

# Survey of Overlay Multicast Technology

Zhi Li and Yongjoo Shin

2nd June 2002

{lizhi, Shinyj}@cs.ucdavis.edu

## Abstract

IP Multicast is a technique that can transmit one copy of data traffic to multiple receivers at one time. Because it can greatly save the network bandwidth consumption as well as many applications are inherently multicast-based, it has been a research focus since the idea was proposed. However, because of many reasons, such as management, security, inter-domain routing, etc, it has not been widely deployed.

Recently, many researchers have put their research focus on Overlay Multicast. In Overlay Multicast, the data replication, multicast routing, group management, and other functions are all achieved at application layer. Because it does not require to change the current Internet infrastructure, it can easily be deployed. Since the idea was proposed, many overlay multicast ideas have been proposed. They differ in multicast tree formation, underlayer support, etc.

In this paper, we did a systematic research on available overlay multicast technologies. We classify them based on their different characteristics and present the different ideas of these protocols.

## 1 Introduction

Many Internet applications, such as video-conference, tele-education, require the underlayer network to support multicast communications.

Though IP-based Multicast technique has been proposed for more than 10 years. Many problems are still plaguing the wide deployment of multicast[17]: 1) per-group state maintaining 2) security problems; 3). Scalable address allocation; 4). reliability, congestion control, flow control problem; 5). slowly deployment. Until now, as far as we can find, only two ISPs; Sprint and UUnet have commercial multicast offer.

According to the end-to-end arguments[32]: Unless implementing some functions at lower layer can achieve large performance benefit that outweighs the cost of additional complexity at lower level, we should push the function to higher level as possible.

Based on this idea, recently, the research focus of mulitcast has been put on overlay multicast. In overlay mulitcast technique, the multicast group members are connected by an overlay multicast tree. All the multicast functions, such as membership management, data replication, are implemented at the end hosts. This method can easily address most of the problems of IP-based multicast. Because it does not require any modification to

the current Internet infrastructure, it can easily be deployed. Inktomi [28] has utilized this technique to provide multicast service to users. Their goal is to construct an infrastructure that can provide television-sized audience as well as television-like quality. Also, overlay has the following merits: adaptable, robust, customizable and standard.

However, overlay multicast technology is not as efficient as IP-based multicast. It will incur some delay and bandwidth penalties, less stability of multicast tree. Till now, many overlay multicast routing algorithms have been proposed to utilize overlay technique to provide scalable and high quality multicast service to applications. They differs in forming the multicast tree, maintaining multicast tree, and applicable applications.

The following are the possible multicast session characteristics:

- 1).Performance requirements: conference, high bandwidth low latency; file transfer, high bandwidth.
- 2).Gracely degradable: conference, tolerate loss; file no.
- 3).Session length: conference, short; file transfer, long.
- 4).Group characteristic: conference, small group; content delivery, large group.
- 5).Source transmission: conferencing, One source, fixed rate.

However, there has not been done any survey or summary work about these previous works about overlay multicast. The goal of this project is to provide a deep survey of these available protocols. In this paper, we first give a rough classification of available overlay multicast technique and then present the basic ideas about available overlay multicast protocols,

The paper is organized as follows. In section 2, we will give the classification of available multicast protocols. Then, section 3 will present the detailed ideas of these protocols one by one. Finally, we will draw our conclusion at section 4.

## **2 Classification Of Overlay Multicast**

As shown in Figure 1, we give the available overlay multicast techiques and their classification. In section 3, we will discuss the details of these protocols according to this classification.

Currently, there are two branches of overlay multicast techniques: fixed nodes based overlay and dynamic nodes based overlay.

In the fixed nodes based multicast, it first strategically places some nodes around the whole Internet. Then, according to the applications' requirement, these nodes autonomously form overlay multicast trees to provide multicast service. There are advantages and disadvantages in this solution. Because it uses the fixed nodes, the multicast tree is stable and can easily provide QoS service to applications. However, the multicast service is not flexible, and still needs ISP support. Besides those, the fixed nodes can easily become the bottleneck. The available proposed ideas include[9, 19, 25, 18, 11, 5].

In dynamic nodes-based overlay multicast, the group members are self-organized into an overlay multicast tree. The data duplication, multicast data forwarding, and group members management and other functions are all achieved at group members. Because in large multicast groups, there will happen frequent joining or leaving events happens, as well the unexpected network situation, How to adapt to this unexpected change is one of the main issue we should consider. How to scalably form an efficient multicast tree is another important issue.

