

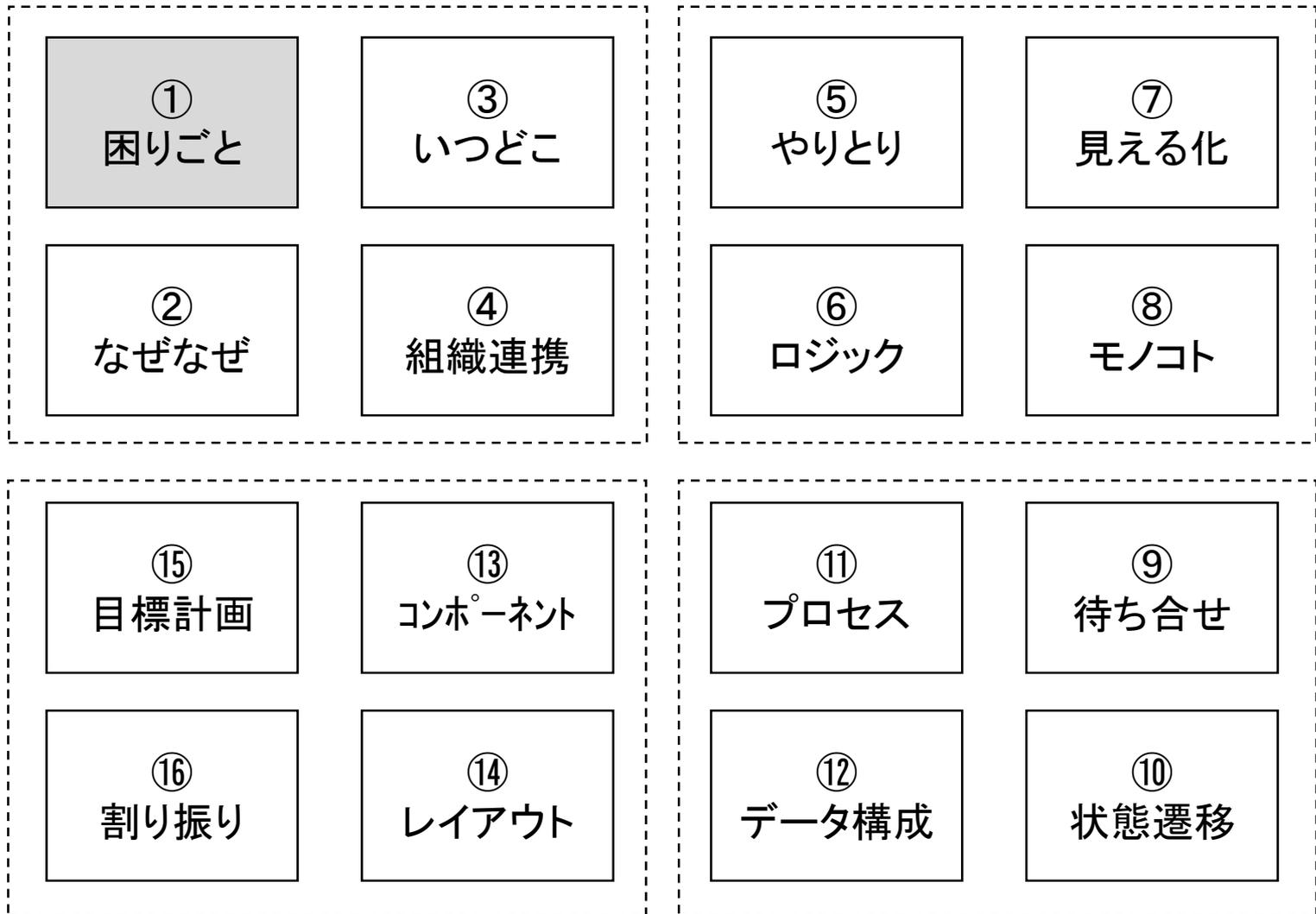
IVIスマートシンキング

# IVIモデラー2020年度版 16チャートの書き方

2020年8月

インダストリアル・バリューチェーン・  
イニシアティブ

# 16チャート

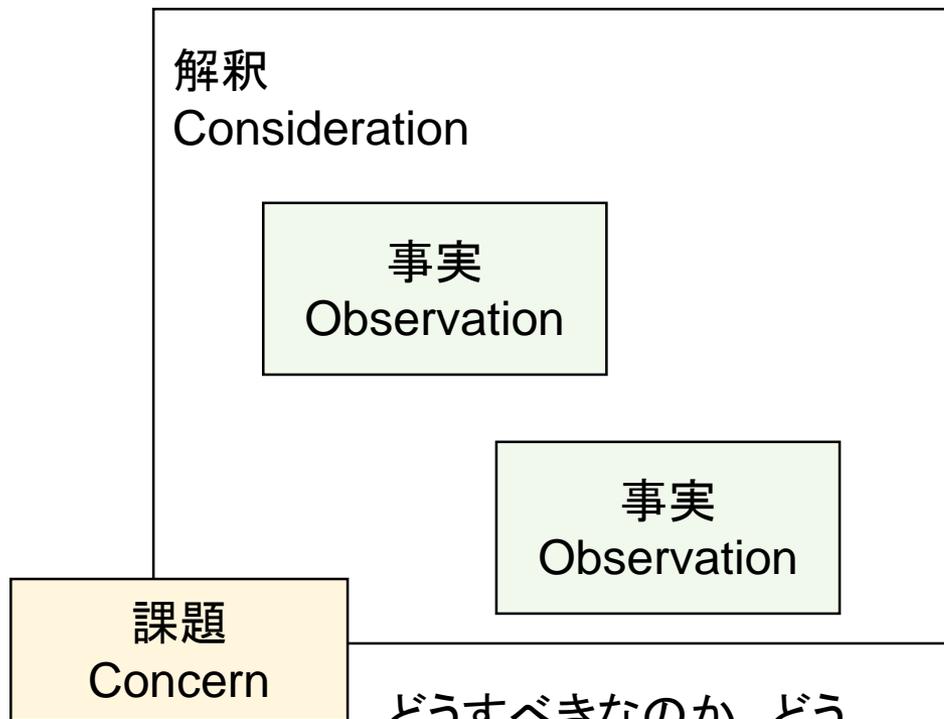


# エクスポレーション(問題発見)

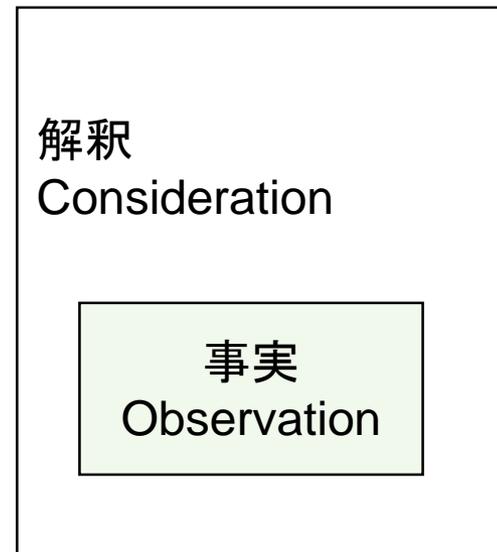
**困りごととチャート**は、観測された客観的な事実と、それに対する解釈を切り分けて構造化するためのチャートである。KJ法の親和図に相当する。解釈は、階層的に設定できる。解釈はその主体となる役者が存在することが前提となっており、必要に応じて役者を配置する。最終的に、課題を抽出し、配置する。

