

Coroutines

coro.patch

This patch provides simple stackful, first-class symmetric and asymmetric coroutines for the HotSpot JVM.

It's currently meant to be used on its own, without the other patches that are part of the mlvm project. This means that the guards to exclude the other patches need to be given to the qselect command (/indy, /meth, etc.).

Apart from the changes to the HotSpot VM there are also a number of new classes in the java/dyn package, and some changes to the java.lang.Thread class, so it's necessary to build the JDK or compile and prepend (bootclasspath/p) the classes to the HotSpot boot class path.

Symmetric coroutines are implemented via the Coroutine class. This is used by either subclassing Coroutine and overriding the run method or passing a target Runnable to the Coroutine constructor.

Coroutine.java

```
public class Coroutine {
    public Coroutine();
    public Coroutine(Runnable target);
    public Coroutine(long stacksize);
    public Coroutine(Runnable target, long stacksize);

    public static void yield();
    public static void yieldTo(Coroutine target);

    protected void run();
}
```

Asymmetric coroutines have a more complicated API (through the class AsymCoroutine), for a number of reasons:

- they are prepared to take an input value and return an output value
- these input and output values are of generic type
- the caller/callee relationship of asymmetric coroutines leads to a differentiation between "calling" a coroutine and "returning" from it
- it implements Iterable so it can be used in for-each loops

AsymCoroutine.java

```
public abstract class AsymCoroutine<InT, OutT> implements Iterable<OutT>
{
    public AsymCoroutine();
    public AsymCoroutine(long stacksize);

    public InT ret(OutT value);
    public InT ret();
    public OutT call(InT input);
    public OutT call();

    protected abstract OutT run(InT value);

    @Override
    public Iterator<OutT> iterator();
}
```

CoroutineLocals work like ThreadLocals with the exception that they are scoped to Coroutines.