

# Documento merging di policy

Gianluca Oldani

Work in progress

# Capitolo 1: Possibili casi di merging

## Casi base

### Asset differenti

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Set",
4   "uid": "http://example.com/policy:1010",
5   "permission": [{
6     "target": "http://example.com/asset:9898.movie",
7     "action": "play"
8   }]
9 }
```

Listing 1.1: La policy 1010 permette di riprodurre l'asset 9898.movie a chiunque

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Set",
4   "uid": "http://example.com/policy:1011",
5   "permission": [{
6     "target": "http://example.com/asset:1349.mp3",
7     "action": "play"
8   }]
9 }
```

Listing 1.2: La policy 1011 consente la riproduzione dell'asset 1349.mp3 a chiunque

Le due policy mostrate si riferiscono ad asset differenti, farne il merging porta semplicemente ad una policy avente entrambe le *permission* già enunciate.

## Rule incompatibili

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Set",
4   "uid": "http://example.com/policy:1010",
5   "permission": [{
6     "target": "http://example.com/asset:9898.movie",
7     "action": "play"
8   }]
9 }
```

Listing 1.3: La policy 1010 consente la riproduzione dell'asset 9898.movie a chiunque

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Set",
4   "uid": "http://example.com/policy:1011",
5   "prohibition": [{
6     "target": "http://example.com/asset:9898.movie",
7     "action": "distribute"
8   }]
9 }
```

Listing 1.4: La policy 1011 proibisce la distribuzione dell'asset 9898.movie a chiunque

Nonostante le due policy si riferiscano allo stesso asset, anche in questo caso, non è possibile combinare le due regole, poiché si riferiscono a domini diversi. Un eventuale merging delle due policy avrebbe 2 regole distinte uguali a quelle di partenza.

## Attori designati differenti

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "permission": [{
6     "target": "http://example.com/asset:1212",
7     "action": "use",
8     "assigner": "http://example.com/owner:181",
9     "assignee":
10      "http://example.com/party:person:billie"
11   }]
12 }
```

Listing 1.5: La policy 0001 permette un qualsiasi utilizzo dell'asset 1212 da parte del soggetto Billie

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0002",
5   "prohibition": [{
6     "target": "http://example.com/asset:1212",
7     "action": "play",
8     "assigner": "http://example.com/owner:181",
9     "assignee":
10      "http://example.com/party:person:alice"
11   }]
12 }
```

Listing 1.6: La policy 0002 proibisce la riproduzione dell'asset 1212 da parte del soggetto Alice

In questo caso, le regole espresse dalle 2 policy sarebbero in conflitto, poiché “use”, permesso dalla prima policy, comprende al suo interno anche “play”<sup>1</sup>, proibito dalla seconda. Siccome però le due regole riguardano due soggetti diversi, un eventuale merging delle policy avrebbe 2 regole distinte uguali a quelle di partenza.

<sup>1</sup>Per le dipendenze tra le azioni, si fa riferimento all'ODRL Core Vocabulary

### Conflitti senza strategia risolutiva indicata

Nel caso si presentino 2 policy in conflitto, se non viene indicato alcun processo di risoluzione dei conflitti, secondo lo standard ODRL entrambe le policy sono da considerarsi non valide. Attuando policy merging è possibile formulare una terza policy che racchiude entrambe le casistiche.

### Conflitti permesso-permesso

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "permission": [{
6     "target": "http://example.com/asset:1212",
7     "action": "play"
8   }]
9 }
```

Listing 1.7: La policy 0001 permette un qualsiasi utilizzo dell'asset 1212 a chiunque

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0002",
5   "permission": [{
6     "target": "http://example.com/asset:1212",
7     "action": "display"
8   }]
9 }
```

Listing 1.8: La policy 0002 permette la riproduzione dell'asset 1212 a chiunque

La seconda policy risulta in conflitto con la prima, poiché quest'ultima permette un numero maggiore di utilizzi. Seguendo la procedura standard, entrambe le 2 policy dovrebbero essere considerate invalidate. Nella realtà dei fatti, facendo policy merging, si otterrebbe solamente la policy più restrittiva, ovvero la seconda. La procedura di policy merging in questione sarebbe quindi:

1. invalidare le policy 0001 e 0002;

2. creare una terza policy 0003, uguale alla 0002 (la più restrittiva delle due), con indicato il procedimento di merging;
3. l'indicazione è opportuna per fare il processo inverso al merging qualora una delle 2 policy originarie venga ritirata.

