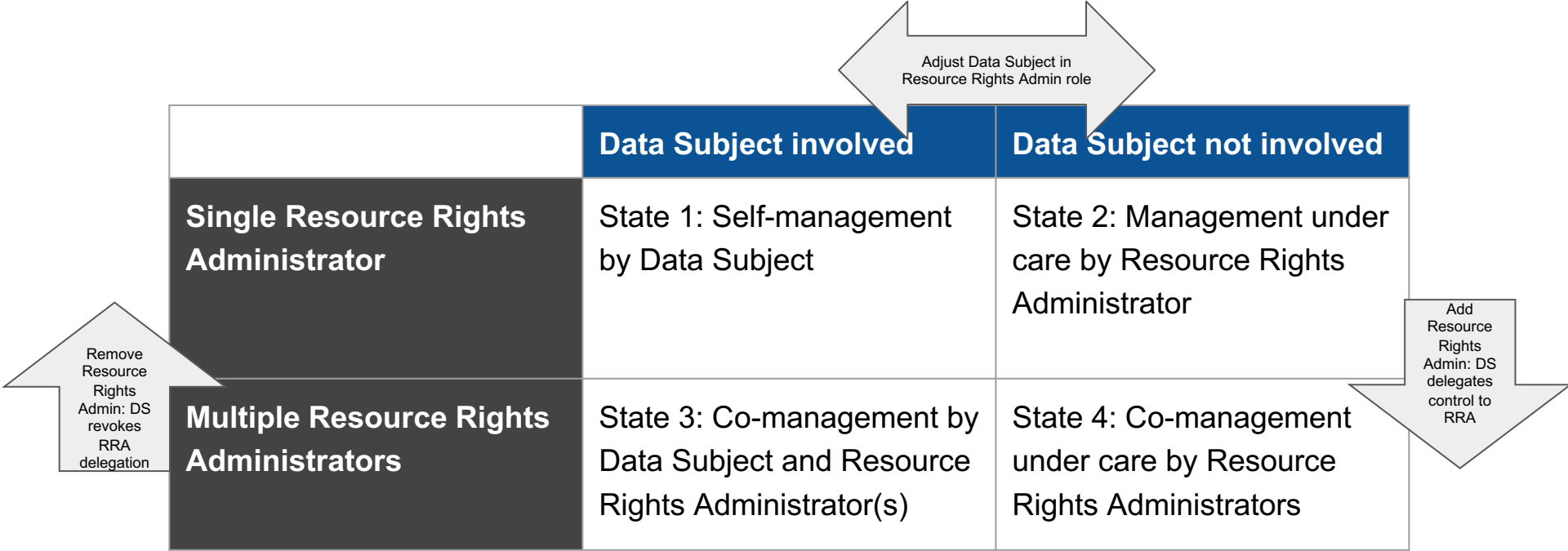


UMA Business Model: Mapping graphics

States related to access granting side

	Data Subject involved	Data Subject not involved
Single Resource Rights Administrator	State 1: Self-management by Data Subject	State 2: Management under care by Resource Rights Administrator
Multiple Resource Rights Administrators	State 3: Co-management by Data Subject and Resource Rights Administrator(s)	State 4: Co-management under care by Resource Rights Administrators

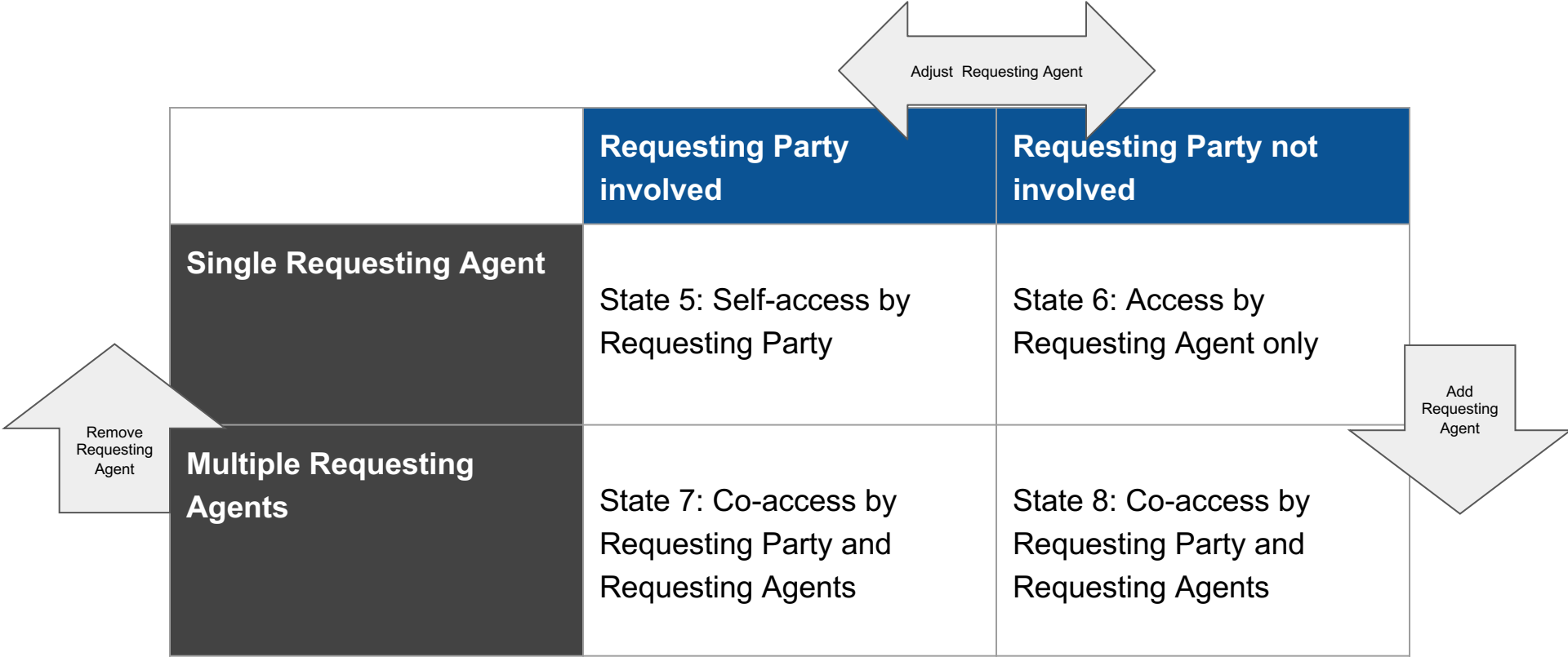
State changes related to access granting side



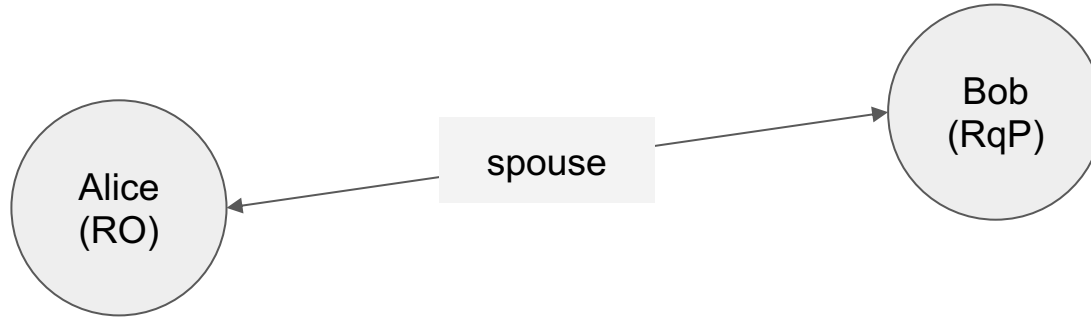
States related to requesting side

	Individual Requesting Party involved	Individual Requesting Party not involved, or Legal Person Requesting Party
Single Requesting Agent	State 5: Self-access by Requesting Party	State 6: Access by Requesting Agent only
Multiple Requesting Agents	State 7: Co-access by Requesting Party and Requesting Agents	State 8: Access by Requesting Agents only

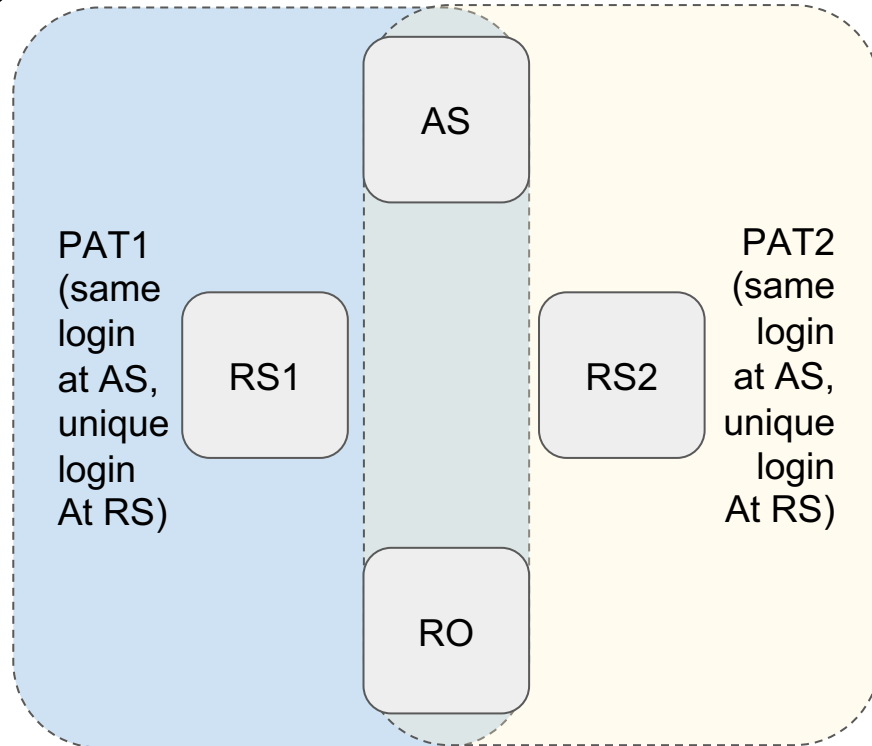
State changes related to requesting side



Policy changes based on relationship changes

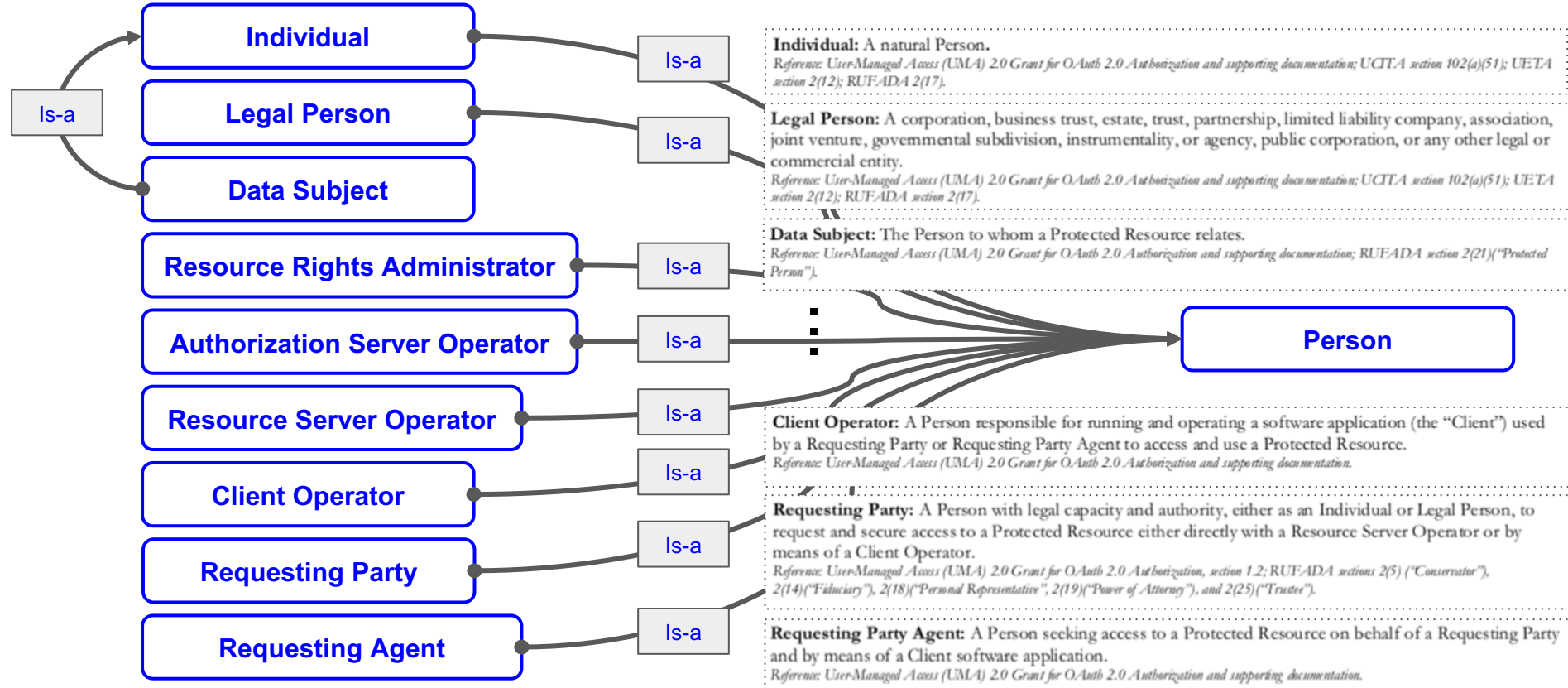


An RO (RRA) and their relationship to an AS and many RS's



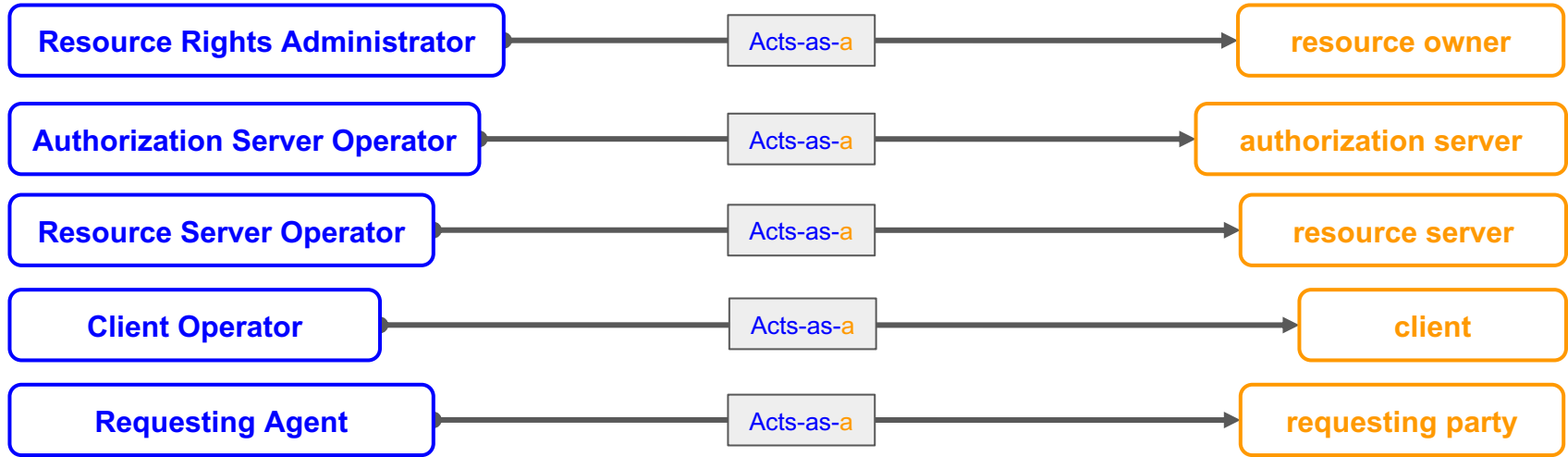
Legal relationships: Persons

Establishing basic party roles: Individual, Legal Person, Data Subject



Legal relationships: Legal-to-technical role bridges

Establishes how parties in legal roles can take part in UMA messaging flows



resource owner

An entity capable of granting access to a protected resource, the "user" in User-Managed Access. The resource owner MAY be an end-user (natural person) or MAY be a non-human entity treated as a person for limited legal purposes (legal person), such as a corporation.

requesting party

A natural or legal person that uses a client to seek access to a protected resource. The requesting party may or may not be the same party as the resource owner.



