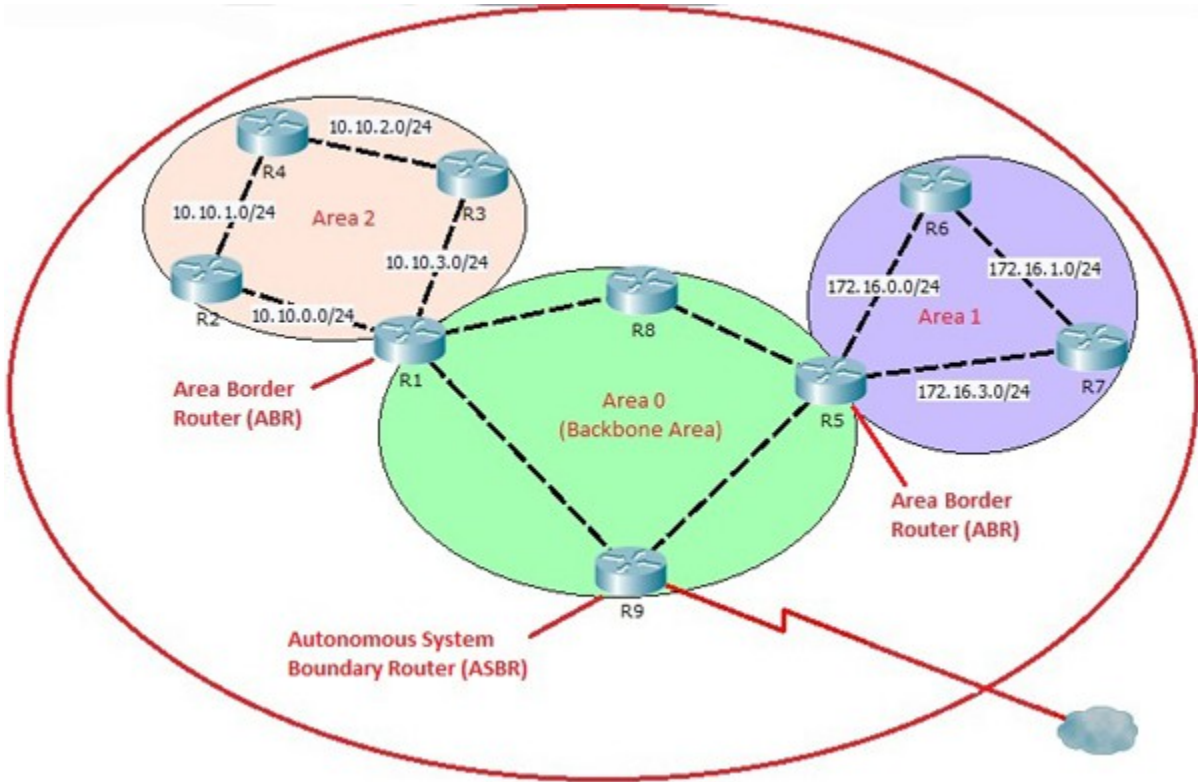


## OSPF Concept

OSPF এর পূর্ণরূপ হলো Open Shortest Path First । এটি একটি Link State রাউটিং প্রটোকল যা Distance Vector রাউটিং প্রটোকল RIP এর রিপ্লেসমেন্ট হিসেবে এসেছে। OSPF ডেভেলপ করার কাজ শুরু হয় 1987 সালে IETF এর মাধ্যমে। 1989 সালে OSPFv1 (Version 1) আবির্ভূত হয়, যা ছিল পরীক্ষামূলক। 1991 সালে OSPFv2 (Version 2) আসে। পরবর্তীতে 1999 সালে IPv6 এর জন্য OSPFv3 (Version 3) উন্মুক্ত হয়।

### OSPF Routing Area

এখন আমরা দেখবো OSPF রাউটিং প্রটোকল কিভাবে এক বা একাধিক রাউটিং এরিয়া নিয়ে কাজ করে।



প্রতিটি OSPF রাউটিং এরিয়া দুই বা ততোধিক রাউটার নিয়ে গঠিত হয়। এরকম যতগুলো এরিয়া থাকুক না কেন তারা যদি একে অন্যের সাথে রাউট বিনিময় করতে চায় তাহলে তাদেরকে অবশ্যই Area 0 এর সাথে যুক্ত থাকতে হবে। Area 0 কে Backbone Area হিসেবে গন্য করা হয়। কোন একটি নির্দিষ্ট এরিয়ার সকল রাউটারের টপোলজি টেবিল একই ধরনের হয়। চিত্রে Area 1 ও Area 2 ব্যাকবোন এরিয়া Area 0 এর সাথে সংযুক্ত। Area 2 এ চারটি রাউটার আছে, এই চারটি রাউটারের প্রতিটিই Area 2 এর সব তথ্য জানে। যদি Area 2 এর মধ্যে কোন একটি লিংক ডাউন হয় তবে সেই আপডেট Area 2 এর এই চারটি রাউটারের প্রতিটির কাছে পৌঁছাবে। আবার যখন কোন লিংক আপ হয় তখন সেই আপডেটও এই চারটি রাউটারের

কাছে আসবে। কিন্তু Area 2 এর কোন আপডেট অন্য কোন এরিয়া এমনকি Area 0 (Backbone Area) এর কাছেও আসবে না। এতে করে রাউটিং আপডেট একটি নির্দিষ্ট এরিয়ার মধ্যে সীমাবদ্ধ থাকে। এতে রাউটিং ইফিসিয়েন্সি বৃদ্ধি পায়।

