

AWSとmyPrestoでの インシリコスクリーニング計算の実際

株式会社バイオモデリングリサーチ

中村寛則

2020年 10月29日

CBI学会2020年大会 企業セッション ES-13

内容:

- myPrestoについて
- myPrestoでのスクリーニング計算
- クラウドでの計算パターン1(PC単体を借りて計算)
- クラウドでの計算パターン2(PCクラスターを構築して計算)
- 計算時間、コスト
- クラウドでのGUIソフト利用例

myPresto5について



医薬品開発支援の 分子シミュレーションシステム

無償利用・商用利用可能

現在、国内30社以上で使われている。

経済産業省、N E D O 及びAMDEDからの委託プロジェクトの中で、開発された。

現在は、[次世代天然物化学技術研究組合](#)がAMED「次世代治療・診断実現のため創薬基盤技術開発事業 革新的中分子創薬技術の開発 中分子シミュレーション技術の開発」にて開発が継続中。

ライセンス管理:[次世代天然物化学技術研究組合](#)

<http://www.mypresto5.jp>

myPrestoの特長と成功例

特長:

- ドッキングシミュレーションと分子動力学シミュレーションの連携が容易
- 300万化合物のインシリコ・スクリーニング計算が容易
- 独自のスクリーニング手法(MTS, ML-MTS, DSI)
- 計算ツール群や化合物データベース等が充実
- 多様な計算機環境で実行可能(PC、スパコン、クラウド)

成功例:

- 合計11標的蛋白質に対して200以上の新規活性化合物を3～50%のヒット率で発見

myPresto5に含まれるプログラム/ データ

MD

cosgene_pack
psygene-G_pack
omegagene_pack
cosgene_sample_pack
SRPG_sample

Preparation

pdbcheck
tplgeneX
tplgeneL
Hgene

Prediction

LogS_predictor
SyntheticAccessibility

GUI

myPresto Potal

Screening

screening_pack
LigandBox
3DdataConstruction
MolSite TGS UAP
substructure_search
MVO_screening_pack

Docking

sievgene_pack
sievgene_M_pack
sievgene_DIAV
sievgene_NMR
sievgene_MVO

Etc

EditMol
LibMyPresto
Tools
doc

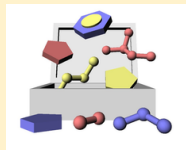
スクリーニング計算で使用するプログラムとデータ



screening_pack :

スクリーニング計算用スクリプト
(コマンド実行)

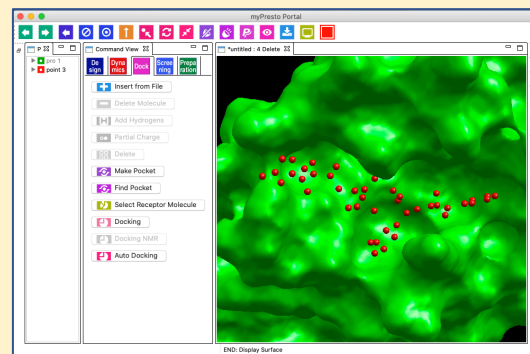
LigandBOX: 化合物ライブラリ



ナミキ商事、キシダ化学から
購入可能な化合物ライブラリ
化学反応性と分子サイズでふるいに
かけたものが提供されている。

ナミキ商事版 300万化合物(500-600万化合物から選出)
キシダ化学版 100万化合物(400-500万化合物から選出)

myPreto Portal: GUIプログラム



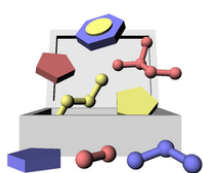
計算用入力ファイルの準備、
ドッキングポーズ観察で使用

myPrestoのプログラムとデータだけでスクリーニング計算が可能

myPrestoでのスクリーニングの流れの例

- (1) LigandBOXを入手
myPresto5のサイトからスクリーニング用LigandBOXをリクエストできる。
ダウンロードサイトについての通知が届くので、ダウンロードする
- (2) screening_packを入手screening_packの練習をする。
ターゲットタンパク質のサンプルデータが用意されている。
- (3) 入力ファイルの用意
myPresto Portal等で、ターゲットタンパク質の計算用ファイルを用意する。
- (4) screening_packで、本番のスクリーニング計算を行う。
- (5) 化合物の発注
- (6) 実験 → ヒット同定

