RDC 18

ROBLOX DEVELOPER CONFERENCE

RDC 18

ROBLOX DEVELOPER CONFERENCE

What Do Game Ratings Mean To You And Your Game

Sorcus

Who I Am

Sorcus

Focus

Build the most trusted platform for online play

To build the most trusted platform for online play, we actually need three things:

- Enable secure and exploit free game development
- Facilitate trusted and safe communication between players
- Deliver age appropriate content to our users

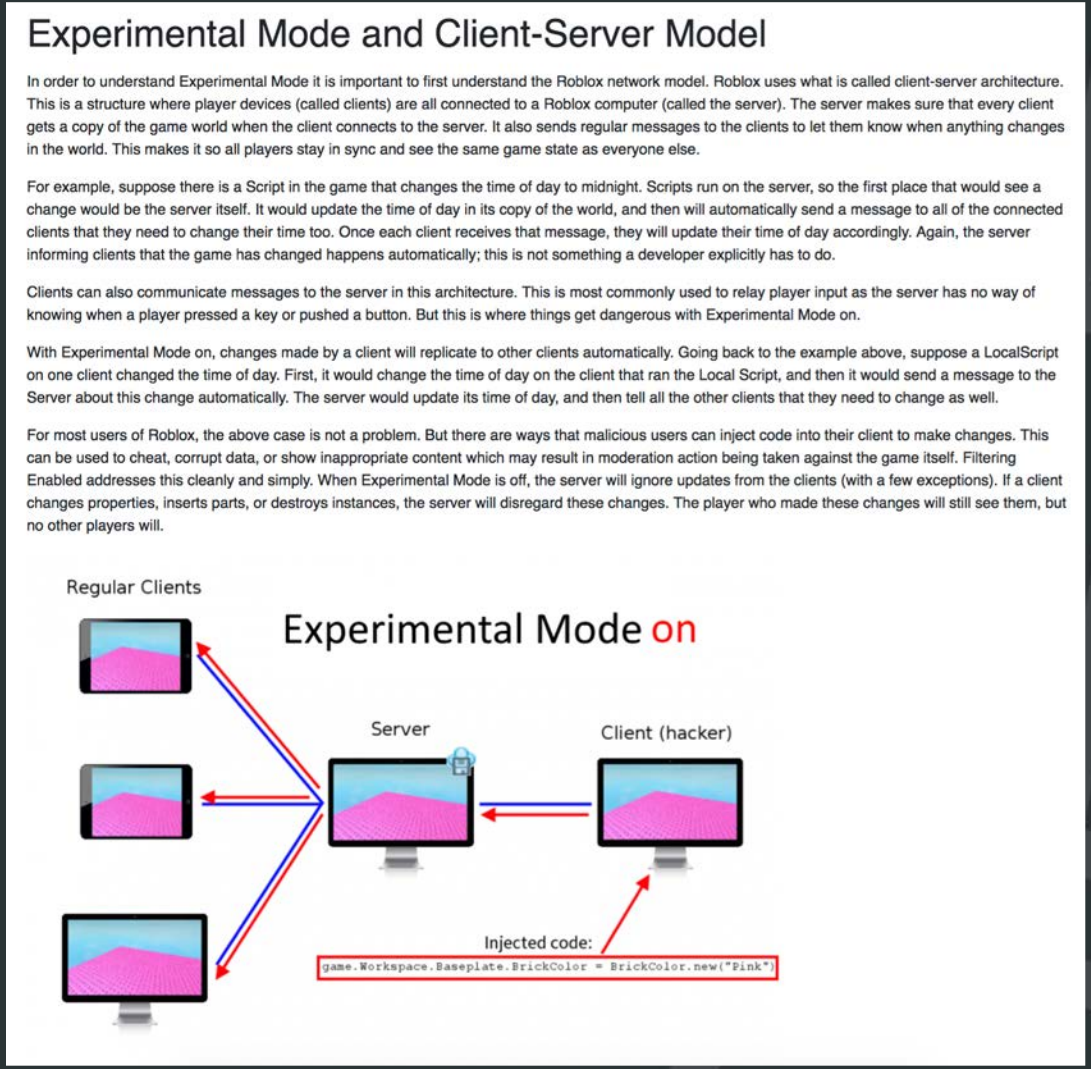
Experimental Mode Games

Making games secure

What is Experimental Mode?

