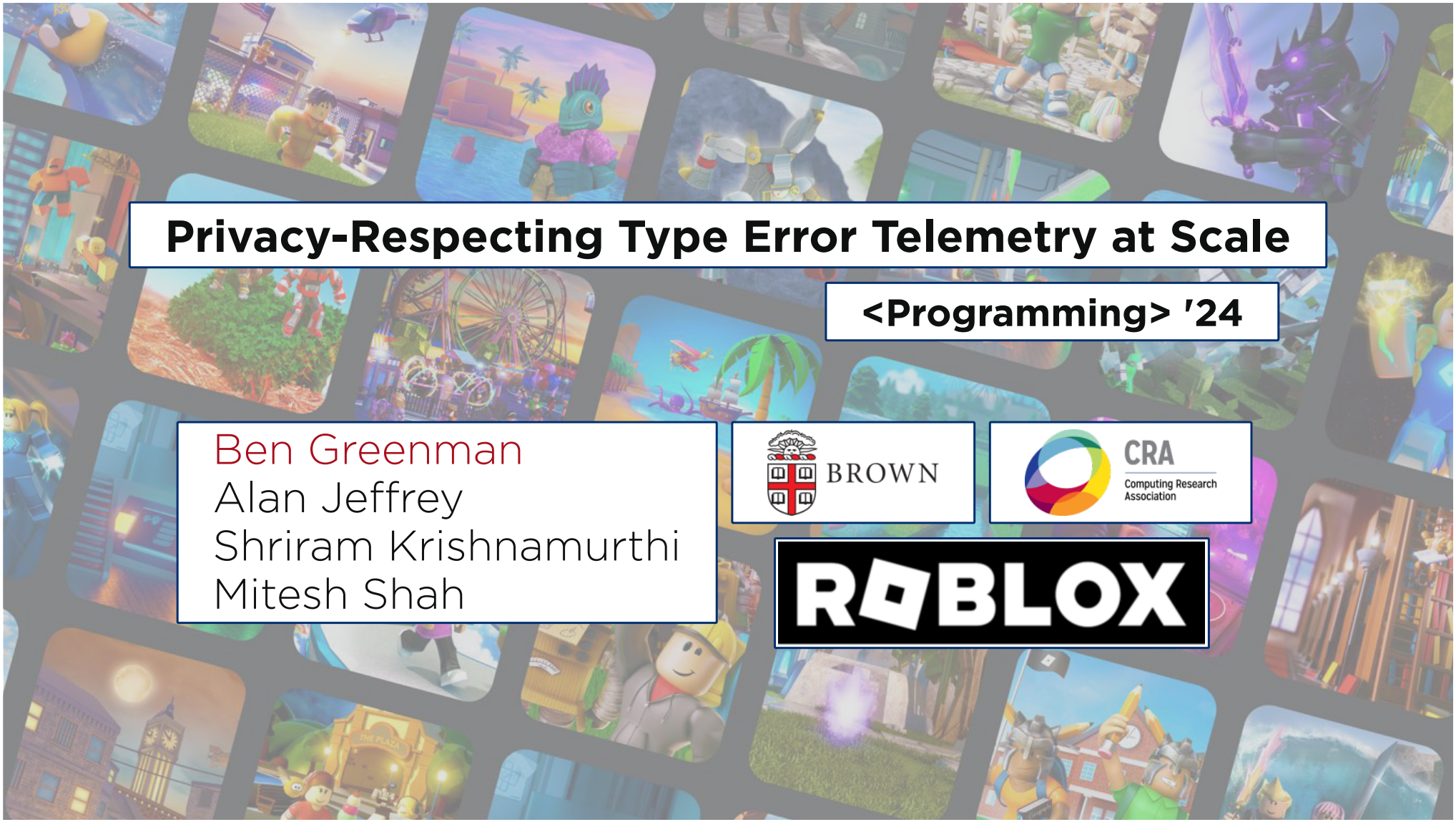


Privacy-Respecting Type Error Telemetry at Scale

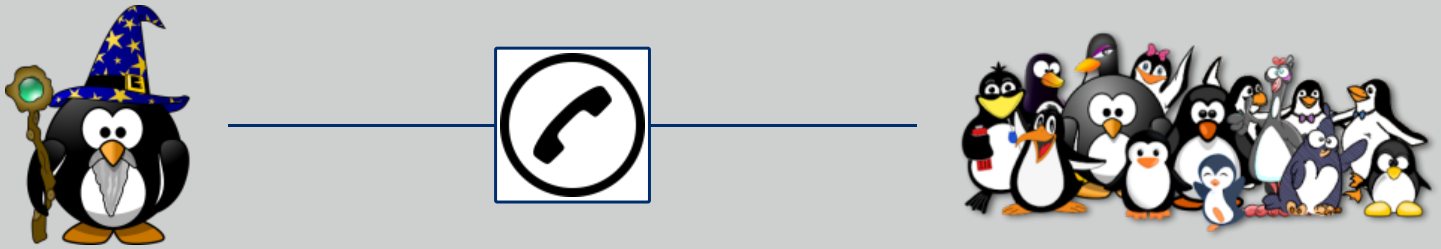
<Programming> '24

Ben Greenman
Alan Jeffrey
Shriram Krishnamurthi
Mitesh Shah

BROWN

CRA
Computing Research Association

ROBLOX



Language Designers & Users need to talk!

Language Designers & Users need to talk!

?

- Interviews
- Surveys
- Experiments

Language Designers & Users need to talk!

?

- Interviews
- Surveys
- Experiments

but Low Bandwidth

Language Designers & Users need to talk!

?

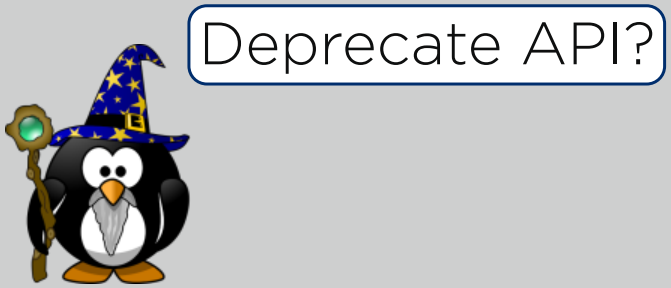
Interviews

Surveys

Experiments

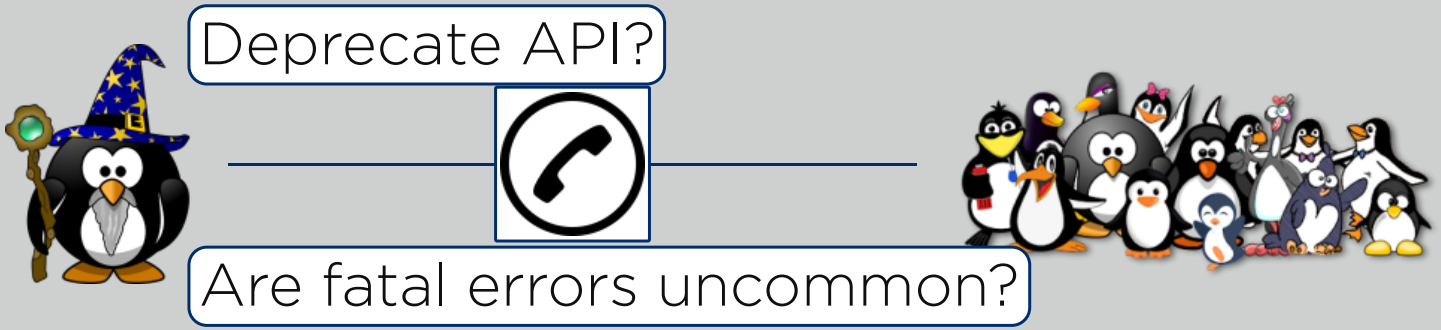
but Low Bandwidth

this work:
Telemetry

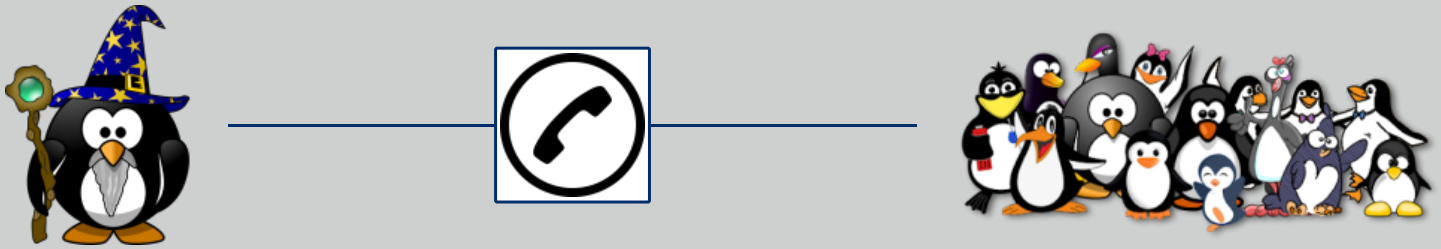

Deprecate API?

