

Party-Driven Actors – A Conceptual Model

Rieks Joosten (rieks.joosten@tno.nl)

Within a large organization, the plethora of organizations (companies, governments) , individuals, and IT applications (and components), and the many ways they interact with each other, make it hard to find out who does what, or is responsible for what. This makes governance, risk-management and compliance very difficult.

The conceptual models that we introduce here provide the concepts (and language) that help us come to grips with this complexity. The models explain how actions can be performed by, or on behalf of, specific parties, what things are prerequisite for that, and how this can be governed, managed and controlled. Specifically, these concepts can also help provide the means and method by which parties can tell whether or not, or to what extent, work is being done by, or on behalf of, other parties, and determine who these parties are.

When we define a specific concept, we do so by stating the term we will use to refer to that concept (printed in **bold**), as well as the criterion that can be used to distinguish between what are instances (examples) of that concept, and what are not. The visual representations of the models use notations and conventions that are common in UML or ArchiMate, and should not be problematic.

Introduction

It is common practice to think actions being taken by individuals (people, natural persons) or organizations (enterprises, governments or governmental bodies, enterprises). This practice, however, poses problems when one needs to answer questions as presented above. One such problem arises from observing that organizations actually cannot do anything. An enterprise cannot hire employees, or sign contracts. They need people that do so on its behalf. Another problem arises when you see someone doing something, e.g., sign a contract. How can one tell whether this is done on behalf of the person itself, or on behalf of some organizations (and if so, what the organization is).

To come to grips with this, we propose a new way of looking at such situations, that is based on somewhat different concepts than that are commonly used, and collectively is called the "Party-Driven Actor model".

Here are some basic definitions (i.e., terms associated with criteria that enable us to distinguish between what is, and what is not, an instance (example) of a particular concept).

An **Entity** is something that is known to exist, e.g., a person, an organization, a computer, an extinct animal, a thought, an idea, a JSON-object – *anything* that anyone can think of as existing. Everything, tangible or not, still existing, extinct or as a future possibility, qualifies as an entity.

An **Actor** is an entity that can act (i.e., actually do things), such as people, or machines. Tables and stones are examples of entities that cannot act. And as we have seen, organizations, too, do not qualify as an actor. In order to continue using the common practice of saying that some organization has done something, we will take that to mean "there is an actor that does this something on behalf of that organization".

An **Action** is something that is actually done: a 'unit of work' that is executed by a single actor, as a single operation, in a specific context. Examples include "drafting a document", "signing a contract", "accepting an order". Business or other processes typically consist of a sequence of actions, where different actions therein may be executed by a single, or multiple actors (people, or machines).

It is typical for actions that they need to be done in particular ways, depending on whom they are done for. If a person is drafting and sending a letter, it matters whether (s)he does so for him/herself, or for the company (s)he works for. In the latter case, the letter may need to be printed on company paper, and the company may have defined styles (typography, layout, etc.), which the person would do otherwise when drafting and sending a letter on its own behalf.

A **Policy** is a (set of) rules, working-instructions, preferences and other guidance for the execution of one or more kinds of actions, and that (the execution) of such actions can be said to comply with.

Policies are to be created and maintained by those that need such actions to be executed on their behalf. You can say that policies are part of their knowledge i.e., the (intangible) sum of what they know.

It is easy to observe that individual people each have and maintain their own, subjective knowledge, which predominantly resides in their minds. However, one can also make the case for organizations to have a 'mind' in which they store 'their knowledge'. This mind would be cabinets with papers, databases, and the like, and the knowledge therein is what is represented by the figures, texts, bits, etc. contained therein.

A **Party** is an entity that sets its objectives, maintains one particular knowledge, and uses that knowledge to pursue its objectives in an autonomous (sovereign) manner, which includes onboarding actors to do (parts of) the actual work. One might say that they have a mind of their own. Typical examples are individuals and organizations.

Their **minds** (i.e., the facilities in which they store their subjective knowledge) and its contents (the knowledge) are what distinguishes one party from another, so every party is 1-1 related to its knowledge (mind). The mind of humans consists of parts of their brains (and bodies). The mind of an organization typically consists of systems such as databases, filing cabinets, etc.

