

# Down the Rabbit Hole

Manipulation, Math, and More



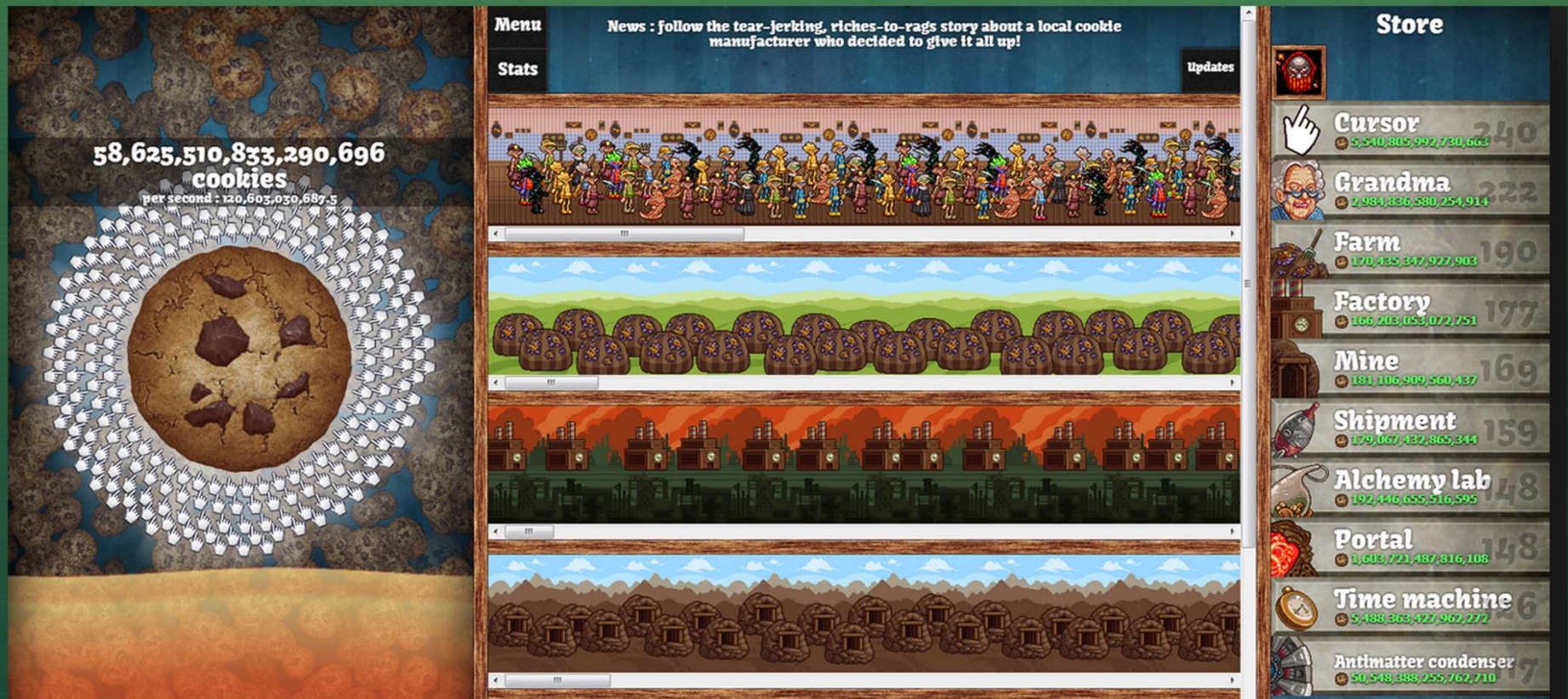
# Let's Talk Incremental Games

- Game where majority of gameplay is spent performing a basic action, with powerful upgrades and new tasks layered on top of one another
- Common sub-genre: Idle games, a “zero-player” game
  - Progress without interaction
  - Constant positive growth and feedback
  - Rapid growth of power and costs
- Incredible depth within simple mechanic
  - Candy Box, A Dark Room, Drowning (Notch)





# Let's Talk Idle Games



# Let's Talk Idle Games

[Save](#) [Load](#) [Export](#) [Import](#) [Molpy Down](#) [Coma](#) 262

Sand Tools:  
Bucket  
Buy 1 Sell  
Owned: 0  
Price:  
- 8 Castles  
Sand/mNP: 0

