Софийски университет “Св. Климент Охридски”
Факултет по математика и информатика

Проект

по

“Представяне и моделиране на знания”

на тема

“Онтология за игри”

Изготвен от: Таня Христова, ф.№4MI3400002, спец. ИИ, I курс
Явор Георгиев, ф.№7MI3400001, спец. ИИ, I курс

30.01.2022 г.

1. Идея на проекта

Идеята на проекта е да се моделират чрез онтология различни понятия от приложната област на игрите. Крайните цели на проекта включват търсене на игра по предварително зададени от потребител ограничения и улеснено класифициране на нова игра.

2. Елементи на онтологията

2.1. Класове

Атомични класове
Game ⊑ Thing
Field ⊑ Thing
Equipment ⊑ Thing
Spectator ⊑ Thing
Player ⊑ Thing

Полета
Lawn ⊑ Field
Board ⊑ Field

Екипировка
Table ⊑ Equipment
Card ⊑ Equipment
Dice ⊑ Equipment
Pawn ⊑ Equipment
Paper ⊑ Equipment
Pencil ⊑ Equipment
Computer ⊑ Equipment
SportEquipment ⊑ Equipment
Ball ⊑ SportEquipment

Зрители
Fan ≐ [AND Spectator [≥ 1 :has_interest_in.Game]]
Kibitzer ≐ [AND Spectator [≥ 1 :watches.Game] [∃ :has_to_remain_silent.{True}]]

Игри според брой играчи
SinglePlayerGame ≐ [AND Game [= 1 :played_by.Player]]
MultiPlayerGame ≐ [AND Game [≥ 2 :played_by.Player]]

Игри според вид активност
IntelectualGame ⊑ Game
LogicalGame ⊑ IntelectualGame
StrategyGame ⊑ IntelectualGame
KnowledgeGame ⊑ IntelectualGame
PhysicalActivityGame ⊑ Game

Спортни игри
TeamGame ≐ [AND MultiPlayerGame [∃ :has_team.{True}]]
SportGame ≐ [AND PhysicalActivityGame [∀ :require.SportEquipment] [∃ :watched_by.Spectator]]
IndividualSportGame ≐ [AND SinglePlayerGame SportGame]
TeamSportGame ≐ [AND TeamGame SportGame]
ProfessionalSportGame ≐ [AND SportGame [∃ :is_professional.{True}]]
ProfessionalBallGameOnLawn ≐ [AND ProfessionalSportGame [≥ 1 :require.Ball] [∀ :played_on.Lawn]]

Карти
PlayingCard ≐ [AND Card [∃ :is_normal_card.{True}]]
MagicCard ≐ [AND Card [∃ :is_custom_card.{True}]]
CardSuit ≐ {Clubs}, {Diamonds}, {Hearts}, {Spades}
CardValue ≐ {Two}, {Three}, {Four}, {Five}, {Six}, {Seven}, {Eight}, {Nine}, {Ten}, {Jack}, {Queen}, {King}, {Ace}

Настолни игри
VideoGame ≐ [AND Game [≥ 1 :require.Computer]]
TableTopGame ≐ [AND StrategyGame LogicalGame [≥ 1 :require.Table]]
BoardGame ≐ [AND TableTopGame [≥ 1 :played_on.Board]]
DiceGame ≐ [AND TableTopGame [∃ :require.Dice]]
PaperAndPencilGame ≐ [AND TableTopGame [∃ :require.[AND Paper Pencil]]]
CardGame ≐ [AND TableTopGame [∃ :require.Card]]

2.2. Свойства

2.2.1. Свойства на обектите

Property	*Domain*	*Range*	*Characteristics*
played_on	Game	Field	-
require	Game	Equipment	-
plays	Player	Game	-
played_by	Game	Player	Inverse property (of plays)
watches	Spectator	Game	-
watched_by	Game	Spectator	Inverse property (of watches)
has_interest_in	Spectator	Game	-
has_greater_number_than	Dice	Dice	Transitive property
has_lower_number_than	Dice	Dice	Transitive property, Inverse property (of has_greater_number_than)
has_favourite	Fan	Player	-
has_fans	Player	Fan	Inverse property (of has_favourite)
has_suit	PlayingCard	CardSuit	Functional property
has_value	PlayingCard	CardValue	Functional property
has_greater_value_than	PlayingCard	PlayingCard	Transitive property
has_lower_value_than	PlayingCard	PlayingCard	Transitive property, Inverse property (of has_greater_value_than)
need	BoardGame	OR[MagicCard Dice Pawn Paper Pencil]	Subproperty (of require)

2.2.2. Свойства на данните

Property	*Domain*	*Range*	*Characteristics*
has_duration	Game	int	-
is_professional	Game	bool	Functional property
is_amateur	Game	bool	Functional property, Not(is_professional)
has_to_remain_silent	Spectator	bool	Functional property
is_custom_board	Board	bool	Functional property
is_normal_card	Card	bool	Functional property
is_custom_card	Card	bool	Functional property, Not(is_normal_card)
has_sides	Dice	int	Functional property
has_number	Dice	int	Functional property
is_blank	Paper	bool	Functional property
has_color	Pencil	str	-
has_players_number	MultiPlayerGame	int	-
has_team	MultiPlayerGame	bool	Functional property
has_opponent	MultiPlayerGame	bool	Functional property
has_referee	ProfessionalSportGame	bool	Functional property
has_type	MagicCard	str	Functional property
has_effect	MagicCard	str	-

