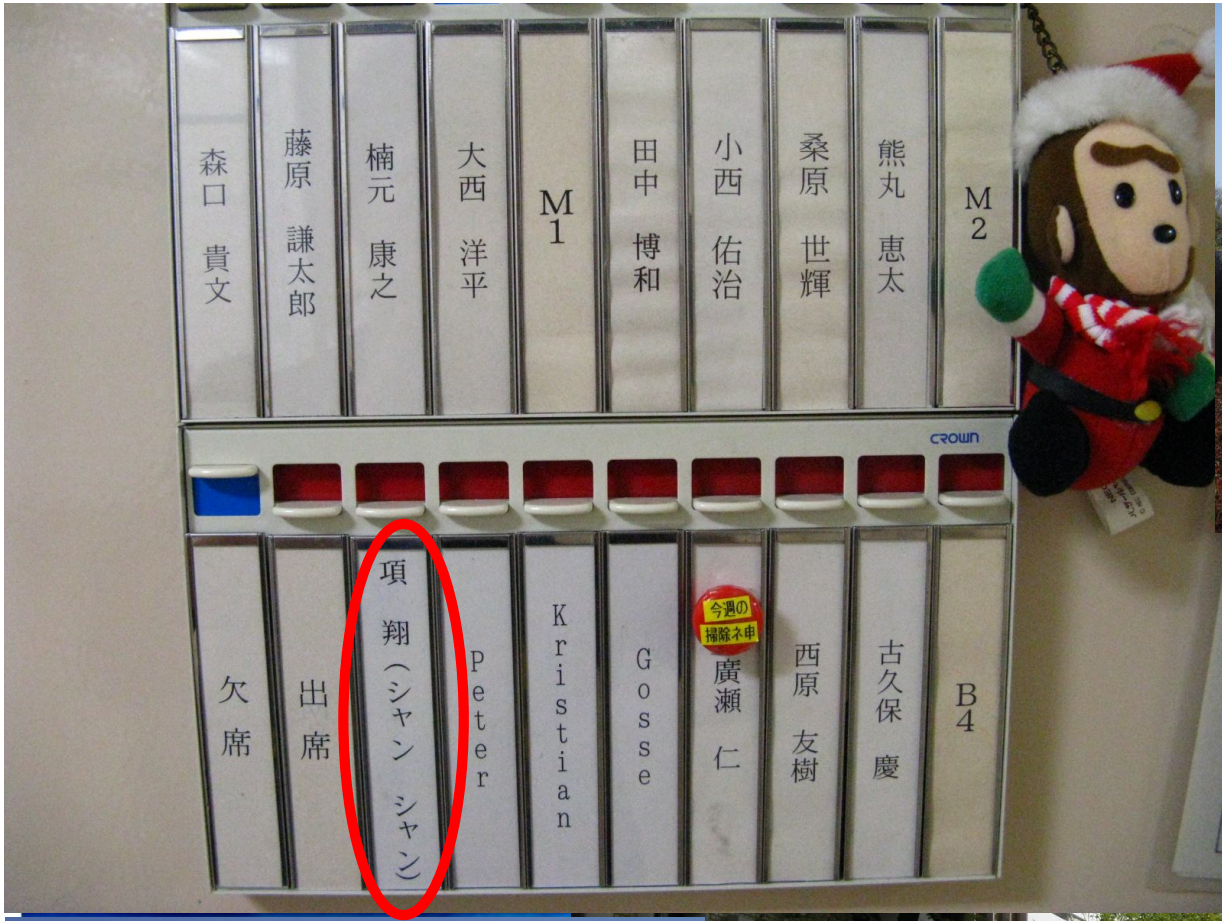
Ubiquitous and Customized P2P Picture Sharing based on PIAX

Progress Presentation

(till Oct. 30th, 2008)

Xiang Xiang (Visiting Researcher)
Graduate School of Engineering,
Osaka University, Japan


森口 貴文
藤原 謙太郎
楠元 康之
大西 洋平
M1
田中 博和
小西 佑治
桑原 世輝
熊丸 恵太
M2
欠席
出席
項 翔（シャン シャン）
Peter
Kristian
Gosse
今週の掃除ネ申
廣瀬 仁
西原 友樹
古久保 慶
B4
CROWN


大阪大学


サイバーメディアセンター
CYBERMEDIA CENTER


大阪大学
吹田キャンパス
千里門
Osaka University
Senri Gate of Suita Campus

Agenda

- I. Overview of Functions
- II. Priciple of P2P and PIAX
- III. System Design [Important]
  (Agent and Peer Desgn)
- IV. Recommendation System
- V. Questions & Comments

One can manage the pictures he/she has taken and the information in the P2P Internet will be updated synchronously.

Without Internet, the P2P network can be built based on wireless LAN.

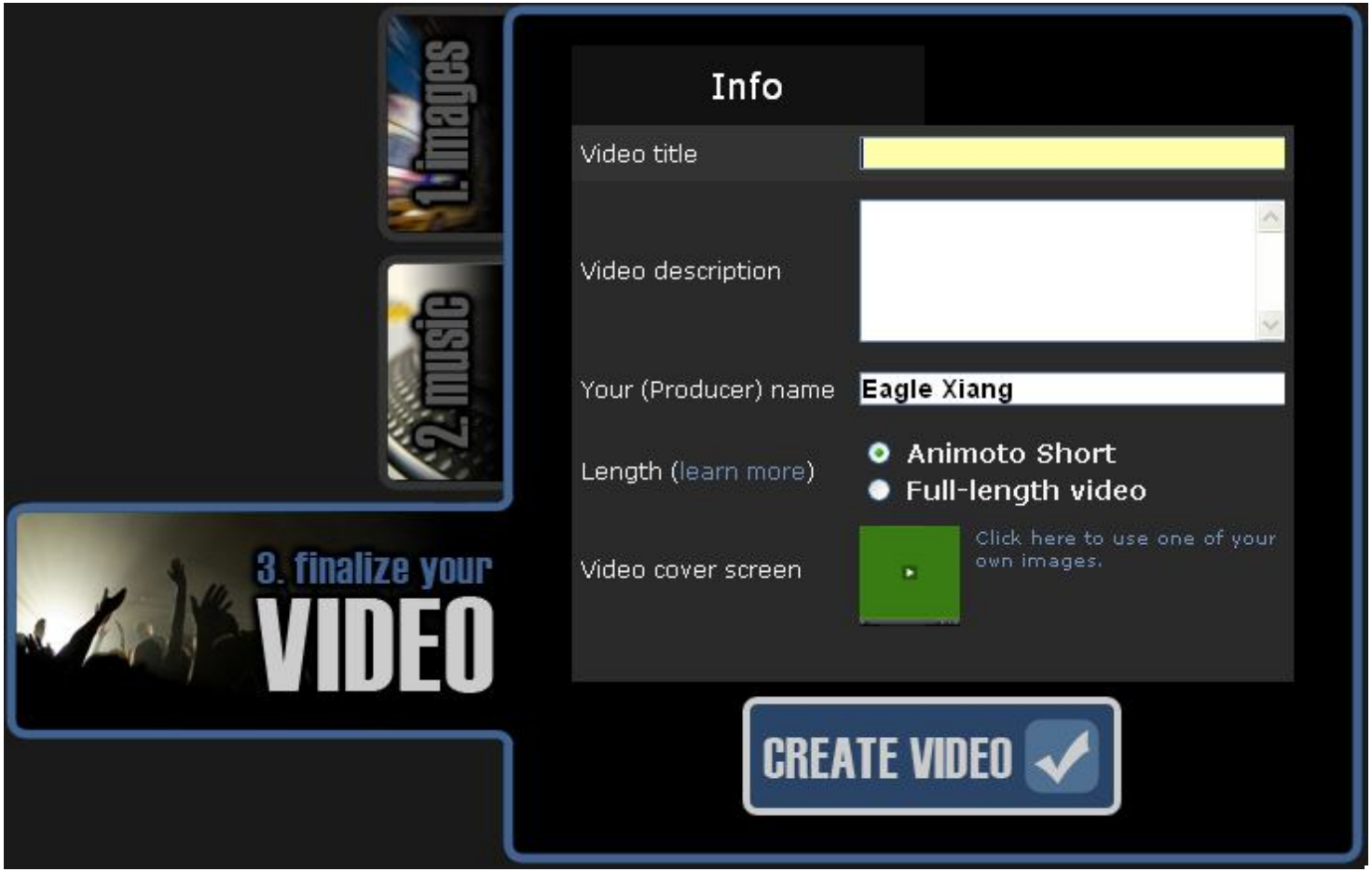
Share ubiquitous pictures with others.

