Graduation Ceremony

With Dr. Prasad during Graduation Ceremony waiting to be officially announced as a "Dr."

Feed aggregator

PRACE 14th Call for Proposals Awards Nearly 2000M Core Hours

HPC Wire - Tue, 04/18/2017 - 10:53

April 18, 2017 — The 14th PRACE Call for Proposals yielded 113 eligible proposals of which 59 were awarded a total of close to 2 thousand million core hours. This brings the total number of projects awarded by PRACE to 524. Taking into account the 3 multi-year projects from the 12th Call that were renewed and the 10 million core hours reserved for Centres of Excellence, the total amount of core hours awarded by PRACE to more than 14 thousand million.

The 59 newly awarded projects are led by principal investigators from 15 different European countries. In addition, two projects are led by PIs from New Zealand and the USA.

This Call was the first under the recently ratified PRACE 2 Programme (http://www.prace-ri.eu/prace2-council-ratification/). As a new feature of this programme, 3 project proposals are tagged Candidate Tier-0 and will receive tailored assistance from a High-Level Support Team (HLST) to push their excellent research towards more advanced use of world-class high performance computers.

Seven scientific domains are represented: 8 projects are linked to the fields of Biochemistry, Bioinformatics and Life Sciences; 22 to Chemistry; 4 to Earth System Sciences; 4 to Engineering; 9 to Fundamental Constituents of Matter; and 12 to Universe Sciences.

The 14th PRACE Call for the first time included the Piz Daint system from CSCS, Switzerland, PRACE's newest Hosting Member. 8 projects were awarded a total of close to 6 million node hours (401 million core hours) on this system. Hosting Members BSC (Spain) and CINECA (Italy) have recently upgraded their systems (MareNostrum and Marconi respectively) and these are now also available for PRACE Calls.

Amongst the renewed multi-year projects from Call 12, Charge and Spin Hall Kubo Conductivity by Order N Real Space Methods led by Dr. Stephan Roche is linked to the FET Graphene Flagship program. The project received 20 million core hours on MareNostrum @ BSC, Spain.

One PRACE-awarded project is linked to the FET Human Brain Flagship program:CBNR – CereBellar Network Reconstruction led by Prof. Egidio D'Angelo. The project received 32 million core hours on Juqueen @ GCS @ JSC, Germany.

One PRACE-awarded project is linked to the Autism Research and Technology Initiative: iHART – Characterization of genetic risk variants in ASD families using a reference-free approach led by Dr. Daniel Geschwind. Taking into account its potential societal impact, it was awarded 5.5 million core hours on MareNostrum @ BSC, Spain.

Two projects are led by PIs from industry: EDF (France, energy-related R&D in the field of materials) and Cenaero (Belgium, public/private R&D centre in aeronautics/combustion), and another involves a Co-PI from industry: Termo Fluids S.L. (Spain, SME in fluid mechanics).

The PRACE 14th Call awarded resources to 9 ERC and 2 Marie Sklodowska Curie funded projects, 2 H2020 funded projects, 2 EC FET Flagship programmes, 2 projects with links to the MaX Centre of Excellence and 2 projects with links to EUROfusion and the European Strategic Energy Technology Plan (https://ec.europa.eu/energy/en/topics/technology-and-innovation/strategic-energy-technology-plan).

All information and the abstracts of the projects awarded under the 14th PRACE Call for Proposals can be found here: http://www.prace-ri.eu/14th-project-call/ (going live soon).

The **14th Call** for Proposals for PRACE Project Access (Tier-0) was open from 10 October until 21 November 2016. Selected proposals will receive allocations to PRACE resources from 3 April 2017 to 31 March 2018.

About PRACE

The Partnership for Advanced Computing in Europe (PRACE) is an international non-profit association with its seat in Brussels. The PRACE Research Infrastructure provides a persistent world-class high performance computing service for scientists and researchers from academia and industry in Europe. The computer systems and their operations accessible through PRACE are provided by 5 PRACE members (BSC representing Spain, CINECA representing Italy, CSCS representing Switzerland, GCS representing Germany and GENCI representing France). The Implementation Phase of PRACE receives funding from the EU's Seventh Framework Programme (FP7/2007-2013) under grant

agreement RI-312763 and from the EU's Horizon 2020 research and innovation programme (2014-2020) under grant agreements 653838 and 730913. For more information, see www.prace-ri.eu.

Source: PRACE

The post PRACE 14th Call for Proposals Awards Nearly 2000M Core Hours appeared first on HPCwire.

Categories: RSS Feed Reader

Article "Styles" In Action

Intel News - Mon, 04/17/2017 - 16:45

The intro paragraph is a wonderful way to highlight the start of your article. This extra large copy helps you focus the developer on your main purpose. Simply select your copy and open the "Styles" dropdown in the WYSIWYG editor. Choose "Intro" and your copy should transform automatically.

This is the default for any copy you enter into the editor. Vestibulum id ligula porta felis euismod semper. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Maecenas sed diam eget risus varius blandit sit amet non magna. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Morbi leo risus, porta ac consectetur ac, vestibulum at eros.

Did You Want to Highlight Some Content?

Go to the "styles" dropdown and select the "outline" style. You'll get a simple grey outline around your copy. A few things to note: bullets, header tags, and multiple paragraphs will create a strange multiple box experience and shouldn't be used with this style. Please use a soft return "Shift + Return" to create a break. Use "bold" styling to emulate a header.

Can you use images in this space? Yes, but... If an image is longer than the copy you have inside this box, the image will appear to "break out" of the box. Please ensure you have enough copy, or a short enough image. You will only get the option to adjust the styling of your image when you import it, so, ensure you select your alignment, sizing and position wisely.
We have a number of versions of this highlight box style that are all found in the styles menu. This one is called "Note" and has a yellow highlight band that appears on the right. This one is great for alerts or information you want to draw your developer to. When you hit a hard enter, you'll notice that you've created another box. Simply type in something, select that copy and click the remove styles button above.

Another one is called "Grey Highlight". You can change the alignment of the text (left, center, right) for all three of these styles. You can't restrict the width of this style to a portion of the page and align right. It is required that this style be full content width. Don't use in a table.

When adding your code snippet, make sure to select "Code Simple" from the styles dropdown to ensure for the best viewing experience.
document.getElementById("demo").style.fontSize = "25px"; document.getElementById('demo').style.fontSize = '25px';

Categories:

DPDK-on-the-Go – Profile DPDK Applications on Your Windows Laptop

Intel News - Mon, 04/17/2017 - 12:14

Introduction
This article gets you started with hands-on development, execution, and profiling of the Data Plane Development Kit (DPDK) application on your own laptop. This enhances portability as well as sharing and teaching developers, customers, and students in a scalable way.

About the Author

M Jay has worked with the DPDK team since 2009. He joined Intel in 1991 and has been in various roles and divisions: 64-bit CPU front side bus architect, 64 bit HAL developer, among others, before he joined the DPDK team. M Jay holds 21 US patents, both individually and jointly, all issued while working at Intel. M Jay was awarded the Intel Achievement Award in 2016, Intel's highest honor based on innovation and results.

Background
To run and profile DPDK on the Linux* platform, please refer to the article Profiling DPDK Code with Intel® VTune™ Amplifier. If you don't want to install Linux on your laptop, follow the steps in this article to learn how to configure your Intel® architecture-based Windows* laptop to develop, run and get started profiling DPDK applications.

Intel® VTune™ Amplifier, a performance profiler, will run natively on the Windows* OS so that it can access all the hardware performance registers. Developing and running DPDK applications will be done on an Oracle VM VirtualBox*.

The instructions in this article were tested on an Intel® Xeon® processor-based desktop, server, and laptop. Here we will use a laptop with the Windows OS.

If you have an Apple* laptop, the appendix provides information about systems based on the Mac OS*.

DPDK Application Building and Profiling – Components

- Intel® Atom™ or Intel Xeon processor-based system

- DPDK Applications
- Oracle VM VirtualBox
- Intel VTune Amplifier profiler

The platform can be any Intel® processor-based platform: desktop, server, laptop, or embedded system.

This article covers the following steps:

- Install and configure the Oracle VM VirtualBox.
- Import, build, and run DPDK applications.
- Install Intel VTune Amplifier and get started profiling.

Install and Configure the Oracle VM VirtualBox*

- Step 1: Make sure Intel® Virtualization Technology for Directed I/O (Intel® VT-d) Intel® Virtualization Technology for IA32, Intel® 64 and Intel® Architecture (Intel® VT-x) are enabled in the UEFI firmware/BIOS.
- Step 2: Download two images: the VirtualBox image and the extension packs for the same version.
- Step 3: Install VirtualBox.
- Step 4: Install the extension packs.
- Step 5: Verify that 64-bit guest virtualization is enabled.

Step 1: Make sure Intel VT-d and Intel VT-x are enabled in UEFI firmware/BIOS.
This is needed to ensure 64-bit guests can be run; VT-d and VT-x need to be on.

Intel VT-d and Intel VT-x enabling will be under Advanced CPU settings or Advanced Chipset settings as mentioned below. First we need to get into safe mode and look into BIOS setting.

1. Press the Windows button. You will see the following screen
2. Press the **Power Switch** icon. You will see the following drop-down menu.
3. To get into BIOS, press SHIFT+RESTART.

   If you have a laptop installed with Windows* 8, go to safe mode (SHIFT+RESTART).

   You will see the following settings. Note that depending on your computer, you may see different options.

4. To use advanced tools, choose **Troubleshoot.**

5. If the following screen displays, choose **Enable Safe Mode** to access the screen for the BIOS change.

   Once you have selected safe mode, you will be able to access additional options, as shown below.

   ;

6. Select **UEFI Firmware Settings**

   Note: In your system, it may be referred to as **BIOS setting**.

   Depending on your vendor and BIOS, you will be able to access the **Advanced** setting or **Advance Chipset Control** or **Advanced CPU Control**. What you need to do is verify whether Intel VT is enabled. In certain BIOS models, it may display as VT-d and VT-x.

   Some systems will have both a CPU section (for Intel VT-x) and a chipset section (for Intel VT-d) so you may have to look at both sections to enable virtualization.

   Below are two screens: the CPU screen followed by the chipset screen. In this system, only the chipset screen has virtualization control.

7. Save and then exit.

   Now the OS and applications come up.

   Step 2: Download two images: the VirtualBox Image and the extension packs for the same version
   To access the downloads, go to https://www.virtualbox.org/wiki/Downloads

For Windows:

1. Select VirtualBox5.1.8 (or the latest) for Windows hosts.
2. Matching version number Extension Packs.

For OS X*:

1. Select VirtualBox5.1.8 (or the latest) for OS X hosts.
2. Click and download the matching version extension.

Why install extension packs? What functionality do they provide?

Extension packs complement the functionality of VirtualBox.

1. Verify that both images downloaded in your system successfully.

Step 3. Install VirtualBox – Run As Administrator

1. To start the install, right-click VirtualBox, and then select **Run as administrator.**
2. Continue through the following screens.
3. For the screen above, press the left-arrow "<" to select INSTALL.

Step 4: Install the Extension Packs

1. Click **File**, and then click **Preferences.**
2. Click **Extensions**.
3. To select the extension packs download, browse, and then click the folder icon on the right, as shown by the arrow below.
4. Select the extension packs to install.

You will see the following success message:

Step 5: Verify that 64-bit guest virtualization is enabled.

1. Select New (shown by the arrow below) and choose OS Type "Linux".
2. Verify in the "Version" sub-menu whether 64-bit Ubuntu* is selectable. If so, the virtualization steps to enable Intel VT-d and Intel VT-x were successful.

Now you are ready to import the VMs.

Note: If you don't see 64-bit versions and see only 32-bit version, you'll need to enable Intel VT-d and Intel VT-x correctly. Return to the BIOS setting steps under "Step 1: "Make sure Intel VT-d and Intel VT-x are enabled in UEFI firmware/BIOS."

Import, Build, and Run DPDK Applications

In this article, we assume that you have plugged in a thumb drive with a copy of an exported DPDK application virtual machine that was built on a native Linux platform running DPDK. When you have connected the thumb drive, follow these instructions to import the VM.

1. Click **File**, and then click **Import Appliance**.
2. Click the folder on the right (as shown by the arrow below) to select the VM to import.
3. Select the VM.
   Example, as shown below: Ubuntu Nov 7 VTune DPDK.

The following screenshot shows verification of the virtual appliance getting imported.

4. Select Import.
   You will see the appliance being imported as shown below.

You have successfully imported the DPDK virtual appliance, as shown by the arrow in the screenshot below.

You have successfully launched DPDK running in the Ubuntu guest OS with VirtualBox on your laptop as shown below.

5. Select the Imported DPDK appliance.

6. To start the imported DPDK appliance, click **Start**.

You have successfully launched DPDK running in the Ubuntu guest OS with VirtualBox on your laptop as shown below

Now you can start your own development by developing applications, building, and running. To get started, locate the README_FIRST file, as shown in the above screenshot. Click open and you'll find instructions to run DPDK microbenchmarks and other applications.

Quick Profile View using Windows Task Manager
Let's say you want to know where cycles are being spent in the system. You can use Task Manager to have get a bird's-eye view first. Then you can dig into Intel VTune Amplifier.

The screenshot below shows the CPU cycles and tasks running, with Windows Task Manager showing CPU utilization running the DPDK application as a guest with VirtualBox.

Install Intel® VTune™ Amplifier

1. Go to https://software.intel.com/en-us/intel-VTune-amplifier-xe
2. Under Get Free Downloads & Trials, choose Windows*, and then click Download FREE Trial.
3. On the form, enter your email information to which the download link will be sent.
4. Check your email inbox for an email titled "Thank you for evaluating Intel® VTune™ Amplifier XE for Windows*. (Note: Search in your e-mail's trash folder in case you don't see it)
5. Click Download.

The **Activation Acknowledgement** page displays with the **Download Now** button, as shown below.

6. Click **Download Now**.
7. Print the "What's New?" document

Once the download has completed, as shown below you will have the VTune _Amplifier_XE_2017_update1_setup.exe image
8. Double-click the extracted setup image. You will see the following confirmation.
9. On the welcome page, click **Next**.
10. In the following screen, select **Evaluate this product (no serial number required).**
11. Click through the successive screen to complete the install.
12. On the following screen, leave "How do you want to open this file?" as the default and click **OK.**
13. Press the **Windows** button. You will see the installed Intel VTune Amplifier title as shown below.

The next step is to open a terminal as an administrator. It is important to access Intel VTune Amplifier as an administrator.

1. In the **Search Windows** box, type: cmd (shown below as the first step).
2. Right-click **Command Prompt** (shown below as the second step).
3. On the scroll-down menu, choose **Run as administrator.**
4. Verify that the terminal opened is titled Administrator.

The next step is to verify that the system is ready and the install was successful.

1. C:cd Program Files (x86)\IntelSWTools\VTune Amplifier XE 2017\bin32
2. To verify whether your system meets the needs of hardware-event-based sampling, type: amplxe-sepreg.exe –c
The following message screen should display.

The screen above indicates that you have successfully verified the correct dependency checks required to install the sampling driver:

- Platform, architecture, and OS environment
- Availability of the sampling driver binaries: sepdrv4_0.syssep3drv.sys and sepdal.sys
- Administrative privileges
- 32/64-bit installation

3. To check whether the sampling driver is loaded, type: amplxe-sepreg.exe –s
   The following message screen should display.

   The screen above indicates that the sampling driver loaded successfully.

   NOTE: If the sampling driver did NOT successfully load, refer to Appendix 3. Do NOT enter the command in Appendix 3 if you see the above success message.

   What's next?

   The default installation path for the Intel VTune™ Amplifier XE is
   [Program Files (x86)]\IntelSWTools\VTune™ Amplifier XE

4. cd \Progam Files (x86)\IntelSWTools\VTune™ Amplifier XE 2017
5. amplxe-vars.bat - run the batch file as shown below
   You have set the needed environment variables successfully. You will get output as shown below

   The final step is to run Intel VTune Amplifier.

6. **amplxe-gui** - run VTune; the GUI version is shown below

You will see the welcome screen as shown below.

Be sure to print the items circles below: Getting Started and Discover Performance Snapshots.

Start practicing by clicking **New Project** (also circled)

To get hands-on practice, please refer to the sections after "Starting Intel VTune Amplifier" in the following article: Profiling DPDK Code with Intel® VTune™ Amplifier

Also refer to the resources given in the reference section of the above article for videos and articles.

Next Steps
With the above hands-on exercise, you have successfully completed your "DPDK-On-The-Go" hands-on exercise.

As your first step, please register on the DPDK mailing list http://www.dpdk.org/ml/listinfo/dev

Also, we encourage you to play an active role in our meetups and DPDK community: www.dpdk.org

Please provide your feedback on this article to Muthurajan.Jayakumar@intel.com within 2 weeks after you go through your hands-on experience.

Exercises for the Readers

1. How do you virtualize network devices in your host to present to VirtualBox appliances?
2. How do you share your host file system with VirtualBox appliances?
3. When you configure for sharing, what happens when you export the Virtual Appliance?
4. What parameters in Intel VTune Amplifier do you look for in case you feel your application is compute bound?
5. Same question as above, with your application being a) memory bound, b) I/O bound?

