



VIVER

[váivər]

次世代Linuxディスクレスブート
テクノロジ VIVER

1. VIVERとは？

2. VIVERの使い方

3. VIVERの仕組み

4. V-FIELD

自己紹介



氏名：古橋貞之（ふるはしさだゆき）

年齢：19

キーワード：冗長化、負荷分散、動的、
C++、Ruby、筑波大学、未踏ユース

自己紹介

2006年度未踏ユースで
VIVER 0.3を開発

氏名ス吉橋貞史(エイサキヤシダヒコ)
年齢：19

認定

キーワード：冗長化、負荷分散、動的、

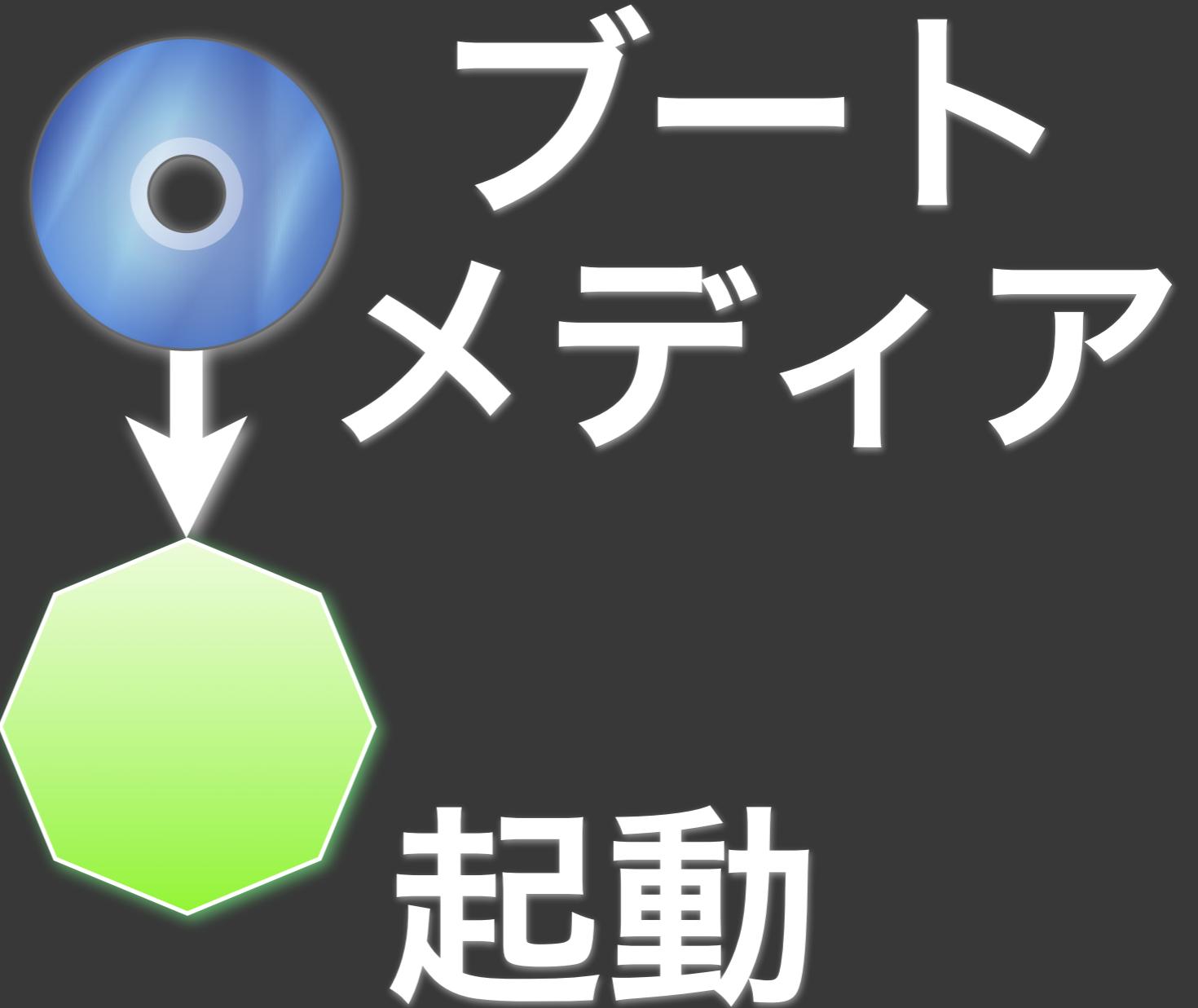
C++、Ruby、筑波大学、未踏ユース

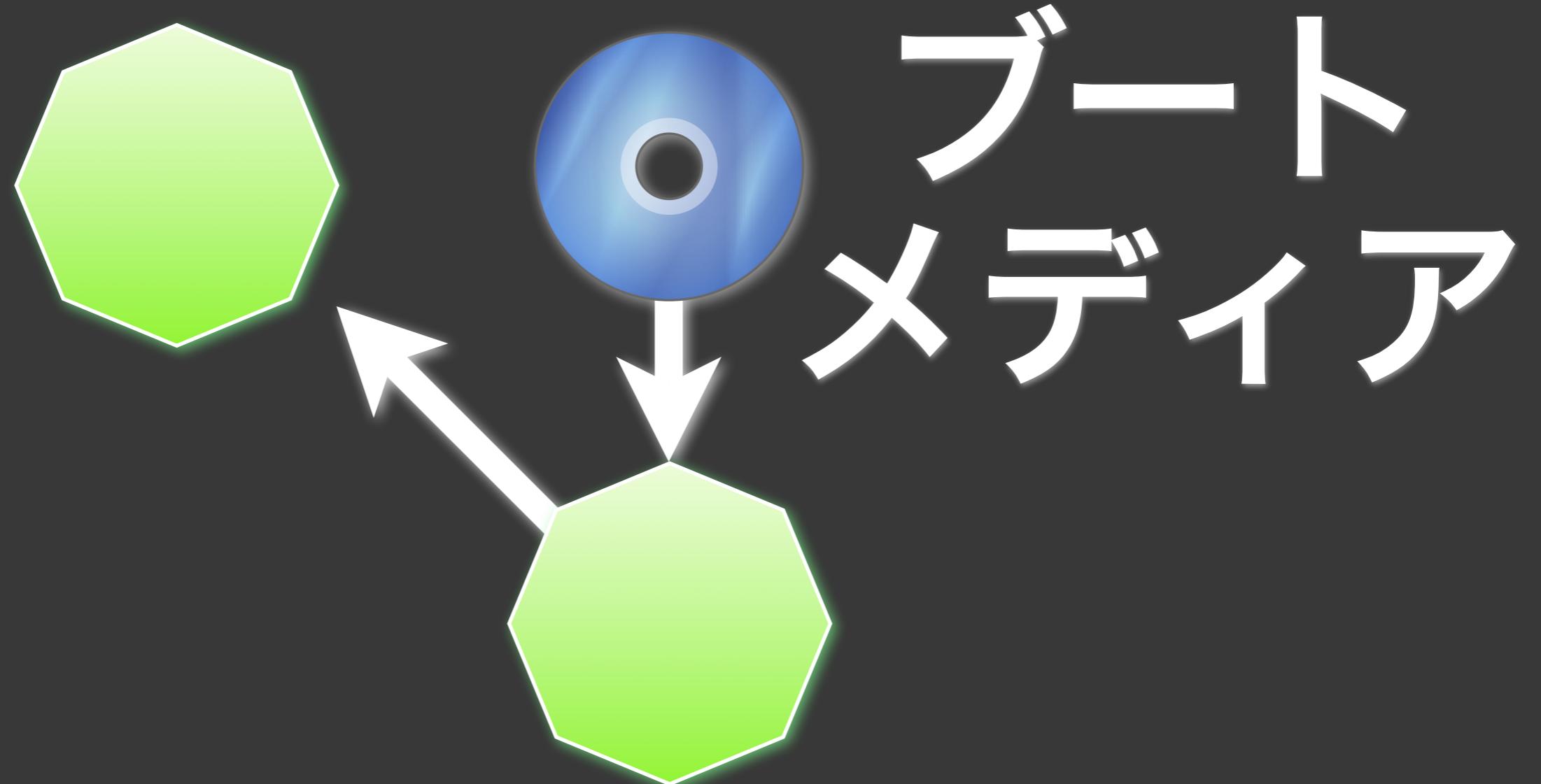


I

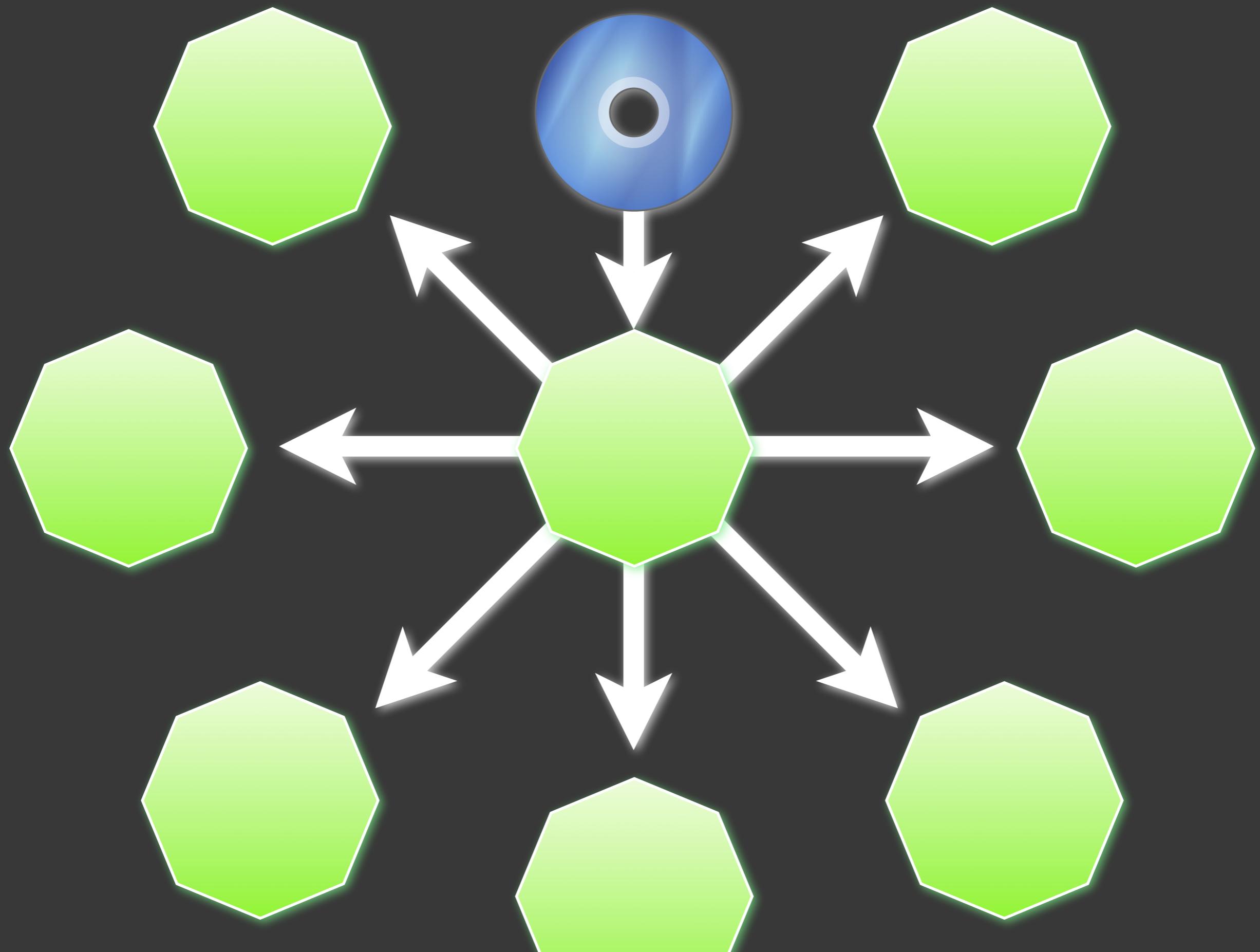
VIVER

& fit?





ネットワーク
ブート(PXE)



Live Linux

(KNOPPIXなど)

とは違う？

全然違う

似ている

ところ：

最初の一一台は

CDから起動

→USBメモリ
でも可

→HDD
でも可

→ SDカード
でも可

ハイードウェア

自動検出

→ 設定不要で
すぐ使える



NFS-Root

(NFSを/にして
ネットワークブート)

とは違う？

すきし

似ている

違う

ところ:

最初の一一台は

CDから起動

→ サーバーの
設定不要

ハートドウェア

自動検出

→異種混合環境
でもOK

NFSを
使わない

高速分散多重化

共有ブロック

デバイス

V-FIELD.