Check out the Dev Hub for more information on the Client-Server Model.

If any place in a game is in Experimental Mode, your whole game is in Experimental Mode.



Experimental Mode and Client-Server Model

In order to understand Experimental Mode it is important to first understand the Roblox network model. Roblox uses what is called client-server architecture. This is a structure where player devices (called clients) are all connected to a Roblox computer (called the server). The server makes sure that every client gets a copy of the game world when the client connects to the server. It also sends regular messages to the clients to let them know when anything changes in the world. This makes it so all players stay in sync and see the same game state as everyone else.

For example, suppose there is a Script in the game that changes the time of day to midnight. Scripts run on the server, so the first place that would see a change would be the server itself. It would update the time of day in its copy of the world, and then will automatically send a message to all of the connected clients that they need to change their time too. Once each client receives that message, they will update their time of day accordingly. Again, the server informing clients that the game has changed happens automatically; this is not something a developer explicitly has to do.

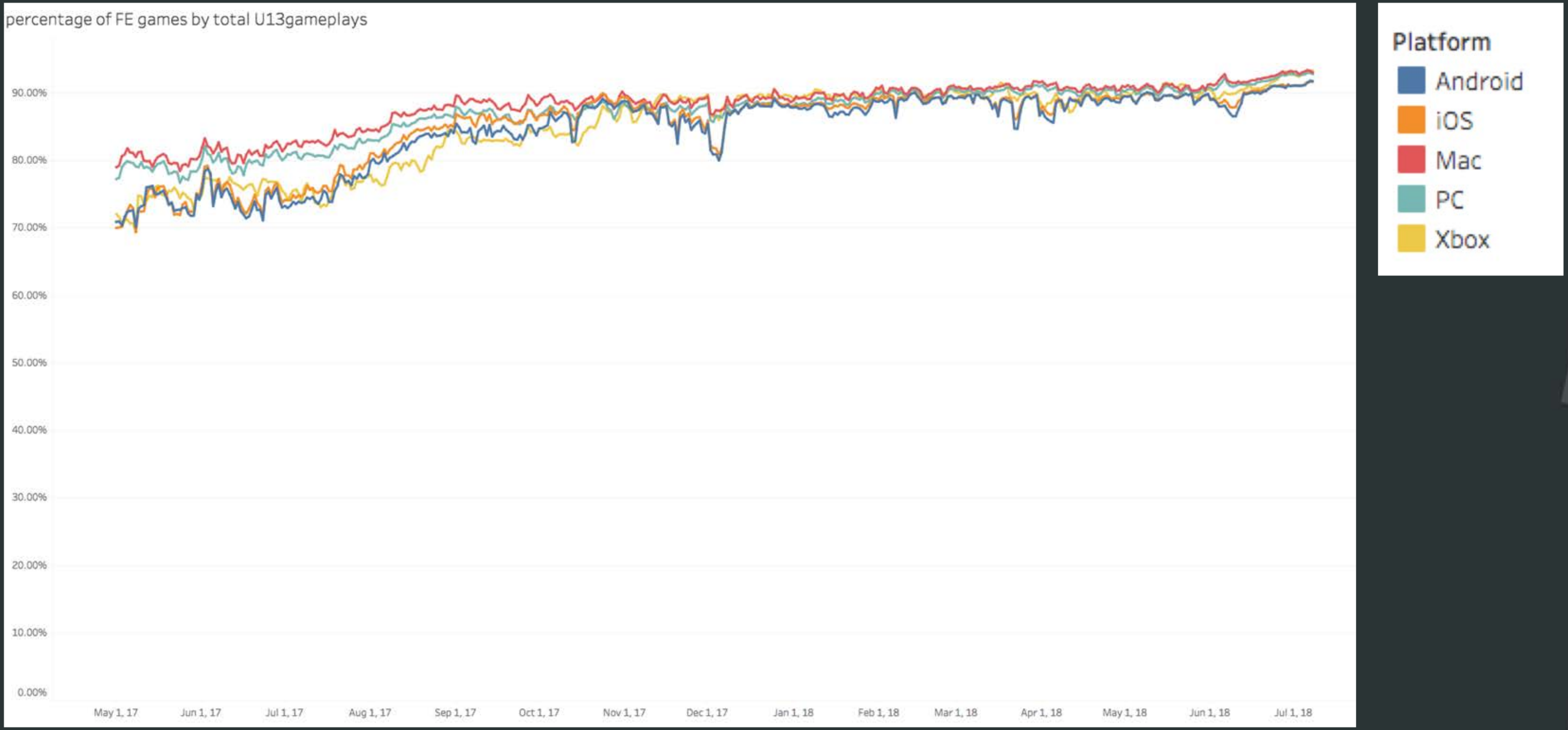
Clients can also communicate messages to the server in this architecture. This is most commonly used to relay player input as the server has no way of knowing when a player pressed a key or pushed a button. But this is where things get dangerous with Experimental Mode on.

