Google
Developers

Mobile Multiplayer Made Manageable

Wolff Dobson, Bruno Oliveira, Dan Rodney
Android

Why multiplayer?

It’s fun.

engagement

challenging

engagement

replay value

challenging

engagement

replay value

challenging

engagement

competition

replay value

challenging

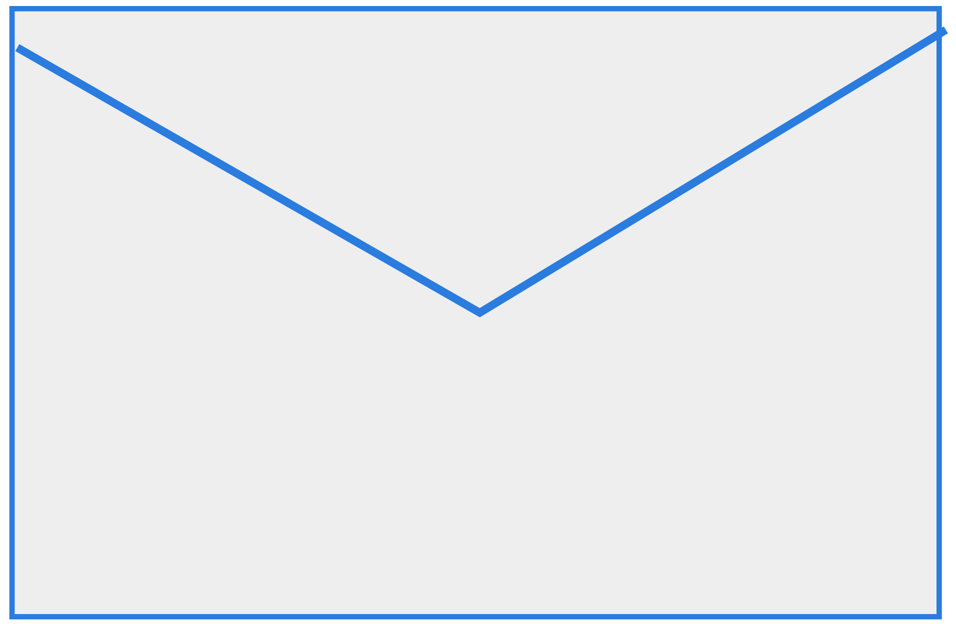
engagement

retention

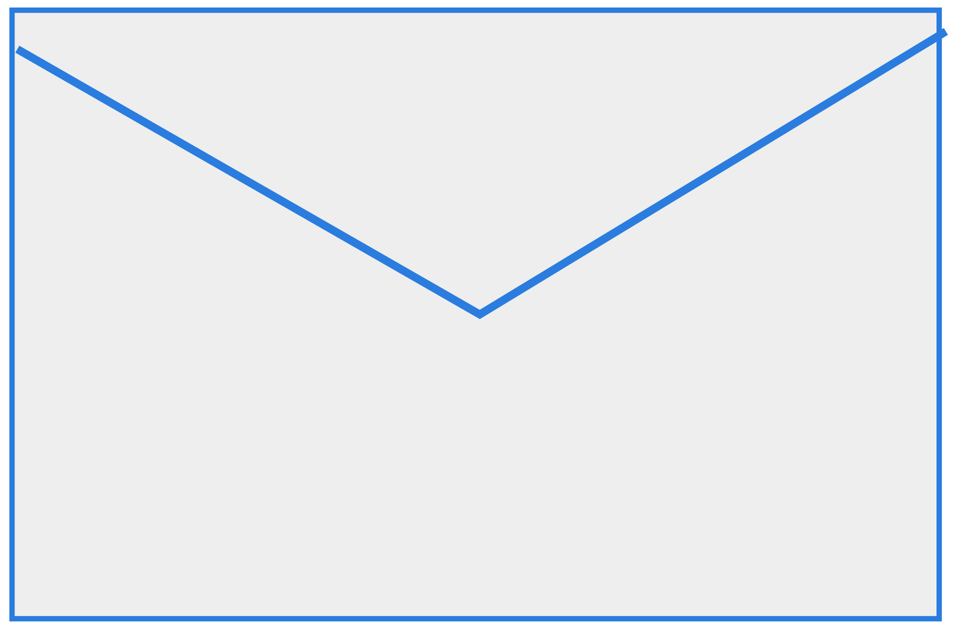
competition

the social graph

the social graph



the social graph



the social graph

disruption

Q: If multiplayer is so cool,
why isn't everybody doing it?

A: It’s hard.

Multiplayer is complicated!

Multiplayer is complicated!

infrastructure

identity

invitations

notification

automatching

connectivity

transport

Multiplayer is complicated!

infrastructure
identity
invitations
notification
automatching
connectivity
transport

game logic
world model
how to make
your game fun

boring stuff

infrastructure
identity
invitations
notification
automatching
connectivity
transport

game logic
world model
how to make
your game fun

boring stuff

- infrastructure
- identity
- invitations
- notification
- automatching
- connectivity
- transport

your job!

- game logic
- world model
- how to make your game fun

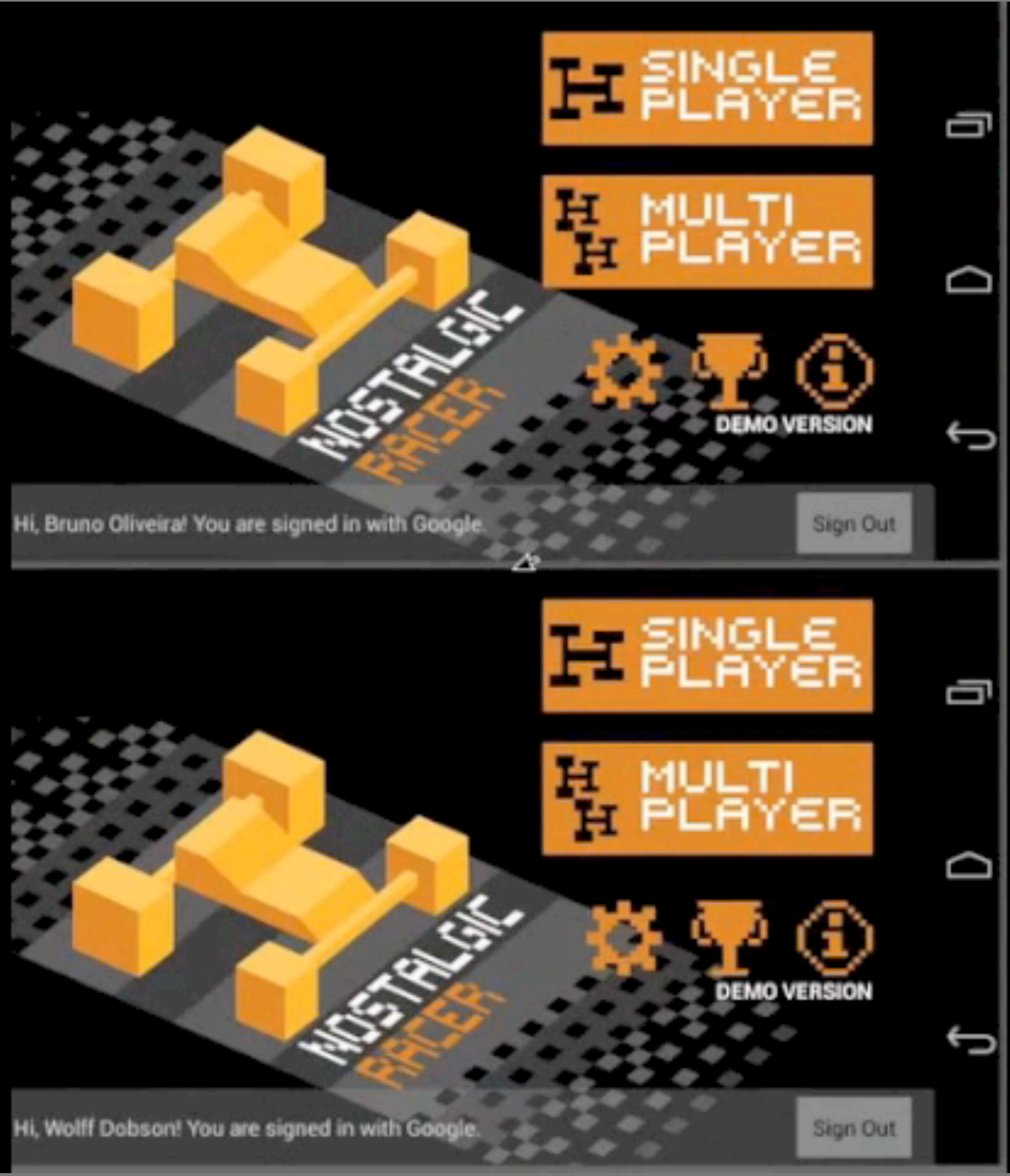
Availability of Multiplayer Features - Android

Froyo and above

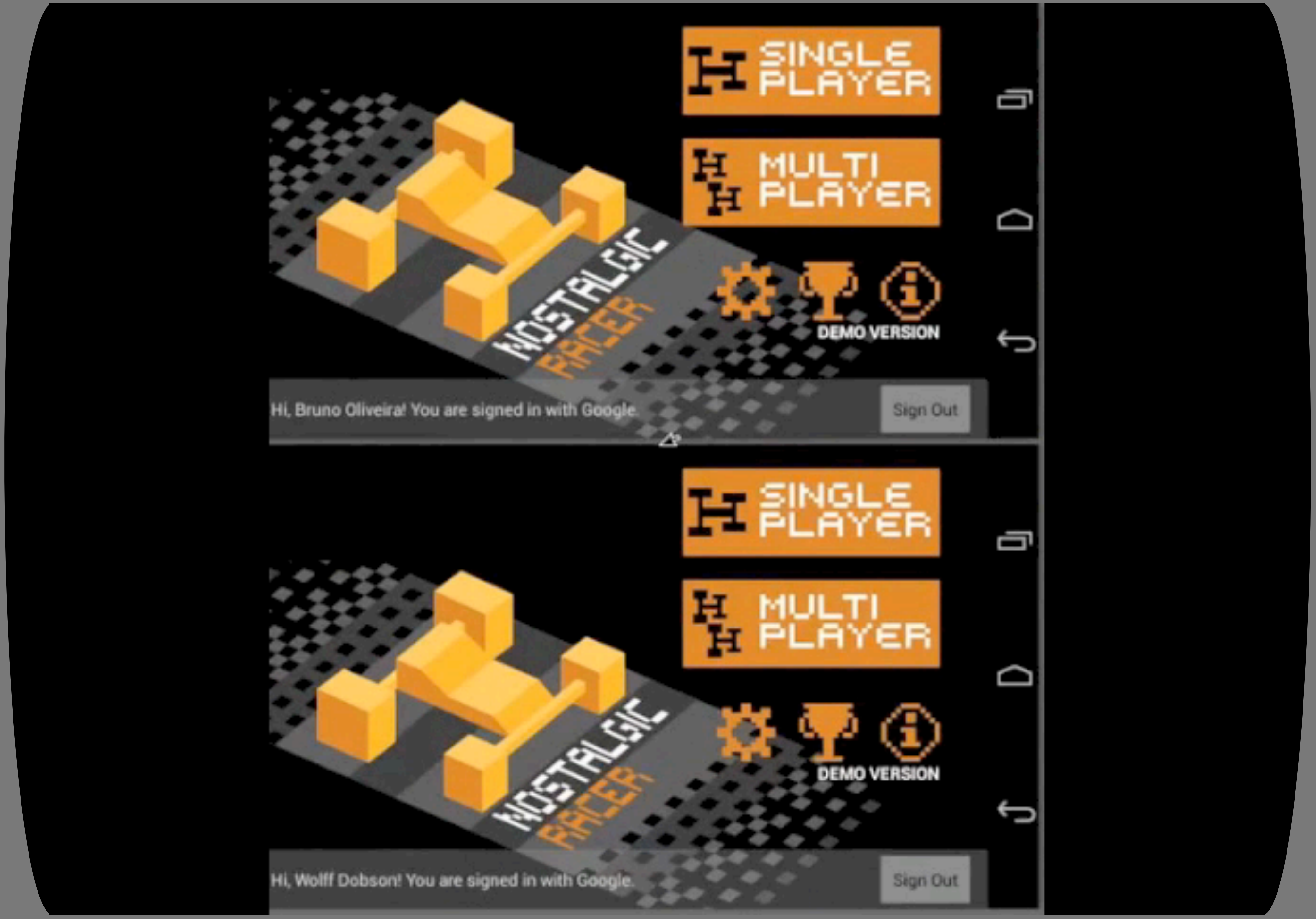
Google Play Services

Google+ account (app not required)

Nostalgic Racer!



NOSTALGIVISION 2000



Hi, Bruno Oliveira! You are signed in with Google.

Hi, Wolff Dobson! You are signed in with Google.

NOSTALGIVISION 2000

That was cool.

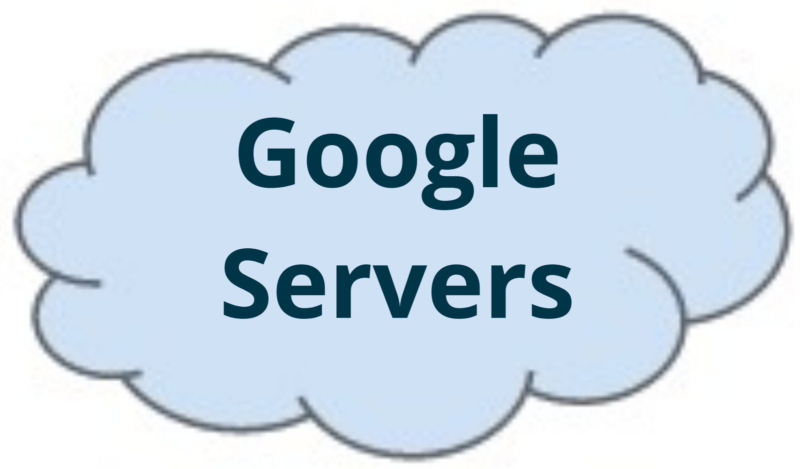
That was cool.

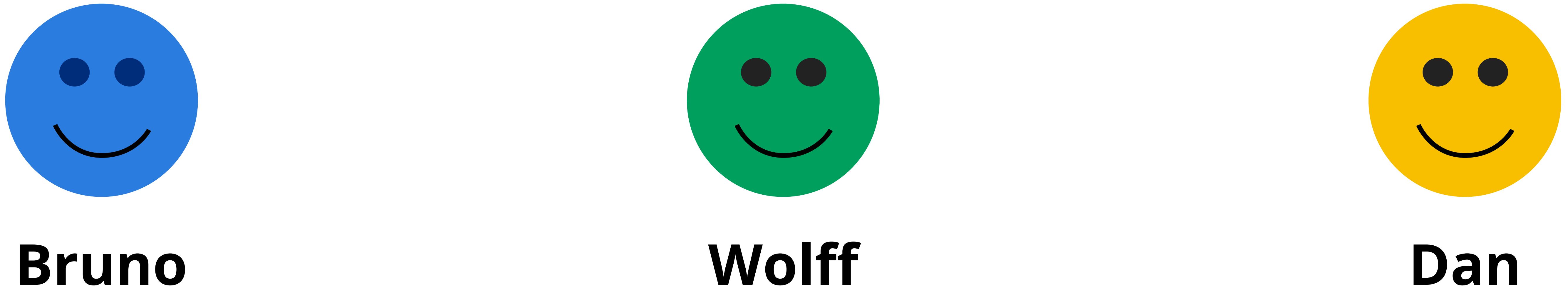
What just happened?

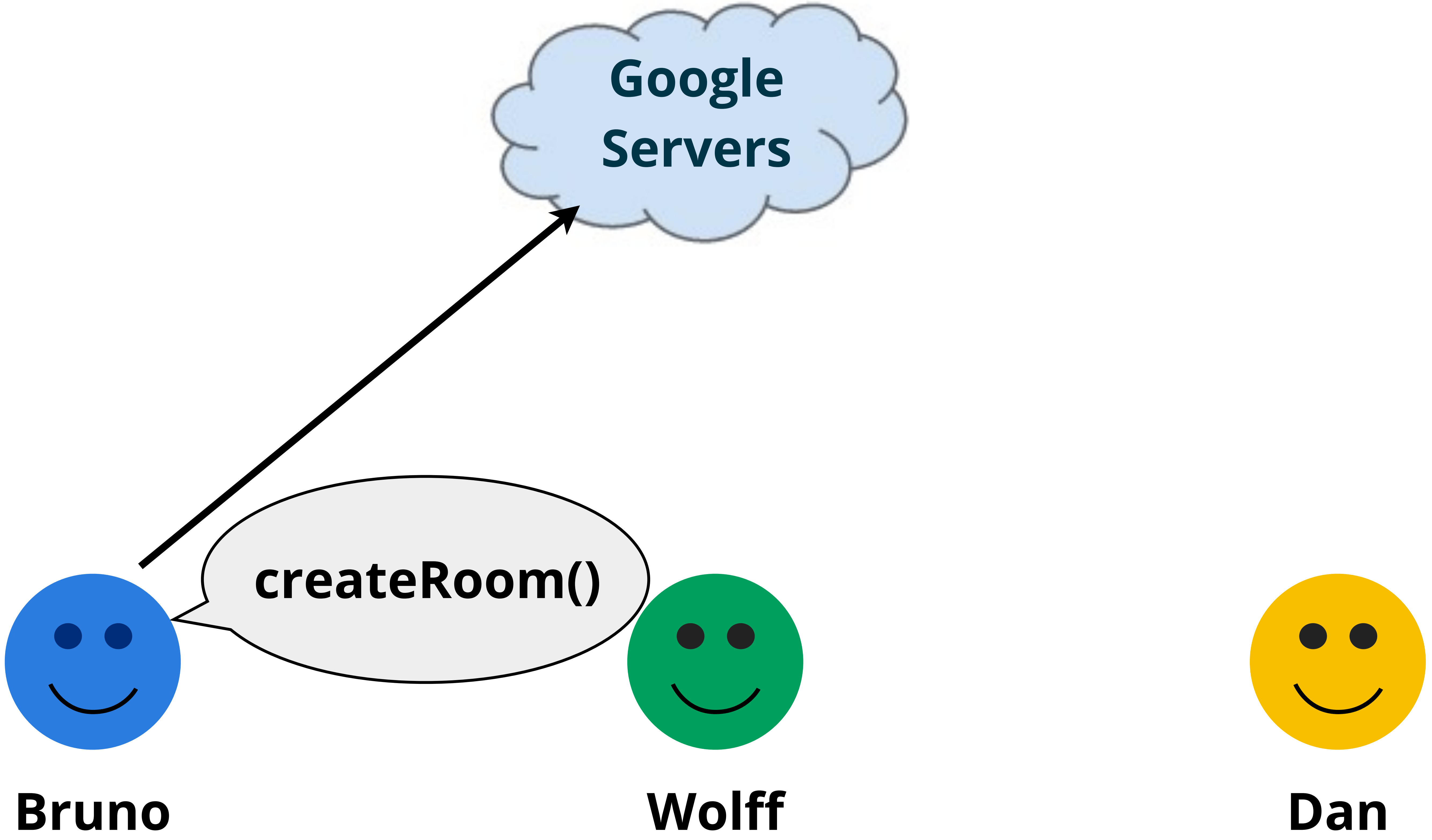
Bruno

Wolff

Dan


Google
Servers

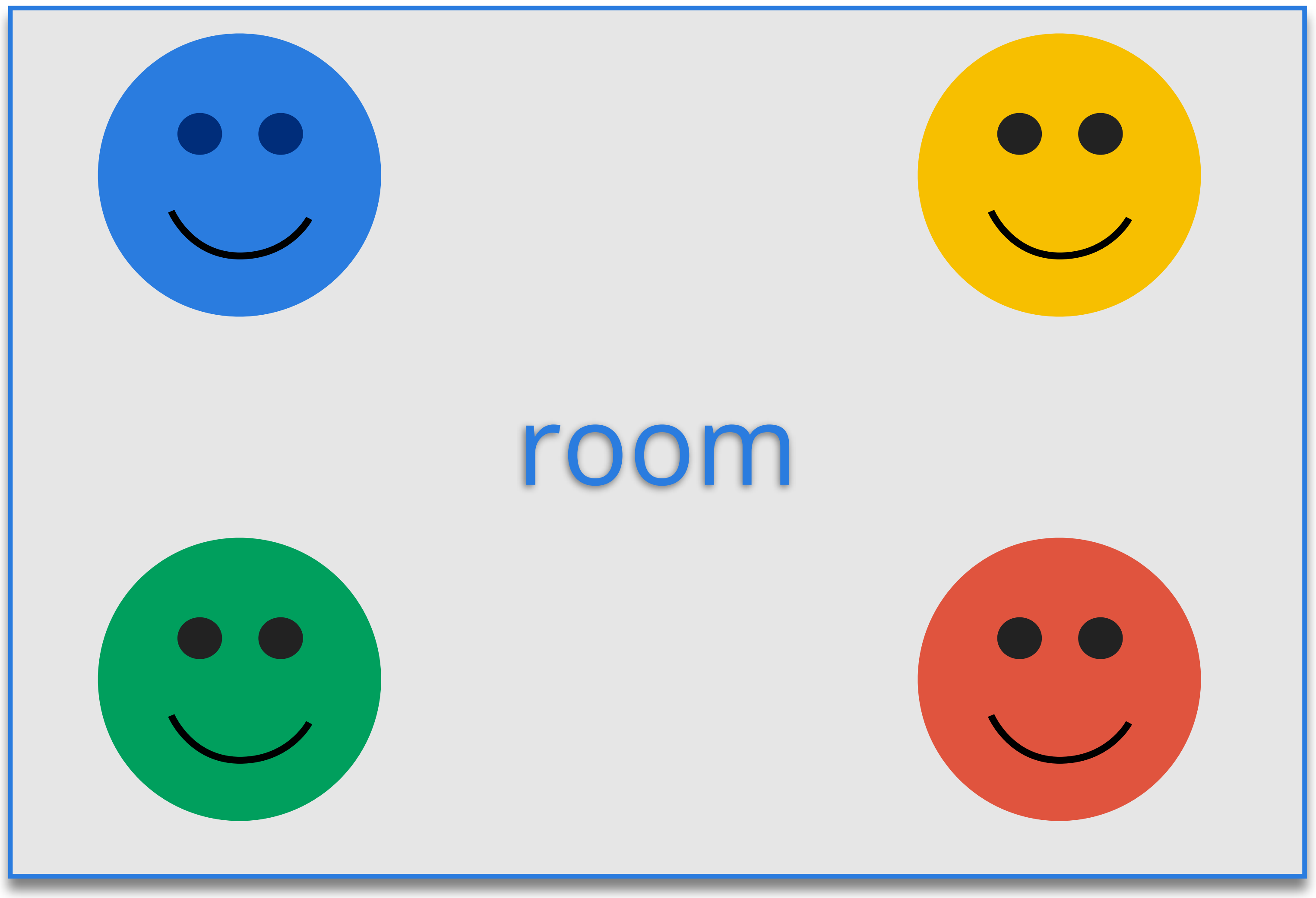

Bruno
Wolff
Dan

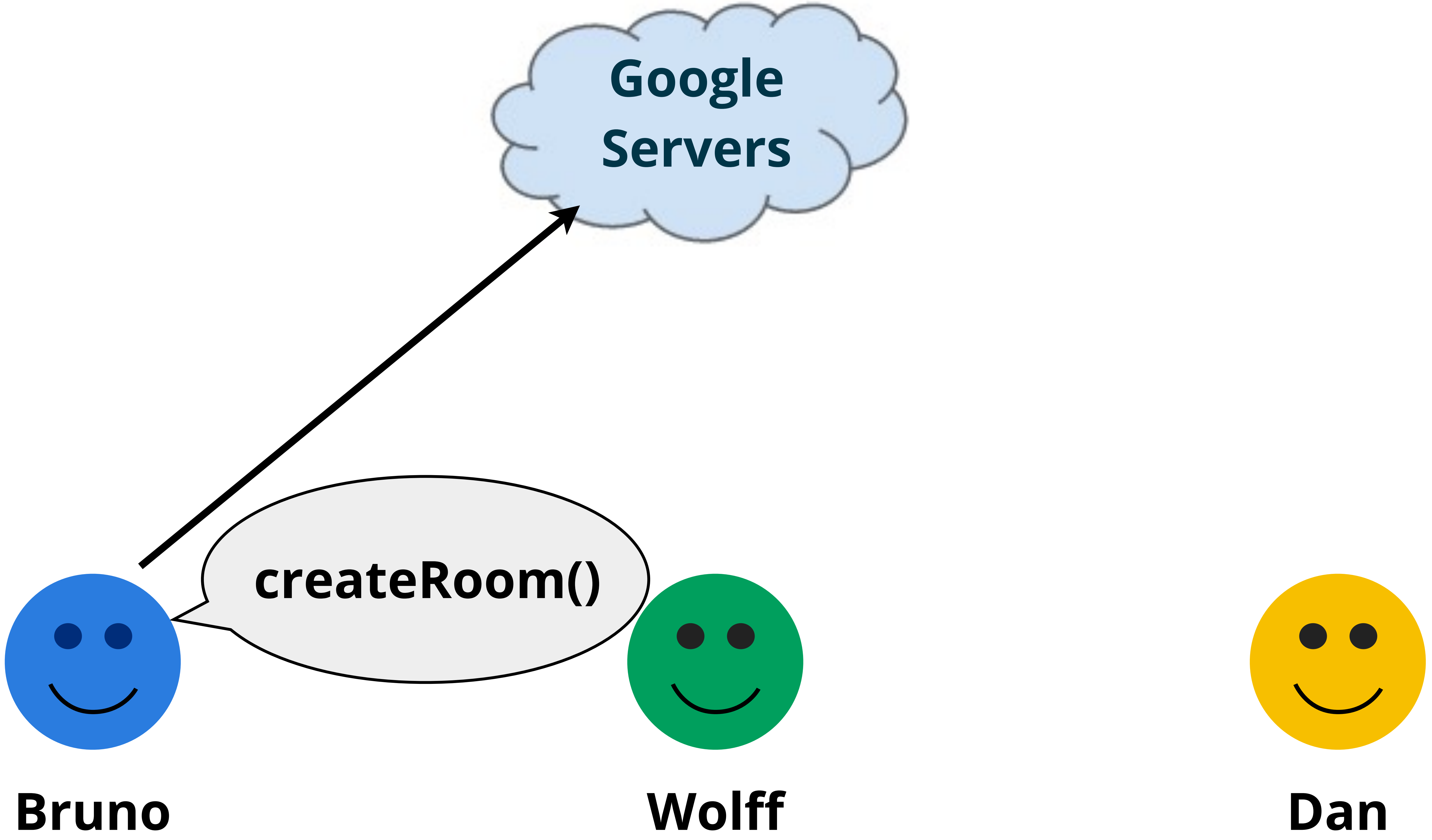

Google Servers
createRoom()
Bruno
Wolff
Dan

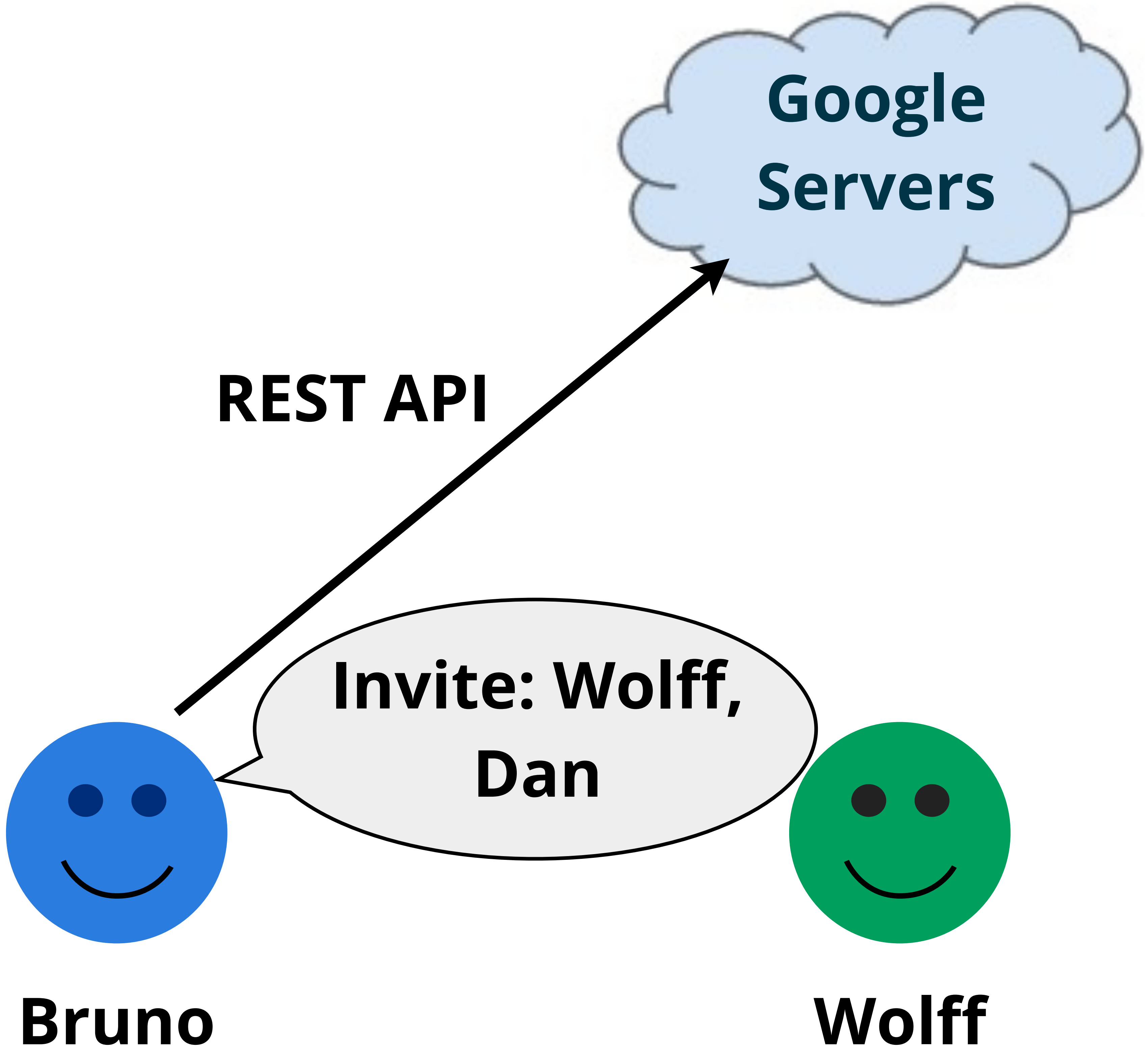
room, n. [/rōōm/]