One can Use a PIAX Nano or a cell phone to take pictures with location information attached at any time and can optionally add comments to them.

When a person is walking and there are pictures taken before at near locations, the application will notify him/her.

It may looks like: (from ANIMOTO)

Although for this project, how it looks like is not important



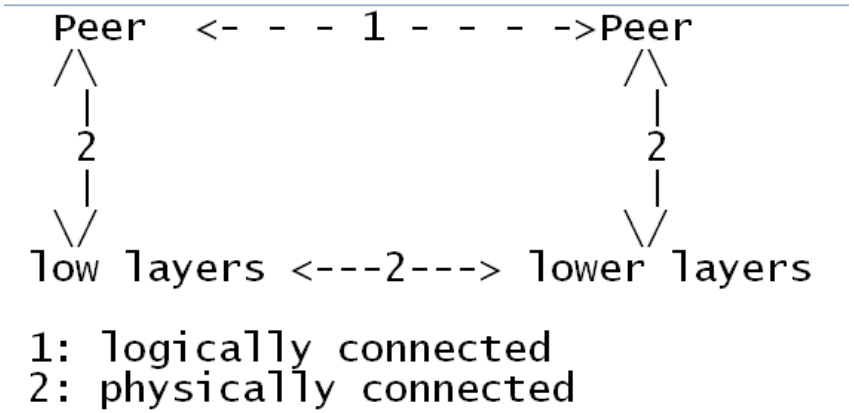
Take Pictures → **Publish Pictures** → **Add Comments**

Agenda

- I. Overview of Functions
- II. Priciple of P2P and PIAX
- III. System Design
  (Agent and Peer Desgn)
- IV. Recommendation System
- V. Questions & Comments

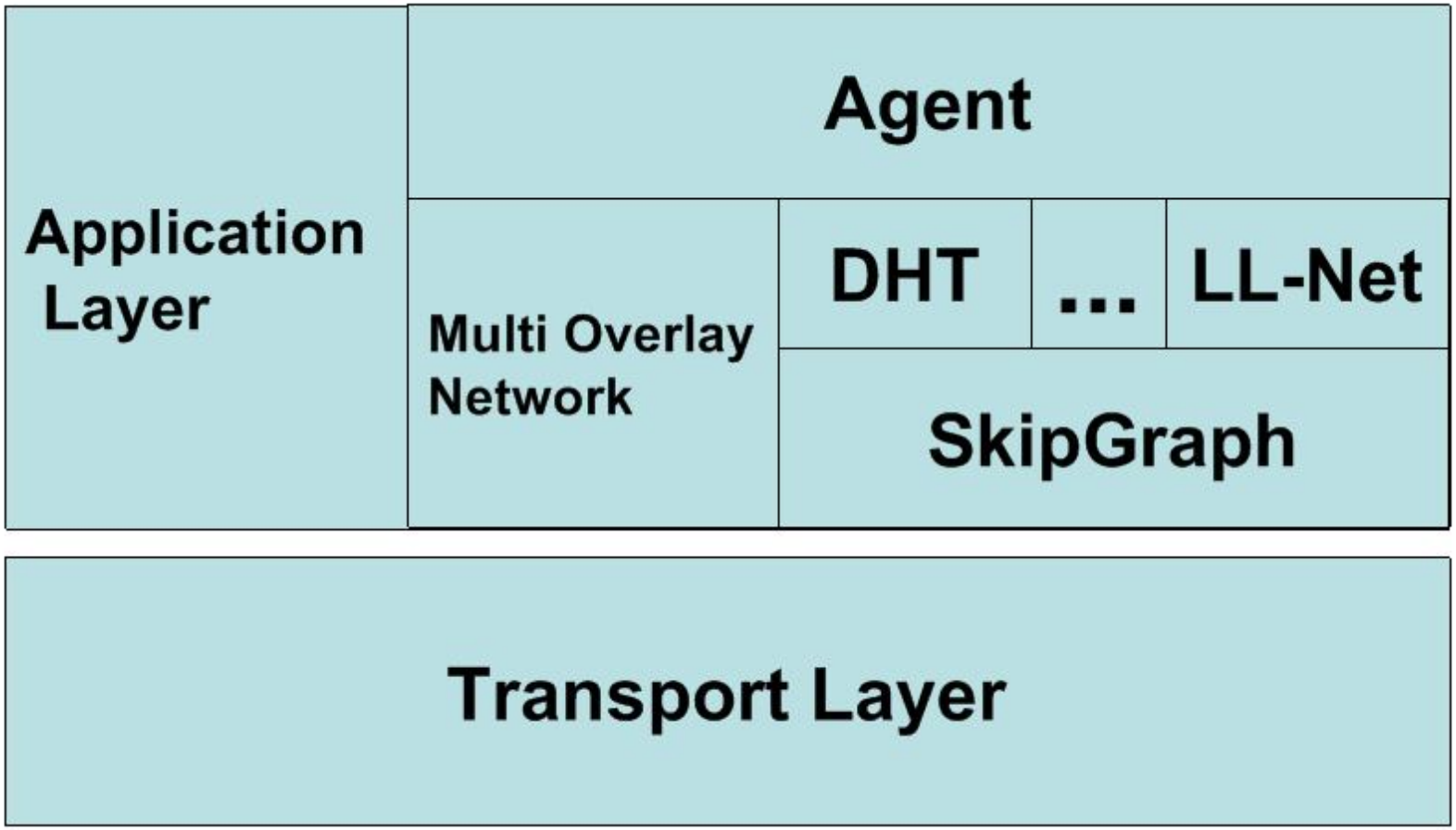
1. Overlay network
(my understanding)

- built on top of another network
- nodes are connected by virtual or logical links



- Overlay network means that an application exists inside one layer(now I talk about Five Layer Architecture), especially the application network. Since the communication between a peer and another is based on the layers below, this kind of network is called "Overlay".

