

CHRIS GERPHEIDE <[CHRIS@BE-SPOKE.IO](mailto:CHRIS@BE-SPOKE.IO)>

2019-09-06

---

# MODELS ALL THE WAY DOWN: CREATING A PROGRAMMING LANGUAGE



# CHRIS

VP of Engineering, Bespoke Inc

B.A. Mathematics and Political Science,  
Grinnell College, USA

M.Sc. Computer Science and Engineering,  
Eindhoven University of Technology, Netherlands

---



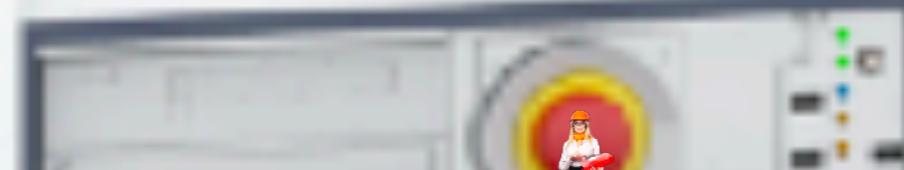
BESPOKE





BESPOKE

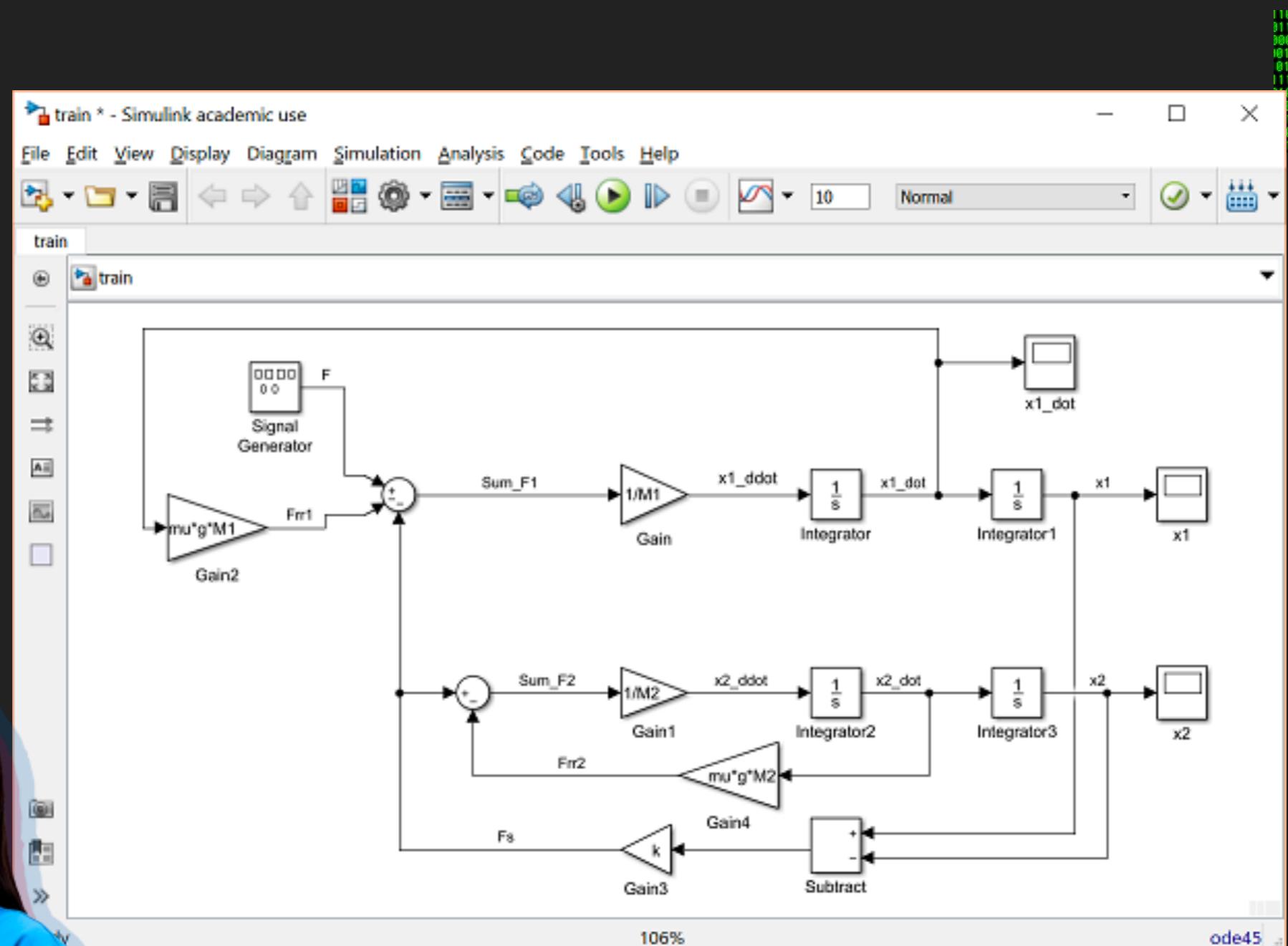
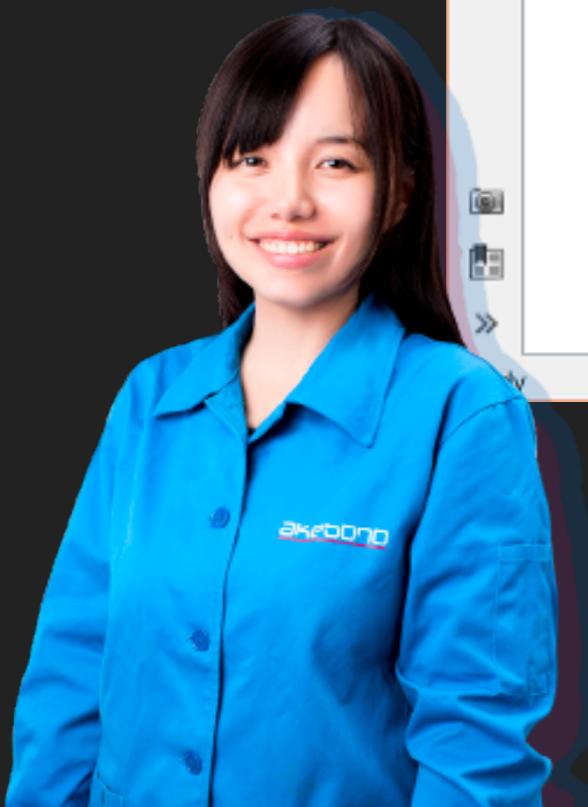
```
011001100111001000000000  
011001010101110010 0 01  
000001100000101110000  
0100110000101110000  
011011100110011001  
0110010 0 0111011011  
0100000001110000001100  
00000011010010110110111  
011000010111000000110  
010011001110010000000  
0110010101110010 0 01
```



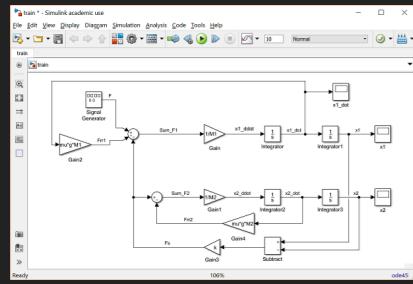


BESPOKE

11001100111001000000  
0110010101110010■01  
00000110000101110000  
10100110000101110000  
01101110011001110011  
1110010■01110111011  
01000000111000001100  
00000110100101101111  
01100001011100000110  
11001100111001000000  
0110010101110010■01



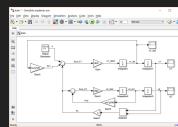
BESPOKE



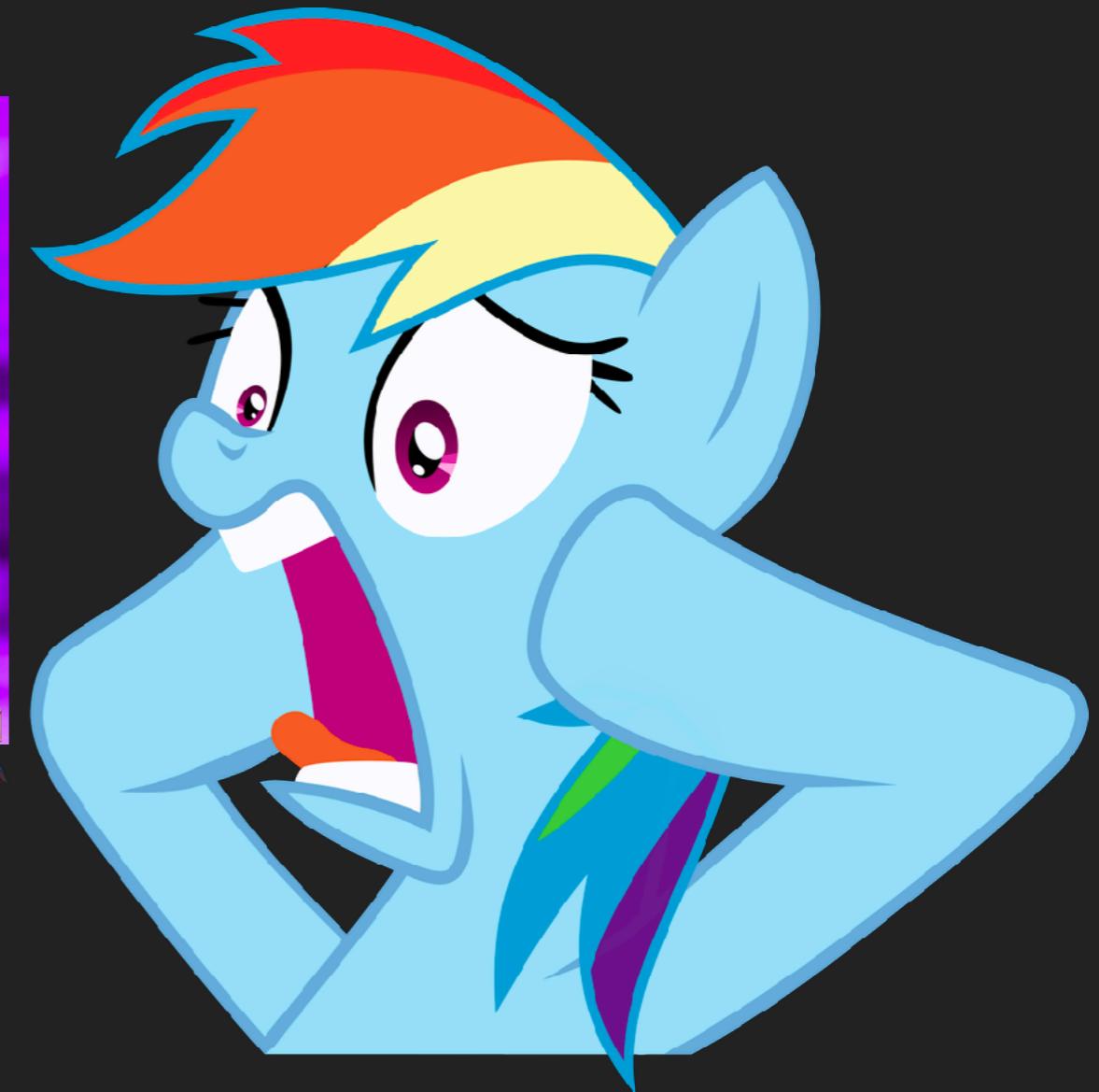
```
11001100111001100000  
111001010111000101  
00000110000101110000  
01000110000101110000  
01101110011001110011  
111001010111000101  
11000000111000001100  
00000110100101101111  
11100001011100000110  
11001100111000101  
111001010111000101
```