Demo

VIVER

ネットワークブート

II

VIVER

の使い方

Step Ø

インストール

Linuxを

HDDに

Debianでも

CentOSでも

Momongaでも

Mandrivaでも

Step 1

VIVER化キット
をインストール

http://viver.sourceforge.jp/

or

VIVER

検索

カチッ♪

Step 2

ブートディスク
を作る

squashfs

+

mkisofs

Step 3

起動！

起動！

起動！

Step 4

コーヒーを飲む

Step 5

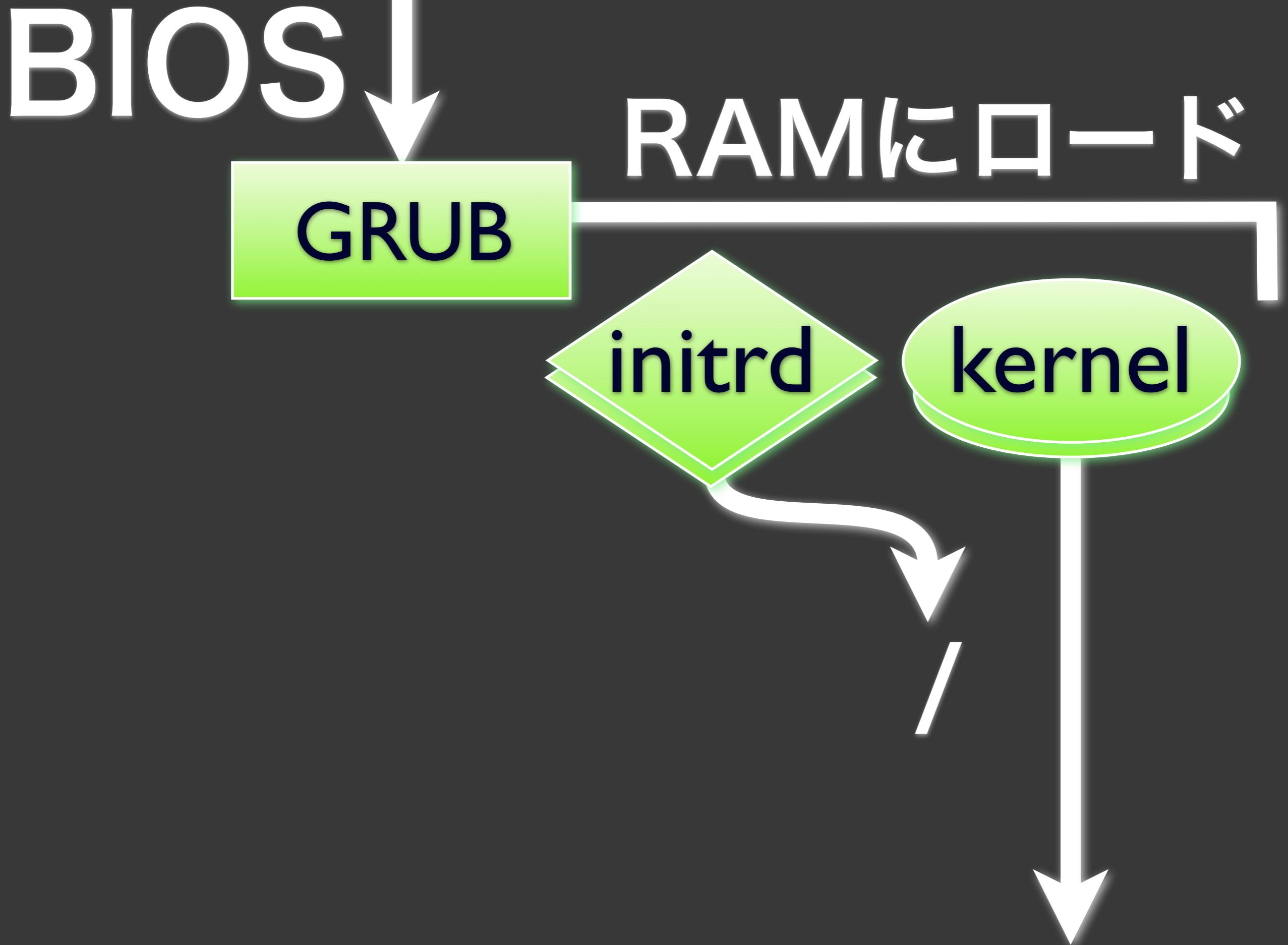
Hackする

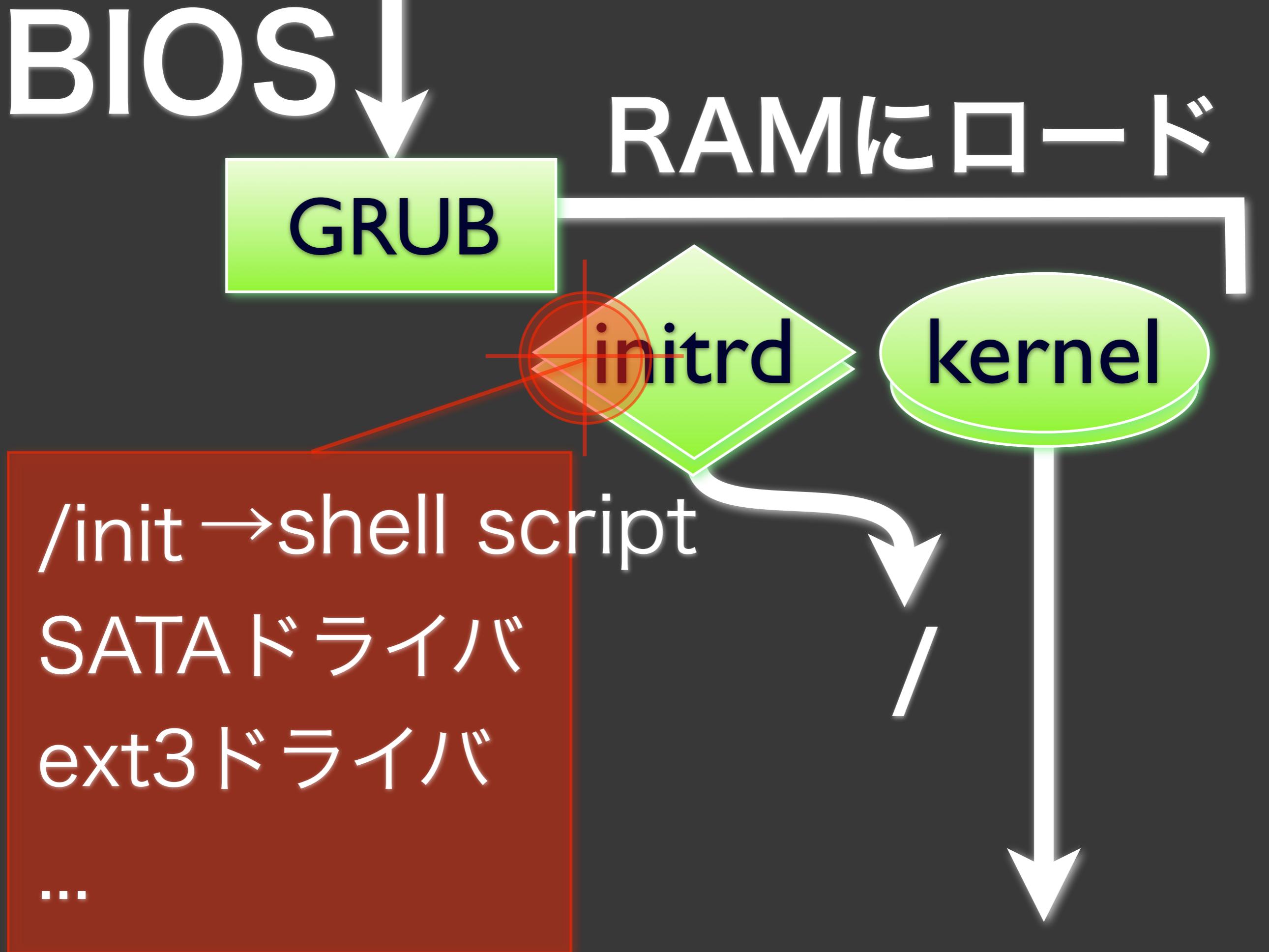
III

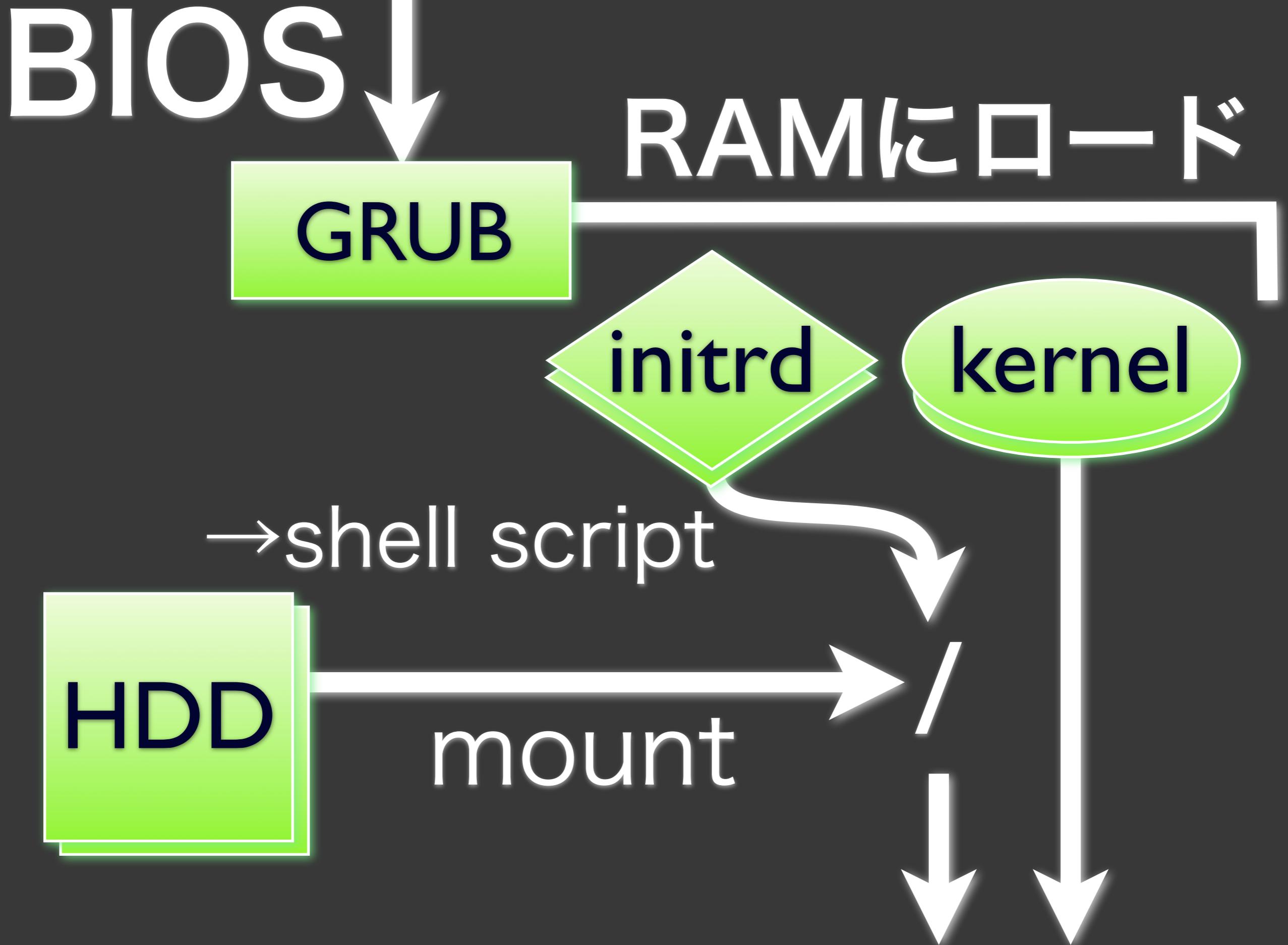
VIVER

の件相談

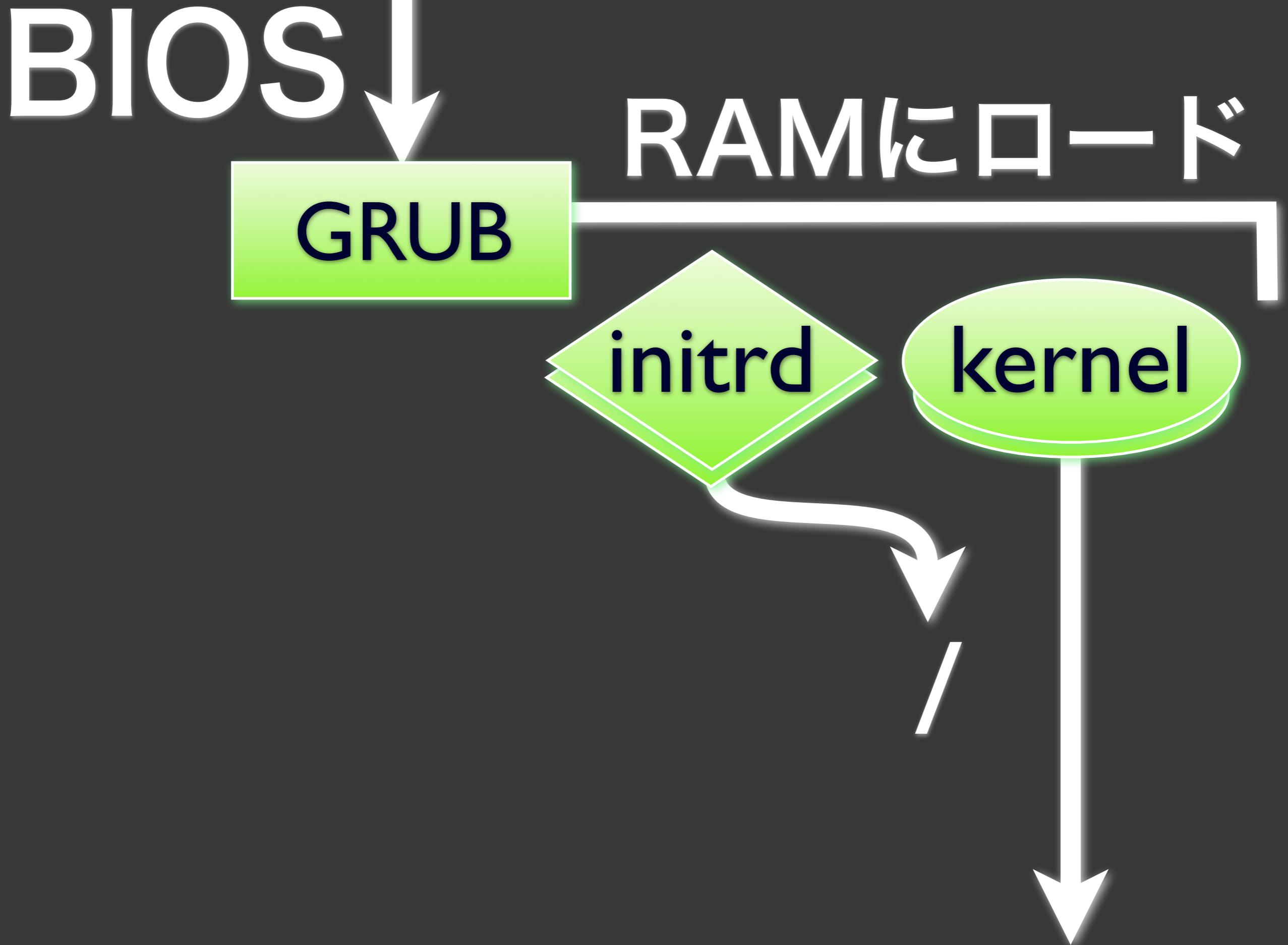
通常のLinuxのブートプロセス

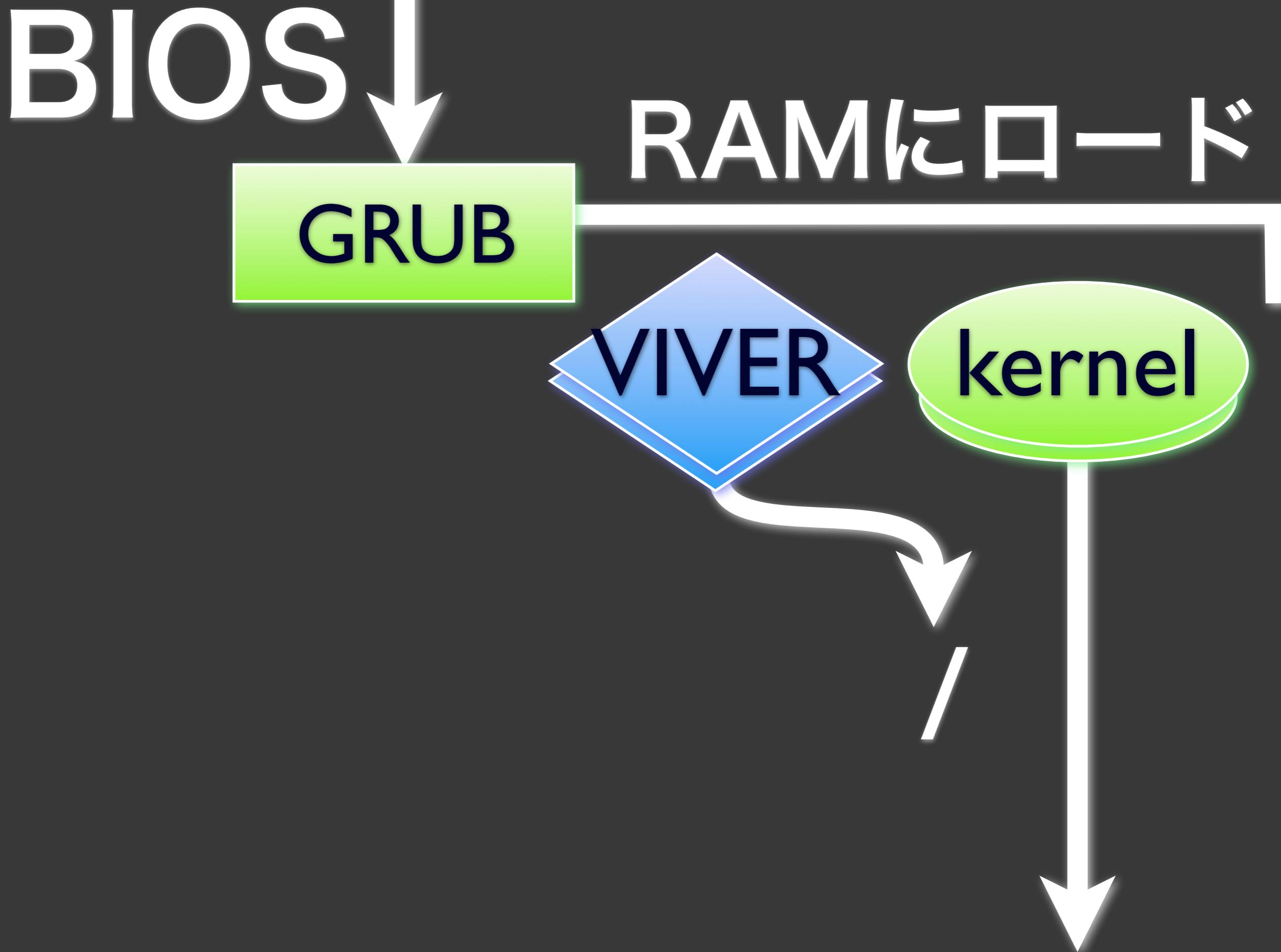


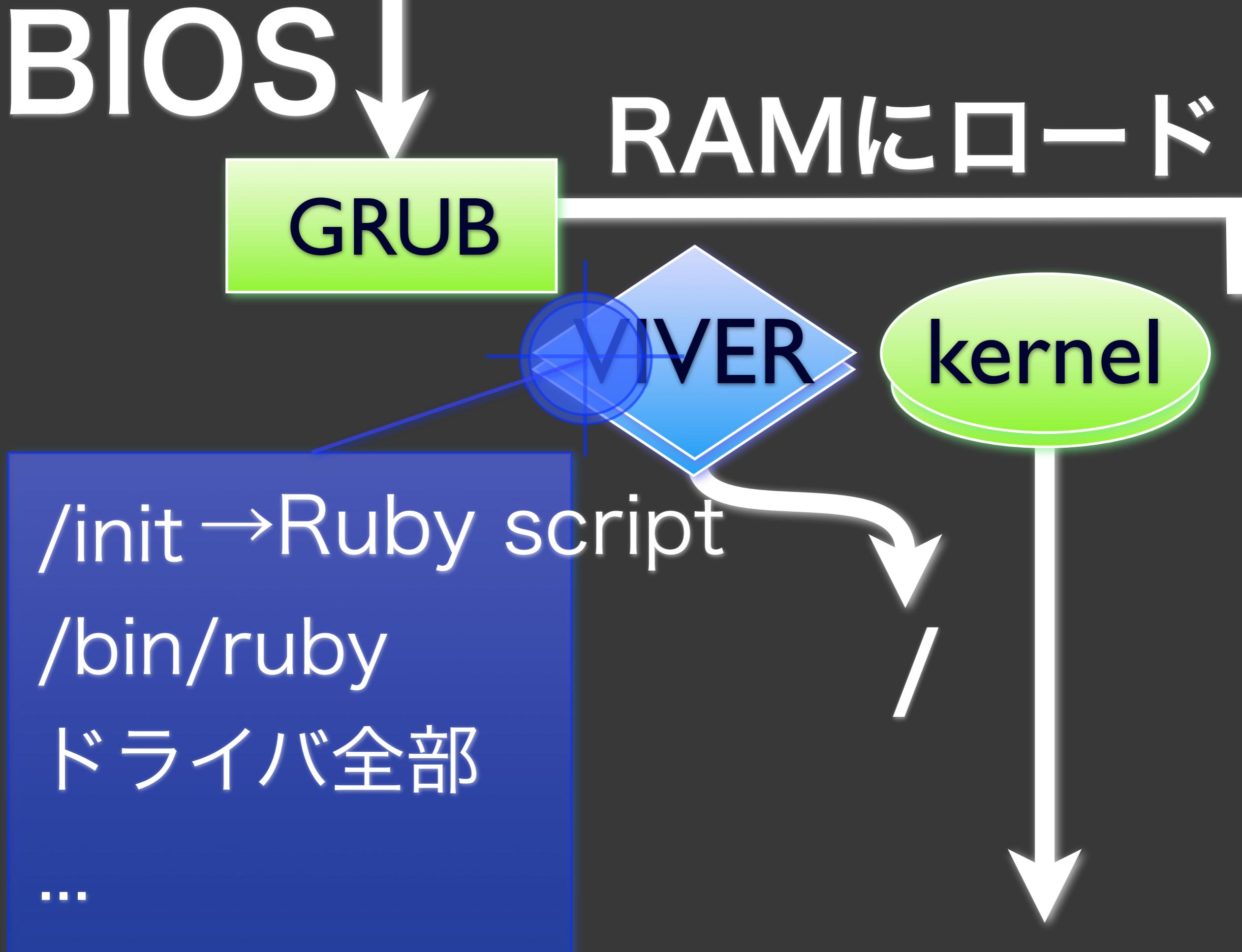


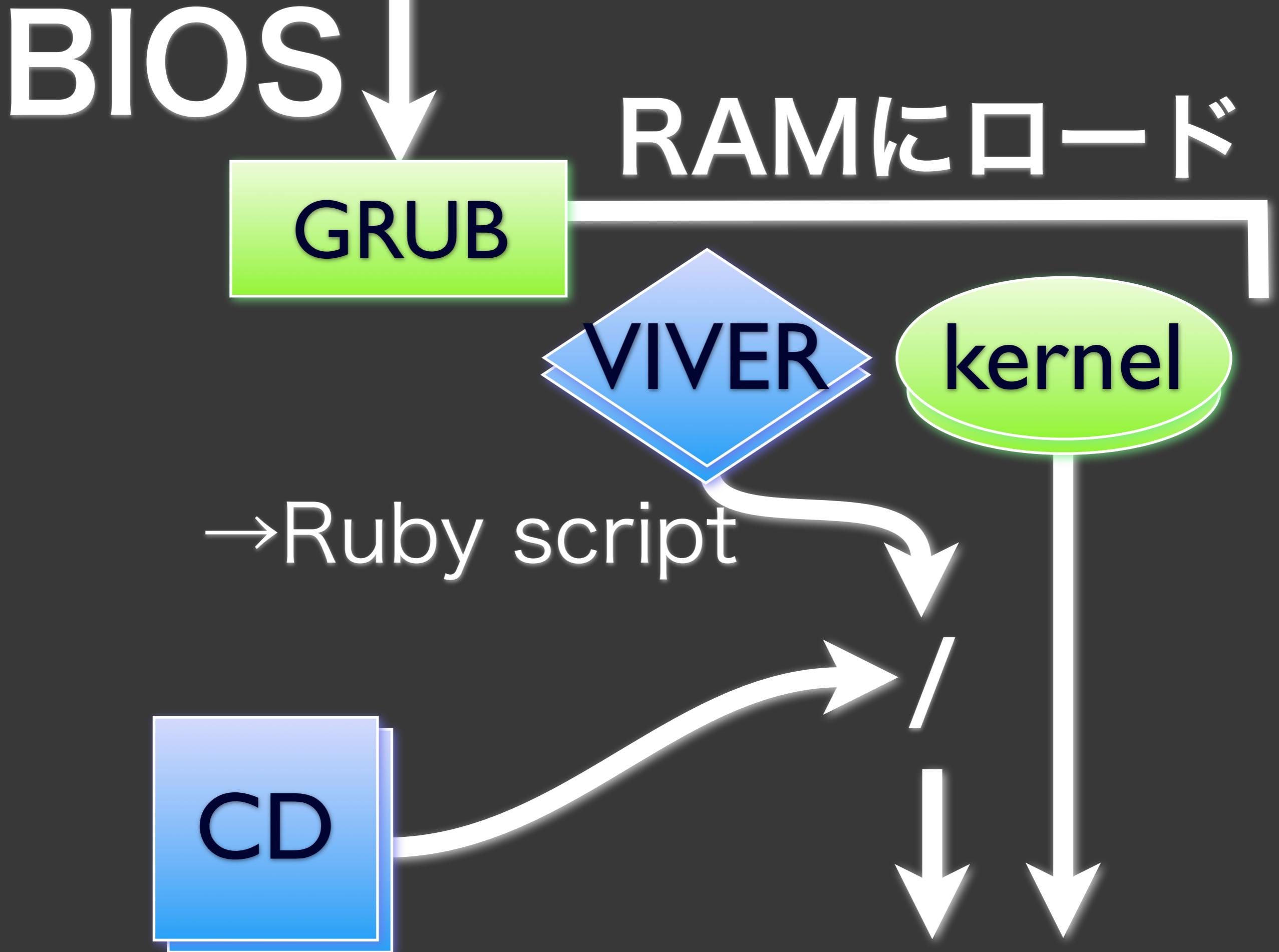


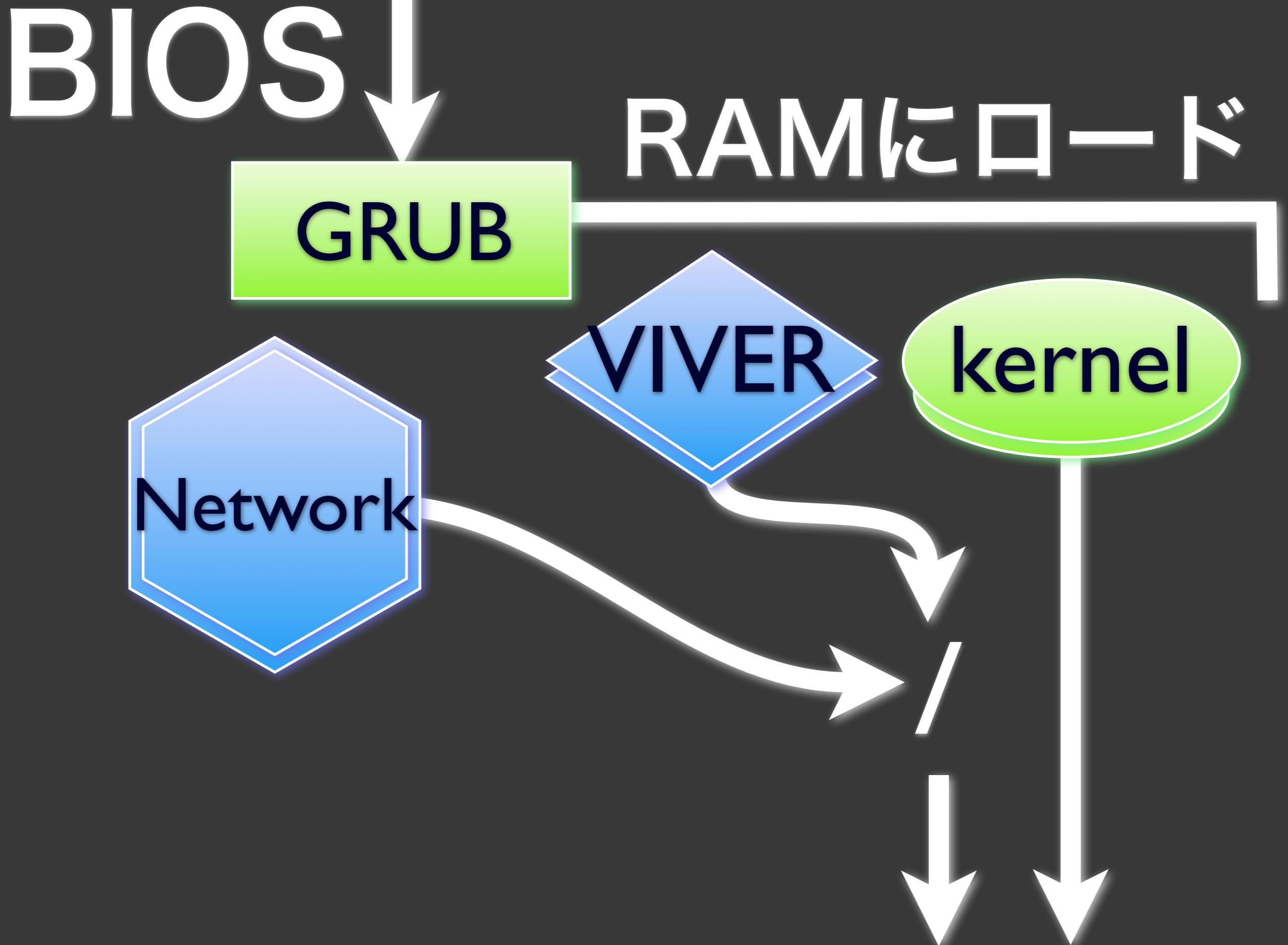
VIVERのブートプロセス







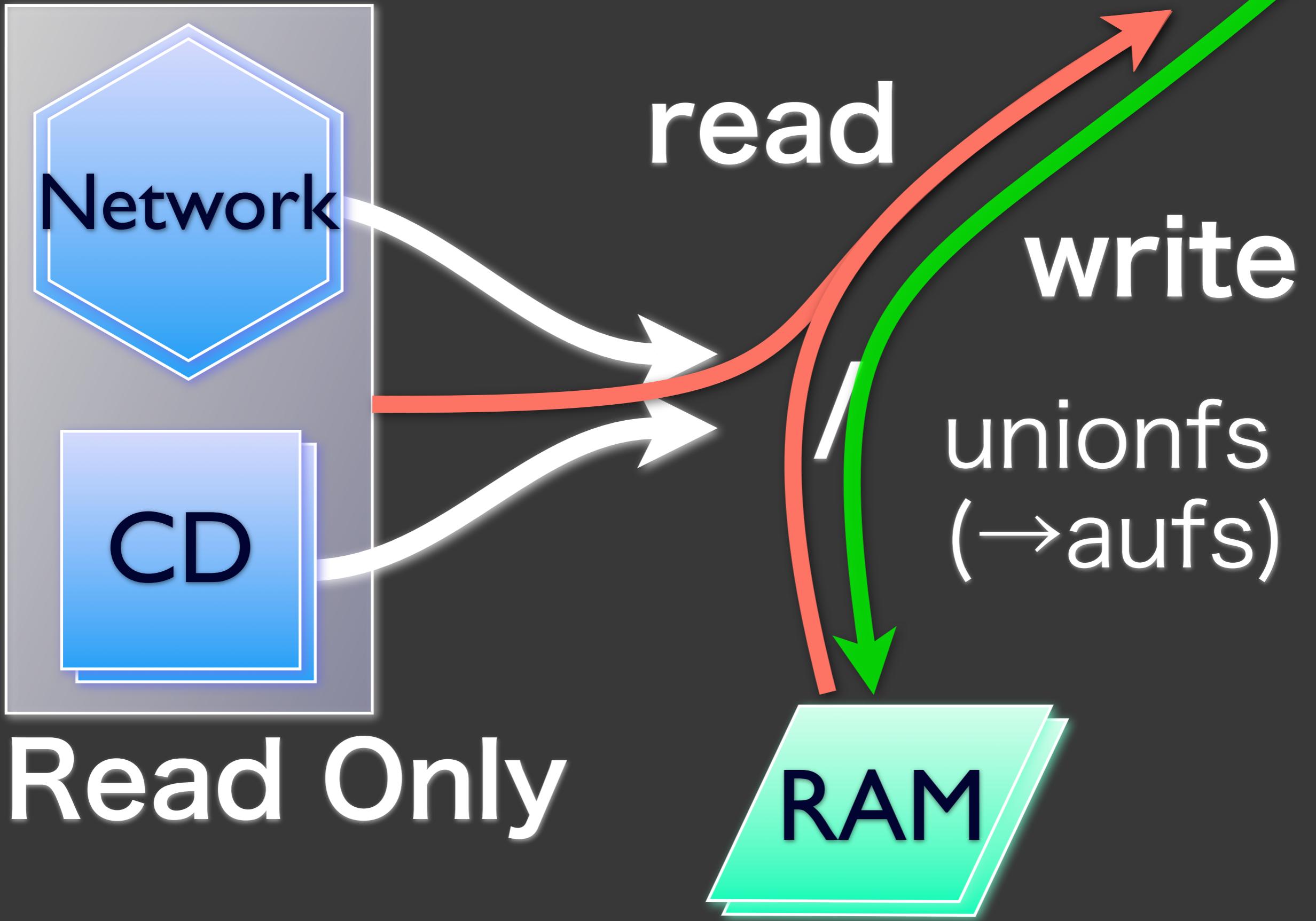


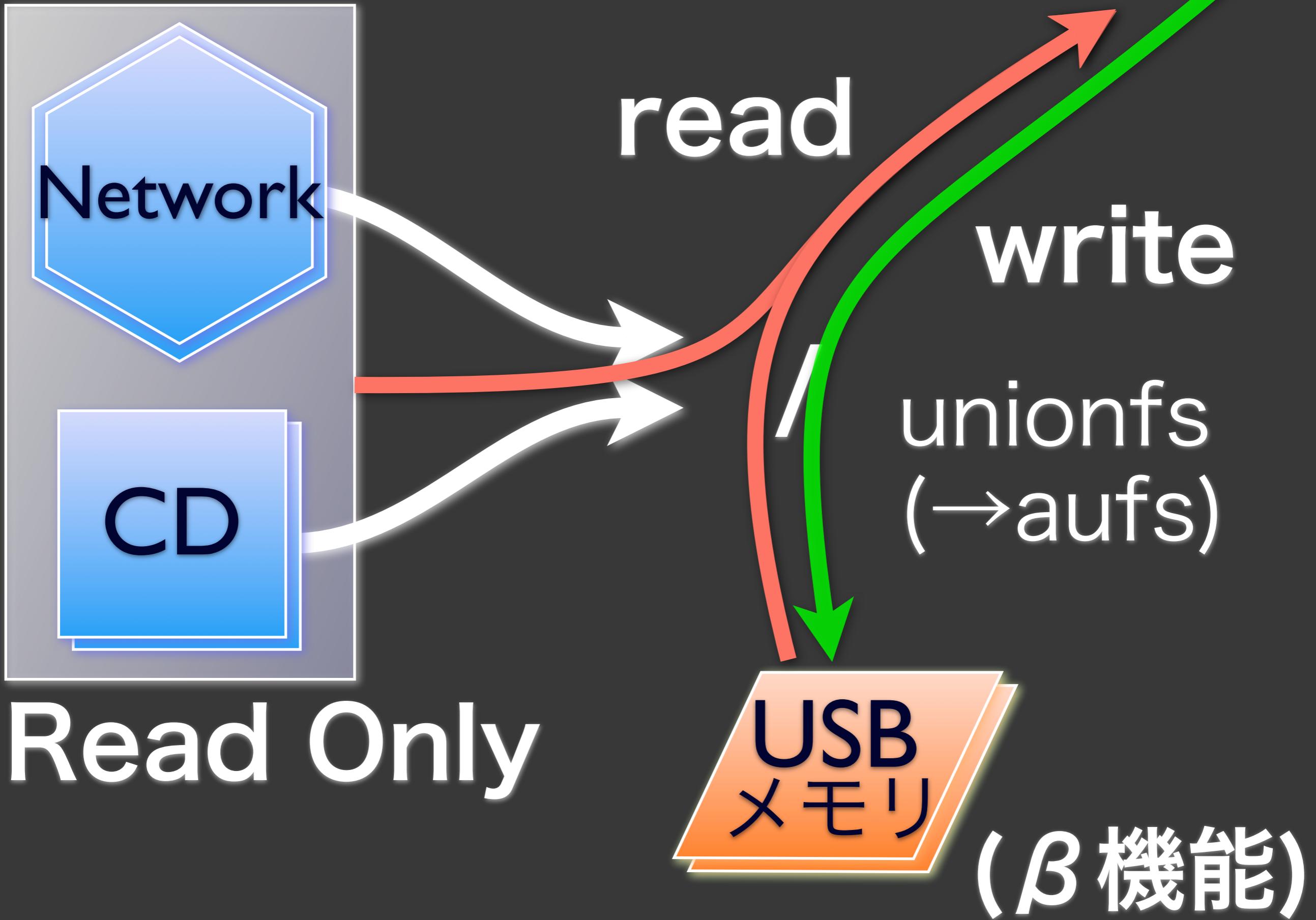


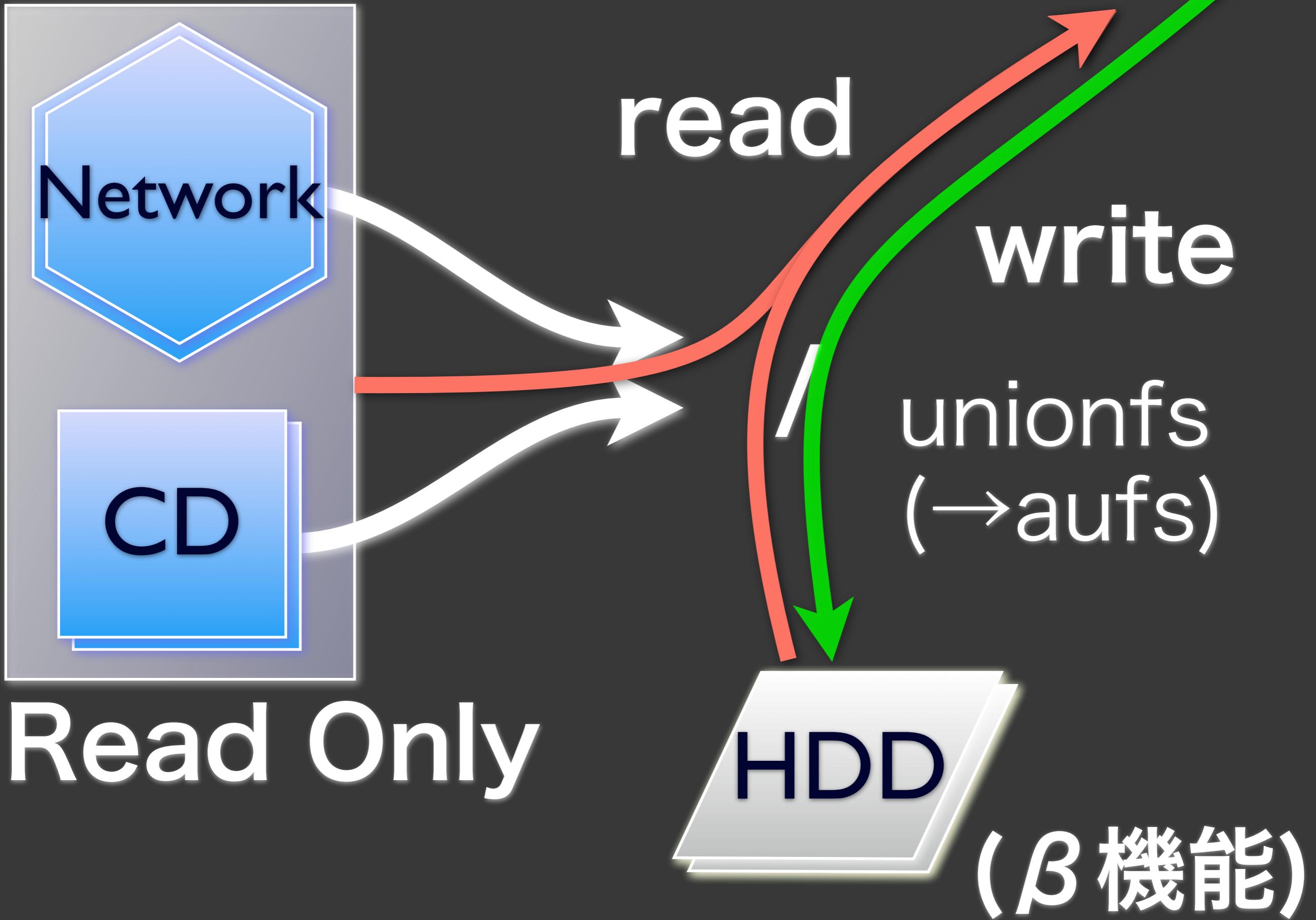
Copy-on-Write



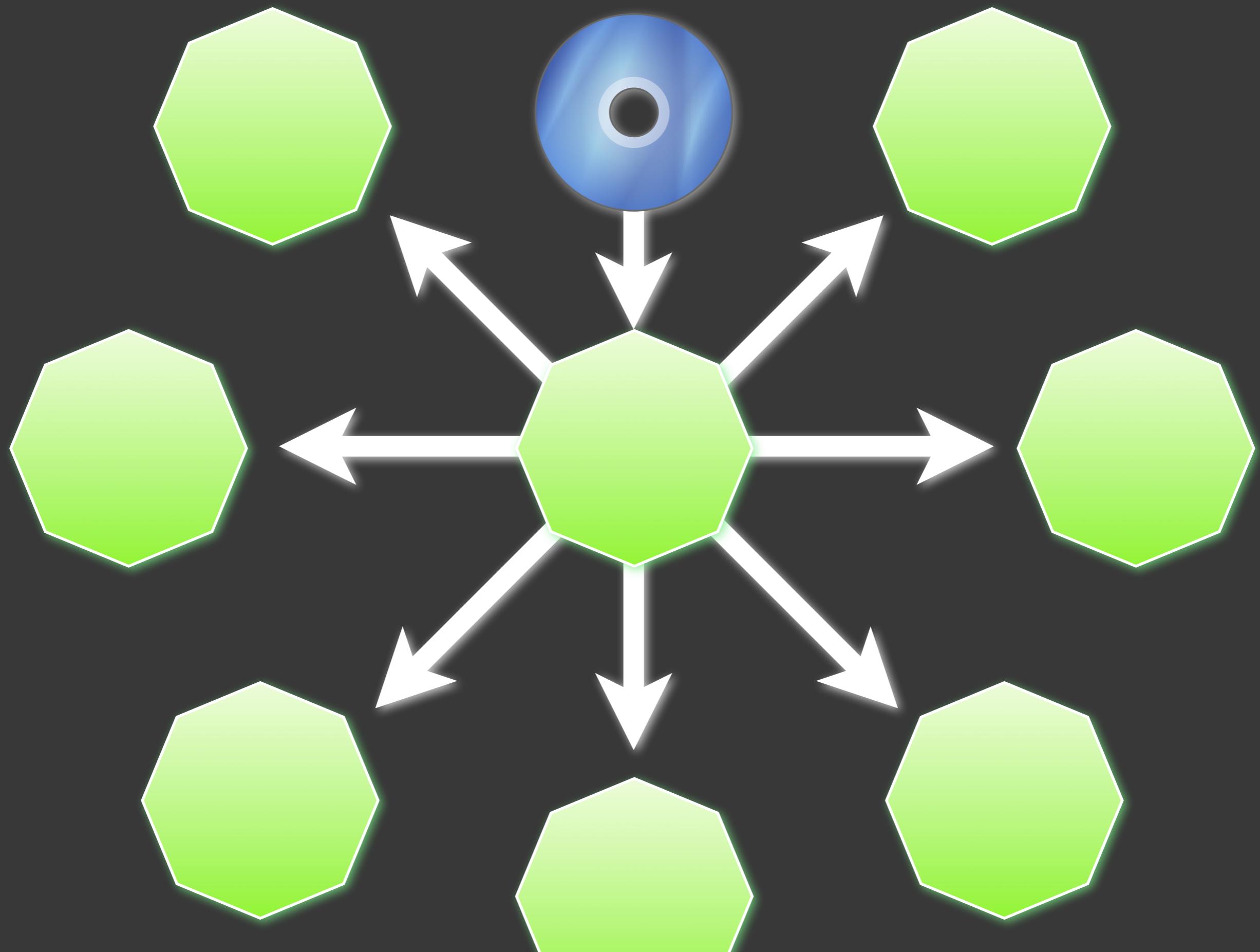
Read Only

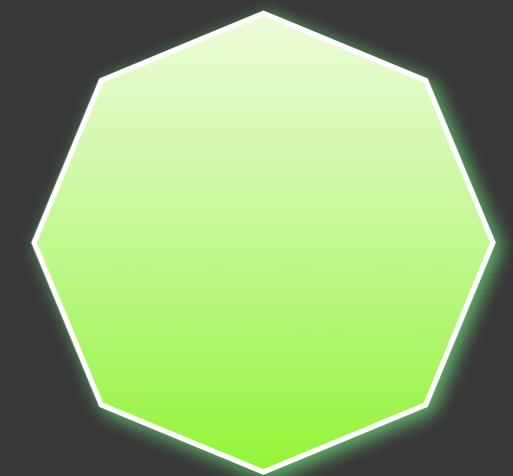