Appendix 1: How to Enable Intel® Virtualization Technology in a Mac* Computer
This article's instructions were tested on a laptop with the Windows OS. Here are some references for the Mac regarding enabling Intel VT.

http://kb.parallels.com/en/5653

https://support.apple.com/en-us/HT203296

Appendix 2 – Potential Items to Watch Out for

- **Event-based sampling analysis**: To install the drivers on Windows 7 and Windows* Server* 2008 R2 operating systems, you must enable the SHA-2 code signing support for these systems by applying Microsoft Security update 3033929: https://technet.microsoft.com/en-us/library/security/3033929. If the security update is not installed, event-based sampling analysis types will not work properly on your system.

Appendix 3 – In case the sampling driver is not installed
If the sampling driver is not installed but the system is supported by Intel VTune Amplifier, execute the following command with the administrative privileges to install the driver:

amplxe-sepreg.exe –l

Appendix 4 – Intel VTune Amplifier for Mac Computers
While Intel VTune Amplifier 2017 can run on Windows and Linux systems, the profiled results can be seen on OS X.

So you can run the DPDK applications with VirtualBox in Mac computers. For profiling you can use native tools that come with OS X.

And you can use the Viewer given below to view the output Intel VTune Amplifier generated on Windows or Linux machines.

Please refer to the article How to Download and Evaluate the VTune™ Amplifier OS X* Viewer

References

DPDK-in-a-Box uses Minnowboard Turbot Single Board Computer.

Profiling DPDK Code with Intel® VTune™ Amplifier

Video: Intel® VTune™ and Performance Optimizations

DPDK Performance Optimization Guidelines White Paper

Categories:

DOE's INCITE Program Seeks Advanced Computational Research Proposals for 2018

HPC Wire - Mon, 04/17/2017 - 10:01

ARGONNE, Ill., April 17, 2017 — The Department of Energy's (DOE's) *Innovative and Novel Computational Impact on Theory and Experiment (INCITE)* program will be accepting proposals for high-impact, computationally intensive research campaigns in a broad array of science, engineering, and computer science domains. DOE's Office of Science plans to award over 6 billion supercomputer processor-hours at Argonne National Laboratory and at Oak Ridge National Laboratory.

From April 17 to June 23, INCITE's open call provides an opportunity for researchers to make transformational advances in science and technology through large allocations of computer time and supporting resources at the Leadership Computing Facility (LCF) centers located at Argonne and Oak Ridge national laboratories. ALCF and OLCF are DOE Office of Science User Facilities.

The winning proposals will receive large awards of time on two primary systems: Mira, a 10-petaflops IBM Blue Gene/Q system at Argonne, and Titan, a 27-petaflops Cray XK7 at Oak Ridge. In addition, certain 2018 INCITE awards will receive time on Argonne's new Intel/Cray system, a 9.65-petaflops system called Theta.

The INCITE program will host open instructional proposal writing webinars on April 19, May 18, and June 6, 2017. Staff from both LCFs will participate in all three sessions. In addition, the ALCF is hosting a Computational Performance Workshop, May 2-5, 2017, to train INCITE users and others on ways to boost their code performance on ALCF's manycore systems.

Proposals will be accepted until the call deadline of 8:00 p.m. EDT on Friday, June 23, 2017. Awards are expected to be announced in November 2017.

To submit an application or for additional details about the proposal requirements, visit the 2018 INCITE Call for Proposals webpage.

For more information on the INCITE program and a list of previous awards, visit the INCITE program website.

About the U.S. Department of Energy's Office of Science

The U.S. Department of Energy's Office of Science is the single largest supporter of basic research in the physical sciences in the United States and is working to address some of the most pressing challenges of our time. For more information, visit the Office of Science website.

Source: Argonne National Laboratory

The post DOE's INCITE Program Seeks Advanced Computational Research Proposals for 2018 appeared first on HPCwire.

Categories: RSS Feed Reader

Performance of Classic Matrix Multiplication Algorithm on Intel® Xeon Phi™ Processor System

Intel News - Fri, 04/14/2017 - 19:46

Contents

Introduction
Matrix multiplication (MM) of two matrices is one of the most fundamental operations in linear algebra. The algorithm for MM is very simple, it could be easily implemented in any programming language, and its performance significantly improves when different optimization techniques are applied.

Several versions of the classic matrix multiplication algorithm (CMMA) to compute a product of square dense matrices are evaluated in four test programs. Performance of these CMMAs is compared to a highly optimized 'cblas_sgemm' function of the Intel® Math Kernel Library (Intel® MKL)7. Tests are completed on a computer system with Intel® Xeon Phi™ processor 72105 running the Linux Red Hat* operating system in 'All2All' Cluster mode and for 'Flat', 'Hybrid 50-50', and 'Cache' MCDRAM modes.

All versions of CMMAs for single and double precision floating point data types described in the article are implemented in the C programming language and compiled with Intel® C++ Compiler versions 17 and 16 for Linux*6.

The article targets experienced C/C++ software engineers and can be considered as a reference on application optimization techniques, analysis of performance, and accuracy of computations related to MMAs.

If needed, the reader may review the contents of References1 or2 for a description of mathematical fundamentals of MM, because theoretical topics related to MM are not covered in this article.

An Overview of the Classic Matrix Multiplication Algorithm
A fundamental property of any algorithm is its asymptotic complexity (AC)3.

In generic form, AC for MMA can be expressed as follows:

MMA AC = O(N^Omega)

where O stands for operation on a data element, also known in computer science as a Big O; N is one dimension of the matrix, and omega is a matrix exponent which equals 3.0 for CMMA. That is:

CMMA AC = O(N^3)

In order to compute a product of two square matrices using CMMA, a cubic number of floating point (FP) multiplication operations is required. In other words, the CMMA runs in O(N^3) time.

An omega lower than 3.0 is possible, and it means that an MMA computes a product of two matrices faster because an optimization technique, mathematical or programming, is applied and fewer FP multiplication operations are required to compute the product.

A list of several MMAs with different values of omega is as follows:

Algorithm Omega Note Francois Le Gall 2.3728639 (1) Virginia Vassilevska Williams 2.3728642 Stothers 2.3740000 Coppersmith-Winograd 2.3760000 Bini 2.7790000 Pan 2.7950000 Strassen 2.8070000 (2) Strassen-Winograd 2.8070000 Classic 3.0000000 (3)
Table 1. Algorithms are sorted by omega in ascending order.

Total Number of Floating Point Operations
Let's assume that:

M x N is a dimension of a matrix A, or A[M,N]
N x P is a dimension of a matrix B, or B[N,P]
M x P is a dimension of a matrix C, or C[M,P]

There are three relations between M, N and P:

Relation #1: A[...,N] = B[N,...]
Relation #2: A[M,...] = C[M,...]
Relation #3: B[...,P] = C[...,P]

If one of these three relations is not met, the product of two matrices cannot be computed.

In this article only square matrices of dimension N, where M = N = P, will be considered. Therefore:

A[N,N] is the same as A[M,N]
B[N,N] is the same as B[N,P]
C[N,N] is the same as C[M,P]

The following table shows how many multiplications are needed to compute a product of two square matrices of different Ns for three algorithms from Table 1 with omega = 2.3728639 (1), omega = 2.807 (2) and omega = 3.0 (3).

N Omega = 2.3728639 (1) Omega = 2.807 (2) Omega = 3.0 (3) 128 100,028 822,126 2,097,152 256 518,114 5,753,466 16,777,216 512 2,683,668 40,264,358 134,217,728 1024 13,900,553 281,781,176 1,073,741,824 2048 72,000,465 1,971,983,042 8,589,934,592 4096 372,939,611 13,800,485,780 68,719,476,736 8192 1,931,709,091 96,579,637,673 549,755,813,888 16384 10,005,641,390 675,891,165,093 4,398,046,511,104 32768 51,826,053,965 4,730,074,351,662 35,184,372,088,832 65536 268,442,548,034 33,102,375,837,652 281,474,976,710,656
Table 2.

For example, to compute a product of two square dense matrices of dimension N equal to 32,768, Francois Le Gall (1) MMA needs ~51,826,053,965 multiplications and Classic (3) MMA needs ~35,184,372,088,832 multiplications.

Imagine the case of the product of two square matrices where *N equals 32,768* needs to be computed on a very slow computer system. It means that if the Francois Le Gall MMA completes the processing in one day, then the classic MMA will need ~679 days on the same computer system, or almost two years. This is because the Francois Le Gall MMA needs ~679x fewer multiplications to compute a product:

~35,184,372,088,832 / ~51,826,053,965 = ~678.9

In the case of using a famous Strassen (2) MMA, ~91 days would be needed:

~4,730,074,351,662 / ~51,826,053,965 = ~91.3

In many software benchmarks the performance of an algorithm, or some processing, is measured in floating point operations per second (FLOPS), and not in elapsed time intervals, like days, hours, minutes, or seconds. That is why it is very important to know an exact total number (TN) of FP operations completed to calculate a FLOPS value.

With modern C++ compilers, it is very difficult to estimate an exact TN of FP operations per unit of time completed at run time due to extensive optimizations of generated binary codes. It means that an analysis of binary codes could be required, and this is outside of the scope of this article.

However, an estimate value of the TN of FP operations, multiplications and additions, for CMMA when square matrices are used can be easily calculated. Here are two simple examples:

Example 1: N = 2

Multiplications = 8 // 2 * 2 * 2 = 2^3 Additions = 4 // 2 * 2 * 1 = 2^2*(2-1) TN FP Ops = 8 + 4 = 12
Example 2: N = 3

Multiplications = 27 // 3 * 3 * 3 = 3^3 Additions = 18 // 3 * 3 * 2 = 3^2*(3-1) TN FP Ops = 27 + 18 = 45
It is apparent that the TN of FP operations to compute a product of two square matrices can be calculated using a simple formula:

TN FP Ops = (N^3) + ((N^2) * (N-1))

Note: Take into account that in the versions of the MMA used for sparse matrices, no FP operations are performed if the matrix element at position (i,j) is equal to zero.

Implementation Complexity
In the C programming language only four lines of code are needed to implement a core part of the CMMA:

for(i = 0; i < N; i += 1) for(j = 0; j < N; j += 1) for(k = 0; k < N; k += 1) C[i][j] += A[i][k] * B[k][j];
Therefore, CMMA's implementation complexity (IC) could be rated as very simple.

Declarations of all intermediate variables, memory allocations, and initialization of matrices are usually not taken into account.

More complex versions of MMA, like Strassen or Strassen-Winograd, could have several thousands of code lines.

Optimization Techniques
In computer programming, matrices could be represented in memory as 1-D or 2-D data structures.

Here is a static declaration of matrices A, B, and C as 1-D data structures of a single precision (SP) FP data type (float):

float fA[N*N]; float fB[N*N]; float fC[N*N];
and this is what a core part of the CMMA looks like:

for(i = 0; i < N; i += 1) for(j = 0; j < N; j += 1) for(k = 0; k < N; k += 1) C[N*i+j] += A[N*i+k] * B[N*k+j];
Here is a static declaration of matrices A, B, and C as 2-D data structures of a single precision (SP) FP data type (float):

float fA[N][N]; float fB[N][N]; float fC[N][N];
and this is what the core part of CMMA looks like:

for(i = 0; i < N; i += 1) for(j = 0; j < N; j += 1) for(k = 0; k < N; k += 1) C[i][j] += A[i][k] * B[k][j];
Many other variants of the core part of CMMA are possible and they will be reviewed.

Memory Allocation Schemes
In the previous section of this article, two examples of a static declaration of matrices A, B, and C were given. In the case of dynamic allocation of memory for matrices, explicit calls to memory allocation functions need to be made. In this case, declarations and allocations of memory can look like the following:

Declaration of matrices A, B, and C as 1-D data structures:

__attribute__((aligned(64))) float *fA; __attribute__((aligned(64))) float *fB; __attribute__((aligned(64))) float *fC;
and this is how memory needs to be allocated:

fA = (float *)_mm_malloc(N * sizeof(float), 64); fB = (float *)_mm_malloc(N * sizeof(float), 64); fC = (float *)_mm_malloc(N * sizeof(float), 64);
Note: Allocated memory blocks are 64-byte aligned, contiguous, and not fragmented by an operating system memory manager; this improves performance of processing.

Declaration of matrices A, B, and C as 2-D data structures:

__attribute__((aligned(64))) float **fA; __attribute__((aligned(64))) float **fB; __attribute__((aligned(64))) float **fC;
and this is how memory needs to be allocated:

fA = (float **)calloc(N, sizeof(float *)); fB = (float **)calloc(N, sizeof(float *)); fC = (float **)calloc(N, sizeof(float *)); for(i = 0; i < N; i += 1) { fA[i] = (float *)calloc(N, sizeof(float)); fB[i] = (float *)calloc(N, sizeof(float)); fC[i] = (float *)calloc(N, sizeof(float)); }
Note: Allocated memory blocks are not contiguous and can be fragmented by an operating system memory manager, and fragmentation can degrade performance of processing.

In the previous examples, a DDR4-type RAM memory was allocated for matrices. However, on an Intel Xeon Phi processor system5 a multichannel DRAM (MCDRAM)-type RAM memory could be allocated as well, using functions from a memkind library11 when MCDRAM mode is configured to 'Flat' or 'Hybrid'. For example, this is how an MCDRAM-type RAM memory can be allocated:

fA = (float *)hbw_malloc(N * sizeof(float)); fB = (float *)hbw_malloc(N * sizeof(float)); fC = (float *)hbw_malloc(N * sizeof(float));
Note: An 'hbw_malloc' function of the memkind library was used instead of an '_mm_malloc' function.

On an Intel Xeon Phi processor system, eight variants of memory allocation for matrices A, B, and C are possible:

Matrix A Matrix B Matrix C Note DDR4 DDR4 DDR4 (1) DDR4 DDR4 MCDRAM (2) DDR4 MCDRAM DDR4 DDR4 MCDRAM MCDRAM MCDRAM DDR4 DDR4 MCDRAM DDR4 MCDRAM MCDRAM MCDRAM DDR4 MCDRAM MCDRAM MCDRAM
Table 3.

It is recommended to use MCDRAM memory as much as possible because its bandwidth is ~400 GB/s, and it is ~5 times faster than the ~80 GB/s bandwidth of DDR4 memory5.

Here is an example of how 'cblas_sgemm' MMA performs for two memory allocation schemes (MASs) (1) and (2):

Matrix multiplication C=A*B where matrix A (32768x32768) and matrix B (32768x32768) Allocating memory for matrices A, B, C: MAS=DDR4:DDR4:DDR4 Initializing matrix data Matrix multiplication started Matrix multiplication completed at 50.918 seconds Allocating memory for matrices A, B, C: MAS=DDR4:DDR4:MCDRAM Initializing matrix data Matrix multiplication started Matrix multiplication completed at 47.385 seconds
It is clear that there is a performance improvement of ~7 percent when an MCDRAM memory was allocated for matrix C.

Loop Processing Schemes
A loop processing scheme (LPS) describes what optimization techniques are applied to the 'for' statements of the C language of the core part of CMMA. For example, the following code:

for(i = 0; i < N; i += 1) // loop 1 for(j = 0; j < N; j += 1) // loop 2 for(k = 0; k < N; k += 1) // loop 3 C[i][j] += A[i][k] * B[k][j];
corresponds to an LPS=1:1:1, and it means that loop counters are incremented by 1.

Table 4 below includes short descriptions of different LPSs:

LPS Note 1:1:1 Loops not unrolled 1:1:2 3rd loop unrolls to 2-in-1 computations 1:1:4 3rd loop unrolls to 4-in-1 computations 1:1:8 3rd loop unrolls to 8-in-1 computations 1:2:1 2nd loop unrolls to 2-in-1 computations 1:4:1 2nd loop unrolls to 4-in-1 computations 1:8:1 2nd loop unrolls to 8-in-1 computations
Table 4.

For example, the following code corresponds to an LPS=1:1:2, and it means that counters 'i' and 'j' for loops 1 and 2 are incremented by 1, and counter 'k' for loop 3 is incremented by 2:

for(i = 0; i < N; i += 1) // :1 { for(j = 0; j < N; j += 1) // :1 { for(k = 0; k < N; k += 2) // :2 (unrolled loop) { C[i][j] += A[i][k] * B[k][j]; C[i][j] += A[i][k+1] * B[k+1][j]; } } }
Note: A C++ compiler could unroll loops as well if command line-options for unrolling are used. A software engineer should prevent such cases when source code unrolling is used at the same time, because it prevents vectorization of inner loops, and degrades performance of processing.

Another optimization technique is the loop interchange optimization technique (LIOT). When the LIOT is used, a core part of CMMA looks as follows:

for(i = 0; i < N; i += 1) // loop 1 for(k = 0; k < N; k += 1) // loop 2 for(j = 0; j < N; j += 1) // loop 3 C[i][j] += A[i][k] * B[k][j];
It is worth noting that counters 'j' and 'k' for loops 2 and 3 were exchanged.

The loops unrolling and LIOT allow for improved performance of processing because elements of matrices A and B are accessed more efficiently.

Compute Schemes
A compute scheme (CS) describes the computation of final or intermediate values and how elements of matrices are accessed.

In a CMMA an element (i,j) of the matrix C is calculated as follows:

C[i][j] += A[i][k] * B[k][j]
and its CS is ij:ik:kj.

However, elements of matrix B are accessed in a very inefficient way. That is, the next element of matrix B, which needs to be used in the calculation, is located at a distance of (N * sizeof (datatype)) bytes. For very small matrices this is not critical because they can fit into CPU caches. However, for larger matrices it affects performance of computations, which can be significantly degraded, due to cache misses.

