ONTOLOGIES (2)

Raisonnement en $\mathcal{ALC}$
Logiques de description riches

LRC - 2025/2026

Gauvain Bourgne

Raisonnement

Raisonnement

Raisonnements terminologiques

- Tests
  - Test de satisfiabilité de concept : C est-il satisfiable selon $\mathcal{T}$? (C $\mathcal{T} - SAT$?)
  - Test de subsomption : C est-il subsumé par D selon $\mathcal{T}$? ($\mathcal{T} \models C \sqsubseteq D$?)
- Tâche
  - Classification de concept : classer les concept d'une Tbox selon la relation de spécialisation pour obtenir la hiérarchie de concepts.

Raisonnement

Raisonnements assertionnels

- Tests
  - Test de cohérence : $\mathcal{A}$ est-elle cohérente selon $\mathcal{T}$? ($\mathcal{A}$ $\mathcal{T} - SAT$?)
  - Vérification d'instance : a est-elle une instance du concept C selon $\langle \mathcal{T}, \mathcal{A} \rangle$ et ? ($\mathcal{T}, \mathcal{A} \models a : C$?)
- Tâche
  - Recherche d'information : trouver tous les individus qui sont des instances d'un concept C selon la KB.
  - Réalisation : étant donné une KB et une constante a, trouver les concepts les plus spécifiques dont a est une instance selon la KB.

La logique $\mathcal{ALC}$ – Raisonnement

SCIENCES SORBONNE UNIVERSITÉ

Type de raisonnements

- Vérification de cohérence : $\mathcal{A}$ est-elle cohérente selon $\mathcal{T}$?
  $\mathcal{A}$ $\mathcal{T} - SAT$? est-ce que $\langle \mathcal{T}, \mathcal{A} \rangle \not\models \bot$?
  Cet ensemble d'affirmations est-il contradictoire ?
- Reconnaissance d'instance : a est-elle une instance du concept C selon $\langle \mathcal{T}, \mathcal{A} \rangle$? $\mathcal{T}, \mathcal{A} \models a : C$?
  Est-ce que, selon nos connaissances, x est un C ?
- Satisfiabilité de concept : C est-il satisfiable selon $\mathcal{T}$?
  C $\mathcal{T} - SAT$? ou est-ce que $\mathcal{T} \not\models C \sqsubseteq \bot$?
  Peut-on être C ? Existe-t-il des individus vérifiant C ? (typiquement C est un concept complexe)
- Test de subsomption : C est-il subsumé par D selon $\mathcal{T}$?
  $\mathcal{T} \models C \sqsubseteq D$?
  Est-ce que tous les C sont forcément des D ($\mathcal{T} \models C \sqsubseteq D$) ? Est-il possible que des C soient des D ($\mathcal{T} \not\models C \sqsubseteq \neg D$) ?

La méthode des tableaux en $\mathcal{ALC}$

SCIENCES SORBONNE UNIVERSITÉ

Principes

- Algorithme pour la vérification de cohérence d'un Abox
- Etapes (sans Tbox)
  1. Traduire le problème en un problème de vérification d'une Abox
  2. Mettre la Abox sous forme normale négative (NNF)
  3. Appliquer les règles tableau jusqu'à trouver une feuille complète ouverte ou avoir complété toutes les feuilles.
  4. Conclure : la Abox est cohérente ssi il y a au moins une feuille (complète) ouverte dans le tableau. (elle est donc incohérente ssi toutes les feuilles du tableau complet sont fermées)
- CLASH : quand on a une contradiction (soit $a : C$ et $a : \neg C$, soit $a : \bot$) dans un nœud, on ferme la feuille.

La méthode des tableaux en $\mathcal{ALC}$

Mise sous forme normale négative

On s'assure que toutes les négations de concept soient devant un concept atomique et non une expression de concept en appliquant les règles de réécriture suivantes :

$$\begin{array}{rcl}
\neg(\neg C) & \leftrightarrow & C \\
\neg(C \sqcap D) & \leftrightarrow & \neg C \sqcup \neg D \\
\neg(C \sqcup D) & \leftrightarrow & \neg C \sqcap \neg D \\
\neg(\forall R.C) & \leftrightarrow & \exists R.\neg C \\
\neg(\exists R.C) & \leftrightarrow & \forall R.\neg C
\end{array}$$

