WEBVTT

1
00:00:03.100 --> 00:00:08.840
Good morning and welcome to the size distinguished lecture series. Today it's my

2
00:00:08.910 --> 00:00:13.889
W2210 x8479 Conf Room: honor and pleasure to introduce our speaker, Jack Tungara.

3
00:00:14.000 --> 00:00:20.119
W2210 x8479 Conf Room: Jack is the recipients of the two thousand and twenty one Acm. A. M. Curing award

4
00:00:20.380 --> 00:00:37.130
W2210 x8479 Conf Room: which is the highest honor in computer science for aspiring contributions to numerical algorithms and libraries that have enabled high performance computational software to keep pace with the exponential hardware improvements over the past four decades.

5
00:00:37.690 --> 00:00:40.989
W2210 x8479 Conf Room: Jack specializes in numerical algorithms

6
00:00:41.000 --> 00:00:42.589
W2210 x8479 Conf Room: in linear algebra

7
00:00:42.750 --> 00:00:50.909
W2210 x8479 Conf Room: parallel, computing the use of advanced computer architectures, programming methodology and tools for parallel computers.

8
00:00:50.970 --> 00:00:57.020
W2210 x8479 Conf Room: His contributions to these fields have truly transformed the field of high-performance scientific computing.

9
00:00:57.270 --> 00:01:07.939

W2210 x8479 Conf Room: Ah, Jack Holds appointments at the University of Manchester, Oakridge National Laboratory, and the University of Tennessee, where he founded the innovative computing laboratory

10
00:01:08.400 --> 00:01:22.179
W2210 x8479 Conf Room: in addition to the Turing award, Jack has received numerous awards, including the actually I, Tripoli Sid fun Back Award The Acm. I Triple Kennedy Award The I Triple Charles Garbage Award,

11
00:01:22.270 --> 00:01:26.029
W2210 x8479 Conf Room: the I. Triple E Computer Society Computer Pioneering award

12
00:01:26.290 --> 00:01:38.189
W2210 x8479 Conf Room: just to name a few. He's, a Fellow of Aaas Acm I. Tripoli and Siam, a Foreign member of the British Royal Society and a member of the Us. Academy of Engineer.

13
00:01:38.770 --> 00:01:44.380
W2210 x8479 Conf Room: I could go on. But you're not here to listen to me, Doc, so let me turn it over to Jack.

14
00:01:48.880 --> 00:01:58.500
Well, it's a pleasure to be here, and we should keep this informal, so I don't know if you have questions from the audience. Just shout them out during the hearing the thing, and we'll! We'll see what

15
00:01:58.510 --> 00:02:14.890
see what we can do with that. So i'm going to talk about high performance computing today, and look a little bit to the future, as mentioned. I'm. Ah, i'm a emeritus professor at the University of Tennessee. I hold a position at Oakbridge National Laboratory, which is about forty forty miles from the University,

16
00:02:14.900 --> 00:02:21.940
and I also hold a faculty position at the University of Manchester in the Maths department, and I go there one one month a year.

17
00:02:21.950 --> 00:02:41.730

So um, you know, and sort of the message here is uh computing is in this rapid transition stage. We have smartphones and and cloud services are really eating the world. We have Hpc. Changes in the wind. It's greater performance now

18
00:02:41.740 --> 00:02:48.840
requires more money and so high-performance computing one of the supercomputers

19
00:02:48.990 --> 00:02:52.730
in the States. Anyway, it goes for about six hundred million dollars.

20
00:02:52.930 --> 00:03:01.620
Well, that's the cost of a big machine like the doe machines in Japan. They paid a billion dollars for the fbacco system.

21
00:03:01.630 --> 00:03:20.590
It was really a little bit ahead of its time, and the money that went into it. So so so so changes in the wind money is, It's going to be greater. You know. We all know this story. Transistors are getting more expensive than our scaling is ended. We got the slowdown of Moore's law.

22
00:03:20.600 --> 00:03:39.180
But you know, this graph here shows a number of trends. The one that ah is a bit striking is the fact that the number of transistors on this graph doesn't appear to be ah bending over, it seems to be going up straight, and and I was thinking about that, and I think the reason for that is that the

23
00:03:39.220 --> 00:03:58.609
chip area is growing. So the chips used to be this size, and now they're getting bigger, and as a result of that, the transistors that are on the chip are increasing, not not necessarily because of Moore's law, because the area of the ship is increasing. So I think that's the reason why that that slope doesn't bend over more than than it shows.

24
00:03:58.620 --> 00:04:15.829
Um, And you know the the center of technology innovation, and the money has has shifted from from what it was before. So there's a lot of changes computing. All those computing is pervasive, and all aspects of society drives

25

00:04:15.840 --> 00:04:30.749
the commercial side and the business side Apple Samsung and Google dominate the world of smartphones, and you know the thing that they're doing is they're designing their own silicon. So they're designing their own processors.

26
00:04:30.890 --> 00:04:33.199
They're not relying on commodity stuff.

27
00:04:33.470 --> 00:04:50.700
And um, you know, Google Microsoft, Amazon, Facebook, you know, they really dominate in terms of the um, the funding that they have the market cap for those companies is over a trillion dollars just incredible amounts of money, and you know the Chinese companies are not far behind

28
00:04:50.710 --> 00:04:55.539
that way, and they're all designing their own silicon. All the companies,

29
00:04:55.960 --> 00:05:09.499
W2210 x8479 Conf Room: you know. We see this with um hyperscaler companies. So you know, companies like Amazon, Google, Microsoft, Alibaba, they all have their own hardware design

30
00:05:09.520 --> 00:05:19.069
for their for their products. So they're targeting the hardware to specialize for their needs. They have tremendous resources,

31
00:05:19.080 --> 00:05:36.510
you know. Ah, consider them to be exothermic in terms of the amount of money they have. They have tremendous resources that they can devote to. Ah building those chips and hardware, and we even see this of carmakers. So Tesla has their own, their own chips that they're using and putting forward,

32
00:05:36.520 --> 00:05:51.439
so it's quite different than the story we have from Hpc. So in the Hpc. Side we really have this mono culture. So if you take a look at the top, five hundred data, So the graph that i'm showing here is a graph over time.

33
00:05:51.450 --> 00:05:58.669
We're showing the five hundred computers. And what I've done is to highlight of the five hundred computers,

34
00:05:58.790 --> 00:06:07.379
the numbers that are being that are used using the X eighty, six architecture for the architecture that's from Intel and Amd.

35
00:06:07.520 --> 00:06:12.719
So what we see is early on in the In the story of the top five hundred list

36
00:06:12.870 --> 00:06:25.570
we have this: We have a few, just a few machines using the X eighty, six architecture. So the top five hundred was started in one thousand nine hundred and ninety, three, and then there was this

37
00:06:25.580 --> 00:06:41.030
drop, where high-performance computing was really dominated by vector supercomputers at that time. So they weren't interested in in microprocessors. They didn't have enough power. And then there was this: You know one thing that's referred to as the attack of the killer microse.

38
00:06:41.040 --> 00:06:57.729
But the processors got much more powerful, and we see this incredible rise in terms of the number of machines on the top, five hundred that are using this X eighty, six architecture. So today we have of the five hundred machines today. Seventy eight percent of them use intel chips.

39
00:06:57.740 --> 00:07:05.950
Another nineteen percent use Amd chips. So ninety seven percent of the top five hundred machines are based on x, eighty, six.

40
00:07:06.930 --> 00:07:08.400
And Why is that?

41

00:07:08.410 --> 00:07:22.940
Well, it's because it's cheap, right? So it's a commodity processor. So we can put together machines for a low cost, getting theoretically high performance and trying to drive scientific computation in that way.

42
00:07:22.950 --> 00:07:40.410
And um, you know the the performance that we see comes about because of Gpus in in most cases, and those gpus are also commodity. So they're being purchased by from intel sorry from India. So if we take a look at this graph, this again is looking at the

43
00:07:40.420 --> 00:07:52.200
the top five hundred data, and we're plotting vendors of accelerators that are in the top five hundred machines. The accelerator started around two thousand and six

44
00:07:52.210 --> 00:08:11.970
and in two thousand and six. We just had a few of the high-performance machines top five hundred machines using accelerators, and then we saw that grow to around. That number today is around one hundred and sixty, something one hundred and sixty. Seven machines are using that mainly being propelled by Nvidia

45
00:08:11.980 --> 00:08:33.060
of the top five hundred machines. They're also using commodity interconnect. So we see it infiniband and ethernet being used in four hundred and twenty six of those machines. So commodity processors. A lot of the accelerators and commodity networks are being used, and Linux of course, is everywhere. One hundred percent of the machine machines.

46
00:08:33.480 --> 00:08:48.819
So doe has a program program called the ex-scale competing program, and that program is seven year program. So we're in your we're in the last year. Now so this is, we have eleven more months to go in the ex-scale community program.

47
00:08:49.270 --> 00:08:58.809
The price tag for the program is four billion dollars the four billion dollars. So the question is, what do you get for four billion four billion dollars? You get three computers.

48
00:08:58.820 --> 00:09:12.349

So the three computers are the machine that's actually in place and running at Oak Ridge. So it's a machine called Frontier. It's based on Amd. Processors and the Cpu and Amd. Gpu commodity processor,

49
00:09:12.460 --> 00:09:15.210
and it's up and running sort of,

50
00:09:15.620 --> 00:09:26.289
because it's just being, you know. It's a large machine. They're shaking it down. They can run some things on it. It has a lot of issues, a lot of software, a lot of hardware issues, and that's being worked out.

