# RT0 syntax

- A, B, D: principals
- r, r1, r2: role names
- A.r: a role (a principal + a role name)

Four types of credentials:

| Type | Explanation |
|---|---|
| Type 1: A.r ← D | Role A.r contains principal D as a member |
| Type 2: A.r ← B.r1 | A.r contains role B.r1 as a subset |
| Type 3: A.r ← A.r1.r2 | A.r ⊇ B.r2 for each B in A.r1 |
| Type 4: A.r ← A1.r1 ∩ A2.r2 | A.r contains the intersection |

| Example | semantics | definition |
|---|---|---|
| Epub.discount ← Alice | Alice ∈ [[Epub.discount]] | Alice belongs to the role Epu.discount |
| Epub.discount ← StateU.student | [[StateU.student]] ⊆ [[Epub.discount]] | if StateU states that X is a student then I state that X gets a discount |
| Epub.discount ← AccredBureau.university.student | For every $X$ ∈ [[AccredBureau.university]], [[X.student]] ⊆ [[Epub.discount]] | If AccredBureau states that $X$ is an accredited university and $X$ states that $Y$ is a student then I state that $Y$ gets a discount |
| ITbizz.maysign ← ITbizz.manager ∩ ITbizz.senior | [[ITbizz.manager]] ∩ [[ITbizz.senior]] ⊆ [[ITbizz.maysign]] | Anyone showing a manager certificate and a senior certificate, both signed by ITbizz may sign |

**Find the semantics**

- Alice.s ← Alice.u.v
- Alice.u ← Bob
- Bob.v ← Charlie
- Bob.v ← Charlie.s
- Charlie.s ← David
- Charlie.s ← Edward

**Solution**

- [[Charlie.s]] = {David, Edward}
- [[Bob.v]] = {Charlie, David, Edward}
- [[Alice.u]] = {Bob}
- [[Alice.s]] = {Charlie, David, Edward}