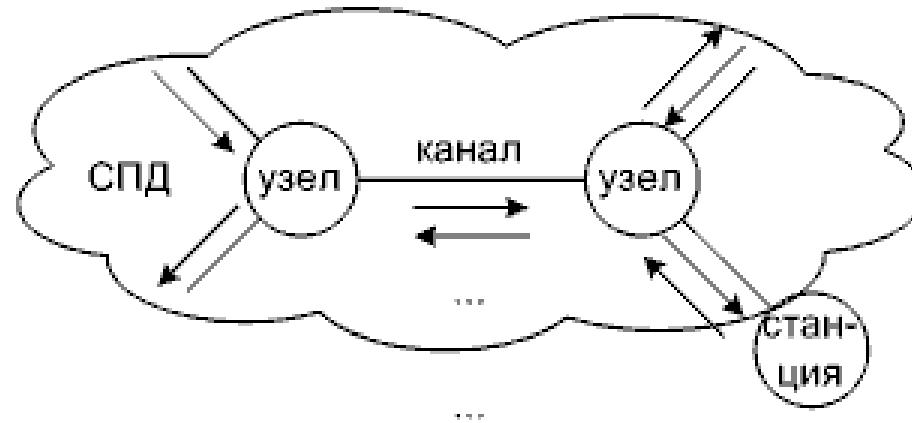


МАРШРУТИЗАЦИЯ IPv4

2.0.1.1

Понятие маршрутизации относится к функционированию СПД.



(На этом рисунке лучше не использовать УГО конкретных сетевых устройств.)

Канал (в традиционной телефонии channel, в КС link) представляет собой СрПД, через которую передаются пакеты.

Узел (так же как и станция, node) представляет собой некоторое устройство, выполняющее прием, передачу или ретрансляцию пакетов.

Узлами и станциями могут быть самые разные устройства.

2.0.1.2

Все узлы делят на два типа:

1. Пассивные (passive).
2. Активные (active).

Пассивность узла означает, что он не выполняет анализ или обработку пакетов.

Активность подразумевает, что пакеты анализируются или обрабатываются.

2.0.1.3

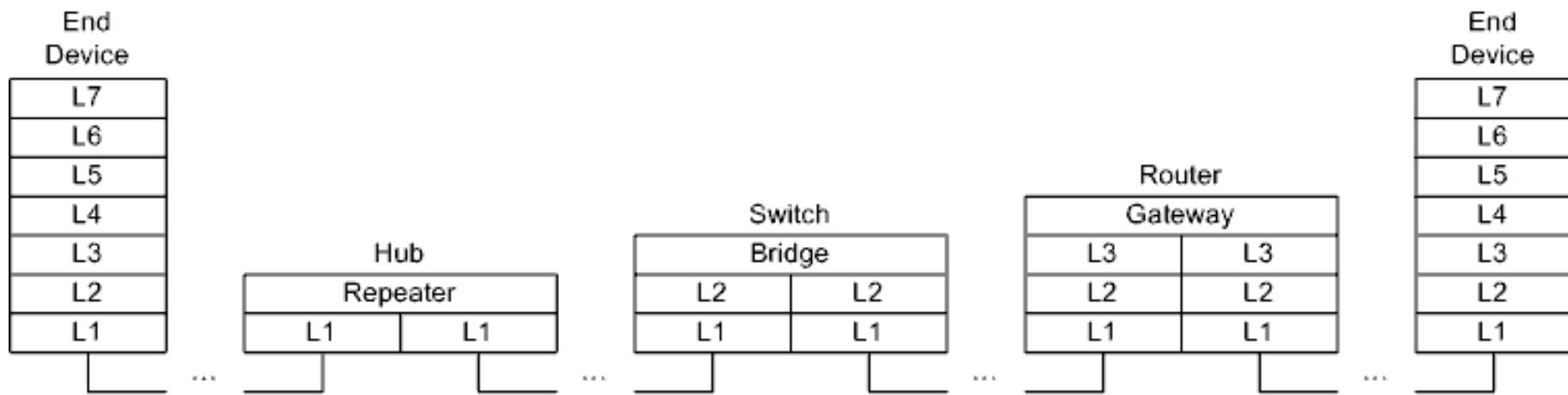
Если соотносить узлы с моделью OSI, то можно выделить:

1. *Повторители* (repeaters) -- аппаратно «срашивают» СПД на физическом уровне, типичными представителями являются *оконечные концентраторы* (hubs) (уже не производят).
2. *Мосты* (bridges) -- аппаратно (но есть и «интеллектуальные») «срашивают» СПД на канальном уровне, типичными современными представителями являются коммутаторы (switches).
3. *Шлюзы* (gateways) -- аппаратно и программно «срашивают» СПД на сетевом уровне, типичными представителями являются *маршрутизаторы* (routers).