myPrestoでのスクリーニング計算について



スクリーニング用LigandBOXでは、
1つのmol2ファイルに1万化合物が格納されている。

300万化合物スクリーニングでは、300本のジョブで計算する。

1万化合物の計算が2時間だとすると、合計600時間の計算時間が必要。

screening_packには2つのスクリプトセットがある。

screening_for_PC

PCで実行可能。ジョブスケジューラーを使用しない。

screening_for_server

サーバーでの実行用。ジョブスケジューラーを使用する。
サーバーのジョブスケジューラーに合わせてスクリプトを
変更する必要がある。

AWSでのスクリーニング計算実行方法は多様

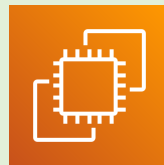


screening_for_PC

screening_for_server

X

AWSの関連サービス



Amazon EC2

計算機(インスタンス)を
借りるサービス

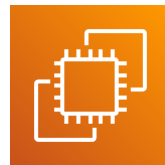


AWS Parallel Cluster

簡単にPCクラスターを構築可能
自動的にAmazon EC2で計算機
を調達

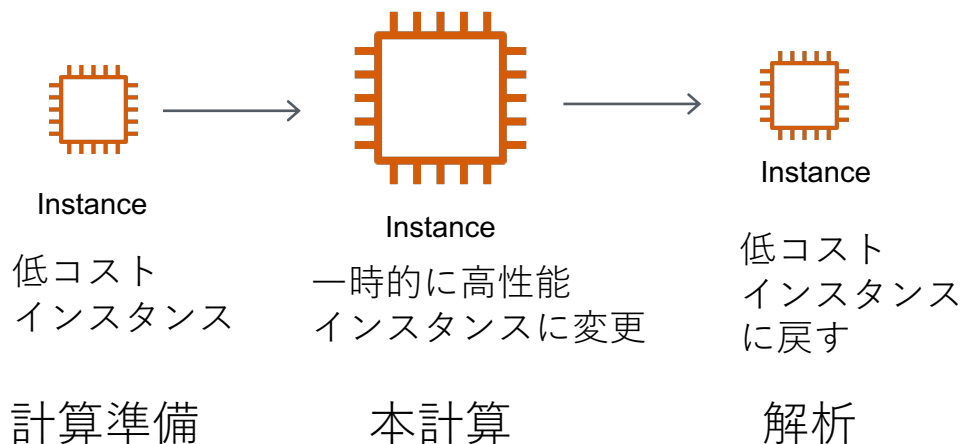
クラウドでの計算パターン1

screening_for_PCを使用すれば、スクリプトを変更せずに
マニュアル通りにスクリーニング計算を実行可能



Amazon EC2

必要な時にだけ、必要な性能で



ディスクの内容そのまま
インスタンスタイプ(CPU,
メモリ容量等)を変更可能

複数インスタンスで
計算を分割すれば、
短時間での計算も可能

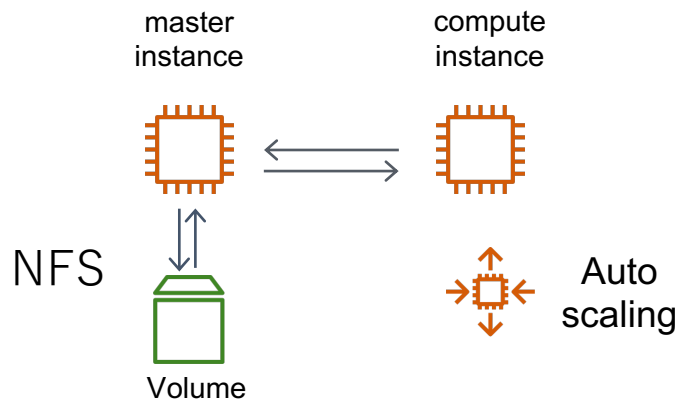
クラウドでの計算パターン2

screening_for_serverのスクリプトを変更すれば、AWS Parallel Clusterでスクリーニング計算を実行可能。マニュアル通りの手順で計算できる。