Deprecate API?

Are fatal errors uncommon?



Telemetry ==> Informed Decisions



But, telemetry can go wrong

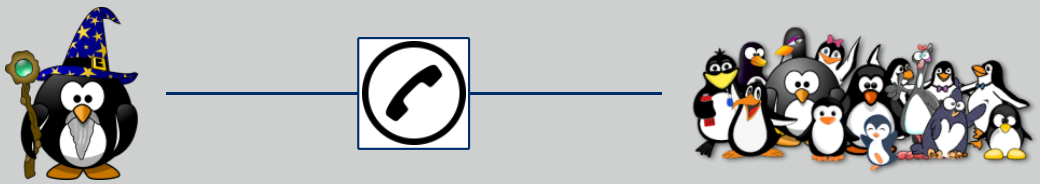
Are you spying on me?

Are you spying on me?

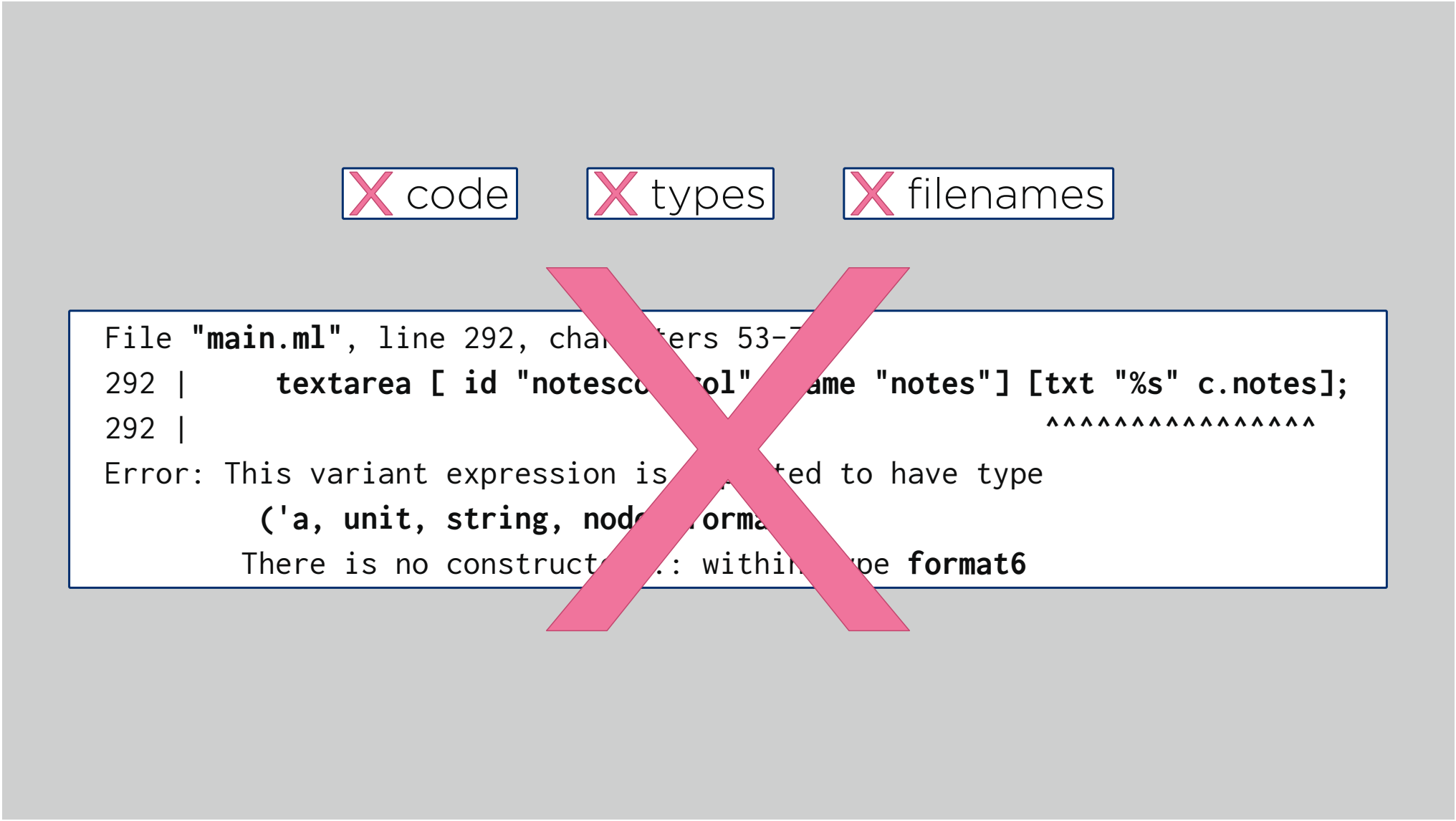
Personal Info

Trade Secrets

How to study type errors without revealing any code?

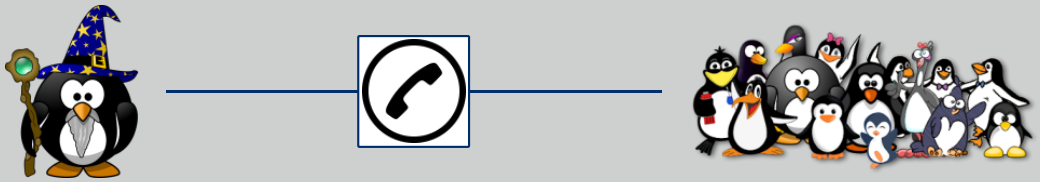


```
File "main.ml", line 292, characters 53-70:
292 |     textarea [ id "notescontrol"; name "notes"] [txt "%s" c.notes];
292 |                                                     ^^^^^^^^^^^^^^^^^
Error: This variant expression is expected to have type
         ('a, unit, string, node) format4
       There is no constructor :: within type format6
```


code
types
filenames
File "main.ml", line 292, cha ters 53-
292 | textarea [id "notesco ol" ame "notes"] [txt "%s" c.notes];
292 | ^^^^^^^^^^^^^^^^
Error: This variant expression is ted to have type
('a, unit, string, nod orma
There is no construct : withi pe format6

How to study type errors without revealing any code?

Formative work, generating hypotheses




Luau

@


ROBLOX

@

2.4 million developers [Dec'23]

gradual types


```
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```


```
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

create a table

update it, maybe

read from table


```
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

create a table

update it, maybe

read from table

`x.r` ==> runtime error

`x.q + x.p` ==> possible error


```
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

nocheck

`x.r` ==> runtime error

`x.q + x.p` ==> possible error


```
--!nonstrict
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

nonstrict

Unknown Property


```
--!nonstrict
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

nonstrict

Unknown Property

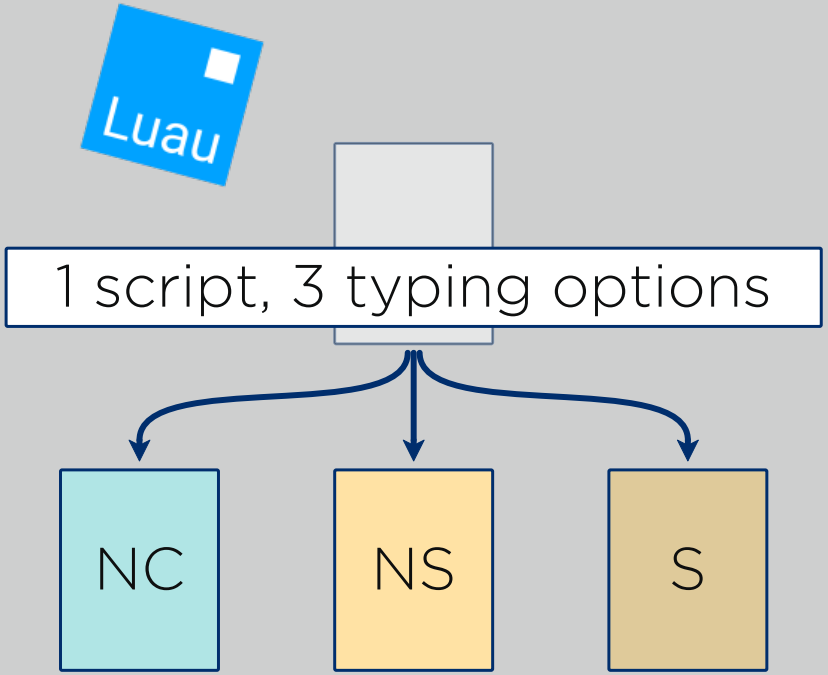
```
--!strict
local x = { p = 5, q = nil }
if condition then x.q = 7 end
local y = x.q + x.p
local z = x.r
```

