

Microsoft Research

Search Microsoft Research

[Our research](#) [Connections](#) [Careers](#) [About us](#)[All](#) [Downloads](#) [Events](#) [Groups](#) [News](#) [People](#) [Projects](#) [Publications](#) [Videos](#)

Multiplayer Gaming for Smartphones

By [Douglas Gantenbein](#)

June 28, 2011 9:00 AM PT

Like 9Tweet 78

People love their smartphones—and they love to play games on them. On common smartphone software platforms, including [Windows Phone](#), games are among the most popular applications and constitute the vast majority of downloads.

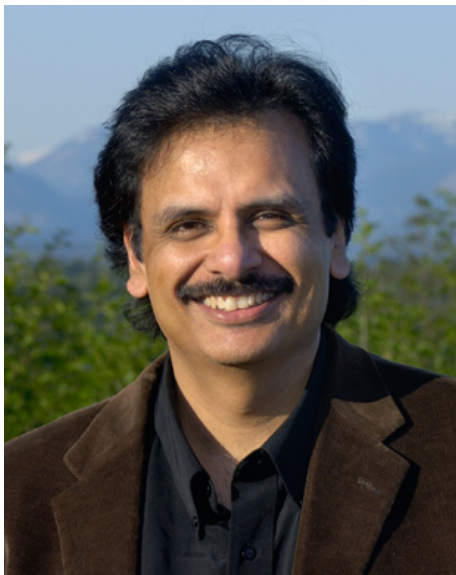
"Games are killer apps for smartphones," says [Victor Bahl](#), director of the [Mobile Computing Research Center](#) at [Microsoft Research Redmond](#). "They're fantastic time-killers, and for smartphones, the killer app is the app that kills time."

Gaming on a smartphone comes with compromises, though. Most smartphone-based games are single-player games—*Angry Birds* is a perfect example—or turn-based, in which two or more players queue up on a single phone to play games.

Bahl and a team of Microsoft Research scientists and interns created Switchboard, a cloud service that enables smartphones to link to each other for multiplayer gaming. Joining Bahl on the project were Justin Manweiler, an intern, and Romit Roy Choudhury, a visiting researcher, both from Duke University; and [Sharad Agarwal](#) and [Ming Zhang](#) from Microsoft Research Redmond. Agarwal was the project leader for Switchboard.

Switchboard: A Matchmaking System for Multiplayer Mobile Games will be one of the papers presented during [MobiSys 2011](#), the ninth International Conference on Mobile Systems, Applications, and Services, scheduled for June 28-July 1 in Washington, D.C. Microsoft Research and the National Science Foundation are supporters of the conference, which is aimed at researchers and academics interested in topics such as security, location management, application support, mobile architectures, sensor networks, energy management, and ways to solve mobility problems. Microsoft Research scientists will present a total of six papers during MobiSys and three associated workshops.

It was 2009 when Bahl started to think about how to give smartphone users the ability to engage in live gaming. Although not a game player himself, he found the technical challenges fascinating.



Victor Bahl

"Multiplayer gaming on smartphones is a very challenging technical problem," he says. "Gamers move around, the wireless channel is hostile, the bandwidth you need is not always there, disconnections can happen, and smartphones are energy-constrained."

"If you just take a game and put it on a smartphone, it is not going to work."

The key to Switchboard is that it works as a "matchmaker," finding potential game players and assigning them to a group most similar for each player's phone and network connection.

"Let's say there are 20 candidates for a game," Bahl says. "You might find that some have very low battery levels—even if they wanted to play the game, they would be able to play for only a few minutes. Others might be moving

around so much that lots of handoffs [between cell towers] will happen, resulting in disconnections, and they will not enjoy the game. Or some might be located in highly congested

Products

- [Windows Phone](#)
- [Halo](#)
- [Fable](#)
- [Windows Mobile 6.5](#)

People

- [Victor Bahl](#)
- [Sharad Agarwal](#)
- [Ming Zhang](#)

Groups

- [Mobile Computing Research Center](#)

Facilities

- [Microsoft Research Redmond](#)

Publications

- [Switchboard: A Matchmaking System for Multiplayer Mobile Games](#)

Events

- [MobiSys 2011](#)

areas where a lot of people are using phones and the available network bandwidth is low.”

Switchboard finds suitable game candidates by “pinging” their phones to estimate latency: the time needed to move a packet of data. Latency is important to gaming because a long lag time—in the hundreds of milliseconds—to move data can affect game play.

“There are techniques for multiplayer games to handle ‘jitter’—occasional spikes in network delay,” Agarwal says. “When those techniques react to jitter, you may see a player or object jump from one spot to another. Such ‘glitches in the matrix’ become problematic if they happen frequently. A major challenge in this work is to predict what a player’s latency will look like for the foreseeable duration of a game and minimize such glitches.”