Функция маршрутизации выполняется собственно маршрутизаторами. Но, нужно учитывать, что и все оконечные устройства должны иметь подсистему маршрутизации.

Между IP-шлюзом и IP-маршрутизатором можно поставить знак равенства. IP-маршрутизатор является частным случаем маршрутизатора.

2.0.1.4



Повторители, мосты, шлюзы и модель OSI

2.0.1.5

Новые условные графические обозначения.



-- оконечный концентратор



-- МОСТ

2.0.1.6

Физические порты маршрутизаторов ограничивают *широковещательные домены* (broadcast domains).

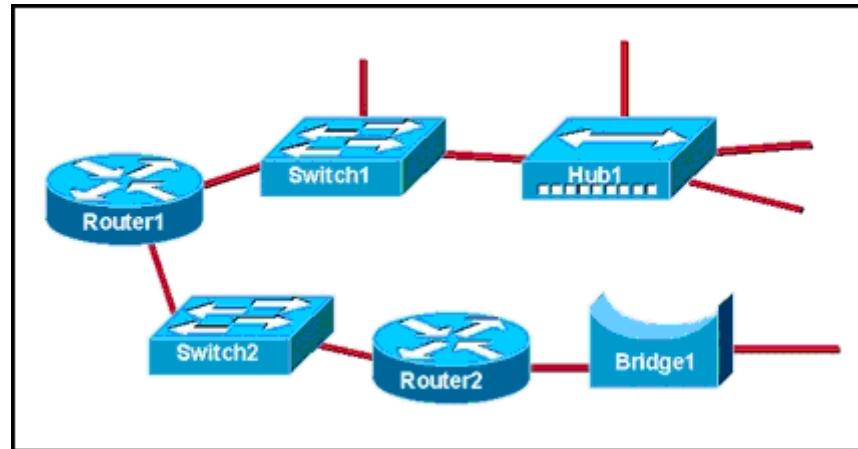
Физические порты коммутаторов ограничивают *домены коллизий* (collision domains).

Оконечные концентраторы входят в домены коллизий и широковещательные домены.

2.0.1.7

Сколько здесь широковещательных доменов?

Сколько здесь доменов коллизий?



Вопрос из CCNA

2.0.1.8

Два узла, подключенные к одному и тому же каналу называют *соседними* (*neighbor*).

Понятие соседства очень важно при маршрутизации, так как маршрутизаторы могут «перепасовывать» пакеты только своим соседям.

2.0.1.9

Алгоритмы маршрутизации реализовать полностью аппаратно невозможно (если не брать в расчет уже давно устаревшие самые примитивные из них). Поэтому под *аппаратной маршрутизацией* в настоящее время подразумевают ту, которая выполняется специализированным «железом».

Применительно к IP-маршрутизации, алгоритмы выражены в виде соответствующих протоколов семейства TCP/IP.

2.0.2.1

Маршрутизацией занимается ядро сетевой ОС, возможно встроенной (например, Cisco IOS).

Маршрут (route) -- это путь, по которому пакет передается от станции-отправителя к станции-получателю или составная часть этого пути.

Выделяют три вида маршрутов:

1. Маршрут до станции (сетевого интерфейса).
2. Маршрут до подсети.
3. Маршрут по умолчанию.

2.0.2.2

Маршруты хранятся в специальной таблице, называемой *таблицей маршрутизации (routing table)*.