Legal relationships: Business relationship types

- Delegates authority for granting and managing access permissions to: **Delegates-perm-authority-to**
 - aka Agency Contract
- Delegates resource management to: **Delegates-mgmt-to**
 - aka Access Contract
- Licenses granting access permissions to: **Licenses-perm-granting-to**
- Licenses receiving access permissions to: **Licenses-perm-getting-to**
- Delegates access seeking authority to: **Delegates-seek-authority-to**
- Delegates permission to know/persist to: **Permits-knowing-claims**
- Party in role A also acts in role B: **Acts-as-a**

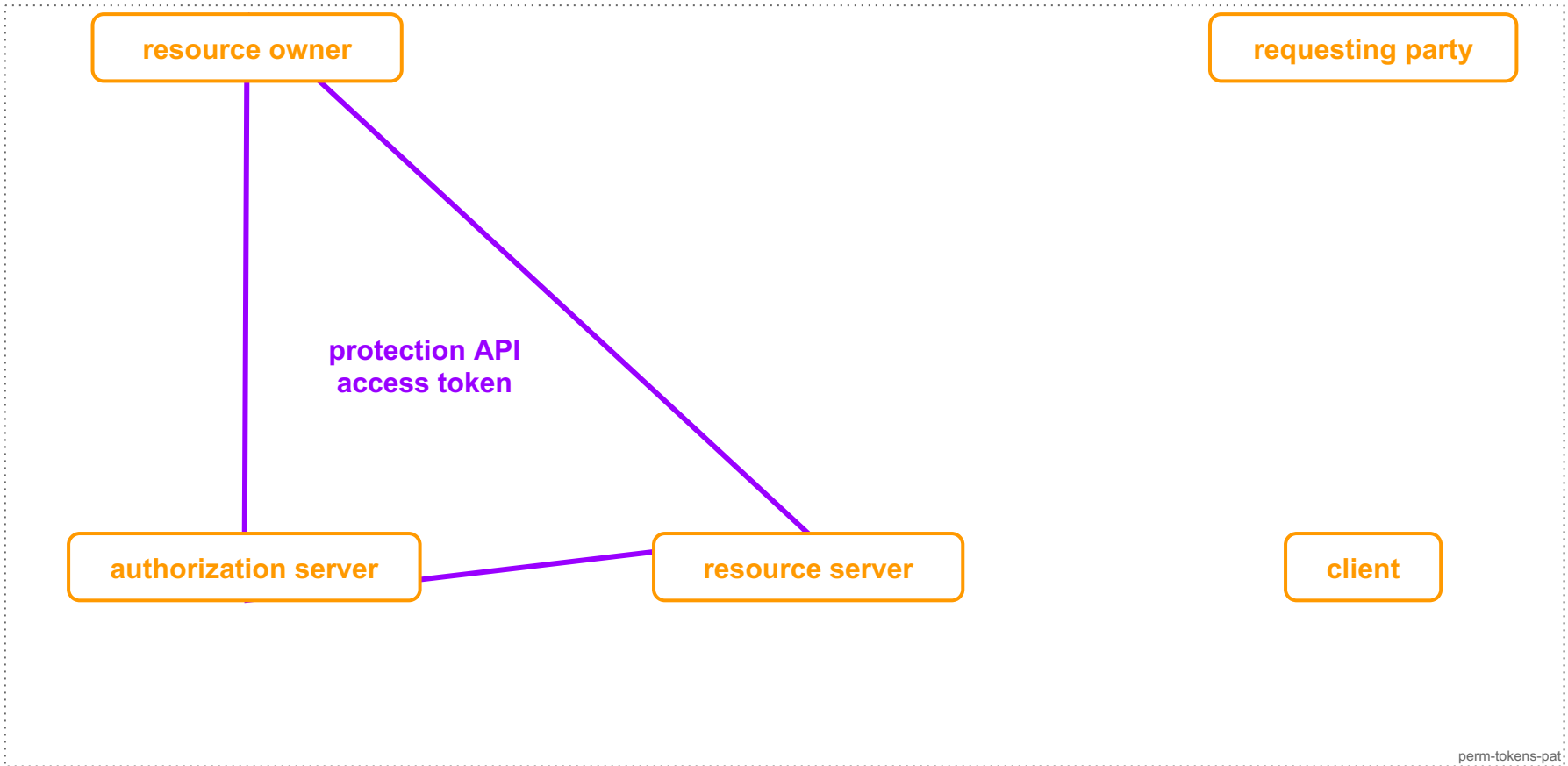
resource owner

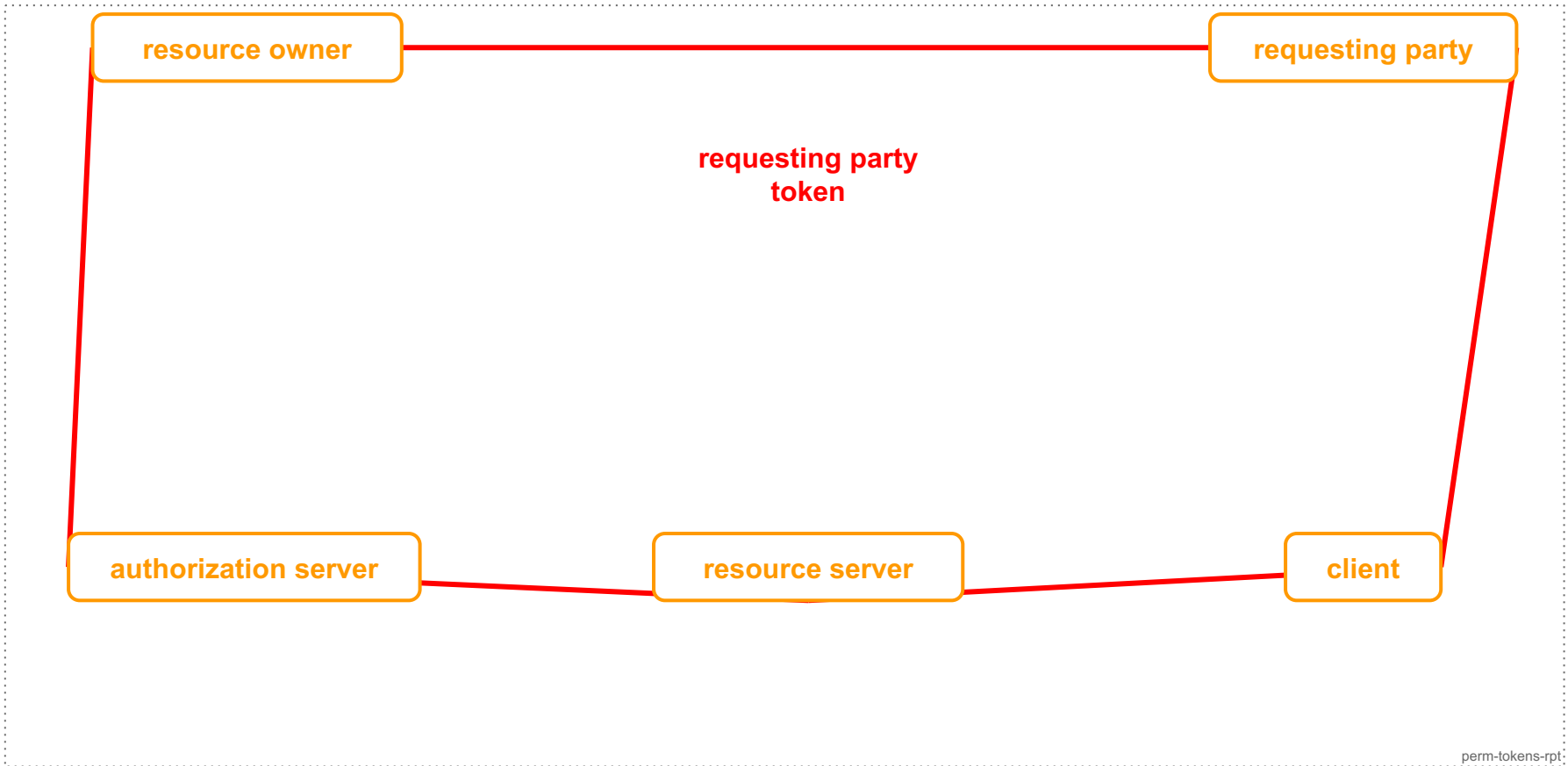
requesting party

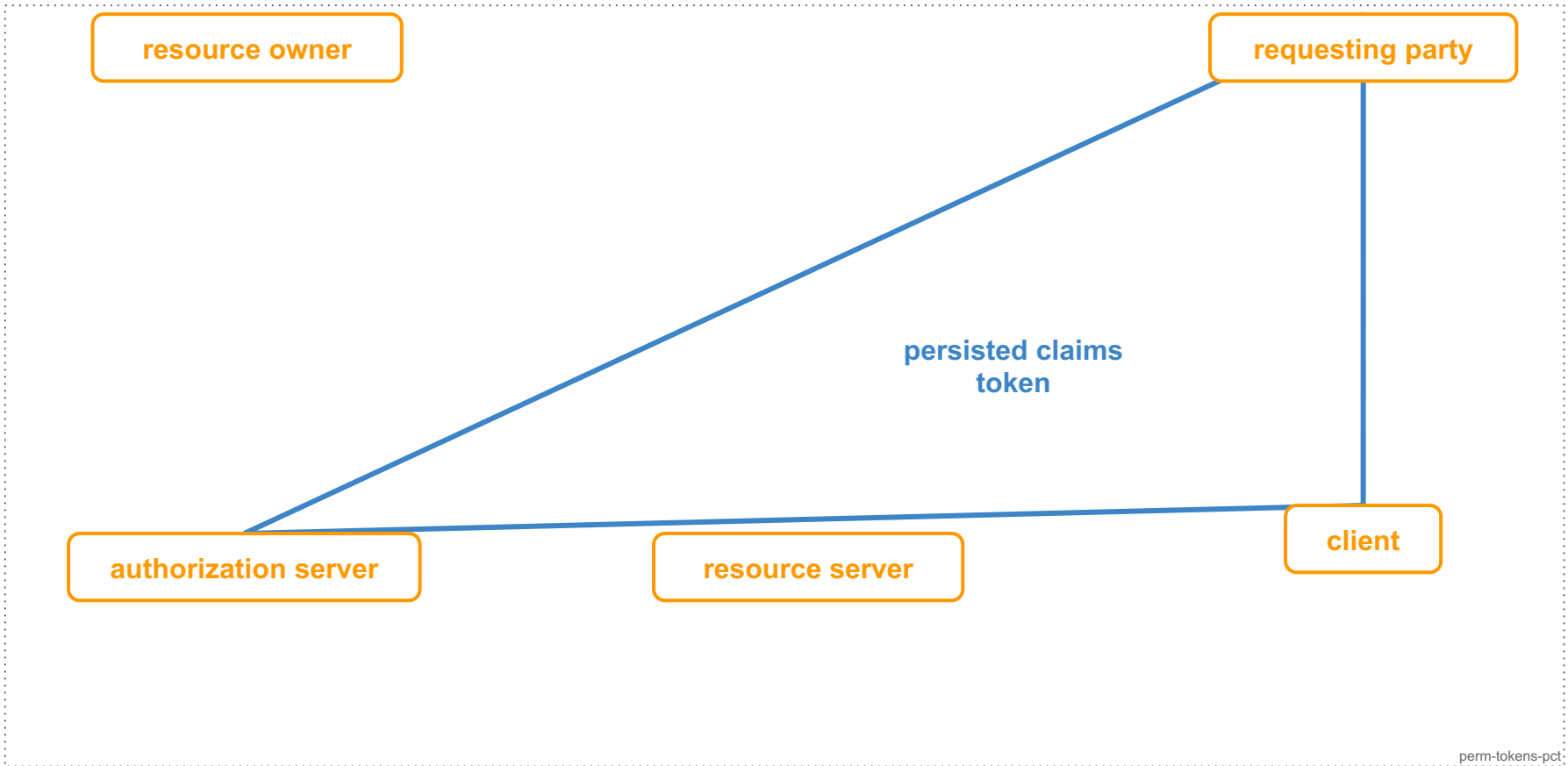
authorization server

