arKItect

Role Playing Game Modeling with arKItect

Bojan Arnaudovski, Department of Computer Science
University of Antwerpen, January 2014

[redacted]

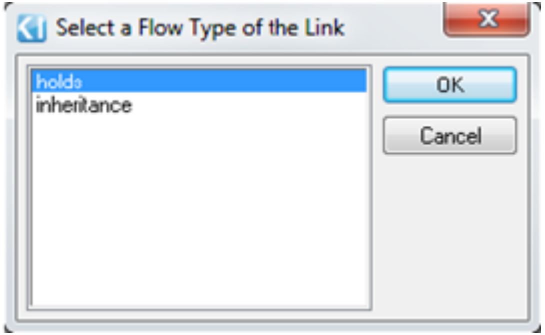
- Introduction
- Abstract Syntax
- Concrete Syntax
- Constraints and Operational Semantics
- References

Introduction

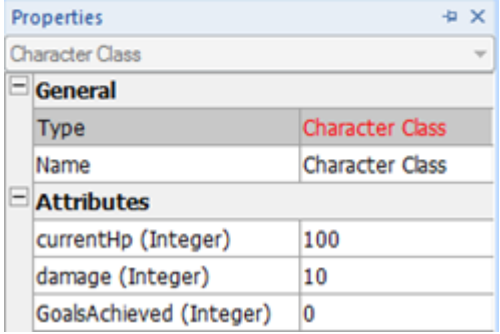
- Representation and Design of complex, hierarchical systems
- First version in 2007 by Knowledge Inside (Samuel Boutin, Joe Matta and Konstantin Smolin)
- Renault, Cheuvreux, Bouygues, PSA, Ansaldo STS, EADS, Thalès (satellites), EDF and Schneider Electric
- meta-modeling and operational semantics for RPG
- arKItect Designer (meta-modeling)

Abstract Syntax

- arKItect objects,attributes and data flows
- Define rules and filters
- No constraints, cardinality and action field
- triggering an event (Python scripts)