With Experimental Mode on, changes made by a client will replicate to other clients automatically. Going back to the example above, suppose a LocalScript on one client changed the time of day. First, it would change the time of day on the client that ran the Local Script, and then it would send a message to the Server about this change automatically. The server would update its time of day, and then tell all the other clients that they need to change as well.

For most users of Roblox, the above case is not a problem. But there are ways that malicious users can inject code into their client to make changes. This can be used to cheat, corrupt data, or show inappropriate content which may result in moderation action being taken against the game itself. Filtering Enabled addresses this cleanly and simply. When Experimental Mode is off, the server will ignore updates from the clients (with a few exceptions). If a client changes properties, inserts parts, or destroys instances, the server will disregard these changes. The player who made these changes will still see them, but no other players will.

```
game.Workspace.Baseplate.BrickColor = BrickColor.new("Pink")
```

Playtime for Experimental Mode games



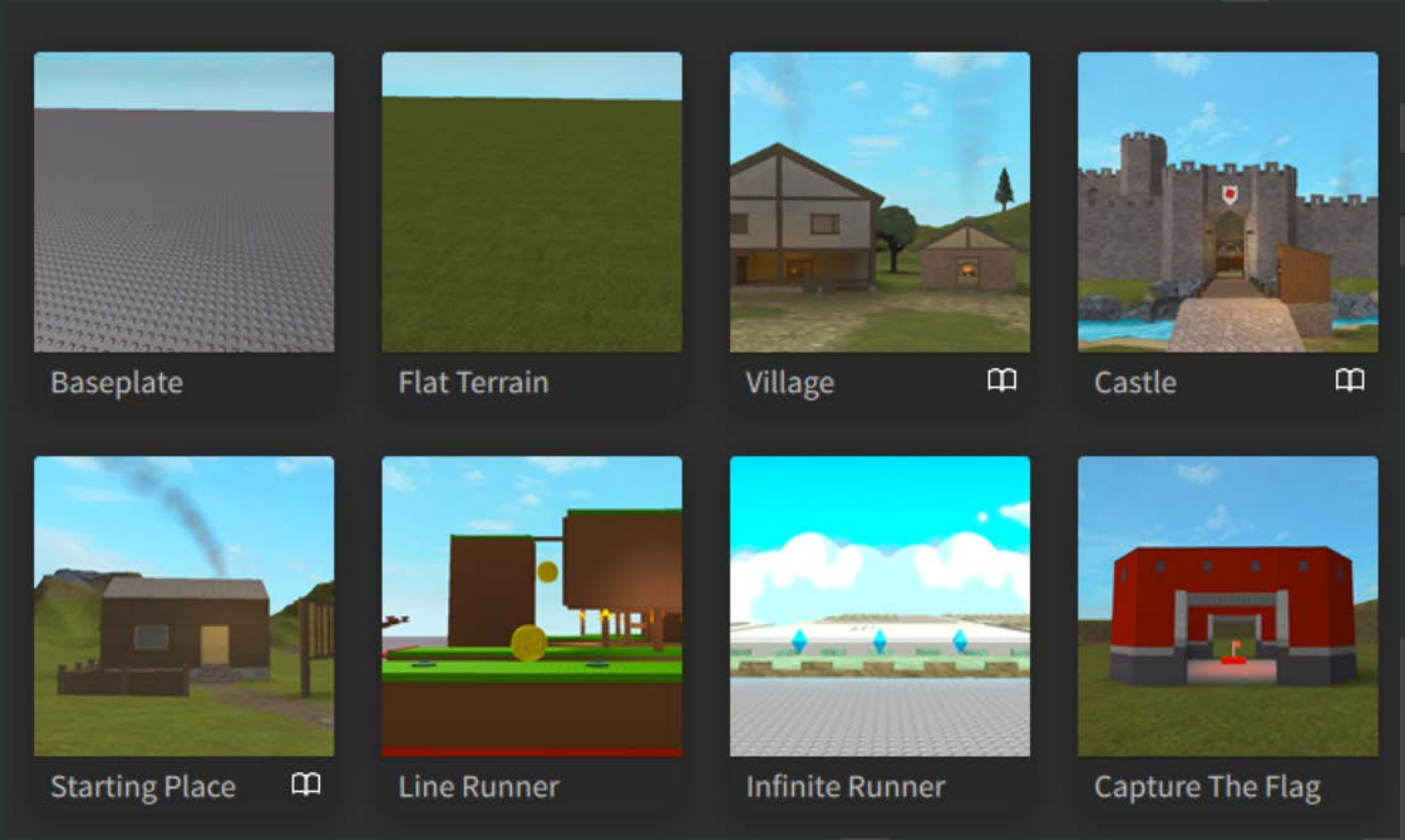
Sorts & Search

Sorts: Turning off Experimental Mode is a requirement for your game to in sorts. This is the current behavior.

Search: Experimental games will be shown to all users in search.

