

École d'été interdisciplinaire en numérique de la santé 2024



Modélisation de la Reconnaissance d'Activités dans un Habitat Intelligent à l'Aide d'une Ontologie

Par Hubert Ngankam

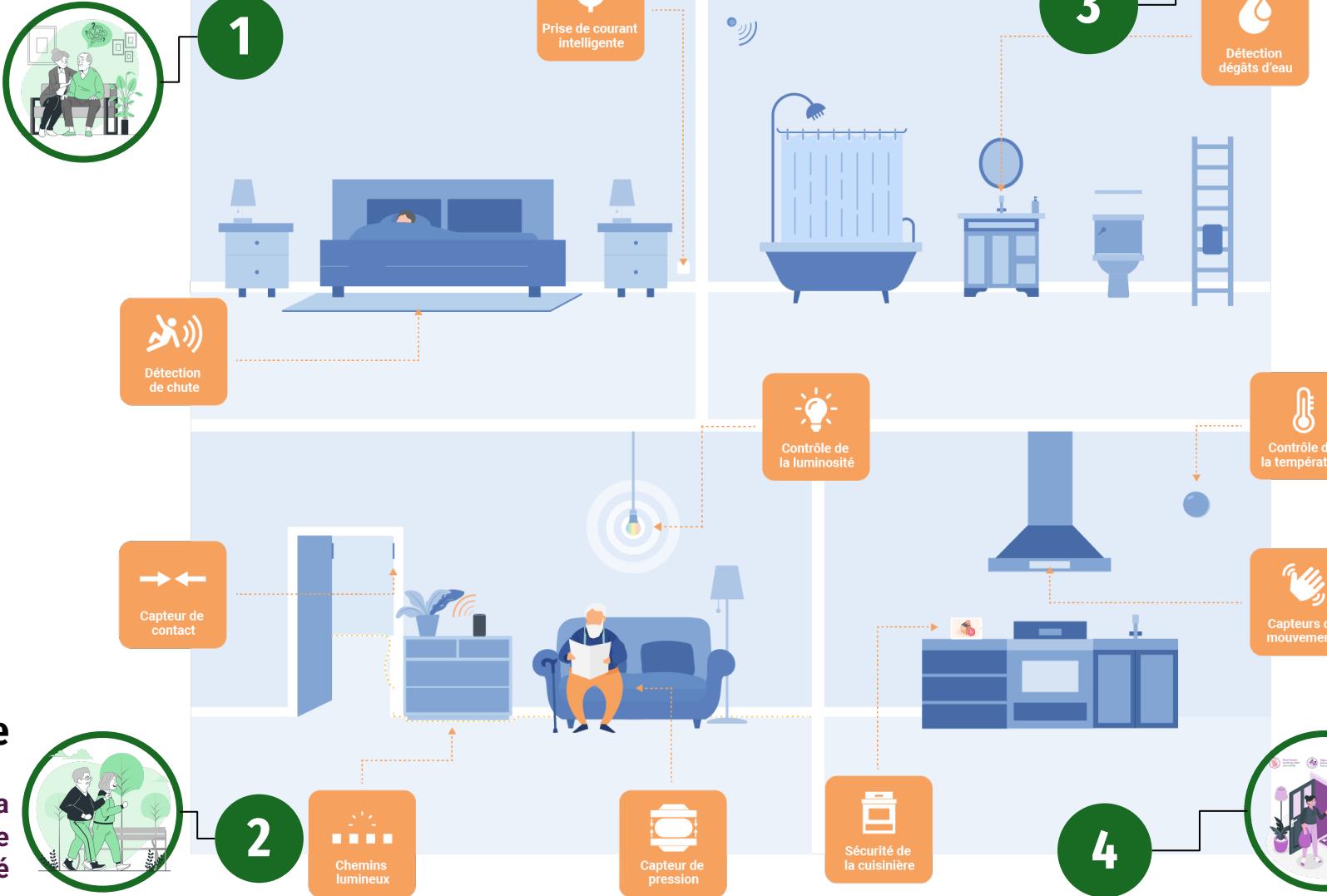
Habitat intelligent

Télémonitoring

Surveillance automatique de l'activité et profil des du vieillissement.

Connecté

Environnement de vie qui intègre différentes solutions technologiques



Santé connectée

Amélioration de la prestation des soins de santé

Assistance

l'environnement domestique plus réactif et adaptatif aux besoins des occupants en simplifiant les tâches quotidiennes des habitants.