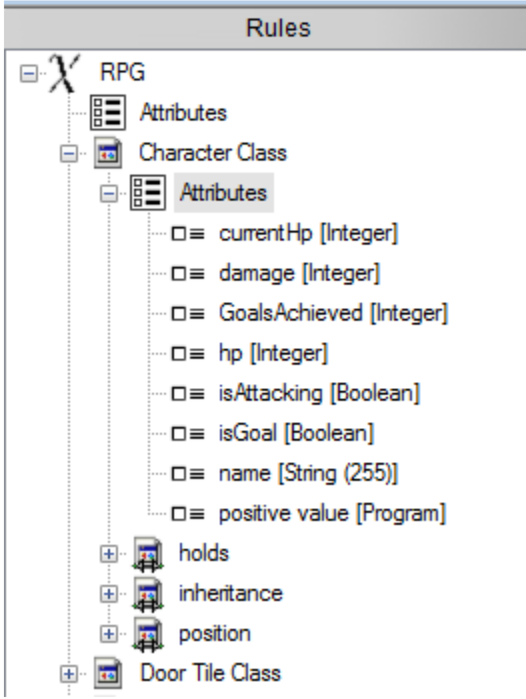

Select a Flow Type of the Link

holds
inheritance

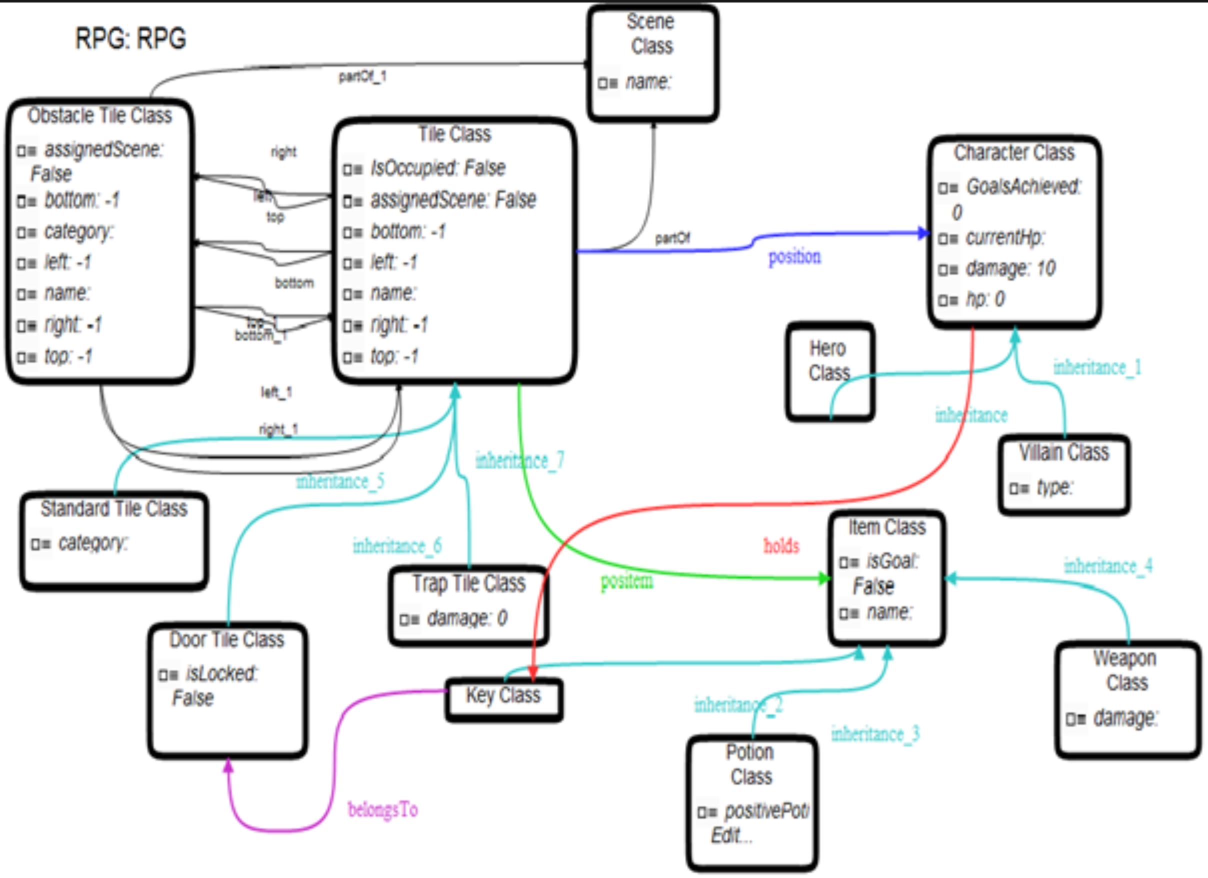

Properties

Character Class

General	
Type	Character Class
Name	Character Class
Attributes	
currentHp (Integer)	100
damage (Integer)	10
GoalsAchieved (Integer)	0