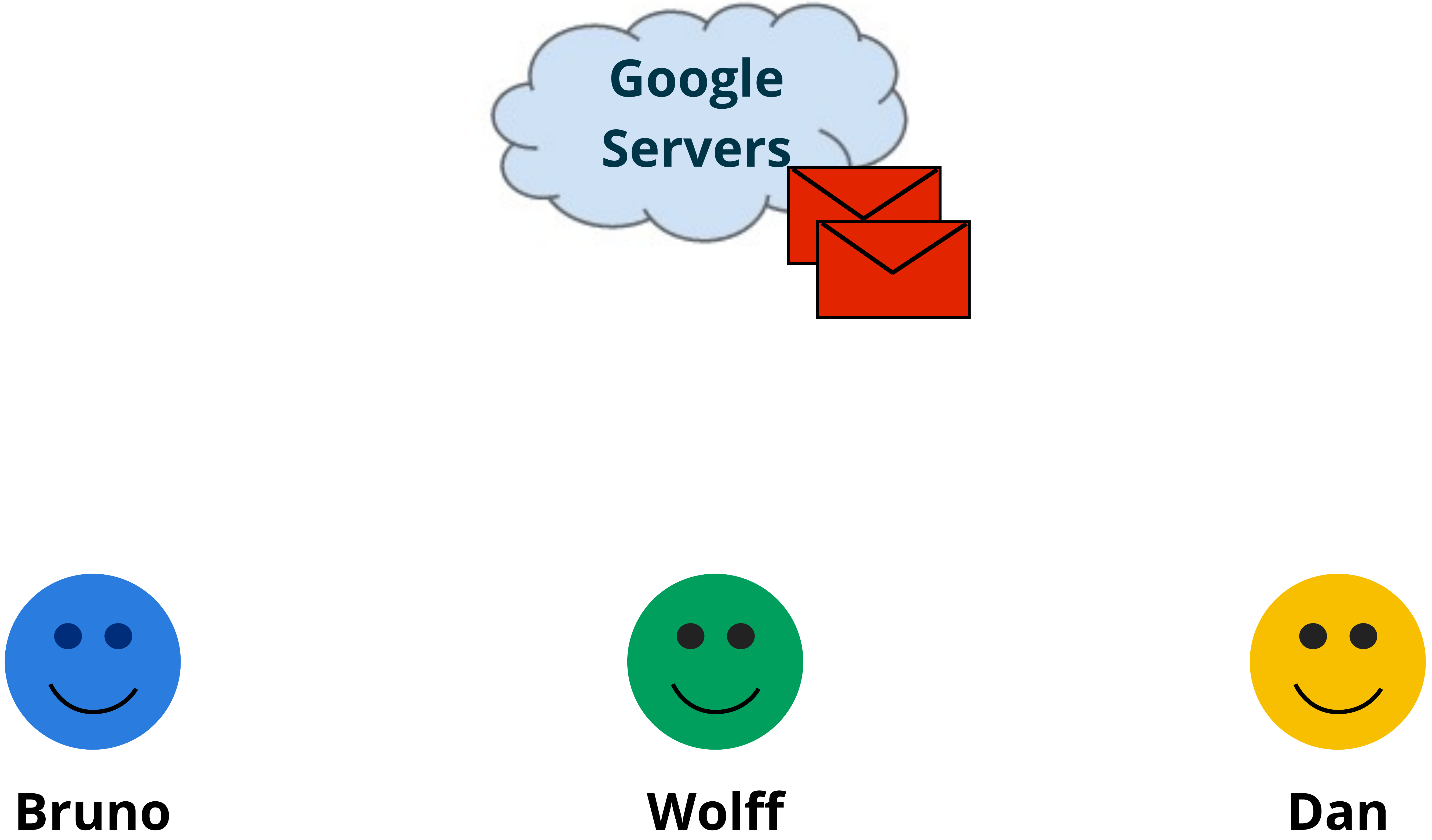
1. space where something can be done; 2. each of the delimited spaces in a building; 3. amount of space; 4. a section of a structure.

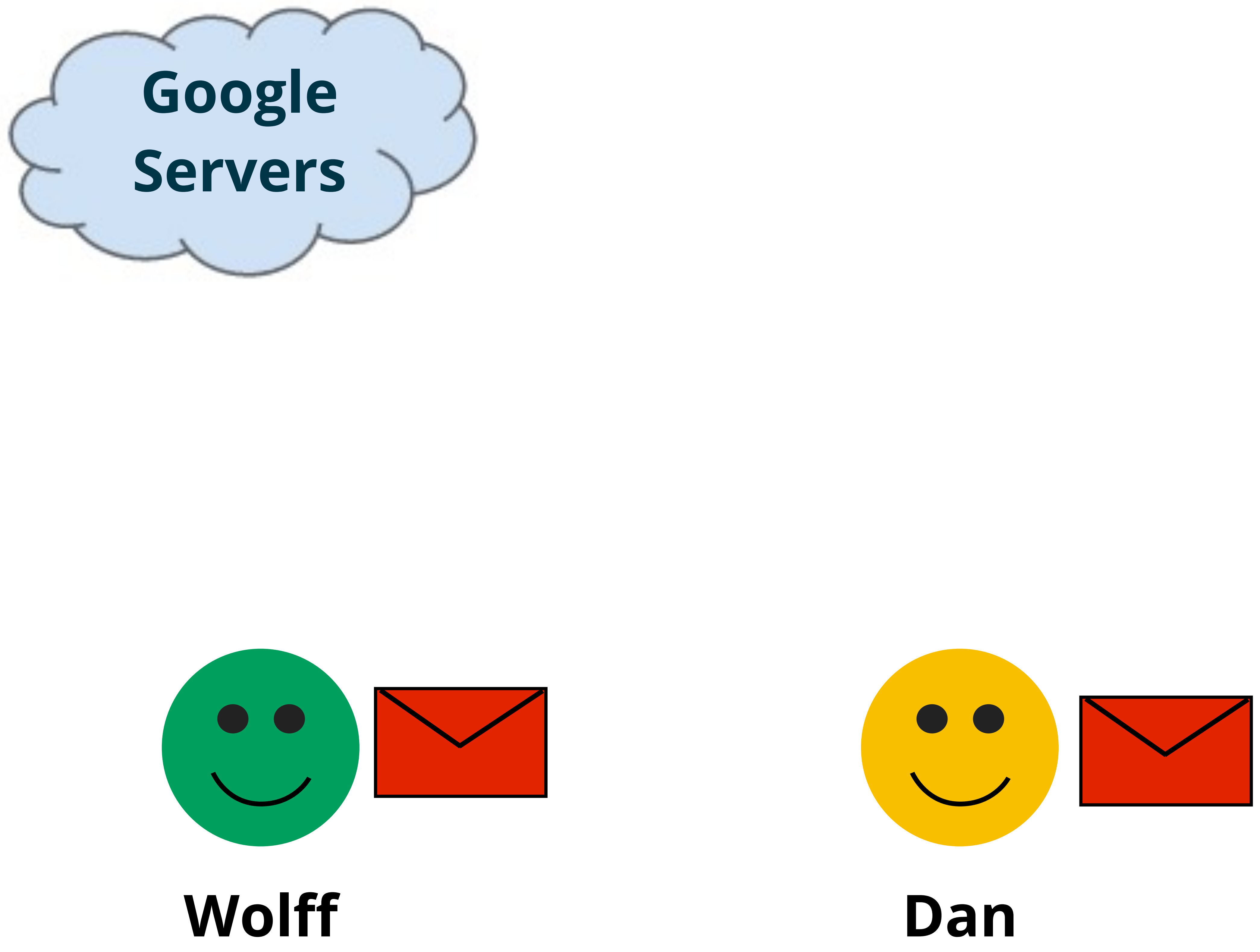
Room is where a game takes place.

