

Introduction to the Flash Platform P2P API

Jun Heider
RealEyes Media



- Jun Heider
 - RealEyes Media
 - <http://www.realeyes.com>
 - Blog
 - <http://www.iheartair.com>
 - Twitter
 - @coderjun
- Special Thanks to Influxis for some server space!



- Reduced Server and Bandwidth Cost (P2P)
- Lower Latency Communications (UDP)
- Data Prioritization (Audio over Video and Data)
- IP Mobility (Rendezvous Service)
- P2P Across Devices! (PC/Tablet/Phone/TV)



- **Demonstration of P2P**
 - Call Center POC
 - Browser
 - Tablet
 - Smartphone



- Gaming:

- MAX Racer:

- <http://alternativaplatform.com/en/demos/maxracer/>

- Battlecell:

- <http://www.influxis.com/battlecell/>

- Haxball:

- <http://haxball.appspot.com/>



- Audio/Visual

- Adobe:

- LiveCycle Collaboration Service
- Internal Communications
- MAX Keynote

- Chatroulette (Early Adopter):

- <http://www.chatroulette.com/>

- Radar:

- <http://www.flashmobileblog.com/radar/>
- <http://www.flashmobileblog.com/2010/08/27/p2p-radar-for-android/>



- Protocol
 - Real Time Media Flow Protocol (RTMFP)
 - UDP-based
 - 128-bit AES encryption over the wire
- Client
 - FP 10 / AIR1.5 for direct connections (1-1)
 - FP 10.1 / AIR 2 for RTMFP Groups
- Server (optional)
 - FMS 4
 - Cirrus (Non-commercial)



- Server is optional?
 - Local LAN peers can use discovery
 - Otherwise, need rendezvous service
 - Bootstrapping
 - Keeping track of peers and their IP addresses



- Local LAN
 - LAN Peer Discovery (OSI Layer 3)
 - Application Level Multicast (OSI Layer 7)
- Over the Internet
 - Application Level Multicast (OSI Layer 7)
- Enterprise Network
 - IP Multicast (OSI Layer 3)
 - Application Level Multicast (OSI Layer 7)
 - Fusion (Both, but IP Multicast preferred)



- Firewall BAD!
 - Symmetric NAT <-> Symmetric NAT BAD!
- Test Connectivity Tool
 - <http://cc.rtmfp.net>
 - Check FMS 4 developer guide for details
- TURN proxy support in Flash Player
 - mms.cfg
 - RTMFPTURNProxy=TURN_proxy (IP or DNSName)



- Day-to-Day Classes
 - NetConnection
 - GroupSpecifier
 - NetGroup
 - NetStream
 - NetStatusEvent



- Create NetConnection*
 - Connect the NetConnection
- Create GroupSpecifier
 - Get the GroupSpec from the GroupSpecifier
- Create NetGroup/NetStream*
 - Pass in NetConnection
 - Pass in GroupSpec
- *Listening for NetStatus Events



- Connecting... Various Endpoint Types
 - Local LAN
 - "rtmfp:"
 - FMS
 - "rtmfp://server/app"
 - Cirrus
 - "rtmfp://p2p.rtmfp.net/cirrusDevKey"



- Basic P2P Application



- Group NetStream
 - Publish Camera and Microphone
 - Use it the same as standard NetStream, passing in group spec string



- NetGroup Posting
 - Call post() on your instance
 - `NetGroup.post(message:Object):String`



- NetGroup Object Replication
 - Use methods on your NetGroup instance
 - instance.addHaveObjects()
 - instance.addWantObjects()
 - instance.removeHaveObjects()
 - instance.removeWantObjects()
 - Listen for NetStatusEvent



- NetGroup Directed Routing
 - Use methods on your NetGroup instance
 - `instance.sendToAllNeighbors()`
 - `instance.sendToNearest()`
 - `instance.sendToNeighbor()`
 - Listen for NetStatusEvent



- Sending Data

- Typed Object via [RemoteClass] metadata
- NetGroup.post() - identical messages are filtered, use timestamps or serials

- Object Replication

- Optimal 64 KB Chunks



- NetStream/NetGroup have info properties
 - NetGroupInfo
 - NetStreamInfo
- Check additional properties for things to tweak
 - NetStream.multicastWindowDuration
 - Smaller value = less latency/reduced quality
 - NetGroup.replicationStrategy
 - Lowest Index or Rarest Index?



- **GroupSpecifier**

- Group Name + Capabilities = Group
- Set posting or publish passwords
 - `GroupSpecifier.setPostingPassword()`
 - `GroupSpecifier.setPublishPassword()`
 - `GroupSpecifier.groupspecWithAuthorizations()`
 - `GroupSpecifier.groupspecWithoutAuthorizations()`



- Let's check out some more of the API



- FMS Multicast App (use FMLE)
- FMS as Client Manager
- FMS as a Persistence Layer
 - Application.onAppStop() return false
- FMS as another RMTFP Group Peer



- Let's check out an FMS Monitoring App



- cocoon-p2p
 - <http://code.google.com/p/cocoon-p2p/>
- HydraP2P
 - <https://github.com/devboy/HydraP2P>
- RealTimeGameLib
 - <https://github.com/tomkrcha/RealtimeGameLib>



- Adobe Flash Media Server on Amazon Web Services
 - <http://www.adobe.com/products/flashmediaserver/amazonwebservices/>
 - You manage
- Influxis Enterprise P2P
 - https://secure1.influxis.com/webrootv10/custom_enterprise/index.aspx
 - Support Staff are FMS Experts :)



- Influxis for Hosting
 - SmartMesh API (Beta)
 - For gaming
 - InfluxisP2P API (Beta)
 - For Posting
 - Offers RTMP fallback
 - API is tied into custom SSAS on Influxis server



- Let's take a look at Influxis



Thank you!



- References:
 - <http://www.delicious.com/coderjun/flashplayerp2p>
- Interested in hosted P2P:
 - https://secure1.influxis.com/webrootv10/custom_enterprise/index.aspx
- Interested in consulting:
 - <http://realeyes.com>
- Any questions? jun@realeyes.com

