

# コンピュータシステムA - ハードウェアを中心に -

---

## #7 性能向上、集中と分散

# 超並列アプローチ

---

- NEC 地球シミュレータ (2002/6~2004/6 TOP500 winner)  
8 CPU からなる計算ノードを高速ネットで640台接続  
5120のスーパーコンピュータで並列計算
- 東京工業大学 TSUBAME (2010/11 TOP500 #4)  
12CPUコア+3GPU (Tesla) をもつ演算サーバを高速ネット  
トで 1408 台 (2.9GHz Intel 64bit core x 16,896)
- 何故このようなスタイルの高速計算機が？  
超高速計算のためにこのスタイルを採った理由は？

# 繰り返し処理

---

- (一般的) コンピュータの特長
  - 単純な装置で複雑な処理を可能にする
- 役割分担の存在
  - ハードウェアは単純・高速に
  - 複雑さはソフトウェアで実現
  - 複雑な処理は単純な処理の組み合わせや繰り返しに分解
- これがハードウェアに高速性が求められる理由
  - 分解処理の例：多数桁の足し算

# 多数桁の足し算

- 筆算は分解処理の例

「多数桁の足し算」を「一桁演算の繰り返し」へ単純化

- 10進で3桁の足し算を分解

10進1桁の足し算を三回(繰り上がり込み)

2進では9桁、足し算も9回

$$234 + 456 = 690 \text{ は?}$$

2	3	4
+	+	+
4	5	6
=	=	=
6	8	10
	+1	
6	9	0

$$234 (11101010) + 456 (111001000) = 690$$

		1	1	1	0	1	0	1	0
		+	+	+	+	+	+	+	+
	1	1	1	0	0	1	0	0	0
	=	=	=	=	=	=	=	=	=
	1	10	10	1	0	10	0	1	0
	+1	+1			+1				
1	0	1	0	1	1	0	0	1	0

