

Project #1

Student Names and SFUIDs:

Chenyu Ru	301323578
Jiangpei Chen	301326516

Purpose: This pair-project focuses on documentation for a software package.

Tasks:

Phase One (Today – Wednesday January 29th):

- Find a partner for this week-long project. Please do not ask me to pair you up. If you cannot find a partner, then you will have to work alone.
- Become familiar with the concept of pair-programming – know the pros and cons and be able to classify it within the context of software engineering practices.
- Read and understand *everything* in the following blog post:

<https://www.ybrikman.com/writing/2014/05/05/you-are-what-you-document/>

Phase Two (Due next Friday, February 7th):

- You will then be set upon a week-long task next class (Friday, June 3rd) involving the *three types of documentation* referred to in the post above.
- Submission instructions will be provided next week by the TA.
- You will have the rest of next week to work on this project outside of class in your pairs.
- Use this document as your cover page

IMPORTANT NOTE: I do not make attendance mandatory. However, it is the student's responsibility to find out what was discussed from your peers. Assignments and in-class activities are designed so that attendance is NOT mandatory. However, those that do not attend will find the assignment somewhat difficult.

As the summary of the principle reference *You are what you document*, it summarized the documents into three different types. Written documentation, Code documentation, Community documentation. *Each type of documentation solves a different problem, so most projects should include some mix of all three types.*

<https://www.ybrikman.com/writing/2014/05/05/you-are-what-you-document/>

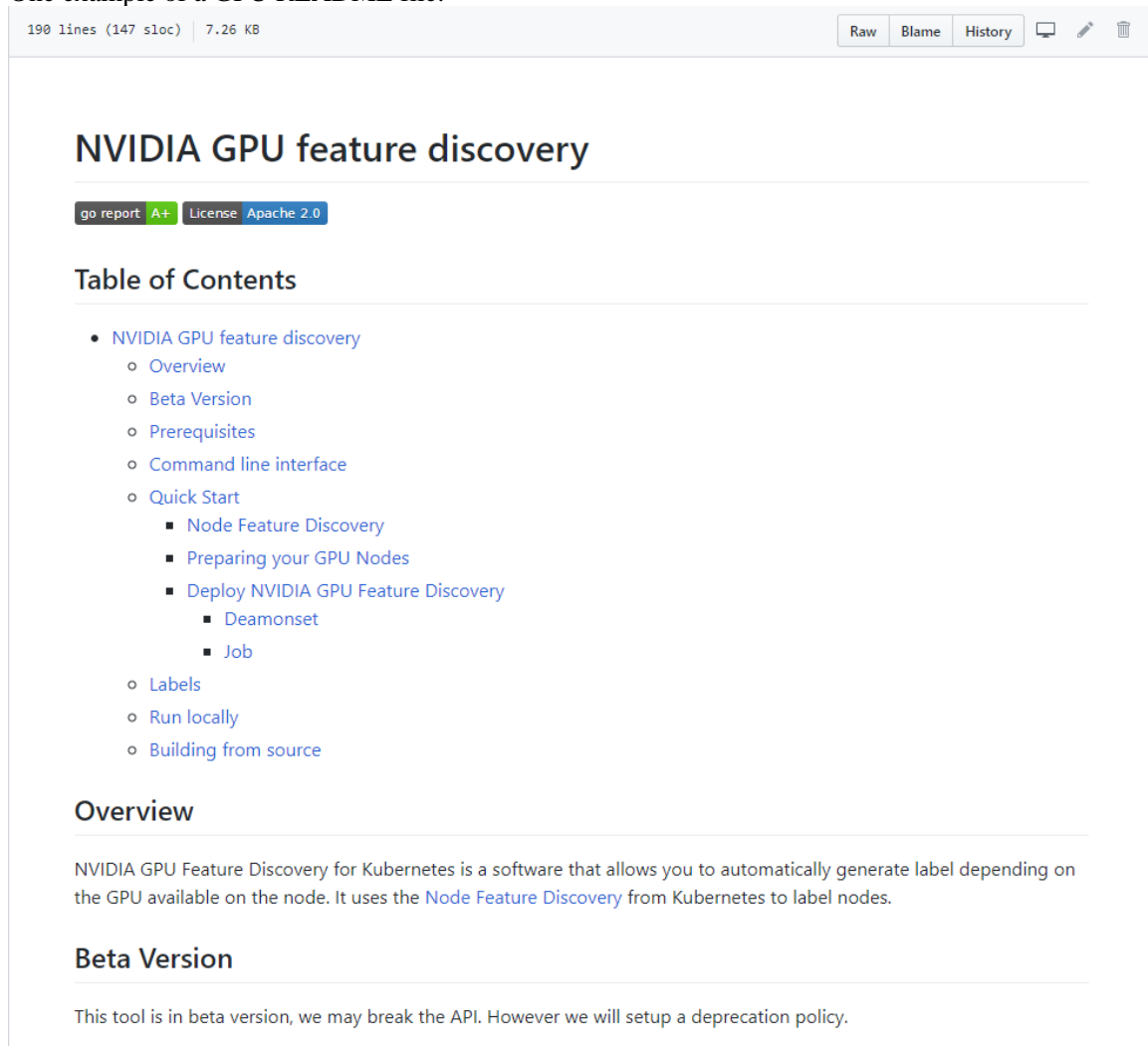
In this project, I will give some examples for those three different types of documents of GPU as the coprocessors that enhances the core functionality of a CPU.