事実に対する解釈  
を視点や意図を含  
めて記述する。

解釈や課題に対する関係者、  
主体は役者として設定する



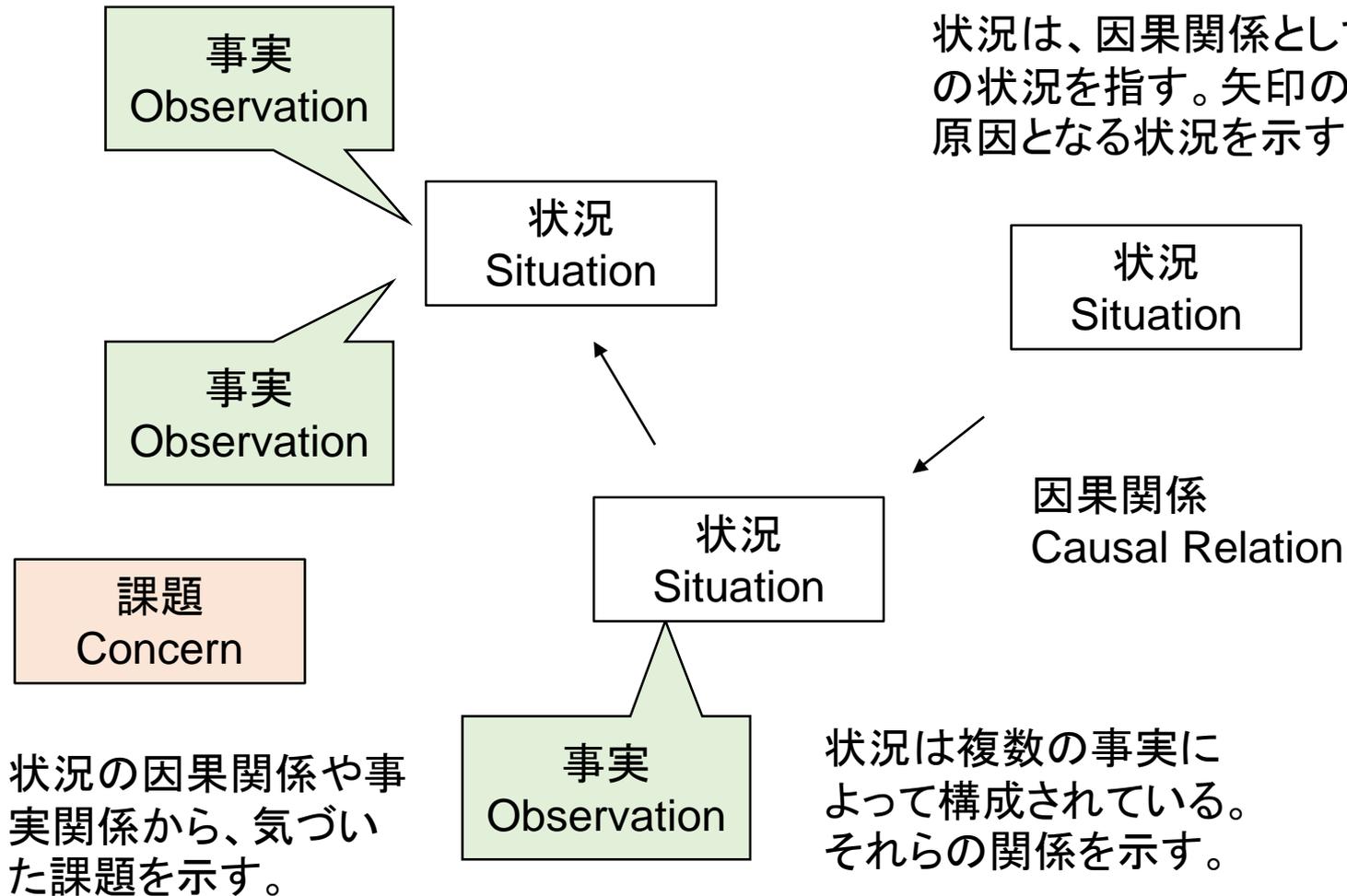
どうすべきなのか、どう  
ありたいのかという目標  
や課題を記述する



ブレインストーミングによ  
り、観測された事実、思  
いついた事実を断片的で  
もよいので記述する。

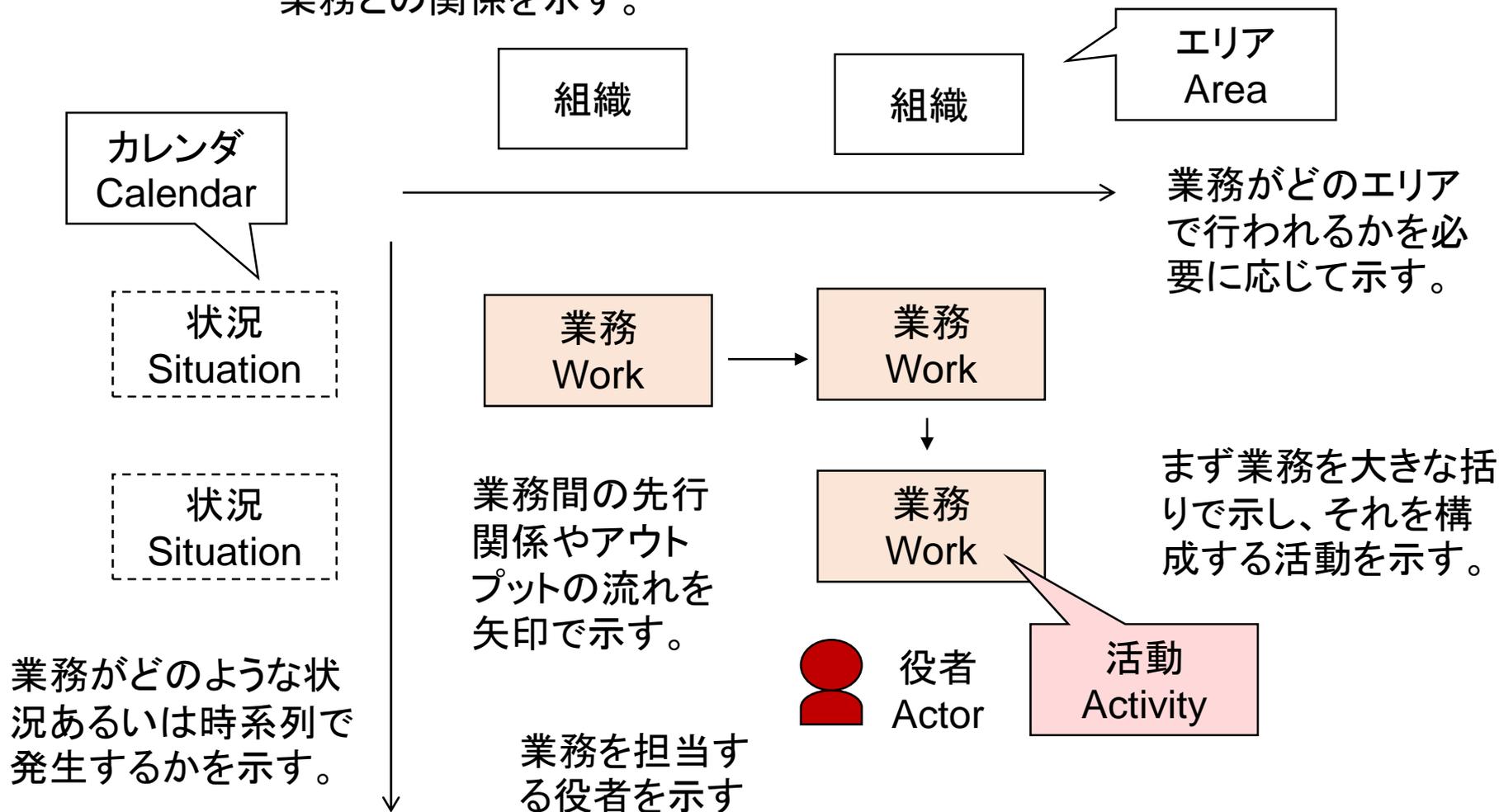


**なぜなぜチャート**では、ある状況の原因となる状況をたどり、根本となる事実を発見する。状況は事実によって構成されている。状況の因果関係の背後にある事実関係を明らかにし、どこに課題があるのかを示す。



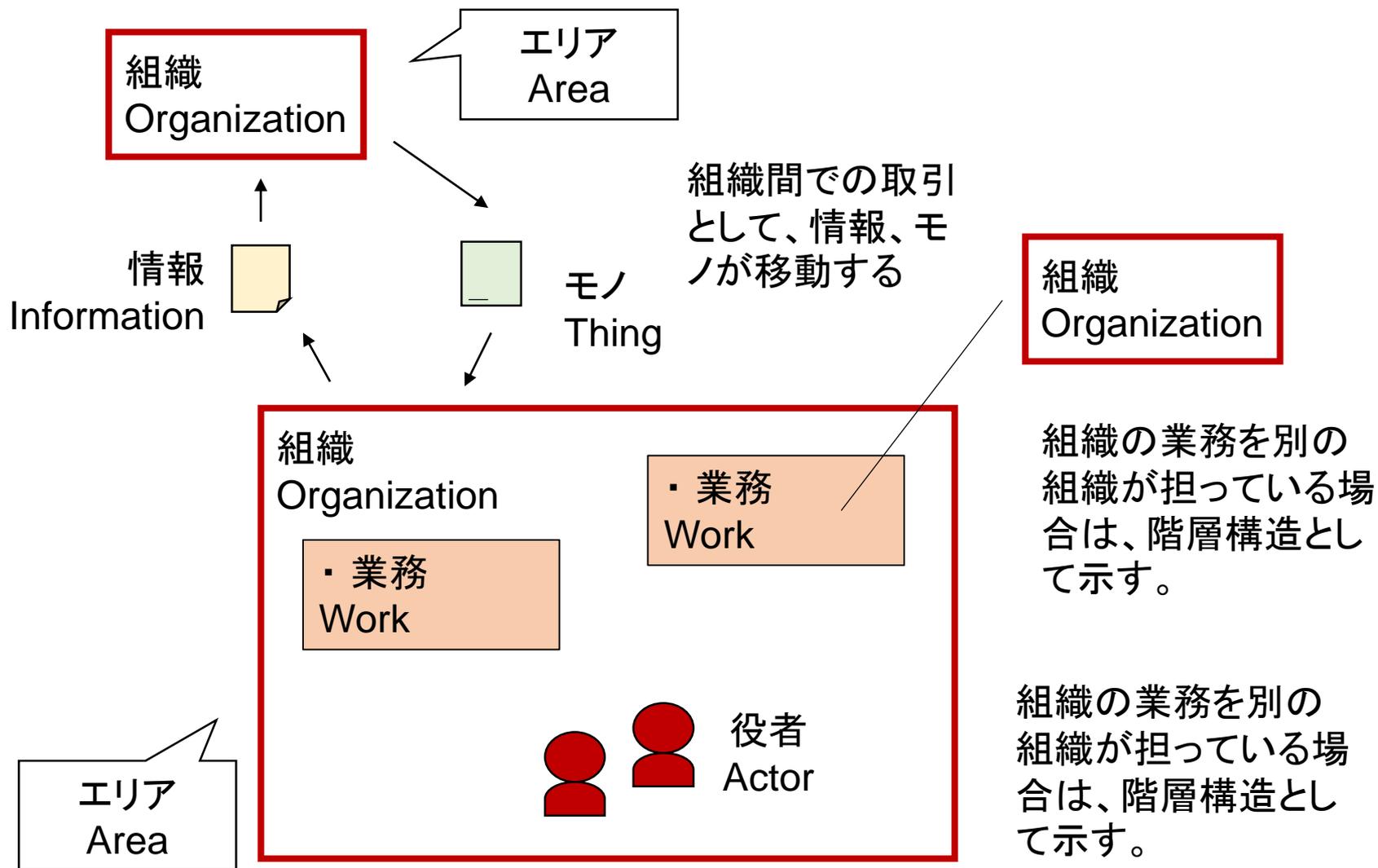
**いっどこチャート**は、組織において、どのような業務があり、それらがどのような活動で構成されているかを示すとともに、それらが状況に対応して時系列的にどのようなフローになっているかを示す。業務を通じた組織間の関係なども明らかにする。また、それぞれの業務が空間的な位置関係として組織によってどのエリアで実施されるかも知る。

組織を横軸(または縦軸)にとり、  
業務との関係を示す。



**組織連携チャート**は、組織に帰属する業務を明らかにし、組織の構成員を役者として示す。また、組織間の業務の連携を、業務間でやりとりされるモノまたは情報によって明らかにする。また、組織の空間的な位置づけをエリアによって示す。組織が階層化されている場合は、組織内の業務と内部の組織を対応づける。