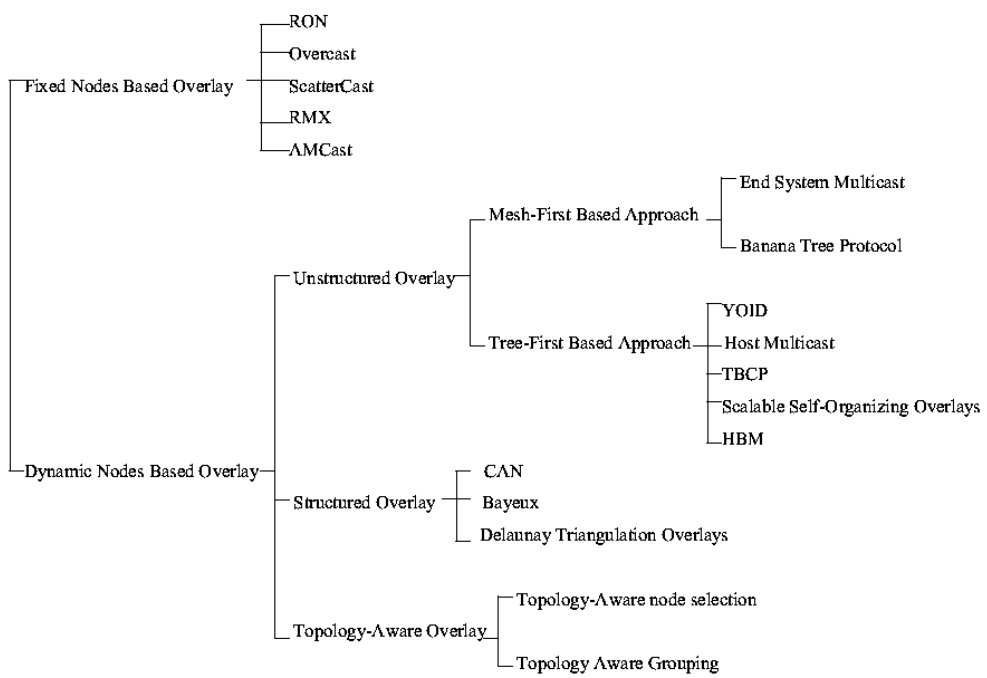


Figure 1: Classification of Overlay Multicast

Dynamic nodes based Overlay multicast can be either structured overlay technique or unstructured overlay. In structured overlay, the nodes can be interconnected together at the application layer in some well-defined manner. The protocols, such as ALMI, End-system multicast, Yoid [13, 3, 12, 2, 8] belong to this kind of dynamic nodes based overlay multicast technique.

For unstructured overlay multicast, some protocols mainly focus on building topology-aware overlay multicast tree, such as [10, 19].

Most overlay multicast routing protocols do not have detailed underlying topology information, they just use the measurement methods to get the relative distance between nodes.

In tree-first approach: members directly construct an overlay tree topology for data delivery, and additional control links are monitored and maintained to allow quick recovery from member failures. Such as Yoid[8], ALMI[13] are tree-first approach.

In mesh-first approach: members distributedly construct a mesh. Each member then participates in a route protocol on the mesh topology, and generates a source-specific tree to all other members based on the mesh topology, such as Narada[3], and Scattercast[6].

In centralized approach: they use some fixed nodes to control the membership information of a whole multicast group, helping the group members to form an efficient overlay multicast tree[13].

In Hierarchical approach, the members form hierarchical structure which can achieve good scalability[31][22]. It can greatly decrease the multicast tree control traffic as well as effectively absorb most of dynamic characteristics.

### **3 Miscellaneous Overlay Multicast Routing Protocols**

#### **3.1 Fixed Nodes Based Overlay**

This approach utilizes some fixed nodes to provide the multicast service to applications. These fixed nodes are strategically placed around Internet. Usually, one domain has one or more nodes. The multicast receivers (group members) connect to these fixed nodes to send or receive multicast traffic. RON[18] was proposed to quickly detect and recover path outages using overlay technique. Though, it does not deal with overlay multicast directly, it provides good directions for afterward overlay research work. Overcast[11] is designed to provide bandwidth sensitive multicast service as well as utilize the network bandwidth efficiently. In ScatterCast[6], the authors focus on how to provide scalable multicast service with heterogeneous receivers. It uses the application knowledge to meet heterogeneous receivers's requirement. In [25, 1], the author mainly focuses on how to set up efficient multicast tree with degree limitation as well as delay limitation.

#### **RON (Resilient Overlay Network)**

RON[18]'s main goal is to allow applications to detect and recover path outages and degraded performance quickly and integrate routing and path selection with application more tightly. Network failures can be categorized as link failure and path failure. BGP takes a long time (several minutes) to converge to a new valid route and it is incapable of expressing fine-grained policies aimed at users or hosts. RONs are well-suited to providing

fine-grained policy routing. RON can improve the reliability of Internet packet delivery by detecting and recovering from outages and path failures more quickly than current inter-domain routing protocols. A RON works by deploying nodes in different Internet routing domains. Each node of RON monitor the quality of the underlying Internet between themselves and use this information to route packets according to application-specified routing metrics either via a direct Internet path or via other RON nodes.

### **Overcast**

Same as RON, Overcast [11] is implemented by some nodes at strategic locations in existing networks. Its main goal is wide-area content distribution. It is designed for bandwidth intensive content and long-running content to be offered to vast number of nodes (large groups). The main focus is to build distribution trees that maximize each node's bandwidth from the source and utilize the substrate network topology effectively. The internal overlay nodes have permanent storage to facilitate the asynchronous time requirement of multicast content.