BESPOKE



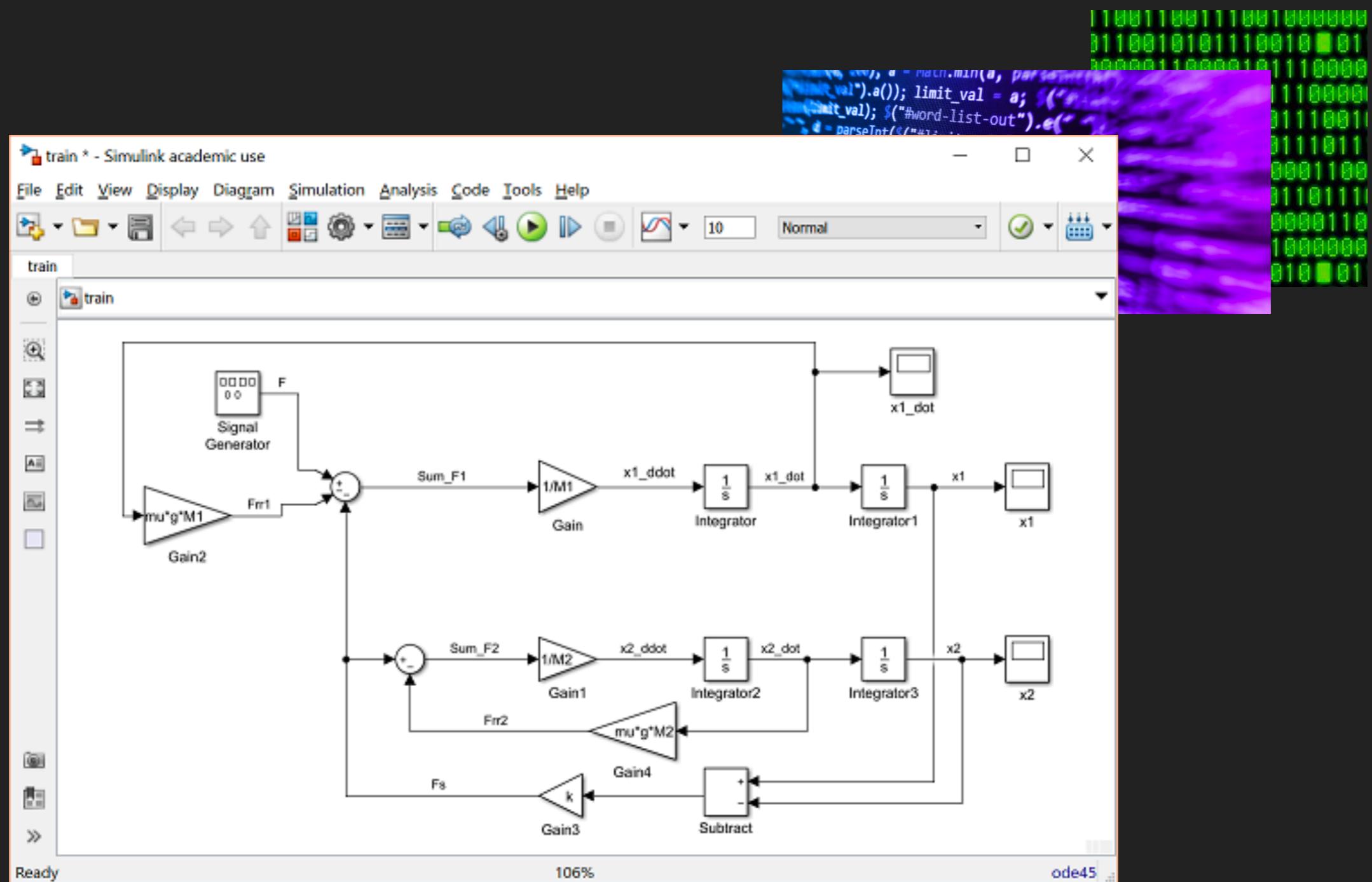
```
    a = Math.min(a, parseInt($("#limit_val").a())); limit_val = a; $( "#word-list-out" ).empty(); d = parseInt($("#limit_val").a()), file_number).e()); function("LIMIT", f = d, function("check_random_use_wystepuje", b, c[g]), -1 < e && b.splice(e, 1); if (0 < c.length) { use_wystepuje:"parameter", word:c[g], b.splice(e, 1); e = m(b, void 0); -1 < e && b.splice(e, 1); for (c = 0; c < d; c++) { b[c].b), "parameter", word:c[g], b[c].b); } } );
```



