

POC CMDB

Benoit Simard (@logisima)

Le scoring

L'objectif

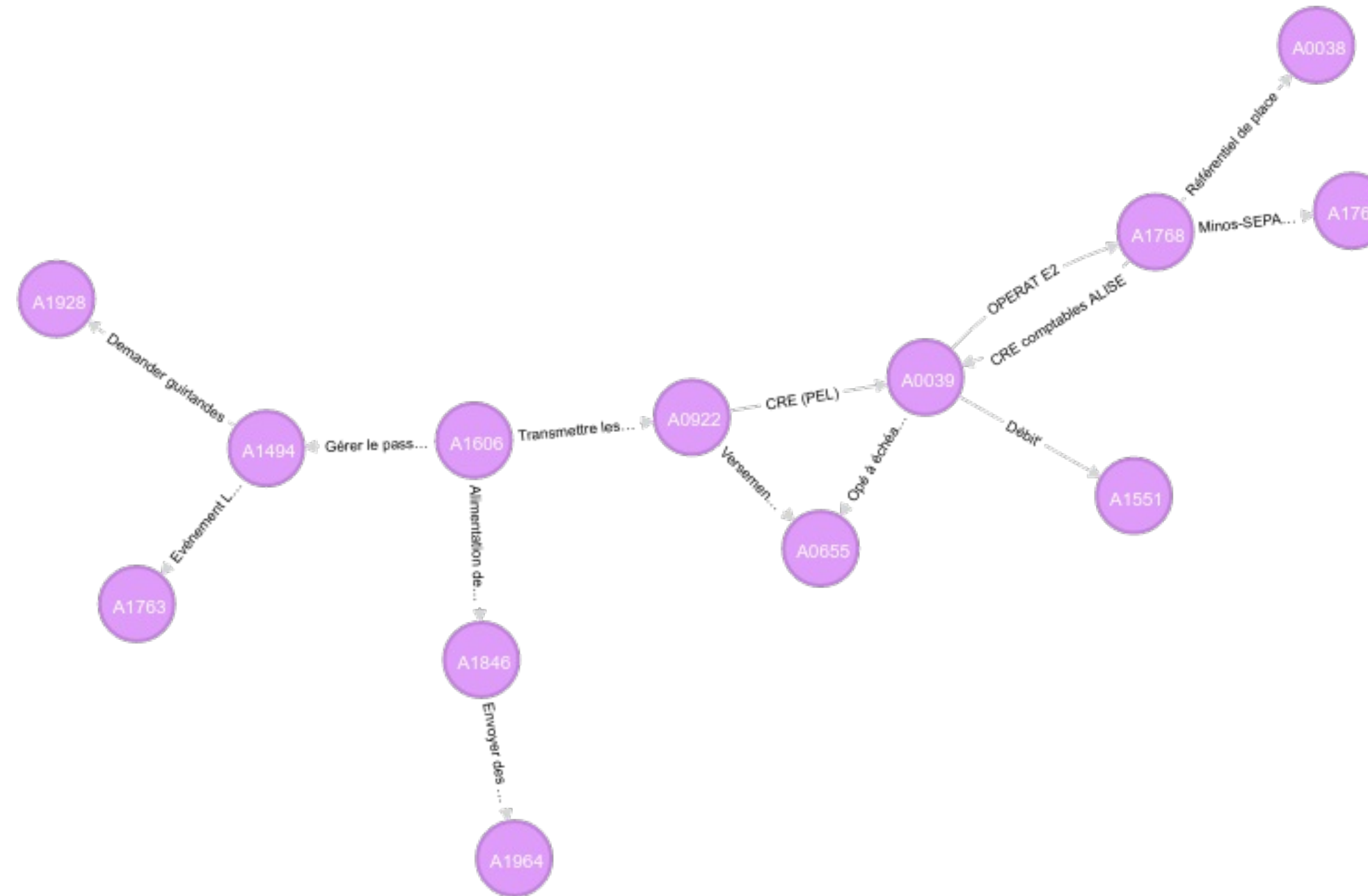
Réaliser un outil de CMDB pour les agents responsables de la qualification (SI) des tickets d'intervention. Permettre d'obtenir simplement une vue agrégée des données du ticket ainsi que de son impact sur le SI.

Obtenir un score pour chaque ticket

- Connaitre la liste des applications impactées
- Avoir une fonction de scoring pour chaque application
- Créer un score global pondérer

La liste des applications impactées

On suit les flux obligatoires



```
1 MATCH (a:Application {irt:"A1606"})-[r:FLUX*..4]-(a2:Application)
2 WHERE WHERE ALL(rel in r WHERE rel.mandatory = '1')
3 RETURN a2
```

Le score par application



Le ratio nombre d'incidents / le nombre total de ticket (tout confondu)

```
1 MATCH (a:Application)
2 WITH
3   size((:Incident)<-[:HAS_INCIDENT]-(a)) AS nbIncident,
4   size((:Ticket)-[:FOR_APPLICATION]->(a)) AS nbTicket
5 RETURN
6   CASE
7     WHEN (nbIncident + nbTicket) = 0 THEN 0 // avoid div / 0
8     WHEN (nbIncident + nbTicket) = nbIncident THEN 0.5 // missing nb ticket information
9     ELSE toFloat(nbIncident) / (nbIncident + nbTicket) // make the ratio
10  END AS score
```

Le score globale

La somme de tous les scores, pondérés par la profondeur

Plus une application est loin dans la chaine d'impacte, moins son score compte dans la note globale

```
1 MATCH p=(a:Application {irt:'A1606'})-[r:FLUX*..4]->(a2:Application)
2 WHERE ALL(rel in r WHERE rel.mandatory = '1')
3 UNWIND NODES(p) AS node
4 WITH node, reduce(x=[0,0], bc IN NODES(p) | CASE WHEN id(node)=id(bc) THEN [x[0]+1, x[0]] ELSE [x[0]+1, x[1]] END
   )[1] AS level
5 WHERE level > 0 AND NOT id(node)= id(a)
6 WITH node AS application, MIN(level) AS level
7 WITH level,
8   size((:Incident)-[:HAS_INCIDENT]-(application)) AS nbIncident,
9   size((:Ticket)-[:FOR_APPLICATION]->(application)) AS nbTicket
10 WITH
11   CASE
12     WHEN (nbIncident + nbTicket) = 0 THEN 0 // avoid div / 0
13     WHEN (nbIncident + nbTicket) = nbIncident THEN 0.5 // missing nb ticket information
14     ELSE toFloat(nbIncident) / (nbIncident + nbTicket) // make the ratio
15   END AS score , level
16 WHERE score > 0
17 RETURN SUM(score*(5-level))/ SUM(5-level)
```

Les points d'améliorations



La méthode actuelle ne prend pas en compte les éléments suivants :

- La classification des tickets
- L'impact des tickets
- La sensibilité de l'application
- Prise en compte temporelle (Time to Live)