51
00:09:26.300 --> 00:09:34.909
The next machine is a machine that's going into Argon. It's called Aurora. It's based on intel part. So it would have an intel processor and an intel accelerator

52
00:09:35.210 --> 00:09:36.190
in it,

53
00:09:36.200 --> 00:09:43.790
and the final machine that's going into Lawrence Livermore will be similar to the Oak Ridge machine Amd. Parts in that machine.

54
00:09:43.800 --> 00:09:47.670
So six hundred million dollars for the hardware on each of those systems,

55
00:09:47.900 --> 00:10:06.580
the do. We also gave vendors four hundred million dollars for something called non-recurring engineering, so that money was an incentive to the companies to make changes to the architecture, so that it would better reflect the applications that we're going to run on those machines.

56
00:10:06.630 --> 00:10:09.189
And to be honest, I really can't point

57

00:10:12.390 --> 00:10:13.730
W2210 x8479 Conf Room: by volume one.

58
00:10:14.080 --> 00:10:15.240
Oh, okay,

59
00:10:15.250 --> 00:10:18.929
W2210 x8479 Conf Room: Um. So Um. Is that is that better now?

60
00:10:19.750 --> 00:10:20.840
W2210 x8479 Conf Room: So

61
00:10:21.920 --> 00:10:23.760
it's better. You think. Okay.

62
00:10:23.900 --> 00:10:37.590
W2210 x8479 Conf Room: So four hundred million dollars going to companies like Nvidia. We've got like Intel and Amd and some of the some of the interconnect companies to help with it. But again I have a hard time pointing to what what was done for that.

63
00:10:37.600 --> 00:10:53.889
W2210 x8479 Conf Room: The whole point of this program is to drive applications at X of scale. So there are twenty one applications that Doe is specified that should participate in this program. And those are all energy Related as you may, you might guess, from the Department of Energy. They deal with wind energy,

64
00:10:53.900 --> 00:11:11.519
W2210 x8479 Conf Room: nuclear energy, combustion, magnetic fusion, chemistry, astrophysics, and a number of things. So those are the twenty one applications that are intended to run on these machines when they finally get up and running. And then there's a software stack,

65
00:11:11.530 --> 00:11:24.959
W2210 x8479 Conf Room: the money that's going into the project funds, the application work, and it also funds the software that's being developed in the program. So the way I think of it is

roughly half of the money, one for hardware and the other half is going for the applications in the software.

66
00:11:25.450 --> 00:11:34.589
W2210 x8479 Conf Room: The problem with this program is it's going to end It's going to end in eleven months. There's a thousand people working on this program.

67
00:11:34.600 --> 00:11:53.379
W2210 x8479 Conf Room: And um, there's a cliff because there's no nothing at the end of that eleven month period There's nothing that will take up those thousand people. There's not enough funding in the system to cause those people to be retained. So at this moment there's people being picked off

68
00:11:53.390 --> 00:12:18.900
W2210 x8479 Conf Room: and and going to other places. So we're losing. We're losing talent. It's a situation where we have incredible talent that was developed over the past seven years on this project. And now those people are going off, and and perhaps joining other other companies. So in some sense it it was a tragic mistake. That the ue doesn't have a follow-on project at at or near the scale that could help and absorb that.

69
00:12:18.910 --> 00:12:24.629
W2210 x8479 Conf Room: So High-performance Computing today is highly parallel. It's a distributed memory it has.

70
00:12:24.640 --> 00:12:53.550
W2210 x8479 Conf Room: It uses Mpi and open Mp. As the programming models. Mpi, of course, between the processes and between the processors, the hardware and the open Mp. For shared memory. Frontier is is the first machine off the rack, and it has a peak performance of two xiflops, and that's sixty four bit flops. So two exaflops, ten to the eighteen floating point operations per second, eight point, eight million cores in the machine, nine thousand four hundred nodes

71
00:12:53.560 --> 00:12:58.180
W2210 x8479 Conf Room: in the system, consuming about thirty megawatts of power,

72
00:12:58.490 --> 00:13:15.010

W2210 x8479 Conf Room: and again fueled by Amd. Amd. Processors and the Machine's performance comes really through those and D processors. So ninety eight percent of the performance is derived by the gpus,

73
00:13:15.020 --> 00:13:32.490
W2210 x8479 Conf Room: and if you're not using the gpus, then your performance is really going to suffer on this on this machine. So it's commodity processors and commodity accelerators being used. And you know, I have to say that when you take a look at the architecture, communication is the most expensive thing.

74
00:13:32.500 --> 00:13:36.589
W2210 x8479 Conf Room: Floating point is over over, provisioned in the machine,

75
00:13:36.600 --> 00:13:46.790
W2210 x8479 Conf Room: and it's really not an issue in terms of time to solution. It's the moving the data that that really is where the problem is.

76
00:13:46.800 --> 00:13:57.039
W2210 x8479 Conf Room: And you know, today we have different levels of floating point precision. I've been talking about sixty four-bit accuracy. But machines have thirty two-bit, sixteen-bit

77
00:13:57.050 --> 00:14:10.829
W2210 x8479 Conf Room: with Nvidia. Even some machines have eight bit floating point arithmetic so using the lower precision. Things run faster. So there's an effort today to understand where we might drive

78
00:14:10.840 --> 00:14:20.069
W2210 x8479 Conf Room: the performance if we can get by with lower lower precision. And can the application sustain that lower precision as we go for

79
00:14:20.080 --> 00:14:33.619
W2210 x8479 Conf Room: okay, So we're talking about ex-flop. So what's an exaflop computer? So one flop is an add or a multiply of a sixty four-bit for sixty four-bit operand. So that's a floating point operation.

80

00:14:33.630 --> 00:15:03.099
W2210 x8479 Conf Room: And an ex-flop is a billion billion floating point operations. Per second. I gave a talk a few months ago, and after the talk a newspaper guy came up to me and we talked about exoscale computing, and I said a billion billion operations per second, and we discussed a number of things. The next day I opened the paper and read the interview, and it said, Two billion. This is not two billion. This is a billion billion

81
00:15:03.110 --> 00:15:17.889
W2210 x8479 Conf Room: operations for a second. So hopefully we get that right? So it's ten to the eighteen operations per second. And the way to think of it is. If you had everybody on the earth doing one operation per second,

82
00:15:17.900 --> 00:15:23.459
W2210 x8479 Conf Room: everybody on the earth, it would take four years for everybody to do

83
00:15:23.470 --> 00:15:39.710
W2210 x8479 Conf Room: an ex aflop right, everybody to do ten to the eighteen operations. And that's what this computer can do in one second. So really, quite stunning. So I became an accidental benchmarker back in one thousand nine hundred and seventy nine. I was involved in a project.

84
00:15:39.720 --> 00:15:48.859
W2210 x8479 Conf Room: I was a graduate student. I was working at Argon national laboratory, and we had a project that was funded by Nsf.

85
00:15:48.870 --> 00:16:05.580
W2210 x8479 Conf Room: To develop a software package called Linpack. So Windpack is a package of mathematical software for solving systems of linear equations. It's written in Fortran. It was intended to be portable. Portability was a big deal back in one thousand nine hundred and seventy nine. We didn't have my triple a standard,

86
00:16:05.590 --> 00:16:35.359
W2210 x8479 Conf Room: and ah! Dealing with numerical things became a bit awkward in the portability sense. Ah! So we put this. Ah, we do this package together on the picture. There, on the right is Um Jim Bunch, from Ah, from the University of California, San Diego. Ah, Pete Stewart from Maryland. Ah, Cleve Molar from New Mexico, and that's me on the far on the far left there with a little bit more hair back in one thousand nine hundred and seventy nine. That's a picture of my car.

87
00:16:35.370 --> 00:16:46.900
W2210 x8479 Conf Room: That's the license plate I had on my car, and when I was assumed in New Mexico. Cleve Muller was my adviser. Cleave. If you don't know, cleave, he he's a guy who did Matlab,

88
00:16:46.910 --> 00:17:03.889
W2210 x8479 Conf Room: so he's the guy who formed the math math works company and still involved. So all of these, all these guys here are are linear algebra guys quite capable from the theoretical side, and also from studying the software. So I was. I was sort of a cabin boy of this project

89
00:17:03.900 --> 00:17:09.129
W2210 x8479 Conf Room: I was given tasks to do, and one of the tasks to do was to go off and do

90
00:17:09.140 --> 00:17:37.099
W2210 x8479 Conf Room: uh with some benchmarking to see how fast these these computers could run. So I put an appendix into the User's guide, and I looked at solving a very simple system of linear equations, Ax and B, using the software and lin pack and ran it on a bunch of machines, twenty three machines, in fact. And I put that I put that list in the appendix to the User's Guide, and this is a

91
00:17:37.110 --> 00:17:48.730
W2210 x8479 Conf Room: this is that list ranking them by the time to solution. So the fastest computer that solved it, using the linpack software for a matrix of size one hundred

92
00:17:48.740 --> 00:18:17.699
W2210 x8479 Conf Room: was the machine at N-car, and it it was a cray one, and I've got some scribble here on the paper i've got a pencil scribble on here, which is showing the the execution rate at translating the time into the execution rate, and the N. Car machine was running at fourteen megaflops. It was its performance, So That was the first impact, a benchmark thing. I kept that table. It grew over the years.

