditionalElimination 50: Before phase IncrementalCanonicalizer 50: After phase IncrementalCanonicalizer 50: After phase IncrementalCanonicalizer 50: After phase IncrementalCanonicalizer 50: Before phase IncrementalCanonicalizer 50:	39: Before subphase DeadCodeElimination	103: After subphase IncrementalCanonicalizer
41: Before subphase Canonicalizer 42: After subphase Canonicalizer 43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase lowering 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 49: After subphase loweringRound 49: After subphase lowering 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 55: Before phase Schedule 56: After phase IncrementalCanonicalizer 57: After phase IncrementalCanonicalizer 58: Before phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 50: Before phase IncrementalC	40: After subphase DeadCodeElimination	
43: After phase EarlyReadElimination 44: Before phase Lowering 45: Before subphase lorementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 48: After subphase LoweringRound 49: After phase Lowering 50: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase LockElimination 55: Before phase LockElimination 56: Before phase FloatingRead 57: After phase FloatingRead 58: After phase IterativeConditionalElimination 60: Before subphase ConditionalElimination 61: After phase IterativeConditionalElimination 62: After phase IterativeConditionalElimination 63: After phase IterativeConditionalElimination 64: After phase IterativeConditionalElimination 65: After phase IterativeConditionalElimination 66: After phase LockTimination 66: After phase IterativeConditionalElimination 66: After phase Schedule 67: After phase IterativeConditionalElimination 68: After phase LockTimination 69: After phase Schedule 60: After phase IterativeConditionalElimination 60: After phase ExpandLoge 60: Af	41: Before subphase Canonicalizer	
44: Before phase Lowering 45: Before subphase LoweringRound 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 55: Before phase LockElimination 55: Before phase LockElimination 55: Before phase FloatingRead 57: After phase FloatingRead 58: Before phase DeadCodeElimination 59: S8: After phase FloatingRead 59: Before phase PropagateDeoptimizeProbability 59: Before phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 59: Before phase ConditionalElimination 60: Before subphase ConditionalElimination 61: After phase IterativeConditionalElimination 62: After phase IterativeConditionalElimination 63: After phase IterativeConditionalElimination 64: After phase IterativeConditionalElimination 65: After phase IterativeConditionalElimination 66: After phase IterativeConditionalElimination 67: After phase IterativeConditionalElimination 68: After phase Schedule 69: After phase IterativeConditionalElimination 60: After phase FixReads 60: After phase Canonicalizer 60: After phase AddressLowering 60: After phase AddressLowering 60: After phase AddressLowering 60: After phase DeadCodeElimination 60: After phase DeadCodeElimination 60: After phase PropagateDeoptimizeProbability 6	42: After subphase Canonicalizer	
44: Before phase Lowering 45: Before subphase IncrementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase Lowering 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 57: After phase FloatingRead 58: After phase PropagateDeoptimizeProbability 59: Before phase IncrementalCanonicalizer 50: After phase IncrementalCanonicalizer 50: Aft	43: After phase EarlyReadElimination	
45: Before subphase IncrementalCanonicalizer 46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 50: 112: After phase PropagateDeoptimizeProbability 50: 122: After phase InsertMembars 50: 123: Before phase Schedule 50: After phase Schedule 51: After phase Schedule 52: After phase LowTier	44: Before phase Lowering	
46: Before subphase LoweringRound 47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase IncrementalCanonicalizer 59: Before phase IterativeConditionalElimination 60: Before subphase ConditionalElimination 61: After subphase Schedule 110: After phase Canonicalizer 111: Before phase AddressLowering 114: After phase UseTrappingNullChecks 115: Before phase UseTrappingNullChecks 116: After phase UseTrappingNullChecks 117: Before phase DeadCodeElimination 118: After phase DeadCodeElimination 119: After phase DeadCodeElimination 110: After phase AddressLowering 112: After phase UseTrappingNullChecks 113: Before phase UseTrappingNullChecks 116: After phase DeadCodeElimination 117: Before phase DeadCodeElimination 119: After phase DeadCodeElimination 119: After phase UseTrappingNullChecks 110: After phase UseTrappingNullChecks 111: Before phase UseTrappingNullChecks 111: Before phase UseTrappingNullChecks 111: Before phase UseTrappingNullChecks 112: Before phase DeadCodeElimination 113: After phase DeadCodeElimination 114: After phase DeadCodeElimination 115: Before phase PropagateDeoptimizeProbability 116: After phase DeadCodeElimination 117: Before phase PropagateDeoptimizeProbability 118: After phase PropagateDeoptimizeProbability 119: Before phase PropagateDeoptimizeProbability 119: Before phase InsertMembars 119: Before phase InsertMembars 110: After phase InsertMembars 110: After phase Canonicalizer 112: After phase UseTrappingNullChecks 111: Before phase UseTrappingNullChecks 111: Before phase InsertMembars 112: After phase InsertMembars 113: Before phase InsertMembars 114: After phase InsertMembars 115: Before phase InsertMembars 116: After phase InsertMembars 117: Before phase InsertMembars 118: After phase Inser		
