

第3回講義：Machinationsフレームワー ク

第3回講義：Machinationsフレームワーク (Chapter 4)

1. Machinationsとは
2. 基本要素（ノード）
3. 接続の種類
 - 3.1 リソース接続（実線矢印 →）
 - 3.2 状態接続（点線矢印 —>）
4. アクティベーションモード
5. プル/プッシュモード
6. ゲートの種類
 - 6.1 確率的ゲート（Probabilistic Gate）
 - 6.2 決定的ゲート（Deterministic Gate）
7. フィードバック構造の分析
 - 7.1 正のフィードバック（Positive Feedback）
 - 7.2 負のフィードバック（Negative Feedback）
8. フィードバックプロファイルの7つの特性
9. SimWarの例
- まとめ

第3回講義：Machinationsフレームワーク (Chapter 4)

1. Machinationsとは

Machinationsは、ゲームの内部経済（Internal Economy）をモデル化するための動的ダイアグラムフレームワークである。リソースの生成・消費・変換・流通をビジュアルに表現し、ゲームバランスの分析やシミュレーションを可能にする。

Machinations基本構造

Machinations基本構造

2. 基本要素（ノード）

要素	記号	機能
プール (Pool)	丸 ○	リソースを蓄積・保持する
ソース (Source)	上向き三角 △	リソースを無限に生成する
ドレイン (Drain)	下向き三角 ▽	リソースを消費・除去する
コンバーター (Converter)	横向き三角 ▷	あるリソースを別の種類に変換する
ゲート (Gate)	ひし形 ◇	リソースの流れを条件分岐する
トレーダー (Trader)	—	プレイヤー間でリソースを交換する

重要ポイント：各ノードの役割を正確に理解することが、ダイアグラム設計の基礎となる。プールは「貯蔵」、ソースは「生成」、ドレインは「消費」、コンバーターは「変換」と覚える。

3. 接続の種類

3.1 リソース接続（実線矢印 →）

リソースの物理的な流れを表す。ソースからプールへ、プールからドレインへなど、実際にリソースが移動する経路。

3.2 状態接続（点線矢印 —>）

情報の流れを表す。リソースそのものは移動しないが、あるノードの状態が別のノードの挙動に影響を与える。

状態接続には以下の4種類がある：

修飾子	機能
ラベル修飾子（Label Modifier）	リソース接続の流量を変更する
ノード修飾子（Node Modifier）	ノードの内部状態を変更する
トリガー（Trigger）	条件を満たしたときにアクションを発火する
アクティベーター（Activator）	ノードの有効/無効を切り替える

重要ポイント：リソース接続は「モノの流れ」、状態接続は「情報の流れ」である。状態接続を使いこなすことで、条件分岐やフィードバックなど複雑なゲームメカニクスを表現できる。

4. アクティベーションモード

ノードがいつ動作するかを決定するモード。

モード	動作	記号表記
自動 (Automatic)	毎ステップで自動的に実行される	*
インタラクティブ (Interactive)	プレイヤーが手動で起動する	ダブルライン
受動 (Passive)	他のノードから起動される	—

重要ポイント：自動モードはシステム駆動のメカニクス（毎ターンの収入など）、インタラクティブモードはプレイヤーの意思決定（カードを引く、アイテムを使うなど）に使う。

5. プル/プッシュモード

リソースの流れる方向の制御方式。

- ・ **プルモード (Pull) :** 下流のノードが上流からリソースを「引っ張る」
 - 例：プレイヤーが山札からカードを引く
- ・ **プッシュモード (Push) :** 上流のノードが下流にリソースを「押し出す」
 - 例：ソースが自動的にリソースを生成して送り出す

重要ポイント：プルは「需要主導」、プッシュは「供給主導」である。リソースが不足したときの挙動が異なるため、ゲームメカニクスの性質に合わせて適切に選択する。

6. ゲートの種類

ゲートはリソースの分配方法を決定する。

6.1 確率的ゲート (Probabilistic Gate)

確率に基づいてリソースの行き先を決定する。サイコロの出目やランダムイベントの表現に使用。

- ・例：50%の確率でルートAへ、50%でルートBへ

6.2 決定的ゲート (Deterministic Gate)

あらかじめ定められた条件・パターンに基づいてリソースを分配する。

- ・例：3回に1回はルートAへ、それ以外はルートBへ

重要ポイント：確率的ゲートはランダム性のあるメカニクス、決定的ゲートは規則的なメカニクスに対応する。ゲームのランダム性の度合いを制御する重要な要素。

7. フィードバック構造の分析

ゲームバランスの核心となるフィードバックループの分析。

7.1 正のフィードバック (Positive Feedback)

成長を加速させるループ。「勝っている者がさらに勝つ」構造。

- ・例：モノポリーの不動産投資 — 物件を持つ → 家賃収入が増える → さらに物件を買える
- ・効果：ゲームを収束させる（勝敗の決着を早める）
- ・リスク：格差が開きすぎてゲームが退屈になる可能性

7.2 負のフィードバック (Negative Feedback)

バランスを維持するループ。「勝っている者にハンデが付く」構造。

- 例：リスクの軍隊分配 — 領土が多い → 防衛が分散する → 弱体化する
- 効果：ゲームを長引かせる（プレイヤー間の格差を縮める）
- リスク：ゲームが膠着して終わらなくなる可能性

重要ポイント：正のフィードバックは収束を生み、負のフィードバックは均衡を生む。良いゲームデザインは両者を適切に組み合わせる。

8. フィードバックプロファイルの7つの特性

フィードバックループを分析する際の7つの観点：

#	特性	説明
1	極性 (Polarity)	正のフィードバックか、負のフィードバックか
2	強度 (Strength)	効果が弱いか、強いか
3	持続性 (Duration)	一時的か、永続的か
4	速度 (Speed)	効果の発現が速いか、遅いか
5	投資の有無 (Investment)	プレイヤーの投資を必要とするか
6	構築的/破壊的 (Constructive/Destructive)	リソースを増やすか、減らすか
7	直接的/間接的 (Direct/Indirect)	フィードバックが直接的か、他の要素を経由するか

重要ポイント：これら7つの特性を用いることで、フィードバックループを多角的に分析でき、ゲームバランスの調整ポイントを特定できる。

9. SimWarの例

SimWarは2人対戦の戦略ゲームであり、Machinationsフレームワークの実践的な分析対象として取り上げられる。

- ・両プレイヤーがリソース（兵力）を生成・配分し、相手と戦闘する
- ・正のフィードバック（領土獲得 → 兵力増加）と負のフィードバック（戦線拡大 → 防衛力分散）が共存
- ・Machinationsダイアグラムでリソースフローを可視化し、バランスの偏りを発見できる

重要ポイント：SimWarの分析を通じて、Machinationsが実際のゲーム分析にどのように活用できるかを理解する。ダイアグラムを描くことで、直感ではわかりにくいバランスの問題を発見できる。

まとめ

- ・Machinationsはゲームの内部経済をビジュアルに分析するツール
- ・6種類のノード、2種類の接続、3種類のアクティベーションモードが基本構成要素
- ・フィードバック構造の理解がゲームバランス設計の鍵
- ・7つのフィードバック特性を用いて体系的に分析する