Formalized PDA conceptual model

The relevance of having this concept is that we can now clearly express how the execution of actions work: an action is executed by a single actor (as a single operation, in a single context), on behalf of a single party, which means that it will execute that action in compliance with the policy that this party maintains (within its knowledge). This is illustrated in Figure 1.

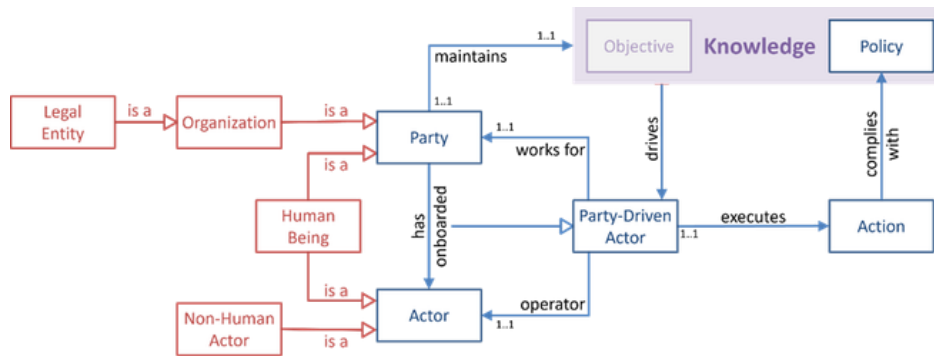


Figure 1: Conceptual Model Diagram for Party-Driven Actors

In this figure, the red boxes and arrows represent concepts and relations that are commonly used (including by ourselves) but are not to be considered as part of the Party-Driven Actor (PDA) model. Rather, they explain the relationships between these commonly used concepts and the concepts of the PDA model.

Note that human beings (people) not only qualify as a party, as they have a mind of their own, but they also qualify as an actor, as they can do things. This 'unity', that organizations lack, is often a cause for confusion in the design and of systems. In our way of thinking (i.e., the perspective of this model), it is easy to circumvent such confusions; we will come to this further on.

Party-Driven Actors (PDAs)

Figure 1 also shows the crucial concept of this model, which is the Party-Driven Actor.

Formally, a **Party-Driven Actor (PDA)** is the (conceptualized) relationship between a single party and a single actor that is the result of a party having onboarded the actor (which we will explain a bit later). The purpose of having a PDA is so to say that it executes actions, which must be taken to mean that the actor that is the operator of the PDA executes that action, and the party for which the PDA works maintains the knowledge that *drives* the PDA (hence the term 'party-driven actor'), and that contains the policies that the actor must follow it/the PDA executes actions.

Note that since a PDA is the (conceptualized) relationship between a single party and a single actor, it follows that any pair of PDAs must differ in either the party for whom the PDA works, or the PDA's operator, or both.

Informally, a PDA can be seen as an actor that operates (executes actions) on behalf of a single party, e.g., in their role as employee, or hired workforce. We considered to use the term 'persona', but it didn't quite fit, as a 'persona' is more a particular way of viewing an actor (specifically, a person) than that it establishes an entity of its own, of which the actor is one of the (two) parts. It might work because it reflects that an actor, as it works for different parties, has different 'faces' to show, as if it were different persons.

Informally, a PDA can also be seen as a party that does things (executes actions) as it pursues objectives that it has set (and that exist within its knowledge). As we have seen, parties that themselves are incapable of acting need actors to do these things on their behalf, which is precisely what a PDA adds. So, when an organization is said to hire an employee, this means that there is a PDA that works

for the organization (which is a party), which implies there is an actor that does the associated work, where the actor is onboarded (we'll get to that) by that party.

According to the PDA model, any text that mentions that either some party or some actor does something, MUST be interpreted to mean that there is a PDA that does this something, where the PDA works for that party, or the actor is the operator of that PDA respectively.