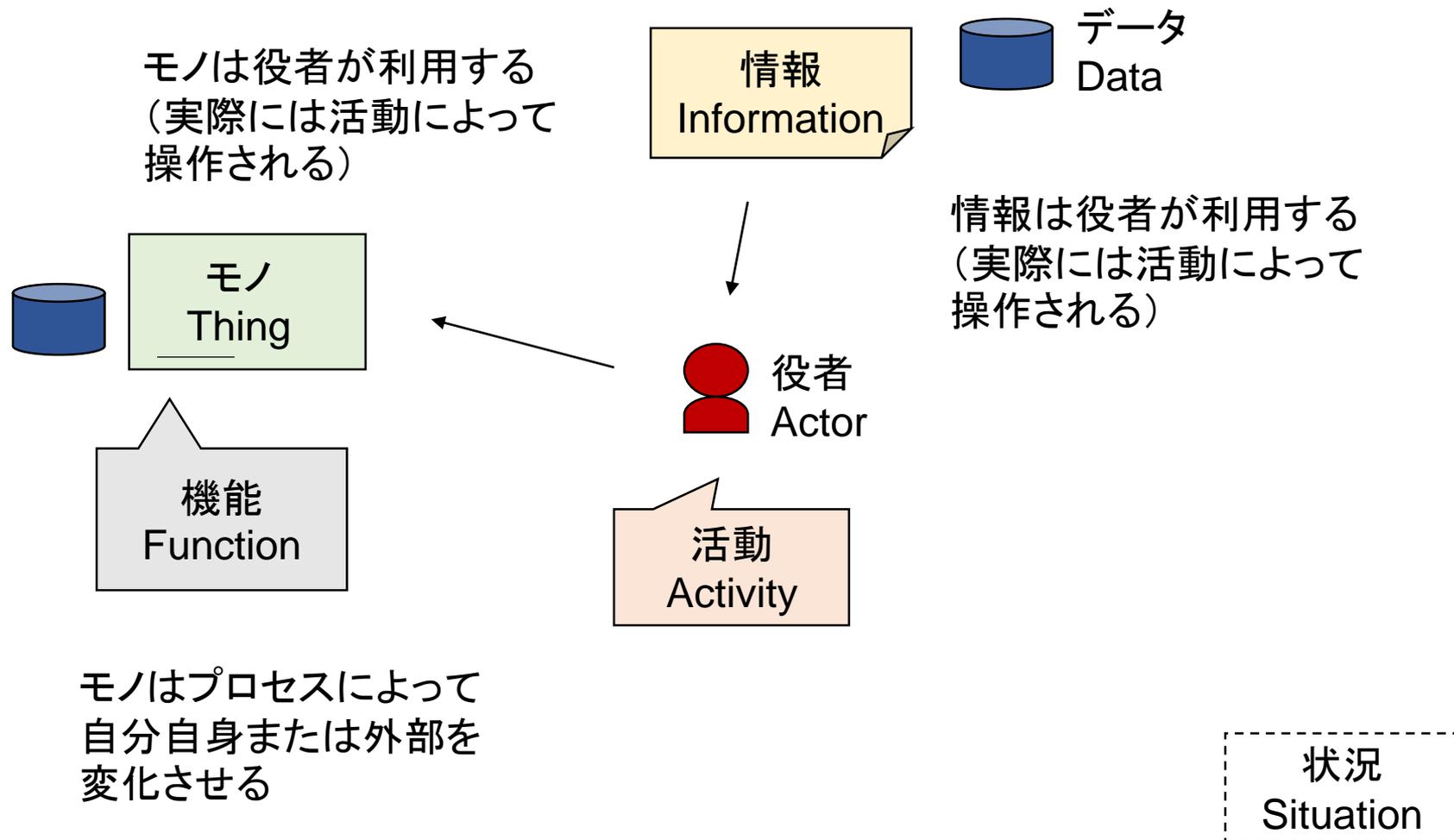
# 組織連携チャート



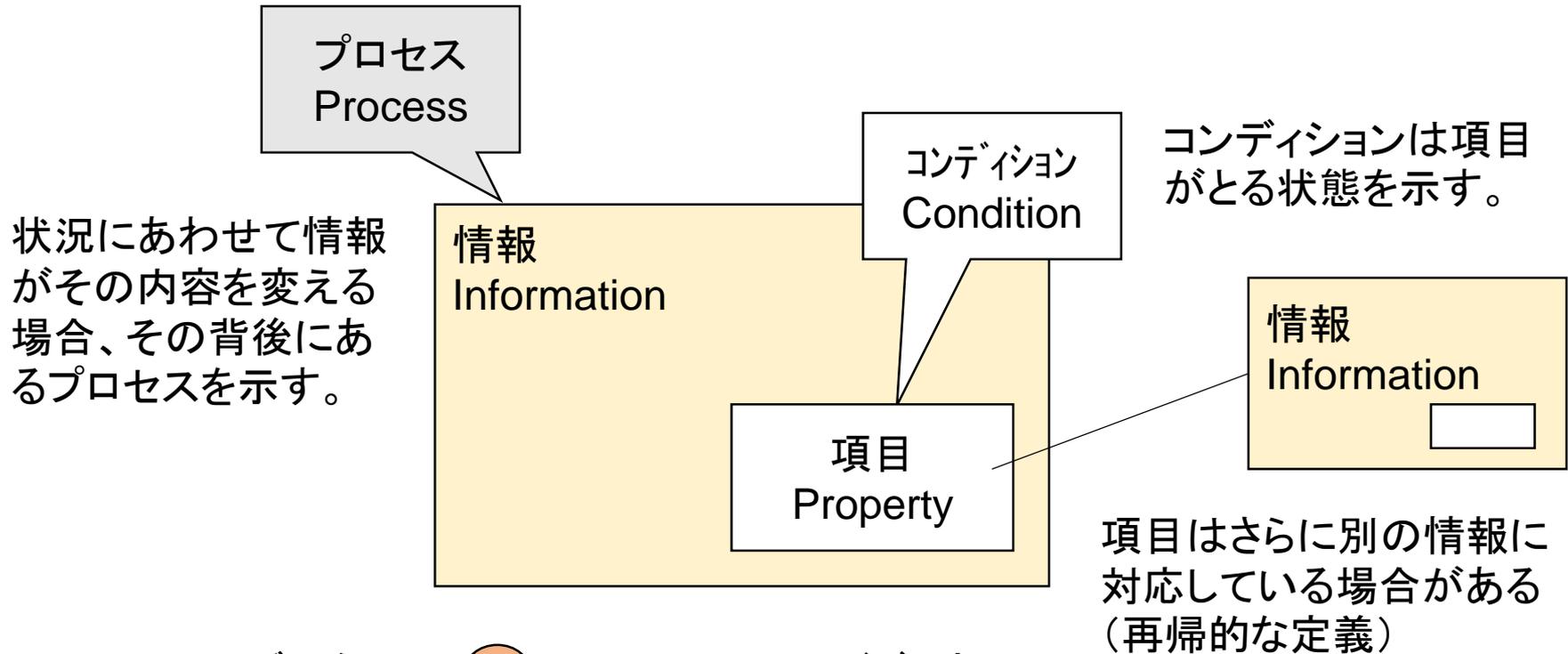
# リコグニション(問題共有)

**やりとりチャート**は、現場で行われている業務の実際の様子を示すために、役者と別の役者の間でモノまたは情報がどのようにやりとりされているかを示す。モノ、情報にデータが定義されている場合は、デジタル化されていることを示す。

情報とモノはデータに対応づけることができる。



**見える化チャート**は、情報の内容や構造を示す。情報の内容として、どのような項目が含まれているかを示し、情報が階層化されている場合は、項目に対応する別の情報として定義する。情報が対象とする役者、情報に対応するデータを定義可能。また、情報にはその操作や変化としてプロセスが定義できる。



データ  
Data

データとして扱う場合にデジタルマークとして設定



役者  
Actor

情報はそれを利用する役者をもつ



イベント  
Event

情報から読取れるイベントがある場合、あるいはある時点での情報である場合などに設定する

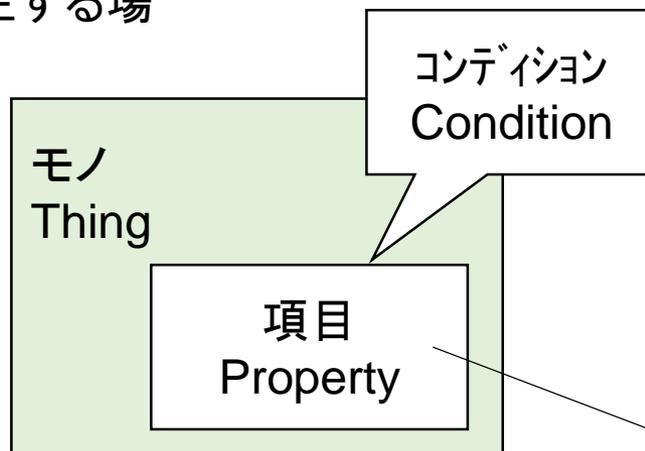


**モノコトチャート**は、モノの内容や構造を定義する。モノの項目として、仕様、設定、履歴など、データとなり得るものを定義し、それらの状態としてコンディションの定義がある場合はあらかじめ示す。モノは構造化(階層化)可能である。機能を定義した場合は、モノに定義したプロセスを主体的に行うことができる。