এখন প্রশ্ন হলো, Area 2 এর কোন লিংক ডাউন বা আপ হওয়ার আপডেট যদি অন্য কোন এরিয়ায় না পৌঁছায় তাহলে অন্য এরিয়ার রাউটারসমূহ ঐ লিংক এর তথ্য কিভাবে পাবে অথবা পাবেই কিনা? এটাই হলো OSPF Hierarchical Network Design এর আসল মজা। চিত্রানুসারে, Area 2 এর R1 রাউটারটি হলো একটি Area Border Router (ABR) যা Area 2 এবং Area 0 এর মধ্যে সংযোগ স্থাপন করেছে, এর একটি ইন্টারফেস Area 2 এ এবং আরেকটি ইন্টারফেস Area 0 এর সাথে যুক্ত। এই R1 রাউটার Area 2 এর সকল নেটওয়ার্ককে সামারাইজ করে অন্য এরিয়ার নিকট পাঠায়। উদাহরণস্বরূপ, Area 2 এর চারটি নেটওয়ার্ক 10.10.0.0/24, 10.10.1.0/24, 10.10.2.0/24 ও 10.10.3.0/24 কে সামারাইজ করে একটি সিঙ্গেল নেটওয়ার্ক 10.10.0.0/22 এ পরিণত করে এবং এই 10.10.0.0/22 কে রাউটিং আপডেট হিসেবে Area 0 এর কাছে পাঠায়। এখন যদি কোন কারনে 10.10.2.0/24 নেটওয়ার্কটি ডাউন হয়ে যায় তাহলে সেই আপডেট Area 0 এর কাছে আসবে না। একে Route Localization বলে। এতে করে সমগ্র নেটওয়ার্কের রাউটিং ইফিসিয়েন্সি বৃদ্ধি পায়।

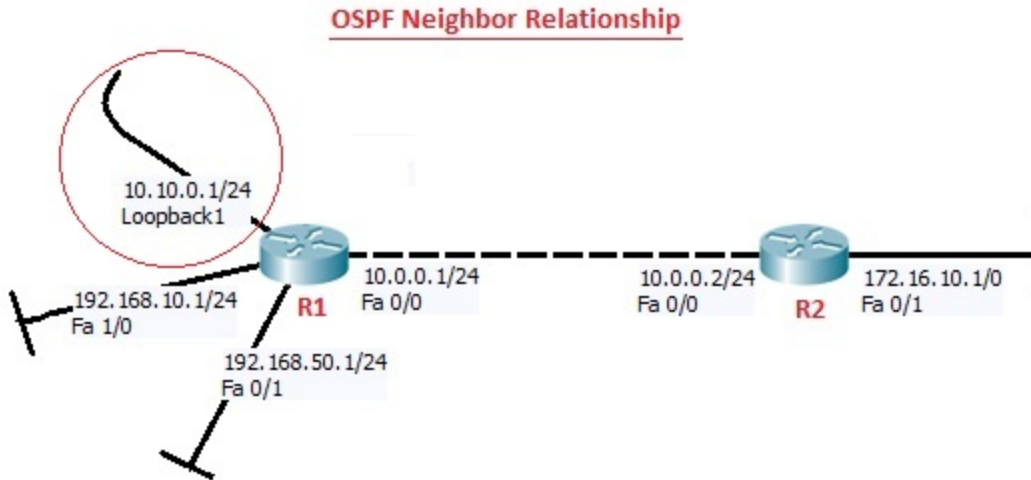
এখন ধরি, যদি কোন কারণে 10.10.2.0/24 নেটওয়ার্কটি Area 2 এর মধ্যে না থেকে Area 1 এর মধ্যে থাকে অথবা 172.16.3.0/24 নেটওয়ার্কটি Area 1 এ না থেকে Area 2 তে থাকে তাহলে R1 বা R5 কোন রাউটারই তাদের নিজ নিজ এরিয়ার নেটওয়ার্কসমূহকে সঠিকভাবে সামারাইজ করতে পারবে না, কারণ নেটওয়ার্কটি Discontiguous হয়ে পড়বে। এতে নেটওয়ার্ক ডিজাইন Hierarchical হবে না এবং সামারি রাউটের সুফলও পাওয়া যাবে না। কারণ OSPF Hierarchical Network Design মূলত নির্ভর করে IP Addressing Scheme এর উপর। তাই নেটওয়ার্ক ডিজাইনের সময় একজন এ্যাডমিনিষ্ট্রেটরকে এ সকল বিষয়সমূহ বিবেচনায় রাখা উচিত।

চিত্রে R9 রাউটারটি হলো একটি Autonomous System Boundary Router (ASBR) যা Internet বা Outside নেটওয়ার্কের সাথে এই নির্দিষ্ট রাউটিং ডোমেইনটিকে যুক্ত করেছে। রাউটার R9 তার রাউটিং ডোমেইনের সবগুলো এরিয়ার সবগুলো নেটওয়ার্ককে সামারাইজ করে Outside নেটওয়ার্কে পাঠিয়ে থাকে। এখানে একটি কথা বিশেষভাবে উল্লেখযোগ্য যে, OSPF এর ক্ষেত্রে শুধুমাত্র ABR এবং ASBR রাউটারই কেবল রাউট সামারাইজ করতে পারে।