+1 CASTLE

Castle Tools:  
NewPixBot  
Buy 1 Sell  
Owned: 0  
Price:  
- 1 Castles

+1 CASTLE

[stuff]  
Castles  
Buy

[boost]  
Hugo  
Buy

[stuff]  
Sand  
Buy

[boost]  
Chromatic Heresy  
Buy  
Price:  
- 200 Sand  
- 10 Castles

[boost]  
Expando  
Buy  
Price:  
- 800 Sand  
- 20 Castles

File

NewPix

Links

Options

Shop

Loot

Loot Srch

Loot Nav

Sand Tools

Castle Tools

Stats

Layouts

Faves

Codex

Alerts

Log

Version

Sand / Castle #s

Tool Factory #s

Stuff #s

Newpix 1  
The Sandcastle Eon  
The Pre-expansion Era  
The Debut/What? Period

7 castles  
4 sand of 21 needed  
0 sand/mNP

Version: 4.05  
Method Not Allowed

BOOST UNLOCKED: HUGO

+1 CASTLE

Per Page: 20 << < 1 / 1 > >>  
Loot:  
[Stuff](#) [Badges](#) [Badges](#) [Tagged](#)



# Let's Talk Idle Games

THE VERGE

TECH ▾

SCIENCE ▾

CULTURE ▾

CARS ▾

REVIEWS ▾

LONGFORM

VIDEO

MORE ▾



GAMING

PLAY THIS

TECH

## TL;DR

### A game about AI making paperclips is the most addictive you'll play today

20

By James Vincent | @jjvincent | Oct 11, 2017, 7:40am EDT

[Idle games on Kongregate](#)

<https://www.kongregate.com/idle-games> ▾

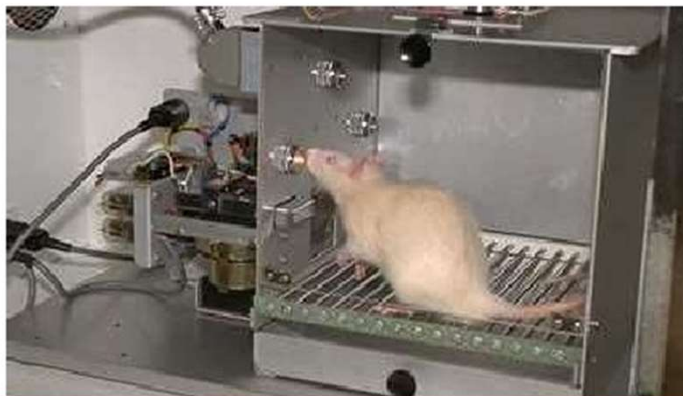
We have over 2057 of the best **Idle games** for you! Play online for free at **Kongregate**, including Clicker Heroes, Trimps, and Realm Grinder.

[Idle](#) · [Browse Idle Games](#) · [Factory idle](#) · [Learn to Fly Idle](#)

# Are we having fun?

## Skinner Box – Postive Reinforcement

- Initially behaviour was random, but can inadvertently tripped lever and was rewarded with a food pellet
- The rat didn't take long to learn that the lever represented a means of obtaining the reinforcer
- The consequence of receiving food (a desirable stimulus) for lever pressing ensured that it would repeat the action
- Skinner was initially using a **continuous reinforcement** schedule
- .



# Growstone (Incremental Game)

- Positive Reinforcement:

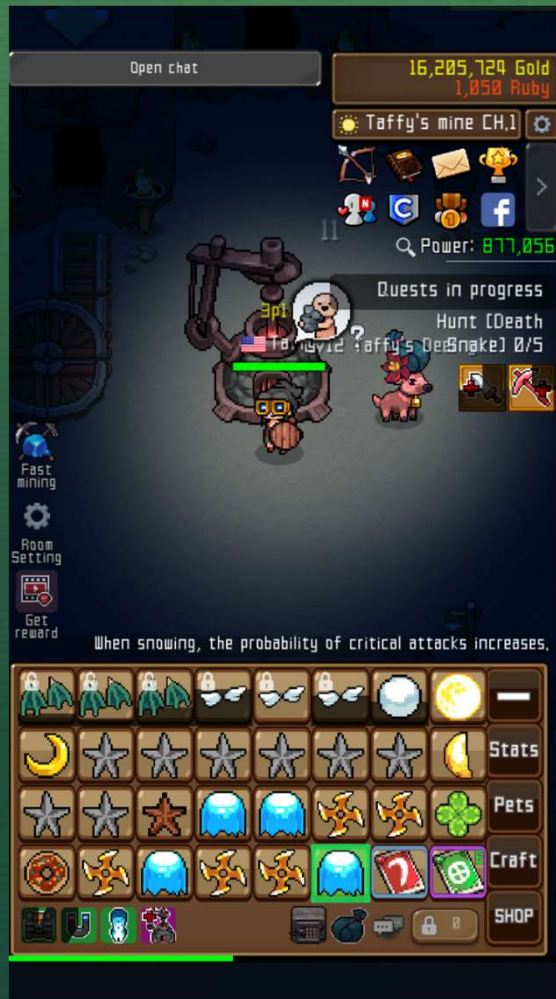
- Earn tokens from fighting enemies with stones that you constantly grow and power up
- Upgrade utilities that automatically merge
- Daily bonuses and time-limited dungeons

- Monetization

- Paid currency and buffs awarded to loyal ad-consuming
- Paid currency and loot packages



# Growstone





# Why Bot?

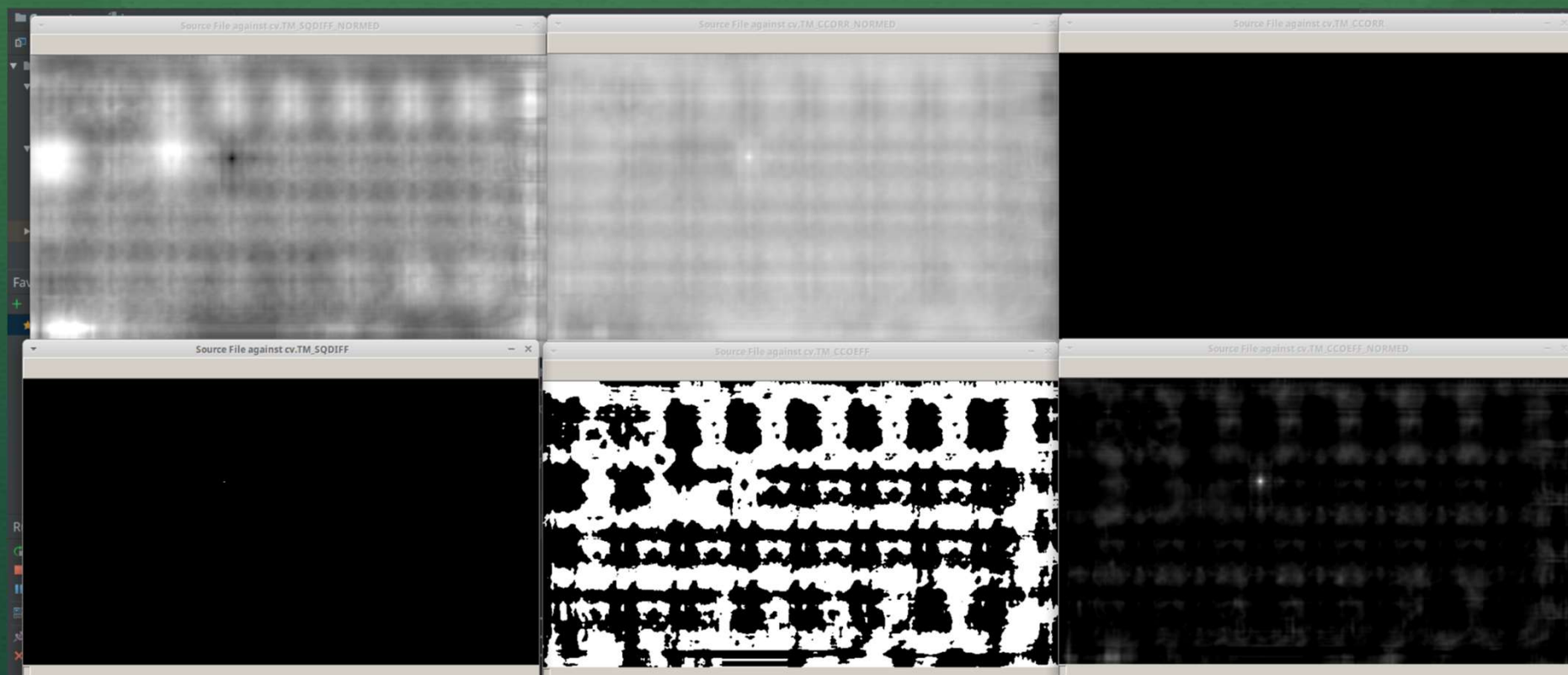
	Skip Chance	0.300	
Levels	Baseline	Plain Value	Expected Value
0	1	1.000	1.000
1	2	2.000	2.000
2	4	4.000	3.400
3	8	6.800	5.960
4	16	11.920	10.384
5	32	20.768	18.114
6	64	36.227	31.589
7	128	63.179	55.093
8	256	110.187	96.084
9	512	192.169	167.574
10	1,024	335.148	292.254
11	2,048	584.509	509.701
12	4,096	1,019.402	888.934
13	8,192	1,777.868	1,550.328
14	16,384	3,100.655	2,703.819
15	32,768	5,407.638	4,715.543
16	65,536	9,431.087	8,224.052
17	131,072	16,448.104	14,342.999
18	262,144	28,685.998	25,014.630
19	524,288	50,029.260	43,626.281
20	1,048,576	87,252.562	76,085.572
Estimated Time in Hours assuming 5sec spawn rate	1,456.356	121.184	105.674