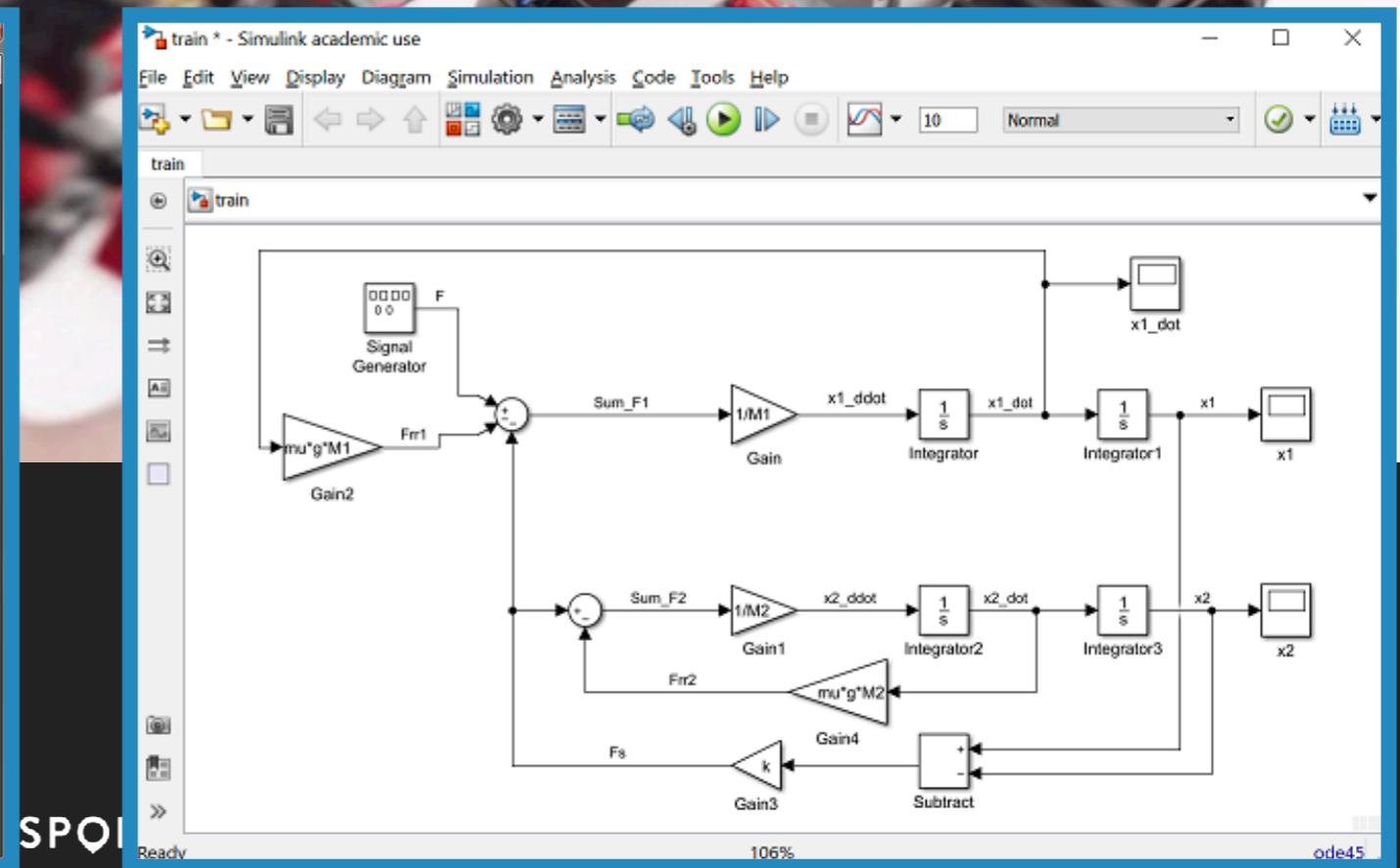
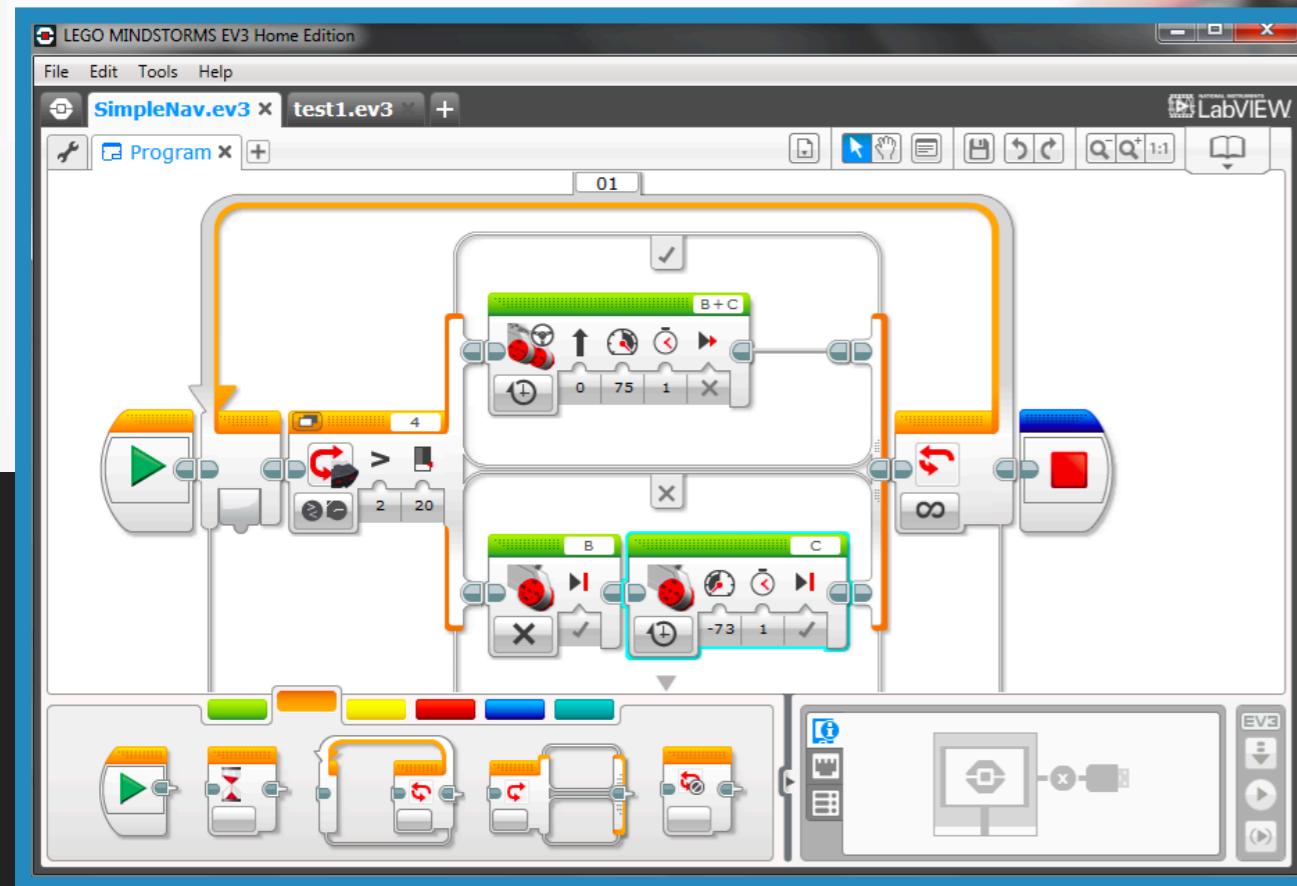
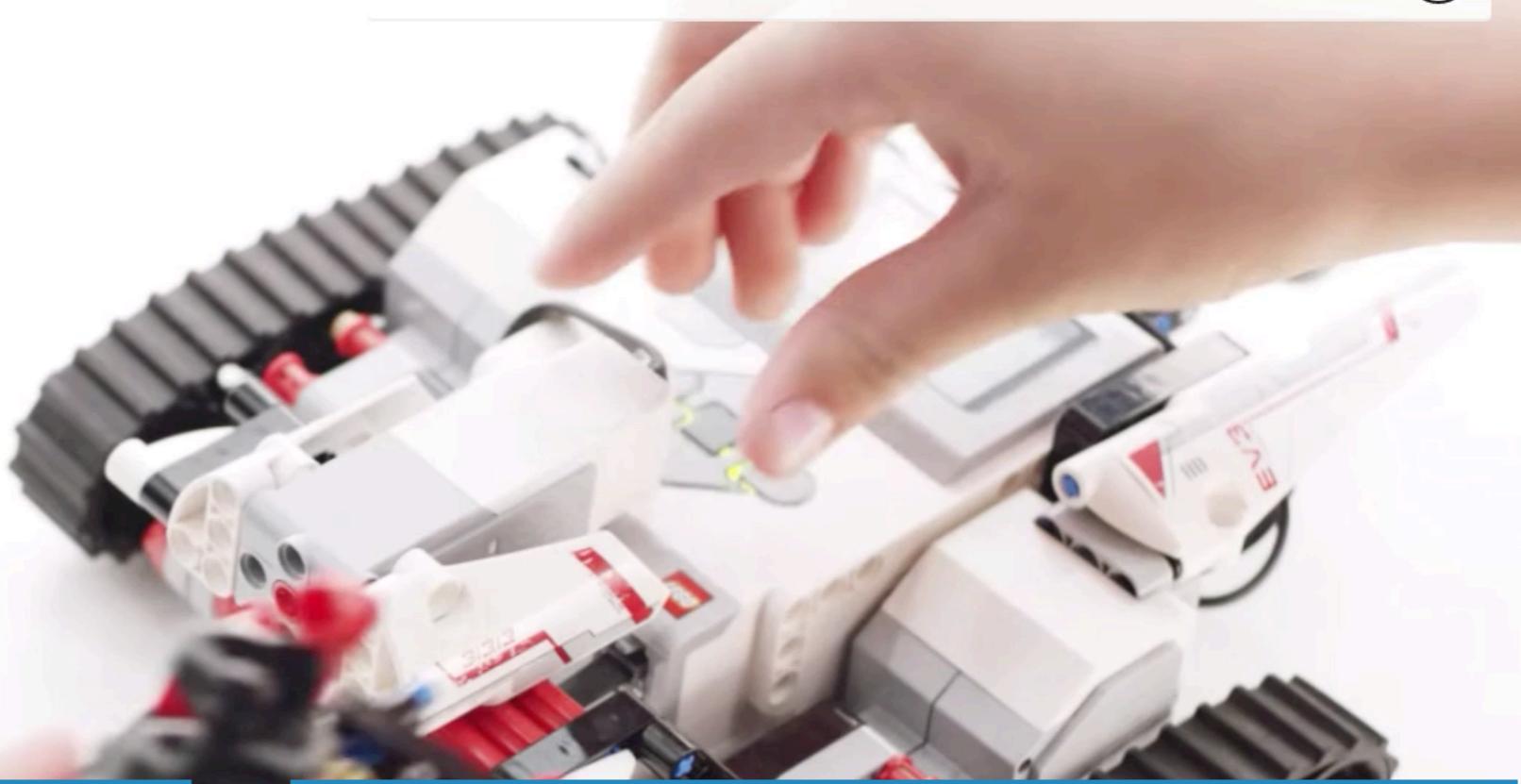
BESPOKE