とこうで…





といふのは
ダメですね

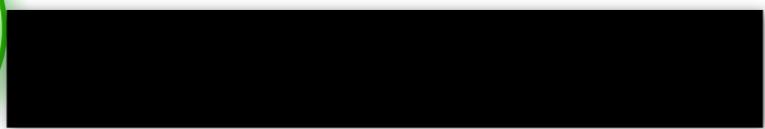
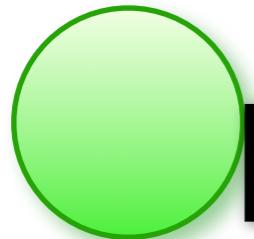
そこで

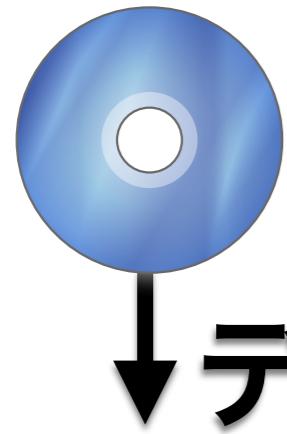
N

W-FIELD

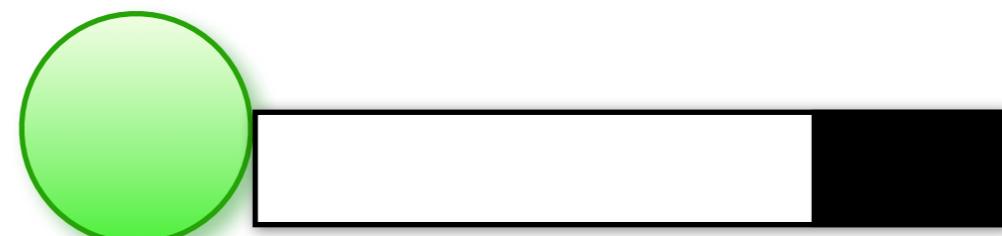
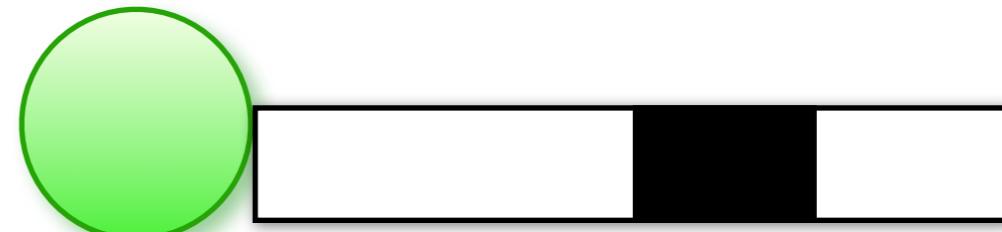
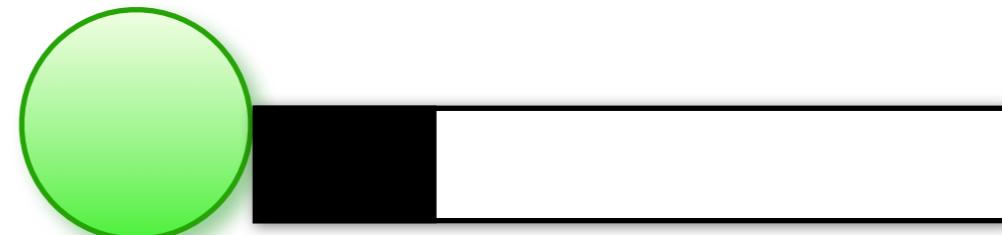
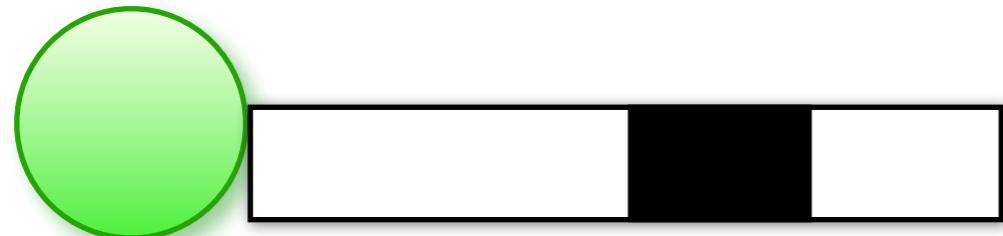
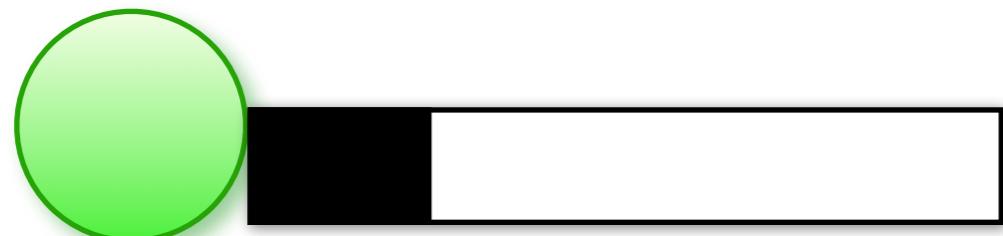


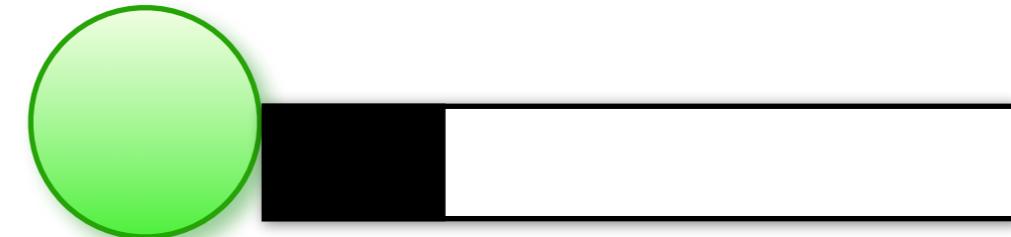
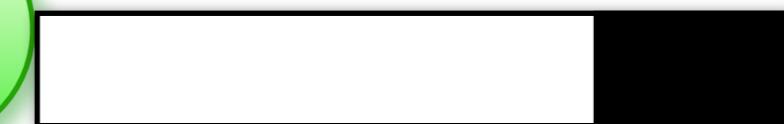
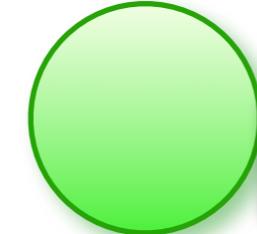
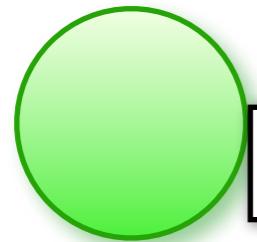
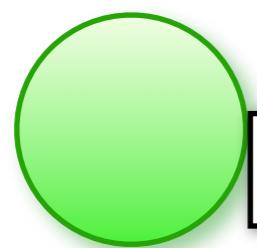
↓ディスクイメージ





ディスクイメージ





九月九日

食
宿

分
離

動的なか

ナーバー

の追加



Demo

**高速分散多重化
共有ブロックデバイス
「V-FIELD」**

転送性能比

V-FIELD

P2P型

**NBD
(Network Block Device)**

クライアント/サーバー型

300MBのディスクイメージを
共有し、全ノードで一斉に
シーケンシャルリード

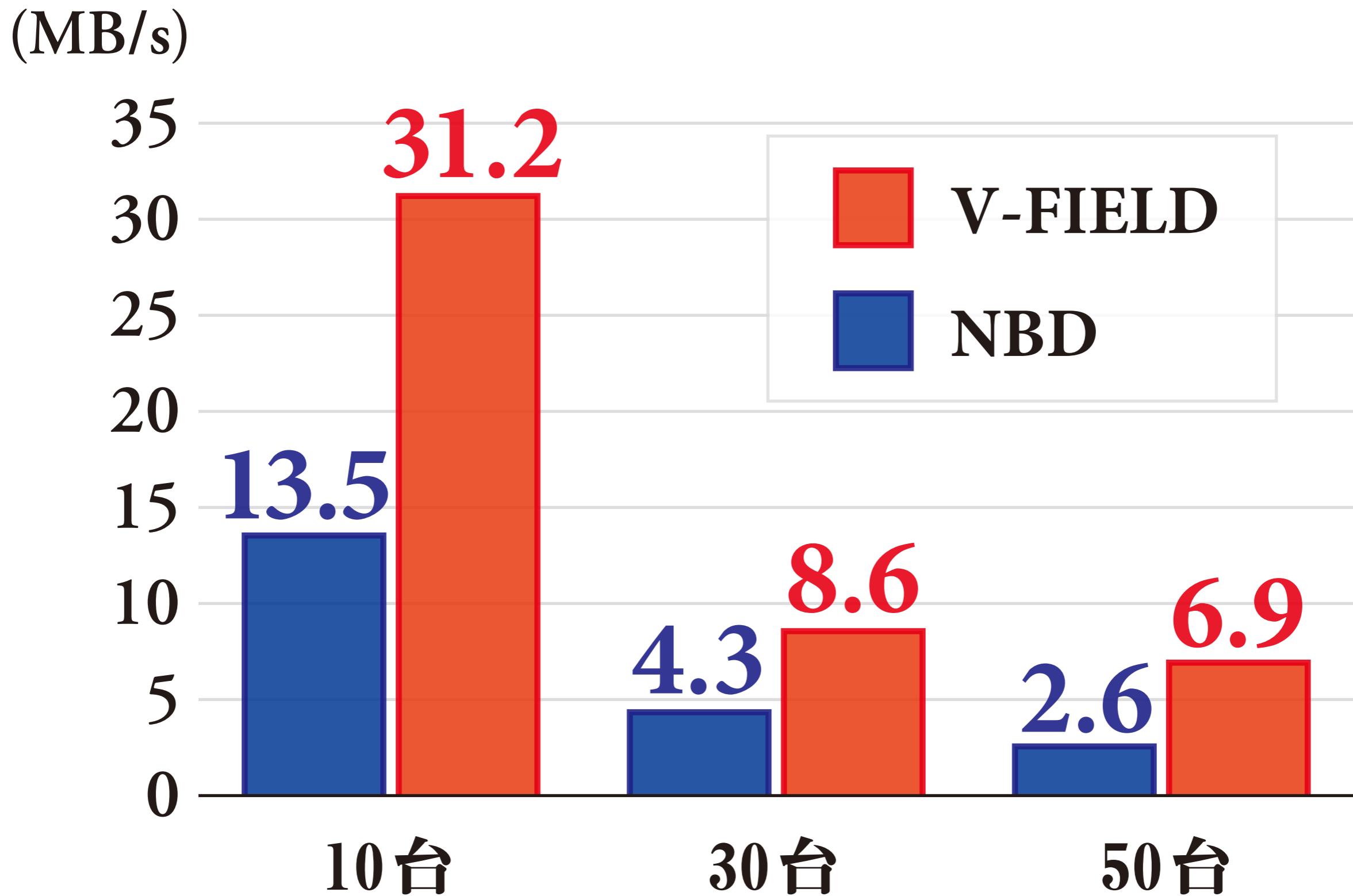
IBM Blade Center HS20

Xeon 3.6GHz ×2

Memory 2GB

NIC BCM5604s (GbE)