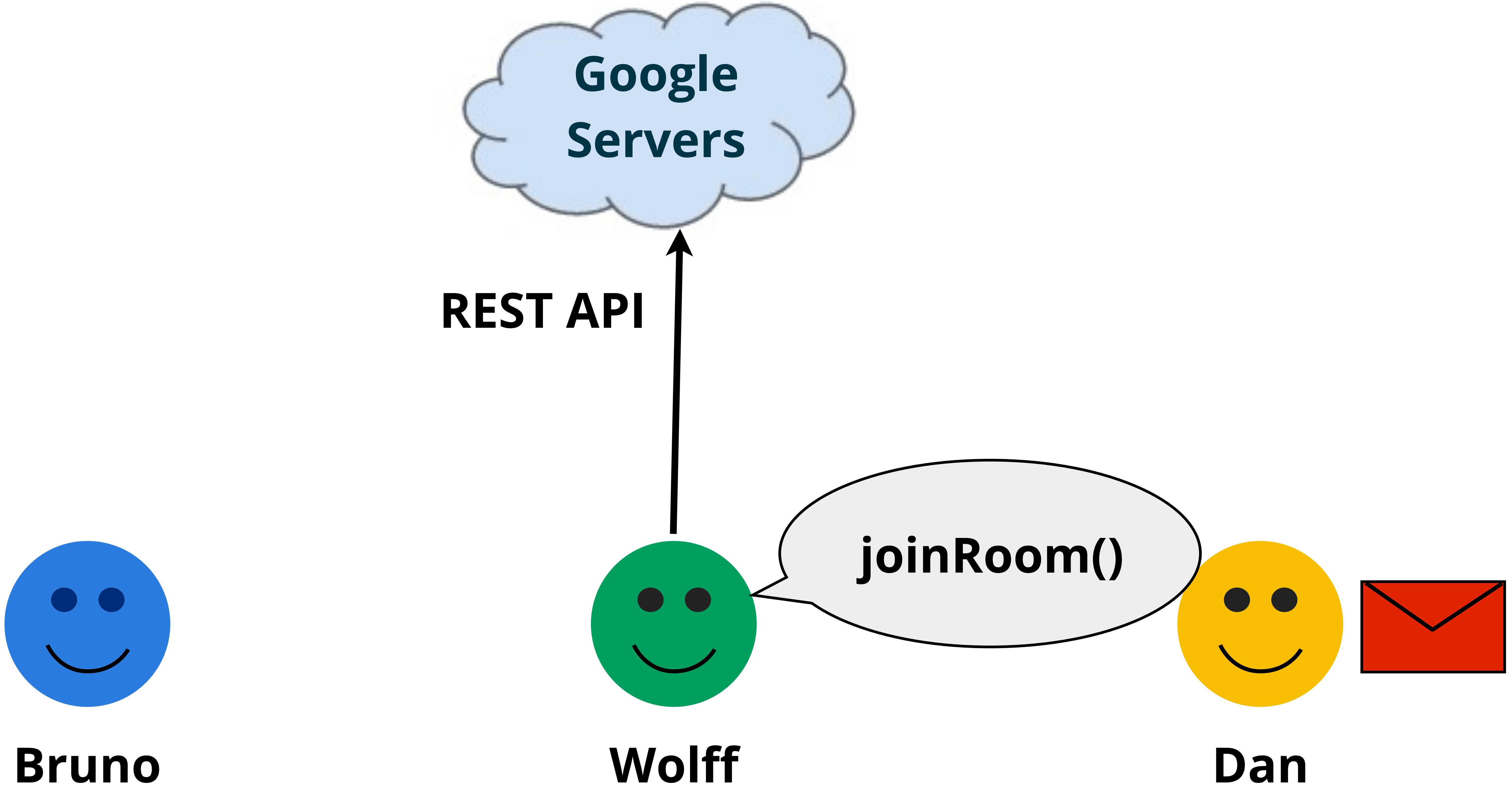


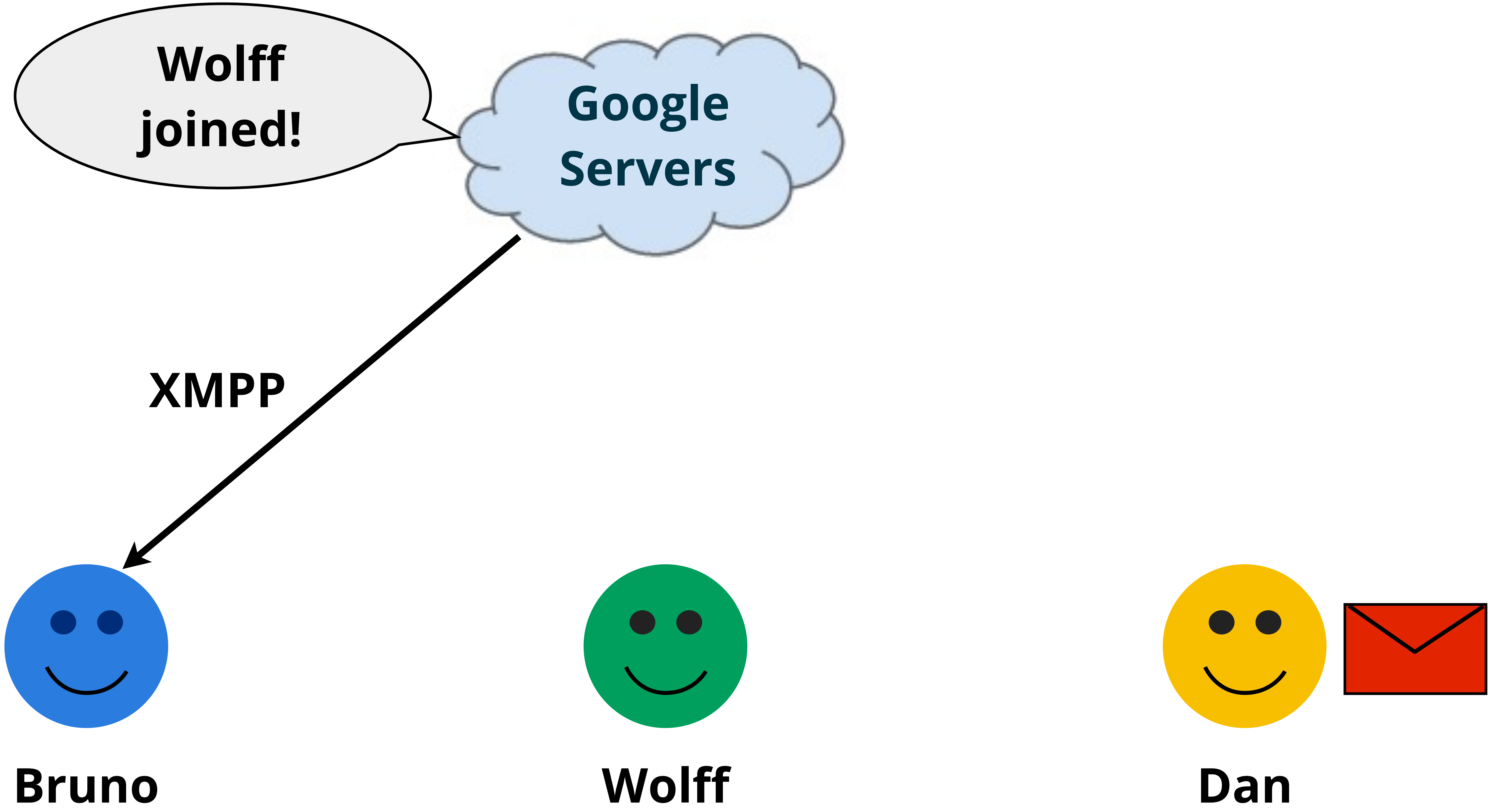

Google Servers
createRoom()
Bruno
Wolff
Dan

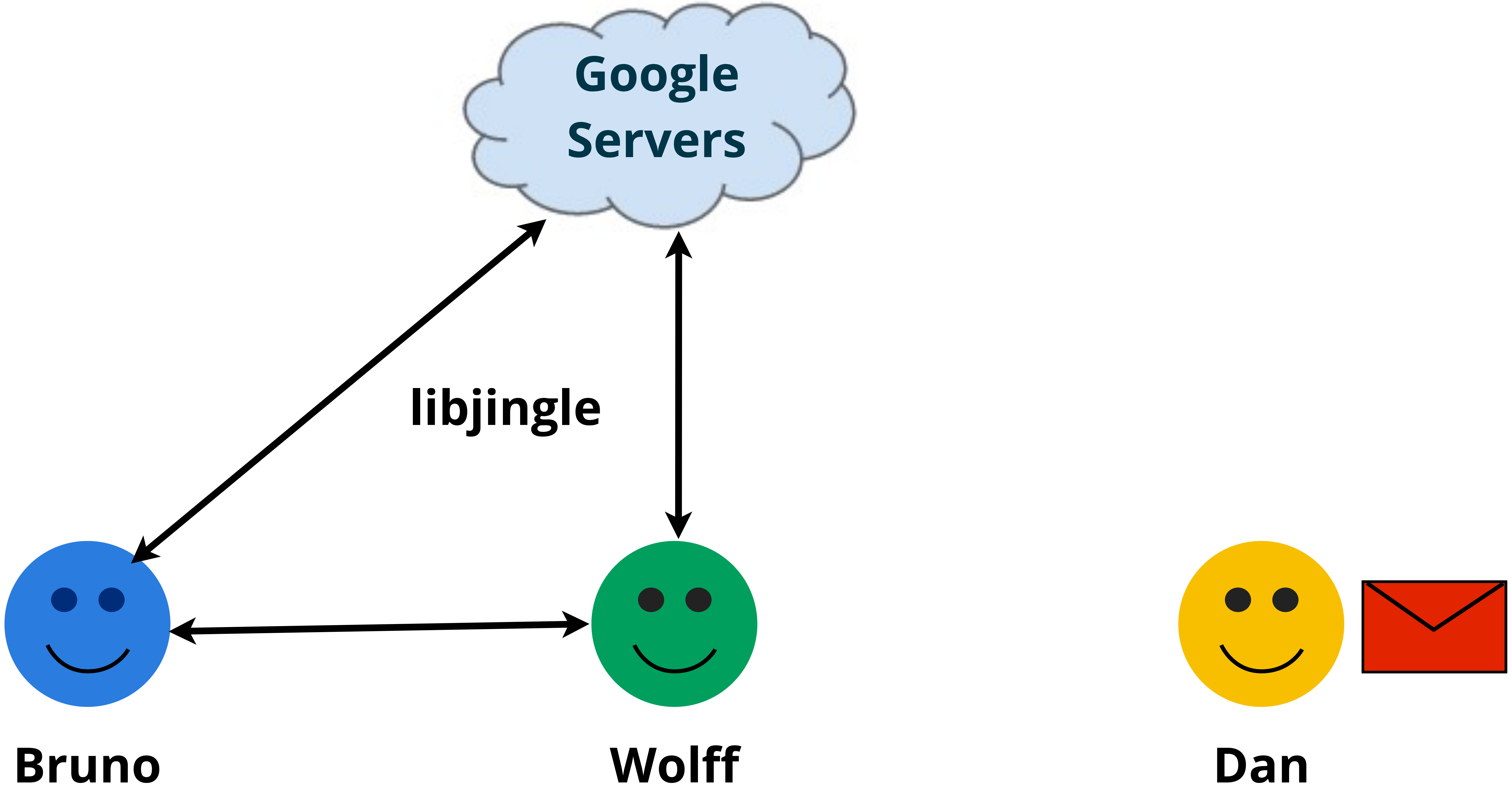

Google Servers
REST API
Invite: Wolff, Dan
Bruno
Wolff
Dan

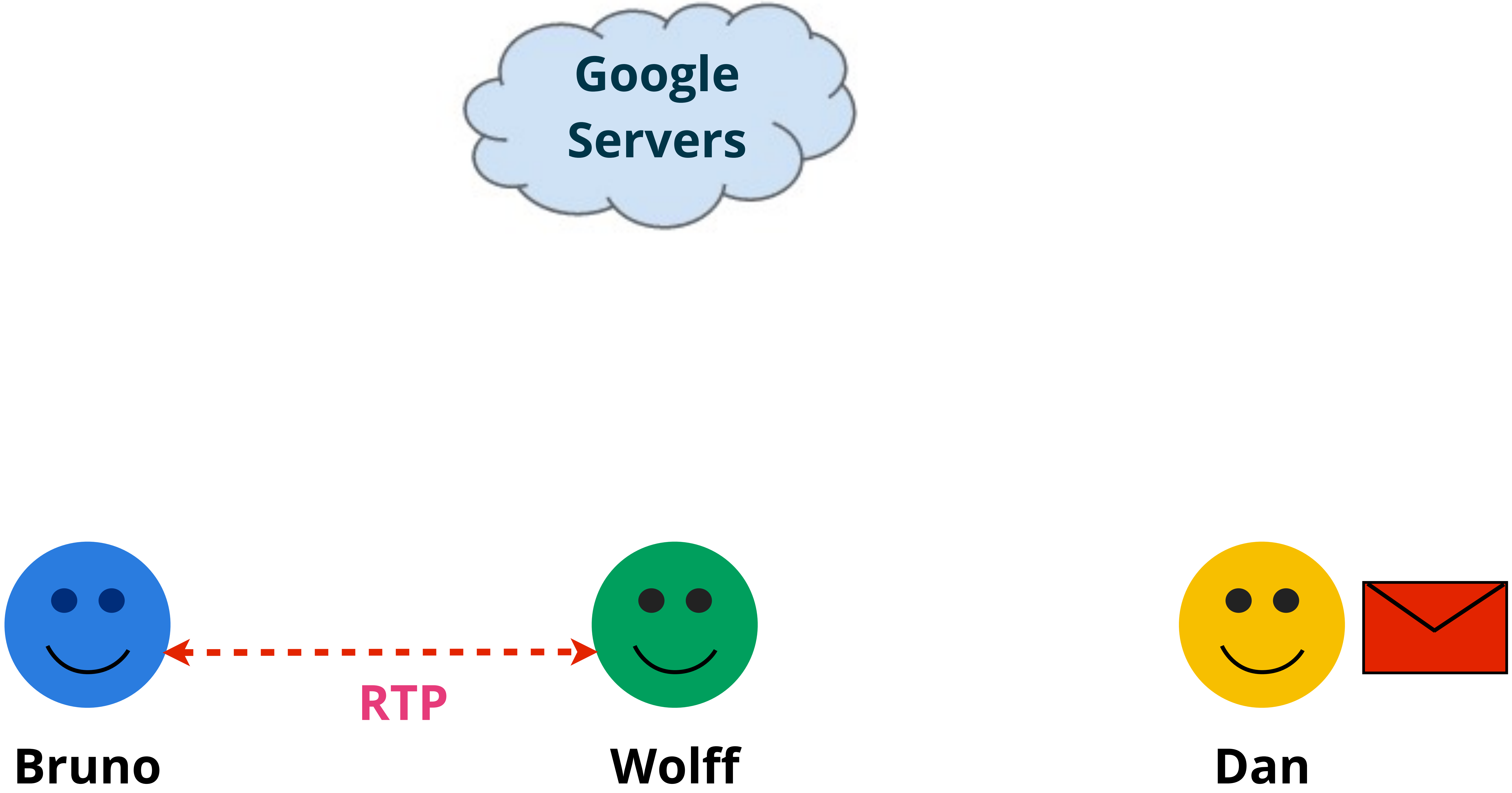

Google
Servers
Bruno
Wolff
Dan

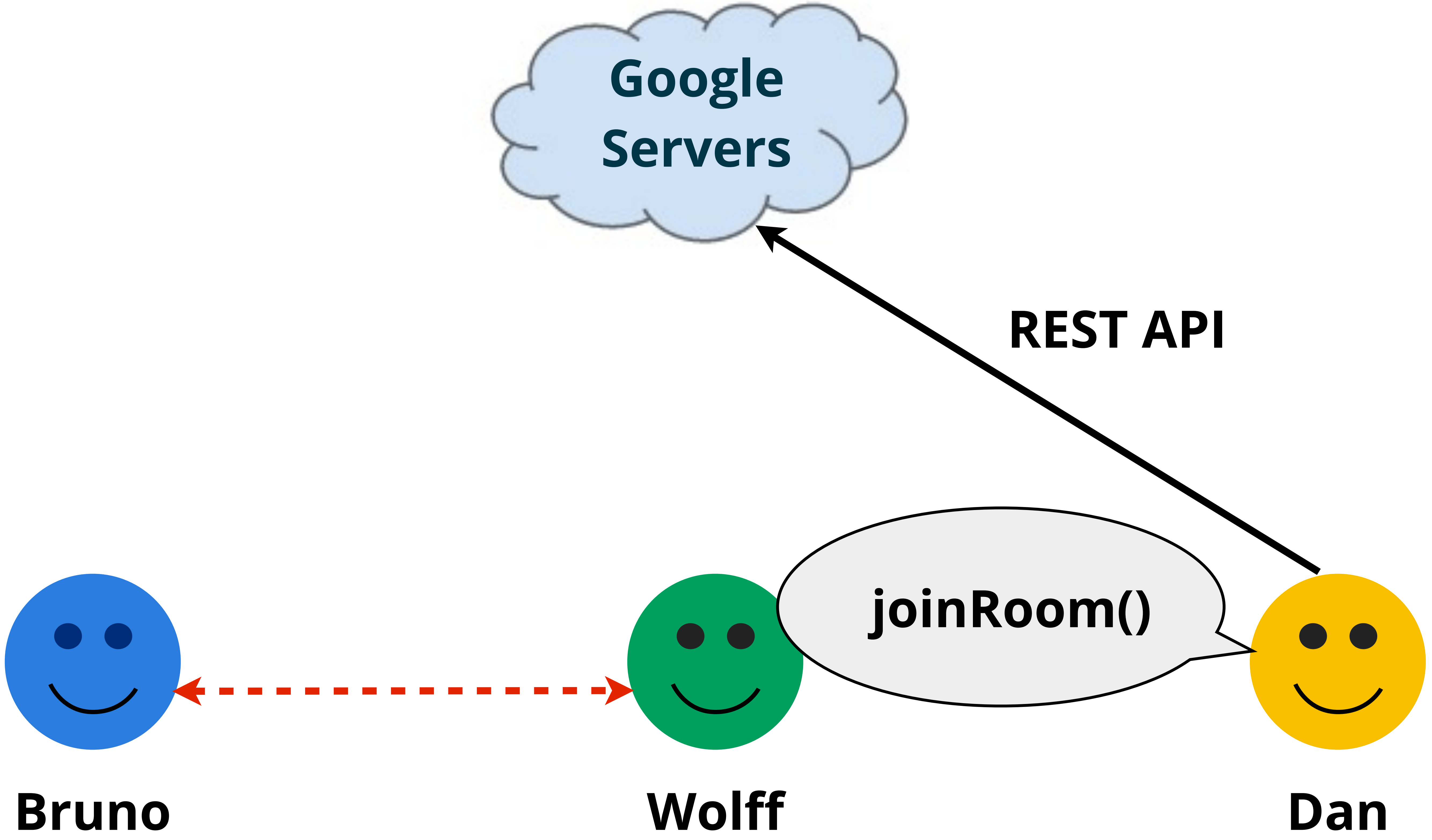

Google Servers
Bruno
Wolff
Dan

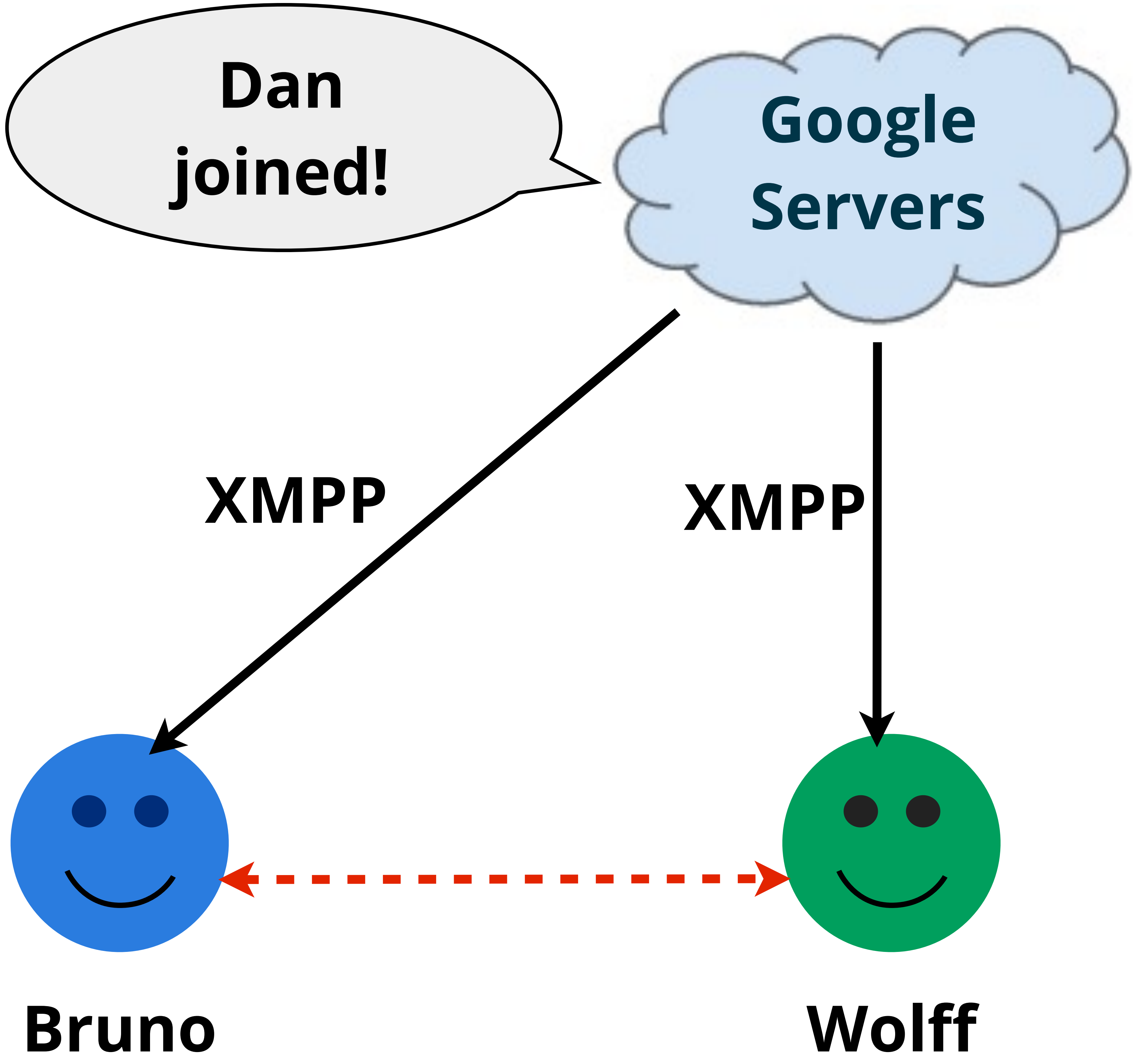

Google Servers
REST API
joinRoom()
Bruno
Wolff
Dan

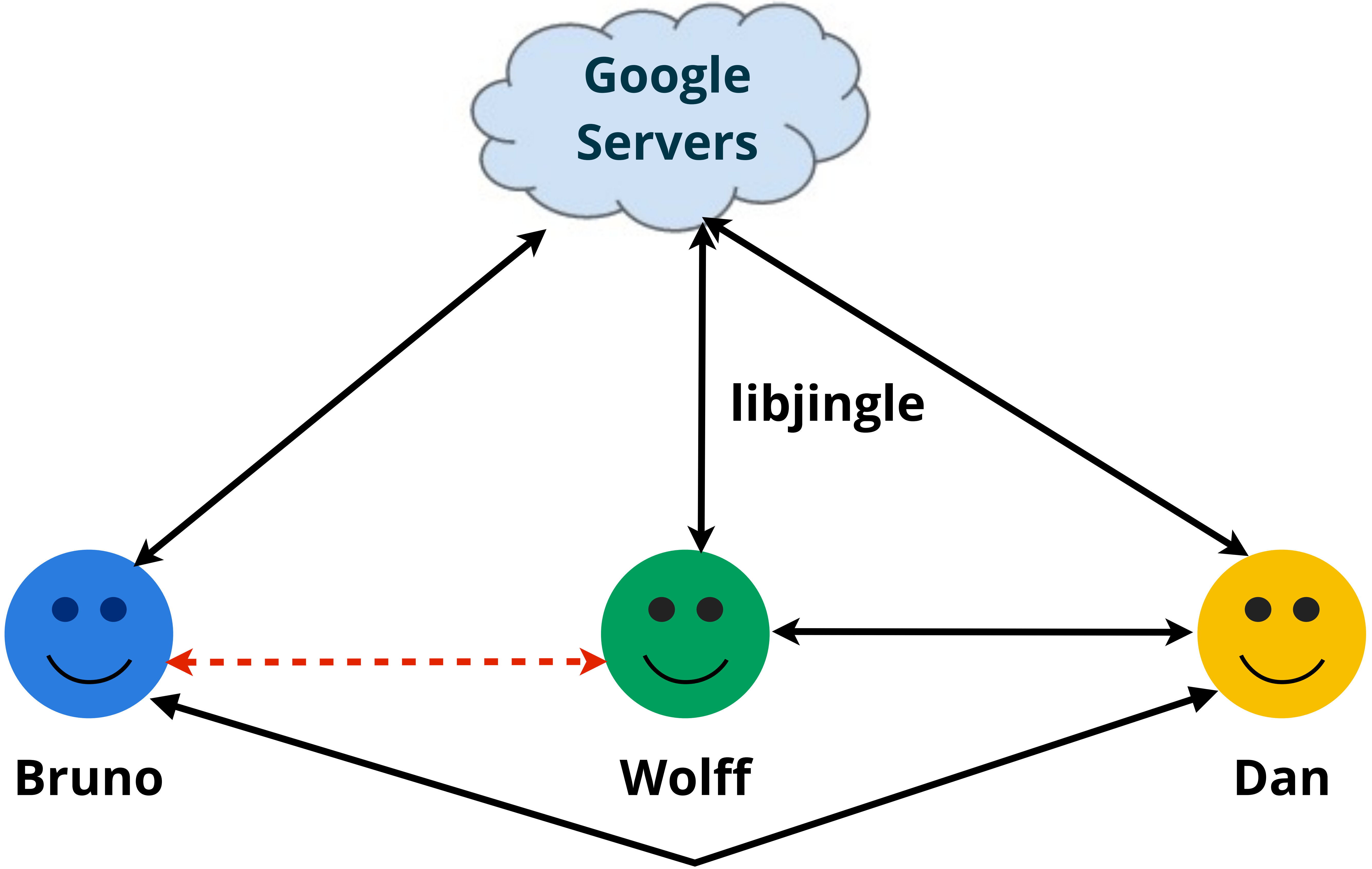

Wolff joined!
Google Servers
XMPP
Bruno
Wolff
Dan

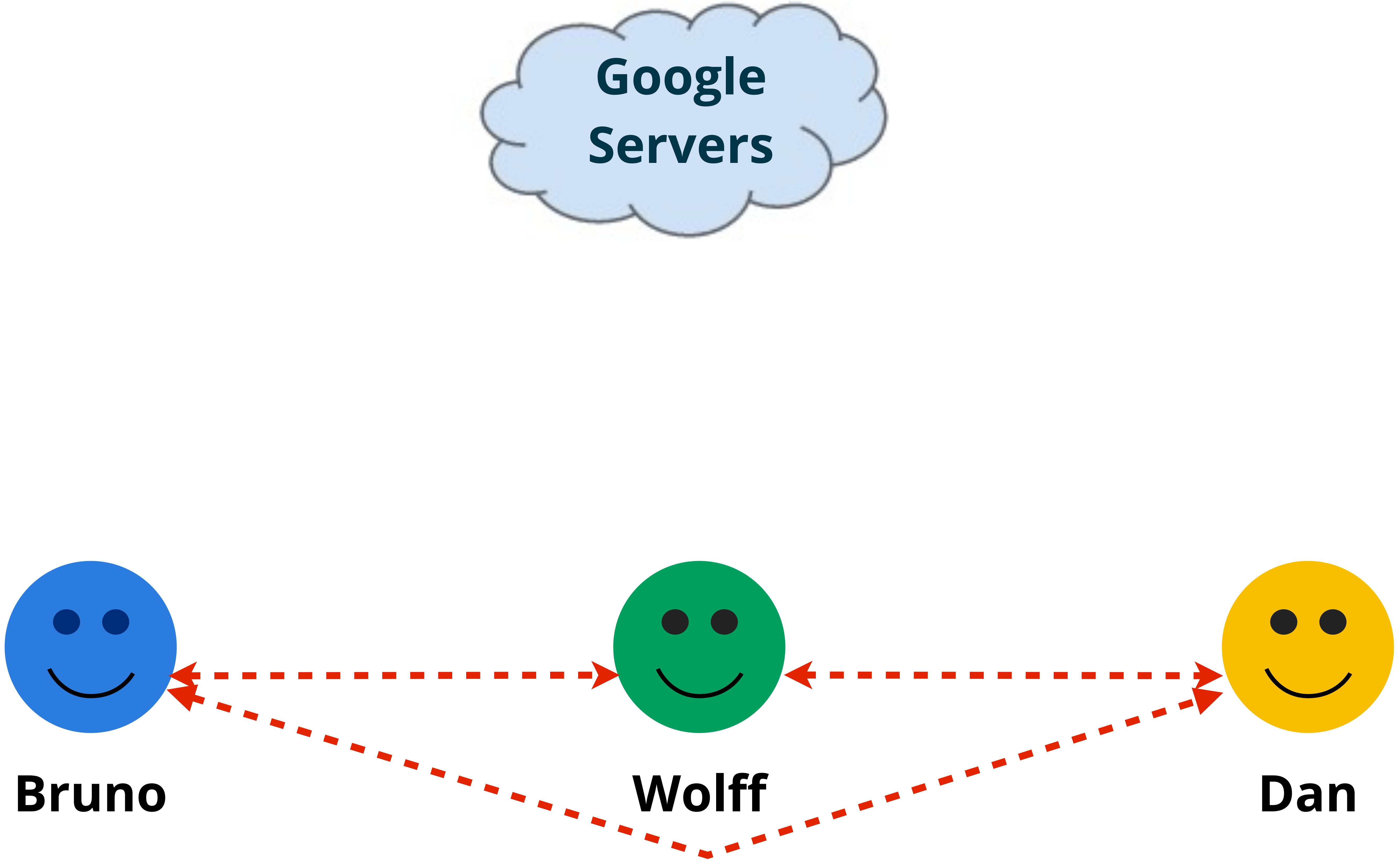

Google Servers
libjingle
Bruno
Wolff
Dan

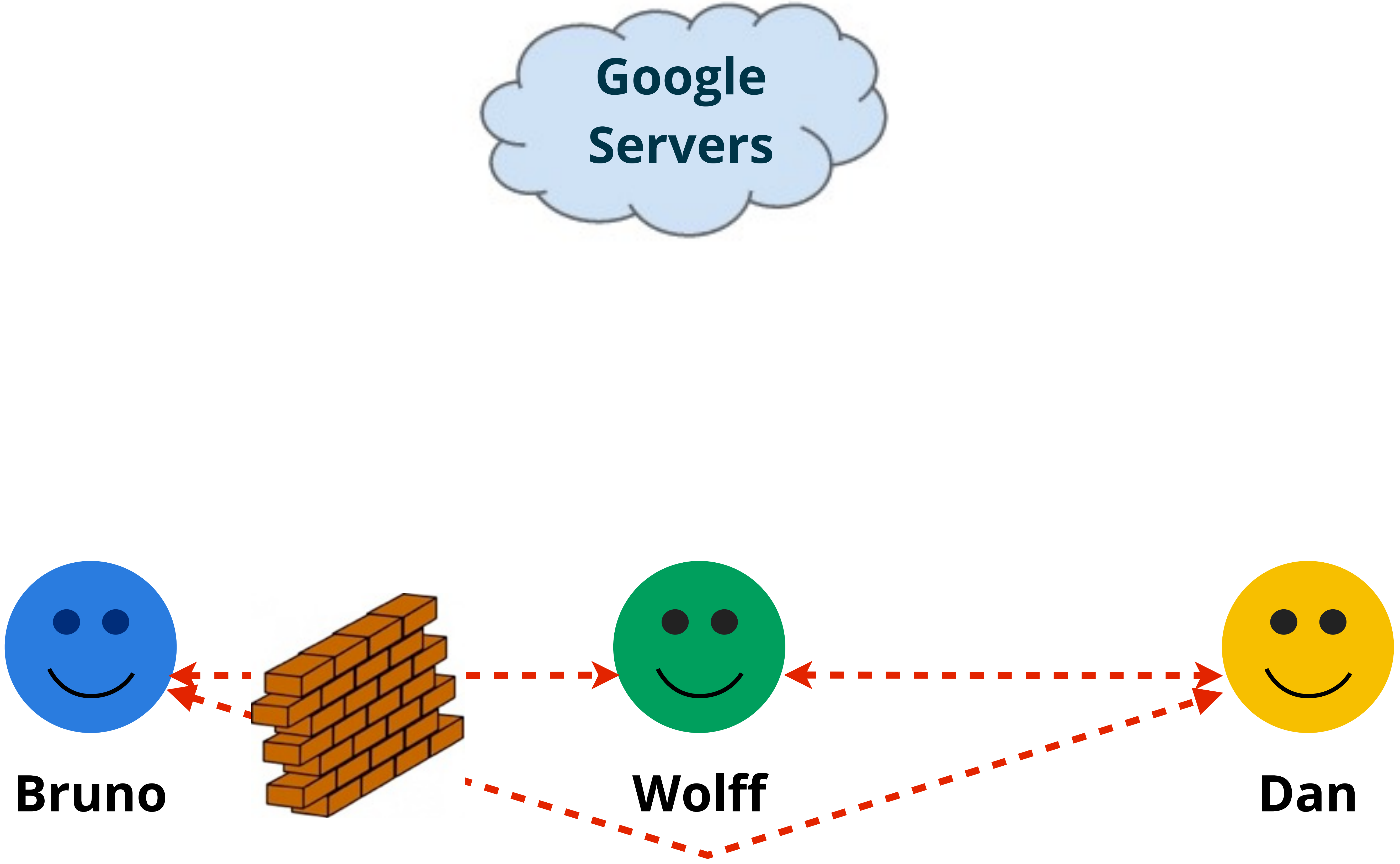

Google
Servers
RTP
Bruno
Wolff
Dan

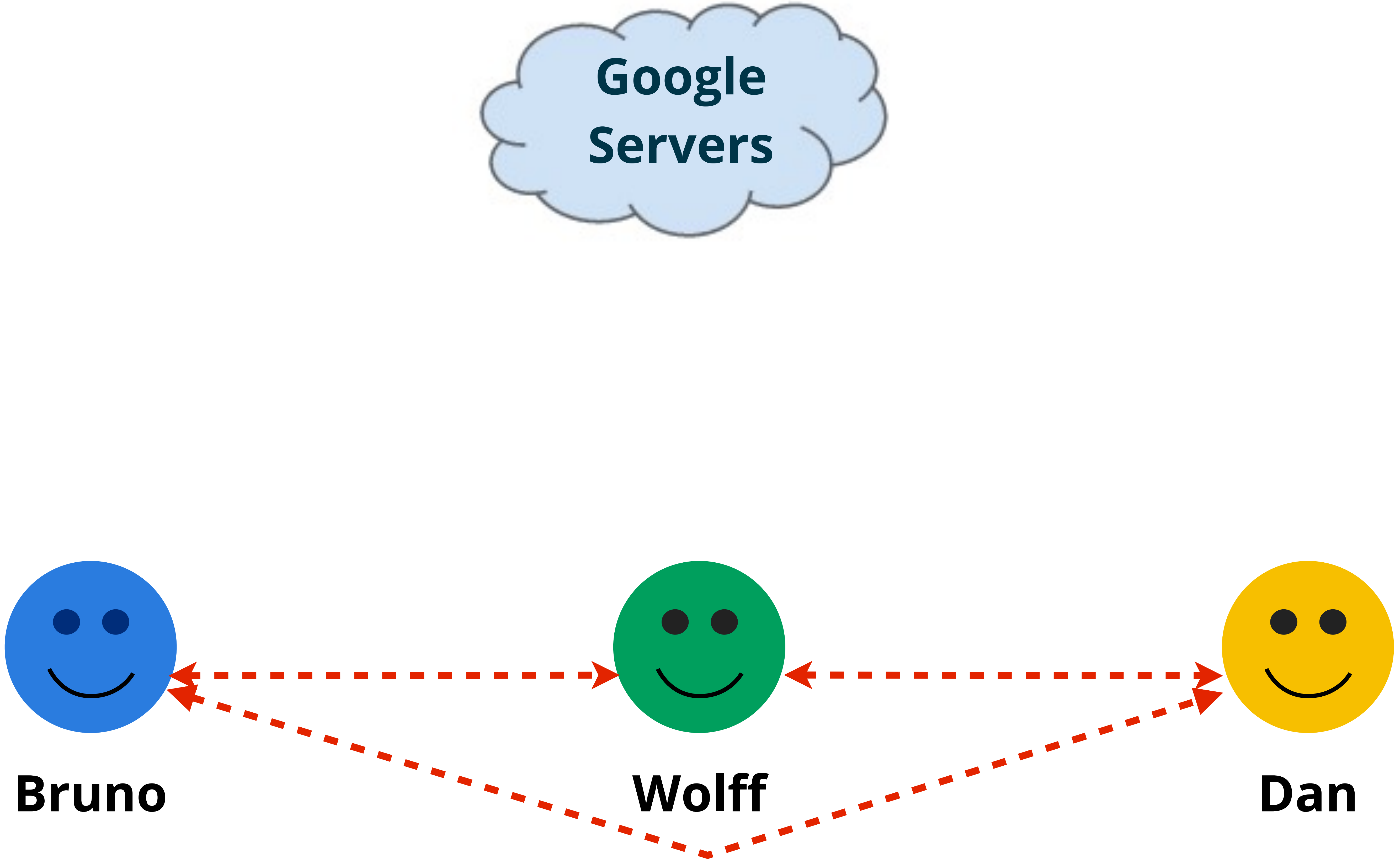

Google Servers
REST API
joinRoom()
Bruno
Wolff
Dan

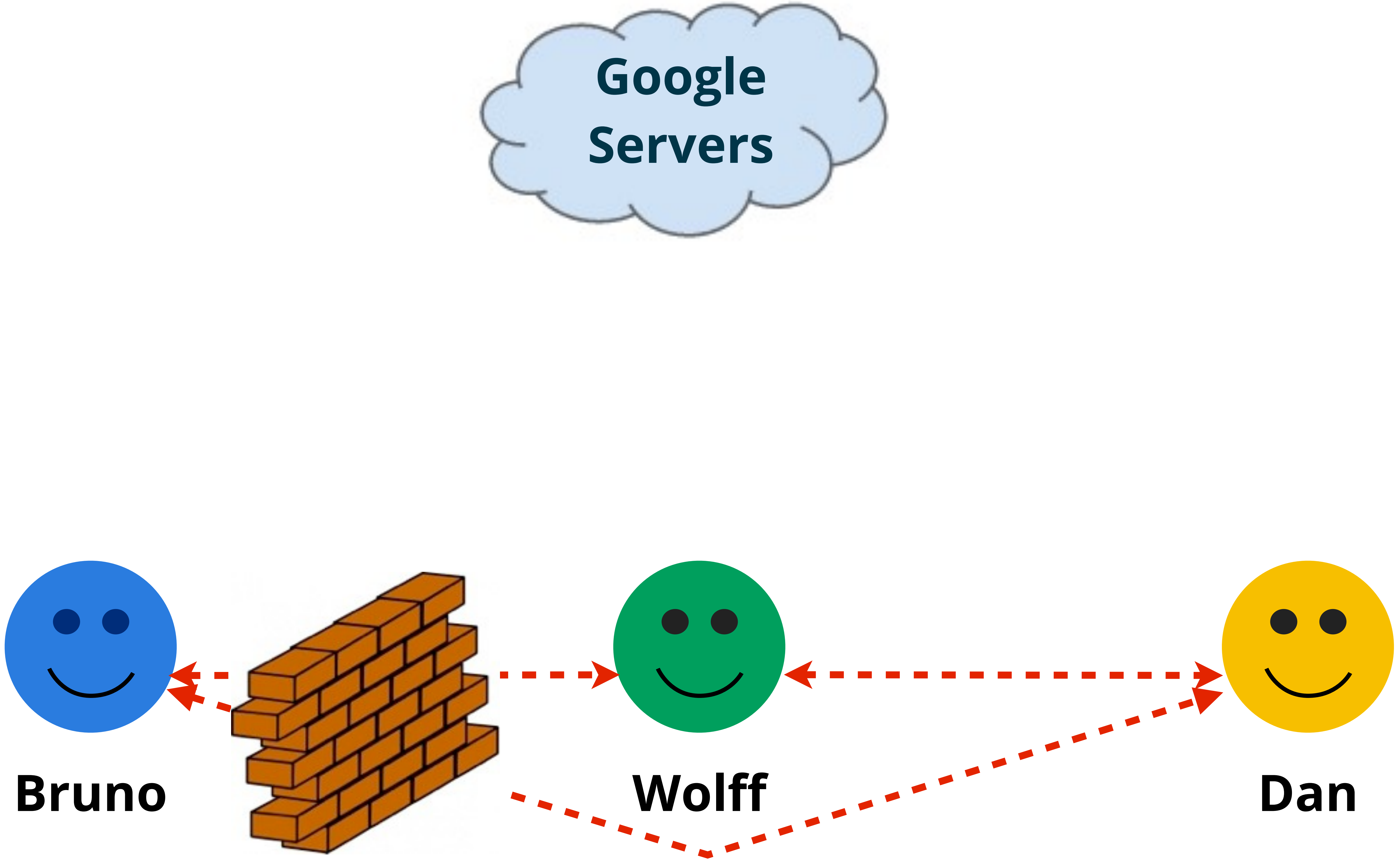

Dan joined!
Google Servers
XMPP
XMPP
Bruno
Wolff
Dan

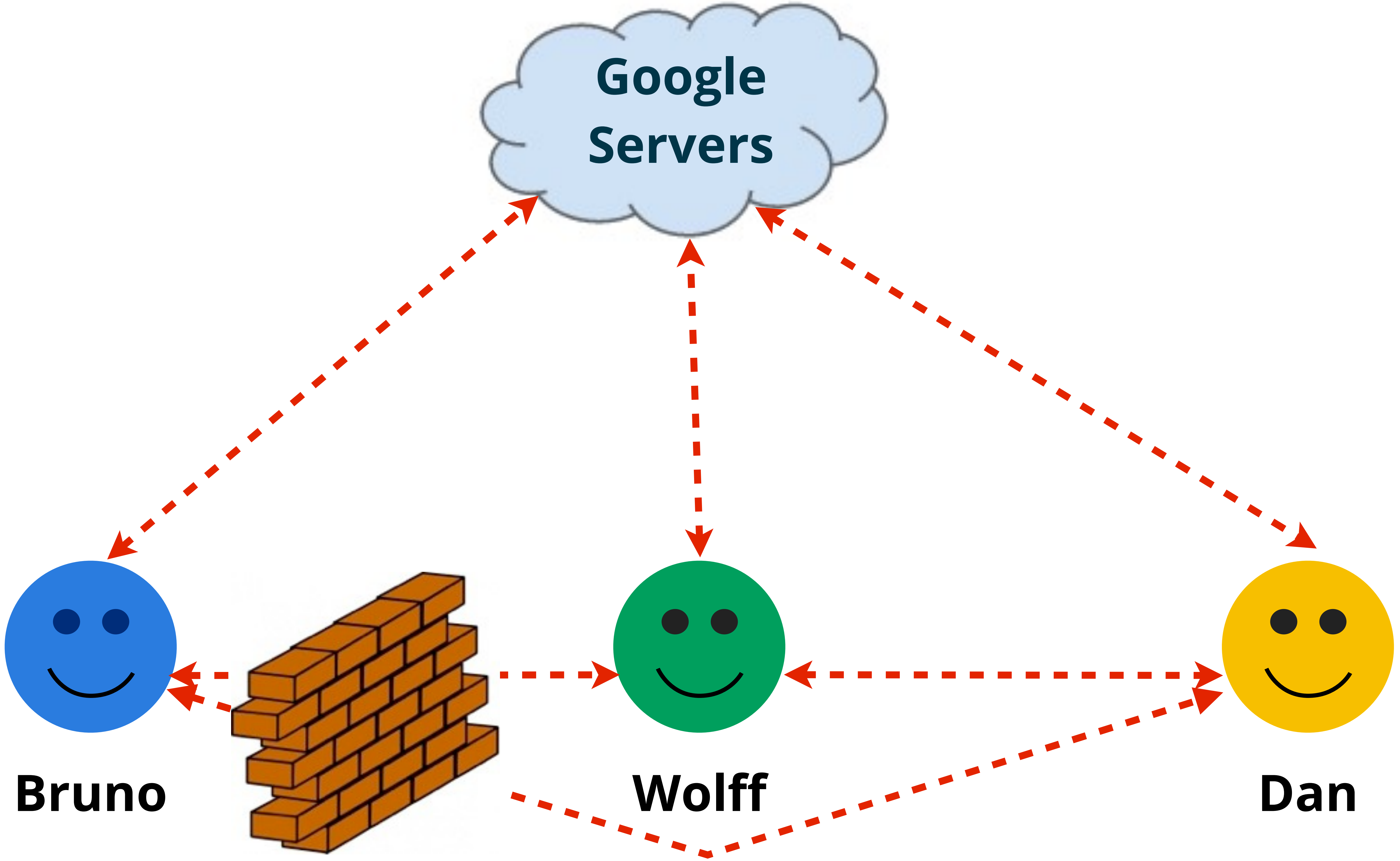

Google Servers
libjingle
Bruno
Wolff
Dan


Google
Servers
Bruno
Wolff
Dan

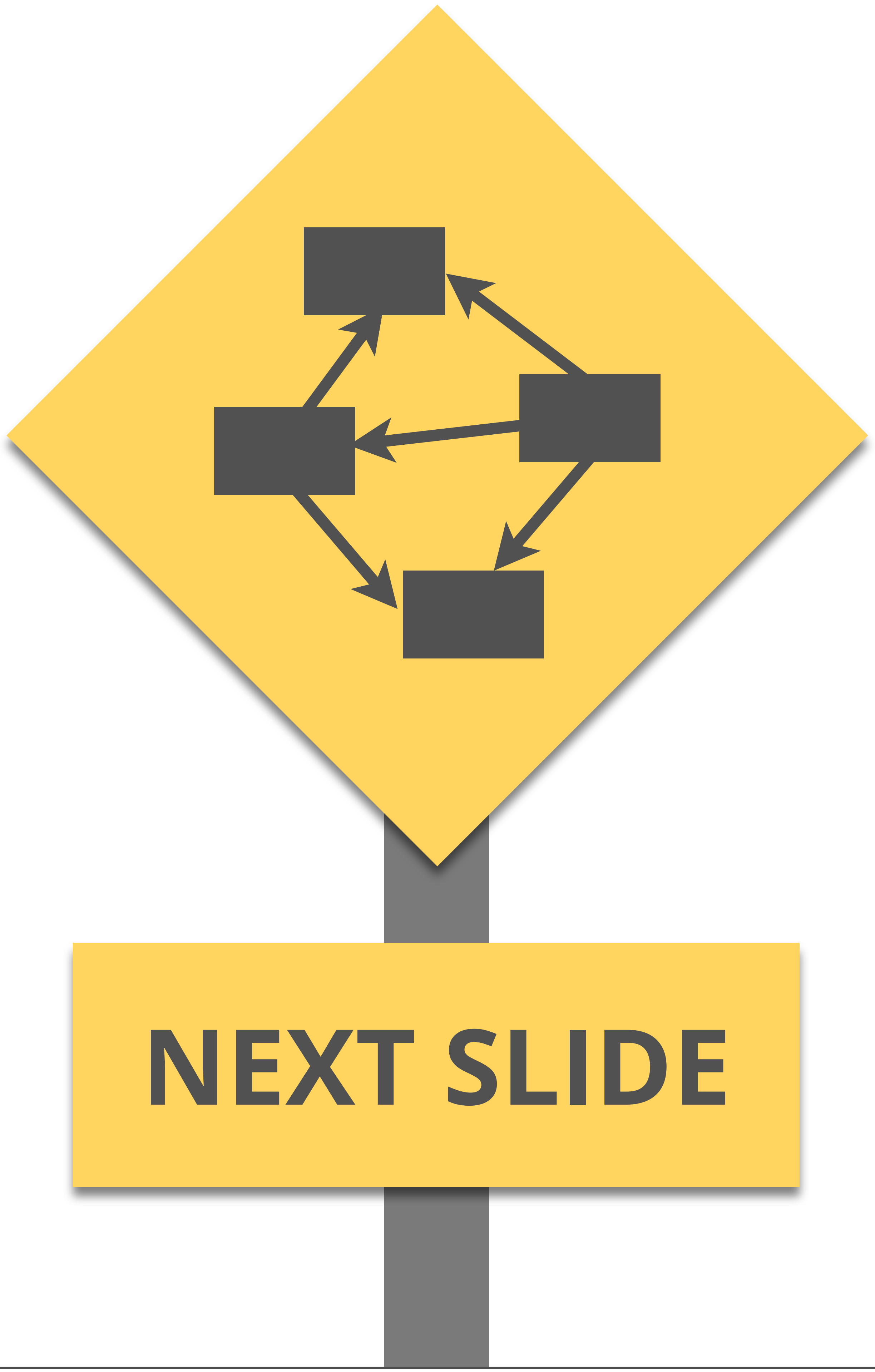

Google
Servers
Bruno
Wolff
Dan

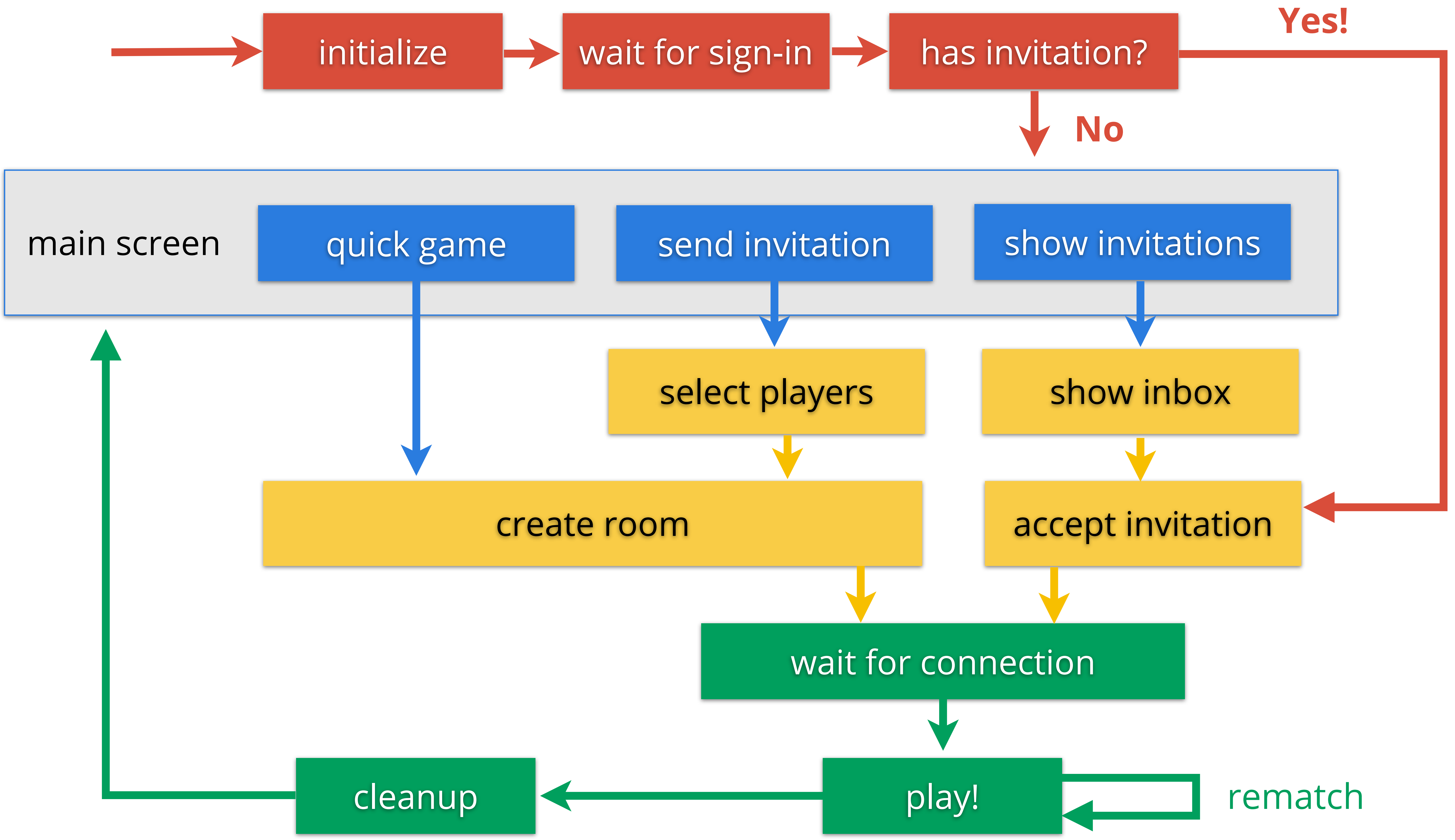

Google
Servers
Bruno
Wolff
Dan

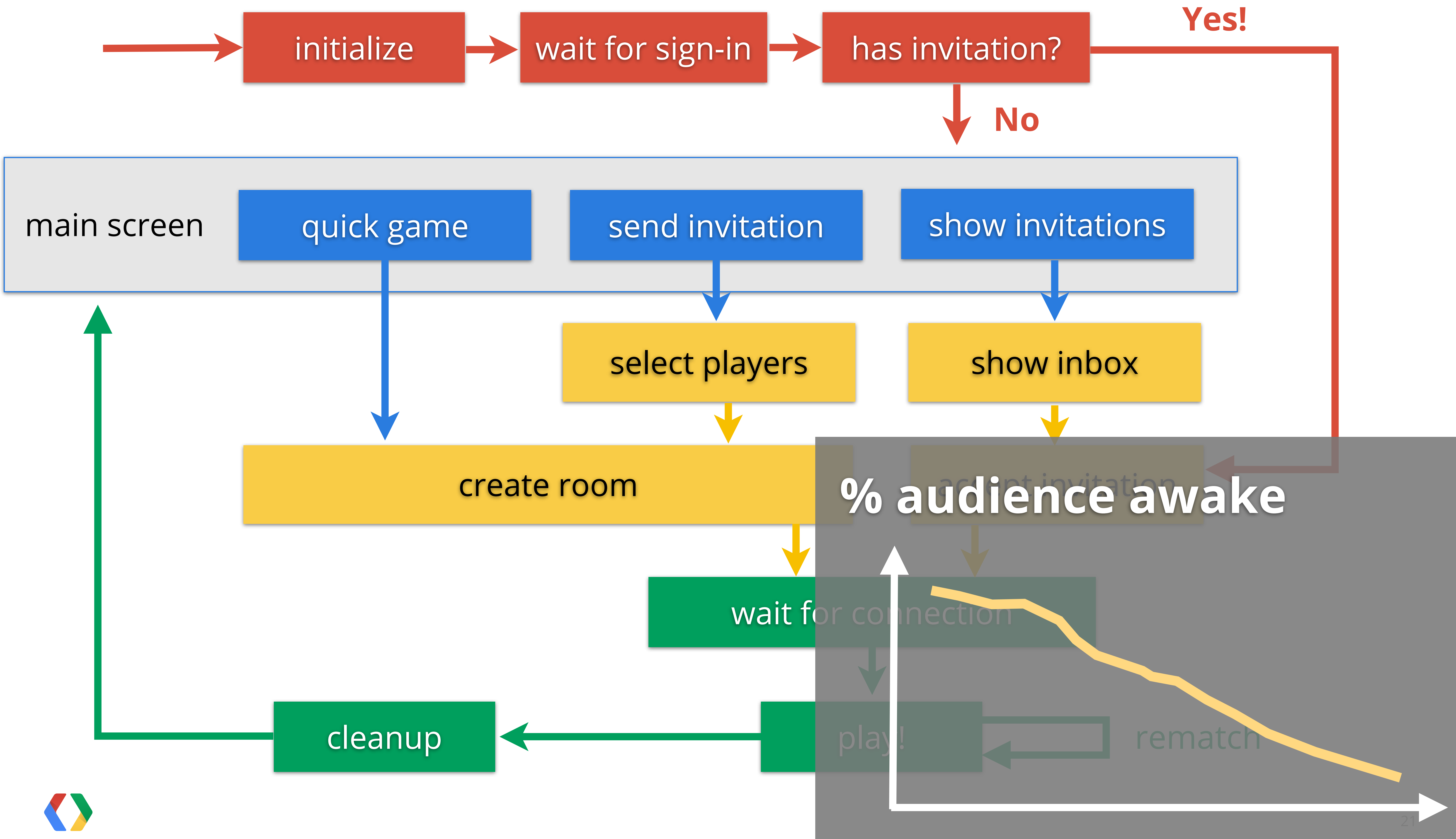

Google Servers
Bruno
Wolff
Dan