В обобщенном виде, с теми или иными вариациями, таблицу можно представить следующим образом.

| | Destination | Netmask | Gateway | Interface | Metric | Options |
|-------|-------------|---------|---------|-----------|--------|---------|
| Route | | | | | | |
| Route | | | | | | |
| ... | | | | | | |
| Route | | | | | | |

2.0.2.3

Назначение полей:

1. Destination -- адрес назначения -- IP-адрес, определяющий вид маршрута.
2. Netmask -- маска подсети -- дополняет адрес назначения с целью его правильной интерпретации (если не указана, то подставляется стандартная).
3. Gateway -- шлюз -- IP-адрес шлюза-соседа, которому нужно передать пакет (если пакет не покидает пределы подсети, то шлюз не нужен).
4. Interface -- интерфейс -- IP-адрес или другой параметр, однозначно определяющий сетевой интерфейс, который должен физически «выдать» пакет в канал.
5. Metric -- метрика -- определяет приоритетность маршрута (основное назначение), часто рассматривают в совокупности с так называемой *административной дистанцией* (administrative distance).
6. Options -- опции -- специфические опции данной реализации.

2.0.2.4

Специальные соглашения в области IP-маршрутизации:

1. Адрес назначения 0.0.0.0 -- маршрут по умолчанию.
2. Маска подсети 255.255.255.255 -- маршрут к одному сетевому интерфейсу.

2.0.2.5

Таблица маршрутизации определяет что делать с уже принятым пакетом, подлежащим ретрансляции, или имеющимся пакетом, сформированным для передачи на вышестоящих уровнях.

При наличии такого пакета, работа с таблицей маршрутизации протекает в две фазы:

1. Поиск маршрутной информации.
2. Применение маршрутной информации.

Возникает ряд вопросов о работе с таблицей маршрутизации:

1. Стоит ли искать более одного маршрута.
2. Стоит ли просматривать всю таблицу маршрутизации.
3. Что делать если найдено несколько маршрутов.
4. Что делать если не найдено ни одного маршрута.
5. Стоит ли передавать пакет более одного раза.

В ответ можно придумать разные алгоритмы, которые вполне имеют право на существование.

2.0.2.6а

В настоящее время, как де facto стандартный, применяется подход согласно принципу *наиболее точного соответствия* (best match, longest match), заключающийся в следующем:

1. Маршрут ищется путем последовательного сравнения IP-адреса назначения, считанного из заголовка пакета, с диапазонами, задаваемыми адресами назначения в связке с масками подсетей, считывамыми из строк таблицы маршрутизации.
2. При попадании (hit) маршрут считается подходящим.
3. Просматривается вся таблица маршрутизации. Конечно, этот процесс разными способами оптимизируется.
4. При наличии нескольких попаданий выбирается наиболее точный маршрут. Точность попадания определяется «размером мишени». Самым точным является маршрут к станции.
5. При одинаковой точности попадания маршрут выбирается исходя из дополнительного критерия -- метрики.
6. Маршрут по умолчанию выбирается если не найдено ни одного более точного маршрута. «Промахнуться» невозможно.
7. При отсутствии попаданий пакет уничтожается (drop).

2.0.2.6b

8. Маршрут ищется для того, чтобы его применить. Применение маршрута заключается в отправке по нему пакета. Пакет передается один раз.

9. На вопросы о том, куда и чем передавать, отвечают соответствующие поля в маршруте.

2.0.2.7

При наличии нескольких альтернативных маршрутов могут совпасть и их метрики, то есть маршруты оказываются абсолютно равноправными (надо отметить, что такое происходит довольно часто).

В некоторых реализациях это считается недопустимым, а в некоторых возникает так называемая балансировка нагрузки, точнее, **эквивалентная балансировка нагрузки** (*equal load balancing*) -- соответствующие пакеты поочередно передаются в разных направлениях.

Существует еще и **неэквивалентная балансировка нагрузки** (*unequal load balancing*) -- отличается тем, что трафик распределяется пропорционально согласно метрикам.

2.0.2.8

В IP-сетях реализованы два типа маршрутизации:

1. Статическая (static).
2. Динамическая (dynamic).

При *статической маршрутизации* таблицы формируются «вручную» или автоматически на основе указанных IP-параметров и хранятся до их «ручной» модификации.

При *динамической маршрутизации* таблицы и формируются, и модифицируются автоматически с воздействием специальных служебных протоколов, что не отменяет возможность вмешательства администратора.

2.0.2.9

В отношении протоколов динамической маршрутизации, все сетевые интерфейсы делят на:

1. *Активные* (active) -- могут использоваться при обмене маршрутной информацией.
2. *Пассивные* (passive) -- не могут использоваться при обмене маршрутной информацией.