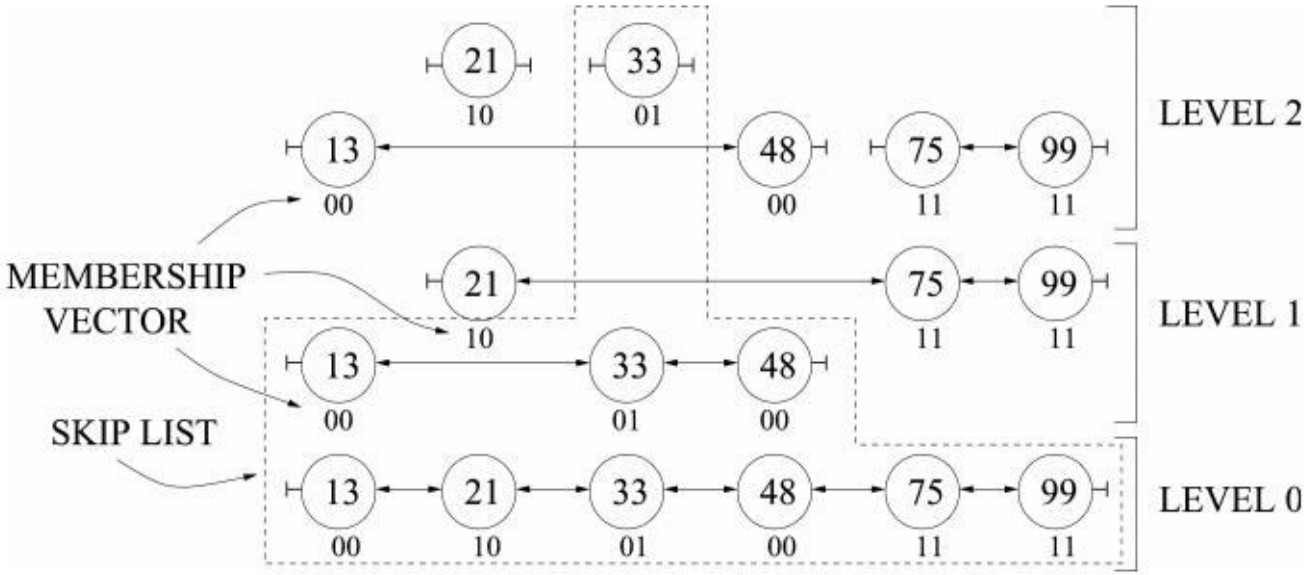
2. Multiple Overlay network



3. How do the DHT, LLNet, and ALM overlays use MSkipGraph?

- DHT and ALM search through SkipGraph according to Key, but LLNet does that according to peerId. The following is the data structure of a skip graph. The valid searching region is inside the dashed region.

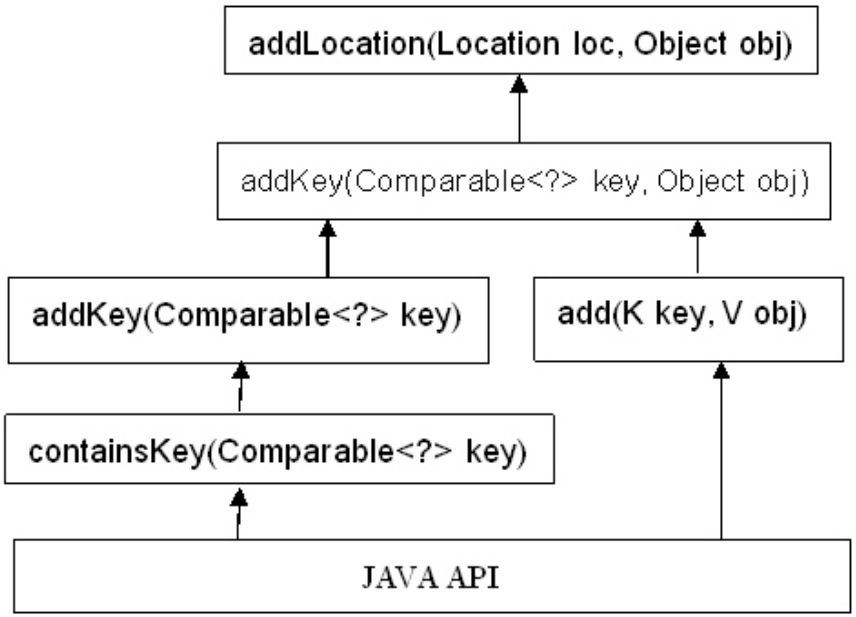
(1) In theory, for a DHT:

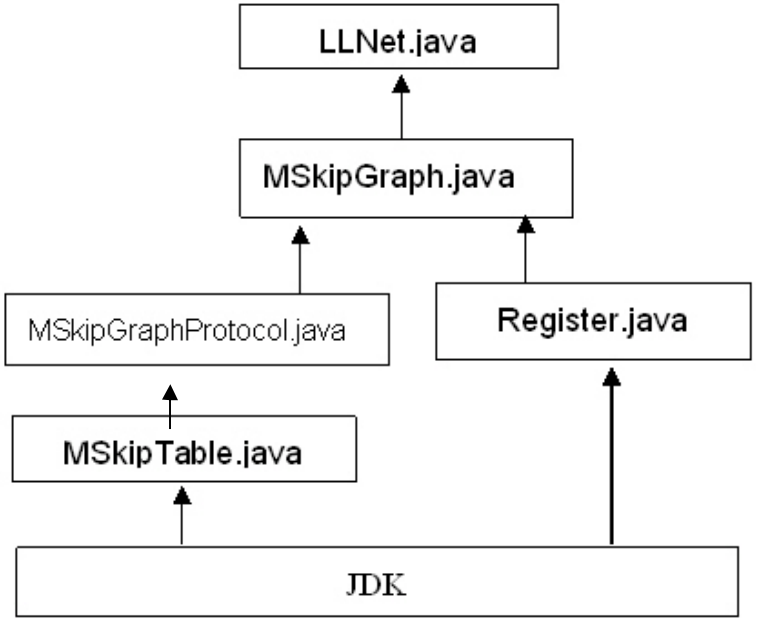


A skip graph with $n = 6$ nodes and $\lceil \log n \rceil = 3$ levels.