イベントに対応する場合、あるいはイベントを発生する場合に指定する

■ イベント  
Event

項目は、モノの属性や設定値、測定値などを示す。コンディションはその値の状態



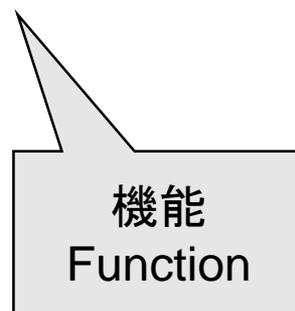
モノは内部にモノを再帰的に定義できる。それぞれの階層に項目が設定できる。



👤 役者  
Actor

🗄️ データ  
Data

モノを操作または利用するユーザとして役者が存在する。また、モノの状態値や操作値などをデータとして表記し外部とやり取りできる場合はデータを設定する。

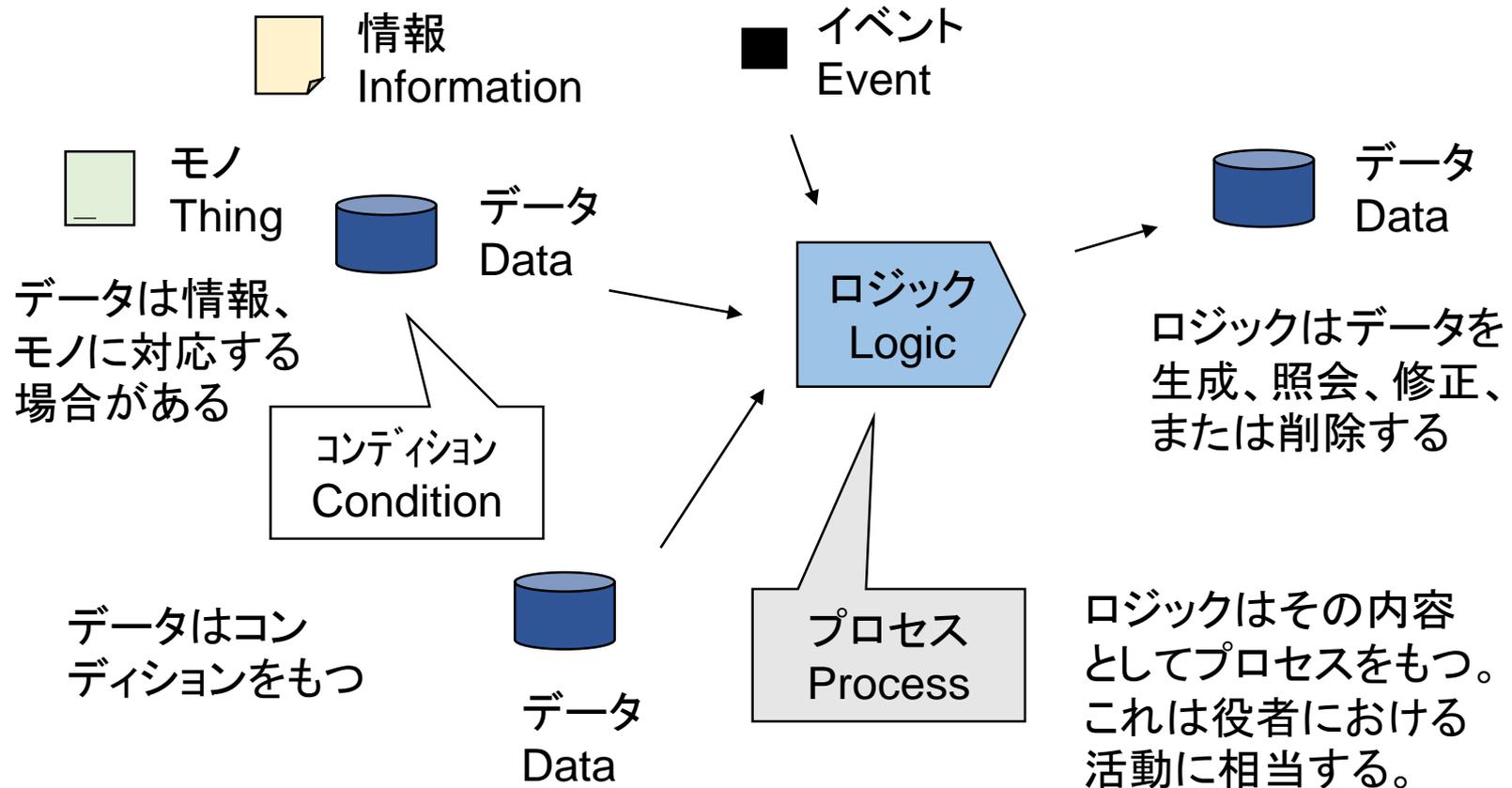


モノがもつ機能。役者における活動に相当する



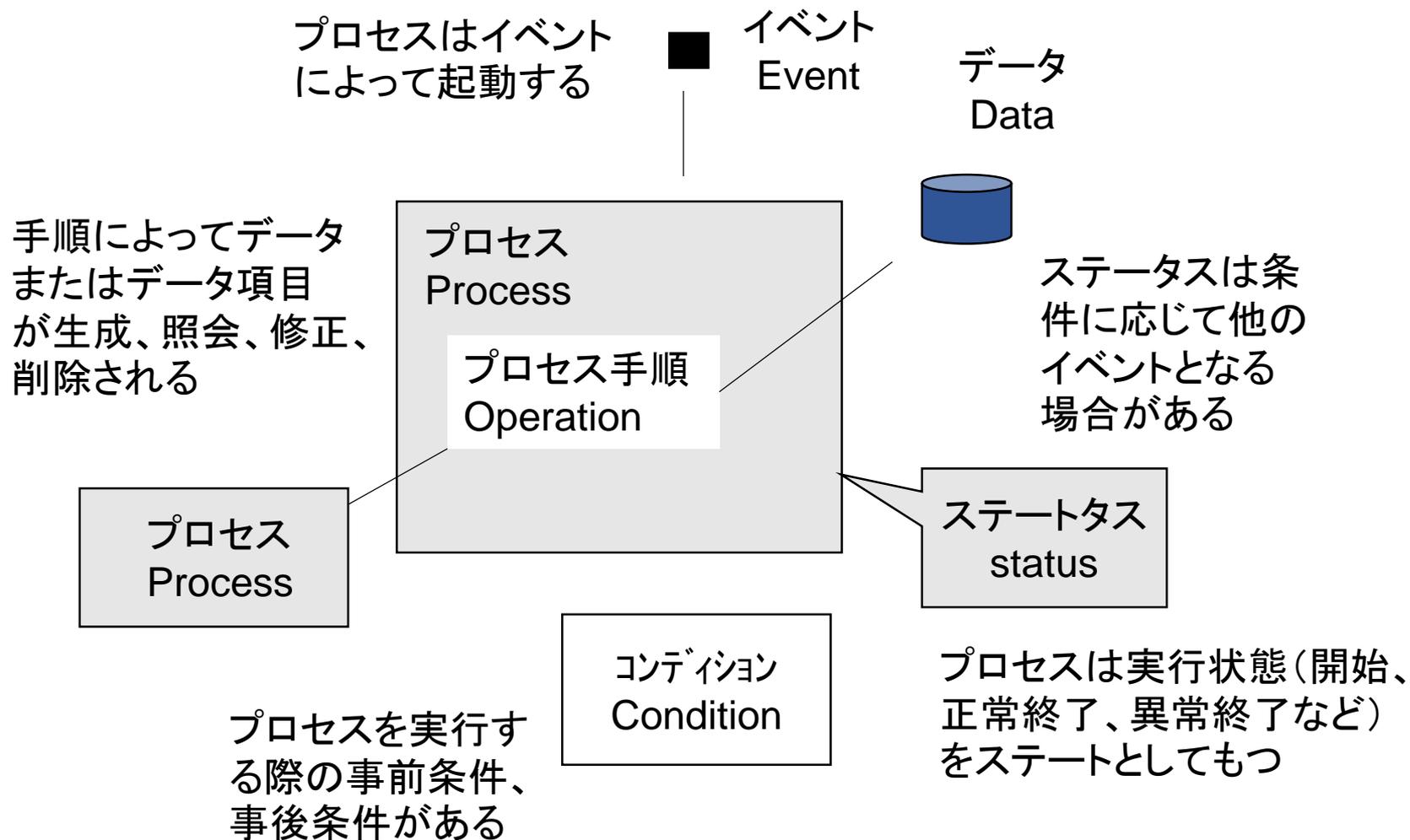
**ロジックチャート**は、サイバー空間上（デジタル処理可能なコンピューターとネットワーク内）で、ロジックがどのようなデータを用いてプロセスを実行しているかを示すためのものである。ロジックはイベントによって起動され、別のイベントを生成する。データには、フィジカル空間上のモノや情報を対応づけることができる。