## OSPF Neighbor Relationship

দুইটি OSPF রাউটারের মধ্যে Neighbor Relationship তৈরী হওয়ার মূল শর্ত হলো রাউটার দুটিকে অবশ্যই একই রাউটিং এরিয়ার মধ্যে থাকতে হবে। দুইটি ভিন্ন এরিয়ার রাউটারের মধ্যে Neighbor Relationship তৈরী হয় না। তবে যে সকল রাউটার Area Border Router (ABR) হিসেবে কাজ করে সেই রাউটারগুলো যেহেতু একাধিক রাউটিং এরিয়ার সাথে যুক্ত থাকে তাই তারা একাধিক এরিয়ার রাউটারসমূহের সাথে Neighbor Relationship তৈরী করতে পারে।

এখন আমরা দেখবো দুইটি OSPF রাউটারের মধ্যে কিভাবে Neighbor Relationship তৈরী হয়।



ধাপ-১: যখন কোন রাউটারের মধ্যে OSPF কনফিগার করা হয় তখন ঐ রাউটারের মধ্যে OSPF প্রসেস শুরু হয়। OSPF প্রসেসের শুরুতেই রাউটার তার Router ID নির্বাচন করে। Router ID হলো একটি OSPF রাউটারের নাম যা ঐ রাউটারের OSPF অপারেশনে ব্যবহৃত হয়। মূলত এই Router ID হল ঐ রাউটারের একটি আই.পি এ্যাড্রেস। অর্থাৎ একটি OSPF রাউটার তার ইন্টারফেসসমূহের মধ্যে বসানো আই.পি এ্যাড্রেসগুলোর মধ্য থেকে একটি আই.পি কে তার Router ID হিসেবে নির্বাচন করে। এখন প্রশ্ন হলো রাউটার তার কোন আই.পি কে Router ID হিসেবে নির্বাচন করবে? এক্ষেত্রে রাউটার তার Highest Active আই.পি টিকে Router ID হিসেবে গন্য করবে।

চিত্রে রাউটার R1 তার 192.168.50.1 আই.পি টিকে Router ID হিসেবে নির্বাচন করবে। কারণ রাউটার R1 এর সবগুলো আই.পি'র মধ্যে সংখ্যার দিক থেকে এটিই সবচেয়ে বড় (ধরি, চিত্রে Loopback1 নাই)। কিন্তু এক্ষেত্রে একটু ব্যতিক্রম রয়েছে। যদি R1 এ কোন Loopback ইন্টারফেস কনফিগার করা থাকে তাহলে ঐ Loopback ইন্টারফেসের আই.পি-টিই হবে রাউটারে Router ID। চিত্রে R1 এর Loopback1 ইন্টারফেসের আই.পি 10.10.0.1 ই হবে এর Router ID। Router ID নির্ধারণের ক্ষেত্রে Loopback ইন্টারফেসের আই.পি সবসময় Physical ইন্টারফেসের আই.পি'র চেয়ে প্রাধান্য পায়। যদি এরকম হয় যে, রাউটারে একাধিক Loopback ইন্টারফেস আছে, তাহলে ঐ Loopback ইন্টারফেসসমূহের আই.পি গুলোর মধ্যে যেটি বড় সেটিই হবে রাউটারের Router ID।

একটি প্রোডাকশন নেটওয়ার্কে চলমান কোন OSPF রাউটারের কোন Physical বা Loopback ইন্টারফেসে যদি এমন কোন আই.পি বসানো হয় যা অন্যান্য আই.পি সমূহ থেকে বড়, তারপরও ঐ রাউটারের Router ID পরিবর্তন হবে না। কোন একটি রাউটারের Router ID তখনই নতুন করে নির্বাচন করা হয় যখন ঐ রাউটারের OSPF প্রসেসটি রিস্টার্ট দেওয়া হয় অথবা ঐ রাউটারটিকেই রিবুট দেওয়া হয়। তবে সব কথার শেষ কথা হলো, একজন এ্যাডমিনিষ্ট্রেটর চাইলে নিজের ইচ্ছামতো কোন রাউটারের Router ID নির্ধারণ করে দিতে পারেন। সেক্ষেত্রে রাউটারের Physical বা Loopback ইন্টারফেসের কোন Highest আই.পি ই বিবেচিত হবে না।

ধাপ-২: দ্বিতীয় ধাপে, যখন কোন রাউটারের OSPF প্রসেসে network কমান্ডের মাধ্যমে কোন নেটওয়ার্ককে যোগ করা হয় তখন রাউটার তার ঐ ইন্টারফেসের নেটওয়ার্কটিকে নিজের Link State ডাটাবেজে যুক্ত করে। ধরি চিত্রে R1 রাউটার তার 10.0.0.0/24, 192.168.10.0/24 ও 192.168.50.0/24 নেটওয়ার্কগুলোকে তার Link State ডাটাবেজে যুক্ত করেছে।

ধাপ-৩: যখন নেটওয়ার্কসমূহ R1 এর Link State ডাটাবেজে যুক্ত হয় তখন R1 ঐ সকল নেটওয়ার্কের সংশ্লিষ্ট ইন্টারফেসসমূহ দিয়ে Neighbor ডিসকভারির জন্য একটি করে Hello Packet পাঠায়। R1 যতক্ষণ পর্যন্ত ঐ ইন্টারফেসসমূহ দিয়ে কোন Hello Reply না পায় ততক্ষণ পর্যন্ত R1 এর Link State ডাটাবেজে ইন্টারফেসসমূহ DOWN STATE এ থাকে।

Broadcast/P2P নেটওয়ার্কে প্রতি 10 সেকেন্ড পরপর এবং NBMA নেটওয়ার্কে প্রতি 30 সেকেন্ড পরপর Hello Packet পাঠানো হয়। উভয়ক্ষেত্রেই Dead Timer হলো Hello Timer এর চারগুন।

