

Streaming Vidéo

IDS - Présentation finale

Amrani Ismaïl

Benoudina Mohamed Tarek

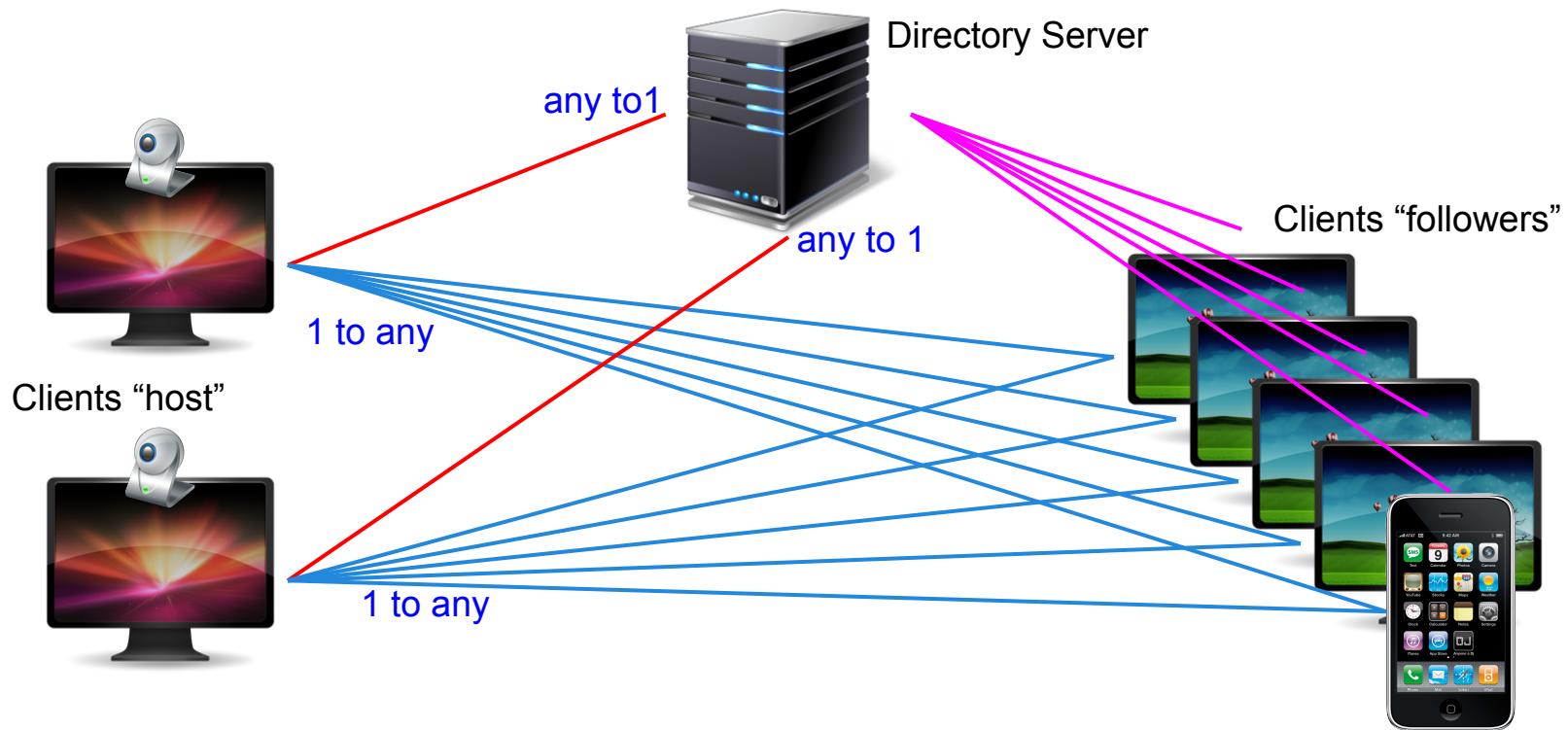
Hess Aurélien

Liu Chongguang

Olé Adrien



Architecture

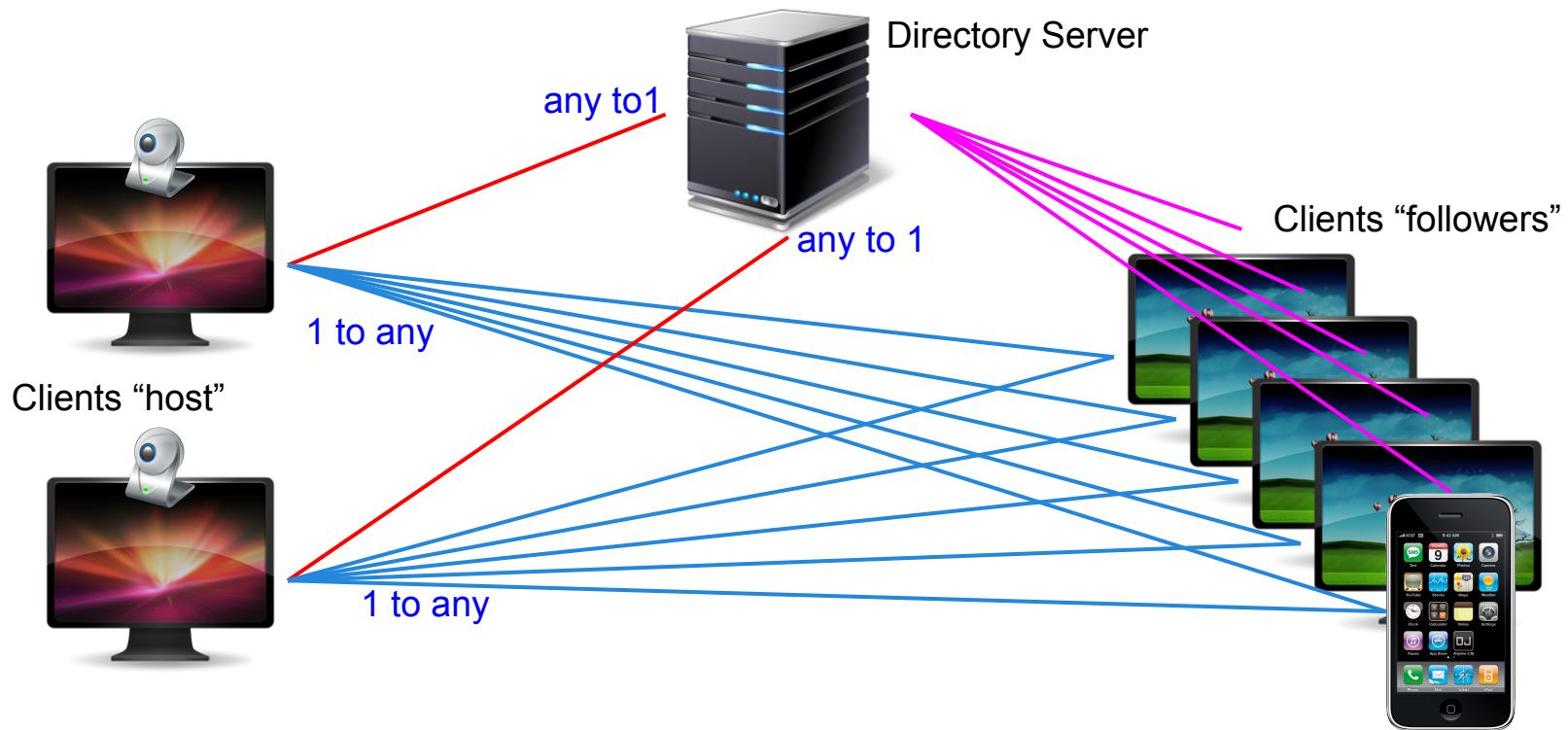


Démonstration





Architecture





Serveur de directory

But: référence les streams vidéo disponibles

- Les clients java communiquent en **RMI**
- Un **thread** séparé gère les clients iOS en **TCP**
- Gestion des **exceptions** avec les clients



Application “host”