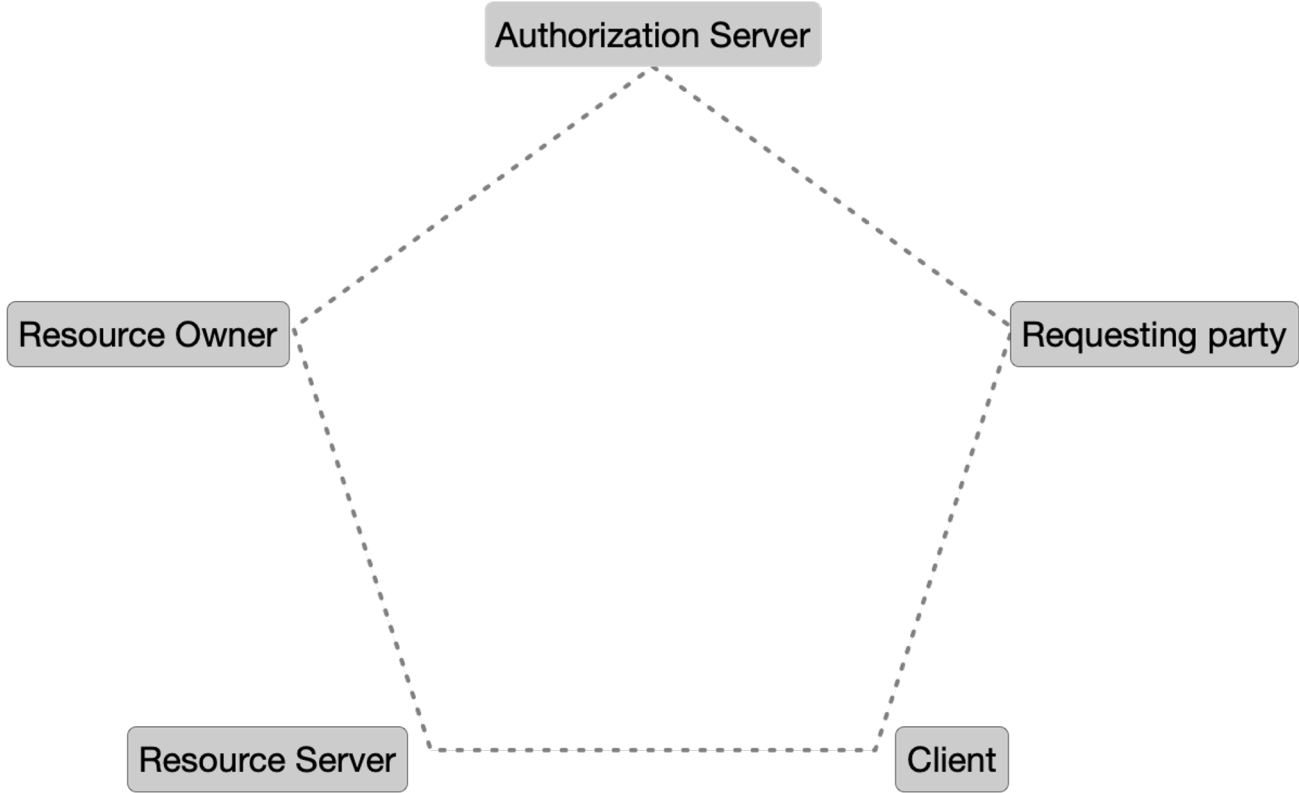
resource server

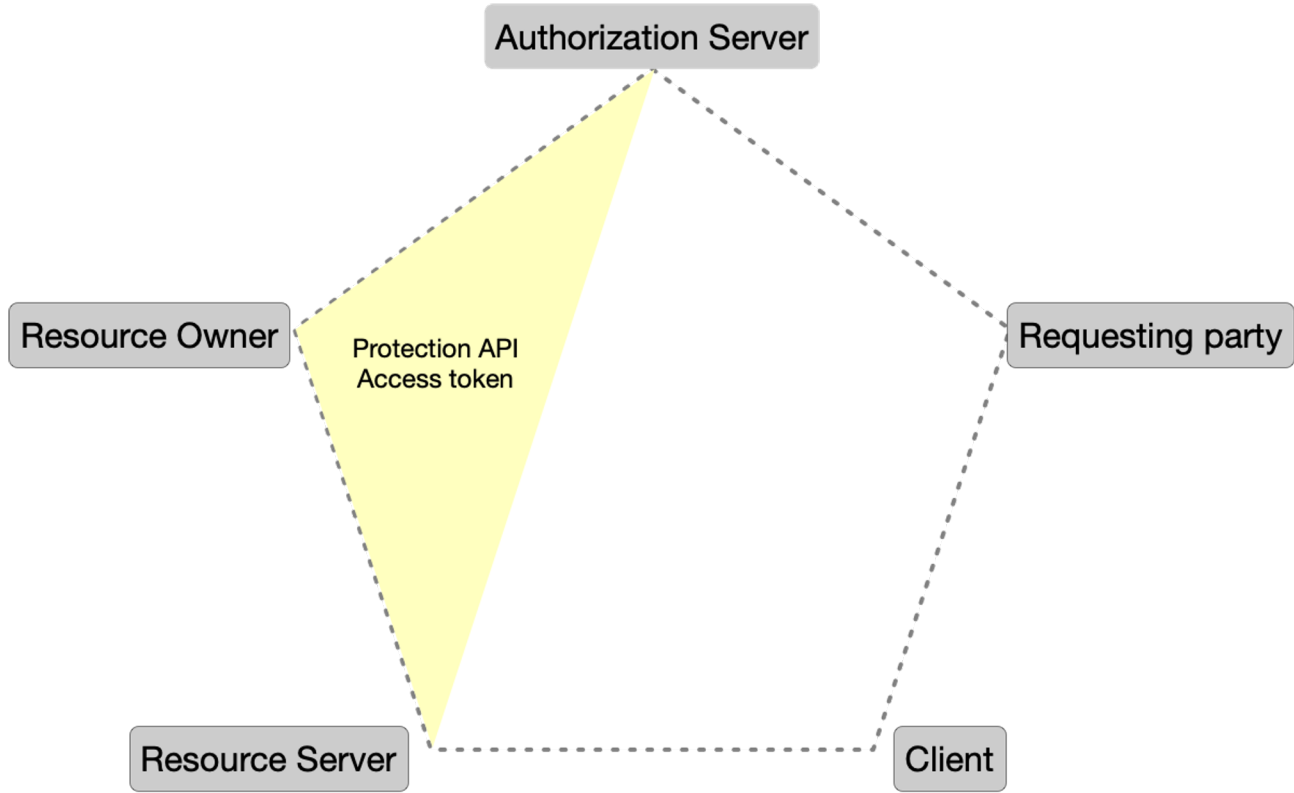
client

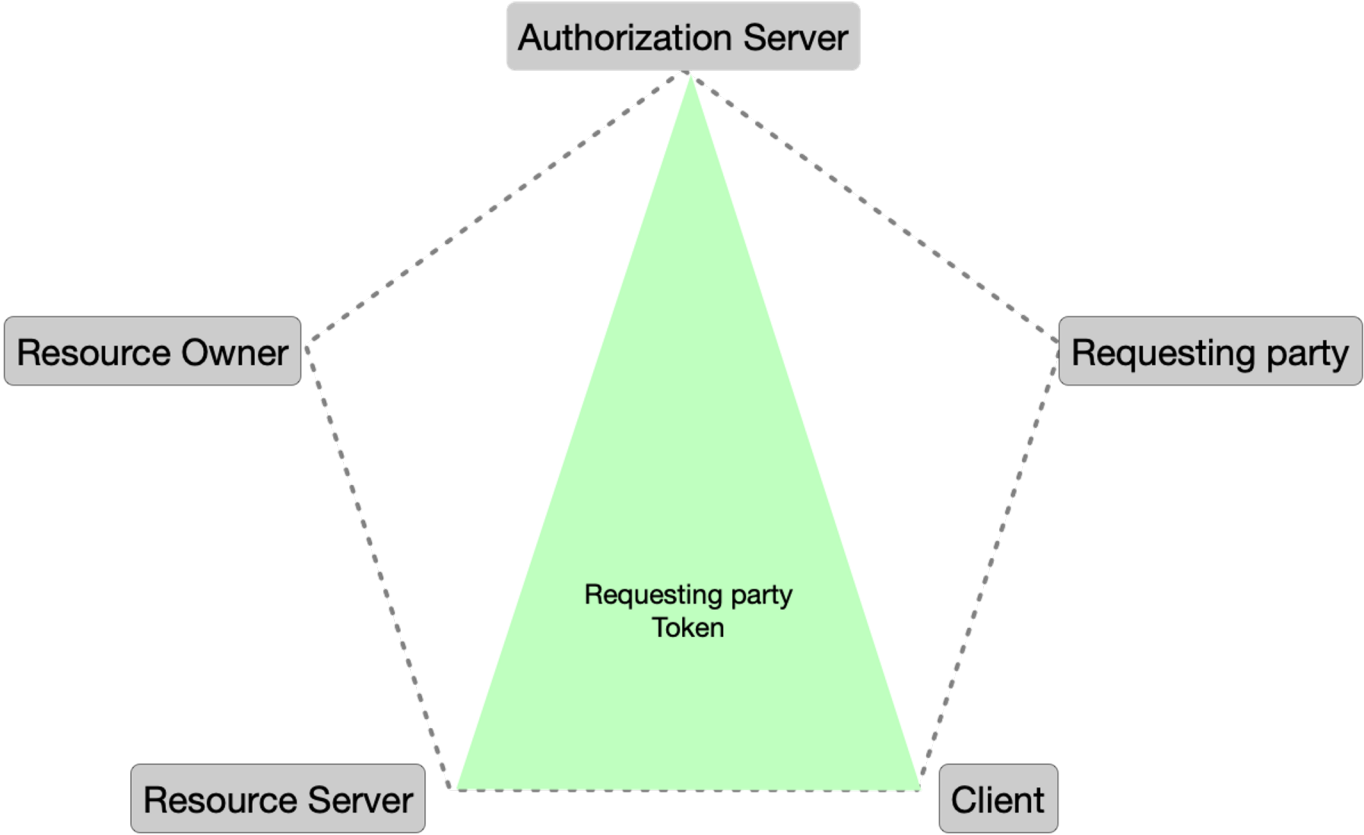


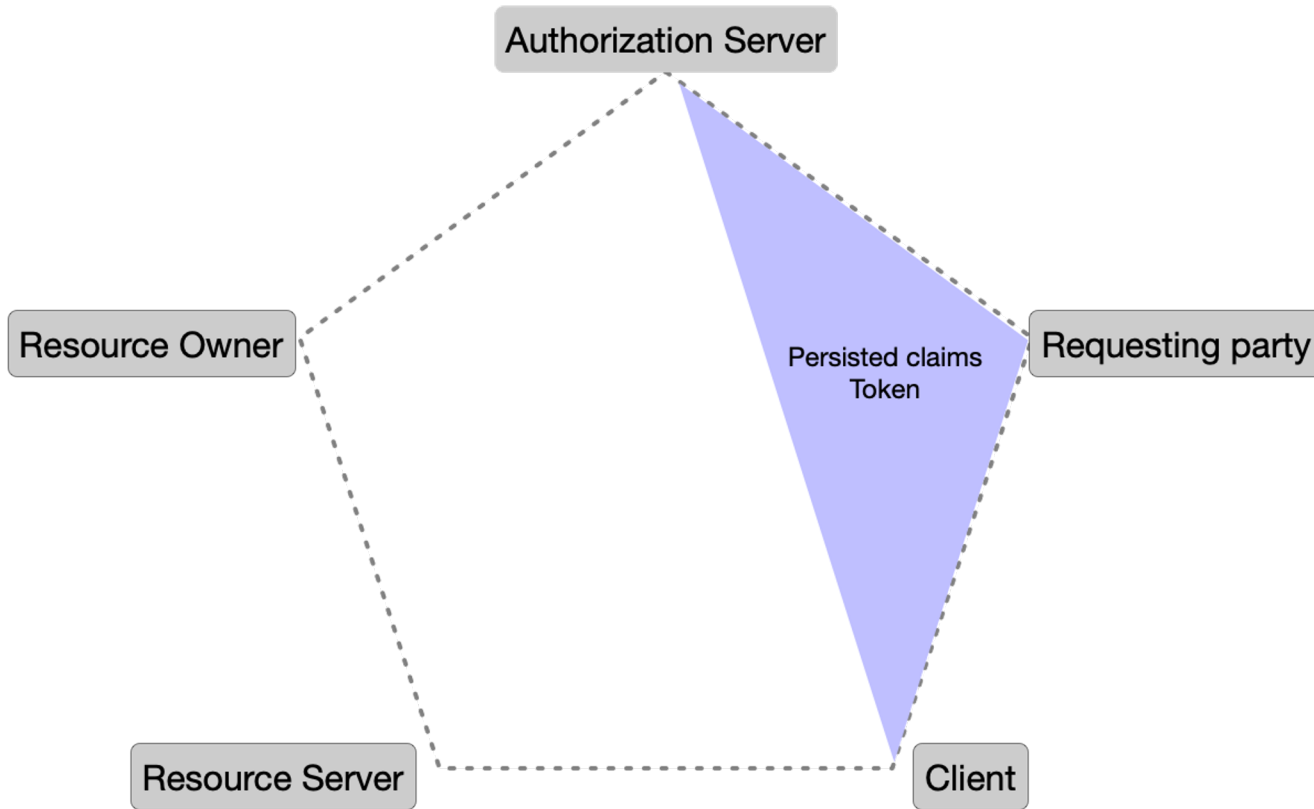


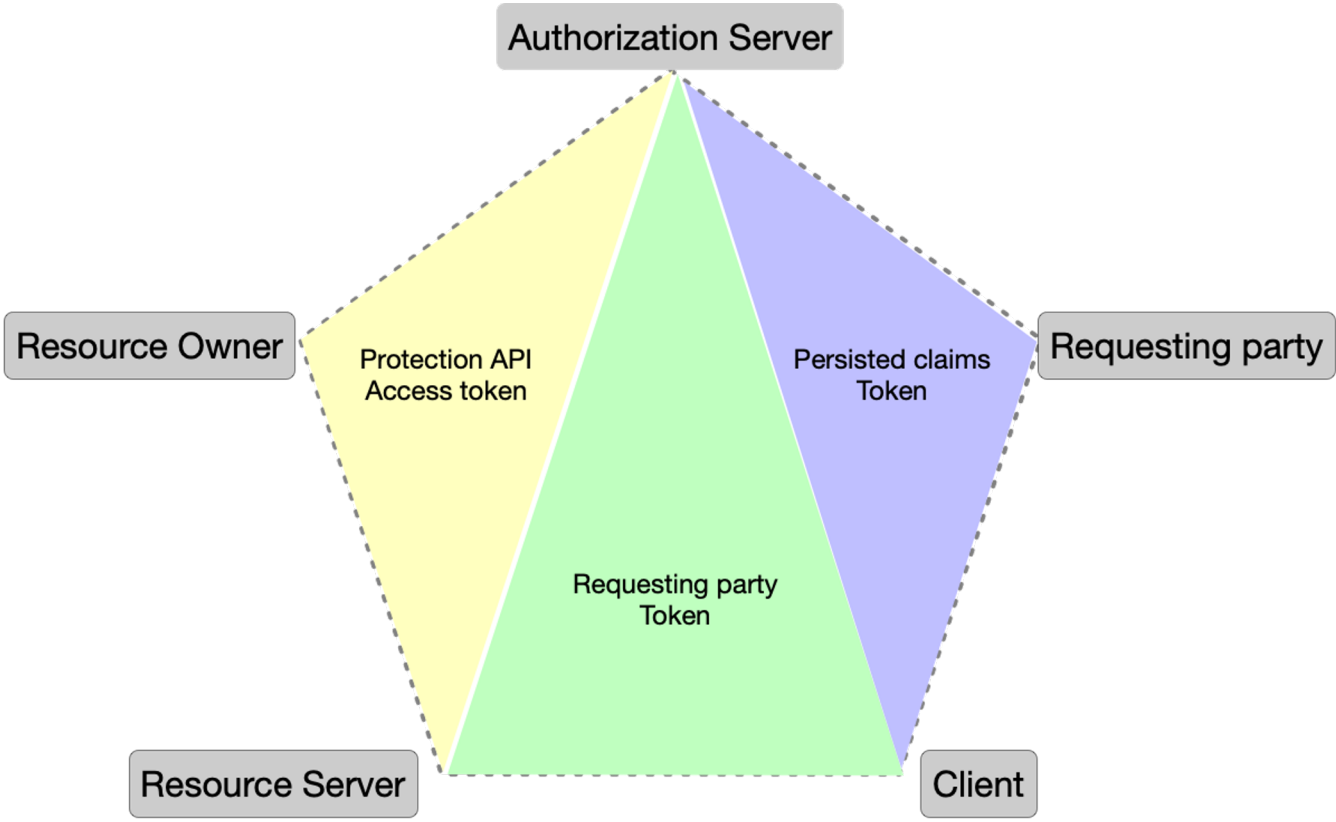


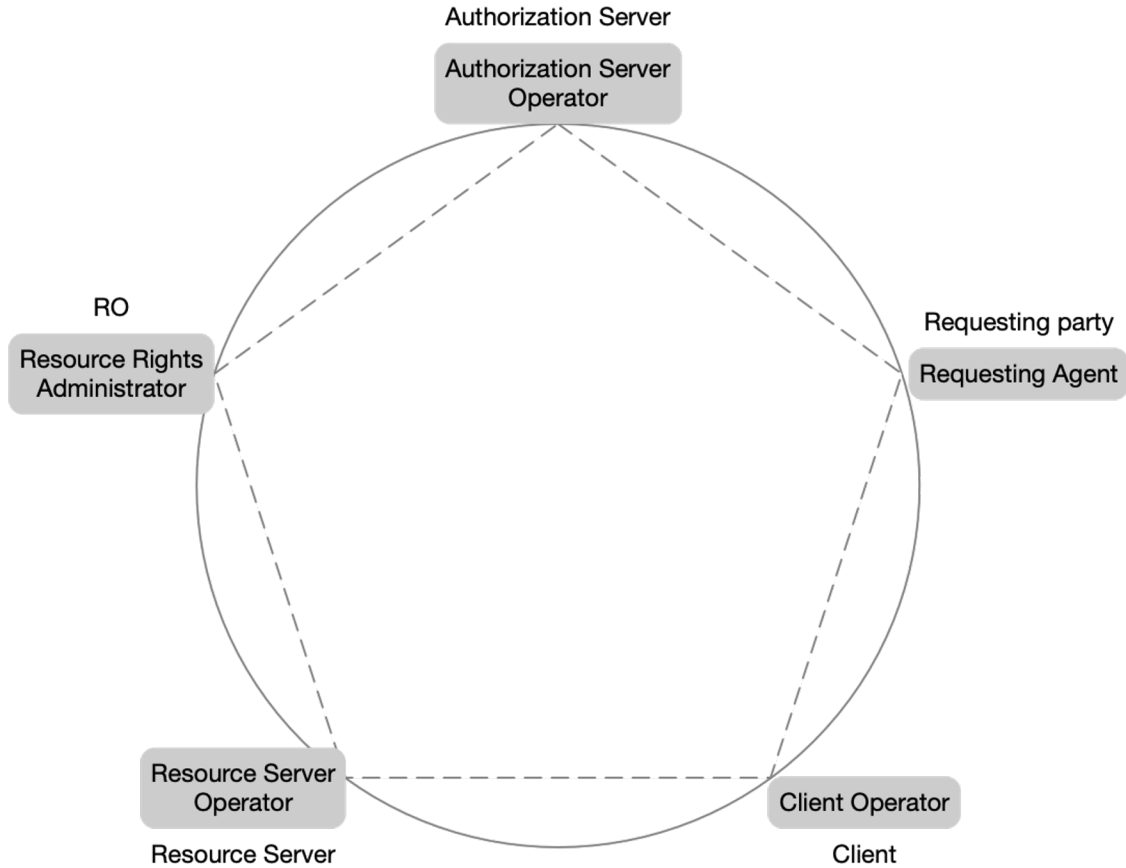


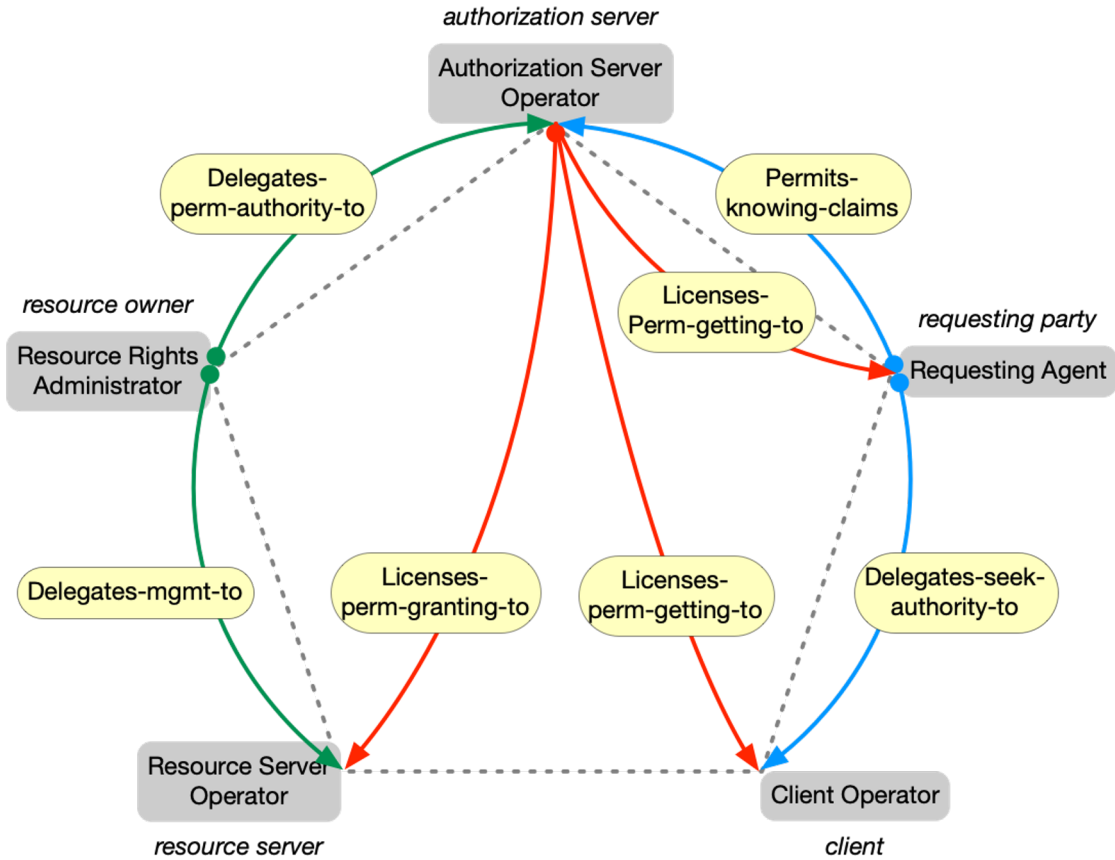












resource owner

Resource Rights Administrator

requesting party

Requesting Agent

authorization server

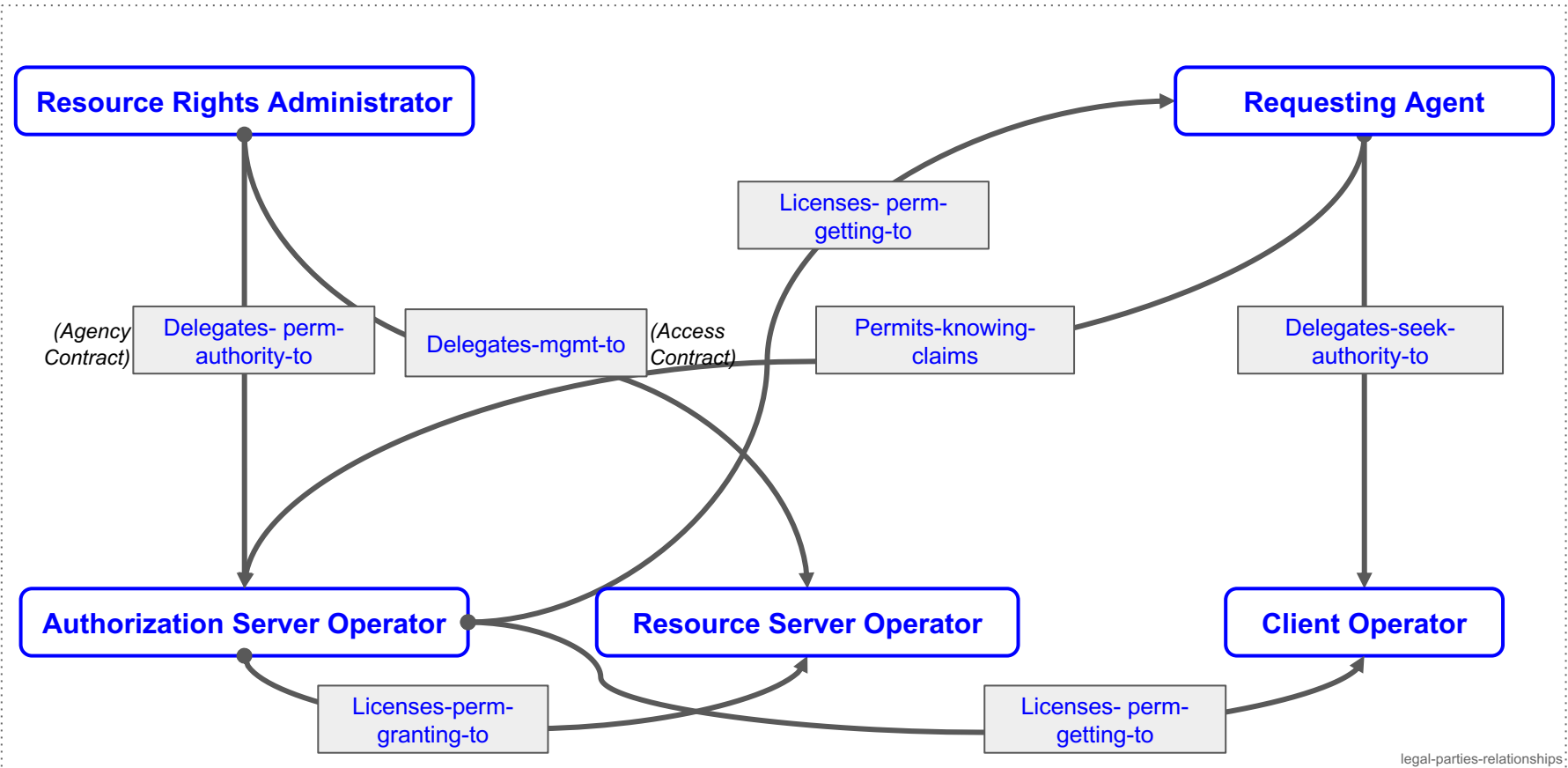
Authorization Server Operator

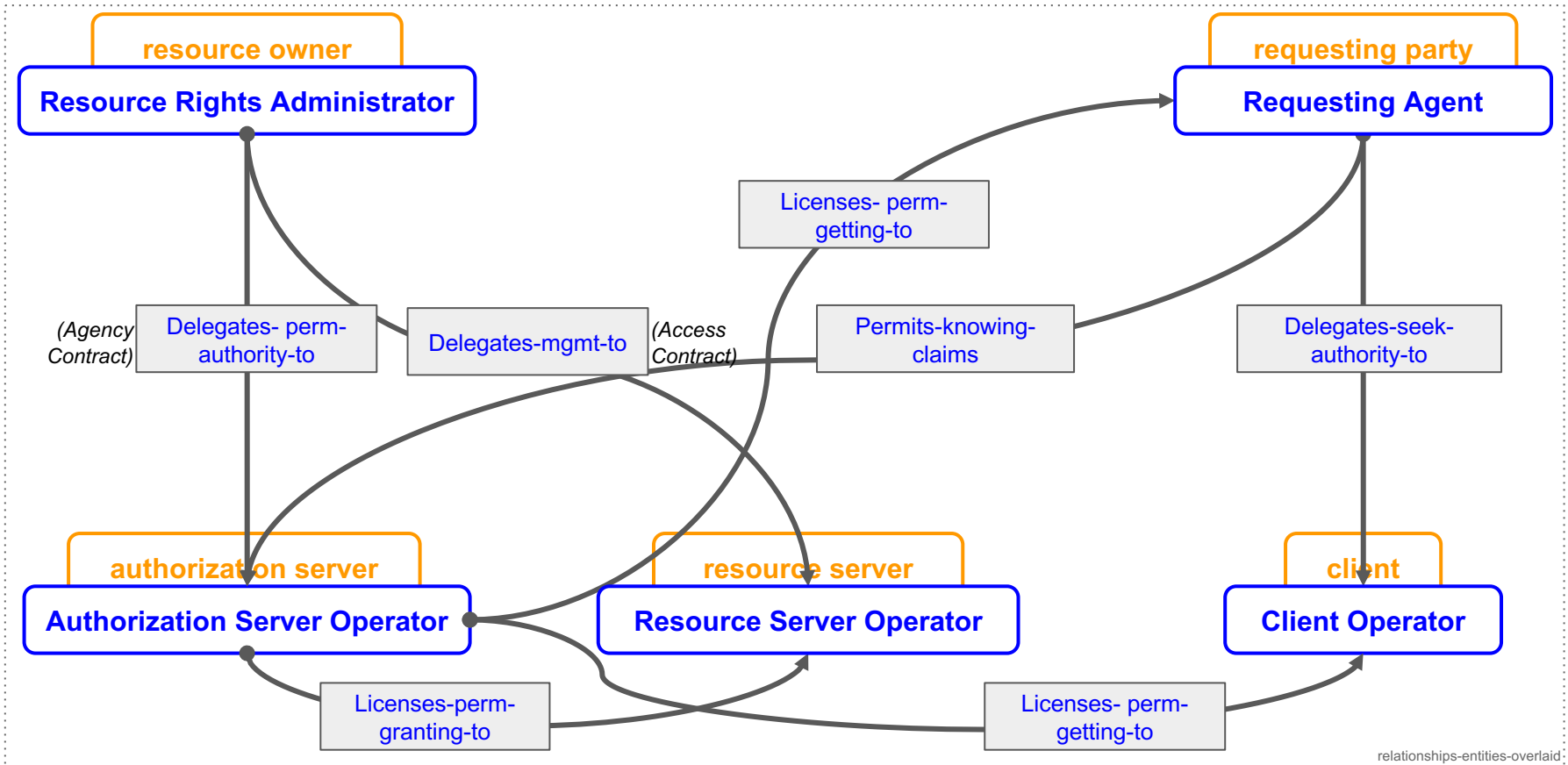
resource server

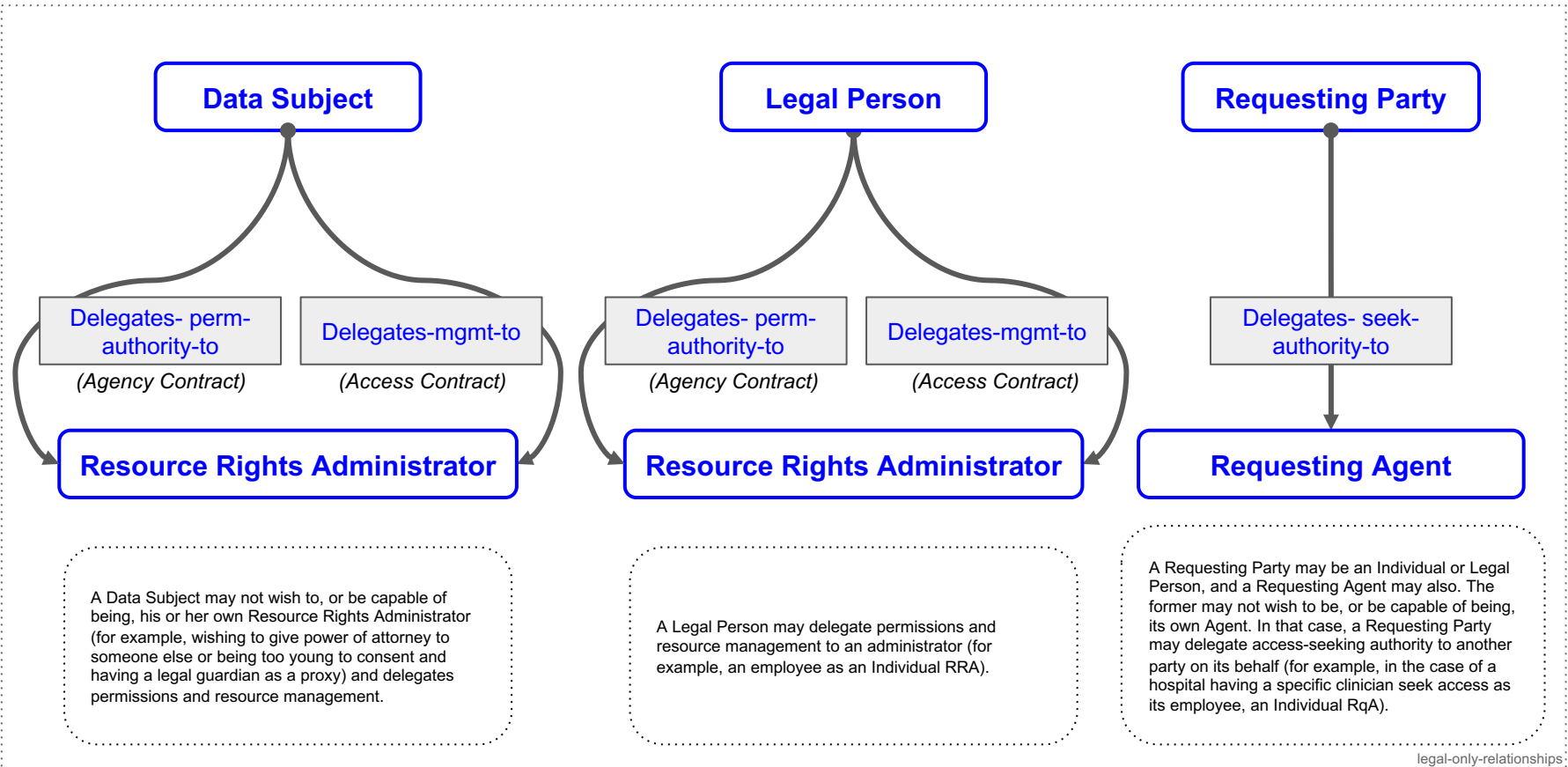
Resource Server Operator

client

Client Operator

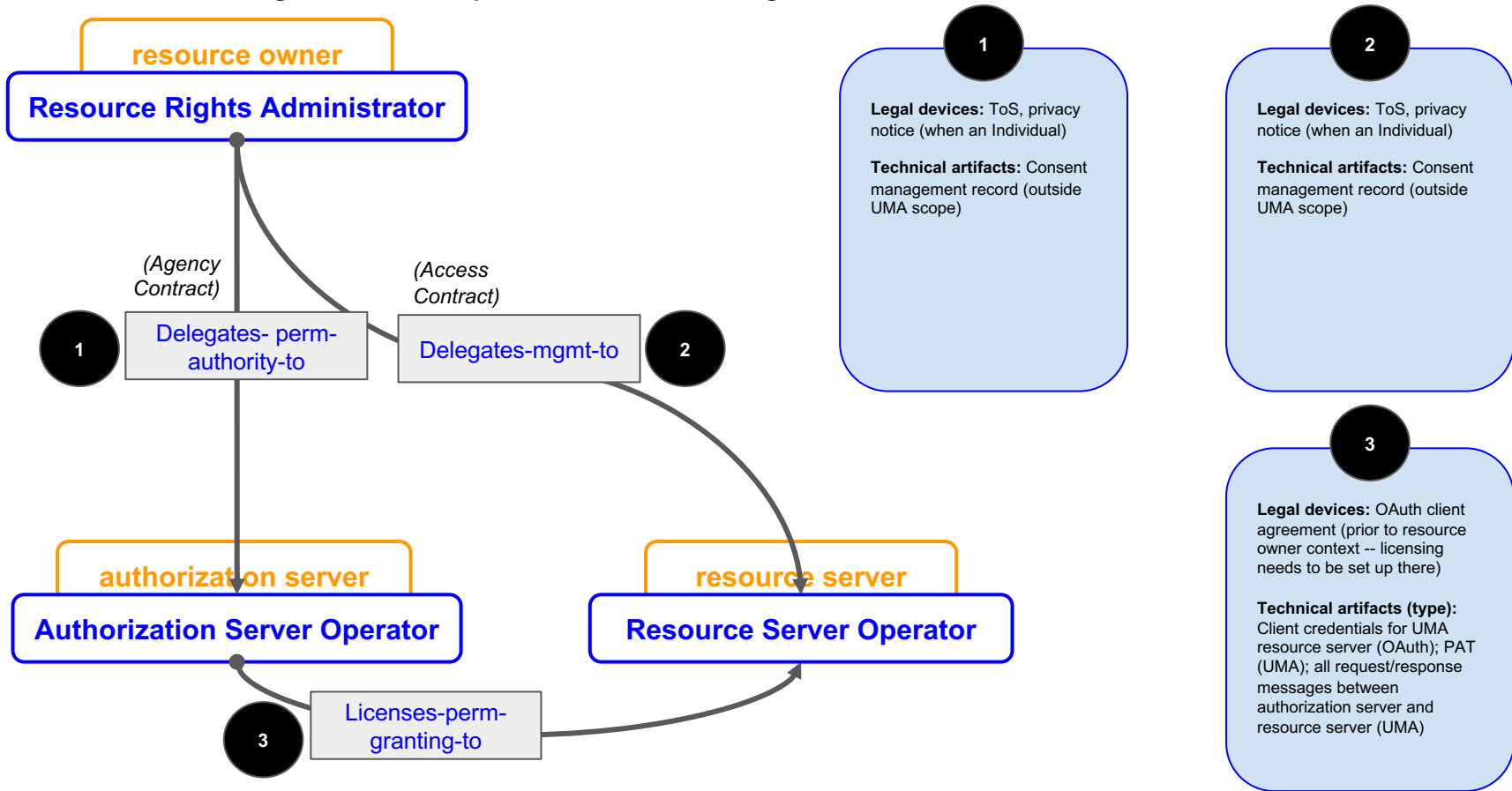






Legal relationships: Devices and artifacts

Making relationships and their changes auditable and machine-readable



Legal relationships: Devices and artifacts

Making relationships and their changes auditable and machine-readable

4

Legal devices: OAuth client agreement for UMA client (prior to requesting party context -- licensing needs to be set up there)

Technical artifacts (type): Client credentials for UMA client (OAuth); RPT (UMA); claim token (UMA); all authorization server/client request/response messages (UMA); policies (outside the scope of UMA)

5

Legal devices: (License builds on previous devices and is carried through technical artifacts)

Technical artifacts (type): PCT (UMA), all authorization server/requesting party request/response messages (UMA) *(These are all front-channel messages; what are options for auditing?)*