The multicast tree building process works as follows. When a new member wants to join a multicast group, it first contacts with the root of the available multicast tree. The root node becomes the current node. Then, after checking the current node and as its children nodes, the new member picks one that can provide the maximal throughput from the root to itself. A node also can periodically reevaluate its position in the tree by measuring the bandwidth to its current siblings, parents and grandparents. To avoid loop, each node also maintains the list of all the children under itself in the hierarchy (root path).

### **ScatterCast**

For ScatterCast [6], it relies on a collection of strategically placed ScatterCast Proxies (SCXs) that collaboratively provides the multicast service for a session. It partitions a heterogeneous set of session participants into disjoint groups at different locations. Each group is serviced by a strategically located SCX. When joining a multicast group, the client can first locate a nearby SCX and tap into the session via the SCX (by unicast or local area multicast). The protocol Gossmer is used by SCXs to locate each other in a decentralized manner and self-configure themselves into an adaptive and efficient overlay mesh of unicast interconnections. SCXs run a variant of a distance-vector routing protocol on top of this mesh structure and effectively build reverse-shortest-path distribution tree. Using SCXs, Scattercast can use application semantics to adaptively modify the content in order to suit the needs of the clients. This makes it very suit for the multicast sessions with heterogeneous requirement. In order to deal with partition, it uses selection algorithm to select a heartbeat generator. It sends heartbeat messages along the mesh periodically. If a node can not get heartbeat message for a long time, it will re-contact with the heartbeat and reconnect the mesh again. To improve the multicast tree efficiency, SCXs periodically probe with each other to evaluate the usefulness of adding new edges or deleting edges.

### **RMX**

In RMX [5], the authors describe a framework for providing reliable multi-point communication based ScatterCast architecture. Its focus is real-time reliable multicast. Its main goal is to reconcile the heterogeneous capabilities and network connections of various clients with the need for reliability. RMX introduces the notion

of semantic reliability. Each receiver defines its own level of reliability and decides how and to what degree individual data objects might be transformed and compressed. RMX builds on top of Scattercast by integrating application-specific intelligence and semantics into forwarding service.

RMX assumes that there is a protocol that can help the co-located receivers with similar network characteristics can self-organize into data groups and strategically placing RMXs to build overlay network across data groups. The notion of semantic reliability and ALF enable application-specific adaptation and transformation of the data that arrives into an RMX before it is forwarded into other links.

## AMcast

In [25], the author proposed AMcast, which uses a set of distributed Multicast Service Nodes (MSN) is used to provide multicast services in overlay multicast networks. The paper [1] focuses on optimizing the access bandwidth of the MSN's interfaces and end-to-end delay by using the appropriate routing algorithms in order to maximize the overlay sessions that can be served. Several algorithms are proposed to optimize the overlay multicast service:

1). Compact Tree Algorithm is a heuristic algorithm to find a minimum diameter, degree-limited spanning tree. The greedy algorithm builds a spanning tree incrementally. Suppose  $d(v)$  is the length of the longest path for  $v$  to other nodes in the partial tree. For each  $v$  not in the tree so far, we add an edge  $e(u,v)$ .  $u$  is chosen to minimize  $d(v) = c(e) + d(u)$ . At each step, they select  $v$  with the smallest value of  $d(v)$  and add it and update the residual degree of  $e(u,v)$  to the tree.

2). Balanced Compact Tree is to find a tree with bounded diameter, degree-balanced spanning tree. Its goal is to maximize the minimum value of residual degree. At each step, it finds the  $M$  vertices that have the smallest values of  $d(v)$ . From this set, it selects the vertex  $v$  with  $e(v) = (u,v)$ , which maximize the smaller of residual degree of  $u$  and  $v$ .

3). Balanced Degree Allocation is to achieve the best-possible residual degree balance.  $k$  is the multicast session fan-out.  $d_a$  is a degree allocation function which meets the following properties: (1).the overall  $d_a(v)$  is  $2(k-1)$ ; (2). there is at least two vertices:  $d_a(v) = d_a(u) = 1$ . In a partial degree allocation, the overall  $d_a(v)$  is less than  $2(k-1)$ . It computes a degree allocation function that maximize the smallest residual degree as follows: for each  $v$ ,  $d_a(v) = 1$ . while all the value of  $d_a(v)$  is less than  $2(k-1)$ , it selects a vertex that maximize the the minimum value of residual degree and increment  $d_a(v)$ . When selecting an edge  $e(u,v)$ , there are several algorithms: (1). Closest Pair (CP) algorithm, selecting the closest  $u$  and  $v$ . ; (2). Compact Component Algorithm: selecting the pair that results in the smallest diameter component in the collection of components.