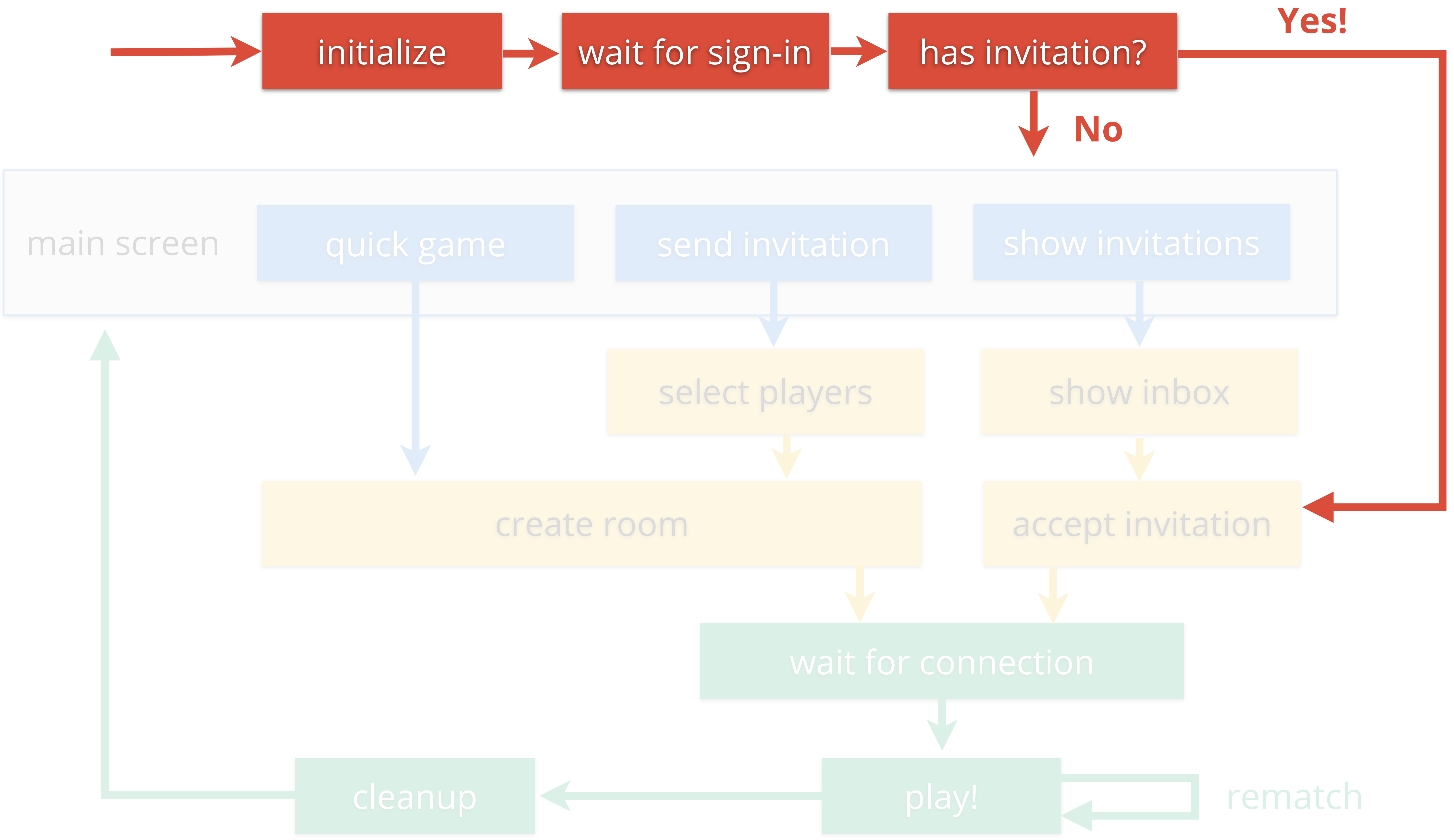

Google
Servers
Bruno
Wolff
Dan

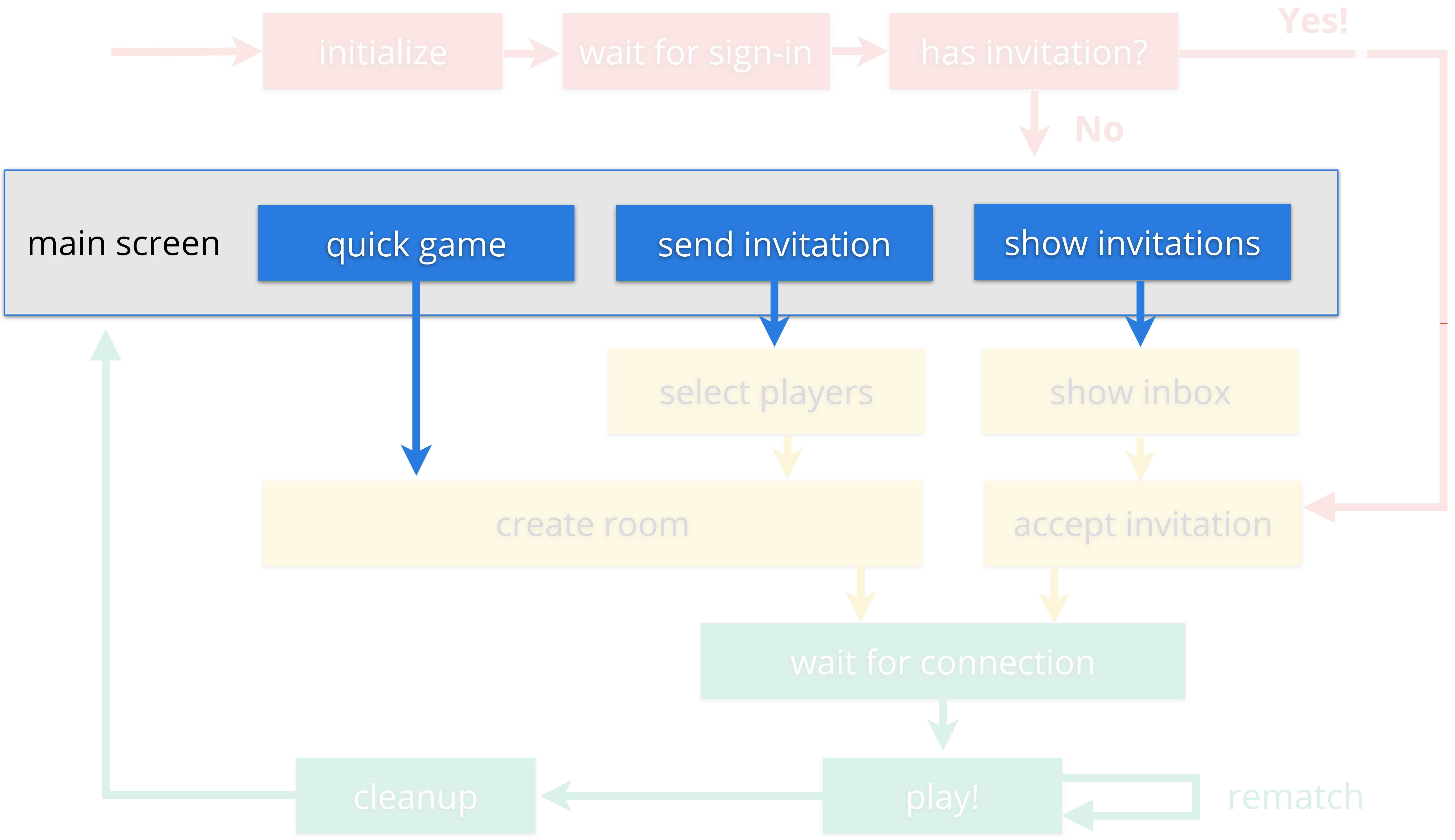
Sure but...
How do I do that?

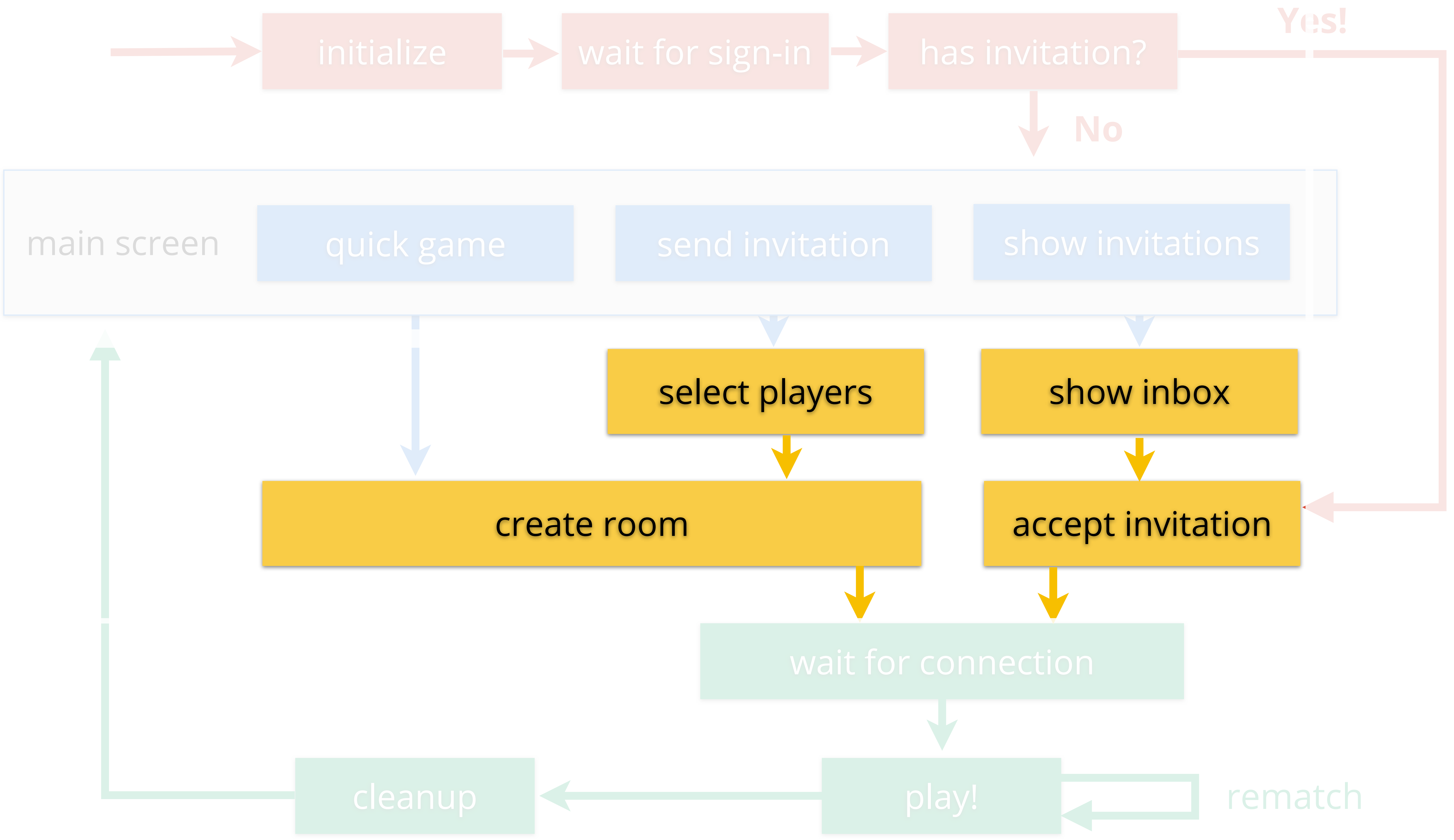

NEXT SLIDE

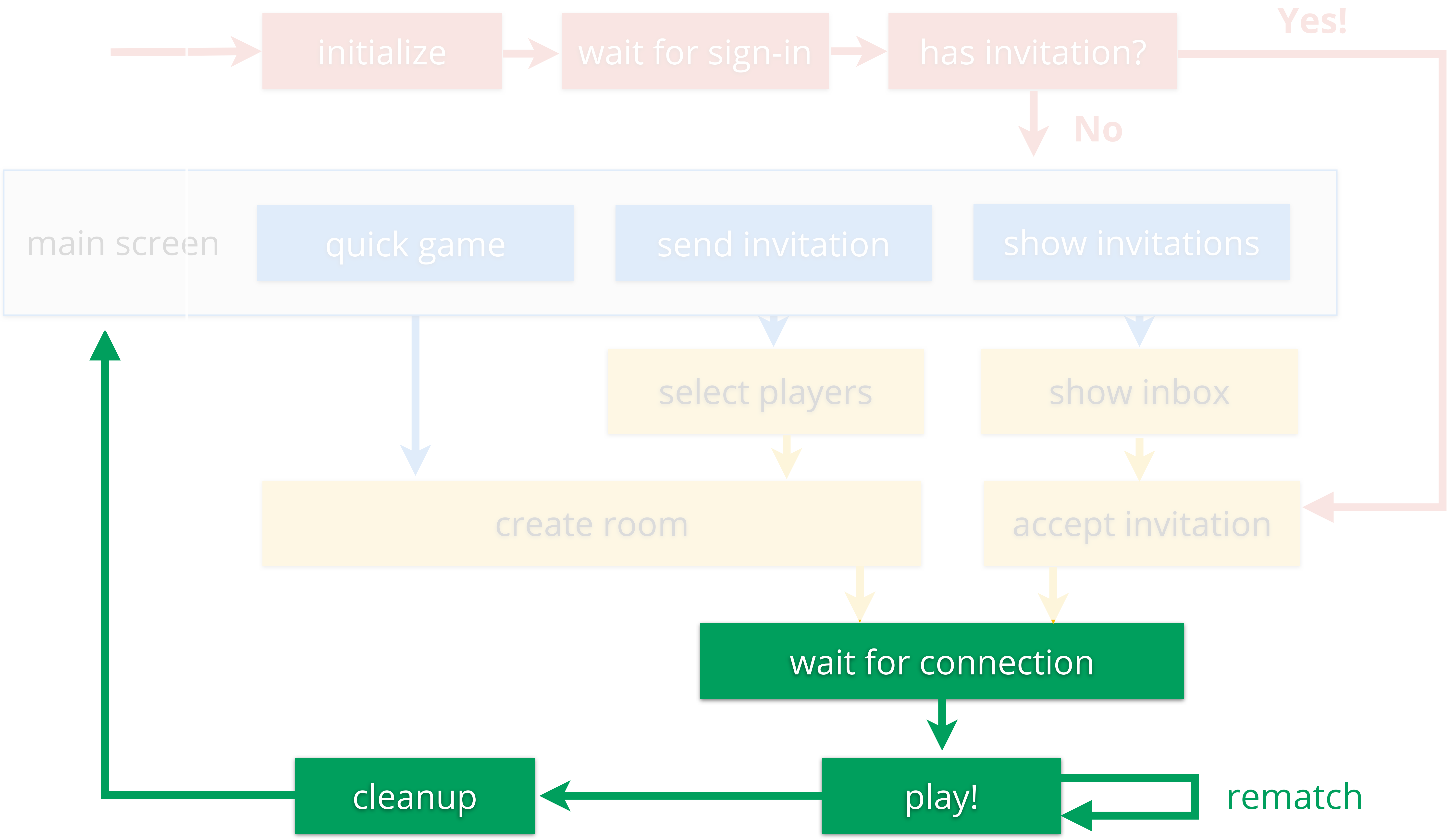

initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup
% audience awake


Yes!
initialize
wait for sign-in
has invitation?
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

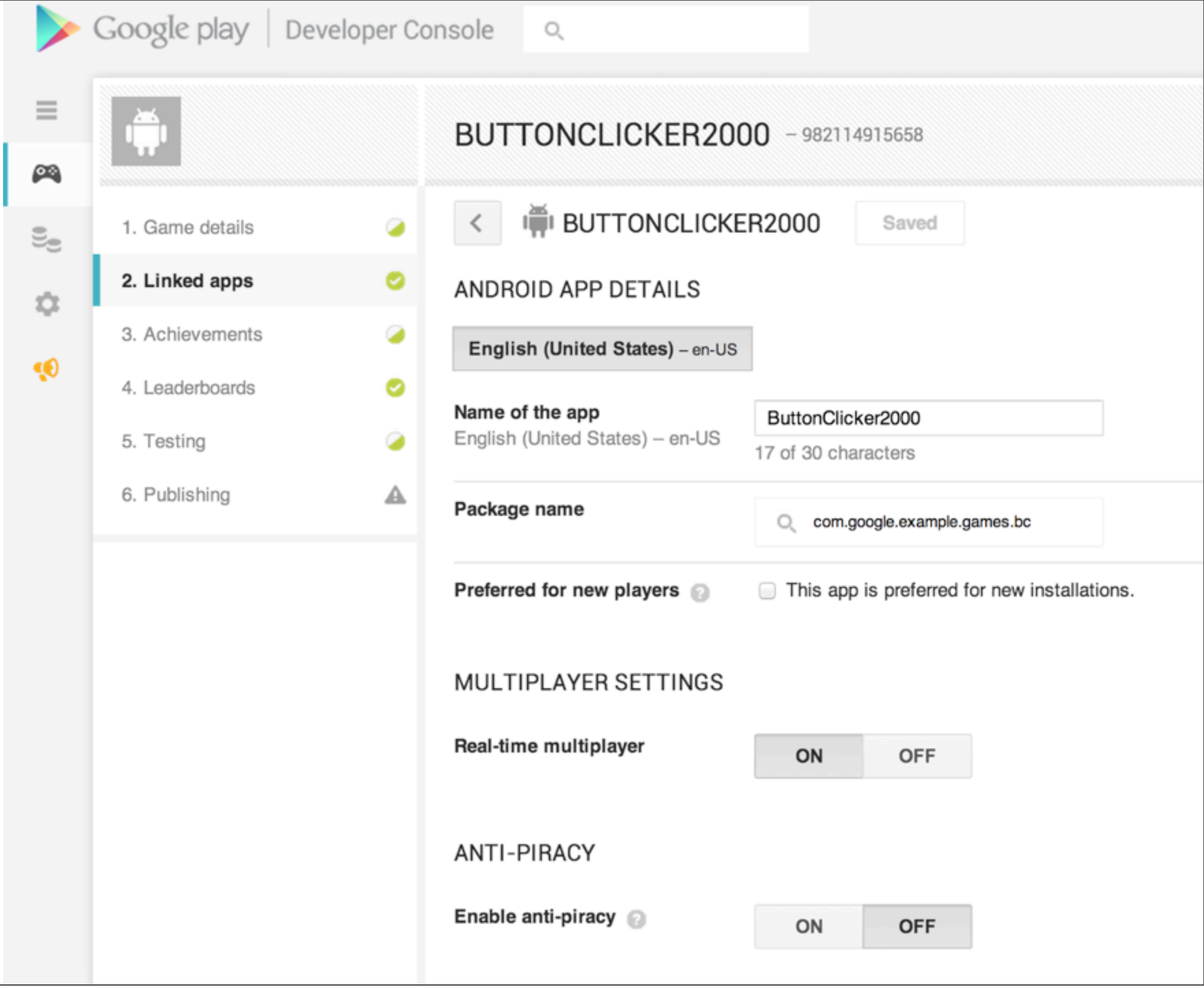

initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

Initialize and sign-in

Review: Implement Sign-in

Java

```java
public class MyActivity extends BaseGameActivity {
    @Override
    public void onSignInSucceeded() {
        // ready!
    }
    @Override
    public void onSignInFailed() {
        // show sign-in button
    }
    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.sign_in_button) {
            beginUserInitiatedSignIn();
        }
    }
}
```



BUTTONCLICKER2000 – 982114915658

BUTTONCLICKER2000

ANDROID APP DETAILS

English (United States) – en-US

Name of the app — English (United States) – en-US: ButtonClicker2000 (17 of 30 characters)

Package name: com.google.example.games.bc

Preferred for new players: ☐ This app is preferred for new installations.