Onboarding

Every PDA represents a relationship between the party that it works for, and the actor that serves as the operator that does the actual, operational work. This relationship, which we call '**onboarding**' exists between a party and an actor during the timeframe in which:

1. The party (continually) ensures that the actor has the capabilities that are required (or handy) for doing the particular kinds of work that the party is tasking it with (which it does to realize its objectives). For example, when an organization wants to hire someone for a specific position, it first ensures that this person is suitable for that position. Similarly, when an organization wants to lease or buy software, it will first ensure that this software is fit for the purposes of the organization. Organizations will typically regularly evaluate the fitness of an actor for its tasks, and take actions either to get its capabilities enhanced, or terminate the relationship.
2. The party (continually) ensures that the rights and duties of the actor (towards the party itself, and/or others, as appropriate) are properly specified, maintained, and enforced if necessary. This may take the form of a contract, as is common between organizations and their employees, or organizations and parties from which they hire/lease machines or software. All this is to ensure that the actor, once it is onboarded, can be relied upon to work as the party expects it to (which includes adhering to the policies that the party specifies and maintains for the execution of actions on its behalf).
3. The party (continually) ensures that the circumstances and conditions exist that are needed for the actor to execute the actions that it tasked with. This includes e.g., providing the actor with appropriate roles/permissions, access to the policies that guide the execution of such actions, resources to work with, etc.

The PDA-model's Perspective on Human Beings

Human beings are special in the sense that they qualify both as a party and an actor, which is easy to verify by checking their definitions.

We postulate that every human being (as a party) has onboarded itself (as an actor) for the entire time that (s)he is alive, for any kind of action that it wants to execute itself. This results in the existence of every person having one (and only one) PDA whose party and actor are both this person.

Of course, a person can also onboard other actors, or be onboarded by other parties, which creates new PDAs in which the person is the party or the actor respectively.

It is easy to observe that individuals differ greatly, and not all of them have all capabilities that they might need or would like to have to onboard themselves, or other actors, in such a way that the resulting PDAs would operate to what is beneficial to the individual. The model allows for 'lousy' onboarding practices to exist, thereby acknowledging the autonomy that individuals (and other parties)

have to make their own decisions as good, bad, or ugly as they like. This is also in line with what conceptual models are for, i.e., come to grips with the world around us. Our conceptual models are not intended to model what good or bad judgments are.

The PDA-model's Perspective on Organizations and Legal Entities

PDA's where the operator and party are the same person are needed to create parties that themselves cannot act, such as enterprises, governments, communities, etc. What it takes is that one or more of these PDA's set objectives, and devise associated policies and other knowledge, where none of these PDA's individually controls them. Rather, they work together to govern and manage such objectives, and to ensure everything gets done to realize them.

Such collaborations can be very informal, e.g., a group of friends that decide to promote social bonding and emotional support, e.g., by sharing hobbies or interests, serving as a support network for its members, celebrating achievements, by group volunteering or community service.

However, collaborations can also arrange for complying with rules/laws in some jurisdiction for the purpose of being recognized as a legal entity (i.e.: an entity that is known within some jurisdiction), which typically means that the collaboration gets rights and duties (such as the right to sue and be sued) that are similar to human beings.

