

23<sup>rd</sup> International Conference on  
Model-Driven Engineering Languages and Systems

MODELS 20

16–23 October 2020



# A Model-driven Alternative to Programming in Blocks using Rule-based Transformations



BALL STATE  
UNIVERSITY

*Huseyin Ergin*

*Wenjun Shi*

THE UNIVERSITY OF  
ALABAMA<sup>®</sup>

*Herart Domingus Nurue*

*Jeff Gray*



Learn computer science.  
Change the world.

Start learning

Watch the video



Are you teaching in a remote or socially-distanced classroom this semester? View our resources.

Get started

40%

of U.S. students have accounts on Code.org

20M

of our students are young women

106M

projects created on Code.org

1M

teachers use Code.org

50

All 50 states support computer science



Learn at Home  
These resources make it easy



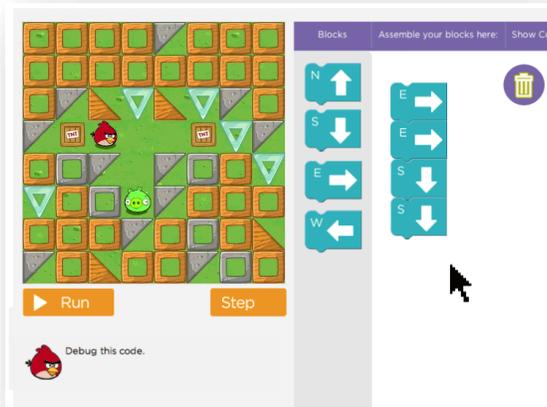
Students  
Explore our courses



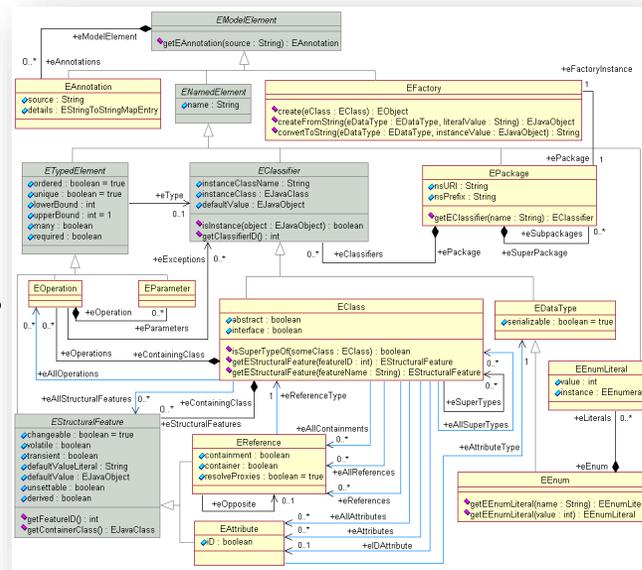
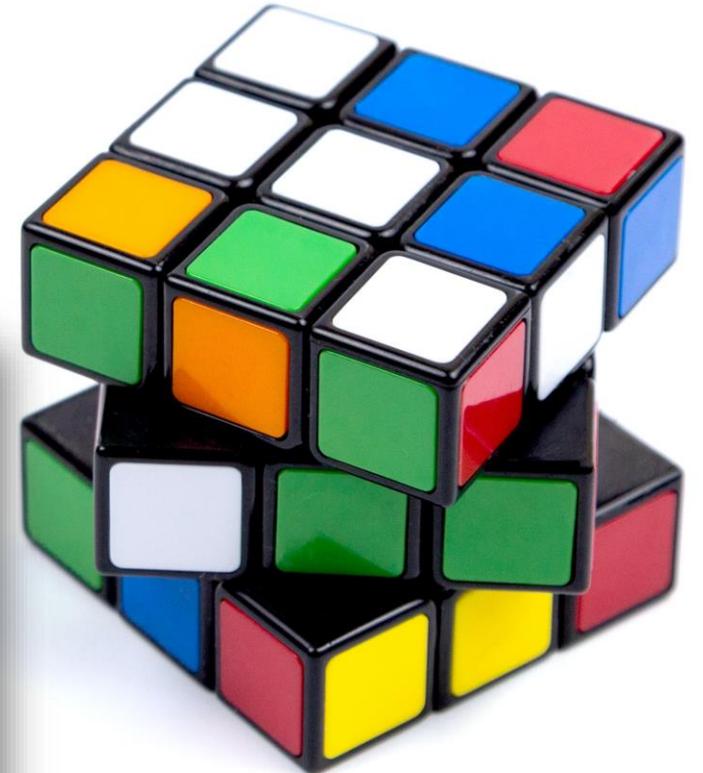
Educators  
Teach your students