Licenses- perm-
getting-to

5

requesting party

Requesting Agent

authorization server

Authorization Server Operator

client

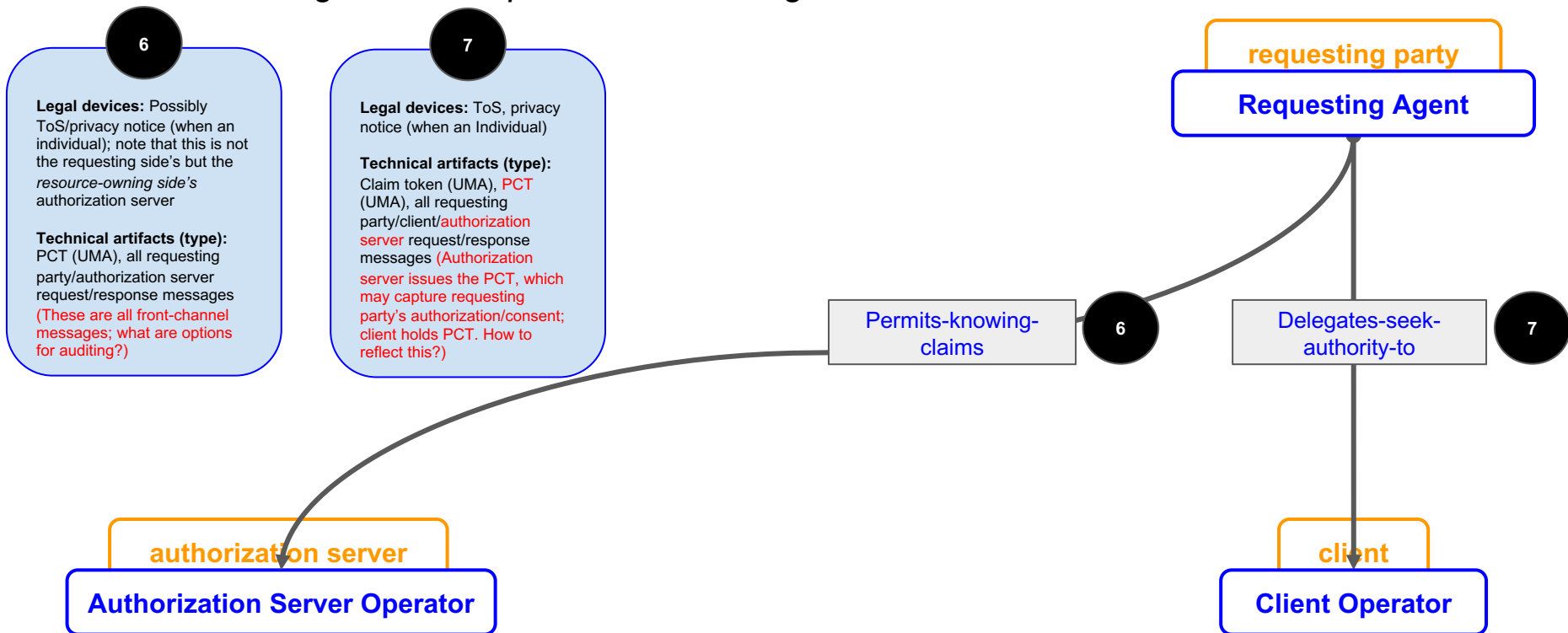
Client Operator

4

Licenses- perm-
getting-to

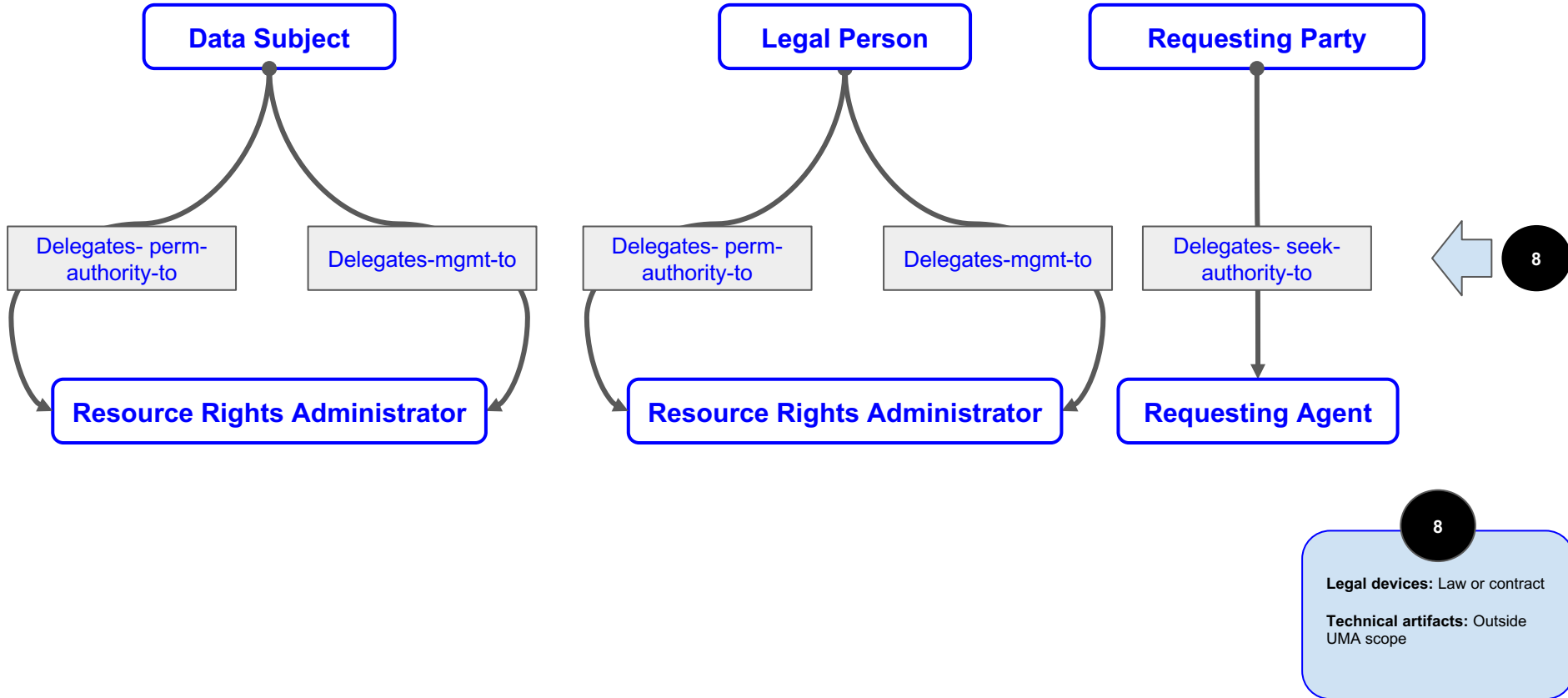
Legal relationships: Devices and artifacts

Making relationships and their changes auditable and machine-readable



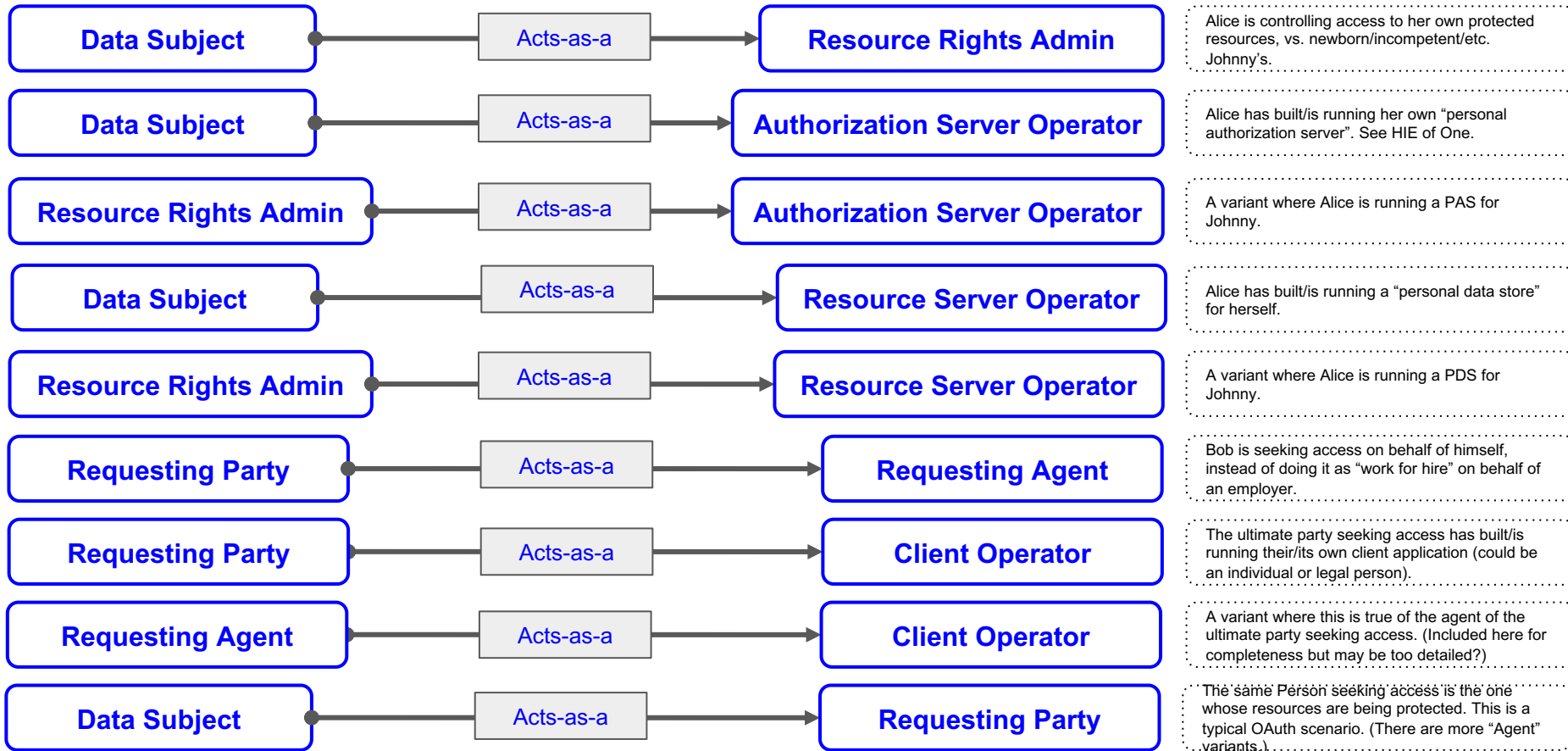
Legal relationships: Devices and artifacts

Making relationships and their changes auditable and machine-readable



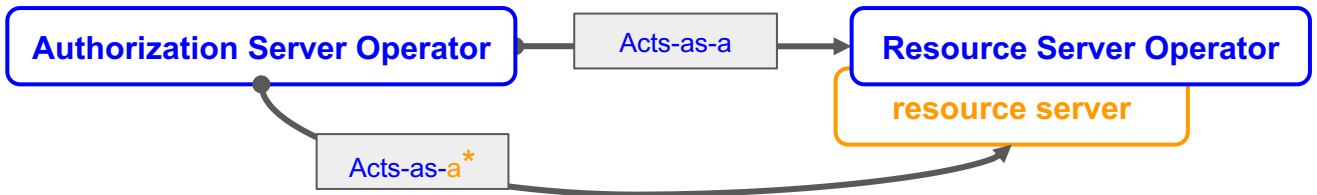
Legal relationships: One-party/multi-role scenario patterns

In some cases...

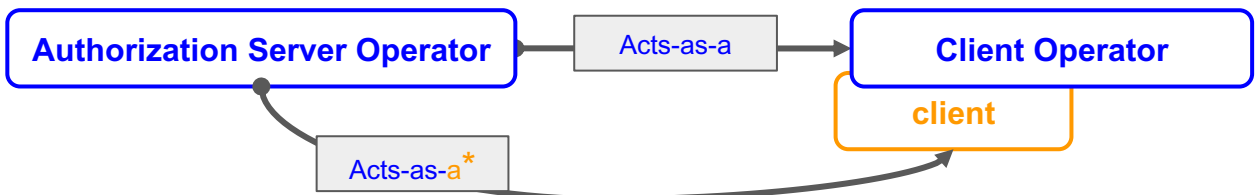


Legal relationships: More scenario patterns

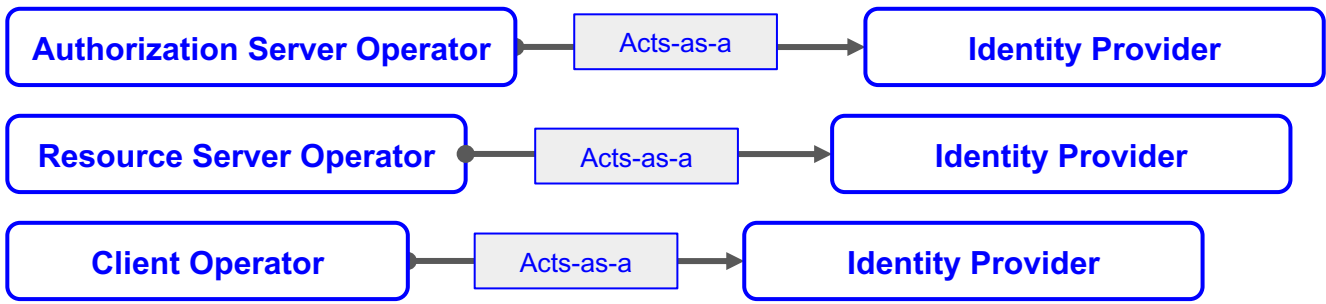
In some cases...



* ...and ASO *runs all available resource servers*. This relatively tighter ecosystem is consistent with how most OAuth deployments are run; it may still be interested in exposing the UMA Federated Authorization (protection API) interface for auditability reasons.



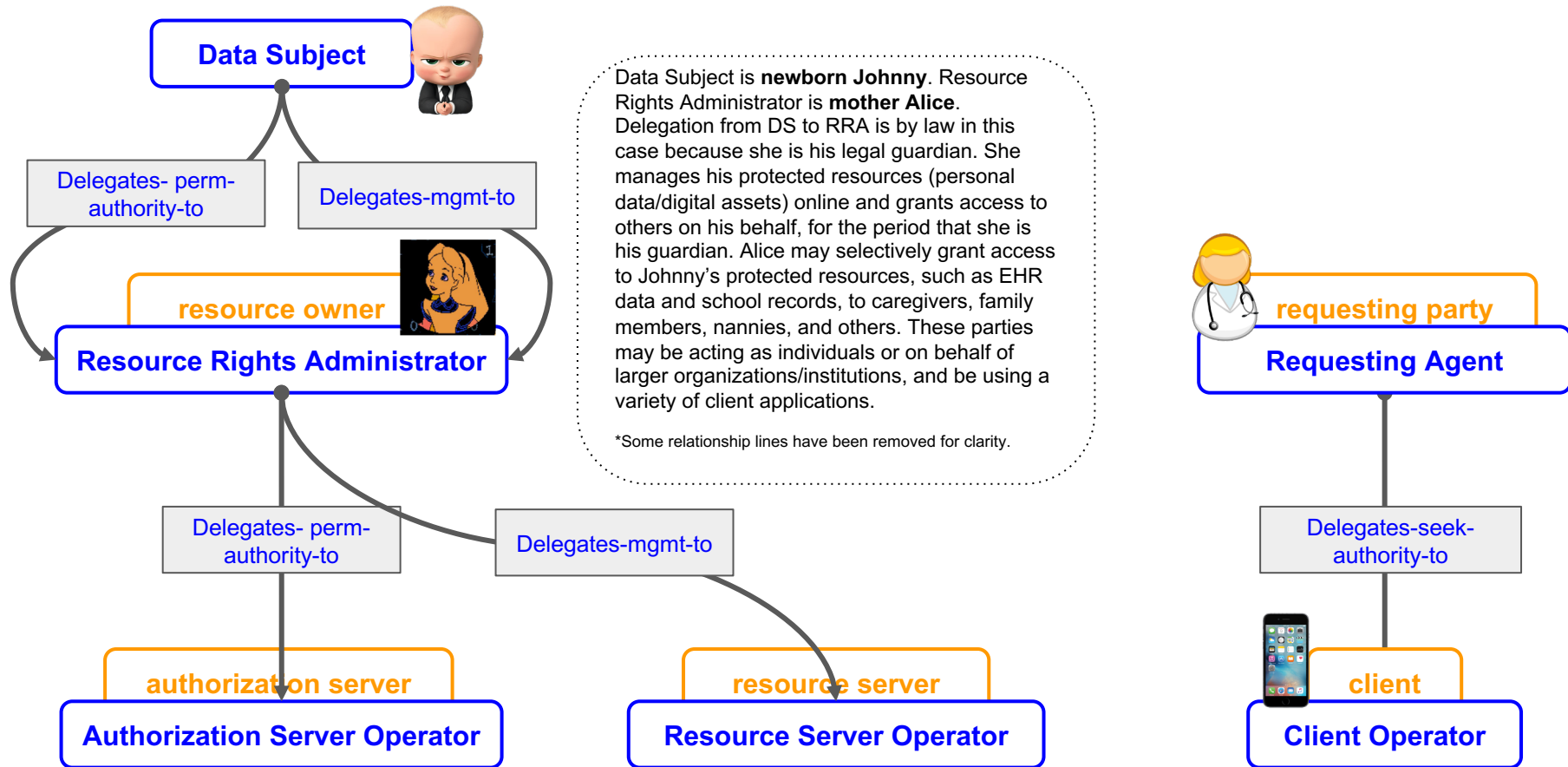
* ...and ASO *runs all available clients*. This tighter ecosystem (possibly in combination with the above) may still be interested in having the authorization server expose the various UMA interfaces for auditability reasons.



There are a variety of deployment options possible for sourcing resource owner identity (and requesting party claims). A business layer such as a trust framework can take into account identity assurance, authentication, and claims requirements. ("Identity Provider" is not an UMA-related party role and UMA is agnostic as to identity, identification, and authentication.)