NBD: 1対多, V-FIELD: 多対多



問題上

**V-FIELDは
ReadOnly**

→元のディスクイメージ
も書き換え不可

正常に終了

できなさい

→電源ボタン長押し

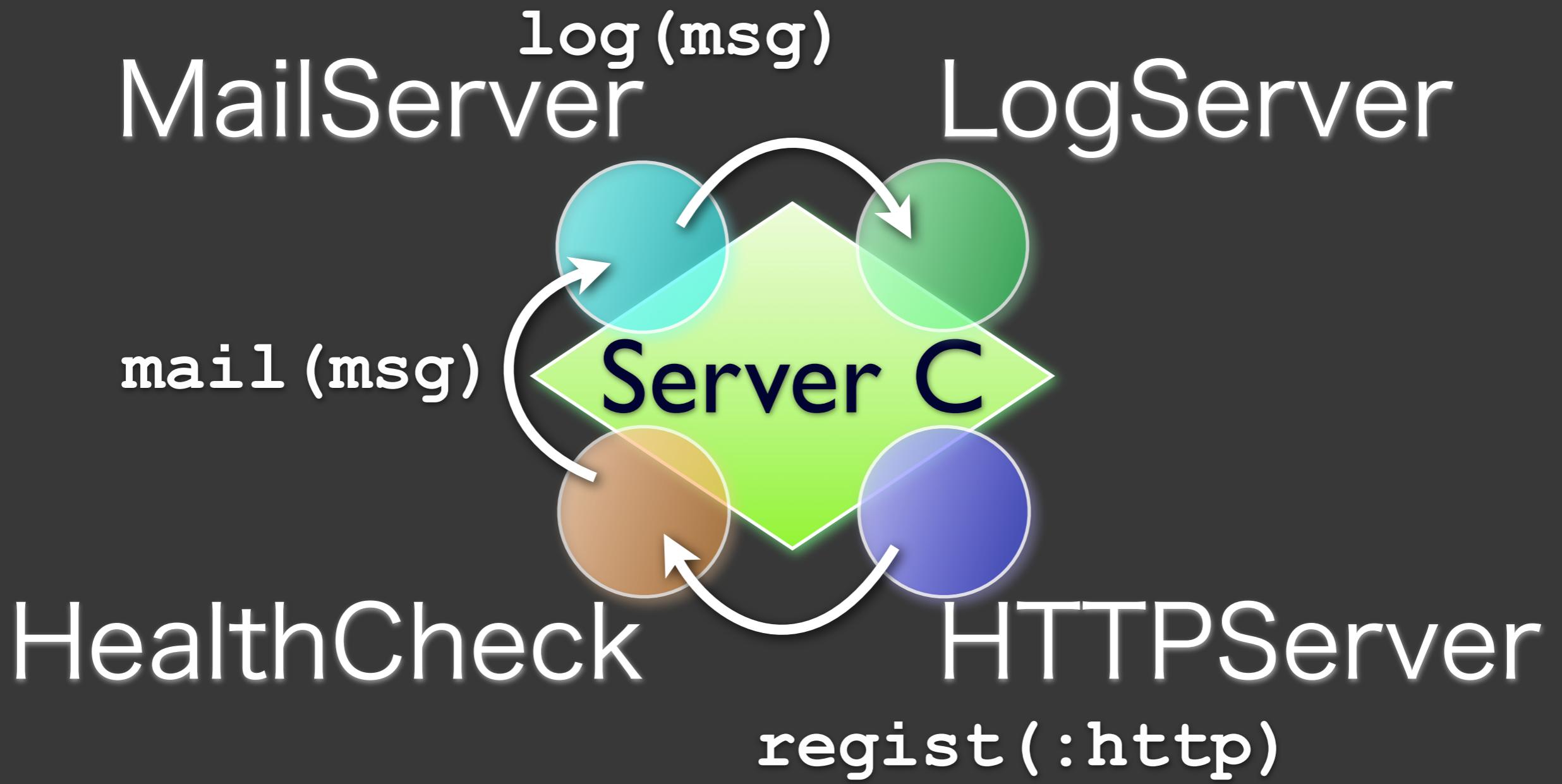
→Copy-on-Writeで
RAMを使っているなら

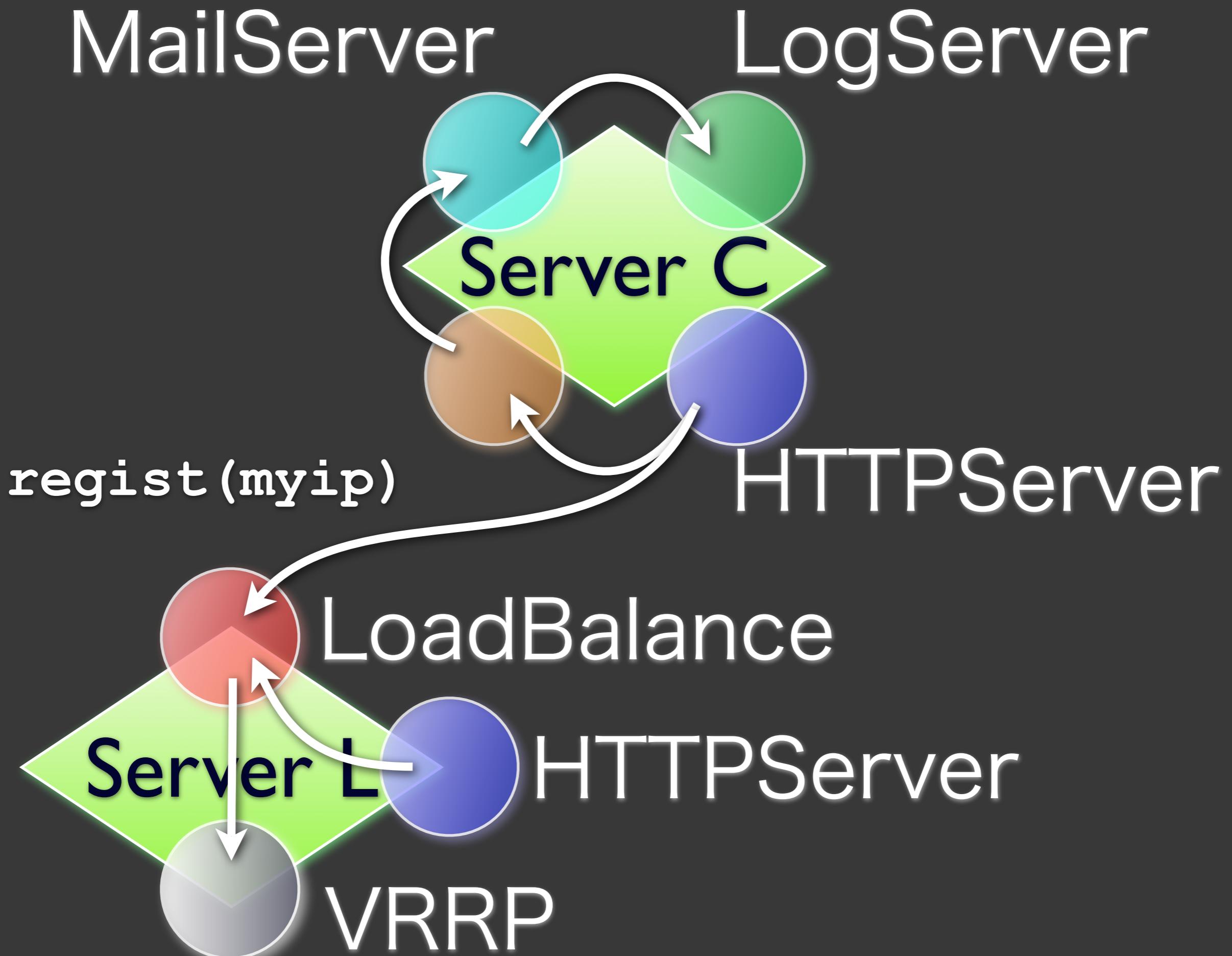
問題なし

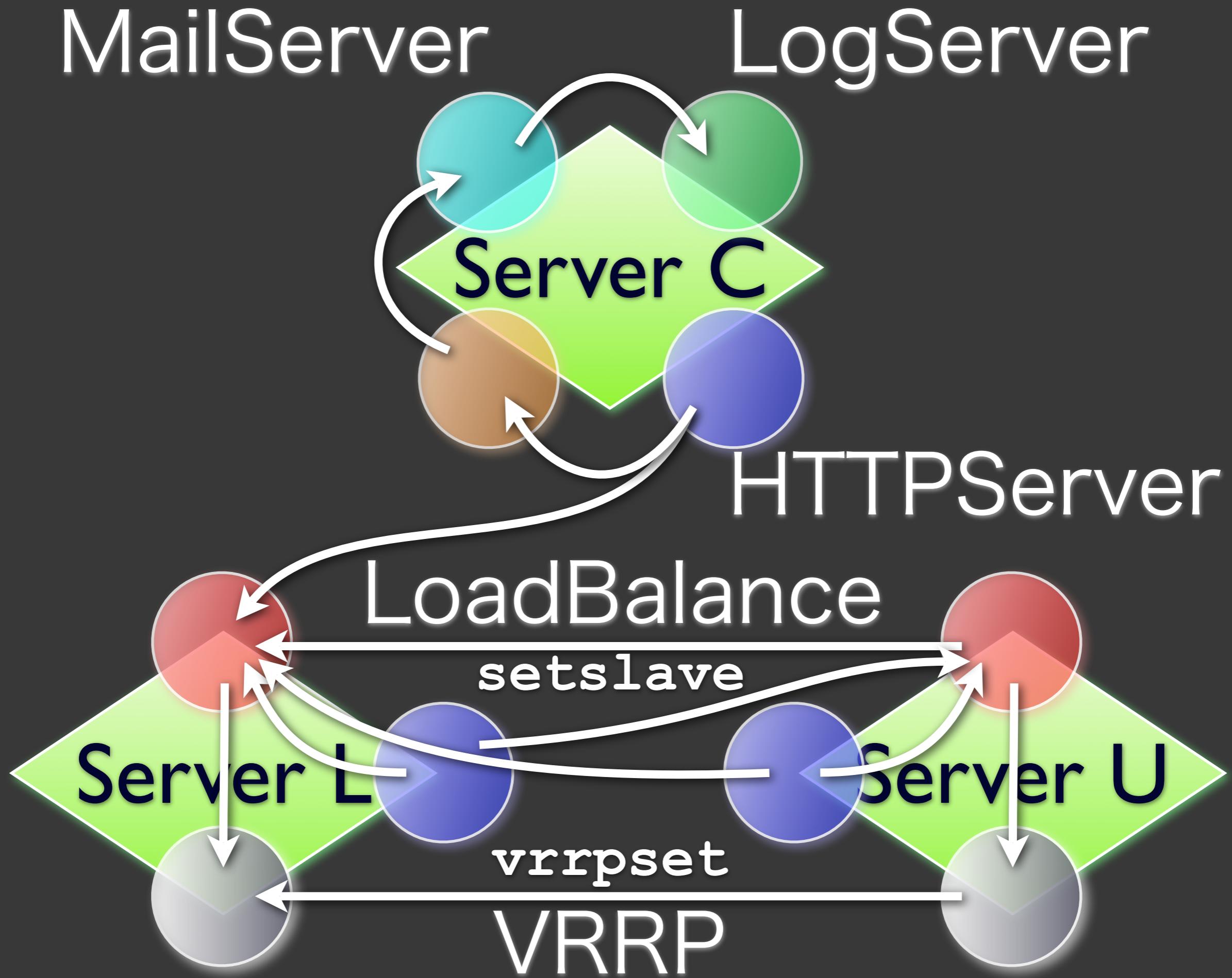
(どちらにしても)
(データは消える)