Get involved  
Support diversity in computing

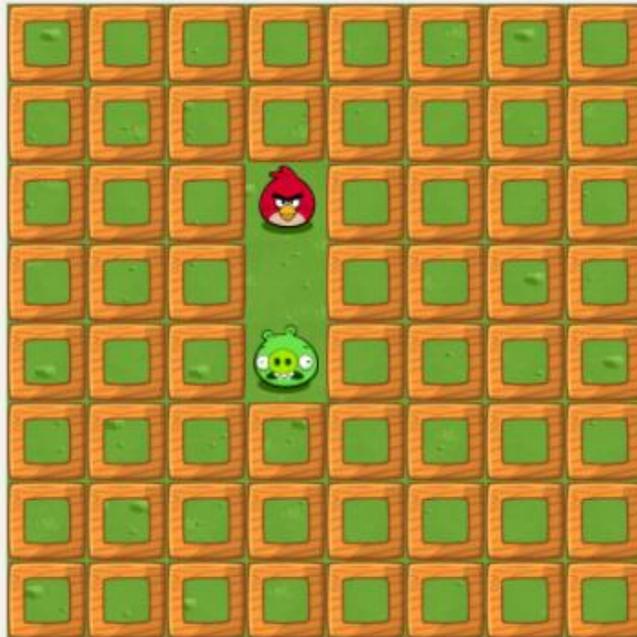


Block-based CS Education



MDE Education?

**We have a problem in MDE!**



▶ Run

Need help? See these videos and hints



Maze Intro

Instructions



Can you help me to catch the naughty pig? Stack a couple of "move forward" blocks together and press "Run" to help me get there.



Blocks

Workspace: 2 / 3 blocks

↺ Start Over

↔ Show Code

move forward

turn left ↺

turn right ↻

when run  
move forward

repeat 5 times  
do

repeat until   
do

if path to the left ↺  
do

# Model-driven Engineering



C O D E

Classic Maze 1 I finished! Sign in ?

Instructions

Can you help me to catch the naughty pig? Stack a couple of "move forward" blocks together and press "Run" to help me get there.

Blocks Workspace: 2 / 3 blocks Start Over Show Code

move forward

turn left ↶

turn right ↷

when run

move forward

Run

Need help? See these videos and hints

Maze Intro

# • Similarity to BBPLs

The screenshot shows the 'Classic Maze' challenge on Code.org. The interface is divided into several sections:

- Header:** 'Classic Maze' with a progress indicator (1 out of 10) and a 'I finished!' message. There are also 'Sign in', a help icon, and a menu icon.
- Maze:** A 10x10 grid of green squares with orange walls. A red Angry Bird character is at the top center, and a green pig character is at the bottom center. A path of green squares leads from the bird to the pig.
- Instructions:** A text box with an Angry Bird icon that says: "Can you help me to catch the naughty pig? Stack a couple of 'move forward' blocks together and press 'Run' to help me get there."
- Blocks:** A workspace with 2/3 blocks. The blocks are: a yellow 'when run' block, a blue 'move forward' block, and a blue 'turn left' block. There are also 'Start Over' and 'Show Code' buttons.
- Run Button:** A large orange button with a play icon and the text 'Run'.
- Need help?:** A section with the text 'See these videos and hints' and a video thumbnail titled 'Maze Intro' showing a woman.

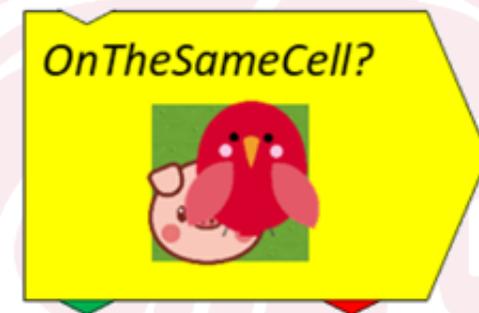
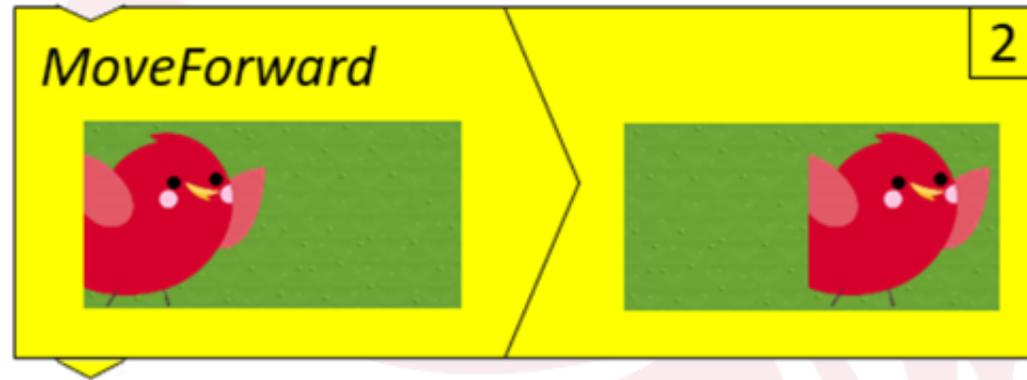
**Key characteristics of a possible alternative MDE environment: #1**

# • Visual

The screenshot shows the Code.org 'Classic Maze' game interface. At the top, there's a teal header with 'C O' and 'D E' icons, the title 'Classic Maze', a progress indicator with 15 circles (the first is filled), and the text 'I finished!'. A 'Sign in' button and a help icon are on the right. The main area is divided into three sections: 1. A 15x15 grid maze with a red bird character at the start and a green pig character at the end. 2. An 'Instructions' panel with a red bird icon and the text: 'Can you help me to catch the naughty pig? Stack a couple of "move forward" blocks together and press "Run" to help me get there.' 3. A 'Workspace' area with 'Blocks' on the left and 'Workspace: 2 / 3 blocks' on the right. The blocks include 'move forward', 'turn left', and 'turn right' in the left column, and a 'when run' block with a 'move forward' block in the right column. A 'Run' button is below the maze. At the bottom left, there's a 'Need help?' section with a video thumbnail titled 'Maze Intro'.

