INTERNATIONAL JOURNAL OF INTELLECTUAL PROPERTY RIGHTS (IJIPR)

ISSN 0976-6529 (Print)
ISSN 0976-6537 (Online)
Volume 5, Issue 1, January - June (2014), pp. 22-29

IJIPR
©IAEME

COPYRIGHT OF THE REVISED ARCHITECTURE FOR IMPROVEMENT IN THE DESIGN OF DATA STRUCTURE OF THE TREE TO ENHANCE THE APPLICABILITY

Dr. Suryakant B Patil[1], Ms. Ketaki G Katre[2], Ms. PriyankaGhegade[3], Dr. Preeti Patil[4]

[1]Professor, JSPM's Imperial College of Engineering & Research, Wagholi, Pune
[2, 3]PG Research Scholar, JSPM's ICOER, Wagholi, Pune
[4]Dean (SA), HOD & Professor, KIT's COE, Kolhapur

ABSTRACT

Intellectual Property Rights (IPR) is the emerging concept in the developing country like India, but the developed countries already taken major note of it. The major share of the asset values of the organizations from the developed countries is occupied by the IPR. In India the awareness has been increasing rapidly since last few years and so the scope is. When consider the data set we first need to distinguish between the structures designed and the elements in the data set (when we use the term 'data' we shall mean the content of the data set itself). Structural elements include things like the field names and a model for the data – the organization of these fields and their inter-relation.In data structures trees play major role from applicability point of view like Multicast Routing, Turing Machine, Graph Theory, CFG, Traversals, Enhancement of Network Life Time, Internal Network Characterization etc. Here we have proposed the revised architecture for the improvement in the design of the data structure of the tree. This improved design will turn up into the enhancement in the traversals, operations, time complexity along with the accessibility during executions.This will be the universal solution for the Binary tree as well as the general tree in the various applications by revising the structural design. The copyright of the same is in the pipeline through Associate Lawyer Ms. PriyaSomaiyya ofKatariya& Associates, a Patents & Trademarks Attorneys.As there are no general rules and the situation varies by Indian jurisdiction. Thus we proceed country by country detailing for (if any) this approach is used in a particular jurisdiction.

Categories and Subject Descriptors

K.5.0**[Legal Aspects of Computing]:** General
E.1 **[Data]:** Data Structures

GENERAL TERMS: Legal Aspects, Algorithms, Design, Experimentation, Performance.

Keywords: IPR, Copyright, Data Structure, Tree, Binary Tree.

1. INTRODUCTION

A tree is a non-linear data structure that consists of a root node and potentially many levels of additional nodes that form a hierarchy. A tree can be empty with no nodes called the null or empty tree or a tree is a structure consisting of one node called the root and one or more subtrees as shown in the Figure 1.

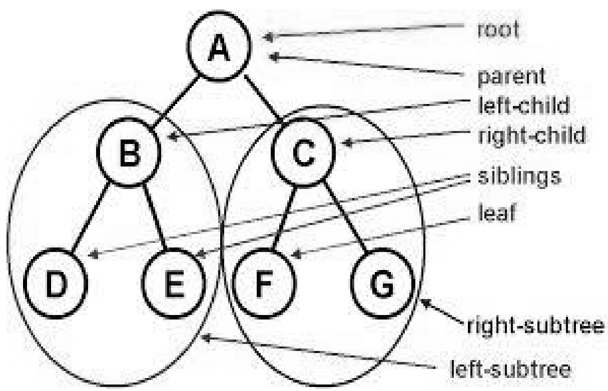

Figure 1: Data Structure of the Binary Tree

Terminologies used in tree:

- Root - the top most node in a tree.
- Parent - node that has a child.
- Siblings - nodes with the same parent.
- Leaves - nodes with no children.
- Internal nodes - nodes with at least one child.
- Degree - number of sub trees of a node.
- Edge - connection between one node to another.
- Height - The level of a node is defined by initially letting the root be at level 1.
- Forest - A forest is a set of $n \geq 0$ disjoint trees.

Types of tree

1) Simple tree

A simple unordered tree; in this diagram, the node labeled 7 has two children, labeled 2 and 6, and one parent, labeled 2. The root node, at the top, has no parent.

2) Binary tree

Binary tree is a tree data structure in which each node has at most two children (referred to as the left child and the right child). In a binary tree, the degree of each node can be at most two. Binary trees are used to implement binary search trees and binary heaps, and are used for efficient searching and sorting. A binary tree is a special case of a K-ary tree, where k is 2.