To satisfy the the bound diameter requirement, they use "loose degree allocation" (small increase of the degree) and allow the tree-building process to construct a suitable tree satisfying the degree limits imposed by the loose allocation. This will result in an iterative overlay multicast routing algorithms. The simulation shows that ICT algorithm, when combined with loosening procedure is more effective at producing small diameter trees than ICP and ICC. However, ICT's greater effectiveness comes with a cost of added complexity, as it iterates through each possible starting vertex in order to find the best tree.

## 3.2 Dynamic Nodes Based Multicast

According to whether the nodes of a multicast group are numbered in a well-defined manner at application layer, we will have unstructured overlay and structured overlay. Unstructured overlay multicast is the early research focus of overlay multicast. End System Multicast [3] and Yoid [8] are the first two overlay multicast techniques. The structured multicast, such as CAN [19] and Bayeux [9], are based on application content searching and retrieving which is widely utilized peer-to-peer file sharing. The main drawback of overlay multicast is that it is not as efficient as IP-based Multicast because multicast multiple branches may pass the same physical link. Recently, several works have been done to improve the efficiency of overlay multicast tree [23, 24]. Their idea is to try to make the overlay multicast tree congruent with the IP-multicast tree.

### 3.2.1 Unstructured Overlay

In dynamic nodes based overlay multicast, the frequent joining and leaving events, the host death, and the change of network situation all will make the overlay multicast tree in an extreme dynamic situation. So, how to construct and maintain the overlay multicast tree is one of the most important issues we need to consider when we construct an overlay multicast tree. According to above, the unstructured overlay can be categorized into the following approaches: mesh-first approach, tree-first approach, hierarchical approach and centralized approach.

In mesh-first approach, all the group members first form a mesh. Each group member should keep all the other group members (or partial) and they also need some method to avoid and detect mesh partition. Then, based on the mesh they formed, they use some available IP multicast routing methods to set up an multicast tree which connects the group members.

In contrast, for tree-first approach, the members directly set up an overlay multicast tree which connect all these members. Then, based on some mechanisms, they make the overlay multicast more robust to the dynamic network environment. However, the tree-first approach leads to complex mechanisms (such as loop detection and avoidance).

Some other protocols, such as [13, 12], they use a centralized point to control the overlay multicast tree. The centralized point control the membership and help the group members to set up an efficient multicast tree.

Most of the above protocols only support small size multicast group. When the size of multicast group increase, the control traffic management and multicast tree setup have scalable problems. To avoid this, in [31, 15], the authors propose to use hierarchical solutions to provide multicast service to large group size. In addition, the hierarchical structure also can absorb the dynamic details of group members, which make the overlay multicast more applicable.

## Mesh-First Based Approach

### End System Multicast

End-system multicast is one of the early proposed overlay multicast protocols [3]. It is aimed at constructing small and sparse group. Narada is the distributed protocol, by which the multicast group members self-organize into an overlay structure and self-improve using a distributed protocol in a dynamic, unpredictable and heterogeneous

Internet environment. It optimizes the efficiency of the overlay routing tree based on end-to-end measurement.

The overlay spanning tree construction needs two steps: setting up mesh and constructing shortest path tree based on the mesh. When a member joins a multicast group, it gets a partial list of group members from some well-know nodes. To overcome the dynamic network and group situation, each member needs to refresh the membership to other members along the mesh. To avoid mesh partition, each member keeps a list of dead members, periodically delete a member from the queue, with some probability, either ignore it, or add a new link to it. To improve the mesh quality, each member needs to probe other members periodically. If adding one link can gain great utility, then, add the link. The dropping should have the following two desirable properties: stability and partition avoidance. To avoid count-to-infinity problems during data delivery process, each member maintains the path that leads to each destination.

### **Banana Tree Protocol**

Banana Tree Protocol [21] is proposed to improve the performance of Narada. The main algorithm it used is Switching-Trees algorithms, which is used for building and improving the overlay mutlicast trees while obeying any given degree limits. The nodes switch its parents to reduce tree cost or latency. It can switch to its nearby sibling or grandparent. When a node wants to switch to a potential parent, it should make sure that switch occurs when 1). the potential parent is not simultaneously attempting to switch to another node; 2). the potential parent is still the node's sibling or grandparent. Experiments show that the protocols can improve the performance of Narada: the cost of overlay multicast tree is less than twice of the according IP multicast tree.

### **Tree-First Based Approach**

#### **Yoid**

Yoid [8] is a general overlay network based content distribution toolkit, which addressing applications as diverse as netnews, streaming broadcasts, and bulk email distribution. Its aim is also to connect the small multicast available islands and servers providing a rudimentary architecture for global multicast. It includes a full framework for overlay multicast implementation. It includes YTMP (Yoid Tree Management Protocol, tree/mesh creation and management), YDP (yoid distribution protocol, it controls the reliable transmission over the tree/mesh), YIDP(Yoid Identification Protocol, packet, sender and receiver identification ) and yTCP, yRTP, yMTCP, and yMRTP (Yoid Transport Layers). Its main goal is to address all the aspects of multi-peer transmissions: connectivity, flow-control, reliability, etc.