# What's the point?

- Manage the status of all our applications, reacting before a consumer does
- Provide insight into application problems. AEM memory leak, and failing servers
- Check assumptions. Is your build failing because of your code? A config? Chef? Nexus/NPM?



# Template Matching



# Normalized correlation coefficient

- Correlation coefficient generated from each image slice
- Dot product quickly compares vector's similarity
- Normalization makes peaks more prominent

- method=CV\_TM\_CCOEFF

$$R(x, y) = \sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))$$

where

$$T'(x', y') = T(x', y') - 1/(w \cdot h) \cdot \sum_{x'', y''} T(x'', y'')$$
$$I'(x + x', y + y') = I(x + x', y + y') - 1/(w \cdot h) \cdot \sum_{x'', y''} I(x + x'', y + y'')$$

- method=CV\_TM\_CCOEFF\_NORMED

$$R(x, y) = \frac{\sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$$



# Normalized correlation coefficient

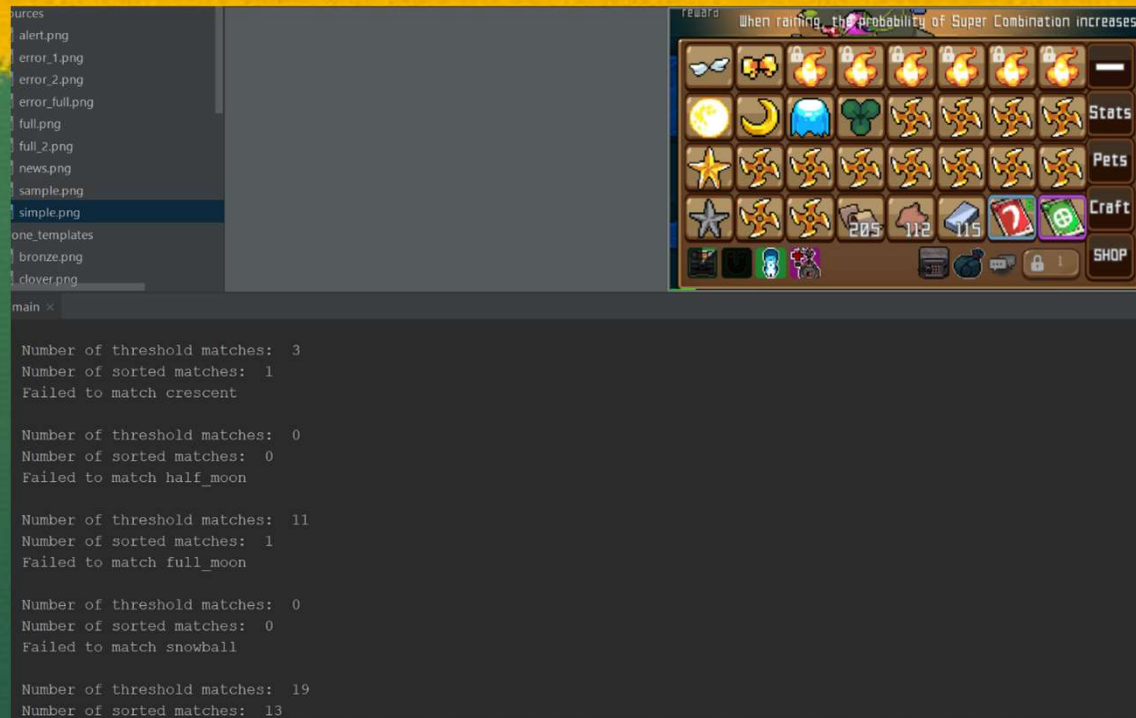
```
for method in methods:

    result_array = cv.matchTemplate(source, template, eval(method))
    match_array = np.where(result_array >= threshold)

    # Filter the matching arrays
    confidence = []
    for point in zip(match_array[1], match_array[0]):
        confidence.append([point, result_array[point[1]][point[0]]])
        cv.rectangle(
            source, point,
            (point[0] + template_width, point[1] + template_height),
            (0, 0, 255), 2)
    confidence = sorted(confidence, key=itemgetter(1), reverse=True)

    # Remove points that are too close to one another
    filtered_points = []
    confidence = [x[0] for x in confidence]
    while len(confidence) > 0:
        baseline = confidence.pop(0)
        confidence = list(
            filter(
                lambda cpoint: (abs(baseline[0] - cpoint[0]) > 5 or abs(baseline[1] - cpoint[1]) > 5),
                confidence))
        filtered_points.append(baseline)
```

# Simple Thesholding



```
Width: 45 Height: 45
Evaluating cv.TM_CCoeff_NORMED
Max Value: 0.970000147819519 Max Loc: (1574, 1155)
Match array dimensions (2, 10)
(array([[1154, 1154, 1155, 1155, 1233, 1233, 1234, 1234, 1234, 1234]], array([1495, 1574, 1495, 1574, 1180, 1338, 1180, 1338, 1416, 1417])))
Point:(1495, 1154), Confidence:0.8231894969940186
Point:(1574, 1154), Confidence:0.8790508508682251
Point:(1495, 1155), Confidence:0.9094637036323547
Point:(1574, 1155), Confidence:0.970000147819519
Point:(1180, 1233), Confidence:0.8397243618965149
Point:(1338, 1233), Confidence:0.8141580820083618
Point:(1180, 1234), Confidence:0.9230763912200928
Point:(1338, 1234), Confidence:0.9048370122909546
Point:(1416, 1234), Confidence:0.8626410365104675
Point:(1417, 1234), Confidence:0.8124781250953674
```

