Scaling Peer-to-Peer Games in Low Bandwidth Environments

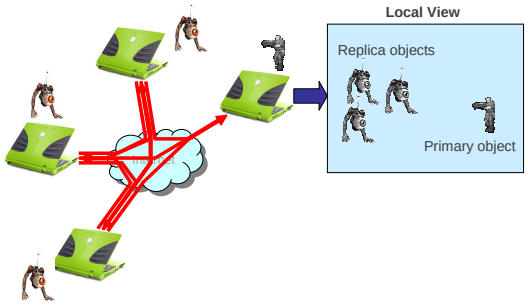
Jeff Pang
Carnegie Mellon

Frank Uyeda
U.C. San Diego

Jacob R. Lorch
Microsoft Research

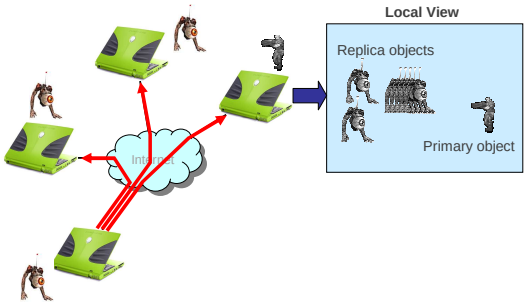
1

P2P Games



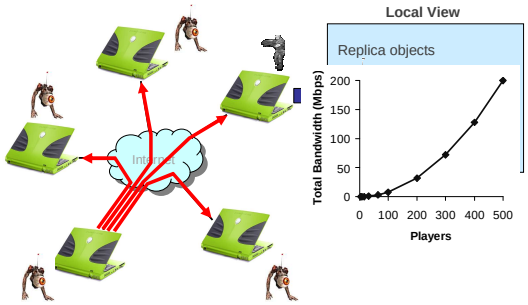
2

P2P Games



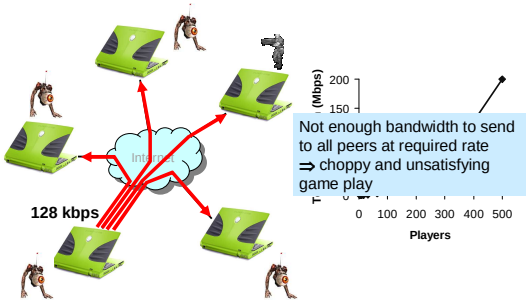
3

P2P Games



4

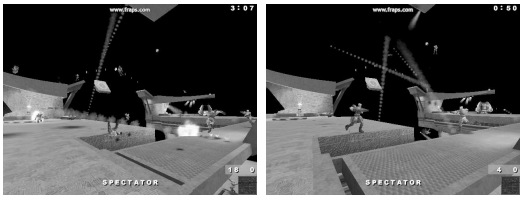
P2P Games



Not enough bandwidth to send to all peers at required rate
⇒ choppy and unsatisfying game play

5

Not Enough Bandwidth



P2P Quake on a LAN

P2P Quake on a Cable Modem

6

Our Solution: **Donnybrook**

- Contributions
  - Provide insights that enable scaling
  - Design techniques that codify these insights
  - A prototype implementing these techniques
- User study demonstrates effectiveness
  - Preserves fun in low bandwidth environments
  - Increases scalability by an order of magnitude

7

Design Principles

- Player attention is bounded by a constant
  Example: I focus on my current target
  ⇒ Total attention scales *linearly* with # players
- Realism should not be sacrificed for accuracy
  Example: "Teleporting" to correct locations is jarring
  ⇒ Don't always minimize error in because of updates
- Interaction must be timely and consistent
  Example: If I shoot you, you should get hit immediately
  ⇒ Prioritize interaction (i.e., inter-object writes)

8

Design Principles

- Player attention is bounded by a constant
  - Technique: **Focus Set**
- Realism should not be sacrificed for accuracy
  - Technique: **Guidable AI**
- Interaction must be timely and consistent
  - Technique: **Pairwise Rapid Agreement**

9

Focus Set

- Each player divides other peers into two classes:
  - **Focus Set**: players most focused on me
    - Update every frame
  - **Best Effort**: everyone else
    - Update when bandwidth is available
- α bandwidth allocated to Focus Set, (1- α) to Best Effort
- Who goes in my Focus Set?
  - Attention(**i**,**j**) = how much a player **i** is focused on player **j**
  - Attention(**i**,**j**) sent from peer **i** to peer **j** periodically
  - Peers with the highest Attention(**i**,**j**) go in peer **j**'s Focus Set