# 実際の演算処理

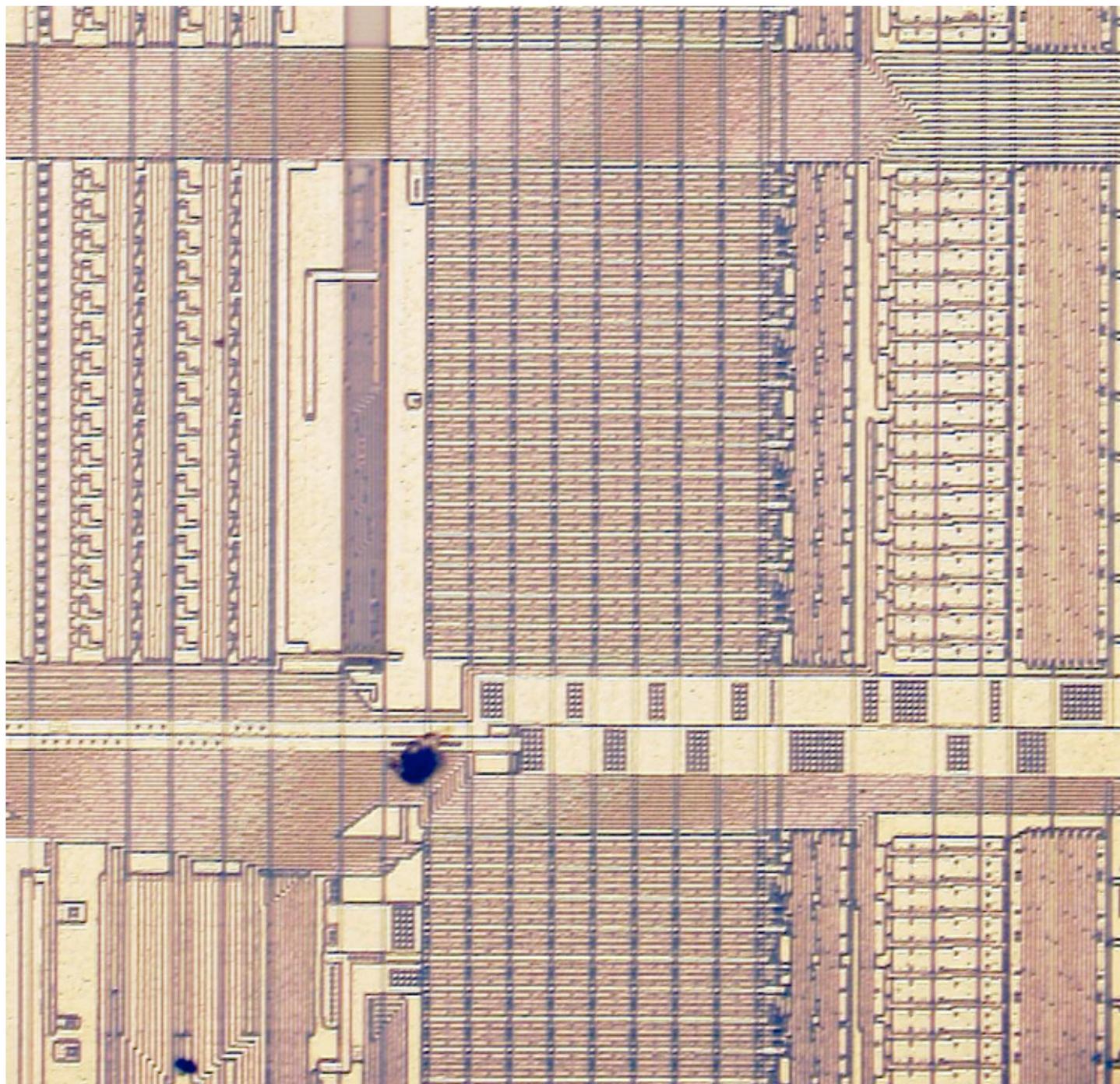
このような方法(筆算)で処理を行なう場合、

1. 一つの素子を9回使い回して処理する
2. 素子を9つ並べて一回で処理するか  
のいずれかとなる。

$$234 (11101010) + 456 (111001000) = 690$$

		1	1	1	0	1	0	1	0
		+	+	+	+	+	+	+	+
	1	1	1	0	0	1	0	0	0
	=	=	=	=	=	=	=	=	=
	1	10	10	1	0	10	0	1	0
	+1	←	+1	←	+1	←			
1	0	1	0	1	1	0	0	1	0

CPU 内部の  
拡大写真



# 性能 (処理速度) は何で決まるか

---

- ビット並列度を高める

性能 = 回路の複雑さ

Intel 4004 (1971)の4bitから 64bit に

- 繰り返し周期をより短く

性能 = 短い繰り返し周期 = 高速な回路

Intel 4004 (1971)の108KHzから 3.8GHz 程度に

- これを微細化によって実現する

# 微細化による利益

---

- 微細化＝一石三鳥
- より細く短い配線：電気の伝わる時間を短縮
- より小さな素子：機能するまでに必要な時間を短縮
- 一定面積に集積できる回路の増加  
処理時間短縮に貢献する機能・部品の追加
- チップ面積縮小によるチップ単価（製造コスト）低減

# 素子技術の発展と高速化

Intel i860XP, 40MHz, 1990  
1  $\mu$ m, 120万トランジスタ



Motorola MC68000, 8MHz, 1980  
3.5  $\mu$ m, 7万トランジスタ



Sun UltraSPARC III, 600MHz, 1999  
0.18  $\mu$ m, 2900万トランジスタ

年ごとに配線幅は狭く、  
高速になり、また集積素  
子数も増える。

# マイクロプロセッサの成功

---

- いわゆるCPU

半導体の微細化、集積化による高速化技術を追及  
他の高速化手法を大きく抜いて成功

- ムーアの法則(Moore, 1965)の体現者

半導体回路の集積度は18-24ヶ月ごとに倍になる

- チップ価格 = 開発費用 / 生産数

共通品、量販品としてのPCの成功

最高速製品が最廉価品であるという矛盾

# さらなる高速化と限界

---

- 微細化による問題

現行の上級製品は 32nm ~ 45nm 程度の配線幅

静電気ですら簡単に配線を壊してしまう

ゲート絶縁膜は1.2nmの厚さ（原子数個分）

# さらなる高速化と限界

---

- 熱問題

過去において、微細化と高速化は同義だった

消費電力 = 熱

トランジスタ数と周波数に比例 × 動作電圧の2乗に比例

熱の集中：あの面積に 100W 程度集めると？

Core i7 3.2GHz = 130W max.