В некоторых реализациях (например, UNIX `routed`) аналогично делят маршруты.

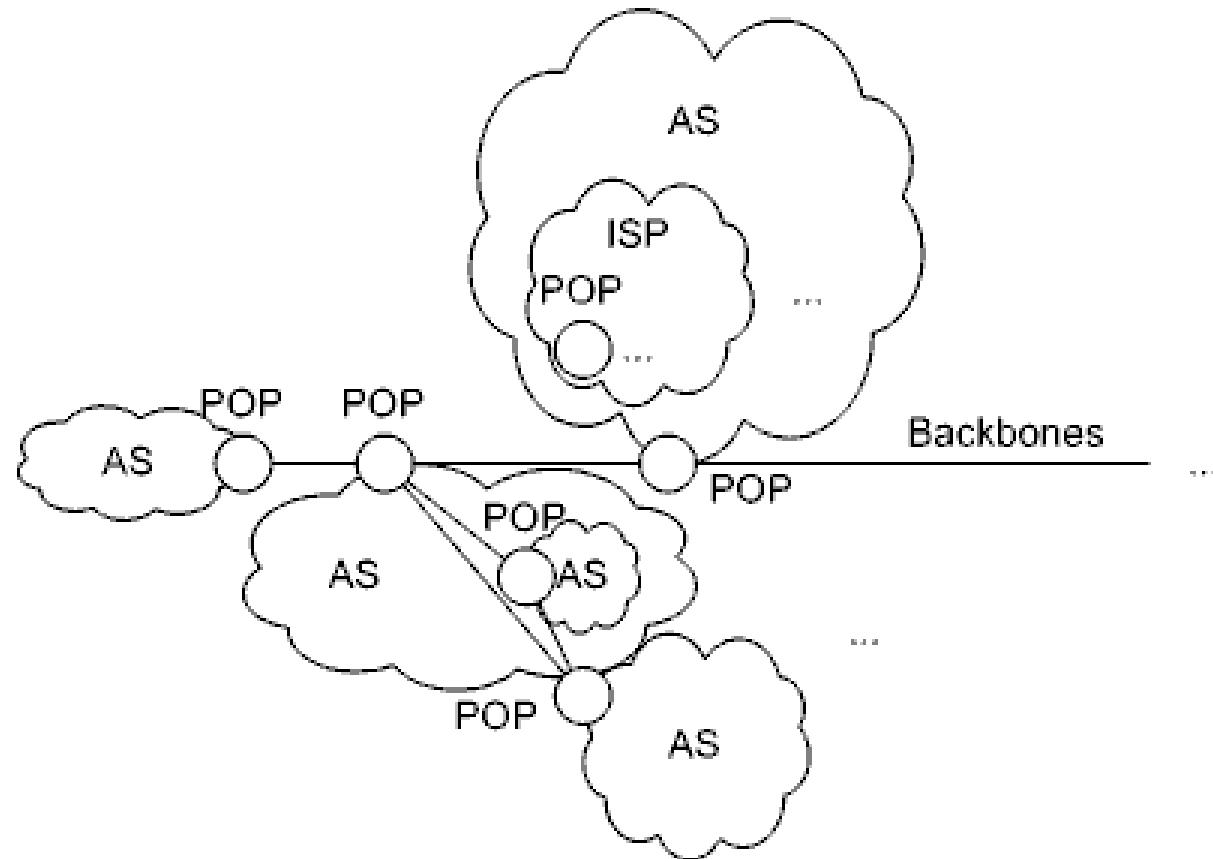
Пассивные маршруты, в отличие от *активных*, не могут быть «затронуты» (считаны или замещены) протоколами динамической маршрутизации.

В реализациях особо выделяют *постоянные* или, по-другому, *персистентные* (persistent) маршруты, которые должны сохраняться после перезагрузки.

2.0.3.1

Различные протоколы динамической маршрутизации предназначены для различных уровней в иерархии Internet. Чтобы обозначить их «место» необходимо рассмотреть структуру сети Internet.

2.0.3.2



Структура Internet

2.0.3.3

Основными структурными единицами Internet являются **автономные системы** -- Autonomous Systems (ASes).