AWS Parallel Cluster

簡単にPCクラスターを構築可能



投入ジョブ本数、計算状況に応じて、計算インスタンスを自動的に増減してくれる。

ParallelClusterでのスクリーニング計算実行例(300万化合物のドッキング)

1 時間 3 時間 **12 時間** 1 日 3 日 1 週 カスタム ▾

線 ▾

アクション ▾



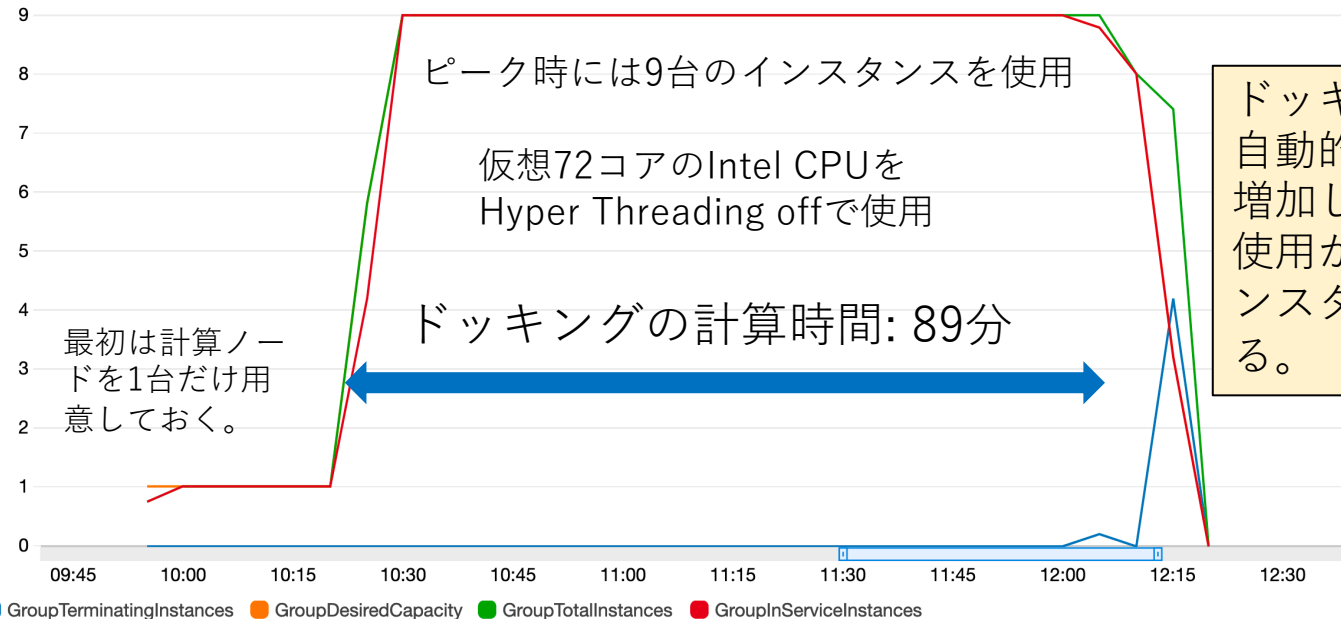
タイトルなしのグラフ

None

時間範囲の適用



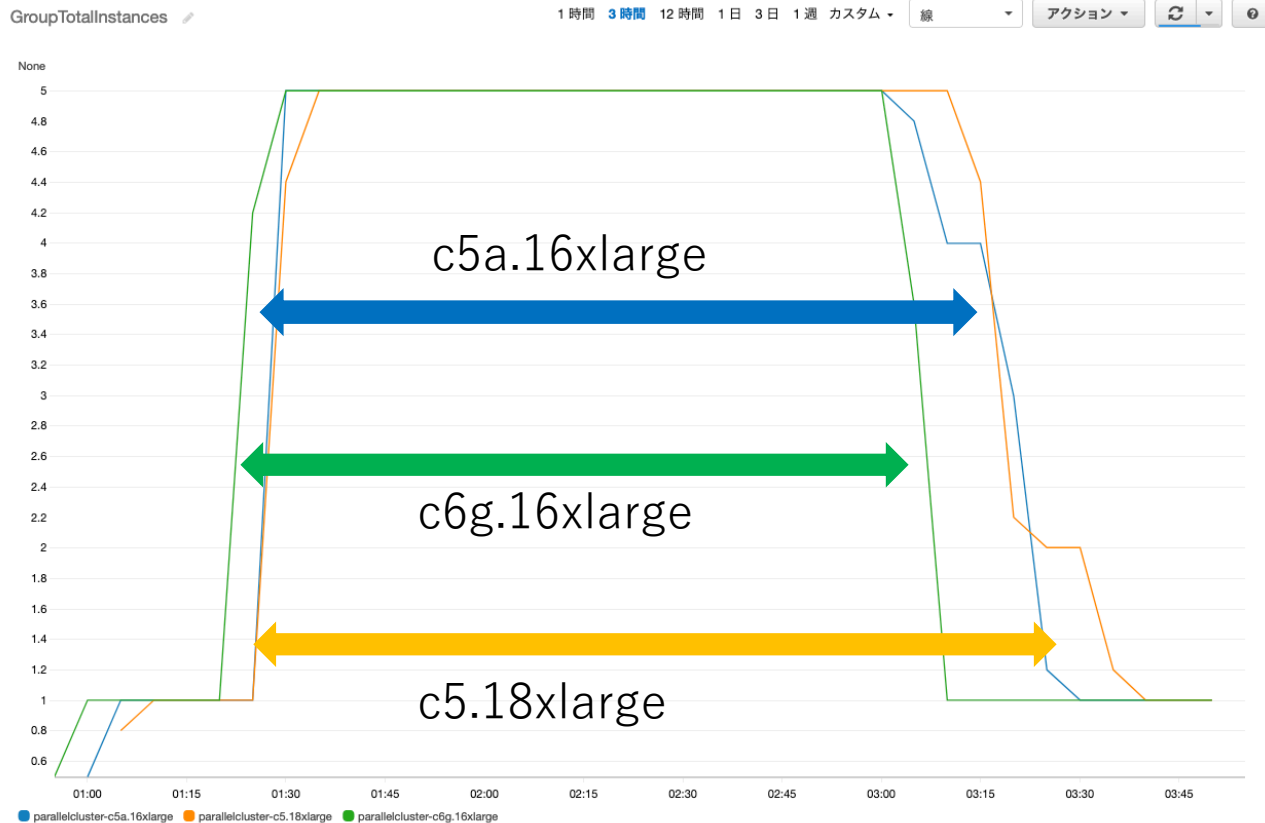
インスタンス数



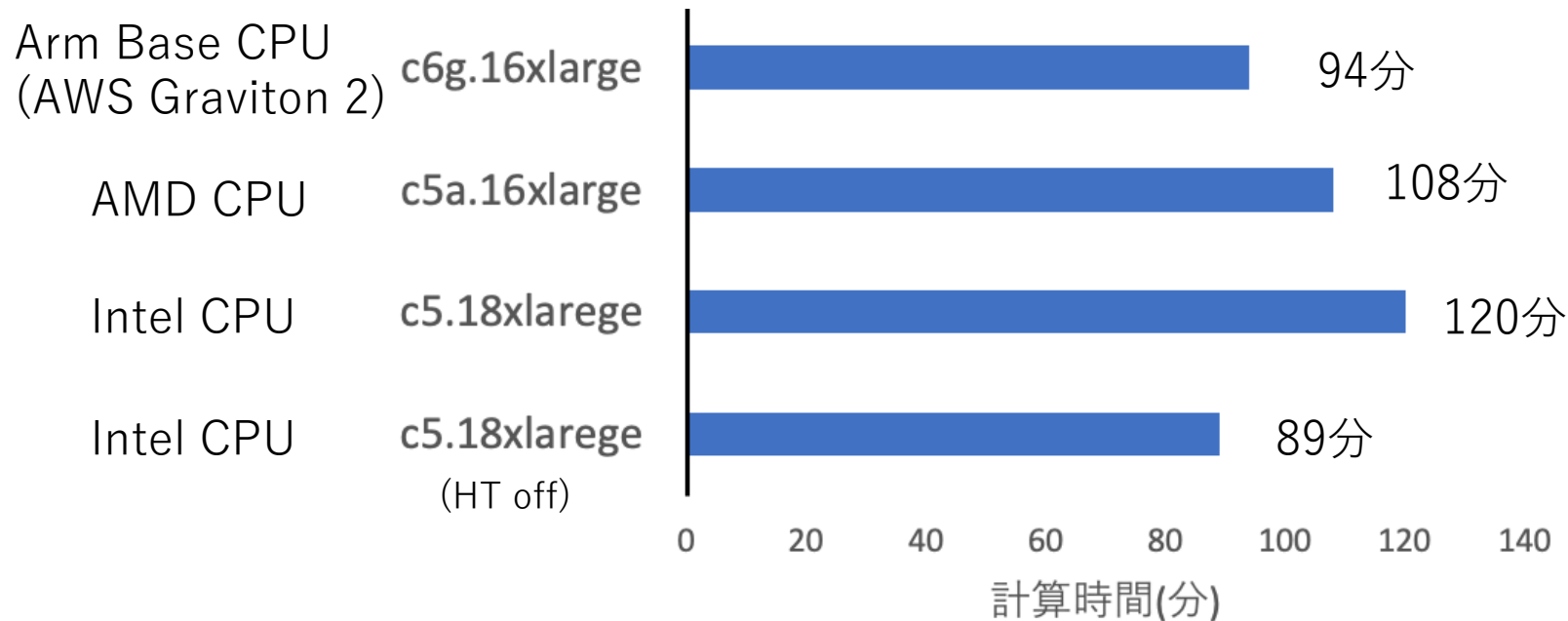