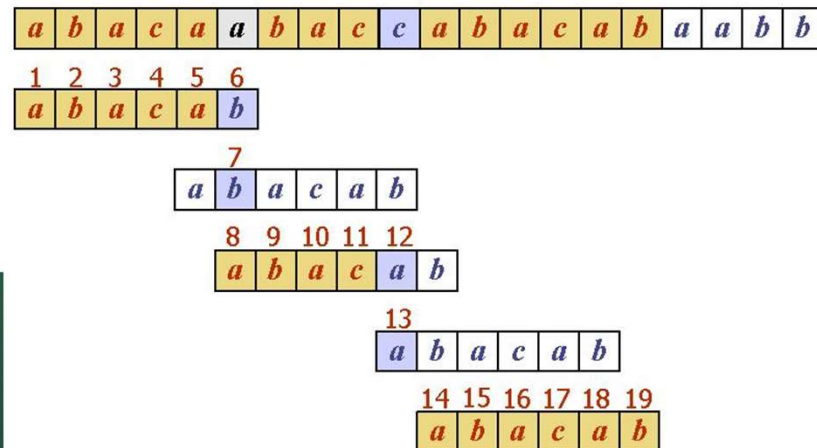


# What are people doing better?

```

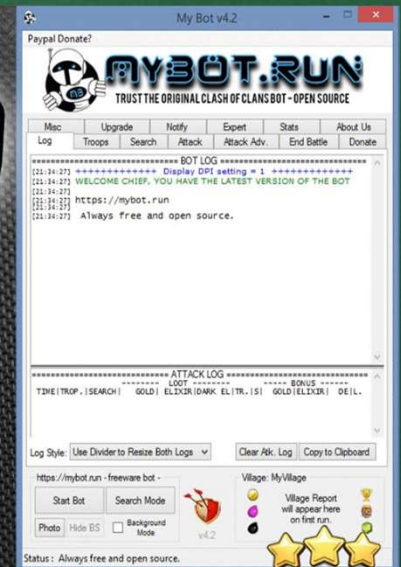
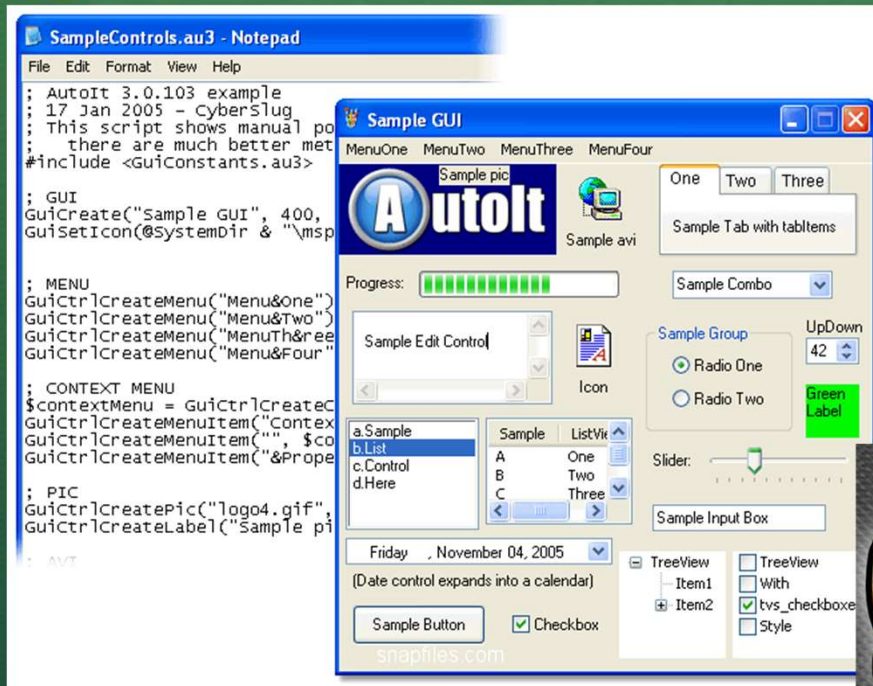
for y in range(haystackHeight):
    for matchx in _kmp(needleImageFirstRow, haystackImageData[y * haystackWidth:(y+1) * haystackWidth]):
        foundMatch = True
        for searchy in range(1, needleHeight):
            haystackStart = (searchy + y) * haystackWidth + matchx
            if needleImageData[searchy * needleWidth:(searchy+1) * needleWidth] != haystackImageData[haystackStart:haystackStart + needleWidth]:
                foundMatch = False
                break
        if foundMatch:
            # Match found, report the x, y, width, height of where the matching region is in haystack.
            numMatchesFound += 1
            yield (matchx, y, needleWidth, needleHeight)
            if limit is not None and numMatchesFound >= limit:
                # Limit has been reached. Close file handles.
                if needleFileObj is not None:
                    needleFileObj.close()
                if haystackFileObj is not None:
                    haystackFileObj.close()