### Conflitti divieto-divieto

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "prohibition": [{
6     "target": "http://example.com/asset:1212",
7     "action": "play"
8   }]
9 }
```

Listing 1.9: La policy 0001 proibisce un qualsiasi utilizzo dell'asset 1212 a chiunque

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0002",
5   "prohibition": [{
6     "target": "http://example.com/asset:1212",
7     "action": "display"
8   }]
9 }
```

Listing 1.10: La policy 0002 proibisce la riproduzione dell'asset 1212 a chiunque

Caso duale del precedente; in questo caso una delle due policy proibisce un numero maggiore di utilizzi rispetto all'altra. La procedura da seguire risulta essere:

1. invalidare le policy 0001 e 0002;
2. creare una terza policy 0003, uguale alla 0001 (la più ampia delle due), con indicato il procedimento di merging;
3. l'indicazione è opportuna per fare il processo inverso al merging qualora una delle 2 policy originarie venga ritirata.

## Conflitti permesso-divieto

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "permission": [{
6     "target": "http://example.com/asset:1212",
7     "assigner": "http://example.com/owner:181",
8     "assignee":
9     "http://example.com/party:person:alice",
10    "action": "transfer"
11  }]
12 }
```

Listing 1.11: La policy 0001 permette il trasferimento dell'asset 1212 al soggetto Alice

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0002",
5   "prohibition": [{
6     "target": "http://example.com/asset:1212",
7     "assigner": "http://example.com/owner:181",
8     "assignee":
9     "http://example.com/party:person:alice",
10    "action": "sell"
11  }]
12 }
```

Listing 1.12: La policy 0002 proibisce la vendita dell'asset 1212 al soggetto Alice

In questo caso, l'azione “transfer”, nell'ODRL Core Vocabulary, è inclusa in 2 sotto-azioni: “give” e “sell”; delle 2 sotto-azioni, solamente “sell” è esplicitamente proibita dalla policy 0002, di conseguenza la seguente policy permette solo azioni che soddisfano entrambe le policy precedenti:

```

1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0002",
5   "permission": [{
6     "target": "http://example.com/asset:1212",
7     "assigner": "http://example.com/owner:181",
8     "assignee":
9     "http://example.com/party:person:alice",
10    "action": "give"
11  }]
12 }

```

Listing 1.13: La policy 0003 consente al soggetto Alice di cedere l’asset 1212 senza richiedere un compenso e cancellando l’asset dal proprio insieme di dati

Sono necessarie alcune valutazioni su questa casistica:

- il numero di regole all’interno della policy ottenuta per merging non è necessariamente minore rispetto al numero di regole di partenza; tale numero, dipende dal numero di sotto-azioni esistenti e da come è formulato il divieto di partenza;
- è necessario un meccanismo per notificare che il le sotto-azioni presentate in realtà indicano “tutte le sotto-azioni possibili, escluse quelle esplicitamente vietate”;
- se “prohibition” e “permission” fossero invertite tra la policy 0001 e la 0002, la policy 0003 risulterebbe una policy non valida.

### Conflitti con strategia risolutiva indicata

Le strategie risolutive indicabili all’interno di ODRL utilizzando il Core Vocabulary sono le seguenti:

1. “perm”: tutte le “permission” hanno la precedenza sulle “prohibition”;
2. “prohibit”: tutte le “prohibition” hanno la precedenza sulle “permission”;
3. “invalid”: qualsiasi conflitto invalida la policy;

La casistica “invalid” è già stata trattata all’interno del paragrafo “Conflitti senza strategia risolutiva indicata”[1].



