Porting the GNAT Tasking Runtime System to the Java Virtual Machine

9 June 1998

Laurent Milliet (ENST), Ted Baker (FSU)
Outline

1. Background
2. Mapping C to JVM
3. Prototyping using JNI
4. Conclusions
allows calling Java methods from C-language code

Java Native Interface (JNI)

Java port of tasking = target of GNU

GNU: low-level task primitives for GNU

GNAT: higher-level primitives for GNU

GNAT: Ada compiler

Precedent: Intermetrics Ada-to-JVM compiler

Java Virtual Machine (JVM)

Background
Implements these in terms of JVM objects and operations.

GUJULI Task Primitive Operations

• Yield
• Set-Priority
• Step, Step-Med, Step-Make
• Write-Lock, UnLock
• SelF
• Abort-Task
• Create-Task
available methods on ATCB include synchronization
derived from Java. Task. Object
• Ada Task Control Block (ATCB) is ordinary Java object
or any class with run method
• use constructor of Java. Thread. Thread

Create Task
Abort occurs only at abort completion polling points

- it body of Abort-Task is null
- JVM has no preemptive abort operation

Abort-Task
With data component that points to ATCB

- Ada task = Java thread
- Java.Lang.Thread.currentThread
- Self
Ada bindings for C Interfaces:
Lessons Learned from the First Implementation

- GNU ML emulates using Thread.setPriority
due to ATCB
- no priority ceiling locking in Java
- hence to ATCB
- monitor.monitor_exit, monitor_exit apply to all objects

writeLock, unLock
Lessons learned from the first implementation:

- Apply to all Java objects, including ATCBS
- Java, Java, Object, Notify, Wakeup
- Java, Java, Object, Wait, Sleep
- Java, Java, Object, Wait, Sleep

Steep, Steep, Steep, Steep, Steep
Java, Java, Thread, Thread Yield

or precisely define the effects of priorities, though

Java does not require priority scheduling support.

Java, Java, Thread, setPriority, setPriority

Scheduling Support
This is only a temporary implementation, for testing.

JVM layer (Java)

JNI

AJI layer (C)

GNUIL layer (Ada)

GNUIL

GNARL layer (Ada)

How to test tasking port without JVM byte-code code compiler?

Prototyping using JNI

Laurent Milliet (ENST), Ted Baker (FSU)
gives thread context in which to execute method

Java Environment pointer

Ada Bindings for C Interfaces:

AdaEurope'98

9 June 1998

Lessons Learned from the First Implementation

Laurent Millier (ENST), Ted Baker (FSU)
```java
{
    System.loadLibrary("wrapper");
}

private native void wrap(int proc, int self-tid);
{
    wrap (proc, self-tid);
}

public void run
{
    this.self-tid = self-tid;
    this.proc = proc;
}

public wrapper (int proc, int self-tid);
private int proc, self-tid;
}

public class wrapper extends Thread

package ALL;

Task Creation
```
- Annex D tests
- 49 Ada 95 tests
- 189 Ada ACVC 83 tests

As of December, passed all basic tasking-related ACVC tests

Tasking "system calls" are all through the JVM, via JNI

Ada programs are compiled to native SPARC object code

Prototype Implementation works

Conclusion

Laurent Millier (ENST), Ted Baker (FSU)
Such as this one, is a prerequisite

• A Java-compatible tasking runtime system,

• Demonstrates feasibility of Interface via JNI

Further Conclusions