```



$x$	0	1	2	3	4	5
$P[x]$	$a$	$b$	$a$	$c$	$a$	$b$
$f(x)$	0	0	1	0	1	2

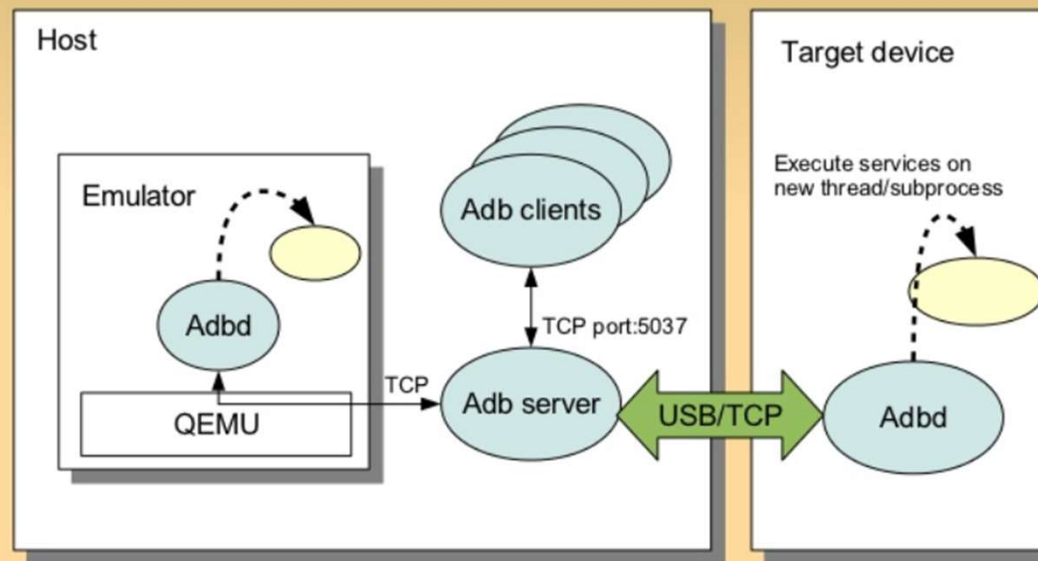
# What are people doing better?





# What are people doing better?

## ADB overview



8

- <https://www.slideshare.net/tetsu.koba/adbandroid-debug-bridge-how-it-works>

# Migration Headaches

- MacOS library differences, pip ordering
- Hunt for the safe emulator
  - Nox: encrypted info, destination unknown
  - Genymotion: ARM translation unsupported
  - BlueStacks: Mac upgrade

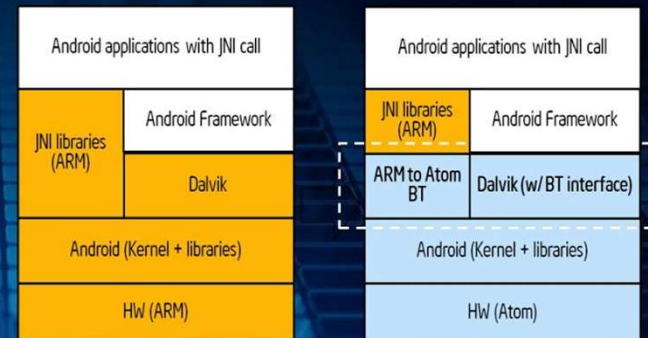


# Migration Headaches

## Intel Houdini - Arm to Intel on Android

### Binary Translator Overview

- Intel's Binary Translator can run Android Applications with native ARM Architecture binaries on Intel Atom powered devices
- Intel and Google have worked together to ensure native applications execution on Intel Atom Smartphone as is without additional porting from the application developers
- Enables most of the NDK applications to run on Intel Atom platforms with no additional effort/cost to develop
- Provides comparable user experience as the existing Android Smartphone
- Addresses end user expectations of an Android Smartphone - Applications that they know and love to just work.



