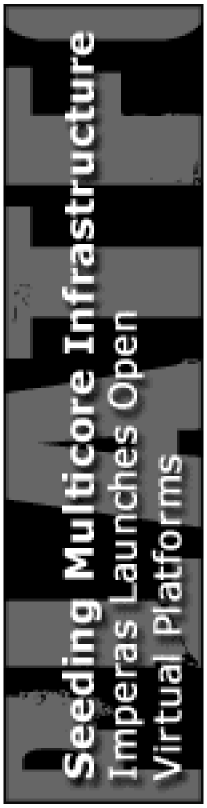
Embedded Technology JOURNAL
TRENDS
TECHNOLOGY
TOOLS
TECHNIQUES
www.embeddedtechjournal.com



Seeding Multicore Infrastructure

Imperas Launches Open Virtual Platforms

by Bryon Moyer, Embedded Technology Journal

Seeding a saturated solution for optimal crystal growth can be a tricky business. The highest-quality, largest crystals grow when given lots of time for the molecules to orient themselves in the lattice. Seeding too late can result in chaotic explosive nucleation, small granularity, and low quality. Seed too early, and, well, there may not technically be a problem, but being an impatient species, if we don't see crystal growth quickly enough, we tend to get bored and move the seed elsewhere.

Saturation is something of a measure of potential, of pent-up demand. There is more and more willingness to orient along coordinated lines, but that initial seed is missing, around which everything can congregate. The multicore market, while ballyhooed for some time now, has grown slowly because of the need for new programming models and development infrastructure. Demand has existed, but it hasn't been compelling enough to drive a robust commercial marketplace. Those companies that were either forced into multicore or saw first-mover opportunities there have tended to create their own tools, keeping them proprietary for convenience or competitive advantage. Things are very fractured, with few clear trends.

We're getting to the point now, however, where embedded multicore is going to have to step out of the shadowy corners and take on mainstream status. Demand has slowly built for common infrastructural elements around which to build a toolchain. In particular, there's a need for a way to validate and debug software programs before the hardware is available. Hardware simulators are too slow and provide more precision than is needed for most application development; a higher-level simulation model can provide accurate enough behavior at reasonable performance. While it's too expensive for everyone to do their own from scratch, everyone seems to be waiting for someone else to go first.

Imperas has decided that this means the solution is now saturated enough to warrant seeding some crystallization, but not super-saturated to the extent that chaos would ensue. It's still early enough for alignment to occur in an orderly fashion, and so they've announced the formation of the Open Virtual Platforms alliance, or OVP, and have seeded it with some of their technology.

There are three basic components to the OVP: APIs, models, and a reference simulator that they have named OVPsim. The APIs provide a consistent interface for modeling all of the elements of a platform, as well as the platform itself. The main APIs are the modestly-named Innovative CPU Manager (ICM) for creating platforms; the Virtual Machine Interface (VMI) for creating processors; Behavioral Hardware Modeling (BHM) for handling processes, delays, and events; and Peripheral Programming Model (PPM) for creating peripheral interconnections. They've provided the C header files and documentation.

The first processor models they have provided are for ARM7, several MIPS processors, and the OpenRISC OR1K, with plans for more to come. They've also modeled a number of standard embedded devices to allow assembly of a complete platform, including various types of memories, traps, bridges, DMA engines, and UARTs, to name a few. Right now the models are hosted on the OVP site in binary form for ARM and MIPS and in source form for the OR1K, but OVP intends to make them freely available as open source via SourceForge, a popular repository of open-source software.

They also have made some complete platform models available, although they admit that these are for the moment "smallish" platforms with 1 to 24 processors. They've got some more realistic larger platform models in the works.

As to the simulator, they have included a reference version that can be freely used. The OVPsim simulator is capable of modeling entire multicore platforms at 500 MIPS and higher using just-in-time code morphing; their website shows results in excess of 1000 MIPS on desktop PCs. Platforms can have heterogeneous multicore configurations and, of course, can include shared resources. OVPsim can be called from other simulation engines, wrapped as needed. C, C++, and SystemC wrappers are included; others can be written. OVPsim can also encapsulate existing instruction-set simulators (ISSes) so that legacy work can be incorporated into the new environment. It can also interface with standard GDB debuggers using the Remote Serial Protocol (RSP).

One of the traditional hurdles that free software like this can raise is the license requirements. The standard Gnu General Public License (GPL) more or less makes it difficult to make money off of software – and makes it difficult for consumers of such software to make money, since their software may be "contaminated" with GPL software. While many software writers like to give software away, most of them like to be paid, so seeding an environment with a way to get some return on investment was seen as critical to attract companies to coalesce around this effort. To this end, the less-restrictive Lesser GPL (LGPL) license will likely be used; this allows proprietary linking and inheritance from free libraries, while still requiring that wholesale derivatives be kept public.

As of announcement date, the list of other companies and entities participating in the OVP effort included Azul Systems, Beyond Semiconductor, Brian Bailey (a consultant), Calypto Design Systems, Carbon Design Systems, CMU Prof. Don Thomas, CriticalBlue, Denali Software, ElementCXI, EVE, Forte Design Systems, Jennic, MIPS, Nova Software, SiBridge Technologies, Sigmatix, and Tensilica. This seems to be something of a grass-roots assemblage, but it stands to reason that some of the larger companies with current proprietary solutions may be slower to move beyond their current wares.

At some point Imperas sees the possibility of standardizing the API, but it's too early for that now. It's more important just to have something that can be pushed and pulled on as it gets used and abused. Imperas does participate in the Multicore Association and so has visibility and a voice in the standardization work going on there.

As to Imperas' own business model, one of the obvious immediate questions is, if you're giving your stuff away, how do you make money? As might be expected, this isn't a charity operation. But they perceive that there's a fundamental critical mass of technology that must be available in order to seed real commercial ventures. They see the three great market requirements being programming models, verification/debug/analysis tools, and simulation platforms. The latter they're giving away via OVP for Windows XP, although they also have faster commercial versions for Linux and other environments. And their main focus for the immediate term will be the verification, debug, and analysis arenas. Clearly they'll be looking to be amongst the first to start forming crystals around the seed.

Bryon Moyer, Embedded Technology Journal

March 18, 2008