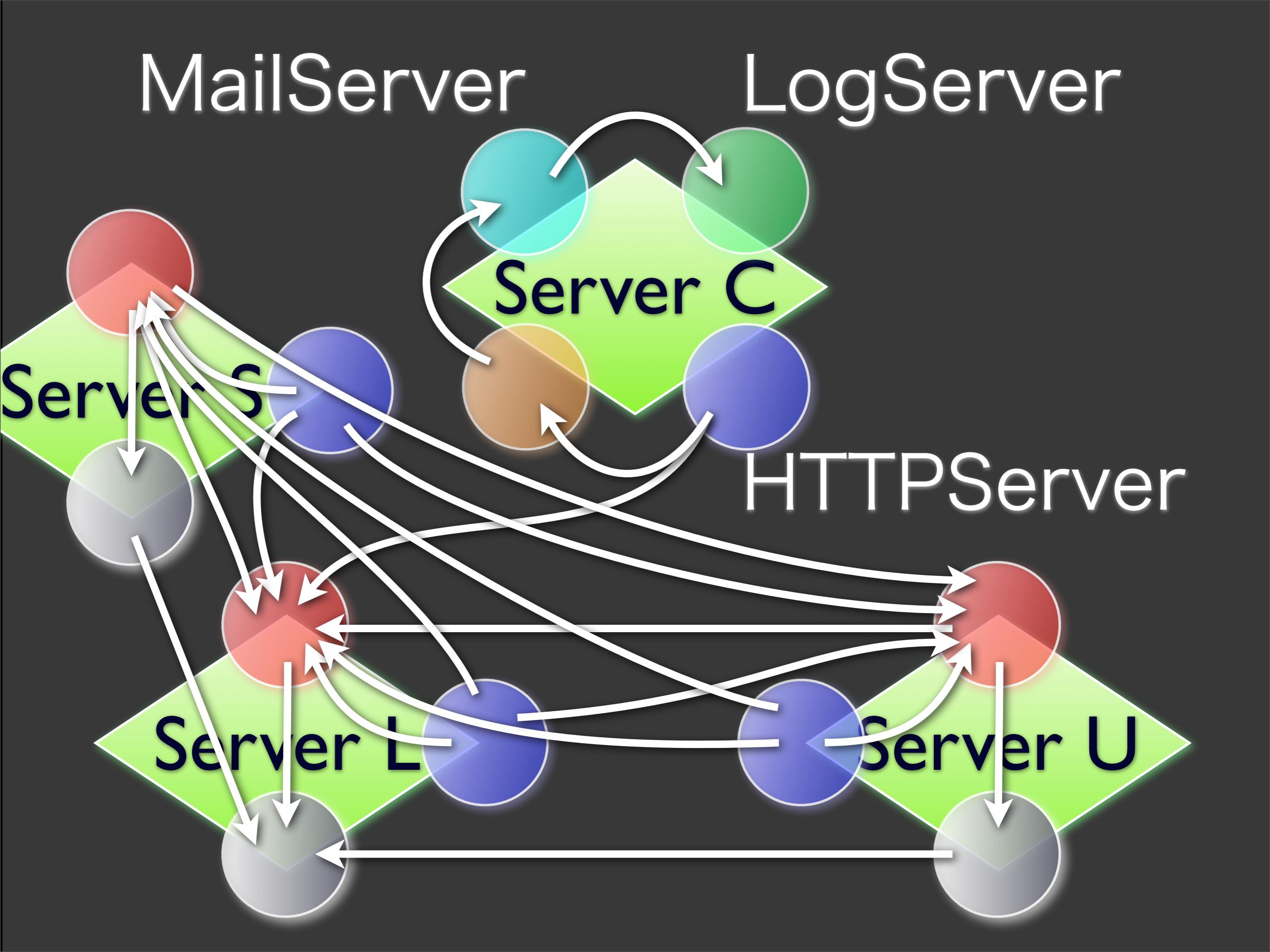
**まだまだ開発を
続けています**

分散ネットワークを自己組織化



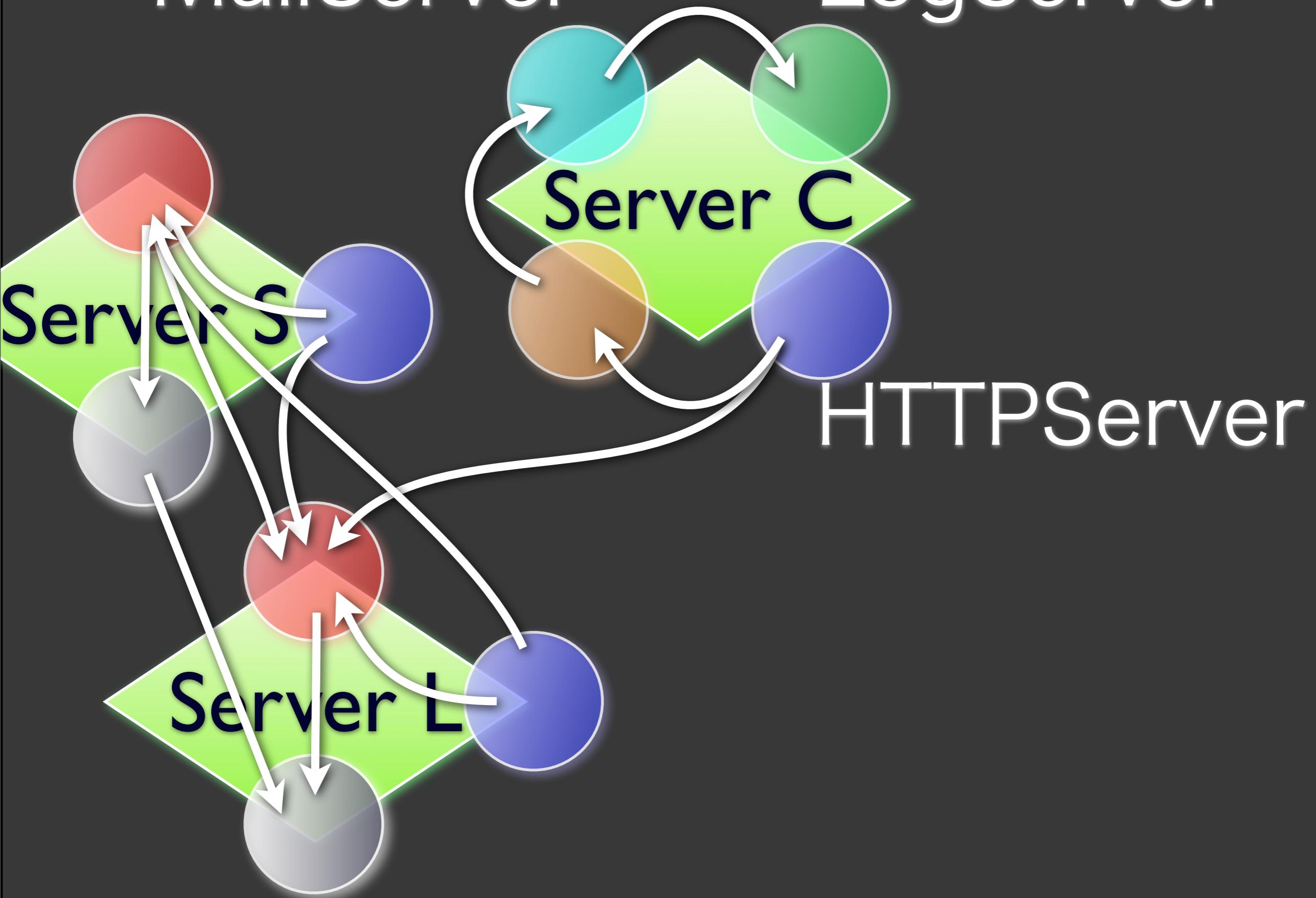






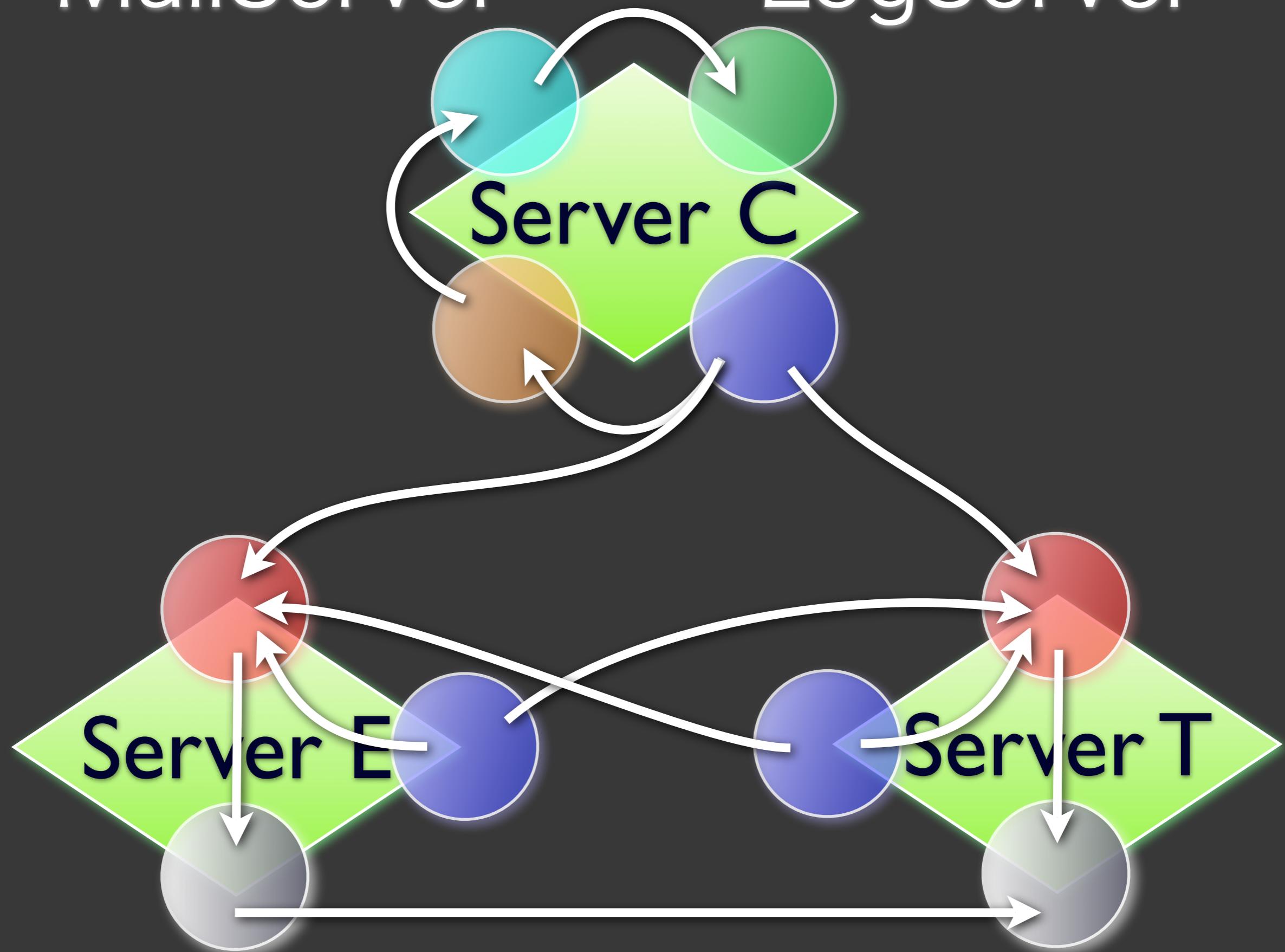
MailServer

LogServer



MailServer

LogServer



電源制御

LogServer

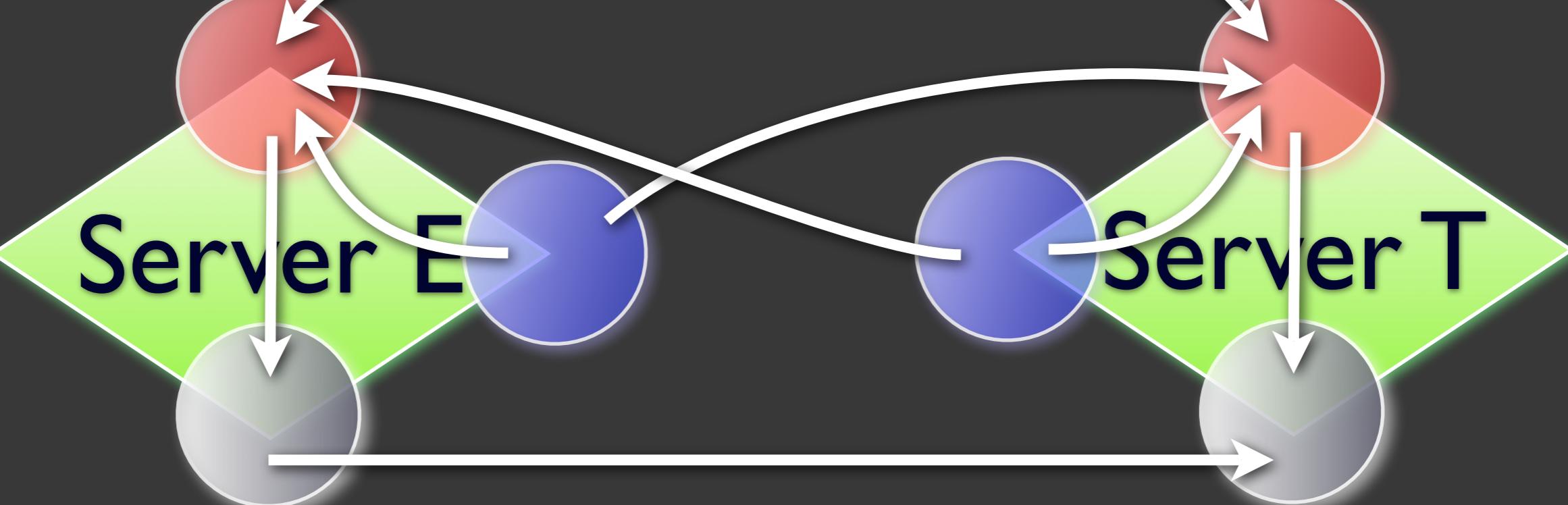
Server R

Server C

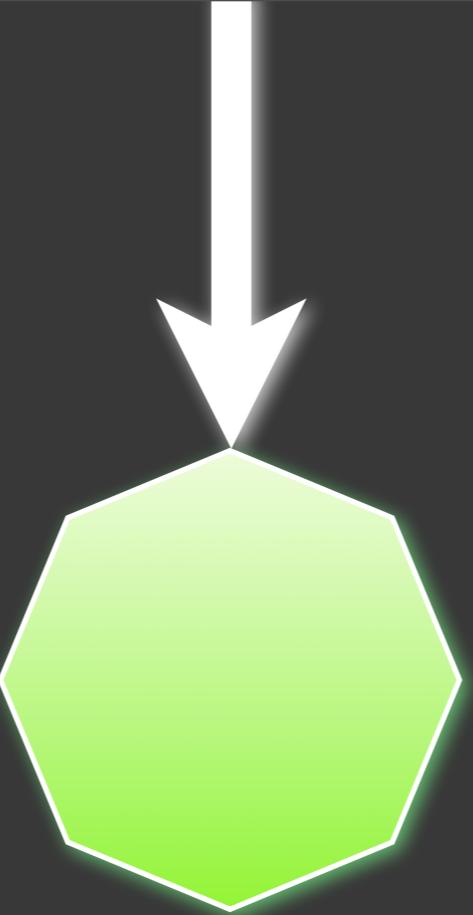
自己最適化

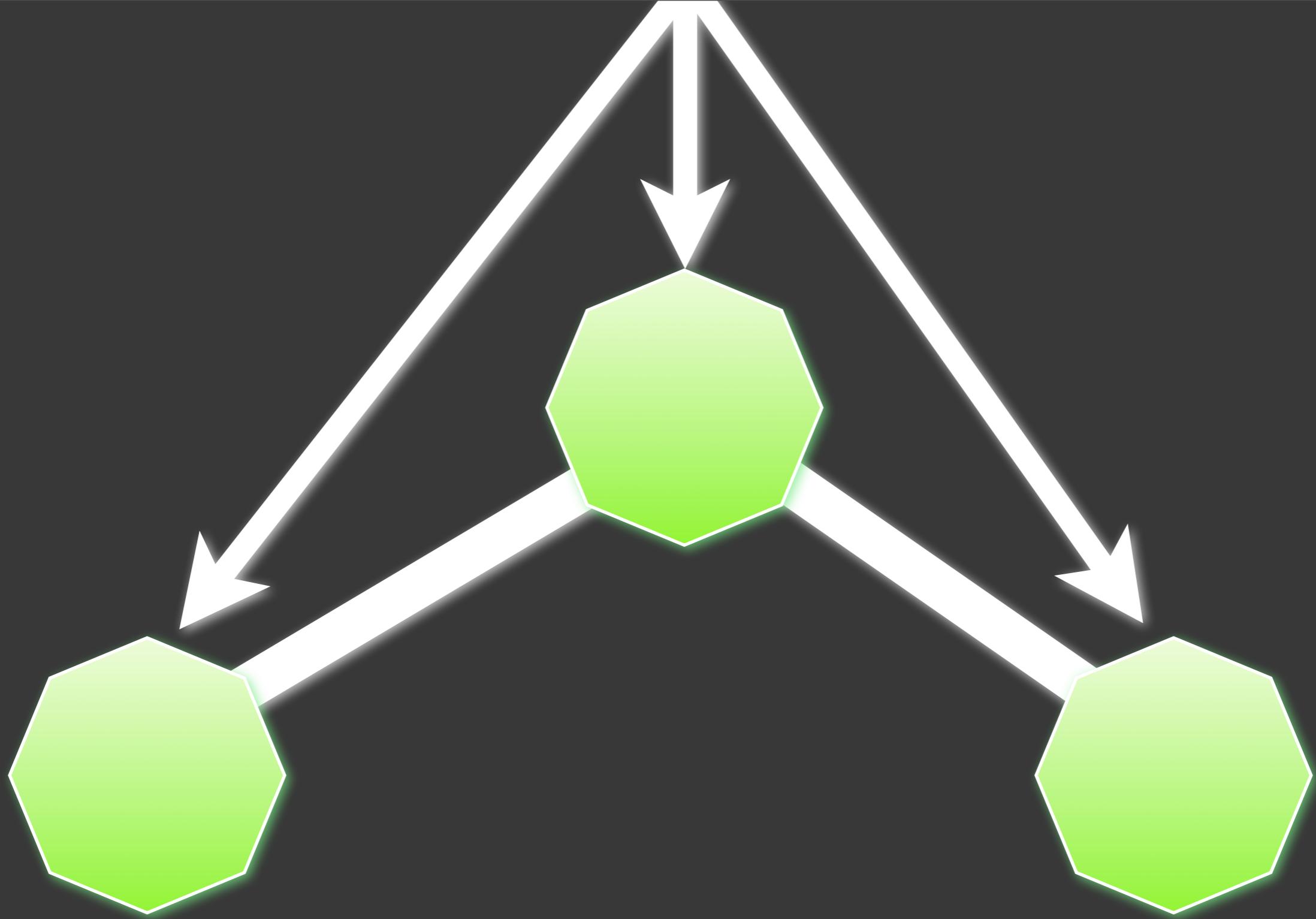
Server E

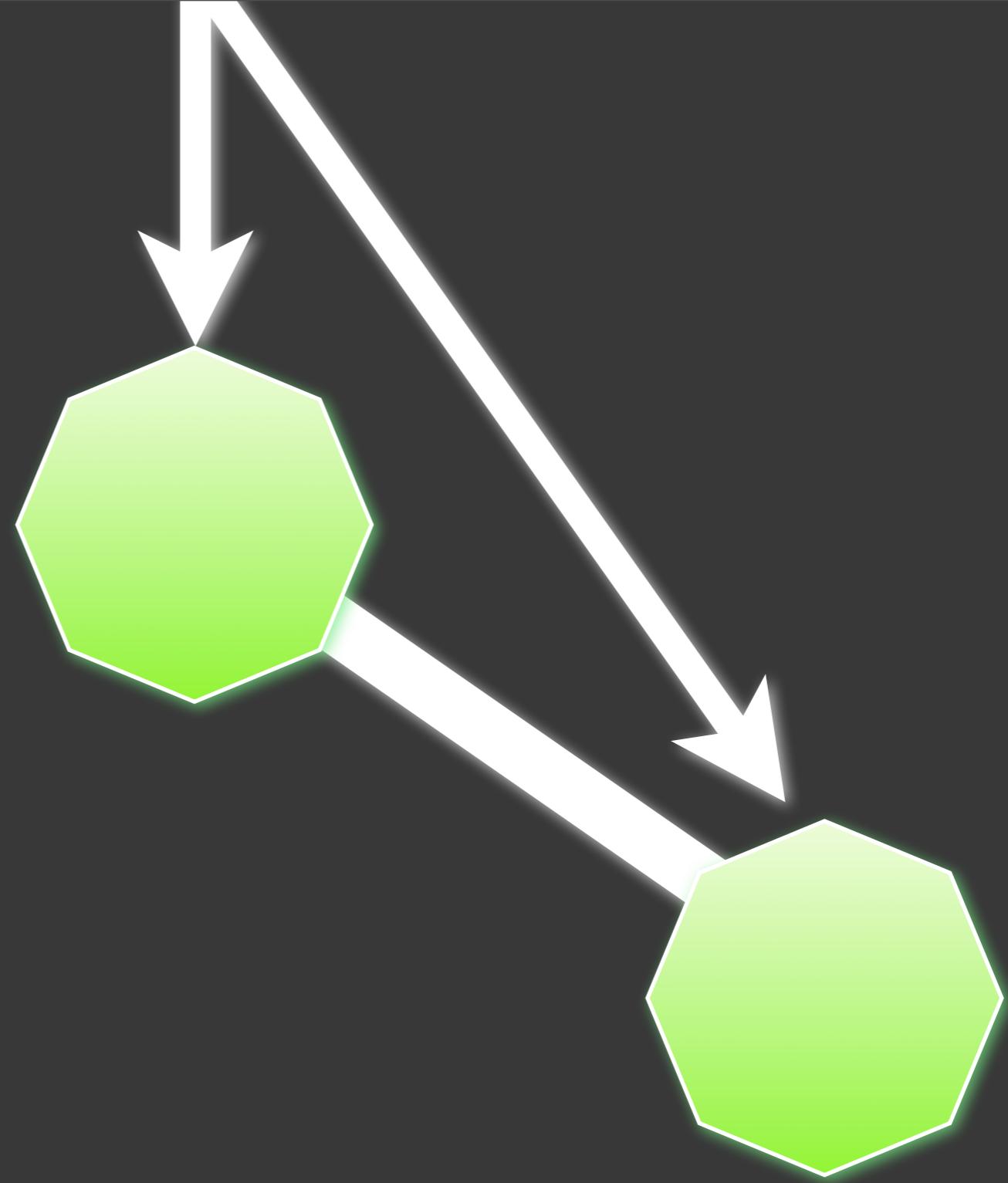
Server T

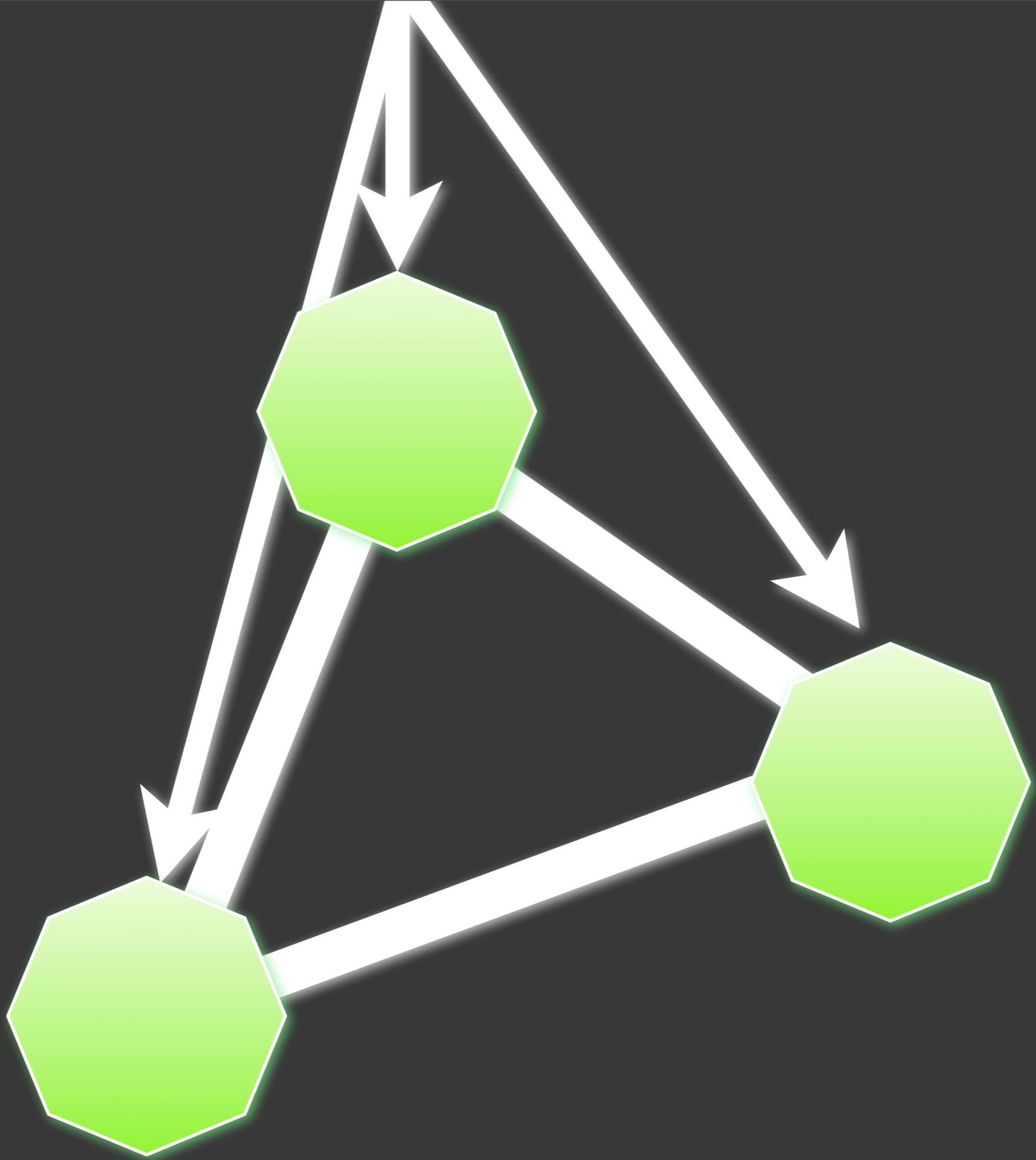


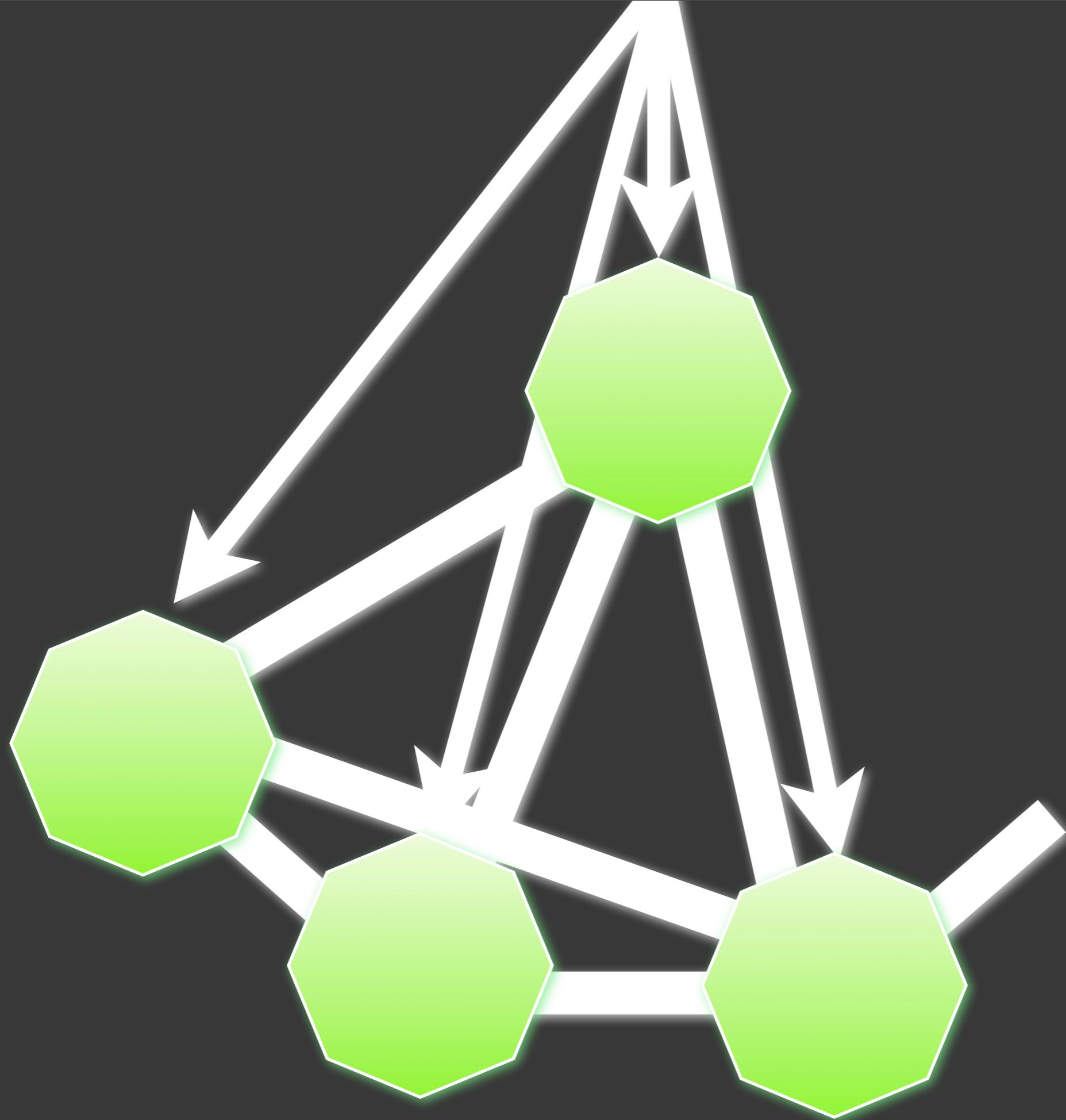
分散ファイルシステム







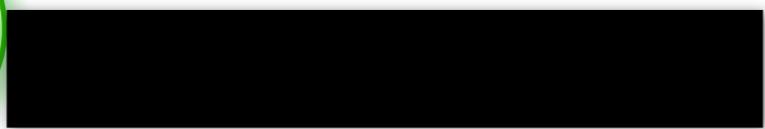
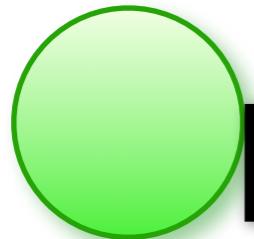


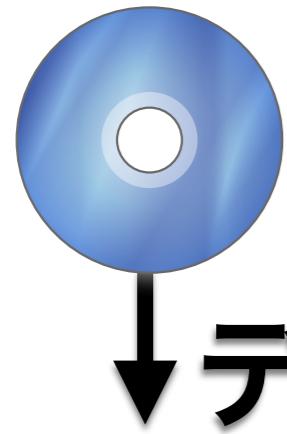


ご静聴ありがとうございました。

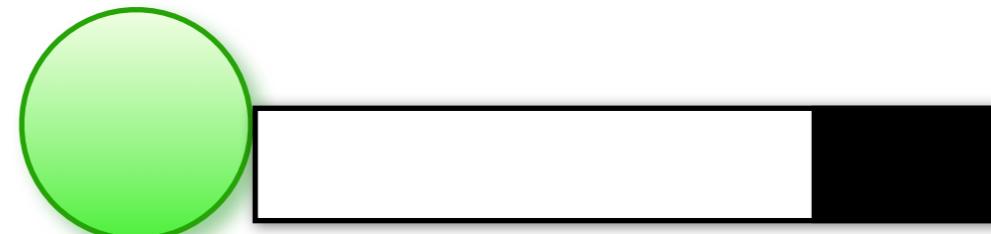
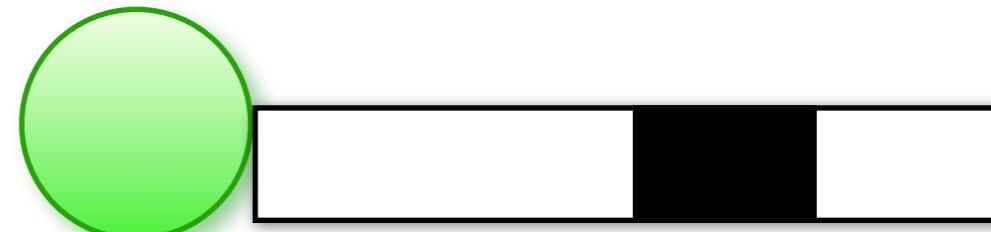
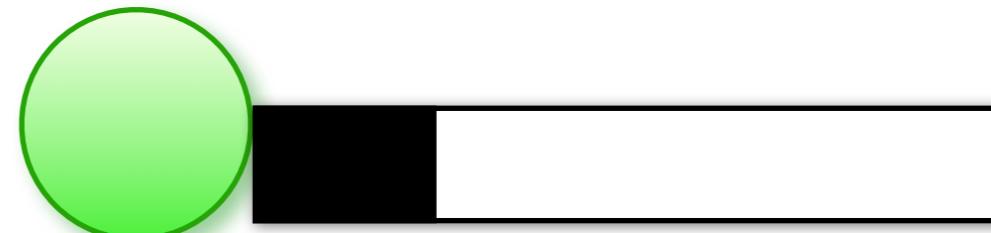
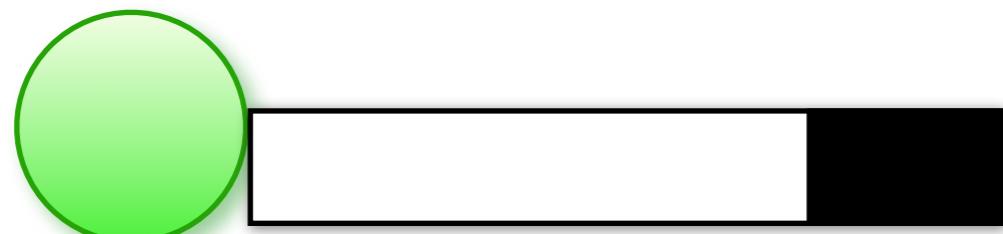
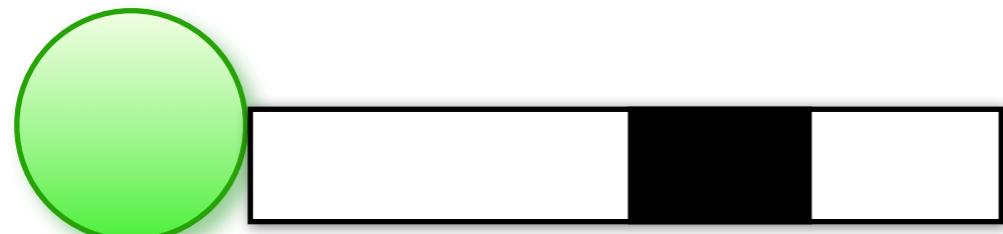


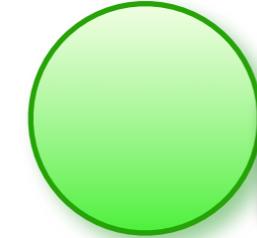
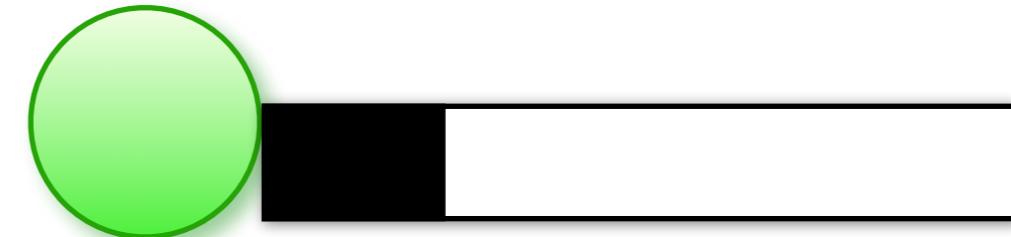
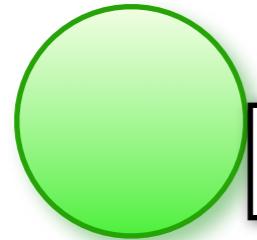
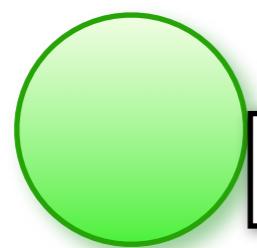
↓ディスクイメージ





