VM List:

Current items in play:

i. 64-bit Spur (simulation of 64-bit VM & SpurOM & image support development)
ii. 64-bit vm support (BochsX64Plugin, IA64???, gdb-7.10)
iii. ARM (32-bit simulation)
iv. multi-threaded VM discussions: capability boundaries & design
v. Pharo (generation & simulation & fork elimination)

We have a huge amount of work to do on Cog:

1. event-driven VM (that hence costs 0% processor time at idle)
2. Pharo modernization to latest Cog code – 32-bit
3. 32-bit ARM Simulator
    a) 32-bit Pharo ARM
4. 64-bits (x64 and ARM and...?)
    a) BochsX64Plugin
    b) 64-bit ARM Simulator
    c) Pharo 64-bit Spur VM
5. Sista adaptive optimizer
6. FFI via dynamic generation of marshaling code, as required for efficient and correct call outs on x64
7. MTVM as defined above
8. an incremental global mark-sweep GC for Spur
9. running on Xen/Unikernels/containers
10. providing a JavaScript plugin to proved rendering and events so we can run an efficient VM in a web browser
11. a port of the Interpreter/Context VM to Spur

IMO, things that can /and should/ wait are

- throwing away Slang and providing a true written-in-pure-Smalltalk VM that is self-bootstrapped a la Gerardo Richarte and Xavier Burroni
- a truly parallel multi/threaded VM

and things we shouldn't go anywhere near are

- using libffi
- targeting JavaScript, Java or any other dynamic language de jour that happens to run in a web browser but either provides abysmal performance or doesn't support full Smalltalk semantics
- implementing the VM in other VM frameworks such as PyPy which simply strengthens that community and weakens our own

Breakout: Multi-Threaded Virtual Machine

I'm defining the MTVM to be the sharing of the VM between threads, and /not/ just the use of threads to implement non-Smalltalk sub tasks of the VM, and /not/ a full-blown multithreaded Smalltalk VM providing concurrent execution of Smalltalk processes in parallel.

Breakout: ARM Simulation – 32-bit/64-bit architectures

I've ported the latest gdb version of the ARM simulator (gdb-7.10), which now includes support for VFP and other instructions that were missing from the gdb-7.6 version. That's now in the queue awaiting final check and inclusion in the distribution. But there may likely be more debug work to get it running as well as the Bochs IA32 simulator.

I did run into issues trying to build the linux32x86 version on my 64-bit Ubuntu system: there are a number of system libraries that don't have support for multiarch which I had to include local version of, and I had to put in some "-m32" flags in the config scripts to force the build to be 32-bit. Once the 64-bit Spur is stable, there will have to be some work to parameterize the simulator builds to build either 32-bit or 64-bit versions.