In order to solve that problem and improve performance of computations, a very simple optimization technique is used. If matrix B is transposed, the next element that needs to be used in the calculation will be located at a distance of (sizeof (datatype)) bytes. Thus, access to the elements of matrix B will be similar to the access to the elements of matrix A.

In a transpose-based CMMA, an element (i,j) of the matrix C is calculated as follows:

C[i][j] += A[i][k] * B[j][k]
and its CS is ij:ik:jk. Here B[j][k] is used instead of B[k][j].

It is very important to use the fastest possible algorithm for the matrix B transposition before processing is started. In Appendix B an example is given on how much time is needed to transpose a square matrix of 32,768 elements, and how much time is needed to compute the product on an Intel Xeon Phi processor system.

Another optimization technique is the loop blocking optimization technique (LBOT) and it allows the use of smaller subsets of A, B, and C matrices to compute the product. When the LBOT is used, a core part of CMMA looks as follows:

for(i = 0; i < N; i += BlockSize) { for(j = 0; j < N; j += BlockSize) { for(k = 0; k < N; k += BlockSize) { for(ii = i; ii < (i+BlockSize); ii += 1) for(jj = j; jj < (j+BlockSize); jj += 1) for(kk = k; kk < (k+BlockSize); kk += 1) C[ii][jj] += A[ii][kk] * B[kk][jj]; } } }
Note: A detailed description of LBOT can be found at10.

Table 5 shows four examples of CSs:

CS Note ij:ik:kj Default ij:ik:jk Transposed iijj:iikk:kkjj Default LBOT iijj:iikk:jjkk Transposed LBOT

Table 5.

Error Analysis
In any version of MMA many FP operations need to be done in order to compute values of elements of matrix C. Since FP data types SP or DP have limited precision4, rounding errors accumulate very quickly. A common misconception is that rounding errors can occur only in cases where large or very large matrices need to be multiplied. This is not true because, in the case of floating point arithmetic (FPA), a rounding error is also a function of the range of an input value, and it is not a function of the size of input matrices.

However, a very simple optimization technique allows improvement in the accuracy of computations.

If matrices A and B are declared as an SP FP data type, then intermediate values could be stored in a variable of DP FP data type:

for(i = 0; i < N; i += 1) { for(j = 0; j < N; j += 1) { double sum = 0.0; for(k = 0; k < N; k += 1) { sum += (double)(A[i][k] * B[k][j]); } C[i][j] = sum; } }
The accuracy of computations will be improved, but performance of processing can be lower.

An error analysis (EA) is completed using the mmatest4.c test program for different sizes of matrices of SP and DP FP data types (see Table 6 in Appendix C with results).

Performance on the Intel® Xeon Phi™ Processor System
Several versions of the CMMA to compute a product of square dense matrices are evaluated in four test programs. Performance of these CMMAs is compared to a highly optimized 'cblas_sgemm' function of the Intel MKL7. Also see Appendix D for more evaluations.

Figure 1. Performance tests for matrix multiply algorithms on Intel® Xeon Phi™ processor using mmatest1.c with KMP_AFFINITY environment variable set to 'scatter', 'balanced', and 'compact'. A lower bar height means faster processing.

Here are the names of source files with a short description of tests:

mmatest1.c - Performance tests matrix multiply algorithms on an Intel Xeon Phi processor.
mmatest2.c - Performance tests matrix multiply algorithms on an Intel Xeon Phi processor in one MCDRAM mode ('Flat') for DDR4:DDR4:DDR4 and DDR4:DDR4:MCDRAM MASs.
mmatest3.c - Performance tests matrix multiply algorithms on an Intel Xeon Phi processor in three MCDRAM modes ('All2All', 'Flat', and 'Cache') for DDR4:DDR4:DDR4 and MCDRAM:MCDRAM:MCDRAM MASs. Note: In 'Cache' MCDRAM mode, MCDRAM:MCDRAM:MCDRAM MAS cannot be used.
mmatest4.c - Verification matrix multiply algorithms accuracy of computations on an Intel Xeon Phi processor.

OpenMP* Product Thread Affinity Control
OpenMP* product compiler directives can be easily used to parallelize processing and significantly speed up processing. However, it is very important to execute OpenMP threads on different logical CPUs of modern multicore processors in order to utilize their internal resources as best as possible.

In the case of using the Intel C++ compiler and Intel OpenMP run-time libraries, the KMP_AFFINITY environment variable provides flexibility and simplifies that task. Here are three simple examples of using the KMP_AFFINITY environment variable:

KMP_AFFINITY = scatter KMP_AFFINITY = balanced KMP_AFFINITY = compact
These two screenshots of the Htop* utility12 demonstrate how OpenMP threads are assigned (pinned) to Intel Xeon Phi processor 72105 logical CPUs during processing of an MMA using 64 cores of the processor:

Screenshot 1. KMP_AFFINITY = scatter or balanced. Note: Processing is faster when compared to KMP_AFFINITY = compact.

Screenshot 2. KMP_AFFINITY = compact. Note: Processing is slower when compared to KMP_AFFINITY = scatter or balanced.

Recommended Intel® C++ Compiler Command-Line Options
Here is a list of Intel C++ Compiler command-line options that a software engineer should consider, which can improve performance of processing of CMMAs:

O3
fp-model
parallel
unroll
unroll-aggressive
opt-streaming-stores
opt-mem-layout-trans

Os
openmp

ansi-alias
fma
opt-matmul
opt-block-factor
opt-prefetch

The reader can use 'icpc -help' or 'icc -help' to learn more about these command-line options.

Conclusion
Application of different optimization techniques to the CMMA were reviewed in this article.

Three versions of CMMA to compute a product of square dense matrices were evaluated in four test programs. Performance of these CMMAs was compared to a highly optimized 'cblas_sgemm' function of the Intel MKL7.

Tests were completed on a computer system with an Intel® Xeon Phi processor 72105 running the Linux Red Hat operating system in 'All2All' Cluster mode and for 'Flat', 'Hybrid 50-50', and 'Cache' MCDRAM modes.

It was demonstrated that CMMA could be used for cases when matrices of small sizes, up to 1,024 x 1,024, need to be multiplied.

It was demonstrated that performance of MMAs is higher when MCDRAM-type RAM memory is allocated for matrices with sizes up to 16,384 x 16,384 instead of DDR4-type RAM memory.

Advantages of using CMMA to compute the product of two matrices are as follows:

- In any programming language, simple to implement to run on CPUs or GPUs9
- Highly portable source codes when implemented in C, C++, or Java programming languages
- Simple to integrate with existing software for a wide range of computer platforms
- Simple to debug and troubleshoot
- Predictable memory footprint at run time
- Easy to optimize using parallelization and vectorization techniques
- Low overheads and very good performance for matrices of sizes ranging from 256 x 256 to 1,024 x 1,024 (see Figures 1 through 5)
- Very good accuracy of computations for matrices of sizes ranging from 8 x 8 to 2,048 x 2,048 (see Table 6 in Appendix C)

Disadvantages of using CMMA to compute a product of two matrices are as follows:

- Poor performance for large matrices with sizes greater than 2048 x 2048
- Poor performance when implemented using high-level programming languages due to processing overheads
- Reduced accuracy of computations for matrices of sizes ranging from 2,048 x 2,048 to 65,536 x 65,536 (see Table 6 in Appendix C)

References

1. Matrix Multiplication on Mathworld

http://mathworld.wolfram.com/MatrixMultiplication.html

2. Matrix Multiplication on Wikipedia

https://en.wikipedia.org/wiki/Matrix_multiplication

3. Asymptotic Complexity of an Algorithm

https://en.wikipedia.org/wiki/Time_complexity

4. The IEEE 754 Standard for Floating Point Arithmetic

http://standards.ieee.org/

5. Intel® Many Integrated Core Architecture

https://software.intel.com/en-us/xeon-phi/x200-processor
http://ark.intel.com/products/94033/Intel-Xeon-Phi-Processor-7210-16GB-1_30-GHz-64-core
https://software.intel.com/en-us/forums/intel-many-integrated-core

6. Intel® C++ Compiler

https://software.intel.com/en-us/c-compilers
https://software.intel.com/en-us/forums/intel-c-compiler

7. Intel® MKL

https://software.intel.com/en-us/intel-mkl
https://software.intel.com/en-us/intel-mkl/benchmarks
https://software.intel.com/en-us/forums/intel-math-kernel-library

8. Intel® Developer Zone Forums

https://software.intel.com/en-us/forum

9. Optimizing Matrix Multiply for Intel® Processor Graphics Architecture Gen 9

https://software.intel.com/en-us/articles/sgemm-ocl-opt

10. Performance Tools for Software Developers Loop Blocking

https://software.intel.com/en-us/articles/performance-tools-for-software-developers-loop-blocking

11. Memkind library

https://github.com/memkind/memkind

12. Htop* monitoring utility

https://sourceforge.net/projects/htop

Downloads
Performance_CMMA_system.zip

List of all files (sources, test reports, and so on):

Performance_CMMA_system.pdf - Copy of this paper.

mmatest1.c - Performance tests for matrix multiply algorithms on Intel® Xeon Phi processors.

dataset1.txt - Results of tests.

mmatest2.c - Performance tests for matrix multiply algorithms on Intel® Xeon Phi processors for DDR4:DDR4:DDR4 and DDR4:DDR4:MCDRAM MASs.

dataset2.txt - Results of tests.

mmatest3.c - Performance tests for matrix multiply algorithms on Intel® Xeon Phi processors in three MCDRAM modes for DDR4:DDR4:DDR4 and MCDRAM:MCDRAM:MCDRA MASs.

dataset3.txt - Results of tests.

mmatest4.c - Verification of matrix multiply algorithms accuracy of computations on Intel® Xeon Phi processors.

dataset4.txt - Results of tests.

Note: Intel C++ Compiler versions used to compile tests:
17.0.1 Update 132 for Linux*
16.0.3 Update 210 for Linux*

Abbreviations
CPU - Central processing unit
GPU - Graphics processing unit
ISA - Instruction set architecture
MIC - Intel® Many Integrated Core Architecture
RAM - Random access memory
DRAM - Dynamic random access memory
MCDRAM - Multichannel DRAM
HBW - High bandwidth memory
DDR4 - Double data rate (generation) 4
SIMD - Single instruction multiple data
SSE - Streaming SIMD extensions
AVX - Advanced vector extensions
FP - Floating point

FPA - Floating point arithmetic4
SP - Single precision4
DP - Double precision4
FLOPS - Floating point operations per second
MM - Matrix multiplication
MMA - Matrix multiplication algorithm
CMMA - Classic matrix multiplication algorithm
MTA - Matrix transpose algorithm
AC - Asymptotic complexity
IC - Implementation complexity
EA - Error analysis
MAS - Memory allocation scheme
LPS - Loop processing scheme
CS - Compute scheme
LIOT - Loop interchange optimization technique
LBOT - Loop blocking optimization technique
ICC - Intel C++ Compiler6
MKL - Math kernel library7
CBLAS - C basic linear algebra subprograms
IDZ - Intel® Developer Zone8
IEEE - Institute of Electrical and Electronics Engineers4
GB - Gigabytes
TN - Total number

Appendix A - Technical Specifications of the Intel® Xeon Phi™ Processor System
Summary of the Intel Xeon Phi processor system used for testing:

Process technology: 14nm
Processor name: Intel Xeon Phi processor 7210
Frequency: 1.30 GHz
Packages (sockets): 1
Cores: 64
Processors (CPUs): 256
Cores per package: 64
Threads per core: 4
On-Package Memory: 16 GB high bandwidth MCDRAM (bandwidth ~400 GB/s)
DDR4 Memory: 96 GB 6 Channel (Bandwidth ~ 80 GB/s)
ISA: Intel® AVX-512 (Vector length 512-bit)

Detailed processor specifications:

http://ark.intel.com/products/94033/Intel-Xeon-Phi-Processor-7210-16GB-1_30-GHz-64-core

Summary of a Linux operating system:

[guest@... ~]$ uname -a

Linux c002-n002 3.10.0-327.13.1.el7.xppsl_1.4.0.3211.x86_64 #1 SMP
Fri Jul 8 11:44:24 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux

[guest@... ~]$ cat /proc/version

Linux version 3.10.0-327.13.1.el7.xppsl_1.4.0.3211.x86_64 (qb_user@89829b4f89a5)
(gcc version 4.8.5 20150623 (Red Hat 4.8.5-4) (GCC)) #1 SMP Fri Jul 8 11:44:24 UTC 2016

Appendix B - Comparison of Processing Times for MMAs versus MTA
Comparison of processing times for Intel MKL 'cblas_sgemm' and CMMA vs. MTA:

[Intel MKL & CMMA]

Matrix A [32768 x 32768] Matrix B [32768 x 32768]
Number of OpenMP threads: 64
MKL - Completed in: 51.2515874 seconds
CMMA - Completed in: 866.5838490 seconds

[MTA]

Matrix size: 32768 x 32768
Transpose Classic - Completed in: 1.730 secs
Transpose Diagonal - Completed in: 1.080 secs
Transpose Eklundh - Completed in: 0.910 secs

When compared processing time of MTA to:
MKL 'cblas_sgemm'. the transposition takes ~2.42 percent of the processing time.
CMMA, the transposition takes ~0.14 percent of the processing time.

Appendix C - Error Analysis (Absolute Errors for SP FP Data Type) N MMA Calculated SP Value Absolute Error 8 MKL 8.000080 0.000000 8 CMMA 8.000080 0.000000 16 MKL 16.000160 0.000000 32 CMMA 16.000160 0.000000 32 MKL 32.000309 -0.000011 32 CMMA 32.000320 0.000000 64 MKL 64.000671 0.000031 128 CMMA 64.000641 0.000001 128 MKL 128.001160 -0.000120 128 CMMA 128.001282 0.000002 256 MKL 256.002319 -0.000241 512 CMMA 256.002563 0.000003 512 MKL 512.004639 -0.000481 512 CMMA 512.005005 -0.000115 1024 MKL 1024.009521 -0.000719 2048 CMMA 1024.009888 -0.000352 2048 MKL 2048.019043 -0.001437 2048 CMMA 2048.021484 0.001004 4096 MKL 4096.038574 -0.002386 8192 CMMA 4096.037109 -0.003851 8192 MKL 8192.074219 -0.007701 8192 CMMA 8192.099609 0.017689 16384 MKL 16384.14648 -0.017356 32768 CMMA 16384.09961 -0.064231 32768 MKL 32768.33594 0.008258 32768 CMMA 32768.10156 -0.226118 65536 MKL 65536.71875 0.063390 65536 CMMA 65536.10156 -0.553798
Table 6.

Appendix D - Performance of MMAs for Different MASs

Figure 2. Performance of Intel® MKL 'cbals_sgemm'. KMP_AFFINITY environment variable set to 'scatter'. Cluster mode: 'All2All'. MCDRAM mode: 'Flat'. Test program mmatest2.c. A lower bar height means faster processing.

Figure 3. Performance of Intel® MKL 'cbals_sgemm' vs. CMMA. KMP_AFFINITY environment variable set to 'scatter'. Cluster mode: 'All2All'. MCDRAM mode: 'Flat'. Test program mmatest3.c. A lower bar height means faster processing.

Figure 4. Performance of Intel® MKL 'cbals_sgemm' vs. CMMA. KMP_AFFINITY environment variable set to 'scatter'. Cluster mode: 'All2All'. MCDRAM mode: 'Hybrid 50-50'. Test program mmatest3.c. A lower bar height means faster processing.

Figure 5. Performance of Intel® MKL 'cbals_sgemm' vs. CMMA. KMP_AFFINITY environment variable set to 'scatter'. Cluster mode: 'All2All'. MCDRAM mode: 'Cache'. Test program mmatest3.c. A lower bar height means faster processing.

About the Author
Sergey Kostrov is a highly experienced C/C++ software engineer and Intel® Black Belt Developer. He is an expert in design and implementation of highly portable C/C++ software for embedded and desktop platforms, scientific algorithms, and high performance computing of big data sets.

Categories:

Intel® XDK FAQs - Crosswalk [RETIRED]

Intel News - Fri, 04/14/2017 - 17:29

[RETIRED] The Crosswalk Project has been retired!
IMPORTANT: on February, 2017, the *Crosswalk Project* was retired. Crosswalk 23 was the last version of the Crosswalk library produced by the Crosswalk team. You can continue to build with the Crosswalk library using Cordova CLI or PhoneGap Build, but no further updates to the Crosswalk library will occur.

No bug fixes will be implemented for Crosswalk components.

You can continue to use Crosswalk in your project, but there will be no new releases of the Crosswalk library and the Intel XDK will not add any new versions of Crosswalk to the build settings. If you are deploying your app to Android 5 or greater there is no reason to use the Crosswalk library, since those versions of Android include an upgradeable native Chromium webview that is up-to-date and is as capable and as performant as the Crosswalk webview. If you are still deploying to Android 4.x devices you may want to continue to use Crosswalk for those devices. Unlike the native webview in Android 5+ devices, the native webview in Android 4.x devices cannot be upgraded and is quite limited.

