1. What is the definition of a p2p system given by the authors in sec 1?
Compare it with at least one of the definitions surveyed in the last paragraph of pg 2.

The definition of a p2p system given by the authors in section 1 is the following:

"The term "peer-to-peer" (P2P) refers to a class of systems and applications that employ distributed resources to perform a critical function in a decentralized manner. "

1.1 Comparison with the Intel definition
The Intel P2P working group defines it as "the sharing of computer resources and services by direct exchange between systems".

In the Intel definition, there is no reference of the location where the computation takes place. On the contrary, the authors' definition is more complete.

1.2 Comparison with Alex Weytsel's definition
Alex Weytsel's definition of Aberdeen defines P2P as "the use of devices on the internet periphery in a non client capacity".

In this definition there is no reference of the location of the resources, so it lacks of completeness, compared to the authors' definition.

1.3 Comparison with Ross Lee Graham's definition
Ross Lee Graham defines P2P through three key requirements:

a) they have an operational computer of server quality;
b) they have an addressing system independent of DNS; and
c) they are able to cope with variable connectivity

This definition has many contradictions, compared to the one given by the paper's authors. The assumption that there should be a server computer opposes the goal of decentralized computation. Also, this definition constraints the name resolution mechanism, while the authors do not pose such restrictions. Finally, the statement of variable connectivity is quite important and is not referred by the authors. Its importance stems from the fact that p2p systems mainly consist of heterogeneous elements. The authors did not seem to have qualified this fact.

1.4 Comparison with Clay Shirky's definition
Clay Shirky of O'Reilly and Associate uses the following definition: "P2P is a class of applications that takes advantage of resources – storage, cycles, content, human presence – available at the edges of the Internet. Because accessing these decentralized resources means operating in an environment of unstable connectivity and unpredictable IP addresses, P2P nodes must operate outside the DNS system and have significant or total autonomy from central servers"

Compared to the definition provided by the authors, this one seems to be almost complete: it does mention most of the authors statements, such as decentralized resources, and it also mentions the problem of connectivity. Moreover, it implies that the computation should be decentralized.

1.5 Comparison with Kindberg's definition

Kindberg defines P2P systems as those with independent lifetimes.

This definition being loose, mainly focuses on p2p connectivity. However it is not stated where the resources and the computation are located.

2. In Fig 2 (pg 3), the authors compare some aspects of the client-server and the p2p computing models. List and explain these aspects.

The aspects compared are the following:

Client - Server	p2p	Explanation
Managed	Self-organized	Client-server systems are centralized and need to be managed by an administrator. P2p systems are self organized
Configured	Ad-hoc	Client-server systems need to be locally configured. P2p systems take ad-hoc decisions i.e. load balancing
Lookup	Discover	In the Client – server model we have to use specific entities which location is well – known in order to locate a resource. In the p2p model this is not always necessary (e.g. flood variations)
Hierarchy	Mesh	Client server models follow a hierarchy since clients take resources from servers. This does not happen in p2p since the organization is usually decentralized and resources may come and go from any peer leading to a formation of a mesh.
Static	Mobile	Servers must be static (e.g reside in the same IP) in order to be located by the clients. In p2p resource sharing entities may change point of presence and be also located.
Server dependencies	Independent lifetime	Servers must have extended uptime in order to provide clients with resources. In p2p systems, nodes might enter or leave the system arbitrarily.
IP-centric	Also non-IP	In p2p systems different protocols may be used in order to facilitate services such as routing.
DNS-based	Custom naming	Client – server systems uses the DNS in order to locate naming. P2p systems mainly use custom naming methods in order to cope with special system properties, such as node independent lifetimes.
RPC	async	Client – server systems may employ RPCs, while this is not possible with p2p systems, in which arbitrary lifetimes require asynchronous communication.

3. What is a hierarchical and what is a flat client-server model?

The client-server model is called flat when all clients only communicate with a single server (possibly replicated for improved reliability).

It can be hierarchical when the servers of one level are acting as clients to higher level servers.

4. What is a super peer?

A SuperPeer is a peer node that contain some of the information that other peers may not have. Other peers typically lookup information at SuperPeers, if they cannot find it otherwise.

5. What is the difference between a compute-intensive and a componentized application? How does this relate to vertical and horizontal distribution?

Compute-intensive applications run the same task on many peers, while componentized applications run different components on each peer.

6. What is according to the authors the main challenge of communication in p2p?

The main challenge of communication in P2P community is to overcome the problems associated with the dynamic nature of peers. Peer groups frequently change either intentionally (e.g., because a user turns off her computer) or unintentionally (e.g., due to a, possibly dial-up, network link failing). Maintaining application-level connectivity in such an environment is one of the biggest challenges facing P2P developers.

7. What is the most common solution to reliability across p2p systems?

The most common solution to reliability across P2P systems is to take advantageof redundancy. For example, in case of compute intensive applications, upon a detection of a failure, the task can be restarted on other available machines. Alternatively, the same task can be initially assigned to multiple peers. In file sharing applications, data can be replicated across many peers. Finally, in messaging applications, lost messages can be resent or can be sent along multiple paths simultaneously.

8. What are the advantages/disadvantages of the centralized directory, the flooded requests, and the document routing models.