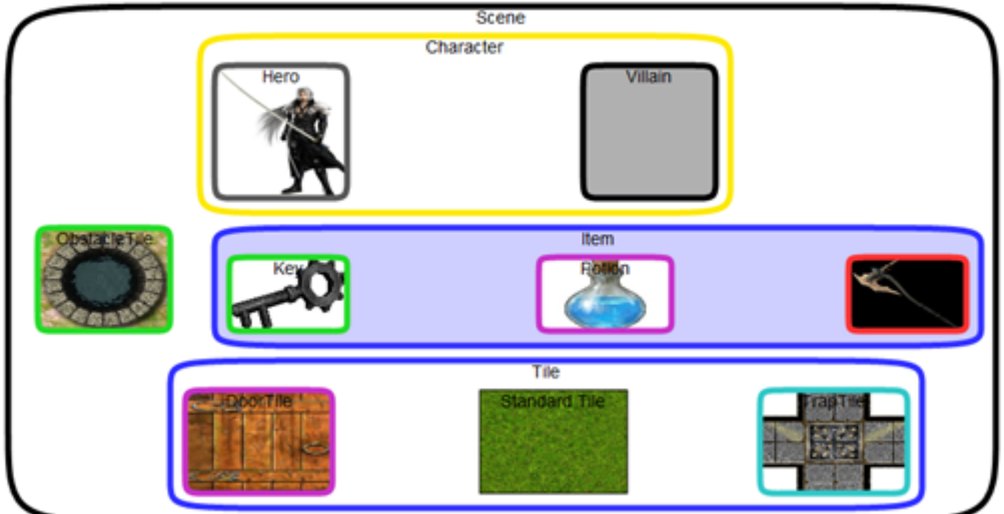

Rules

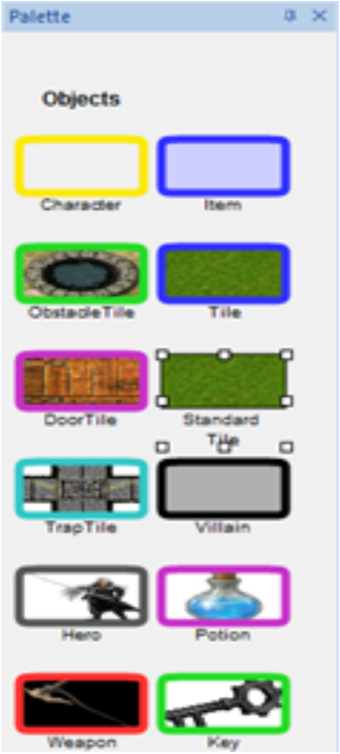
- RPG
  - Attributes
  - Character Class
    - Attributes
      - currentHp [Integer]
      - damage [Integer]
      - GoalsAchieved [Integer]
      - hp [Integer]
      - isAttacking [Boolean]
      - isGoal [Boolean]
      - name [String (255)]
      - positive value [Program]
    - holds
    - inheritance
    - position
  - Door Tile Class


RPG: RPG
Scene Class
name:
partOf_1
Obstacle Tile Class
assignedScene: False
bottom: -1
category:
left: -1
name:
right: -1
top: -1
Tile Class
IsOccupied: False
assignedScene: False
bottom: -1
left: -1
name:
right: -1
top: -1
right
left
top
bottom
top_1
bottom_1
left_1
right_1
partOf
position
Character Class
GoalsAchieved: 0
currentHp:
damage: 10
hp: 0
Hero Class
inheritance_1
inheritance
Villain Class
type:
inheritance_7
inheritance_5
Standard Tile Class
category:
inheritance_6
Trap Tile Class
damage: 0
holds
positem
Item Class
isGoal: False
name:
inheritance_4
Weapon Class
damage:
Door Tile Class
isLocked: False
Key Class
belongsTo
inheritance_2
inheritance_3
Potion Class
positivePoti
Edit...

Concrete Syntax

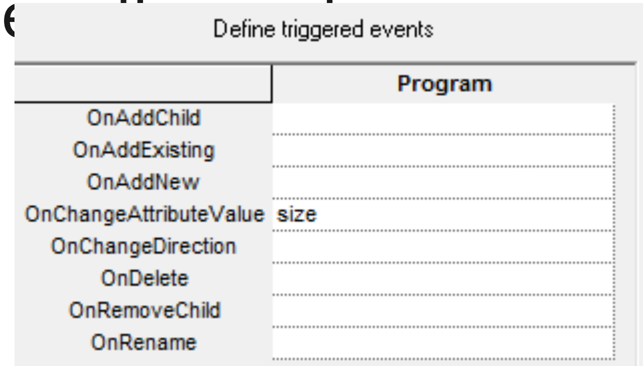
- Additional rules and filters
- Restrictions when creating objects
- Overview of the RPG model





Constraints and Operational Semantics

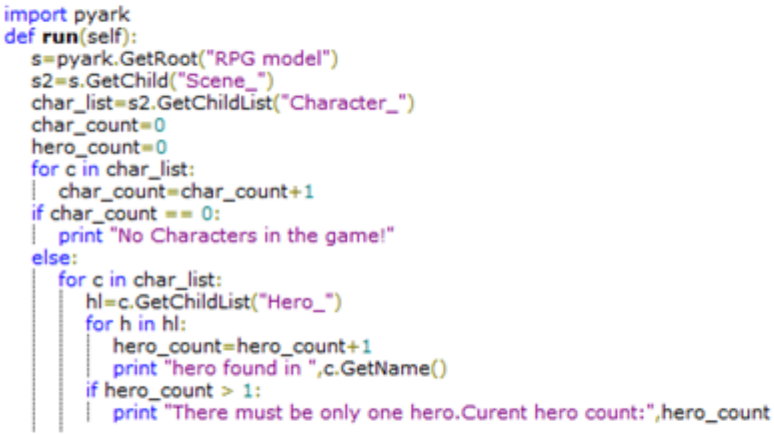
- No support for operational semantics and transformation
- “Program” attribute - Python Scripts
- Executed manually or triggered by an event
- Constraints and Op.semantics done
- Output similar to metaDepth



