UMA2 Legal role definitions

Some visualizations presented at IIW 26

(See the companion draft report *A Proposed Licensing Model for User-Managed Access*, available at: https://kantarainitiative.org/reports-recommendations/)
Attempt at formal model
Legal relationships: Conventions and terminology

Conventions:
- **Legal** (upper capital, blue)  
  **technical** (lowercase, orange)  
  **issue/question** (red)

Relationship types:
- **Is-a** *(is a kind/species of; for more detail, see the Business Model definitions appendix)*
- **Acts-as-a** *(maps a party defined in the Business Model to a technical entity defined in the specs)*
- Delegates authority for granting and managing access permissions to: **Delegates-perm-authority-to**
- Delegates resource management to: **Delegates-mgmt-to**
- 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**
- **Acts-as-a** *(for business scenarios in cases where two roles are served by a single party)*
Legal relationships: Persons

Establishes basic party roles

- Individual
- Legal Person
- Data Subject
- Data Subject Agent
- Authorization Server Operator
- Resource Server Operator
- Client Operator
- Requesting Party
- Requesting Party Agent

Legal Person: A corporation, business trust, estate, trust, partnership, limited liability company, association, joint venture, governmental subdivision, instrumentality, or agency, public corporation, or any other legal or commercial entity.

Data Subject: The Person to whom a Protected Resource relates.

Client Operator: A Person responsible for running and operating a software application (the “Client”) used by a Requesting Party or Requesting Party Agent to access and use a Protected Resource.

Requesting Party: A Person with legal capacity and authority, either as an Individual or Legal Person, to request and secure access to a Protected Resource either directly with a Resource Server Operator or by means of a Client Operator.

Requesting Party Agent: A Person seeking access to a Protected Resource on behalf of a Requesting Party and by means of a Client software application.
Legal relationships: Delegation and licensing
Establishes how party roles relate to each other in a business sense

1. Delegates-perm-authority-to
2. Delegates-perm-authority-to
3. Delegates-mgmt-to
4. Delegates-mgmt-to
5. Licenses-perm-granting-to
6. Licenses-perm-getting-to
7. Licenses-perm-getting-to
8. Delegates-seek-authority-to
9. Delegates-seek-authority-to
10. Permits-knowing-claims
Legal relationships: Legal-to-technical role bridges

*Establishes how party roles can actually take part in UMA flows*

- **Data Subject Agent**
  - Acts-as-a
  - resource owner
- **Authorization Server Operator**
  - Acts-as-a
  - authorization server
- **Resource Server Operator**
  - Acts-as-a
  - resource server
- **Client Operator**
  - Acts-as-a
  - client
- **Requesting Party Agent**
  - Acts-as-a
  - requesting party

---

**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.
<table>
<thead>
<tr>
<th>#</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Relationship</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Possible OAuth/UMA artifacts</th>
<th>Typical legal devices</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Data Subject</td>
<td>(None)</td>
<td>Delegates-perm-authority-to</td>
<td>Data Subject Agent</td>
<td>resource owner</td>
<td>(None)</td>
<td>Law or private contract</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Data Subject Agent</td>
<td>resource owner</td>
<td>Delegates-perm-authority-to</td>
<td>Authorization Server Operator</td>
<td>authorization server</td>
<td>(None)</td>
<td>T/Cs, privacy notice (when DSA is an Individual)</td>
<td>If DSA==ASO, then possibly EULA or nothing (T/Cs: CRs?)</td>
</tr>
<tr>
<td>3</td>
<td>Data Subject</td>
<td>(None)</td>
<td>Delegates-mgmt-to</td>
<td>Data Subject Agent</td>
<td>resource owner</td>
<td>(None)</td>
<td>Law or private contract</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Data Subject Agent</td>
<td>resource owner</td>
<td>Delegates-mgmt-to</td>
<td>Resource Server Operator</td>
<td>resource server</td>
<td>(None)</td>
<td>T/Cs, privacy notice (when DSA is an Individual)</td>
<td>If DSA==RSO, then possibly EULA or nothing (T/Cs: CRs?)</td>
</tr>
<tr>
<td>5</td>
<td>Authorization Server Operator</td>
<td>authorization server</td>
<td>Licenses-perm-granting-to</td>
<td>Resource Server Operator</td>
<td>resource server</td>
<td>RS OAuth client credentials; PAT with RO context; all AS/RS request/response messages</td>
<td>OAuth client agreement</td>
<td>Agreement is outside/before RO context -- licensing needs to be set up/ prepared there</td>
</tr>
</tbody>
</table>

Legal relationships: Devices and artifacts (1 of 3)

Maps party roles to auditable and machine-readable artifacts
## Legal relationships: Devices and artifacts (2 of 3)

**Maps party roles to auditable and machine-readable artifacts**

<table>
<thead>
<tr>
<th>#</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Relationship</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Possible OAuth/UMA artifacts</th>
<th>Typical legal devices</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Authorization Server</td>
<td>authorization server</td>
<td>Licenses-perm-getting-to</td>
<td>Client Operator</td>
<td>client</td>
<td>Client OAuth client credentials; RPT with permissions; claim token; all AS/client request/response messages</td>
<td>OAuth client agreement</td>
<td>Agreement is outside/before RqP context – licensing needs to be set up/ prepared there; important but non-UMA artifacts include policies</td>
</tr>
<tr>
<td>7</td>
<td>Authorization Server</td>
<td>authorization server</td>
<td>Licenses-perm-getting-to</td>
<td>Requesting Party Agent</td>
<td>requesting party</td>
<td>PCT if used, all AS/RqP request/response messages</td>
<td>On DS/DSA's behalf, carried through technical artifacts</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Requesting Party</td>
<td>(None)</td>
<td>Delegates-seek-authority-to</td>
<td>Requesting Party Agent</td>
<td>requesting party</td>
<td>(None)</td>
<td>Law or private contract</td>
<td></td>
</tr>
</tbody>
</table>
### Legal relationships: Devices and artifacts (3 of 3)