93
00:18:17.710 --> 00:18:30.950

W2210 x8479 Conf Room: Um! I probably had a few thousand machines on it, and I met up with um. Hans Lawyer Hans Meyer was a professor at Ah Mannheim, and he had a list along with Eric Straw Meer,

94
00:18:30.960 --> 00:18:56.330
W2210 x8479 Conf Room: of um of the fastest machines and Hans uh, and that was, they were ranked by pizza, and I had a benchmark that had uh based on this impact thing, and Han suggested that we put our list together and call it the Top Five hundred. So the top five hundred list was created, five hundred machines ranked, solving that system of linear equations.

95
00:18:56.340 --> 00:19:00.140
W2210 x8479 Conf Room: So the basic the ground rules are You have to.

96
00:19:00.150 --> 00:19:20.120
W2210 x8479 Conf Room: You have to use the the algorithm Gauss elimination with partial T: So you can do that, you can implement it, however you want. But that's the basic, algorithm and I will give you the matrix, and you'll solve it. Sixty four bit arithmetic, and we check to see that you got the radius. And then we look at the time to solution, and that gets centered into the table.

97
00:19:20.130 --> 00:19:28.330
W2210 x8479 Conf Room: So that's sort of the general markings for the benchmark itself, and it solves a dense matrix. Problem.

98
00:19:28.340 --> 00:19:57.429
W2210 x8479 Conf Room: Um! And it's quite quite simple to ah to generate a matrix quite simple to check the solution and um getting the performance ah should be Ah should be relatively straightforward as well. So this list comes out twice a year. It comes out in November and June. So the most recent, the most recent list we have is from November, and the idea here is that we want to look at the asetonic rate. So if you think about increasing the matrix size,

99
00:19:57.440 --> 00:20:15.600
W2210 x8479 Conf Room: the rate of execution is going to go up until it reaches some asymptotic point, and that asymptotic point is what we want to capture. So you know, make the matrix as big as possible. Solve it, show us the time, and then we convert it into a rate of execution, and the rate gets entered into the into the table itself.

100
00:20:16.050 --> 00:20:18.990
W2210 x8479 Conf Room: Well, this is a snapshot, if you will, of

101
00:20:19.360 --> 00:20:29.060
W2210 x8479 Conf Room: of supercomputing since we started collecting that top five hundred data. So thirty years of data here, and we're looking at

102
00:20:29.070 --> 00:20:46.779
W2210 x8479 Conf Room: uh the uh, we're looking at three numbers on this on this charge here. The first one is this: this: the red set of dots that's the computer at position Number one. So that's the fastest computer of any time today. That's at one point one exiflox that's the frontier system

103
00:20:46.790 --> 00:21:15.259
W2210 x8479 Conf Room: and the orange dots are the machine at position Number five hundred. So that's the guy that just made it on the list. So one point seven, three pedophile. So that's the range, if you will of supercomputing if you, if you think about the five hundred machines as characterizing supercomputing, and the blue dots represent the sum of the five hundred machines. So that's at four point eight

104
00:21:15.270 --> 00:21:40.929
W2210 x8479 Conf Room: exaflops today. So um, you know. There's a few interesting things on the list. There's this sort of ah inflection point that's happened in two thousand and eight, where the graph changes the trend line. And you know what's the cause of that? Well, probably Moore's law slowing down an art scaling, ending the financial crisis that happened which causes delays and refreshing machines, you know, maybe a combination of those things

105
00:21:40.940 --> 00:21:57.619
W2210 x8479 Conf Room: and the machine that was Number one. When this list was first put together back in one thousand nine hundred and ninety three was the thinking machines at Los Alamos National Laboratory. It was used for nuclear weapon simulation. It was a machine with one thousand processors

106
00:21:57.630 --> 00:22:17.110
W2210 x8479 Conf Room: quite innovative for its time, and had accelerators in it. Um! And you know, my laptop, this machine here that I'm using to give this talk. I can run the benchmark on

it, and when I do I get four hundred and twenty six gigaflop, You know that's a stunning achievement, i'll say, for a device which I use mainly to read email,

107
00:22:17.120 --> 00:22:30.860
W2210 x8479 Conf Room: and you know, four hundred and twenty-six gigaflops, so that would have been faster than the machine at um at Los Alibos. That was number one, in fact, seven times faster. So you know the device I'm holding is seven times faster than that.

108
00:22:30.870 --> 00:22:47.189
W2210 x8479 Conf Room: A thousand processor machine at Los Alamos is uh driving those kind of calculations. In fact, it's uh it's equivalent to the Hitachi machine in one thousand nine hundred and ninety, seven, with two thousand uh processors, a machine that was used in Scuba. So you know

109
00:22:47.200 --> 00:22:56.029
W2210 x8479 Conf Room: a lot of changes over a short amount of time. This is the the top ten list. These are the ten fastest machines

110
00:22:56.040 --> 00:23:26.029
W2210 x8479 Conf Room: in the top ten machines, you know, if we take a if you take the sum of the five hundred computers and use that as a number for the top, for the for what the five hundred computers have. The top ten machines. Ah represents fifty, three percent of that. So this list is really stewed as a very long tail, I guess, is another way to say it. But these top ten machines have a tremendous amount of of performance associated with them. So the first machine on Number One machine is the machine that I mentioned.

111
00:23:26.040 --> 00:23:55.180
W2210 x8479 Conf Room: It's at Pokerage National Lab. It's a department of Energy machine run by the office of Science. It's called frontier. It has amd processors and amd accelerators. The red indicator here under this column, which has computers, is a list of the accelerator. So this one here is using this this accelerator, and you can just look down the list here and see of the top ten, nine of them use accelerators.

112
00:23:55.190 --> 00:24:03.780
W2210 x8479 Conf Room: The one that doesn't is the machine in Japan, so the fugitive system does not use an accelerator. It has an arm processor,

113

00:24:03.790 --> 00:24:33.270
W2210 x8479 Conf Room: and it gets its performance through the use of vector instructions. So they they augmented the arm instruction set with vector instructions, and that's being used to drive it. So you know we have the number one machine about seven point seven billion cores in it. It has a impact number of Hpl. Number of one point, one exaflops, and that's sixty. Five percent of the theoretical peak performance. So the theoretical peak is a paper and pencil calculation.

114
00:24:33.330 --> 00:24:41.980
W2210 x8479 Conf Room: So you look at the Cpu and the Gpu with the theoretical peak for those. How many flaps can you do for a cycle?

115
00:24:41.990 --> 00:25:00.790
W2210 x8479 Conf Room: Compute the the theoretical peak rate, and then and then compare that to what we actually a Cpc. On the computer, and that's sixty, five, and this is thirty, one sorry, twenty, one megawatts. So twenty, one megawatts under load, you know, in Tennessee. If I were to buy a megawatt in my house,

116
00:25:00.800 --> 00:25:25.369
W2210 x8479 Conf Room: and I use that for one year. I'll get a bill for a million dollars. So a mega-wide year is a million dollars in Tennessee. So to run this machine to turn it on is about twenty, one million dollars. So just to put some some things in perspective, and the last column is giving us a sense of the efficiency. So the efficiency here is fifty, fifty, two gigaflops per watt. So that's

117
00:25:25.550 --> 00:25:41.569
W2210 x8479 Conf Room: that's the performance, and it's this machine's doing pretty good in terms of that, and you could scan down the list here. The Us. Has five machines on the list. Four of them are at Doe Labs. What is that? Nvidia in-house? That's being used for some of their ai work?

118
00:25:41.580 --> 00:25:57.980
W2210 x8479 Conf Room: Two machines are in China. There's been a lot of discussion about us technology going to China. One of these machines is using us technology and machine at position. Number Ten uses, intel processors, and the next generation of that machine.

119
00:25:57.990 --> 00:26:27.589

W2210 x8479 Conf Room: Um, the Tianha Iii is. Ah, they've taken out the intel processors, and they're using their own processor. So in China has developed their own processors based on arm, which is, which is being used used to fuel the next machine, and i'll talk some about that in a little bit. Um let me see I didn't quite finish that. So Italy has the machine. Finland and Japan sort of round out the the top. The top machines

120
00:26:27.600 --> 00:26:38.690
W2210 x8479 Conf Room: just looking at the list here. There's a couple of things, I guess, to to point out the percent of peak performance, so we get a very high return from the fugaku system. Eighty two percent

121
00:26:38.700 --> 00:26:52.639
W2210 x8479 Conf Room: is what they get in terms of returning to peak performance. And that's also true of the Nvidia machine. They're getting around eighty percent of the theoretical peak performance back when they run the benchmark. So those are sort of interesting numbers to look at.