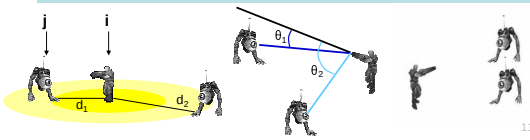
10

Focus Set

- Who goes in my Focus Set?
  - Attention(**i**,**j**) = how much a player **i** is focused on player **j**
  - Attention(**i**,**j**) sent from peer **i** to peer **j** periodically
  - Peers with the highest Attention(**i**,**j**) go in peer **j**'s Focus Set

Attention(**i**,**j**) =

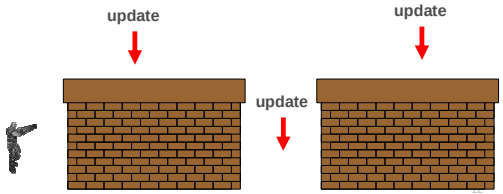
$f_{proximity}(d_1)$ + $f_{aim}(\theta_1)$ + $f_{interaction\text{-}recency}(t_1)$



11

Guidable AI

- **Problem:** Best effort peers receive infrequent updates
- **Solution:** Smooth state changes with AI
  - **Example**: use existing path finding code to make replica move



12

Donnybrook in Action

P2P Quake on a Cable Modem

P2P Quake on a Cable Modem with Donnybrook

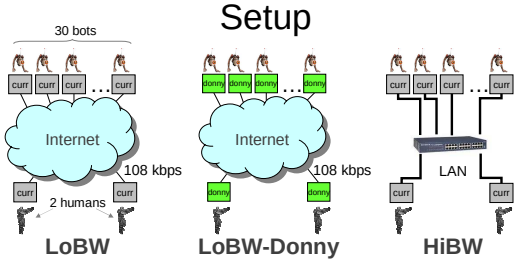
13

Evaluation

- Questions
  - How much does Donnybrook improve playability in low bandwidth environments?
  - How close is Donnybrook in a low bandwidth environment to a LAN ("optimal")?
- Methodology
  - Modified Quake III to run on Donnybrook
  - User study to evaluate user satisfaction
  - Simulation to evaluate fairness (See paper)

14

Setup



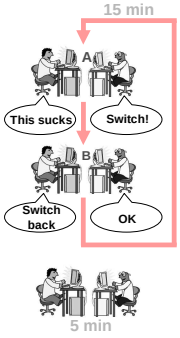
- ~5 updates/sec
- Focus set size = ~4
- Best effort rate = ~1 update/sec
- 20 updates/sec

15

Setup

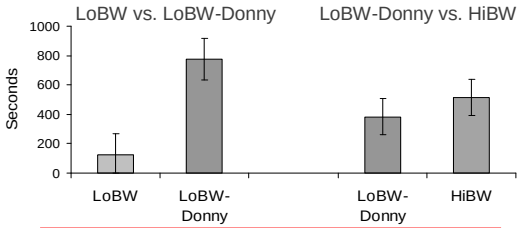
User Study Procedure

- Before experiment, practice on HiBW
- Tell players two Quake III "servers" exist: A and B
- Start playing on A, can vote to switch to B
- When both players vote, game continues on B
- Can vote to switch back and forth
- Analog to how players choose game servers (if good, stay, otherwise leave and try another)
- Play new game on least-used version so they can compare



16

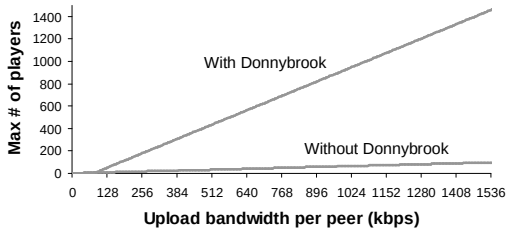
Total Stay Time



Three additional metrics in the paper support these conclusions.