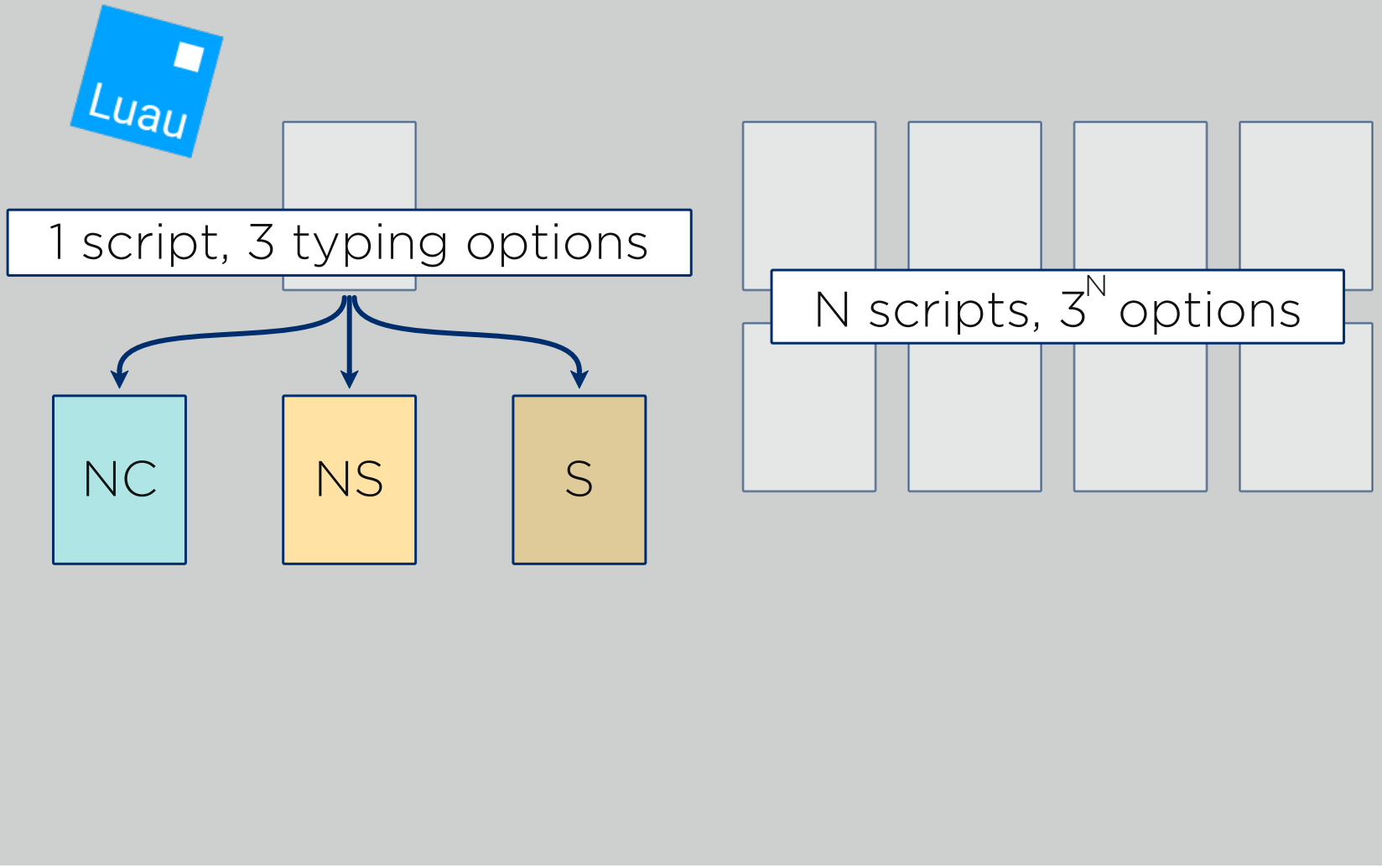
strict

Type Mismatch

Unknown Property

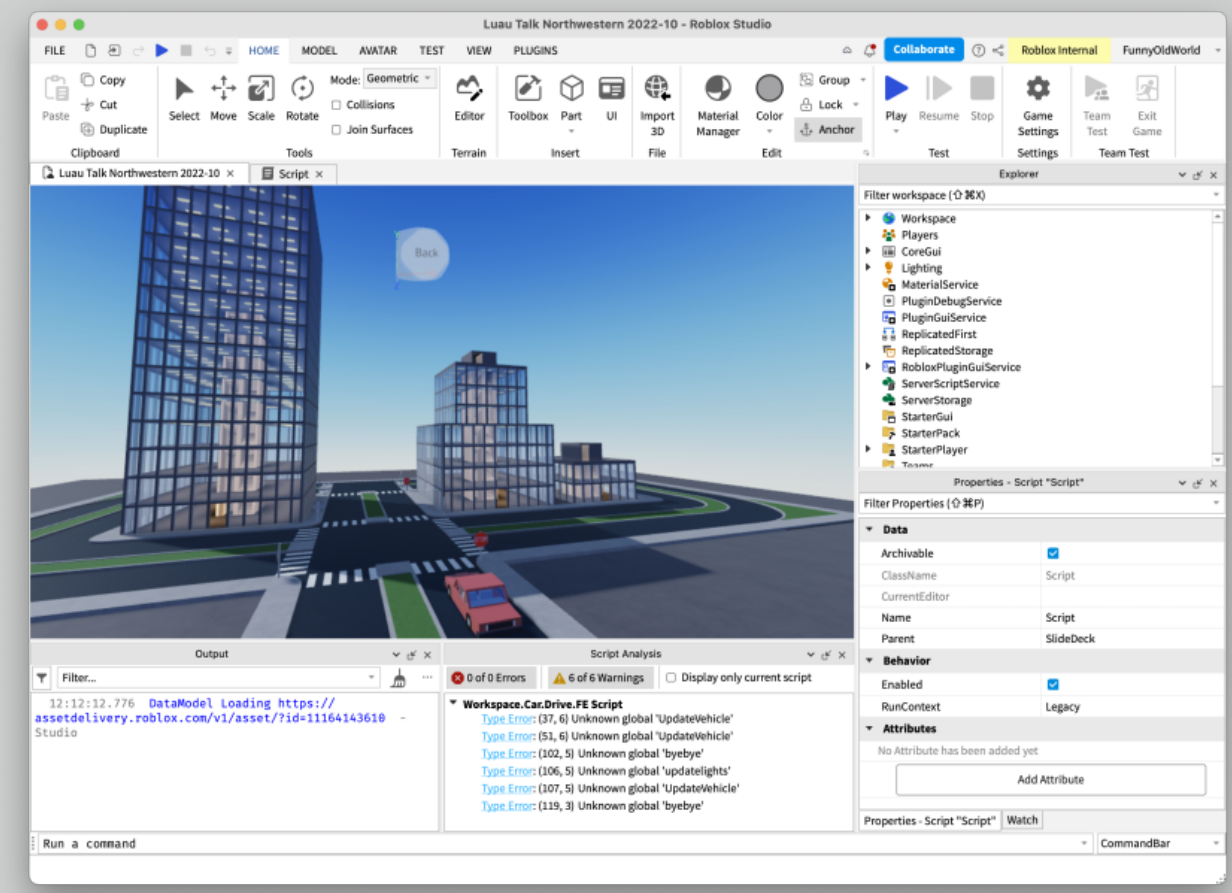
nocheck	syntax errors
nonstrict	high-confidence errors
strict	full type analysis