La reconnaissance d'activités

- Est une technique utilisée pour **identifier** et **comprendre** les actions effectuées par les individus dans un environnement donné afin de **surveiller** et **d'interpréter** les mouvements et les comportements des occupants.

Exemple de reconnaissance d'activités

Par exemple, **des capteurs de mouvement, des caméras, des capteurs de pression, des microphones, des capteurs électriques ou des capteurs de contacts** installés dans des zones de la maison pour détecter des activités telles que **marcher, cuisiner, regarder** la télévision, **dormir**, etc.

Pourquoi choisir les ontologies

- Prenons un exemple où la *simple programmation* peut être laborieux, mais où les *ontologies* peuvent apporter une solution plus flexible et évolutive
- *Exemple*
 - Gérer les **préférences et les contraintes** des occupants dans un habitat intelligent de manière à optimiser l'utilisation des ressources tout en respectant les **préférences individuelles**.

Joseph et Marie

- Imaginons un scénario où un couple, **Joseph et Marie**, vit dans une maison intelligente.
- Jean aime que la température de la maison soit maintenue **à 20°C** pendant la journée, tandis que Marie préfère **22°C**.
- De plus, ils ont des contraintes budgétaires qui limitent l'utilisation excessive de l'énergie.



Solution avec une simple programmation



Difficile de gérer de manière efficace ces préférences et contraintes variables.



Par exemple, un système de contrôle de la température basé sur **des règles programmées** pourrait simplement suivre une **plage de température** fixe sans tenir **compte des préférences individuelles**.



De même, un système basé uniquement sur des règles de contrôle budgétaire pourrait ne pas prendre en compte le confort des occupants.



Atelier pratique

Ontologie pour la gestion des dispositifs et des capteurs dans un habitat intelligent