- How do I play audio with different playback rates?
- Why are Intel XDK Android Crosswalk build files so large?
- Why is the size of my installed app much larger than the apk for a Crosswalk application?
- Why does my Android Crosswalk build fail with the com.google.playservices plugin?
- Why does my app fail to run on some devices?
- How do I stop "pull to refresh" from resetting and restarting my Crosswalk app?
- Which versions of Crosswalk are supported and why do you not support version X, Y or Z?
- How do I prevent my Crosswalk app from auto-completing passwords?
- How can I improve the performance of my Construct2 game built with Crosswalk?
- Why does the Google store refuse to publish my Crosswalk app?
- Why is my Crosswalk app generating **errno 12 Out of memory** errors on some devices?

- Construct2 Tutorial: How to use AdMob and IAP plugins with Crosswalk and the Intel XDK.
- What is the correct "Target Android API" value that I should use when building for Crosswalk on Android?
- Can I build my app with a version of Crosswalk that is not listed in the Intel XDK Build Settings UI?
- My Construct2 Crosswalk app flashes a white box or white band after the splash screen.

How do I play audio with different playback rates?
Here is a code snippet that allows you to specify playback rate:

var myAudio = new Audio('/path/to/audio.mp3'); myAudio.play(); myAudio.playbackRate = 1.5; Why are Intel XDK Android Crosswalk build files so large?
When your app is built with Crosswalk it will be a minimum of 15-18MB in size because it includes a complete web browser (the Crosswalk runtime or webview) for rendering your app instead of the built-in webview on the device. Despite the additional size, this is the preferred solution for Android, because the built-in webviews on the majority of Android devices are inconsistent and poorly performing.

See these articles for more information:

- *Build High-Performance HTML5 Cordova Apps with Crosswalk*
- *Why Use Crosswalk?*

Why is the size of my installed app much larger than the apk for a Crosswalk application?
This is because the apk is a compressed image, so when installed it occupies more space due to being decompressed. Also, when your Crosswalk app starts running on your device it will create some data files for caching purposes which will increase the installed size of the application.

Why does my Android Crosswalk build fail with the com.google.playservices plugin?
The Intel XDK Crosswalk build system used with CLI 4.1.2 Crosswalk builds does not support the library project format that was introduced in the "com.google.playservices@21.0.0" plugin. Use "com.google.playservices@19.0.0" instead.

Why does my app fail to run on some devices?
There are some Android devices in which the GPU hardware/software subsystem does not work properly. This is typically due to poor design or improper validation by the manufacturer of that Android device. Your problem Android device probably falls under this category.

How do I stop "pull to refresh" from resetting and restarting my Crosswalk app?
See the code posted in this forum thread for a solution: /en-us/forums/topic/557191#comment-1827376.

An alternate solution is to add the following lines to your intelxdk.config.additions.xml file:

<!-- disable reset on vertical swipe down --> <intelxdk:crosswalk xwalk-command-line="--disable-pull-to-refresh-effect" /> Which versions of Crosswalk are supported and why do you not support version X, Y or Z?
The specific versions of Crosswalk that are offered via the Intel XDK are based on what the Crosswalk project releases and the timing of those releases relative to Intel XDK build system updates. This is one of the reasons you do not see *every* version of Crosswalk supported by our Android-Crosswalk build system.

With the September, 2015 release of the Intel XDK, the method used to build embedded Android-Crosswalk versions changed to the "pluggable" webview Cordova build system. This new build system was implemented with the help of the Cordova project and became available with their release of the Android Cordova 4.0 framework (coincident with their Cordova CLI 5 release). With this change to the Android Cordova framework and the Cordova CLI build system, we can now more quickly adapt to new version releases of the Crosswalk project. Support for previous Crosswalk releases required updating a special build system that was forked from the Cordova Android project. This new "pluggable" webview build system means that the build system can now use the standard Cordova build system, because it now includes the Crosswalk library as a "pluggable" component.

The "old" method of building Android-Crosswalk APKs relied on a "forked" version of the Cordova Android framework, and is based on the Cordova Android 3.6.3 framework and is used when you select CLI 4.1.2 in the Project tab's build settings page. Only Crosswalk versions 7, 10, 11, 12 and 14 are supported by the Intel XDK when using this build setting.

Selecting CLI 5.1.1 in the build settings will generate a "pluggable" webview built app. A "pluggable" webview app (built with CLI 5.1.1) results in an app built with the Cordova Android 4.1.0 framework. As of the latest update to this FAQ, the CLI 5.1.1 build system supported Crosswalk 15. Future releases of the Intel XDK and the build system will support higher versions of Crosswalk and the Cordova Android framework.

In both cases, above, the net result (when performing an "embedded" build) will be two processor architecture-specific APKs: one for use on an x86 device and one for use on an ARM device. The version codes of those APKs are modified to insure that both can be uploaded to the Android store under the same app name, insuring that the appropriate APK is automatically delivered to the matching device (i.e., the x86 APK is delivered to Intel-based Android devices and the ARM APK is delivered to ARM-based Android devices).

For more information regarding Crosswalk and the Intel XDK, please review these documents:

- *Intel XDK Crosswalk Chapter*

- *Using the Crosswalk for Android Build Option*
- *Choosing Crosswalk* Build Options: Shared or Embedded*

How do I prevent my Crosswalk app from auto-completing passwords?
Use the Ionic Keyboard plugin and set the spellcheck attribute to false.

How can I improve the performance of my Construct2 game build with Crosswalk?
Beginning with the Intel XDK CLI 5.1.1 build system you *must* add the --ignore-gpu-blacklist option to your intelxdk.config.additions.xml file if you want the additional performance this option provides to blacklisted devices. See this forum post for additional details.

If you are a Construct2 game developer, please read this blog by another Construct2 game developer regarding how to properly configure your game for proper Crosswalk performance > *How to build optimized Intel XDK Crosswalk app properly?* <

Also, you can experiment with the **CrosswalkAnimatable** option in your intelxdk.config.additions.xml file (details regarding the **CrosswalkAnimatable** option are available in this Crosswalk Project wiki post: *Android SurfaceView vs TextureView*).

<!-- Controls configuration of Crosswalk-Android "SurfaceView" or "TextureView" --> <!-- Default is SurfaceView if >= CW15 and TextureView if <= CW14 --> <!-- Option can only be used with Intel XDK CLI5+ build systems --> <!-- SurfaceView is preferred, TextureView should only be used in special cases --> <!-- Enable Crosswalk-Android TextureView by setting this option to true --> <preference name="CrosswalkAnimatable" value="false" />
See *Chromium Command-Line Options for Crosswalk Builds with the Intel XDK* for some additional tools that can be used to modify the Crosswalk's webview runtime parameters, especially the --ignore-gpu-blacklist option.

Why does the Google store refuse to publish my Crosswalk app?
For full details, please read *Android and Crosswalk Cordova Version Code Issues*. For a summary, read this FAQ.

There is a change to the version code handling by the Crosswalk and Android build systems based on Cordova CLI 5.0 and later. This change was implemented by the Apache Cordova project. This new version of Cordova CLI automatically modifies the **android:versionCode** when building for Crosswalk and Android. Because our CLI 5.1.1 build system is now more compatible with standard Cordova CLI, this change results in a discrepancy in the way your **android:versionCode** is handled when building for Crosswalk (15) or Android with CLI 5.1.1 when compared to building with CLI 4.1.2.

If you have never published an app to an Android store this change will have little or no impact on you. This change might affect attempts to *side-load* an app onto a device, in which case the simplest solution is to uninstall the previously side-loaded app before installing the new app.

Here's what Cordova CLI 5.1.1 (Cordova-Android 4.x) is doing with the **android:versionCode** number (which you specify in the *App Version Code* field within the **Build Settings** section of the **Projects** tab):

Cordova-Android 4.x (Intel XDK CLI 5.1.1 for Crosswalk or Android builds) does this:

- multiplies your **android:versionCode** by 10

then, if you are doing a Crosswalk (15) build:

- adds 2 to the **android:versionCode** for ARM builds
- adds 4 to the **android:versionCode** for x86 builds

otherwise, if you are performing a standard Android build (non-Crosswalk):

- adds 0 to the **android:versionCode** if the *Minimum Android API* is < 14
- adds 8 to the **android:versionCode** if the *Minimum Android API* is 14-19
- adds 9 to the **android:versionCode** if the *Minimum Android API* is > 19 (i.e., >= 20)

*If you **HAVE PUBLISHED** a Crosswalk app to an Android store this change may impact your ability to publish a newer version of your app!* In that case, *if you are building for Crosswalk*, add 6000 (six with three zeroes) to your existing *App Version Code* field in the Crosswalk **Build Settings** section of the **Projects** tab. If you have only published standard Android apps in the past and are still publishing only standard Android apps you should not have to make any changes to the *App Version Code* field in the Android **Builds Settings** section of the **Projects** tab.

The workaround described above *only* applies to Crosswalk CLI 5.1.1 and later builds!

When you build a Crosswalk app with CLI 4.1.2 (which uses Cordova-Android 3.6) you will get the old Intel XDK behavior where: 60000 and 20000 (six with four zeros and two with four zeroes) are added to the **android:versionCode** for Crosswalk builds and no change is made to the **android:versionCode** for standard Android builds.

NOTE:

- Android API 14 corresponds to Android 4.0
- Android API 19 corresponds to Android 4.4
- Android API 20 corresponds to Android 5.0
- CLI 5.1.1 (Cordova-Android 4.x) does not allow building for Android 2.x or Android 3.x

Why is my Crosswalk app generating **errno 12 Out of memory** errors on some devices?
If you are using the WebGL 2D canvas APIs and your app crashes on some devices because you added the *--ignore-gpu-blacklist* flag to your intelxdk.config.additions.xml file, you may need to also add the *--disable-accelerated-2d-canvas* flag. Using the *--ignore-gpu-blacklist* flag enables the use of the GPU in some problem devices, but can then result in problems with some GPUs that are not blacklisted. The *--disable-accelerated-2d-canvas* flag allows those non-blacklisted devices to operate properly in the presence of WebGL 2D canvas APIs and the *--ignore-gpu-blacklist* flag.

You likely have this problem if your app crashes after running a few seconds with the an error like the following:

<gsl_ldd_control:364>: ioctl fd 46 code 0xc00c092f (IOCTL_KGSL_GPMEM_ALLOC) failed: errno 12 Out of memory
<ioctl_kgsl_sharedmem_alloc:1176>: ioctl_kgsl_sharedmem_alloc: FATAL ERROR : (null).

See *Chromium Command-Line Options for Crosswalk Builds with the Intel XDK* for additional info regarding the *--ignore-gpu-blacklist* flag and other Chromium option flags.

Construct2 Tutorial: How to use AdMob and IAP plugins with Crosswalk and the Intel XDK.
See this tutorial on the Scirra tutorials site > How to use AdMob and IAP official plugins on Android-Crosswalk/XDK < written by Construct2 developer Kyatric.

Also, see this blog written by a Construct2 game developer regarding how to build a Construct2 app using the Appodeal ad plugin with your Construct2 app and the Intel XDK > *How to fix the build error with Intel XDK and Appodeal?* <.

What is the correct "Target Android API" value that I should use when building for Crosswalk on Android?
The "Target Android API" value (aka android-targetSdkVersion), found in the Build Settings section of the **Projects** tab, is the version of Android that your app and the libraries associated with your app are tested against, it **DOES NOT** represent the maximum level of Android onto which you can install and run your app. When building a Crosswalk app you should set to this value to that value recommend by the Crosswalk project.

The recommended "Target Android API" levels for Crosswalk on Android apps are:

- 18 for Crosswalk 1 thru Crosswalk 4
- 19 for Crosswalk 5 thru Crosswalk 10
- 21 for Crosswalk 11 thru Crosswalk 18

As of release 3088 of the Intel XDK, the recommended value for your android-targetSdkVersion is 21. In previous versions of the Intel XDK the recommended value was 19. If you have it set to a higher number (such as 23), we recommend that you change your setting to 21.

Can I build my app with a version of Crosswalk that is not listed in the Intel XDK Build Settings UI?
As of release 3088 of the Intel XDK, it is possible to build your Crosswalk for Android app using versions of the Crosswalk library that are *not* listed in the **Project** tab's *Build Settings* section. You can override the value that is selected in the Build Settings UI by adding a line to the intelxdk.config.additions.xml file.

NOTE: The process described below is for experts only! By using this process you are effectively disabling the Crosswalk version that is selected in the Build Settings UI and you are overriding the version of Crosswalk that will be used when you build a custom debug module with the Debug tab.

When building a Crosswalk for Android application, with CLI 5.x and higher, the Cordova Crosswalk Webview Plugin is used to facilitate adding the Crosswalk webview library to the build package (the APK). That plugin effectively "includes" the specified Crosswalk library when the app is built. The version of the Crosswalk library selected in the *Build Settings* UI is effected by a line in the Android build config file, similar to the following:

<intelxdk:crosswalk version="16"/>
The line above is added automatically to the intelxdk.config.**android**.xml file by the Intel XDK. If you attempt to change lines in the Android build config file they will be overwritten by the Intel XDK each time you use the **Build** tab (perform a build) or the **Test** tab. In order to modify (or override) this line in the Android config file you need to add a line to the intelxdk.config.**additions**.xml file.

The precise line you include in the intelxdk.config.**additions**.xml file depends on the version of the Crosswalk library you want to include.

<!-- Set the Crosswalk embedded library to something other than those listed in the UI. --> <!-- In practice use only one, multiple examples are shown for illustration. --> <preference name="xwalkVersion" value="17+"/> <preference name="xwalkVersion" value="14.43.343.24" /> <preference name="xwalkVersion" value="org.xwalk:xwalk_core_library_beta:18+"/>
The first example line in the code snippet above asks the Intel XDK to build with the "last" or "latest" version of the Crosswalk 17 release library (the '+' character means "last available" for the specified version). The second example requests an explicit version of Crosswalk 14

when building the app (e.g., version 14.43.343.24). The third example shows how to request the "latest" version of Crosswalk 18 from the Crosswalk beta Maven repository.

NOTE: only one such "xwalkVersion" preference tag should be used. If you include more than one "xwalkVersion" only the last one specified in the intelxdk.config.additions.xml file will be used.

The specific versions of Crosswalk that you can use can be determined by reviewing the Crosswalk Maven repositories: one for released Crosswalk libraries and one for beta versions of the Crosswalk library.

Not all Crosswalk libraries are guaranteed to work with your built app, especially the beta versions of the Crosswalk library. There may be library dependencies on the specific version of the Cordova Crosswalk Webview Plugin or the Cordova-Android framework. If a library does not work, select a different version.

Detailed instructions on the preference tag being used here are available in the Crosswalk Webview Plugin README.md documentation.

If you are curious when a specific version of Chromium will be supported by Crosswalk, please see the Crosswalk Release Dates wiki published by the Crosswalk Project.

My Construct2 Crosswalk app flashes a white box or white band after the splash screen.
The white box or white bands you see between the ending of the splash screen and the beginning of your app appears to be due to some webview initialization. It also appears in non-Crosswalk apps on Android, but does not show up as white. The white band that does appear can cause an initial "100% image" to bounce up and down momentarily. This issue is not being caused by the splash screen plugin or the Intel XDK; it appears to be interference caused by the Cordova webview initialization.

The following solution appears to work, although there may be some situations that it does not help. As this problem is better understood more information will be provided in this FAQ.

Add the following lines to your intelxdk.config.additions.xml file:

<platform name="android"> <!-- set Crosswalk default background color --> <!-- see http://developer.android.com/reference/android/graphics/Color.html --> <preference name="BackgroundColor" value="0x00000000" /> </platform>
The value 0x00000000 configures the webview background color to be "transparent black," according to the Cordova documentation and the Crosswalk webview plugin code. You should be able to set that color to anything you want. However, this color appears to work the best.

You may also want to add the following to your intelxdk.config.additions.xml file:

<platform name="android"> <!-- following requires the splash screen plugin --> <!-- see https://github.com/apache/cordova-plugin-splashscreen for details --> <preference name="SplashScreen" value="screen" /> <preference name="AutoHideSplashScreen" value="false" /> <!-- <preference name="SplashScreenDelay" value="30000" /> --> <preference name="FadeSplashScreen" value="false"/> <!-- <preference name="FadeSplashScreenDuration" value="3000"/> --> <preference name="ShowSplashScreenSpinner" value="false"/> <preference name="SplashMaintainAspectRatio" value="false" /> <preference name="SplashShowOnlyFirstTime" value="false" /> </platform>
Testing of this fix was done with Crosswalk 17 on an Android 4.4, Android 5.0 and an Android 6.0 device.

Back to FAQs Main

Categories:

Intel® XDK FAQs

Intel News - Fri, 04/14/2017 - 17:18

Click the FAQ titles below to view specific FAQ pages.

General FAQs
Getting started as a new user; installing and updating the Intel XDK; questions related to the Brackets editor; differences between mobile platforms, etc.

Cordova FAQs
Using Cordova* APIs; adding and using third-party plugins; selecting plugins for your app; AdMob and in-app purchases; Intel App Security; image capture; camera, etc.

Crosswalk FAQs [RETIRED]
Using the Crosswalk* runtime with Android*; why are Crosswalk app packages so large; controlling audio playback rate; Crosswalk GPU support; Crosswalk options, etc.

Debug & Test FAQs
Enable testing via wifi with Intel App Preview; limitations of the Intel XDK simulator; debugging third-party plugins over USB, etc.

App Designer FAQs [DEPRECATED]
The App Designer layout editor; the App Framework library; creating and sizing UI elements; widget attributes; updating App Framework versions, etc.

IoT FAQs
Developing Internet of Things (IoT) NodeJS* apps using the Intel XDK; updating the MRAA library; connecting the Intel XDK to your IoT device; using the WebService API, etc.