Luau
1 script, 3 typing options
NC
NS
S


Luau
1 script, 3 typing options
NC
NS
S
N scripts, 3^N options

ROBLOX

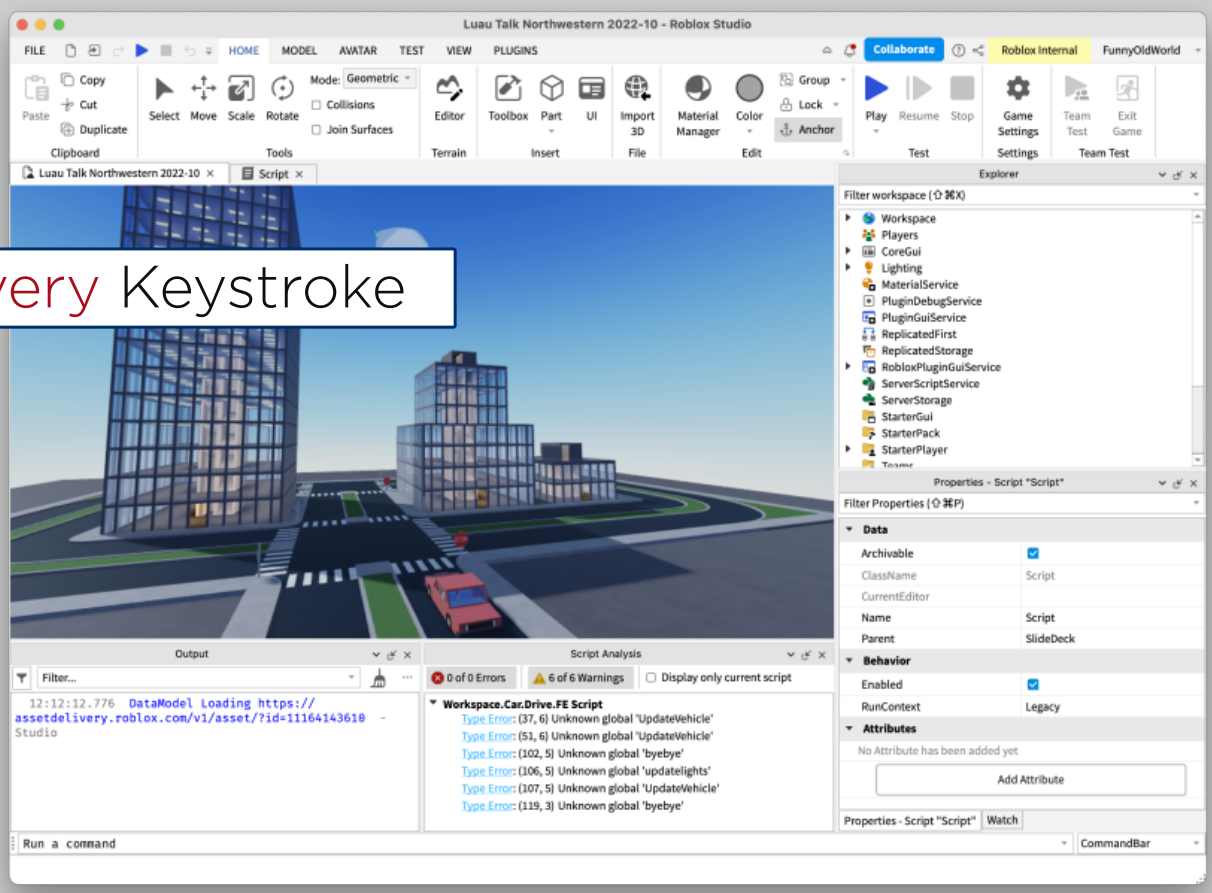
Luau


Luau Talk Northwestern 2022-10 - Roblox Studio
FILE
HOME
MODEL
AVATAR
TEST
VIEW
PLUGINS
Collaborate
Roblox Internal
FunnyOldWorld
Copy
Cut
Duplicate
Paste
Clipboard
Select
Move
Scale
Rotate
Mode: Geometric
Collisions
Join Surfaces
Tools
Editor
Terrain
Toolbox
Part
UI
Insert
Import 3D
File
Material Manager
Color
Group
Lock
Anchor
Edit
Play
Resume
Stop
Test
Game Settings
Settings
Team Test
Exit Game
Team Test
Luau Talk Northwestern 2022-10
Script
Back
Explorer
Filter workspace (⇧⌘X)
Workspace
Players
CoreGui
Lighting
MaterialService
PluginDebugService
PluginGuiService
ReplicatedFirst
ReplicatedStorage
RobloxPluginGuiService
ServerScriptService
ServerStorage
StarterGui
StarterPack
StarterPlayer
Properties - Script "Script"
Filter Properties (⇧⌘P)
Data
Archivable
ClassName
Script
CurrentEditor
Name
Script
Parent
SlideDeck
Behavior
Enabled
RunContext
Legacy
Attributes
No Attribute has been added yet
Add Attribute
Properties - Script "Script"
Watch
Output
Filter...
12:12:12.776 DataModel Loading https://assetdelivery.roblox.com/v1/asset/?id=11164143610 - Studio
Script Analysis
0 of 0 Errors
6 of 6 Warnings
Display only current script
Workspace.Car.Drive.FE Script
Type Error: (37, 6) Unknown global 'UpdateVehicle'
Type Error: (51, 6) Unknown global 'UpdateVehicle'
Type Error: (102, 5) Unknown global 'byebye'
Type Error: (106, 5) Unknown global 'updatelights'
Type Error: (107, 5) Unknown global 'UpdateVehicle'
Type Error: (119, 3) Unknown global 'byebye'
Run a command
CommandBar