Appendix 1: Envisaged Benefits of Adopting the PDA Model

1. **Clarity and Precision:** The PDA model offers a clear and precise way of understanding and describing complex relationships between parties, actors, actions, and policies. By providing well-defined definitions and concepts, it eliminates ambiguity and enhances communication and understanding.
2. **Accurate Representation of Reality:** The model addresses the inherent limitations in traditional ways of attributing actions to individuals or organizations. It recognizes that organizations, as entities, cannot directly perform actions but rely on actors to act on their behalf. By acknowledging this distinction, the PDA model provides a more accurate representation of how actions are executed in reality.
3. **Improved Design and System Development:** Adopting the PDA model can lead to more effective design and development of systems, processes, and organizations. By clearly identifying the relationships between parties and actors and their respective roles, it becomes easier to define responsibilities, assign tasks, and align policies and objectives. This clarity can help avoid confusion and miscommunication during system design and implementation.
4. **Enhanced Accountability and Compliance:** The PDA model enables better tracking of accountability and compliance. By explicitly linking actions to parties and their policies, it becomes easier to trace the origin and responsibility for specific actions. This can be valuable in auditing, regulatory compliance, legal proceedings, and ensuring that actions are aligned with organizational objectives and standards.
5. **Efficient Collaboration and Resource Allocation:** Understanding the PDA model can facilitate more efficient collaboration and resource allocation within organizations and across different parties. By clearly defining the relationships between parties and actors, it becomes easier to assign tasks, delegate authority, and optimize resource allocation based on individual capabilities and organizational needs.
6. **Flexibility and Adaptability:** The PDA model accommodates different scenarios and levels of autonomy. It recognizes that individuals can be both parties and actors, allowing for flexibility in self-onboarding and onboarding others. This flexibility supports various organizational structures, such as partnerships, collaborations, and hierarchical relationships, while maintaining the clarity of roles and responsibilities.
7. **Alignment with Legal and Regulatory Frameworks:** The PDA model aligns well with legal and regulatory frameworks, especially when dealing with liability, contractual obligations, and compliance. By explicitly defining the relationships between parties and actors and their associated rights and duties, it provides a solid foundation for legal interpretations and contractual agreements.
8. **Improved Decision-Making and Governance:** By understanding the PDA model, decision-makers can have a more comprehensive view of the parties involved and their knowledge (including policies) when making strategic decisions. This can lead to better-informed governance, improved risk management, and more effective resource allocation.

Glossary

The following table contains the definitions for the terms used in this document:

Term	Criterion
Action	something that is actually done – a 'unit of work' that is executed by a single actor, as a single operation, in a specific context. Examples include "drafting a document", "signing a contract", "accepting an order".
Actor	an entity that can act (i.e., do things), such as people, or machines. Tables and stones are examples of entities that cannot act. Organizations (corporations, governments, etc.) also do not qualify as an actor.
Entity	is something that is known to exist, e.g., a person, an organization, a computer, an extinct animal, a thought, an idea, a JSON-object – <i>anything</i> that anyone can think of as existing. Everything, tangible or not, still existing, extinct or as a future possibility, qualifies as an entity.
has onboarded	see: Onboarding.
Legal Entity	an entity that is known to exist within a particular jurisdiction. These typically include citizens, corporations, and other organizations.
Mind (of a party)	the combined facilities that the party uses to store its knowledge. The mind of humans consists of parts of their brains (and bodies). The mind of an organization typically consists of systems such as databases, filing cabinets, etc.
Onboarding	the relationship between a single party and a single actor , that exists during a (continuous) timeframe, in which the party continually ensures that 1. the actor has the capabilities required to perform the tasks it is expected to be doing, 2. the rights and duties between the actor and party are properly specified, maintained and enforced, and 3. the circumstances and conditions exist that the actor needs to perform the tasks it is expected to be doing.
Operator (of a PDA)	the actor that will do the actual work when we say that PDA is executing some action .
Party	an entity that sets its objectives, maintains one particular knowledge, and uses that knowledge to pursue its objectives in an autonomous (sovereign) manner, which includes onboarding actors to do (parts of) the actual work. One might say that they have a mind of their own. Typical examples are individuals and organizations. The (contents of their) minds (subjective knowledge) are what distinguishes one party from another, so every party is 1-1 related to its knowledge (mind).

Party-Driven Actor (PDA)	the (conceptualized) relationship between a single party and a single actor that is the result of that party having onboarded the actor . The purpose of having a PDA is so to say that it executes actions , which must be taken to mean that the actor that is the operator of the PDA executes that action , and the party for which the PDA works maintains the knowledge that drives the PDA (hence the term 'party-driven actor'), and that contains the policies that the actor must follow it/the PDA executes actions.
Policy	a (set of) rules, working-instructions, preferences and other guidance for the execution of one or more kinds of actions , and that (the execution) of such actions can be said to comply with.
Works for	the party that takes responsibility for the work that (the operator of) a PDA performs as it executes an action .