Каждая AS выделяется исходя из наличия собственной системы маршрутизации (возможно оригинальной), то есть состояние AS не должно зависеть от состояния других ASes.

Все ASes имеют уникальные 16-битные номера.

Номера ASes поделены на:

1. Публичные (public): 1 -- 64511.
2. Приватные (private): 64512 -- 65535.

В связи с исчерпанием, не так давно были введены дополнительные 32-битные номера.

2.0.3.4

ASes связаны между собой посредством базовых магистралей (backbones).

Изначально в структуре Internet была задумана и реализована одна базовая магистраль, но сейчас это лишь условность. Поскольку на практике далеко не всегда удавалось осуществить непосредственное примыкание той или иной AS к базовой магистрали, к настоящему времени возникла очень сильная фрагментация.

Реально, ASes соединены друг с другом через так называемые *пиринговые точки* (peering points) или, по-другому, *точки присутствия* -- Points-Of-Presence (POPs).

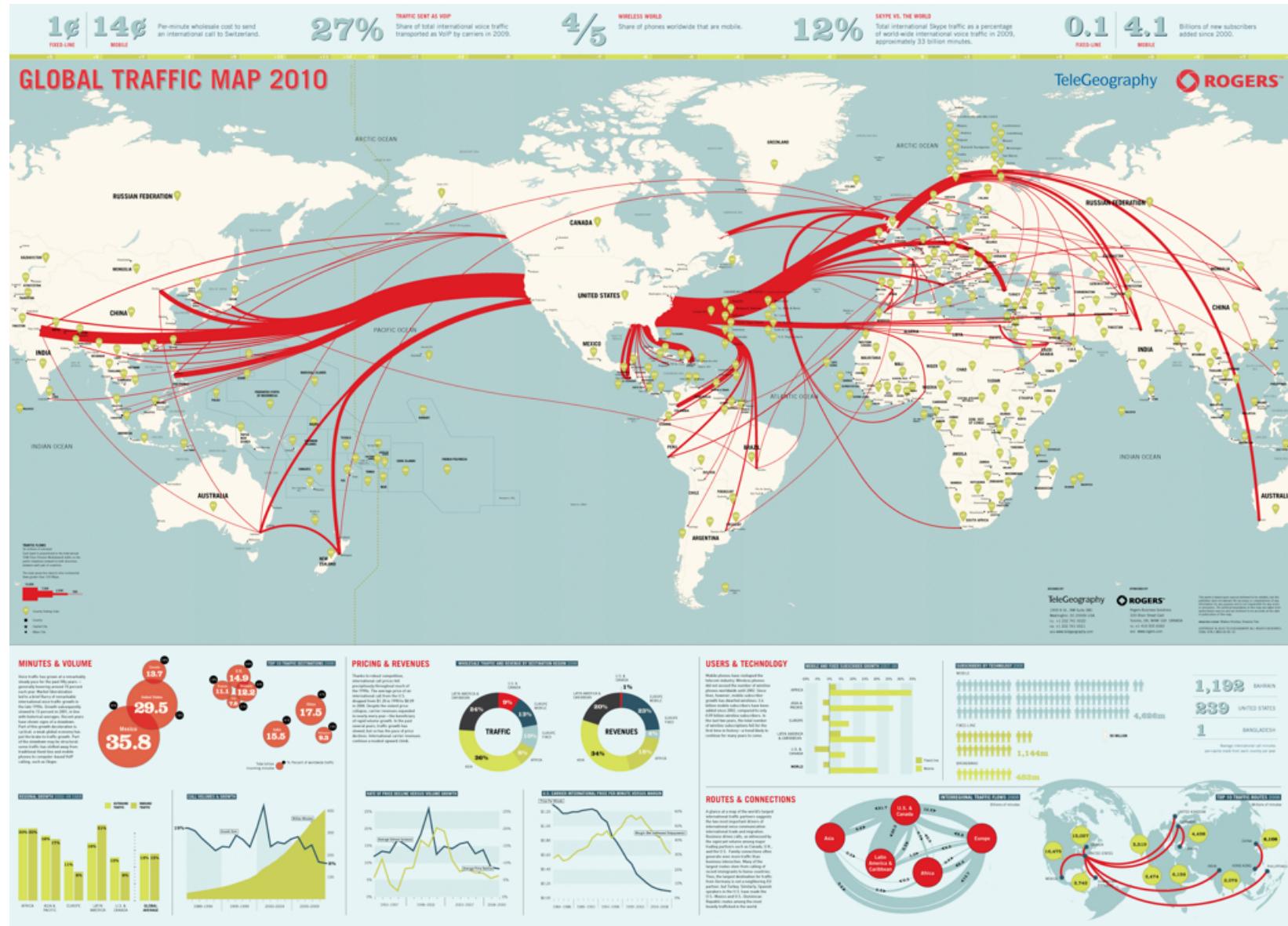
2.0.3.5

Внутри ASes работают *провайдеры* -- Internet Service Providers (ISPs).

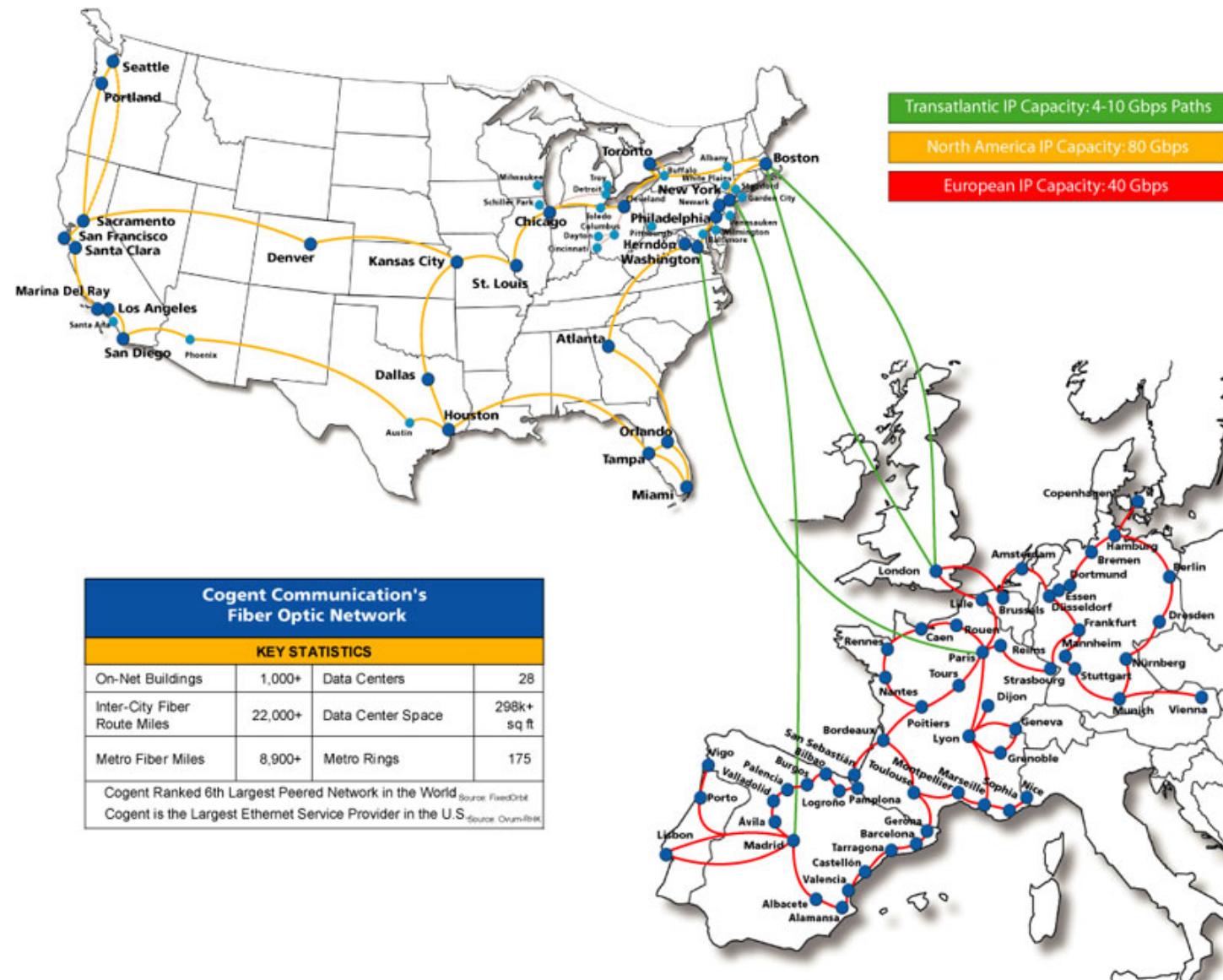
Касательно POPs, следует уточнить, что терминологически это, в первую очередь, точки предоставления коммуникационных услуг пользователям Internet.