*Maps party roles to auditable and machine-readable artifacts*

<table>
<thead>
<tr>
<th>#</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Relationship</th>
<th>Party role</th>
<th>Corresponding entity role</th>
<th>Possible OAuth/UMA artifacts</th>
<th>Typical legal devices</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Requesting Party Agent</td>
<td>requesting party</td>
<td>Delegates-seek-authority-to</td>
<td>Client Operator</td>
<td>client</td>
<td>Claim token if used, PCT if used, all RqP/client/AS request/response messages <em>Does the AS belong in this list?</em></td>
<td>T/Cs, privacy notice (when RqPA is an Individual)</td>
<td>If DSA==CO, then possibly EULA or nothing (T/Cs: CRs?)</td>
</tr>
<tr>
<td>10</td>
<td>Requesting Party Agent</td>
<td>requesting party</td>
<td>Permits-knowing-claims</td>
<td>Authorization Server Operator</td>
<td>authorization server</td>
<td>PCT if used, all RqP/AS request/response messages</td>
<td>Possibly T/Cs, privacy notice</td>
<td>This is the DS/DSA’s ASO (the RO’s AS), not (necessarily also) the RqP’s AS depending on topology (T/Cs: CRs?)</td>
</tr>
</tbody>
</table>
Legal relationships: One-party/multi-role scenario patterns

In some cases...

- Alice is controlling access to her own protected resources, vs. newborn/incompetent/etc. Johnny's.
- Alice has built/is running her own "personal authorization server". See HIE of One.
- A variant where Alice is running a PAS for Johnny.
- Alice has built/is running a "personal data store" for herself.
- A variant where Alice is running a PDS for Johnny.
- Bob is seeking access on behalf of himself, instead of doing it as "work for hire" on behalf of an employer.
- The ultimate party seeking access has built/is running their/its own client application (could be an individual or legal person).
- A variant where this is true of the agent of the ultimate party seeking access. (Included here for completeness but may be too detailed?)
- The same Person seeking access is the one whose resources are being protected. This is a typical OAuth scenario. (There are more "Agent" variants.)
Legal relationships: More scenario patterns

**In some cases...**

* Acts-as-a Authorization Server Operator
  * Acts-as-a Resource Server Operator
    * resource server
    * acts-as-a

* Acts-as-a Authorization Server Operator
  * Acts-as-a Client Operator
    * client

* Acts-as-a Authorization Server Operator
  * Acts-as-a Identity Provider

* Acts-as-a Resource Server Operator
  * Acts-as-a Identity Provider

* Acts-as-a Client Operator
  * Identity Provider

* ...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: Parent-child resource management

Life stage 1

Data Subject

Delegates-perm-authority-to

Delegates-mgmt-to

Resource owner

Data Subject Agent

Delegates-perm-authority-to

Delegates-mgmt-to

authorization server

Authorization Server Operator

Delegates-perm-authority-to

Delegates-mgmt-to

resource server

Resource Server Operator

Licenses-perm-granting-to

Requesting Party

Delegates-mgmt-to

Requesting Party Agent

Delegates-perm-authority-to

Client

Client Operator

DS is newborn Johnny. DSA is mother Alice. Delegation from DS to DSA is by law in this case because she is his legal guardian. She manages his protected resources (personal data/digital assets) online and grants access to others on his behalf, for the period that she is his guardian.

(UMA delegation/licensing details on this side elided.) Alice may selectively grant access to Johnny’s protected resources, such as EHR data and school records, to caregivers, family members, nannies, and others. These parties may be acting as individuals or on behalf of larger organizations/institutions, and be using a variety of client applications.
Scenario: Parent-child resource management

Life stage 2

**Data Subject**
- Delegates-perm-authority-to
- Delegates-mgmt-to

**Authorization Server**
- Authorization Server Operator
- Licenses-perm-granting-to

**Resource Server**
- Resource Server Operator

**Requesting Party**
- Requesting Party
- Requesting Party Agent

**Client**
- Client Operator

**DS Johnny** grows old enough to begin using online services. **DSA Alice** begins to give some control of his resources (personal data/digital assets) to him. One way to handle this is by enabling Alice to grant access to Johnny's own resources to him as a Requesting Party Agent on his own behalf as a Requesting Party. (In certain jurisdictions, a verified citizen identity may have been created for him at birth, which he could claim and use now.)
Scenario: Parent-child resource management

*Life stage 3*

**Authorization Server Operator**

**Resource Server Operator**

**Client Operator**

**Requesting Party Agent**

**Data Subject Agent**

**Requesting Party**

**Data Subject**

**Resource owner**

**Delegates-perm-authority-to**

**Delegates-mgmt-to**

**Delegates-perm-authority-to**

**Authorization Server**

**Resource Server**

**Licenses-perm-granting-to**

DS Johnny is old enough to need a legal guardian no longer and may even wish to withdraw his own mother (former DSA) Alice’s access to his resources (personal data). This may be true at least for certain resources, possibly based on standardized data types, correlated to jurisdictional law. For a start, the relevant delegations to her could be rescinded, which cascades into revoking relevant UMA tokens, likely policies, and other artifacts and replacing Alice as the resource owner with himself. (Such UMA “molecular bond” rearrangements are not part of UMA per se, but could be part of an “identity relationship management” automation layer.)