47: After subphase LoweringRound 48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase Schedule 50: After phase Canonicalizer 112: After phase UseTrappingNullChecks 116: After phase UseTrappingNullChecks 117: Before phase DeadCodeElimination 118: After phase DeadCodeElimination 119: Before phase PropagateDeoptimizeProbability 120: After phase PropagateDeoptimizeProbability 121: Before phase IncrementalCanonicalizer 122: After phase InsertMembars 123: Before phase Schedule 124: After phase Schedule 125: After phase LowTier		
48: After subphase IncrementalCanonicalizer 49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase IncrementalCanonicalizer 57: After phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase IterativeConditionalElimination 50: Before subphase ConditionalElimination 50: Before phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 50: After phase IterativeConditionalElimination 51: Before phase UseTrappingNullChecks 51: Before phase UseTrappingNullChecks 51: After phase DeadCodeElimination 51: Before phase DeadCodeElimination 51: Before phase DeadCodeElimination 51: Before phase DeadCodeElimination 51: Before phase IterativeConditionalElimination 52: After phase PropagateDeoptimizeProbability 53: After phase InsertMembars 54: After phase InsertMembars 55: Before phase InsertMembars 56: Before phase IterativeConditionalElimination 56: Before phase IterativeConditionalElimination 57: After phase Schedule 58: After phase IterativeConditionalElimination 59: Before phase LockElimination 50: Before phase IterativeConditionalElimination 50: Before phase IterativeConditionalElimination 50: Before phase Canonicalizer 51: Before phase PropagateDeoptimizeProbability 58: After phase InsertMembars 59: Before phase Schedule 50: After phase Schedule 50: After phase LockElimination 50: Before phase IterativeConditionalElimination 50: Before phase LockElimination 50: Before phase InsertMembars 50: Before phase IterativeConditionalElimination 50: Before ph		
49: After phase Lowering 50: After phase HighTier 51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase LockElimination 55: Before phase LockElimination 56: Before phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 50: After phase C		111: Before phase Canonicalizer
So: After phase HighTier  So: After phase HighTier  So: After high tier  So: Before phase MidTier  So: Before phase LockElimination  So: Before phase IncrementalCanonicalizer  So: Before phase FloatingRead  So: After phase FloatingRead  So: Before phase FloatingRead  So: Before phase IncrementalCanonicalizer  So: Before phase PropagateDeoptimizeProbability  So: After phase InsertMembars  So: Before phase InsertMembars  So: Before phase InsertMembars  So: Before phase InsertMembars  So: Before phase IncrementalCanonicalizer  So: Before phase InsertMembars  So: Before phase IncrementalCanonicalizer  So: After phase InsertMembars  So: Before phase IncrementalCanonicalizer  So: After phase InsertMembars  So: Before phase InsertMemba		112: After phase Canonicalizer
51: After high tier 52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase Schedule 50: After phase InsertMembars 50: After phase InsertMembars 50: After phase Schedule 50: After phase InsertMembars 50: After phase Schedule 50: After phase InsertMembars 5123: Before phase Schedule 5124: After phase Schedule 5125: After phase LowTier		113: Before phase AddressLowering
52: Before phase MidTier 53: Before phase LockElimination 54: After phase LockElimination 55: Before phase IncrementalCanonicalizer 56: Before phase FloatingRead 57: After phase FloatingRead 57: After phase FloatingRead 58: After phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 59: Before phase IncrementalCanonicalizer 60: Before subphase ConditionalElimination 60: Before subphase ConditionalElimination 61: After subphase ConditionalElimination 62: After phase IterativeConditionalElimination 63: After phase IterativeConditionalElimination 64: After phase IterativeConditionalElimination 65: After phase IterativeConditionalElimination 66: After phase IterativeConditionalElimination 67: After phase IterativeConditionalElimination 68: After phase IterativeConditionalElimination 69: After phase IterativeConditionalElimination 60: After phase IterativeConditionalElimination		
53: Before phase LockElimination  54: After phase LockElimination  55: Before phase IncrementalCanonicalizer  56: Before phase FloatingRead  57: After phase FloatingRead  58: After phase FloatingRead  59: Before phase IncrementalCanonicalizer  59: Before phase IncrementalCanonicalizer  59: Before phase IncrementalCanonicalizer  60: Before subphase ConditionalElimination  60: Before subphase ConditionalElimination  60: After phase InsertMembars  122: After phase InsertMembars  123: Before phase Schedule  124: After phase Schedule  125: After phase LowTier		
