業界最安値の GPUaaS 「GPUSOROBAN」 を実際に試してみた





業界最安値の GPUaaS 「GPUSOROBAN」を実際に試してみた

発行日:2022年6月27日

【概要】

本資料は最新の GPU が使える上に、とても安いと噂の「GPUSOROBAN」がどんな感じのものなのか、 CUDA 初心者が 3 日間の無償評価で会員登録から CUDA のサンプルを動かすまでのレポートです。

「GPUSOROBAN」は株式会社ハイレゾが運営する AI 開発や高解像度映像のレンダリング、交通イン フラや医療などのシミュレーションなどに活用される GPU リソースを、低コスト・定額料金で利用で きる業界最安値の GPU クラウドサービスです。

本資料はインスタンスを使い始める入り口までですが、今回は CUDA が使える状態まで進めておき、 次回の資料では何か動かしてみたいと思います。

Windows でのインスタンスの作成から接続までの手順は「GPUSOROBAN」公式 Web サイトの「SOROBAN で仮想インスタンスを作成するには〜WINDOWS10の場合〜」にも掲載されています。

【目次】

1.	利用前の確認	2
2.	会員登録	2
3.	インスタンスの作成	5
4.	インスタンスの起動	7
5.	インスタンスへの接続と環境の確認	8
6.	CUDA のサンプルでベンチマークを実行	12
7.	まとめ	14
改版履	夏歴	14



1. 利用前の確認

弊社内の「GPUSOROBAN」担当と連絡を取り、無償評価をしたいことを伝えました。インスタン スにインストールされている CUDA のバージョンが分からなかったので、それも聞いてみたのです が、そこで Docker についての説明もありました。

なお、無償評価については、以下問合せ画面から問合せていただくとスムーズかと思います。 URL:https://www.paltek.co.jp/solution/gpucloud/index.html

◆サマリ

- ・評価開始希望日を伝えると、その日から3日間が無償評価期間になる
- ・会員登録はいつでもできる
- ・インスタンスの作成は評価開始日から可能になる
- ・CUDA のバージョンは 11.1
- ・Docker を使うには仮想インスタンスではなく、ベアメタルインスタンスでの提供になる

現在の私の開発環境は、CUDA が 11.4、環境は Docker で作っています。Docker を使いたかったの でベアメタルインスタンスを相談中です。ベアメタルインスタンスは仮想インスタンスのように即 座に使用可能とはいかないので仮想インスタンスで進めます。

2. 会員登録

① サイトヘアクセス

「GPUSOROBAN」のWebサイトにアクセスして「新規申し込み」をクリックします。

URL : https://soroban.highreso.jp/



① 会員登録

会員登録画面が表示されますので、名前、メールアドレス、パスワードを入力し、利用規約に 同意をチェックして「登録」ボタンを押します。

次に、登録完了の表示に切り替わり、登録したメールアドレスに登録認証メールが送信されま す。

最後に届いたメールに記載されている URL をクリックすると登録完了です。

	SOROBAN
	登録完了
云貝豆琢 お名前個人名、会社名、回体名等):	登録メールアドレスに確認用メールを送りました。 メールの内容に従い、メールアドレスの確認を行ってください。 既に会員の方はごちら 会員ログイン
お名前を入力してください	メール:SOROBAN 会員登録のご確認
メールアドレス: メールアドレス パスワード:	SOROBAN をご利用いただきありがとうございます。
パスワード パスワード確認用): パスワード確認用	会員登録を申込みいただきありがとうございます。 URL をタップしてメールアドレスの確認を
利用規約に同意します。	完了してください。 https://gpu-console.highreso.jp/auth/confirm/xxxxxxxxxxxxxxxxxxxxxxxxxx
登録	URL の有効期限は 30 分です。
既に会員の方はこちら 会員ログイン	SOROBAN サポート https://gpu-console.highreso.jp/



② ログイン

下記の URL よりログインしてみましょう。 ログイン URL

https://gpu-console.highreso.jp/auth/login

SOROBAN						
ログイン						
/0.70-1F:						