La méthode des tableaux en $\mathcal{ALC}$

SCIENCES SORBONNE UNIVERSITÉ

Les règles de tableau

Règles conjonctives

Règle	**condition**	**action**
$R_\sqcap$	$a : C \sqcap D$	ajout de $a : C$ et de $a : D$
$R_\forall$	$a : \forall R.C$ et $a, e : R$	ajout de $e : C$
$R_\exists$	$a : \exists R.C$ et aucun e tel que $e : C$ et $a, e : R$	ajout de $a, b : R$ et de $b : C$ où b nouvelle constante

Règle disjonctive

Règle	**condition**	**action**
$R_\sqcup$	$a : C \sqcup D$	Duplication en $\mathcal{A}_g$, $\mathcal{A}_d$ ajout de $a : C$ à $\mathcal{A}_g$ et de $a : D$ à $\mathcal{A}_d$

Exemple : vérification de cohérence d'une Abox simple

$\mathcal{A}_1$:

(1) $\texttt{pierreRichard} : \forall\texttt{joueDans}.\texttt{Comédie}$
(2) $\texttt{pierreRichard} : (\exists\texttt{joueDans}.\texttt{Romance}) \sqcap \texttt{Réalisateur}$
(3) $\texttt{pierreRichard}, \texttt{laChevre} : \texttt{joueDans}$
(4) $\texttt{laChevre} : \neg(\texttt{Comédie} \sqcap \texttt{Romance})$

La logique $\mathcal{ALC}$ – Raisonnement

Réduction à la vérification de cohérence

- Reconnaissance d'instance :
  $$\begin{array}{lll} \mathcal{T}, \mathcal{A} \models a : C & \text{ssi} & \mathcal{T}, \mathcal{A} \cup \{a : \neg C\} \models \bot \\ & \text{ssi} & \mathcal{A} \cup \{a : \neg C\} \text{ est } \mathcal{T} - UNSAT \end{array}$$
- Satisfiabilité de concept :
  $$\begin{array}{lll} C\ \mathcal{T} - SAT & \text{ssi} & \mathcal{T} \not\models C \sqsubseteq \bot \\ & \text{ssi} & \mathcal{T}, \{e : C\} \not\models \bot \\ & \text{ssi} & \{e : C\} \text{ est } \mathcal{T} - SAT \end{array}$$
- Test de subsomption
  $$\begin{array}{lll} \mathcal{T} \models C \sqsubseteq D & \text{ssi} & \mathcal{T} \models C \sqcap \neg D \sqsubseteq \bot \\ & \text{ssi} & C \sqcap \neg D \text{ est } \mathcal{T} - UNSAT \\ & \text{ssi} & \mathcal{T}, \{e : C \sqcap \neg D\} \models \bot \\ & \text{ssi} & \{e : C \sqcap \neg D\} \text{ est } \mathcal{T} - UNSAT \end{array}$$

La logique $\mathcal{ALC}$ – Syntaxe

SCIENCES SORBONNE UNIVERSITÉ

Typologie d'axiomes

Dans cette boîte, on note A les concepts atomiques, C, C_1, C_2 les expression de concepts (atomique ou non) et E les concepts complexes (expression de concept non atomique)

- $A \equiv C$ est un axiome définitoire
- $A \sqsubseteq C$ est un axiome d'inclusion simple
- $E \sqsubseteq C$ est un axiome d'inclusion générale
- On néglige les axiomes d'équivalence générale ($E \equiv C$), qui peuvent être décomposé en deux inclusions générale
- Un axiome $C_1 \equiv C_2$ ou $C_1 \sqsubseteq C_2$ est dit cyclique s'il existe un concept atomique A qui apparaît à la fois dans C_1 et C_2.

La logique $\mathcal{ALC}$ – Syntaxe

SCIENCES SORBONNE UNIVERSITÉ

Typologie de Tbox

- Une Tbox définitoire est une Tbox qui ne contient que des axiomes définitoires
- Une Tbox simple est une Tbox qui ne contient que des axiomes d'inclusions simples ou des axiomes définitoires
- Une Tbox générale est une Tbox qui contient au moins un axiome d'inclusion générale.
- Un Tbox est dite acyclique si elle ne contient pas d'axiome cyclique ni de cycle (ensemble d'axiomes $G_1 \sqsubseteq D_1, \ldots, G_n \sqsubseteq D_n$ tel que pour chaque i, il existe A_i apparaissant à la fois dans D_i et dans G_{i+1} (ou G_1 si $i = n$)).