# “Assessing and improving quality of QVTo model transformations”





Suggested: Introduction for Parents - LEGO MINDSTORMS EV3

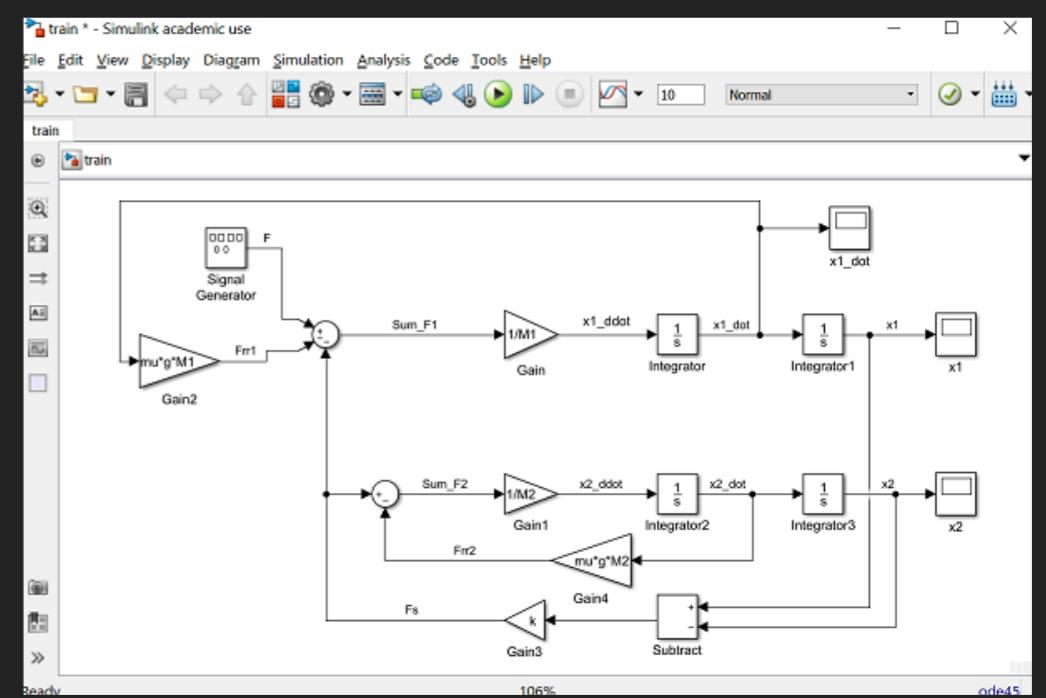


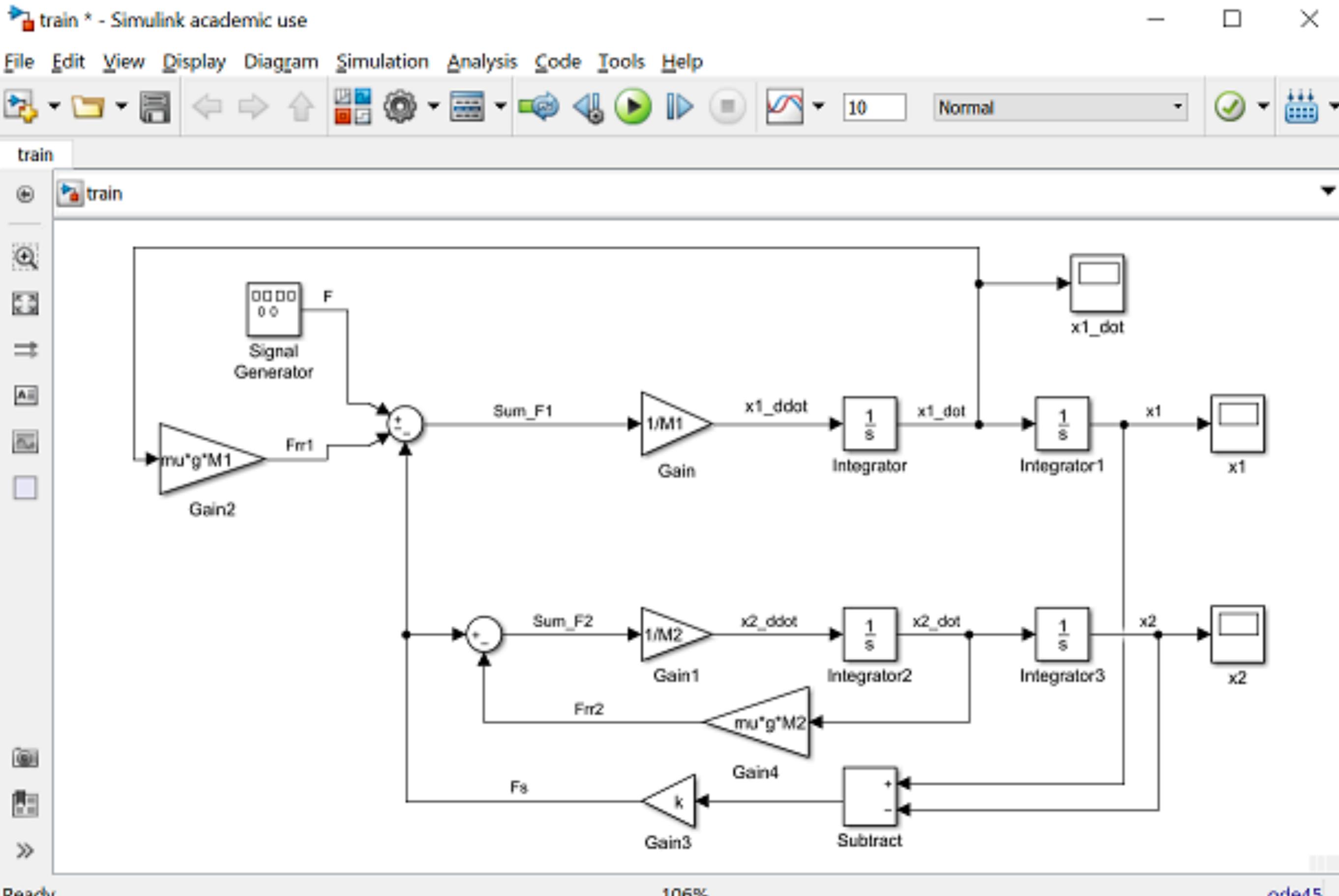


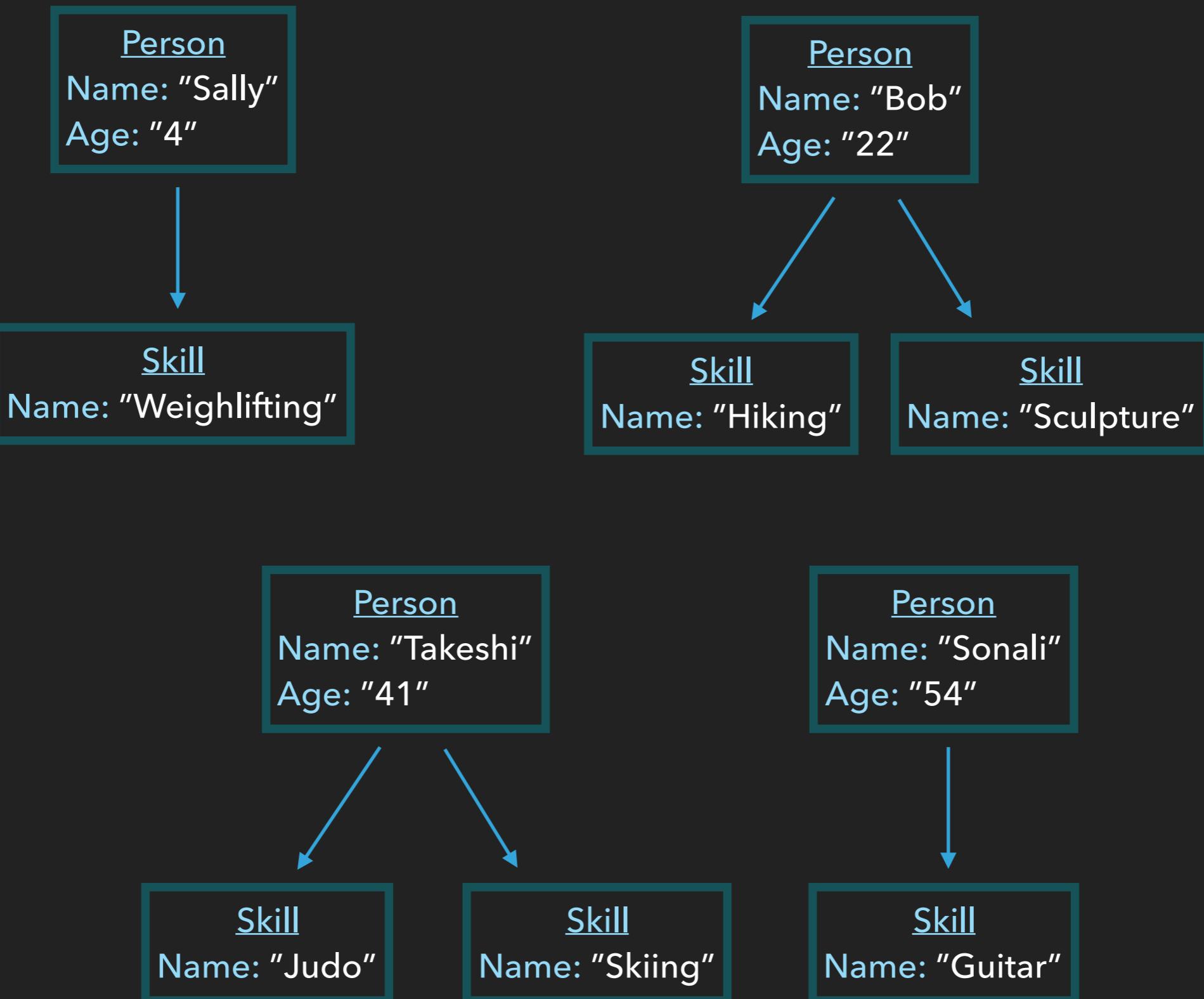
BESPOKE

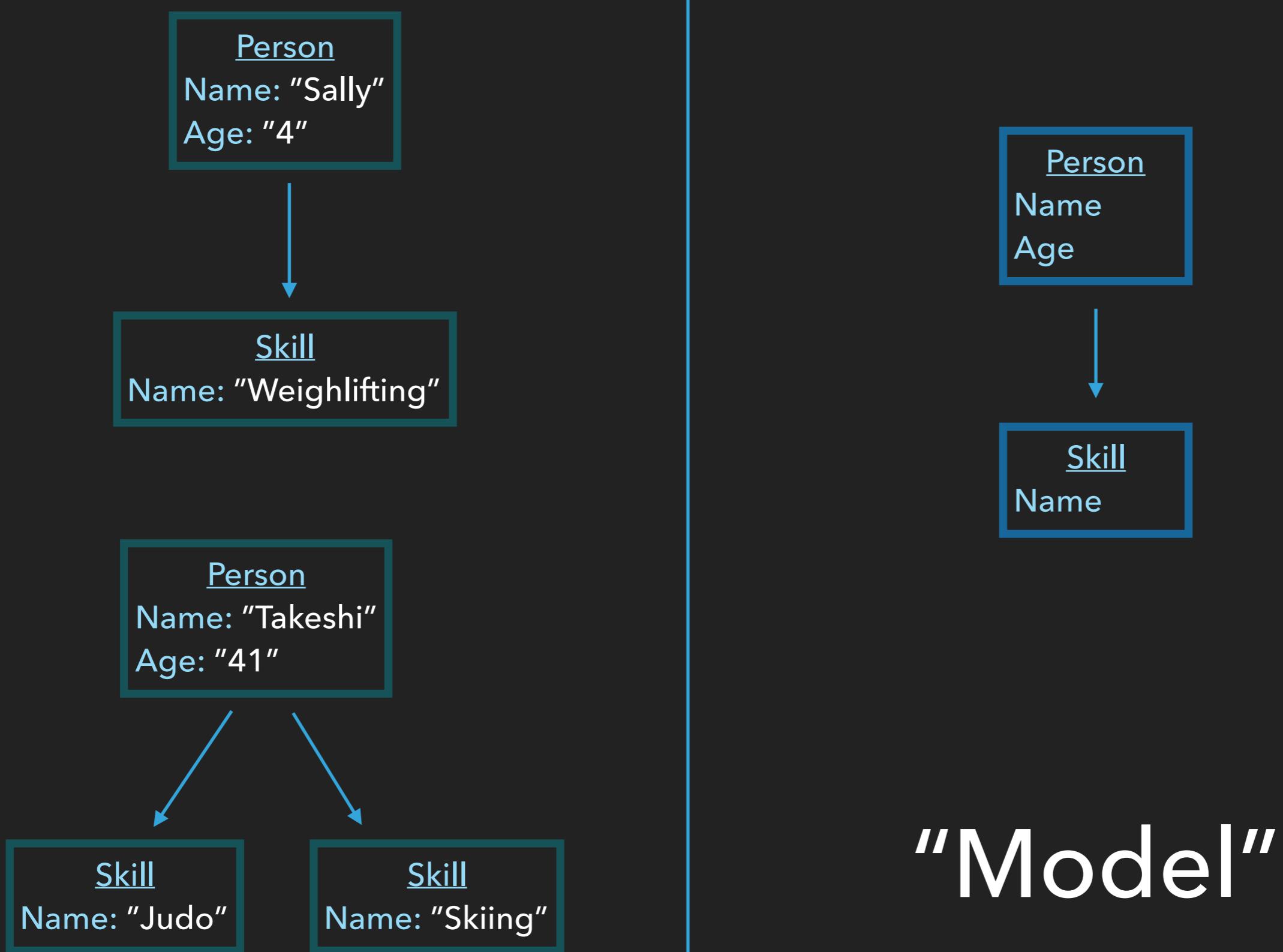
Binary code grid:

```
1100110011100010000000  
0110010101110010 01  
00000110000101110000  
10100110000101110000  
0110111001100111001  
1110010 01110111011  
01000000111000001100  
0000011010010110111  
01100001011100000110  
11001100111001000000  
0110010101110010 01
```