ロジイン						
バスワードをお忘れの場合 再設定 アカウントをお持ちで新い場合 会員登録						

ログインするとコントロールパネルが表示されます。

= SOROBAN									
П номе	お知らせ								
🖵 インスタンス作成 🔷 🔇	5月31日 法定点検による	3データセンター停留	Mのお知らせ						
© ssн≠—	12月14日 年末年始サポ	ート休業のお知らせ ート休業のお知らせ							
☆ 会員情報						-覧へ→			
利用状況/料会プラン変更	会員インスタンス								
請求明細(確定)	起動されていません。								
三支払い機関 インスタンス22000000									
□ サ ポート 〈	※ 各インスタンスの料金プランの変更は利用状況/料金プラン変更へ④								
[→ ログアウト	インスタンス名	料金プラン	タイプ	配動	削除	RIN			

3. インスタンスの作成

ここからはインスタンスを作成していきます。

今回は、ウィザードタ形式で進められる「クイック・スタート」で行います。

左のメニューバーより「インスタンス作成」、「クイック・スタート」の順にクリックし、作成するインスタンス名を入力して「NEXT」ボタンをクリックします。

🔓 номе	クイック・スタート・	ステッパー					
🖵 インスタンス作成 🛛 🗸	(1) —	2	3	(4)	5		
00 クイック・スタート	インスタンス名	インスタンスタイプ	SSH秘密鍵	課金タイプ	確認		
🖵 カスタム構成	クイック	ウスタート機	能				
₿ ssн≠—	こちらはインスター	ンス作成のクイックスタート版 範略化されていますが、選択で	。 「たなります。 「きろインスタンスタイプ(こ制限があります.			
○ 会員情報	より多くのインスタンスタイプから選択したい場合は、カスタム構成を選択してください。						
利用状況/料金プラン変更	1. インスタンス名						
請求明細(確定)	インスタンス名を 半角小文字英数の	入力してください。 み、4文字以上、32文字以下で	入力してください。				
注 支払い履歴	test01						
□ サポート <	NEXT						
[→ ログアウト							

② 後々機械学習をやってみたいので、インスタンスに nv4-1dl を選択します。

クイック・スタート	・ステッパー			
 — 	2	3	- (4)	5
インスタンス名	インスタンスタイプ	SSH秘密鍵	課金タイプ	確認
2. インス・ 以下の表からイン	タンスタイプの ッスタンスタイプを決定します	?定		
	タイプ名	用途		
選択	nvd4-1dl	AI・ディープラーニング		
選択	nvd4-1pr	レンダリング		
選択	nvd4-1ub	その他		
PREV				

インスタンスにアクセスするための"SSH 秘密鍵の生成"を行います。
 ここは「NEXT」をクリックして進めるだけで、OK です。



業界最安値の GPUaaS 「GPUSOROBAN」を実際に試してみた

クイック・スタート・ステッパー							
インスタンス名		3 SSH秘密鍵	4 課金タイプ	5 確認			
3. SSH秘密	密鍵の生成						

④ 今回は無償試用なので「従量」のまま「NEXT」をクリックします。

クイック・スタート・ス	テッパー			
く インスタンス名			4 課金タイプ	5 確認
4. 課金タイン 以下から課金タイプを	プの選択 を選択してください。			
● 従量 月額 PREV NEXT	🔵 1年契約 🛛 2年契約			
従量プラン(使用した時間に応じ) まずはインスタンスの	こついて こ料金を請求させていただき D操作を体験してみたい、性	ます。 能を確認したいなどの場	合にこちらの料金プランをお道	難びください。

⑤ 確認画面が表示されますので、内容に問題のないことを確認して「CREATE」をクリックしま す。従量に金額が表示されていますが、試用中は課金されないのでご安心ください。

				5
2入92入省	1 289 2891 2	いるでは	詠並ウイノ	191.20
5.確認				
インスタンス名	test01			
タイプ	nvd4-1dl			
SSH丰一名	defaultinskey			
従量	361円/h			
作成の確認	STEP 5			
よろしけたいちょうちみて	EIボタンをタップしてくださ	ちし。		
PREV CREATE	-,			

以上の設定でインスタンスを作成できました。

4. インスタンスの起動

作成したインスタンスは「会員インスタンス」に表示されます。

① 「会員インスタンス」より「起動」をクリックしてインスタンスを起動します。

会員インスタンス							
インスタンスへのアクセス方	「法⊖ 利用状況/	料金プラン変	更へ⊖				
	インスタンス名	料金プラン	タイプ	+-	外部ストレージ	IPアドレス	
	test01	従量	nvd4-1dl	defaultinskey		停止中	
※ インスタンス作成には状況	※ インスタンス作成には状況により10数分ほどの時間がかかる場合があります。						