একটি Hello Packet এ নিচের Field গুলো থাকে।

১. **Router ID:** রাউটিং অপারেশনে ব্যবহৃত কোন রাউটারের নাম।
২. **Area ID\*:** OSPF রাউটিং এরিয়ার এরিয়া নম্বর।
৩. **Network Mask\*:** যে নেটওয়ার্কটিকে OSPF রাউটিং প্রসেসে যোগ করা হয় তার সাবনেট মাস্ক।
৪. **Hello Interval/Dead Interval\***
৫. **List of Neighbor:** রাউটারের অন্যান্য OSPF Neighbor সমূহের তালিকা (যদি থাকে)।
৬. **Router Priority:** পরের অংশে আলোচনা করা হবে।
৭. **Designated Router (DR):** পরের অংশে আলোচনা করা হবে।
৮. **Backup Designated Router (BDR):** পরের অংশে আলোচনা করা হবে।
৯. **Authentication Password\*:** OSPF রাউটিং এর নিরাপত্তা বৃদ্ধির জন্য ব্যবহৃত বিশেষ পাসওয়ার্ড (যদি থাকে)।

উপরিউক্ত Field সমূহের মধ্যে \* চিহ্নিত Field গুলো অবশ্যই দুইটি রাউটারে এক হতে হবে। যদি এই \* চিহ্নিত Field সমূহ ভিন্ন হয় তাহলে Neighbor Relationship তৈরী হবে না।

ধাপ-৪: এই ধাপে R2 রাউটার R1 এর Hello প্যাকেট গ্রহন করে। এসময় R2 রাউটার INIT STATE এ থাকে। R2 রাউটার প্রাপ্ত Hello প্যাকেটের Area ID, Netmask, Hello/Dead Timer ও Authentication Password এই চারটি Field নিজের Field এর সাথে চেক করে দেখে যে ঠিক আছে কিনা।

ধাপ-৫: যদি Field সমূহ ঠিক থাকে তাহলে R2 রাউটার R1 এর কাছে Hello Reply পাঠায়। এসময় R1 ও R2 উভয় রাউটার 2-WAY STATE এ পদার্পন করে। অর্থাৎ রাউটার দুটির মধ্যে Neighbor Relationship তৈরী হয়। এসময় রাউটারদ্বয় চেক করে দেখে যে, তাদের নিজেদের মধ্যে আগে থেকেই Neighbor Relationship ছিল কিনা। এখানে একটি কথা মনে রাখা দরকার যে, শুধুমাত্র অপরিচিত দুইটি রাউটারই Relationship তৈরীর জন্য নিজেদের মধ্যে Hello প্যাকেট বিনিময় করে না, নিজেদের মধ্যে Neighbor Relationship আছে এমন রাউটারসমূহও সেই Relationship বজায় রাখার জন্য নিয়মিত বিরতিতে Hello প্যাকেট বিনিময় করে (Broadcast/P2P নেটওয়ার্কে প্রতি 10 সেকেন্ড পরপর এবং NBMA নেটওয়ার্কে প্রতি 30 সেকেন্ড পরপর)।

যদি আগে থেকেই Relationship থেকে থাকে তাহলে রাউটারদ্বয় নিজেদের Dead Timer টি রিসেট করে নেয় এবং নিজেদের মধ্যে রাউটিং টেবিল বিনিময় করে না। আর যদি নতুন Relationship হয় তাহলে তারা নিজেদের মধ্যে রাউটিং টেবিল বিনিময় করার জন্য ধাপ-৬ এ গমন করে।

ধাপ-৬: এই ধাপে রাউটারদ্বয় নিজেদের মধ্যে Master-Slave রিলেশনশীপ নির্ণয় করে অর্থাৎ কে Master আর কে Slave। এসময় রাউটারদ্বয় EX-START State এ থাকে (Exchange Start)। নতুন দুইটি Neighbor এর মধ্যে Master-Slave রিলেশনশীপ তেমন বিশেষ কিছুই নয়। এখানে যে রাউটারটি Master হবে সে আগে Slave এর নিকট নিজের Link State Database পাঠাবে। আর যে Slave হবে সে পরে Master এর নিকট তার নিজের Link State Database পাঠাবে। এখন দেখা যাক কিভাবে নির্ধারিত হয় কে Master আর কে Slave?

যে রাউটারের Router Priority বেশি হবে সে রাউটার হবে Master আর অন্যটি হবে Slave। যদি রাউটার দুইটির Priority ভ্যালু একই হয় (যেহেতু রাউটারসমূহের Priority বাই ডিফল্ট 1), তাহলে তারা Router ID ব্যবহার করে Master-Slave নির্বাচন করে থাকে। এক্ষেত্রে যে রাউটারের Router ID বড় সে হবে Master আর অন্যটি হবে Slave।

Master রাউটার Slave এর নিকট প্রথমে নিজের Database Description (DBD) টেবিল পাঠায়। DBD হলো রাউটারের Link State ডাটাবেজের একটি সার সংক্ষেপ। অতঃপর Slave রাউটারও Master এর কাছে নিজের DBD পাঠায়। ধরি, চিত্রে R1 হলো Master রাউটার এবং R2 হলো Slave রাউটার।

ধাপ-৭: এই ধাপে রাউটারদ্বয় এক অন্যের থেকে পাওয়া DBD এর জন্য Acknowledgement পাঠায় (যেহেতু OSPF একটি রিলায়েবল প্রটোকল)। এসময় রাউটারদ্বয় LOADING STATE এ থাকে।