The core of Yoid is a topology management protocol: TYMP, which allows a group of hosts to dynamically auto-configure into two topologies: mesh and tree. The tree and the mesh are created separately. The tree is optimized for efficiency while the mesh is for robutness. The mesh also has other functions: discovery of tree partitions, distribution of content (when tree is partitioned), detection and notification of member unreachability, verification of content reception. Each member can have several mesh neighbors, which can avoid partition.

Yoid also uses a rendezvous host to bootstrap group members into multicast tree. It informs the new member about several current members as well .

YTMP is the protocol to build and maintain the shared tree and the mesh. YTMP uses tree-first approach. It also uses root-path to detect loop and avoidance.

Partition Discovery: If a member finds that it becomes a root, it periodically broadcasts an “I am the root” message over the mesh and informs the rendezvous that it is the root. Then, these nodes will join each other’s partitions.

### **Host Multicast**

Host Multicast mainly deals with the deployment of multicast [2]. It uses Host Multicast Tree Protocol (HMTP) connecting multicast islands and provide multicast service to those places where multicast is not available. For each island, one member host is selected as the Designated Member (DM). Different islands are connected by UDP tunnels between DMs. All the DMs run HMTP to self-organize into a bi-direction shared tree. A special node is assigned the root of the tree. Each group has a Host Multicast Rendezvous Point (HMRP). The new member can get the root address by querying HMRP. Then, starting from the root, It gets the children list of the root. Then, it picks the closest one as potential parent. Then, it repeats this step until the potential parent accepts it. The potential parent decides whether to accept a join request based on its policy, bandwidth, traffic load, etc. If the potential parent rejects it, it goes back up one level and resumes the search for parents. This process stops when it reaches a leaf node, or a node that is closer than all its neighbors. The member also can adapt to the changes in the network condition to switch the parents, detect and break routing loops(using root path), and recover from network partition.

### **TBCP**

TBCP [20] is a generic tree building protocol which is designed to build overlay spanning tree while reducing the convergence time given the restricted membership/topology information available. It is a tree-first, distributed overlay spanning tree building protocol whose strategy is to place members in a near optimal position at joining time. Either the new node or the candidate’s children nodes can be redirected to its child node. In this protocol, it assumes that each node has a fixed number of children it wants to serve: fan-out of the entity. When joining a multicast group, the new member repeatedly contacts the candidate parent nodes from the root. The candidate parent node then replies the new member of its children nodes list. Based on some cost function, the new node evaluates the parent node as well as the children nodes to select an joining point. To improve the efficiency of the tree, it uses the concept of domain, each domain has its own root node. When a new member wants to join one group, the join request is first redirected to its domain root.

### **Hierarchical Approach**

#### **NICE**

NICE[31] was designed to scale overlay multicast to large groups. It is based upon a hierarchical clustering of the application-layer multicast peers and can be used to produce a number of different data delivery trees with specific properties. NICE constructs a hierarchically-connected control topology which connects all the group members. The data delivery path also will follow this structure and no additional computations are needed. most

members are in the bottom of the hierarchy and maintain only state of a constant number of other members. The members at the very top of the hierarchy maintain state of  $O(\log N)$  other members.

The lowest level of the hierarchy is layer 0. In each layer, the hosts are partitioned into a set of clusters. Each cluster has a leader, which has the minimum distance to all other host in this cluster. The cluster leaders of all the clusters in layer  $L_i$  join layer  $L_{i+1}$ . The host hierarchy can be used to define different overlay structures for the control and data delivery paths. Because the size of cluster is fixed (between  $k$  and  $2*k$ ). Using this method, the protocol can effectively reduce the number of refresh messages and number of hosts every host need to keep.

It also assumes that there is a Rendezvous Point (RP) that is the leader of the highest layer of the hierarchy. When a new member joins, it first contacts the RP. The RP will reply it with the members in the highest layer. Then, the new member will select the closest member. Then, the closest member will inform the new member with the list of numbers in the  $l$  level lower. Using this method, the new member can iteratively uses this procedure to find its  $L_0$  cluster. The cluster maintenance is done by each member periodically sending HeartBeat message to peer members within the same cluster. The cluster leader's HeartBeat message also includes the members of higher level cluster. If a member wants to leave, it sends Remove message to all clusters to which it is joined. If the cluster leader leaves without notification, the remaining members within the cluster will independently select a new leader. The simulation results show that NICE can support wide-area size multicast group with lower overhead.

### **Scalable Self-Organizing Overlays**

The target of [22] is to form an overlay with tens of thousands of nodes. The key contribution is to apply the hierarchy to overlay management to achieve scalability without degrading the quality of the resulting overlay. Unlike NICE, It is based on two-level overlay. The topology is partitioned into clusters. Each cluster has a unique representative node called head node. It is based on two techniques: clustering and mesh management. Clustering deals with building the hierarchy dynamically by forming the clusters. Mesh management determines how the nodes at the same level are connected to each other; how the clusters are connected to each other; how the nodes are connected to each other. For mesh management, it extends the Narada to manage mesh. Simulation result shows that it can reduce the bandwidth requirement for control traffic and hierarchical overlays absorbs changes better than flat overlays.

