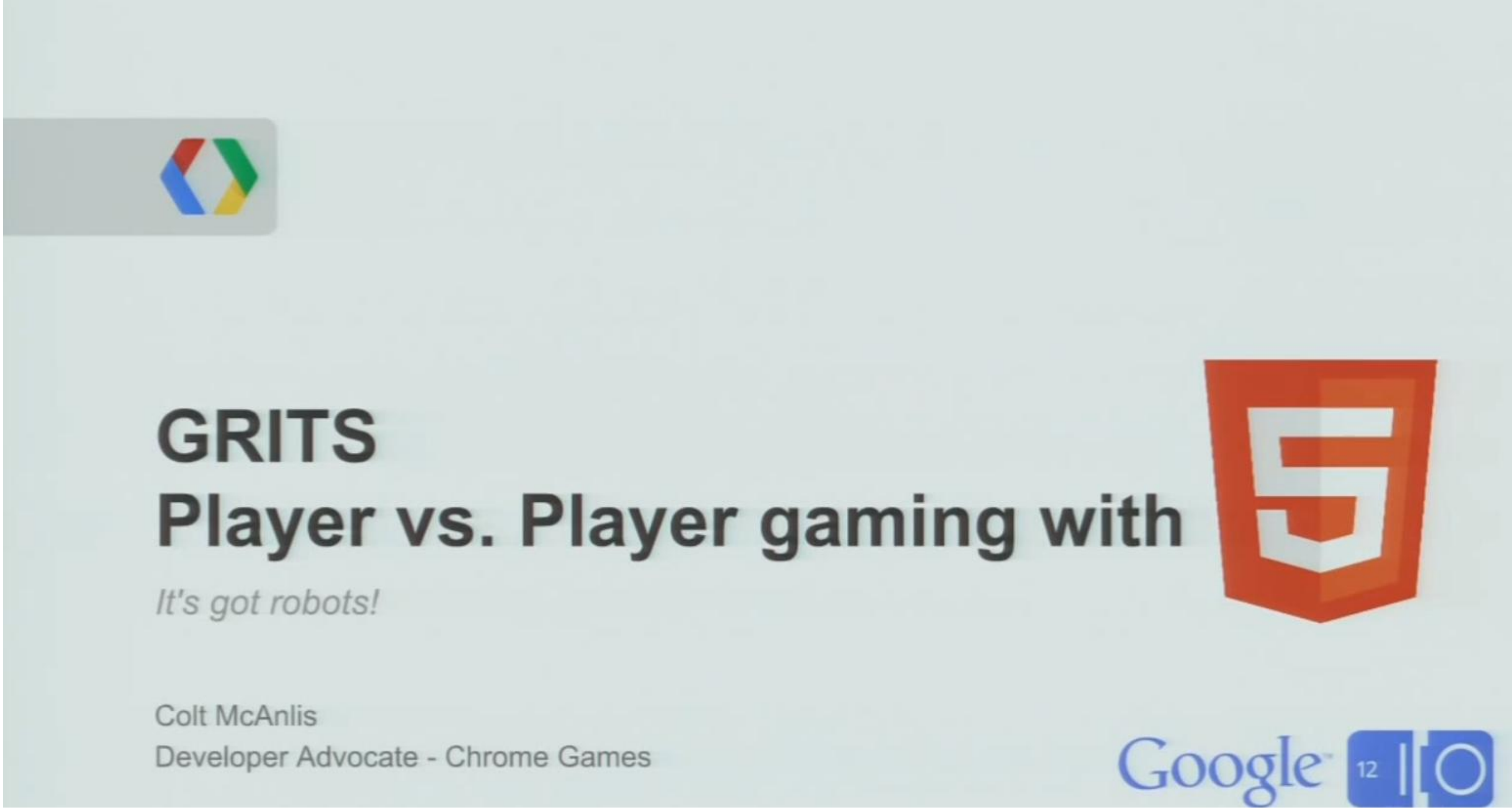


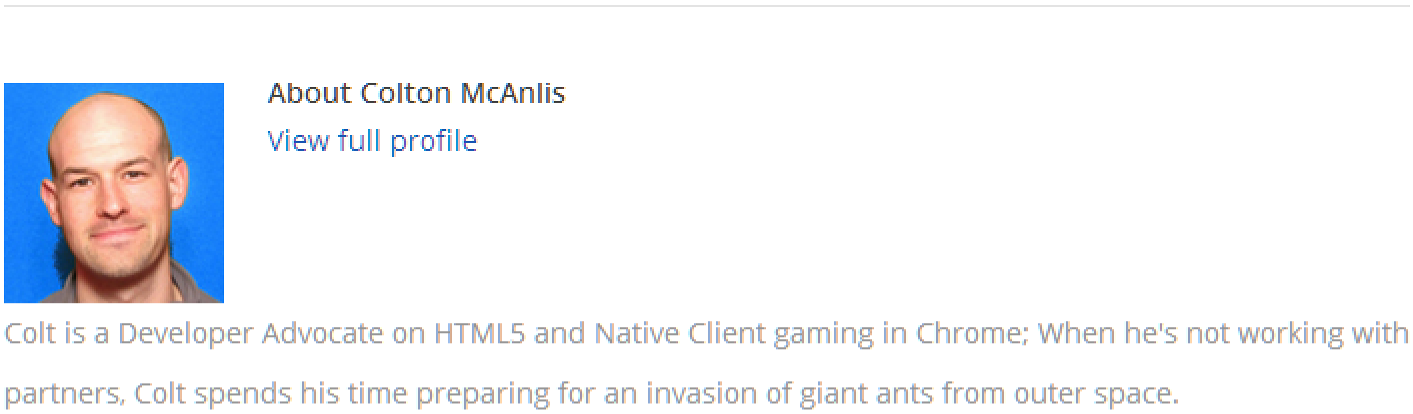
GRITS
Player vs. Player gaming with 5

It's got robots!

Colt McAnlis
Developer Advocate - Chrome Games

Google 12 I O

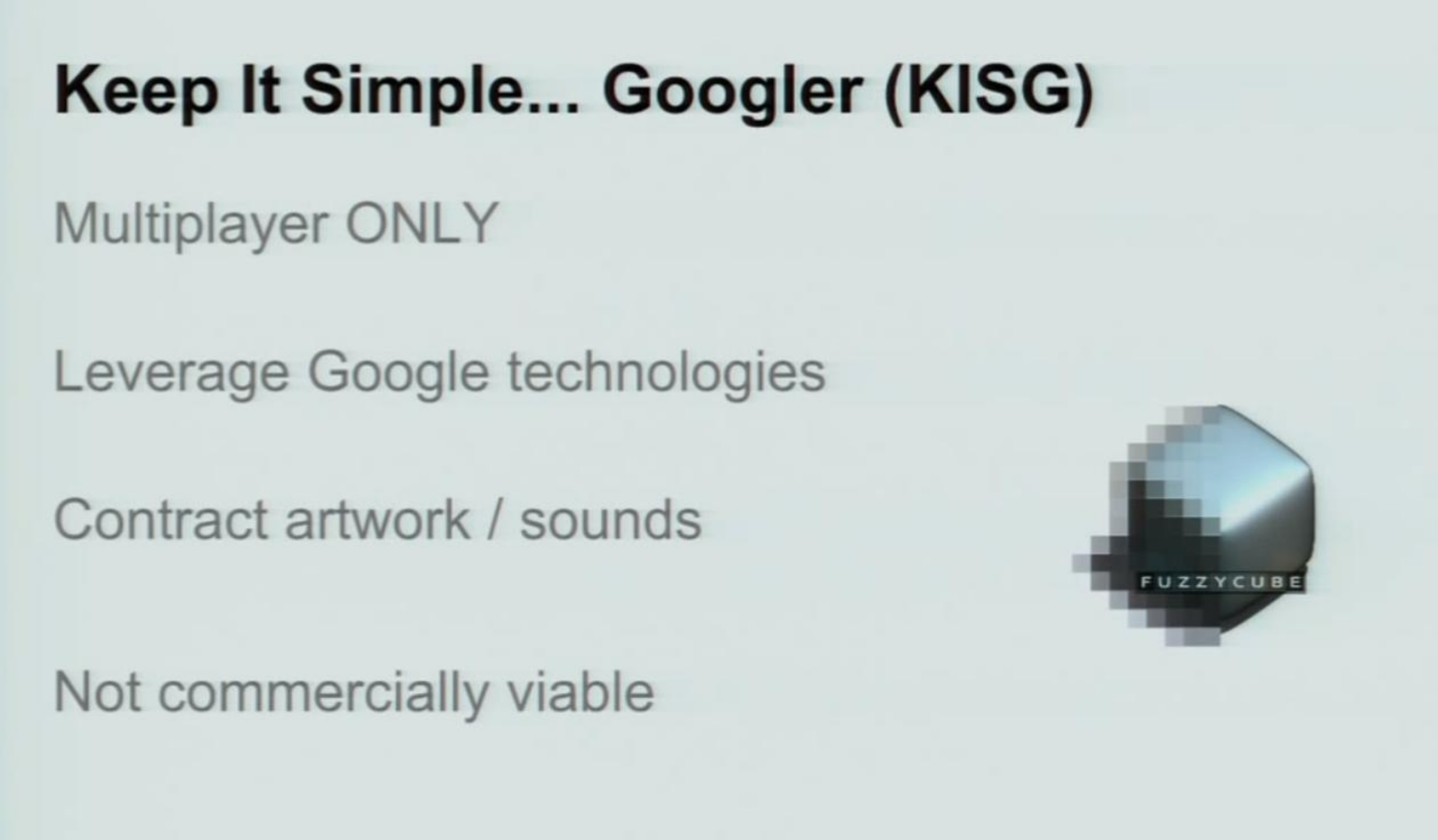
This session will present GRITS, a player vs. player shooter game, built entirely using Google technologies. In this talk, we'll walk through building an HTML5 canvas engine, serving the content, networking using Websockets, using NodeJS, social integration and more. Attendees to this session will walk away with a big-picture view of all the Google technologies that are relevant to web gaming, a deep understanding of how to get started with them, and have the ability to see them live, in action with the source code to the published game.



About Colton McAnlis

View full profile

Colt is a Developer Advocate on HTML5 and Native Client gaming in Chrome; When he's not working with partners, Colt spends his time preparing for an invasion of giant ants from outer space.



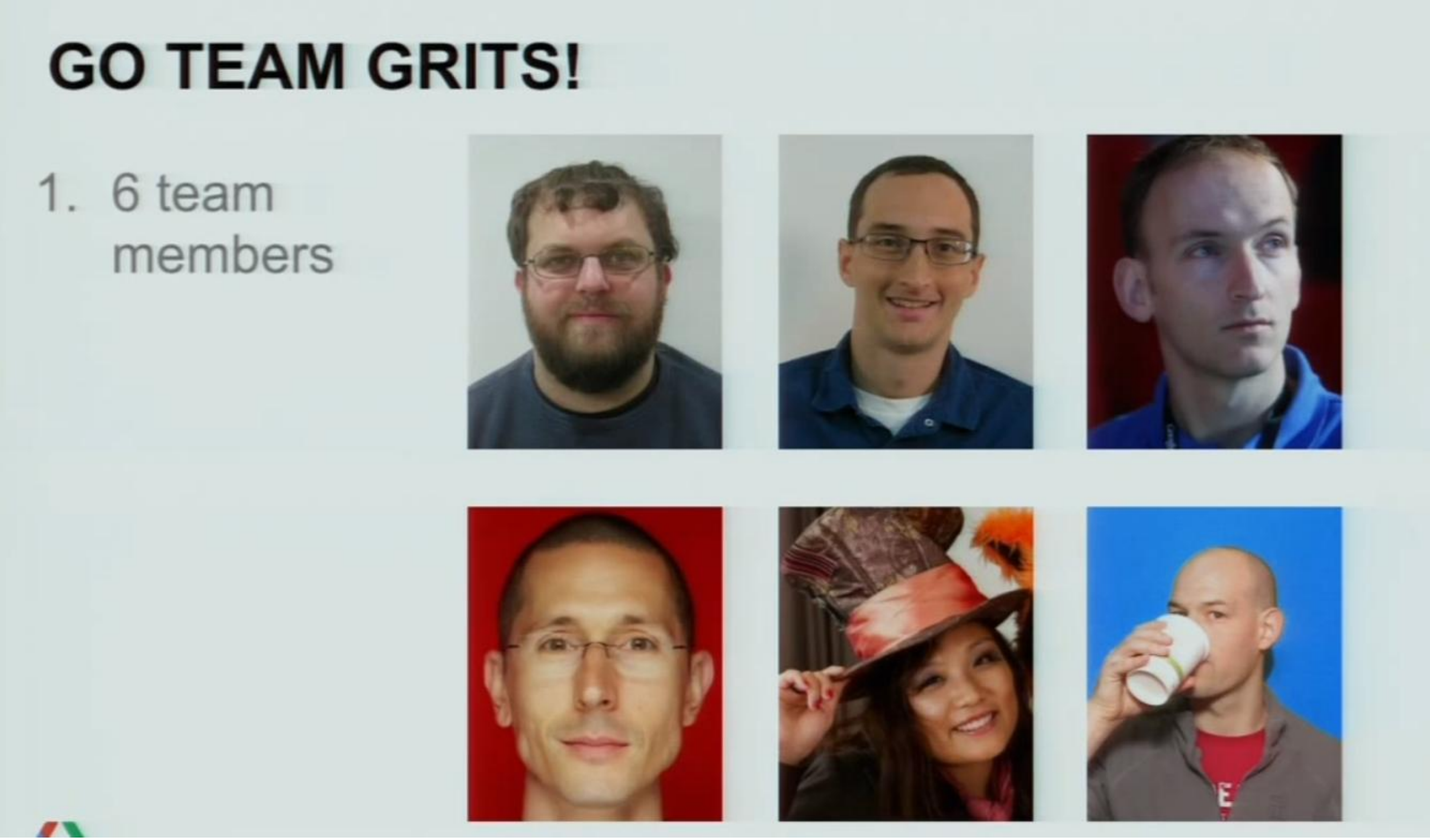
Keep It Simple... Googler (KISG)

Multiplayer ONLY

Leverage Google technologies

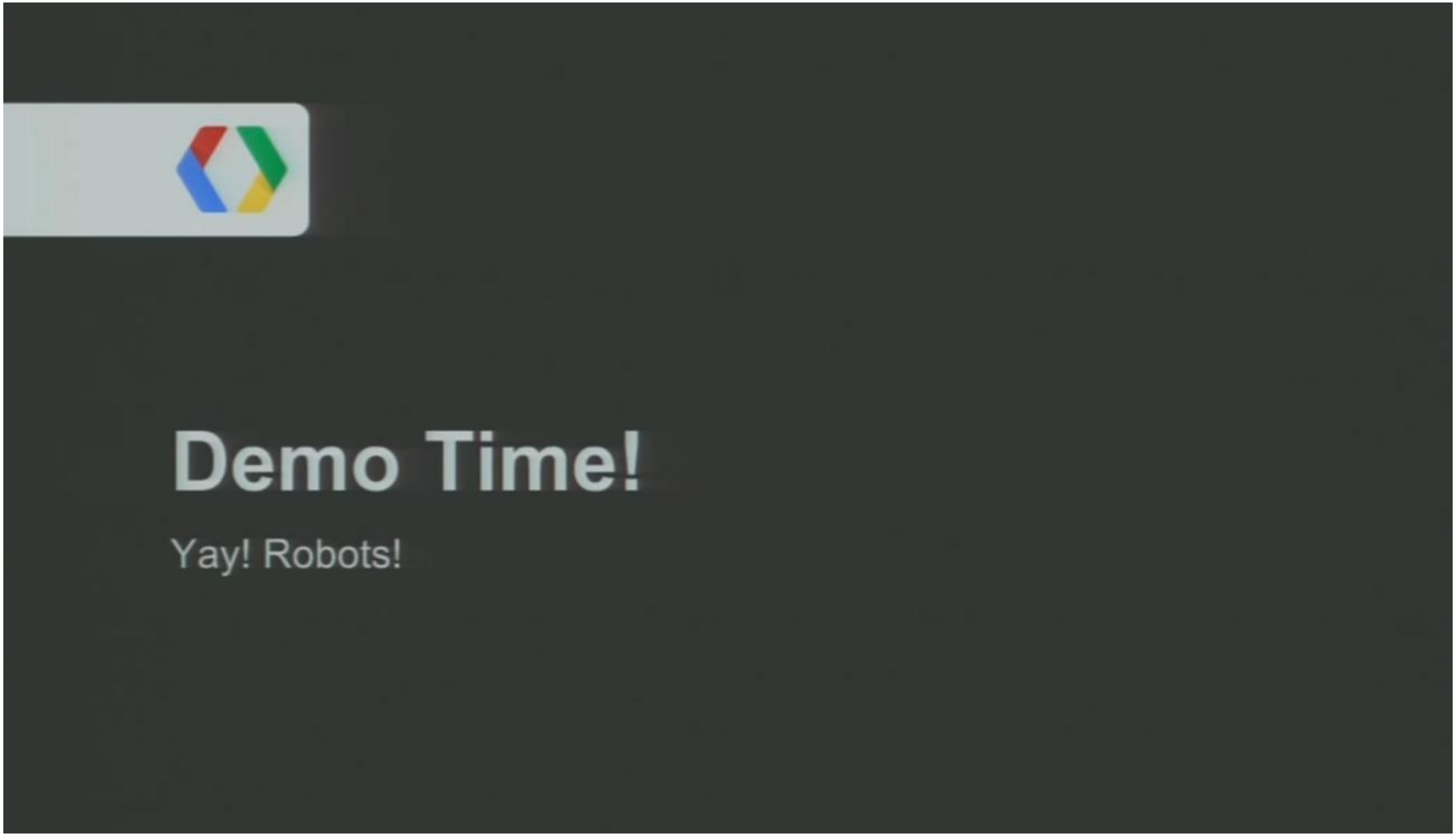
Contract artwork / sounds

Not commercially viable



GO TEAM GRITS!

