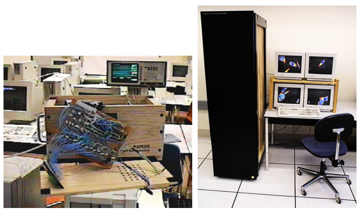
20 Years Of AGGREGATE.ORG SC Exhibits

SC13 marks the 20th AGGREGATE.ORG research exhibit at SC. Our research consortium is rooted in the research group Professor Hank Dietz formed in 1986 when he joined the faculty at PURDUE UNIVERSITY, and it is still led by him, since 1999 based at the UNIVERSITY OF KENTUCKY. Some highlights of our SC exhibits:



1994: In 2/94, we built the world's 1st LINUX PC cluster supercomputer (shown above) to test our new AGGREGATE FUNCTION COMMUNICATION model and the open source software and hardware that implemented it. We augmented 10Mb/s Ethernet with PAPERS (PURDUE'S ADAPTER FOR PARALLEL EXECUTION AND RAPID SYNCHRONIZATION), which communicated between nodes without an OS call, with total latency as low as 3us – several orders of magnitude faster than the Ethernet-only BEOWULF cluster obtained later that year. Our SC94 exhibit disseminated the new technologies and demonstrated 386 and 486 LINUX PC clusters, the 1st IBM POWERPC AIX cluster, and the 1st DEC ALPHA OSF cluster. Demos included cluster-driven synchronized video displays and multi-channel audio.

1995: The TTL_PAPERS MICROCLUSTER fit a cluster of four 486 systems in a "luggable" 30-pound 1-foot cube. Simplified and scalable versions of our *open source* PAPERS hardware and software including our LINUX patch for SECURE OS-BYPASS I/O were highlighted.

1996: Our exhibit housed 37 computers and 27 monitors, including 2x2 and 4x4 LINUX cluster VIDEO WALLS – demonstrating a four-player SWARM GAME and a dynamically load-balanced shared-memory MANDELBROT FRACTAL computation. The AFAPI (AGGREGATE FUNCTION API) library was released for PAPERS clusters and generic SMPs.

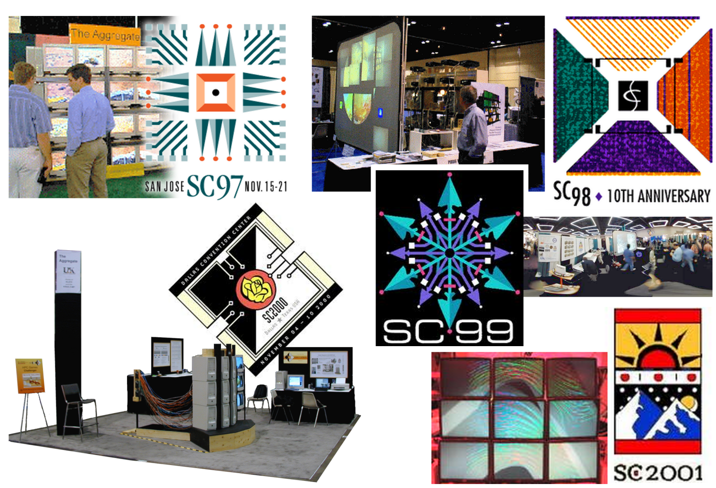
1997: We demonstrated a 5,120x4,096 pixel video wall driven by a PAPERS cluster of 16 HP PA-RISC systems, an MPI 2.0 port built on top of AFAPI, and GIGABIT ETHERNET. A major update of Dietz's PARALLEL PROCESSING HOWTO (published by the LINUX DOCUMENTATION PROJECT) included chapters on SMP, CLUSTERS, SWAR (SIMD WITHIN A REGISTER) using MMX, and ATTACHED PROCESSORS (DSPs and FPGAs). With many collaborators outside PURDUE, our group adopted its current name: **AGGREGATE.ORG**

1998: A 7.5'x10' rear-projection video wall driven by a fleet of AMD K6-2 processors demonstrated applications using 3DNOW! floating point – compiled using our SWARC parallel C dialect compiler. Demos included interactive pan and zoom of high-resolution images (up to 200MB).

1999: A projection video wall driven by an AMD Athlon cluster allowed interactive pan and zoom of 360° images sent via 802.11 by pairs of fisheye cameras tethered to laptops mounted on autonomous robots. VWLIB was released, and literally on the show floor we used it to create the world's 1st video wall MPEG player.