ドッキング
ジョブ投入

ドッキング
ジョブ終了

ParallelClusterを使用したスクリーニング計算の例

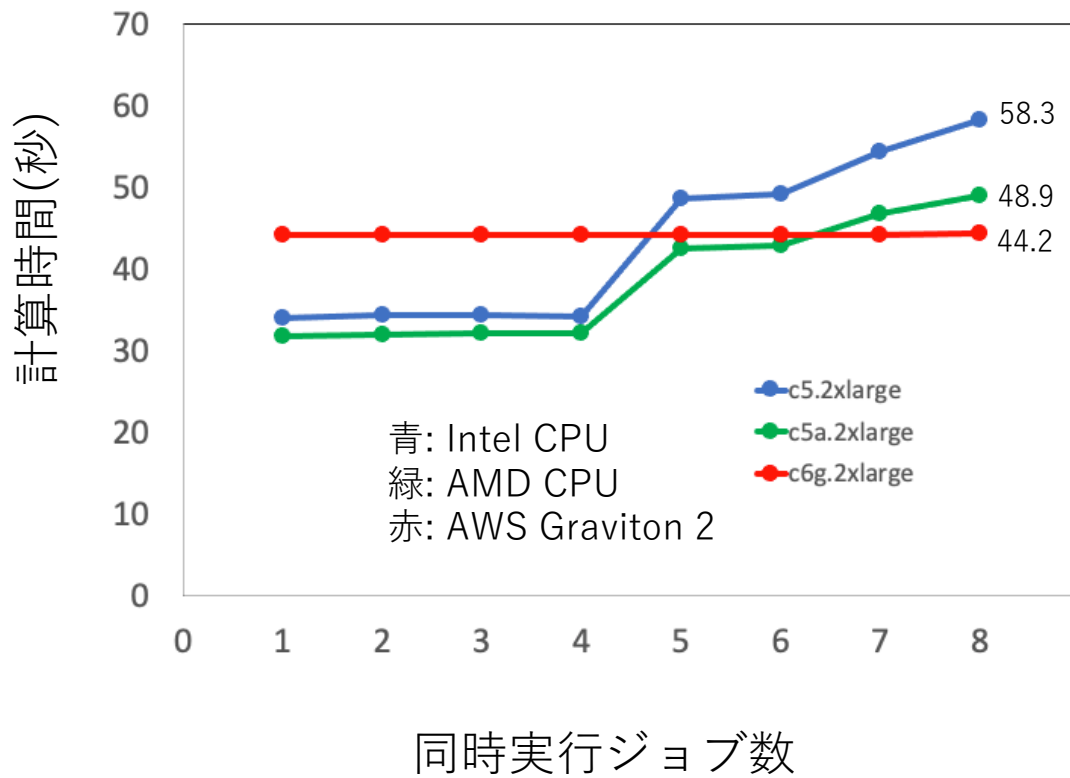


ドッキング計算時間



300万化合物に対するドッキング計算の計算時間(300コアを使用)