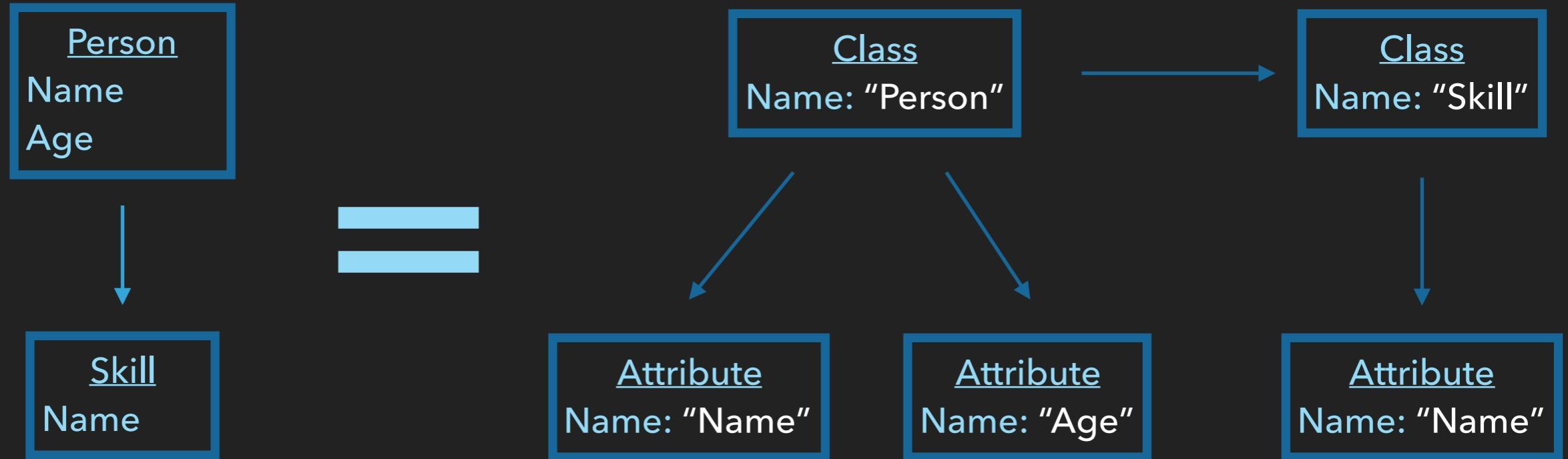




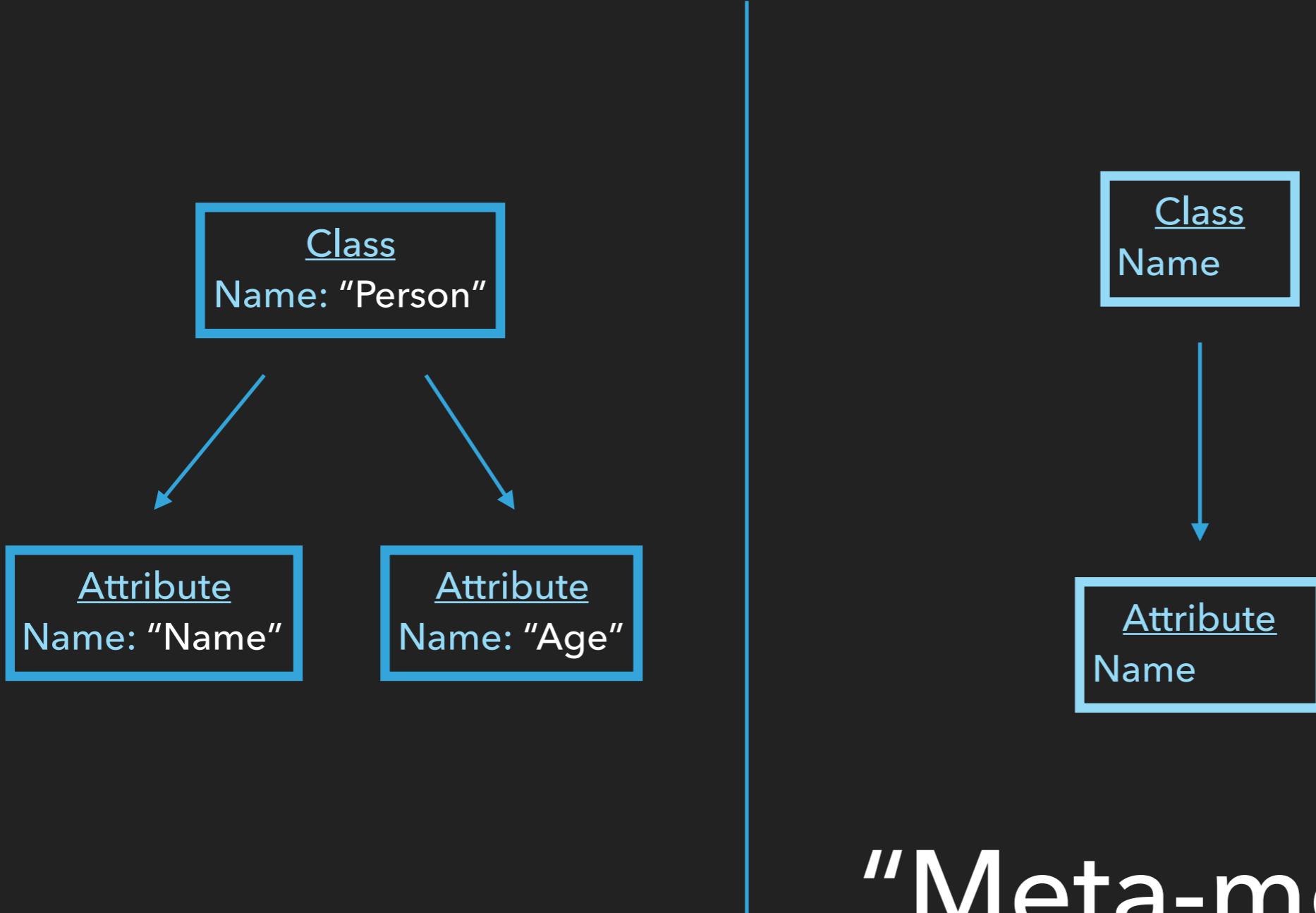




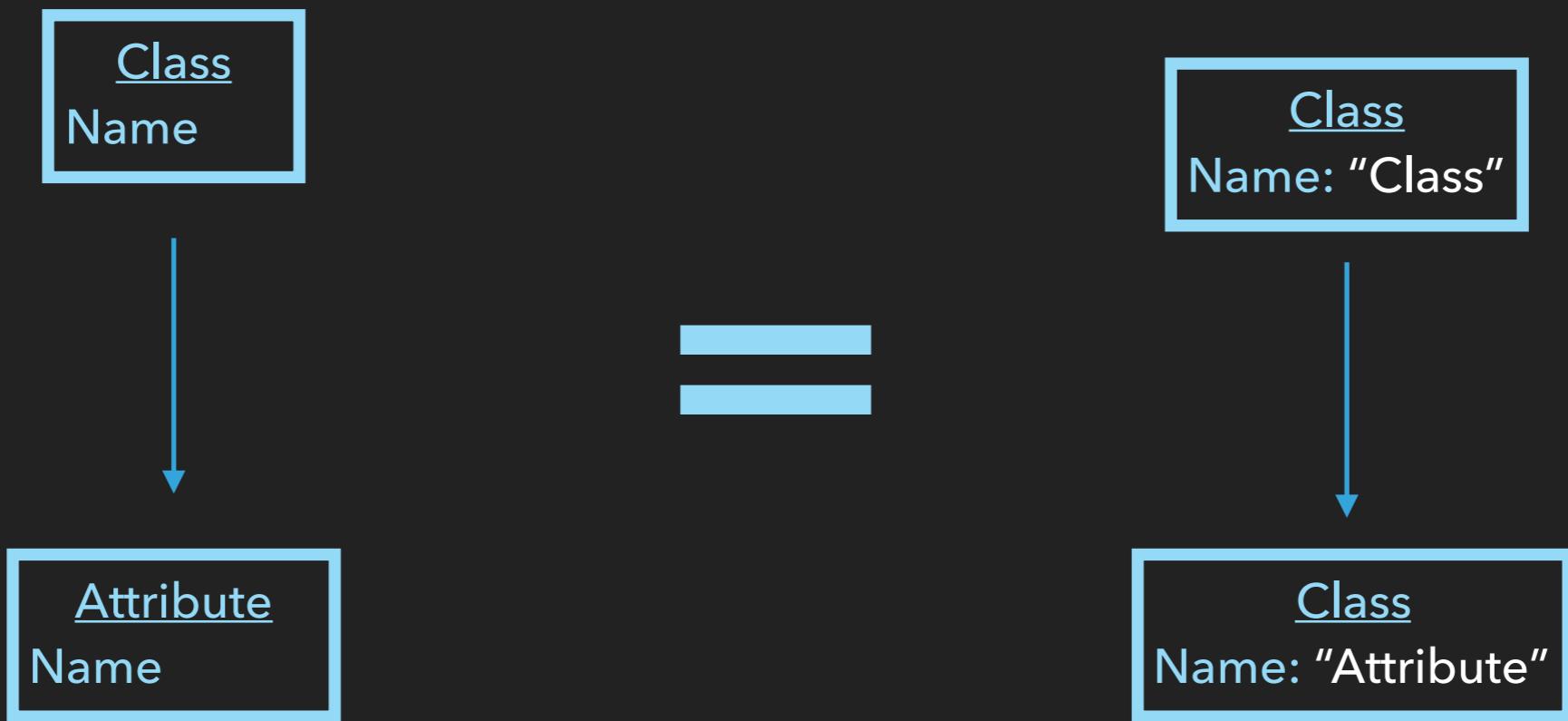
"Model"