2000: This was the year we built KLAT2 (KENTUCKY LINUX ATHLON TESTBED 2), which set various world records for supercomputer price/performance and was recognized by GORDON BELL and COMPUTERWORLD SMITHSONIAN awards. The sub-$10K KRAA Z-MP cluster, our entry in the HPC games, demonstrated the key technologies on the exhibit floor at SC2000: an FNN (FLAT NEIGHBORHOOD NETWORK) designed by a genetic algorithm and our ATLAS support for 3DNOW! – it even had a small video wall.

2001: In addition to a highly-portable 3x3 video wall made by careful mounting of a cluster of 9 ATHLON laptops, we released the public domain CDR (CLUSTER DESIGN RULES – known as the BDR, BEOWULF DESIGN RULES on SOURCEFORGE), "the first software package to exhaustively analyze a complete cluster design space to determine the best cluster design for a particular application(s), site characteristics, and budget."



2002: The invention of "Application-Specific FNNs" (now called SFNNs – Sparse FNNs) and improvements to the CDR were accompanied by application demos.

2003: Using an SFNN, the KASY0 (Kentucky ASYmmetric zero) cluster became the first supercomputer under \$100/GFLOPS, again setting various world records... many of which were held by KLAT2 until then. Kentucky Architecture Nanocontrollers were introduced as a way to bring MIMD programmability to ~100 transistors per processor (using our MIMD-on-SIMD technologies). This is also when we became co-developers of Warewulf, reducing the memory needed in diskless nodes.

2004: Refinements on various things were presented, but this was the SC where we first presented support for general-purpose computation using GPUs (Graphics Processing Units). We also began hosting both the CAOS and Centos Linux distributions.

2005: Various improvements and presentation of new technology for using microphone arrays to track sound locations.

2006: Still more improvements. A major new topic was a method for implementing true predictive power management based on compile-time analysis to generate annotation for runtime predictions of future power use. We also presented native pair arithmetic algorithms for obtaining high accuracy while using fast single-precision floating point on GPUs, SWAR, etc.

2007: In Reno, NV, our exhibit had fake slot machines that would simulate summing floating-point number sequences using single, double, and arbitrary precision. The interesting result is that in the rare cases when single isn't sufficiently accurate, double usually isn't either. Thus, we presented methods for speculative precision.

2008: We demonstrated MOG (MIMD On GPU): the ability to take arbitrary code written for a MIMD model and to compile it into a form that executes efficiently inside a GPU. Two different versions were demonstrated: one based on generating an optimized interpreter the other using our meta-state conversion algorithm.

2009: Major improvements to the interpreter-based MOG were the big news.

2010: The MOG system was completely rewritten to reprocess arbitrary code compiled using any MIPSEL compiler toolchain and was released as full source code. We also demonstrated some computational photography work involving 3D capture using single-capture, single-lens, anaglyphs.

2011: Improvements to various things, ranging from predictive power management to LAR (Line Associative Register) architecture. An MPI message-passing library for use inside GPUs was Dalton Young's MS Thesis.

2012: NodeScape, a tool for physically-correct visual summary of status of cluster nodes was demonstrated projecting color status information on node cases. KNITT (Kentucky's Network Implementation Topology Tool) uses a genetic algorithm to optimize physical layout of cluster components. GAMake automatically, and in parallel, uses a genetic algorithm to find optimal parameter values for anything that can be built using unix-like make.

2013: Various improvements plus the public introduction of TDI (Time-Domain continuous Imaging). TDI uses massively-parallel nanoprocessor computing as an integral part of a new type of image sensor that doesn't capture images, but waveforms describing how light levels are varying over time at each sensel.

The future: Aggregate.Org has been a leader in developing, and freely disseminating, new technologies that make high-performance parallel computing more cost-effective, capable, and available. We'll continue to do that. If you'd like to help, we'd like that too – be it as a source of research funding, an academic or industry collaborator, or a student at the University of Kentucky.

This document should be cited as:

```
@techreport{sc13year20,
author={Henry Dietz},
title={{20 Years Of Aggregate.Org SC Exhibits}},
institution={University of Kentucky},
address={http://aggregate.org/WHITE/sc13year20.pdf},
month={Nov}, year={2013}}
```

SC13 Denver, CO 2013

Aggregate.Org
UNBRIDLED COMPUTING