ロジックはイベントによって起動する。また、ロジックは、実行結果として別のイベントを発動する

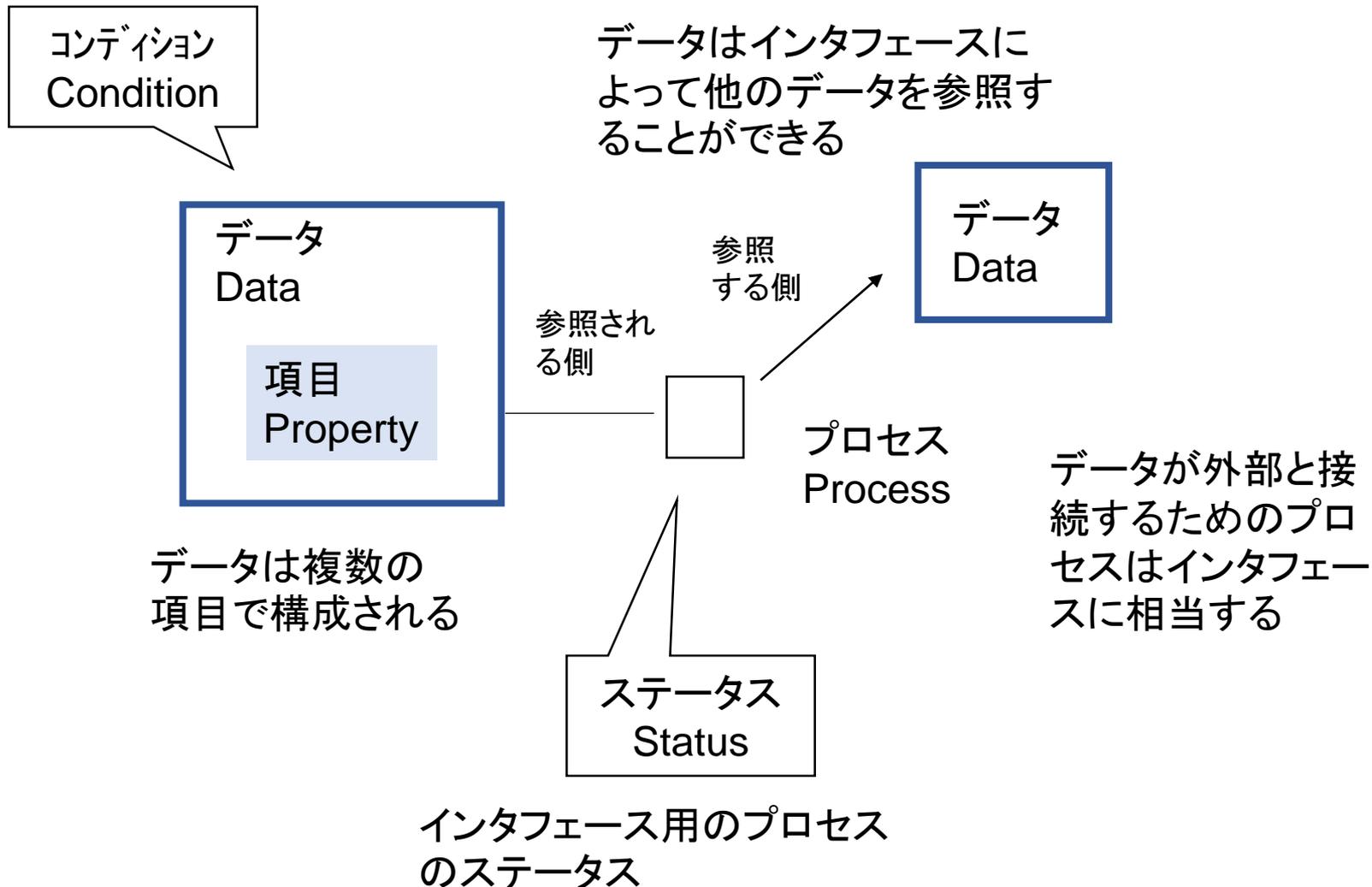


# オーケストレーション(課題設定)

**プロセスチャート**は、主にロジックの内容として定義されたプロセス（サイバー側のプロセス）について、その手順および対象とするデータの関係を示す。プロセスは、イベントで起動し、事前条件、事後条件を持ち、ステートが変化する。



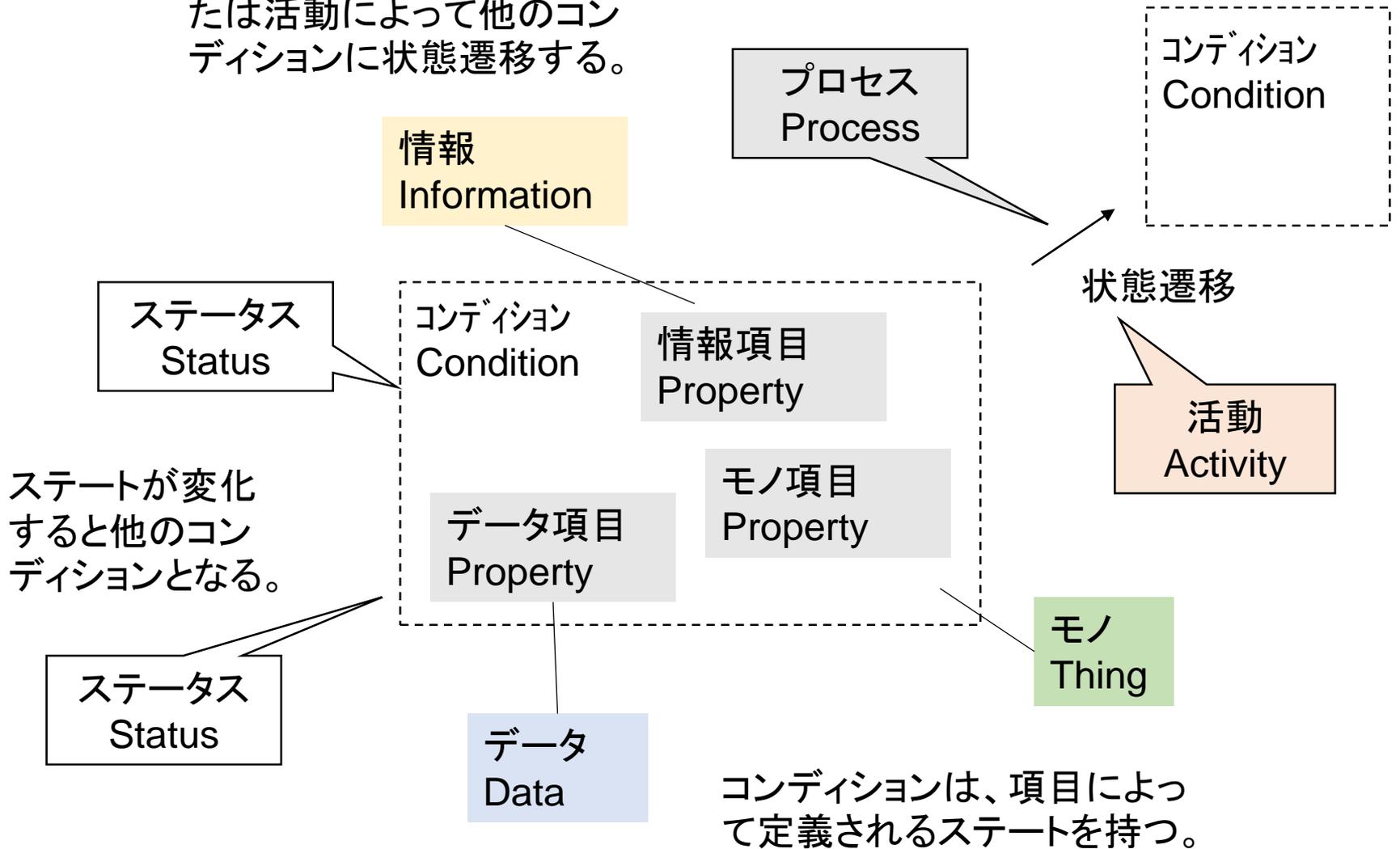
**データ定義チャート**は、データとその内容としてのデータ項目、およびデータ間の参照関係をしめす。ER図に相当する。データ間の関係では、データを参照される側がインタフェースを介して、参照する側にそのプロセスを提供する形となる。



**状態遷移チャート**は、コンディションとして定義した状態が、プロセスや活動によって別のコンディションに遷移する様子を示す。コンディションは、モノ、情報、データを構成する項目とその値に対する制約としてのステートによって定義される。プロセスや活動は、これらの値を変更することになる。