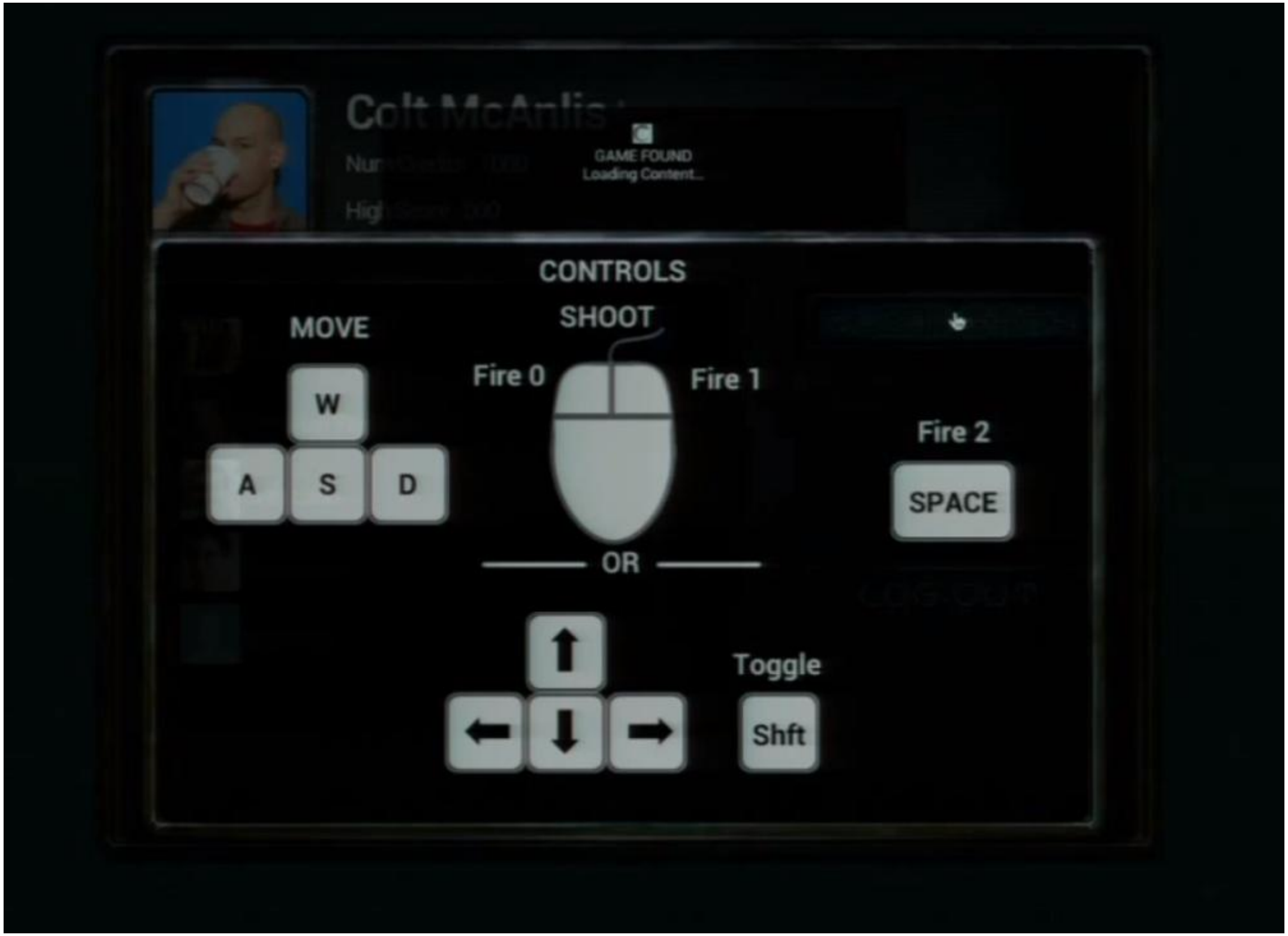
1. 6 team members



Demo Time!

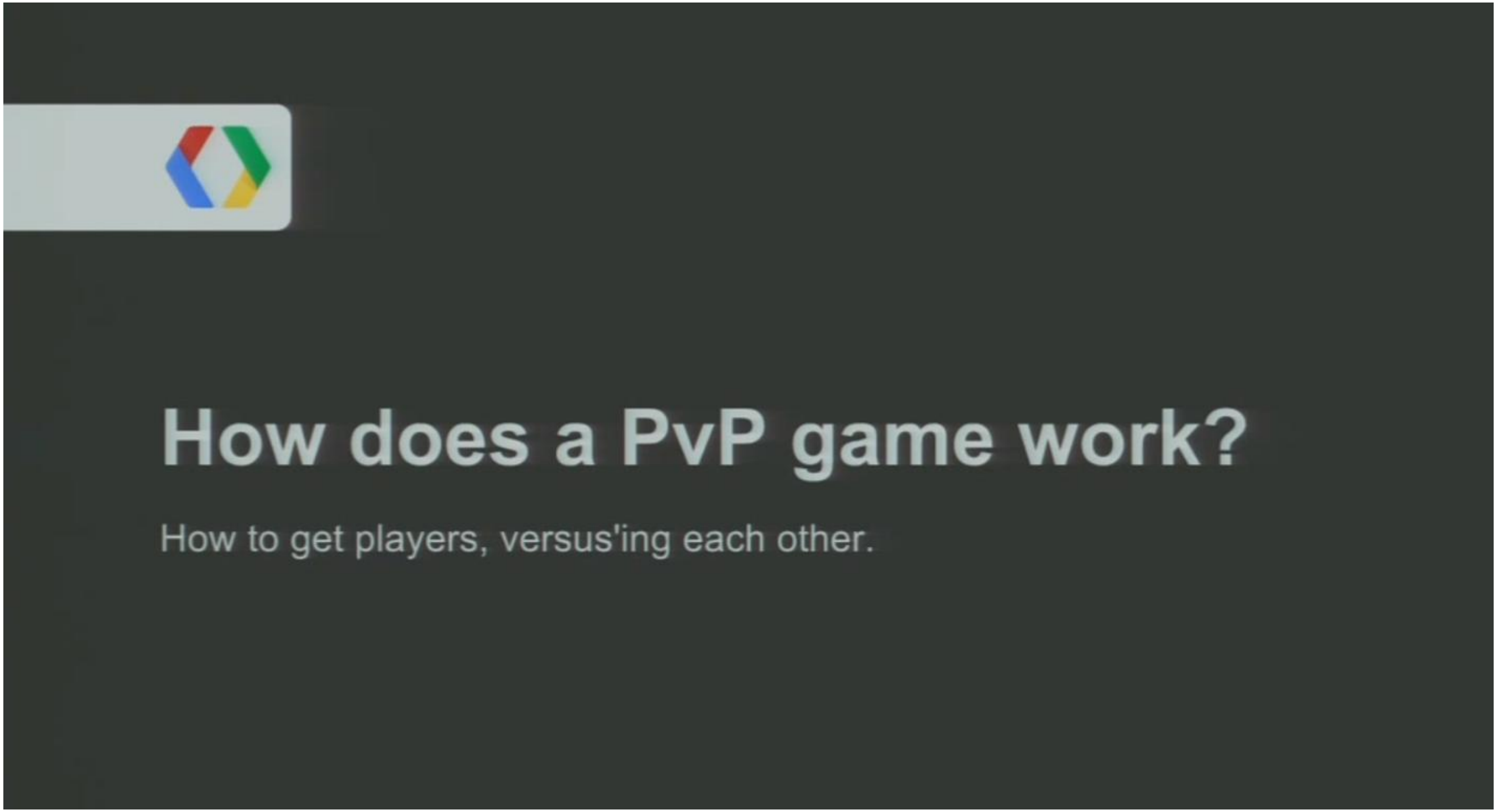
Yay! Robots!

GRITS
QUICK GAME
LOG-IN
00:00:01
-00:01:41



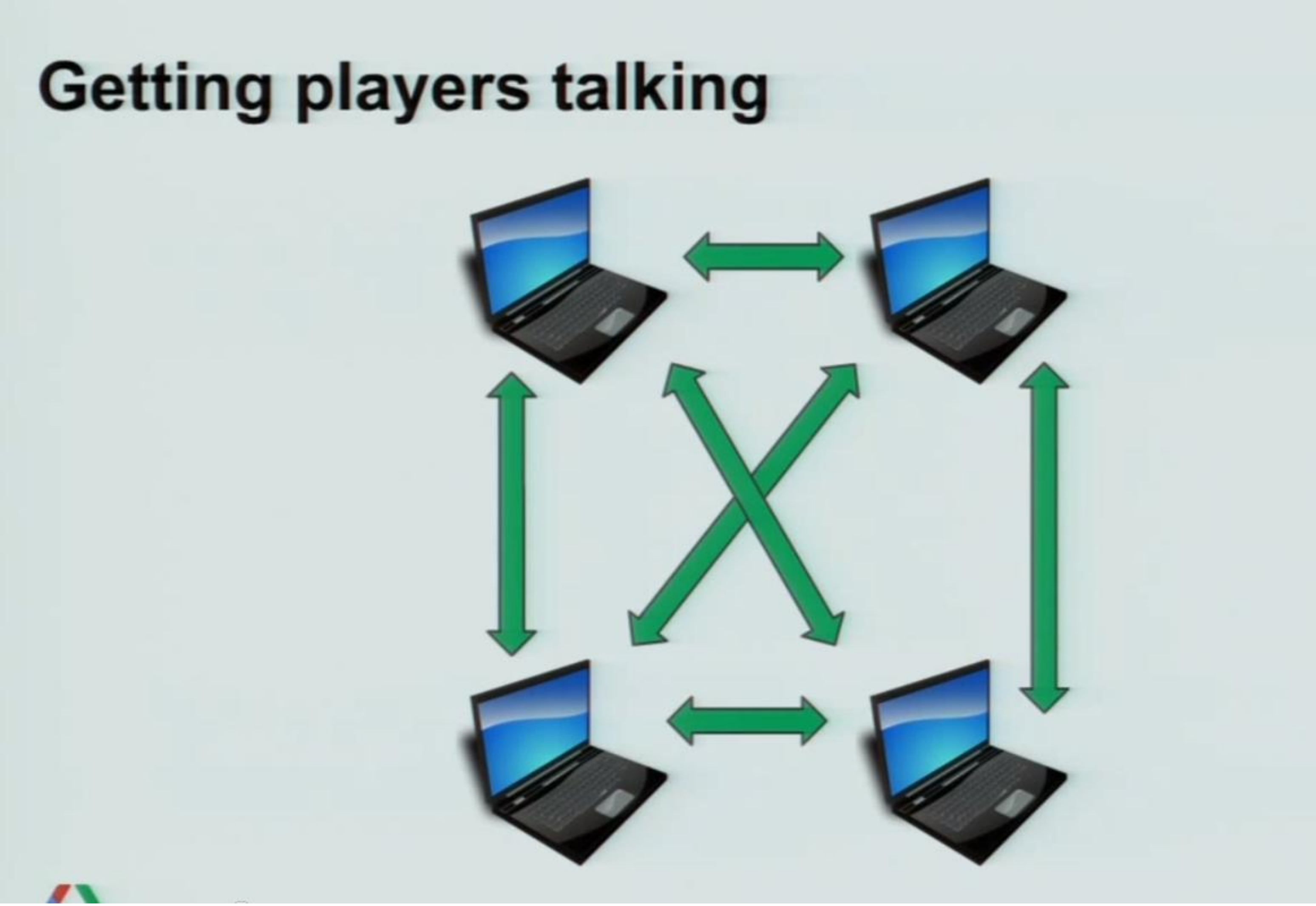
GAME FOUND
Loading Content...

OL WUT?
bot-1593
bot-6351

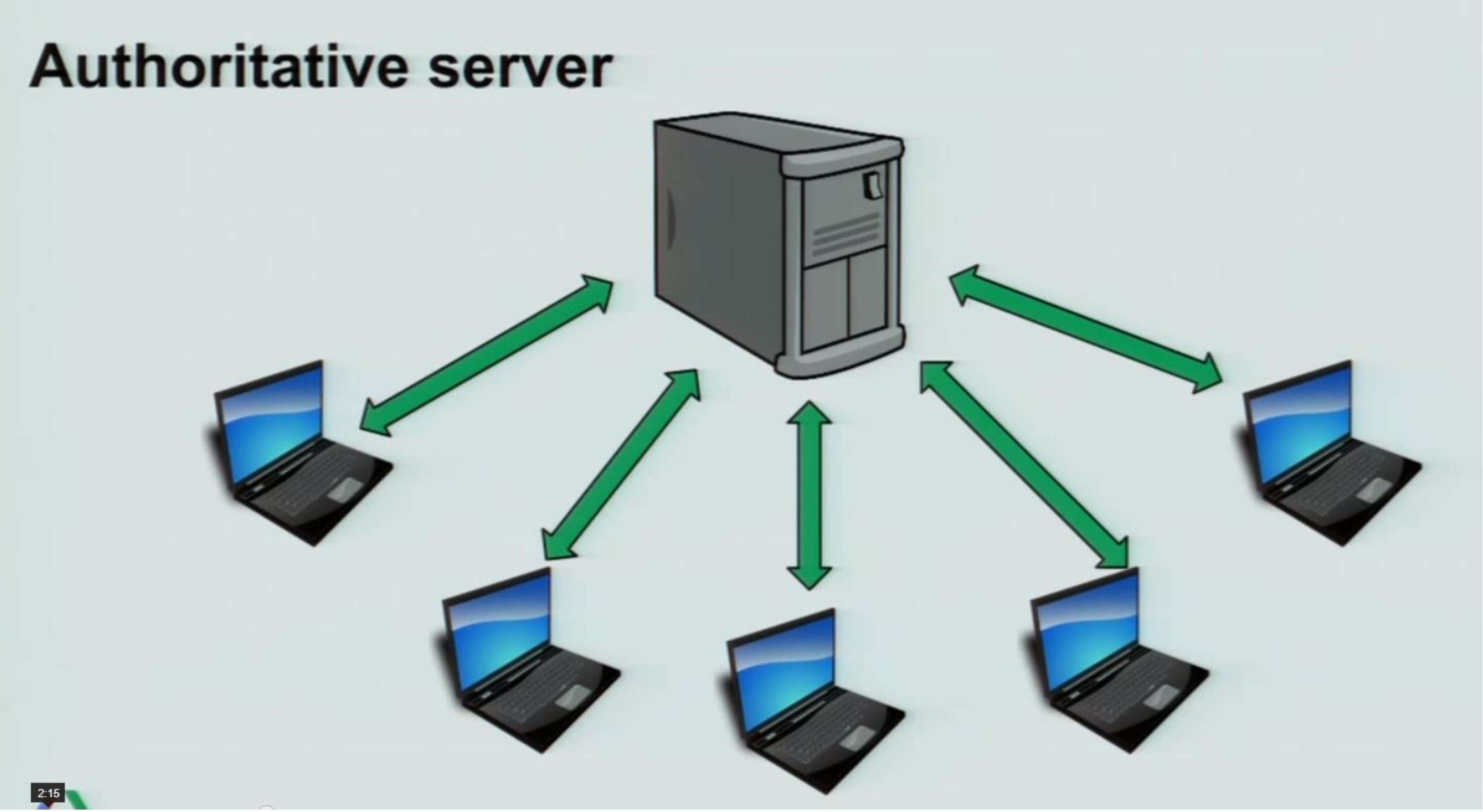


How does a PvP game work?