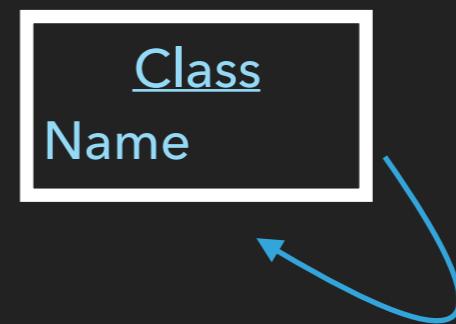
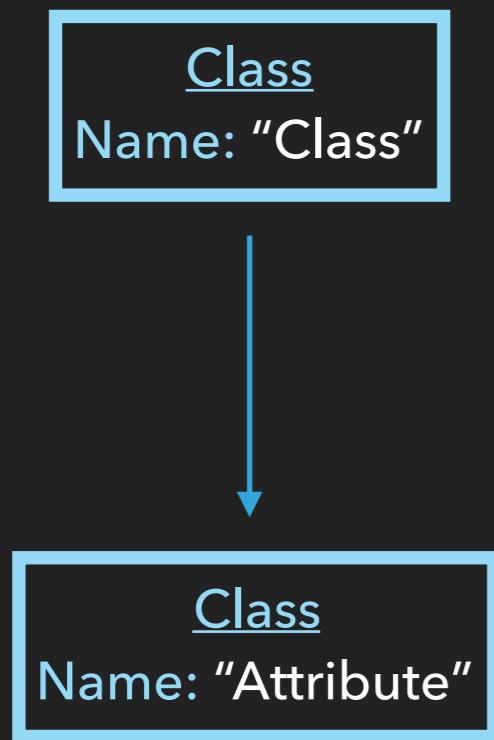
“Model”



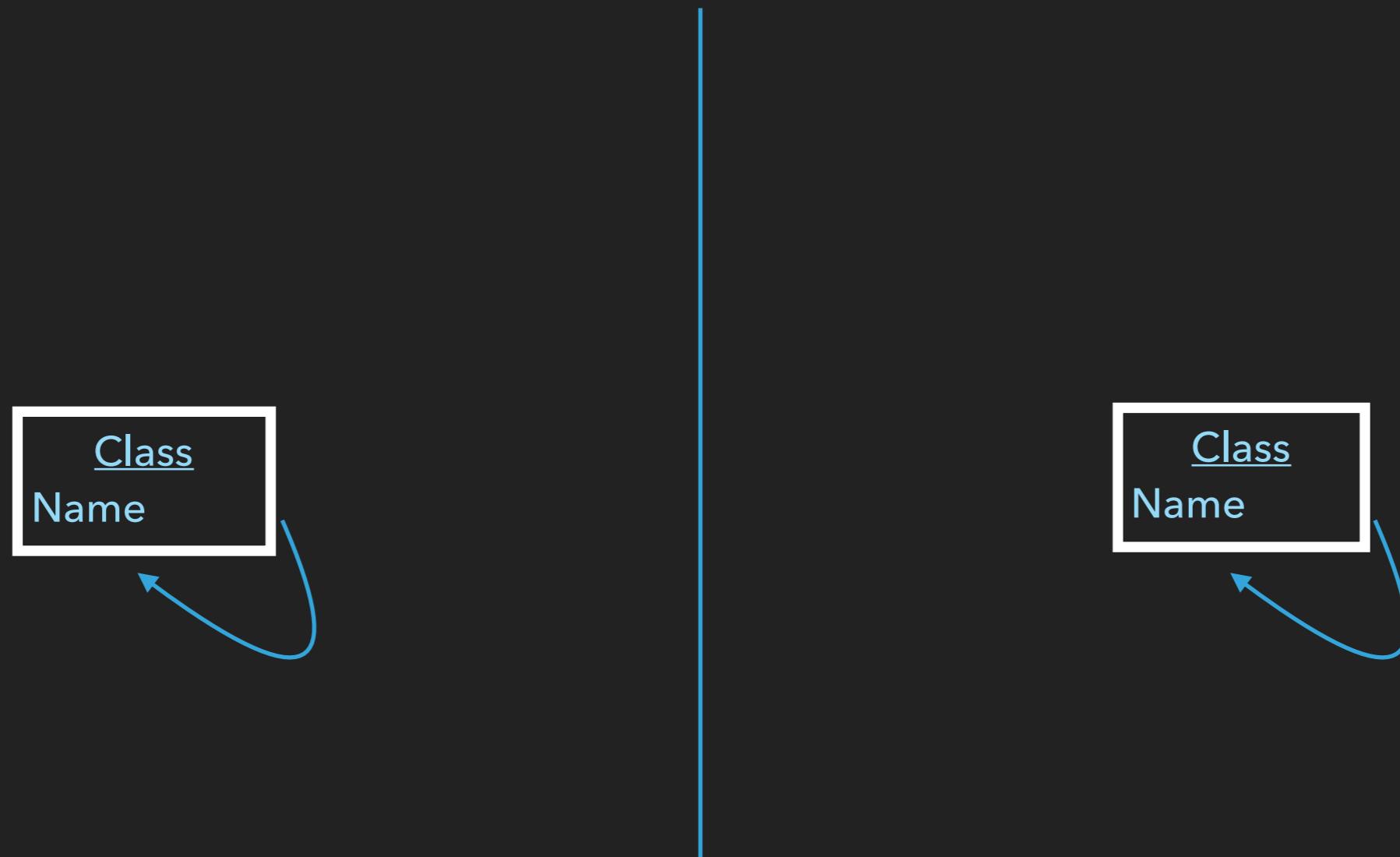
“Meta-model”



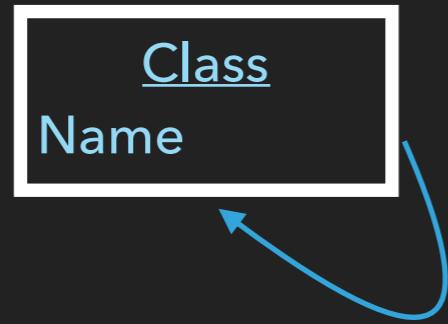
“Meta-model”



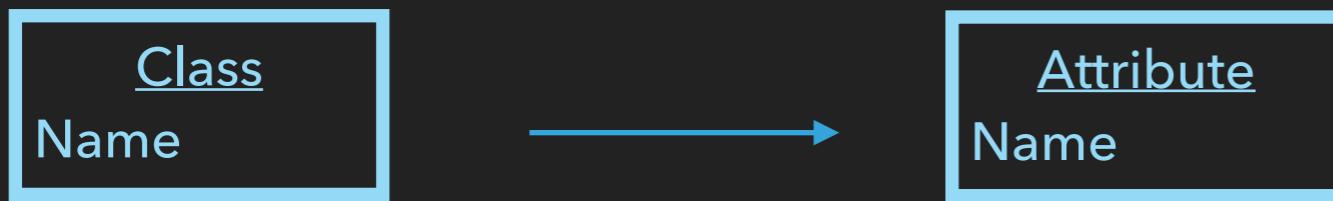
**"Meta-meta-model"**



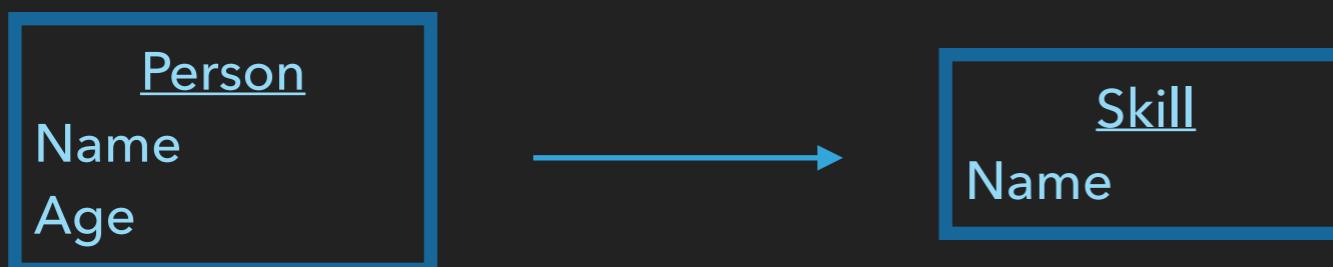
“Meta-meta-model”