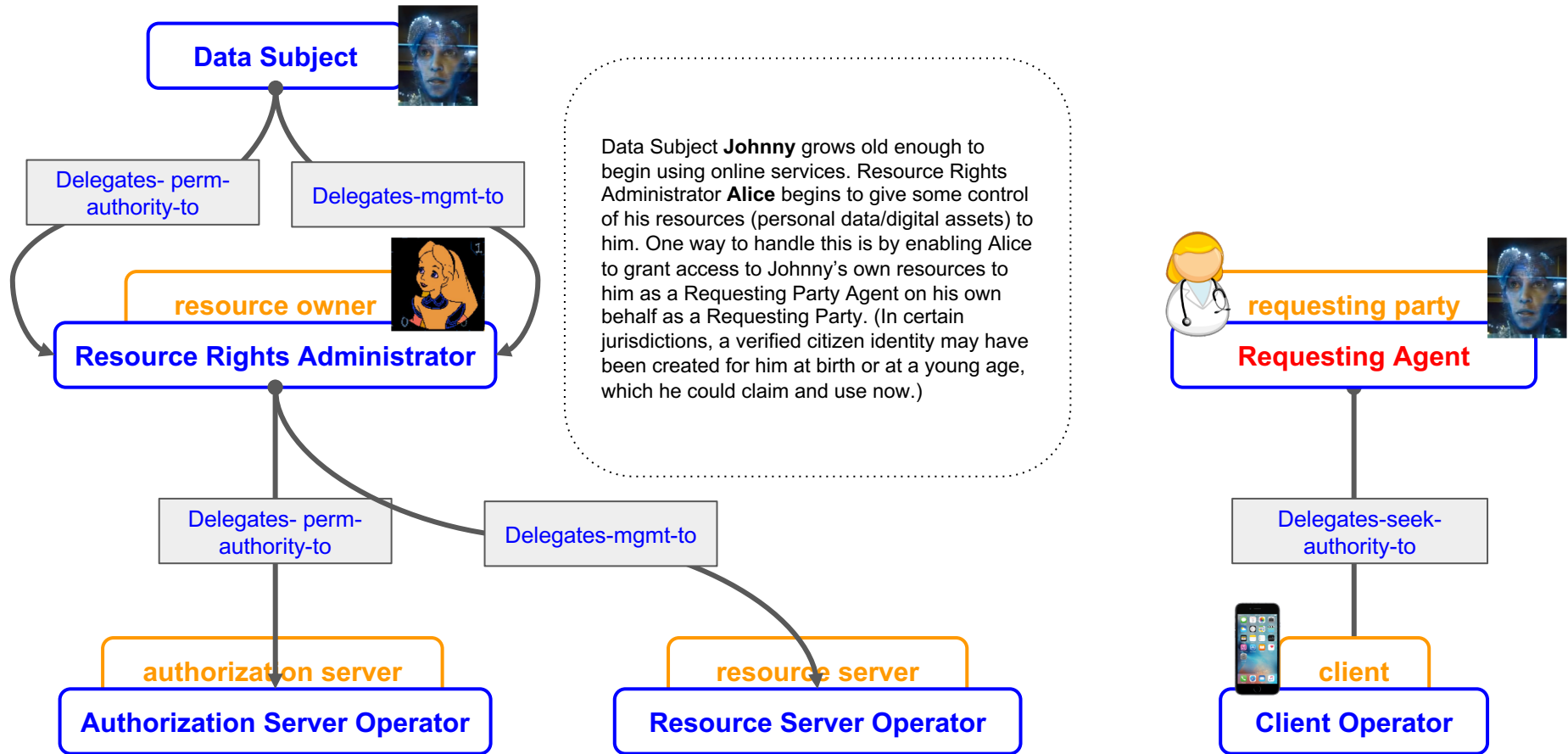
Scenario: Cradle-to-grave

1. Data Subject is too young to use digital assets



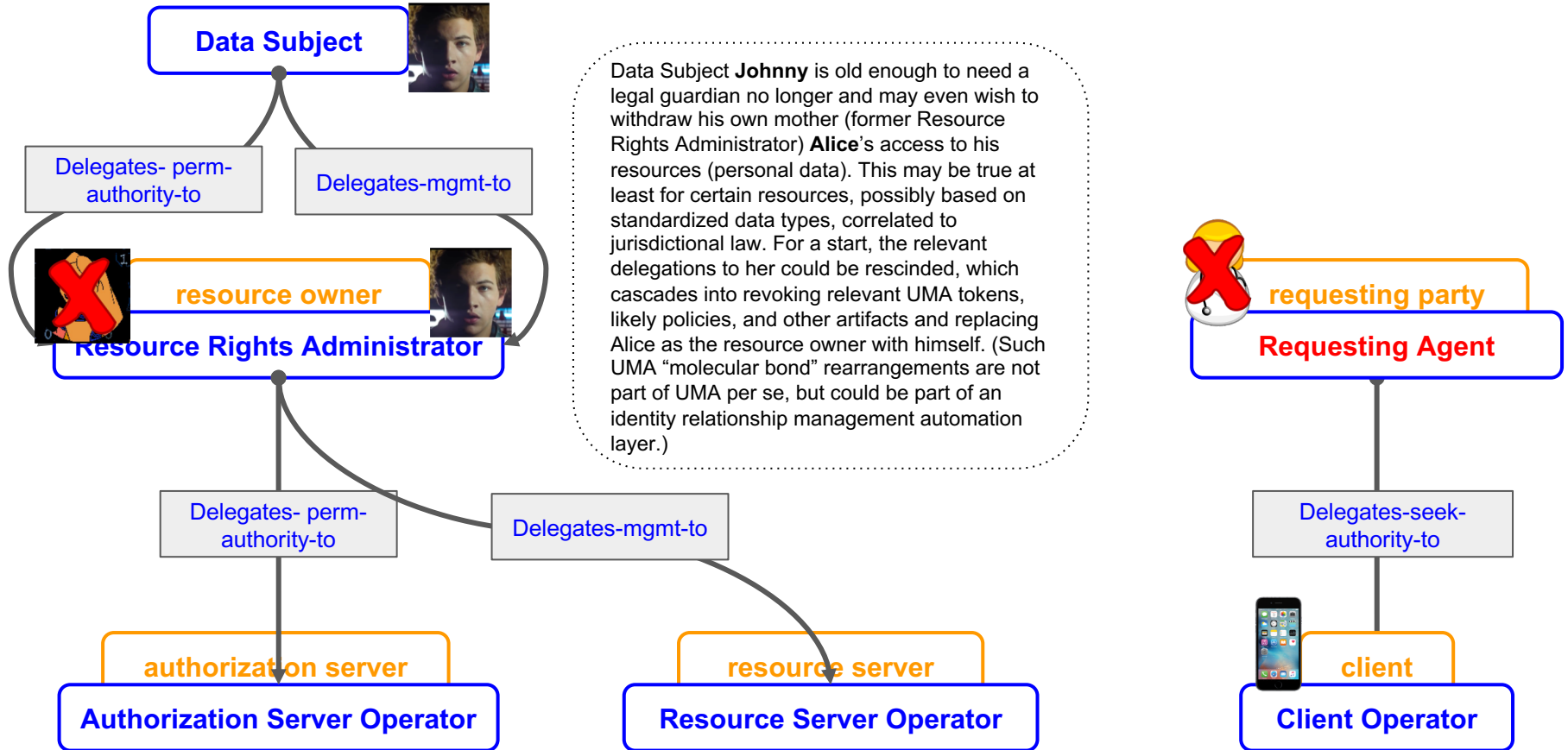
Scenario: Cradle-to-grave

2. Data Subject is old enough to use assets but too young to consent to their use



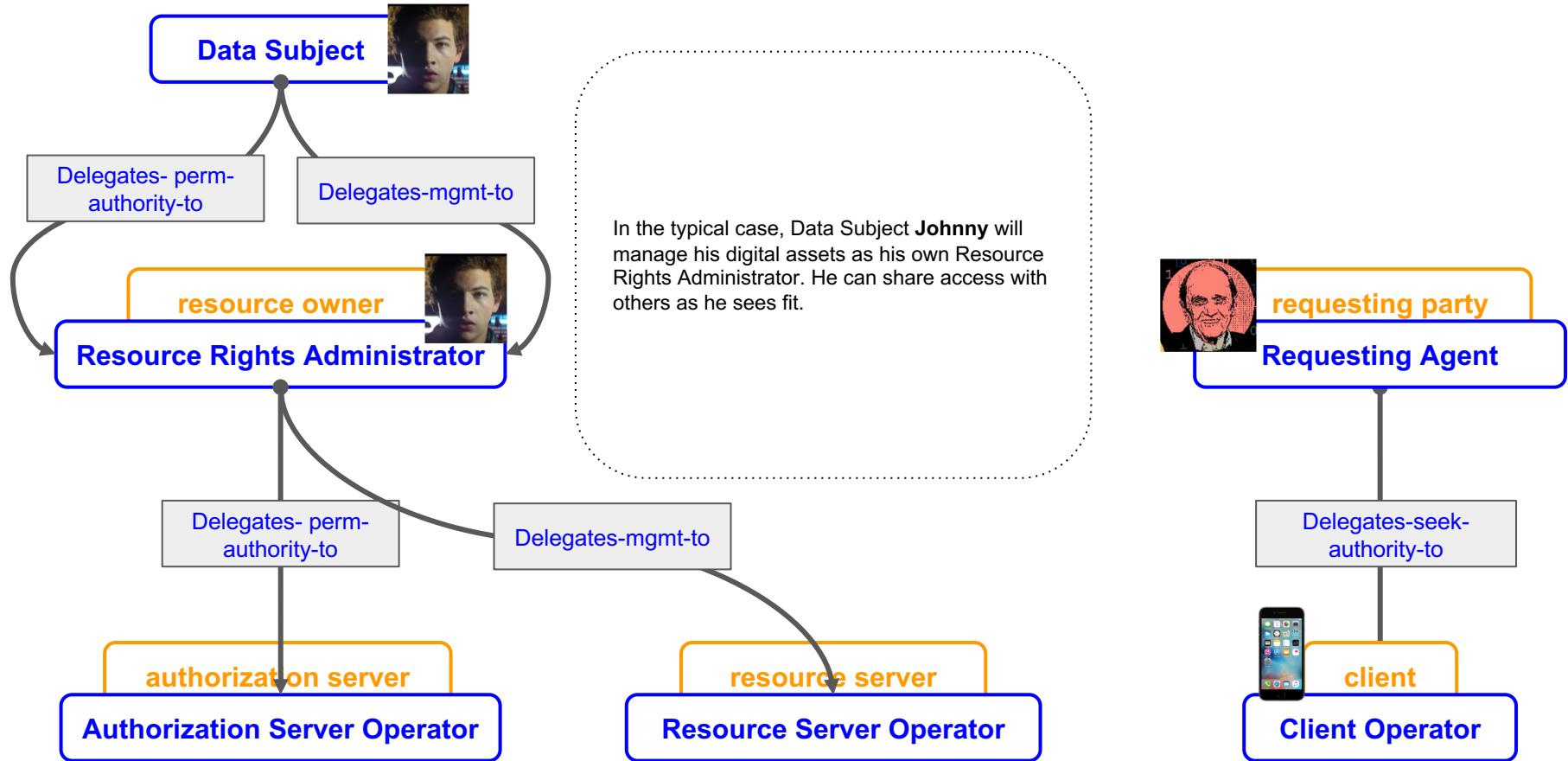
Scenario: Cradle-to-grave

3. Data Subject is old enough to consent to their use and manages digital assets themselves



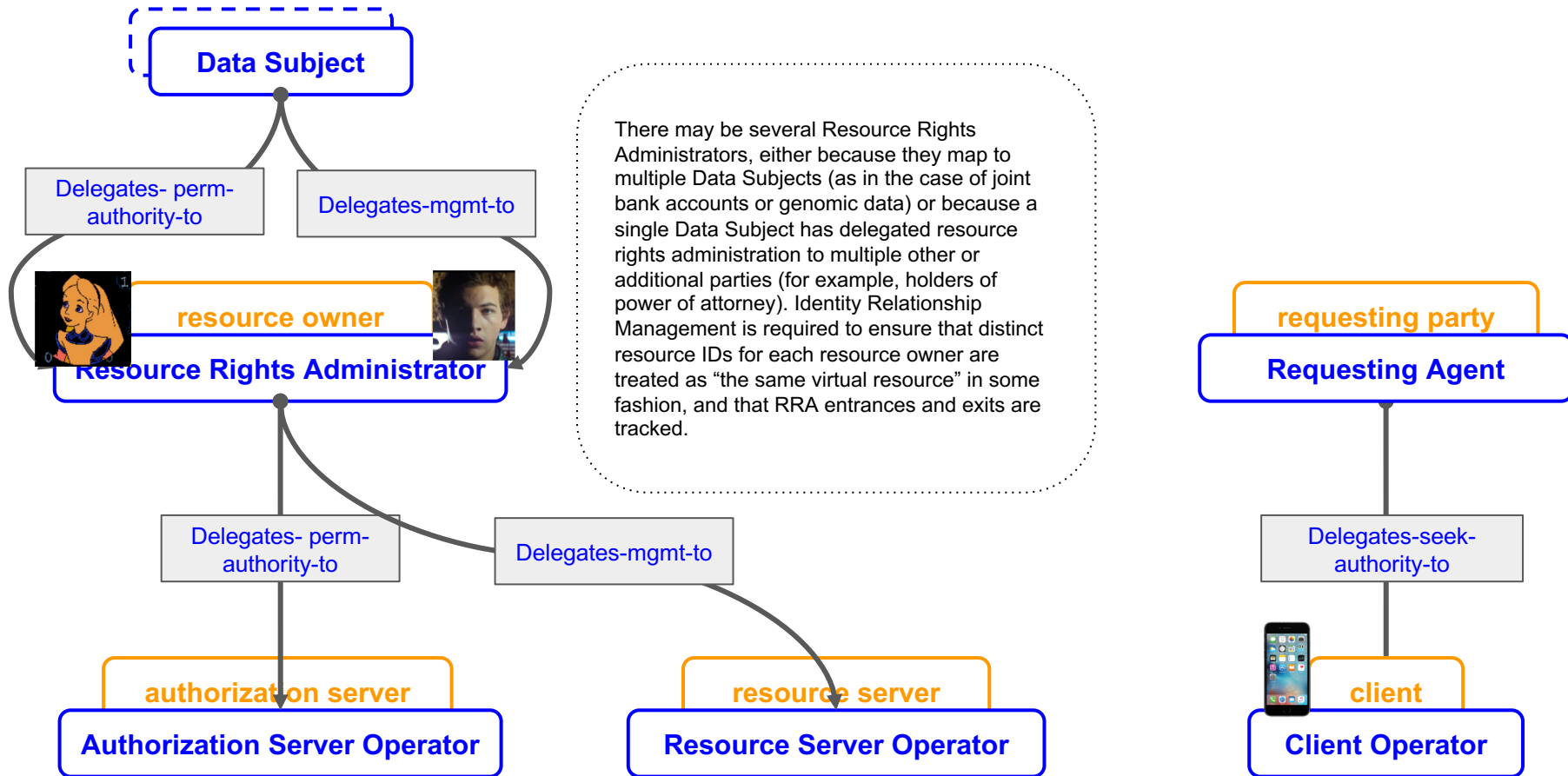
Scenario: Cradle-to-grave

3a. Steady state: Data Subject manages their own digital assets



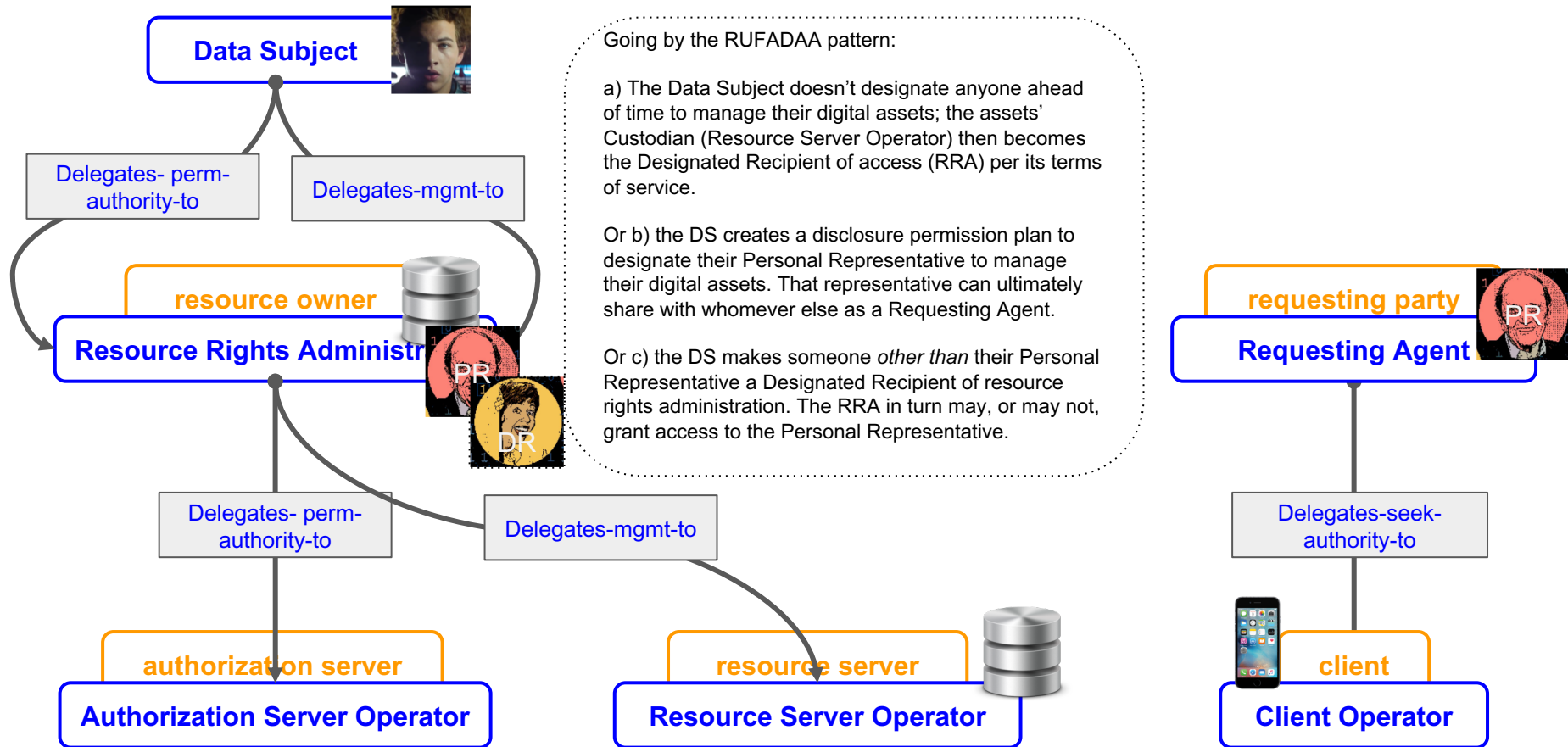
Scenario: Cradle-to-grave

4. There are multiple administrators of resource rights



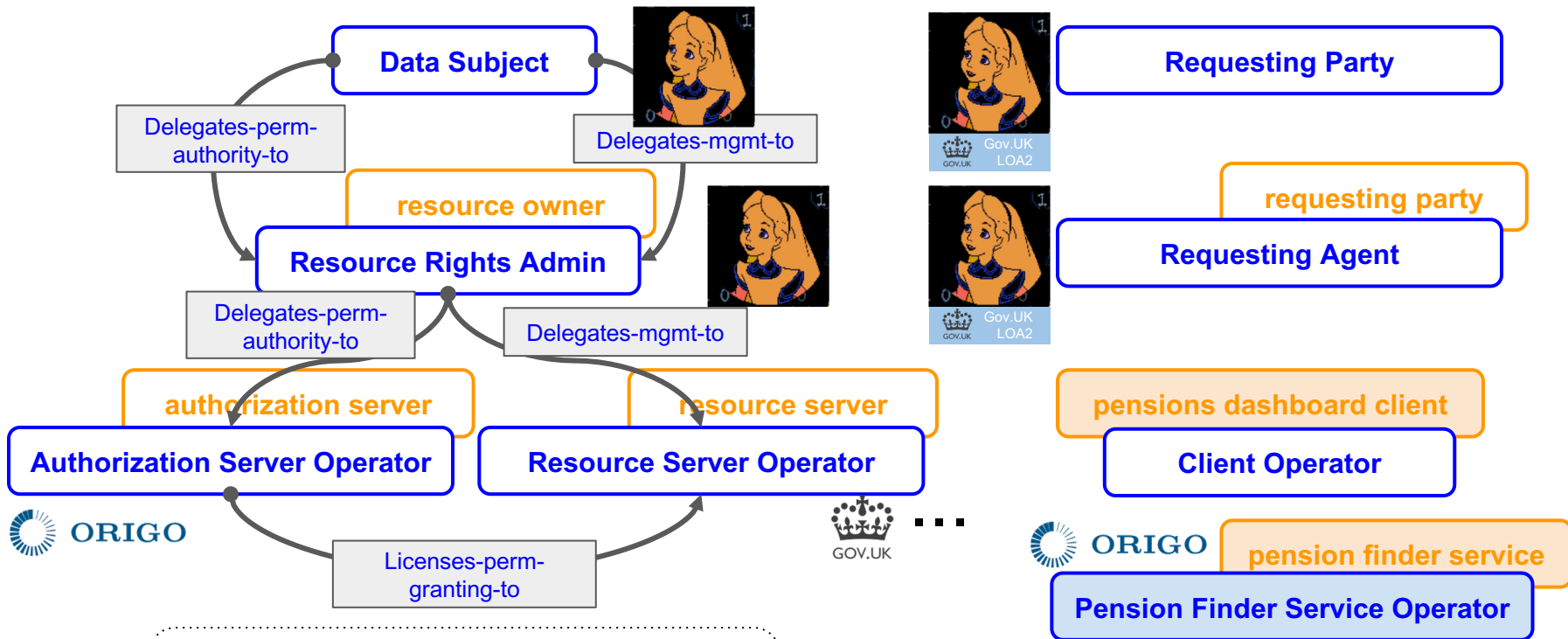
Scenario: Cradle-to-grave

5. Data Subject becomes mentally incapacitated or dies



Scenario: UK Pensions Dashboard

Step 1

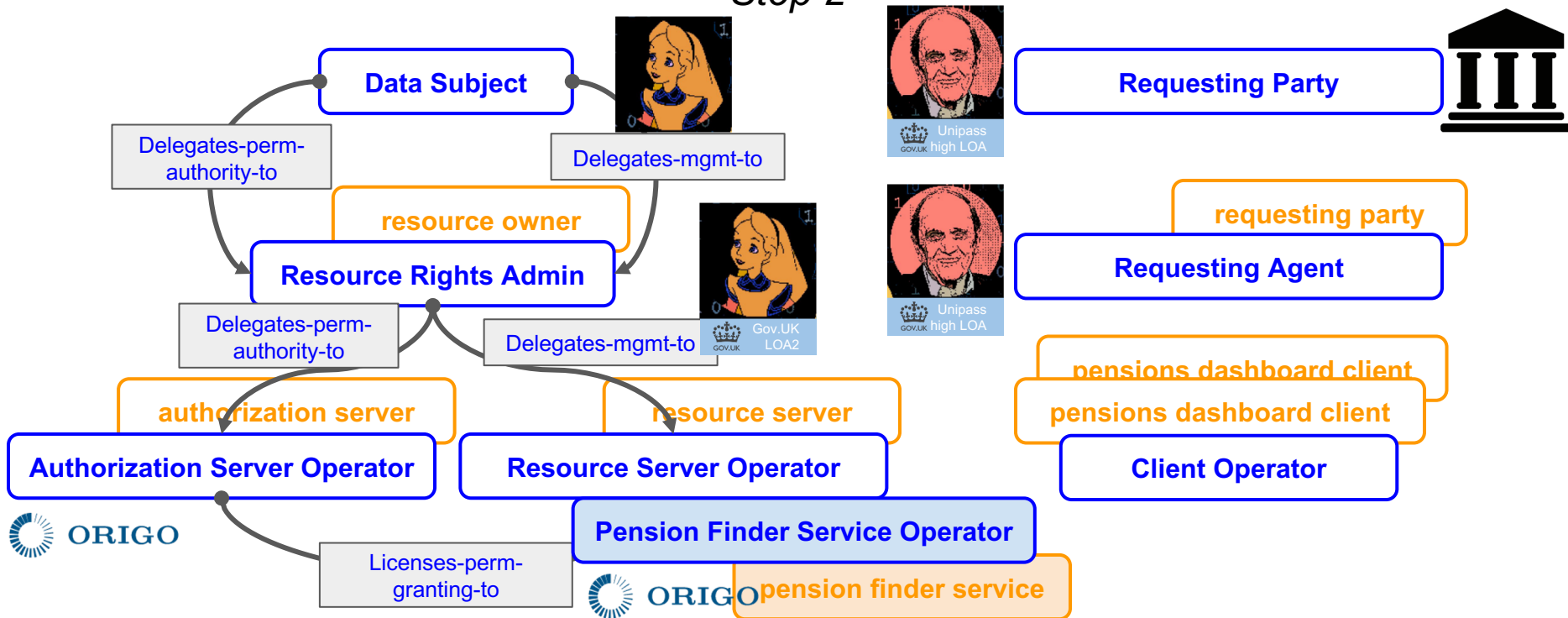


The Pensions Dashboard project is a government fintech initiative for the UK consumer. The Origo solution is securely identifying the consumer before orchestrating a search of pensions created in previously held jobs across the industry. "Wee Alice" (acting as her own DSA) first grants pension access to an LOA version of herself, "Big Alice". The government runs the AS and the single RS hosting state pension accounts; private state pension accounts are run separately. **Is the AS the low- and high-LOA IdP?**