3) Binary search tree

A binary search tree (BST), sometimes also called anordered or sorted binary tree, is a node-based binary tree data structure where each node has a comparable key (and an associated value) and satisfies the restriction that the key in any node is larger than the keys in all nodes in that node's left subtree and smaller than the keys in all nodes in that node's right sub-tree. Each node has no more than two child nodes. Each child must either be a leaf node or the root of another binary search tree. The left sub-tree contains only nodes with keys less than the parent node; the right sub-tree contains only nodes with keys greater than the parent node.

II. PROPOSED DESIGN STRUCTURE

There are three fundamental types of IPv4 addresses: unicast, broadcast, and multicast. A unicast address is designed to transmit a packet to a single destination. A broadcast address is used to send a datagram to an entire subnetwork. A multicast address is designed to enable the delivery of datagrams to a set of hosts that have been configured as members of a multicast group in various scattered subnetworks.

Multicasting is not connection oriented. A multicast datagram is delivered to destination group members with the same "best-effort" reliability as a standard unicast IP datagram. This means that a multicast datagram is not guaranteed to reach all members of the group, or arrive in the same order relative to the transmission of other packets.

The only difference between a multicast IP packet and a unicast IP packet is the presence of a "group address" in the Destination Address field of the IP header. Instead of a Class A, B, or C IP address, multicasting employs a Class D destination address format (224.0.0.0- 239.255.255.255).

In the tree data structure we propose two extra fields as H (height of the tree) and C (count of the nodes). In the binary tree there will be left and right child. Initially H=0 and C=0, when we add root node then both will be H=1, C=1. If the next node of the child node will be NULL then only the addition of the new node is possible and for deletion there should be minimum one node available in the tree.

```
Initial Declaration:
int H=0, C=0;
Node -> Ref1 = H;
Node -> Ref2 = C;
ADD (Root) || ADD (New Node)
C+=1;
if (Node ->Next == NULL)
then H+=1;
else repeat;
```

```
Traversal:
Display ( &Node -> Ref1, &Node -> Ref2 )
Value = Node -> Value;
Node = Node -> Next;
(*Node -> Ref2) - = 1;
While (*Node -> Ref2 == 0)
Repeat;
*(Node -> Ref2) = 2X(*(Node -> Ref1)) + 1;
Repeat;
```

Existing IPRs and Background:

1. **Binary trees for multicast traffic**

A network device may include an input device and a packet forwarding engine. The input device receives a data packet. The packet forwarding engine includes logic configured to determine that the data packet is to be multicast. The logic is also configured to identify one of multiple forwarding schemes, where each of the forwarding schemes is configured such that the packet forwarding engine generates and forwards no more than two copies of the data packet.

Publication number US7710963 B1
Publication type Grant
Application number US 11/696,929
Publication date 4 May 2010
Also published as US7983263, US20100165989
Inventors Amit Jain, Yong Luo

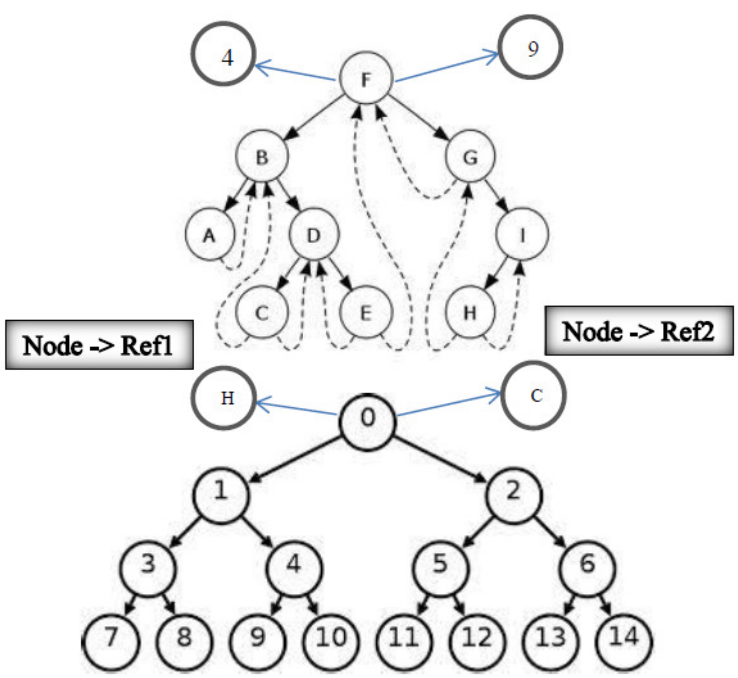

Figure 2: CLAIM - Swotted Design Augmentation Contemplated

2. **Communications network**

A method of operating a datagram network is disclosed. The method involves placing a group address in the source address field of the datagram. Notes within the datagram network forward the datagram in dependence on the contents of the source address field. This provides many of the scalability advantages offered by Multiprotocol Label Switched (MPLS) networks without introducing the overheads caused by connection set-up in MPLS networks. The method can also easily provide different quality of service levels to different types of packets and is especially useful in providing Virtual Private Networks across a shared internet work such as the public Internet.