**Key characteristics of a possible alternative MDE environment: #2**

# • Declarative model transformation



***Key characteristics of a possible alternative MDE environment: #3***

- **Easy for first-timers/early-learners**

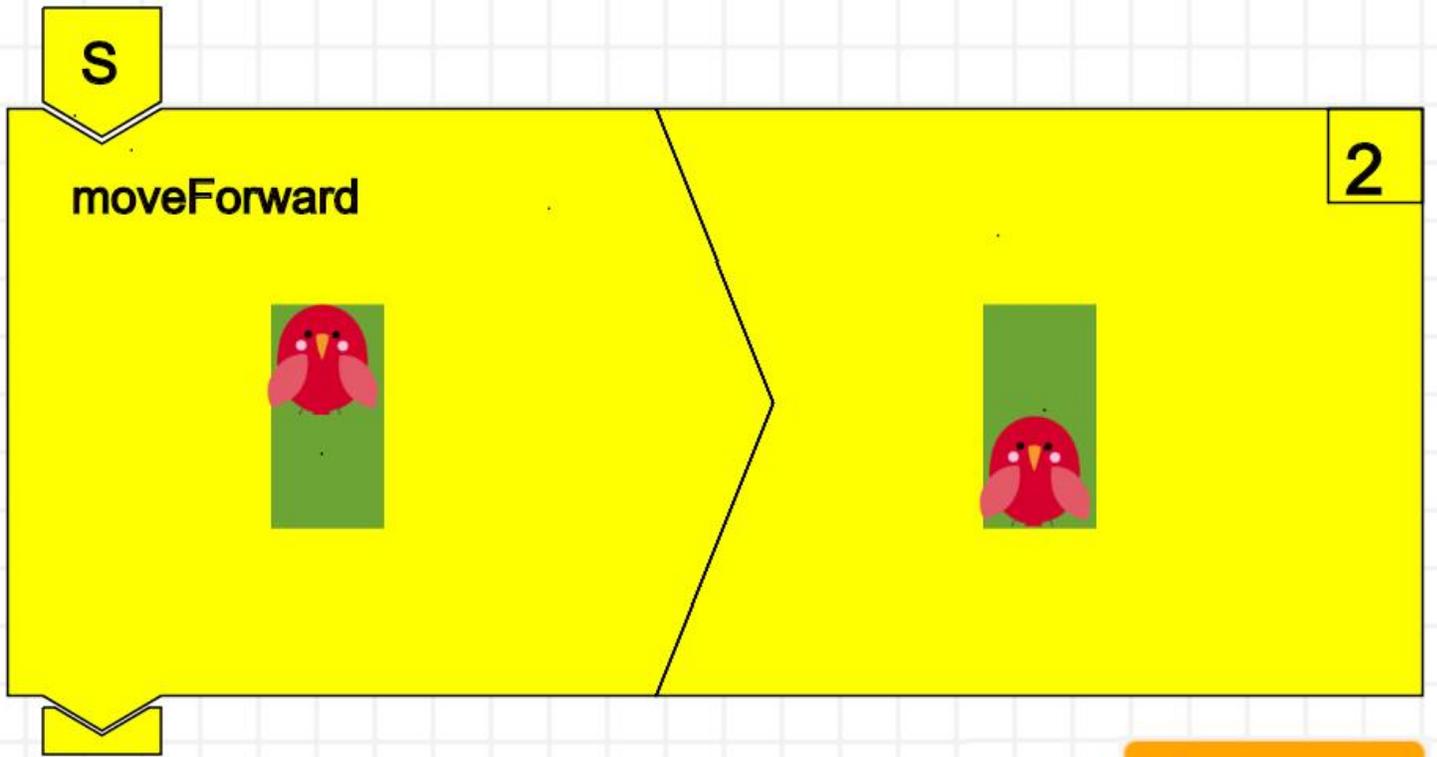
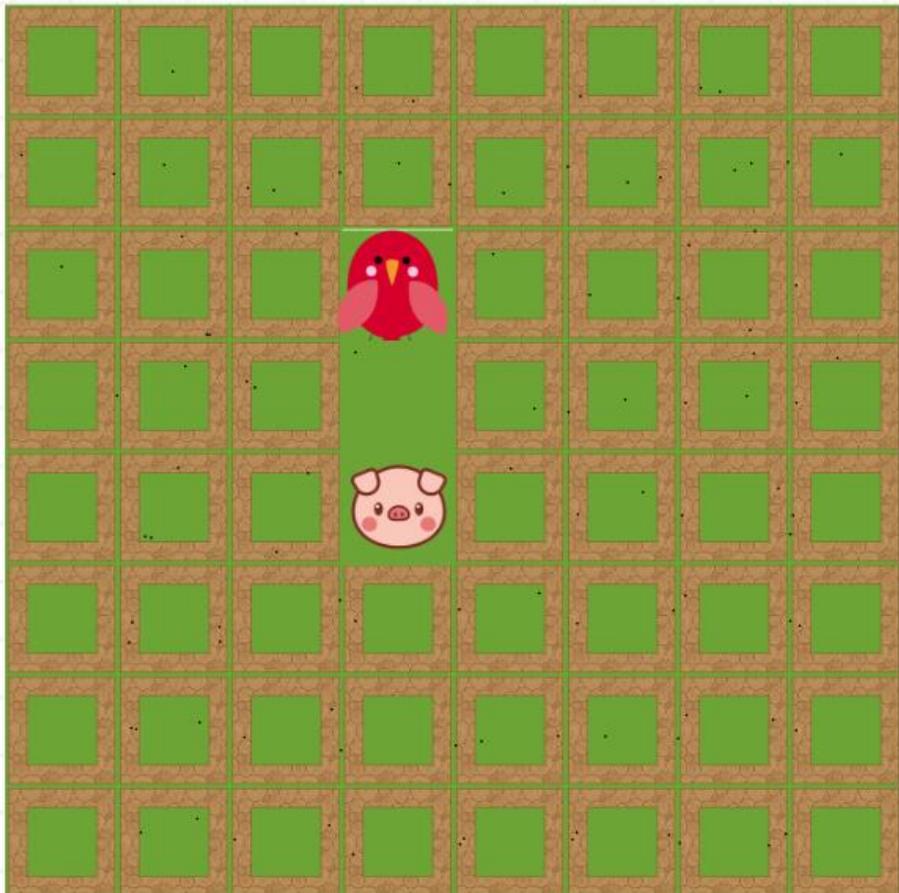
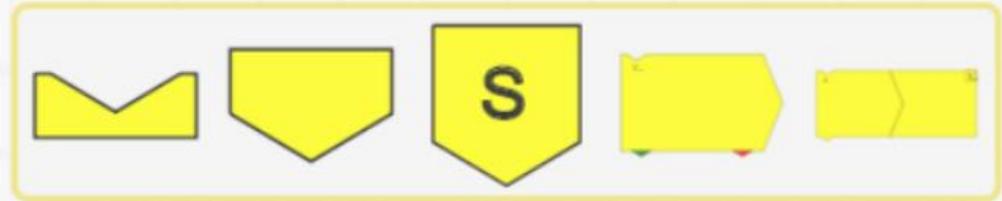