122
00:26:52.870 --> 00:27:11.090
W2210 x8479 Conf Room: Okay, this machine. That's number one, Two exaflops is the theoretical peak performance based on Amd. Parts, nine thousand nodes. Again, you know you. You'd better be using those gpus on this machine. Ninety nine percent of the performance comes from the Gpus on this particular

123
00:27:11.100 --> 00:27:23.830
W2210 x8479 Conf Room: architecture. If you use sixteen bit floating point arithmetic because of the gpus. You can do sixteen bit floating point, and you could see a theoretical peak performance of eleven

124
00:27:23.840 --> 00:27:52.279
W2210 x8479 Conf Room: excel flocks, so eleven X of flops using sixteen bit, So that's an additional leverage, if you can use it. Sixteen bit is really there, of course, for machine learning applications and artificial intelligence you can get by with a lot less precision than you can. So here's that list of the number one machines back in one thousand nine hundred and ninety seven. We had. We hit a terraform. So the first terrorflop machine was at Stadia National Lab

125
00:27:52.290 --> 00:28:01.060
W2210 x8479 Conf Room: Ascii Red Machine. Eleven years later there was a Petaflot machine at Los Alamos, so it took eleven years to get three orders of magnitude

126
00:28:01.070 --> 00:28:29.340
W2210 x8479 Conf Room: increase in our in our supercomputer. And then, if we look at the exas scale fourteen years later. We've reached another three orders of magnitude, so we're not able to keep pace with things at the same rate that we have been. So the question might be. Where are we going into the future? What happens for the next? Rather this. So if you take a look at the the data that's there from the top five hundred, and then do a projection of that data. So we're looking now

127
00:28:29.350 --> 00:28:42.309
W2210 x8479 Conf Room: at the number one machine and trying to project where we go into the future. And it basically is telling us that it's going to take eight years just to get one order of magnitude in perform. So we would expect.

128
00:28:42.320 --> 00:28:56.330
W2210 x8479 Conf Room: If this is true. If the trend line is correct, it will be eight years before we get just one order, and getting to Zen. A scale is going to take a much, much longer time than we we've seen in the past to gain that three orders of magnitude.

129
00:28:56.340 --> 00:29:13.659
W2210 x8479 Conf Room: So yes, so that's one of the things. This is a graph which shows the top five hundred machines, So I've got a dot for each of the five hundred machines and the rank the y-axis is is by their performance. So this is the number one machine. This is the number two machine

130
00:29:13.670 --> 00:29:27.910
W2210 x8479 Conf Room: and you see the machines that that are there. So fifty again, the top ten machines, fifty-three percent of the overall performance, and then there's this long tail. So if I was doing the top Five hundred again, I would probably say we should probably look at

131
00:29:27.920 --> 00:29:48.590
W2210 x8479 Conf Room: about five hundred machines, but maybe top fifty machines might be a better reflection of high performance computing because those machines at the at the tail there really are not in my mind involved in scientific and scientific computing. If you're interested in where the Nsf. Machines place. So Nsf. Has nine machines on the top five hundred.

132

00:29:48.600 --> 00:29:49.460
W2210 x8479 Conf Room: Yes,

133
00:29:49.590 --> 00:29:50.720
is the lifespan

134
00:29:51.220 --> 00:29:56.609
W2210 x8479 Conf Room: right? So what's the How long can these machines be kept in service, I would say typically

135
00:29:56.620 --> 00:30:25.260
W2210 x8479 Conf Room: typically three years, three to four years is the lifespan. So I think six hundred million dollars. After three years you're going to get rid of it, and you have to replace it. If you want to be involved in high performance. Computing you may be able to squeeze out a little bit more. But you know the the maintenance costs start to start to creep up on these machines, so the cost continues to go up as we look at it. If the technology has changed, it's not worth the the effort to keep the machines at that at that point.

136
00:30:25.440 --> 00:30:27.519
W2210 x8479 Conf Room: So yeah, So there. Yes,

137
00:30:29.530 --> 00:30:30.630
W2210 x8479 Conf Room: yes,

138
00:30:34.400 --> 00:30:35.430
yeah,

139
00:30:37.930 --> 00:31:02.809
W2210 x8479 Conf Room: right. So we can always accelerate the the line by increasing the amount of money that we dump into it. I mentioned earlier Japan. Japan's paid one billion dollars for their computer, and they really accelerated in some sense, pushed the technology ahead of its time. But they were able to do that investment, and the result was to get a machine which was much, much faster than just using the standard stuff.

140
00:31:02.820 --> 00:31:21.709

W2210 x8479 Conf Room: And that's an example of that. So the answer is, yes. So if we need to, we can just dump more money in and cause that to go up. How much it will go up is a question so we can't really predict. I would say, how much we can get for the investment today at that point. But I think that's a good point that we would be

141
00:31:23.200 --> 00:31:24.810
W2210 x8479 Conf Room: Oh, good!

142
00:31:32.370 --> 00:32:01.409
W2210 x8479 Conf Room: Right? So um! So why is the Italian machine doing better in terms of the um in terms of the performance. So the Italian machine is is here, and it's at five point six megawatts. The Italian machine is driven by Nvidia Gpus one hundred and I'm. Sure that has something to do with that notion of having low power, and still having a relatively good.

143
00:32:01.420 --> 00:32:29.490
W2210 x8479 Conf Room: So it's it's. It's merit. The figure of merit. Here is thirty, one which is a reasonably high, a number for flops per watt, which is what's being reflected at. So I would guess it's because of the Nvidia thing and other machines have in video. This is the one that has an a one hundred. We see the similar kind of thing down here with the machine Pearl mother at Berkeley,

144
00:32:29.500 --> 00:32:39.790
W2210 x8479 Conf Room: and also at the Nvidia site itself, using the a one hundred. So it's a much better effective machine for doing computations. I think that's the overall reason.

145
00:32:43.140 --> 00:33:00.599
W2210 x8479 Conf Room: Okay. So there's the top five hundred numbers. There's the machines from Nsf: in that long tail as we see it. China. So China is an interesting case. China has

146
00:33:00.610 --> 00:33:05.219
W2210 x8479 Conf Room: has the lead in terms of the number of supercomputers

147
00:33:05.230 --> 00:33:21.489
W2210 x8479 Conf Room: in the country. So if we take a look at the number of supercomputers. China has one hundred and sixty, two in the Us. We have one hundred and twenty five supercomputers, and then going down the list we see Germany and Japan roughly tied France, Uk. And so on.

148
00:33:21.500 --> 00:33:44.789
W2210 x8479 Conf Room: It also is the one that produces the most supercomputers, so they consume. They have the most supercomputers installed, and they produce the most supercomputers, and they sell those so they sell them to companies like Sugan Lenovo, Innsburg, Howway and and Nu Dt. Is a national University for defense technology. So you know, Lenovo sells supercomputers.

149
00:33:44.800 --> 00:33:47.729
W2210 x8479 Conf Room: I think there's one at

150
00:33:49.280 --> 00:33:50.590
W2210 x8479 Conf Room: okay for even

151
00:33:50.600 --> 00:34:07.989
W2210 x8479 Conf Room: Ah, so some place on the east coast. So anyway, they're they're there, you know. I see a number of Lenovo machines in the audience here, So they're they're a big ah vendor for hive for computers and high-performance machines as well. There's a rumor that there are two ex-scale computers in China.

152
00:34:08.000 --> 00:34:14.149
W2210 x8479 Conf Room: So we know about them because of papers that have been published

153
00:34:14.159 --> 00:34:35.409
W2210 x8479 Conf Room: that reflect applications that have been run on these machines. The first one is the ocean light machine that's the follow on. Ah! To the machine that's listed on the top five hundred today. So this machine is, and there's rumors that they ran the benchmark, but they Haven't submitted it. So the way you get entered in the top five hundred, unless you have to run the benchmark,

154
00:34:35.420 --> 00:34:48.600
W2210 x8479 Conf Room: submit it to the website, and then it'll be looked at, and then put into the into the list. They haven't done that. But we have rumors. We've seen results that suggest that they ran the benchmark at one point three X of,

155
00:34:48.610 --> 00:35:04.389

W2210 x8479 Conf Room: but they Haven't submitted the results. They also have entered their machine in the Gordon Bell Prize, and in fact, they won the Gordon Bell prize two years ago with this machine, and they document what the architecture is, and we know pretty precisely what what's going on in the machine itself.

156
00:35:04.400 --> 00:35:26.270
W2210 x8479 Conf Room: Um, but they Haven't submitted the results, and the same thing is true of the follow-on to the Chinese machine, the tianja Iii, which uses this arm processor plus Ah! An accelerator that was developed in house. It's my speculation that they're concerned about some effect, the the retaliation the Us. May have.

157
00:35:26.280 --> 00:35:33.309
W2210 x8479 Conf Room: So these are Chinese chips they were designed in China. But the question really is, where were they fabricated?

158
00:35:33.320 --> 00:35:51.999
W2210 x8479 Conf Room: And the fabrication place was probably Tsmc. Over in Taiwan. So I think they're concerned that the Us. May cut off Chips going to Taiwan. And if you've read the papers, that's certainly what's what's in the air today? So it's Ah, it's unfortunate that they have it. But I guess I understand why,

159
00:35:52.010 --> 00:35:58.900
W2210 x8479 Conf Room: if we take a look at performance of these of these machines and take a look at the processors.