- (1)let's suppose that we want to search the peer with the key 75. We search from higher levels to lower levels, and from head (left) to tail (right). Hence, the key route of our searching will be: Head of Level 2 -> 33 in Level 2 -> 33 in Level 1 -> 48 in Level 1 -> 48 in Level 0 -> 75 in Level 0. Not until we find the Key 75, did we find the peer. Through that kind of searching, DHT uses the MSkipGraph. And in implementation, DHT implements MSkipGraphWrapper.

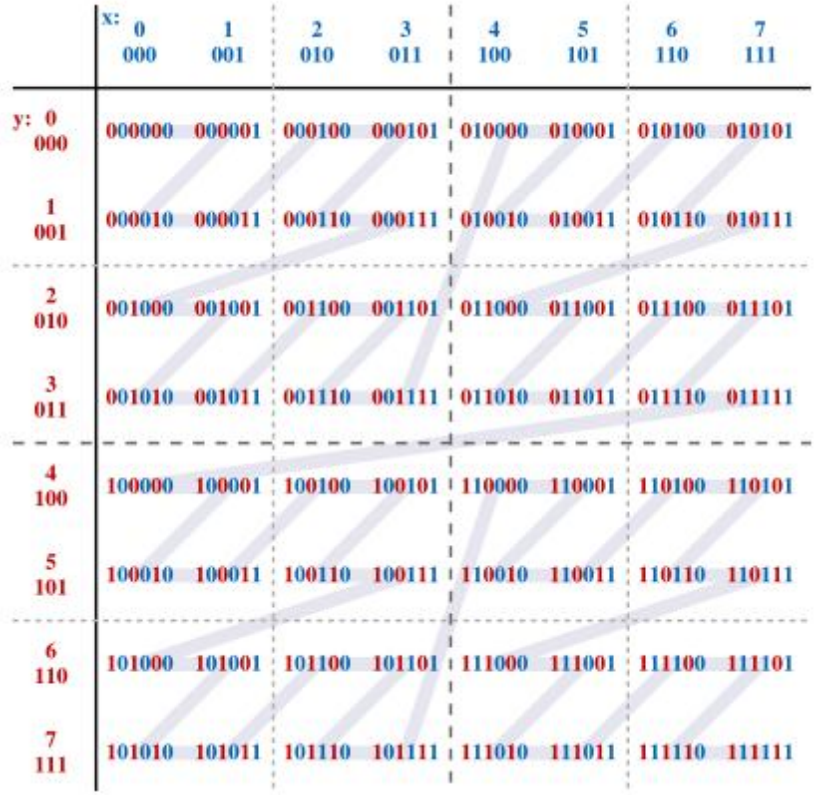
(2) In implementation, for a LL-Net:



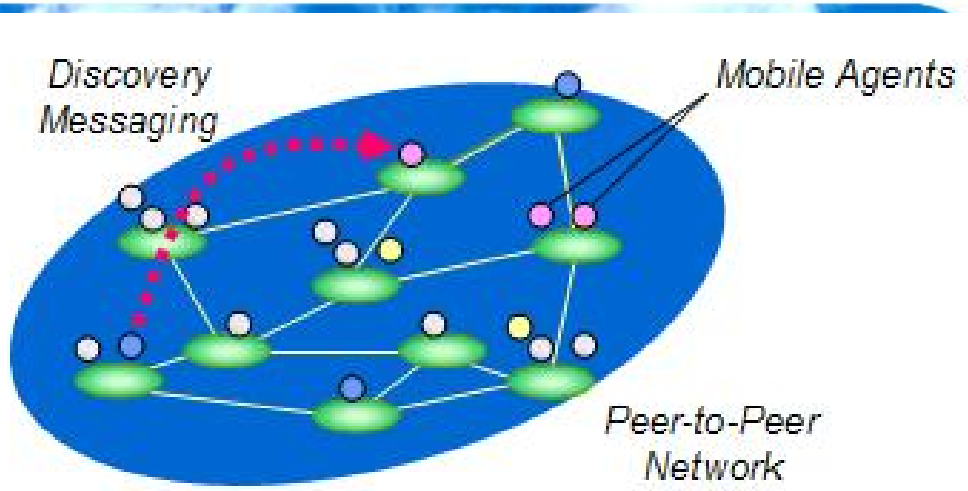


4. How to describe a peer's position?

- we can use a key to describe the accurate position of one peer. It's because the key is the combination of (x,y) according to the Z-sorting (Z-order) rules. And (x,y) stands for the unique position of a peer in the 2-dimension map (longitude, latitude).



5. PIAX principle
(my understanding)



- The P2P network needs seeds to allow external peers to connect into the network. Seeds is not necessarily be fixed, and can be flexible.
- In PIAX, peers use peerID or peerName to communicate with other peers. Agents exist on peers. A peer can have many agents, and a agent can move from one peer to another peer. That kind of agent is called mobile agent; the agent which cannot move is called persistent agent.

- We can use a Key to describe the unique position of a peer; and Key is stored in the multiple overlay networks, such as DHT, LL-Net, ALM. In fact, a peer(peer1) can have multiple positions, so it can have multiple keys. So then, if another peer(peer2) want to find peer1 by searching according to one key, then any key of peer1's multiple keys will be okay.

6. PIAX Programming Guide

- In PIAX, when we want to make a peer to join the existed P2P network, we should call the method join();
- If we want to change the position of this peer, we can call the method move(double x, double y);
- If we want to start the peer with a specific position, we can directly call the method join(double x, double y). In fact, that method is the combination of join() and move(double x, double y).
- If we want to know the information of this peer, we can call the method info(). Then we'll know the peerName, peerID, (host:port), location, [key and objects].

- Since Key can be accurate identity of a peer's position, then how can we know the value of a Key? In fact, we can use the method get(String keyName) and then it will return the value of that Key.Then how can we set a new Key or change the value of a Key? We can use the method put(String keyName, String keyValue).
- In order to communicate with other peers, only joining the P2P network is not enough. It also need s initialization. We can use init() to do that. In fact, init() calls the method put(String keyName, String keyValue) to set the peerID to be the value of its Key.