***Key characteristics of a possible alternative MDE environment: #4***

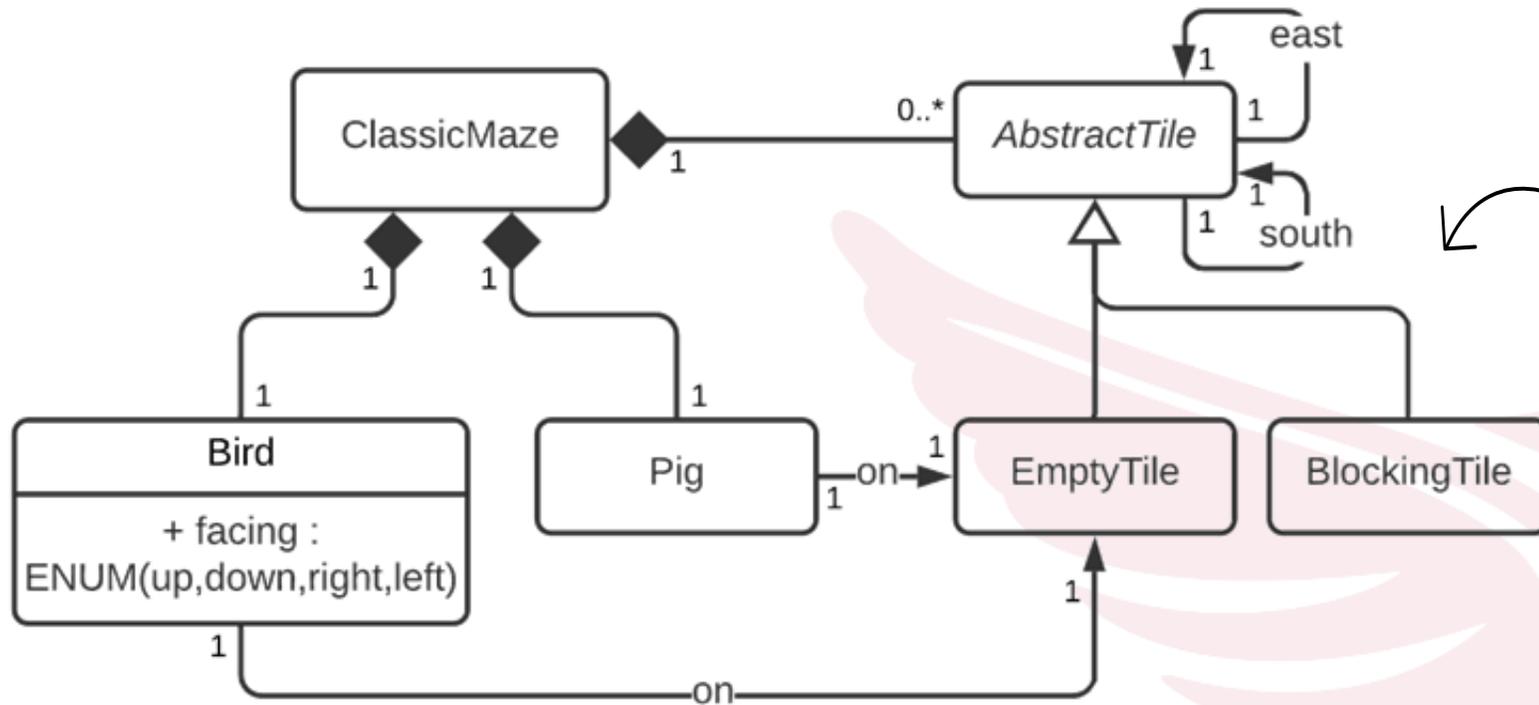
- **Web-based**



***Key characteristics of a possible alternative MDE environment: #5***

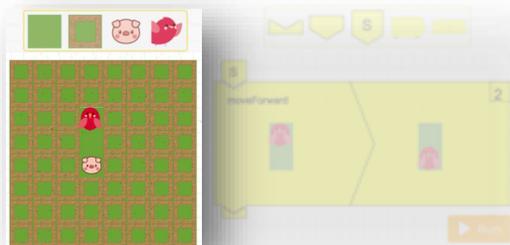


***A new block-based programming environment using MDE***



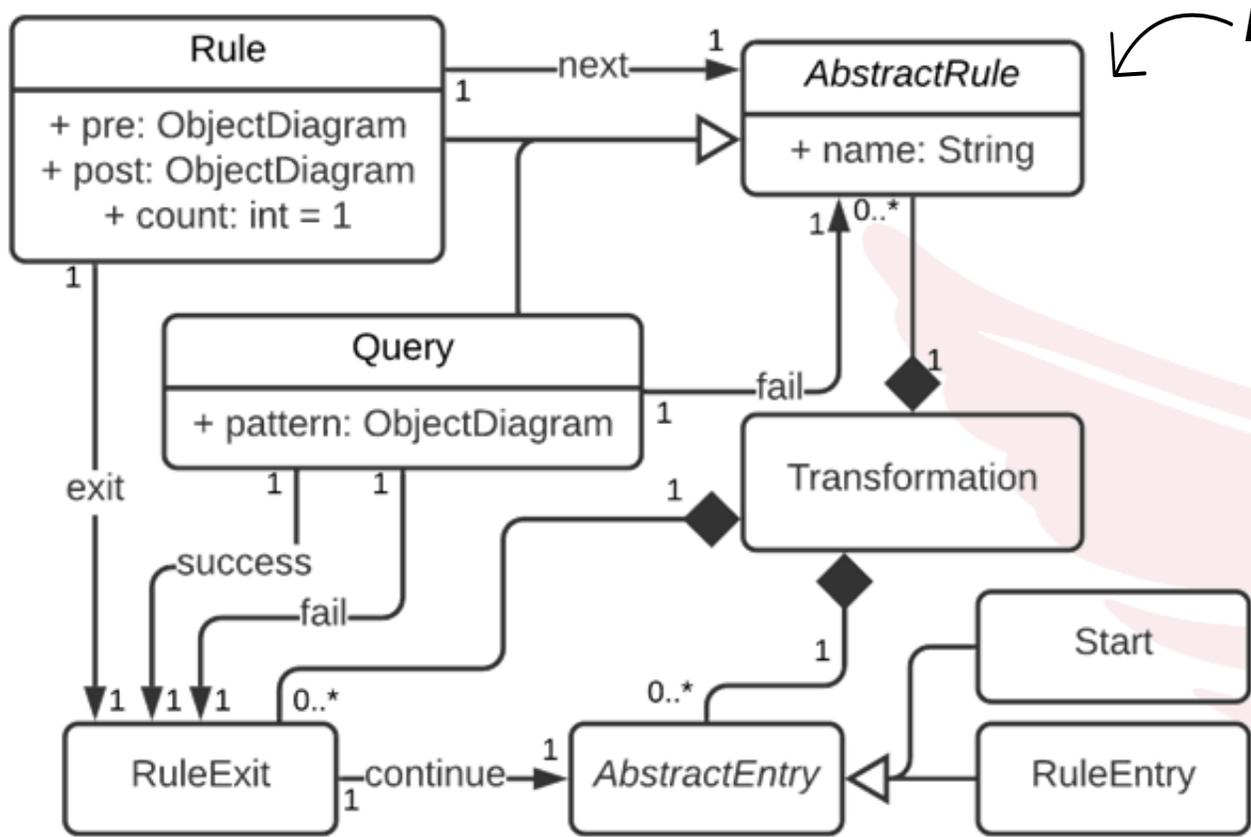