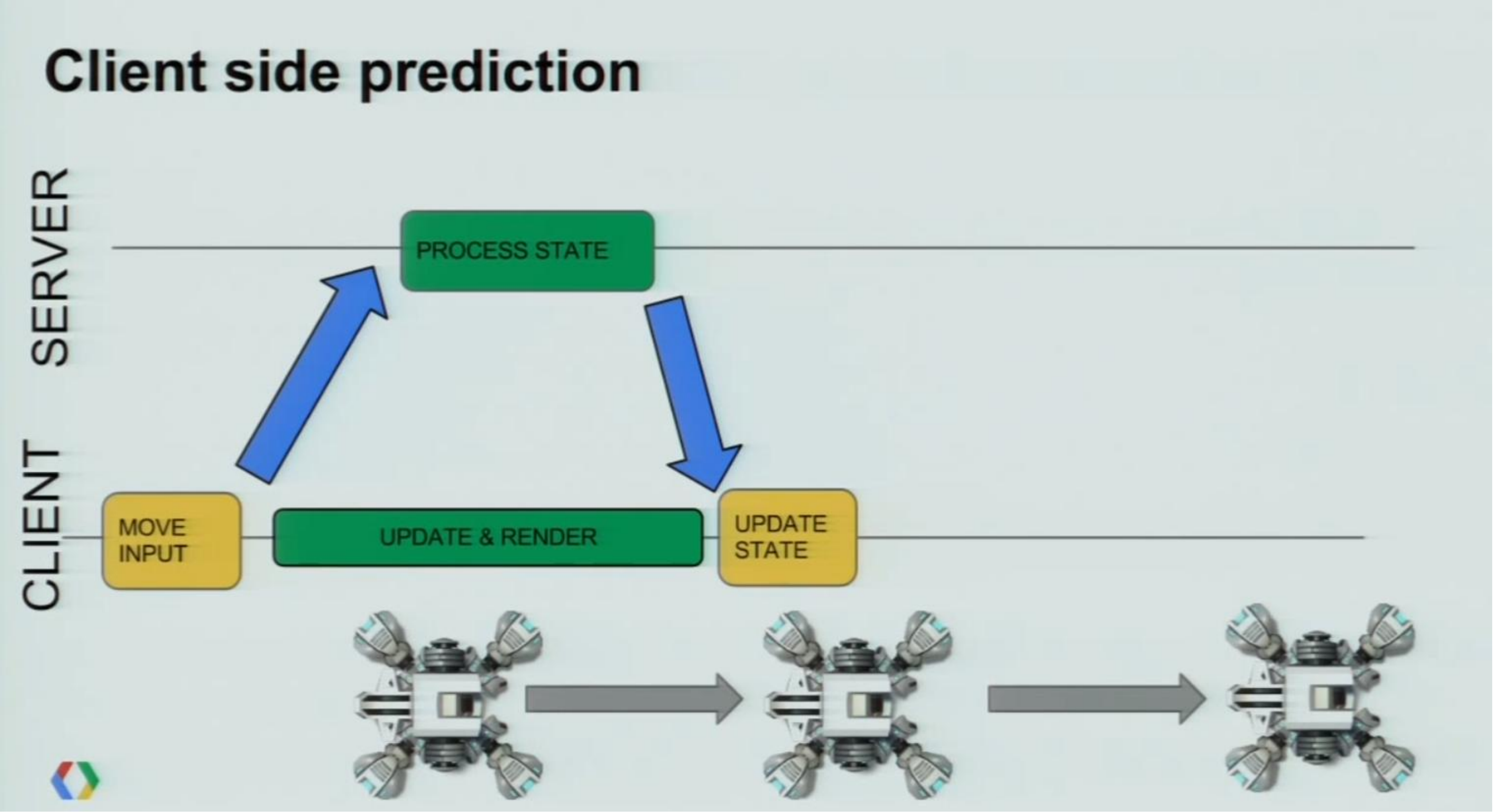
How to get players, versus'ing each other.