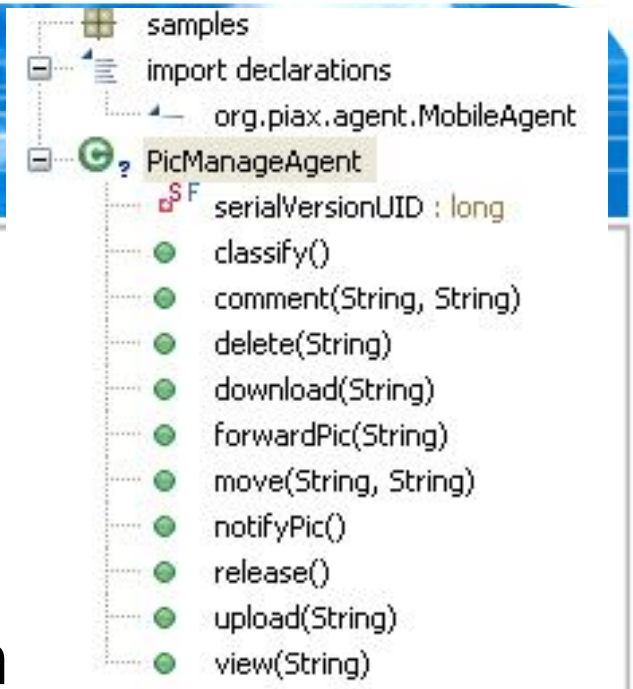
- If peer1 want to get to know another peer's (peer2's) information, it can call the method peek(String peer2_Name), as long as it knows the peerName of the peer2. The most important information peer1 will get is the peerID of peer2.

- If we want to set a new agent for a peer, we just need to call the method mkagent(String agentClassName, String agentName). There're several kinds of agents defined, including MobileAgent, PersistentAgent. In reality, we can define agent (or called design agent Class) by ourselves, such as EchoAgent, HelloAgent, IshiAgt, SleepyAgent and TravelAgent in the packet samples.

- After we set a new agent, we must want to check whether the agent has been added to the peer. In fact, we just need to call the method agents() for the peer, and then we can get a list of the agents existing on the peer.

- and so on.

Since I need to use the picture data with its location, I should use LL-net.



And the picture management class "PicManageAgent" should extend the class "Agent" and can have the methods showed in the right diagram:

- **First, what kind of query con applications or people will issue?**

  a. location-based search (pics taken in a specified region),

  b. time-based search (pics taken during specified time period),

  c. other attribute-based search (e.g, pics of person, animal pics, scenery pics, etc.)

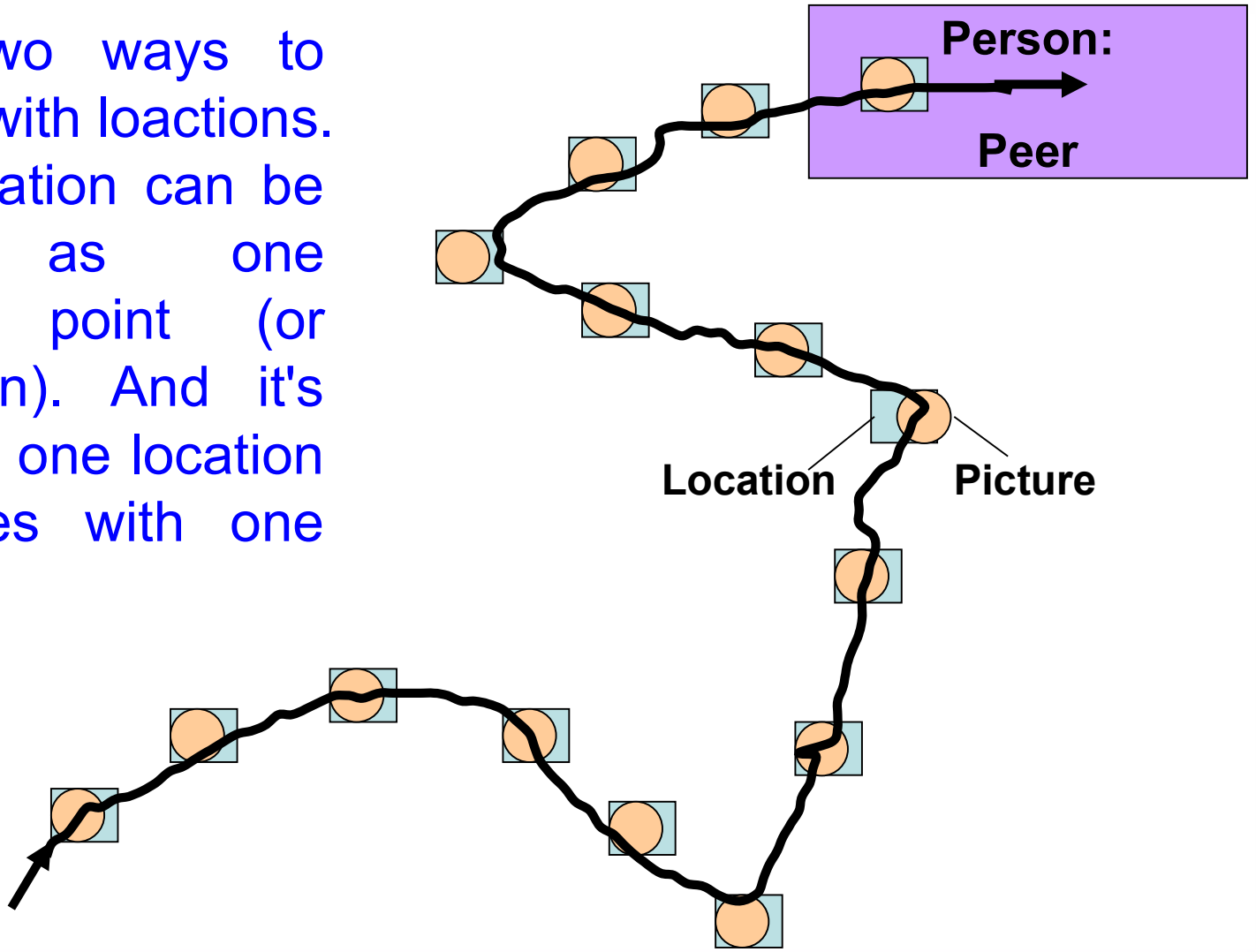
- **Then, give keys to pictures.**

Agenda

- I. Overview of Functions
- II. Priciple of P2P and PIAX
- III. System Design [Important]
  (Agent and Peer Desgn)
- IV. Recommendation System
- V. Questions & Comments