La méthode des tableaux en $\mathcal{ALC}$

SCIENCES SORBONNE UNIVERSITÉ

Prise en compte d'une Tbox définitoire acyclique

Un simple prétraitement de la Abox $\mathcal{A}$ permet de se ramener au cas d'une Tbox vide.
Réécriture avec les définitions simples : Tant que la Abox contient un concept atomique A tel que A a une définition dans la Tbox ($A \equiv C \in \mathcal{T}$), on **remplace** A par C dans la Abox.
A la fin, la Abox finale $\mathcal{A}_{\mathcal{T}}$ ne contient que des concepts primitifs (c'est-à-dire des concepts qui n'apparaissent jamais à gauche d'une définition ou d'une inclusion dans la Tbox).
Alors : $\mathcal{A}$ est SAT selon $\mathcal{T}$ ssi $\mathcal{A}_{\mathcal{T}}$ est SAT (selon $\emptyset$).

Exemple : Satisfiabilité de concept avec Tbox définitoire

SCIENCES SORBONNE UNIVERSITÉ

$$\mathcal{T}: \boxed{\begin{array}{l} \texttt{SansOption} \equiv \forall \texttt{suit}.\texttt{TroncCommun} \\ \texttt{RétifInfo} \equiv \forall \texttt{suit}.\neg\texttt{Info} \\ \texttt{Info} \equiv \texttt{troncCommun} \sqcup \texttt{OptionsInfos} \\ \texttt{Etudiant} \equiv \exists \texttt{suit}.\top \\ \texttt{EtuSansOption} \equiv \texttt{Etudiant} \sqcap \texttt{SansOption} \end{array}}$$

On veut s'il peut y avoir des étudiants sans option rétifs à l'informatique, c'est-à-dire savoir si $\texttt{EtuSansOption} \sqcap \texttt{RétifInfo}$ est $\mathcal{T}$.

La méthode des tableaux en $\mathcal{ALC}$

SCIENCES SORBONNE UNIVERSITÉ

Prise en compte d'une Tbox simple acyclique

On met la Tbox sous forme normale négative.

Déploiement des inclusions simples : Pour chaque triplet (e, A, C) (avec e constante, A concept atomique et C expression de concept) tel que la Abox contienne l'assertion $e : A$ et que la Tbox contienne l'inclusion $A \sqsubseteq C$, on **ajoute** (si non présent) l'assertion $e : C$. On itère tant que cela génère des ajouts.

Ici, il est nécessaire, après le déploiement initial, de refaire un déploiement *chaque fois qu'on ajoute une assertion $e : A$ où A concept est un concept atomique apparaissant à gauche d'une inclusion*.

Exemple : Reconnaissance d'instance avec Tbox simple

$$\mathcal{T}: \boxed{\begin{array}{l} \texttt{Célèbre} \sqcap \texttt{Nain} \sqsubseteq \texttt{Apprécié} \\ \exists \texttt{joue.Nain} \sqsubseteq \texttt{Nain} \end{array}}$$

$$\mathcal{A}: \boxed{\begin{array}{l} \texttt{peterDinklage} : \texttt{Célèbre} \\ \texttt{tyrion} : \texttt{Nain} \\ \texttt{peterDinklage}, \texttt{tyrion} : \texttt{joue} \end{array}}$$

On veut montrer que d'après ces connaissances, Peter Dinklage est apprécié, c'est-à-dire que : $\mathcal{A}, \mathcal{T} \vdash \texttt{p} : \texttt{A}$

La méthode des tableaux en $\mathcal{ALC}$

Prise en compte d'une Tbox générale

Réecriture des axiomes généraux. Chaque axiome d'inclusion générale $C \sqsubseteq D$ est remplacé par sa réécriture $\top \sqsubseteq \neg C \sqcup D$. On met ensuite la Tbox sous NNF.

Déploiement des axiomes généraux : Pour chaque constante a apparaissant dans la Abox et pour chaque inclusion $\top \sqsubseteq E$ de la Tbox, on **ajoute** (si non présent) l'assertion $a : E$.