ディスクイメージ





分散ディスク共有

プロトコル

TCP

Join

ネットワークに参加する

Get Data

データをダウンロードする

UDP

Notify Up

持っているデータ範囲を知らせる

Notify Down

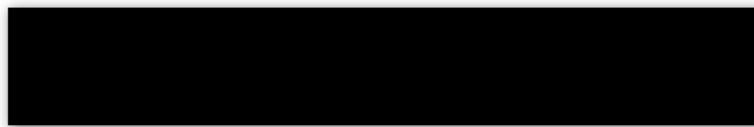
今からダウンすることを伝える

Find

データを持っていないか聞く

分散ディスク共有

ノードの起動



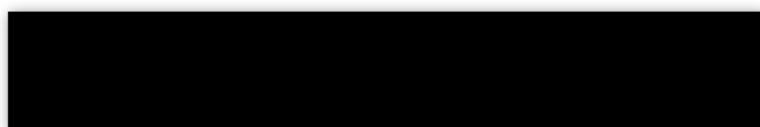
データ



V-Table

分散ディスク共有

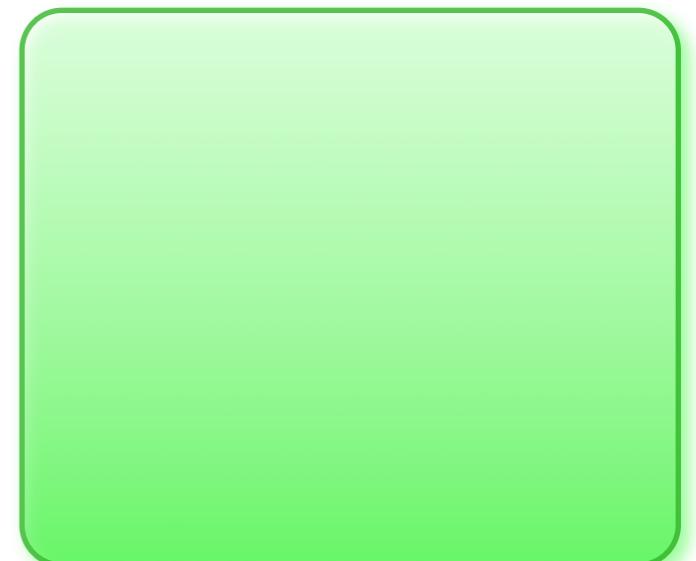
ノードの起動



データ

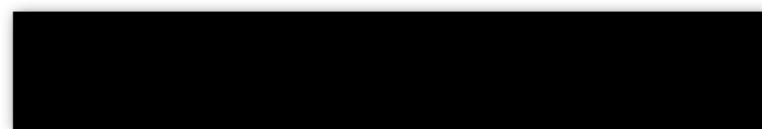


V-Table



分散ディスク共有

ノードの起動



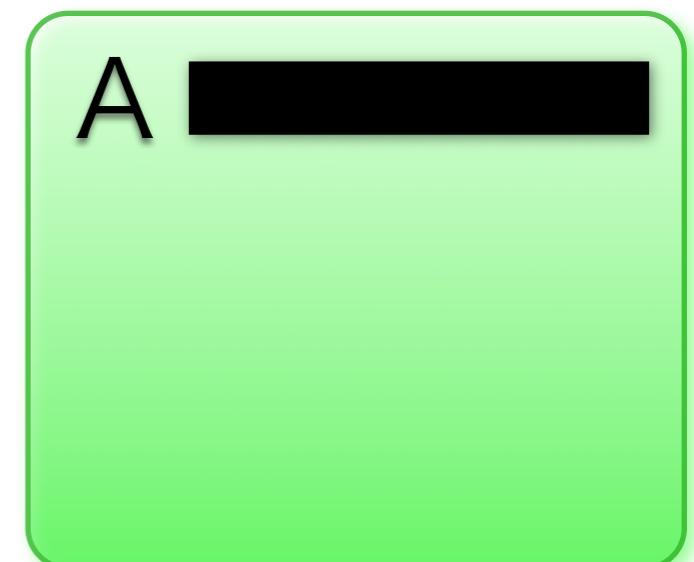
データ



Join

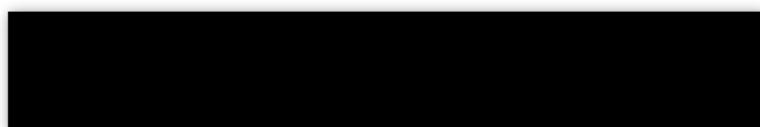


V-Table

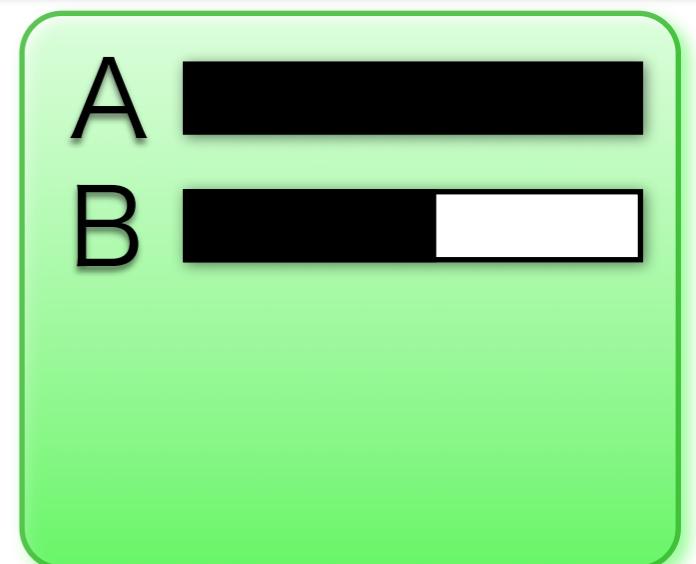


分散ディスク共有

ノードの起動

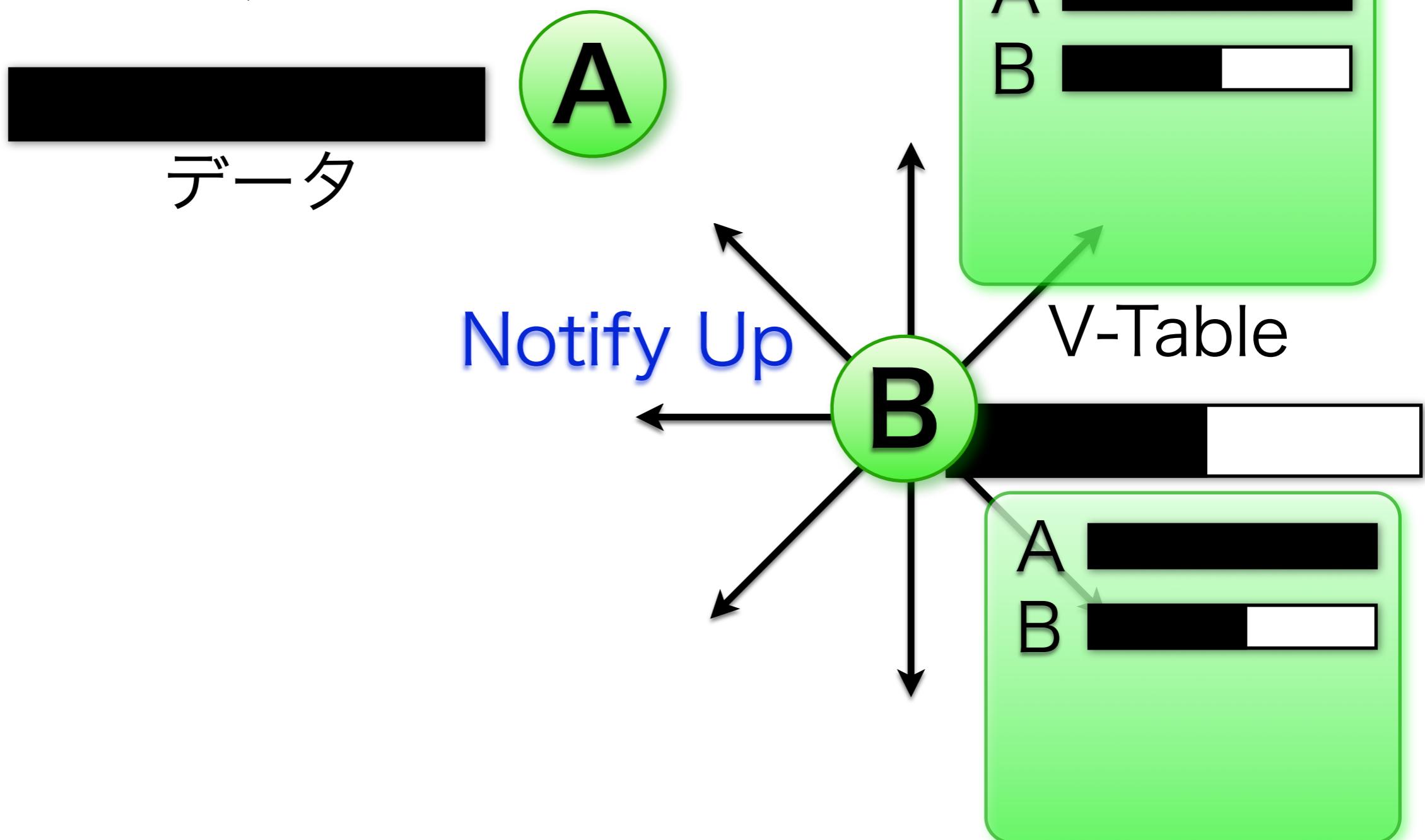


データ



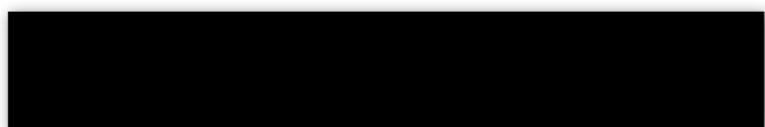
分散ディスク共有

ノードの起動

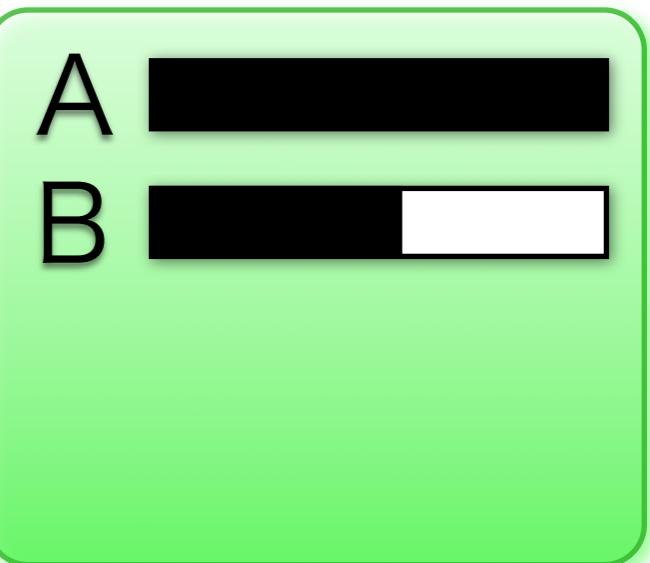
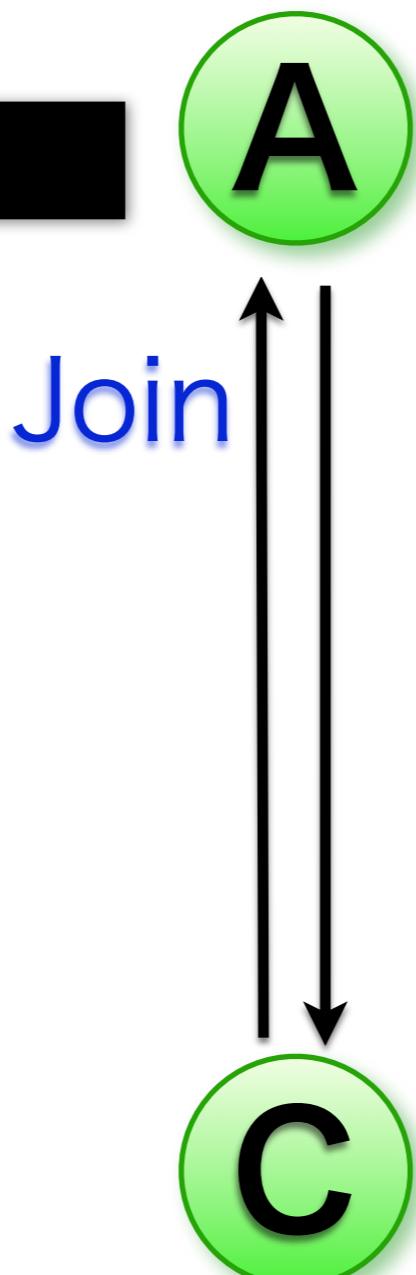


分散ディスク共有

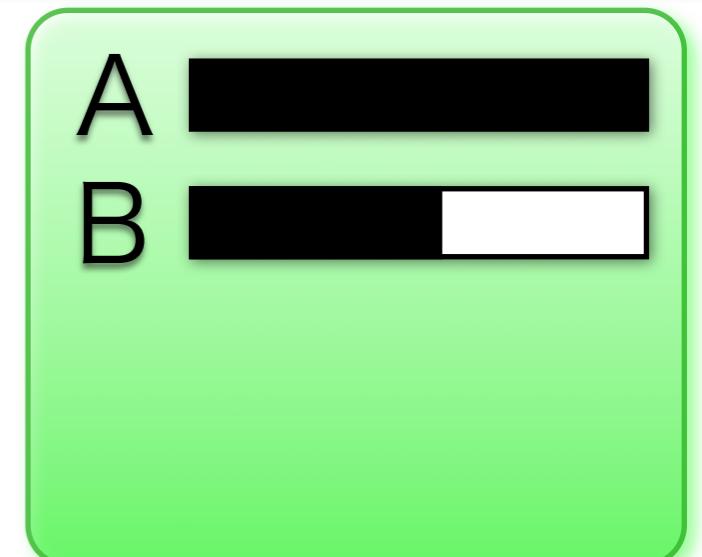
ノードの起動



データ

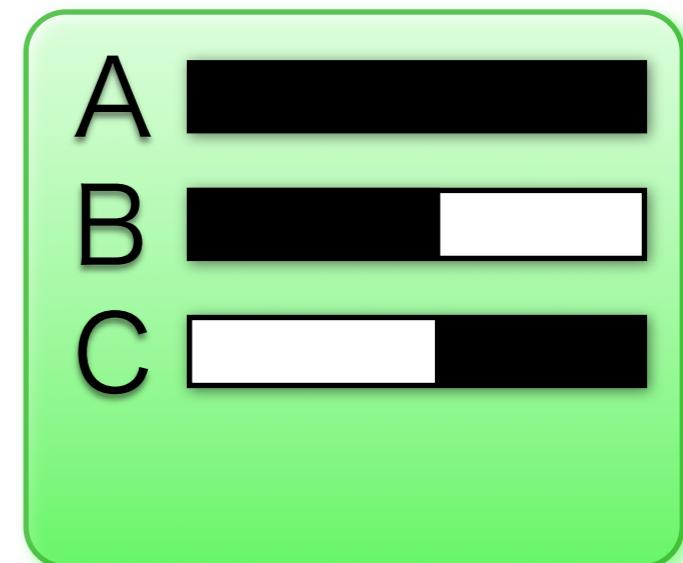


V-Table

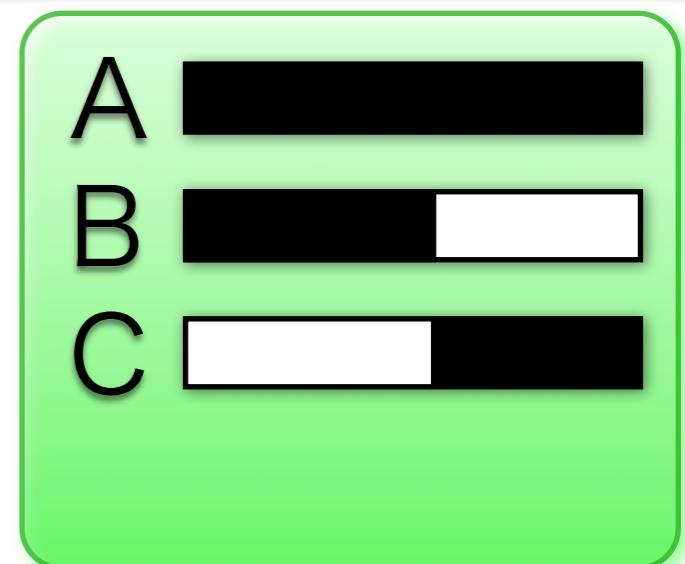
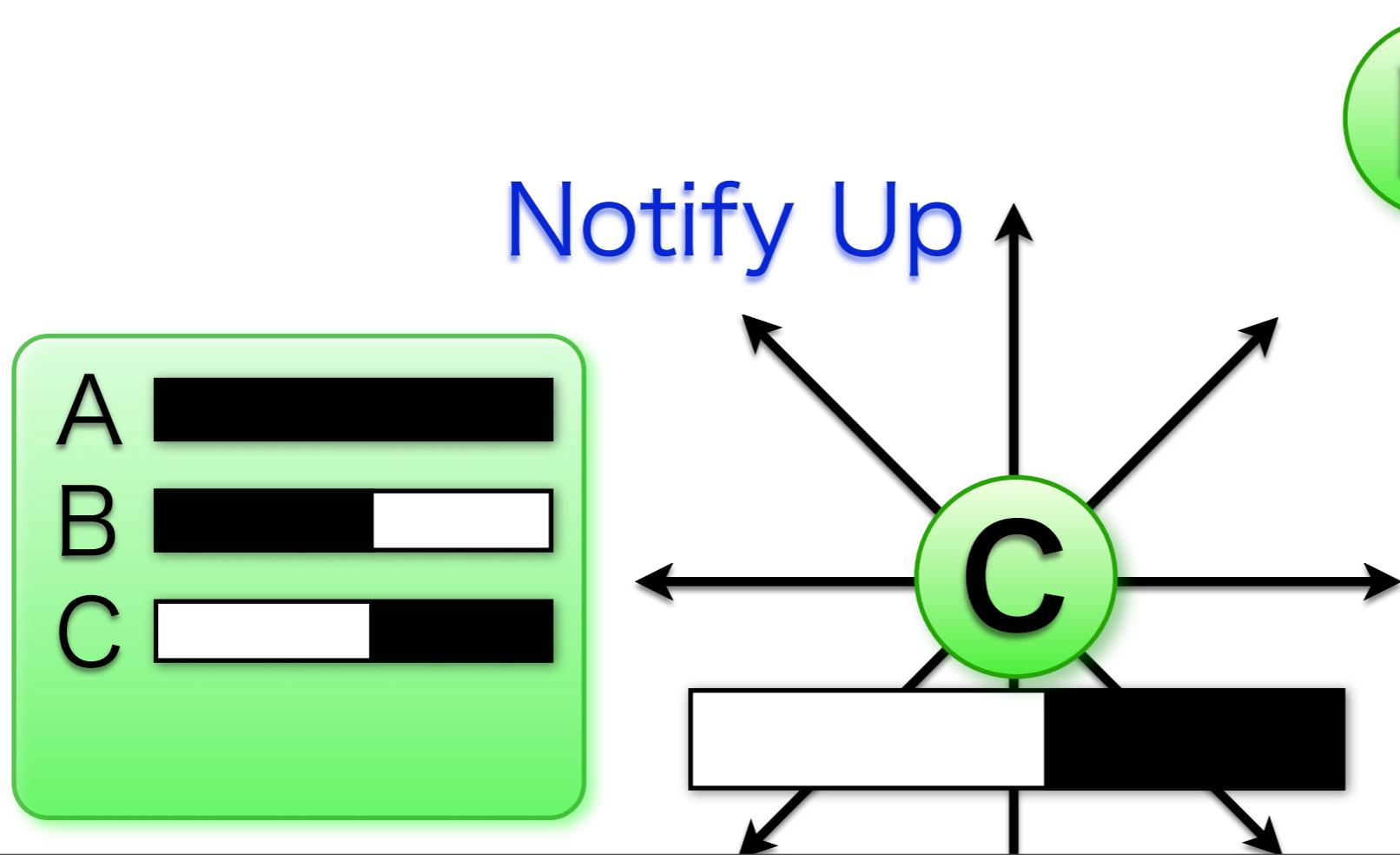


分散ディスク共有

ノードの起動

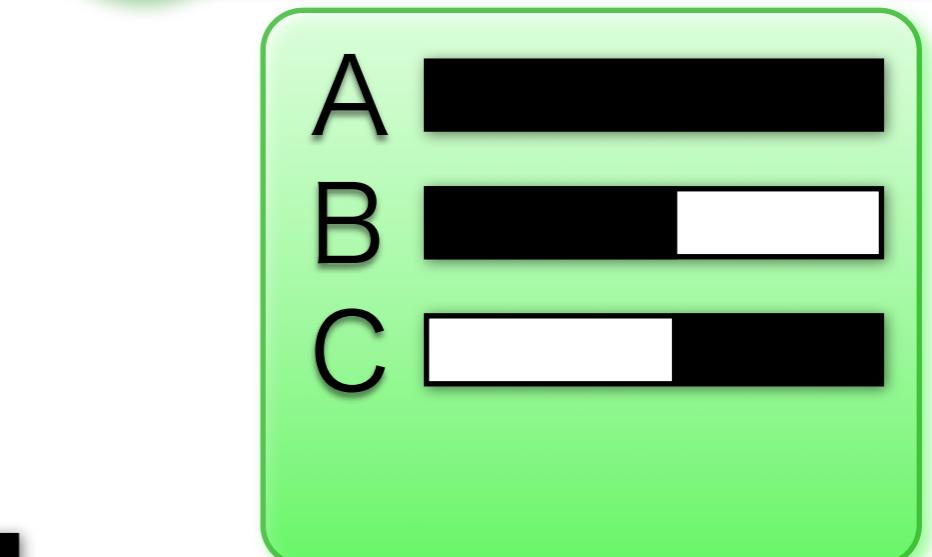
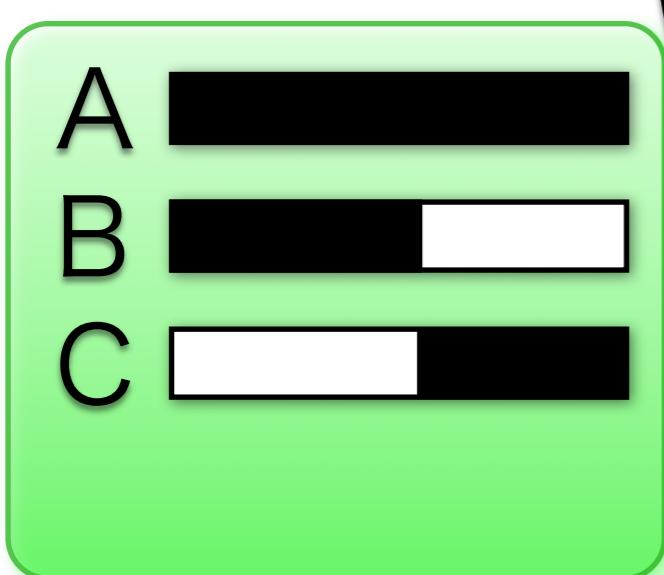
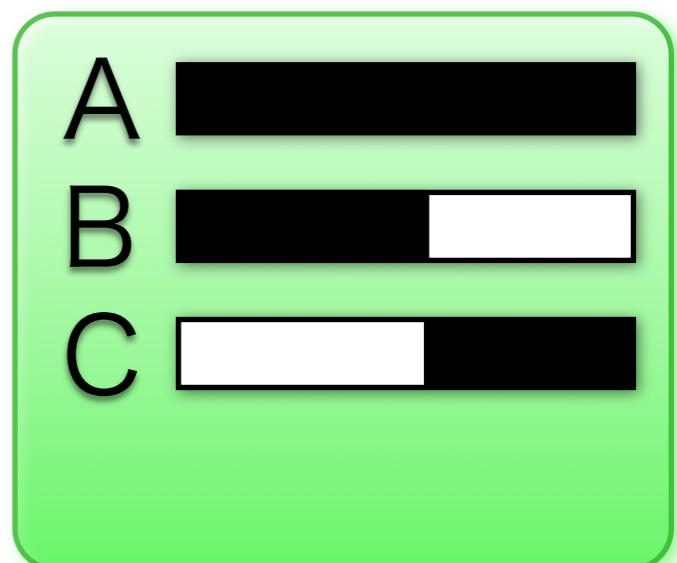