Master রাউটার R1 Slave রাউটার R2 এর কাছে যে DBD পাঠিয়েছে তার মধ্যে তিনটি নেটওয়ার্ক আছে (10.0.0.0/24, 192.168.10.0/24 ও 192.168.50.0/24)। এখন Slave R2 যেহেতু 10.0.0.0/24 নেটওয়ার্ককে তার ডিরেক্টলি কানেক্টেড নেটওয়ার্ক হিসেবে চেনে তাই এটি বাকি দুইটি নেটওয়ার্কের বিস্তারিত চেয়ে Master R1 এর কাছে দুইটি আলাদা Link State Request (LSR) পাঠায় এবং R1 প্রাপ্ত LSR গুলো চেক করে কাঙ্ক্ষিত নেটওয়ার্কের বিস্তারিত তথ্য Link State Update (LSU) আকারে R2 এর নিকট পাঠায়। অনুরূপভাবে, Slave R2 রাউটার Master R1 এর কাছে যে DBD পাঠিয়েছে তার মধ্যে দুইটি নেটওয়ার্ক আছে (10.0.0.0/24 ও 172.16.10.0/24)। এখন Master R1 যেহেতু 10.0.0.0/24 নেটওয়ার্ককে তার ডিরেক্টলি কানেক্টেড নেটওয়ার্ক হিসেবে চেনে তাই এটি বাকি একটি নেটওয়ার্কের বিস্তারিত চেয়ে Slave R2 এর কাছে একটি LSR পাঠায় এবং R2 সেই LSR চেক করে কাঙ্ক্ষিত নেটওয়ার্কের বিস্তারিত তথ্য LSU আকারে R1 এর নিকট পাঠায়।

ধাপ-৮: যখন Master ও Slave রাউটার নিজেদের মধ্যে তথ্য বিনিময় করে উভয়ই একই ধরনের Link State ডাটাবেজ তৈরী করে তখন তাদের মধ্যকার Neighbor রিলেশনশীপকে Synchronized হিসেবে গন্য করা হয়। এসময় উভয় রাউটারই Full State এ গমন করে। যখন কোন রাউটার FULL STATE এ থাকে তখন ইহা তার Neighbor এর সাথে রাউটিং টেবিল বিনিময় করে এবং Dijkstra's Algorithm এর মাধ্যমে SPF Tree তৈরী করে বিভিন্ন ডেস্টিনেশনে যাওয়ার জন্য Shortest Path নির্ণয় করে।



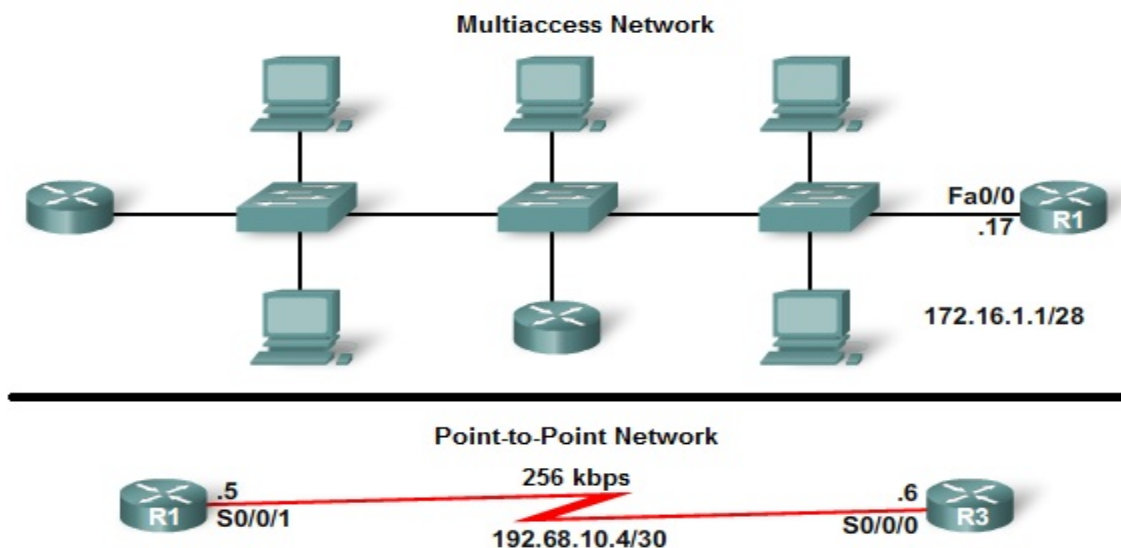
## OSPF Packet Types

আমরা OSPF Neighbor Relationship আলোচনার সময় যে OSPF Packet সমূহ সম্পর্কে জেনেছি এক নজরে তা নিম্নরূপঃ

১. **Hello:** এই প্যাকেটটি Neighbor ডিসকভারির জন্য ব্যবহৃত হয়।
২. **Database Description (DBD):** এই প্যাকেটের মধ্যে একটি রাউটারের Link State ডাটাবেজের সামারী থাকে যা Neighbor রাউটারের নিকট পাঠানো হয়।
৩. **Link State Request (LSR):** এই প্যাকেট পাঠানোর মাধ্যমে একটি রাউটার আরেকটি রাউটারের কাছে কোন নির্দিষ্ট নেটওয়ার্কের আপডেট জানতে চায়।
৪. **Link State Update (LSU):** LSR এর মাধ্যমে জানতে চাওয়া কোন আপডেট রিকোয়েস্ট এর জবাবে এই LSU পাঠানো হয়। একটি LSU এর মধ্যে এক বা একাধিক LSA থাকে।
৫. **Link State Advertisement (LSA):** এই LSA এর মাধ্যমে একটি রাউটার আরেকটি রাউটারের কাছে রাউটিং আপডেট পাঠিয়ে থাকে। পাঠানো রাউটের সংখ্যা যত হয়, LSA এর সংখ্যাও তত হয়। অর্থাৎ একটি LSA তে কেবল একটি রাউটের আপডেট থাকে।
৬. **Link State Acknowledgement (LSACK):** যেহেতু OSPF একটি রিয়ায়েবল প্রটোকল তাই OSPF অপারেশনের সময় এই LSACK এর মাধ্যমে অন্যান্য প্যাকেট যেমন DBD, LSR, LSU, LSA এর Acknowledgement প্রদান করা হয়।