### **Centralized Approach**

#### **ALMI**

ALMI is targeted to the multicast applications with small groups[13]. An ALMI session consists of a session controller and multiple session members. The session controller handles member registration and maintains the multicast tree. When a new member joins or leave one multicast group, it needs to contact the controller. Then, the controller returns a list of peering points from which the member should accept connection requests and the parent to which the new member should initiate a connection. It ensures the efficiency of the multicast tree by periodically calculating a minimum spanning tree based on the measurement updates received from the all the members and inform the new parent and children's ID. To collect measurements, the controller needs

each member to monitor a set of other members and report. Using this centralized control approach, it greatly simplifies the routing problem compared to the distributed approaches.

### **HBM (Host-Based Multicast)**

In Host-Based Multicast[12], it distinguishes Core members (CM) and non-core members(nonCM). The CMs form a distribution tree. It assumes that Rendezvous Point (RP) knows CMs and nonCMs and the distance between them. and it can compute the overlay multicast topology and informs the members.

In the paper[12], it uses three methods to keep robust: 1). add Redundant Virtual Links (RVL). if a member gets message both from tree and RVLs, it informs the RVL to stop. Otherwise, RVL becomes its parent node. 2). Fast Failure Discovery and Recovery. 3).Adaption: to have stable transit nodes while unstable ones are moved to the leaves of the topology. They proposed several topologies to connect all the members: bus, tree, ring, star, etc.

### **3.2.2 Structured Overlay**

Because the available overlay multicasts have the following shortcomings: 1). The group members need periodically announce its estimated distance from other nodes and every node maintains the state for each other node in the topology; 3). when topology changes, every node needs to learn the information quickly. Structured Overlay technique can effectively deal with those problems.

### **Bayeux:**

Bayeux is an architecture for Scalable and Fault-tolerant Wide-area Data Dissemination[9]. Its goal is for streaming multimedia applications with arbitrarily large receiver groups. Bayeux is an efficient, source-specific, explicit-join, and application-layer multicast system that provides scalability to large number of receivers with failure tolerance in routers and network links using a prefix-based routing scheme inherited from an existing application-level routing protocol called Tapestry. Bayeux also provides specific mechanisms to provide load-balancing across replicated root nodes as well as more efficient bandwidth consumption by clustering receivers by identifier.

It uses the natural hierarchy of Tapestry routing to forward packets while conserving bandwidth. In Tapestry, each node has names independent of their location and semantic properties, in the form of fixed-length bit-sequence. Each node has neighbor map, which incrementally route overlay messages to the destination ID digit by digit. The neighbor maps have multiple levels, where each level represents a matching suffix up to a digit position in the ID. When routing, the nth hop shares a suffix of at least length n with the destination ID. To find the next router, we look at the its (n+1)th level map, and look up the entry matching the value of the next digit in the destination ID, which will gives us the next hop address. This guarantees that it will take at most  $\log_b N$  logical hops to arrive at the destination (system has N nodes, using base b). In addition to this, Tapestry also provides a set of fault-tolerance mechanisms which allows routers to quickly route around link and node failures. Tapestry facilitates the multicast by forwarding packets according to suffixes of listener IDs. The node ID base defines the fan-out used in the multiplexing of data packets to different paths on each router.

To maintain the multicast tree, It utilizes dedicated servers in the network infrastructure to help construct more efficient data distribution trees.

### **Content-Addressable Multicast**

It is designed to support large group size without restricting the service model to a single source. It is based on Content Addressable Networks(CAN)[19].

In CAN, the nodes form a virtual d-dimensional Cartesian Coordinate space. Every node owns a portion of the total space. Based on this algorithm, we can eliminate the need for a multicast routing algorithm to construct distribution trees. The virtual coordinate space is used to store (key, value) pairs as follows: to store a pair (K1, V1), key k1 is deterministically mapped on to a point (x,y) using a uniform hash function. Then, the pair (K1, V1) is stored at (x,y). The nodes self-organize into an overlay network that represents this virtual coordinate space. Two nodes are neighbors if their coordinate span overlap along d-1 dimensions and abut along one dimension. using its neighbor coordinate set, a node routes a message toward its destination by simple greedy forwarding to the neighbor with coordinates closest to the destination coordinate. The CAN construction process is as follows: 1) first the new node must find a node already in the CAN; 2). Next, using the CAN routing mechanisms, it must find a nodes whose zone will be split; 3). Finally, the neighbors of the split zone must be notified so that routing can include the new node.