Publication number WO2001052482 A1
Publication type Application
Application number PCT/GB2001/000092
Publication date Jul 19, 2001
Inventors Parminder Singh Mudhar, Alan William O'neill
Applicant British Telecomm, Parminder Singh Mudhar, Neill Alan William O

Reducing Network Load

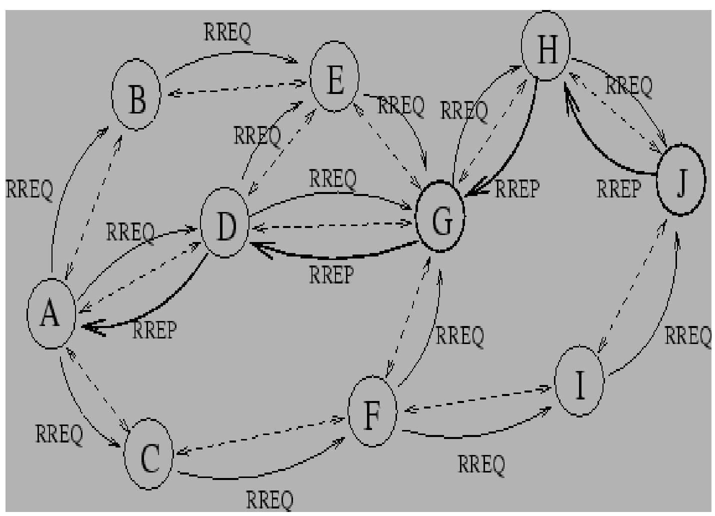

Figure 2: Multi-Cast Tree

Multicast routes are set up in a similar manner. A node wishing to join a multicast group broadcasts a RREQ with the destination IP address set to that of the multicast group and with the 'J'(join) flag set to indicate that it would like to join the group. Any node receiving this RREQ that is a member of the multicast tree that has a fresh enough sequence number for the multicast group may send a RREP. As the RREPs propagate back to the source, the nodes forwarding the message set up pointers in their multicast route tables. As the source node receives the RREPs, it keeps track of the route with the freshest sequence number, and beyond that the smallest hop count to the next multicast group member. After the specified discovery period, the source nodes will unicast a Multicast Activation (MACT) message to its selected next hop.

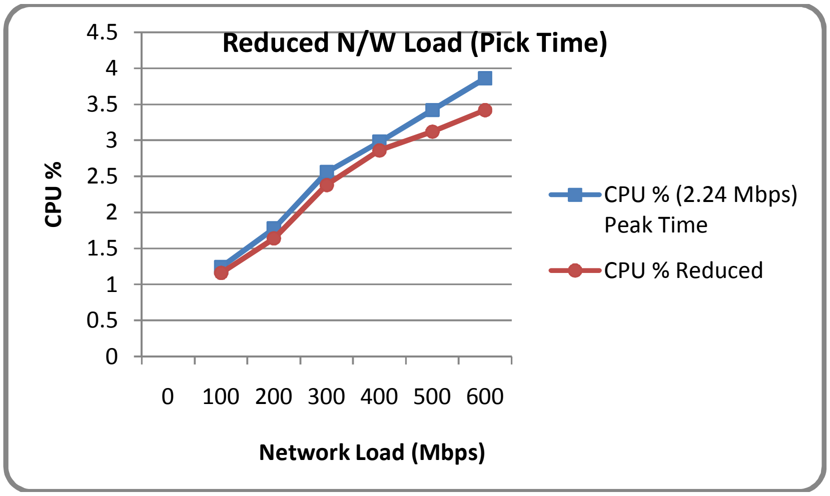

Figure 3: N/W Load Vs CPU % (PT 2.24 Mbps)

This message serves the purpose of activating the route. A node that does not receive this message had set up a multicast route pointer will timeout and deletes the pointer. If the node receiving the MACT was not already a part of the multicast tree, it will also have been keeping track of the best route from the RREPs it received. Hence it must also unicast a MACT to its next hop, and so on until a node that was previously a member of the multicast tree is reached.