Ici, il est nécessaire, après le déploiement initial, de refaire un déploiement *chaque fois qu'on ajoute une nouvelle constante (règle $R_\exists$).*

Exemple : Subsomption avec Tbox générale

$$\mathcal{T} : \boxed{\begin{array}{l} \mathtt{A} \sqcup \mathtt{B} \sqsubseteq \mathtt{C} \\ \forall \mathtt{r}.\mathtt{D} \sqsubseteq \mathtt{B} \\ \forall \mathtt{r}.\mathtt{E} \sqsubseteq \mathtt{A} \\ \mathtt{F} \sqcup \mathtt{D} \sqsubseteq \mathtt{E} \end{array}}$$

On veut montrer que d'après ces connaissances, $\mathcal{T} \vdash \forall \mathtt{r}.\mathtt{F} \subseteq \mathtt{C}$

Extensions de $\mathcal{ALC}$

Concepts plus expressifs

Restrictions de cardinalité – $\mathcal{N}$

- *Syntaxe.* $\geq n\ R$. Constructeur de concept binaire prenant comme arguments un entier naturel n et un rôle R.
- *Signification.* Désigne *les individus en relation* R *avec au moins* n *individus.*
  Exemples : $\texttt{Espion} \sqsubseteq \geq 3\ \texttt{aIdentite}$
  $\texttt{Colocation} \equiv \texttt{Location} \sqcap \geq 2\ aLocataire$
- *Sémantique formelle.* Pour une structure M
  $(\geq n\ R)^M = \{x \in \Delta^M | \mathrm{Card}(\{y \in \Delta^M | (x,y) \in R^M\}) \geq n\}$
- *Traduction LPPO.* $\tau_x(\geq n\ R) =$
  $\exists Y_1 \ldots \exists Y_n (\bigwedge_{i \in \{1,\ldots,n\}} R(x, Y_i) \wedge \bigwedge_{(i,j) \in \{1,\ldots,n\}^2, i<j} (Y_i \neq Y_j))$
- On construit aussi $\leq n\ R = \neg(\geq n+1\ R)$.
  Exemple : $\texttt{EnfantUnique} \equiv \forall \texttt{aParent}. \leq 1\ \texttt{aEnfant}$

Concepts plus expressifs

Restrictions de cardinalité qualifiées – $\mathcal{Q}$

- *Syntaxe.* $\geq n\ R.C$. Constructeur de concept ternaire prenant comme arguments un entier naturel n, un rôle R et un concept C.
- *Signification.* Désigne *les individus en relation R avec au moins n individus du type C.*
  Exemples : ≥ 2 `enseigne.Informatique`
  `stendhal` $:\geq 2$ `aEcrit.Roman`
- *Sémantique formelle.* Pour une structure M $(\geq n\ R.C)^M = \{x \in \Delta^M | \text{Card}(\{y \in \Delta^M | (x,y) \in R^M \text{ et } y \in C^M\}) \geq n\}$
- *Traduction LPPO.* $\tau_x(\geq n\ R) = \exists Y_1 \ldots \exists Y_n$ $(\bigwedge_{i \in \{1,\ldots,n\}}(R(x,Y_i) \wedge C(Y_i)) \wedge \bigwedge_{(i,j) \in \{1,\ldots,n\}^2, i<j}(Y_i \neq Y_j))$
- On construit aussi $\leq n\ R.C = \neg(\geq n+1\ R.C)$.
  Exemple : ≤ 3 `aLu.Polar`

Concepts plus expressifs

Concepts nominaux – $\mathcal{O}$

- *Syntaxe.* $\{a\}$ Constructeur de concept unaire prenant en argument une constante.
  Ecriture simplifiée pour les groupes :
  $\{a_1, \ldots, a_n\} = \{a_1\} \sqcup \ldots \sqcup \{a_n\}$.
- *Signification.* Considère *l'individu nommé* comme un concept.
  Exemple :
  $\texttt{EtudiantM1} \sqcap \exists \texttt{inscrit}.\{dac, androide\} \sqsubseteq \exists \texttt{suit}.\{lrc\}$
- *Sémantique formelle.* Pour une structure M, $\{a\}^M = \{a^M\}$
- *Traduction LPPO.* $\tau_x(\{a\}) = (x = a)$

Expressions de rôles

Rôle inverse – $\mathcal{I}$

- *Syntaxe.* R^{-1} Constructeur de rôle unaire prenant en argument un rôle.
- *Signification.* Désigne *la relation inverse (ie ordre des arguments inversé)*.
  Exemple : $\texttt{EnfantUnique} \equiv \forall \texttt{aEnfant}^{-1}. \leq 1\ \texttt{aEnfant}$
- *Sémantique formelle.* Pour une structure M, $(R^{-1})^M = \{(y, x) \in \Delta^M | (x, y) \in R^M\}$
- *Traduction LPPO.* $\tau_{x,y}(R^{-1}) = R(y, x)$