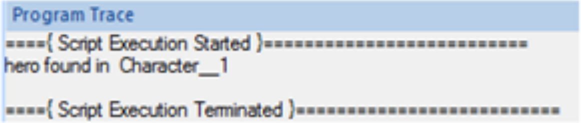
Define triggered events

	Program
OnAddChild	
OnAddExisting	
OnAddNew	
OnChangeAttributeValue	size
OnChangeDirection	
OnDelete	
OnRemoveChild	
OnRename	

Constraints and Operational Semantics (2)

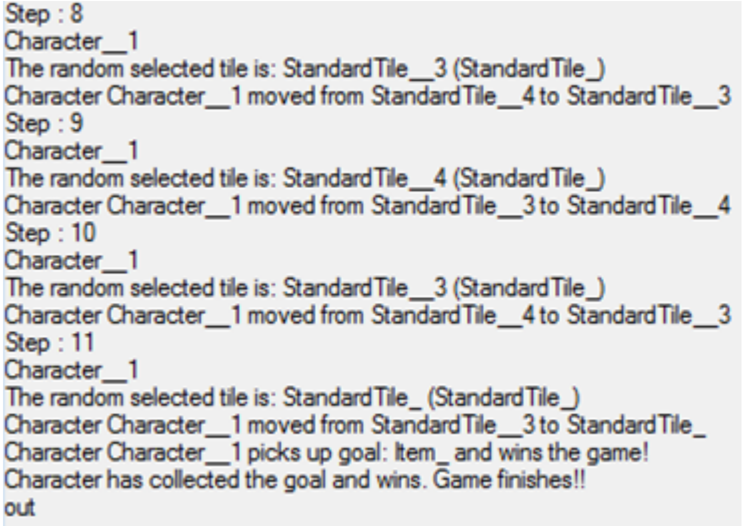


```
import pyark
def run(self):
    s=pyark.GetRoot("RPG model")
    s2=s.GetChild("Scene_")
    char_list=s2.GetChildList("Character_")
    char_count=0
    hero_count=0
    for c in char_list:
        char_count=char_count+1
    if char_count == 0:
        print "No Characters in the game!"
    else:
        for c in char_list:
            hl=c.GetChildList("Hero_")
            for h in hl:
                hero_count=hero_count+1
                print "hero found in ",c.GetName()
            if hero_count > 1:
                print "There must be only one hero.Curent hero count:",hero_count
```



```
===={ Script Execution Started }===========================
hero found in  Character__1

===={ Script Execution Terminated }===========================
```



```
Step : 8
Character__1
The random selected tile is: StandardTile__3 (StandardTile_)
Character Character__1 moved from StandardTile__4 to StandardTile__3
Step : 9
Character__1
The random selected tile is: StandardTile__4 (StandardTile_)
Character Character__1 moved from StandardTile__3 to StandardTile__4
Step : 10
Character__1
The random selected tile is: StandardTile__3 (StandardTile_)
Character Character__1 moved from StandardTile__4 to StandardTile__3
Step : 11
Character__1
The random selected tile is: StandardTile_ (StandardTile_)
Character Character__1 moved from StandardTile__3 to StandardTile_
Character Character__1 picks up goal: Item_ and wins the game!
Character has collected the goal and wins. Game finishes!!
out
```

Constraints and Operational Semantics (3)

- “hero_count”
- “item_tile_check”
- “positiveX” and “positiveY”
- “right_num_tiles”
- “size”
- “one item”
- “get all Tiles”
- “update connection”
- “same type”
- “hero or villain”
- “set position”
- “move” and “simulate”

References

- arKItect home page, *https://support.k-inside.com/display/ARKI22/arKItect+2.2.x+documentation+home*
- arKItect documentation *http://www.k-inside.com/web/produits-et-services/produits/arkitect-designer/*