Written documentation

Written documentation is usually the most straight forward documentation that we feel, when we mention “software documentation”. For some examples of written documentation, I choose README, reference guides (Manual).

README is more like a brief introduction of that software. It basically includes Description, Quick examples, Quick start, Further documentation, Project organization, Legal notices as the definition of *You are what you document*.

One example of a GPU README file.



The screenshot shows the GitHub interface for the 'NVIDIA GPU feature discovery' repository. At the top, it indicates '190 lines (147 sloc)' and '7.26 KB'. Below the repository name, there are buttons for 'go report', 'A+', 'License', and 'Apache 2.0'. The 'Table of Contents' section lists the following items:

- NVIDIA GPU feature discovery
 - Overview
 - Beta Version
 - Prerequisites
 - Command line interface
 - Quick Start
 - Node Feature Discovery
 - Preparing your GPU Nodes
 - Deploy NVIDIA GPU Feature Discovery
 - Daemonset
 - Job
 - Labels
 - Run locally
 - Building from source

The 'Overview' section states: 'NVIDIA GPU Feature Discovery for Kubernetes is a software that allows you to automatically generate label depending on the GPU available on the node. It uses the [Node Feature Discovery](#) from Kubernetes to label nodes.'

The 'Beta Version' section states: 'This tool is in beta version, we may break the API. However we will setup a deprecation policy.'

<https://github.com/NVIDIA/gpu-feature-discovery/blob/master/README.md>

Other example of a NVIDIA GPU README files

https://download.nvidia.com/XFree86/Linux-x86_64/165.33.09/README/

<https://github.com/NVIDIA/gpu-operator/blob/master/README.md>

Then as reference guides, I think it more likes a detailed of a manual of the software or hardware. It contains all the topics and the users can use this reference guides to find all the information of the software (hardware) and can navigate the functionality, implement training, troubleshooting as well. One simple reference guide of NVIDIA virtual GPU

Chapter 1. INTRODUCTION TO NVIDIA vGPU SOFTWARE

NVIDIA vGPU software is a graphics virtualization platform that provides virtual machines (VMs) access to NVIDIA GPU technology.

1.1. How NVIDIA vGPU Software Is Used

NVIDIA vGPU software can be used in several ways.

1.1.1. NVIDIA vGPU

NVIDIA Virtual GPU (vGPU) enables multiple virtual machines (VMs) to have simultaneous, direct access to a single physical GPU, using the same NVIDIA graphics drivers that are deployed on non-virtualized operating systems. By doing this, NVIDIA vGPU provides VMs with unparalleled graphics performance, compute performance, and application compatibility, together with the cost-effectiveness and scalability brought about by sharing a GPU among multiple workloads.

For more information, see [Installing and Configuring NVIDIA Virtual GPU Manager](#).