ROBLOX

Luau

Typechecking Every Keystroke

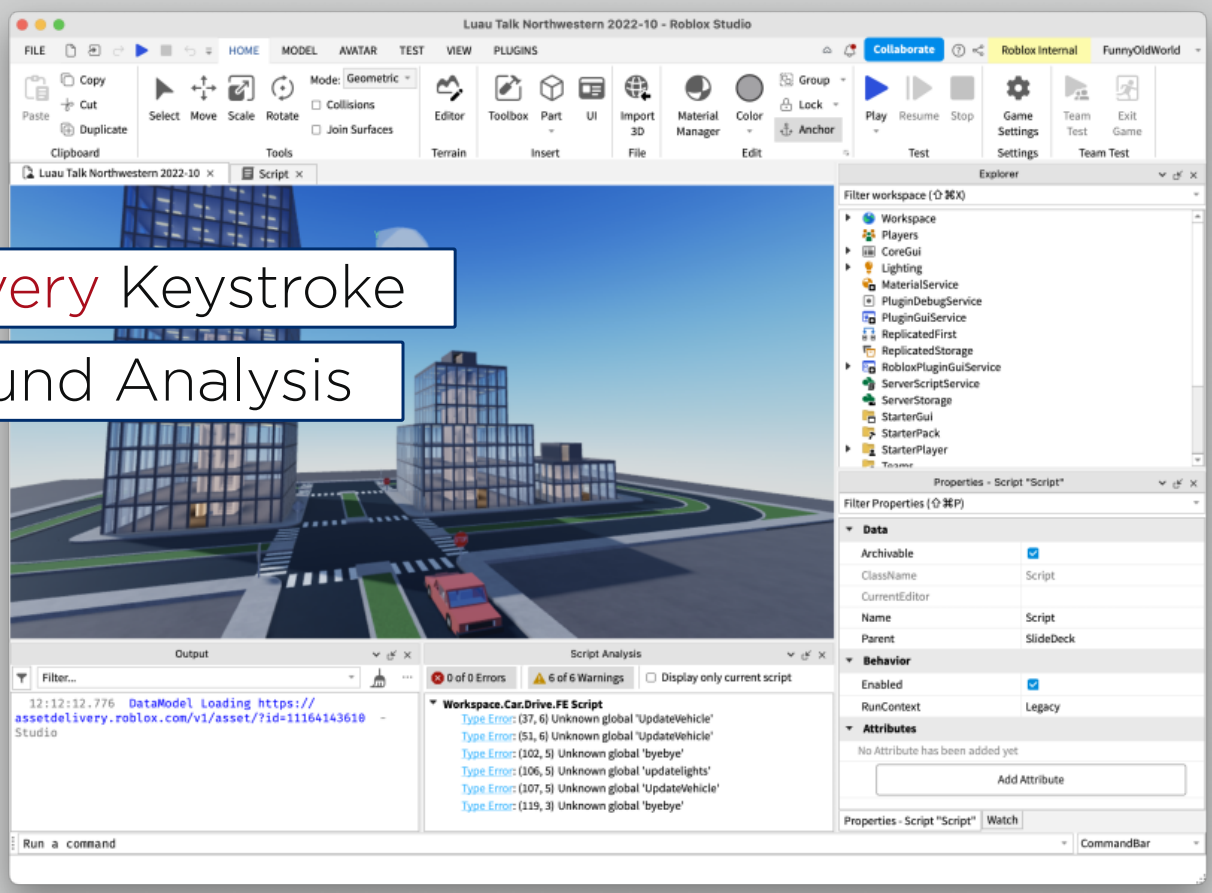


ROBLOX

Luau

Typechecking Every Keystroke

Strict Background Analysis



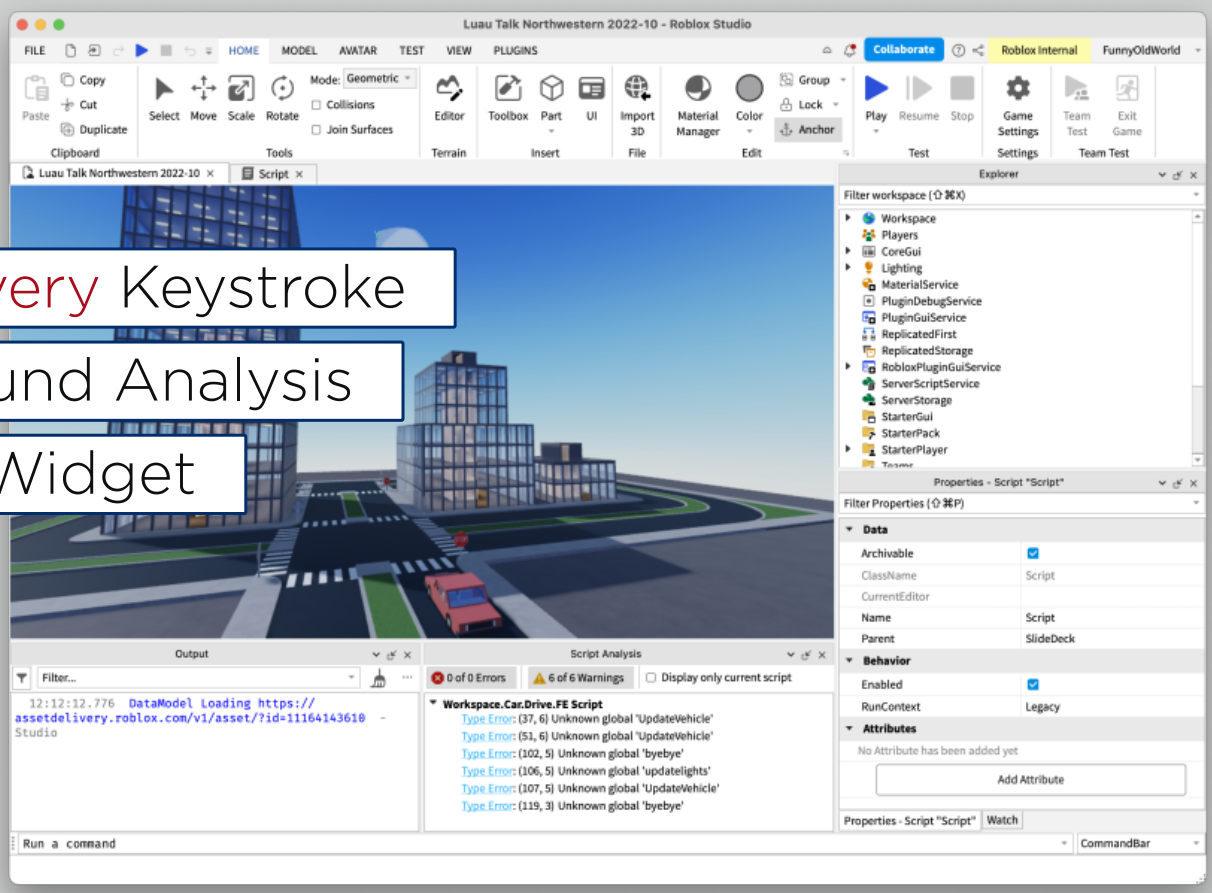
ROBLOX

Luau

Typechecking Every Keystroke

Strict Background Analysis

Analysis Widget



ROBLOX

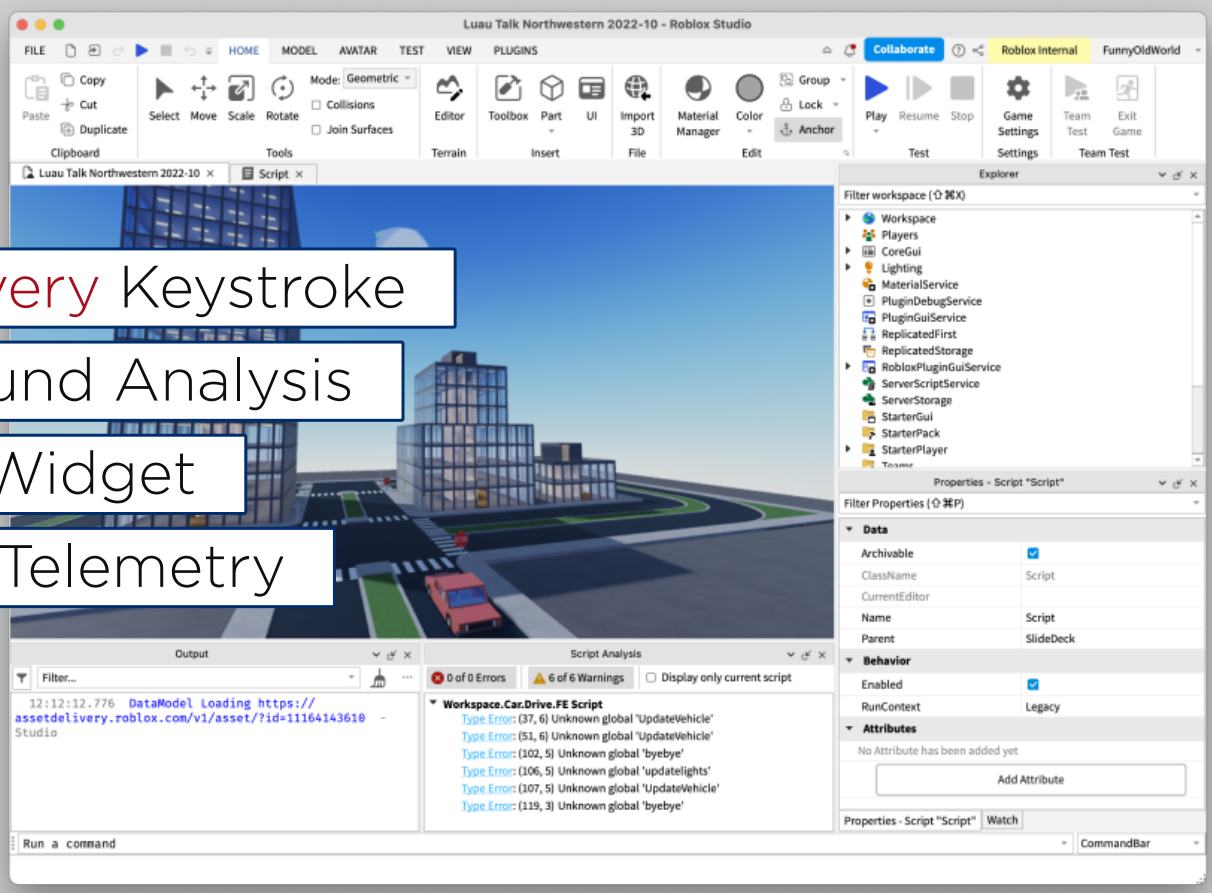
Luau

Typechecking Every Keystroke

Strict Background Analysis

Analysis Widget

Usage-Data Telemetry



Research Topics

1. Adoption

How many use types?
How many mix analysis modes?
How many change analysis modes?

Research Topics

1. Adoption

How many use types?
How many mix analysis modes?
How many change analysis modes?

2. Errors and Repairs

Which errors are common?
Which errors tend to survive edits?

Research Topics

1. Adoption

How many use types?
How many mix analysis modes?
How many change analysis modes?

2. Errors and Repairs

Which errors are common?
Which errors tend to survive edits?

3. Impact on Background Errors

nocheck ==> more background errors?