Categories:

Intel® XDK FAQs - App Designer [DEPRECATED]

Intel News - Fri, 04/14/2017 - 17:15

[DEPRECATED] App Designer (UI layout tool) has been deprecated!
IMPORTANT: the Intel XDK App Designer component (aka the UI layout tool) has been deprecated. It will be retired in an upcoming release. Once retired, existing App Designer projects will continue to work, but you will not be able to create new App Designer projects.

No bug fixes will be implemented for the existing App Designer component nor for any of the UI frameworks that were supported by App Designer.

If you have designed your layout by hand or by using an external tool, there will be no changes to your project. This change ONLY affects projects that have been created using the App Designer UI layout tool. If you are just starting with the Intel XDK we recommend that you do NOT use App Designer to create your layout, since the editor will not be maintained and may eventually be discontinued.

There are many UI frameworks and tools available for creating UI layouts; too many to enumerate here. The vast majority of layout tools that generate standard HTML5 code (HTML/CSS/JavaScript) should work with no issue. The Intel XDK creates standard Cordova CLI (aka PhoneGap) applications, so any UI frameworks and tools that work in the Cordova CLI environment will work with your Intel XDK applications.

Which App Designer framework should I use? Which Intel XDK layout framework is best?

There is no "best" UI framework for your application. Each UI framework has pros and cons. You should choose that UI framework which serves your application needs the best. Using App Designer to create your UI is not a requirement to building a mobile app with the Intel XDK. You can create your layout by hand or using any UI framework (by hand) that is compatible with the Cordova CLI (aka PhoneGap) webview environment.

- **Twitter Bootstrap 3** -- This UI framework has been ***deprecated*** and will be retired from App Designer in a future release of the Intel XDK. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using Twitter Bootstrap please visit the Twitter Bootstrap website and the Twitter Bootstrap GitHub repo for documentation and help.

- **Framework7** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using Framework7 please visit the Framework7 project page and the Framework7 GitHub repo for documentation and help.

- **Ionic** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using Ionic please visit the Ionic project page and the Ionic GitHub repo for documentation and help.

- **App Framework 3** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to

continue using App Framework please visit the App Framework project page and the App Framework GitHub repo for documentation and help.

- **Topcoat** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using Topcoat please visit the Topcoat project page and the Topcoat GitHub repo for documentation and help.

- **Ratchet** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using Ratchet please visit the Ratchet project page and the Ratchet GitHub repo for documentation and help.

- **jQuery Mobile** -- This UI framework has been ***retired*** from App Designer. You can always use this (or any mobile) framework with the Intel XDK, but you will have to do so manually, without the help of the Intel XDK App Designer UI layout tool. If you wish to continue using jQuery Mobile please visit the jQuery Mobile API page and jQuery Mobile GitHub page for documentation and help.

What does the Google* Map widget's "center type" attribute and its values "Auto calculate," "Address" and "Lat/Long" mean?
The "center type" parameter defines how the map view is centered in your div. It is used to initialize the map as follows:

- Lat/Long: center the map on a specific latitude and longitude (that you provide on the properties page)
- Address: center the map on a specific address (that you provide on the properties page)
- Auto Calculate: center the map on a collection of markers

This is just for initialization of the map widget. Beyond that you must use the standard Google maps APIs to move and/or modify the map. See the "google_maps.js" code for initialization of the widget and some calls to the Google maps APIs. There is also a pointer to the Google maps API at the beginning of the JS file.

To get the current position, you have to use the Geo API, and then push that into the Maps API to display it. The Google Maps API will not give you any device data, it will only display information for you. Please refer to the Intel XDK "Hello, Cordova" sample app for some help with the Geo API. There are a lot of useful comments and console.log messages.

How do I size UI elements in my project?
Trying to implement "pixel perfect" user interfaces with HTML5 apps is not recommended as there is a wide array of device resolutions and aspect ratios and it is impossible to insure you are sized properly for every device. Instead, you should use "responsive web design" techniques to build your UI so that it adapts to different sizes automatically. You can also use the CSS media query directive to build CSS rules that are specific to different screen dimensions.

Note:The viewport is sized in CSS pixels (aka virtual pixels or device independent pixels) and so the physical pixel dimensions are not what you will normally be designing for.

How do I create lists, buttons and other UI elements with the Intel XDK?
The Intel XDK provides you with a way to build HTML5 apps that are run in a webview on the target device. This is analogous to running in an embedded browser (refer to this blog for details). Thus, the programming techniques are the same as those you would use inside a browser, when writing a single-page client-side HTML5 app. You can use the Intel XDK App Designer tool to drag and drop UI elements.

Why is the user interface for Chrome on Android* unresponsive?
It could be that you are using an outdated version of the App Framework* files. You can find the recent versions here. You can safely replace any App Framework files that App Designer installed in your project with more recent copies as App Designer will not overwrite the new files.

How do I work with more recent versions of App Framework* since the latest Intel XDK release?
You can replace the App Framework* files that the Intel XDK automatically inserted with more recent versions that can be found here. App designer will not overwrite your replacement.

Is there a replacement to XPATH in App Framework* for selecting nodes from an XML document?
This FAQ applies only to App Framework 2. App Framework 3 no longer includes a replacement for the jQuery selector library, it expects that you are using standard jQuery.

App Framework is a UI library that implements a subset of the jQuery* selector library. If you wish to use jQuery for XPath manipulation, it is recommend that you use jQuery as your selector library and not App Framework. However, it is also possible to use jQuery with the UI components of App Framework. Please refer to this entry in the App Framework docs.

It would look similar to this:

<script src="lib/jq/jquery.js"></script> <script src="lib/af/jq.appframework.js"></script> <script src="lib/af/appframework.ui.js"></script> Why does my App Framework* app that was previously working suddenly start having issues with Android* 4.4?
Ensure you have upgraded to the latest version of App Framework. If your app was built with the now retired Intel XDK "legacy" build system be sure to set the "Targeted Android Version" to 19 in the Android-Crosswalk build settings. The legacy build targeted Android 4.2.

How do I manually set a theme?

If you want to, for example, change the theme only on Android*, you can add the following lines of code:

1. $.ui.autoLaunch = false; //Stop the App Framework* auto launch right after you load App Framework*
2. Detect the underlying platform using either navigator.userAgent or intel.xdk.device.platform or window.device.platform. If the platform detected is Android*, set $.ui.useOSThemes=false todisable custom themes and set <div id=”afui” class=”android light”>
3. Otherwise, set $.ui.useOSThemes=true;
4. When device ready and document ready have been detected, add $.ui.launch();

How does page background color work in App Framework?

In App Framework the BODY is in the background and the page is in the foreground. If you set the background color on the body, you will see the page's background color. If you set the theme to default App Framework uses a native-like theme based on the device at runtime. Otherwise, it uses the App Framework Theme. This is normally done using the following:

<script> $(document).ready(function(){ $.ui.useOSThemes = false; }); </script>
Please see *Customizing App Framework UI Skin* for additional details.

What kind of templates can I use to create App Designer projects?

Currently, you can only create App Designer projects by selecting the blank 'HTML5+Cordova' template with app designer (select the app designer check box at the bottom of the template box) and the blank 'Standard HTML5' template with app designer.

App Designer versions of the layout and user interface templates were removed in the Intel XDK 3088 version.

My AJAX calls do not work on Android; I'm getting valid JSON data with an invalid return code.

The jQuery 1 library appears to be incompatible with the latest versions of the cordova-android framework. To fix this issue you can either upgrade your jQuery library to jQuery 2 or use a technique similar to that shown in the following test code fragment to check your AJAX return codes. See this forum thread for more details.

The jQuery site only tests jQuery 2 against Cordova/PhoneGap apps (the Intel XDK builds Cordova apps). See the **How to Use It** section of this jQuery project blog > https://blog.jquery.com/2013/04/18/jquery-2-0-released/ for more information.

If you built your app using App Designer, it may still be using jQuery 1.x rather than jQuery 2.x, in which case you need to replace the version of jQuery in your project. Simply download and replace the existing copy of jQuery 1.x in your project with the equivalent copy of jQuery 2.x.

Note, in particular, the switch case that checks for zero and 200. This test fragment does not cover all possible AJAX return codes, but should help you if you wish to continue to use a jQuery 1 library as part of your Cordova application.

function jqueryAjaxTest() { /* button #botRunAjax */ $(document).on("click", "#botRunAjax", function (evt) { console.log("function started"); var wpost = "e=132&c=abcdef&s=demoBASICA"; $.ajax({ type: "POST", crossDomain: true, //;paf; see http://stackoverflow.com/a/25109061/2914328 url: "http://your.server.url/address", data: wpost, dataType: 'json', timeout: 10000 }) .always(function (retorno, textStatus, jqXHR) { //;paf; see http://stackoverflow.com/a/19498463/2914328 console.log("jQuery version: " + $.fn.jquery) ; console.log("arg1:", retorno) ; console.log("arg2:", textStatus) ; console.log("arg3:", jqXHR) ; if(parseInt($.fn.jquery) === 1) { switch (retorno.status) { case 0: case 200: console.log("exit OK"); console.log(JSON.stringify(retorno.responseJSON)); break; case 404: console.log("exit by FAIL"); console.log(JSON.stringify(retorno.responseJSON)); break; default: console.log("default switch happened") ; console.log(JSON.stringify(retorno.responseJSON)); break ; } } if((parseInt($.fn.jquery) === 2) && (textStatus === "success")) { switch (jqXHR.status) { case 0: case 200: console.log("exit OK"); console.log(JSON.stringify(jqXHR.responseJSON)); break; case 404: console.log("exit by FAIL"); console.log(JSON.stringify(jqXHR.responseJSON)); break; default: console.log("default switch happened") ; console.log(JSON.stringify(jqXHR.responseJSON)); break ; } } else { console.log("unknown") ; } }); }); } What do the data-uib and data-ver properties do in an App Designer project?

App Designer adds the data-uib and data-ver properties to many of the UI elements it creates. These property names only appear in the index.html file on various UI elements. There are other similar data properties, like data-sm, that only are required when you are using a service method.

The data-uib and data-ver properties are used only by App Designer. They are not needed by the UI frameworks supported by App Designer; they are used by App Designer to correctly display and apply widget properties when you are operating in the "design" view within App Designer. These properties *are not critical* to the functioning of your app; however, removing them will cause problems with the "design" view of App Designer.

The data-sm property is inserted by App Designer, and it may be used by data_support.js, along with other support libraries. The data-sm property *is relevant* to the proper functioning of your app.

Unable to select App Designer UI option when I create a new App Designer project.

If you previously created an App Designer project named 'ui-test' that you then delete and then create another App Designer project using the same name (e.g., 'ui-test'), you will not be given the option to select the UI framework for the new project named 'ui-test.' This is because the Intel XDK remembers a framework name for each project name that has been used and does not delete that entry from the global-settings.xdk file when you delete a project (e.g. if you chose "Framework 7" the first time you created an App Designer project with the name 'ui-test' then deleting 'ui-test' and creating a new 'ui-test' will result in another "Framework 7" project).

Because the UI framework name is not removed from the global-settings.xdk file when you delete the project, you must either use a new unique project name or edit the global-settings.xdk file to delete that old UI framework association. This is a bug that has been reported, but has not been fixed. Following is a workaround:

- Close the Intel XDK.
- Open the global-settings.xdk file with a text or code editor (Sublime, Notepad++, Brackets, etc.). See this FAQ for how to locate the global-settings.xdk file on your system.
- Look for the project name you are trying to reuse at the end of the file. For example for a project name 'ui-test':

"FILE-/C/Users/xxx/Downloads/pkg/ui-test/www/index.html": { "canvas_width": 320, "canvas_height": 480, "framework": "framework 7" }

- Remove the last line ("framework": "framework 7") from the JSON object (remember to remove the comma at the end of the preceding line or you won't have a proper JSON file and your global-settings.xdk file will be considered corrupt).
- Save and close the global-settings.xdk file.
- Launch the Intel XDK.
- Create a new project with old name you are reusing.

You should now see the list of App Designer framework UI selection options when you create the new project with a previously used project name that you have deleted.

Back to FAQs Main

Categories:

Intel® MKL 2018 Beta is now available - draft

Intel News - Fri, 04/14/2017 - 02:56

Intel® MKL 2018 Beta is now available as part of the Parallel Studio XE 2018 Beta.
Check the **Join the Intel® Parallel Studio XE 2018 Beta program** post to learn how to join the Beta program, and the provide your feedback.

What's New in Intel® MKL 2018 Beta:

- DNN:
  - Added initial convolution and inner product optimizations for Intel(R) Xeon Phi(TM) processors based on Intel(R) Advanced Vector Extensions 512 (Intel(R) AVX-512) with support of AVX512_4FMAPS and AVX512_4VNNIW instruction groups.
  - Average pooling has an option to include padding into mean values computation
- BLAS Features:
  - Introduced optimized integer matrix-matrix multiplication routines (GEMM_S16S16S16 and GEMM_S16S16S32) to work with quantized matrices for all architectures.
  - Introduced ?TRSM_BATCH to complement the batched BLAS for all architectures
- BLAS Optimizations:
  - Optimized SGEMM, GEMM_S16S16S16 and GEMM_S16S16S32 for Intel(R) Xeon Phi(TM) processors based on Intel(R) Advanced Vector Extensions 512 (Intel(R) AVX-512) with support of AVX512_4FMAPS and AVX512_4VNNIW instruction groups
  - Improved ?GEMM_BATCH performance for all architectures
  - Improved single and multi-threaded {D,S}SYMV performance for Intel® Advanced Vector Extensions 512 (Intel® AVX-512) and the Intel® Xeon Phi™ processor x200
- Sparse BLAS:
  - Improved performance of CSRMV/BSRMV functionality for Intel® AVX-512 instruction set in Inspector-Executor mode
- LAPACK:
  - Introduced factorization and solve routines based on Aasen's algorithm: ?sytrf_aa/?hetrf_aa, ?sytrs_aa/?hetrs_aa
- Vector Mathematics:
  - Added 24 new functions: v?Fmod, v?Remainder, v?Powr, v?Exp2; v?Exp10; v?Log2; v?Logb; v?Cospi; v?Sinpi; v?Tanpi; v?Acospi; v?Asinpi; v?Atanpi; v?Atan2pi; v?Cosd; v?Sind; v?Tand; v?CopySign; v?NextAfter; v?Fdim; v?Fmax; v?Fmin; v?MaxMag; v?MinMag
- Library Engineering:
  - Introduced support for Intel(R) Xeon Phi(TM) processors based on Intel(R) Advanced Vector Extensions 512 (Intel(R) AVX-512) with support of AVX512_4FMAPS and AVX512_4VNNIW instruction groups.

Optimizations are not dispatched unless explicitly enabled with mkl_enable_instructions function call or MKL_ENABLE_INSTRUCTIONS environment variable.

- Documentation:
  - Starting with this version of Intel MKL, most of the documentation for Parallel Studio XE is only available online at https://software.intel.com/en-us/articles/intel-math-kernel-library-documentation. You can also download it from the Intel Registration Center > Product List > Intel® Parallel Studio XE Documentation Beta.
- **Hardware Support for Intel® Xeon Phi™ Coprocessors (code name Knights Corner) is removed.** Customers are recommended to stay on MKL 2017 given they continue to use and develop for Intel® Xeon Phi™ Coprocessors (aka Knight Corner)

Categories:

DOE Supercomputer Achieves Record 45-Qubit Quantum Simulation

HPC Wire - Thu, 04/13/2017 - 20:21

In order to simulate larger and larger quantum systems and usher in an age of “quantum supremacy,” researchers are stretching the limits of today’s most advanced supercomputers. Just as classical computing systems have been instrumental in advancing their own forward progression, today’s fastest machines are helping pave the way for quantum computing breakthroughs, which will be revolutionary for applications in quantum chemistry, material science, machine learning, and cryptography.

A research team from ETH Zurich in Switzerland recently succeeded in simulating the largest quantum device yet — a 45-qubit circuit — using the Knights Landing-based Cori II machine at Lawrence Berkeley National Laboratory. Cori II has 9,304 compute nodes each outfitted with one 68-core Intel “Knights Landing” Xeon Phi 7250 processor, linked with the Cray Aries interconnect, with a total peak performance of 29.1 petaflops and 1 PB of aggregate memory.

Thomas Häner and Damian S. Steiger of the Institute for Theoretical Physics at ETH Zurich performed simulations of low-depth random quantum circuits, which were proposed by Google to demonstrate quantum supremacy. In the paper describing the work, the authors specify that “the execution of low-depth random quantum circuits is not scientifically useful on its own” but “running such circuits is of great use to calibrate, validate, and benchmark near-term quantum devices.”

“In order to make use of the full potential of systems featuring multi- and many-core processors, we use automatic code generation and optimization of compute kernels, which also enables performance portability,” they write.

Summary of all simulation results carried out on Cori II. (Source)
To simulate the 45-qubit quantum circuit, Häner and Steiger used 8,192 Cori II nodes and a total of 0.5PB of memory, achieving an average 0.428 petaflops. In explaining the low performance in relation to peak FLOPS, the team refer to 1) heavy communication overhead (75 percent of circuit simulation time spent in communication) and state 2) kernel performance suffers where few kqubit gates are applied before a global-to-local swap needs to be performed (see section 4.1.2 for further analysis and discussion).