② 課金に関する確認のダイアログが表示されますのでお読みいただき、「内容を確認しました」を チェックして、「起動する」をクリックします。

母インスタンス情報	
基本情報 アタッチ デタッチ	
基本情	報
インスタンス名:	test01
インスタンスタイプ:	nvd4-1dl
アタッチ済ストレージ:	なし
従量課金に関	する確認
 び置調金ブランでは、インスタンス起動後、接続 よろしければ内容を確認しました)をチェックし () 病容を確認しました () 病面にない () 病面です () 前面です () 病面です () 前面です () 前面です	町能伏観になった時点で課金開始されます。 て(起動する)ボタンをタップしてください。

③ 「IP アドレス」欄が「停止中」から「Pending」に変わりました。





インスタンスを作成後、ちょっと離席してから起動しました。その間にインスタンスの作成は 終わっていたようで、15秒程度で起動して「IPアドレス」欄に IP アドレスが表示されました。

会員インスタンス							
インスタンスへのアクセス方法 ④ 利用状況/料金プラン変更へ ④							
	インスタンス名	料金プラン	タイプ	+-	外部ストレージ	IPアドレス	
前除 🕕 停止	test01	従量	nvd4-1dl	defaultinskey		10.233.122.16	
※ インスタンス作成には	状況により10数分(まどの時間がた	いかる場合カ	べあります。			
起動リクエスト完了							
ご利用ありがとうございます。 インスタンス起動リクエストが完了しました。 起動完了までしばらくおまちください。							

5. インスタンスへの接続と環境の確認

「会員インスタンス」の下の方に OS 別の「接続方法」が記載されています。今回は Linux から接続してみます。

接続方法
1. アクセスサーバ用秘密鍵の保存 以下のボタンを押してアクセスサーバ用の秘密鍵「ackey.txt」をダウンロードします。
アクセス織ダウンロード
2. インスタンス用秘密鍵の保存 以下のボタンを押してインスタンス用の秘密鍵「mykey.txt」をダウンロードします。
※インスタンス作成後に「インスタンス鍵ダウンロード」ボタンが有効化されない場合は、ブラウザをリロードしてくださ い。
キー名:defaultinskey インスタンス編ダウンロード
3. インスタンス接続用コマンド OSごとのインスタンス接続コマンド例を示します。
WINDOWS MAC LINUX
LINUX OSの場合

 「アクセス鍵ダウンロード」、「インスタンス鍵ダウンロード」を順にクリックして、接続に使用 する Linux PC に秘密鍵をダウンロードします。



今回は、ユーザのホームディレクトリ下の.ssh内にダウンロードして、パーミッションを変更しました。

kirin@xeon-E3:~\$ Is -la ~/.ssh/
合計 16
drwxrwxr-x 2 kirin kirin 4096 6月 16 10:36 .
drwxr-xr-x 7 kirin kirin 4096 6月 16 10:28
-rwxrr 1 kirin kirin 3243 6月 16 10:36 ackey.txt
-rwxrr 1 kirin kirin 3243 6月 16 10:36 mykey.txt
kirin@xeon-E3: [*] \$ chmod 600 [*] /.ssh/ackey.txt [*] /.ssh/mykey.txt

インスタンスへの接続経路は以下のようになっています。



図1インスタンスへの接続経路

インスタンスに接続するにはアクセスサーバを経由する必要があり、インスタンスに直接 ssh でログインすることができません。

「<u>インスタンスへのアクセス方法」</u>を参考に ssh のポートフォワーディングを使ってアクセス サーバと ssh で接続し、インスタンスのポート 22(ssh 用ポート)をローカルの Linux PC のポー ト 20122 にフォワードします。

※上記リンク先は会員登録をしていないと開けません

③ インスタンスの IP アドレスは、インスタンスを再起動するたびに変化しますので、環境変数に 割り当てました。では、ssh でトンネルを掘ります。ssh の鍵はアクセスサーバ用秘密鍵を使用 します。

kirin@xeon-E3:[~]\$ export VM_ADR=10.233.122.16 # インスタンスの IP アドレス kirin@xeon-E3:[~]\$ export AC_ADR=202.122.50.154 # アクセスサーバの IP アドレス



kirin@xeon-E3:~\$ export CONNECT_PORT=20122 # インスタンスの ssh ポートを割り当てるローカル PC のポート kirin@xeon-E3:~\$ ssh -L \$CONNECT_PORT:\$VM_ADR:22 -I user \$AC_ADR -p 30022 -i ~/.ssh/ackey.txt Highreso GPU Advance. This is access server.