Getting players talking



Authoritative server



Client side prediction

SERVER

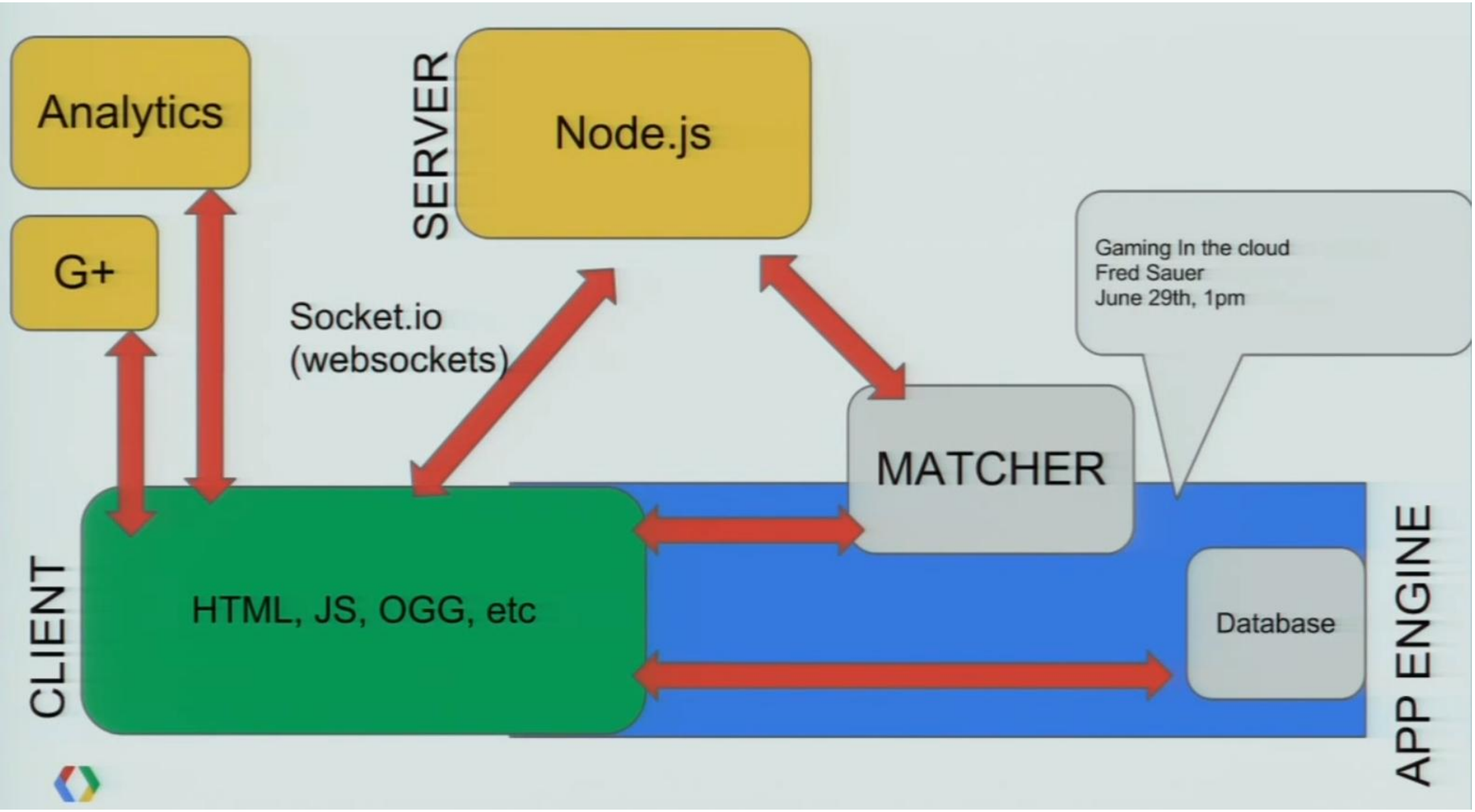
CLIENT

PROCESS STATE

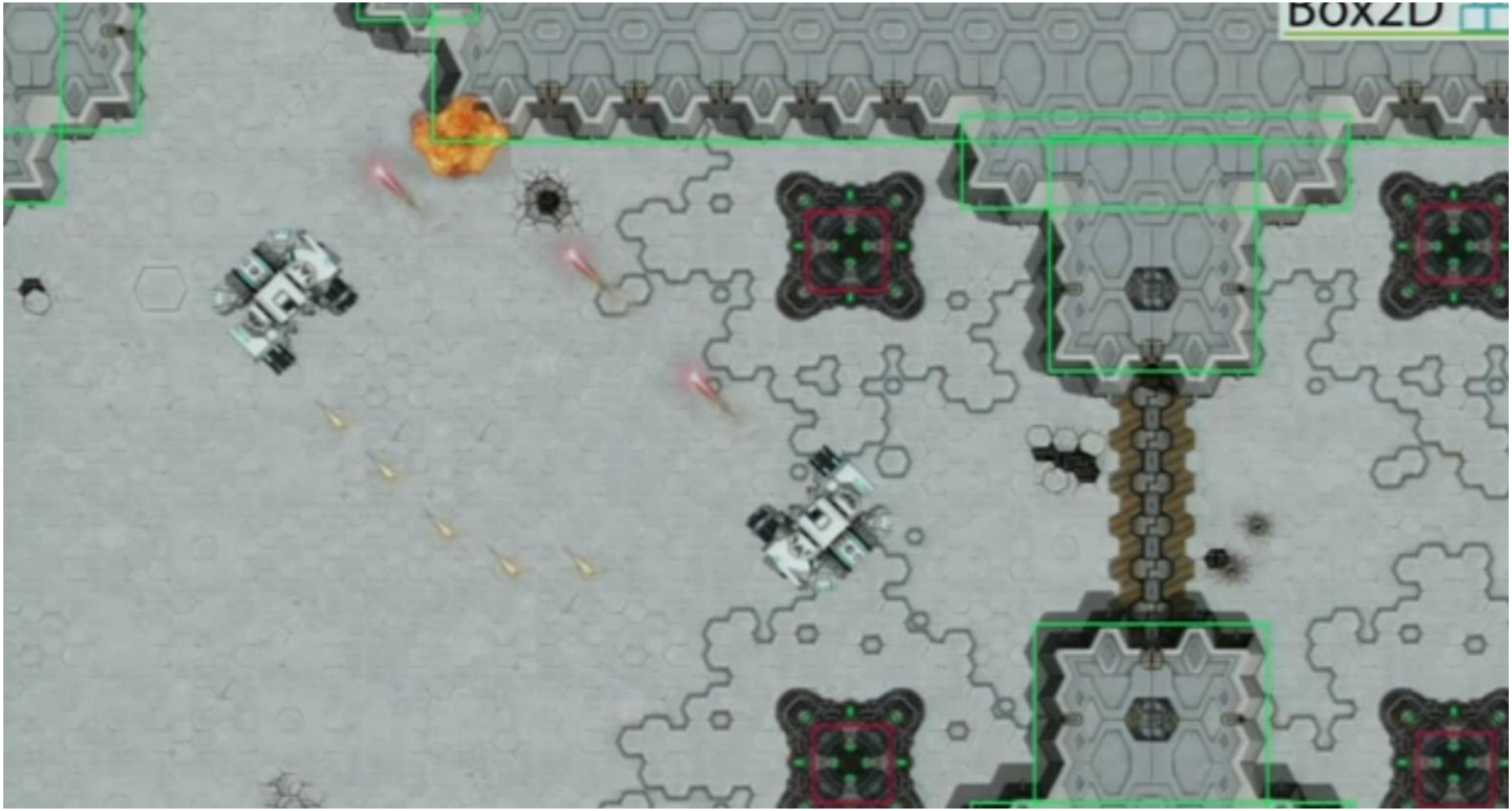
MOVE INPUT

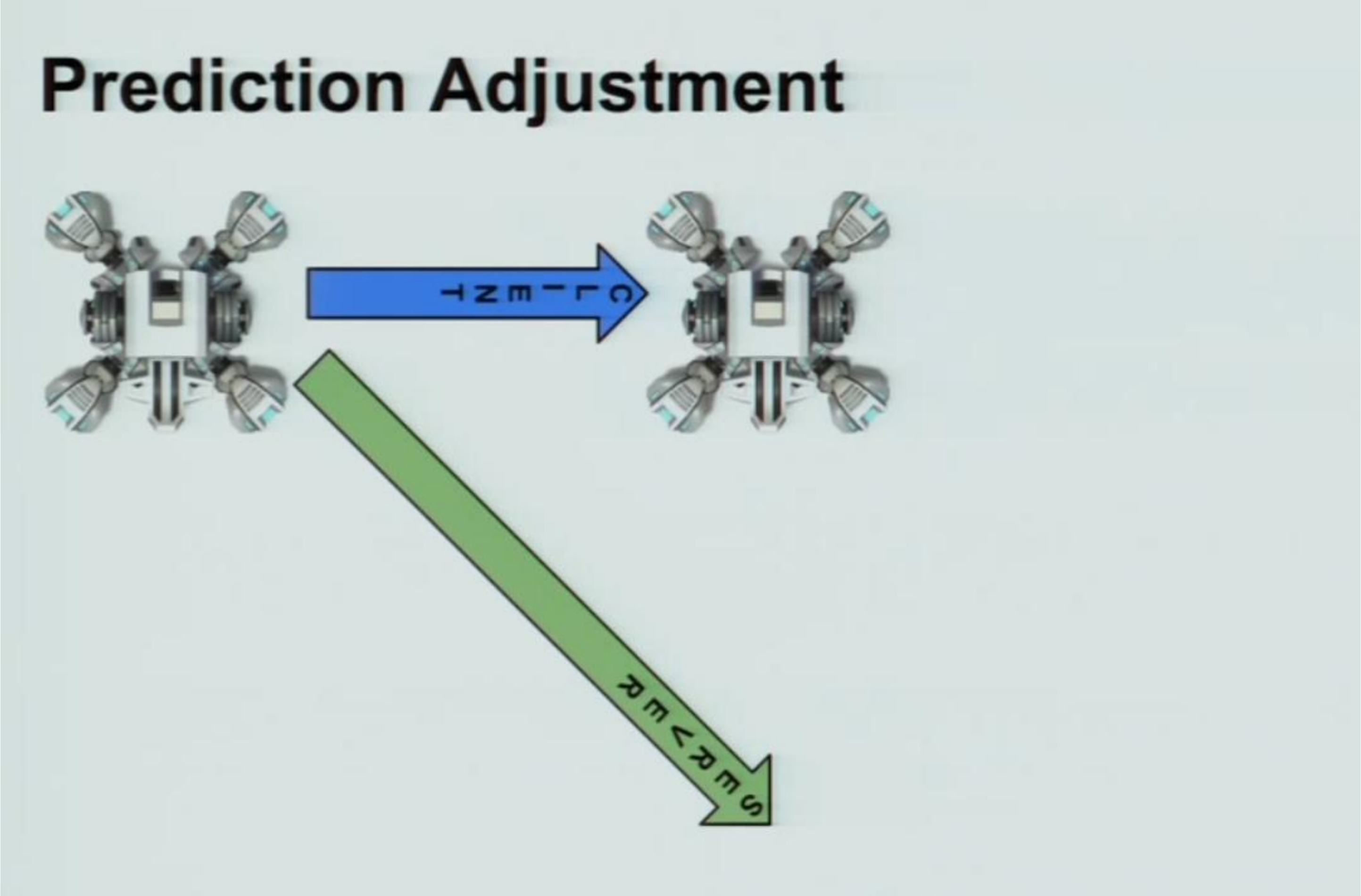
UPDATE & RENDER

UPDATE STATE



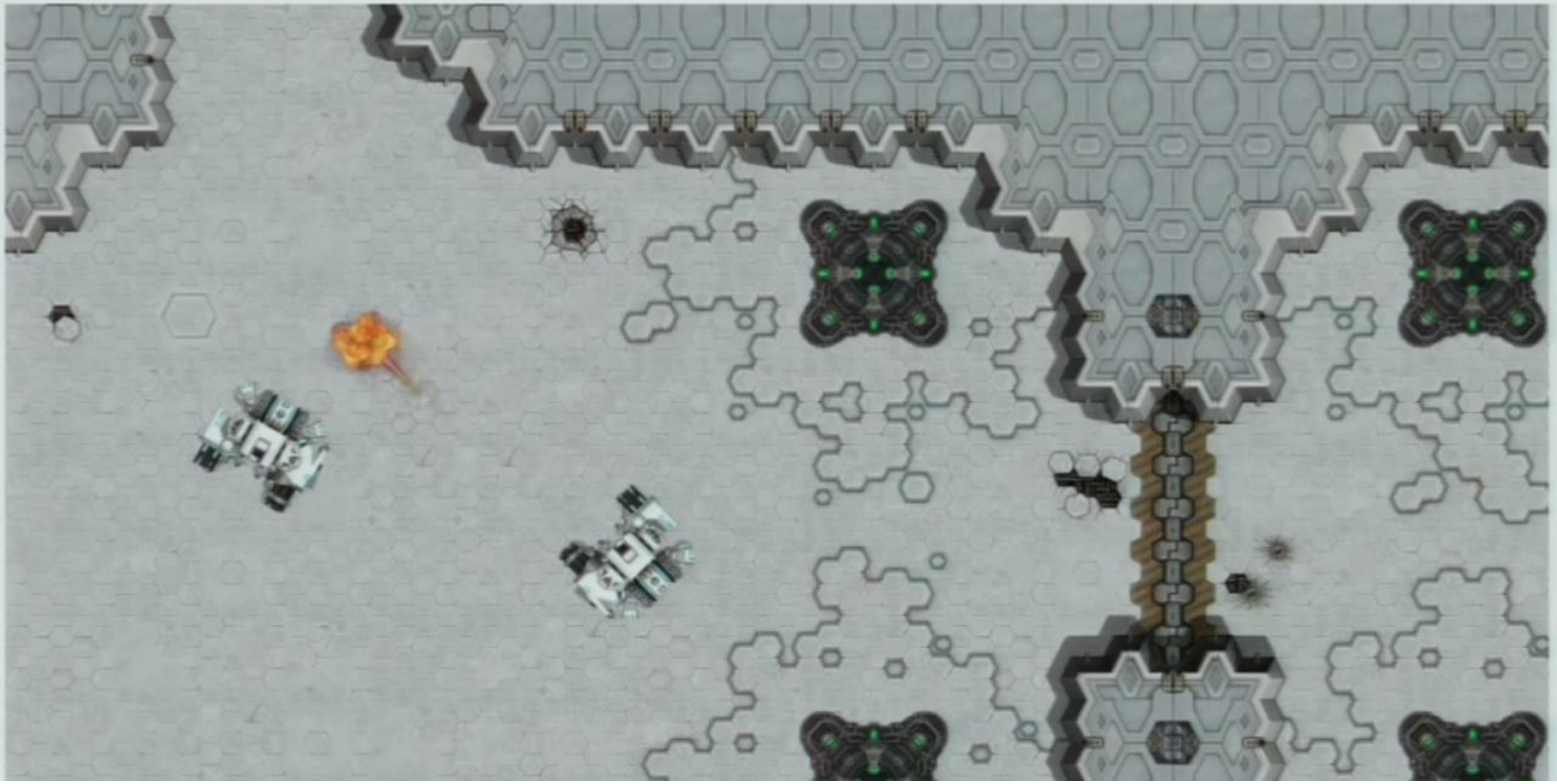
Gaming In the cloud
Fred Sauer
June 29th, 1pm


Box2D



Prediction Adjustment

Authoritative server



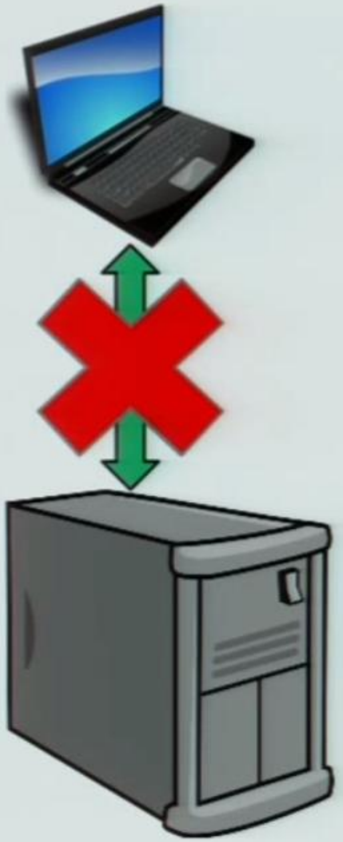
Networking GRITS

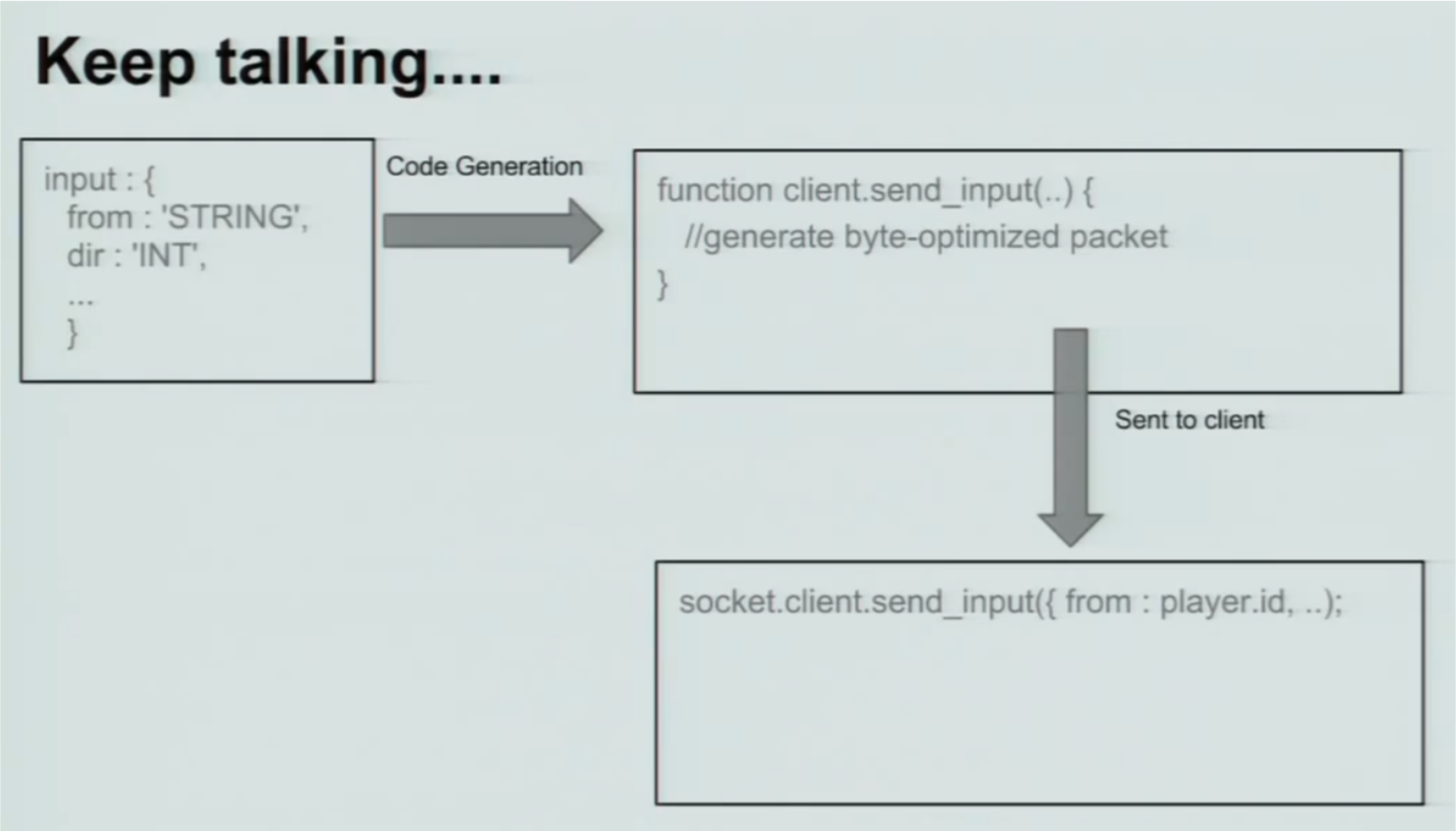
It's not *who* you know, it's what technology stack allows you to talk to them.

Using packets in C++

```
struct input
{
  uchar pktType;
  uint32 from;
  float32 dir[2];
}
```

```
struct input_v3
{
  uint32 from;
  float32 dir_x;
  float32 dir_y;
}
```



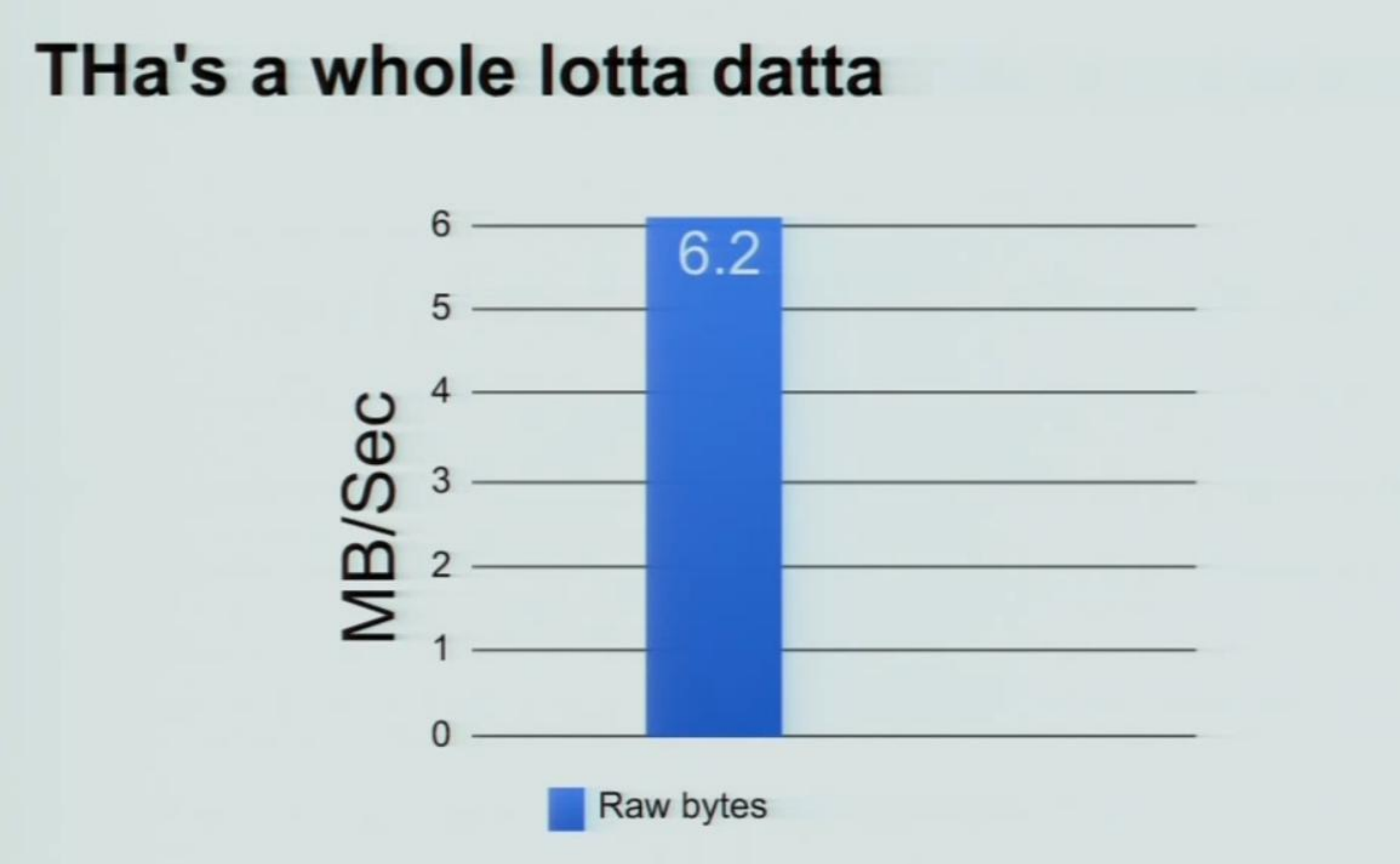


Keep talking....

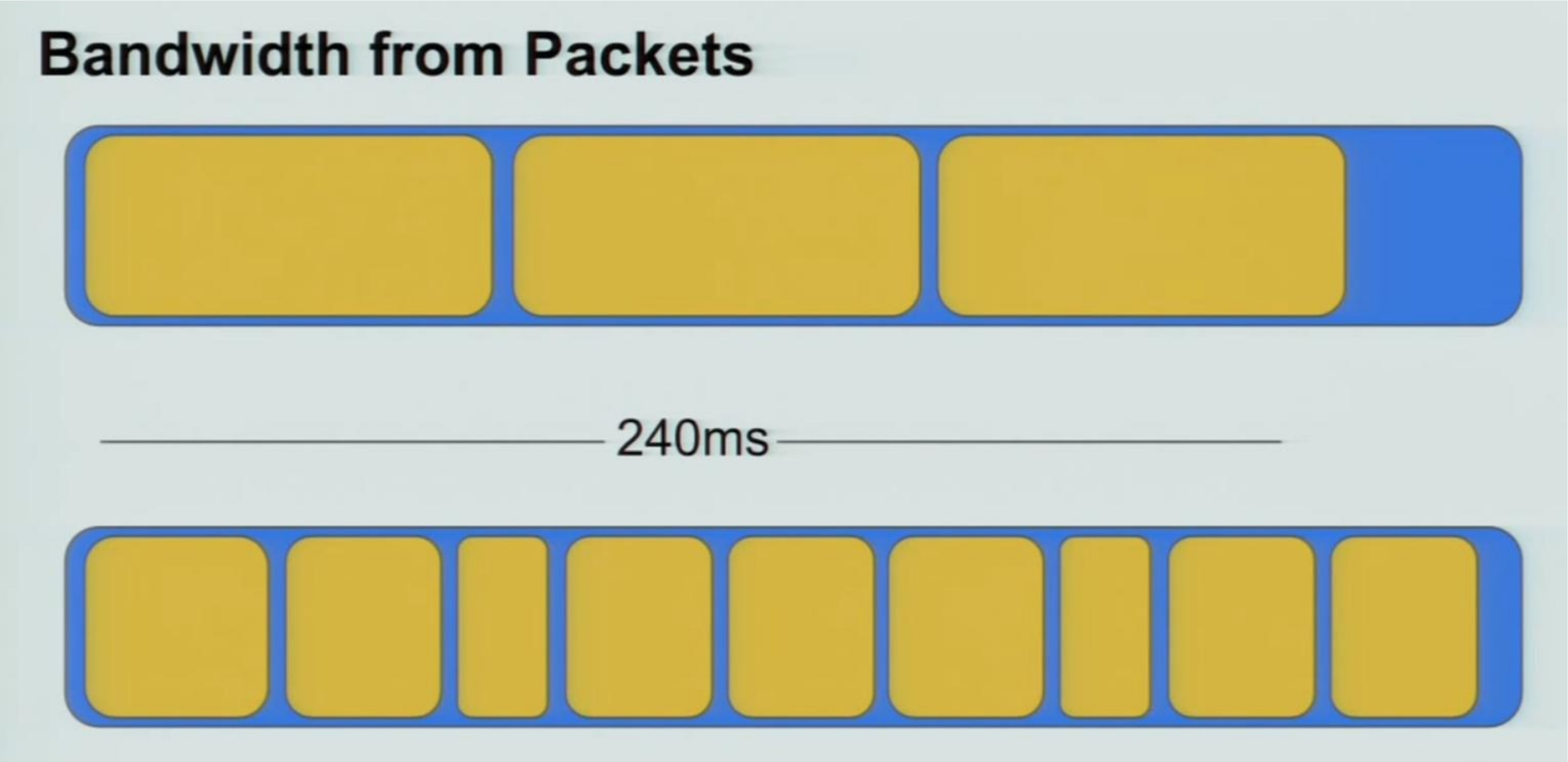
```
input : {
  from : 'STRING',
  dir : 'INT',
  ...
  }
```

```
function client.send_input(..) {
  //generate byte-optimized packet
}
```

```
socket.client.send_input({ from : player.id, ..);
```