The research team says to the best of their knowledge, this 45-qubit quantum circuit simulation is the largest ever conducted. “Our highly-tuned kernels in combination with the reduced communication requirements allow an improvement in time-to-solution over state-of-the-art simulators by more than an order of magnitude at every scale,” they write.

The next step for the ETH Zurich team is to add more qubits to their simulation. Although 49-qubits is widely-held as the point at which quantum devices surpass the most capable traditional supercomputers and thwart larger simulations, the researchers have a plan to reach this threshold.

“While we do not carry out a classical simulation of 49 qubits, we provide numerical evidence that this may be possible,” the research team states. “Our optimizations allow reducing the number of communication steps required to simulate the entire circuit to just two all-to-alls, making it possible to use, e.g., solid-state drives if the available memory is less than the 8 petabytes required.”

Cori is the flagship resource of the DOE National Energy Research Scientific Computing Center (NERSC). The system was named in honor of the American biochemist Gerty Cori, the first American woman to win a Nobel Prize in science. Cori II is currently ranked number five on the (November 2016) Top500 list. The full system is comprised of two partitions: 2,004 Intel Xeon “Haswell” processor nodes and 9,300 Intel Xeon Phi “Knight’s Landing” nodes. According to its Top500 submission, the KNL partition (Cori II) has a peak performance of 27.9 petaflops and a measured Linpack score of 14 petaflops.

The 11-page paper, “0.5 Petabyte Simulation of a 45-Qubit Quantum Circuit,” is published on arXiv.org. There’s also a writeup of the research at *MIT Technology Review*.

The post DOE Supercomputer Achieves Record 45-Qubit Quantum Simulation appeared first on HPCwire.

Categories: RSS Feed Reader

Getting Started with Intel® Cluster Checker for Linux*

Intel News - Thu, 04/13/2017 - 19:47

Intel® Cluster Checker verifies the configuration and performance of Linux based clusters and checks compliance with the Intel® Scalable System Framework architecture specification. If issues are found, Intel® Cluster Checker diagnoses the problems and may provide

recommendations on how to repair the cluster.

Intel® Cluster Checker has the following features:

- Dynamic detection of cluster configuration, operation, and performance issues.
- Problem diagnoses with severity and confidence levels.
- On-demand data collection.

Intel® Cluster Checker is installed as part of the following suites:

- Intel® Parallel Studio XE 2018 Cluster Edition.
- For Intel® Scalable System Framework partners, as a stand-alone product.

The following flowchart represents the usage model for working with the Intel® Cluster Checker.

Prerequisites

1. Install Intel® Cluster Checker using the bundled installer.
2. We recommend running the tool as a non-root user. Before using Intel® Cluster Checker for the first time, the runtime environment must be setup. Two files are included to setup the runtime environment, clckvars.sh for shells with Bourne syntax and clckvars.csh for shells with C-shell syntax. Source the appropriate file from the command line. For example:

   source /opt/intel/clck/2018.0/bin/clckvars.sh

3. Create a text file that lists the compute nodes in the cluster using one hostname per line. In these examples, this file is named "nodefile". Here is an example for one head node and four compute nodes:

   frontend #role: head
   node1
   node2
   node3
   node4

   For detailed system requirements, see the "System Requirements" section in the *Intel® Cluster Checker Release Notes*

Step 1: Collect data
Run the following from a command line. *nodefile* should be in a shared & writeable location.

clck-collect -a -f nodefile

Step 2: Analyze the data
Run this from a command line:

clck-analyze -f nodefile

Resolve any issues reported in step 2 and repeat steps 1 and 2 until you are satisfied with the results.

By default, diagnosed signs are not included in the analyzer output. If the analyzer reports issues, then it will be beneficial to output diagnosed signs on subsequent runs. More data about signs and diagnoses can be found in the *User's Guide*. Run this from a command line to print diagnosed signs:

clck-analyze -f nodefile -p diagnosed_signs

There will be occasions where modifications of the default XML configuration file are needed. This can happen when more output is desired, test parameters need to be modified, the log level must be changed, etc. More information can be found in the *User's Guide*.

Troubleshooting/FAQ
Files will be installed into /opt/intel/clck/2018.0.

- **For help with the collector, run**:

  clck-collect --help

- **For help with the analyzer, run**:

clck-analyze --help

- **To view collected data, use the database query tool.**
  For help with the query tool, run:

  clckdb --help

- **To customize the analysis behavior**:
  Make a copy of the default XML file.

  cp /opt/intel/clck/2018.0/etc/clck.xml ~

  Edit the XML file options.

  To use a custom XML file with the analyzer, run the following (if the custom XML file is named "~/clck.xml"):

  clck-analyze -f nodefile -c ~/clck.xml

Documentation and Resources
All of the following documents can be found at https://software.intel.com/en-us/intel-cluster-checker-support/documentation:

Document Description Intel® Cluster Checker Developer's Guide Contains a breakdown of the following components: the knowledge base, the connector, and the database schema. Intel® Cluster Checker User's Guide Contains a description of the product, including the following components and processes: the analyzer, knowledge base, connector, data collection, data providers, and the database schema. Intel® Cluster Checker Release notes Contains a brief overview of the product, new features, system requirements, installation notes, documentation, known limitations, technical support, and the disclaimer and legal information.

Categories:

Recent Updates - Request Form

Intel News - Thu, 04/13/2017 - 18:24

Categories:

Larry Smarr Talks Machine Intelligence at Jackson State

HPC Wire - Thu, 04/13/2017 - 16:01

JACKSON, Miss., April 13, 2017 — Dr. Larry Smarr, physicist and Big Data thought leader, practices what he preaches.

When asked about the auto-pilot option packaged in the Tesla, he proudly says: "I own one. I have the model X. It drives me from UC San Diego up to UC Irvine and back every time."

It's Monday, April 10, the day before Smarr, founding director of California Institute of Telecommunications and Information Technology (Calit2), is to appear in Jackson State University's CSET Engineering auditorium to present his theory on the mounting growth of machine artificial intelligence.

Big Data thought leader Larry Smarr, holder of the Harry E. Gruber professorship in UCSD's Department of Computer Science and Engineering lectured and facilitated an open discussion in the auditorium of the College of Science, Engineering and Technology. Professor Smart speculated on the exponentially growing machine intelligence and how it will increasingly inter-operate with human intelligence. (Charles A. Smith/University Communications)

JSU and Calit2 are partners on National Science Foundation-funded grants, like SCOPE — Scalable Omnipresent Environment — a visual metaphor for a combined microscope and telescope that enables users to explore data from the nano to the micro to the macro to the mega scale. Another is already implemented at Jackson State – a camera system called SENSEI, a/k/a the virtual wall, capable of capturing 3D stereo and still images for viewing in nationally networked, virtual reality systems.

Seemingly unfazed by a fatal 2016 accident that involved a Tesla set on auto pilot, Smarr points out that the U.S. National Highway Traffic Safety Administration cleared the automaker of any fault in the incident.

"I'm much less concerned that my Tesla will have an accident than some stupid human driving a car is going to have an accident," he quips then smiles.

Smarr, of course, backs his opinion with science, saying, "There are ultrasound, radar and optical sensors on each of the Teslas. As they're driving, the data stream of how they're interacting with the world is sent wirelessly over the Internet to the Tesla cloud."

What happens next, according to Smarr, is that a “hive mind” develops where the experiences of all the Teslas are shared with each other in the cloud, and a machine-learning algorithm goes in and improves the way that all Teslas drive.

“So, the more Teslas there are and the more miles they drive, the safer everybody becomes,” he says matter of factly.

The Harvard Junior Fellow adds that the average car driver’s knowledge base is dependent upon whatever the driver recalls and not 10,000 similar vehicles second to second sharing data.

To call Smarr a big deal is an understatement. His bio contains phrases like “theoretical observational” and “computational astrophysics,” a field he has pioneered for 25 years. He is credited with galvanizing the early development of foundational components of U.S. global cyberinfrastructure and most recently became a leader of the quantified self-movement or “life logging.”

In addition to being a member of the National Academy of Engineering, Smarr holds memberships with several distinguished science organizations. He has also served on the NASA Advisory council to four NASA administrators as well as other esteemed positions.

If who and what Smarr is does not stimulate the brain even slightly, then his description of the work conducted at the Calit2 Pattern Recognition Laboratory should be an eye-opener.

“So there are new kinds of computer chips emerging that are specialized for recognizing patterns in images, for instance, a hearing aid. What a hearing aid is trying to do is recognize the pattern of a voice in a noisy environment,” he says.

Through specialized processors like hearing aids or voice authentication technology, Calit2 develops machine learning that can identify patterns and collects this information into a lab that faculty and students can access in order to conduct experiments to understand how, particularly, Big Data can be analyzed more efficiently.

“So, it’s a pattern that’s in a bunch of data, which as a human, if you just look at all of the data then you can’t figure anything out. But if you have a specialized computer that can do it then maybe you can,” he says.

Smarr and Calit2 are working with everyone from startups to big name companies like IBM that may be in the infancy stages of new technology and need assistance in determining how it may be best used.

“So, by having a lot of faculty and particularly students who may have a lot of clever ideas that faculty may not think of,” he chuckles, “they get access to the lab so they can conduct experiments.”

In addition to determining uses for new technology, Smarr adds that identifying market niches is also an outcome of Big Data research.

Testing science in more ways than cruising in his Tesla, Smarr is engaged in a computer-aided study of his own body – lifelogging. Every month, the noted director relinquishes 5-6 vials of his blood for analysis and every quarter he hands over up to 20 vials of the life-giving substance.

Smarr says his venture into the “quantified self” movement stems from his 25 years of experience as an astrophysicist. Over that period, he has conducted studies like taking measurements over time (a time series) to figure out how an unusual point came to be in the sky and then determine what that point is doing at various stages.

“I began to think about the body as a dynamic, multi-component, nonlinear system. And I thought, well, surely, that’s the way people are trying to figure out what’s going on with you before it gets to the point that you have a symptom,” he says.

But to his confoundment, Smarr discovered his theory was not the way medicine is conducted.

“So, I said why don’t I make myself a lab animal and turn my body into an observatory and see if we can begin to understand the dynamics of some fundamental things like inflammation, glucose and insulin and things like cholesterol. All that kind of stuff,” he says.

Expounding on his quest to become a human guinea pig, Smarr discloses one of his goals: “Can we figure out unintended consequences of some of our activities? Whether it’s medical treatments or whether it’s things like our nutrition, exercise, sleep or other things that people are arbitrary about what they do as if there are not going to be any consequences and there are.”

Pulling out his smartphone, he reveals an image of what he describes as “70 different biomarkers in me” taken over 20 years with each marker representing things like glucose, cholesterol, sodium and potassium levels. Through a color-coordinated effort, Smarr is able to see when his body is healthy and when it exceeds the healthy range by a certain percentage.

“I thought I was healthy,” he laughs as he glances at the rainbow-colored images.

Smarr has data compiled from over 20 years but explains that he’s been doing the blood work and intensively studying microbiome – the general composite material present in or on the human body – for about five years.He declares that if others were able to see clear indicators that revealed the onslaught of Alzheimer’s or diabetes, then people could possibly make some behavioral changes and choices to stop diseases before they develop.“By bringing in the big data that can be read out of your body,” Smarr hopes to improve various areas of medicine.

“I just lifelog a little more intensely than most people. And I’m a scientist, and I have a big laboratory, so I can afford to do it. He explains, “I’m not saying you should be like me. What I’m trying to be is the patient of the future, so in the future, the cost will go way down, and it will be

much less evasive to measure these things."

Smarr readily admits that it's an experiment, and "it's a little weird being me, but somebody has to do it."

Experimentation appears to be working for the scientist whose research and curiosity is taking him to heights he had not imagined if that's easy to believe after reading his bio.

"I never in a million years thought I would end up working with surgeons and medical doctors. I didn't have any training on biomedical things," says Smarr, referring to his recent surgery to remove a portion of his colon.

In 2012, after an MRI, Smarr was less than inspired by the 2D black and white splices he was shown by his radiologist. He then requested the images and the 3D data that is routinely included in individual medical evaluations.

Together with his "virtual reality guy," Jurgen Schulze, he created a 3D image of his abdomen.

After being diagnosed with colonic Crohn's disease late last year, he offered the 3D image to his surgeon, Dr. Sonia Ramamoorthy, several days before the removal of the inflamed area.

In short, Ramamoorthy's review of the detailed image led her to change her original plan for the incisions she would make to Smarr's abdomen and reduced his operating time by nearly 30 minutes.

Smarr again pulls out his phone and this time he shows a video of himself lying unconscious on the operating table while Ramamoorthy works simultaneously with the surgical robot and the 3D imagery.

"It was like having a 3D Google map," he jokes.

A week after his procedure, he recalls Ramamoorthy saying, "Oh my God, I had a patient today. It would've been so helpful if we had this."

Smarr's innovative thinking has the potential to not only benefit the medical industry but society as a whole. "The fact that you can be under anesthesia less means that recovery chances would be improved." He says, "So, we're making a program at UC San Diego, in the medical school, to then simplify the software to make it more integrated with the robot and then train and do another 10 patients."

After learning that a lot of Smarr's esteemed accomplishments started with having an organic knack for investigation, it is easy to imagine him as a kid making a Tesla coil and blacking out all the televisions and radios in his neighborhood as he once said he did.

When asked what type of skills or characteristics a student would need to become a Larry Smarr, he says: "Well, I think you have to be intensely curious, and then I guess you can do what President Reagan said about the Russians – 'Trust but verify.'"

"Think" is the operative term Smarr wants kids to learn. "Doing your Facebook updates a billion times a day, well, I'm sure there is some good that comes from that – it is not thinking. And the other thing is reading is so important."

He conveys that while playing video games results in some positives, like the ability to multitask, an inadvertent outcome is that kids' attention spans have grown shorter. He becomes quiet as if searching for the right words to describe his thoughts; then he adds that the younger generation is no longer reading.

"When you think about the way books are written – the very best books, like the very best art and the very best music – you are having a personal mentor on how to organize your thoughts, how to communicate, how to convince someone of something or how to paint a word picture," he says.

"You don't know how to do that to start with, but as you read, your brain is remembering patterns like that's how you write, that's how you phrase your words, that's how you communicate. So this is not generally talked about – why you need to read. But if you don't, then you dumb down the population because an incredible number of things are going to happen in your lifetime and how are you going to be prepared to know how to react?

About Jackson State University: Challenging Minds, Changing Lives

Jackson State University, founded in 1877, is a historically black, high research activity university located in Jackson, the capital city of Mississippi. Jackson State's nurturing academic environment challenges individuals to change lives through teaching, research and service. Officially designated as Mississippi's Urban University, Jackson State continues to enhance the state, nation and world through comprehensive economic development, health-care, technological and educational initiatives. The only public university in the Jackson metropolitan area, Jackson State is located near downtown, with five satellite locations throughout the area. For more information, visit www.jsums.edu or call 601-979-2121.

Source: Rachel James-Terry, Jackson State University

The post Larry Smarr Talks Machine Intelligence at Jackson State appeared first on HPCwire.

Categories: RSS Feed Reader

CERN openlab Explores New CPU/FPGA Processing Solutions

HPC Wire - Thu, 04/13/2017 - 15:00

At CERN, the European Organization for Nuclear Research, physicists and engineers are probing the fundamental structure of the universe. The Large Hadron Collider (LHC), which began working in 2008, is the world's largest and most powerful particle accelerator; it is housed in an underground tunnel at CERN. Niko Neufeld is a deputy project leader at CERN who works on the Large Hadron Collider beauty (LHCb) experiment, which explores what happened after the Big Bang that allowed matter to survive and build the Universe we inhabit today.

"CERN experiments produce an enormous amount of data with forty million proton collisions every second, which leads to primary data rates of terabits per second," says Neufeld when speaking on a recent FPGA vs. CPU panel. "This is an enormous amount of data and there are a number of technical challenges in our work. We use a number of processing solutions including central processing units (CPUs), field-programmable gate arrays (FPGAs), and graphic processing units (GPUs), but each of these solutions have some limitations. We are collaborating with Intel in experimenting with a co-packaged Intel Xeon processor plus FPGA Quick Path Interconnect (QPI) processor in our LHCb research to try to determine which technology provides the best results."

CERN collaborates with leading ICT companies and other research institutes through a unique public-private partnership known as 'CERN openlab'. Its goal is to accelerate the development of cutting-edge solutions for the worldwide LHC community and wider scientific research. Through a CERN openlab project known as the 'High-Throughput Computing Collaboration,' researchers are investigating the use of various Intel technologies in data filtering and data acquisition systems.

Figure 1. CERN researchers shown in the Large Hadron Collider tunnel in front of the LHCb detector. Courtesy of CERN (courtesty CERN).

Introducing the co-packaged Intel CPU / FPGA Processor

Today the CPU and FPGA are used as discrete chips in a solution – with an Intel Xeon processor and an FPGA which is typically attached via a PCIe interconnect to the CPU. And the development environment is also discrete using independent development tools from Intel and tools such as OpenCL and C++. Intel is working toward a common workflow and development flow to better integrate FPGAs.