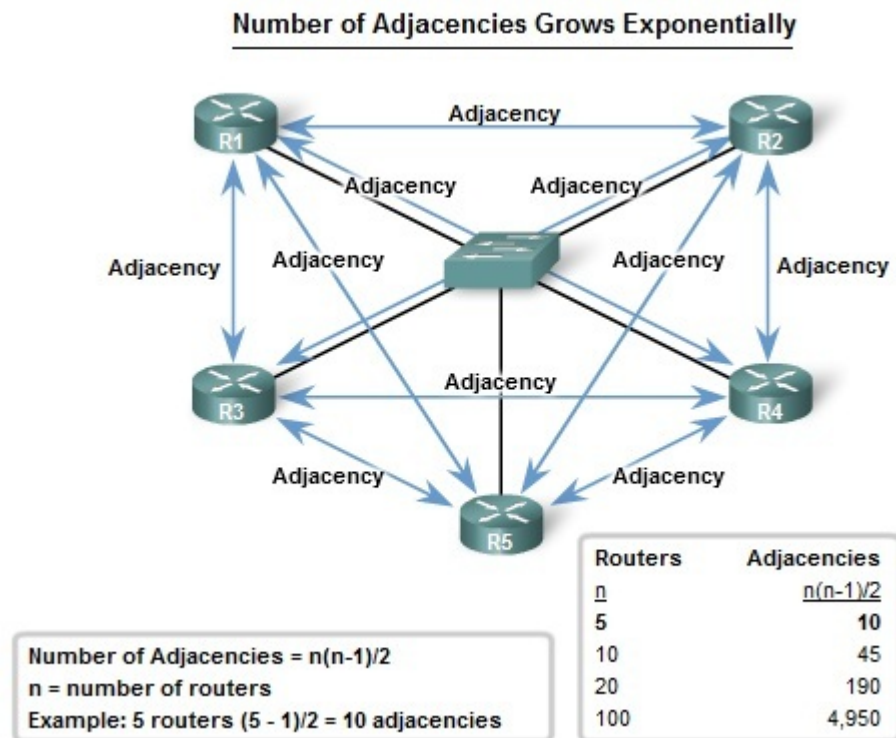
## OSPF Multi Access Network: DR and BDR

আমরা ইতিমধ্যে OSPF এর যে আলোচনা করেছি তাতে দুইটি রাউটারের মধ্যে Point-to-Point সংযোগ দেখেছি এবং এটাও দেখেছি যে এই Point-to-Point নেটওয়ার্কে দুইটি রাউটার কিভাবে একে অন্যের সাথে Neighbor Relationship তৈরী করে এবং প্যাকেট বিনিময় করে। এখন আমরা দেখবো একটি Multi Access নেটওয়ার্কে রাউটারসমূহ কিভাবে নিজেদের মধ্যে OSPF অপারেশন চালায়।



Multi Access নেটওয়ার্ক বলতে এমন একটি নেটওয়ার্ককে বুঝায় যেখানে দুইয়ের বেশি রাউটার একটি Shared মিডিয়ার সাথে যুক্ত থাকে।

Multi Access নেটওয়ার্কের একটি বড় অসুবিধা হল, এই নেটওয়ার্কে যুক্ত প্রতিটি রাউটার একটি আরেকটির সাথে Neighbor Relationship/Adjacency তৈরী করে এবং আলাদা আলাদা ভাবে DBD ও LSP আদান-প্রদান করে। Multi Access নেটওয়ার্কে রাউটারের সংখ্যা যত বেশি হবে এই রাউটারসমূহের মধ্যে Number of Adjacency ও তত বেশি হবে। এতে করে এই নেটওয়ার্কে LSP Flooding এমন অবস্থায় পৌঁছাবে যে ব্যান্ডউইথ স্বল্পতার জন্য নেটওয়ার্কের Real Traffic আদান-প্রদান বাধাগ্রস্ত হবে।