V-Table



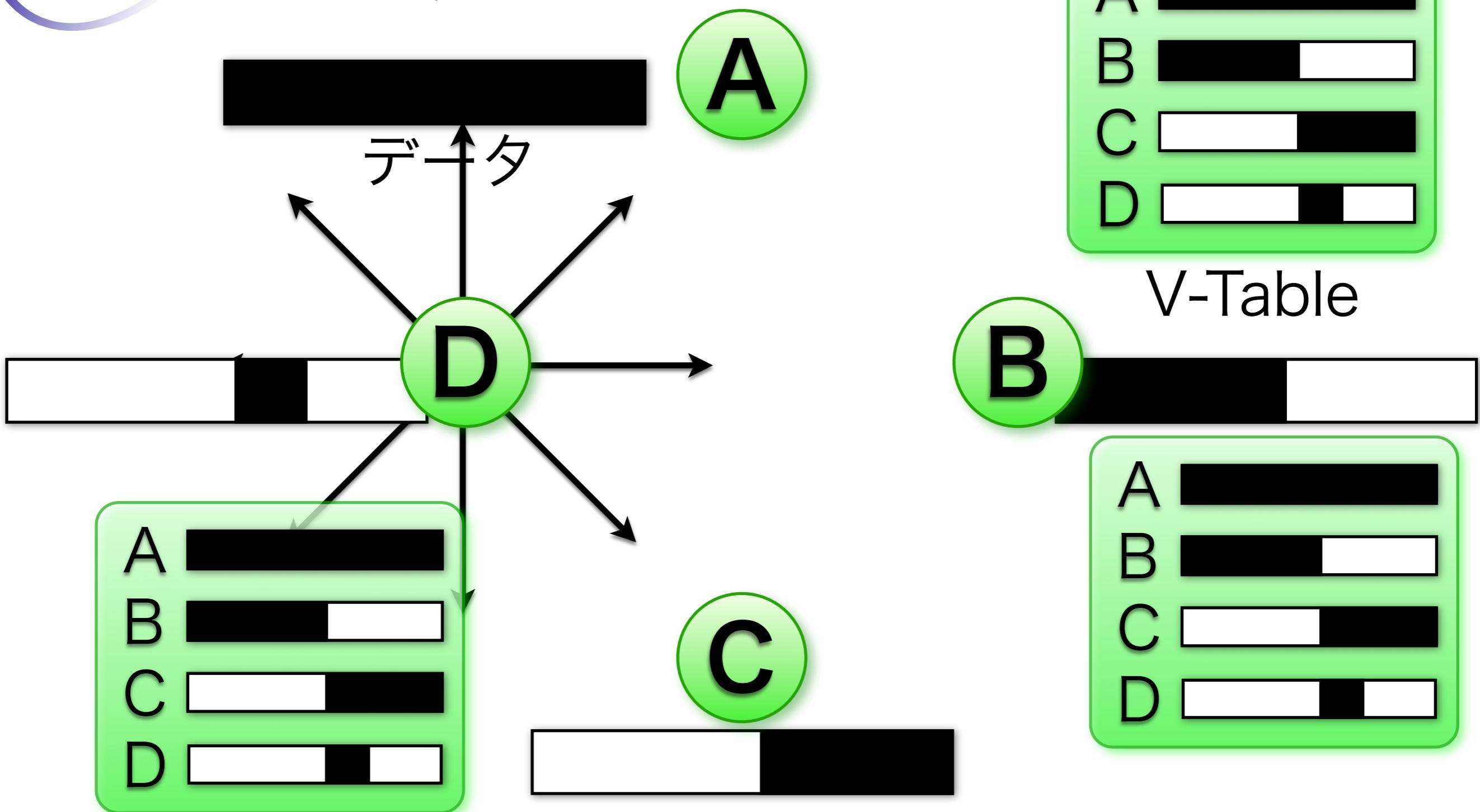
分散ディスク共有

ノードの起動



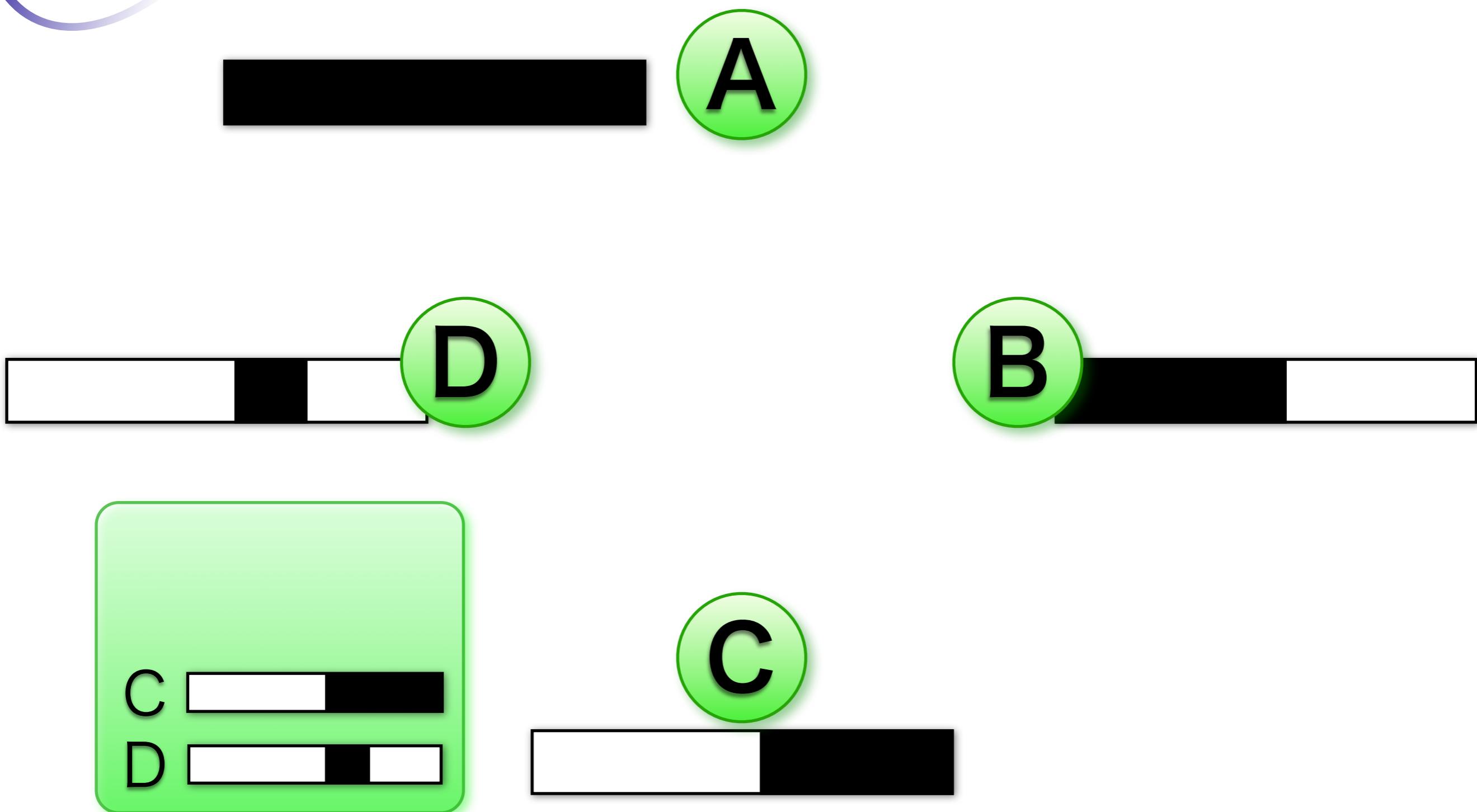
分散ディスク共有

ノードの起動



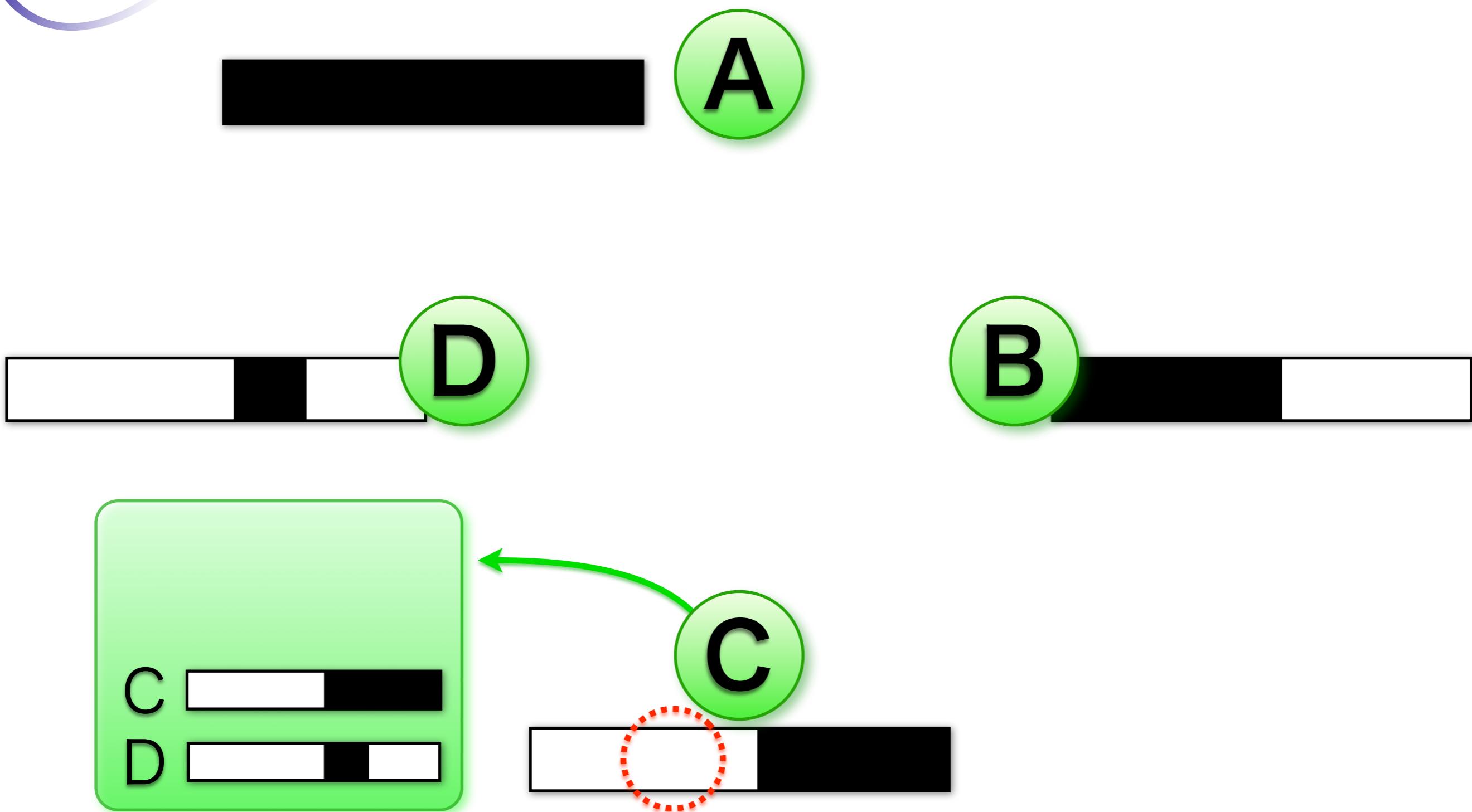
分散ディスク共有

V-Tableの補完



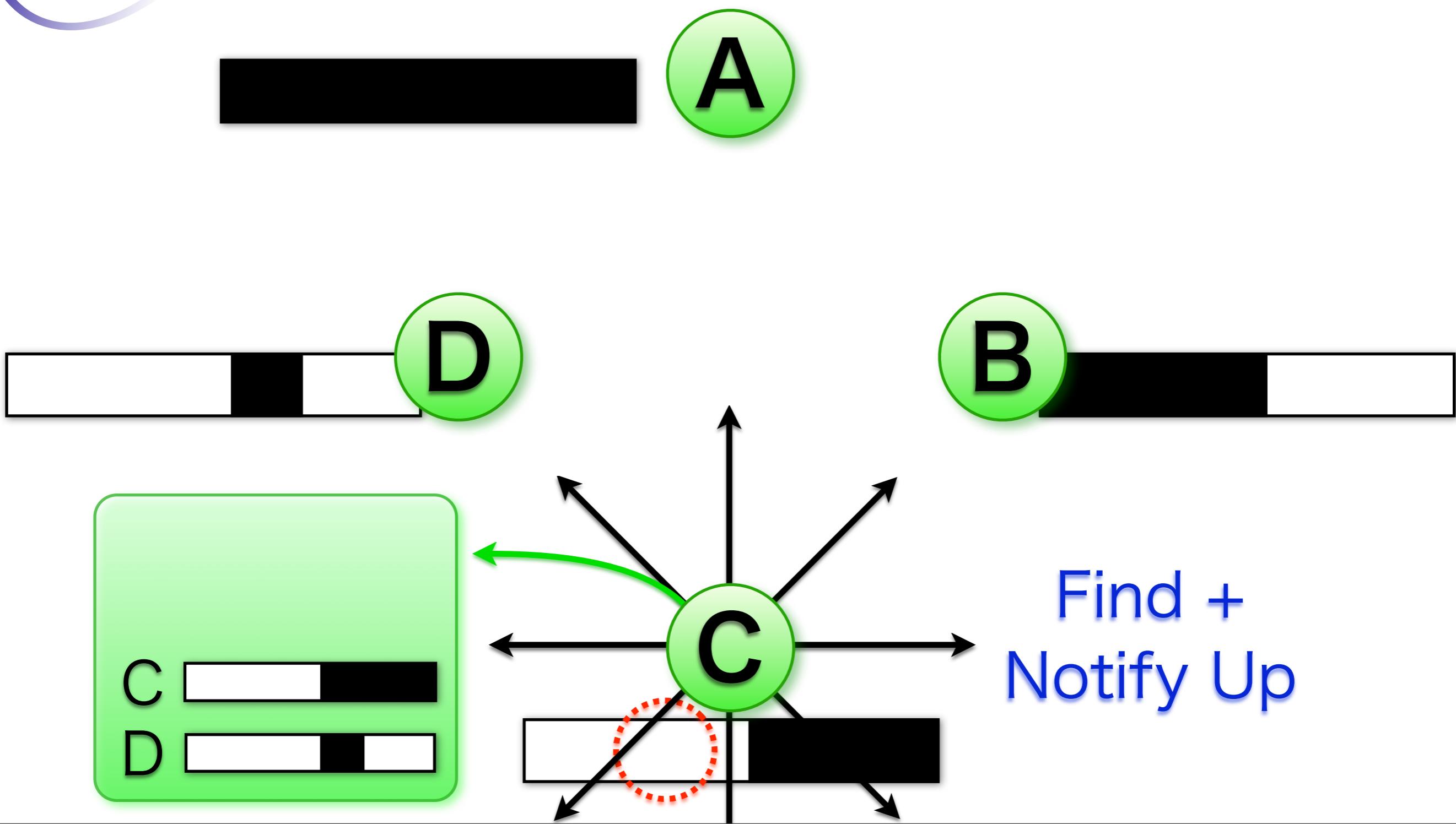
分散ディスク共有

V-Tableの補完



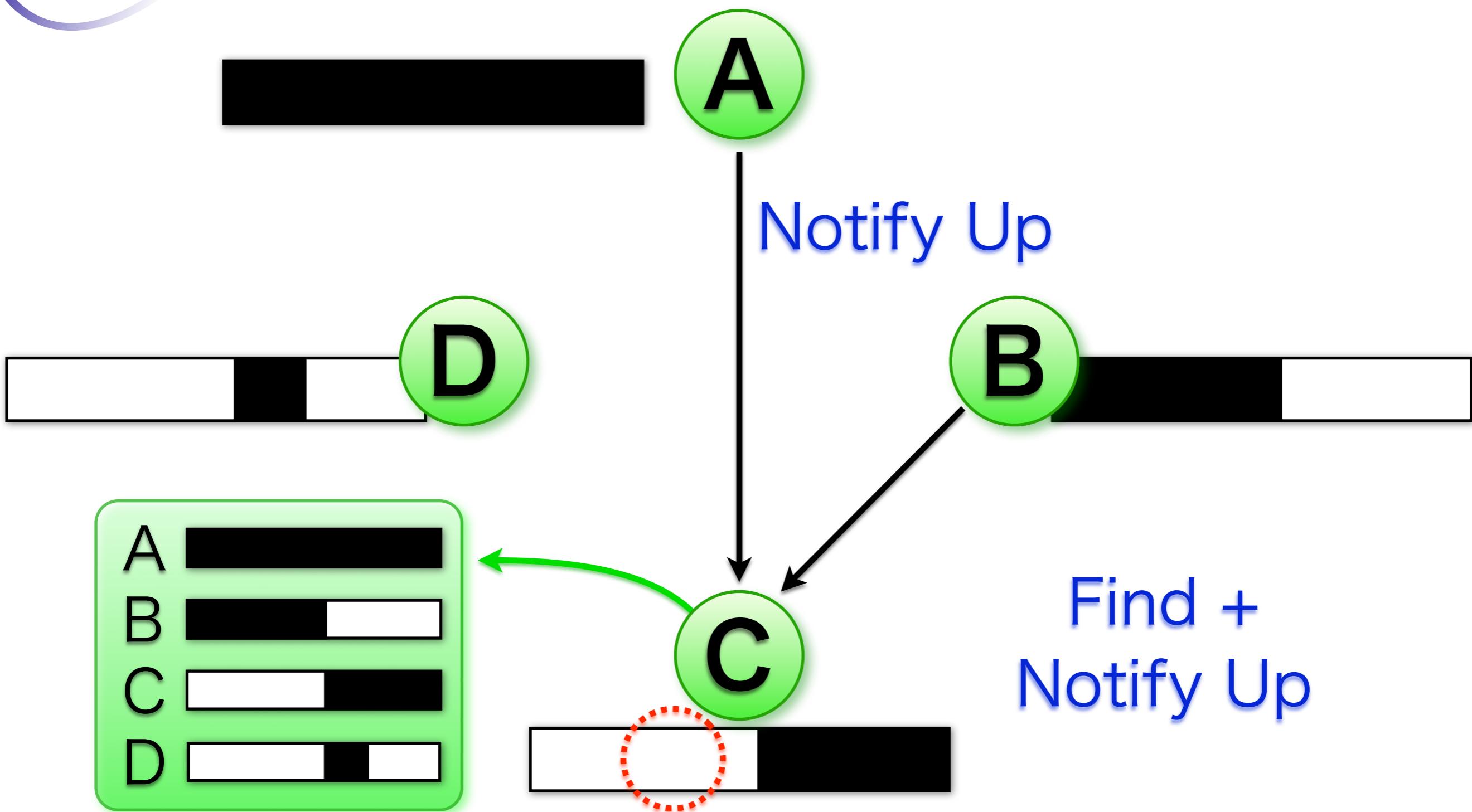
分散ディスク共有

V-Tableの補完



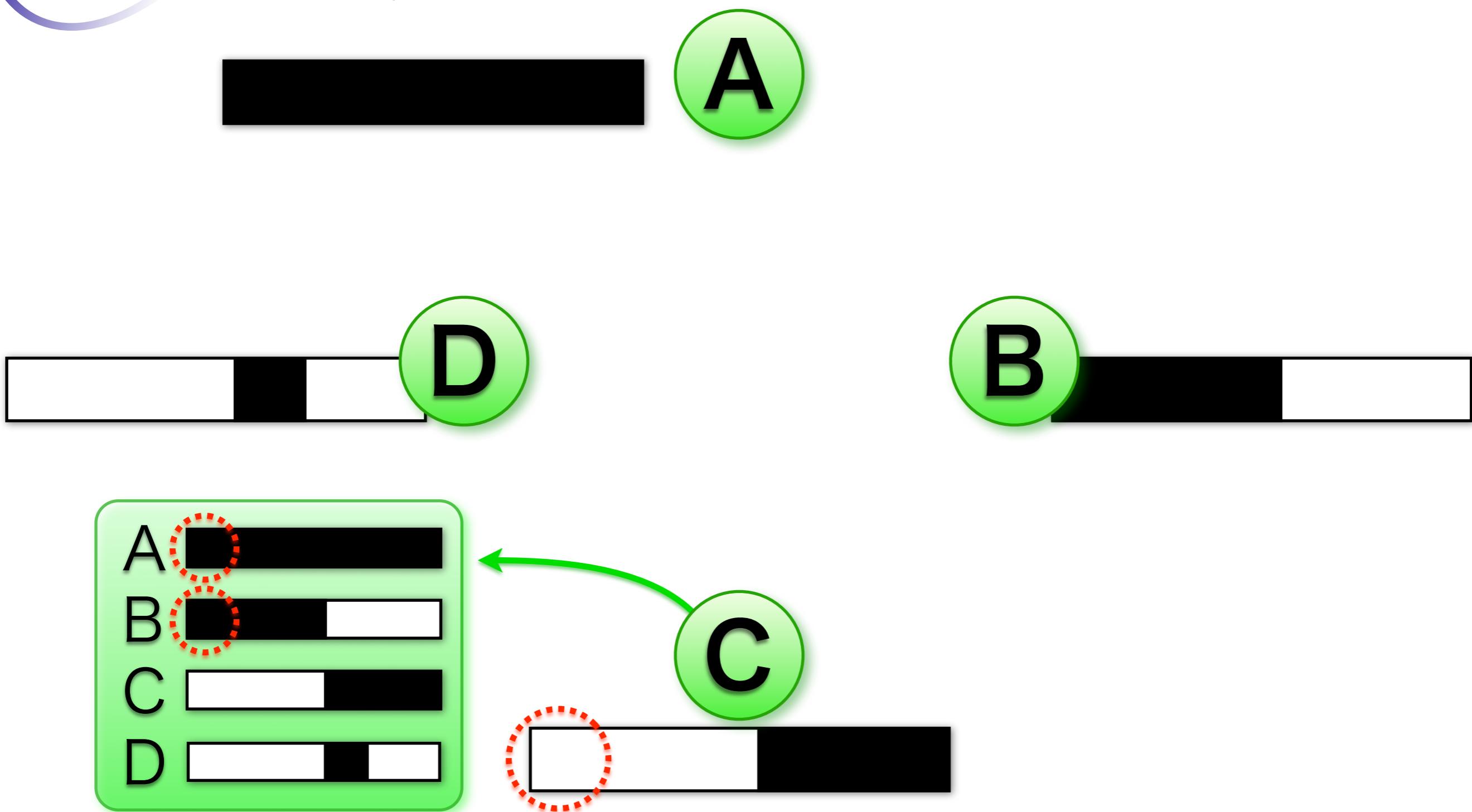
分散ディスク共有

V-Tableの補完



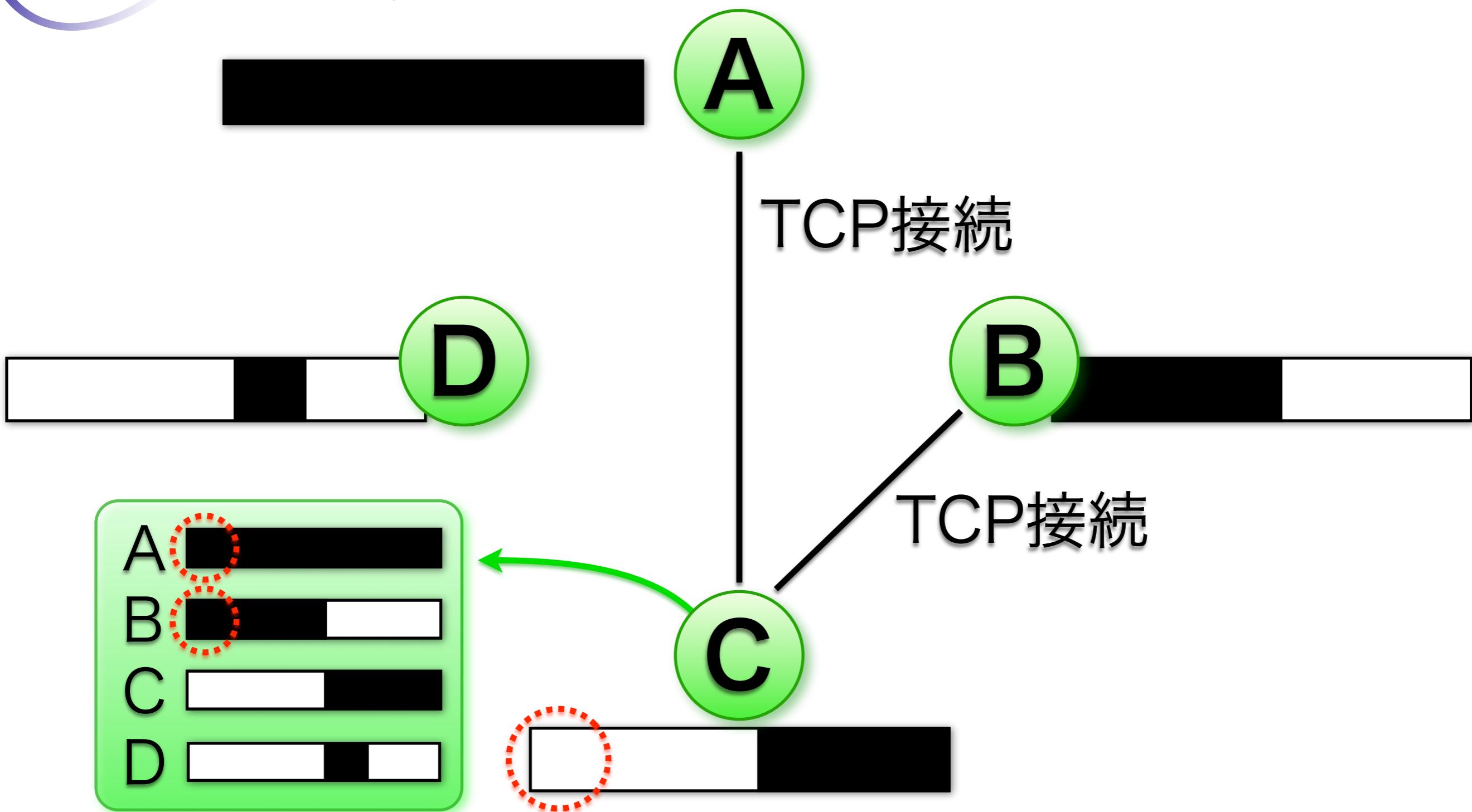
分散ディスク共有

データの取得



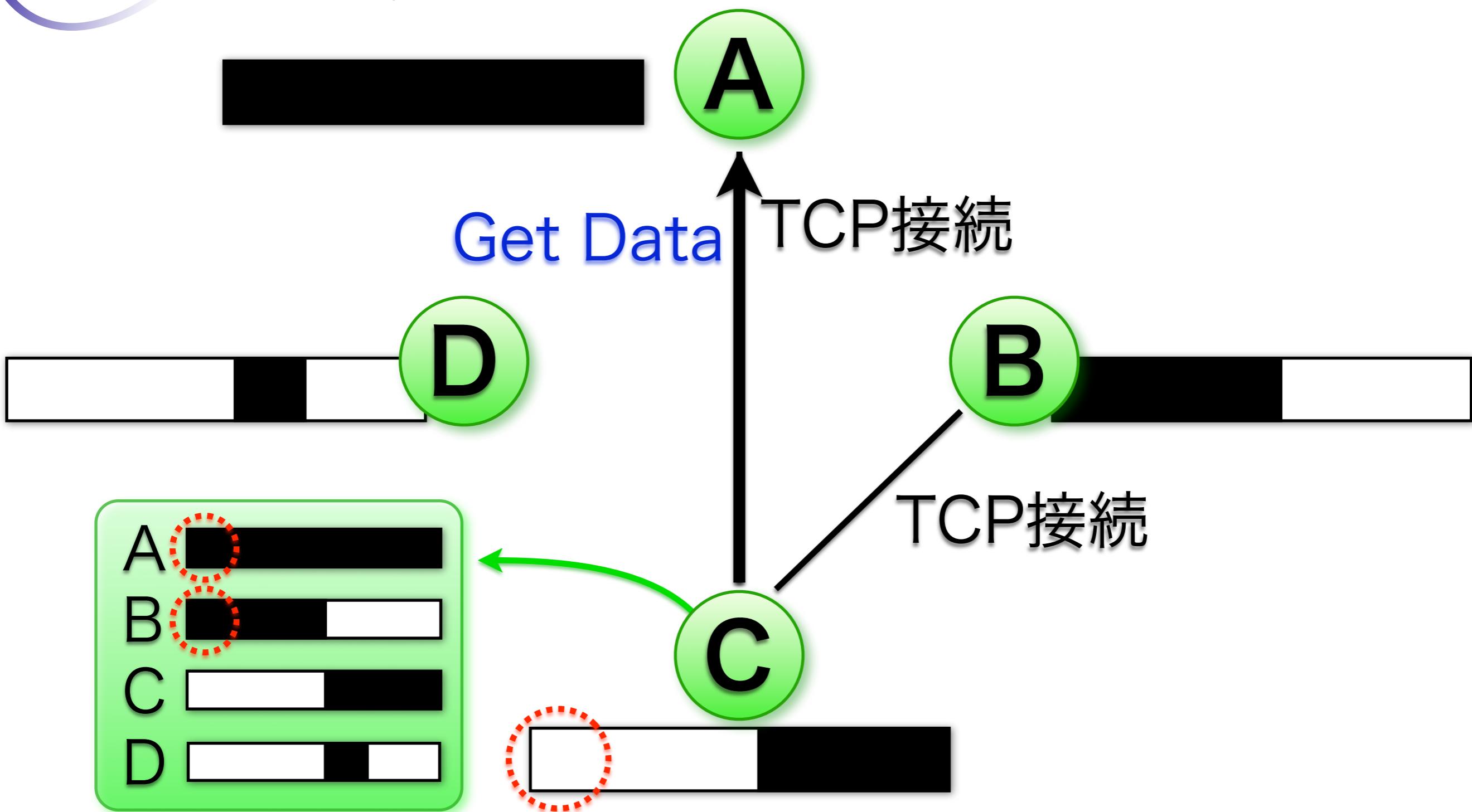
分散ディスク共有

データの取得



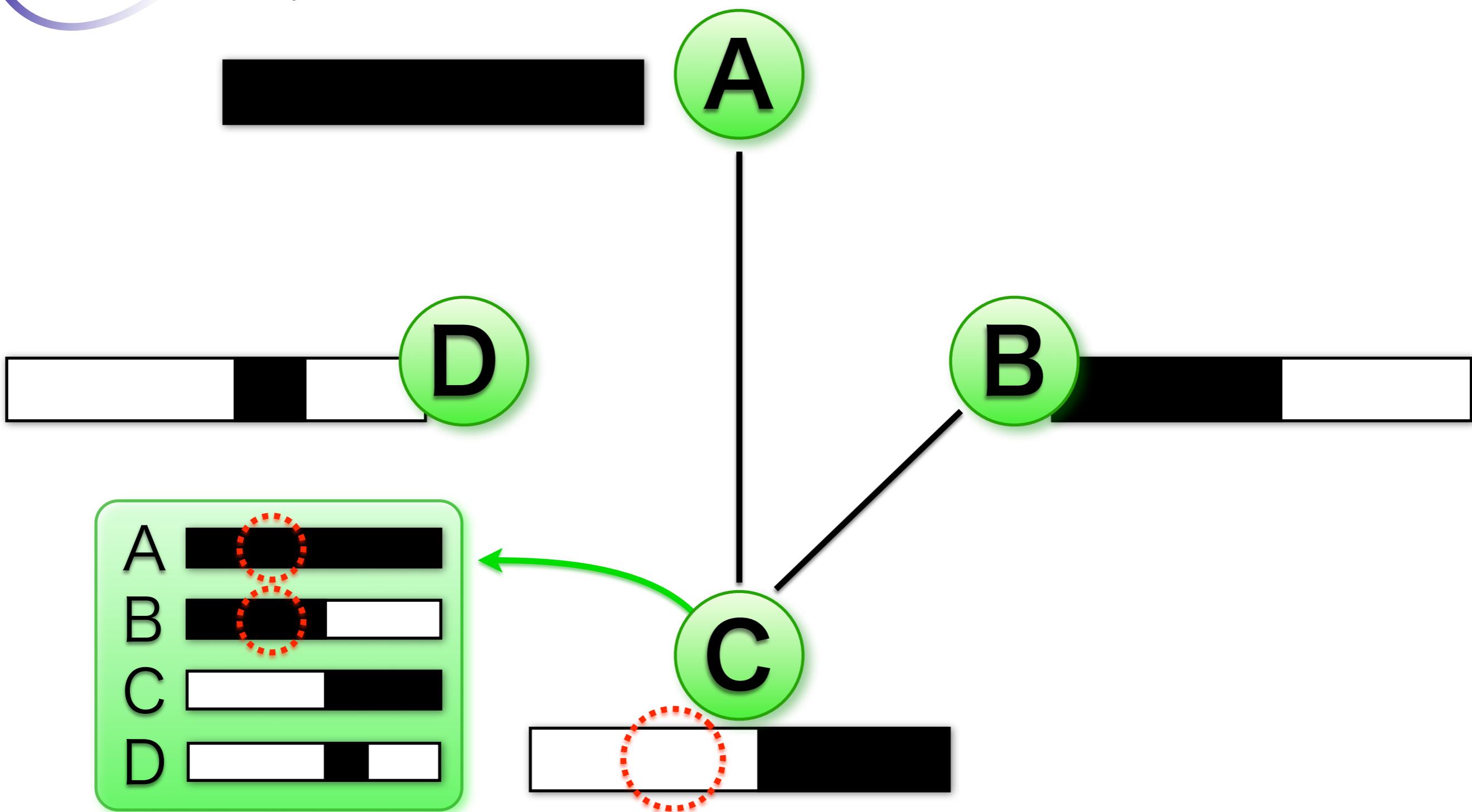