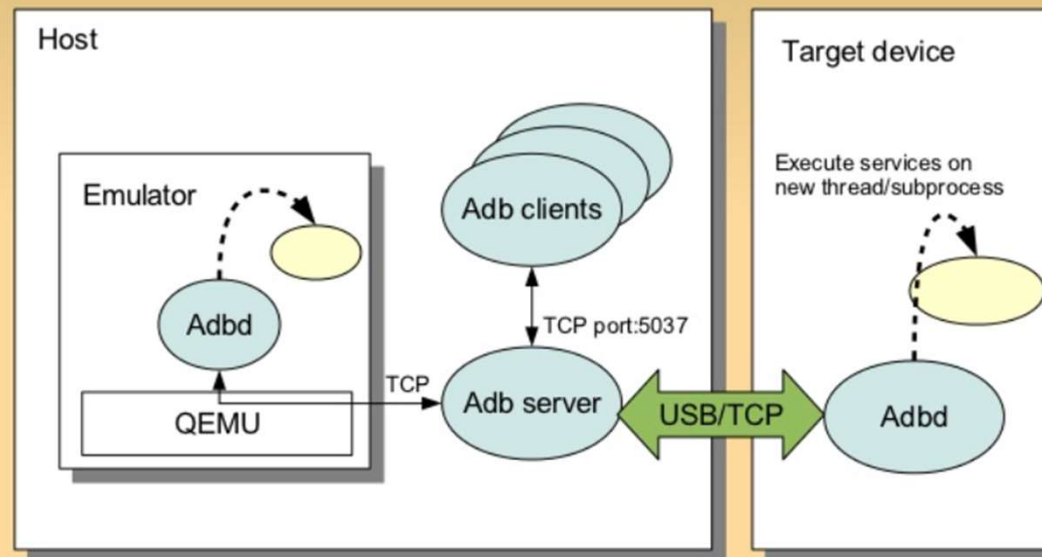
- Applications are executed by JVM for Java and BT for ARM native libraries; fully transparent to end users
  - Very minimal power overhead
- Unnoticeable performance impact for most applications



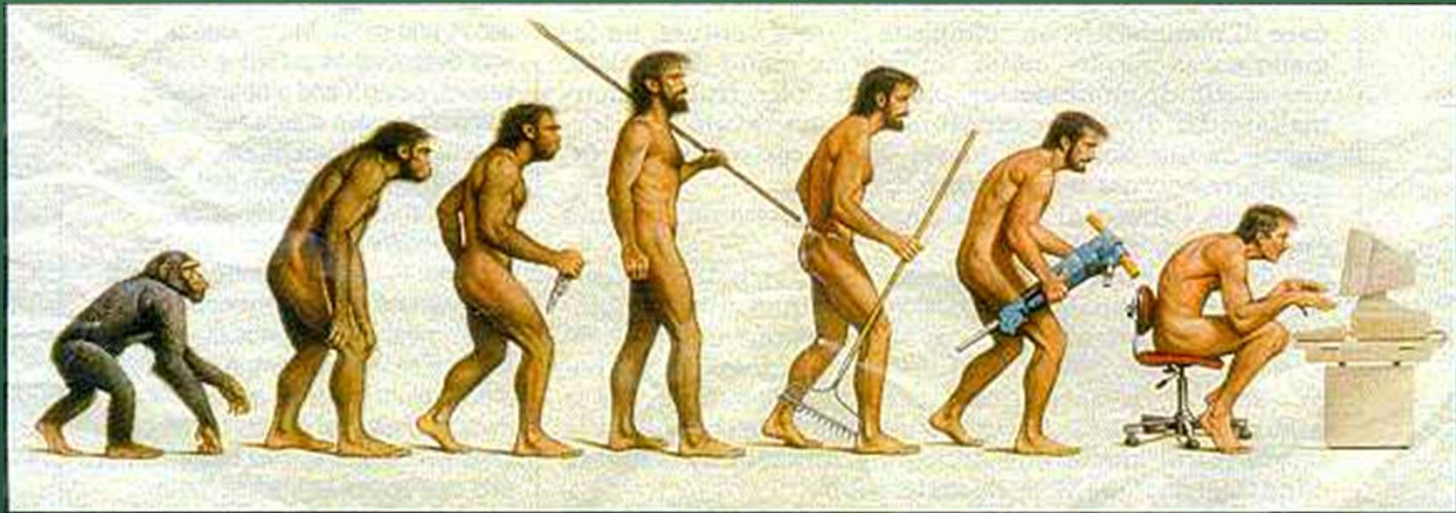


# What are people doing better?

## ADB overview



# Things you don't gotta know



# Recommendations

- Progression Systems – How good games avoid Skinner Boxes

- <https://www.youtube.com/watch?v=S5camMoNw-o>

- OpenCV Object Detection implementations

- [https://docs.opencv.org/3.4/df/dfb/group\\_imgproc\\_\\_object.html](https://docs.opencv.org/3.4/df/dfb/group_imgproc__object.html)

- RailsConf 2018: FIXME by David Heinemeier Hansson

- <https://www.youtube.com/watch?v=zKyv-ICvaCE>