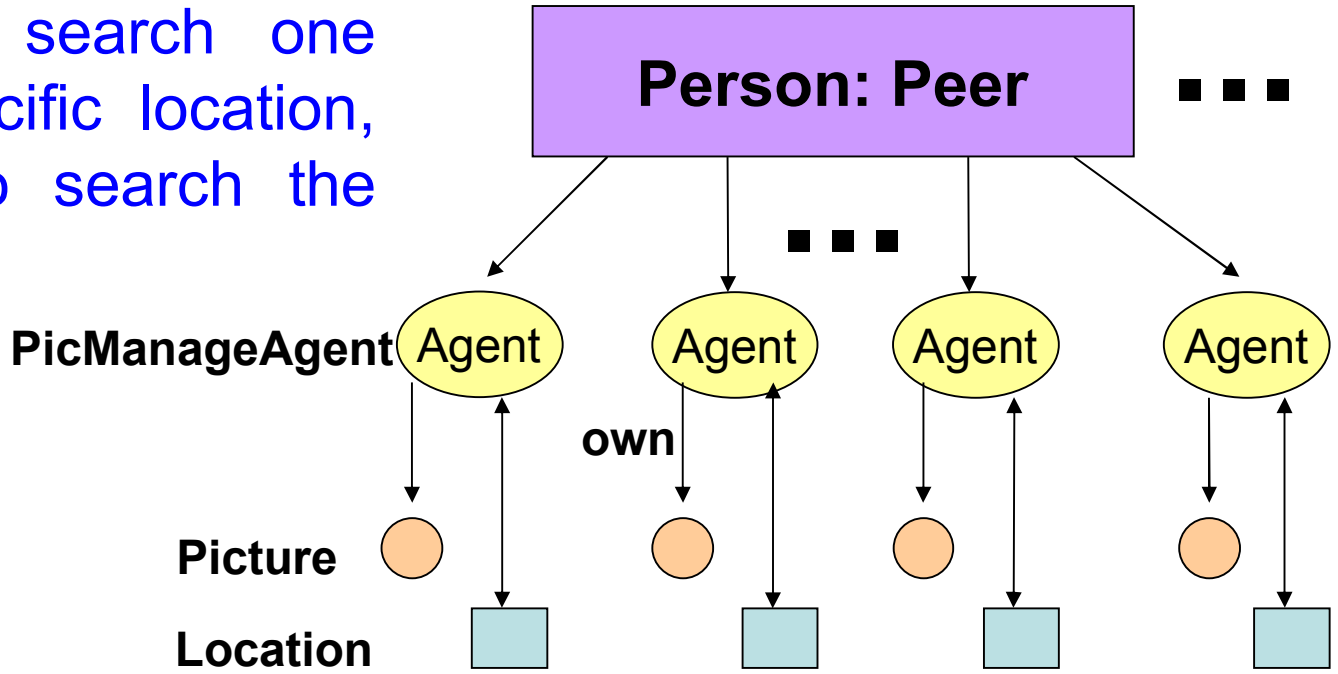
1. Application Scene of Peers (Location-based)

There are two ways to relate agents with loactions. Here, one location can be understood as one geographical point (or called position). And it's supposed that one location just associates with one picture.



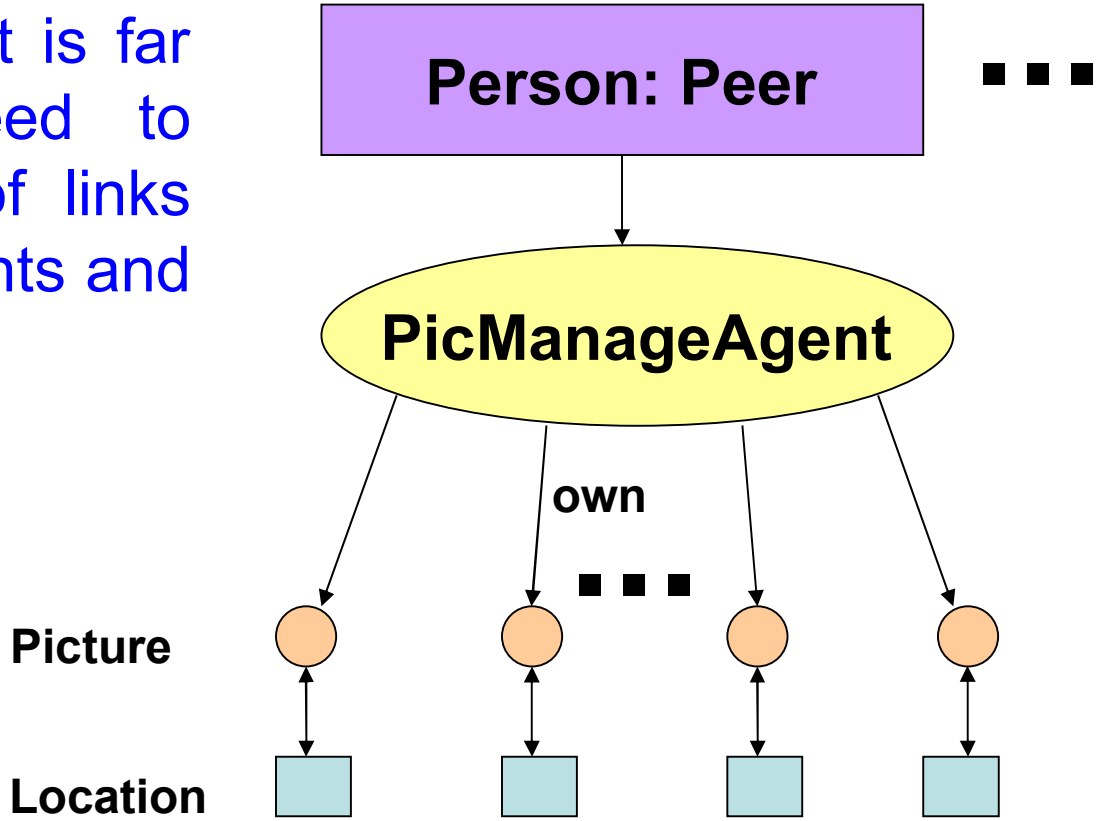
2. One Agent with One Picture

If we want to search one picture in a specific location, we just need to search the agent.



3. One Agent with Multiple Pictures

Just searching the agent is far from enough. We need to record the characters of links between PicManageAgents and pictures.