54: After phase LockElimination  55: Before phase IncrementalCanonicalizer  56: Before phase FloatingRead  57: After phase FloatingRead  58: After phase IncrementalCanonicalizer  59: Before phase IncrementalCanonicalizer  59: Before phase IncrementalCanonicalizer  59: Before phase IterativeConditionalElimination  60: Before subphase ConditionalElimination  61: After subphase ConditionalElimination  62: After phase IterativeConditionalElimination  63: After phase IterativeConditionalElimination  64: After phase IterativeConditionalElimination  65: After phase IterativeConditionalElimination  66: After phase IterativeConditionalElimination  67: After phase IterativeConditionalElimination  68: After phase DeadCodeElimination  118: After phase PropagateDeoptimizeProbability  120: After phase InsertMembars  121: Before phase InsertMembars  122: After phase InsertMembars  123: Before phase Schedule  124: After phase Schedule  125: After phase LowTier		
<ul> <li>55: Before phase IncrementalCanonicalizer</li> <li>56: Before phase FloatingRead</li> <li>57: After phase FloatingRead</li> <li>58: After phase IncrementalCanonicalizer</li> <li>59: Before phase IterativeConditionalElimination</li> <li>60: Before subphase ConditionalElimination</li> <li>61: After subphase ConditionalElimination</li> <li>62: After phase IterativeConditionalElimination</li> <li>63: After phase IterativeConditionalElimination</li> <li>64: After phase IterativeConditionalElimination</li> <li>65: After phase IterativeConditionalElimination</li> <li>65: After phase IterativeConditionalElimination</li> <li>65: After phase IterativeConditionalElimination</li> <li>66: After phase IterativeConditionalElimination</li> <li>67: After phase IterativeConditionalElimination</li> <li>68: After phase IterativeConditionalElimination</li> <li>69: After phase IterativeConditionalElimination</li> <li>60: Before phase InsertMembars</li> <li>60: Before phase InsertMembars</li> <li>60: Before phase IterativeConditionalElimination</li> <li>60: Before phase IterativeConditionalElimination</li> <li>60: Before phase InsertMembars</li> <li>60: Bef</li></ul>		
№ 56: Before phase FloatingRead       № 119: Before phase PropagateDeoptimizeProbability         № 57: After phase FloatingRead       № 120: After phase PropagateDeoptimizeProbability         № 58: After phase IncrementalCanonicalizer       № 121: Before phase InsertMembars         № 59: Before phase IterativeConditionalElimination       № 122: After phase InsertMembars         № 60: Before subphase ConditionalElimination       № 123: Before phase Schedule         № 61: After subphase ConditionalElimination       № 124: After phase Schedule         № 62: After phase IterativeConditionalElimination       № 125: After phase LowTier		
<ul> <li>57: After phase FloatingRead</li> <li>58: After phase IncrementalCanonicalizer</li> <li>59: Before phase IterativeConditionalElimination</li> <li>60: Before subphase ConditionalElimination</li> <li>61: After subphase ConditionalElimination</li> <li>62: After phase IterativeConditionalElimination</li> <li>62: After phase IterativeConditionalElimination</li> <li>120: After phase PropagateDeoptimizeProbability</li> <li>121: Before phase InsertMembars</li> <li>122: After phase Schedule</li> <li>123: Before phase Schedule</li> <li>124: After phase Schedule</li> <li>125: After phase LowTier</li> </ul>	55: Before phase IncrementalCanonicalizer	
№ 57: After phase FloatingRead       № 120: After phase PropagateDeoptimizeProbability         № 58: After phase IncrementalCanonicalizer       № 121: Before phase InsertMembars         № 59: Before phase IterativeConditionalElimination       № 122: After phase InsertMembars         № 60: Before subphase ConditionalElimination       № 123: Before phase Schedule         № 61: After subphase ConditionalElimination       № 124: After phase Schedule         № 62: After phase IterativeConditionalElimination       № 125: After phase LowTier	56: Before phase FloatingRead	119: Before phase PropagateDeoptimizeProbability
Image: Second transport of the problem of the pro		120: After phase PropagateDeoptimizeProbability
№ 59: Before phase IterativeConditionalElimination       № 122: After phase InsertMembars         № 60: Before subphase ConditionalElimination       № 123: Before phase Schedule         № 61: After subphase ConditionalElimination       № 124: After phase Schedule         № 62: After phase IterativeConditionalElimination       № 125: After phase LowTier		
60: Before subphase ConditionalElimination  61: After subphase ConditionalElimination  62: After phase IterativeConditionalElimination  63: After phase IterativeConditionalElimination  64: After phase Schedule  65: After phase LowTier		
<ul> <li>61: After subphase ConditionalElimination</li> <li>62: After phase IterativeConditionalElimination</li> <li>124: After phase Schedule</li> <li>125: After phase LowTier</li> </ul>		
62: After phase IterativeConditionalElimination 125: After phase LowTier		
© 63: Before phase LoopSatepointElimination		
	<b>७</b> ७४: Before phase LoopSafepointElimination	126: After low tier

