Current developments in microprocessor design favor increased core counts over frequency scaling to improve processor performance and energy efficiency. Coupling this architectural trend with a message-passing protocol helps realize a data-center-on-a-die. The prototype chip (Figs. 5.7.1 and 5.7.7) described in this paper integrates 48 Pentium class IA-32 cores [1] on a 6×4 2D-mesh network of tiled core clusters with high-speed I/Os on the periphery. The chip contains 1.3B transistors. Each core has a private 256KB L2 cache (12MB total on-die) and is optimized to support a message-passing-programming model whereby cores communicate through shared memory. A 16KB message-passing buffer (MPB) is present in every tile, giving a total of 384KB on-die shared memory, for increased performance. Power is kept at a minimum by transmitting dynamic, fine-grained voltage-change commands over the network to an on-die voltage-regulator controller (VRC). Further power savings are achieved through active frequency scaling at the tile granularity. Memory accesses are distributed over four on-die DDR3 controllers for an aggregate peak memory bandwidth of 21GB/s at 4× burst. Additionally, an 8-byte bidirectional system interface (SIF) provides 6.4GB/s of I/O bandwidth. The die area is 567mm² and is implemented in 45nm high-k metal-gate CMOS [2].

The design is organized in a 6×4 2D-array of tiles [3] to increase scalability. Each tile is a cluster of two enhanced IA-32 cores sharing a router for inter-tile communication. Cores operate in-order and are two-way superscalar. A 256 entry lookup table (LUT) extension of the 64-entry TLB translates 32-bit virtual addresses to 36-bit physical addresses. The separate L1 instruction and data caches are upsized to 16KB and support both write-through and write-back. Each L1 cache is reinforced by a unified 256KB 4-way write-back L2 cache. The L2 uses a 32-byte line size, matching the cache line size internal to the core, and has a 10-cycle hit latency. The L2 also uses in-line double-error-detection and single-error-correction for improved performance and several programmable sleep modes for power reduction. The L2 cache controller features a time-out-and-retry mechanism for increased system reliability.

Shared memory coherency is maintained through software protocols, such as MPI and OpenMP [4], in an effort to eliminate the communication and hardware overhead required for a memory coherent 2D-mesh. A new message-passing memory type (MPMT) is introduced as an architectural enhancement to optimize data sharing using these software procedures. A single bit in a core’s TLB designates MPMT cache lines. MPMT retains all the performance benefits of a conventional cache line, but distinguishes itself by providing coherence only in user-visible cache blocks. All MPMT cache lines are invalidated before reads/writes to the shared memory to prevent a core from working on stale data. A new instruction, MBINV, is added to the core to invalidate all MPMT cache entries in a single cycle. Subsequent reads/writes to invalidated cache lines cause cache misses, forcing the data to be readwritten from the shared memory. Figure 5.7.2 illustrates the strict software-control protocol used to maintain cache coherency of the MPMT space. The MPB, used as a distributed on-die shared memory, further optimizes the design by decreasing the latency of memory access. Messages passed through the MPB see a 15× latency improvement over messages sent through DDR3.

References:
Figure 5.7.1: Block diagram and tile architecture.

Figure 5.7.2: Message passing vs. DDR3-800 speed grade.

Figure 5.7.3: Router architecture and latency.

Figure 5.7.4: Voltage/Frequency Islands, Clock Crossing FIFOs and Clock Gating.

Figure 5.7.5: Measured HPL performance and energy benefits.

Figure 5.7.6: Measured full-chip power breakdowns, and maximum frequency (Fmax) and power versus VCC.
Figure 5.7.7: Full-Chip and tile micrograph and characteristics.