これでローカルの Linux PC からインスタンスに ssh で接続できるようになりました。ローカル PC のポート 20122 にアクセスするとインスタンスのポート 22 にフォワードされます。このタ ーミナルは、このままにしておきます。

④ インスタンスに ssh でログインします。ssh の鍵はインスタンス用秘密鍵を使用します。

インスタンスに作られているアカウントは「user」です。

kirin@xeon-E3:~\$ ssh user@localhost -p 20122 -i ~/.ssh/mykey.txt
The authenticity of host '[localhost]:20122 ([127.0.0.1]:20122)' can't be established.
ECDSA key fingerprint is SHA256:FDO/Kab5WuIJF44iXFiVfdeaNz18XMOT2nH5m8FuA+k.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[localhost]:20122' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-167-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
The programs included with the Ubuntu system are free software:
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

applicable law.

ログインに成功しました。



※下記のエラーが発生した場合はインスタンスの IP アドレスの指定に間違いがないかご確認く ださい。

kirin@xeon-E3:~\$ ssh user@localhost -p 20122 -i ~/.ssh/mykey.txt ssh_exchange_identification: read: Connection reset by peer

ssh トンネルを掘ったターミナルには下記のエラーが表示されている

channel 3: open failed: connect failed: Host is unreachable

CUDA 関連の情報を見てみます。窓口から聞いた通り 11.1 ですね。

(base) user@test01:~\$ nvidia-smi		
Thu Jun 16 02:44:21 2022		
+ NVIDIA-SMI 455.23.05 Driver	Version: 455.23.05	CUDA Version: 11.1
GPU Name Persistence-M Fan Temp Perf Pwr:Usage/Cap 	' Bus-Id Disp.A Memory-Usage 	Volatile Uncorr. ECC GPU-Util Compute M. MIG M.
0 A100-PCIE-40GB Off N/A 44C PO 38W / 250W 	00000000:C1:00.0 Off OMiB / 40536MiB 	0 0% Default Disabled
+ Processes: GPU GI CI PID Ty ID ID	pe Process name	GPU Memory Usage
 No running processes found		
(base) user@test01:~ \$ cat /proc/ NVRM version: NVIDIA UNIX x86_64 GCC version: gcc version 7.5.0 (base) user@test01:~ \$ cat /usr/I CUDA Version 11.0.228 (base) user@test01:~ \$ nvcc -V nvcc: NVIDIA (R) Cuda compiler d	driver/nvidia/version Kernel Module 455.23. (Ubuntu 7.5.0-3ubuntu1 [°] ocal/cuda/version.txt	. <mark>05</mark> Fri Sep 18 19:37:1 ~18.04)

Built on Wed_Jul_22_19:09:09_PDT_2020 Cuda compilation tools, release 11.0, V11.0.221 Build cuda_11.0_bu.TC445_37.28845127_0

他にも気になる gcc、Python、Anaconda、Docker の状態を見てみます。

(base) user@test01:~\$ gcc --version
gcc (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0
Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: see the source for copying conditions. There is NO
warranty: not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
(base) user@test01:~\$ python -V
Python 3.7.6
(base) user@test01:~\$ python3 -V
Python 3.7.6
(base) user@test01:~\$ pip list | egrep "(tensor|cv)"
(base) user@test01:~\$
(base) user@t

事前に聞いていた通り Docker はないですね。Docker はインストールしても使えないそうです。

6. CUDA のサンプルでベンチマークを実行

CUDA のサンプルをビルドします。

```
サンプルのソースは/usr/local/cuda-11.0 にあるのでこちらを使用します。
```

```
(base) user@test01:~$ cd /usr/local/cuda-11.0/bin
(base) user@test01:/usr/local/cuda-11.0/bin$ ./cuda-install-samples-11.0.sh ~/cuda_sample
Copying samples to /home/user/cuda_sample/NVIDIA_CUDA-11.0_Samples now...
Finished copying samples.
(base) user@test01:/usr/local/cuda-11.0/bin$ cd ~/cuda_sample/NVIDIA_CUDA-11.0_Samples/
(base) user@test01:~/cuda_sample/NVIDIA_CUDA-11.0_Samples$ time make -j16
make[1]: ディレクトリ '/home/user/cuda_sample/NVIDIA_CUDA-11.0_Samples/0_Simple/cpp0verload' に入りま
す
: 省略
Finished building CUDA samples
```