But: crée des streams vidéo à partir de la webcam ou de captures d'écran

- Le nouveau stream est enregistré sur le serveur de directory en **RMI**
- Chaque stream est diffusé sur un **socket multicast**
- L'utilisateur peut gérer ses streams grâce à une **interface graphique simple**



Client java

But: récupère la liste des streams vidéo et affiche ceux désirés

- Récupération de la liste des streams vidéo sur le serveur de directory via **RMI**
- Récuperation de streams vidéo depuis le **socket multicast**
- Dispose d'une **interface graphique** simple d'utilisation



Transférer les images sur réseaux

1

2

3

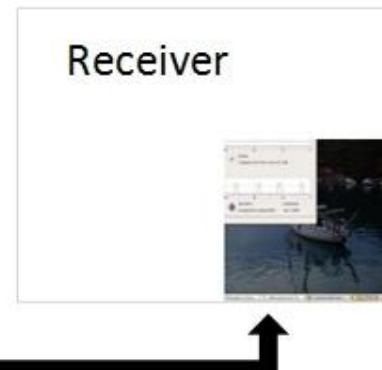
1 - IP Header (20 bytes) 2 - UDP Header (8 bytes) 3 - UDP Data (max 65507 bytes)

UDP Packet

- 65507 Bytes n'est pas suffisant pour une image de capture d'écran ou webcam.
- On doit découper les images en petits morceaux et les envoyer un par un.



Sender



Receiver

Image Sender and Receiver



Transférer les images sur réseaux

Protocole UDP ne garantit pas la bonne livraison des datagrammes à destination, ni leur ordre d'arrivée, on doit le gérer nous même.

Flags	8 bit	Contains SESSION_END and SESSION_START flag
Session Number	8 bit	Session the packet belongs to
Packets	8 bit	Number of packets in total
Maximum Packet Size	16 bit	Maximum size of each packet
Packet Number	8 bit	The number of the current package
Size	16 bit	The data size of the current package



Client iOS

But: récupère la liste des streams vidéos

- Connection au DirectoryServer java: port et adresse multicast
- Affichage de la liste de streams
- Lecture de packets **UDP**

Conclusion



- Beaucoup de temps passé pour faire fonctionner la webcam (drivers)
- Communication UDP iOS complexe et mal documentée
- Backend important, peu visible dans la démo



GROUP 5 IN ASSOCIATION WITH BENOIT GARBINATO AND BEHNAZ BOSTANIPOUR PRESENT IN TECHNICOLOR A
UNIVERSITY OF LAUSANNE AND UNIVERSITY OF NEUCHATEL PRODUCTION

AN IDS PROJECT "THE VIDEO STREAMER"

ADRIEN OLÉ AURÉLIEN HESS ISMAÏL AMRANI CHONGGUANG LIU TAREK BENOUDINA

IOS CODING TAREK BENOUDINA SYSTEM ARCHITECTURE CHONGGUANG LIU AND ISMAÏL AMRANI DIRECTORY SERVER
CHONGGUANG LIU AND ISMAÏL AMRANI EXCEPTION HANDLING ISMAÏL AMRANI WEBCAM HANDLING AURÉLIEN
HESS TESTING AND DOCUMENTATION ADRIEN OLÉ USER INTERFACE CHONGGUANG LIU AND ISMAÏL AMRANI
RESEARCH AND DEVELOPMENT AURÉLIEN HESS AND ADRIEN OLÉ SCREENSHOT AND IMAGE PROCESSING CHONGGUANG LIU
UDP COMMUNICATIONS CHONGGUANG LIU SOCKET COMMUNICATION ISMAÏL AMRANI JAVA CLIENT CHONGGUANG LIU
JAVA HOST CHONGGUANG LIU AND AURÉLIEN HESS DIRECTOR OF PRESENTATION AURÉLIEN HESS AND ISMAÏL
AMRANI ART DIRECTOR AURÉLIEN HESS PROJECT COORDINATION ISMAÏL AMRANI

DIRECTED BY GROUP 5 PRODUCTIONS
BASED ON A TRUE STORY

Question(s)?



That's all Folks!