**Metamodel of the Bird Maze World**

**Concrete Graphical Syntax**

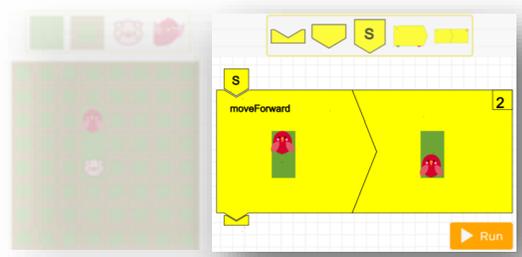
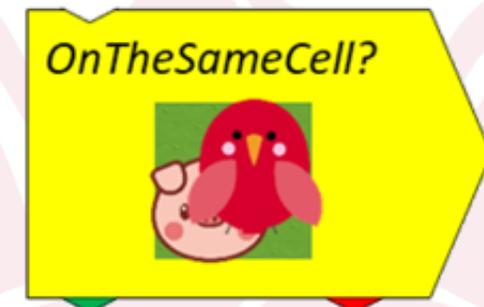
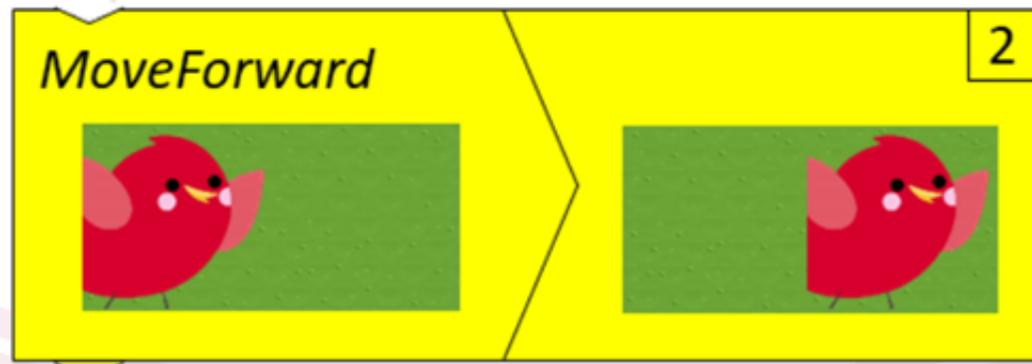


Bird	Pig	EmptyTile	BlockingTile
			

***A new block-based programming environment using MDE***



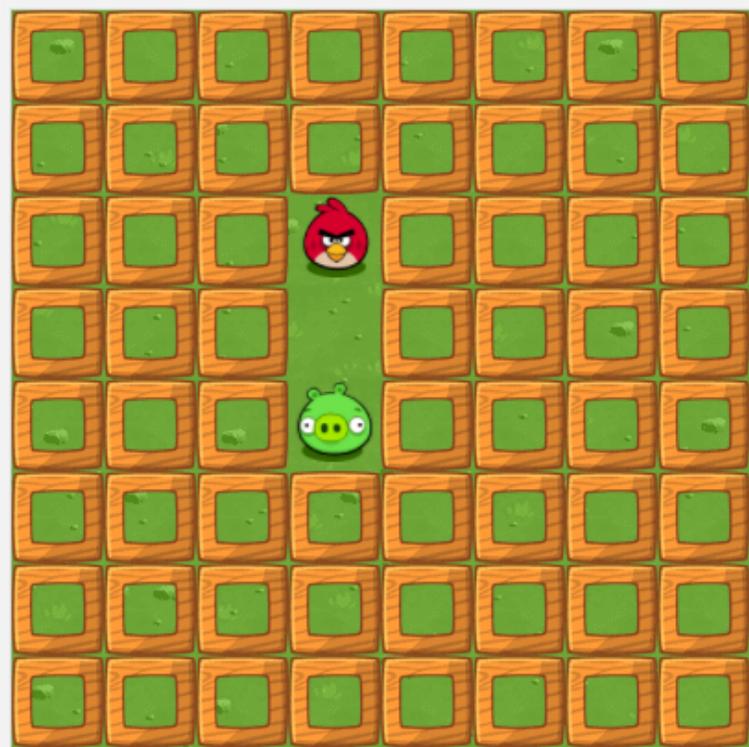
**Metamodel of the Transformation System**