(UMA delegation/licensing details on this side elided.)
The client application is a special one: a Pensions Dashboard that can aggregate a view of all found pensions. A special Pension Finder Service (not part of UMA) performs the aggregation process.

Scenario: UK Pensions Dashboard

Step 2



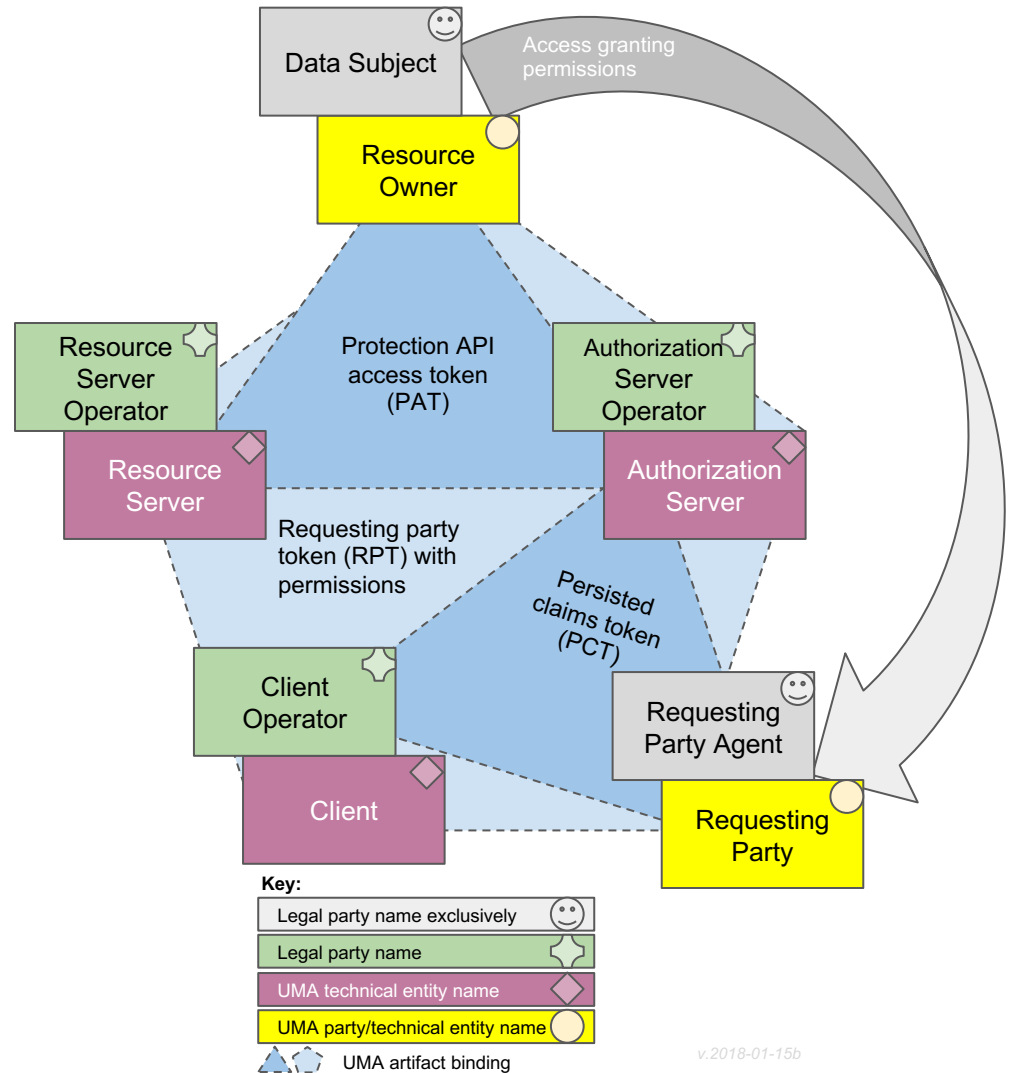
Alice, now in her shared-with role as "Big Alice", can now selectively share pension account information to financial advisors from a resource server run by the government that was sourced from the Pension Finder Service.
Guessing about the relationships between the services.

Through the Unipass IdP run by Origo for financial advisors, Bob provides high-LOA claims to get access. He may work for himself or a larger firm. **Guessing about varying RqP/RqPA relationships.**

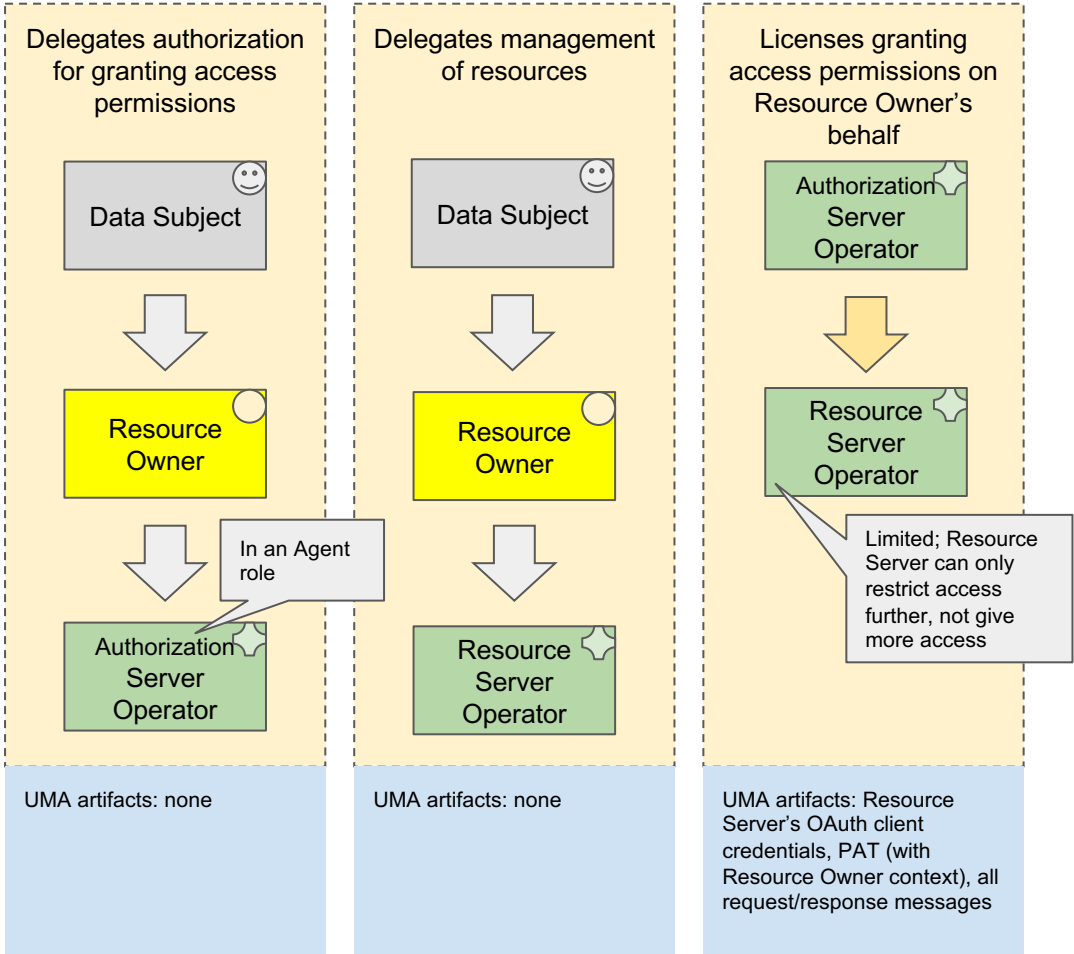
Diagrams used in report

(now a bit historical)

Legal roles and artifact interactions



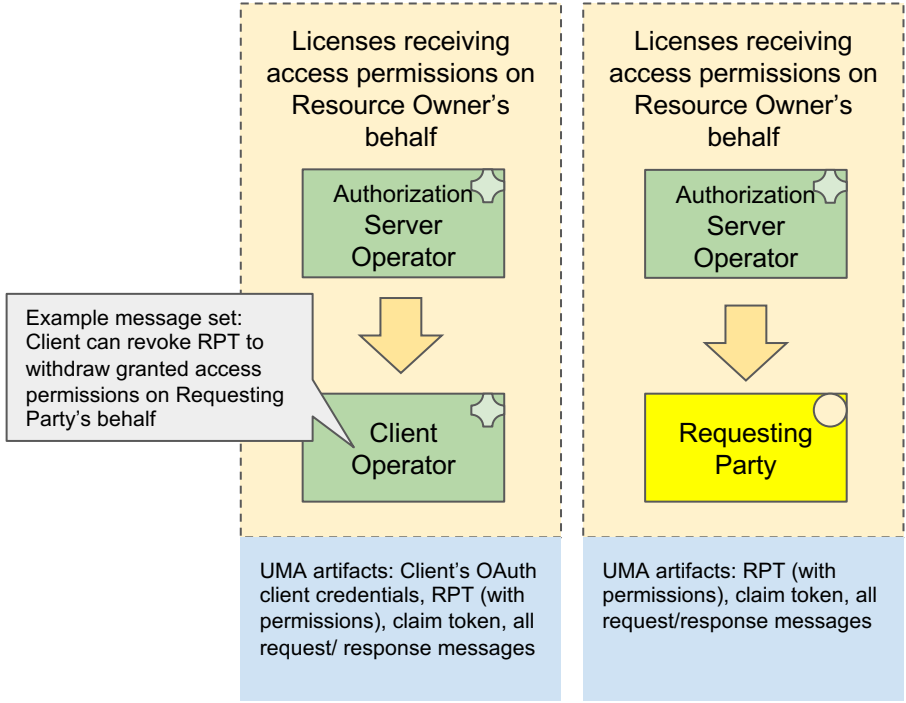
Delegation and licensing: RO-centered



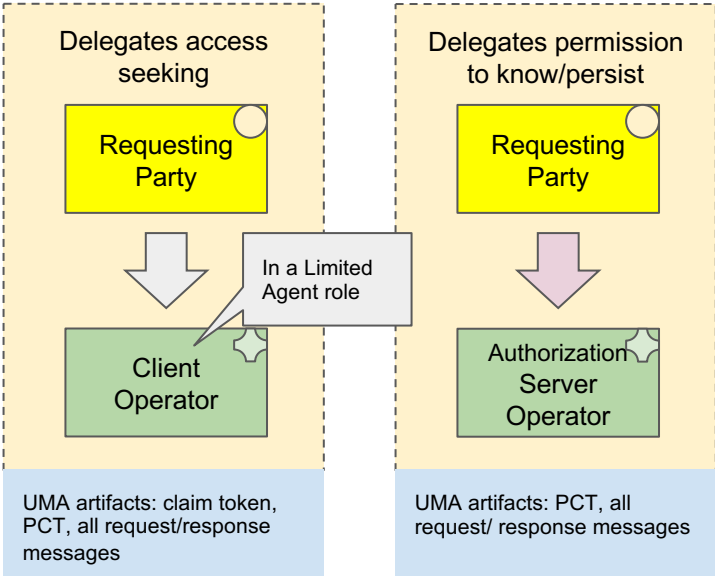
Key:

- ↓ Legal devices only
- ↓ Bound to UMA artifacts

Delegation and licensing: receiving permissions






Delegation and licensing: RqP-centered












Key:
↓ Partially bound to UMA artifacts

Earlier group musings







End-to-end licensing relationship

Requesting Party Resource Owner		Individual (Bob)	Legal Person (VendorCo)	licensee of resource permissions
				
Individual (Alice)	 licensor of resource permissions	Should these be switched? <-> Left is Bob to Alice JW You are correct! - Eve Individual-to-Vendor	Individual-to-Individual	Sharing Scenario











Sub-licensing intermediaries

		Individual (self - Alice)	Individual (other - Bob)	Legal Person (VendorCo)	
Requesting Party Options	Resource Owner				licensee
		TOS	TOS	TOS	
		 Client Operator	 Client Operator	 Client Operator	
Individual		 Resource Server Operator			sub-licensor
	TOS	 Authorization Server Operator			
		Individual-to-Self Sharing	Individual-to-Individual Sharing	Individual-to-Vendor Sharing	Sharing Scenario

End-to-end licensing relationship (new candidate 2)

		Requesting Party			
		Individual (Self)	Individual (other)	Legal Person	
	Resource Owner (Individual)	Individual-to-Self Sharing	Individual-to-Individual Sharing	Individual-to-Vendor Sharing	Scenarios
Client Operator					Sub-Licensors
Resource Server Operator					
Authorization Server Operator					

End-to-end licensing relationship sharing scenarios

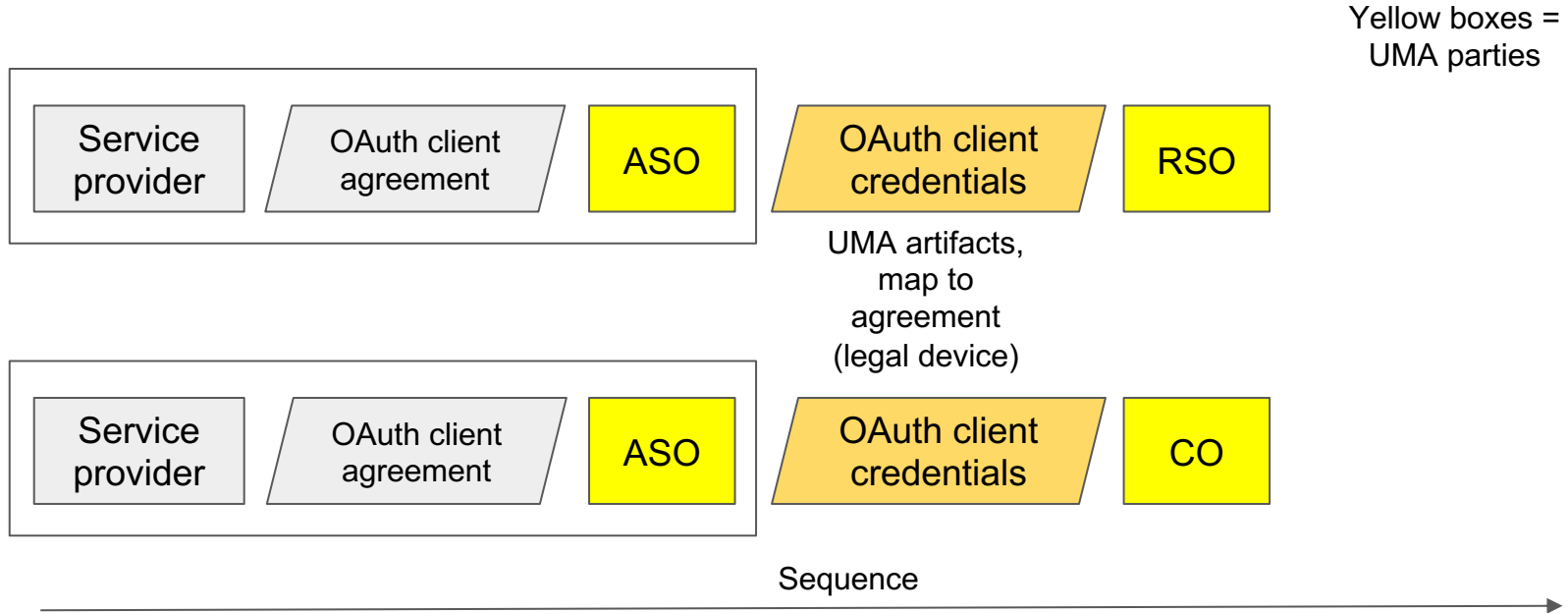
		Requesting Party				
		Individual (Self)	Individual (other)	Legal Person		
		Resource Owner (Individual)	Individual-to-Self Sharing	Individual-to-Individual Sharing	Individual-to-Vendor Sharing	Scenarios
Client Operator						Sub-Licensor
Resource Server Operator						
Authorization Server Operator						

Not sure if this can be incorporated visually, but the arrow of autonomy might be nice. That is to say, who WRITES the TOS or LIC

If written by RO or rep, autonomy favouring. If by other entity, less so.

JW

How RSO and CO become known to ASO

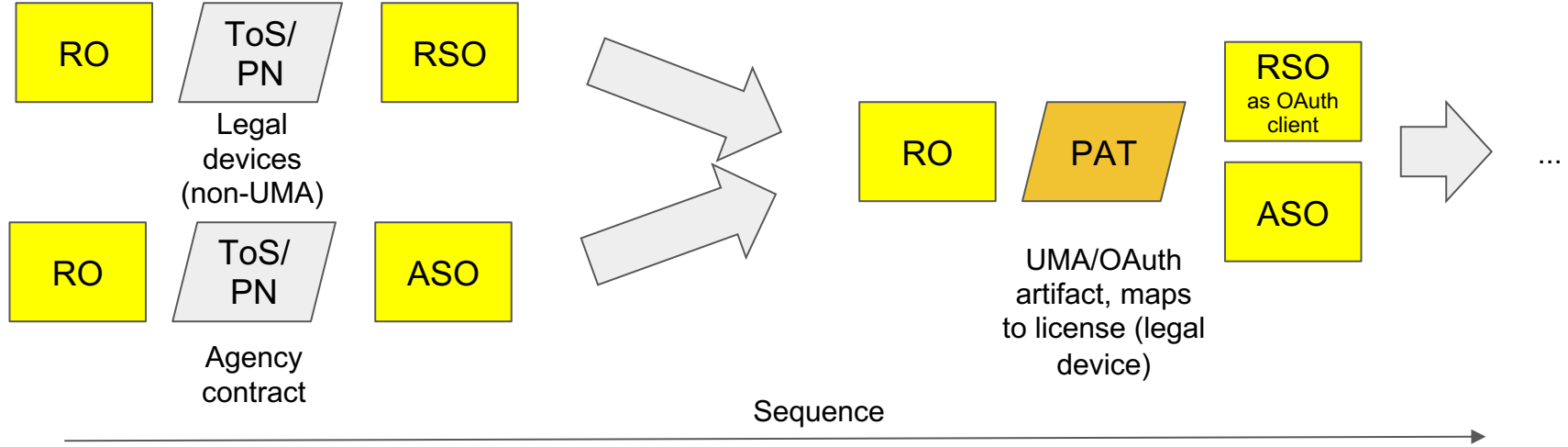


- Clause text would be supplied for both ToS/PN (non-UMA) and PAT artifacts
- This diagram does not include the RqP-side provisions
- Arrows imply ability for clause text to have the indicated order dependencies

Merging RO-RSO, RO-ASO, and RO-RSO-ASO relationship train tracks

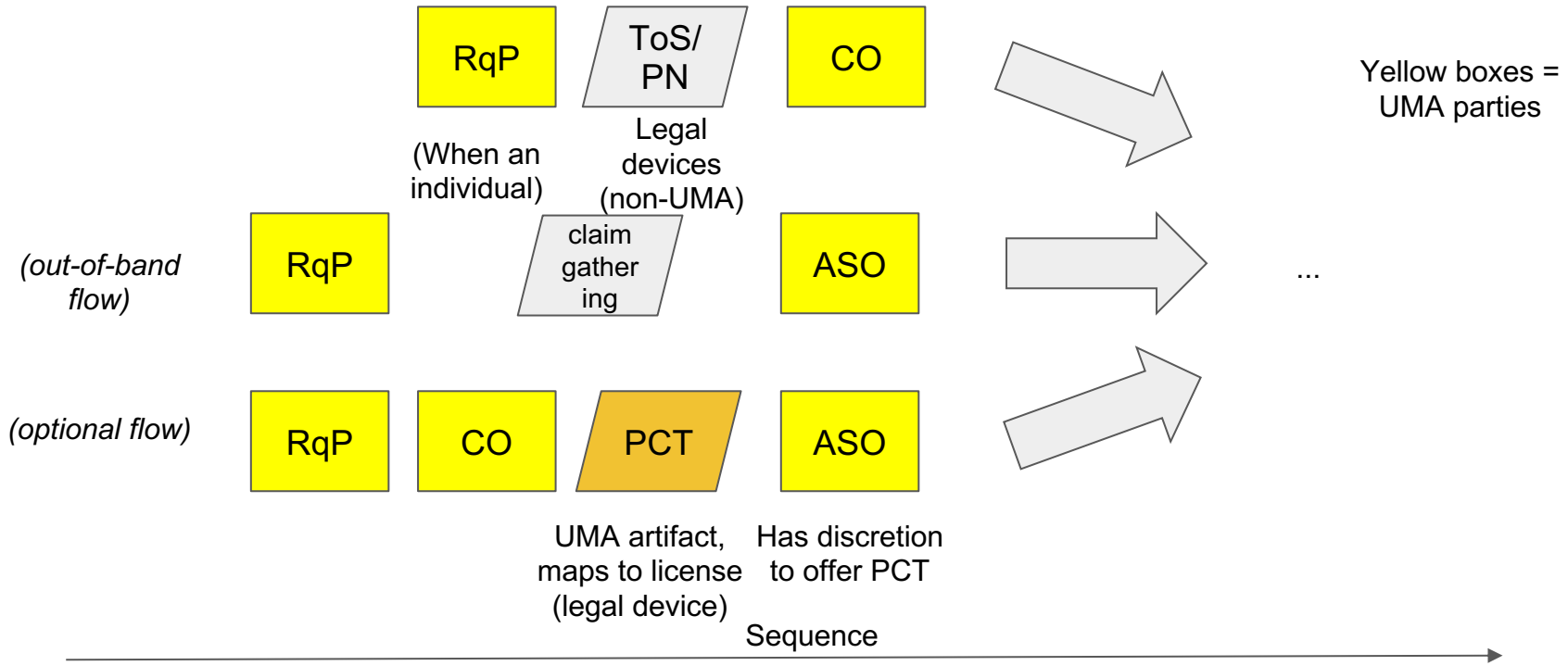
(Always an individual in our scope)

Yellow boxes = UMA parties



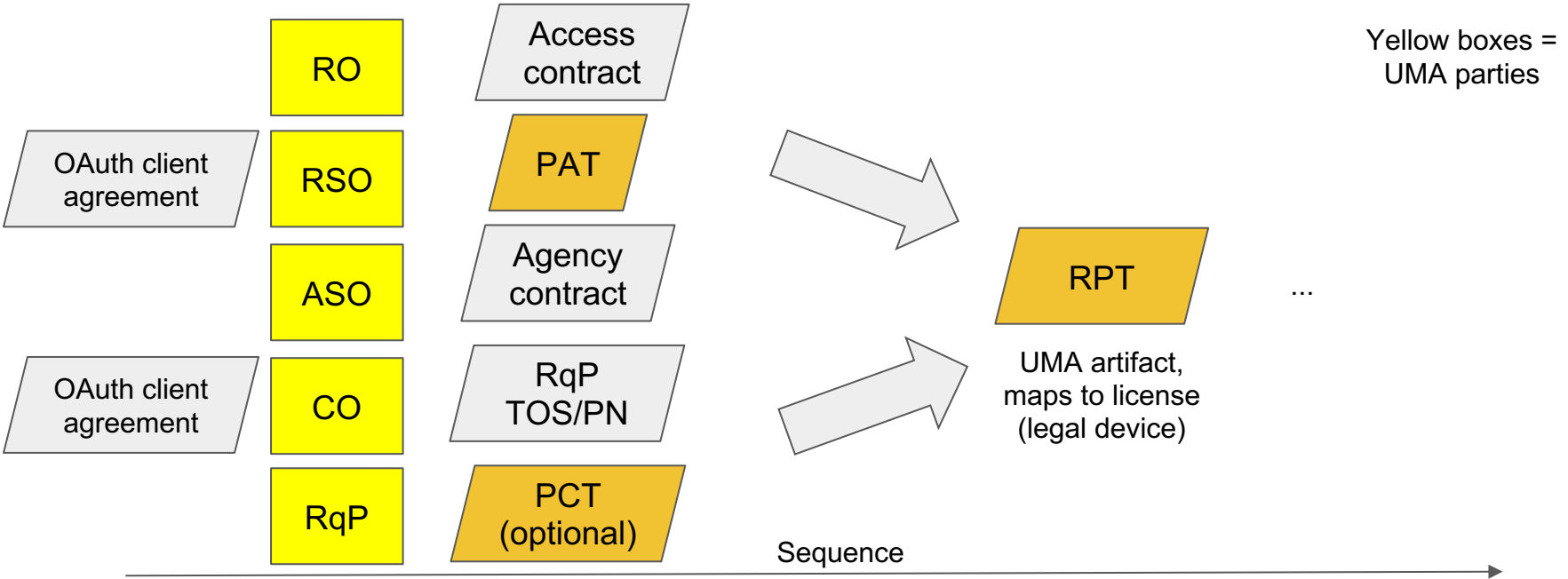
- Clause text would be supplied for both ToS/PN (non-UMA) and PAT artifacts
- This diagram does not include the RqP-side provisions
- Arrows imply ability for clause text to have the indicated order dependencies

Merging RqP-CO, RqP-ASO, and RqP-CO-ASO relationship train tracks



- Clause text would be supplied for ToS/PN (non-UMA) artifacts????? Not sure right now
- This diagram does not include the RO-side provisions
- Arrows imply ability for clause text to have the indicated order dependencies

RO-RSO-ASO-CO-RqP relationship



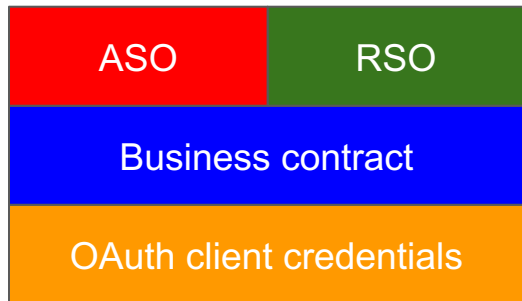
- Arrows imply ability for clause text to have the indicated order dependencies

(Fill in withdrawal/undoing flows)

Example of relationship, legal device, and technical artifact

Legend:

- Red: Pairwise relationship role with greater power
- Green: Pairwise relationship role with lesser power
- Blue: Legal device used between them
- Orange: Technical artifact on the UMA wire



The ASO and the RSO have a business contract wherein the ASO, as sub-licensor of resource permissions on behalf of the RO, sub-licenses to the RSO and enables the RSO to sub-license to COs and RqPs by virtue of giving access/giving content.

RqP vs RqPA relationship

RqPA

RqP

Sharing Scenario B: RqPA was shared with directly by the RO; they are human (Individual). They work for an organization (Legal Person) with which they have an employment agreement (or similar) that is outside the scope of any UMA technical artifacts. Others in the organization might get access by non-UMA methods in downstream fashion, as must be governed by the UMA-enabled license.

RqP

(what to call?
anything?)

Sharing Scenario B: RqP was shared with directly by the RO; they are an organization (Legal Person). They have humans (Individuals) working for them, with an employment agreement (or similar) that is outside the scope of any UMA technical artifacts, who gets access through non-UMA methods in downstream fashion as governed by the UMA-enabled license.

Technology/legal stack relationships

Consent Receipts?
HL7 Consents?
id-events?
PSD2 Consents?

UMA legal framework	Framework extension?
UMA protocol	Some consent tech