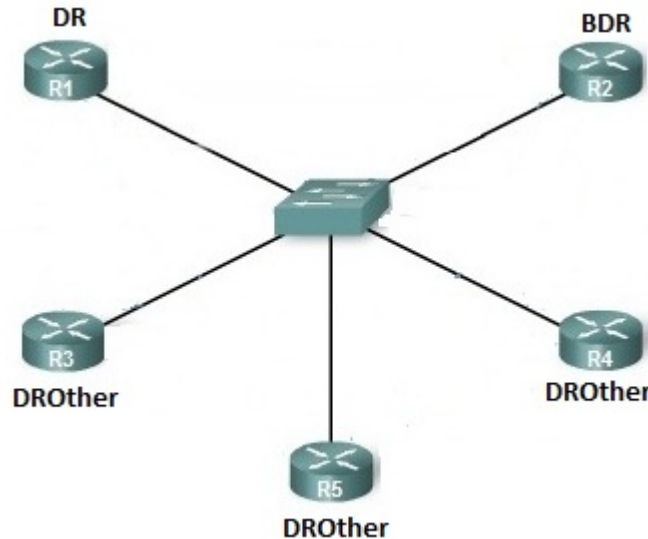
ব্যাপারটা আরেকটু সহজ করা যাক। ধরি একটি কক্ষে 10 জন ব্যক্তি আছে। এখন এই 10 জন ব্যক্তির প্রত্যেকে যদি প্রত্যেকের সাথে নাম-পরিচয় বিনিময় করে তাহলে যে সময় লাগবে ও ঝামেলা হবে তার চেয়ে যদি এই 10 ব্যক্তির মধ্যে মাত্র একজন সামনে দাড়িয়ে এক জন এক জন করে সবার নাম-পরিচয় বর্ণনা করে তাহলে ব্যাপারটা অনেক সুন্দর ও সাবলীল হয়। ঠিক তেমনি Multi Access নেটওয়ার্কে এই সমস্যা দূর করার জন্য সবগুলো রাউটারের মধ্য থেকে একটি রাউটারকে Designated Router (DR) হিসেবে নির্বাচন করা হয়, যার কাজ হলো এই Multi Access নেটওয়ার্কের সব রাউটারের কাছে থেকে আগে আপডেট নিয়ে আপডেটগুলোকে একত্রিত করে আবার সবগুলো রাউটারের কাছে পৌঁছে দেওয়া। এই Multi Access নেটওয়ার্কে আবার একটি Backup Designated Router (BDR) থাকে। যদি কোন কারণে DR ফেইল করে

তাহলে BDR তার স্থলাভিষিক্ত হয় এবং OSPF অপারেশন সমন্বয় করে। DR ও BDR ছাড়া এই Multi Access নেটওয়ার্কের অন্যান্য রাউটারসমূহকে DROther হিসেবে গন্য করা হয়।

## DR/BDR Election Process

এখন আমরা দেখবো একটি Multi Access নেটওয়ার্কে দুইয়ের বেশি রাউটারের মধ্য থেকে কোন রাউটারটি DR এবং কোন রাউটারটি BDR হিসেবে নির্বাচিত হয় এবং এই নির্বাচন প্রক্রিয়া কিভাবে সম্পন্ন হয়।

একটি Multi Access নেটওয়ার্কে যে রাউটারটির Router Priority সবচেয়ে বেশি সে রাউটারটি DR হিসেবে বিবেচিত হয় এবং Router Priority এর দিক থেকে যে রাউটারটি দ্বিতীয় সে রাউটারটি BDR হিসেবে নির্বাচিত হয়। কোন রাউটারের Router Priority ইচ্ছামতো কনফিগার করা যায়। যেহেতু সব রাউটারের Router Priority বাই ডিফল্ট 1, তাই Multi Access নেটওয়ার্কে সবগুলো রাউটারের Router Priority সমান হয়ে গেলে টাই ব্রেকার হিসেবে সর্বোচ্চ Router ID এর ভিত্তিতে DR ও BDR নির্বাচিত হয়। এই DR ও BDR ছাড়া অন্যান্য রাউটারসমূহ DROther হিসেবে থাকে। একটি DROther রাউটার অন্য কোন DROther রাউটারের সাথে Neighbor Relationship তৈরী করে না, এটি শুধুমাত্র DR ও BDR এর সাথে Neighbor Relationship তৈরী করে। তাই একটি DROther এর সাথে আরেকটি DROther এর রিলেশনশীপ 2-WAY STATE এ থাকে এবং একটি DROther এর সাথে DR/BDR এর রিলেশনশীপ FULL STATE এ থাকে।



ধরি, চিত্রে প্রদত্ত Multi Access নেটওয়ার্কটিতে R1 হলো DR, R2 হলো BDR এবং অন্যান্য রাউটারসমূহ হলো DROther। R1 রাউটার ততক্ষণ পর্যন্ত DR হিসেবে কাজ করবে যতক্ষণ না পর্যন্ত এটি ডাউন হয় অথবা এর OSPF প্রসেস রিস্টার্ট দেওয়া হয়। যখন R1 রাউটার উল্লেখিত কোন একটি কারণে ফেইল করে তখন BDR রাউটার R2 সাথে সাথে DR এর জায়গা দখল করে এবং R3 রাউটার BDR হিসেবে কাজ শুরু করে। ধরি কিছুক্ষণ পর আবার R1 রাউটারটি আপ হল, তখন কিন্তু এটি আর DR/BDR হিসেবে বিবেচিত হবে না যদিও এর Router ID অন্যান্য রাউটারসমূহের চেয়ে বড়। R1 রাউটার শুধুমাত্র তখনই BDR হিসেবে



গন্য হবে যখন BDR রাউটার R3 ফেইল করবে এবং শুধুমাত্র তখনই DR হিসেবে গন্য হবে যখন R2 ও R3 (DR ও BDR) উভয়ই ফেইল করবে।

একটি Multi Access নেটওয়ার্কে DR রাউটারকে যেহেতু অনেক বেশি LSA হ্যান্ডেল করতে হয় তাই এর পর্যাপ্ত CPU ও Memory ক্যাপাসিটি থাকা বাঞ্ছনীয়। এজন্য নেটওয়ার্কের সবচেয়ে ভাল কনফিগারেশনের রাউটারটিই যাতে DR হিসেবে নির্বাচিত হতে পারে তাই এর Router Priority (বাই ডিফল্ট 1) পরিবর্তন করে একটি বড় সংখ্যা নির্ধারণ করে দেওয়া উচিত।