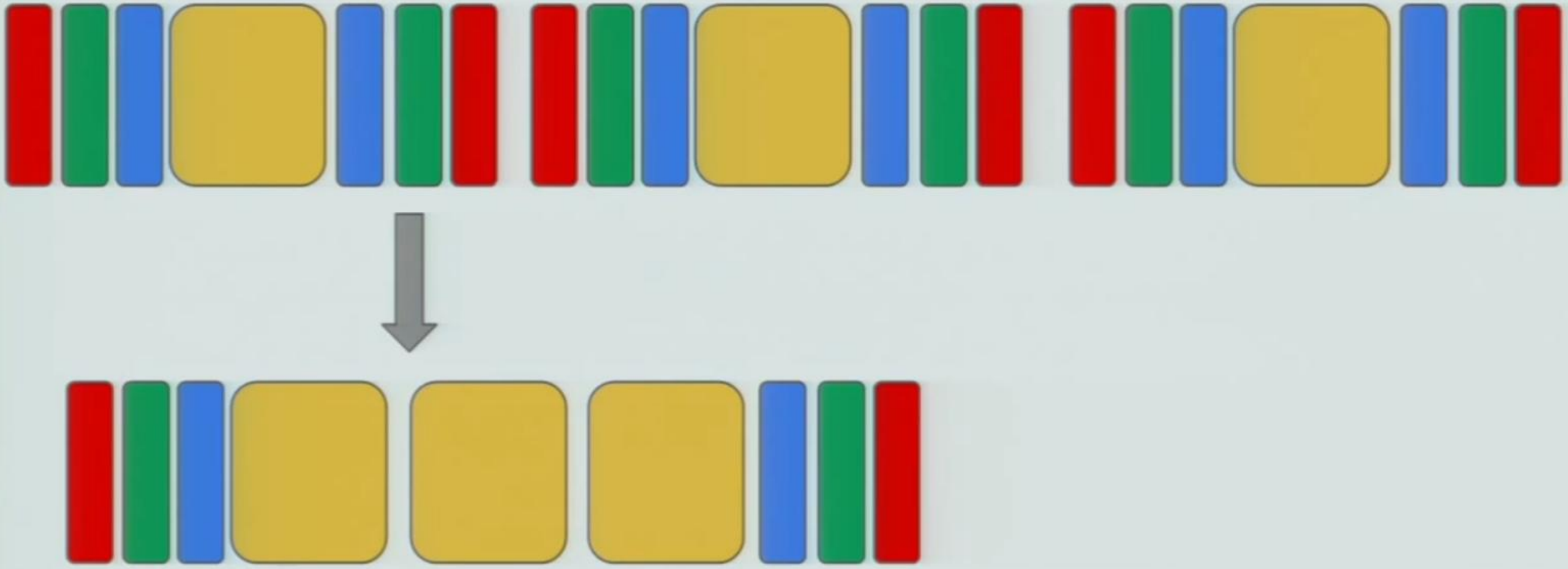


THa's a whole lotta datta

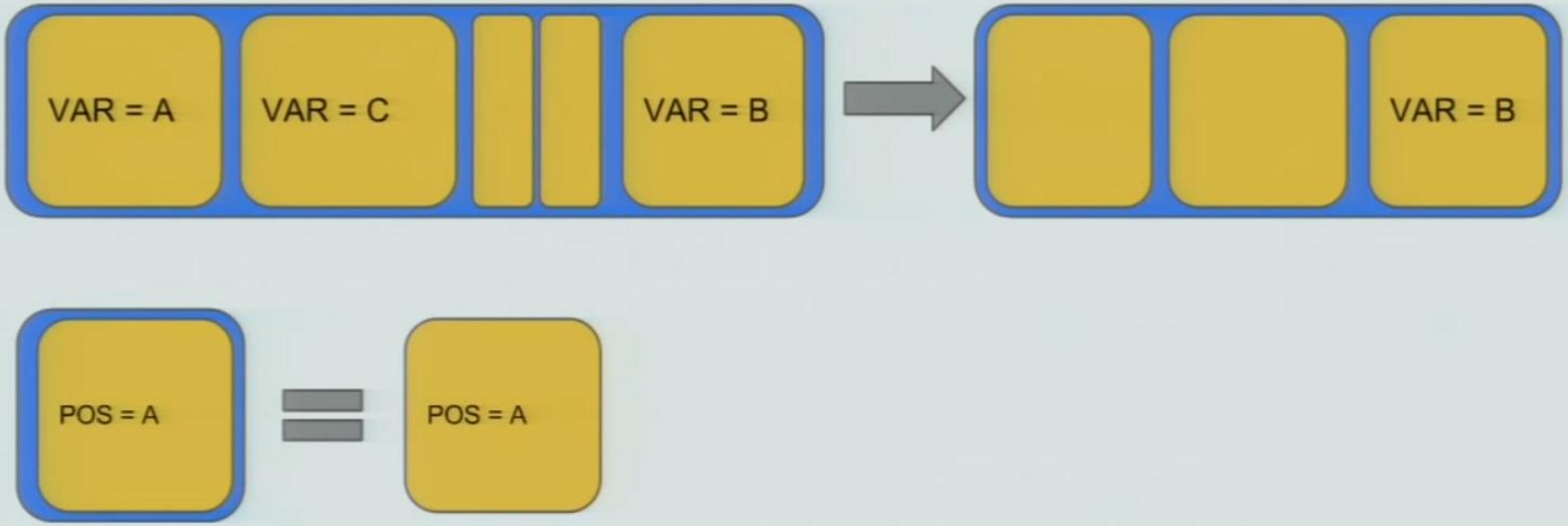


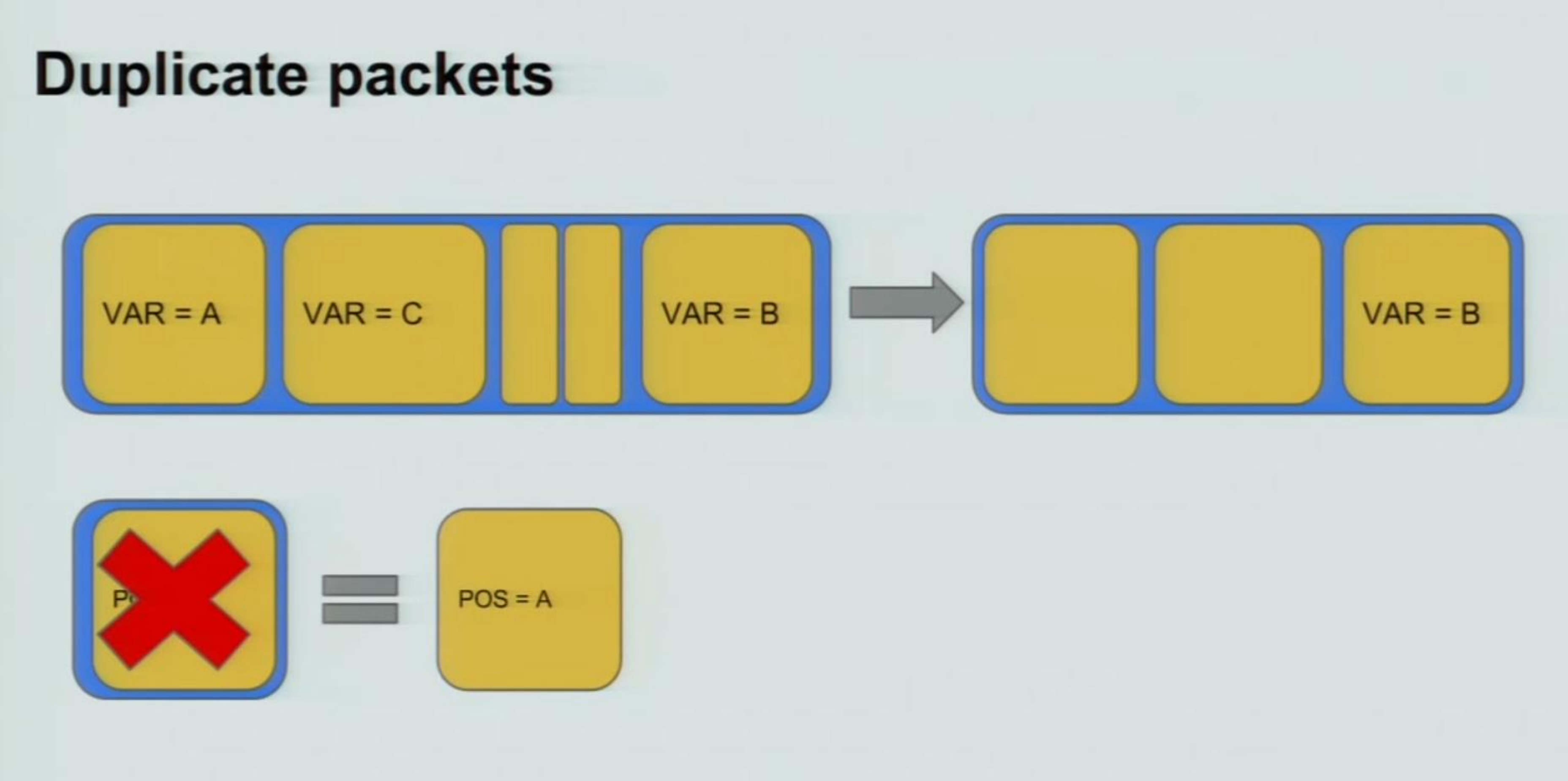
Bandwidth from Packets

Packet grouping



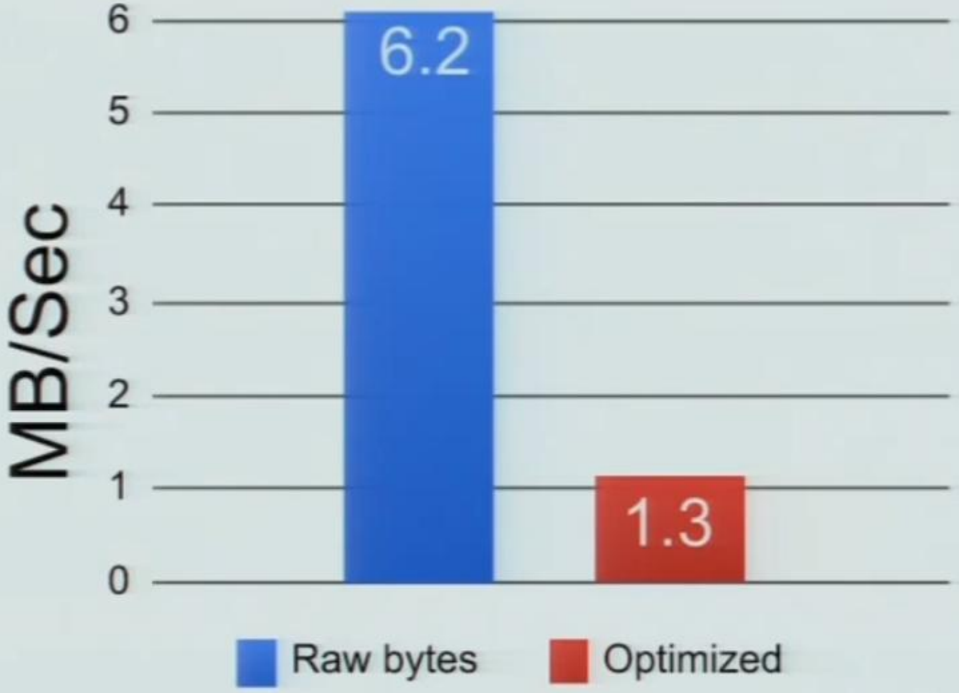
Duplicate packets



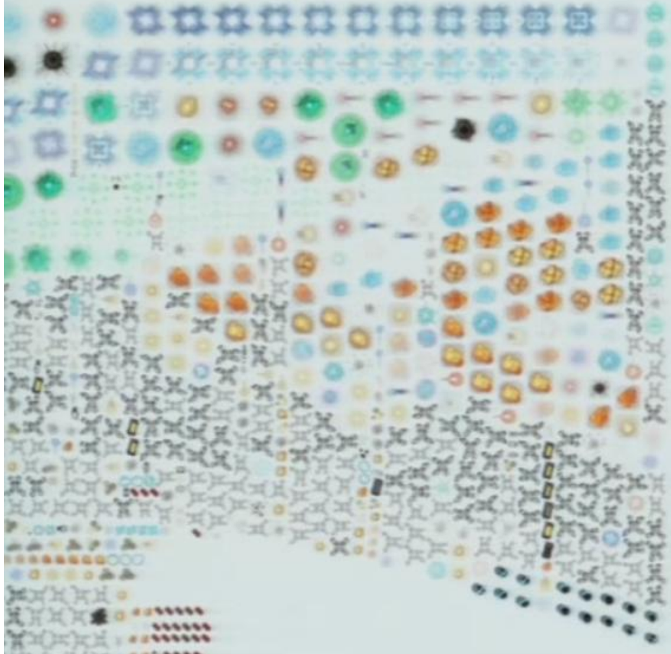


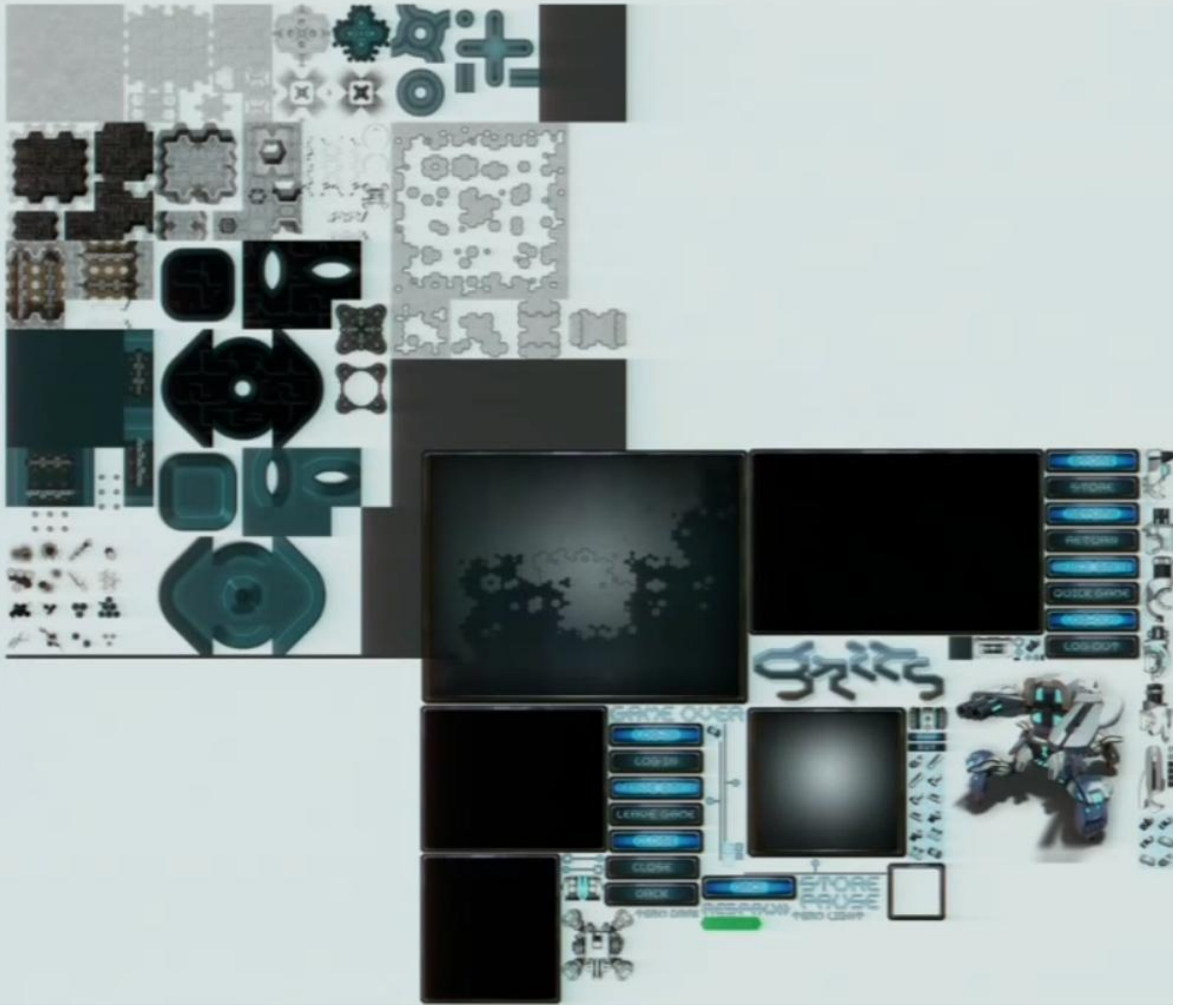
Duplicate packets

Shoop da dupe



Atlas the planet





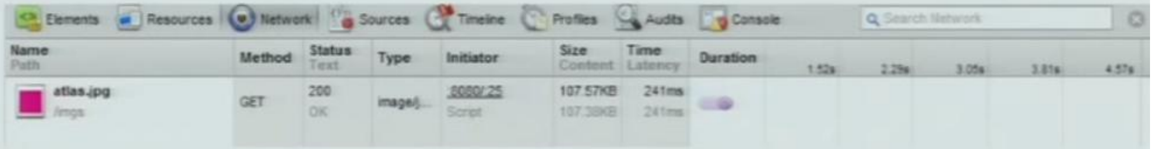
Use an Atlas, save the world.