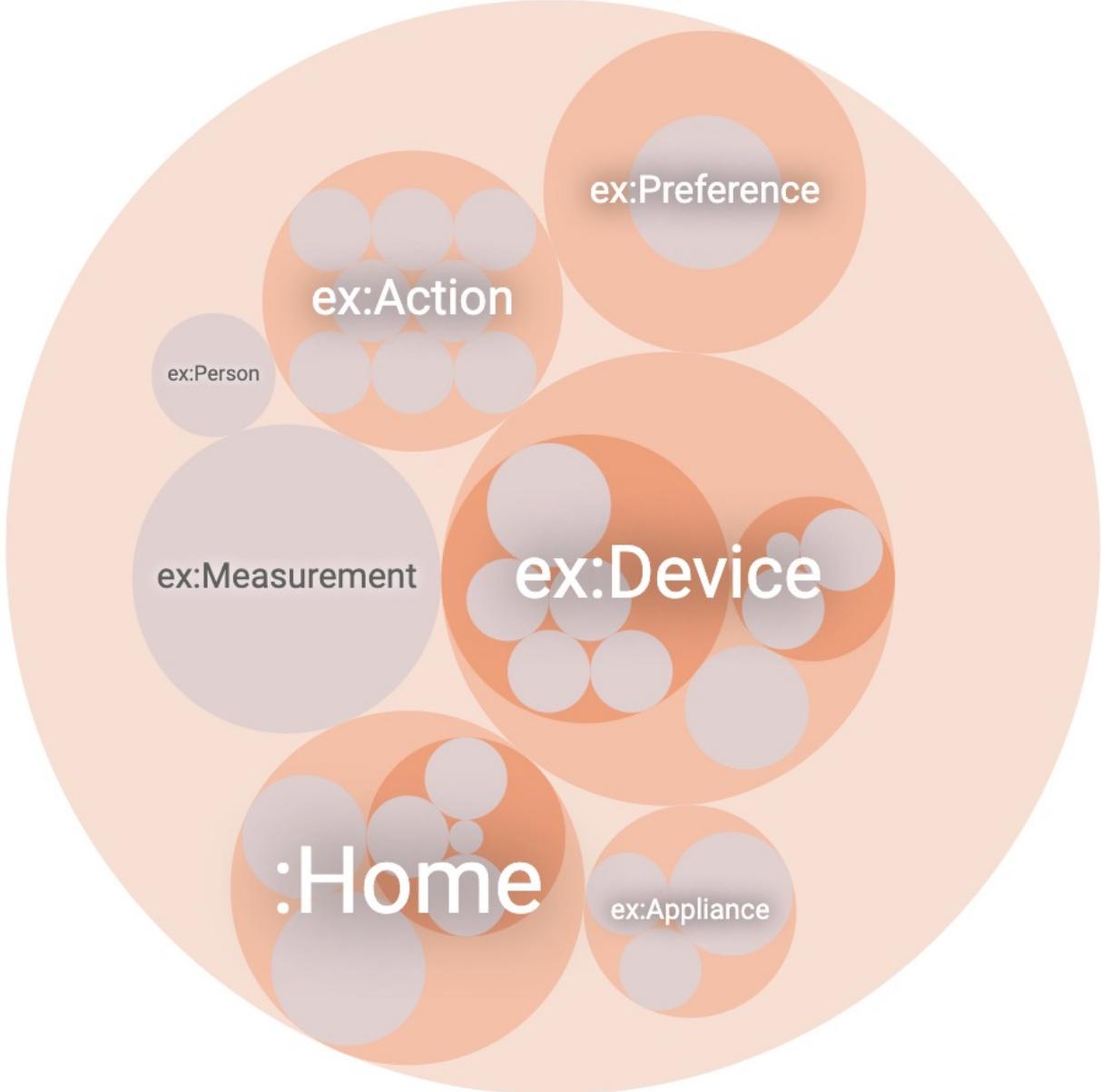


+

.

o

Ce que l'on veut faire



Une ontologie

- Pour faciliter la gestion des dispositifs,
- Pour le suivi de leur état, et permettre au système de prendre des décisions en fonction des données collectées.
- Pour par exemple, si la **température dépasse un certain seuil**, le système **peut activer le système de climatisation** via l'actionneur correspondant, en se basant sur les règles définies dans l'ontologie.



protégé



Classe Dispositif (*Device*)

Sous classes et propriétés



Sous-classes

Capteur (*Sensor*)

Actionneur (*Actuator*)

Luminaire (*Luminaire*)

Propriétés

Identifiant

Emplacement (*hasLocation*)

Type de dispositif (*hasType*)

La classe Capteur

- Sous-classes
 - Capteur de température (**TemperatureSensor**)
 - Capteur de luminosité (**LightSensor**)
 - DéTECTEUR de présence (**PresenceSensor**)
 - Capteur de pression (**PressureSensor**)
- Propriétés
 - Unité de mesure (**hasUnitOfMeasure**)
 - Plage de mesure (**hasMeasurementRange**)
 - Fréquence d'échantillonnage (**hasSamplingFrequency**)

La classe Actionneur



Sous-classes

Chauffage (*Heating*)

Climatisation (*AirConditioning*)

Éclairage (*Lighting*)



Propriétés

État (Allumé/éteint) (*hasState*)

Plage de contrôle (*hasControlRange*)

Puissance nominale (*hasPowerRating*)

Ontologie des dispositifs

- Avec cette ontologie, le système d'habitat intelligent peut organiser et interpréter les informations sur les dispositifs et les capteurs de manière cohérente.
- Par exemple, lorsque le système reçoit des données d'un capteur de température, il peut utiliser l'ontologie pour déterminer l'unité de mesure (comme Celsius), la plage de mesure acceptable, et même l'emplacement du capteur dans la maison.



Requêtes SPARQL

Gestion de l'éclairage automatique

- Récupérer les capteurs de luminosité (instances de la classe ex:LightSensor) et associe chaque capteur à une intensité lumineuse en fonction de la valeur de la luminosité mesurée.

```
1 PREFIX ex: <http://domus.usherbrooke.ca/ontologies/moredaha/2024/01/>
2
3 SELECT ?capteur ?intensite
4 WHERE {
5   ?capteur a ex:LightSensor ;
6           ex:hasMeasurement ?luminosite .
7   ?luminosite ex:hasValue ?value .
8   BIND(IF(?value < 100, "Élevée", "Basse") AS ?intensite)
9 }
10 |
```

Contrôle de la température ambiante

- Récupère les capteurs de température (instances de la classe ex:TemperatureSensor) et associe chaque capteur à une action de contrôle de la température en fonction de la valeur de température mesurée.

```
1 PREFIX ex: <http://domus.usherbrooke.ca/ontologies/moredaha/2024/01/>
2
3 SELECT ?capteur ?action
4 WHERE {
5   ?capteur a ex:TemperatureSensor ;
6           ex:hasMeasurement ?temperature .
7   ?temperature ex:hasValue ?value .
8   BIND(IF(?value < 20, "Activer Chauffage",
9          IF(?value > 25, "Activer Climatisation", "Aucune Action")) AS ?action)
10 }
11
```