160
00:35:58.910 --> 00:36:21.059
W2210 x8479 Conf Room: You know the data movement. I mentioned this before. Data movement is one of the most expensive things. If we take a look at data movement. And how it's changed over time. This graph here. This chart here is looking at processors different processors, over time on the X-axis and the Y-axis is looking at the the rate of floating point

161
00:36:21.220 --> 00:36:37.599
W2210 x8479 Conf Room: speed. The flops per second, divided by the data transfer from memory words per second. So we're trying to get a handle on how much floating point capability there is, and how much data movement you can do. So you want this number flops

162
00:36:37.630 --> 00:36:56.530

W2210 x8479 Conf Room: her word to be around one. You're going to be matching floating-point operations to words. If that gets out of balance. If we're doing too many floating point operations or cpu is capable of doing too many. But we can't deliver the data from memory to the Cpu, and we're not going to see the performance. We would expect

163
00:36:56.540 --> 00:37:21.490
W2210 x8479 Conf Room: so numbers greater than one show that there's going to be a problem getting performance on these machines. So in the early days we had machines that had a ratio of one. So the early vaxes nearly cray machines. There was one data word and one one floating point operation being transferred. So that was a very good match of words to data. As time progress the processors got faster.

164
00:37:21.500 --> 00:37:26.880
W2210 x8479 Conf Room: The result was the memory Speeds Haven't kept up with that with that rate.

165
00:37:26.890 --> 00:37:55.819
W2210 x8479 Conf Room: So this is the memory. Wall we talk about, and going forward. We see machines that we are being designed for. Ah, that have the the trade-off or the ratio of one hundred floating point. Operations are capable for every board that can be transferred from memory. So a tremendous imbalance there between them, and that suggests that you had better be using the cash effectively on these machines in order to get performance out of them. And you know we're seeing architectures today that are

166
00:37:55.830 --> 00:38:21.080
W2210 x8479 Conf Room: ah really at ah close to two hundred floating point operations per word transfer, some of those being the gpus that we have on these architectures. So that's ah! That's causing us to get a very low performance on our floating point. Ah applications um, and that brings us, I guess, to this story about full impact benchmark. So the impact benchmark was started in one thousand nine hundred and seventy nine,

167
00:38:21.090 --> 00:38:34.709
W2210 x8479 Conf Room: and it has a lot of positive things associated with the benchmark. It's it's easy to understand. It chose trends, it's. It's quite easy to implement, but you know a lot's changed since one thousand nine hundred and seventy. Nine arithmetic

168
00:38:34.720 --> 00:38:48.359

W2210 x8479 Conf Room: was expensive then. Today, today we're over provision for floating-point operations and they're very inexpensive today on our machines. So the performance that we have on linpack

169
00:38:48.370 --> 00:39:07.859
W2210 x8479 Conf Room: uh, while it perhaps was a good correlator back in one thousand nine hundred and seventy-nine doesn't really correlate Well, today, with real real applications on the machines. The impact benchmark is is really based on doing a dense matrix. Computation. Matrix multiply is what the computational kernel is

170
00:39:07.990 --> 00:39:20.089
W2210 x8479 Conf Room: for that so designing a system which has a good linen pack number Ah can lead to this. Ah, poor ah! Choice of design for for many of the applications that we have,

171
00:39:20.370 --> 00:39:39.689
W2210 x8479 Conf Room: And if we take a look at the applications, we see a number of applications being run on these high-performance machines, climate combustion, nuclear reactor fusion, stockpiled stewardship materials accelerators. Those applications are being driven by things like

172
00:39:39.700 --> 00:39:54.749
W2210 x8479 Conf Room: they are being modeled by three-dimensional partial differential equations. So the three-dimensional partial, differential equation gets discretized that leads to a system of equations that has to be solved. But that system of equations is sparse. It's not dense,

173
00:39:54.760 --> 00:40:24.150
W2210 x8479 Conf Room: so the the basic model that's being being done. Here is a three-dimensional, partial, differential equation and a sparse matrix calculation, which is iterative over time until we converge to a solution that we want So we've designed a benchmark that tries to capture that particular feat of those particular features, and that's a benchmark that's called Hpcg for conjugate gradients a benchmark that solves a system of linear equations. Again,

174
00:40:24.160 --> 00:40:52.359
W2210 x8479 Conf Room: we have a large sparse matrix that was composed ah, based on a twenty, seven point stencil, similar to Ah, what's used in the three-dimensional problem of the data structures that we have are represented by this Ah, by this pattern here of this matrix and

the benchmark tries to exercise a number of things of an architecture, to see how it would perform for real applications. So this is a

175
00:40:52.370 --> 00:41:20.370
W2210 x8479 Conf Room: a list of the top ten for this, this, this other benchmark. Hpcg: So this is the most recent list, and the machine at the top of the list is is the machine from Japan. So the Rican system, Fogaku, the Fujitsu machine is the number one machine, and what I've done here is to put the limp pack number for the machines. That's the impact number of a fugaku system ran at four hundred and forty, two petaflops,

176
00:41:20.380 --> 00:41:42.990
W2210 x8479 Conf Room: and the the performance for Hpcg is listed here, and that's at sixteen sixteen petaflops. So you could see the difference between a dense matrix calculation and the sparse matrix calculation. And what's really being done on these machines is more like that sparse matrix calculation,

177
00:41:43.000 --> 00:41:58.660
W2210 x8479 Conf Room: and the the last column here reflects the percent of peak performance. So again, the theoretical peak performance of a machine is just a paper and pencil calculation that we do. And now we're going to look at the performance we got on Hpcg. And compared to that theoretical peak.

178
00:41:58.730 --> 00:42:00.350
W2210 x8479 Conf Room: In this case

179
00:42:00.480 --> 00:42:05.469
W2210 x8479 Conf Room: the fugitive system gets three percent of the theoretical peak, three percent.

180
00:42:05.530 --> 00:42:12.399
W2210 x8479 Conf Room: The machine at Oak Ridge gets zero point eight percent. So it's getting less than one percent of the theoretical peak

181
00:42:12.410 --> 00:42:32.230
W2210 x8479 Conf Room: in this application. And this application is more characteristic of real applications that are that are being run on these machines, and you know the the number three

machine is Ah is a a carbon copy, but smaller version of the machine that operates machine in Finland,

182
00:42:32.240 --> 00:42:39.590
W2210 x8479 Conf Room: and it has the same processor as just a lower number, and they also get zero point. Eight percent of the theoretical peak.

183
00:42:39.600 --> 00:42:46.690
W2210 x8479 Conf Room: So think about a race car. Think about a race car that has the ability to go two hundred miles per hour, and we're getting

184
00:42:46.700 --> 00:43:14.950
W2210 x8479 Conf Room: two miles per hour out of this race car so you could walk faster. You know. That's that. Says there's a problem somewhere, and that needs to be looked at or addressed somehow. And the problem is, you know, we're not designing these machines for the kinds of applications that we have, or ever intending to run at. So this is a graph again at the top five hundred numbers and ranking them. So what's being plotted here is the theoretical peak performance compared to the

185
00:43:14.960 --> 00:43:30.099
W2210 x8479 Conf Room: Lynn Patton or Hpl. We're very close together, and this is the Hpcg number. So again, there's a spread between between those two things showing that the performance is not there.

186
00:43:30.720 --> 00:43:58.390
W2210 x8479 Conf Room: Okay, So recently we've seen machine learning and Ai taking off our machine. Learning's been around for a long time. So why? Why, now? Why is it a big deal Now, as the question. Well, one reason is, we have this tremendous amount of data that we can use to train our machine learning of ah applications. We have this increased computational power that can be used to drive that training where we have a growing progress that we've made in algorithms of theory

187
00:43:58.400 --> 00:44:04.949
W2210 x8479 Conf Room: that's been behind that, and increasing support from it from industry itself.

188
00:44:04.960 --> 00:44:33.780

W2210 x8479 Conf Room: And you know artificial intelligence is a broad thing. So there's machine learning. There's a natural language processing thing, chat, gpt. There is expert systems. There's vision, speech, recognition, planning, robotics. Go into the overall, mix of Ai, and we're seeing more and more of machine learning being put into the scientific applications in terms of climate, biology, materials, cosmology.

189
00:44:33.790 --> 00:44:40.489
W2210 x8479 Conf Room: Almost every place you turn, you see a paper being written about what's been what's been learned, or how it's been.

190
00:44:40.500 --> 00:44:48.690
W2210 x8479 Conf Room: Move forward through the help of machine learning. Machine learning is not solving the problem, but it's helping in the solution to the problem.

191
00:44:48.700 --> 00:45:07.010
W2210 x8479 Conf Room: And machine learning needs matrix computations. We need small matrix computations, but they can get by with less precision. So that's where the sixteen bit floating point comes in in terms of doing that, doing those operations. And what we've seen is a number of companies being spun up

192
00:45:07.020 --> 00:45:22.089
W2210 x8479 Conf Room: to develop hardware that can address the issues around machine learning, you know, companies like Cerebrus and and Sombinova and Graph Corps and Baidu has a company, and and

193
00:45:22.100 --> 00:45:32.090
W2210 x8479 Conf Room: and so on. So a lot of activities. It's sort of reminiscent of the old days when a lot of parallel processing companies were put in place.

194
00:45:32.400 --> 00:46:01.689
W2210 x8479 Conf Room: If we take a look at the money that's going into these. We have this incredible situation. So here's the market capitalization of a number of companies. So these are the perhaps traditional companies that have been involved in Hpc. Here and aggregate are about a trillion dollars, and when you take a look at some of the cloud-based companies that we have companies that are doing things in the machine learning space. We have this incredible amount of

195

00:46:01.700 --> 00:46:17.870
W2210 x8479 Conf Room: that they have going into them, and that's also true of the Chinese companies as well. So you know well over trillions of dollars going into them, pushing them forward. So we have that incredible situation.