CPUのパフォーマンス特性の違い



仮想8コアのシステムで1-8本のジョブを同時投入した際の全てのジョブ完了までの時間をそれぞれ測定した。
1本のジョブは、100化合物のドッキング。

AWS Graviton 2は、同時実行ジョブ数によらず、計算時間が一定。
Intel CPUとAMD CPUは、物理コア数までは計算時間が一定だが、物理コア数を超えるジョブ実行で遅くなっていた。

コストの比較1(CPU特性データから算出)

仮想8コアシステムでの8ジョブ実行時のコスト比較:

c5.2xlarge (HT on) : 1.00 (基準)

c5.2xlarge (HT off) : 1.17

c5a.2xlarge (HT on) : 0.93

c6g.2xlarge : 0.68

c5.2xlargeのコストパフォーマンスを1した。

**AWS Graviton CPU搭載インスタンスの計算コストは、
Intel CPU搭載インスタンスの68%だった。**

(ただし、仮想コアを全て使った場合)

インスタンス料金

c5a.2xlarge (8 vCPU, 0.384 USD/時間 = 40.2円/時間)

c5.2xlarge (8 vCPU, 0.428 USD/時間 = 44.8円/時間)

c6g.2xlarge (8 vCPU, 0.3424USD/時間 = 35.9円/時間)

東京リージョン, オンデマンド (2020/10/26)

円の計算は1ドル=104.75円で換算

スクリーニング計算の時間とコスト(Parallel Cluster)

スクリーニング計算全体の時間は、ドッキング計算時間+約1時間。
(準備30分、ドッキング後計算30分)

Parallel Clusterでマスターノードと同じインスタンスを使用し、
計算ノードは常に1台起動していると想定して、c6g. 16xlargeの場合
で計算コストを概算すると

ドッキング計算時間：94分

スクリーニング計算時間：154分(90+60)

スクリーニング計算時間中は常時2ノードを起動(master+1)

ドッキング計算の時だけ:+4ノードを追加で起動

c6g. 16xlarge料金(東京、オンデマンド)：3.072 USD/時間

$(94+60) / 60 * 2 \text{ (ノード)} * 3.072$

$+ 94/60 * 4 \text{ (ノード)} * 3.072$

$= 35.02 \text{ USD} = 3,669\text{円} \text{ (1ドル=104.75円で換算)}$

スクリーニング計算の時間とコストまとめ(Parallel Cluster)

c6g.16xlarge	:	154 分	, 3,669円	, (0.68)	
c5.18xlarge (HT on)	:	180 分	, 5,380円	, (1.00)	(ドッキング計算時間を補正-10分)
c5.18xlarge (HT of)	:	149 分	, 6,788円	, (1.26)	
c5a.16xlarge (HT on)	:	168 分	, 4,011円	, (0.74)	(ドッキング計算時間を補正-5分)

HT on/offは、ハイパースレッディングのon/off

1ドル, 104.75円で換算

括弧内の数字は、c5.18xlarge(HT on)を1とした場合のコストの比率。

スクリーニング計算時間は、ドッキング計算時間+60分で算出。

補正分は、計算機コストをより正確に求めるために、ノード起動数の時間変化グラフから概算。

インスタンス料金

c5a.16xlarge (64 vCPU, 3.072 USD/時間 = 321.8円/時間)