Также следует отметить, что крупные телекоммуникационные компании могут обладать несколькими ASes, а их СПД могут иметь межконтинентальную протяженность.

2.0.3.6a

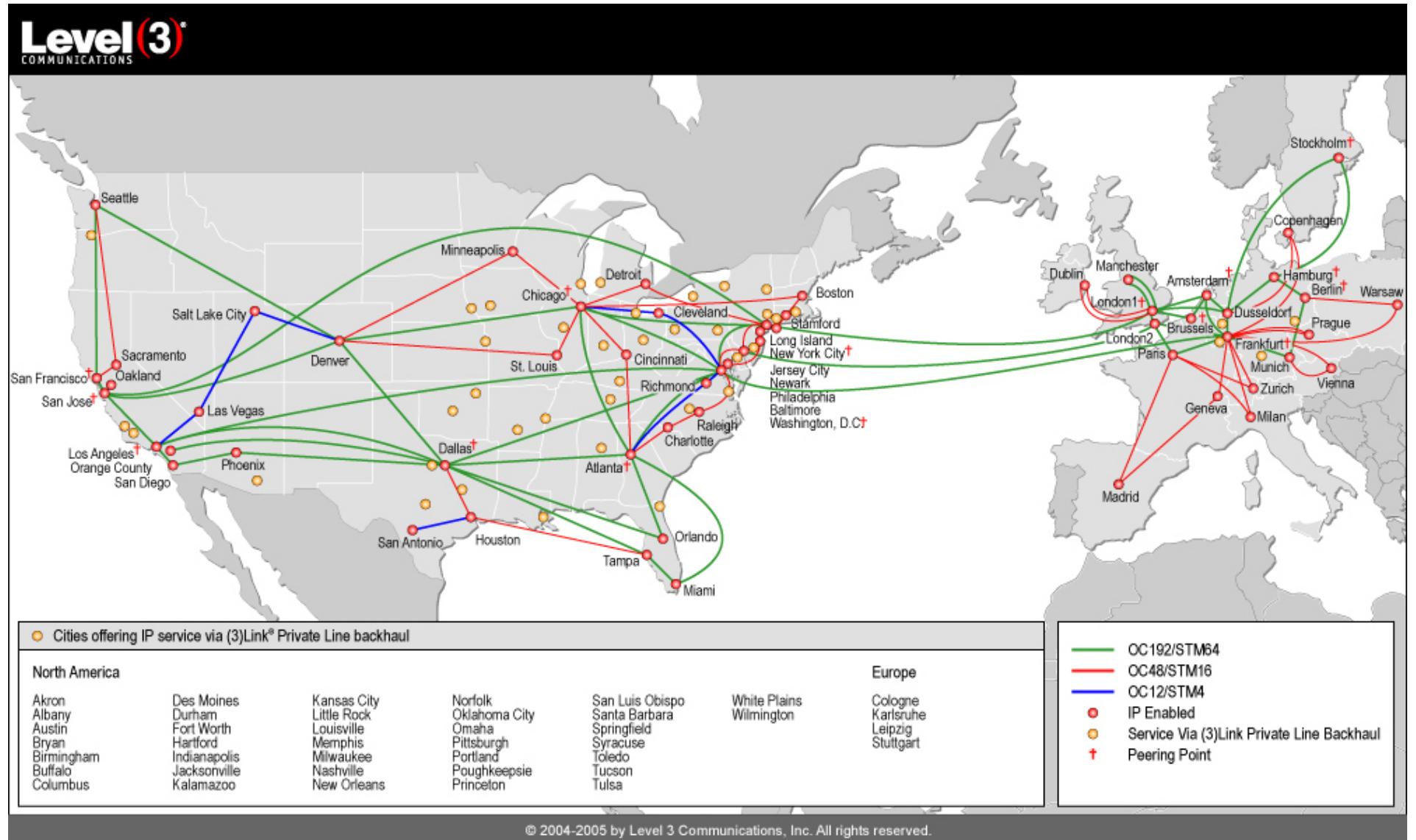


2.0.3.6b



Cogent 2006 [Cogent]