Expressions de rôles

Composition de rôle – $\_(\circ)$

- *Syntaxe.* $R_1 \circ R_2$ Constructeur de rôle binaire prenant en argument deux rôles.
- *Signification.* Désigne *la relation composée (en prenant les rôles en chaîne (aucun gain d'expressivité sans axiomes de rôles)*.
  Exemple : $\mathtt{aEnfant} \circ \mathtt{aEnfant}$ pour $\mathtt{aPetitEnfant}$
  $\mathtt{enseigne} \circ \mathtt{suit}^{-1}$
- *Sémantique formelle.* Pour une structure M, $(R_1 \circ R_2)^M = \{(x, y) \in \Delta^M | \exists z \in \Delta^M, (x, z) \in R_1^M \text{ et } (z, y) \in R_2^M\}$
- *Traduction LPPO.* $\tau_{x,y}(R_1 \circ R_2) = \exists z(R_1(x, z) \wedge R_2(z, x))$

Expressions de rôles

Autres constructeurs de rôles

- Union de rôle ($\sqcup$),
  rôle complément ($\neg$),
  intersection de rôle ($\sqcap$)...
- Exemples :
  $\texttt{Fidele} \equiv \leq 1\ \texttt{enCouple} \sqcap \neg(\exists(\texttt{embrasse} \sqcap \neg\texttt{enCouple}).\top$
  $\texttt{a} : \exists(\texttt{aPere} \sqcup \texttt{aMere}).\exists(\texttt{aPere} \sqcup \texttt{aMere}).\texttt{Ouvrier}$

Axiomes de rôles

SCIENCES SORBONNE UNIVERSITÉ

Transitivité de rôle – $\_{R^+}$

$\mathcal{S} = \mathcal{ALC}_{R^+}$

- *Syntaxe.* Nouvel axiome possible : Trans R
- *Signification. Le rôle R est transitif.*
  Exemple : Trans `aAncetre`
  Trans `contient`
- *Sémantique formelle.* Pour une structure M, M est un modèle de Trans R ssi R^M est une relation transitive
  càd, R^M vérifie $\forall(x, y, z) \in (\Delta^M)^3$, si $(x, y) \in R^M$ et $(y, z) \in R^M$, alors $(x, z) \in R^M$.
- *Traduction LPPO.*
  $\forall X \forall Y \forall Z((\tau_{X,Y}(R) \wedge \tau_{Y,Z}(R)) \rightarrow \tau_{X,Z}(R))$

Axiomes de rôles

Hiérarchie de rôles simples – $\mathcal{H}$

- *Syntaxe.* $R_1 \sqsubseteq R_2$ où R_1 et R_2 sont des rôles atomiques ou des inversions de rôles atomiques.
- *Signification. R_1 est inclus dans R_2 / est une spécialisation de R_2.*
  Exemple : $\texttt{responsableUE} \sqsubseteq \texttt{enseigne}$
  $\texttt{dirige} \sqsubseteq \texttt{membreDe}$
  $\texttt{aPere}^{-1} \sqsubseteq \texttt{aEnfant}$
- *Sémantique formelle.* Pour une structure M, M est un modèle de $R_1 \sqsubseteq R_2$ ssi $(R_1)^M \subseteq (R_2)^M$.
- *Traduction LPPO.* $\forall X \forall Y(\tau_{X,Y}(R_1) \rightarrow \tau_{X,Y}(R_2))$

Axiomes de rôles

SCIENCES SORBONNE UNIVERSITÉ

Hiérarchies complexes – $s\mathcal{R}$

- *Nouveaux axiomes.*
  - $\rho \sqsubseteq R$ où R est un rôle atomique et $\rho = R_1 \circ \ldots R_n$ une chaîne de composition de rôles atomiques ou d'inversions de rôles atomiques.
  - Sym R, Ref R, Irr R, Trans R, Dis (R_1, R_2) sont des axiomes de rôles indiquant respectivement que R est Symétrique, Réflexif, Irréflexif, Transitif ou que R_1 et R_2 sont disjoints.
- *Nouveau constructeur de concept.*
  - $\exists R.Self$ définit le concept des individus en relation R avec eux-mêmes.
- *Assertions individuelles (de rôle).*
  - $a, b : R$ où R est un rôle ou une inversion de rôle atomique
  - $a, b : \neg S$ où S est un rôle simple (définition plus restreinte que R - ne doit pas avoir d'axiomes d'inclusion complexe, même de façon indirecte)