17

Projected Scalability



18

Summary

- Donnybrook enables large-scale P2P games in low bandwidth environments
- Three key design principles:
  - Player attention is bounded by a constant
  - Interaction must be timely and consistent
  - Realism should not be sacrificed for accuracy
- Future work:
  - Constant-size Focus Set doesn't work if everyone is focused on one player (e.g., the flag carrier in CTF)
  - Use a multicast tree to deliver frequent updates in this scenario

Donnybrook: The Missing Slides

Expansion Pack

Meta-Conclusions

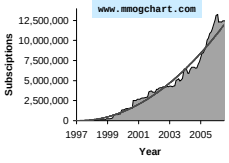
- We should do more games research
  - Has the market-share and user base
  - Numerous large scale distributed systems problems
- Bounded player attention applicable to other systems
  - Prioritizing data collection in P2P sensor aggregation
  - Varying fidelity in P2P video conferencing
- "End-to-end" evaluation should consider user experience
  - Donnybrook's replica consistency is not great
  - Users don't notice or don't care

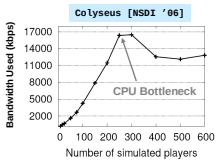
Other P2P Games Challenges

- Cheating
  - Migration, Witnesses
    - Bharambe [CMU thesis work]
  - Trusted Computing / Hardware Sensors
    - Johnson et al. [Intel]
- Fault Tolerance
  - DHT self-healing / replica placement:
    - Colyseus [NSDI 2006]
    - SimMUD [INFOCOM 2004]
- Persistence
  - General P2P storage replacements for DB

Why P2P Games?

- Massively multiplayer games are popular
  - Recent research:
    - Colyseus [NSDI '06]
    - SimMUD [INFOCOM '04]
- Centralized First Person Shooter (FPS) games scale poorly
  - Quadratic Bandwidth
  - CPU Limited





Game Execution Model

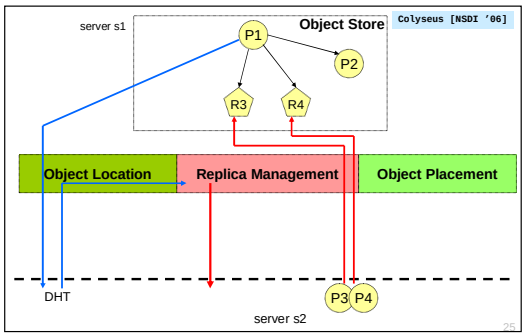
- **Game State:**
  - Collection of distinct *objects* (players, missiles, items, etc.)
- **Game Execution:**
  - Each object has a `Think` function:

```
Think() { ReadPlayerInput(); DoActions(); ... }
```

  - Execute each object once per *frame*:

```
Each 50ms do {
    foreach object do {
        object->Think();
    }
}
```

Area of Interest Filtering



25

Design Comparison

- Existing P2P game architectures
  - Replicas can be as stale as the update interval
  - Update rate per peer is fixed
- Donnybrook's solution
  - QoS: Vary update rate per peer, some get priority
  - Guidable AI: Secondary replicas think for themselves

26

Pairwise Rapid Agreement

- **Interaction**: when player A modifies player B (i.e. A performs a write on B)
- **Goal**: modification is consistent and applied quickly
- **Insight**: # interactions scales slowly
  - Occur at human time scales ⇒ infrequent
  - Involve only 2 players ⇒ unicast
- **Solution**: prioritize all inter-object writes
  - Player A sends mod to Player B
  - Player B checks preconditions locally
  - Player B applies mod, sends result to A
- PRAs required in Quake III:
  - Damage, Death, Item Pickup, Door Opening

27

Prediction

- **Motivation**: state snapshots get stale fast
  - **Example**: players can traverse the entire diameter of a map in several seconds in Quake III
  - **Goal**: send prediction of state at time of next expected update
  - **Example**: predict where a player will be at the next update
- **Predicted Properties**:
  - **Predict position**: use in-game simulation to figure how where physics brings player in next second
  - **Predict viewing angle**: use view angles to estimate the target a player is aiming at
  - **Predict Events**: use number-of-shots-fired to estimate when a player is "shooty," etc.

