## 自動計算システム

#### 仮定

- 計算サーバーはPCと別に存在する。
- 計算サーバーが計算している状態はPCからは分からない。計算サーバーはbatchを用いて計算を行う。

PCでのstatusと計算サーバーでのstatusの2つの状態を持つ。

## 自動計算システム

Inputdata/fcc/random structure1

Inputdata/fcc/random\_structure2

Inputdata/fcc/random\_structure3

Inputdata/fcc/deformedlattice structure1

Inputdata/fcc/deformedlattice structure2

Inputdata/fcc/deformedlattice\_structure3

ID1.0

ID2.0

ID3.0

ID4.0

ID5.0

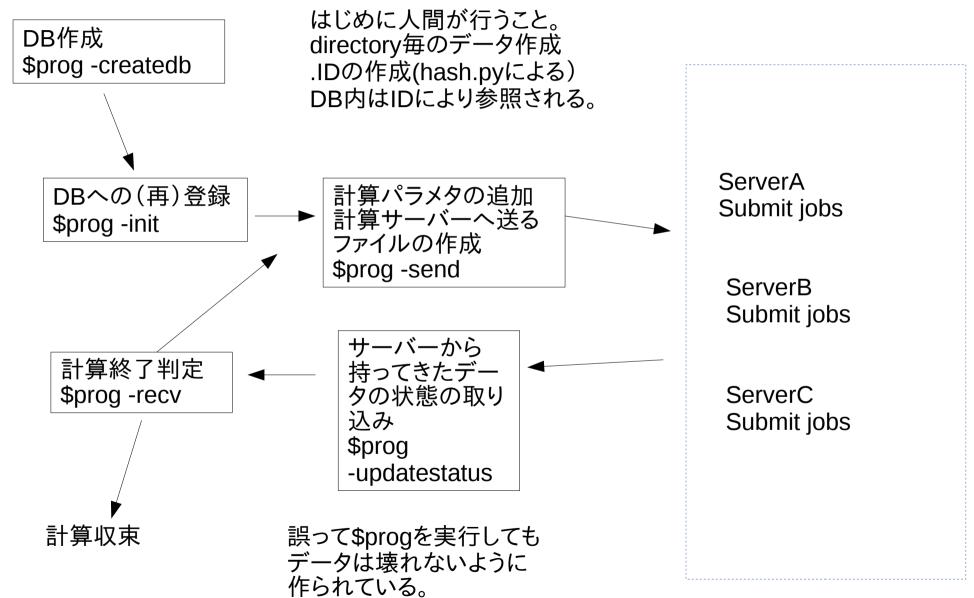
ID6.0

• • •

. . .

Inputdata/hcp/deformedlattice\_structure1

# Autcc (半自動計算システム)



### autccの実装

- autcc -createdbdatabaseを作る。
- · autcc -init

Inputdir/以下でfile .IDがあるdirectoryを検索する。

IDがDBにある場合はstatus=(new,idle)にする。count=0

IDがDBにあるものに対しては何もしない。(inputdir/にdirectoryを随時追加して良い。)

autcc -send

calcdata/ID.count/をつくる。countによりinputfileをつくる。入力ファイルからinputfileをつくるroutineをつくる必要がある。

(count>0の場合は先の計算で失敗しているので収束パラメタを変更することになる。)

必要なファイルのコピーなども可能。

calcdata/ID.count/.EXECSTATUSをつくる。サーバー上の計算状態を表す。ファイルの中身はidle。

IDの状態を(submitted,idle)にする。

ファイルを計算サーバーへ持っていく。jobが正常終了したらID.count/.EXECSTATUSの中身をfinishedとしておく。(PCからはjobの状態は分からない。) 計算サーバーからファイルをcalcdata/ID.countへコピーする。

· autcc -updatestatus

ID.count/.EXECSTATUSの状態をDBに反映させる。計算が終了していなければ(submitted,idle)、終了していれば(submitted,finished)になる。

autcc -recv

(submitted,finished)のみ処理を行う。

Calcdata/ID.count/output\_scf.txt からSCFが収束したかを判定する。収束していたら(claculated,finished)、収束していなかったら(new,idle),count++をする。