"FPGA typically uses a higher level machine abstraction language (such as Verilog and VHDL) which have a painful low-level hardware programming model for most people. As a next step, Intel has a solution that co-packages the CPU and FPGA in the same Multichip Chip Product (MCP) package to deliver higher performance and lower latency than a discrete solution," states Bill Jenkins, Intel Senior AI Marketing Manager. The Intel MCP is supported by a cross-platform development framework like OpenCL that can be used to develop applications for both the CPU and FPGA. The Intel solution includes a fully unified intellectual property (IP) and development suite, including languages, libraries and development environments. The roadmap to a unified development flow leverages common tools and libraries to support both FPGA and Intel Xeon processor + FPGA systems along with an expansive ecosystem network of Intel and vendors working on independent development tools for demanding workloads such as HPC, imaging identification, security and big data.

Abstracting away FPGA Coding

Intel is building an abstraction layer (as part of the product containing the Intel CPU and FPGA in the same MCP package), called the Orchestration Software layer. This layer and the higher level IP and software models help make development less complex so that developers don't need to code specifically to the FPGA. The FPGA-enabled Orchestration software layer abstracts away the API to communicate with the FPGA as shown in the following example.

Figure 2. Example of Intel implementation of user IP implemented into FPGA via an abstraction Orchestration software layer

There is a cloud-based library of functions and end-user IP that have been pre-compiled and built that is loaded into the FPGA at runtime. The user first launches a workload from the host and it goes into the Orchestration software which pushes a function into the FPGA. This produces a bitstream that is pre-compiled on the FPGA to bring the data in—it is almost like a fixed architecture I/O interface.

In the example scenario, users simply download the image from the abstraction Orchestration software layer to the FPGA and it is ready to run without compilation. "With the abstraction Orchestration software layer," Jenkins explained, "Intel is abstracting away all the difficulties of FPGA programming using machine language tools while enabling all the higher level Intel frameworks including the Intel Trusted Analytics Platform (TAP) and Intel Scalable System Framework (SSI) and tying the FPGA into the frameworks. Intel is developing this approach for a variety of markets including visual understanding, analytics, enterprise, Network Function Virtualization (NFV), VPN, genomics, HPC and storage."

Large Hadron Collider High-Energy Physics Research at CERN

Neufeld indicates that the experiments at CERN — through what they refer to as 'online computing' — require a first-level data-filtering to reduce the data to an amount that can be stored and processed on more traditional processing units such as Intel Xeon processors. Figure 3 shows a schematic view of the future LHCb readout system. At the top level, there is a detector and optical fiber links, which transfer data out of the detector. CERN uses FPGAs to acquire data from the detector. There are also large switching fabrics, as well as clusters of processing elements including CPUs, FPGAs, and GPUs to reduce the amount of data. One of the questions the CERN team is testing is "Which technologies should we use and which provide the best performance and lowest energy usage results?"

Figure 3. Schematic diagram showing future LHCb first-level data-filtering system. Courtesy of CERN.

CERN Tests Complex Cherenkov Angle Reconstruction Calculation

CERN has extensive experience using FPGAs in their research work. "We typically use FPGAs in our research to run algorithms looking for simple integer signatures, or for other less complicated calculations. When we heard about the Intel Xeon / FPGA combined processor, we chose a test using a complex algorithm to do a Cherenkov angle reconstruction of light emission in a particle detector, which is not typically performed on an FPGA. This involves tracing a light particle — photon — through a complex arrangement of optical reflection and deflection

systems. Our test case used a rich PID algorithm to calculate the Cherenkov angle for each track and detection point. This is a complex mathematical calculation that involves hyperbolic functions, roots, square roots, etc., as shown in Figure 4. It is one of the most costly calculations done in online reconstruction," states Neufeld.

Figure 4. Test case running Rich PID algorithm to calculate Cherenkov angle. Courtesy of CERN.

Coding the Cherenkov Angle Reconstruction in Verilog versus OpenCL

The CERN team first implemented the Cherenkov angle reconstruction by coding it in the Verilog HDL. The team wrote a 748 clock-cycle long pipeline in Verilog, along with additional blocks developed for the test including: cubic root, complex square root, rotational matrix, and cross/scalar product. It was a lengthy task doing this coding in Verilog with 3,400 lines of code. With all test benches, the implementation took 2.5 months.

Next, the team recoded the Cherenkov angle code using the OpenCL and the BSP (board support package) designed to work across a variety of hardware platforms. Because OpenCL is an abstraction language, it required only 250 lines of code and took two weeks of coding. Not only was coding in OpenCL much faster but the performance results were similar. Figure 5 shows the results of the Verilog versus OpenCL implementation.

Figure 5. Result of Verilog (CQRT) versus OpenCL (RICH) code and performance. Courtesy of CERN.

CERN Compares Co-packaged Intel Xeon – FPGA Processor against Nallatech PCIe Stratix V FPGA Board

To test performance of the Verilog code, the CERN team used a commercially available Stratix V GXA7 FPGA board / Nallatech 385 board for testing. They achieved an acceleration of a factor up to six with the Stratix – Nallatech PCIe board. However, they found a bottleneck in data transfer—they could not keep the pipeline busy because the PCIe card was limited to an eight-lane interface. Next, the CERN team did tests with the Cherenkov angle code comparing a Nallatech FPGA Board with the co-packaged Intel Xeon/FPGA QPI processor.

Finally, the CERN team tested an Intel Xeon CPU, PCIe Stratix V FPGA and Intel Xeon processor/Stratix V QPI (where only the interconnect was different). As shown in Figure 6, there was a factor of 9 speed up for the PCIe Stratix V FPGA and a 26 factor speed up for the Intel Xeon processor/Stratix V QPI with the faster interconnect.

Figure 6. Test results from the CERN team comparing Intel Xeon CPU, PCIe Stratix V FPGA and Intel Xeon processor/FPGA QPI. Courtesy of CERN.

CERN Plans to do Future Testing using co-packaged Intel Xeon/ Intel Arria10 FPGA Processor

"Our CERN team found the results of using the co-packaged Intel Xeon processor/Stratix V QPI processors to be very encouraging. In addition, we find the programming model with OpenCL attractive and it will be mandatory for the High-Energy Physics (HEP) field. Intel will be launching a co-packaged Intel Xeon processor / Intel Arria 10 FPGA processor in the future. We want to do other experiments with the co-packaged Intel Xeon processor/ Arria 10 FPGA. We expect that the high-bandwidth interconnect and modern Arria 10 FPGA card will provide high performance and performance per Joule for HEP algorithms," states Neufeld.

Linda Barney is the founder and owner of Barney and Associates, a technical/marketing writing, training and web design firm in Beaverton, OR.

The post CERN openlab Explores New CPU/FPGA Processing Solutions appeared first on HPCwire.

Categories: RSS Feed Reader

Engility to Pursue NASA Advanced Computing Services Opportunity

HPC Wire - Thu, 04/13/2017 - 10:08

CHANTILLY, Va., April 13, 2017 — Engility Holdings, Inc. (NYSE: EGL), announced today that the company will bring its world-class high performance computing (HPC) capabilities to bear as it competes to win NASA's Advanced Computing Services contract.

"HPC is a strategic, enabling capability for NASA," said Lynn Dugle, CEO of Engility. "Engility's cadre of renowned computational scientists and HPC experts, coupled with our proven high performance data analytics solutions, will help increase NASA's science and engineering capabilities."

Engility subject matter experts are on the forefront of developing systems and solutions that leverage integrated, multi-scaled scientific and engineering models. Engility applies sophisticated visualization, numerical algorithms and computational frameworks to complex scientific and engineering problems. These can be executed on multi-peta scale computing systems to develop HPC solutions, simulations and tools. These solutions, simulations and tools enable warfighters, academics and policymakers to better understand complex phenomena, reduce time to action, enhance their productivity and move complex technologies from the laboratory to the operational mission while saving and improving lives.

Engility's legacy of HPC success stretches back a quarter century. In 1993, the company reduced U.S. Air Force weather model time by 75 percent. Engility also delivered the National Oceanic and Atmospheric Administration's (NOAA's) first commodity-based 64-bit cluster in 2004. In addition, the company established the advanced computing research program in support of the Army's HPC Research Center, partnering with universities and government labs such as NASA Ames Research Center in 2007. Just last year, Engility won a $112 million Food and Drug Administration HPC contract that improves scientific computing. For instance, bioinformatics insights provide faster, better-informed regulatory decisions and enhance collaboration among scientists and the worldwide scientific community.

Engility's HPC team:

- Includes some of the best computational and supercomputing scientists in the industry, enabling advanced HPC users to work across all scientific and engineering domains. Engility scientists worked with NOAA to enable the design and acquisition of their first large research and development HPC systems.
- Secures data. Engility delivers in-depth knowledge, support and direction for all IT security-related activities at NOAA's Geophysical Fluid Dynamics Laboratory's Research and Development HPC Program.
- Helps customers strategize and prioritize investments in HPC. Engility developed a tool that enabled NOAA to make strategic investments in HPC technology and acquisition that resulted in an accelerated path to next-generation weather and climate modeling.

For more information about Engility's high performance computing expertise, please visit http://www.engilitycorp.com/services/hpc/.

About Engility

Engility (NYSE: EGL) is engineered to make a difference. Built on six decades of heritage, Engility is a leading provider of integrated solutions and services, supporting U.S. government customers in the defense, federal civilian, intelligence and space communities. Our innovative, highly technical solutions and engineering capabilities address diverse client missions. We draw upon our team's intimate understanding of customer needs, deep domain expertise and technical skills to help solve our nation's toughest challenges. Headquartered in Chantilly, Virginia, and with offices around the world, Engility's array of specialized technical service offerings include high-performance computing, cyber security, enterprise modernization and systems engineering. To learn more about Engility, please visit www.engilitycorp.com and connect with us on Facebook, LinkedIn and Twitter.

Source: Engility

The post Engility to Pursue NASA Advanced Computing Services Opportunity appeared first on HPCwire.

Categories: RSS Feed Reader

Data Science is an Ocean of Information—Stay Focused!

Intel News - Wed, 04/12/2017 - 19:25

A primer on how to become a data scientist

How do I become a good data scientist? Should I learn R* or Python*? Or both? Do I need to get a PhD? Do I need to take tons of math classes? What soft skills do I need to become successful? What about project management experience? What skills are transferable? Where do I start?

Data science is a popular topic in the tech world today. It is the science that powers many of the trends in this world, from machine learning to artificial intelligence.

In this article, we discuss our teachings about data science in a series of steps so that any product manager or business manager interested in exploring this science will be able take their first step toward becoming a data scientist or at least develop a deeper understanding of this science.

Step 1: Define a Problem Statement
We all have heard conversations that go sometime like this: "Look at the data and tell me what you find." This approach may work when the volume of data is small, structured, and limited. But when we are dealing with gigabytes or terabytes of data, it can lead to an endless, daunting detective hunt, which provides no answers because there were no questions to begin with.

As powerful as science is, it's not magic. Inventions in any field of science solve a problem. Similarly, the first step in using data science is to define a problem statement, a hypothesis to be validated, or a question to be answered. It may also focus on a trend to be discovered, an estimate, a prediction to be made, and so on.

For example, take MyFitnessPal*, which is a mobile app for monitoring health and fitness. A few of my friends and I downloaded it about a year ago, and then used it almost daily for a while. But over the past 6 months, most of us have completely stopped using it. If I were a product manager for MyFitnessPal, a problem I might want to solve would be: how can we drive customer engagement and retention for the app?

Step 2: Get the Data
Today's data scientists access data from several sources. This data may be structured or unstructured. The **raw data** that we often get is unstructured and/or dirty data, which needs to be cleaned and structured before it can be used for analysis. Most of the common sources of data now offer connectors to import the raw data in R or Python.

Common data sources include the following:

- Databases
- CSV files
- Social media feeds like Twitter, Facebook, and so one (unstructured)

- JSON
- Web-scraping data (unstructured)
- Web analytics
- Sensor data driven by the Internet of Things
- Hadoop*
- Spark*
- Customer interview data
- Excel* analysis
- Academic documents
- Government research documents and libraries like www.data.gov
- Financial data; for example, from Yahoo Finance*

In the data science world, common vocabulary includes:

- **Observations or examples.** These can be thought of as horizontal database records from a typical database.
- **Variables, signals, characteristics.** These equate to the fields or columns in the database world. A variable could be qualitative or quantitative.

⇄ **Observations or examples** ⇄ are like the rows in a database. ⇄ For example: A customer record for Joe Allen. ⇅ **Variables, signals, or characteristics** ⇅ are like the columns ⇅ For example: Joe's Height. Step 3: Cleaning the Data
Several terms are used to refer to data cleaning, such as data munging, data preprocessing, data transformation, and data wrangling. These terms all refer to the process of preparing the raw data to be used for data analysis.

As much as 70–80 percent of the efforts in a data science analysis involve data cleansing.
A data scientist analyzes each variable in the data to evaluate whether it is worthy of being a feature in the model. If including the variable increases the model's predictive power, it is considered a predictor for the model. Such a variable is then considered a **feature**, and together all the features create a **feature vector** for the model. This analysis is called **feature engineering**.

Sometimes a variable may need to be cleaned or transformed to be used as a feature in the model. To do that we write scripts, which are also referred to as **munging scripts.** Scripts can perform a range of functions like:

- Rename a variable (which helps with readability and code sharing)
- Transform text (if "variable = "big" set variable = "HUGE")
- Truncate data
- Create new variables or transpose data (for example, given the birth date, calculate age)
- Supplement existing data with additional data (for example, given the zip code, get the city and state)
- Convert discrete numerical variables into a continuous range (for example: salary-to-salary range; age-to-age range)
- Date and time conversions
- Convert a categorical variable into multiple binary variables. For example, a categorical variable for region (with possible values being east, west, north, and south) could be converted into four binary variables, east, west, north, and south, with only one of them being true for an observation. This approach helps create easier joins in the data.

Sometimes the data has numerical values that vary in magnitude, making it difficult to visualize the information. We can resolve this issue using **feature scaling.** For example,consider the square footage and number of rooms in a house. If we normalize the square footage of a house by making it a similar magnitude as the number of bedrooms, our analysis becomes easier.

A series of scripts are applied to the data in an iterative manner until we get data that is clean enough for analysis. To get a continuous supply of data for analysis, the series of data munging scripts need to be rerun on the new raw data. **Data pipeline** is the term given to this series of processing steps applied to raw data to make it analysis ready.

Step 4: Data Analysis and Model Selection
Now we have clean data and we are ready for analysis. Our next goal is to become familiar with the data using statistical modeling, visualizations, discovery-oriented data analysis, and so on.

For simple problems, we can use **simple statistical analysis** using the mean, medium, mode, min, max, average, range, quartile, and so on.

Supervised Learning
We could also use **supervised learning** with data sets that gives us access to actual values of response variables (dependent variables) for a given set of feature variables (independent variables). For example, we could find trends based on the tenure, seniority, and title for employees who have left the company (resigned=true) from actual data, and then use those trends to predict whether other employees will resign too. Or we could use historic data to correlate a trend between the number of visitors (an independent variable or a predictor) and revenue generated (a dependent variable or response variable). This correlation could then be used to predict future revenue for the site based on the number of visitors.

The key requirement for supervised learning is the availability of ACTUAL Values and a clear question that needs to be answered.
For example: Will this employee leave? How much revenue can we expect? Data scientists often refer to this as "*Response variable is labeled*

for existing data."

Regression is a common tool used for supervised learning. A one-factor regression uses one variable; a multifactor regression uses many variables.
Linear regression assumes that the unknown relation between the factor and the response variable is a linear relation Y = a + bx, where b is the **coefficient** of x.

A part of the existing data is used as **training data** to calculate the value of this coefficient. Data scientists often use 60 percent, 80 percent, or at times 90 percent of the data for training. Once the value of the coefficient is calculated for the **trained model**, it is tested with the remaining data also referred to as the **test data** to predict thevalue of the response variable. The difference between the predicted response value and the actual value is the Holy Grail of metrics referred to as the **test error metric.**

Our quest in data science modeling is to **minimize the test error metrics** in order to increase the predictive power of the model by:

- Selecting effective factor variables
- Writing efficient data munging scripts
- Selecting the appropriate statistical algorithms
- Selecting the required amount of test and training data

Unsupervised Learning
Unsupervised learning is applied when we are trying to learn the structure of the underlying data itself. There is NO RESPONSE VARIABLE. Data sets are unlabeled and pre-existing insights are unclear. We are not clear about anything ahead of time so we are not trying to predict anything!
This technique is effective for exploratory analysis and can be used to answer questions like

- Grouping. How many types of customer segments do we have?
- Anomaly detection. Is this normal?

Analysis of variance (ANOVA) is a common technique used to compare the means of two or more groups. It's named ANOVA though since the "estimates of variance" is the main intermediate statistics calculated. The means of various groups are compared using various distance metrics, **Euclidean distance** being a popular one.

ANOVA is used to organize observations into similar groups, called **clusters**. The observations can be **classified** into these clusters based on their respective predictors.
http://www.statsdirect.com/help/content/analysis_of_variance/anova.htm

Two common clustering applications are:

- **Hierarchical clustering.** A bottom-up approach. We start with individual observations and merge them with the closest one. We then calculate the means of these grouped observations and merge the groups with the means closest to each other. This is repeated until larger groups are formed. The distance metrics is defined ahead of time. This technique is complex and not advisable with a high-dimension data set.
  Hierarchical ANOVA

- **K-means clustering.** Uses a partitioning approach.
  - We assume that the data has a **fixed number of clusters** in advance based on our intuition.
  - We also assume the **starting center for each cluster**.
  - Each observation is then assigned to the cluster with the mean closest to the observation
  - The step is repeated until all observations have been assigned to a cluster.
  - Now we recalculate the mean for the clusters based on the average of all the observations assigned to the cluster.
  - Observations are reclassified to these new cluster and steps c, d, and e are repeated until they reach a stable state.

