

;login:

THE MAGAZINE OF USENIX & SAGE

August 2001 • Volume 26 • Number 5

inside:

CLUSTERS

EDEN: A HOME BEOWULF

by Robert G. Brown

Special Focus
Issue: Clustering

Guest Editor: Joseph L. Kaiser

USENIX & SAGE

The Advanced Computing Systems Association &
The System Administrators Guild

eden: a home beowulf

Introduction

The Duke University Physics Department is a longtime user of cluster computing, and was an early adopter of Intel/Linux clusters when Intel's P6 family was introduced in the form of the Pentium Pro (which had quite tolerable numerical performance for its time at a far lower cost than competing proprietary UNIX platforms). Over the last eight years I've probably been the largest consumer of compute cycles on campus, at one time running a single computation on more than 100 hosts scattered all over campus and more recently running more or less continuously on the department's Beowulf, Brahma, which I originally designed and built.

Over the years, participation in the Beowulf mailing list has taught me a great deal about Beowulfery, parallel computation, cluster computation, and so forth. Beowulfery has become something more than a hobby and less than my primary profession (which is theoretical physics). I find designing, building, and working with Beowulf-style compute clusters both interesting and just plain fun. I also have an obsession with penguins and open source. Consequently, my clusters tend to run Linux rather than, say, Solaris or Irix.

Naturally, I've had a computer at home capable of connecting to the department network for nearly 20 years now. Over the years, this has transmogrified from an IBM PC with a 1,200-baud modem running Kermit into my current dual Celeron with a fire-walled DSL connection; somewhere in there (about the time I could afford a 100BT switch) my home computer(s) became my home network became my home Beowulf: eden. To be really picky, my home cluster, since I have both dedicated and desktop nodes necessarily mixed on my home network.

eden has proven to be an immensely valuable tool for me professionally. It serves as a totally protected sandbox where I can try things out, prototype things, experiment, and develop code. It has been a solid base from which to begin writing a book on the engineering of Beowulf clusters. It has enabled me to develop tools like `procstatd` (available on the Brahma page) to help monitor and manage Linux LANs and clusters. It has helped me develop benchmark tools and methods like `cpu-rate` (also available on the Brahma page) to support the engineering and management of clusters.

Sure, a lot of this could be (and was) done at Duke on Brahma's nodes and workstations (also a mixed cluster, with a Beowulf component and a distributed desktop component), but Brahma is generally a production environment and it isn't always easy to find an idle node to benchmark or an idle set of nodes to run a code prototype on.

One final advantage that eden has provided me is that five or six years ago I co-founded a company, Market Driven, by writing and contributing its core operational modeling software (an advanced neural network engine). Market Driven isn't really a "Beowulf company" (it actually does stuff like Web design and product development coupled with predictive modeling and database technology), but of course Beowulf methodology and cluster methodology have formed an important component of tools like the neural network. Having a home cluster to work on has kept my evening and weekend efforts "Duke Resource Free," which is an important thing to be able to demonstrate when worrying about ownership and interest.

One great thing about a home Beowulf is how easy it has become to make, or perhaps a better term might be "grow," one. In the following section I'll discuss the engineering of

by Robert G. Brown

Robert G. Brown is a theoretical physicist teaching and doing research in condensed matter physics at Duke University. Over the last decade his research has demanded ever more computer power and expertise which he has obtained by using entire networks of workstations as one big "cluster." He is one of the most prolific contributors to the "beowulf" list, where he is known as "rgb."



rgb@phy.duke.edu



Figure 1: eden's core systems "neatly" arranged on over-the-counter shelving. Left to right: 300MHz Celeron rgb.adsl.duke.edu/adam (the gateway, note DSL plug on the wall), 466MHz Celeron abel, and dual 466MHz Celeron lucifer. The switches and so forth are on the middle shelf, printers on the top.



Figure 2: The front-end arrangement of eden. The monitor is connected to lucifer (left bottom), a dual 466MHz Celeron. Note KV switch

eden and show how just about anybody with a similar need or interest can spend at most a few thousand dollars (of their own or, better, somebody else's money) and end up with a perfectly useful and viable home Beowulf for code prototyping, production, or play.

eden's Design

eden is the simplest of all Beowulfish/cluster designs. It is basically an eight-port 100BT switch (a Netgear FS108) with a collection of mid-tower PCs (all running Linux, naturally) connected to each other via the switched network. On the front side, I have (strictly for convenience in a cluster where, like as not, a node will be opened up with some new piece of junk hardware hanging off the bus in the bottom of the case) a very cheap keyboard and video switch.