2.3. Индивиди

<table>
<tr><th colspan="2"><i>Тенис</i></th></tr>
<tr><td><b>tennis_ball</b> → Ball</td><td><b>tennis_court_clay</b> → Field</td></tr>
<tr><td><b>tennis_court_hard</b> → Field</td><td><b>tennis_racket</b> → Equipment</td></tr>
<tr><td colspan="2"><b>tennis</b> → [AND ProfessionalSportGame
[FILLS :require [AND tennis_ball
tennis_racket]]
[FILLS :played_on [AND tennis_court_clay
tennis_court_hard]]]</td></tr>
</table>

<table>
<tr><th colspan="3"><i>Голф</i></th></tr>
<tr><td><b>green</b> → Lawn</td><td><b>golf_ball</b> → Ball</td><td><b>golf_stick</b> → SportEquipment</td></tr>
<tr><td colspan="3"><b>golf</b> → [AND ProfessionalBallGameOnLawn
IndividualSportGame
[FILLS :require [AND golf_ball
golf_stick]]
[FILLS :played_on green]]</td></tr>
</table>

<table>
<tr><th><i>Футбол</i></th></tr>
<tr><td><b>football</b> → [AND ProfessionalBallGameOnLawn
TeamSportGame
[FILLS :has_referee True]
[FILLS :has_duration 90]]</td></tr>
</table>

<table>
<tr><th colspan="2"><i>Hardstone</i></th></tr>
<tr><td><b>desk</b> → Table</td><td><b>laptop</b> → Computer</td></tr>
<tr><td colspan="2"><b>hardstone</b> → [AND VideoGame
[FILLS :require [AND laptop
desk]]]</td></tr>
</table>

<table>
<tr><th><i>Стани богат</i></th></tr>
<tr><td><b>who_wants_to_be_a_millionaire</b> → [AND KnowledgeGame
SinglePlayerGame]</td></tr>
</table>

Tic-Tac-Toe
paper → [AND Paper [FILLS :is_blank True]]
pencil → [AND Pencil [FILLS :has_color [AND “red” “blue”]]]
tic_tac_toe → [AND PaperAndPencilGame [FILLS :require [AND paper pencil]] [FILLS :is_amateur True]]

Шах			
pawn → Pawn	**knight** → Pawn	**bishop** → Pawn	**rook** → Pawn
queen → Pawn	**king** → Pawn	**chess_board** → Board	
chess → [AND BoardGame MultiPlayerGame [FILLS :played_on chess_board] [FILLS :need [AND pawn knight bishop rook queen king]] [FILLS :has_opponent True] [FILLS :has_players_number 2]]			

Go
table → Table
go_board → [AND Board [FILLS :is_custom_board True]]
go → [AND IntelectualGame LogicalGame StrategyGame [FILLS :has_players_number 2] [FILLS :has_opponent True] [FILLS :require table] [FILLS :played_on go_board]]

Катан
Catan → [AND BoardGame [FILLS :require MagicCard] [FILLS :has_team False]]

Табла

```
dice_6 → [AND Dice
                [FILLS :has_sides 6]
                [FILLS :has_number 5]]
```

```
dice_12 → [AND Dice
                 [FILLS :has_sides 12]
                 [FILLS :has_number 3]
                 [FILLS :has_lower_number_than dice_6]]
```

```
dice_8 → [AND Dice
                [FILLS :has_sides 8]
                [FILLS :has_number 7]
                [FILLS :has_greater_number_than [AND dice_6 dice_12]]]
```

```
Backgammon → [AND BoardGame
                    MultiPlayerGame
                    [FILLS :require dice_6]
                    [FILLS :has_opponent True]]
```

Бридж

```
North → [AND Player
               [FILLS :plays [AND golf football]]]
```

```
East → [AND Player
              [FILLS :plays hardstone]]
```

```
South → Player
```

```
West → Player
```

```
kibitzer → [AND Kibitzer
                 [FILLS :watches football]
                 [FILLS :has_interest_in hardstone]]
```

```
fan → [AND Fan
            [FILLS :has_favourite [AND East West]]
            [FILLS :watches tennis]]
```

```
spectator → [AND Spectator
                  [FILLS :has_interest_in [AND football chess]]]
```