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Id

Pseudonymized session id

Telemetry Design

Who? randomly-selected sessions
When? random keystrokes + module switch
What? counts, not code

Id

Time

Client-side timestamp

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Id | Time | Mode

Current type mode

NC | NS | S

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Id | Time | Mode | Reason

Keystroke or module switch

Telemetry Design

Who? randomly-selected sessions
When? random keystrokes + module switch
What? counts, not code

Id | Time | Mode | Reason | Sizes

files
lines in codebase
lines in edit range

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

Id | Time | Mode | Reason | Sizes

Global Counts

errors overall
errors in script
errors in edit range

Telemetry Design

Who? randomly-selected sessions
When? random keystrokes + module switch
What? counts, not code

Id | Time | Mode | Reason | Sizes

Global Counts | Edit Range Counts

```
{ Type Error : #, ... }
{ BG   Error : #, ... }
```

in edit range,
up to 70 counts

Telemetry Design

Who?	randomly-selected sessions
When?	random keystrokes + module switch
What?	counts, not code

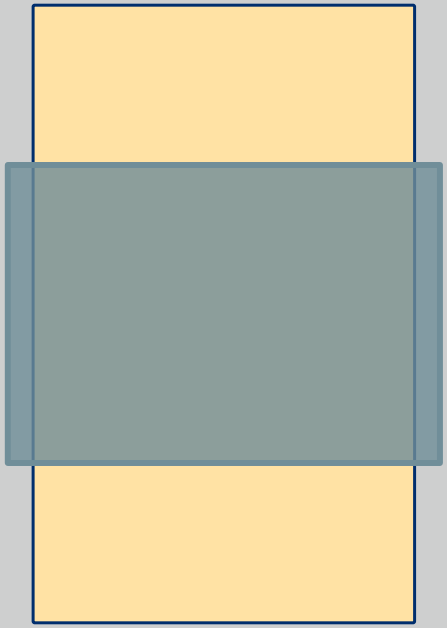
Id | Time | Mode | Reason | Sizes

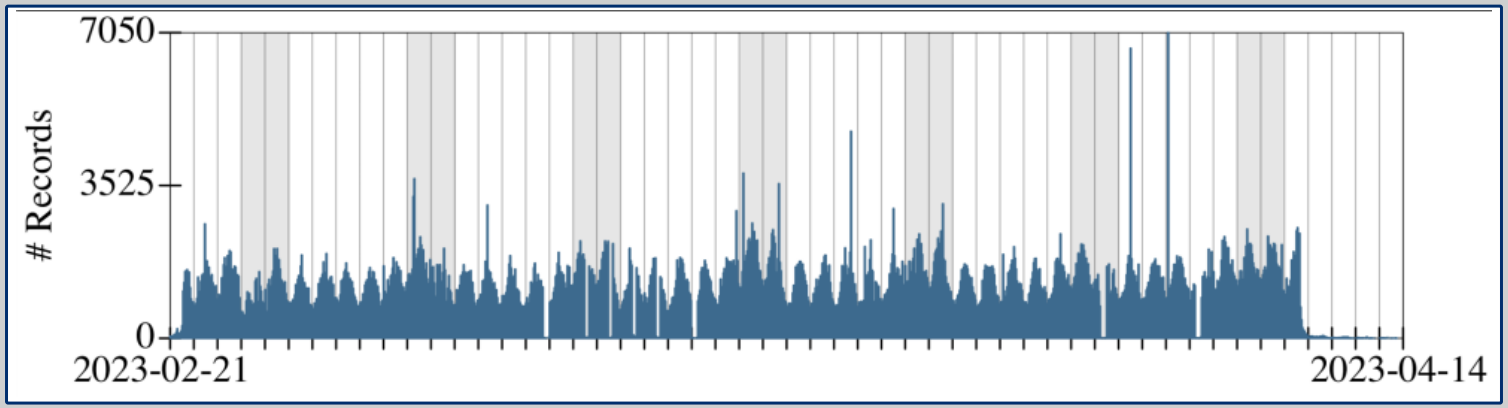
Global Counts | Edit Range Counts

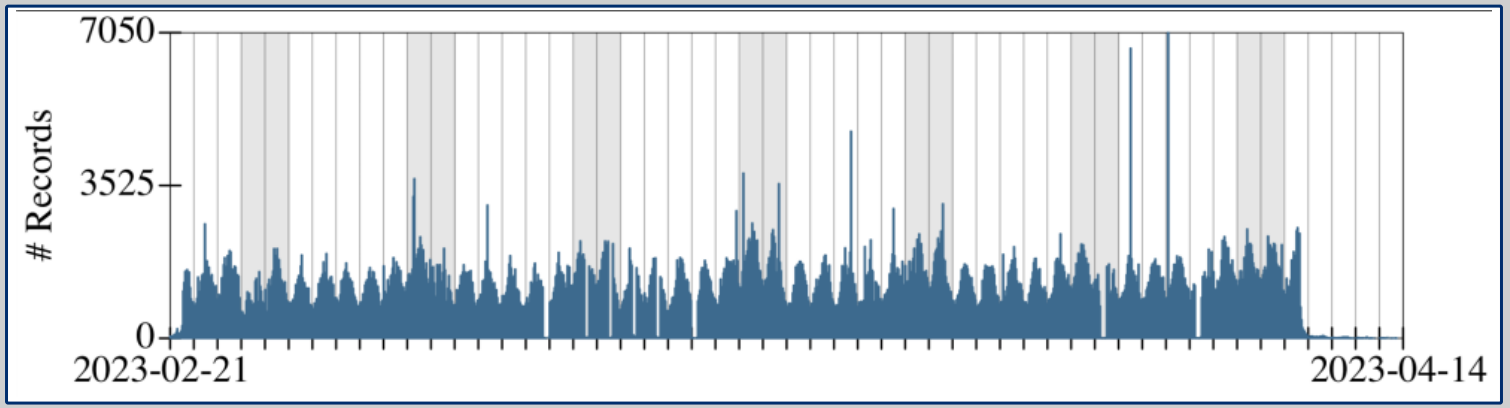
Edit Range =

`[min line, max line]`

Coarse approximation
Not shared!
Filters overlapping errors




7050
3525
0
Records
2023-02-21
2023-04-14



3 months of data

+1.5 million records

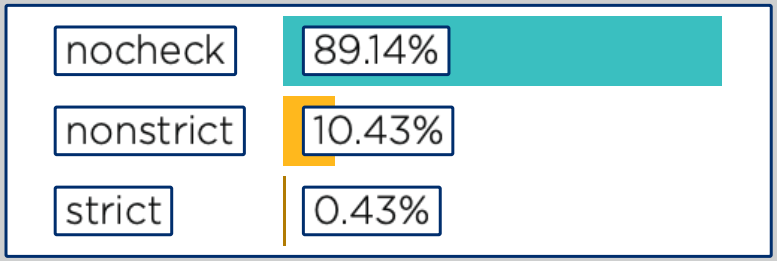
66% from keystrokes, 34% from module switches