Détection de présence et gestion de la sécurité

- Récupère les détecteurs de présence (instances de la classe *ex:PresenceSensor*) et associe chaque détecteur à une action en fonction de la présence détectée.

```
1 PREFIX ex: <http://domus.usherbrooke.ca/ontologies/moredaha/2024/01/>
2
3 SELECT ?detecteur ?action
4 WHERE {
5   ?detecteur a ex:PresenceSensor ;
6           ex:hasMeasurement ?presence .
7   ?presence ex:hasValue ?value .
8   BIND(IF(?value = "true", "Allumer Lumières", "Aucune Action") AS ?action)
9 }
10
11
```



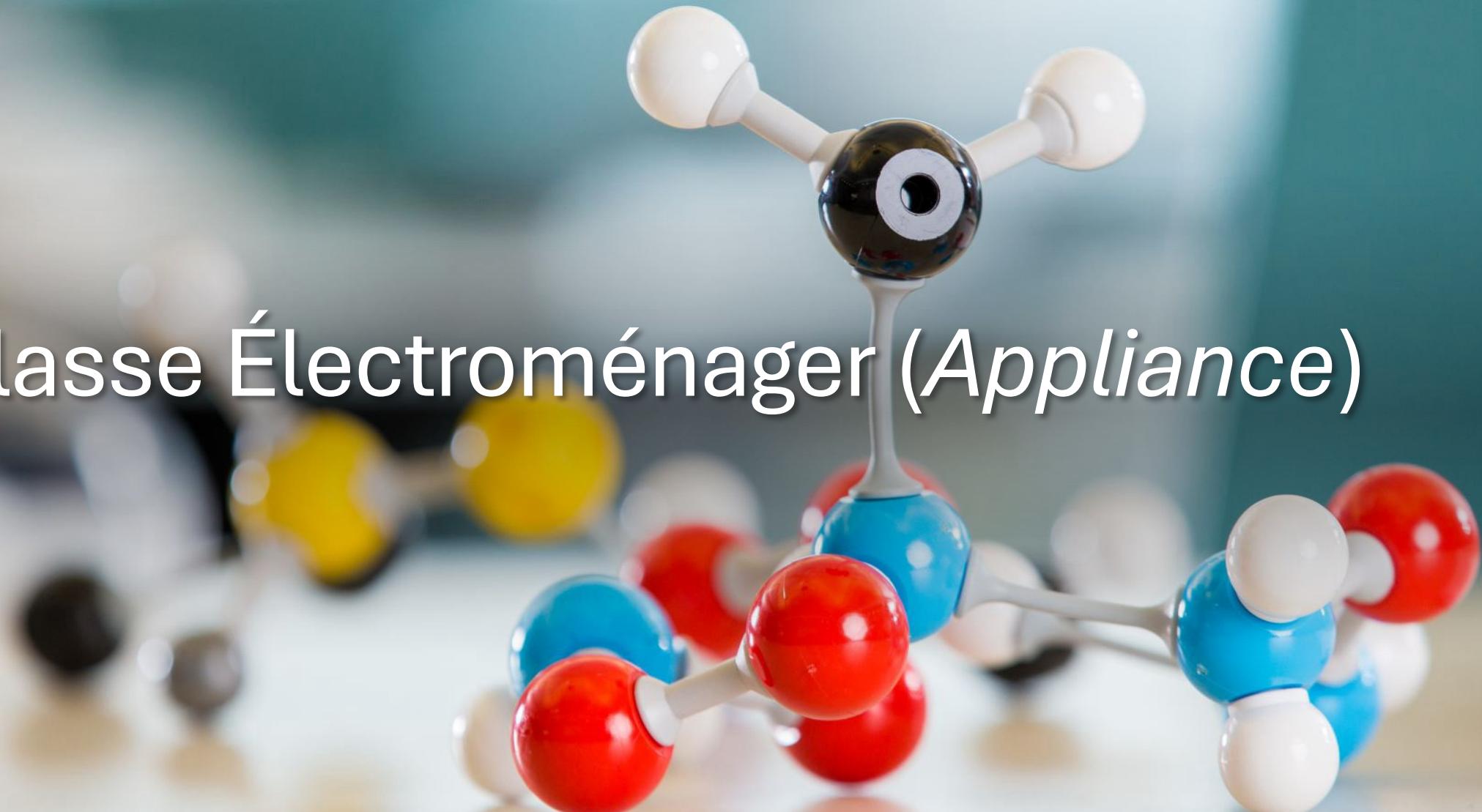
Classe Maison (*Home*)



Sous classes et propriétés

- Sous-classes
 - Porte (Door)
 - Pièce (Room)
 - Fenêtre (Window)
- Propriétés
 - hasDevice
 - isLocatedIn
 - hasDoor
 - hasWindow