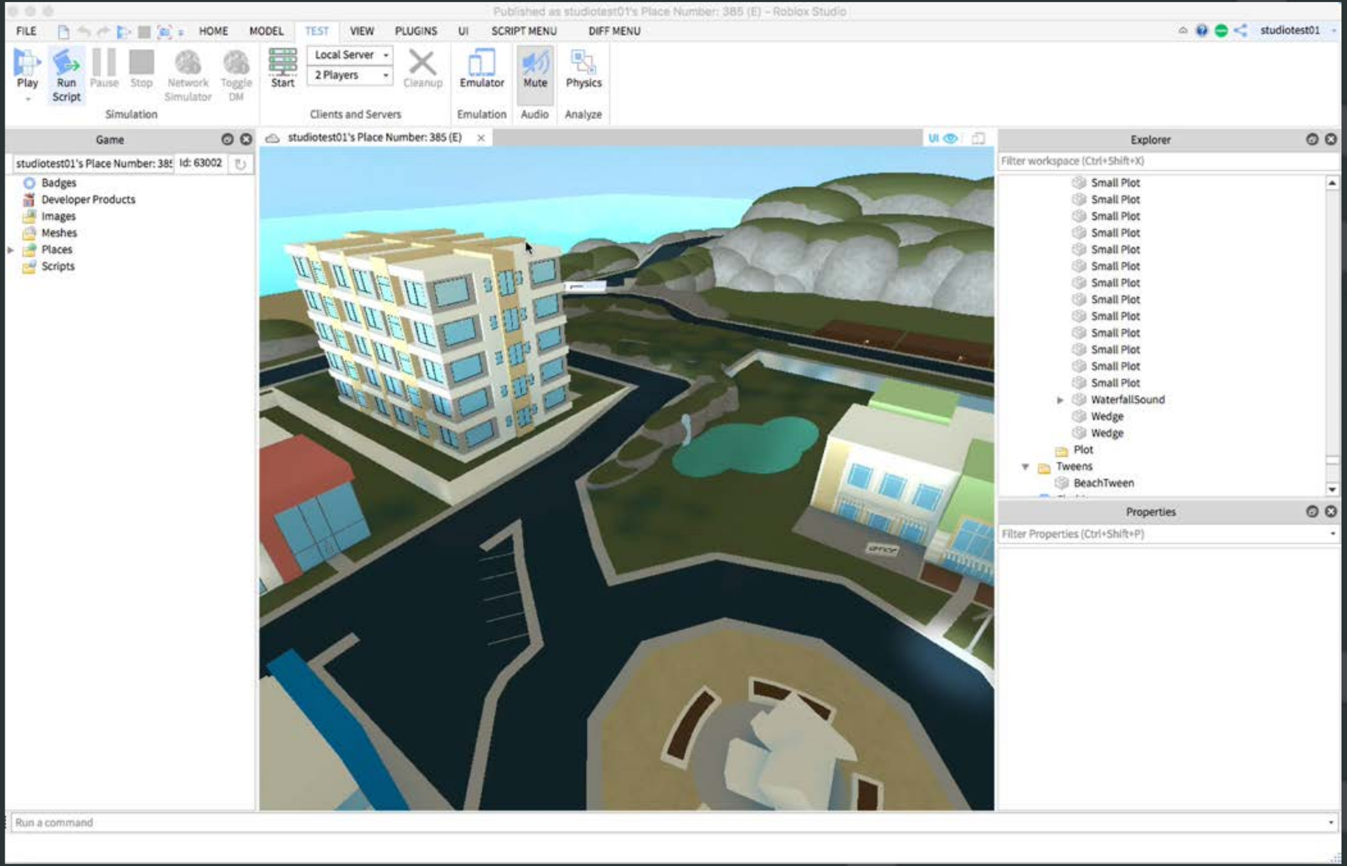
Templates Are Updated

All templates have been set to have Experimental Mode off by default



Improved testing in Studio

Play solo testing in Studio is now Server/Client compatible



Game Ratings

Rating games for appropriate audiences

Game Ratings overview

Game Ratings will:

- Automatically limit play of more mature games to U13 audience
- Empower parents with necessary ability to customize what types of games are available for their kids

What are the ratings?

- Games for everyone
- Games for all users 8 and above
- Games for all users 13 and above
- Mature – Not allowed
- Unrated – default for all games until rated

** Rating names and definitions still TBD*

Unrated Games

Unrated games include:

- Games where developers have not yet assigned a rating
- Newly rated games which have not been vetted by the community yet
- Experimental Mode games (they can never get a rating)
- Your friends can still play your unrated games

Unrated games can only be played by 13+ users.

Verifying Ratings

- 13+ players can provide feedback on the game rating.
- Falsifying ratings may result in some moderation action.
- If majority of the feedback matches your rating for the game it becomes verified.
- Game now becomes available for U13.

Account Settings

- Parents have ability to lock down an U13 account to restrict ratings that can be played.

- By default U13 can play G, PG rated games.

Game Updates

Updating your game does not affect your ratings unless you choose to change the ratings.

Changing your rates will have different behavior depending on whether it becomes more restrictive or more open

For ex. G to PG will not require re-vetting by the community but PG to G will trigger the vetting process.

RDC[18]

ROBLOX DEVELOPER CONFERENCE

Q & A