+340 thousand sessions

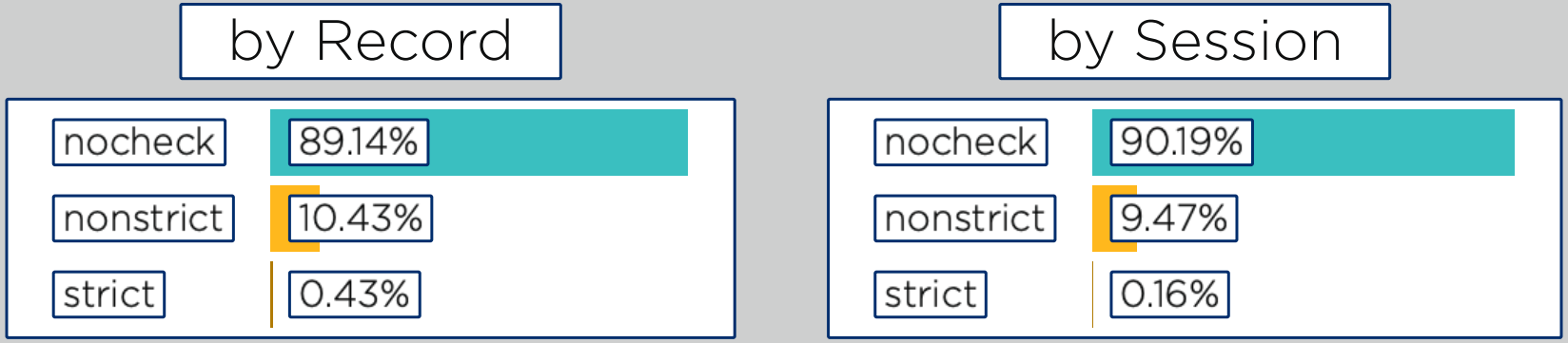
1. Adoption

1. Adoption

by Record

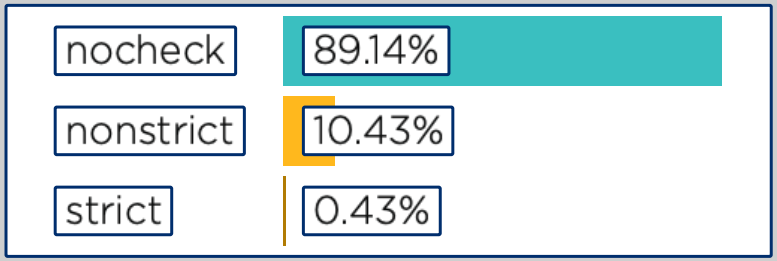


1. Adoption

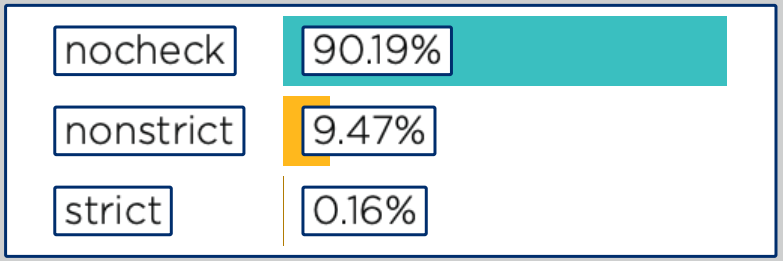

by Record
nocheck
89.14%
nonstrict
10.43%
strict
0.43%
by Session
nocheck
90.19%
nonstrict
9.47%
strict
0.16%

1. Adoption

by Record



by Session



90% nocheck

0.18% of sessions use +1 modes

2. Errors

2. Errors

Internal Limits:

`Code Too Complex` `Unification Too Complex` `Normalization Too Complex`

2. Errors

Internal Limits:

`Code Too Complex` `Unification Too Complex` `Normalization Too Complex`

26 total, only in 3 sessions

2. Errors

nonstrict

UnknownSymbol	62.13%
SyntaxError	15.42%
UnknownProperty	8.28%
UnknownRequire	3.13%
...	
CannotInferBinaryOperation	0.02%
OnlyTablesCanHaveMethods	0.01%
DuplicateTypeDefinition	<0.01%
TypesAreUnrelated	<0.01%

strict

UnknownSymbol	23.97%
TypeMismatch	20.46%
UnknownProperty	18.88%
SyntaxError	9.31%
...	
ModuleHasCyclicDependency	0.09%
CannotExtendTable	0.09%
OccursCheckFailed	0.09%
TypesAreUnrelated	0.09%

2. Errors

Likely to survive edits:

`Optional Value Access` `Cannot Infer Binary Operation`

2. Errors

Likely to survive edits:

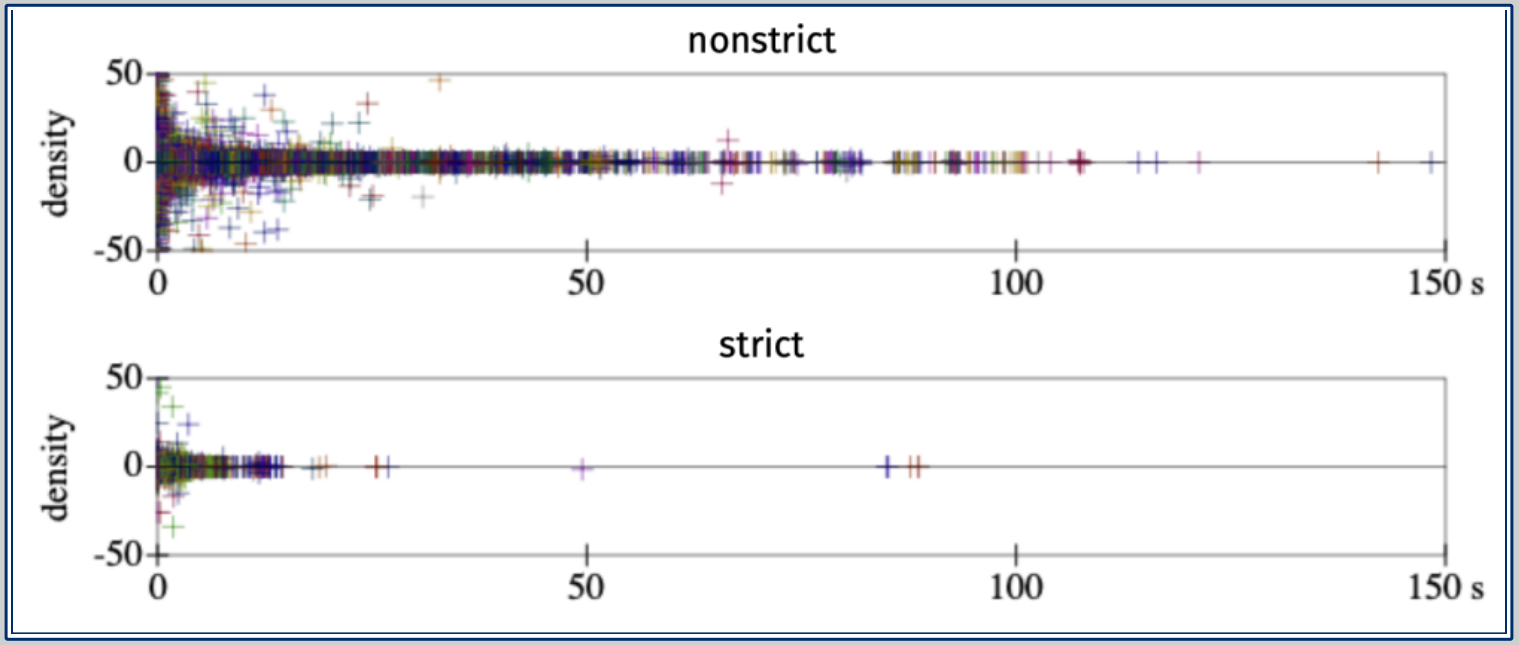
`Optional Value Access` `Cannot Infer Binary Operation`

More in paper:

	nocheck			nonstrict			strict		
	↑	=	↓	↑	=	↓	↑	=	↓
CannotCallNonFunction	-	-	-	12	2	13	1	-	1
CannotExtendTable	-	-	-	6	8	1	-	-	1
CannotInferBinaryOperation					1		5	6	6

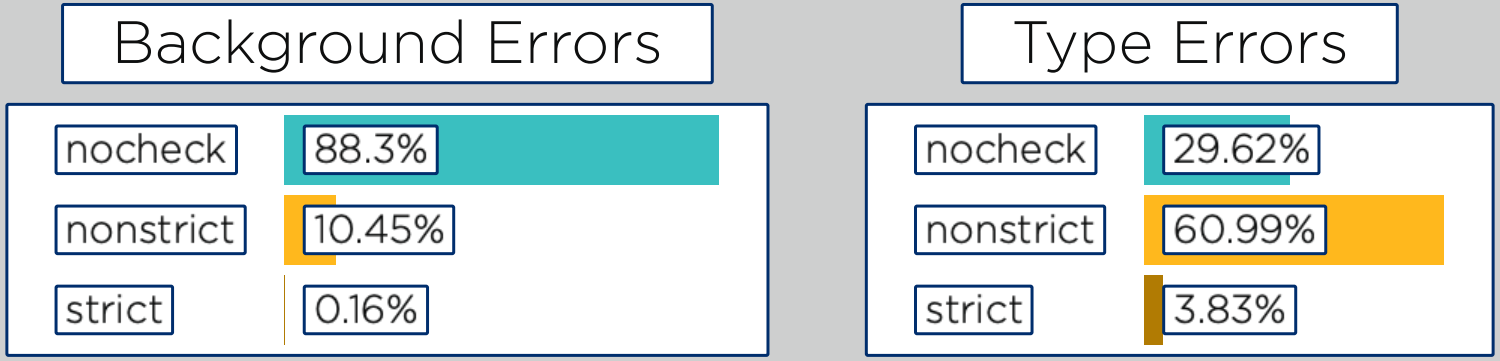
2. Errors + Repairs

Density changes over time (curr - old / lines)