## OSPF Cost

এখন আমরা দেখবো, কিভাবে একটি OSPF রাউটার কোন নেটওয়ার্কে যাওয়ার জন্য ঐ নেটওয়ার্কের Cost ক্যালকুলেট করে। আমরা জানি, OSPF রাউটিং প্রটোকল ব্যান্ডউইথ এর পরিমানের উপর ভিত্তি করে Best পথ নির্ধারণ করে। কিন্তু প্রকৃতপক্ষে OSPF অপারেশনের ব্যাকগ্রাউন্ডে এই Bandwidth ভ্যালুটি সরাসরি ব্যবহৃত হয় না।

এই Bandwidth ভ্যালুটিকে একটু আলাদাভাবে হিসেব করে প্রতিটি পথের Cost বের করা হয়। আর এই Cost এর উপর ভিত্তি করেই Best পথ নির্ধারিত হয়ে থাকে।

লিংক এর ধরনের উপর ভিত্তি করে এর ব্যান্ডউইথের মান ভিন্ন ভিন্ন হয়ে থাকে। যেমনঃ

FastEthernet -----> 100 Mbps

Ethernet -----> 10 Mbps

E1 -----> 2.048 Mbps

T1 -----> 1.544 Mbps

এখন এই লিংক এর ধরণ অনুযায়ী Cost বের করার জন্য নিচের সূত্রটি ব্যবহার করা হয়।

$Cost = 10^8 / \text{bandwidth in bps}$

উদাহরণস্বরূপ, E1 এর ক্ষেত্রেঃ

$Cost = 10^8 / 2048000$  [যেহেতু  $2.048 \text{ Mbps} = 2048000 \text{ bps}$ ]

$= 48.83$

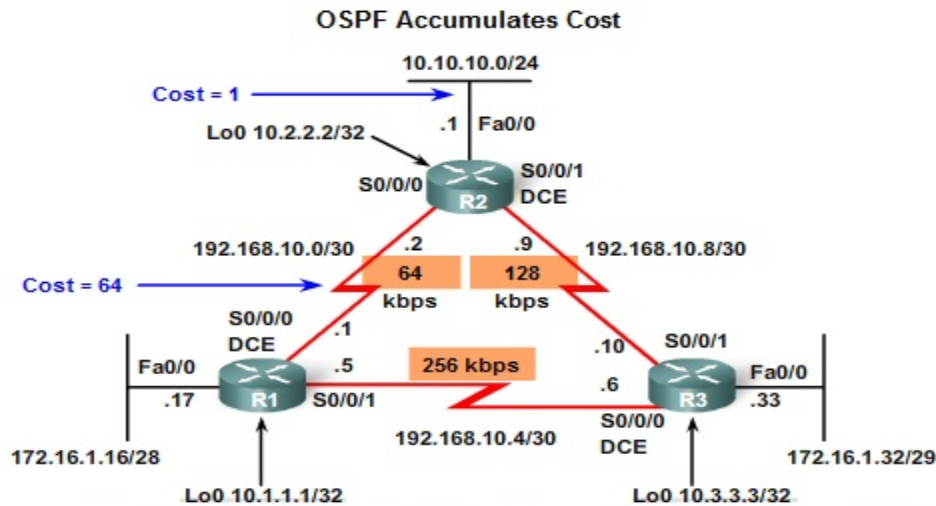
$= 48$  (রাউন্ড ফিগার)

অনুরূপভাবে হিসেব করলে অন্যান্য Link Type এর Cost সমূহ হবে নিম্নরূপ:

Cisco OSPF Cost Values

Interface Type	$10^8/\text{bps} = \text{Cost}$
Fast Ethernet and faster	$10^8/100,000,000 \text{ bps} = 1$
Ethernet	$10^8/10,000,000 \text{ bps} = 10$
E1	$10^8/2,048,000 \text{ bps} = 48$
T1	$10^8/1,544,000 \text{ bps} = 64$
128 kbps	$10^8/128,000 \text{ bps} = 781$
64 kbps	$10^8/64,000 \text{ bps} = 1562$
56 kbps	$10^8/56,000 \text{ bps} = 1785$

এখন একটা প্রশ্ন হলো যে, যদি FastEthernet এর Cost 1 হয়, তাহলে GigEthernet বা 10GigEthernet এর Cost কত হবে? উত্তর হলো, এদের Cost ও হবে 1। কারণটা হলো, আশির দশকে যখন OSPF আবির্ভূত হয় ঐ সময় GE, 10GE, 100GE এর চিন্তা কেউ করে নাই। ঐ সময়ে যে সূত্রের উপর ভিত্তি করে যেভাবে ক্যালকুলেশন করা হতো এখনো সেভাবেই করা হচ্ছে। তবে সব কথার শেষ কথা হলো, একজন এ্যাডমিনিষ্ট্রেটর চাইলে তার ইচ্ছানুযায়ী কোন লিংক এর Cost স্ট্যাটিকভাবে নির্ধারণ করে দিতে পারেন।



চিত্রানুযায়ী, R1 থেকে 10.10.10.0/24 নেটওয়ার্কে যাওয়ার মোট Cost হলো 65। কারণ, 10.10.10.0/24 নেটওয়ার্কটি R2 এর FastEthernet লিংক যার Cost হলো 1। অন্যদিকে R1 ও R2 এর মধ্যবর্তী সংযোগটি হলো একটি T1 লিংক যার Cost হলো 64। তাই 10.10.10.0/24 নেটওয়ার্কে যেতে R1 এর মোট Cost হলো 64+1 বা 65।