图11:对Foo.add()所做的处理

了前面课程的铺垫,你现在看它们应该就消除了一些陌生感。

另外,你会发现,在这些处理阶段中,有一个Canonicalizer的阶段出现了好几次,并且你可能对这个词也 比较陌生,所以下面我们不妨来看看,这个阶段都做了些什么。

#### 规范化(Canonicalizer)

Canonicalize的意思是规范化。如果某段程序有多种写法,那么编译器会处理成一种统一的、标准的写法。

比如,对于下面这个简单的函数,它是把a乘以2。在CanonicalizerPhase运行之后,乘法运算被替换成了 移位运算,也就是a<<1。它的效果与乘以2是相同的,但运行效率更高。

```
public int doDouble(int a){
   return 2*a;
}
```

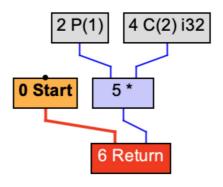


图12:未做规范化优化之前,是乘法运算

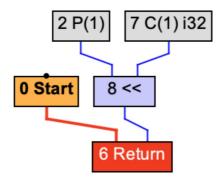


图13: 做完规范化优化之后,变成移位运算

你还可以试一下对某个变量取两次负号的操作。在规范化阶段以后,两个负号就会被去掉,直接返回a。

```
public int negneg(int a){
  return -(-a);
}
```