Start	RuleEntry	RuleExit	Usage

**Concrete Graphical Syntax**

***A new block-based programming environment using MDE***



▶ Run

### Instructions



Can you help me to catch the

### Blocks

move forward

turn left ↶

turn right ↷

when run

move forward

move forward

S

*MoveForwardV2*



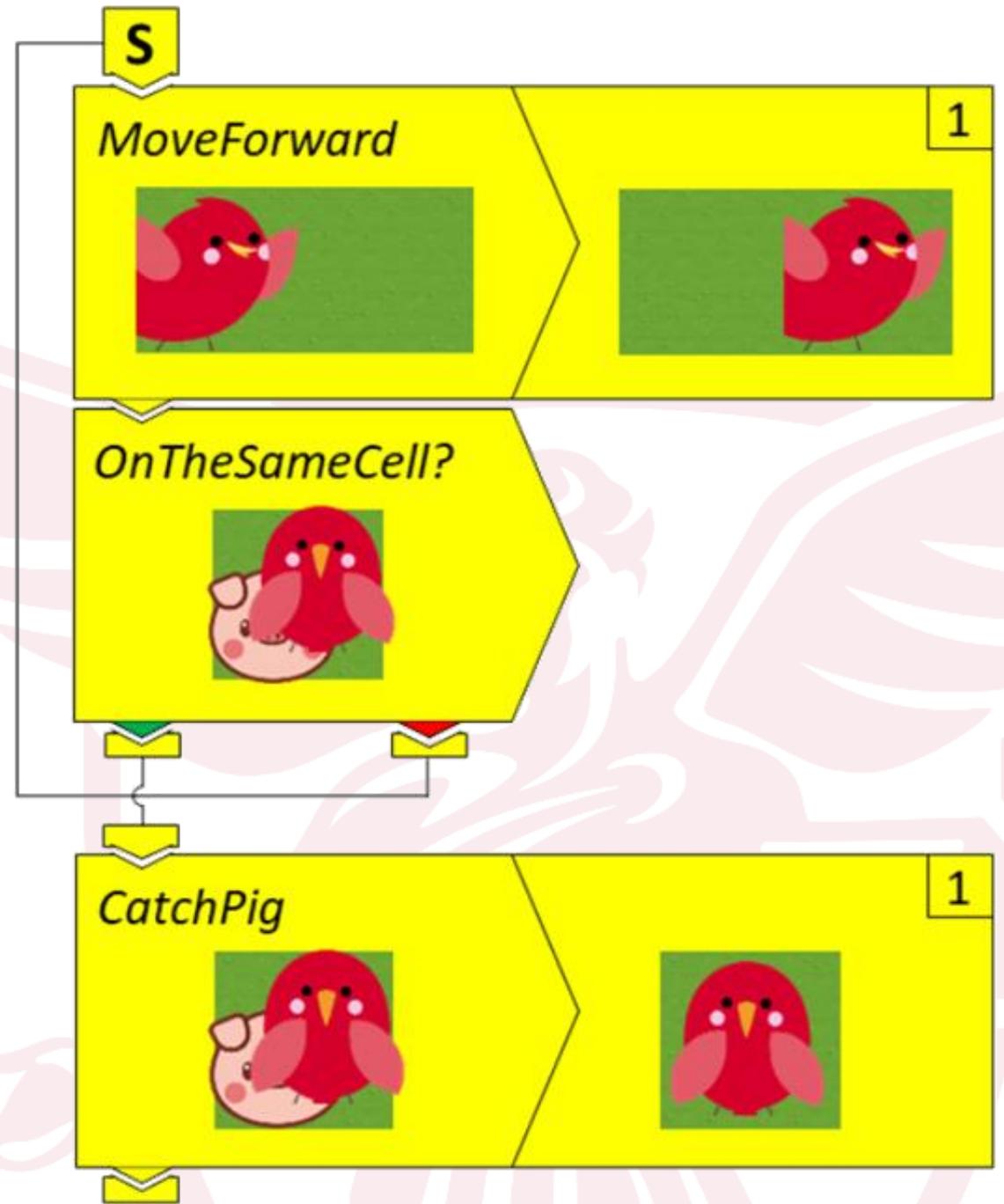
2

Traditional imperative approach

Our new vision

**Maze 1**

**For more examples, please see:**  
<https://bit.ly/mde-blocks>



***Maze 1 solution with built-in rules modeled***



# CS Education

MDE

**A research vision**

# • Empirical and evidence-based pedagogy



**A research vision #1**

Image credit: <http://blogs.egusd.net/mrsjones/2015/08/20/code-org/>

# • Minimizing the footprint of the MDE tools



***A research vision #2***

# • Student customization of worlds (new metamodels)

Instructions

Help the harvester pick th

Instructions

Free Play: This is Laurel t

Instructions

**Make a prediction:** Here's a sprite that looks like a  
What do you think will happen when you click on

Enter your answer here

when run

set background to sci-fi

make new sprite at (200, 100)