Model	Advantages	Disadvantages
Centralized Directory	1. Actual system proved to be strong and efficient	1.Limited Scalability 2.Requires Big central servers
Flooded Requests	1.Efficient in small company networks. 2. No need for centralized servers.	1. Requires a lot of network bandwidth 2. Not very scalable
Document Routing	1. Very efficient for large, global communities. 2. No need for centralized servers.	1. The document IDs must be known before 2. Quite difficult to implement a search 3. the community might split into independent sub-communities, that don't have links to each other

9. In the centralized directory approach, after the best peer is located, the file exchange occurs directly between it and the requesting peer. What are the advantages/disadvantages of this?

The advantages of the centralized directory approach are the following:

- If a resource exists in the system, then the requesting peer will find it.
- The requesting peer will find the best source, if many copies exist.
- Easy implementation.
- Efficient lookup

The disadvantages of the centralized directory approach are the following:

- Management for the central server is required.
- Scalability problems arise when the client number increases.
- Single point of failure.

10. What can be considered as a closure mechanism in Gnutella?

The fact that in Gnutella new nodes must know the address of another Gnutella node or use a host list with known IP addresses of other peers can be addressed as a closure mechanism.

11. What are the factors that affect scalability, give one example for each.

The factors that affect scalability are the following:

- The amount of centralized operations that needs to be performed, for example synchronization and coordination.
- The amount of state that needs to be maintained. The algorithms employed might be centralized, and need to gather system wide information.
- The inherent parallelism an application exhibits. Some operations may have small degree of parallelism, because some operations cannot be parellelized, e.g. centralized naming.
- The programming model that is used to represent the computation. Parallel languages might be used with message passing interfaces which do not scale well.

12. Given the ad-hoc nature of connectivity in p2p, comment on what type of (message-oriented) communication (i.e., synchronous/asynchronous, transient/persistent) would be more appropriate.

The more appropriate type of communication in a p2p environment would be transient asynchronous (UDP), since connectivity is intermittent and peers enter and leave the system in an arbitrary way.

13. pg 17, 1st column, last par
"The geographical distribution of the peers helps to reduce congestion on both peers and the network". Explain.

As geographical distribution of the peers is high, traffic is not directed towards a particular location, the load is balanced, as the requests do not burden a particular e.g. cluster but are scattered.

14. What is the goal of caching in p2p? What are the advantages/disadvantages of caching the reply at all nodes in the return path? Can you think of any alternatives? Is this possible in Gnutella?

The goal of caching in p2p is to reduce the path length required to fetch a file/object and therefore the number of messages exchanged between the peers.

Some of the advantages of caching in the return path, except for the reduced path length, are the following:

- Load balancing, since an object can be found at more than one locations.
- Higher availability, since objects can be located by more nodes in the network, e.g. in the case of a limited search horizon technique.
- Locality of data, since when a node needs to retrieve an object, this can be available by the node's neighborhood.

Some of the disadvantages are the following:

- Consumption of nodes' bandwidth in the return path without they having requested a download, e.g. especially for large files.
- File versioning problem created by cache coherence problem.
- Security: download of unwanted material (harmful offending, illegal, etc.).
- Higher latency in the first time, if the object to be cached, is not multicasted to the path nodes but sent individually from one peer to another along the search path.

Instead of caching the reply at all nodes in the return path, the following alternatives can be used:

- Caching the reply every n nodes in the return path
- Caching the reply only to nodes that their local policy allows caching of alien files.
- Caching only at dedicated peers the number and availability of which is high.
- Aggressive caching, where a node pushes its contents either to its immediate neighbors, or to all possible paths of length l or to i of the possible paths of length l.

This strategy could be possible in Gnutella, if a copy of the file is not transmitted to the requesting peer directly, but also following back the search route. In order to achieve this, each peer should keep a table of the incoming, forwarding messages, so that the search path can be retrieved.

15. What does the "power-law distribution of the p2p network" (pg 17) mean?

This term means that p2p networks have power-law link distributions, containing a few nodes that have a very high degree and many with low degree.

16. Compare/relate the definition of distributed systems in sec 5.2 (pg 21) with sec 1.4 of the textbook.

The textbook's definition is the following:

Distributed systems hide the intricacies and heterogeneous nature of the underlying hardware by providing a virtual machine on which applications can be easily executed

The definition provided by the paper is the following:

Distributed computing is addressed in a community of machines, focusing on the delegation or migration of computing tasks from machine to machine.
This definition implies that all the machines should have the same execution environment. The idea of a virtual machine also contains this fact but also demands, that the user does not know where the task is executed. Delegation and migration have to do with the fact that data and/or execution move from one system to another. Summing up, the two definitions are related if we exclude the fact that in the definition provided by the paper it is not stated that the user should address the distributed system as if it was a single system.

I will also use the definition of grid computing for comparison with the textbook's definition:

A computing grid can be seen and used as a single, transparent computer. A user logs in, starts jobs, moves files, and receives results in a standard way.
It is easy to see that these definitions are almost equal, if we exclude the fact that in grid computing we use high-end machines. Both definitions focus on the attempt to hide heterogeneity.

17. Why is the fault tolerance problem a greater challenge in collaborative p2p systems than in file sharing p2p systems?

Because in collaborative systems message ordering may be important and the group communication techniques are not suitable for P2P applications since such strict guarantees are not required.