1.1.2. GPU Pass-Through

<https://docs.nvidia.com/grid/latest/pdf/grid-vgpu-user-guide.pdf>

Other examples of NVIDIA GPU.

https://download.nvidia.com/XFree86/Linux-x86_64/165.33.09/README/

<https://docs.nvidia.com/grid/latest/>

Code Documentation

Code Documentation is more like a core documentation of a software, it is more self-documented as the software itself. But not only the core code of a software can be regarded as code documentation.

Any code that can help the programmers, testers or debuggers to navigate their demands, can be traded as Code Documentation.

I will choose code comments and testing cases as examples of Code Documentation.

Code Comments

Code comment is an interpretation of the real code by using human-languages. It can help the developer itself or other programmers to understand how the specific function works and what is the logic behinds.

One example of the code comment for a benchmark function in GPU development.

Branch: master ▾ DeepBench / code / nvidia / rnn_bench.cu Find file Copy path

Sharan Narang Fix RNN benchmark to measure backprop wrt params. 4b3781e on May 16, 2018

0 contributors

487 lines (395 sloc) | 19 KB Raw Blame History

```
1  #include <chrono>
2  #include <iomanip>
3  #include <memory>
4  #include <stdexcept>
5  #include <tuple>
6
7  #include <cuda.h>
8  #include <curand.h>
9
10 #include <thrust/device_ptr.h>
11 #include <thrust/fill.h>
12
13 #include "tensor.h"
14 #include "cudnn_helper.h"
15 #include "rnn_problems.h"
16
17 /*
18  Usage:
19
20  The default precision is set based on the architecture and mode.
21
22  By default, the program runs the benchmark in training mode.
23
24  bin/rnn_bench
25
26  To run inference mode, use the following command:
27
28  bin/rnn_bench inference
29
30
31  To change the precision for training/inference, use:
32
33  bin/rnn_bench train <precision>
34  bin/rnn_bench inference <precision>
35
36  Supported precision types:
37
38  For Maxwell GPUS:
39  float for training and inference
40
41  For Pascal GPUS:
42  float, half for training
43  float, half, int8 for inference
```

https://github.com/baidu-research/DeepBench/blob/master/code/nvidia/rnn_bench.cu

Other examples of code comments

<https://github.com/tbennun/mgbench/blob/master/src/L0/devinfo.cpp>

Test cases

Test case is a special code that can demonstrate the expected functionality of code. It can assist the developers to quickly detect the error or bugs especially if it is a huge project.

One example of a test file in GPU development.

Kernel Code - test.cl

The kernel code is generated from the unmodified source code of the tested C program. The tool turns the C source into OpenCL code, by turning the `main` function into an OpenCL kernel function. It also modifies the way inputs are being read, by replacing references to `argv` and standard input with references to the auto-generated **input** structure.

To illustrate this, consider the resulting OpenCL code for **add.c**:

```
1 int add(int a, int b) {  
2     return a + b;  
3 }  
4  
5 __kernel void main_kernel(  
6     __global struct input* inputs,
```

<https://wyaneva.github.io/papers/thesis2016.pdf>

Another example of a test file

```
#define ARRAY_SIZE 9

void quickSort(int[], int, int);
int partition(int[], int, int);

int main(void) {
    // sample input array as test case
    int a[] = { 7,12,1,-2,0,15,4,11,9 };
    quickSort(a, 0, ARRAY_SIZE-1);
    return 0;
}
```

Figure 2: Harness for testing Quicksort with one test input

```
#define ARRAY_SIZE 9
#define NUM_TESTS 256

__device__ void quickSort(int[], int, int);
__device__ int partition(int[], int, int);

// GPU function
__global__ void compute(int *tests) {
    // The thread ID identifies the test case
    int test_case = threadIdx.x*ARRAY_SIZE;
    quickSort(tests+test_case, 0, ARRAY_SIZE-1);
}

int main(void) {
    int host_inputs[NUM_TESTS][ARRAY_SIZE] = { ... };
    int *device_inputs;

    cudaMalloc((void **)&device_inputs, sizeof(host_inputs));
    cudaMemcpy(device_inputs, host_inputs,
               sizeof(host_inputs), cudaMemcpyHostToDevice);

    // Number of blocks is 1 and number of threads per block is 256.
    compute<<<1, NUM_TESTS>>>>(device_inputs);
    return 0;
}
```