1 atlas request

4096x4096 pixels

107.57kb total

247ms total

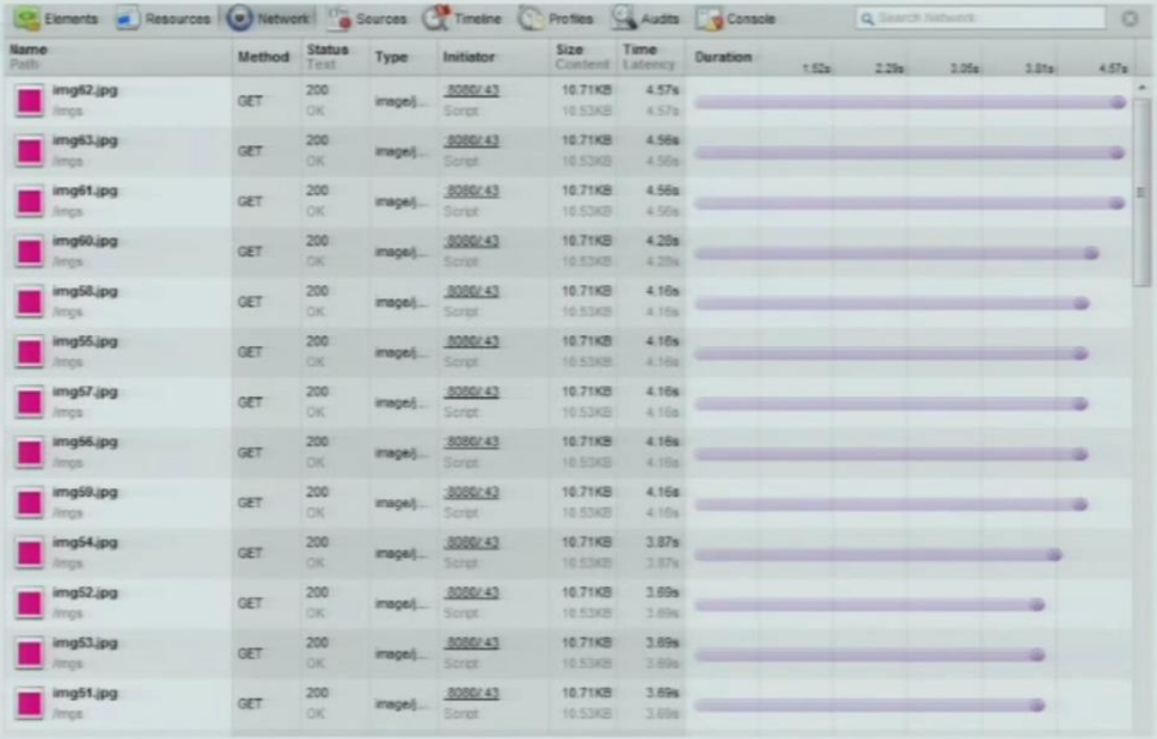


Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Duration	1.52s	2.29s	3.05s	3.81s	4.57s
atlas.jpg /imgs	GET	200 OK	image/j...	:8080/:25 Script	107.57KB 107.38KB	241ms 241ms						

Use an Atlas, save the world.

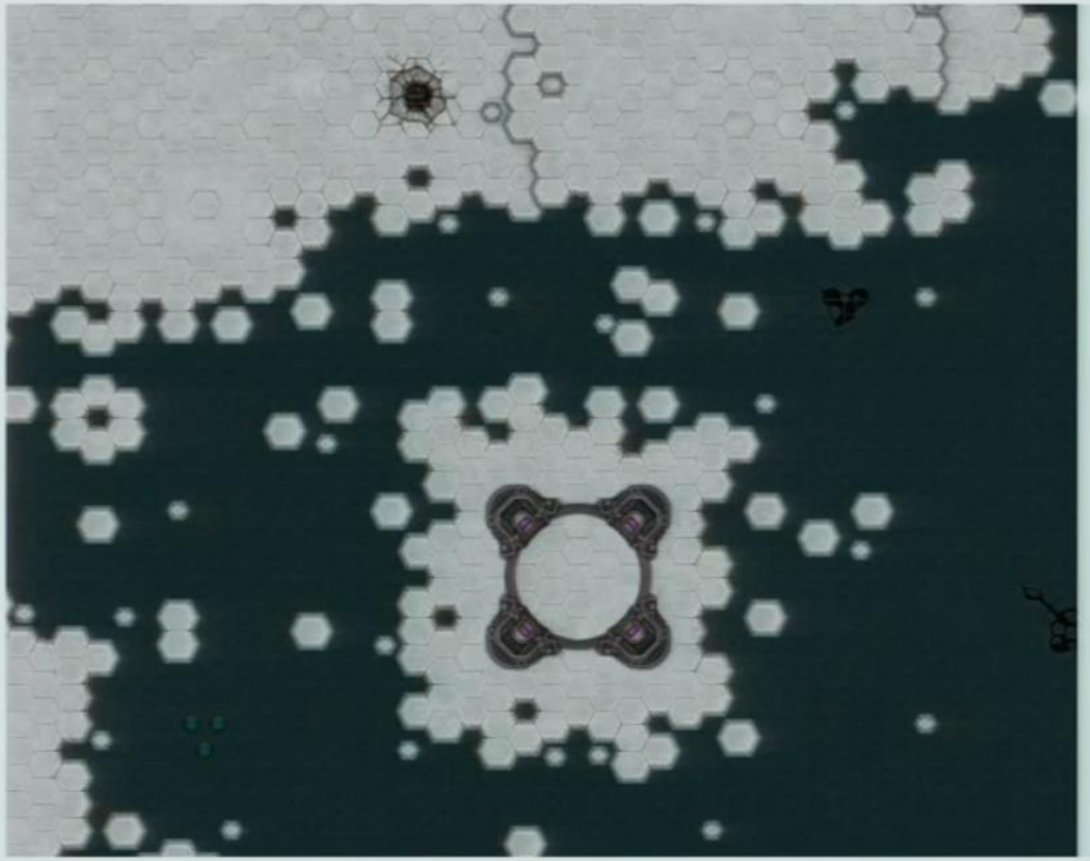
4096 individual requests

- **10.71kb** each
- **685.55k** total
- **~4.63** seconds total

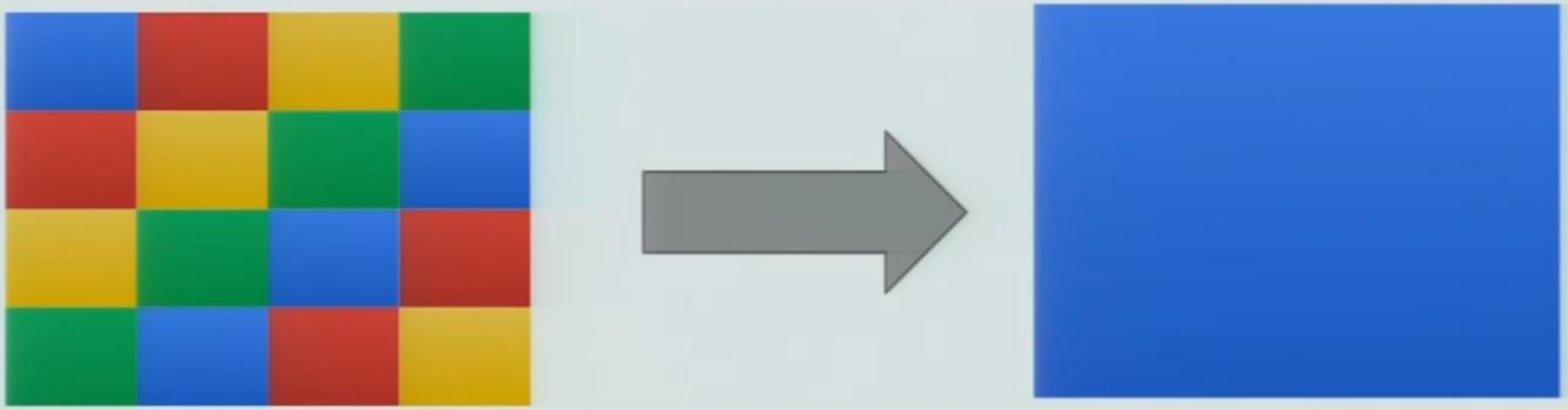


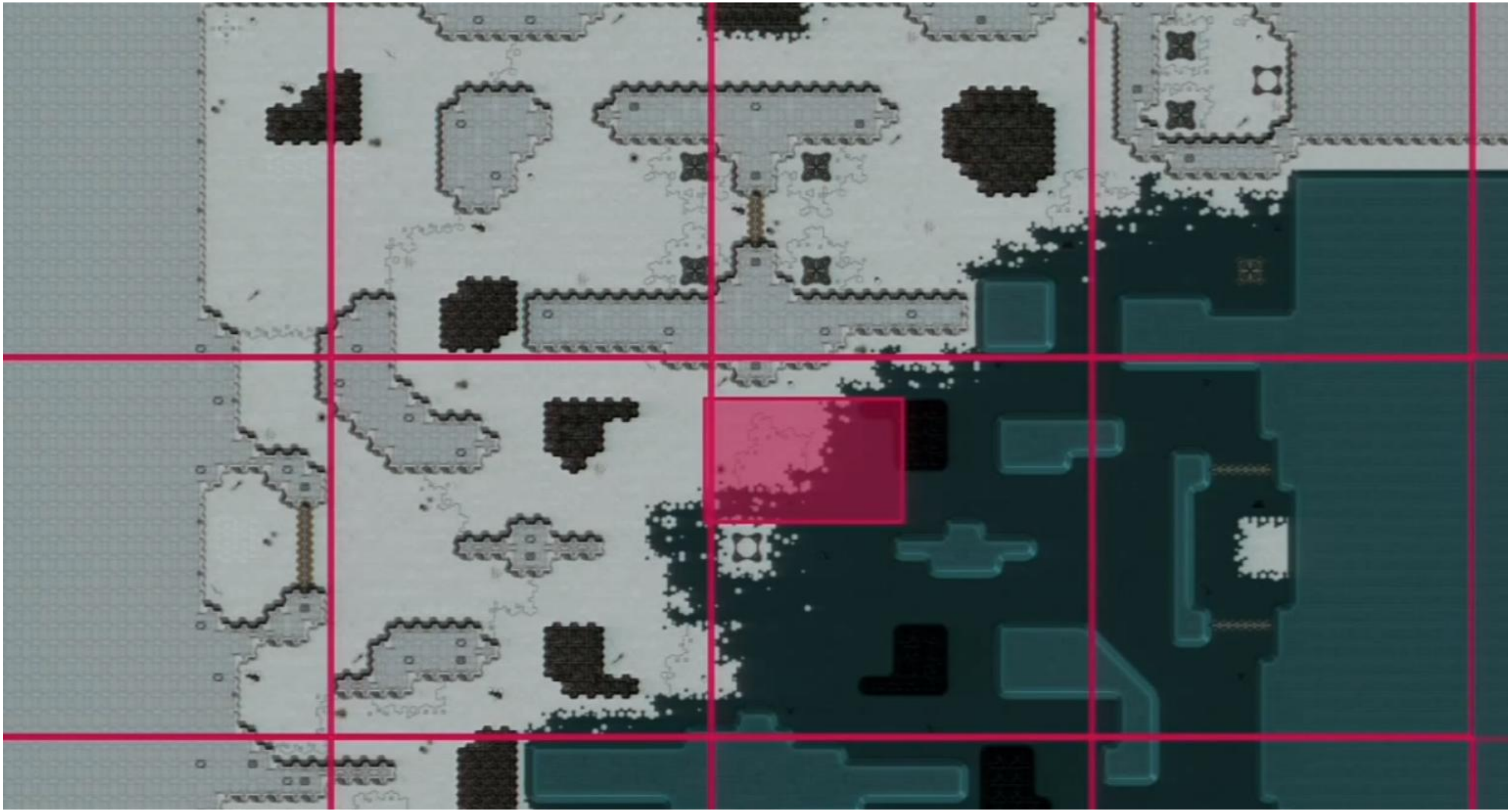
Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Duration
img62.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.57s 4.57s	
img63.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.56s 4.56s	
img61.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.56s 4.56s	
img60.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.28s 4.28s	
img58.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.16s 4.16s	
img55.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.16s 4.16s	
img57.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.16s 4.16s	
img56.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.16s 4.16s	
img59.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	4.16s 4.16s	
img54.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	3.87s 3.87s	
img52.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	3.69s 3.69s	
img53.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	3.69s 3.69s	
img51.jpg /imgs	GET	200 OK	image/j...	:8080/:43 Script	10.71KB 10.53KB	3.69s 3.69s	