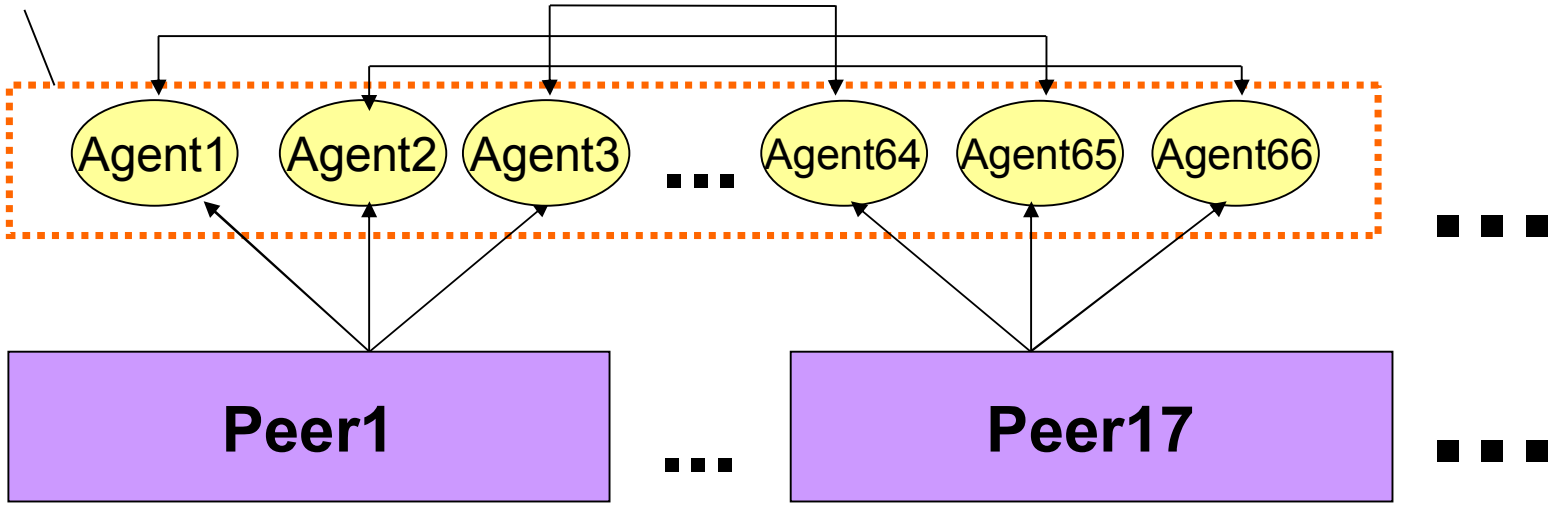
4. Pros and Cons of the Two Designs

	1agent with 1 picture	1 agent with multi-pictures
Advantage	Easy to search pictures	Hard to search pics
Disadvantage	Larger memory space (more agents; high degaree of agents, so too many links Between peers) [When the data is large-scale, this advantage should be considered]	Smaller memory space (just the opposite)

5. Pros and Cons of the Two Designs

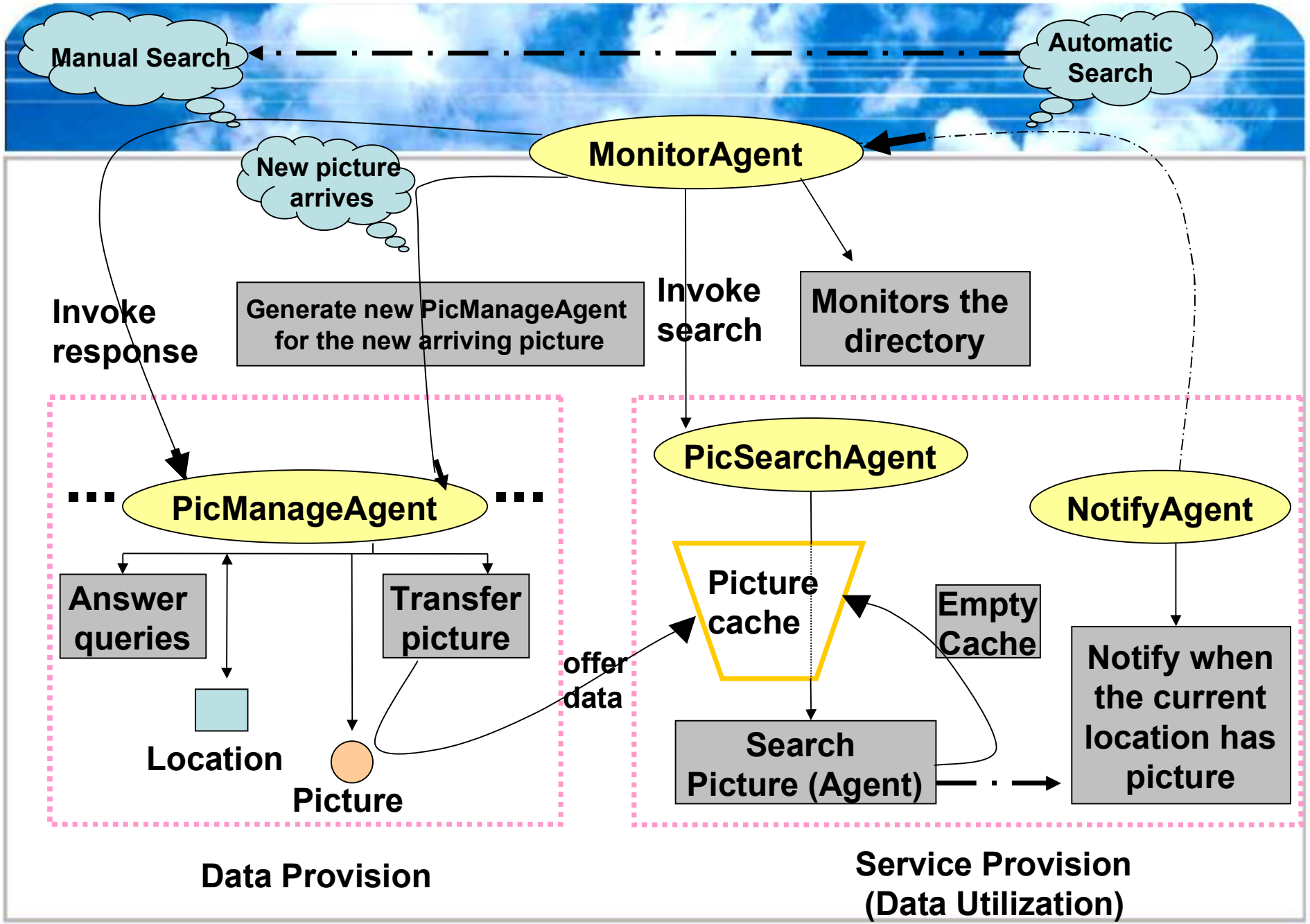
Links between peers

SkipGraph



In the perspective of Skip Graph, the query process requires to jump from one agent to another agent; thus, in the perspective of application layer, the query process requires to jump from one peer to another peer.