Figure 3: CUDA test harness for Quicksort with 256 tests

<http://www.kroening.com/papers/ase2014.pdf>

Other examples

<http://www.prace-ri.eu/IMG/pdf/wp67.pdf>

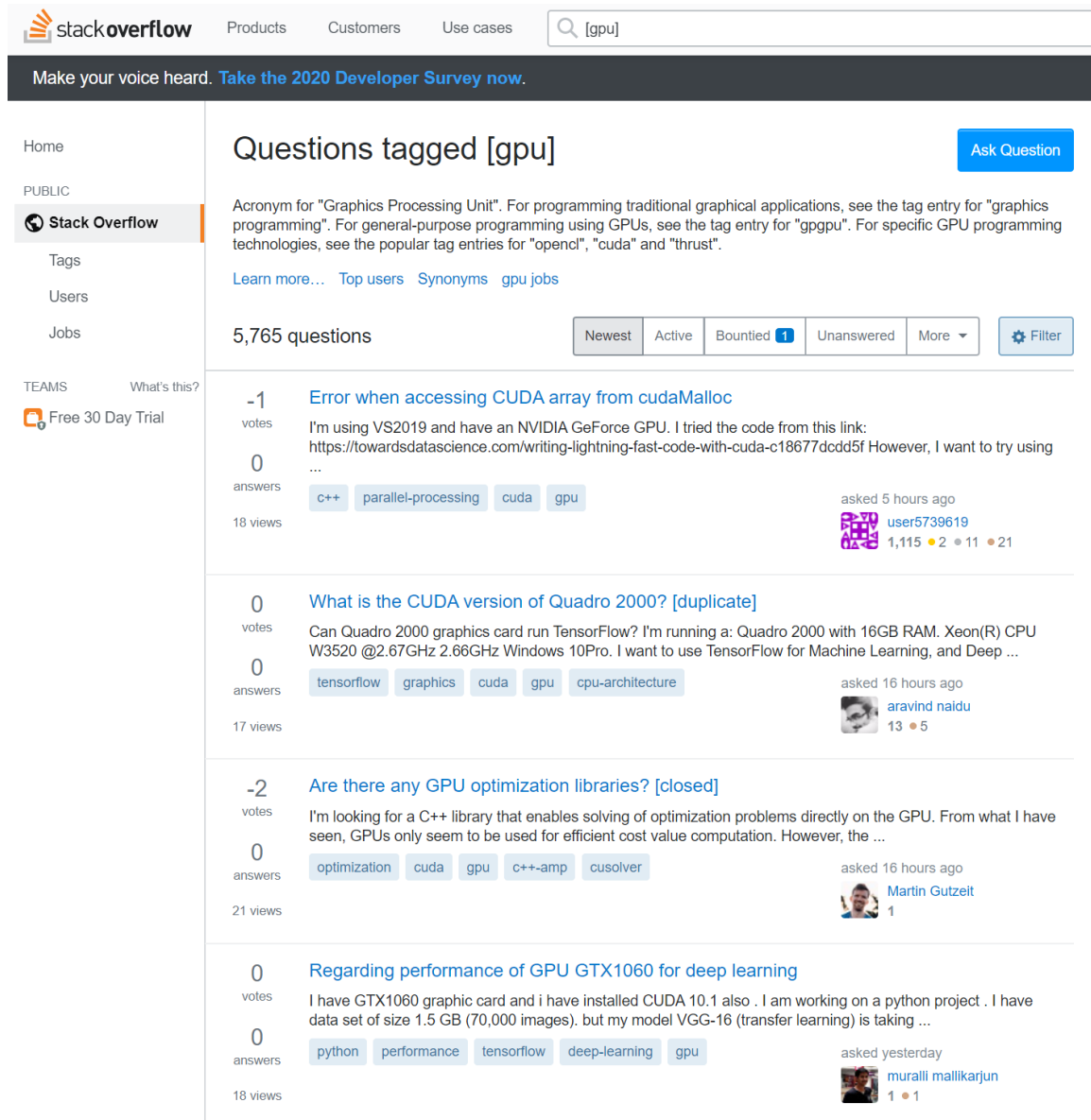
Community documentation

This is also an important documentation for a software. It is used for the people who involved with the project discuss questions and take some help.