"meta-meta-model": Model of a model of a model



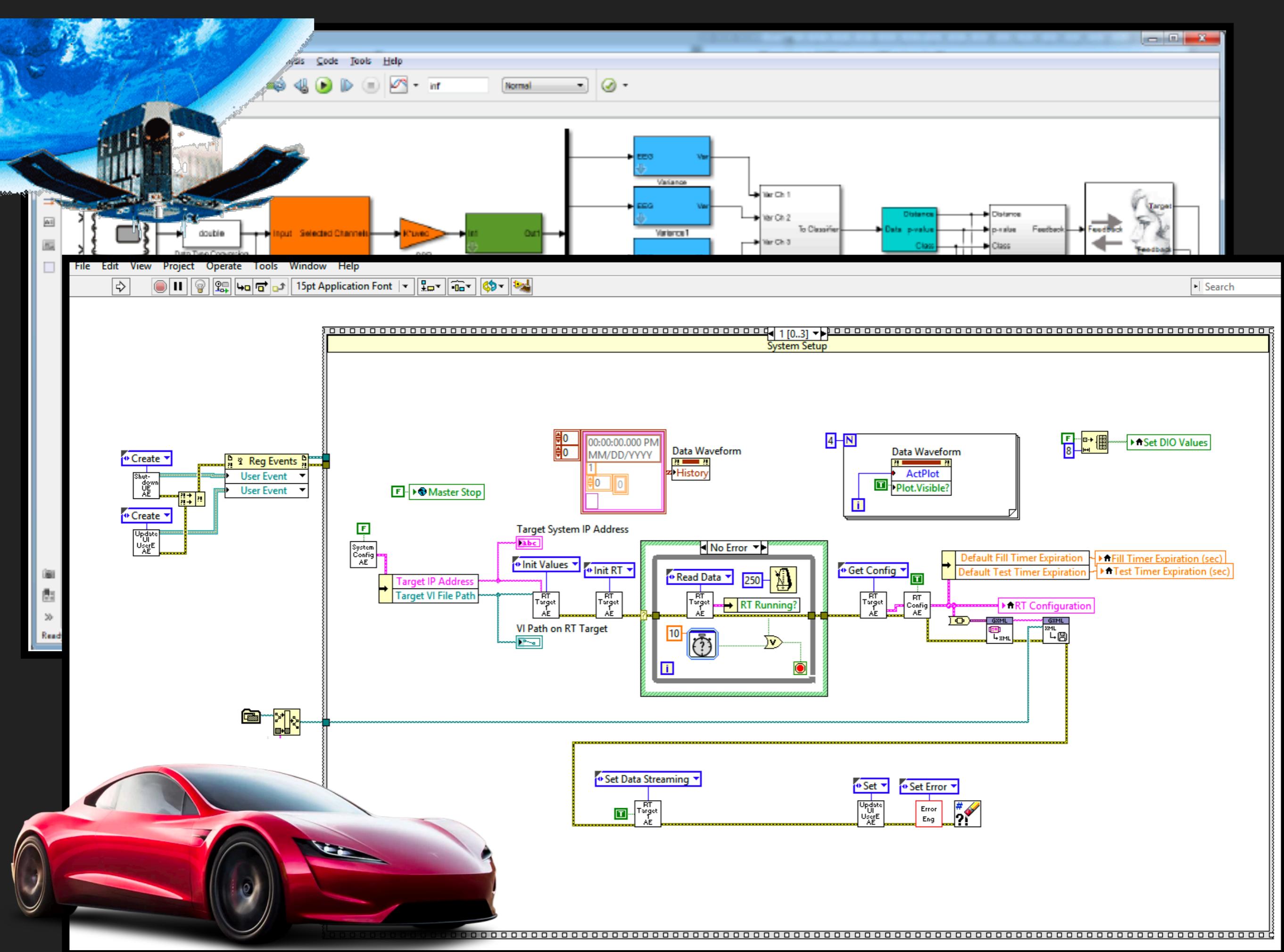
"meta-model": Model of a model



"model": Model of a system



"concrete model": System



**LEGO MINDSTORMS**

Ages 10+ 31313 601

Infrared Sensor  
Please your robot see

Programmable Brick  
The brain and voice of your robot

Scratch File Edit

Beginners by ctchome1 (shared) v448

**PaintPot2**

Screen1 Add Screen ... Remove Screen Designer Block

**Blocks**

- Built-in
  - Control
  - Logic
  - Math
  - Text
  - Lists
  - Colors
  - Variables
  - Procedures
- Screen1
  - HorizontalArrangement1
    - RedButton
    - BlueButton
    - GreenButton
  - DrawingCanvas
  - HorizontalArrangement2
    - ButtonWipe
    - ButtonBig
    - ButtonSmall

Rename Delete

**Media**

kitty.png Upload File ...

⚠ 0 ⚡ 0 Show Warnings

set x to 0  
change y by 10  
set y to 0

**Viewer**

```

when RedButton .Click
do set DrawingCanvas . PaintColor to [red]
when BlueButton .Click
do set DrawingCanvas . PaintColor to [blue]
when GreenButton .Click
do set DrawingCanvas . PaintColor to [green]
when ButtonWipe .Click
do call DrawingCanvas . Clear
initialize global [small] to [2]
initialize global [big] to [8]
initialize global [dotsize] to [2]
when ButtonBig .Click
do set global dotsize to [get global big]
when DrawingCanvas .Touched
x [x] y [y] touchedAnySprite
do ? call DrawingCanvas .DrawCircle
centerX [get x]
centerY [get y]
radius [get global dotsize]
fill [true]
when DrawingCanvas .Dragged
startX [startX] startY [startY] prevX [prevX] prevY [prevY] currentX [currentX] currentY [currentY] draggedAnySprite
do call DrawingCanvas .DrawLine
x1 [get prevX]
y1 [get prevY]
x2 [get currentX]
y2 [get currentY]
when ButtonSmall .Click
do set global dotsize to [get global small]
  
```

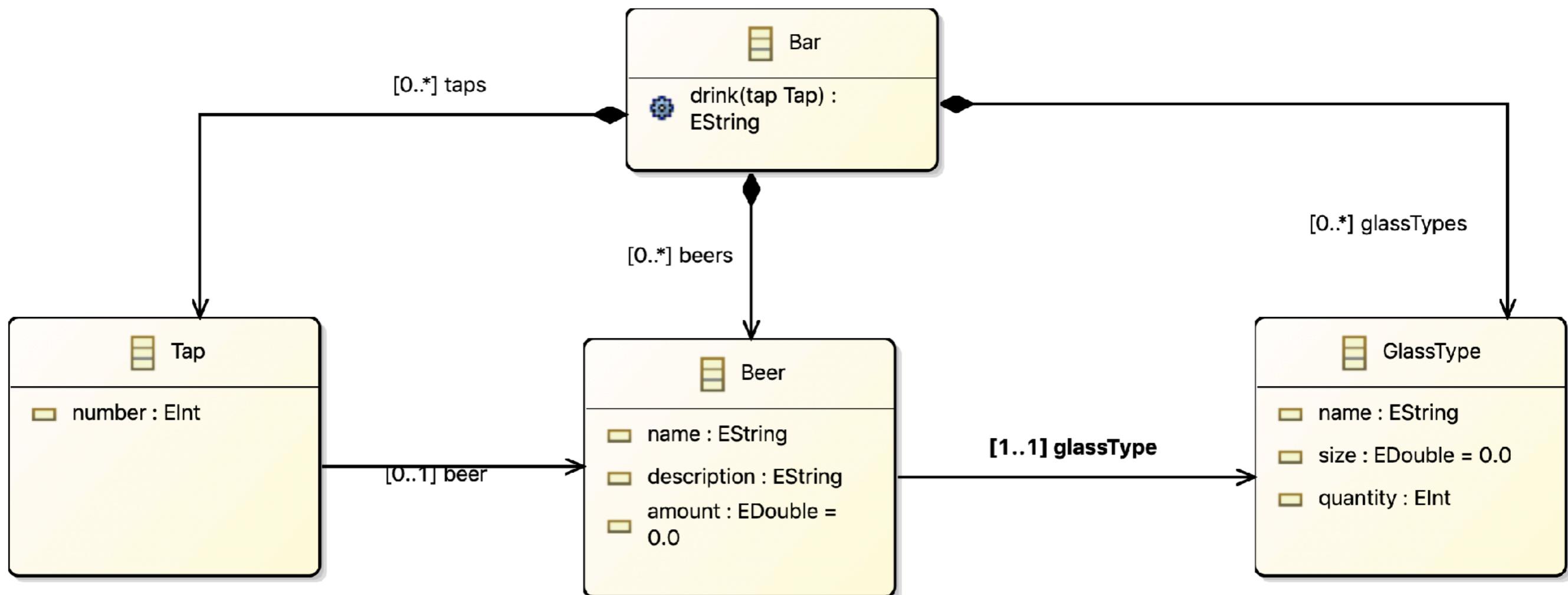
**Stage**  
4 backdrops  
New backdrop:

**Sprites**

Sprite1

**Backpack**

# Model of a bar



# Concrete model of the “Nerd Nite” bar

The screenshot shows a software interface for managing a bar's inventory. At the top, there is a title bar with the text "My.beer" and a close button. Below the title bar is a toolbar with several icons: a folder icon labeled "Resource Set", a magnifying glass icon, a star icon, a gear icon, and a refresh icon.

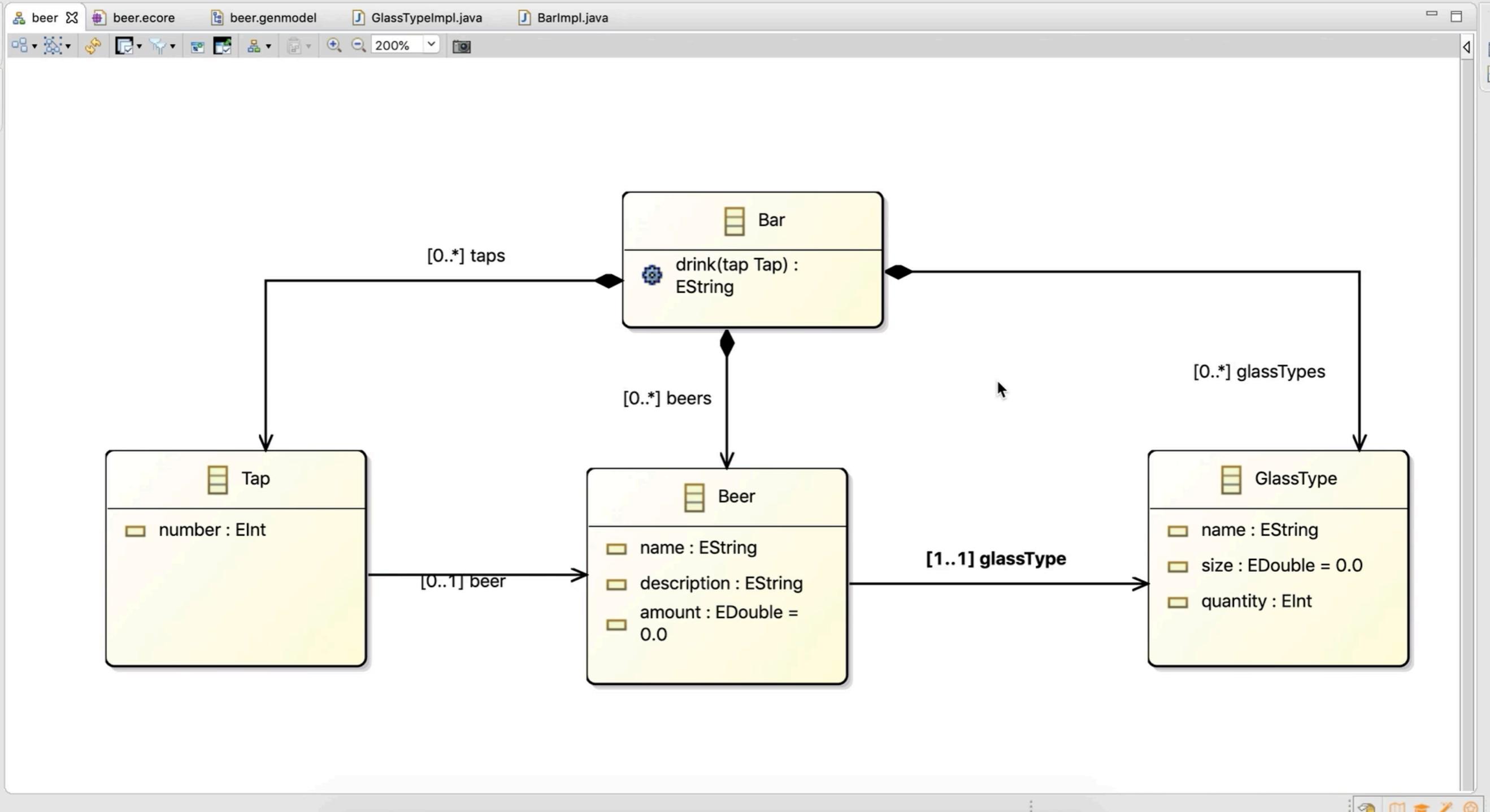
The main area is a tree view titled "Resource Set". It shows a hierarchy under "platform:/resource/GRID%20Bar%20Program/bar/My.beer":