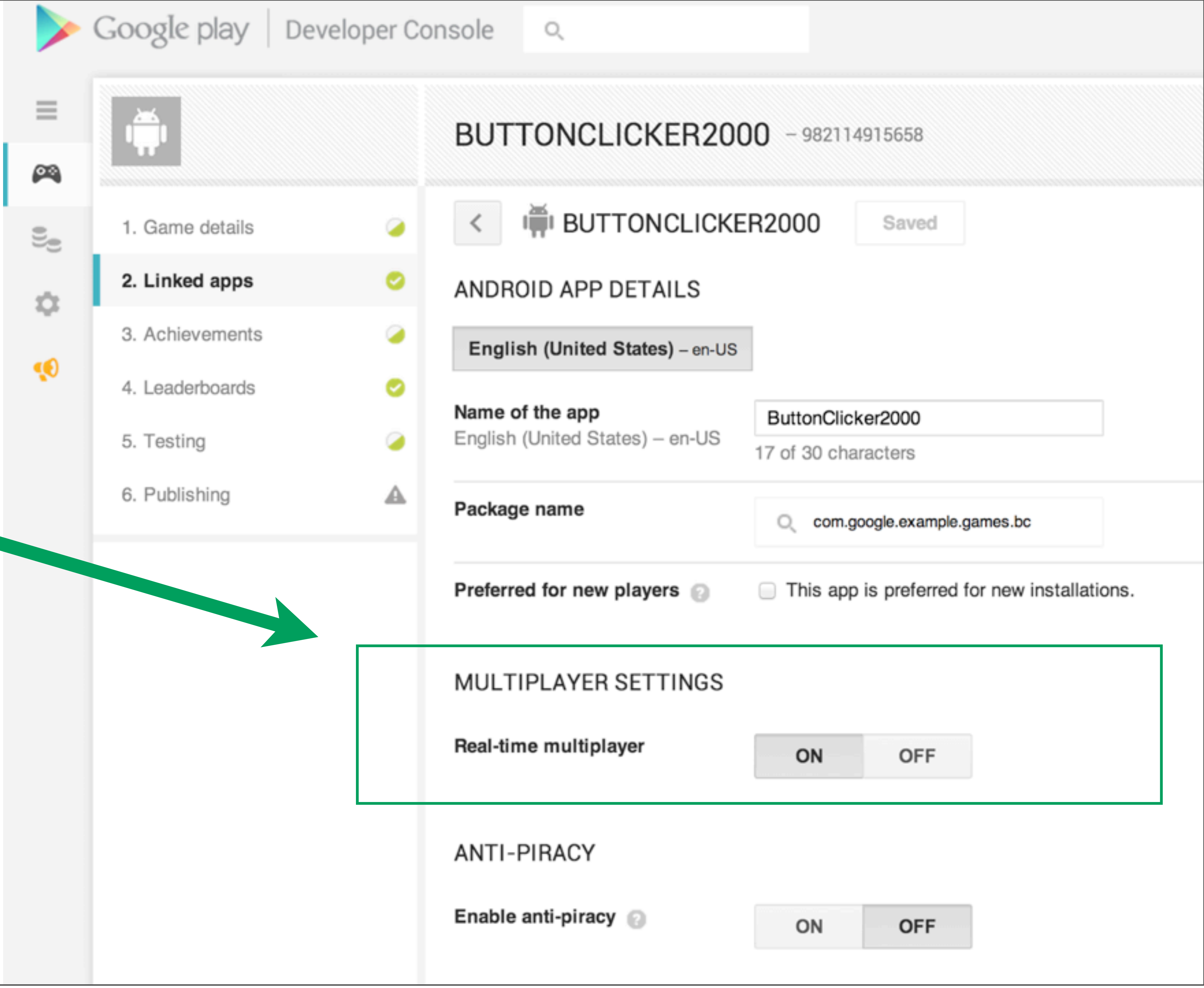
MULTIPLAYER SETTINGS

Real-time multiplayer: [✓ ON] OFF

ANTI-PIRACY

Enable anti-piracy: ON [✓ OFF]

Make sure
this is on



Google play | Developer Console

BUTTONCLICKER2000 – 982114915658

1. Game details
2. Linked apps
3. Achievements
4. Leaderboards
5. Testing
6. Publishing

BUTTONCLICKER2000 Saved

ANDROID APP DETAILS

English (United States) – en-US

Name of the app
English (United States) – en-US
ButtonClicker2000
17 of 30 characters

Package name: com.google.example.games.bc

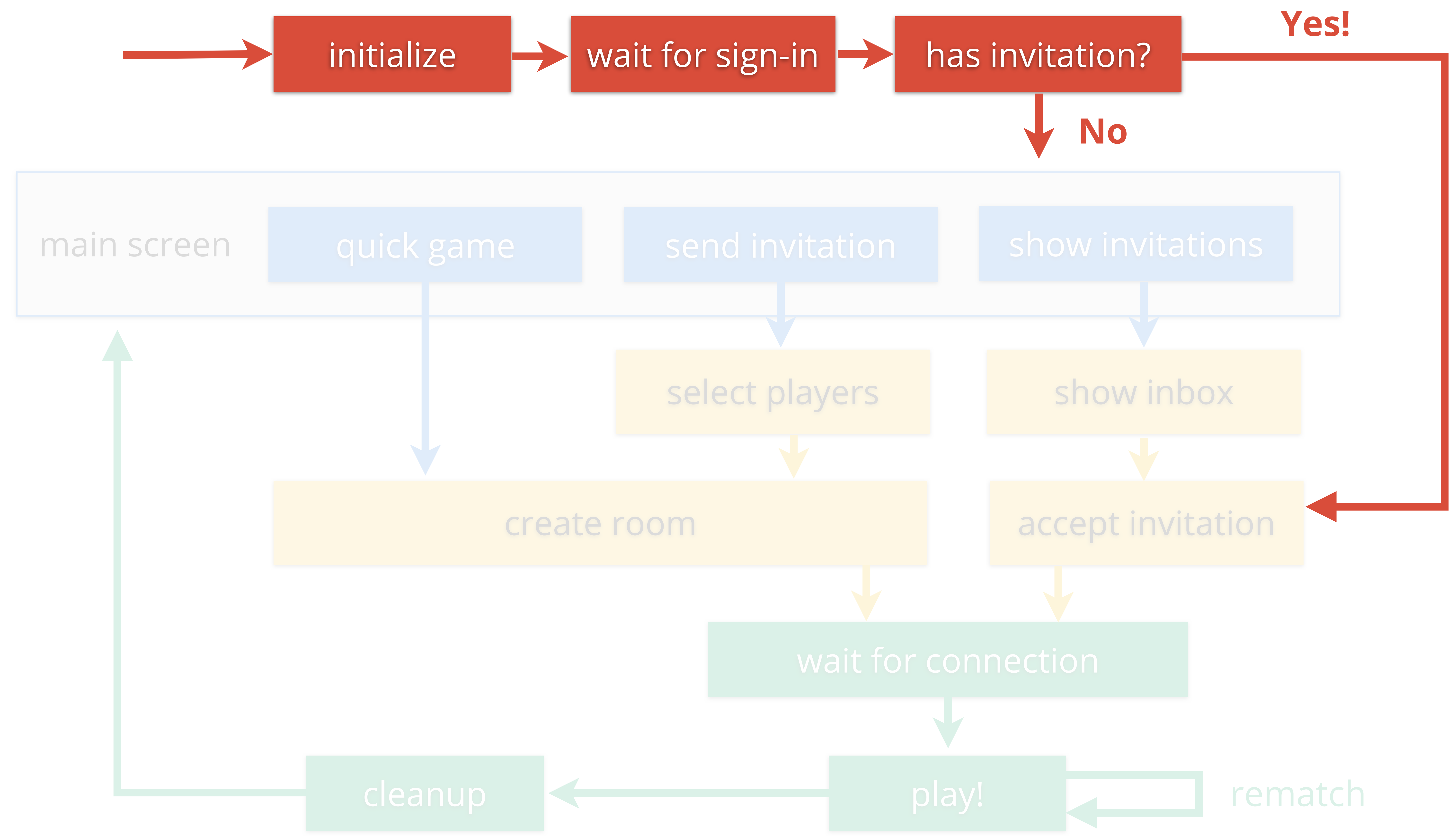
Preferred for new players: This app is preferred for new installations.

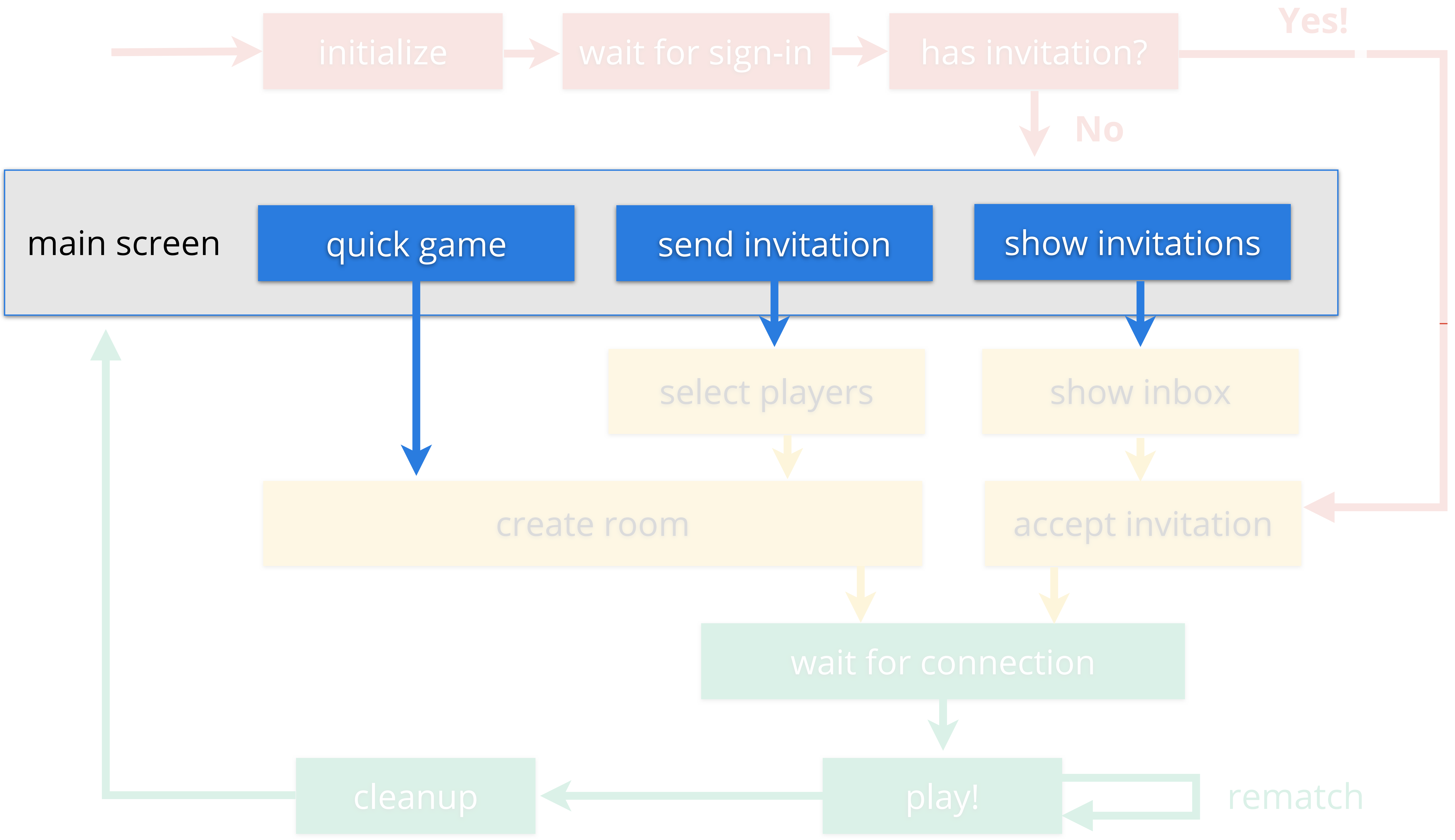
MULTIPLAYER SETTINGS

Real-time multiplayer: [✓ ON] OFF

ANTI-PIRACY

Enable anti-piracy: ON [✓ OFF]


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

Main Screen

Main Screen

quick game

random opponent

Main Screen

quick game	random opponent
send invitation	player can invite friends

Main Screen

quick game	random opponent
send invitation	player can invite friends
show invitations	show “invitation inbox”

Main Screen

quick game

send invitation

show invitations

Main Screen

quick game

send invitation

show invitations

create a room +
invite random player

Main Screen

quick game

create a room +
invite random player

send invitation

create a room +
invite specific players

show invitations

Main Screen

quick game

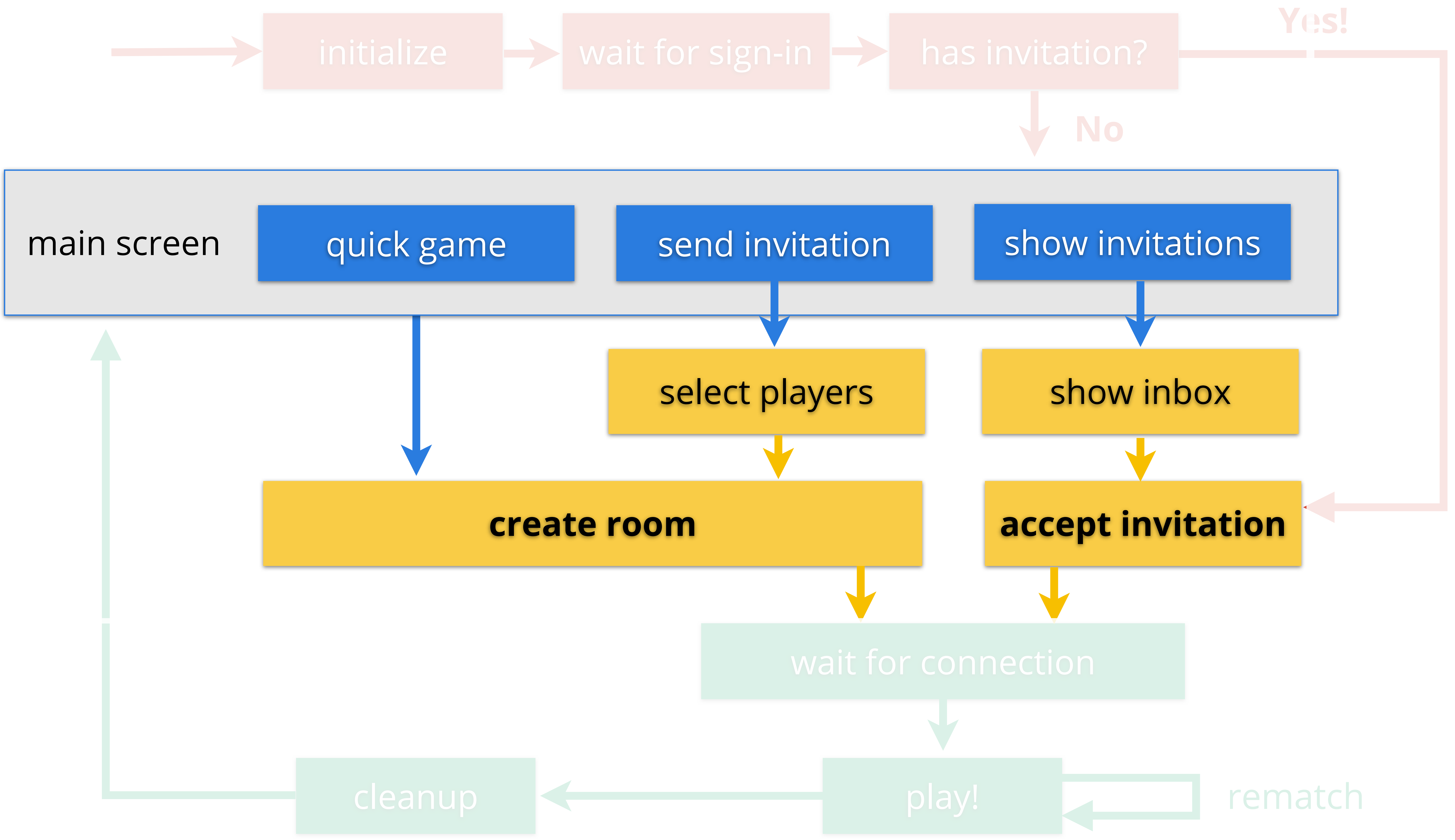
create a room +
invite random player

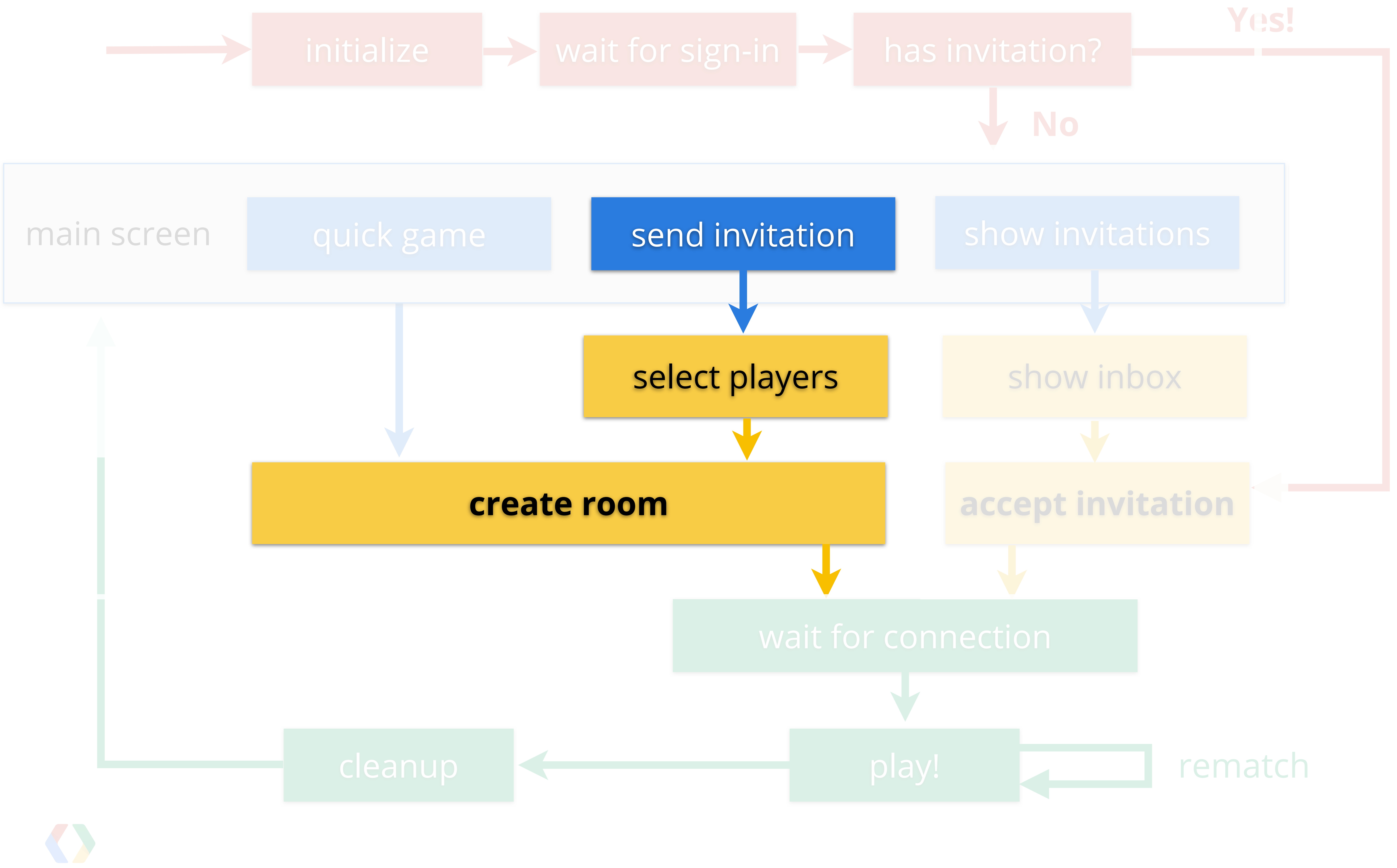
send invitation

create a room +
invite specific players

show invitations

show me what rooms I'm
being invited into.


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch

To create a room:

1. Who to invite?
2. Automatch settings
3. Callbacks

To create a **room**:

RoomConfig

1. Who to invite?
2. Automatch settings
3. Callbacks

Create the room config

Java

```java
// create room config builder
RoomConfig.Builder configBuilder =
          RoomConfig.builder(this);
```

Show the "Select Players" dialog

Java

```java
// request code for the "select players" UI
final static int RC_SEL_PLAYERS = 10000;

// launch the player selection screen
final static int MIN_OPPONENTS = 1;
final static int MAX_OPPONENTS = 1;
```

Show the “Select Players” dialog

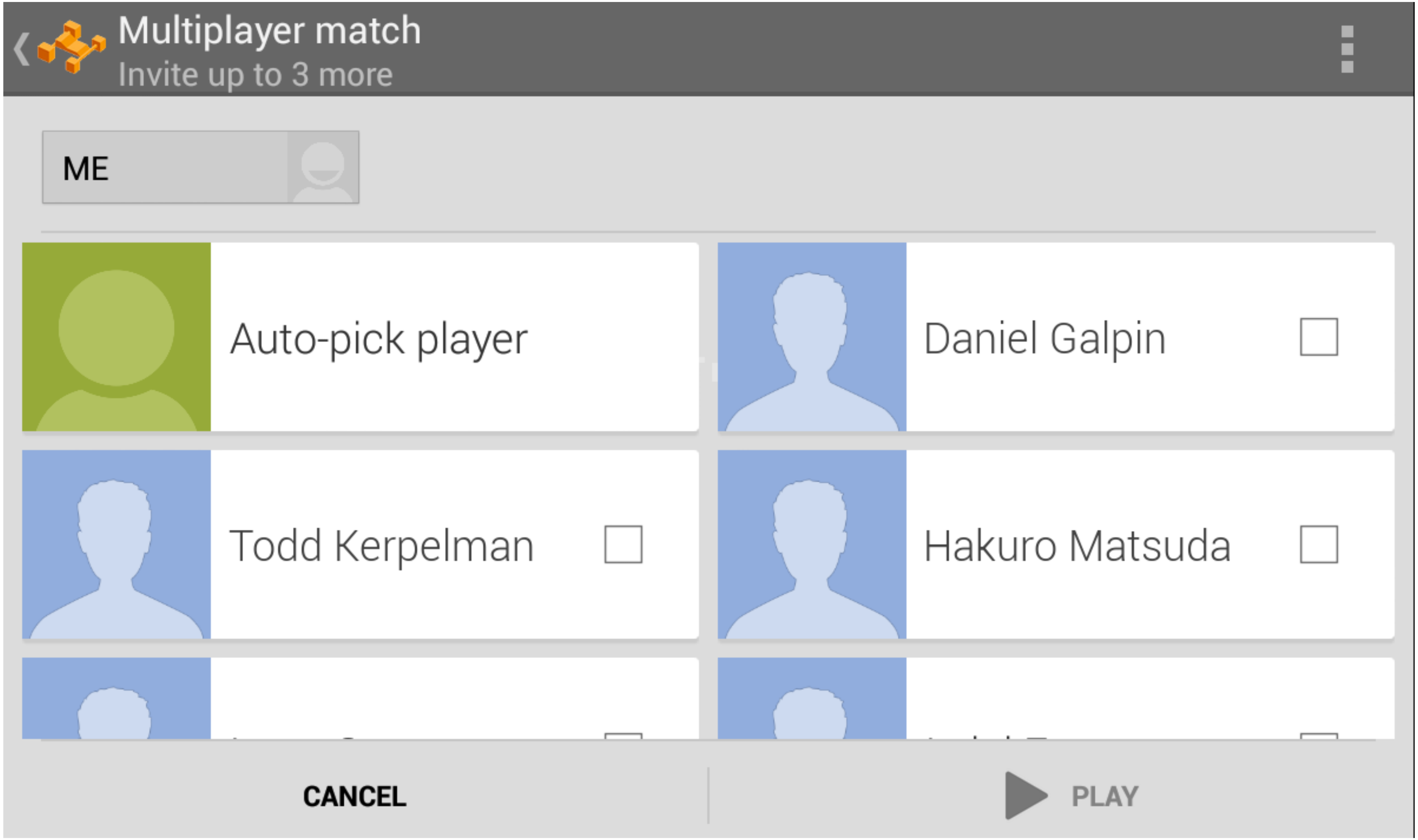
Java

```
// request code for the "select players" UI
final static int RC_SEL_PLAYERS = 10000;

// launch the player selection screen
final static int MIN_OPPONENTS = 1;
final static int MAX_OPPONENTS = 1;

Intent intent = getGamesClient().getSelectPlayersIntent(
                    MIN_OPPONENTS, MAX_OPPONENTS);
```

Show the “Select Players” dialog

Java

```
// request code for the "select players" UI
final static int RC_SEL_PLAYERS = 10000;

// launch the player selection screen
final static int MIN_OPPONENTS = 1;
final static int MAX_OPPONENTS = 1;

Intent intent = getGamesClient().getSelectPlayersIntent(
                    MIN_OPPONENTS, MAX_OPPONENTS);

startActivityForResult(intent, RC_SEL_PLAYERS);
```


Multiplayer match
Invite up to 3 more
ME
Auto-pick player
Daniel Galpin
Todd Kerpelman
Hakuro Matsuda
CANCEL
PLAY

Handle the result of the “Select Players” dialog

Java

```java
@Override
public void onActivityResult(int request, int response,
    Intent data) {
```

Handle the result of the “Select Players” dialog

Java

```java
@Override
public void onActivityResult(int request, int response,
    Intent data) {

    if (request == RC_SELECT_PLAYERS) {
        if (response != Activity.RESULT_OK) {
            // user cancelled
            return;
        }

        // (continued)
```

Handle the result of the "Select Players" dialog

Java

```
// get the list of selected players

Bundle extras = data.getExtras();
```

Handle the result of the “Select Players” dialog

Java

```java
// get the list of selected players

Bundle extras = data.getExtras();

ArrayList<String> invitees;
invitees = data.getStringArrayListExtra(
     GamesClient.EXTRA_PLAYERS);
```

Add that to the room config

Java

```java
// set players to invite
configBuilder.addPlayersToInvite(invitees);
```

Add that to the room config

~~1. Who to invite?~~

2. Automatch settings

3. Callbacks

```
// set players to invite
configBuilder.addPlayersToInvite(invitees);
```

Get automatch settings

Java

```java
// get the automatch settings
int amMin = data.getIntExtra(
        GamesClient.EXTRA_MIN_AUTOMATCH_PLAYERS, 0);
int amMax = data.getIntExtra(
        GamesClient.EXTRA_MAX_AUTOMATCH_PLAYERS, 0);
```

Add that to the room config

Java

```java
// add automatch settings to the room config
Bundle amCrit = null;
if (amMin > 0) {
    amCrit = RoomConfig.createAutoMatchCriteria(
                      amMin, amMax, 0);
    configBuilder.setAutoMatchCriteria(amCrit);
}
```

Add that to the room config

~~1. Who to invite?~~

~~2. Automatch settings~~

3. Callbacks

```
// add automatch settings to the
Bundle amCrit = null;
if (amMin > 0) {
    amCrit = RoomConfig.createAutoMatchCriteria(
                           amMin, amMax, 0);
    configBuilder.setAutoMatchCriteria(amCrit);
}
```

Set up callbacks and create room!