無事、ビルドに成功しました。



サンプルのベンチマークを実行して動作確認をします。今回は nbody を使用し、パラメータの numbodies は 4194304 にします。

(base) user@test01:~\$ cc	l ~/cuda_sample/NVIDIA_CUDA-11.0_Samples				
(base) user@test01:~/cuda_sample/NVIDIA_CUDA-11.0_Samples\$./bin/x86_64/linux/release/nbody ¥					
-benchmark -numbodies=4194304					
Run "nbody -benchmark [-numbodies= <numbodies>]" to measure performance.</numbodies>					
-fullscreen	(run n-body simulation in fullscreen mode)				
-fp64	(use double precision floating point values for simulation)				
-hostmem	(stores simulation data in host memory)				
-benchmark	(run benchmark to measure performance)				
-numbodies= <n></n>	(number of bodies (>= 1) to run in simulation)				
-device= <d></d>	(where d=0,1,2 for the CUDA device to use)				
-numdevices= <i></i>	(where i=(number of CUDA devices $>$ 0) to use for simulation)				
-compare	(compares simulation results running once on the default GPU and once on the				
CPU)					
-cpu	(run n-body simulation on the CPU)				
-tipsy= <file.bir< td=""><td>n> (load a tipsy model file for simulation)</td></file.bir<>	n> (load a tipsy model file for simulation)				
NOTE: The CUDA Samples are not meant for performance measurements. Results may vary when GPU Boost is enabled.					
> Windowed mode					
Simulation data stored in video memory					
<pre>> Single precision floating point simulation</pre>					
> 1 Devices used for simulation					
GPU Device 0: "Ampere" with compute capability 8.0					
1					
> Compute 8.0 CUDA device: [A100-PCIE-40GB]					
number of bodies = 4194304					
4194304 bodies, total time for 10 iterations: 541977.688 ms					
= 324.592 billion interactions per second					
= 6491.849 single-precision GFLOP/s at 20 flops per interaction					
10分近くかかりました。					

A100、GTX 1080Ti、GTX 745 で同じベンチマー クを走らせてみました。A100 の得意なところを引 き出せるベンチマークではないのかなという印象で す。ベンチマークはよく考えなければいけません ね。





7. まとめ

会員登録の開始から5分位でインスタンスを起動できました。

起動したインスタンスへの接続には踏み台(アクセスサーバ)を経由する必要があり、ローカルの PC、踏み台、インスタンスそれぞれのポートを整理するために絵を描いたり、インスタンスの IP アドレスを打ち間違えてエラーが発生したりして 10 分位かかってしまいました。そのようなことを しなければインスタンスへの接続は数分で終了すると思います。

CUDA のドライバや toolkit はインストールされていたので、CUDA のサンプルはビルドするだけ で動かすことができました。ベンチマークの実行中の待ちが 10 分位かかりましたが、それを除けば 5 分位で終了します。

登録開始からサンプルを動かすまで、特に急ぐことなく進めて 30 分あれば十分でした。Docker は 不要であれば、今回作成したインスタンスを使って始められそうです。

では、次回お会いしましょう。

改版履歴

Version	日付	改版内容
1.0	2022/6/27	・新規作成

Attention

1:書面での承諾なしに本書の一部、または全部を無断転用することを禁止致します。

- 2:本書は予告なしに変更する事があります。予めご了承下さい。
- 3:本書は万全を期して作成しておりますが、不明な点や誤り、記載もれ等がある可能性があります。 お知らせくだされば幸いです。
- 4:本書を利用した結果に関しては、上記の如何に関わらず責任を負いかねますので、その旨ご了承下さ います様お願い致します。
- 5:本資料は製品を利用する際の補助的なものとして書かれたものです。製品をご使用になる場合は メーカーがリリースしている最新版英語資料も合わせてご使用下さい。