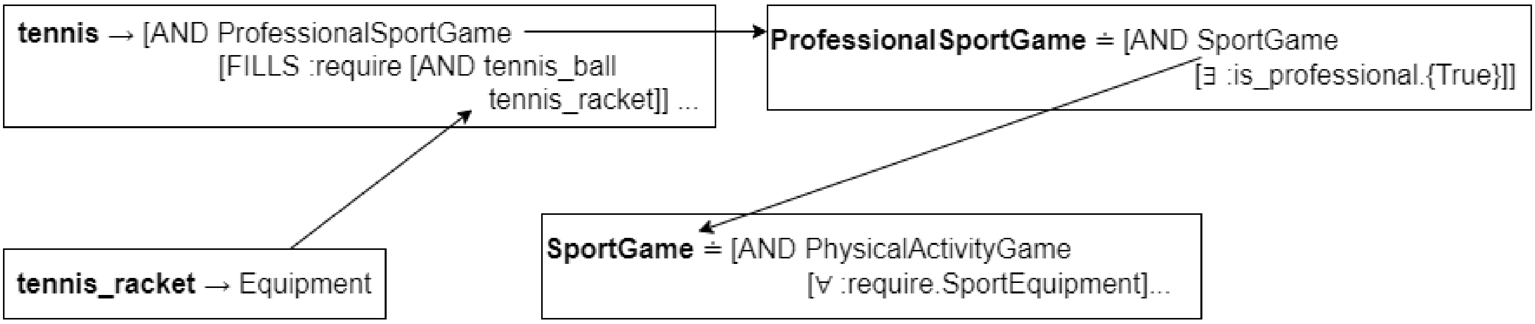
```
Two_of_Clubs → [AND PlayingCard
                     [FILLS :has_suit Clubs]
                     [FILLS :has_value Two]]

Three_of_Clubs → [AND PlayingCard
                       [FILLS :has_suit Clubs]
                       [FILLS :has_value Three]
                       [FILLS :has_greater_value_than Two_of_Clubs]]

…
```

Ace_of_Spades → [AND PlayingCard [FILLS :has_suit Spades] [FILLS :has_value Ace] [FILLS :has_greater_value_than King_of_Spades]]
Bridge → [AND CardGame [FILLS :played_by [AND North East South West]] [FILLS :has_team True] [FILLS :watched_by [AND fan kibitzer]] [FILLS :require [AND Two_of_Clubs Three_of_Clubs … Ace_of_Spades]]

3. Примери за разсъждение на базата знания

3.1. Примери за извършване на логически извод

- KB |= (tennis_racket → SportEquipment)



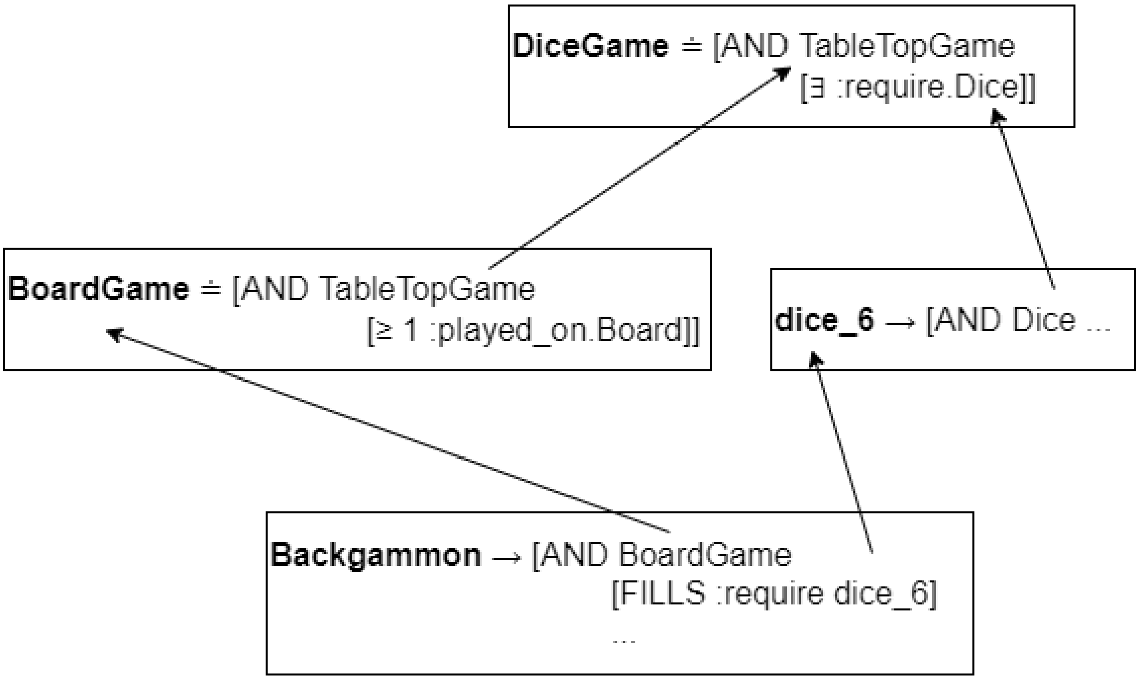
tennis → [AND ProfessionalSportGame
[FILLS :require [AND tennis_ball
tennis_racket]] ...

ProfessionalSportGame ≐ [AND SportGame
[∃ :is_professional.{True}]]

tennis_racket → Equipment

SportGame ≐ [AND PhysicalActivityGame
[∀ :require.SportEquipment]...