Java

```
// set up callbacks (listeners)
configBuilder.setMessageReceivedListener(this)
configBuilder.setRoomStatusUpdateListener(this);
```

Set up callbacks and create roo

~~1. Who to invite?~~

~~2. Automatch settings~~

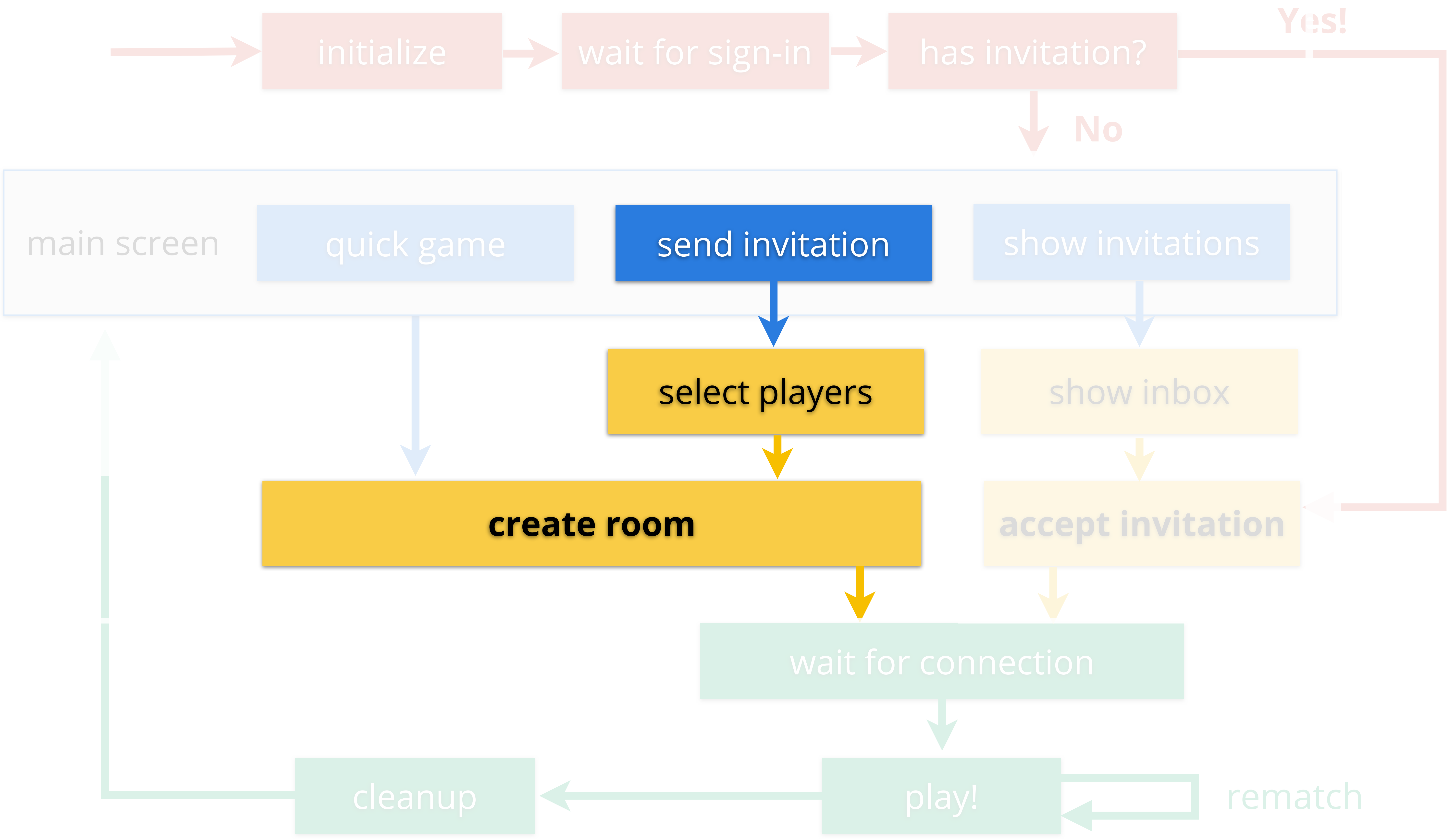
~~3. Callbacks~~

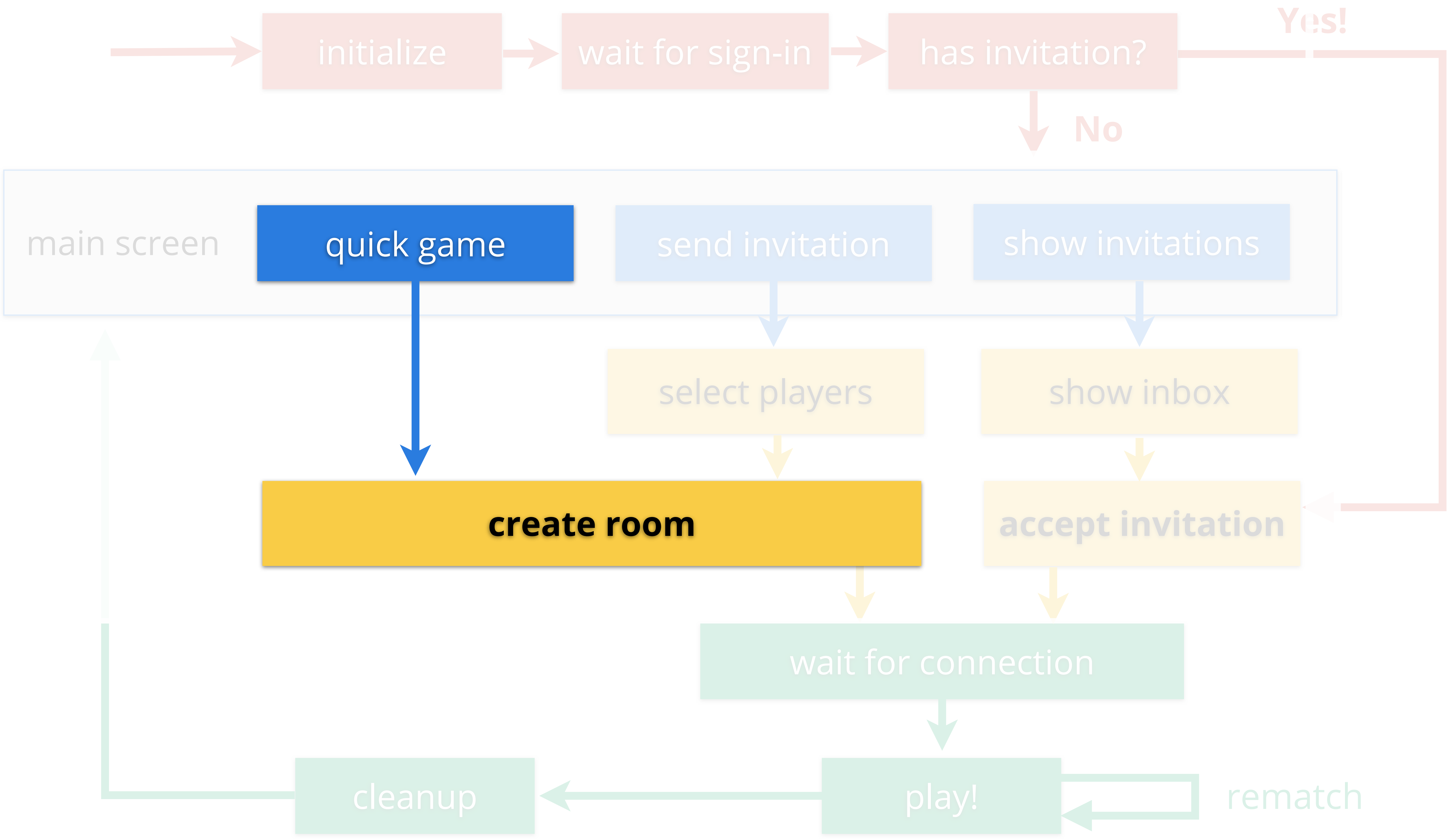
```
// set up callbacks (listeners)
configBuilder.setMessageReceivedListener(this)
configBuilder.setRoomStatusUpdateListener(this);
```

Set up callbacks and create roo

~~1. Who to invite?~~
~~2. Automatch settings~~
~~3. Callbacks~~

```
// set up callbacks (listeners)
configBuilder.setMessageReceivedListener(this)
configBuilder.setRoomStatusUpdateListener(this);

// create room!
getGamesClient().createRoom(configBuilder.build());
```


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

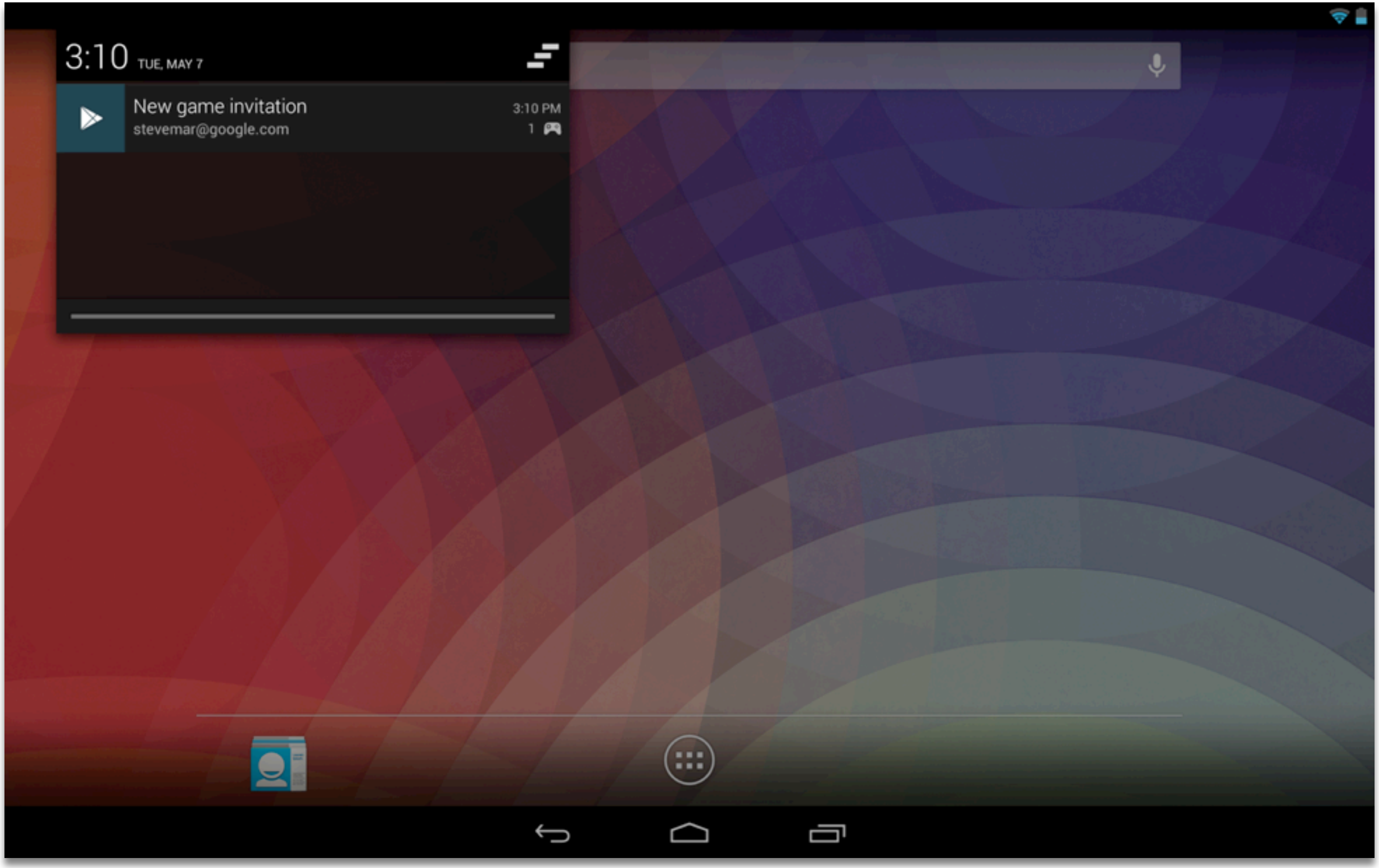

initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch

Quick game with a random opponent

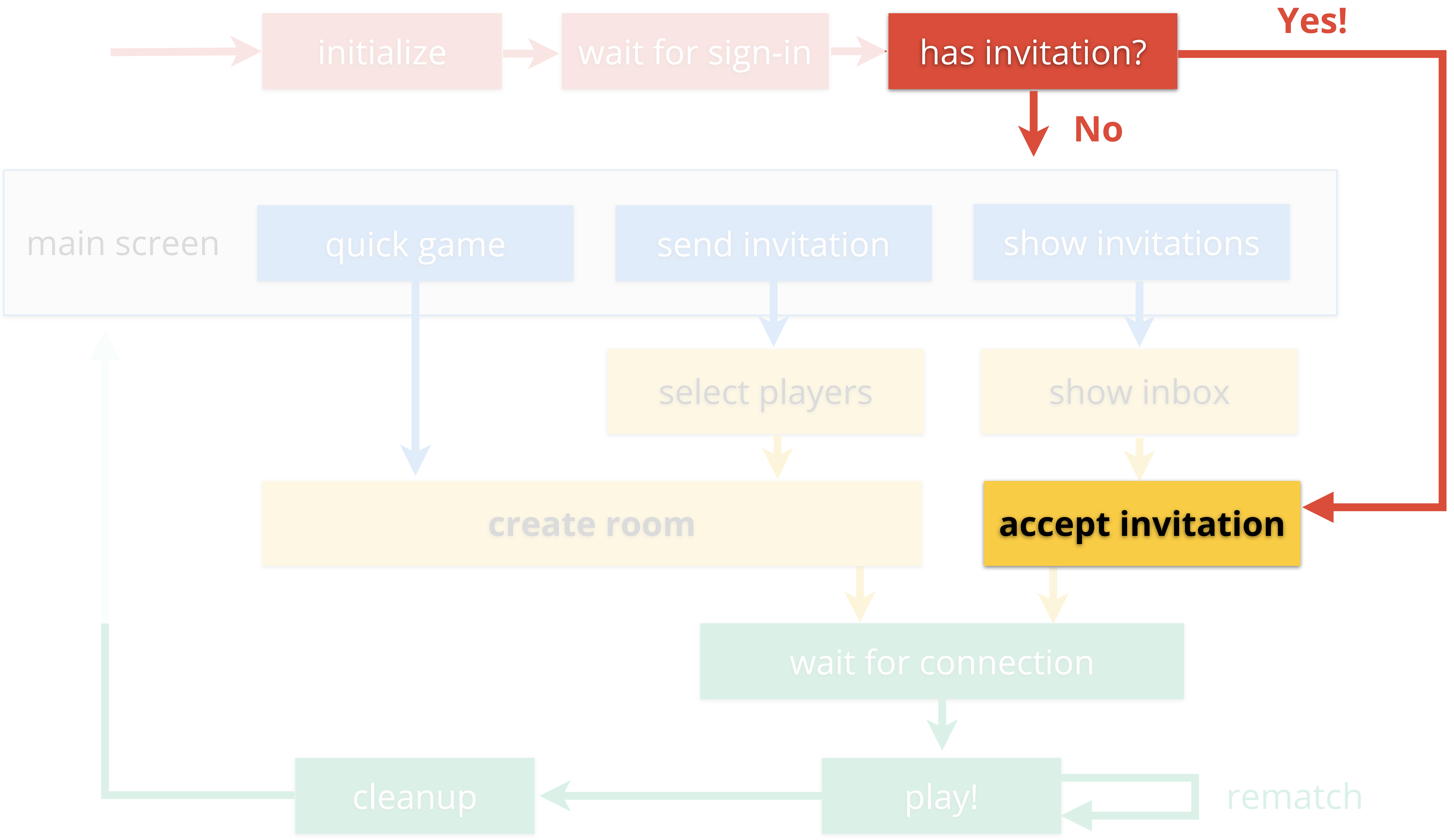
Java

```java
// automatch criteria to invite 1 random automatch opponent:
Bundle am = RoomConfig.createAutoMatchCriteria(1, 1, 0);

// build the room config
RoomConfig.Builder configBuilder = RoomConfig.builder(this)
     .setMessageReceivedListener(this)
     .setRoomStatusUpdateListener(this);
     .setAutoMatchCriteria(am);

// create room
getGamesClient().createRoom(configBuilder.build());
```

Receiving an invitation



3:10 TUE, MAY 7
New game invitation
stevemar@google.com
3:10 PM
1


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch

Check for an invitation

Java

```
public void onConnected(Bundle connectionHint) {
```

Check for an invitation

Java

```java
public void onConnected(Bundle connectionHint) {
   if (connectionHint != null) {
       Invitation inv = connectionHint.getParcelable(
                            GamesClient.EXTRA_INVITATION);

       if (inv != null && inv.getInvitationId() != null) {
           // accept invitation
           // (continued)
       }
```

Accept the invitation

Java

```java
// room config (set callbacks and invitation ID)
RoomConfig.Builder configBuilder =
    RoomConfig.builder(this)
```

Accept the invitation

Java

```java
// room config (set callbacks and invitation ID)
RoomConfig.Builder configBuilder =
    RoomConfig.builder(this)
      .setMessageReceivedListener(this)
      .setRoomStatusUpdateListener(this)
```

Accept the invitation

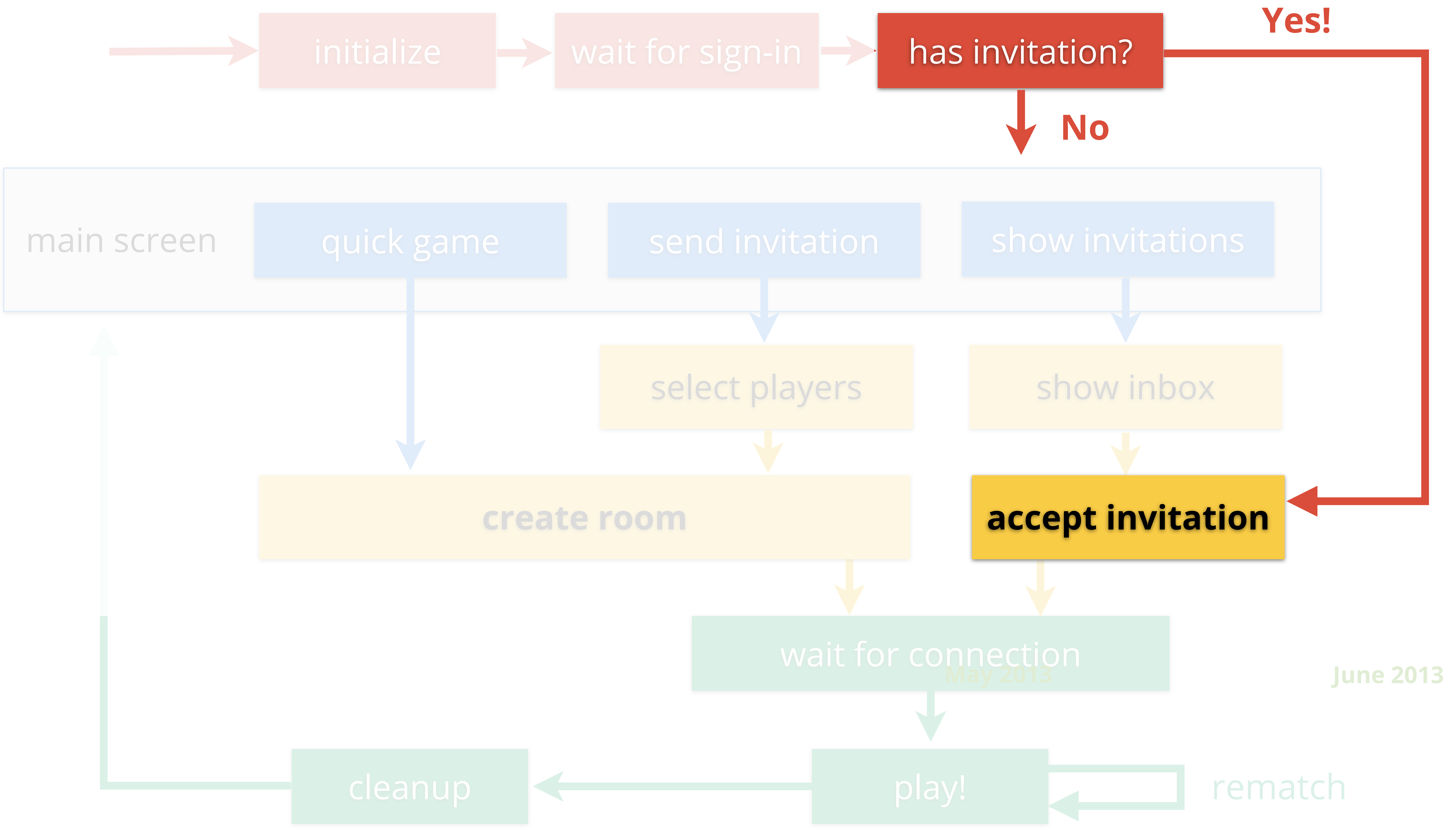
Java

```
// room config (set callbacks and invitation ID)
RoomConfig.Builder configBuilder =
    RoomConfig.builder(this)
      .setMessageReceivedListener(this)
      .setRoomStatusUpdateListener(this)
      .setInvitationToAccept(inv.getInvitationId())
```

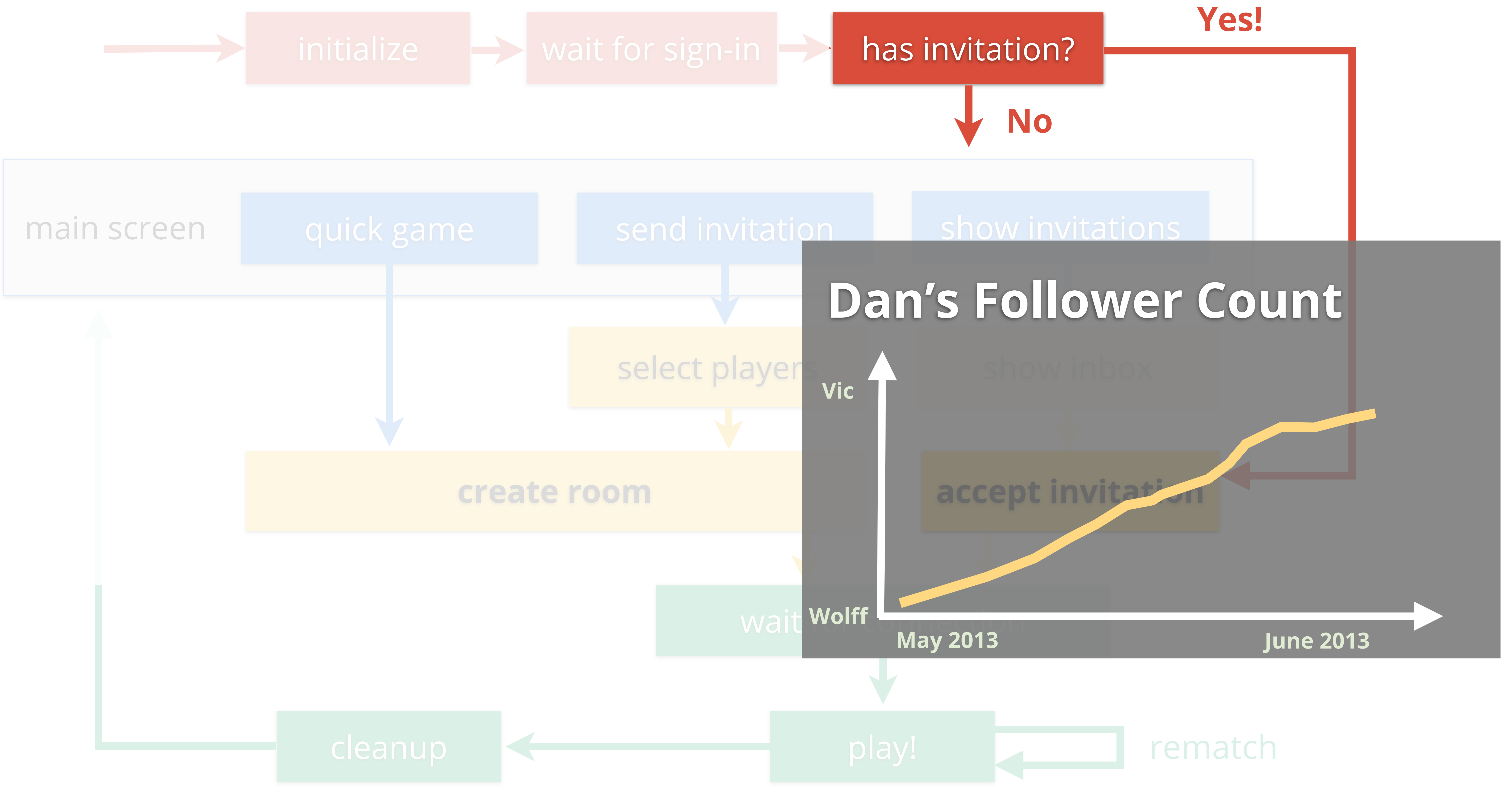
Accept the invitation

Java

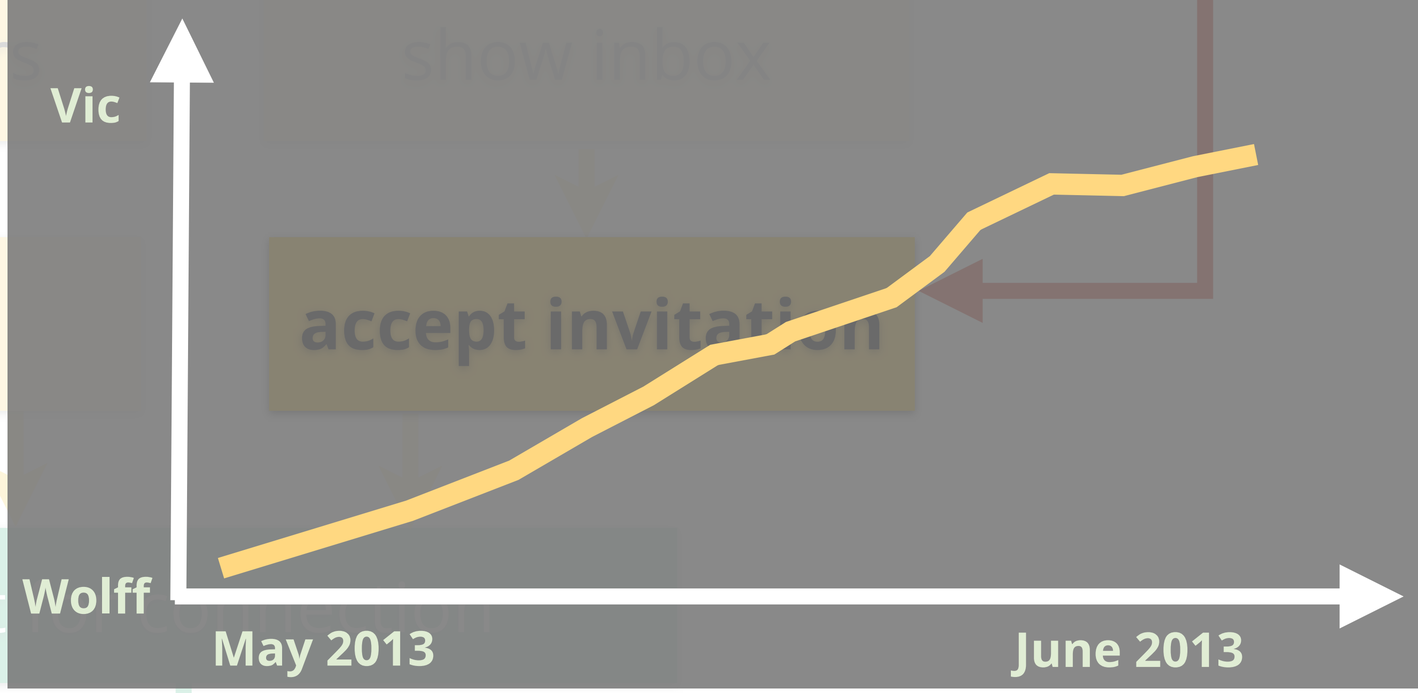
```
// room config (set callbacks and invitation ID)
RoomConfig.Builder configBuilder =
    RoomConfig.builder(this)
      .setMessageReceivedListener(this)
      .setRoomStatusUpdateListener(this)
      .setInvitationToAccept(inv.getInvitationId())

// join room
mGamesClient.joinRoom(configBuilder.build());
```