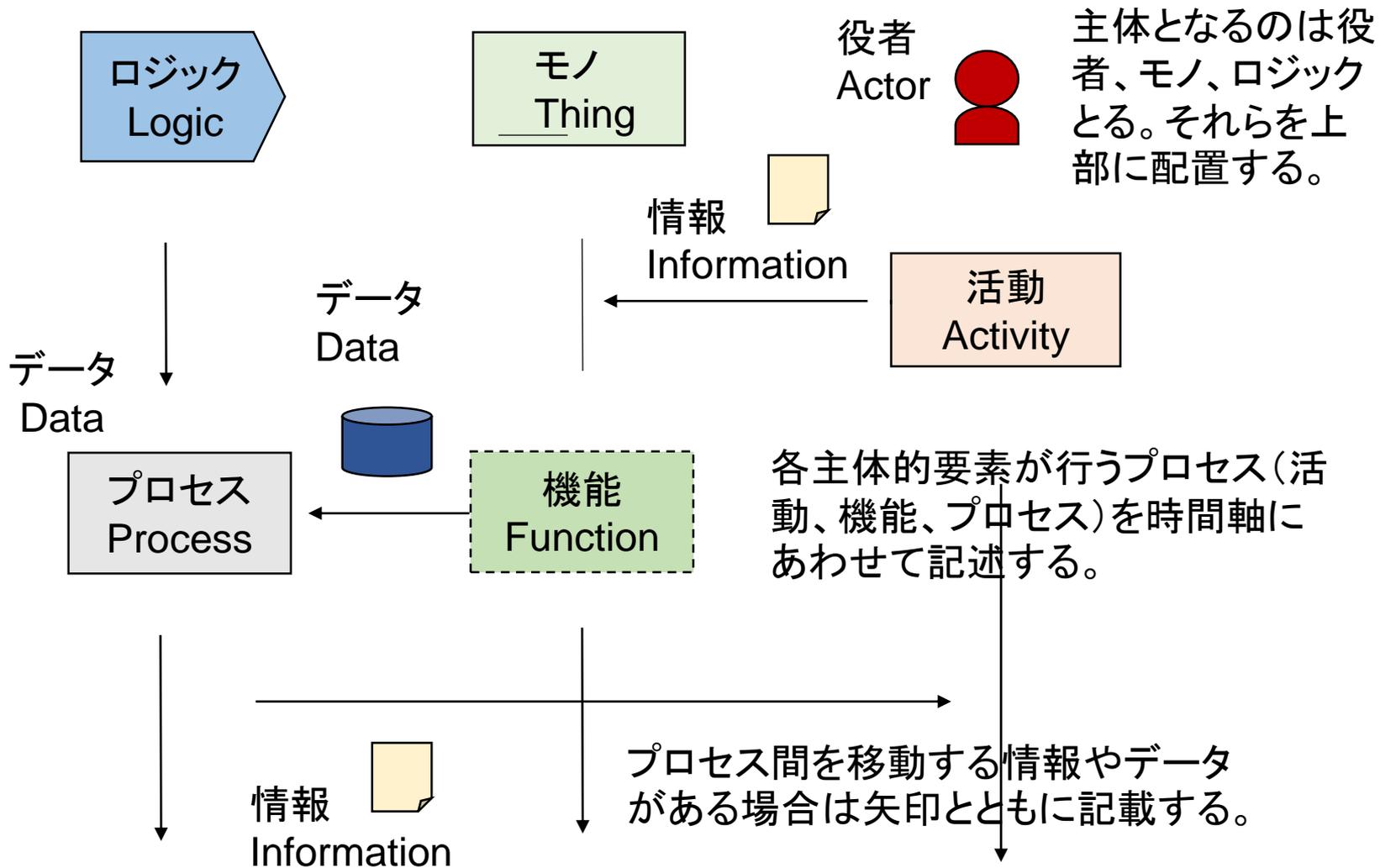
# 状態遷移チャート

コンディションは、プロセスまたは活動によって他のコンディションに状態遷移する。



**待ち合せチャート**は、役者、モノ、またはロジックによって実行されるプロセス間のシーケンスを確認し、対象とするデータや情報の生成、照会、修正、削除などの操作とプロセスとの時間的な関係を確認するために利用する。複数のプロセスが同期をとり、待ち合せを行う様子を表現することができる。

# 待ち合せチャート

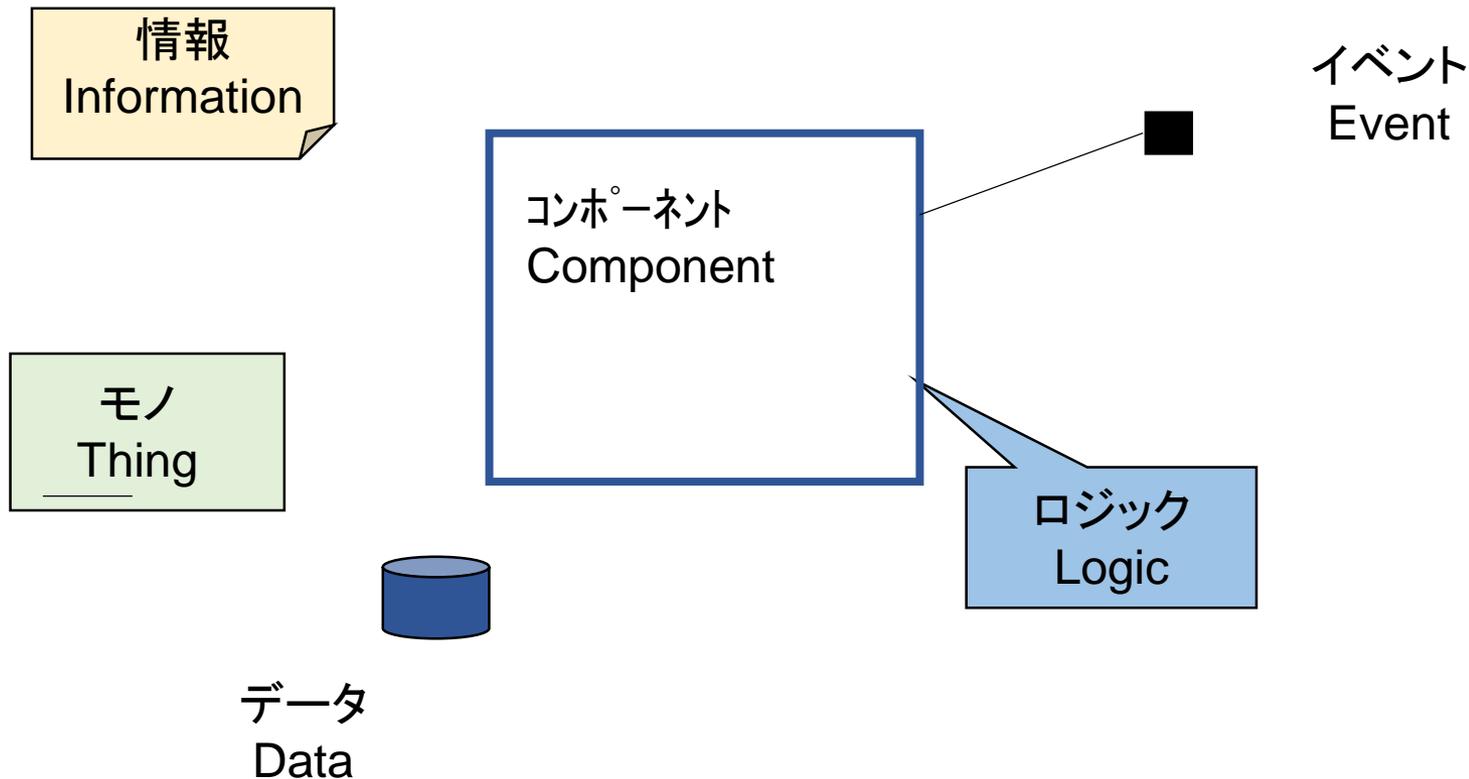


# トランスフォーメーション(課題解決)



**コンポーネントチャート**は、1つ以上のロジックやデータをまとめてコンポーネントとして定義した場合の構成内容を示す。また、ロジックが外部呼出し可能な場合に、コンポーネントが外部に提供するインタフェースとしてそのデータおよびプロセスを示す。