CAN-based Multicast has two steps: 1). The members of the group first form a group specific “mini” CAN. 2). Multicasting is achieved by flooding over the mini CAN. Firstly, Using a well-know hash function, the group address G is deterministically mapped onto a point, say (x,y). and the node one C (the whole CAN) that owns the point (x,y) serves as the bootstrap node as the construction of G\_g. This is done by repeating the usual CAN construction process with (x,y) as the bootstrap node. Every node only maintains the number of groups it belongs to which is independent of the number of traffic sources in the multicast group. Because all the members of group G belong to the associated CAN C\_g, the multicasting to G is achieved by flooding on the CAN C\_g.

### **Summary of Structured Overlay:**

Both Tapestry and Can rely on an embedding of the nodes in a virtual address space. The neighbors of the node in the space are selected such that the overall network is structured in the well-defined manner. However, Tapestry based multicast only supports single source based multicast. It uses an explicit protocol to set up and tear down a distribution tree for the source node to the current set of receiver nodes.

Delaunay Triangulation Overlays[30] is another kind of structured overlay. Its main idea is to assign each node with a logical coordinate in a plane. Then, based on this logical address space, it forms a overlay multicast tree which connect all the members. It shares the same merits as Bayeux and CAN: need no routing algorithm and scalable to large group size.

### 3.2.3 Topology-Aware Overlay

#### Topology-Aware node selection[23]

For overlay applications, the performance will be improved if the application-level connectivity between the nodes in this networks is congruent with the underlying IP-level topology. In this paper, the authors proposed BINNING scheme whereby nodes partition themselves into different bins so that the nodes within the same bin h are relatively close to each other. The strategy is simple (requiring less support), scalable (no global knowledge needed) and completely distributed. This solution requires a set of well-known Landmark machines spread across the Internet. Each node measures the distance (round-trip time) to this set of well- known landmarks and independently selects a particular bin based on these measurements. The range of possible latency values is divided into a number of levels . Then, it augments the landmark ordering of a node with a level vector: one level number corresponding to each landmark in the ordering. The nodes who have the same value of level vector are grouped into one bin. The paper applies this method into structures overlays, such as CAN, pastry, as well as unstructured overlays, such as end-system multicast, and server selection.

#### Topology Aware Grouping (TAG) [24]

It is for group with large number of members. Its goal is to exploit the underlying network topology information to build efficient overlay networks among multicast group members. This releases the member from exchanging end-to-end measurements. Each new member of a multicast session determines the path from the root of the session to itself, and uses path overlap information to partially traverse the overlay data delivery tree and determine its parent and children. It selects as a parent the destination whose shortest path from the source has maximal overlap with its own path, to reduce increase in number of hops over a unicast path, while satisfying loose bandwidth constraints. Each TAG node maintains the Family Table (FT), which includes the parent node and the children nodes as well as the path from root to them (spath). When a new member wants to join a group, it sends a join request to the root. Then, the root begins from itself to search for a node which have the maximal prefix of spath as the new member.

## 4 Summary and Conclusion

In this paper, we do a survey of available overlay multicast technique. Though overlay multicast protocols have many merits, we think that IP-based multicast is a good choce for applications that cannot tolerate delay, some loss, and require high throughput with upper-bound: audio-video conferencing and Internet games while overlay multicast approach is great for those applications that can tolerate delay and can not tolerate loss, such as email and file transfer.

Asdiscussed in previous sections, the available technique can be categorized into two parts: fixed nodes based overlay or dynamic nodes based overlay. They have their own pros and cons. Fixed nodes based multicast can provide stable multicast service to applications. However, it needs preset multicast service nodes to achieve this. Besides this, these preset nodes can easily become the bottlenecks. So, the research focus of fixed nodes based overlay is to evenly distre the multicast traffic among these fixed nodes.

For dynamic nodes based multicast, all the dynamic members self-organize into an overlay multicast tree which connects all the group members. Because of the dynamic network situation and the dynamic group membership situation, the multicast service based on this technique is not stable. How to form an efficient multicast tree and maintain the multicast tree are the research focus. The tree maintenance cost is usually very high to overcome the group membership management and avoid partition.

Most of the available work focuses on how to form a multicast tree as well as how to maintain the multicast tree. For small size multicast group, most of the available solutions give us good result. It can overcome most of the inherent problems of IP-based Multicast. However, for large size group, how to provide scalable multicast service using Overlay technique is a research issue which needs us to more works. Besides this, few of the work deals with QoS issue in multicast tree forming process. This is also another topic which can need more research.

In conclusion, overlay multicast is still new research direction of multicast. How to effectively utilize the overlay technique is a future research topic, such as the issues about security, QoS, multicast tree stability issue, application layer routing, etc. With the more and more users can feel the advantage of multicast, the days of fully deployment and utilization of multicast in Internet community will come soon.