分散ディスク共有

データの取得



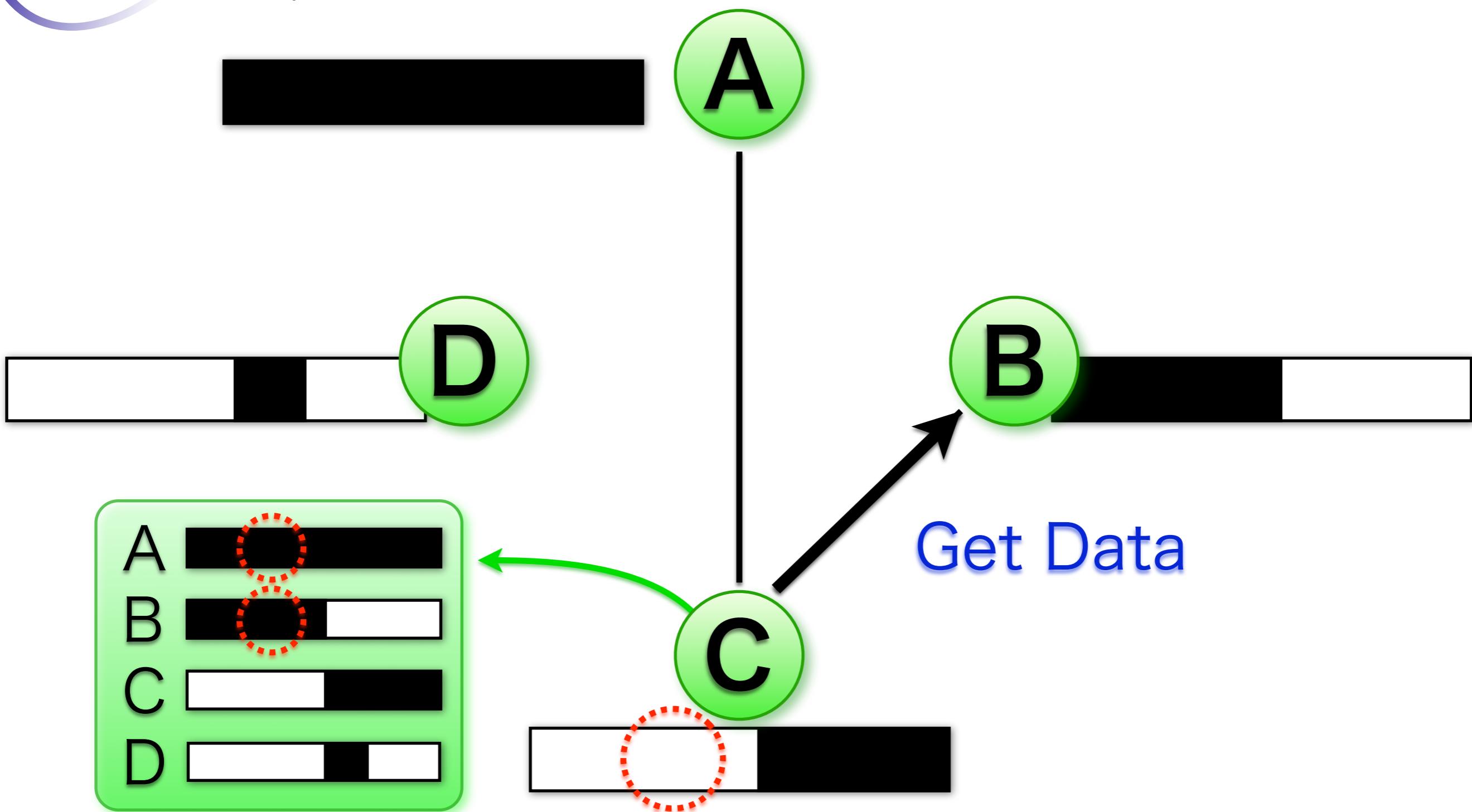
分散ディスク共有

TCP接続の維持



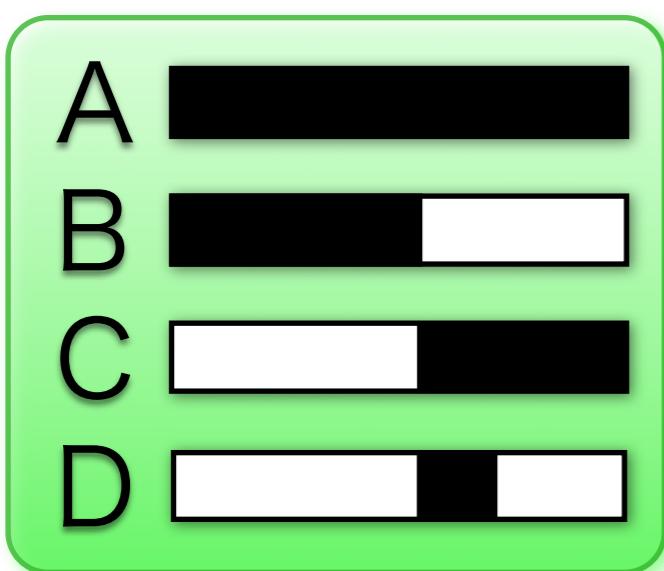
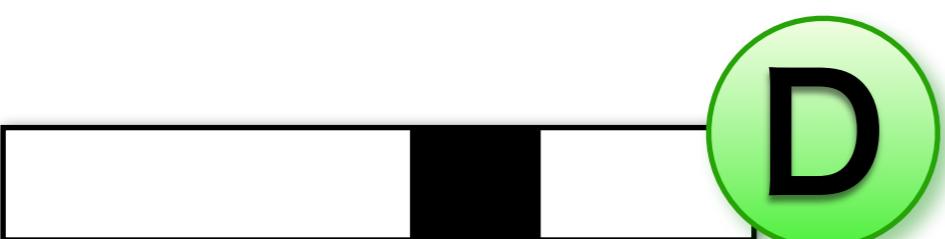
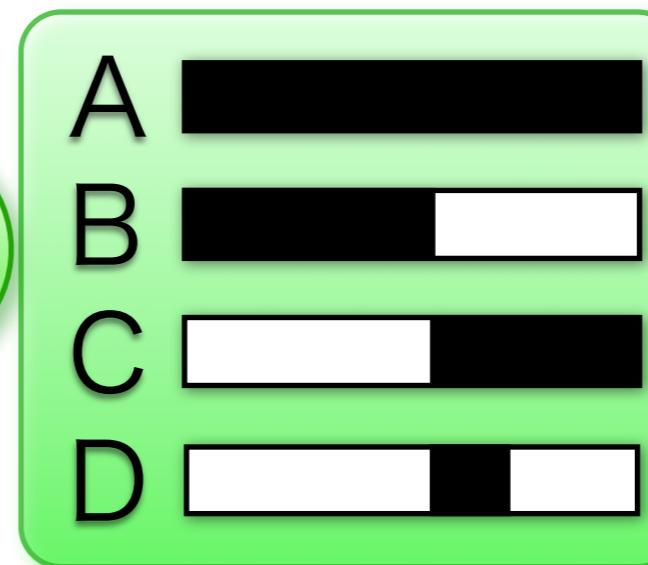
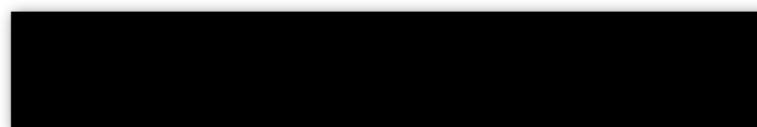
分散ディスク共有

TCP接続の維持



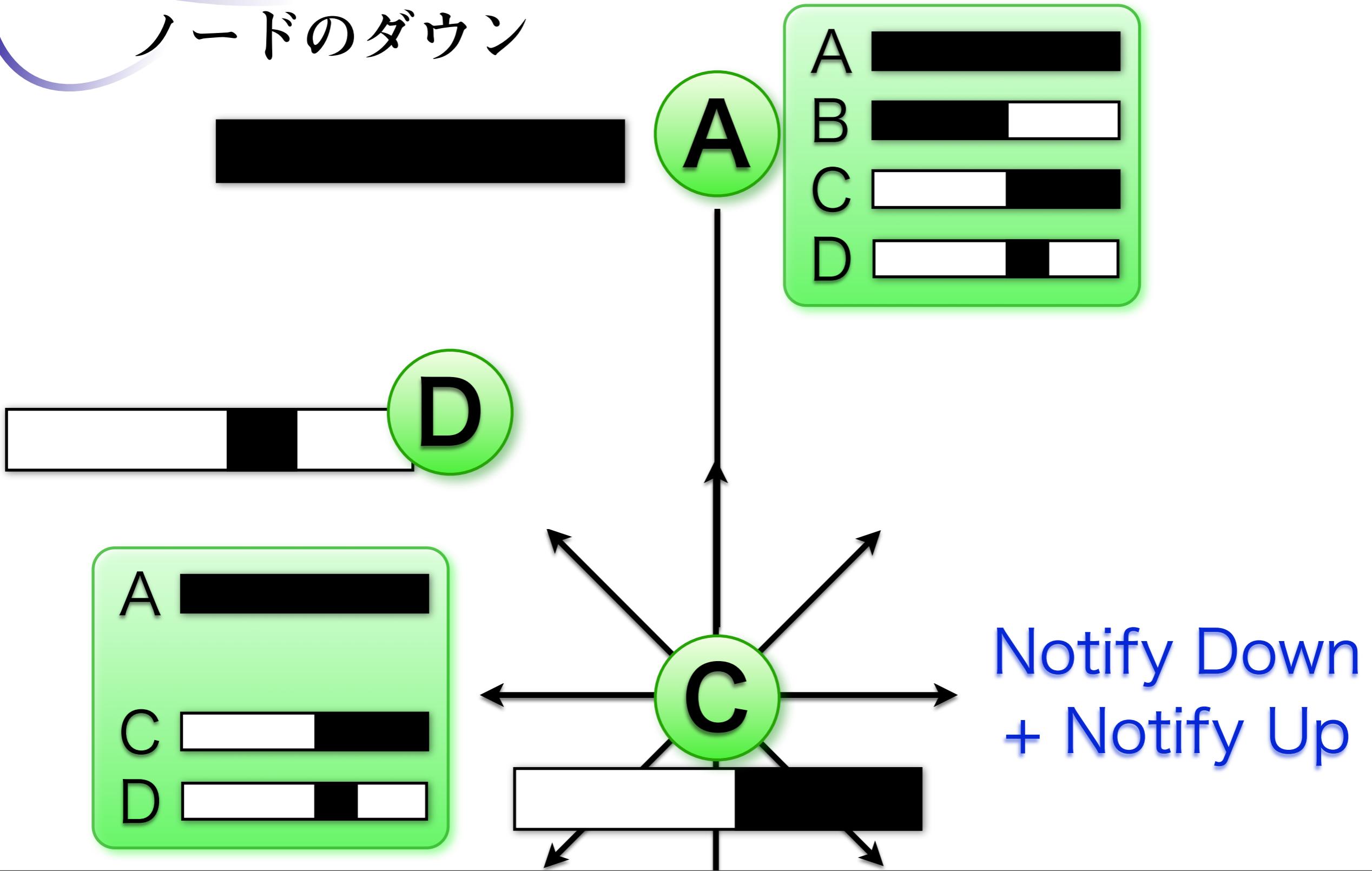
分散ディスク共有

ノードのダウン



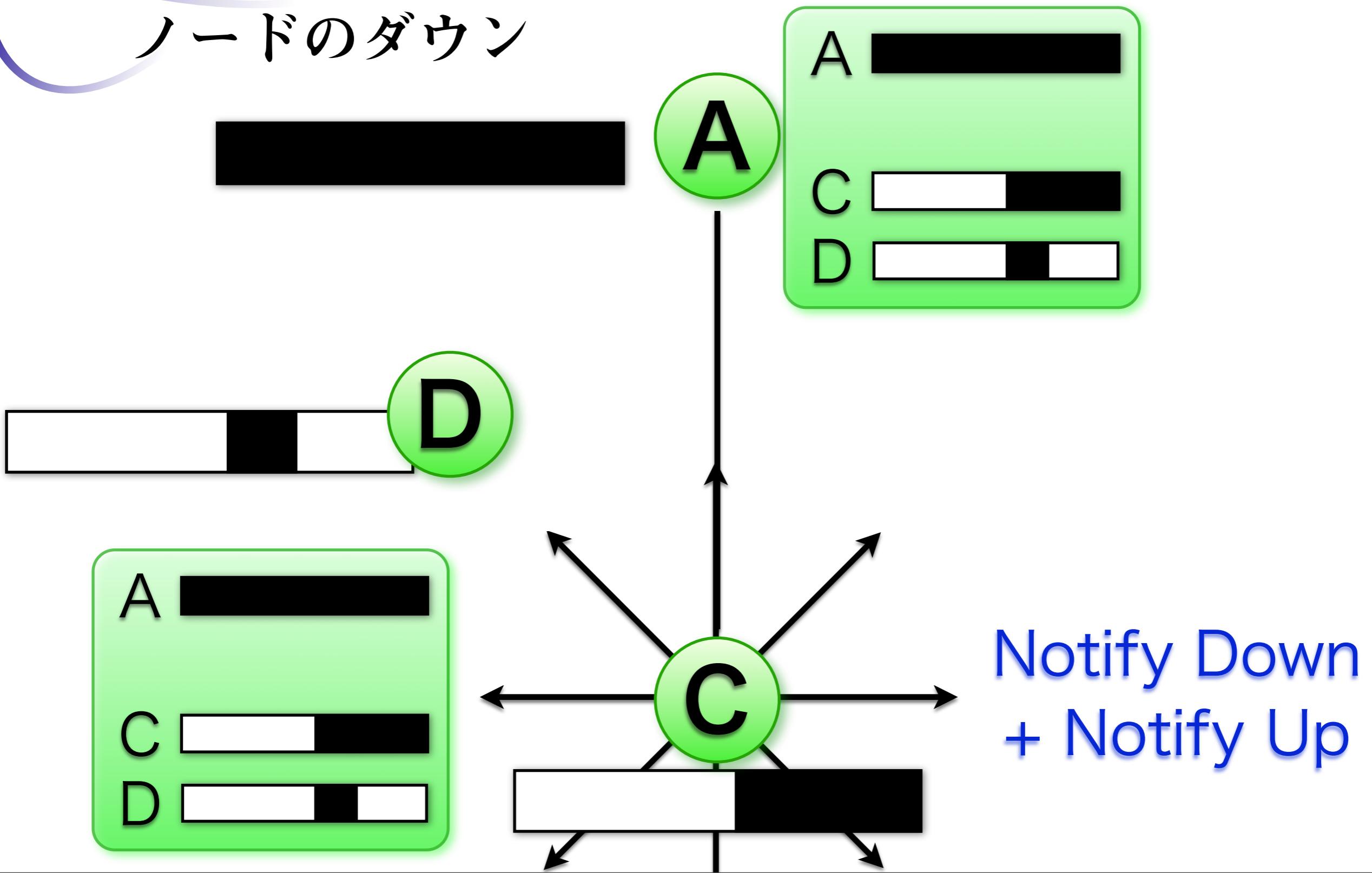
分散ディスク共有

ノードのダウン



分散ディスク共有

ノードのダウン



Notify Down
+ Notify Up

分散ディスク共有

- TCP接続を維持する
 - 3-way handshakeを減らす
 - ノードのダウンを検出
- **Find**と**Notify Down**にも**Notify Up**を載せる
 - V-Tableの精度が上がる