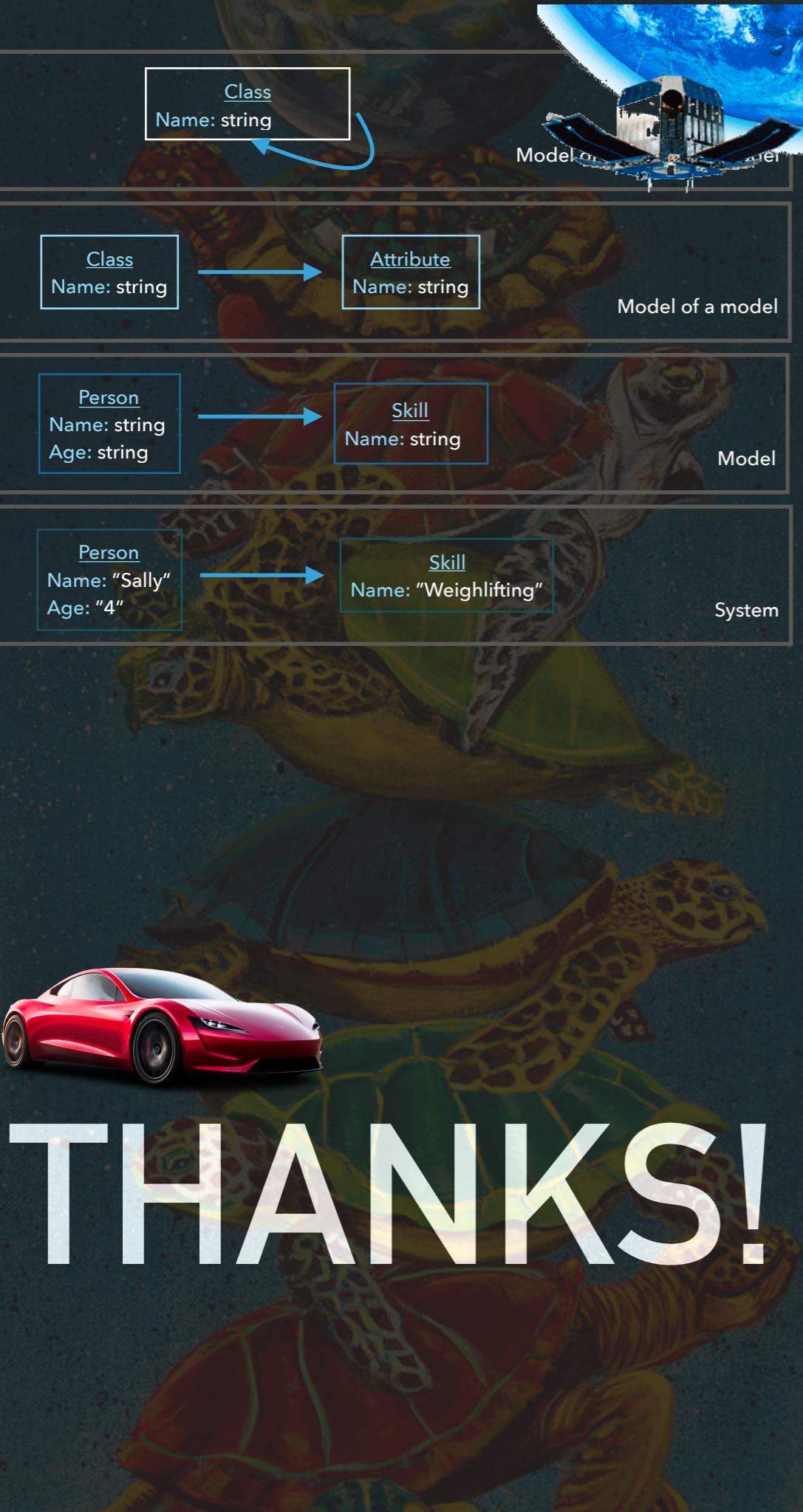
- Bar
  - Beer Westmalle Dubbel
  - Beer Hoegaarden
  - Glass Type Nerd Nite Beaker (selected item, highlighted with a blue background)
  - Tap 1
  - Tap 2

Below the tree view is a navigation bar with tabs: "Selection" (which is selected and highlighted in blue), "Parent", "List", "Tree", "Table", and "Tree with Columns".

At the bottom of the interface is a "Properties" tab, which is currently active. It displays a table of properties and their values:

Property	Value
Name	Nerd Nite Beaker
Quantity	100
Size	5.0





Chris Gerpheide  
@phoxicle  
chris@be-spoke.io

---

## Resources:

Scratch:  
<https://scratch.mit.edu>

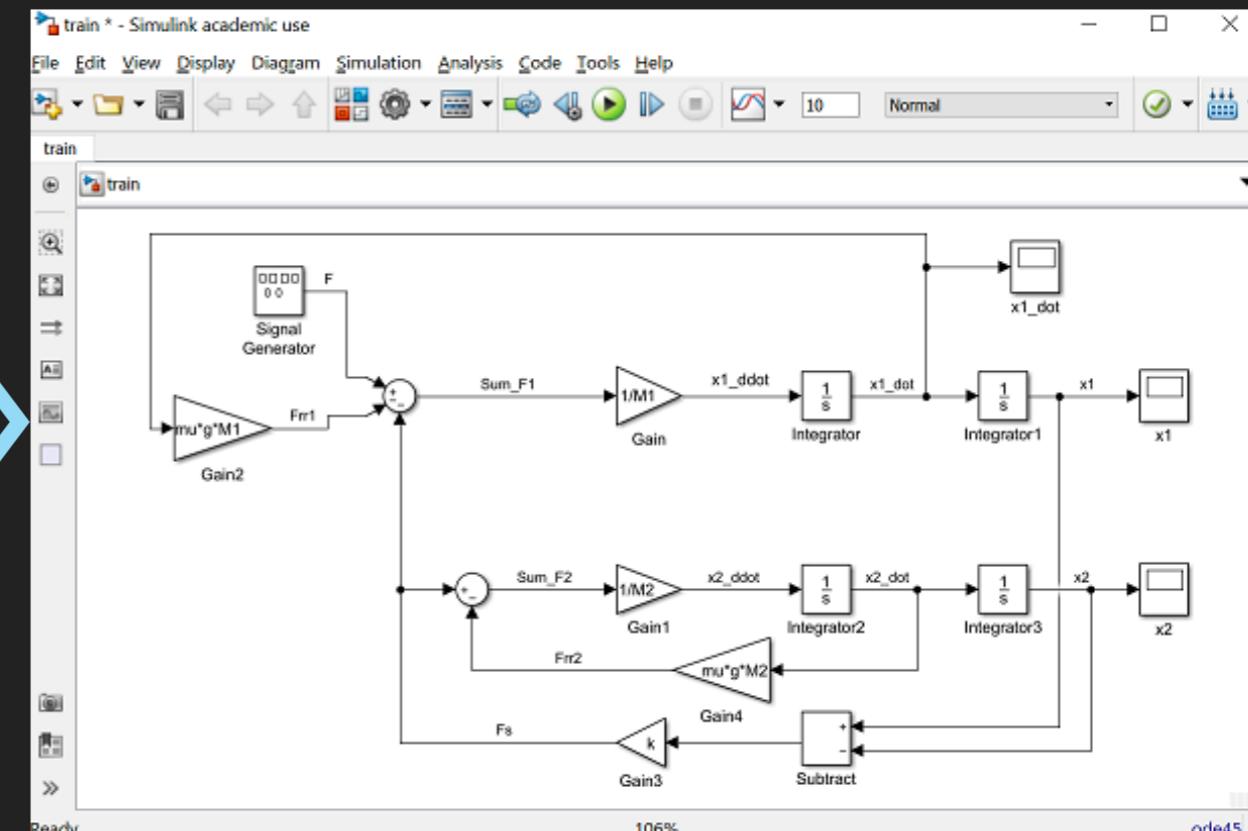
AppInventor:  
<https://appinventor.mit.edu/>

Eclipse Modeling Framework:  
<https://www.vogella.com/tutorials/EclipseEMF/article.html>

Object Management Group:  
<https://omg.org>

```
Server.java /home/taiwo/Desktop
```

```
1 public class Server {  
2     private String ip;  
3     private int port;  
4  
5     public Server(String ip, int port){  
6         this.ip = ip;  
7         this.port = port;  
8     }  
9  
10    public String getIp(){  
11        return this.ip;  
12    }  
13  
14    public void setIp(String ip){  
15        this.ip = ip;  
16    }  
17  
18    public int getPort(){  
19        return this.port;  
20    }  
21  
22    }  
23
```



# Extra: Textual language grammars

## Grammar

Arithmetic {

  Exp = AddExp

  AddExp = AddExp "+" MulExp -- plus  
    | AddExp "-" MulExp -- minus  
    | MulExp

  MulExp = MulExp "\*" PriExp -- times  
    | MulExp "/" PriExp -- divide  
    | PriExp

  PriExp = "(" Exp ")" -- paren  
    | number

  number = digit\* "." digit+ -- fract  
    | digit+ -- whole

}

Server.java /home/taiwo/Desktop

```
1 public class Server {  
2     private String ip;  
3     private int port;  
4  
5     public Server(String ip, int port){  
6         this.ip = ip;  
7         this.port = port;  
8     }  
9  
10    public String getIp(){  
11        return this.ip;  
12    }  
13  
14    public void setIp(String ip){  
15        this.ip = ip;  
16    }  
17  
18    public int getPort(){  
19        return this.port;  
20    }  
21  
22}
```