If a stable state is not achieved, we may need to refine the number of clusters (i.e., K) we assumed in the beginning or use a different distance metrics.

Step 5: Visualize and Communicate Effectively
The final clusters can be visualized for easy communication using tools like Tableau* or graphing libraries.

Tips from Data Science Practitioners
In my quest to understand data science, I met with practitioners working in companies, including Facebook, eBay, LinkedIn, Uber, and some consulting firms, that are effectively leveraging the power of data. Here are some powerful words of advice I received:

- **Know your data.** It's important to fully understand the data and the assumptions behind it. Otherwise, the data may be ineffectively used, which may lead to arriving at the wrong answer, solving the wrong problem, or both.
- **Understand the domain and the problem.** The data scientist must have a deep understanding of the business domain and the problem to be solved to be able to extract the appropriate insights from the data.

- **Ethics.** Don't compromise data quality to fit a hypothesis. **The problem often is not ignorance, but our preconceived notions!**
- It's a myth that a larger data set always offers better insights. Although an increased amount of data becomes statistically significant, a large data set also comes with higher noise. It's common to see the R-squared of a larger data set that is smaller than that of a smaller data set.
- While data science is not a product by itself, it can power brilliant products that solve complex problems. Product managers and data scientists that communicate effectively can become strong partners:
    - The product manager initially brings to the conversation the business problem to be solved, questions to be answered, and constraints to be discovered and/or defined.
    - The data scientist, who brings deep expertise in machine learning and mathematics, focuses on the theoretical aspects of the business problem. Modern data sets are used to perform data analysis, transformations, model selections, and validations to establish the **foundations of the theory to be applied to the business problem**.
    - The software engineer works to operationalize the theory and the solution. He or she needs a strong understanding of the mechanics of machine learning (Hadoop clusters, data storage hardware, writing production code, and so on).
- Learn a programming language. Python is easiest to learn; R is considered the most powerful.

Commonly Used Data Science Tools R
R is a favorite tool of many data scientists and holds a special place in the world of academia, where data science problems are worked on from a mathematician's and statistician's perspective. R is an open source and rich language, with about 9,000 additional packages available. The tool used to program in R is called R Studio*. R has a steep learning curve, though its footprint is steadily increasing in enterprise world and owes some of it popularity to the rich and powerful Regular Expression-based algorithms already available.

Python
Python is slowly becoming the most extensively used language in the data science community. Like R, it is also an open source language and is used primarily by software engineers who view data science as a tool to solve real customer-facing business problems using data. Python is easier to learn than R, because the language emphasizes readability and productivity. It is also more flexible and simpler.

SQL
SQL is the basic language used to interact with databases and is required for all tools.

Other Tools
- Apache Spark offer **Scala***
- **MATLAB*** is a mathematical environment that academia has used for a long time. It offers an open source version called Octave*
- Java* is used for Hadoop environments

What about the Soft Skills?
Below is a list of important soft skills to have, many of which you might already have in your portfolio.

- **Communication.** A data scientist doesn't sit in a cube and code Python programs. The data science process requires that you mingle with your team. You need to connect and build rapport with executives, product owners, product managers, developers, big data engineers, NoSQL* experts, and more. Your goal is to understand what they are trying to build and how data science and machine learning can help.
- **Coaching.** As a data scientist, your coaching skills will shine. You are not an individual contributor of the company; you are the CEO's best friend, who can help him or her shape the company—the product and the territory based on data science. For example, based on your data science results, you give perspective analysis results to the executive team recommending that the company launch dark-green shoes in Brazil; the same product will fail in Silicon Valley in the United States. Your findings can save millions of dollars for the company.
- **Storyteller.** A good data scientist is a good storyteller. During your data science project, you will have tons of data, tons of theories, and tons of results. Sometimes you'll feel as if you are lost in an ocean of data. If this happens, step back and think: What are we trying to achieve? For example, if your audience is a CEO and COO, they might need to make an executive decision in a couple of minutes based on your presentation. They aren't interested in learning about your ROC curve or in going through the 4 terabytes of data and 3,000 lines of your Python code.
  Your goal is to give them direct recommendations based on your solid prediction algorithm and accurate results. We recommend that you create four or five slides where you clearly tell this story—storytelling backed by solid data and solid research.**Visualization.** Good data scientist needs to communicate results and recommendations using visualization. You cannot give 200-page report for someone to read. You need to present using pictures, images, charts, and graphs.

- **Mindset.** A good data scientist has a "hacker" mind—"hacker" being used here in a good way—and is relentlessly looking for patterns in the data set.
- **Love thy data**. You need to live with your data and let it tell you the story. Of course, there are many tools you can use to more fully understand the data, but just a superficial glance at it will give you lots of information.

What Can I Become?
Now it's time to decide. What type of data scientist should I become?

- **Understand the pipeline.** You need to start somewhere. You can be a Python developer working in a data science project. You can gather input data coming from logs, sensors, CSV file, and so on. You can write scripts to consume and ingest the incoming data. Data can be still or in-motion. You might decide to become a big data engineer working with technologies like Hadoop or Hive*. Or a machine learning algorithm specialist—someone who has mastered the skills and understands which algorithm works best under which problem. You can be a math genius, who can play with machine learning out-of-the-box algorithms and modify them according to your needs. You might become a data persistence expert. You might use SQL or NoSQL technologies to persist/serve data. Or you might become a data visualization expert and build dashboards and data stories using tools like Tableau. So check out the above pipeline one more time: **from ingestion to visualization.** Make an opportunity list. For example, "D3 expert, Python script expert, Spark master" and so on.
- **Check out the job market.** Look at the various job portals to get an idea of the current demand. How many jobs are there? What jobs are in highest demand? What is the salary structure? A casual glance at data science jobs in the Bay Area shows promising opportunities ►
- **Understand yourself.** You have explored the pipeline and type of jobs you can get. Now it's time to think about yourself and your skills. What do you enjoy most and what experience do you have? Do you love project management? Databases? Think about your previous success stories. Do you love writing complex scripts to correlate and manipulate data? Are you a visualization person, who is an expert at creating compelling presentations? Make a "love-to-do-list." For example, "love to code, love scripts, love Python."
- **Create a match.** Match your opportunity list with your love-to-do list and get on with the program. Data science is an ocean of information. Stay focused!!

Categories:

The Ports of the Intel(R) Software License Manager

Intel News - Wed, 04/12/2017 - 18:24

The Intel® Software License Manager uses **two** ports to serve licenses - one for lmgrd (the main license service) and one for the INTEL vendor daemon. **Both ports must be open and not blocked by a firewall.**

lmgrd - FlexNet daemon
This is the main process that controls license management, and is provided by FlexNet Publisher, formerly Flexlm. The Intel Software License Manager uses port 28518 as a default to avoid conflicts with other vendors. This can be entered through the Intel Registration Center during activation, or changed for activated licenses by following these steps.

INTEL - vendor daemon
This is the vendor daemon that serves Intel licenses. When lmgrd is started or restarted, it starts the vendor daemon which determines a port to use. At this time, the selected port number is displayed in the startup output, which is either written to a log file or stdout. There is no additional reporting by the license manager utilities on this port, causing it to be overlooked.

As firewalls have become more common, so have reports of issues stemming from the INTEL vendor daemon port being blocked. Even if the port was not previously blocked, restarting the license manager can cause the port number to change and be subsequently blocked. To determine the port number, run a command such as netstat and look for the INTEL daemon.

The INTEL vendor daemon port can be specified by modifying the license file. Change the second line as follows:

VENDOR INTEL port=<port>

Take care in changing the license file, or you may invalidate it.
Be sure to restart the license manager after any license file changes.

Categories:

Steps to register a floating license

Intel News - Wed, 04/12/2017 - 17:27

How you register your floating license depends on how it was issued. Registration is the process of owning a particular serial number, while Activation is assigning the owned serial number to a license server.

Registration
If you have a serial number which has no owner, you may register it by following this process:

1. Go to the Intel® Registration Center
2. On the registration screen in the Register a Product section, enter your email address in the Email box.
3. Enter the same email address in the Confirm Email box.
4. Enter the serial number in the Serial Number box and then click Register Product.
5. Follow the instructions on the web page to download and install your product.
6. After registering, you will receive an email confirming the registration. A license file will not provided at this point.

If you already have a registration center account, you may login and enter the unregistered serial number in the upper-right serial number box.

If the serial number is already registered, the above process will automatically add you as a user of the license. This grants the ability to download the products available with the license. If you expected to become the license owner, you can contact support to assist with determining the current owner and/or license transfer.

Activation
To activate your floating license, you must provide the host ID and host name of the server running the license manager. This can be done in one of two ways:

- If the server has connectivity with the registration center, you can provide the serial number during installation of the Intel Software License Manager and it will automatically provide the host information to generate the license file.
- If the server cannot submit the host information, or if you want to manually activate the serial number, you must do the following:

1. Login to the registration center
2. Click the serial number under the "Serial Numbers" tab - if the serial number is greyed out, you are not the owner or administrator
3. Enter the host information
4. Click "Activate Serial Number"

After the serial number is activated, you may download the license file.

Categories:

Tutorials Schedule Announced for PEARC17

HPC Wire - Wed, 04/12/2017 - 16:52

NEW ORLEANS, April 12, 2017 — PEARC17 organizers today announced 20 full-day and half-day tutorials for attendees at the Practice & Experience in Advanced Research Computing 2017 conference, July 9-13, in New Orleans.

All tutorials will be held Monday, July 10. Of two full-day tutorials, one focuses on enabling and advancing on-campus research computing and the other on manycore programming. Half-day tutorials are scheduled on topics ranging from using science gateways, cloud computing, virtual clusters, programming, and managing scientific data to visualization, data sharing, and building data portals.

When registering for tutorials day, registrants will be asked to select the tutorials they are likely to attend. While attendees may ultimately choose to attend different tutorials on site, tutorials selected by fewer than five attendees during the registration period may be cancelled.

Anyone interested in advanced research computing challenges is encouraged to review the schedule and register soon.

About PEARC

The PEARC (Practice & Experience in Advanced Research Computing) conference series is being ushered in with support from many organizations and will build upon earlier conferences' success and core audiences to serve the broader community. In addition to XSEDE, organizations supporting the new conference include the Advancing Research Computing on Campuses: Best Practices Workshop (ARCC), the Science Gateways Community Institute (SGCI), the Campus Research Computing Consortium (CaRC), the ACI-REF consortium, the Blue Waters project, ESnet, Open Science Grid, Compute Canada, the EGI Foundation, the Coalition for Academic Scientific Computation (CASC), and Internet2.

Source: PEARC

The post Tutorials Schedule Announced for PEARC17 appeared first on HPCwire.

Categories: RSS Feed Reader

Penguin Takes a Run at the Big Cloud Providers

HPC Wire - Wed, 04/12/2017 - 12:57

Fighting for sway among the big HPC-in-the-cloud providers – AWS, Azure, etc. – is challenging for smaller players. The giants own the mindshare. HPC specialist Penguin Computing recently re-ran benchmarks from a study of its larger brethren and says the results show its 'public cloud' – Penguin on Demand (POD) – is among the leaders in cost and performance. The natural caveat here is Penguin ran the tests. The company says the tests accurately recreate those in the study and that it is willing to let prospects or third parties run the tests for themselves at no charge.

A marketing exercise? Of course. The Penguin effort was in response to a paper – *Comparative benchmarking of cloud computing vendors with High Performance Linpack* – authored by Exabyte.io and posted on arXiv.org last month. But besides simply being an effort by Penguin to gain user attention and push its way more forcefully into the land of giants such as Microsoft, Google, AWS, and IBM/SoftLayer, and Rackspace, all examined in the paper, the effort also reveals the growing competitive zeal among many providers to offer HPC in the cloud.

Where Microsoft Azure was the top performer in the original study, Penguin reports that POD comes out ahead in the new benchmarking it performed on both price and performance measures. Penguin, according to its measurements, takes the lead in terms of Linpack gigaflops-

per-core, Linpack gigaflops-per-node and has it has the most favorable speedup at scale.

To some extent one would expect Penguin to outperform public cloud providers. POD is best characterized as an on-demand HPC cluster rather than a typical public cloud. It is a tightly coupled environment (with the option of OmniPath 100Gb/s interconnect, which was used for this testing). Definitional nuances aside, the bigger takeaway, if accurate, may be the POD is cheaper for HPC workloads than the major cloud players.

HPCwire covered the earlier paper (see, Azure Edges AWS in Linpack Benchmark Study). In it, performances varied with Azure besting the pack, which included a NERSC run. The primary purpose of the study was to demonstrate whether or not HPC workloads can be reliably and cost-effectively run in the cloud. Using the high performance Linpack as the performance metric and a cost comparison with traditional infrastructure, the authors' answer is an unqualified yes.

"We benchmarked the performance of the best available computing hardware from public cloud providers with high performance Linpack. We optimized the benchmark for each computing environment and evaluated the relative performance for distributed memory calculations…Based on our findings we suggest that the concept of high performance computing in the cloud is ready for a widespread adoption and can provide a viable and cost-efficient alternative to capital-intensive on-premises hardware deployments," write the authors, Mohammad Mohammadi, Timur Bazhirov of Exabyte.io.

Penguin, of course, is well known for its HPC expertise and Tundra servers. In November the company had seven systems on the Top500. Not surprisingly, the POD offering is HPC-centric. "We recognize the need for high performance nodes with modern processors, bare metal, non-blocking fabric, the kind of infrastructure you would design if you were building a large scale HPC cluster in house," says Victor Gregario, Sr. VP of Cloud Services.

POD provides two locations for cloud computing that can be accessed through a single POD Portal, MT1 and MT2. Each location has localized storage, and high-speed interconnects between the sites to facilitate easy migration of data from one location to another.

Login nodes and storage volumes (user home directories) are local to each location, but usernames are global to all of POD's locations. The POD Portal's usage reports are global to all POD locations.

The tests were run at MT2 datacenter on POD's B30 nodes, which feature Intel Broadwell E5-2680 v4 processors (2.4Ghz, 14 cores, 16 double precision Flops/cycle) and deliver 1.07 Tflops peak per node. The MT2 location uses Intel OmniPath interconnect. The other clouds in the study do not have the current highest-end networking, although the Azure nodes (AZ-A and AZ-H) and NERSC Edison do employ high-speed interconnects (40Gbs InfiniBand and Cray Aries respectively) — details of the various node architectures/processors are best gleaned from the Exabyte.io paper. Penguin ran the High Performance Linpack test.

As shown in the table and figures below, Penguin argues POD performs on par with the best of the big providers and that its costs are less, at least as evaluated in the study.

By this comparison, POD consistently performed at or near the top with performance scaling well with increased node/core count. The cost comparison is even more disparate, despite POD having a significantly higher initial rate per node, $2.80 per node hour versus the highest cost of the majors, $1.90/node/hr. for the Azure IB-A node. Buyer beware: Penguin compared publicly disclosed rates and did not factor in potential discounts.

Source: Penguin Computing

Source: Penguin Computing

The cost reduction drivers, according to Penguin, include higher performance, reduced wall clock time, fine grain metering down to roughly three seconds versus rounding up to the nearest hour used, and the lack of many common additional charges such as data transfer, bandwidth, set-up, etc., says Gregario.

"You are not billed for idle time, you are only billed for compute time. But the biggest thing is the other cloud vendors will round up to the hour," he says. A job that ran 90 minutes would cost the same as one that ran 61 minutes or 62 minutes in billing schemes that round up. Most jobs, he notes, are not nicely divisible by hours. Penguin meters down to roughly three seconds of use. "When you look at what I feel is a real-world situation we are clearly more cost-effective than other providers for HPC workloads," Gregario says.

Currently, there are five different queues on POD and each pointing to a cluster with different resources and capabilities. Lower performing environments are lower. That said, all of them have been designed for HPC workloads according to Gregario.

Penguin doesn't say much about the size of the POD customer base. Sid Mair, SVP, Federal Systems, emphasized POD is the fastest growing part of the company and that its customers are not drawn only from existing Penguin customers for on-premises equipment and services. One POD customer is running multi-thousand core jobs, he says while there's a university with on the order of 300 students submitting "tiny" jobs daily. Weather forecasting, automotive, and traditional engineering disciplines are all represented. Gregario considers a 4000-core job to be a big one.

"Almost every HPC application on the market place runs on POD and many of them are already loaded in the environment we use. We have many relationships where customers transfer their corporate licenses right into POD rather than have to worry about managing them," says Gregario, adding the list of applications on the web-site slightly lags their ongoing efforts add more applications and tools. Most of the familiar names are already there such as ANSYS, Dassult Systems, and MathWorks.

Perhaps surprisingly, the GPU node offering (NVIDIA K40) is modest when compared to recent aggressive adoption of K80 and P100 devices by the major cloud players. Gregario says Penguin is demand-driven and able to scale as the need arises.

“As we see demand in the market place for things like deep learning, we’ll adopt it. We are currently working with a number of customers for on-premises deep learning/machine learning environments and we’re using that experience to understand the needs for optimal HPC environment in the cloud. We aren’t ready to publicly disclose our plans yet,” says Gregario.

The post Penguin Takes a Run at the Big Cloud Providers appeared first on HPCwire.

Categories: RSS Feed Reader

1 2 3 4 5 6 7 8 9 … next › last »