## Strategia concorde

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "conflict": {"@type": "perm"},
6   "permission": [{
7     "target": "http://example.com/asset:1212",
8     "action": "use",
9     "assigner": "http://example.com/owner:181"
10  }]
11 }
```

Listing 1.14: La policy 0001 consente qualsiasi utilizzo dell'asset 1212 a chiunque, dando precedenza ai permessi in caso di conflitto

```
1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "conflict": {"@type": "perm"},
6   "permission": [{
7     "target": "http://example.com/asset:1212",
8     "action": "play",
9     "assigner": "http://example.com/owner:182"
10  }],
11  "prohibition": [{
12    "target": "http://example.com/asset:1212",
13    "action": "display",
14    "assigner": "http://example.com/owner:181"
15  }]
16 }
```

Listing 1.15: La policy 0002 permette la riproduzione dell'asset 1212 a chiunque, mentre ne vieta la proiezione, dando precedenza ai permessi in caso di conflitto

In questo caso è possibile ricondursi alla procedura indicata nel sottoparagrafo “Conflitti permesso-permesso”[1] ignorando tutti i divieti espressi nelle policy. In seguito al merging delle policy si ottiene:

```

1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Policy",
4   "uid": "http://example.com/policy:0001",
5   "conflict": {"@type": "perm"},
6   "permission": [{
7     "target": "http://example.com/asset:1212",
8     "action": "play",
9     "assigner": "http://example.com/owner:182"
10  }]
11 }

```

Listing 1.16: La policy 0003 risulta essere uguale ai permessi della policy 0002, più restrittiva

Dando la precedenza ai permessi, rimane comunque da tener in conto che nella policy 0002 il permesso è più ristretto rispetto a quello espresso nella policy 0001.

Dualmente, nel caso di strategia di conflitto concorde di tipo “prohibit”, ci si riconduce al caso mostrato nel sottoparagrafo “Conflitti divieto-divieto”[1], tenendo conto solo dei divieti.

### Strategia discorde

Utilizzando il normale sistema di ereditarietà delle policy di ODRL, una policy contenente 2 strategie di risoluzione dei conflitti dovrebbe essere considerata non valida. Questa casistica si può però ridurre al caso mostrato nel sottoparagrafo “Conflitti permesso-divieto” [1]. In questo modo, entrambe le policy risultano rispettate. Un’aggiunta che può essere fatta a questa casistica, rispetto a quella già mostrata, sarebbe considerare solamente i divieti o i permessi della singola policy, in base alla strategia indicata; quest’ultima opzione porterebbe ad un ibrido tra le due soluzioni precedenti.

### Restrizioni su regole, asset, party

Possono essere poste delle restrizioni sulla validità di una regola o sulla composizione di una collezione di asset o gruppo di soggetti. Anche queste restrizioni possono essere composte similmente alle regole a sé stanti.

```

1 {
2   "@context": "http://www.w3.org/ns/odrl.jsonld",
3   "@type": "Offer",
4   "uid": "http://example.com/policy:6163",
5   "profile": "http://example.com/odrl:profile:10",
6   "permission": [{
7     "target": "http://example.com/document:1234",
8     "assigner": "http://example.com/org:616",
9     "action": "distribute",
10    "constraint": [{
11      "leftOperand": "dateTime",
12      "operator": "lt",
13      "rightOperand":
14        { "@value": "2018-01-01",
15          "@type": "xsd:date" }
16    }]
17  }]
18 }

```

Listing 1.17: Esempio di limite temporale per una regola, può essere composto similamente ai limiti sull'azione regolata

Problematiche relative alla gestione di questo tipo di merging, non più necessariamente relativo a conflitti, possono essere rilevate nella grande varietà che può assumere:

- limiti di tipo temporale;
- limiti di tipo spaziale;
- limiti di tipo quantitativo(numero di volte che un permesso può essere sfruttato);
- limiti relativi al tipo di dato trattato(dimensione dell'immagine, durata del video).

A causa di questa varietà, la gestione di questo tipo di merging può facilmente esplodere. Vi è anche da contare che un “constraint” può essere ottenuto come operazione in logica booleana di altri “constraint”. Questo tipo di merging non va necessariamente a risolvere conflitti nelle policy, ma può rilevare quando un numero eccessivo di constraint va a rendere la collezione di asset inutilizzabile(una policy permette l'uso di un asset solo la mattina, la seconda solo la sera, unendole l'asset risulta inutilizzabile).