The Microsoft Research team used two HTC FUZE phones running [Windows Mobile 6.5](#) over AT&T’s wireless network to measure latency variation. They also used phones from other manufacturers operating over the T-Mobile network. Collecting those measurements was challenging.

“We had to take an awful lot of measurements in different parts of the country,” Bahl says. “You have to have a lot of data to say this is going to work with consistency and regularity.”

Some of those tests involved driving around city streets and highways while measuring latency. As the Switchboard paper wryly notes, “Unfortunately, the highway patrol did not let us conduct experiments much beyond 60 mph.”

Tool Development

By collecting extensive data and studying wireless tendencies, the team was able to develop tools that could ping phones and reliably predict which potential game players could participate. They found that phones needed to be pinged about 60 times over a 15-minute period to collect reliable information. That places a heavy burden on the network, so the team also devised a method for using the latency of one phone to predict the latency of others.

In addition, team members determined how much latency in a game stems from the wireless network and how much is caused by the Internet, because the Internet also plays a role in connecting wireless gamers.

The next step was to devise a way to group prospective game players. Switchboard achieves this matchmaking by looking at the latency requirements of a game—a fast-paced shooting game might require much lower latency than another game—and then finding potential players whose phones can handle those requirements.

Switchboard builds a “lobby service,” which works with the game developer’s code, and a cloud-based “grouping service.” The lobby service acts as a waiting room for potential players, parking players for a short time while their phones’ capabilities are determined. Once appropriate players are identified, the lobby service sends them to the grouping service, which then forms groups of players and enables them to start play.

Supporting the grouping service are complex algorithms that assign players so that they fit within the latency and group-size constraints determined by a game developer, while also maximizing the number of players in the group. All of this happens rapidly, so players are not left waiting in the “lobby.”

The Switchboard paper also shows how to scale up game play so that it could meet the demands of a growing gaming environment.

“Just like most media content,” Agarwal says, “mobile games suffer from popularity distributions—a few games are tremendously popular at any point in time. In contrast to game consoles, phone markets are measured in the hundreds of millions. Matchmaking for mobile games has to consider scale as a primary design and research challenge from the start.”

Switchboard demonstrates that a phone-to-phone service with strict latency requirements can be built. One hurdle, however, is that a smartphone needs a public IP address to connect it to other phones. AT&T offers this service for a \$3 monthly fee, and Sprint offers it for free, but other carriers do not. Bahl believes that as IPv6, the next generation of Internet protocol, gains more usage, wireless phone carriers will offer public IP addresses as a standard service.

Bahl also says that although Switchboard was developed using 3G wireless networks, it will work perfectly well with high-speed 4G networks being implemented.

“It doesn’t matter what kind of network



Sharad Agarwal

people are using or what kind of device," he says. "The issue of matchmaking will always exist, and the issue of grouping players will always exist."

People with time on their hands will always exist, too. With help from Switchboard, they might be able to band together to fight alien forces attempting to conquer the Earth, instead of idly watching airport LCD screens to learn when their flight *finally* will take off.

Microsoft Research Contributions to MobiSys 2011

Computer Viruses in Urban Indian Telecenters: Characterizing an Unsolved Problem

Prasanta Bhattacharya, Microsoft Research India; and William Thies, Microsoft Research India.

Computing Security in the Developing World: A Case for Multidisciplinary Research

Yahel Ben-David, University of California, Berkeley; Shaddi Hasan, University of California, Berkeley; Joyojeet Pal, University of Michigan; Matthias Vallentin, University of California, Berkeley; Saurabh Panjwani, Microsoft Research India; Philipp Gutheim, University of California, Berkeley; Jay Chen, New York University; and Eric Brewer, University of California, Berkeley.

Hyke: A Low-cost Remote Attendance Tracking System for Developing Regions

Azarias Reda, University of Michigan; Saurabh Panjwani, Microsoft Research India; and Edward Cutrell, Microsoft Research India.

Switchboard: A Matchmaking System for Multiplayer Mobile Games

Justin Manweiler, Duke University; Sharad Agarwal, Microsoft Research Redmond; Ming Zhang, Microsoft Research Redmond; Romit Roy Choudhury, Duke University; and Paramvir Bahl, Microsoft Research Redmond.

SystemSens: A Tool for Monitoring Usage in Smartphone Research Deployments

Hossein Falaki, UCLA; Ratul Mahajan, Microsoft Research Redmond; and Deborah Estrin, UCLA.

Towards Improved Web Acceleration: Leveraging the Personal Web

Azarias Reda, University of Michigan; Edward Cutrell, Microsoft Research India; and Brian Noble, University of Michigan.



Add a comment...

☒ Also post on Facebook

Posting as **Milind Gokhale** ▾

Comment

Facebook social plugin

Follow us



[Contact us](#) [Privacy & cookies](#) [Terms of use](#) [Trademarks](#) [Code of conduct](#) [Feedback](#) [Mobile](#)

©2015 Microsoft