196
00:46:17.880 --> 00:46:46.820
W2210 x8479 Conf Room: If we take a look at the hardware architectures that we have, because this graph here is trying to. But ah, we're trying to help us sort through Ah, architectures that have evolved over time. Ah, in the sixty s there was Ah, these superscalar machines, Cdc. Six thousand six hundred of the Cdc. Seven thousand six hundred used for scientific computations. There were experimental machines like the Iliac that were put in place to do experiments with parallel processing.

197
00:46:46.830 --> 00:47:15.530
W2210 x8479 Conf Room: We had vector supercomputing and a lot of experimental machines developed for parallel processing, you know, sort of a rich time to learn and experiment with with these in the eightys. Then we had the attack of the killer micros, where we had a tremendous number of companies coming in place, supercomputers being built with those microprocessors, and today we have this mixture of accelerators

198
00:47:15.540 --> 00:47:43.690
W2210 x8479 Conf Room: put together with commodity processors that fuel the high-performance computing and we have the cloud companies going off and designing their own hardware to really satisfy to really satisfy the needs of of their specific applications. So some of the conclusions here that you know the the computing ecosystem is is an enormous flux. Today it's creating both opportunities and challenges

199
00:47:43.700 --> 00:48:12.680
W2210 x8479 Conf Room: for the future of advanced scientific computing. Looking forward, it seems unlikely that the future Ah will be able to procure and assemble these machines. Ah, solely from commercial. Ah, integrators! Um! We have advances that are coming about. We really need what I would call an end to end co-design of our hardware. That does not take commodity processors and assemble it because it's the cheapest thing that we could

200
00:48:12.690 --> 00:48:41.129
W2210 x8479 Conf Room: put together in a box, but really sit down and develop a a design that really matches the needs of the applications that we're intending to use on these machines. So these leading edge systems are going to be similar, perhaps, to large-scale scientific

instruments which are designed to really carry out advanced science, like the large Hadron collider of Wygose gravitational

201
00:48:41.140 --> 00:48:53.640
W2210 x8479 Conf Room: thing, or the square kilometer array for radio telescope. There's limited economic incentives, of course, for the commercial development of these systems.

202
00:48:54.280 --> 00:49:23.430
W2210 x8479 Conf Room: There is some hope, I would say in terms of this idea of Co-design Ah! With with the use of chiplets and the ability to put together based on standards, a processor that has certain components that can be fabricated at a reasonable cost. So if we think about putting together a system like this based on chiplets, we might think of a chiplet that has the ability to do Ffts

203
00:49:23.440 --> 00:49:53.420
W2210 x8479 Conf Room: put that chiplet on the substrate itself, so that it could do it very rapidly, or or other components, perhaps to a singular value decomposition around the chiplet. So we can do that very rapidly, and not have to pass data around throughout the chip to engage in that. So there's I think, a time where we'll be looking at this technology for our for our hardware. So the hardware is constantly under change going from scalar to

204
00:49:53.430 --> 00:50:22.110
W2210 x8479 Conf Room: to distribute it to accelerated to the use of mixed precision. There is really sort of three computer revolutions.

205
00:50:22.120 --> 00:50:51.759
W2210 x8479 Conf Room: It's a situation where the hardware guys put together an architecture based on some specifications and based on some amount of funding that's available. They put that together, using commodity parts, and Then they throw it over the fence and the applications. People and the software guys have to go out and try to figure out how to use that architecture for the next three or four years before the next machine gets thrown over the fence, and then they scramble again to do it. So we need a better system.

206
00:50:51.770 --> 00:51:20.920
W2210 x8479 Conf Room: We need a system where we get in the room with the hardware guys, along with the applications, people and the software of people to design a machine that can effectively solve the applications that we have, and that's that's in a sense, what's going on with the cloud guys designing their ah, their processors that can solve the needs for their

specific problem in a way that makes sense. So there was this great paper that was written a couple of years ago by Charles Leicester and a Company.

207
00:51:20.930 --> 00:51:37.990
W2210 x8479 Conf Room: A paper title is something like There's plenty of room at the top. What will drive computer performance after Moore's law. They look at some of these issues about data movement, how we could do things a little bit better and take care of it. I would recommend

208
00:51:38.000 --> 00:51:42.890
W2210 x8479 Conf Room: looking at it. It's a paper that appeared in science back in two thousand and twenty,

209
00:51:42.900 --> 00:52:00.079
W2210 x8479 Conf Room: and and of course it's a riff off off of Ah Feynman's lecture about. There's plenty of room at the bottom where he talks about quantum effects and quantum computing. Ah, back in one thousand nine hundred and fifty nine. So with that I think i'll end, and i'd be happy to take any questions that we might have

210
00:52:00.090 --> 00:52:03.000
W2210 x8479 Conf Room: uh from people here or there,

211
00:52:10.320 --> 00:52:12.160
open up for questions

212
00:52:19.260 --> 00:52:21.700
questions online. But let me open it up.

213
00:52:24.390 --> 00:52:25.830
Thank you

214
00:52:45.990 --> 00:52:47.080
to the

215
00:52:50.770 --> 00:52:52.589

presentations at the Ceo

216
00:52:53.430 --> 00:52:55.860
generally just looking at

217
00:53:08.220 --> 00:53:26.130
So well, we have the top five hundred data that we can that we can mine And look at. If you take a look at the top five hundred data, and you look at where the machines are being deployed. Half of the half of the machines are being used

218
00:53:26.140 --> 00:53:43.860
W2210 x8479 Conf Room: for industry. So industry, you know, sort of understands that they can use high performance computing. Now, most of those machines are not at the high end. They're at the low end or at the longer tail of the five hundred machines, but they are. They're they're invested.

219
00:53:43.870 --> 00:53:57.080
W2210 x8479 Conf Room: They understand that these machines provide some kind of competitive advantage, and they're willing to buy those machines to carry out their applications bring on a good point. If the machines are too high, too expensive, who's going to buy them?

220
00:53:57.090 --> 00:54:09.620
W2210 x8479 Conf Room: So we had a situation back in the back in the seventies, when Craig was designing a special-purpose machine, he sold a few machines, but they couldn't sustain that business

221
00:54:09.630 --> 00:54:26.590
W2210 x8479 Conf Room: over the long run, because the market wasn't. There. People were not willing to pay for it, and we had technology changing very rapidly, as well tech on the killer microprocessors. But you know. The real strangle point here

222
00:54:26.600 --> 00:54:43.189
W2210 x8479 Conf Room: is moving data, and if we can design a system which would be able to effectively move the data and match in some sense what the processors are capable of doing in terms of the floating point operations. We have a much more effective machine

223

00:54:43.200 --> 00:54:52.420
W2210 x8479 Conf Room: when you're getting less than one percent of the theoretical peak performance out of the machine. It says you've done something wrong. It's a radical

224
00:54:52.430 --> 00:55:06.630
W2210 x8479 Conf Room: for a machine to be used for scientific purposes. And you know that's being designed because we were trying to drive the price down by using commodity parts. That's That's how we came to that to that for that decision

225
00:55:19.780 --> 00:55:20.970
we've been writing.

226
00:55:26.430 --> 00:55:34.450
W2210 x8479 Conf Room: So they are. You know you have a car manufacturer is doing it. People designing jet engines, I mean every place you turn almost an industry.

227
00:55:34.880 --> 00:55:55.900
There's there's a need for solving these kind of problems and having a very effective machine to do, it would solve that. Now it's the cost of the machine that that's really going to drive. Whether or not the take uptake is, is there, so that that's going to be the the fragile point here, the point where everything can collapse over so designing a machine at that level,

228
00:55:55.910 --> 00:56:04.709
on the other hand, is, you know, if these machines are important enough for the Government, and the government's going to do it. So you know we

229
00:56:04.900 --> 00:56:10.320
So you think of an aircraft carrier, right? So that that's an important device that we have to have,

230
00:56:10.330 --> 00:56:26.250
W2210 x8479 Conf Room: and we invest, you know, a tremendous amount of money and getting that resource, we're only going to build a few of them, but because of the need for that, we were willing to invest in it. So if these machines are useful enough, the money should be invested.

231

00:56:34.020 --> 00:56:35.509
Oh, thank you. Yes,

232
00:56:38.670 --> 00:56:39.700
questions.

233
00:56:43.690 --> 00:56:44.740
Five ups.

234
00:56:51.430 --> 00:56:53.210
We have the quantum.

235
00:57:01.210 --> 00:57:12.189
W2210 x8479 Conf Room: Ok. So quantum computing we're entering the quantum winter. In some sense I would say so. A lot of hype has been made about quantum computing,

236
00:57:12.200 --> 00:57:29.189
W2210 x8479 Conf Room: and you know I have to say Ah, it has a definite role. Quantum Computing is important, and it will play a part in it, but it's not going to replace what we have. It's not going to replace digital the digital computers that we know and love and quantum computing is going to be a device which is used.