规范化需要的操作,都是对本节点进行修改和替换,一般都不太复杂。某节点如果实现了Canonicalizable 接口,在CanonicalizerPhase就会对它做规范化。

在规范化阶段实现的优化算法包括:常数折叠(Constant Folding)、强度折减(Strength reduction)、全局值编号(Global Value Numbering,GVN),等等。它们的原理,我在<mark>第7讲</mark>都介绍过,这里就不赘述了。

## 课程小结

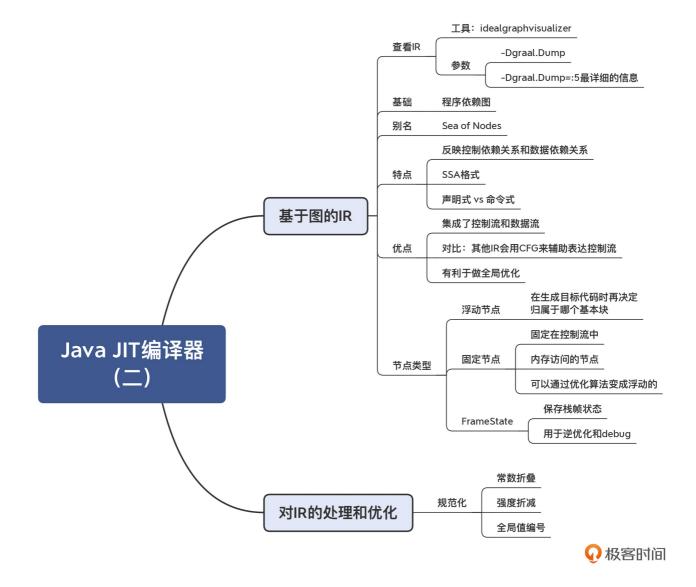
这一讲,我给你介绍了Graal的IR:它整合了控制流图与数据流图,符合SSA格式,有利于优化算法的编写和维护。

我还带你了解了对IR的一个优化处理过程:规范化。规范化所需要的操作一般并不复杂,它都是对本节点进行修改和替换。在下一讲中,我会带你分析另外两个重要的算法,内联和逃逸分析。

另外,Graal的IR格式是声明式的(Declarative),它通过描述一个节点及其之间的关系来反映源代码的语义。而我们之前见到的类似<u>三地址代码</u>那样的格式,是命令式的(Imperative),它的风格是通过命令直接告诉计算机,来做一个个的动作。

声明式和命令式是编程的两种风格,在Graal编译器里,我们可以看到声明式的IR会更加简洁,对概念的表达也更加清晰。我们在后面介绍MySQL编译器的实现机制当中,在讲DSL的时候,还会再回到这两个概念,到时你还会有更加深刻的认识。

本讲的思维导图我也放在了这里,供你参考:



## 一课一思

了解了Graal IR的特点以后,通过对比我们在第7讲中学过的优化算法,你觉得哪些优化算法在Graal IR上实现起来会更方便?为什么?欢迎在留言区分享你的看法。

如果你觉得有收获,也欢迎你把今天的内容分享给更多的朋友。

## 参考资料

基于图的IR,有三篇论文必须提到:

- 1. 程序依赖图: J. Ferrante, K. J. Ottenstein, and J. D. Warren. The program dependence graph and its use in optimization. July 1987。有关程序依赖图的概念在1987年就提出来了。
- 2. Click的论文: <u>A Simple Graph-Based Intermediate Representation</u>。这篇文章比较易读,属于必读文献。Click还发表了一些论文,讲述了基于图的IR上的优化算法。
- 3. 介绍Graal IR的论文: <u>Graal IR: An Extensible Declarative Intermediate Representation</u>。这篇论文也很易读,建议你一定要读一下。