when clicked

sprite begins patrolling up and down

Run

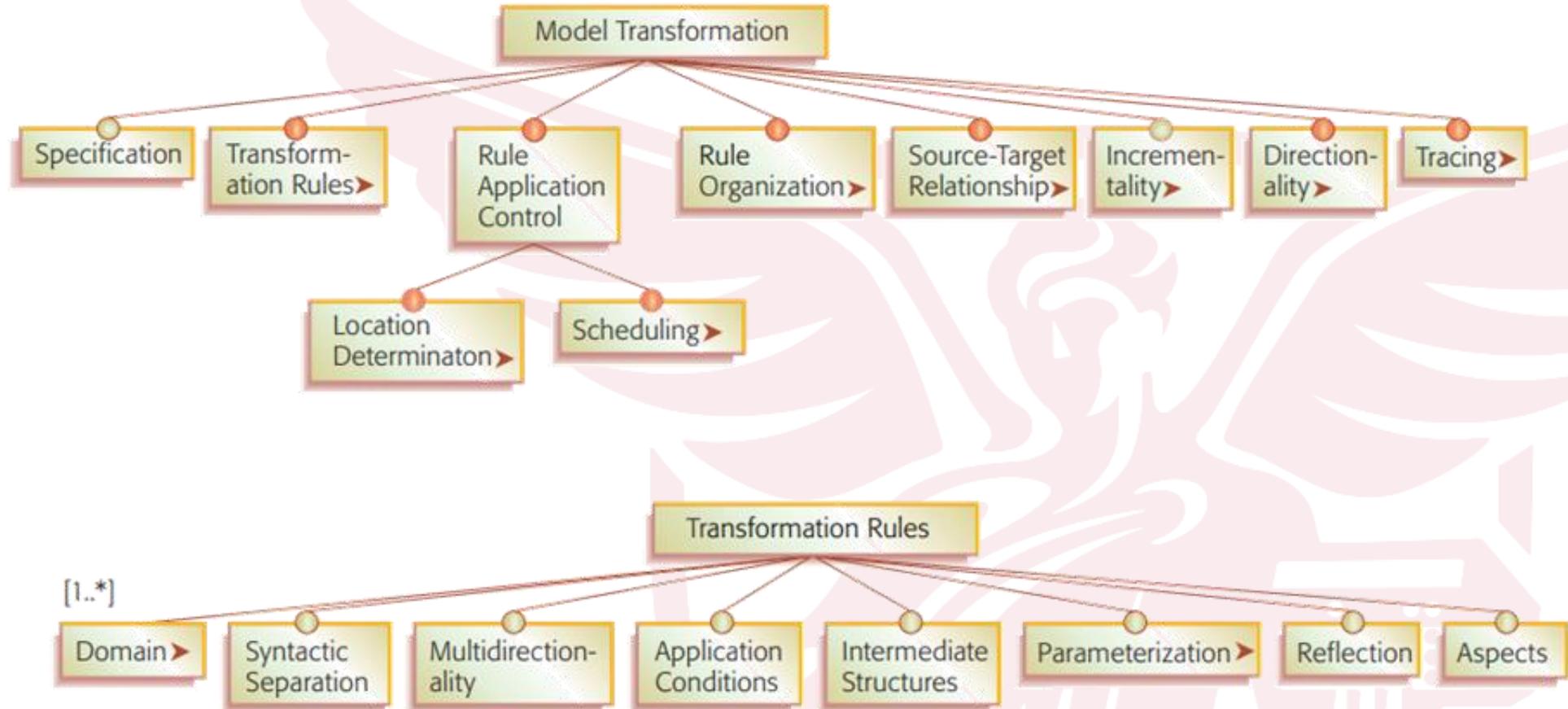
Run

Run

Finish

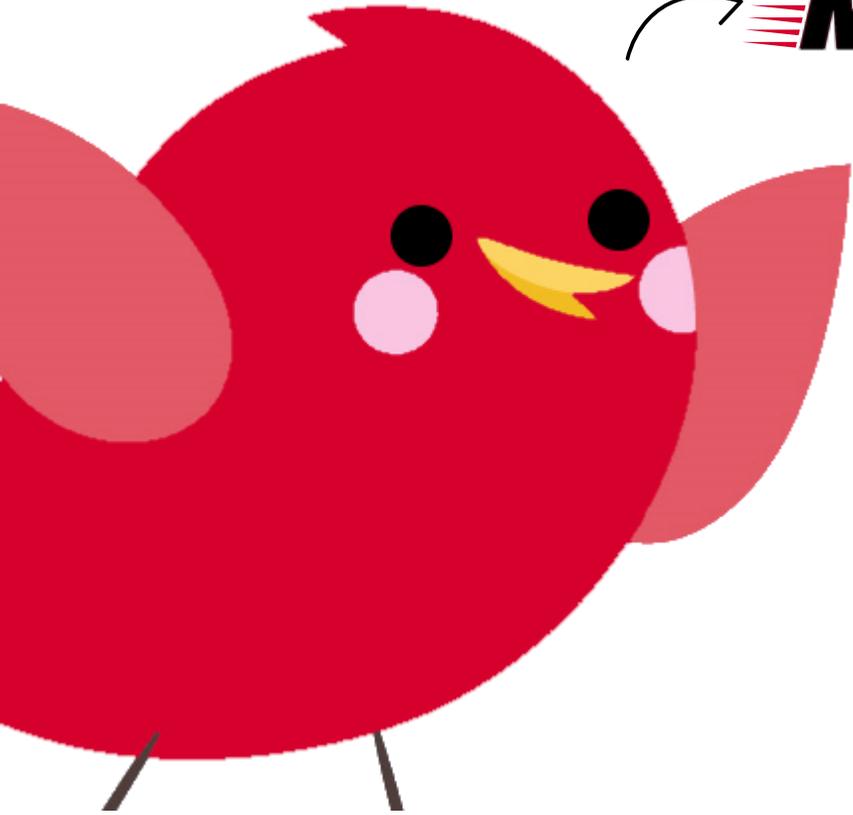
Make a prediction before you run the program.

# • Alternative transformation approaches





 **MDE FOR ALL**



**Questions?**

**Conclusion**