- KB |= (Backgammon → DiceGame)



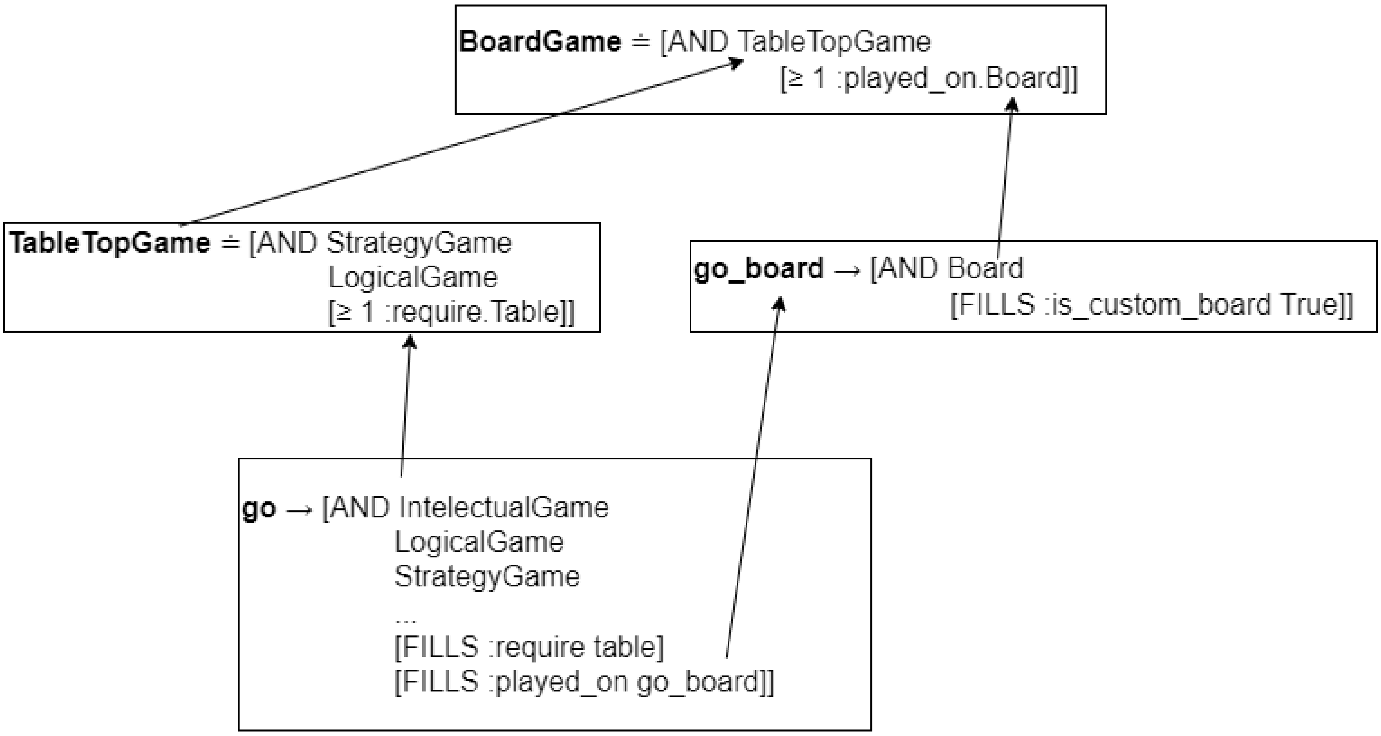
DiceGame ≐ [AND TableTopGame
[∃ :require.Dice]]

BoardGame ≐ [AND TableTopGame
[≥ 1 :played_on.Board]]

dice_6 → [AND Dice ...

Backgammon → [AND BoardGame
[FILLS :require dice_6]
...

- KB |= (go → BoardGame)



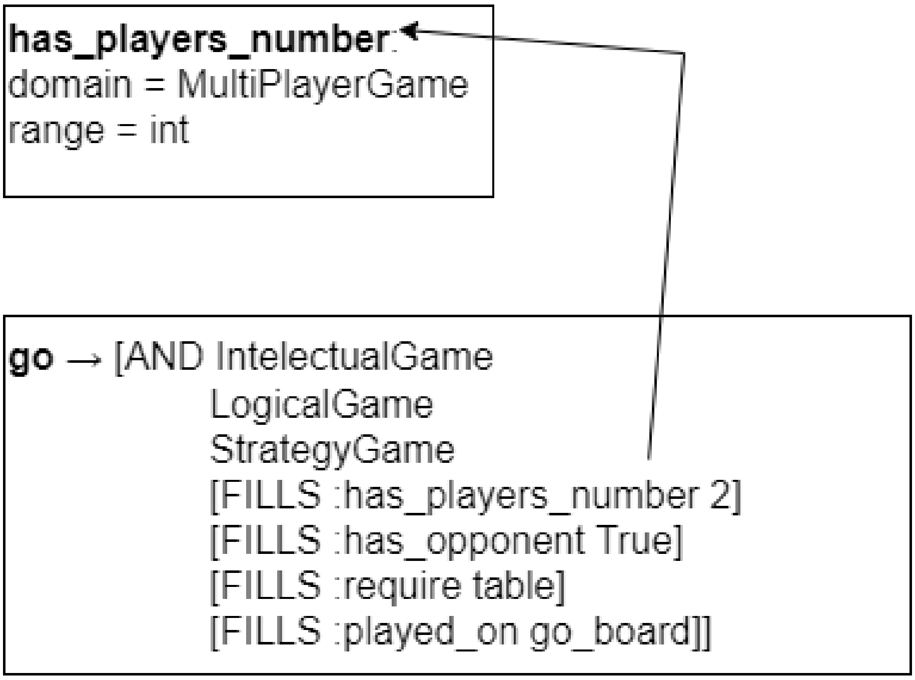
BoardGame ≐ [AND TableTopGame
[≥ 1 :played_on.Board]]