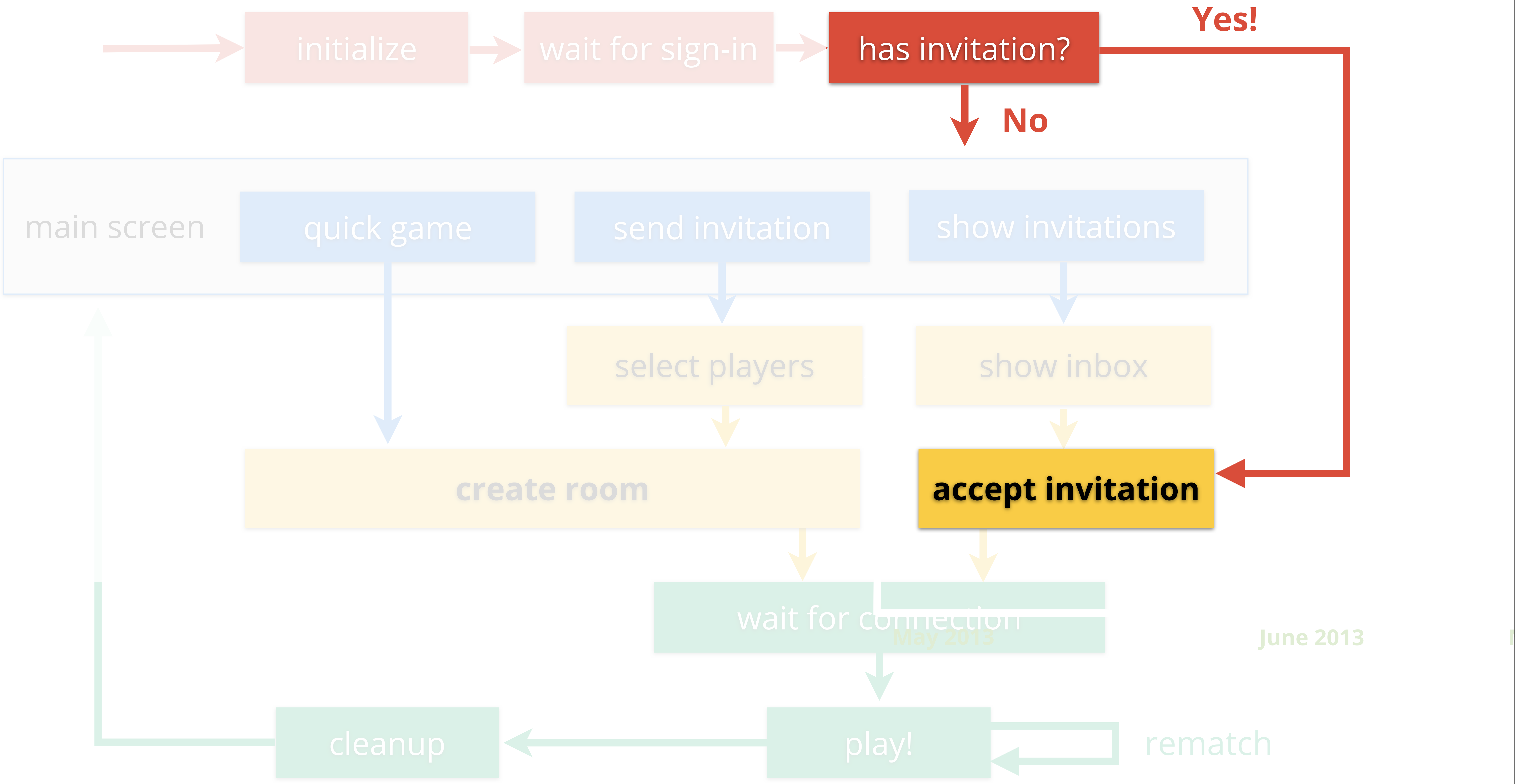


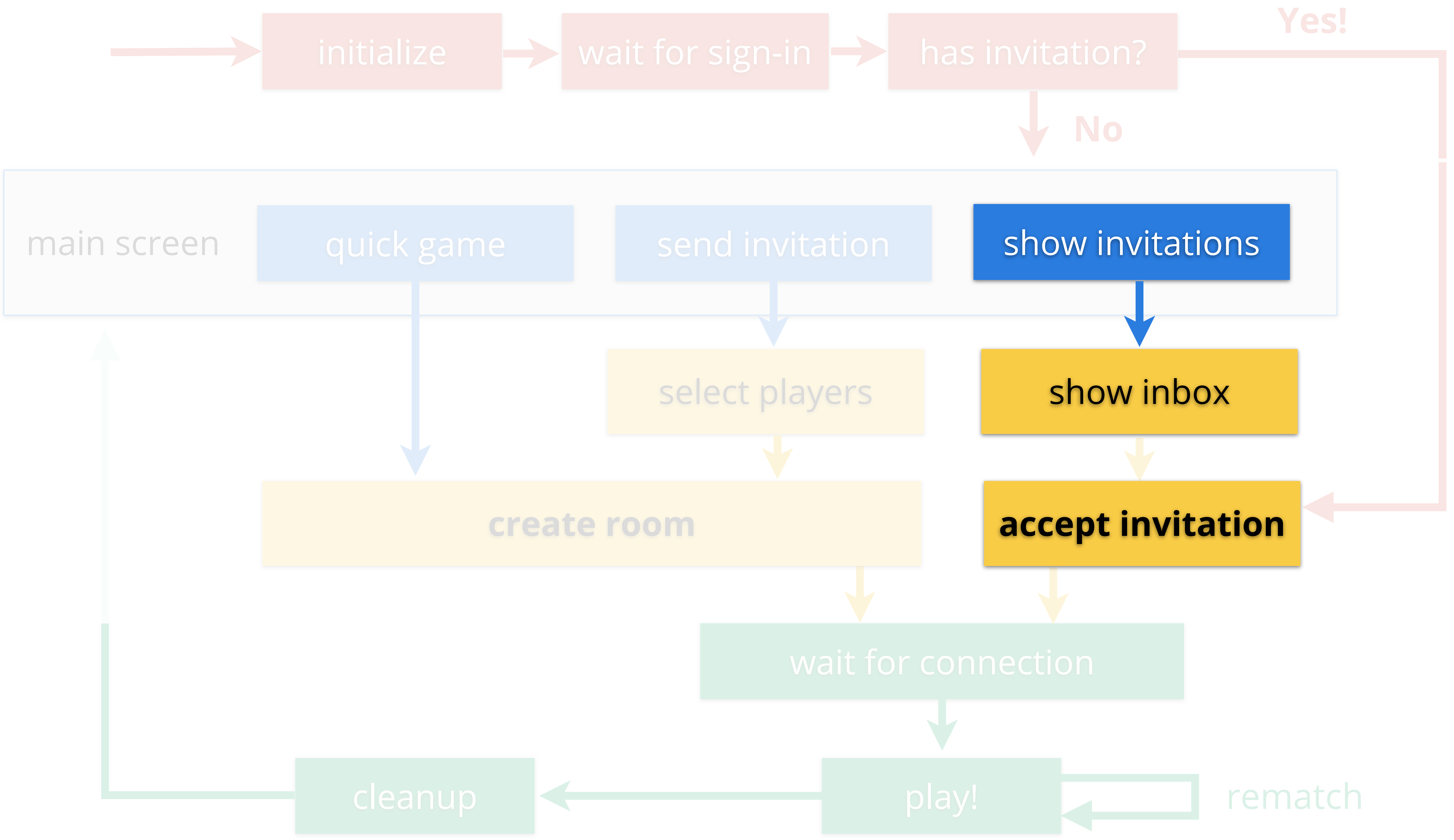
May 2013 June 2013



Dan’s Follower Count



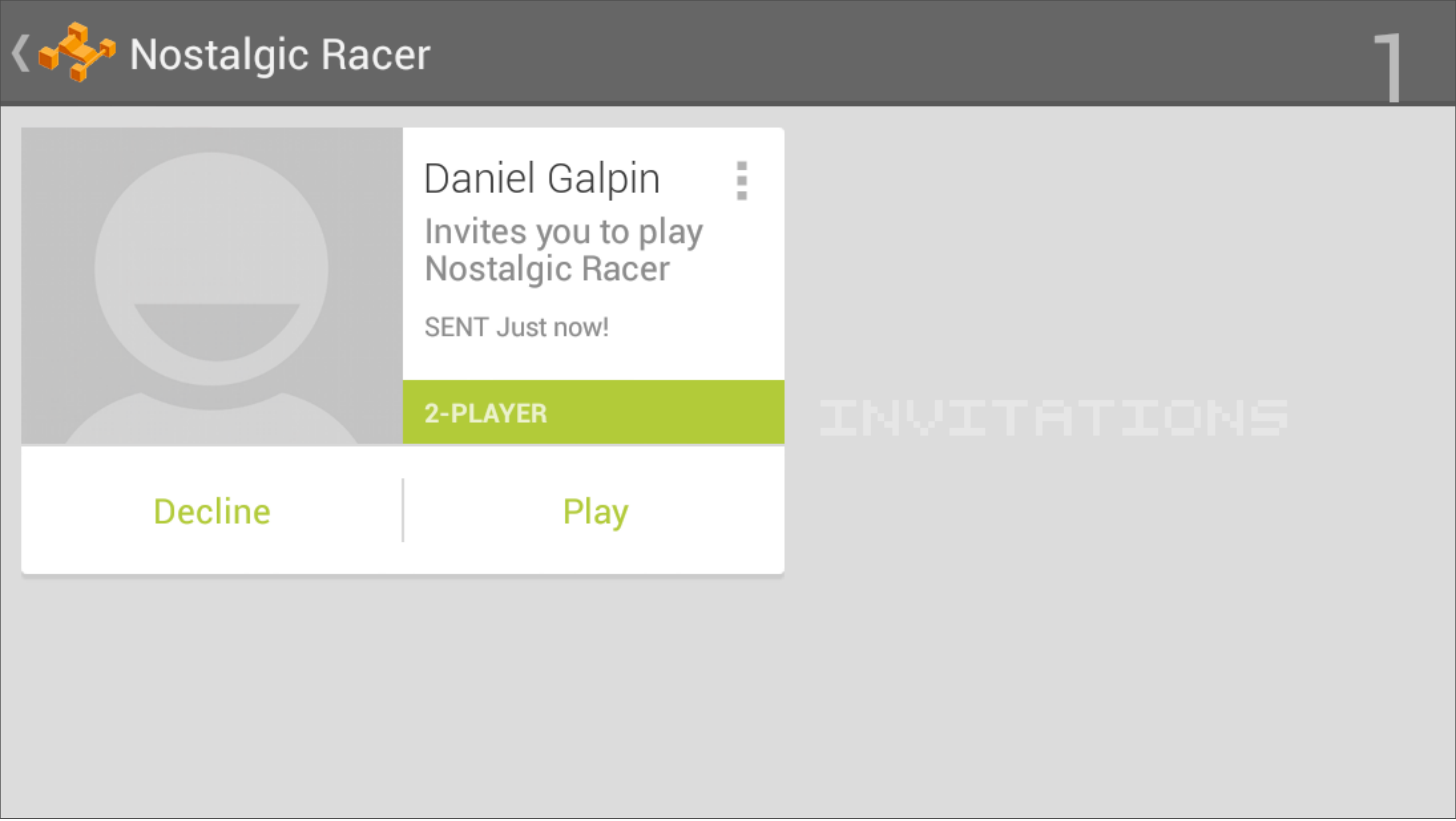

initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
May 2013
June 2013
cleanup
play!
rematch


Yes!
initialize
wait for sign-in
has invitation?
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch

Accept the invitation

Java

```java
// request code
final static int RC_INBOX = 10001;

// launch the intent to show the
// invitation inbox screen
Intent intent = mGamesClient.getInvitationInboxIntent();
startActivityForResult(intent, RC_INBOX);
```



Nostalgic Racer

1

Daniel Galpin

Invites you to play Nostalgic Racer

SENT Just now!

Accept the invitation

Java

```
@Override
public void onActivityResult(int request, int response,
                             Intent data) {
```

Accept the invitation

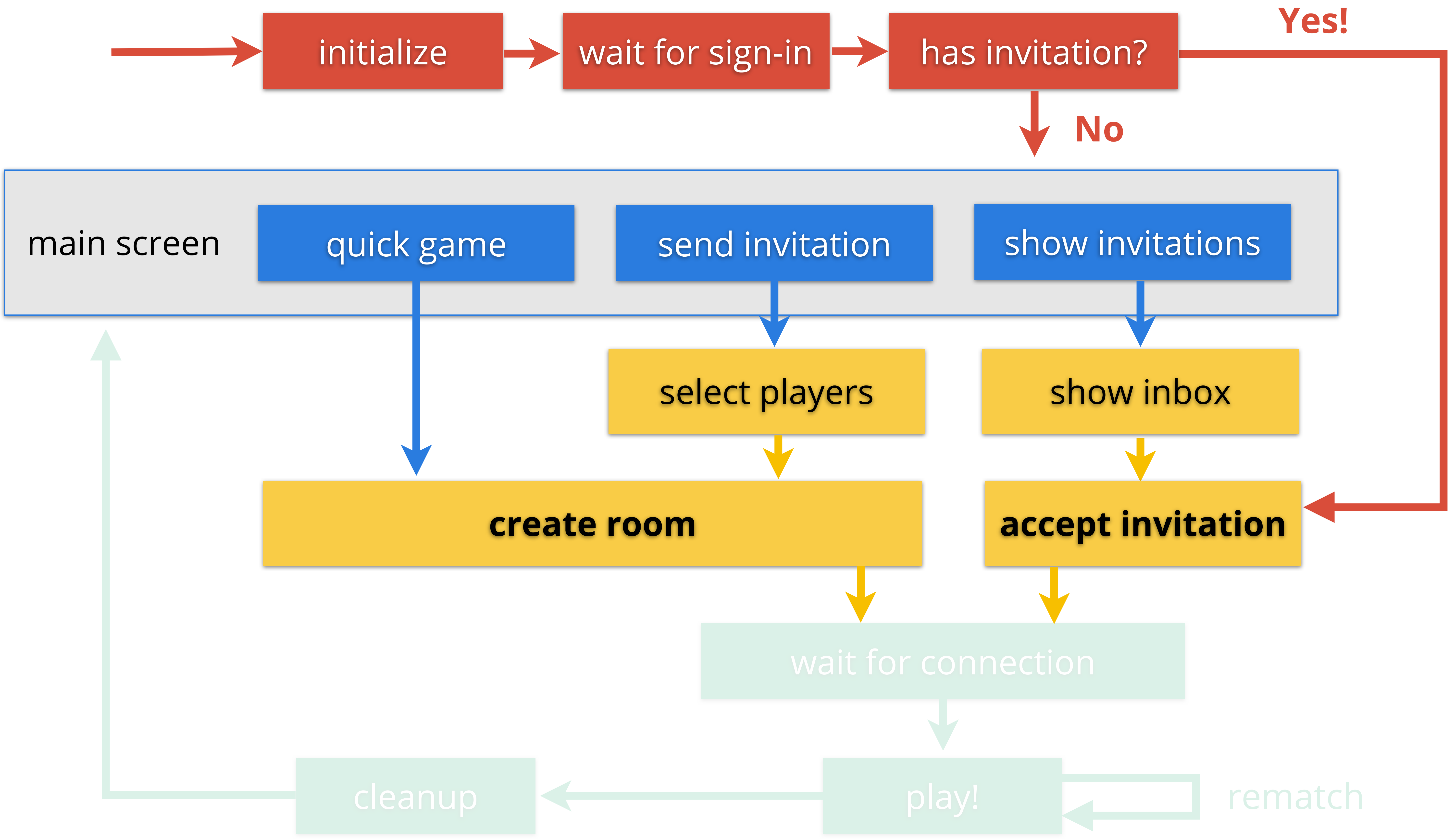
Java

```java
@Override
public void onActivityResult(int request, int response,
                             Intent data) {

    if (request == RC_INV_INBOX) {
        if (response != Activity.RESULT_OK) return;

        Bundle extras = data.getExtras();
        Invitation inv = extras.getParcelable(
                    GamesClient.EXTRA_INVITATION);
```

Accept the invitation (exactly as before)

Java

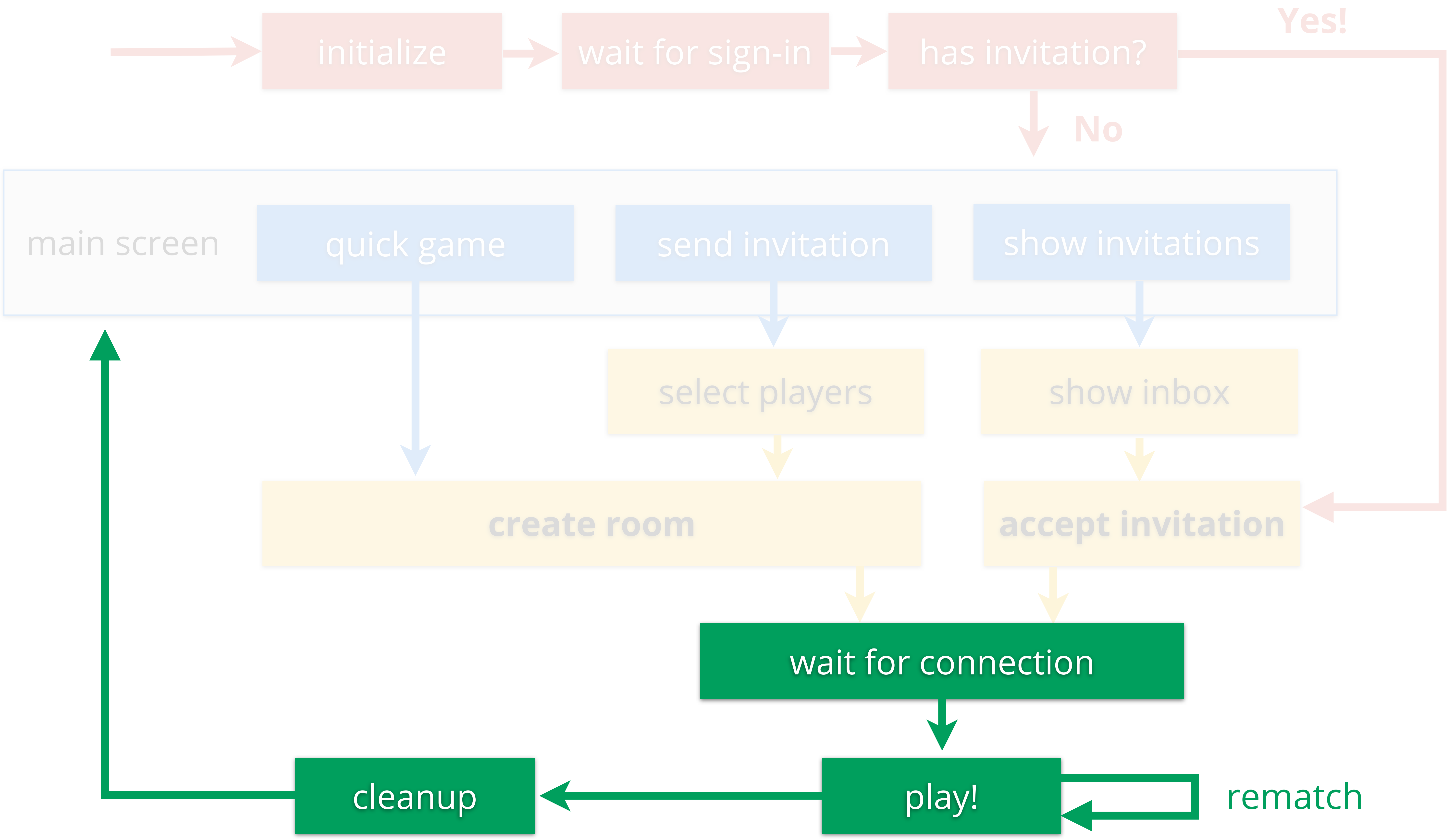
```java
// room config (set callbacks and invitation ID)
RoomConfig.Builder configBuilder =
    RoomConfig.builder(this)
      .setMessageReceivedListener(this)
      .setRoomStatusUpdateListener(this)
      .setInvitationToAccept(inv.getInvitationId())

// join room
mGamesClient.joinRoom(configBuilder.build());
```


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

Invitation Listener - Advanced Games Topics

I/O 2013, **Room 5**, Time **4:40pm**

Looking for invitation listeners? Come to our advanced talk and we'll cover that and much more!


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

Wait for connection.

Wait is boring.

Wait is boring.

Boring wait screen? - Advanced Games Topics!

I/O 2013, **Room 5**, Time **4:40pm**

Keep your users happy during the wait with our ready-to-use **Waiting Room UI**.

Wait for connection

Java

```java
@Override
public void onRoomConnected(int code, Room room) {
```

Wait for connection

Java

```java
@Override
public void onRoomConnected(int code, Room room) {
    if (code != GamesClient.STATUS_OK) {
        // failed to connect room
        // (show error)
        return;
    }
```

Getting basic room info

Java

```java
// Get room ID
mRoomId = room.getRoomId();

// Get list of participants
mParticipants = room.getParticipants();

// Get current player's participant ID
mMyId = room.getParticipantId(
       mGamesClient.getCurrentPlayerId());
```

Player ID vs. Participant ID

Player ID

Participant ID

Player ID vs. Participant ID

Player ID

Participant ID

real life

permanent

Player ID vs. Participant ID

Player ID	Participant ID
real life	**in-game**
permanent	temporary

Player ID vs. Participant ID

Player ID

real life

permanent

1055235951615183875329

Participant ID

in-game

temporary

Player ID vs. Participant ID

Player ID	Participant ID
real life	**in-game**
permanent	temporary
1055235951615183875329	**Cw241_vc6j4h_323871_fd38_Q**

invited friends $\longrightarrow$ Have both player ID and participant ID

automated $\longrightarrow$ **Only** have participant ID

In-game, use participant ID.