28

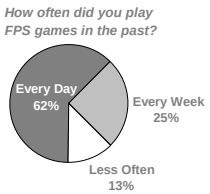
Guidable AI: Convergence

- **Problem:** Best Effort peers receive very infrequent updates
- **Solution:** Smooth state changes with AI
  - **Position**: use existing path finding code to make replica move smoothly
  - **Angle**: have AI turn smoothly toward predicted targets
- **Convergence**
  - **Motivation**: Players in focus should be represented more accurately, but should not "warp" to actual position
  - **Solution**: Converge to actual state when receiving frequent updates
    - Focus on player B
      ⇒ In player B's Focus Set, get frequent updates
      ⇒ Error(replica, actual) decreases with each update
    - When Error() < ε, use player B's update snapshots instead of AI
    - Error(a,b) = distance(a.pos,b.pos)

29

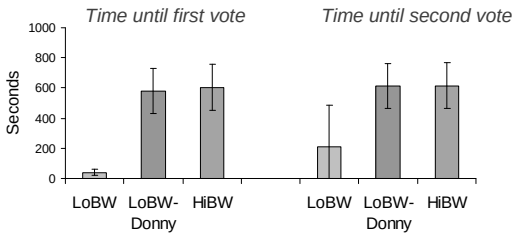
Setup

- User Study Stats
  - **LoBW-Donny vs. LoBW**: 12 trials
  - **LoBW-Donny vs. HiBW**: 32 trials
  - 98 total participants



30

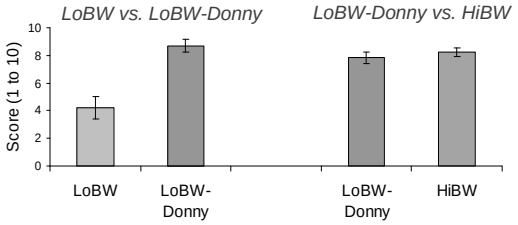
Departure Time



How long before a player wants to switch?

31

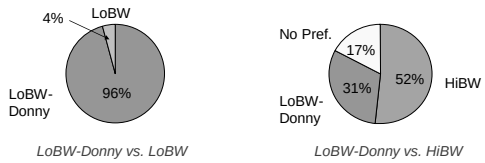
Fun Score



Survey: How fun was server A? Server B?

32

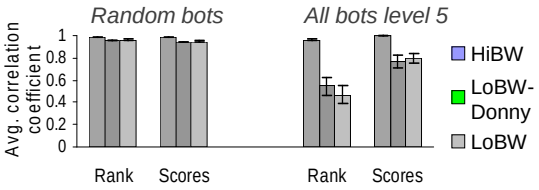
Preference



Survey: Was A or B more Fun?

33

Fairness



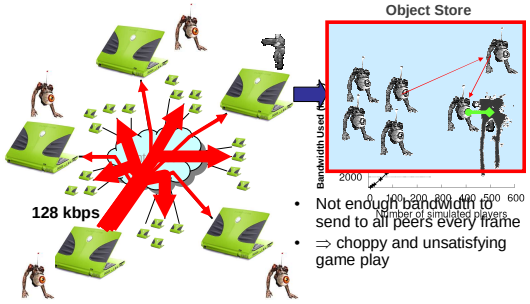
34

Donnybrook: The Elder Slides

Beta Test

35

P2P Games



- Not enough bandwidth to send to all peers every frame
- ⇒ choppy and unsatisfying game play

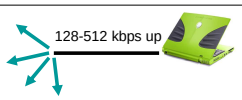
36

Outbound Capacity Problem

- Residential broadband is asymmetric
- In many games, players can see many others
- Insufficient upload capacity to send updates at required frequency

⇒

- Must send updates at lower frequency
- Replicas can be as stale as the update interval



Cost per Peer in Quake III:

Update rate = 20 updates/sec
Update size = ~100 bytes

⇒

Bwidth per peer ≥ 16kbps

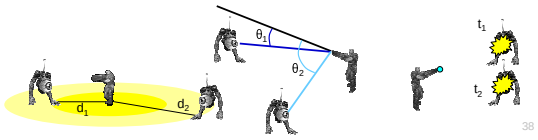
⇒

Supportable peers at 128kbps ≤ 8

37

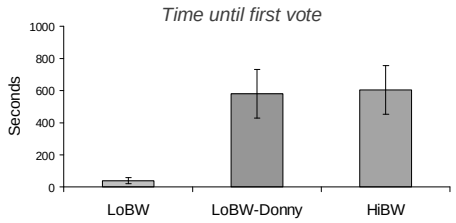
Focus Set

- Who goes in my Focus Set?
  - Attention(*i*,*j*) = how much a player *i* is focused on player *j*
  - Attention(*i*,*j*) sent from peer *i* to peer *j* periodically
  - Peers with the highest Attention(i,j) go in peer ***j***'s Focus Set



38

Departure Time



How long before a player wants to switch?

39