TableTopGame ≐ [AND StrategyGame
LogicalGame
[≥ 1 :require.Table]]

go_board → [AND Board
[FILLS :is_custom_board True]]

go → [AND IntelectualGame
LogicalGame
StrategyGame
...
[FILLS :require table]
[FILLS :played_on go_board]]

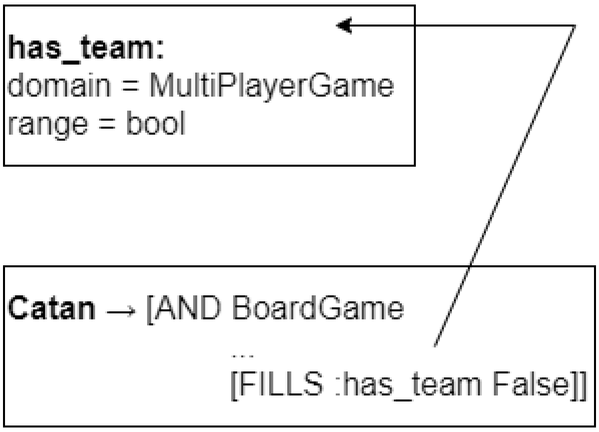
- KB |= (go → MultiPlayerGame)



has_players_number:
domain = MultiPlayerGame
range = int

go → [AND IntelectualGame
LogicalGame
StrategyGame
[FILLS :has_players_number 2]
[FILLS :has_opponent True]
[FILLS :require table]
[FILLS :played_on go_board]]

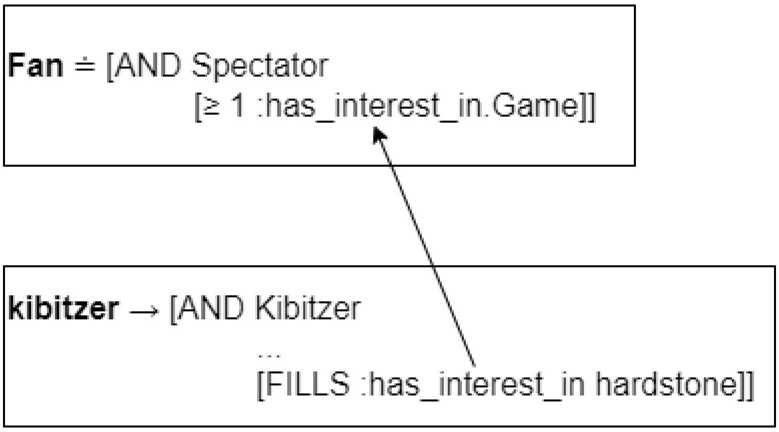
- KB |= (Catan → MultiPlayerGame)



has_team:
domain = MultiPlayerGame
range = bool

Catan → [AND BoardGame
...
[FILLS :has_team False]]

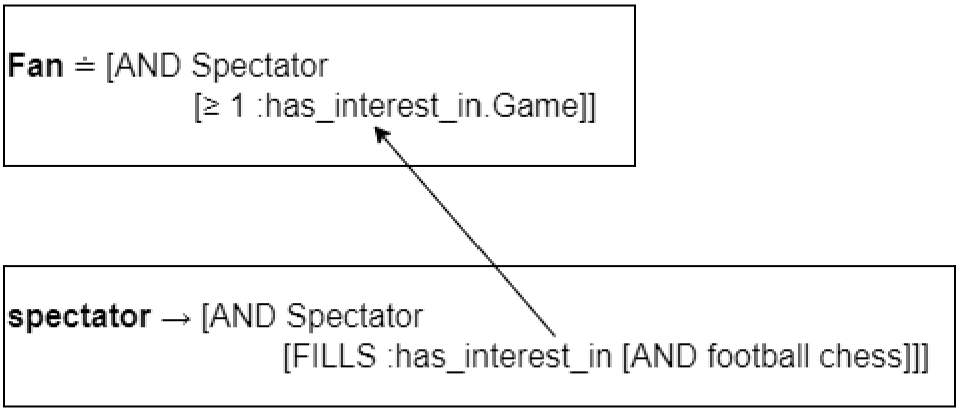
- KB |= (kibitzer → Fan)



Fan ≐ [AND Spectator
[≥ 1 :has_interest_in.Game]]

kibitzer → [AND Kibitzer
...
[FILLS :has_interest_in hardstone]]

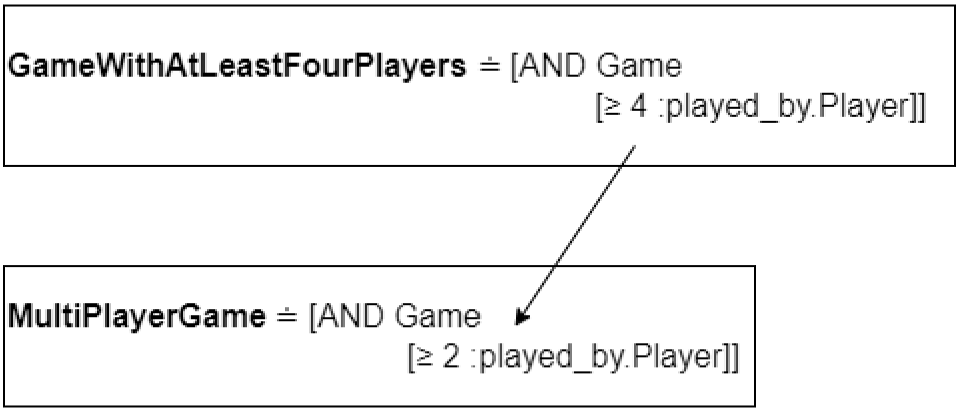
- KB |= (spectator → Fan)