# ブレイクスルー

---

既存技術の限界を別の視点から  
打ち破る動きが必要な時がある

# 並列分散・グリッド・P2P

---

- 並列処理

単体プロセッサの速度に依存するモデルからのシフト

複数のプロセッサを同時に利用するモデル

SMP ・ マルチコア

- 超並列（ネットワーク接続）

大量のコンピュータを集めて大きな計算資源を

グリッド

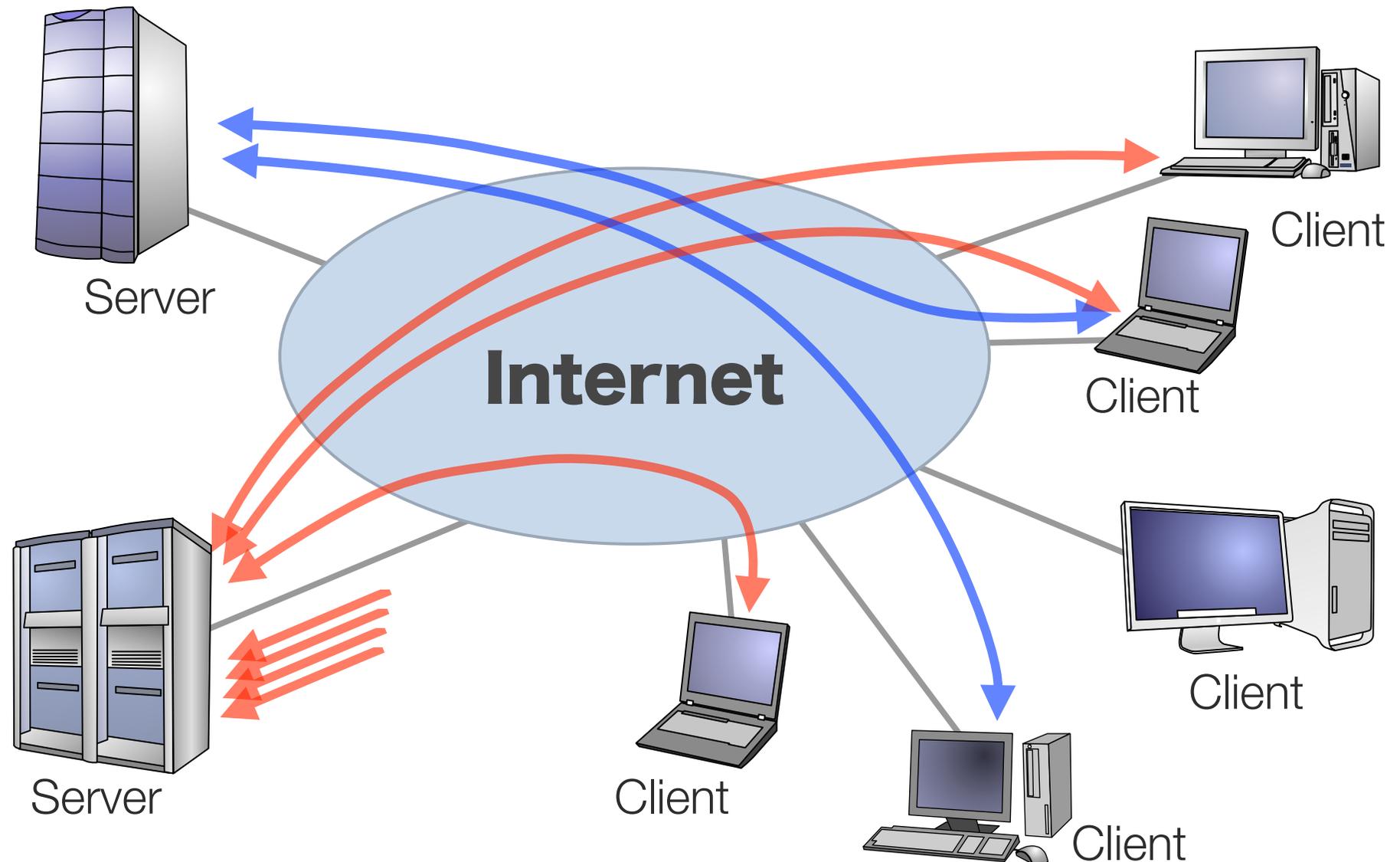
- P2P

集中サーバによるネットワーク処理の限界

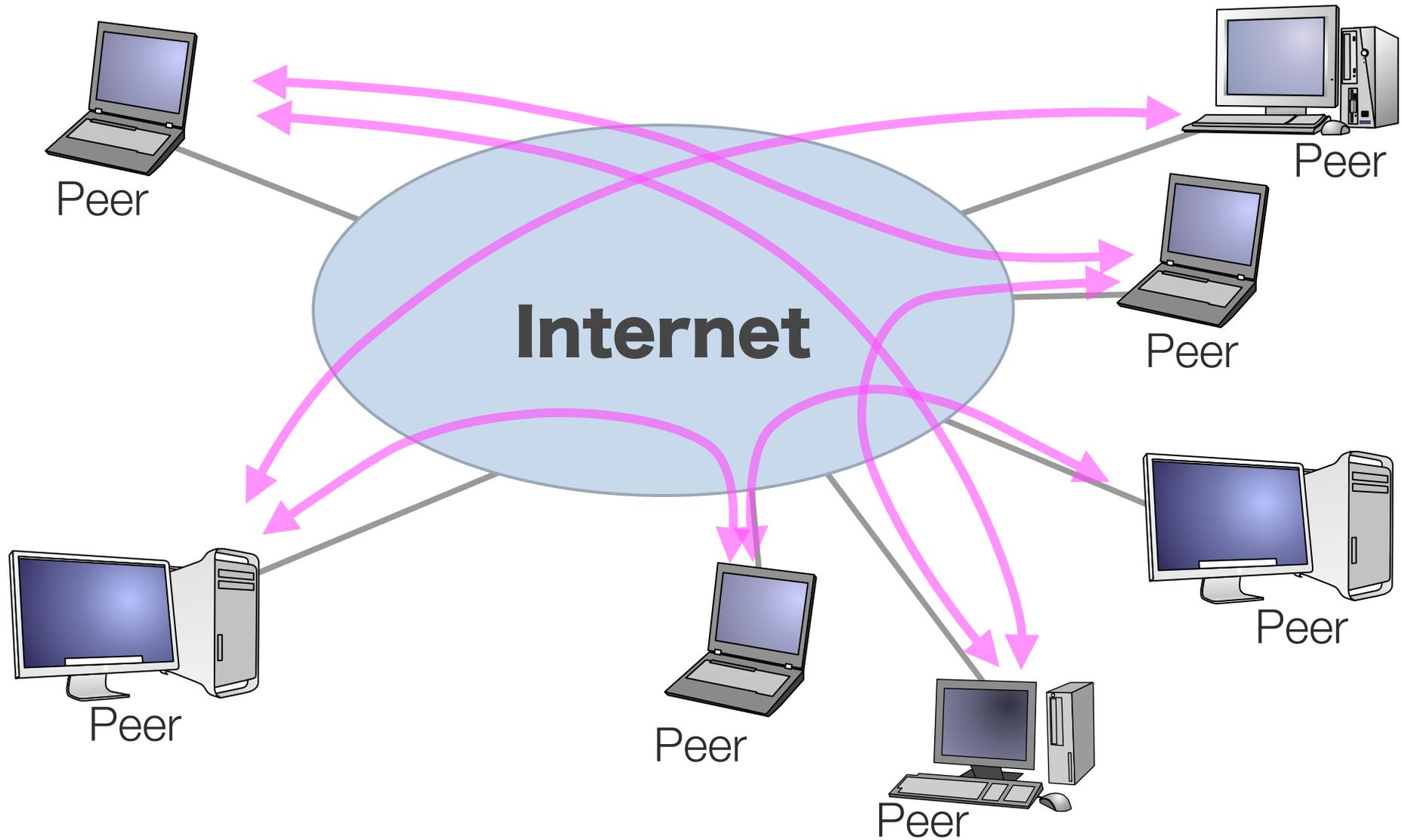
互いに対等なコンピュータを接続して協調動作

サーバ・クライアントとは異なる新しいモデル

# インターネットのサービスモデル



# P2Pサービスモデル



# 集中と分散、技術のバランス

---

- 集積回路への技術集中
  - 従来手法での高速化の限界
- それを補う処理能力向上の手法
  - 並列処理・分散処理（実は両者は同じもの）
- 素子・デバイス技術の開発
  - 光スイッチ、スピントロニクス（racetrack）、etc..
- 歴史
  - 計算機が実用化されて50年
  - そのダイナミズムを感じる
- 次の判断をその波の上で行う感覚が求められている