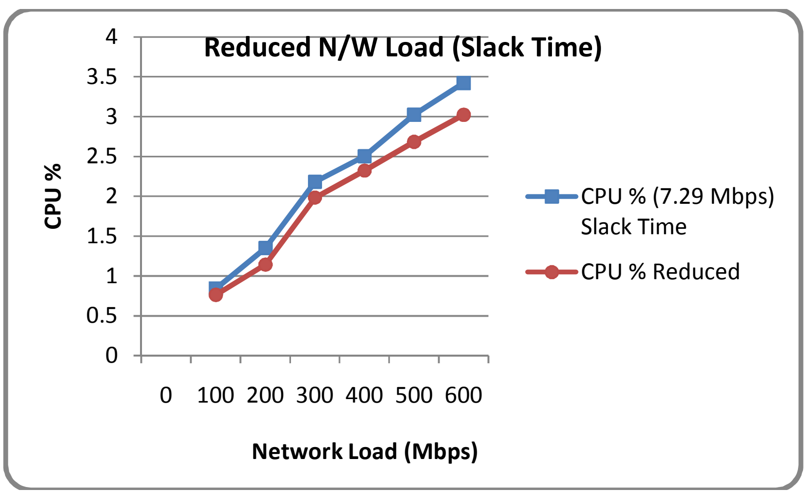

Figure 4: N/W Load Vs CPU % (ST 7.29 Mbps)

Assume that a stock ticker application is required to transmit packets to 100 stations within an organization's network. Unicast transmission to the group of stations will require the periodic transmission of 100 packets where many packets may be required to traverse the same link(s). Multicast transmission is the ideal solution for this type of application since it requires only a single packet transmission by the source which is then replicated at forks in the multicast delivery tree.

III. CLAIMS

The additional soft links provided for the root node as a Ref1 and Ref2 will be useful during the executions and traversals throughout respective applications.

1. The Ref1 will have the height of the tree which is creates along with the creation of the root node. The same is persistent throughout till the existence of the last node. This link Ref1 can be used directly rather than calculating the height of the tree or binary tree with the help of the some algorithms or calculations. The direct availability of this value throughout the all operations and traversal will be an added advantage to calculate the Prefix/ Infix/ Postfix etc as well as during the Preorder, Inorder and Post order Traversals. The additional toll will be negligible compared with the reduced time complexity for faster processing during relative applications.
2. The Ref2 will be the zero before create the root node and will have continuous updations as per the addition/ deletions operations on the tree or binary tree. The addition will have the simultaneously update this pointer value and will available throughout for further use rather than to calculate the height and followed by the count. Either traversals or Grammar operations or multicast routing table the repetitive calculations for the same will be minimized which results in the improvement in the latency time with faster results during the execution. As this value is readily available, no need to traverse till the NULL pointer to check and it can have predefined loop with this count value as a limit.

Both the pointers will be deleted whenever the root gets deleted and its memory location will get free.

IV. REFERENCES

PATENT CITATIONS

Cited Patent	Filing date	Date	Applicant	Title
US7983263 *	12 Mar 2010	19 Jul 2011	Juniper Networks, Inc.	Binary trees for multicast traffic
WO199800 2821A1 *	Jul 15, 1997	Jan 22, 1998	3Com Corp	Virtual network architecture for connectionless LAN backbone
US5910955 *	Aug 25, 1997	Jun 8, 1999	Fujitsu Limited	Switching hub capable of controlling communication quality in LAN
US5402415 *	22 Apr 1993	28 Mar 1995	Washington University	Multicast virtual circuit switch using cell recycling

Cited Patent	Filing date	Date	Applicant	Title
US7263099 *	14 Aug 2002	28 Aug 2007	Juniper Networks	Multicast packet replication
US7266120 *	18 Nov 2002	4 Sep 2007	Fortinet, Inc.	System and method for hardware accelerated packet multicast in a virtual routing system
US7289503 *	30 Jul 2002	30 Oct 2007	Juniper Networks, Inc.	Systems and methods for efficient multicast handling
US7420972 *	3 Aug 2007	2 Sep 2008	Juniper Networks, Inc.	Multicast packet replication
US7519054 *	27 Jan 2005	14 Apr 2009	Intel Corporation	Replication of multicast data packets in a multi-stage switching system
US20060114903 *	29 Nov 2004	1 Jun 2006	Egenera, Inc.	Distributed multicast system and method in a network
US20060215698 *	17 Nov 2004	28 Sep 2006	Amen Hamdan	Communication subsystem controlled information dissemination