Axiomes de rôles

Hiérarchies complexes ($s\mathcal{R}$)

- Exemples :

$\texttt{aParent} \circ \texttt{aParent} \sqsubseteq \texttt{aGrandParent}$

$\texttt{aParent} \sqsubseteq \texttt{aAncetre}$

Trans $\texttt{aAncetre}$

Sym $\texttt{marié\_e}$

$\texttt{Narcissique} \equiv \exists \texttt{adore}.\mathit{Self}$

Logiques de description légères

Logiques de descriptions légères

$\mathcal{EL}$

- Expressions de concepts (A : concept atomique)

$$C ::= A | C \sqcap C | \exists R.C$$

- Pas de négations de concept ni d'union ! (réduction à la vérification de cohérence n'est plus possible pour tous les problèmes)
- Test de subsomption (Tbox définitoire acyclique) : polynomial
- Test de subsomption (Tbox générale) : polynomial

Logiques de descriptions légères

SCIENCES SORBONNE UNIVERSITÉ

$\mathcal{FL}_0$

- Expressions de concepts (A : concept atomique)

$$C ::= A | C \sqcap C | \forall R.C$$

- Pas de négations de concept, ni d'union !
- Test de subsomption (Tbox définitoire acyclique) : polynomial
- Test de subsomption (Tbox générale) : EXPTIME-complet

Logiques de descriptions légères

SCIENCES SORBONNE UNIVERSITÉ

$\mathcal{FL}^-$

$\mathcal{FL}^-$: $\mathcal{FL}_0$ + constructeur $\exists R$

- Expressions de concepts (A : concept atomique)

$$C ::= A | C \sqcap C | \exists R | \forall R.C$$

- Nouveau constructeur de concept $\exists R$
  - Sémantique : $(\exists R)^M = \{x \in \Delta^M | \exists y \in \Delta^M, (x, y) \in R^M\}$
  - Traduction LPPO : $\tau_x(\exists R) = \exists Y R(x, Y)$
  - correspondrait à $\exists R.\top$ en $\mathcal{ALC}$ (mais $\top$ non présent en $\mathcal{FL}^-$)
- Toujours pas de $\neg, \sqcup$, $\top$ ou $\bot$.
- Test de subsomption (Tbox définitoire acyclique) : reste polynomial (algo : subsomption structurelle).

Logiques de descriptions légères

SCIENCES SORBONNE UNIVERSITÉ

DL-lite

modélise primitives des modèles Entité-Association en restant dans une complexite polynomiale.

- Notations : A concept atomique, B concept de base, C concept, P rôle atomique, R rôle de base, E rôle général
- Concepts : $B ::= A|\exists R$; $C ::= B|\neg B$
- Rôles : $R ::= P|P^{-1}$; $E ::= R|\neg R$
- Tbox : uniquement axiomes de la forme $B \sqsubseteq C$

Extensions polynomiales

- Dans DL-lite$_{\mathcal{R}}$, Tbox autorise aussi $R \sqsubseteq E$
- Dans DL-lite$_{\mathcal{F}}$, Tbox autorise axiome de fonctionnalité Funct R (R^M doit être une fonction, ie si $(a, b) \in R^M$ et $(a, c) \in R^M$, on doit avoir $b = c$)

Une famille de logiques

Nomenclature

- sur base $\mathcal{ALC}$: $\mathcal{ALC}$, $\mathcal{ALCN}$, $\mathcal{ALCQ}$, $\mathcal{ALCIN}$, ... $\mathcal{ALCIQ}$
- Logiques complexes : $\mathcal{SHIQ}$, $\mathcal{SHOIN}$
- L'une des plus expressives décidables : $s\mathcal{ROIQ}$

Focus de l'UE

$$\mathcal{ALC}, \mathcal{SHOIN}, \mathcal{FL}^-$$

OWL

- `OWL-Lite` : $\mathcal{SHIF}$
- `OWL-DL` : $\mathcal{SHOIN}$
- `OWL 1.1` :$s\mathcal{ROIQ}$