And here is an example of stack overflow, which is a website used for people post questions and share their idea, solutions etc.

Here is the website:

<https://stackoverflow.com/questions/tagged/gpu>



The screenshot displays the Stack Overflow interface for the 'gpu' tag. The top navigation bar includes the Stack Overflow logo, links to Products, Customers, and Use cases, and a search bar containing '[gpu]'. Below the navigation bar is a banner for the 2020 Developer Survey. The left sidebar shows the 'Stack Overflow' logo and links to Home, PUBLIC, Tags, Users, Jobs, TEAMS, and What's this? (Free 30 Day Trial). The main content area is titled 'Questions tagged [gpu]' and features an 'Ask Question' button. Below the title is a description of the 'gpu' tag and a list of related tags: 'c++', 'parallel-processing', 'cuda', and 'gpu'. The questions are listed in a table with columns for votes, answers, views, question text, tags, and user information. The first question is 'Error when accessing CUDA array from cudaMalloc' with -1 votes, 0 answers, and 18 views. The second question is 'What is the CUDA version of Quadro 2000? [duplicate]' with 0 votes, 0 answers, and 17 views. The third question is 'Are there any GPU optimization libraries? [closed]' with -2 votes, 0 answers, and 21 views. The fourth question is 'Regarding performance of GPU GTX1060 for deep learning' with 0 votes, 0 answers, and 18 views.

stackoverflow Products Customers Use cases [gpu]

Make your voice heard. [Take the 2020 Developer Survey now.](#)

Home

PUBLIC

Stack Overflow

Tags

Users

Jobs

TEAMS What's this?

Free 30 Day Trial

Questions tagged [gpu]

[Ask Question](#)

Acronym for "Graphics Processing Unit". For programming traditional graphical applications, see the tag entry for "graphics programming". For general-purpose programming using GPUs, see the tag entry for "gpgpu". For specific GPU programming technologies, see the popular tag entries for "opengl", "cuda" and "thrust".

[Learn more...](#) [Top users](#) [Synonyms](#) [gpu jobs](#)

5,765 questions

Newest Active Bountied 1 Unanswered More Filter

-1 votes
0 answers
18 views

Error when accessing CUDA array from cudaMalloc

I'm using VS2019 and have an NVIDIA GeForce GPU. I tried the code from this link: <https://towardsdatascience.com/writing-lightning-fast-code-with-cuda-c18677dcd5f> However, I want to try using ...

[c++](#) [parallel-processing](#) [cuda](#) [gpu](#)

asked 5 hours ago
 [user5739619](#)
1,115 ● 2 ● 11 ● 21

0 votes
0 answers
17 views

What is the CUDA version of Quadro 2000? [duplicate]

Can Quadro 2000 graphics card run TensorFlow? I'm running a: Quadro 2000 with 16GB RAM. Xeon(R) CPU W3520 @2.67GHz 2.66GHz Windows 10Pro. I want to use TensorFlow for Machine Learning, and Deep ...

[tensorflow](#) [graphics](#) [cuda](#) [gpu](#) [cpu-architecture](#)

asked 16 hours ago
 [aravind naidu](#)
13 ● 5

-2 votes
0 answers
21 views

Are there any GPU optimization libraries? [closed]

I'm looking for a C++ library that enables solving of optimization problems directly on the GPU. From what I have seen, GPUs only seem to be used for efficient cost value computation. However, the ...

[optimization](#) [cuda](#) [gpu](#) [c++-amp](#) [cusolver](#)

asked 16 hours ago
 [Martin Gutzelt](#)
1

0 votes
0 answers
18 views

Regarding performance of GPU GTX1060 for deep learning

I have GTX1060 graphic card and i have installed CUDA 10.1 also . I am working on a python project . I have data set of size 1.5 GB (70,000 images). but my model VGG-16 (transfer learning) is taking ...

[python](#) [performance](#) [tensorflow](#) [deep-learning](#) [gpu](#)

asked yesterday
 [muralli mallikarjun](#)
1 ● 1

Another example for stack overflow

<https://discourse.julialang.org/t/stack-overflow-on-cuda/10735>