237
00:57:29.200 --> 00:57:40.490
W2210 x8479 Conf Room: It helps solve certain problems. It's not It's not the thing that's going to solve the at those pdes, for example, there's no way that it's going to be able to do that, at least in my lifetime. It's not going to,

238
00:57:40.500 --> 00:57:58.969
W2210 x8479 Conf Room: And so so I think we should invest in research into understanding how they can effectively be used. That would be a good investment, I would say, for and Nsf. To make in quantum computer. Try to understand better how these machines can be used and what the limits, and where the applications are.

239
00:57:58.980 --> 00:58:06.290
W2210 x8479 Conf Room: So in some sense, I think, of what's the architecture that we have. So today we have commodity processors

240
00:58:06.300 --> 00:58:19.089
W2210 x8479 Conf Room: and accelerators, and maybe in the future we'll have a quantum device attached to that mix, and that quantum device will be used not to solve the problem, but to solve a part of the problem

241
00:58:19.100 --> 00:58:35.949
W2210 x8479 Conf Room: that can effectively use that quantum, that quantum device. So I would say that you know that's how I see it fitting in into the spectrum of things along with other devices that will be put on that spectrum miramorphic computing or optical computing. Or,

242
00:58:35.960 --> 00:58:52.940
W2210 x8479 Conf Room: you know, machine learning, computing specialized processors to help with that, and they would effectively solve that part of the problem. But the overall thing would be controlled through a natural to through the conventional way of doing the computation.

243
00:58:52.990 --> 00:58:55.189
That's my that's my view of,

244
00:58:55.200 --> 00:59:06.919
W2210 x8479 Conf Room: and yes, cubits will increase. People will make outrageous claims about what the capabilities are, and they'll go into their winter, and people will be discouraged

245
00:59:08.350 --> 00:59:09.930
about the whole process.

246
00:59:15.460 --> 00:59:16.539
We're pretty cool,

247
00:59:28.950 --> 00:59:36.709
W2210 x8479 Conf Room: incredible. Are you important? Did you get the right answer So you can get her fast. But did you get it.

248

00:59:36.790 --> 01:00:04.889
W2210 x8479 Conf Room: Yeah, the question is about what about correctness? And what can we do to help ensure correctness? Perhaps in our in our confrontations I always think we should have a sense of what the answer is going to be, and try to match that up. We have some way of checking. Ah, what the what the answer. When I was a I was a graduate student. I worked at Los Alamos, and I was running on the cray. One frame of just came in, and I was running for the loophole,

249
01:00:04.900 --> 01:00:07.880
W2210 x8479 Conf Room: and I got the wrong answer.

250
01:00:07.890 --> 01:00:37.099
W2210 x8479 Conf Room: And you know the the the linpack benchmark has a test to see if you've got the right answer, you calculate some residual, so it's not a big deal, but it but it but it could tell you if you got the right answer. We're up to rounding here right so there. There's some very serious theory that's in place, and um! I was going to run it by studying my code, looked at it for a couple of weeks. I couldn't see what the I couldn't see any bug if I looked at the assembled code to see what the compiler had maybe had done something wrong.

251
01:00:37.170 --> 01:00:39.549
There's nothing wrong that the compiler was doing.

252
01:00:39.580 --> 01:00:42.889
I went to the hardware guys, and I said,

253
01:00:42.900 --> 01:00:44.059
I think maybe

254
01:00:44.070 --> 01:00:56.930
i'm seeing a problem here. Maybe there's something wrong with the hardware, and you know the guys got out there stuff the first day. I said, Yes, we're right. It was persistent, and they finally they finally tracked it there, and there was a problem with the hardware.

255
01:00:56.940 --> 01:01:24.470
W2210 x8479 Conf Room: So you know, I think it's important that we have a sense of what the answers are. We can't rely on these machines, or believe these machines that they're going to

give you on the correct answer. So yes, I think it's an important area. There's a lot of work going on in

256
01:01:24.480 --> 01:01:40.639
W2210 x8479 Conf Room: on verification and uncertainty, quantification, and I think all of those things are important, and we need to stress that in our applications we shouldn't just believe that the numbers that come out and trust that they're correct,

257
01:01:41.480 --> 01:01:55.520
W2210 x8479 Conf Room: you know. He has this problem, you know, running the thing today and tomorrow. You hope that it's deterministic, and you're going to get the right. The same answer. But you're not, and you're not because it's done in parallel, so I can't guarantee you

258
01:01:55.560 --> 01:01:59.420
which things get summed or accumulated so

259
01:01:59.430 --> 01:02:20.920
W2210 x8479 Conf Room: parallel can't guarantee that I can if I spend more time precision. If i'm willing to live with that, and everything is okay, then I should be okay.

260
01:02:20.930 --> 01:02:30.709
W2210 x8479 Conf Room: But some people insist on getting the same results today and tomorrow, and they have to do something which slows down the computation to ensure that the order is done correctly.

261
01:02:32.150 --> 01:02:36.829
Maybe you'd like to have that switch that you could turn to to do that verification at some point

262
01:02:44.350 --> 01:02:45.470
I dragged me up in.

263
01:02:46.200 --> 01:02:48.820
How much is this hardware in the algorithm

264
01:02:55.240 --> 01:02:58.499

W2210 x8479 Conf Room: So we're talking now about computing at the edge,

265
01:02:58.510 --> 01:03:28.370
W2210 x8479 Conf Room: talking about pushing more computational stuff at the edge of the at the edge of the computation being able to try to do some machine learning so that we can filter data with the edge, computing It's all about filtering data. So you don't want to pass data all the way to the main thing. So we have edge computing as to some other computing device, perhaps up to Dan's supercomputer, and where the computation is done, so we need something

266
01:03:28.380 --> 01:03:54.490
W2210 x8479 Conf Room: that'll help refine it and to filter that data before it gets pushed upstream to the

267
01:03:54.500 --> 01:04:04.850
W2210 x8479 Conf Room: is actually done so there will be compute at the edge. I don't think i'll have the same capability as the compute at the far reaches. But there will be some computer.

268
01:04:20.320 --> 01:04:28.989
W2210 x8479 Conf Room: So the question is about ah, processor and memory, Tim, architecture, and what does that? What does that hold is that something we should be thinking about.

269
01:04:29.000 --> 01:04:35.160
W2210 x8479 Conf Room: So I think that's ah, that's a great, that's a great idea to be investigated to continue to be investigated.

270
01:04:35.170 --> 01:04:36.840
So putting

271
01:04:36.870 --> 01:05:04.920
W2210 x8479 Conf Room: so it's like a memory space, and then think of scattering processors in the memory, so that the processors are very close to the data. So we don't have to do that long distance communication, wasting hundreds, thousands of cycles from one point to another. That architecture is a research architecture. We don't have existence of machines like that today, or the machines that I know about. But that's an important research here.

272
01:05:04.930 --> 01:05:13.189
W2210 x8479 Conf Room: So I think that should be investigated as a research area, and that could lead to advanced architectures. I would say

273
01:05:32.270 --> 01:05:34.539
so, and I just We arrived for that.

274
01:05:59.110 --> 01:06:20.890
W2210 x8479 Conf Room: So the question is about the brain also has this problem of moving data around. It takes a one time from a point where you get stabbed until you feel the pain or hand as birds until it comes there. So it's a data movement problem in some sense. So the brain has that. And then part of it. The second part was about

275
01:06:20.900 --> 01:06:21.990
all right.

276
01:06:22.000 --> 01:06:31.689
W2210 x8479 Conf Room: Energy, usage, that energy consumption of these things. So how can we reduce the energy, consumption and footprint for the computer itself?

277
01:06:31.700 --> 01:06:50.849
W2210 x8479 Conf Room: So, Um. So a tremendous amount of energy is going into moving data, and not a lot of energy is going into doing the computation. So the vibration there is saying, You know you don't spend much on the of doing the floating point operations. You really are paying for moving the data. So again, it suggests that we need some way to

278
01:06:51.010 --> 01:07:07.219
W2210 x8479 Conf Room: effectively move the data to the processor or have it closer to the I mean. The other thing we can think about is having shorter precision.

279
01:07:07.230 --> 01:07:35.049
W2210 x8479 Conf Room: So we're moving around sixty four-bit words doing a computation if we can move around sixteen bit words. That's a better trade-off four times faster in some sense. By doing that, and the operations move a lot quicker as well, so sort of double bonus. Unfortunately, the ratio is going to be the same four times faster, moving to data, and four

times faster, doing the operation. So we don't gain anything in that except the energy cost is going to.

280
01:07:35.060 --> 01:07:44.230
W2210 x8479 Conf Room: It is a win in terms of the energy budget for that it may not be for the computation. So I would say that we should look at

281
01:07:44.390 --> 01:07:48.780
research going into the use of short precision

282
01:07:49.370 --> 01:07:55.929
in our computations can we get by with mixing precision, and we get by with using

283
01:07:55.940 --> 01:08:18.140
W2210 x8479 Conf Room: sixteen bit arithmetic points in the computation, and then switch to higher precision when needed. We still want to get this full accuracy, perhaps the full fidelity of the computation. But maybe we could start the computation with shorter precision, and that would have a less impact in terms of that energy budget at the beginning. At the end. We need to have full precision.