Fan ≐ [AND Spectator
[≥ 1 :has_interest_in.Game]]

spectator → [AND Spectator
[FILLS :has_interest_in [AND football chess]]]

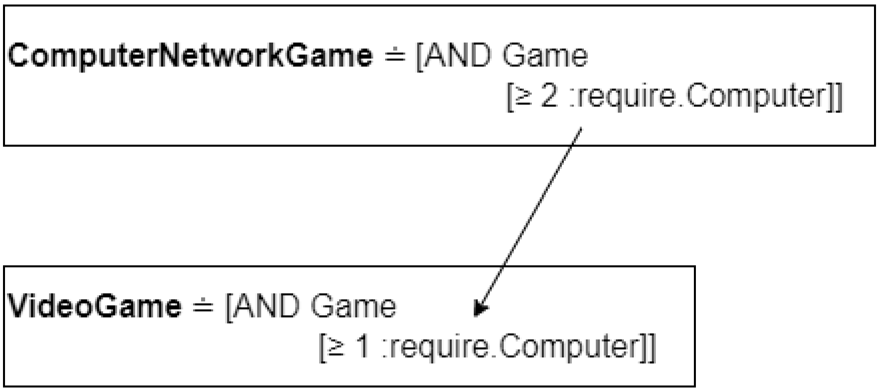
- GameWithAtLeastFourPlayers ≐ [AND Game
  [≥ 4 :played_by.Player]] ⊑
  ⊑ MultiPlayerGame



GameWithAtLeastFourPlayers ≐ [AND Game
[≥ 4 :played_by.Player]]

MultiPlayerGame ≐ [AND Game
[≥ 2 :played_by.Player]]