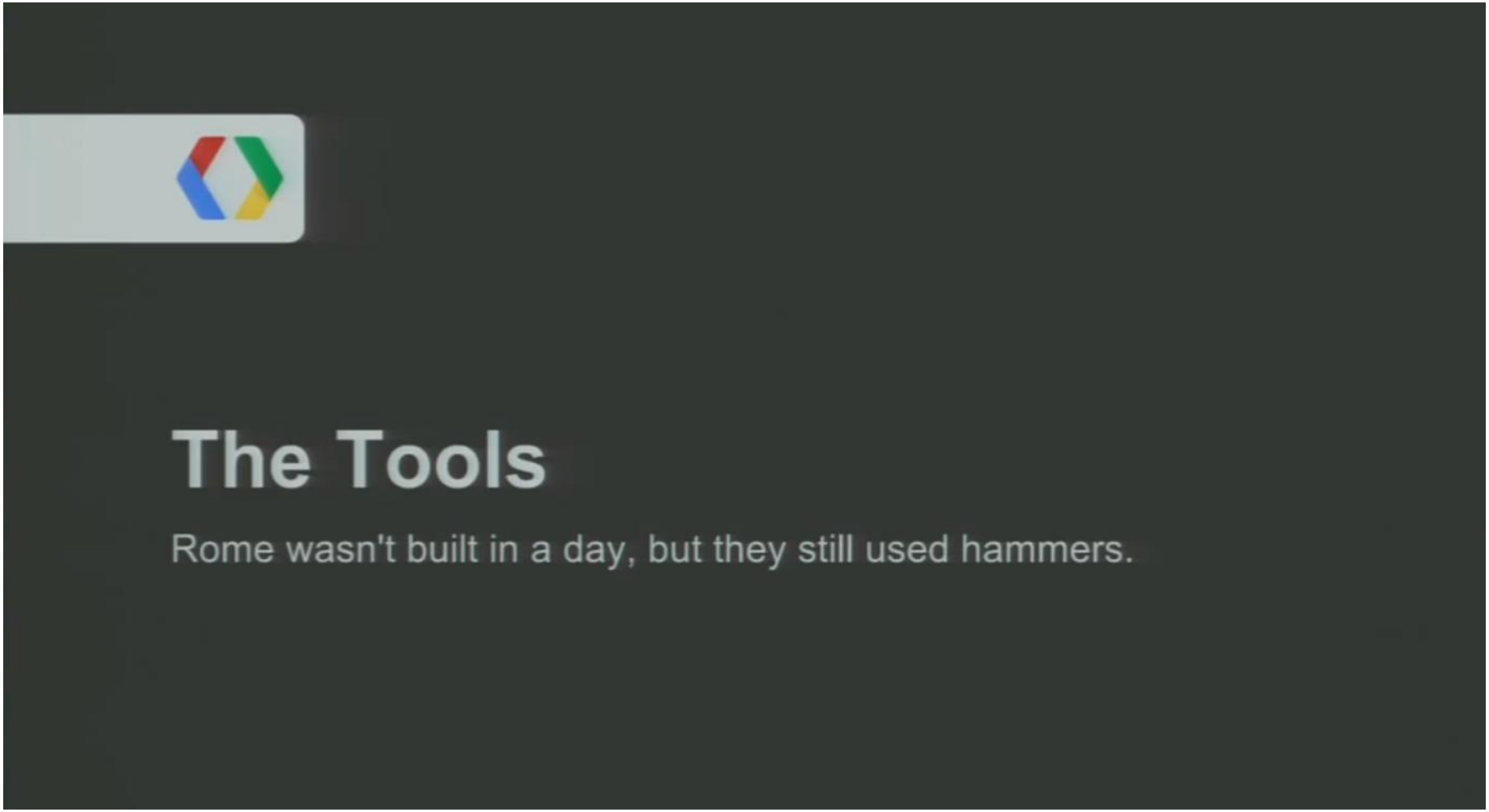
Tiles = Draws = performance



Off DOM canvas



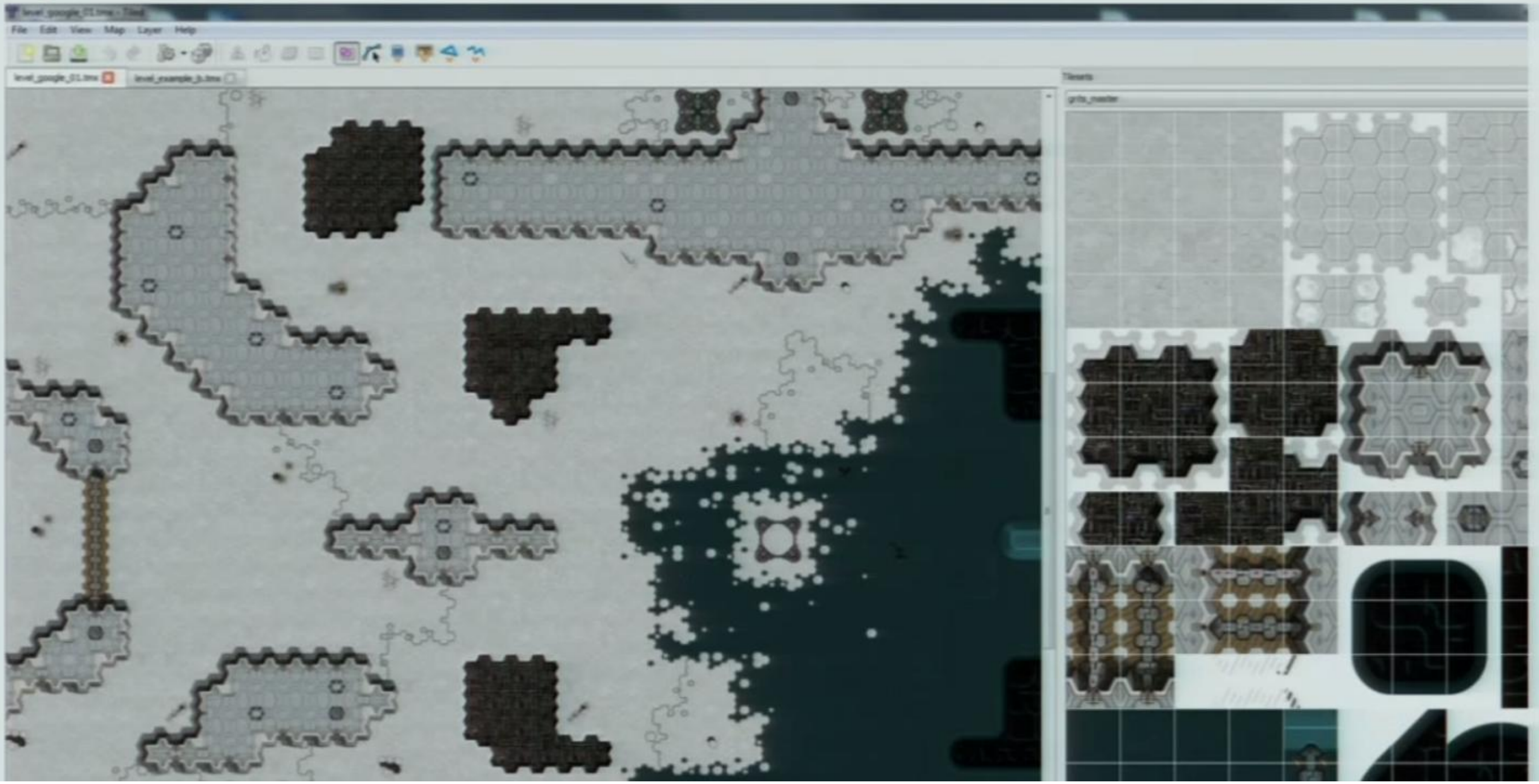




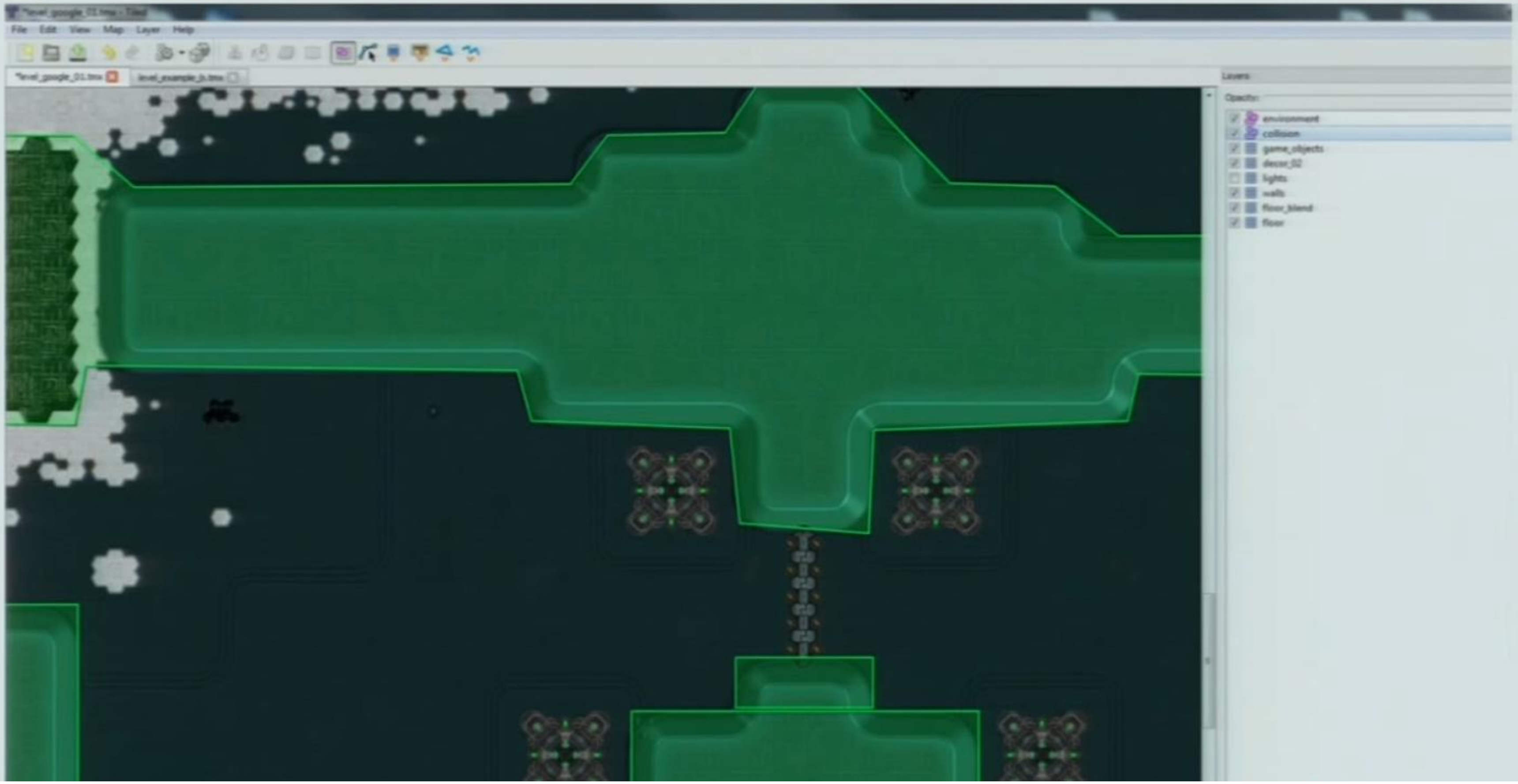
The Tools

Rome wasn't built in a day, but they still used hammers.

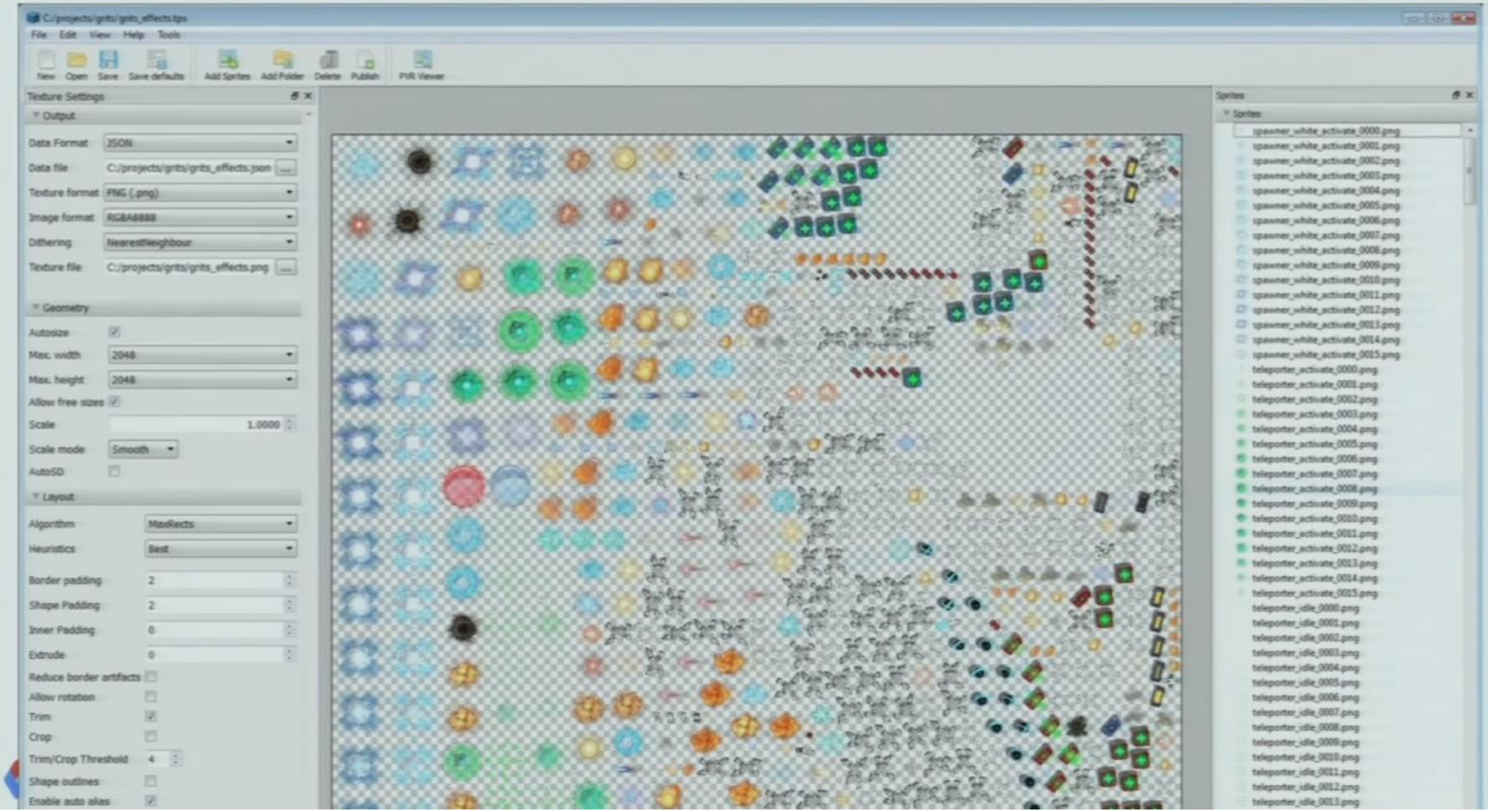
Map Editor : TILED



Map Editor : TILED



Texture Packer



Output

Data Format: JSON

Data file: C:/projects/grits/grits_effects.json

Texture format: PNG (.png)