In the primary core of eden, I have lucifer as a front end and server node with two processors and abel and adam as compute nodes. Most of the components also have to do some sort of double duty: adam is a gateway, abel is a printer/scanner server, lucifer has a printer and is the primary cluster server in addition to being my desktop. The core has four processors with a total of around 1.7GHz of Celeron cycles at its disposal.

However, there are more computers in the house and all of them are or can be in the cluster (up to the capacity of my current switch). There is eve, my wife's computer (an 800MHz Tbird and the latest and fastest addition), which is available for cluster work at all times since it always runs Linux and is always on the network. Then there are my sons' computers (which they only "have" because they are part of my cluster): caine (400MHz Celeron), serpent (400MHz Celeron), and apple (don't ask). These aren't always available, but can be made available if I need their cycles. They are pictured below, in all of their kid-environment glory. The gerbil isn't really running apple (a 133MHz AMD K5),s but it might as well be.

The nodes in the boys' rooms are dual-boot nodes that can boot Windows so that the boys can do their homework on them if absolutely necessary. Note that serpent is running a typical "homework" application (Might, Myth and Magic IV). When all of them are running Linux and on the network, they represent another 1.7GHz of Intel or AMD cycles available to a computation. Even my 200MHz Pentium laptop can participate as a master in a master-slave computation with a wireless interface.

These nodes are all running RedHat Linux, although there is some version drift across them because I've been using one to test 7.1, have 7.0 on most of them, and am running the gateway with a stripped and frequently updated 6.2. They are all running PVM (my personal favorite parallel library) and can be configured into a PVM virtual machine. I also am fond of working on raw socket-based parallel code, being a bit of a masochist. An article on Brahma (see the "Extreme Linux Tutorial" from the 1999 Linux Expo in Raleigh, NC) was developed using eden, and eden was in fact carried to the EL booth and demonstrated at the Expo since my house is only about 30 miles from the Expo site.

The purpose of eden isn't really production, so I won't bore you with benchmark results or claims of fabulous speed. In fact, its 500–600 aggregate Mflops are relatively slow – two 1.33GHz Tbirds would likely replace the whole motley lot and give you back change for your dollar. eden is also highly speed-heterogeneous (as it must be, given that I buy a node or so a year), which gives me an excellent opportunity to work on load balancing parallel code. It is still perfectly adequate to give excellent parallel speedup of some chores that I need to do, and it is ideal for allowing me to do code development and testing without having to work over the DSL link.

The Future of eden

The next-generation makeover of eden (currently underway) will be significantly improved. We are having the attic remodeled into a “proper” home office for my wife (a physician) and me, and this whole space will be networked, wired, and cooled for a much more devoted-function Beowulf. In particular, I’ll have some 12 built-in lines in the attic alone and a patch panel that will allow me to add the various rooms and systems downstairs to a larger switch. In addition there is the wireless network in the house that will let me work with my laptop from a hammock in the backyard on a nice day, with a cooling and refreshing beverage close at hand.

I also expect to recycle into the arrangement some old Pentium Pro and PIII-class systems that are no longer useful at Duke to give me more dedicated compute nodes (and maybe upgrade the K5 while I’m at it). Then, of course, it is almost time to buy this year’s new addition from my PDA – probably a 1.33GHz Tbird, unless something better comes along in the meantime.

I doubt that I’ll ever use this cluster to do a lot of real computing (I have much more powerful resources at Duke to do that with), although I certainly have put it to work in the past. However, it is capable of doing very useful computational work indeed, and as a vocational/avocational tool it is without peer. Then there is a certain cachet associated with having a home supercomputer with an aggregate performance on favorable code to what one might expect from a multimillion-dollar supercomputer only five or six years ago. Home Beowulfery is indeed a source of fun and profit for those who are so inclined. There is no reason that this little home Beowulf cannot do rational drug design or model and predict stock market performance just like its bigger brothers.

Hmmmm, now there’s a useful project for the devoted hobbyist . . .



Figure 3: Four distributed desktop units (eve, caine, serpent, apple), which can participate in a parallel computation. The mess visible in these pictures is a real mess, not a simulated mess, as this cluster is in a real house with real boys and real working parents.