- ComputerNetworkGame ≐ [AND Game
  [≥ 2 :require.Computer]] ⊑
  ⊑ VideoGame

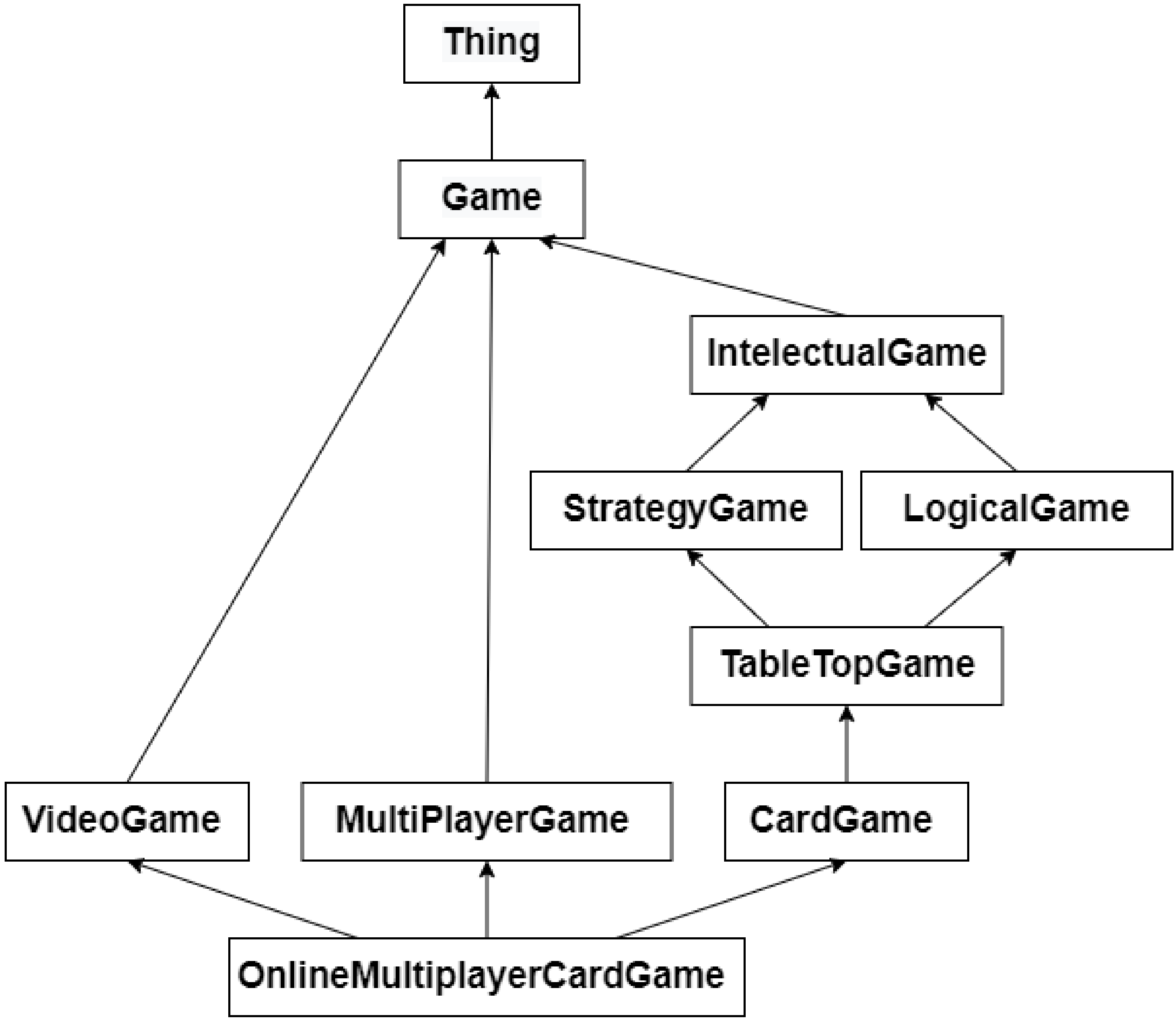


ComputerNetworkGame ≐ [AND Game
[≥ 2 :require.Computer]]

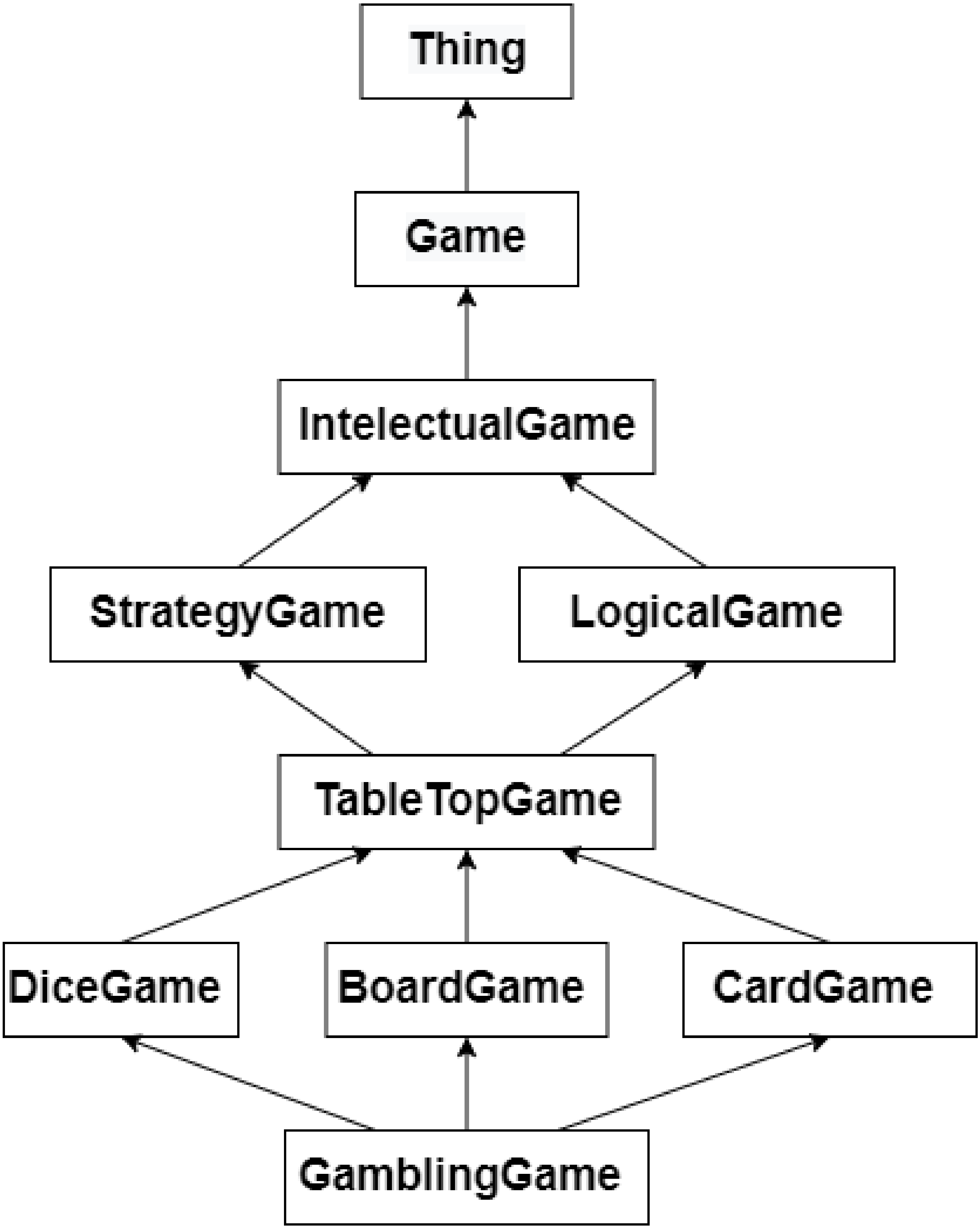
VideoGame ≐ [AND Game
[≥ 1 :require.Computer]]