## References

- [1] S. Shi, J. S. Turner, Routing in Overlay Multicast Networks, Infocom 2002
- [2] B. Zhang, S. Jamin, L. Zhang, Host Multicast: A Framework for Delivering Multicast to End Users, Infocom 2002
- [3] Y.H. Chu, S. G. Rao, and H. Zhang, A Case for End System Multicast, SIGMETRICS 2000
- [4] Y.H.Chu, S.G. Rao, S. Seshan and H.Zhang, Enabling Conferencing Applications on the Internet Using an Overlay Multicast Architecture, SigComm 2001
- [5] Y. Chawathe, S. Maccanne, E. A. Brewer, RMX: Reliable Multicast for Heterogeneous Networks , Infocom 2000
- [6] Y. Chawathe, S. Maccanne, E. A. Brewer, An Architecture for Internet Content Distribution as an Infrastructure Service, unpublished paper, 2000.
- [7] S. Banerjee, B. Bhattacharjee, S. Partheasarathy, A Protocol for Scalable Application Layer Multicast <http://www.cs.umd.edu/projects/nice/talks/ccw-01.pdf>
- [8] P. Francis, Yoid: Extending the Internet Multicast Architecture, Unpublished at <http://www.aciri.org/yoid/docs/index.htm>, Apr. 2000
- [9] S. Q. Zhuang, B. Y. Zhao, A. D. Joseph, R. H. Katz, J. D. Kubiatowicz, Bayeux: An Architecture for scalable and Fault-tolerant Wide-area Data Dissemination, NOSSDAV 2001, June 2001

- [10] B. Y. Zhao, J. D. Kubiatowicz, A. D. Joseph, Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing.
- [11] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaasheok, J. W. O. Overcast: Reliable Multicasting with an Overlay Network, In Proc. 4th USENIX OSDI, October 2000
- [12] Vi. Roca and A. El-Sayed, A Host-Based Multicast (HBM) Solution for Group Communications, 1st IEEE International Conference on Networking (ICN'01), Colmar, France, July 2001
- [13] D. Pendarakis, S. Shi, D. Verma, M. Waldvogel, ALMI: An Application Level Multicast Infrastructure, In Proc. of 3rd Usenix Symposium on Internet Technologies and Systems, March, 2001
- [14] J. Liebeherr, T. K. Beam, HyperCast: A Protocol for Maintaining Multicast Group Members in a Logical Hypercube Topology, University of Virginia, Technical CS-2001-26, Nov. 2001
- [15] S. Jain, R. Mahajan, D. Wetherall, and G. Borriello, Scalable Self-organizing Overlays, <http://www.cs.washington.edu/homes/sushjain/overlays.html>
- [16] J. Liebeherr, M. Nahas, and W. Si, Application-layer multicasting with delaunay, [www.cs.virginia.edu/~jorg/archive/papers/CS-01-26.pdf](http://www.cs.virginia.edu/~jorg/archive/papers/CS-01-26.pdf)
- [17] Diot, Deployment Issues for the IP Multicast service and architecture, IEEE Network vol. 14, num 1, 2000
- [18] D. Anderson, H. Balakrishnan, F. Kaashoek, R. Morris, Resilient Overlay Networks, Infocom 2000
- [19] S. Ratnasamy, M. Handley, R. Karp and S. Shenker, Application-level Multicast using Content-Addressable Networks, Lecture Notes in Computer Science, 2233, 2001
- [20] L. Mathy, R. Canonico, and D. Hutchison, An Overlay Tree Building Protocol, Proceeding of NGC 2001
- [21] D. A. Helder, S. Jamin, End-host Multicast Communication using Switching-Trees Protocols, GP2PC'02
- [22] S. Jain, R. Mahajan, D. Wetherall, and G. Borriello, Scalable Self-Organizing Overlays, IPTPS' 02
- [23] S. Ratnasamy, M. Handley, R. Karp, S. Shenker, Topologically-Aware Overlay Construction and Server Selection, Proceeding of Infocom 2002
- [24] M. Kwon, S. Fahmy, Topology-Aware Overlay Networks for Group Communication, NOSSDAV'02,
- [25] S. Y. Shi, J. S. Turner, M. Waldvogel, Dimensioning Server Access Bandwidth and Multicast Routing in Overlay Networks, Proceeding of NOSSDAV 2001
- [26] K.W. Lee, S. Ha, J.R. Li, V. Bharghavan, An Application-level Multicast Architecture for Multimedia Communications, ACM Multimedia, 398-400, 2000
- [27] V. N. Padmanabhan, H.J. Wang, P.A. Chou, K. Sripanidkulchai, Distributing Streaming Media Content using Cooperative Networking, Microsoft Research Technical Report MSR-TR-2002-37, April 2002

- [28] The Inktomi Overlay Solution for Streaming Media Broadcasts, Inktomi White paper.
- [29] An Evaluation of Scalable Application-Level Multicast Build Using Peer-To-Peer Overlay Networks, unpublished paper.
- [30] J. Liebeherr, M. Nahas, W. Si, Application-Layer Multicasting with Delaunay Triangulation Overlays, University of Virginia Technical Report CS-2001-26
- [31] S. Banerjee, B. Bhattacharjee, C. Kommareddy, Scalable Application Layer Multicast, Sigcomm 2002
- [32] J. Saltzer, D. Reed, and D. Clark, End-to-end arguments in system design. ACM Transactions on Computer Systems, 2(4):195-206, 1984