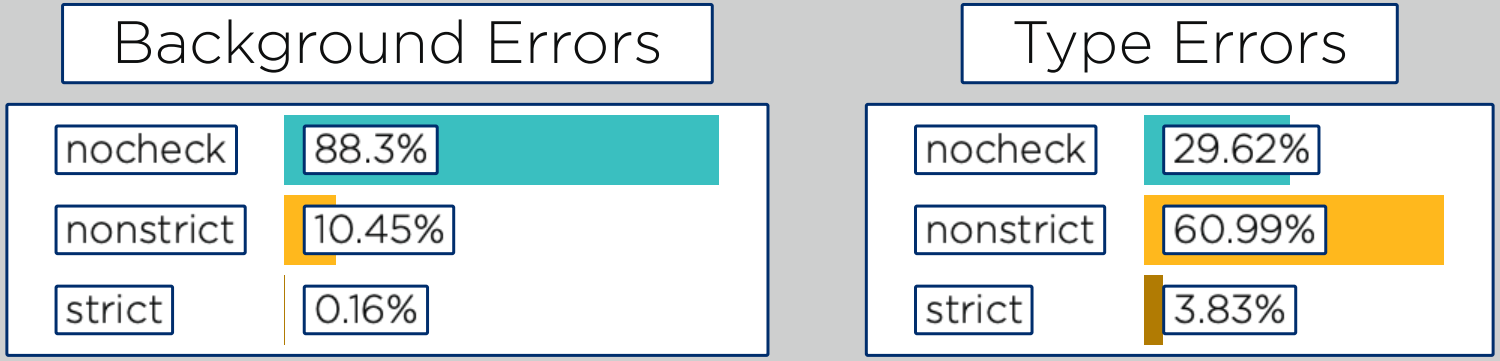


3. Types vs. Background Errors

3. Types vs. Background Errors



3. Types vs. Background Errors



BG rates proportional to adoption rates

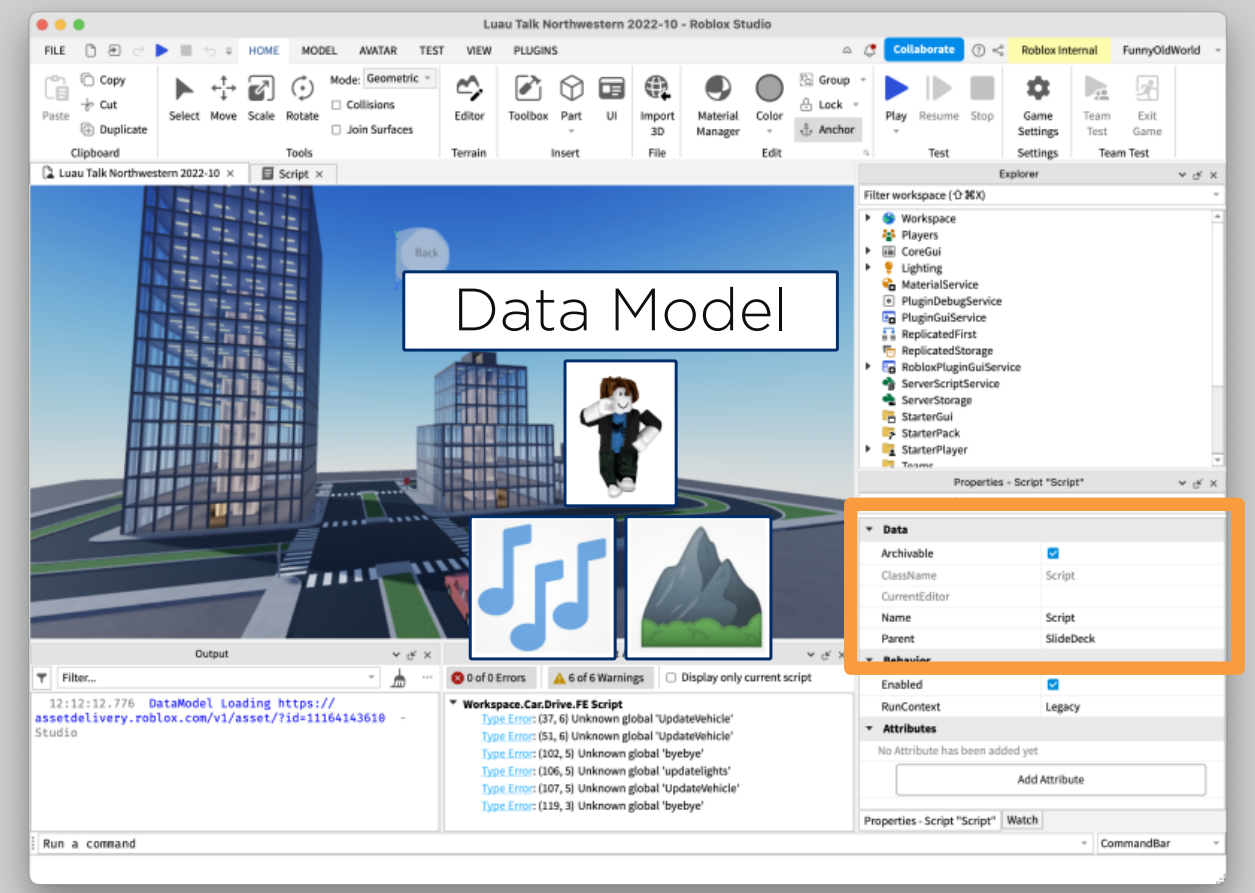
3. Types vs. Background Errors

16% of strict records
increase overall type errors
but not background errors

3. Types vs. Background Errors

16% of strict records
increase overall type errors
but not background errors

Strict is too picky about data assets


Luau Talk Northwestern 2022-10 - Roblox Studio
Data Model
Explorer
Workspace
Players
CoreGui
Lighting
MaterialService
PluginDebugService
PluginGuiService
ReplicatedFirst
ReplicatedStorage
RobloxPluginGuiService
ServerScriptService
ServerStorage
StarterGui
StarterPack
StarterPlayer
Properties - Script "Script"
Data
Archivable
ClassName
Script
CurrentEditor
Name
Script
Parent
SlideDeck
Enabled
RunContext
Legacy
Attributes
No Attribute has been added yet
Add Attribute
Output
Run a command
CommandBar

Findings

1. Adoption

10% use types, 1% use strict

<0.15% mix analysis modes
<0.13% change modes

Findings

1. Adoption

10% use types, 1% use strict 

<0.15% mix analysis modes
<0.13% change modes

2. Errors and Repairs

common errors: syntax (50%), arity (2%), option unpacking (2%)
internal limits are rare
errors rarely pile up 

Findings

1. Adoption

10% use types, 1% use strict 

<0.15% mix analysis modes
<0.13% change modes

2. Errors and Repairs

common errors: syntax (50%), arity (2%), option unpacking (2%)
internal limits are rare
errors rarely pile up 

3. Impact on Background Errors

no correlation

ROBLOX

Luau

Make types the default!

ROBLOX

Luau

Make types the default!

Strict needs work

- data model needs types
- low adoption ==> inexpressive?

Lite telemetry ==> useful analyses

- gradual adoption
- error frequency
- repairs

Lite telemetry ==> useful analyses

- gradual adoption
- error frequency
- repairs

Much more to explore!

How do devs actually use gradual types?

https://doi.org/10.5281/zenodo.10275213

Data + Analysis Scripts

https://doi.org/10.5281/zenodo.10275213

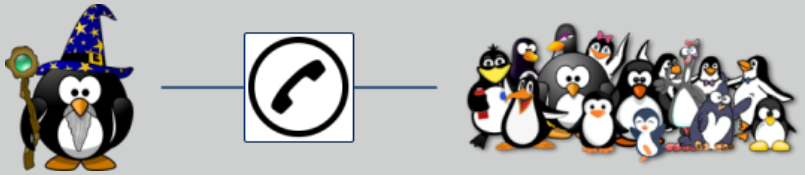
Data + Analysis Scripts

Statistical models of programmers?


The End

ROBLOX

Luau



Id | Time | Mode | Reason | Sizes

Global Counts | Edit Range Counts

Background Errors

nocheck	88.3%
nonstrict	10.45%
strict	0.16%

Type Errors

nocheck	29.62%
nonstrict	60.99%
strict	3.83%

Threats

sampling is incomplete
stx errors dominate global counts
edit ranges are coarse
no data for the intention behind edits