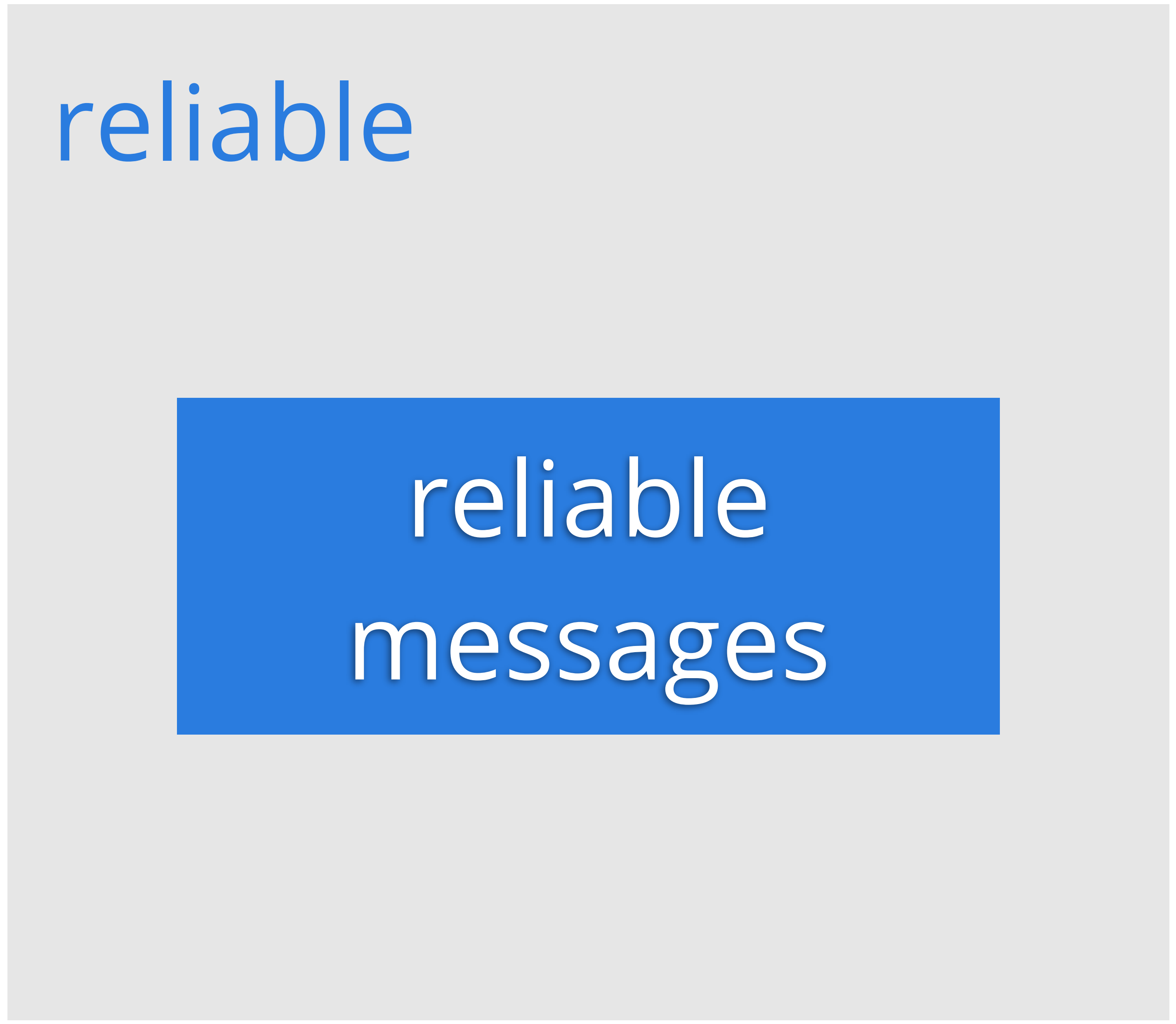
Java

```java
// Get room ID
mRoomId = room.getRoomId();

// Get list of participants
mParticipants = room.getParticipants();

// Get current player's participant ID
mMyId = room.getParticipantId(
       mGamesClient.getCurrentPlayerId());
```

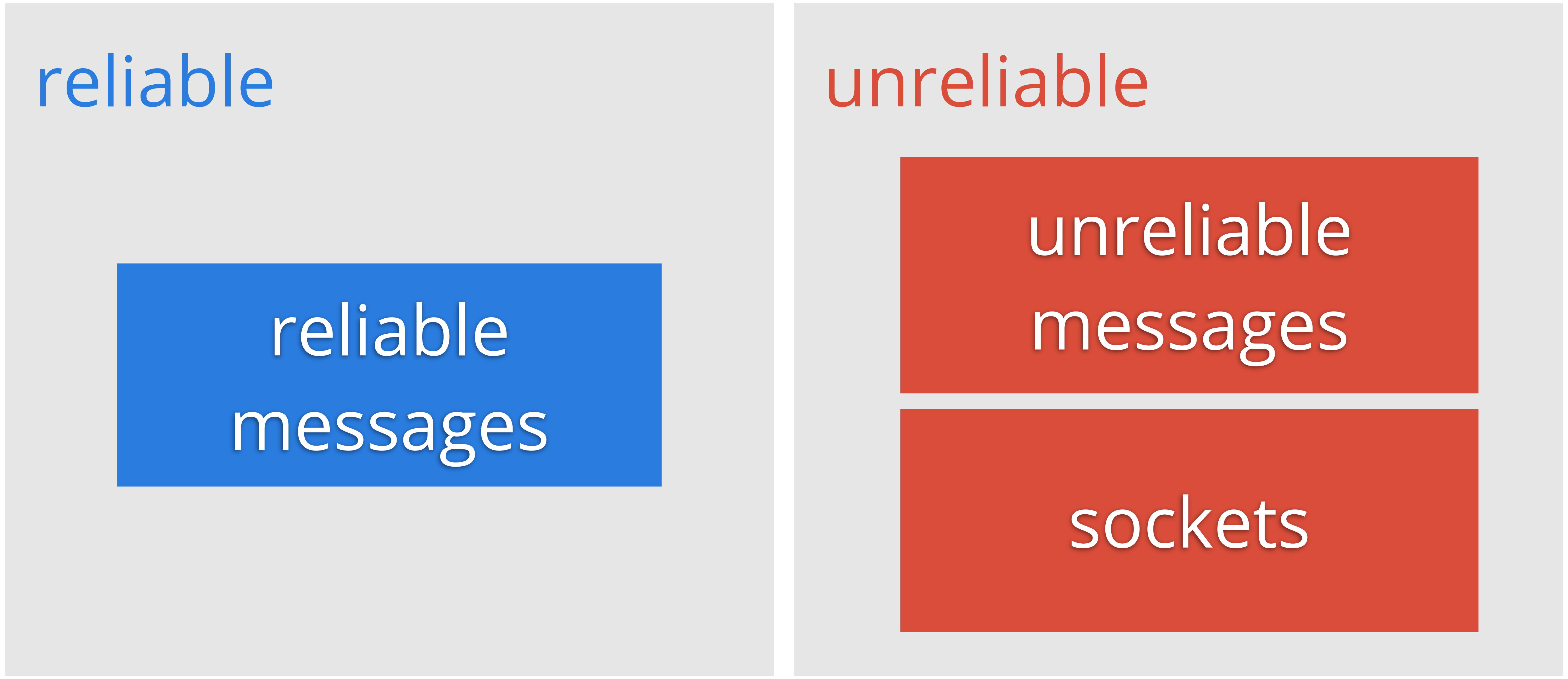
Ready to play

playing = sending/receiving messages

playing = sending/receiving messages



playing = sending/receiving messages



	reliable	**unreliable**
	think TCP	*think UDP*
delivered?		
in order?		
integrity?	always	

	reliable *think TCP*	**unreliable** *think UDP*
delivered?	definitely	
in order?	definitely	
integrity?	always	

	reliable *think TCP*	**unreliable** *think UDP*
delivered?	definitely	probably
in order?	definitely	probably
integrity?	always	always

Send a reliable message

Java

```java
String participantId = ....; // recipient
byte[] msg = ....; // data
```

Send a reliable message

Java

```
String participantId = ....; // recipient
byte[] msg = ....; // data

mGamesClient.sendReliableRealTimeMessage(
    null, msg,
    mRoomId, participantId);
```

Send an unreliable message

Java

```java
String participantId = ....; // recipient
byte[] msg = ....; // data
```

Send an unreliable message

Java

```java
String participantId = ....; // recipient
byte[] msg = ....; // data

mGamesClient.sendUnreliableRealTimeMessage(
    msg, mRoomId, participantId);
```

Send an unreliable message

Java

```java
String participantId = ....; // recipient
byte[] msg = ....; // data

mGamesClient.sendUnreliableRealTimeMessage(
    msg, mRoomId, participantId);
```

```
sendReliableRealTimeMessageToAll()
sendUnreliableRealTimeMessageToAll()
```

Receive a message

Java

```
// remember this?
```

Receive a message

Java

```
// remember this?
RoomConfigBuilder builder = RoomConfig.builder(this)
                .setMessageReceivedListener(this)
                .setRoomStatusUpdateListener(this);
```

Receive a message

Java

```java
@Override
public void onRealTimeMessageReceived(
                     RealTimeMessage rtm) {




}
```

Receive a message

Java

```java
@Override
public void onRealTimeMessageReceived(
                                        RealTimeMessage rtm) {
    // get real-time message
    byte[] b = rtm.getMessageData();

    // process message
}
```

caveats

don’t litter the heap

don't litter the heap

let go of the buffer

don't litter the heap

let go of the buffer

use fast serialization

don't litter the heap

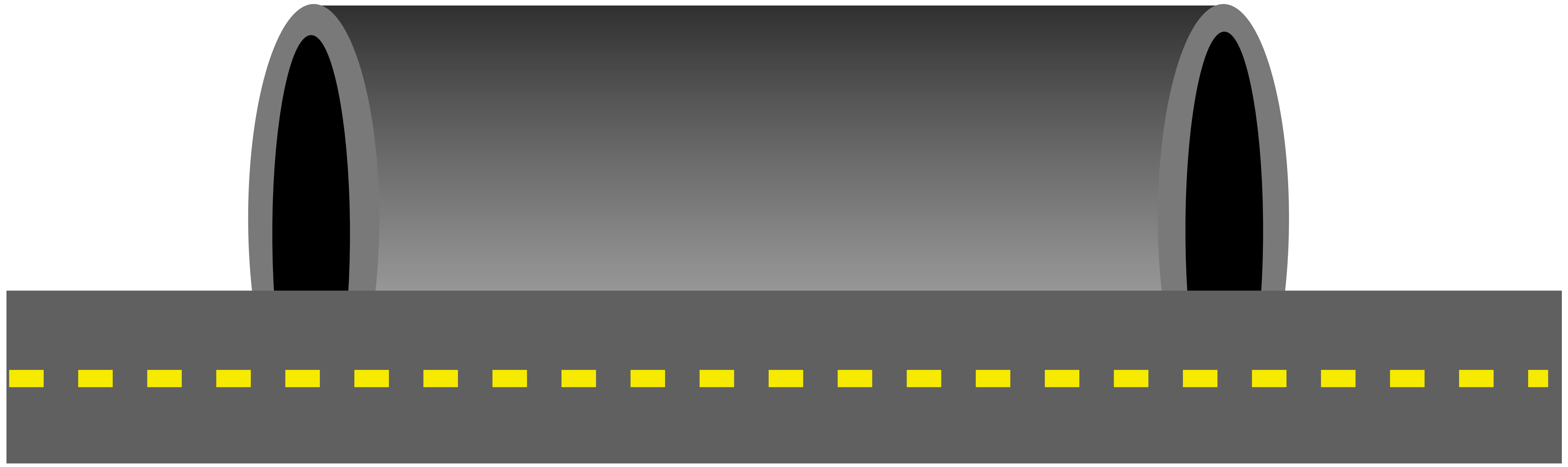
let go of the buffer

use fast serialization

trust no one

Problems

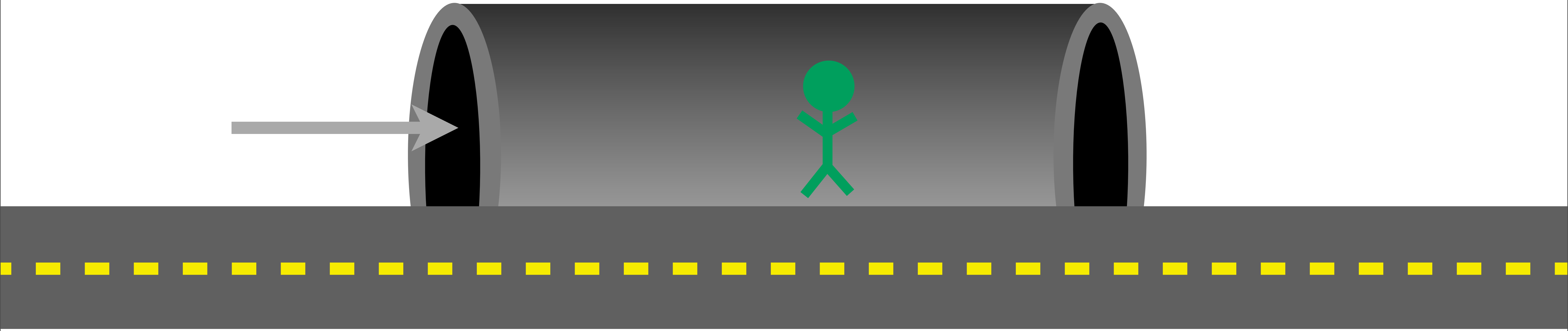
a tunnel (*)



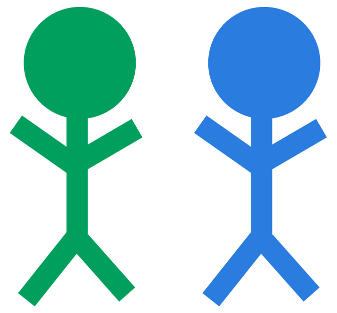
(*) imagination required.

You got disconnected

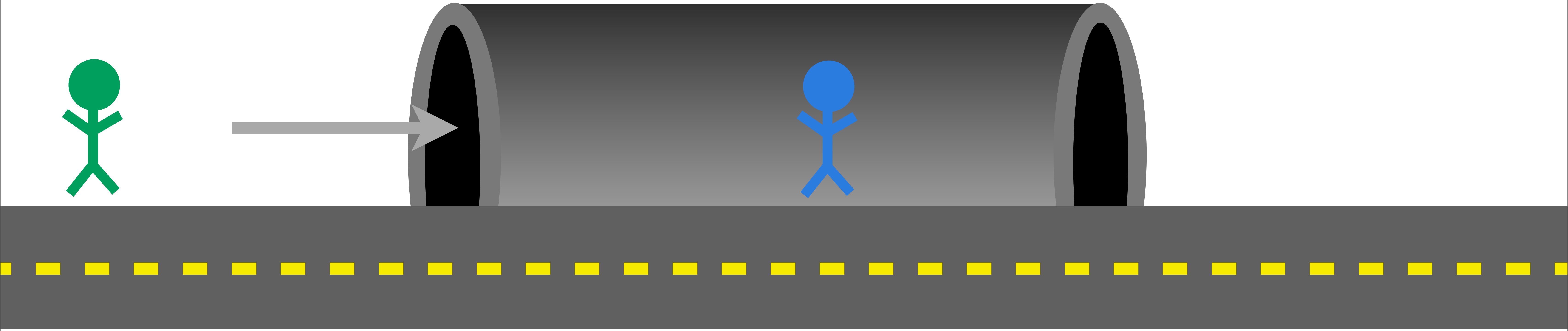
You got disconnected



Peer disconnected



Peer disconnected



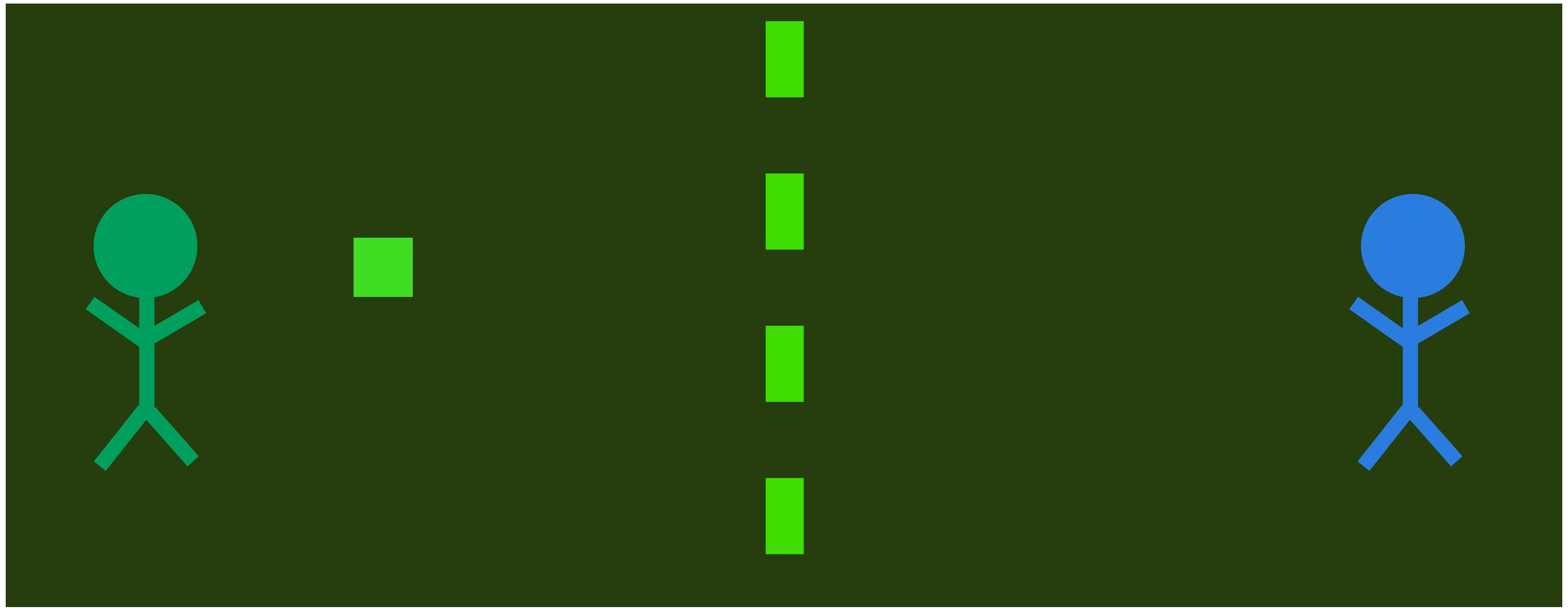
meh

meh

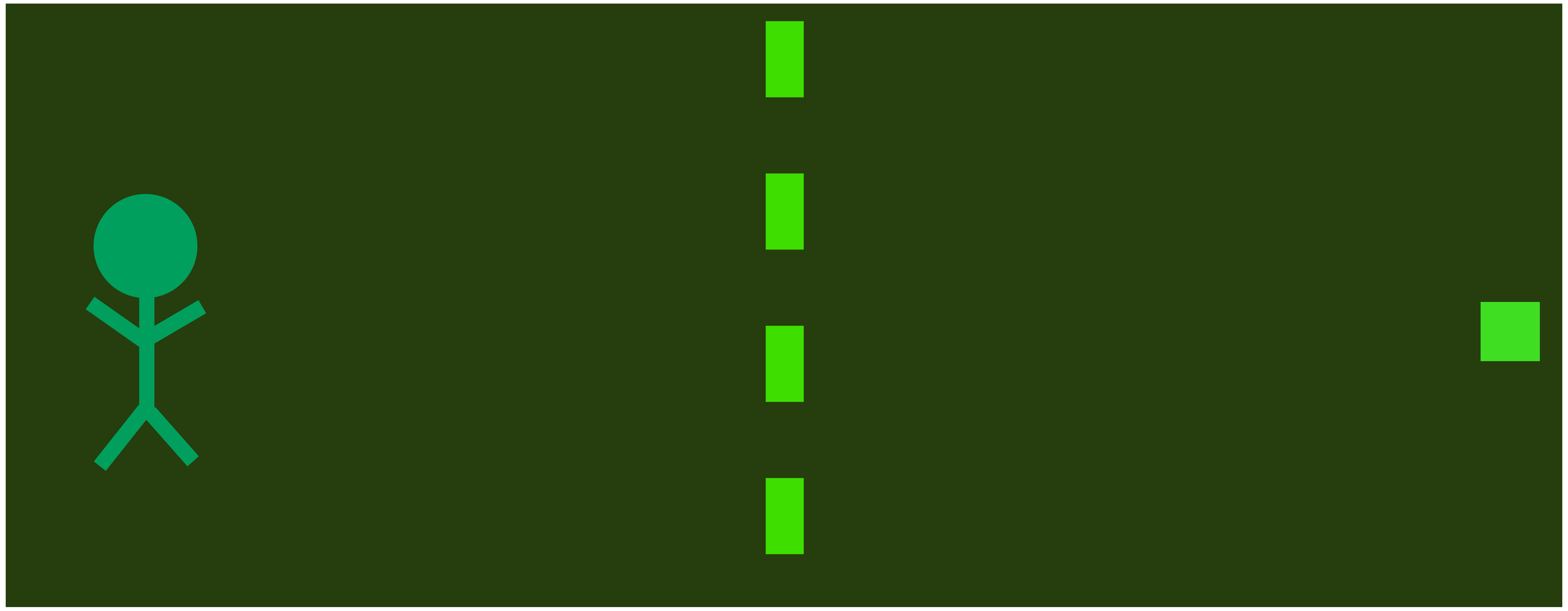
meh

Peer declines invitation.

meh



meh



Peer leaves game.

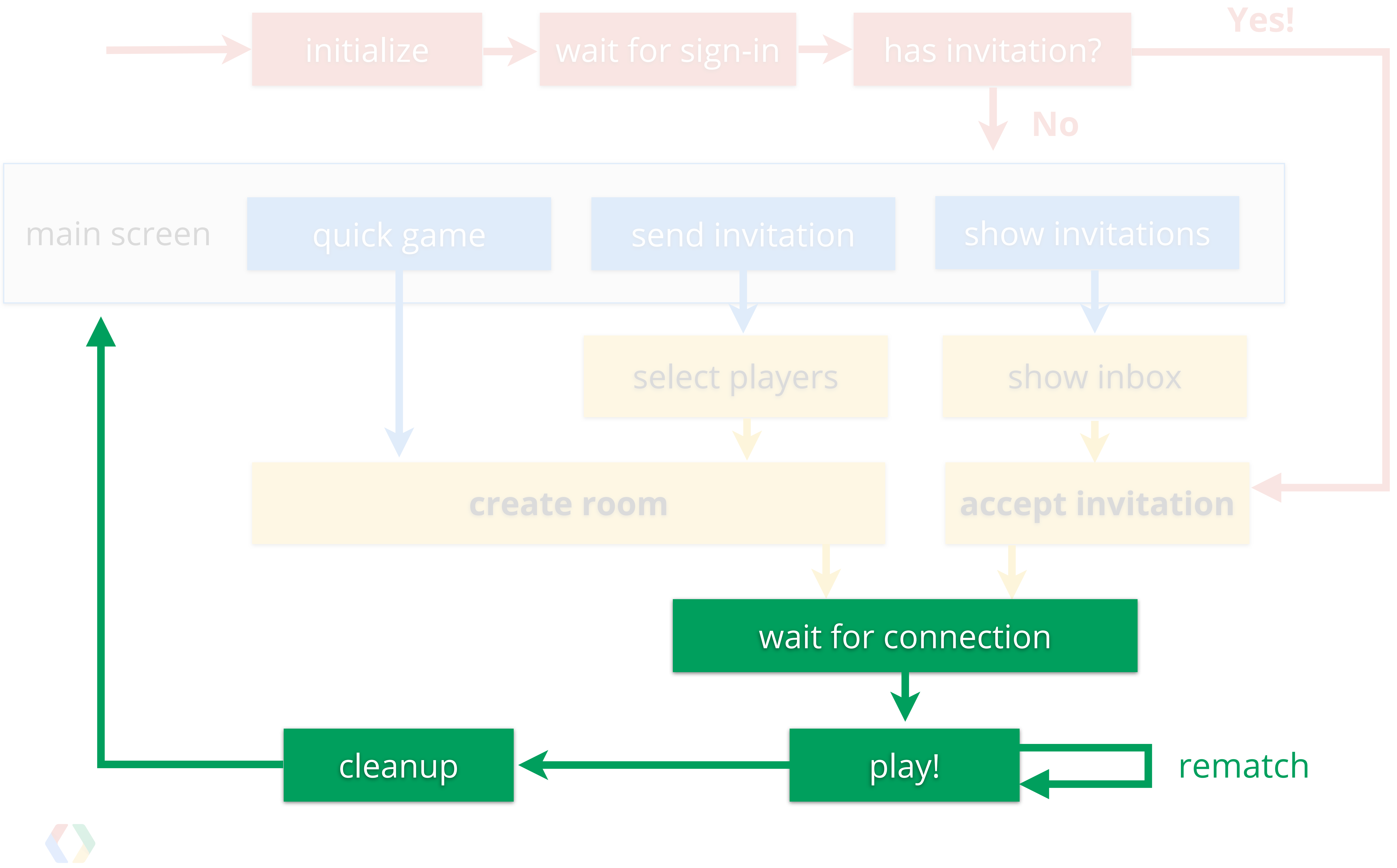
Callbacks

Java

```java
void onDisconnectedFromRoom(Room room)
void onPeersDeclined(Room room, List<String> who)
void onPeersDisconnected(Room room, List<String> who)
void onPeerLeft(Room room, List<String> who)

// see RoomStatusUpdateListener
// and RoomUpdateListener
```

If peers decline, you won’t get **onRoomConnected()**.

```
void onPeersConnected(Room room, List<String> who)
```


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
play!
rematch
cleanup

Leave the room

Java

```java
// leave room
mGamesClient.leaveRoom(this, mRoomId);
```

Leave the room

Java

```java
// leave room
mGamesClient.leaveRoom(this, mRoomId);

@Override
public void onLeftRoom(int code, String roomId) {
    // Left room. Ready to create or join
    // another room.
}
```

Don’t forget to leave on onStop()

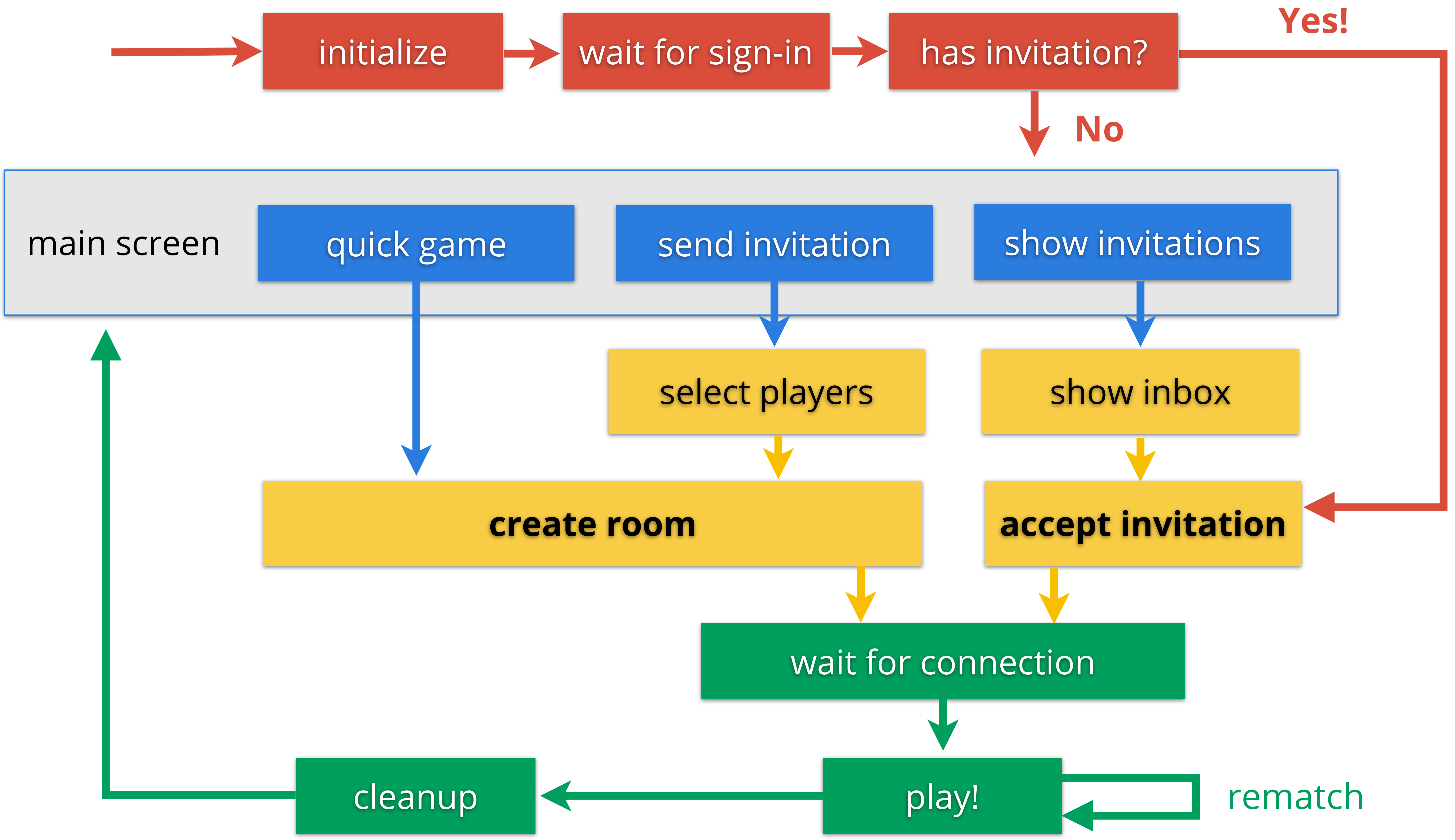
Java

```java
@Override
void onStop() {







}
```

Don't forget to leave on onStop()

Java

```java
@Override
void onStop() {
    super.onStop();
    // leave room
    if (mRoomId != null) {
        mGamesClient.leaveRoom(this, mRoomId);
        mRoomId = null;
    }
}
```


initialize
wait for sign-in
has invitation?
Yes!
No
main screen
quick game
send invitation
show invitations
select players
show inbox
create room
accept invitation
wait for connection
cleanup
play!
rematch

Where do I go from here?

Where do you go from here?

Monday, May 20,

Reasons not to start now

about:blank

Thank You!

Bruno: plus.google.com/+BrunoOliveira

Wolff: www.wolffdobson.com

Dan: goo.gl/yf2Gy