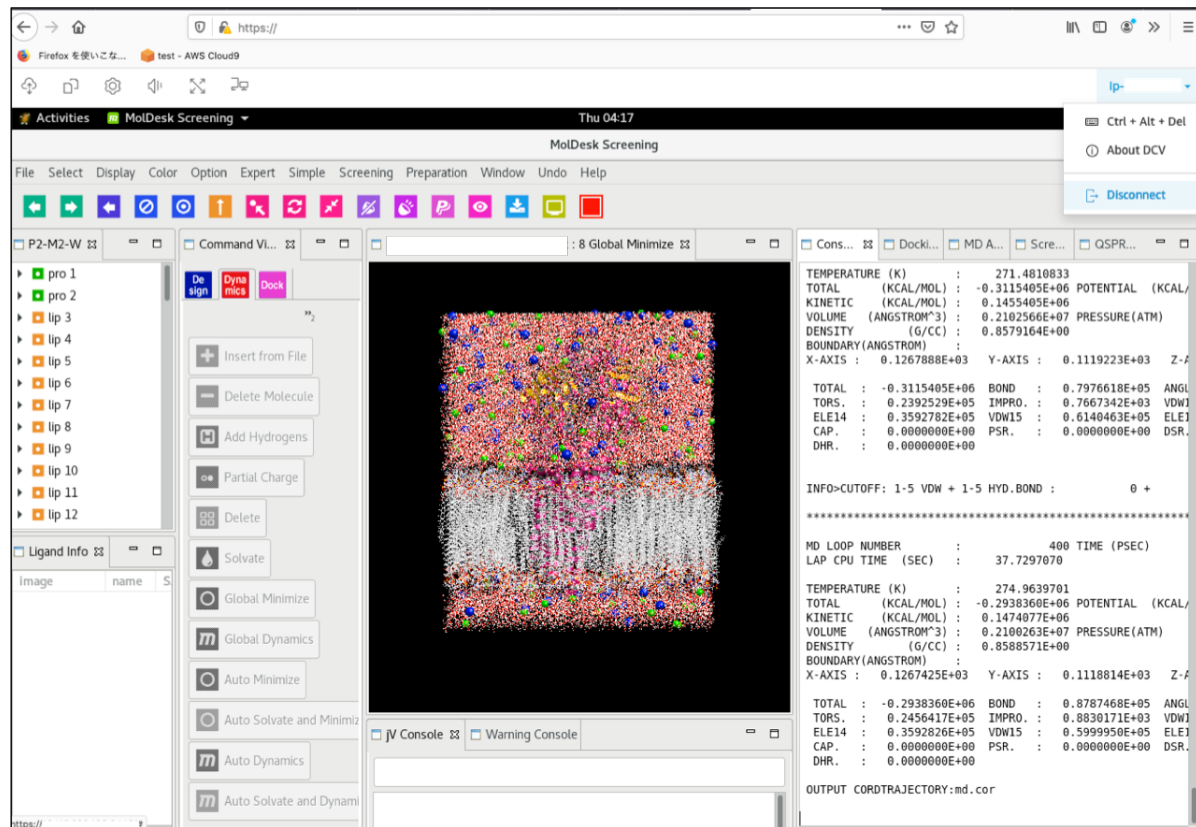
c5.18xlarge (72 vCPU, 3.852 USD/時間 = 403.5円/時間)

c6g.16xlarge (64 vCPU, 2.739 USD/時間 = 286.9円/時間)

オンデマンド料金、東京リージョン (2020/10/26)

円の計算は1ドル=104.75円で換算

クラウドでのGUIソフト利用例



webブラウザから
クラウド計算機の
デスクトップを
利用する機能も利用できる。
(NICE-DCV)

Parallel Clusterの
環境自動構築機能を利用すると、

- NICE-DCV
- CUDA
- MPI

の設定も簡単。

MolDesk Screeningでは、
GUI上の操作でGPU+MPIを
使ったMD計算も実行できる。

まとめと展望

myPrestoで300万化合物のスクリーニング計算は、
3時間以内の時間で、3,600-7,000円程度の計算機費用で実施できる。

クラウドでは、多様なシステム構成を簡単に構築して利用できるため、
自分の計算にあったシステム構成の評価も簡単にできる。
今回評価したものの中では、スクリーニング計算では、ArmベースCPU
であるAWS Graviton 2を搭載したインスタンスがコストパフォーマンス
が良いことがわかった。

AWS Parallel Clusterを使用すると、簡単に大規模な計算を実施できる。
自動スケーリング機能で並列数を自動的に拡大できるため、今後、
計算対象化合物数が増えても、容易に対応可能だと思われる。