Image format: RGBA8888

Dithering: NearestNeighbour

Texture file: C:/projects/grits/grits_effects.png

Geometry

Autosize ☑

Max. width: 2048

Max. height: 2048

Allow free sizes ☑

Scale: 1.0000

Scale mode: Smooth

AutoSD □

Layout

Algorithm: MaxRects

Heuristics: Best

Border padding: 2

Shape Padding: 2

Inner Padding: 0

Extrude: 0

Reduce border artifacts □

Allow rotation □

Trim ☑

Crop □

Trim/Crop Threshold: 4

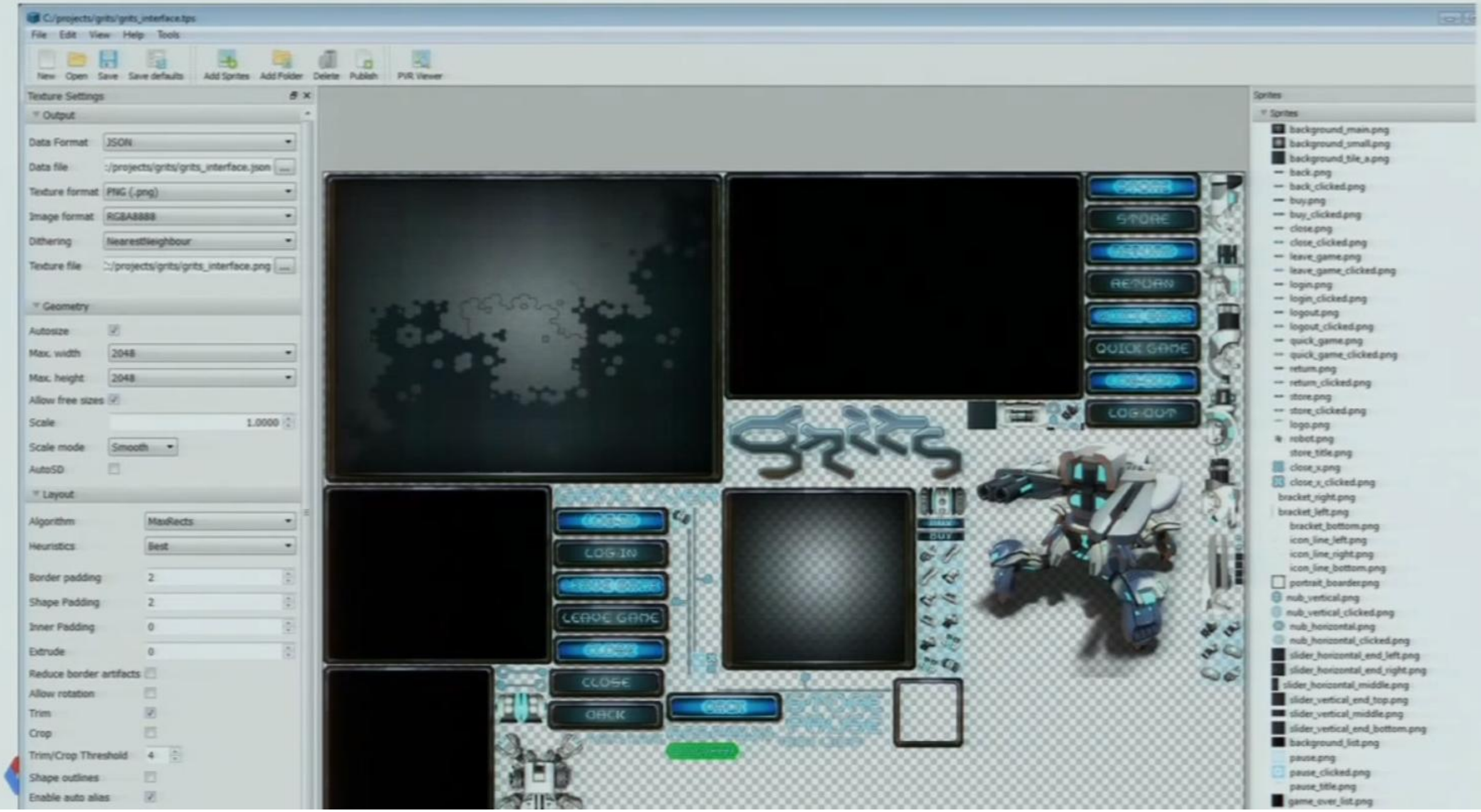
Shape outlines □

Enable auto alias ☑

Sprites

spawner_white_activate_0000.png
spawner_white_activate_0001.png
spawner_white_activate_0002.png
spawner_white_activate_0003.png
spawner_white_activate_0004.png
spawner_white_activate_0005.png
spawner_white_activate_0006.png
spawner_white_activate_0007.png
spawner_white_activate_0008.png
spawner_white_activate_0009.png
spawner_white_activate_0010.png
spawner_white_activate_0011.png
spawner_white_activate_0012.png
spawner_white_activate_0013.png
spawner_white_activate_0014.png
spawner_white_activate_0015.png
teleporter_activate_0000.png
teleporter_activate_0001.png
teleporter_activate_0002.png
teleporter_activate_0003.png
teleporter_activate_0004.png
teleporter_activate_0005.png
teleporter_activate_0006.png
teleporter_activate_0007.png
teleporter_activate_0008.png
teleporter_activate_0009.png
teleporter_activate_0010.png
teleporter_activate_0011.png
teleporter_activate_0012.png
teleporter_activate_0013.png
teleporter_activate_0014.png
teleporter_activate_0015.png
teleporter_idle_0000.png
teleporter_idle_0001.png
teleporter_idle_0002.png
teleporter_idle_0003.png
teleporter_idle_0004.png
teleporter_idle_0005.png
teleporter_idle_0006.png
teleporter_idle_0007.png
teleporter_idle_0008.png
teleporter_idle_0009.png
teleporter_idle_0010.png
teleporter_idle_0011.png
teleporter_idle_0012.png
teleporter_idle_0013.png

Texture Packer



Output

Data Format: JSON

Data file: :/projects/grits/grits_interface.json

Texture format: PNG (.png)

Image format: RGBA8888

Dithering: NearestNeighbour

Texture file: :/projects/grits/grits_interface.png

Geometry

Autosize ☑

Max. width: 2048

Max. height: 2048

Allow free sizes ☑

Scale: 1.0000

Scale mode: Smooth

AutoSD □

Layout

Algorithm: MaxRects

Heuristics: Best

Border padding: 2

Shape Padding: 2

Inner Padding: 0

Extrude: 0

Reduce border artifacts □

Allow rotation □

Trim ☑

Crop □

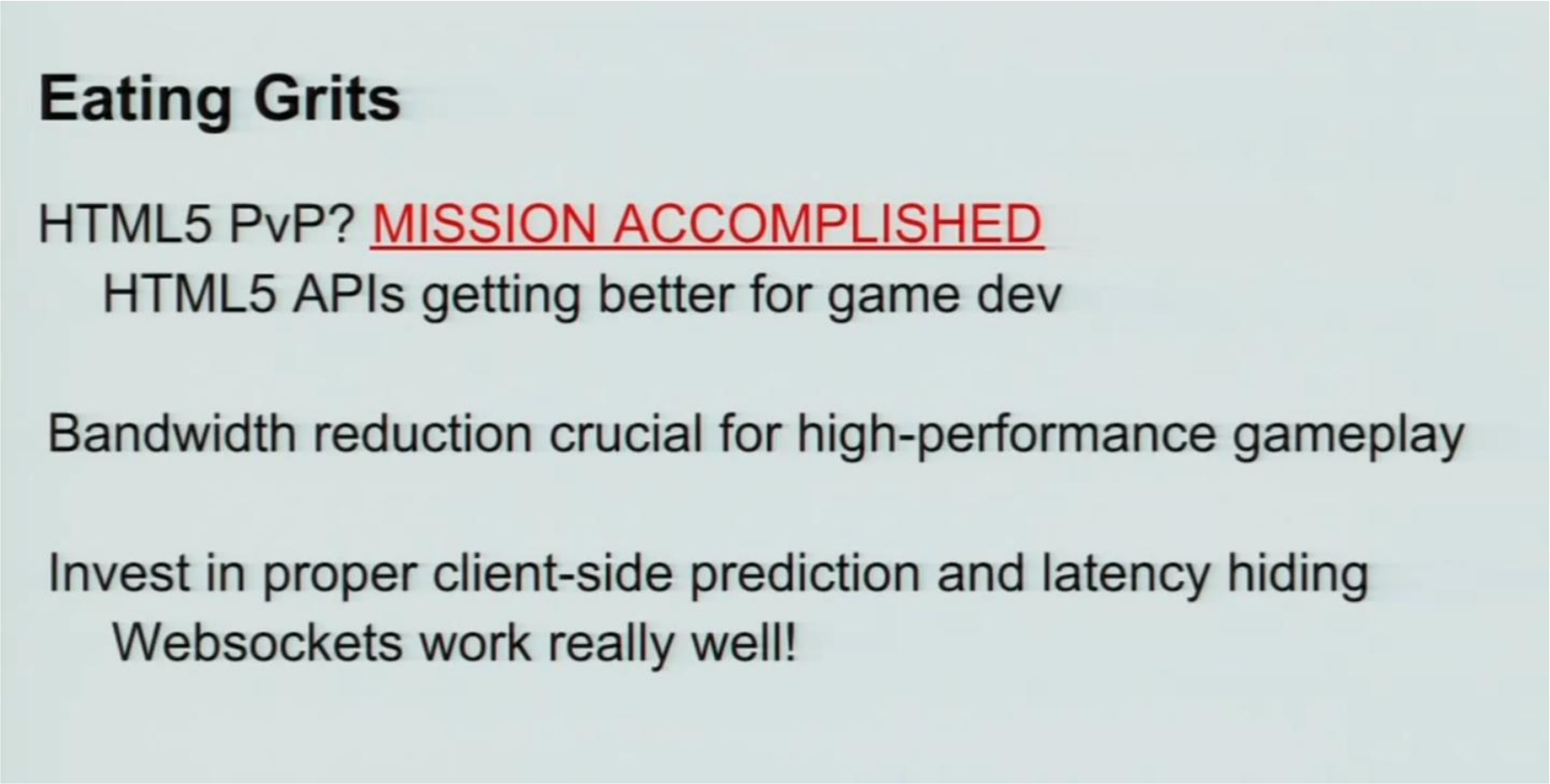
Trim/Crop Threshold: 4

Shape outlines □

Enable auto alias ☑

Sprites

- background_main.png
- background_small.png
- background_tile_a.png
- back.png
- back_clicked.png
- buy.png
- buy_clicked.png
- close.png
- close_clicked.png
- leave_game.png
- leave_game_clicked.png
- login.png
- login_clicked.png
- logout.png
- logout_clicked.png
- quick_game.png
- quick_game_clicked.png
- return.png
- return_clicked.png
- store.png
- store_clicked.png
- logo.png
- robot.png
- store_title.png
- close_x.png
- close_x_clicked.png
- bracket_right.png
- bracket_left.png
- bracket_bottom.png
- icon_line_left.png
- icon_line_right.png
- icon_line_bottom.png
- portrait_boarder.png
- nub_vertical.png
- nub_vertical_clicked.png
- nub_horizontal.png
- nub_horizontal_clicked.png
- slider_horizontal_end_left.png
- slider_horizontal_end_right.png
- slider_horizontal_middle.png
- slider_vertical_end_top.png
- slider_vertical_middle.png
- slider_vertical_end_bottom.png
- background_list.png
- pause.png
- pause_clicked.png
- pause_title.png
- game_over_list.png



Eating Grits

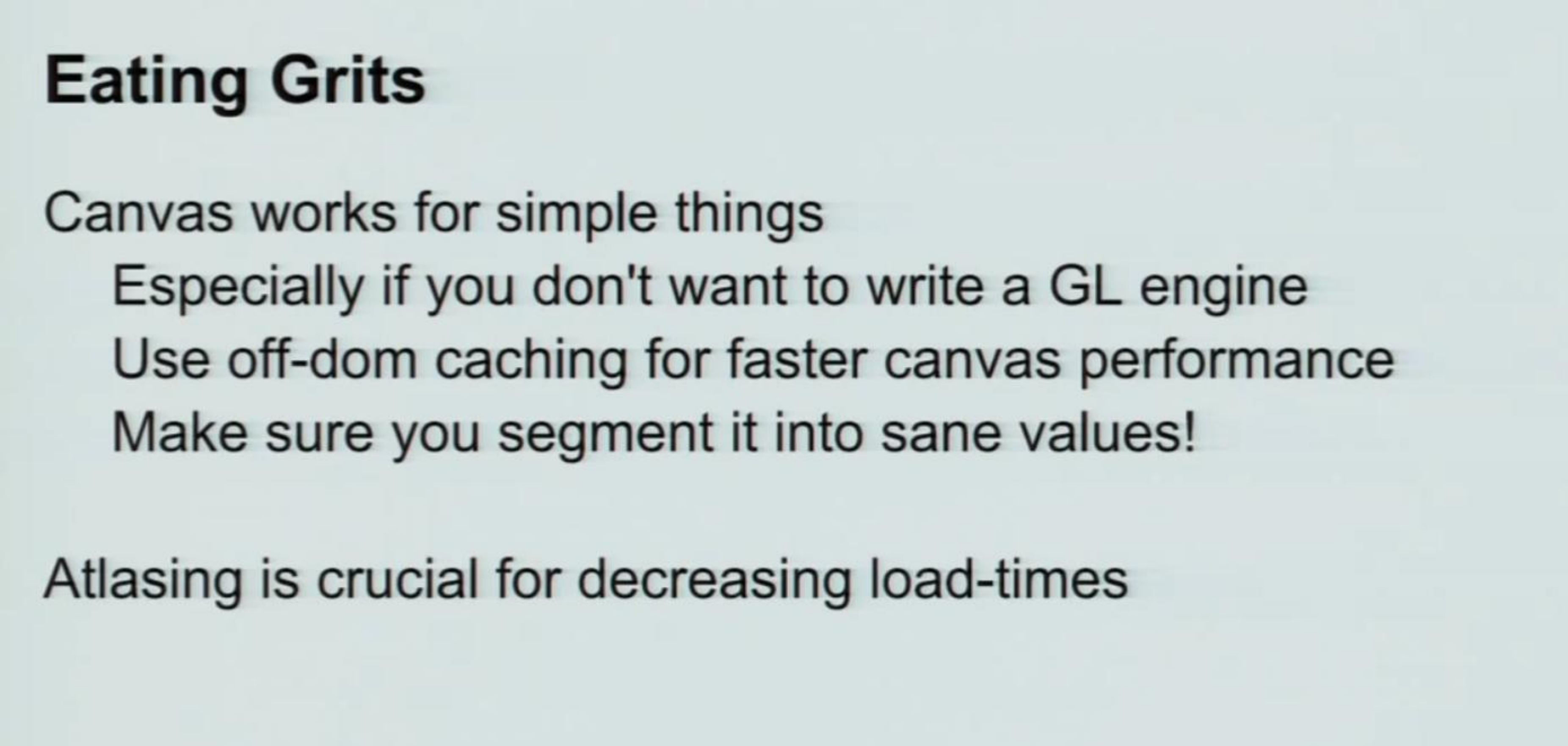
HTML5 PvP? MISSION ACCOMPLISHED

- HTML5 APIs getting better for game dev

Bandwidth reduction crucial for high-performance gameplay

Invest in proper client-side prediction and latency hiding

- Websockets work really well!



Eating Grits

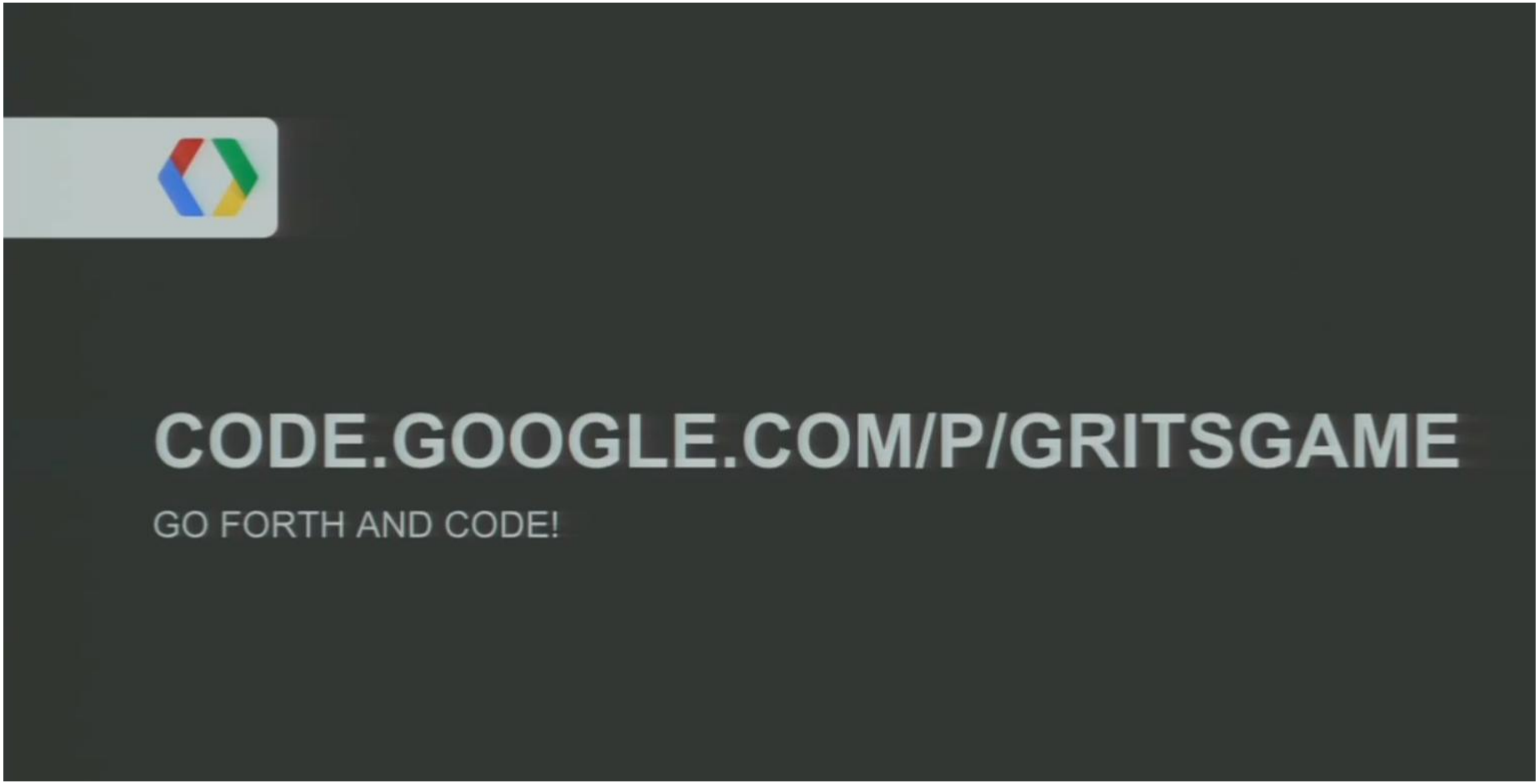
Canvas works for simple things

Especially if you don't want to write a GL engine

Use off-dom caching for faster canvas performance

Make sure you segment it into sane values!

Atlasing is crucial for decreasing load-times



CODE.GOOGLE.COM/P/GRITSGAME

GO FORTH AND CODE!

HTML
5
Strike Fortress
ELECTRONIC ARTS – OCCO
EA
OCCO