3.2. Примери за извършване на класификация

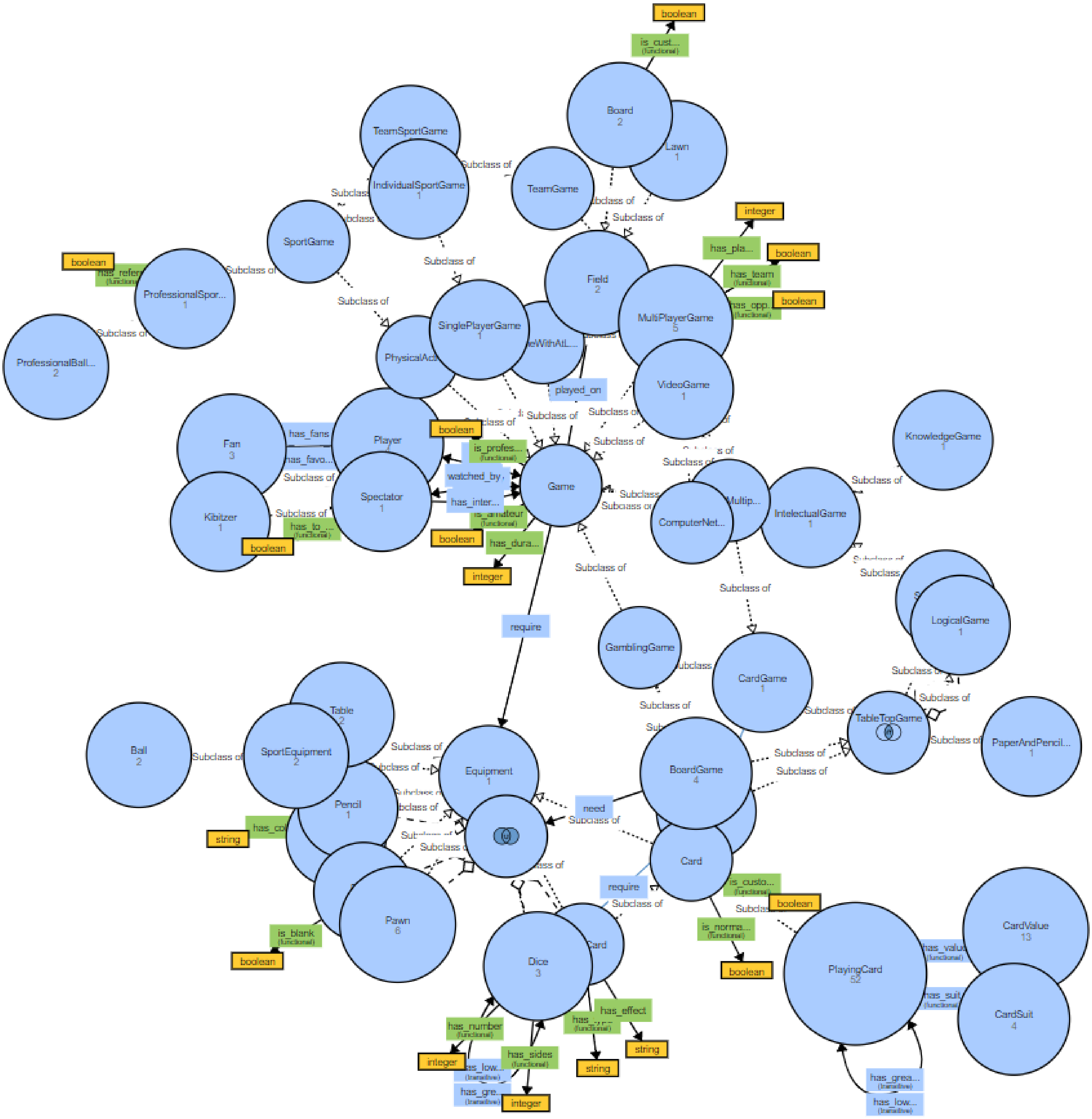
- **OnlineMultiplayerCardGame** ≐ [AND Game
  [≥ 1 :require.Computer]
  [≥ 2 :played_by.Player]
  [∃ :require.Card]]



- **GamblingGame** ≐ [AND Game
  [∃ :require.PlayingCard]
  [= 2 :need.Dice]
  [≥ 1 :require.Table]]



4. Схема на онтологията



5. Бъдещо развитие

- Добавяне на още екземпляри към онтологията с цел увеличаване на знанието за игрите.
- Разширяване на онтологията с още концепти и свойства с цел по-добро описание на предметната област.
- Сливане на онтологията с други онтологии, например такава за видеоигри.
- Публикуване на кода на онтологията онлайн с цел създаване на проект с отворен код, който може да се ползва и допълва от други любители на игрите, разработчици и инженери на знания.

6. Използвани технологии

- Проектът е имплементиран чрез Python и Owlready2.
- За визуализация на схемата на онтологията е използван онлайн инструментът WebVOWL (http://vowl.visualdataweb.org/webvowl.html).

7. Библиография

- Статия за игрите в Уикипедия: https://en.wikipedia.org/wiki/Game
- Документация на Owlready2: https://owlready2.readthedocs.io/en/v0.36/intro.html
- *The great table of Description Logics and formal ontology notations*, Jean-Baptiste Lamy, June 22, 2018: https://hal.archives-ouvertes.fr/hal-01849822/document
- *Knowledge Representation and Reasoning*, Ronald J. Brachman and Hector J. Levesque