Classe Électroménager (Appliance)



Sous-classes et propriétés



Sous-Classe

Réfrigérateur (*Refrigerator*)

Télévision (*TV*)



Propriétés

hasAppliance

isLocated



Classe Personne (*Person*)

Classes et propriétés

Classe

- Personne
(Person)

Propriétés

- isLocatedIn
- isUsing
- isWatching
- isCooking

Qui regarde la TV

- Récupère les personnes qui regardent la television dans le salon

```
1 PREFIX ex: <http://domus.usherbrooke.ca/ontologies/moredaha/2024/01/>
2
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 SELECT ?person
5 WHERE {
6     ?person rdf:type ex:Person .
7     ?person ex:isLocatedIn ex:livingroom1 .
8     ex:tv1 ex:isLocatedIn ex:livingroom1 .
9     ex:tv1 ex:hasState "Allumé" .
10 }
11
```



Reconnaissance d'activité

Classe pour les actions



Action

Une classe générique pour représenter toutes les actions possibles.



TurnOnLightAction

Une sous-classe spécifique pour représenter l'action d'allumer les lumières.



TurnOnTVAction

Une sous-classe spécifique pour représenter l'action d'allumer la télévision.



TurnOnStoveAction

Une sous-classe spécifique pour représenter l'action d'allumer le four.



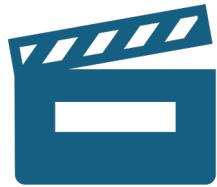
GoOutAction

Une sous-classe spécifique pour représenter l'action de sortir.

Classe pour les actions

- **GoToBedAction :**
 - Une sous-classe spécifique pour représenter l'action de se coucher.
- **PrepareMealAction :**
 - Une sous-classe spécifique pour représenter l'action de préparer un repas.
- **UseFridgeAction :**
 - Une sous-classe spécifique pour représenter l'action d'utiliser le frigo.
- **TurnOnAirConditionerAction**
 - Une sous-classe spécifique pour représenter l'action de mettre la climatisation.

Propriétés : actions – individus - devices



performsAction

Une propriété pour indiquer qu'un individu exécute une action.



actionTargetsDevice

Une propriété pour indiquer que l'action cible un dispositif spécifique.



actionInvolvesRoom

Une propriété pour indiquer que l'action se déroule dans une pièce spécifique.

Modélisation de la relation préparer un repas

- ***PrepareMealAction1***
 - est l'identifiant de l'action de préparer un repas.
 - associé aux actions spécifiques
 - telles que ***UseFridgeAction1***, ***TurnOnLightAction1*** et ***TurnOnStoveAction1*** en utilisant la propriété **isEquivalentTo** pour exprimer que ces actions sont équivalentes à l'action de préparer un repas.
- Ensuite,
- nous allons associer la présence dans la cuisine et l'utilisation de la cuisinière à des individus spécifiques (***KitchenPresence*** et ***isUsing***) en utilisant la propriété ***actionTargetsDevice***.



TV

personnes regardant la télévision dans le salon lorsque la luminosité est faible

```
2 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
3 SELECT ?person
4 WHERE {
5     ?person ex:isWatchingTV true .
6     ?person ex:isLocatedIn ex:livingroom1 .
7     ?tv ex:isLocatedIn ex:livingroom1 .
8     ?tv ex:hasState "Allumé" .
9     ?tv ex:isMeasuredBy ?tvSensor .
10    ?tvSensor rdf:type ex:LightSensor .
11    ?tvSensor ex:hasMeasurement ?measurement .
12    ?measurement ex:hasValue ?lightValue .
13    FILTER (?lightValue < 50) .
14 }
```

Joseph et Marie

- Imaginons un scénario où un couple, **Joseph et Marie**, vit dans une maison intelligente.
- Jean aime que la température de la maison soit maintenue **à 20°C** pendant la journée, tandis que Marie préfère **22°C**.
- De plus, ils ont des contraintes budgétaires qui limitent l'utilisation excessive de l'énergie.



```
1 PREFIX ex: <http://domus.usherbrooke.ca/ontologies/moredaha/2024/01/>
2
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 SELECT ?person ?preferredTemperature
5 WHERE {
6     ?person rdf:type ex:Person .
7     ?person ex:hasPreference ?preference .
8     ?preference ex:hasTemperature ?preferredTemperature .
9     FILTER (?preferredTemperature = 20 || ?preferredTemperature = 22)
10 }
11
```