284
01:08:40.300 --> 01:08:59.650
W2210 x8479 Conf Room: So dark. Silicon we're talking about. We're going to do so. I don't think we can get by with

285
01:08:59.720 --> 01:09:03.069
with turning those things uh turning those things off,

286
01:09:03.080 --> 01:09:23.070
W2210 x8479 Conf Room: you know, with the gpus. It has less enough energy to use the less energy for a floating-point operation, because it's simpler so in gpus. They've removed a lot of stuff it's Turn a lot of things out that are conventional. Cpus have a lot of the instruction decoding, and a lot of the

287
01:09:23.080 --> 01:09:25.090
sophisticated

288
01:09:25.100 --> 01:09:31.489
W2210 x8479 Conf Room: processing of instructions has been thrown away, and they just do floating point. Very

289
01:09:31.500 --> 01:09:49.960
W2210 x8479 Conf Room: only essentially is to the flowing point in parallel. So they know what operations they're going to do, and they can do them in parallel. And now it's a question of just moving the data. So we have data floating point operations, and they can reduce the cost compared to something else. So that's why the gpus are

290
01:09:49.970 --> 01:10:01.289
energy efficient overall than we see on the other machines. That's why in the benchmarks we see them turning out to be a better deal, if you will, for that, for that reason.

291
01:10:01.300 --> 01:10:07.569
W2210 x8479 Conf Room: So yes, if we could turn off parts of the machine, that would be a good thing.

292
01:10:07.780 --> 01:10:17.590
W2210 x8479 Conf Room: We still need to move data. That's an expensive part of the energy budget floating point isn't as expensive as that,

293
01:10:26.060 --> 01:10:27.099
all of

294
01:10:46.630 --> 01:10:49.599
W2210 x8479 Conf Room: right. So it's a little bit out of my uh

295
01:10:49.840 --> 01:11:07.509
W2210 x8479 Conf Room: my lane talking about these things. But you know, at one time people thought would be programming, and all of our scientific computing.

296
01:11:07.520 --> 01:11:15.999
It's done in a very traditional way. C. Using that. Yeah, I don't think i'm prepared to answer the kubernetes for a call

297
01:11:39.750 --> 01:11:57.839
this is the hard question is, Ah, the training aspects. And how do you train people that Ah can be? Ah, our leaders in terms of developing software algorithms even hardware for high-performance computing systems that are going to come through the universities.

298
01:11:57.850 --> 01:12:03.190
They need a strong sense of you know for the scientific things they need strong senses,

299
01:12:03.200 --> 01:12:09.729
W2210 x8479 Conf Room: numerical computations, of course, programming understanding how to program in parallel.

300
01:12:09.740 --> 01:12:37.599
W2210 x8479 Conf Room: You know, advanced concepts of programming. I teach, of course, scientific computing for engineers, and we typically have people not in computer science, but from the engineering field. And I ask, who knows how to program and they all raise their hand. But then, when you start probing and say, well, you know, what do you know about recursion? What do you know about data structures about databases? And they're not quite certain they know how to program loops. They know how to program.

301
01:12:37.610 --> 01:12:49.399
W2210 x8479 Conf Room: So we need a better way of training people who are who can be skilled in these things to help build the applications. That's why it's so. I'm so concerned about this.

302
01:12:49.570 --> 01:13:06.690
W2210 x8479 Conf Room: I mentioned earlier about the Department of Energy at the end of the ex-scale computing program. There's one thousand people employed very trained, highly skilled knowing how to use high-performance systems sort of the leaders in the field.

303
01:13:06.700 --> 01:13:15.970
And now, without funding, I see many of them being picked off and going to other companies, Amazon and Facebook. Google.

304

01:13:15.990 --> 01:13:37.819
W2210 x8479 Conf Room: That, I would say, is something we need to pay attention to, so that, I would say is something we need to pay attention to

305
01:13:37.830 --> 01:13:40.039
and look at it more serious.

306
01:14:09.010 --> 01:14:20.889
W2210 x8479 Conf Room: So that's an important thing it's about benchmarking we should have in my mind we should never have just one bench that makes no sense at all. You can have a whole spectrum of benchmarks

307
01:14:20.900 --> 01:14:38.889
W2210 x8479 Conf Room: that probe the architectures in a way that can be reflective of the applications that we are using today, and that we intend to use in the future. That would be the right way to do it. There. There are activities going on in the Commons, I think, is one that that people are experimenting with different benchmark,

308
01:14:38.900 --> 01:14:54.190
W2210 x8479 Conf Room: looking at the performance, trying to understand how these things can be compared, and those are all very excellent ideas,

309
01:14:54.200 --> 01:15:12.739
W2210 x8479 Conf Room: and in some sense, Then, with that suite we can dial up which ones are important for our applications, and then see what the what the effects would be on our application for a specific machine. That would be the right way to do it, and people have tried it and have experimented with that over the years.

310
01:15:12.750 --> 01:15:16.590
But we don't have anything concrete and in place

311
01:15:16.600 --> 01:15:28.250
W2210 x8479 Conf Room: Yes, we should do that. We should have a community or the community should get together, and they maybe make a decision about benchmarks and having that scale filled out so that we can

312

01:15:38.790 --> 01:15:40.139
brain of one of our

313
01:15:40.650 --> 01:15:42.559
It's an executive. Software

314
01:15:47.860 --> 01:16:06.380
W2210 x8479 Conf Room: Well, software is the important, most important part. So I would. I would look at it, and I would say we have definitely not taken it as seriously as we should in terms of the development, you know, software is something that gets developed.

315
01:16:06.390 --> 01:16:16.769
It gets implemented, and it has to be maintained over its lifetime. So we often don't factor into the equation, we

316
01:16:16.860 --> 01:16:29.149
the maintenance of the software in the long term. Again, I'll come back to the Department of Energy Example, where we have the Ecp projects develop this software stack

317
01:16:29.160 --> 01:16:47.740
W2210 x8479 Conf Room: and that software stack is in place. But in eleven months there'll be no funding to maintain that. Software So we've invested this enormous amount of funding into the development of it, but no way to maintain it as we go forward. And I think that's really a

318
01:16:47.750 --> 01:17:02.499
a reflection of what that question is asking, and we need to pay more attention to that, and put that into the original funding that we place on the system itself

319
01:17:02.570 --> 01:17:06.270
that maintenance of that long term maintenance of the software

320
01:17:41.130 --> 01:17:42.140
pretty much with it.

321
01:17:55.760 --> 01:18:07.690

W2210 x8479 Conf Room: So we hope that we hope we can get to a point where that's not making a small change to a code as this tremendous impact.

322
01:18:07.700 --> 01:18:16.190
W2210 x8479 Conf Room: Kill the scalability. Well, that's always going to happen. But yeah, So that's one of these things that's going to be hard to predict.

323
01:18:16.200 --> 01:18:33.199
W2210 x8479 Conf Room: So yeah, I don't. I don't think there's a there's a clean, clean solution to that problem being able to have someone other than the author of the code going in and modifying things by somebody who is untrained in the code,

324
01:18:33.210 --> 01:18:48.689
making changes to it that may have an impact in terms of the solution, the accuracy, or, as you point out, the performance of some innocuous modification to a code.

325
01:18:48.700 --> 01:18:52.529
Uh, so I I I don't know how to resolve that

326
01:18:59.730 --> 01:19:00.760
all of them

327
01:19:07.090 --> 01:19:08.350
it was.

328
01:19:14.350 --> 01:19:19.669
W2210 x8479 Conf Room: Yes, I want advice. So what do I tell my graduate students,

329
01:19:20.590 --> 01:19:35.559
So I tell my graduate students, they should find something that they're passionate about, that. They have a real passion. What they want to affect things they should aim high. They should look at problems

330
01:19:35.570 --> 01:19:50.880

which are challenging problems. They shouldn't look at things that are simple to solve. They should expect to fail. So failure is part of research. It's something that's part of the game. We're going to try out things. And you're going to fail

331
01:19:50.890 --> 01:20:03.450
W2210 x8479 Conf Room: at times, and and that's that's just part of the way it's done. Can't it all run every every turn you should network. You should get together and interact with other people and talk to them

332
01:20:03.460 --> 01:20:19.140
W2210 x8479 Conf Room: about the problems you're having, and ask advice, and maybe you can help somebody else in

333
01:20:19.230 --> 01:20:36.589
W2210 x8479 Conf Room: a problem to go off and work on. So they go off and work on the problem. We'll come back two weeks later. They say I can't solve this, I say, Well, it's a research problem. I didn't expect you to go off and solve it in a day or two. You should think harder about it and and work at it, and try to come up with a solution.

334
01:20:36.600 --> 01:20:37.589
Look at it!

335
01:20:37.600 --> 01:20:47.090
What other people have done in the field to to look at this problem, and they go off, and eventually they they do come back and and have a handle on on what they do to solve it.

336
01:20:47.100 --> 01:20:52.590
W2210 x8479 Conf Room: So yeah, So it's a complicated thing. They should

337
01:20:52.610 --> 01:21:04.560
find something that they're passionate about that's going to drive them. Try to get something that is not trivial to solve. Aim high,

338
01:21:04.590 --> 01:21:23.390

W2210 x8479 Conf Room: try to engage with the colleagues to try to solve the problem and bounce ideas off of other people to get a handle on what's going on, and an expected family. That's just part of the game. That's what we all have to do in our research.

339
01:21:28.250 --> 01:21:29.279
All the parents

340
01:21:33.600 --> 01:21:39.219
you can. Somebody can email me, of course, and i'll be happy to follow up with that, and be easy to do

341
01:21:39.760 --> 01:21:40.989
all of this. Thank you.