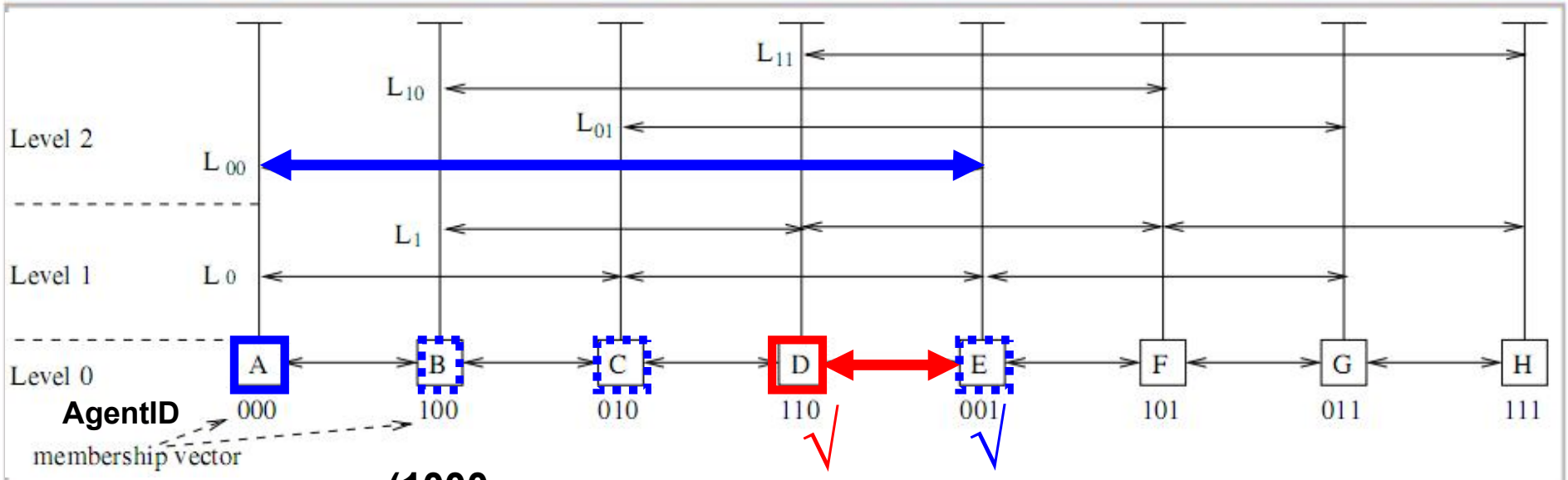
Then, in the 1 agent with 1 picture (location) design, one peer has many links with its agents. Now peer1 has 3 links with peer17. However, if one peer only has one agent (1 agents with multi-pictures), the links between peer1 and peer 17 is 1 at the most. Therefore, the 1 agent with 1 picture design will require larger memory space to store the agents and record the links.



6. Structure of the Peer

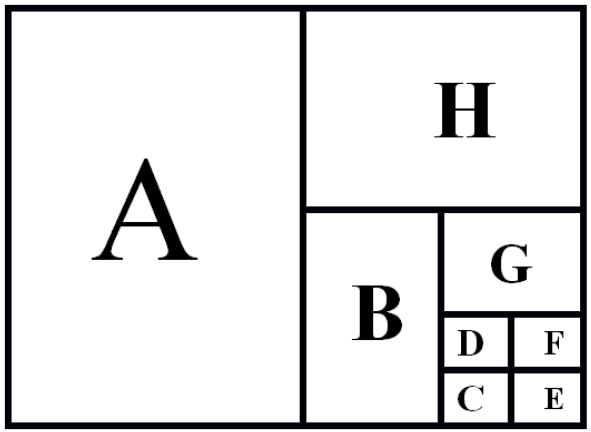
7. Process of Location-based Search (Z-Net, or called LL-Net in PIAX)

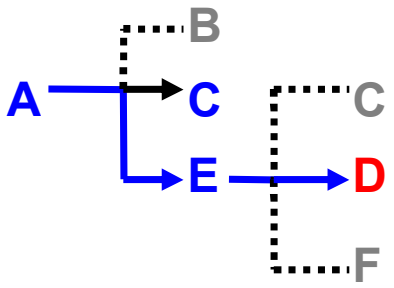
Suppose A receives a point query, whose destination is D.



Z-address (00,01) (1000, 1001) (101000) (101001) (101010) (101011) (1011) (11)

Z-level 1 2 3 3 3 3 2 1





8. Use PIAX API to Implement Location-based Search

Agent.java:

setAttrib(String name, Object value)

discoveryCall(...)

9. Work to do: Time-based and other attribute-based Search

Use special key to record shooting time and picture topic ?

(1) Picture Table:

AgentID	Location (Z-curve Key)	Picture Path	Shooting Time	Topic
B49b...e9	....	C:\P2P\1.jpg		
....				

(2) Agent Table:

AgentID	AgentName	HostPeer	AgentNo
...			

(3) Peer Table:

PeerId	PeerName	Host	Port	Location - X	Location - Y	Key

(4) Key Table:

Key	Object (stored in the position the key describes)

Agenda

- I. Overview of Functions
- II. Priciple of P2P and PIAX
- III. System Design [Important]
  (Agent and Peer Desgn)
- IV. Recommendation System
- V. Questions & Comments

New Challenge: recommendation system of pictures

- Since one agent manages one position for a user in the "1 agent with 1 picture design", one user can have at most 1 picture in one location. But many people may pass by that location, so then that position can be related with many pictures. When a new user passe by that location, which picture is the best one to be recommended to that user?

- That's a complex question. One position can be related with many pictures, but those pictures can have different shooting time (day or night, different seasons, latest or old), can represent different topics (happy or sad, etc.; people, animal, object or sights). So we need a recommendation system to choose one picture from a lot of pictures to suit the user's interest. Such as, if the user ofen publish (or called upload) pictures which are usually about sights, represent happiness, and are taken in the daytime. So the three characters can be used as constraints in query, then the query result will be fewer and more accurate, which will help the system to select one for the user. If after several selections, there are still several similar pictures, the system will choose one randomly. Of course, that is depended on how well the recommendation algorithm is.There are many kinds of recommedation (filtering) algorithms, such as Assocaition Rules, Content Filtering, Collaborative Filtering and so on.

Agenda

- I. Overview of Functions
- II. Priciple of P2P and PIAX
- III. System Design [Important]
  (Agent and Peer Desgn)
- IV. Recommendation System
- V. Questions & Comments

Questions / Comments

Thanks!

Please feel free to contact me:
E-mail: eglxiang@gmail.com
Website: http://eglxiang.blogspot.com MSN: x6x@sohu.com
Mail Add: P.O.Box F26, Information Campus, Wuhan University, Wuhan, Hubei, 430079, P.R.China

@ China: International School of Software, Wuhan University
Tel: +86-138-7197-8695; Fax: +86-027-87519101
@ Japan: Graduate School of Engineering, Osaka University
Tel: +81-06-6879-8795 (Daytime); +81-72-6416900->626 (Night)

Xiang Xiang (項翔)

Oct. 30th, 2008