インタフェースはロジックに対応する  
ロジックはインタフェースを介してデータを外部アクセス可能とする



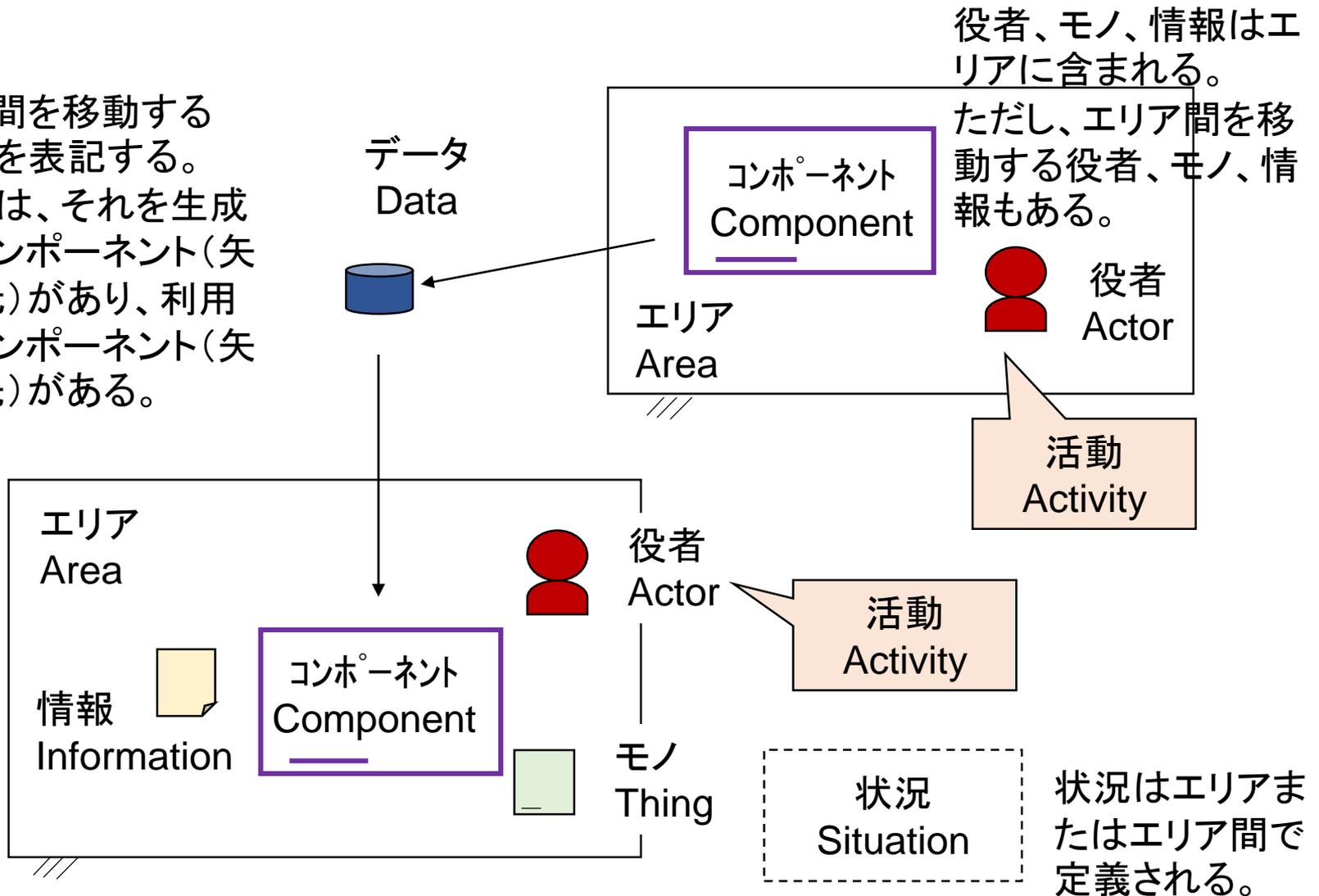
ロジックの機能を利用するためのプロセスが定義される  
ロジック側からイベントを受け取ることができる



**レイアウトチャート**は、コンポーネントが物理的に配置されたエリアを示し、エリア間でデータがどのように移動するかを示す。また、コンポーネントと関連して、各エリアで行われる活動およびその役者を示す。また、エリア内、またはエリア間で移動するモノ、情報をあわせて記述する。

# レイアウトチャート

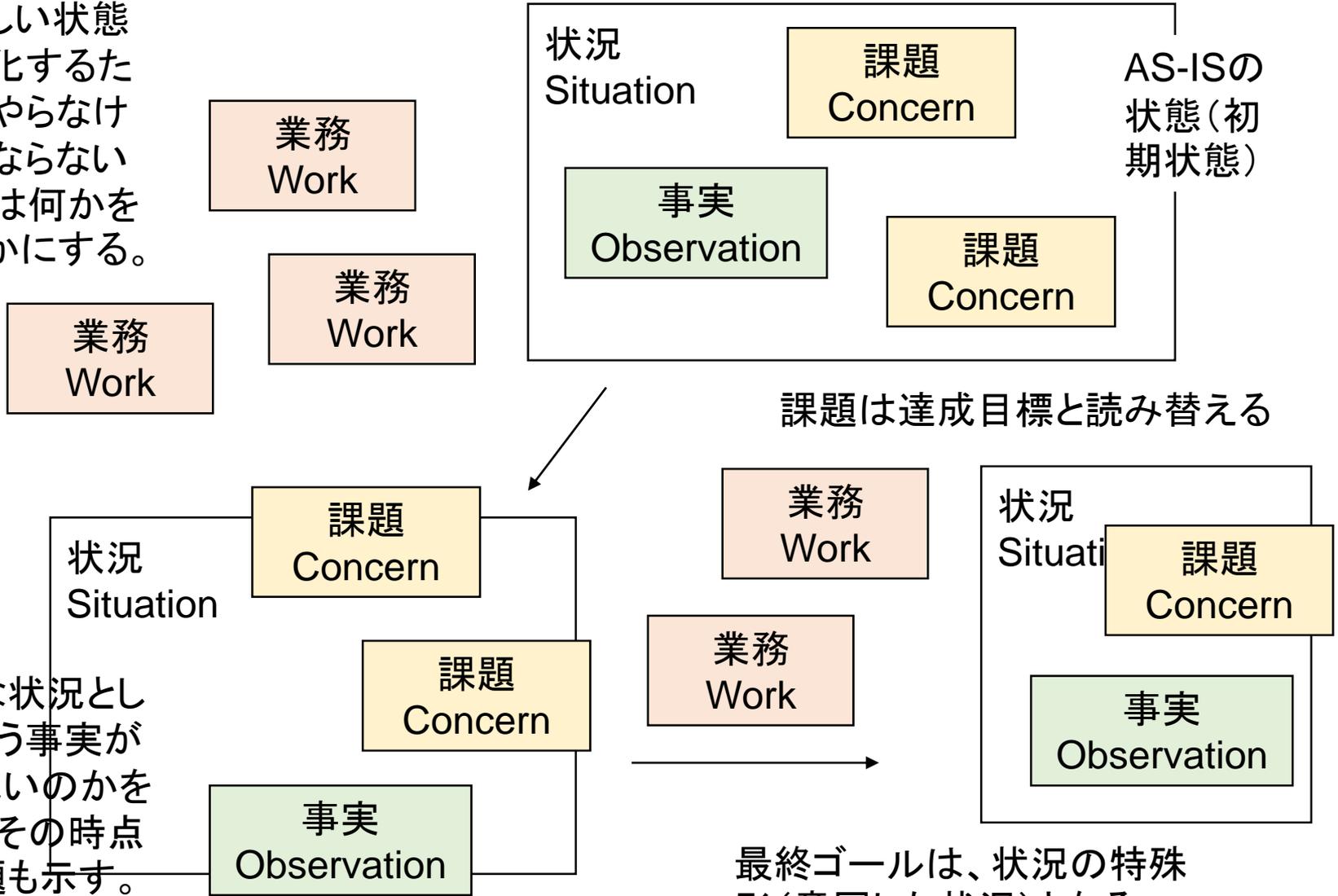
エリア間を移動するデータを表記する。データは、それを生成するコンポーネント(矢印の元)があり、利用するコンポーネント(矢印の先)がある。



**目標計画チャート**は、目標を達成するための段階的なステップを状況ごとに示し、ある状況から最終的に望ましい目標とする状況へと到達するための道のりを検討するために利用する。ある状況は、業務によって別の状況となり、最終的に目標としての状況となる。状況は業務によって枝分かれまたは合流する場合がある。

# 目標計画チャート

望ましい状態  
に変化するためにはやらなければならぬ  
業務は何かを  
明らかにする。



**割り振りチャート**は、目標を組織ごとに割り振り、短期、中期、長期などのカレンダー上に割り振り、それらの関係を概観するために利用する。組織を縦方向、期間(カレンダー)を横方向として目標を配置する。右に行くほど大目標となる。各目標には、その詳細内容として課題が設定できる。

# 割り振りチャート

短期、中期、長期といった時間軸に配置する。

どの組織が主に  
関与するかを明  
らかにする。

カレンダー  
Calendar

カレンダー  
Calendar

カレンダー  
Calendar

組織  
Organization

課題  
Concern



課題  
Concern



課題  
Concern

組織  
Organization

業務  
Work

課題  
Concern



業務  
Work

それぞれの組織  
がいつまでに何  
をしなければな  
らないかを示す。