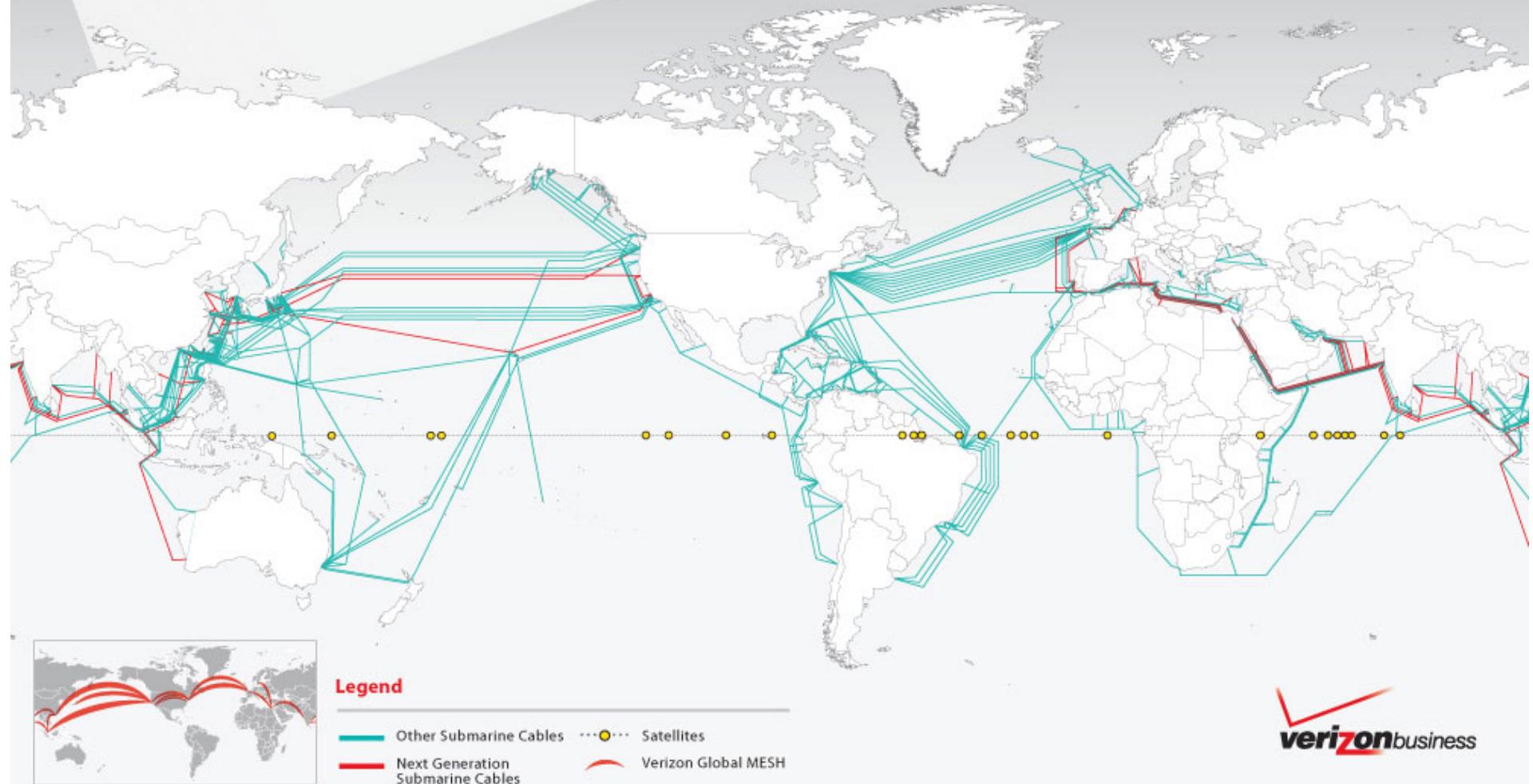
2.0.3.6c



Level(3) 2005 [Level (3)]

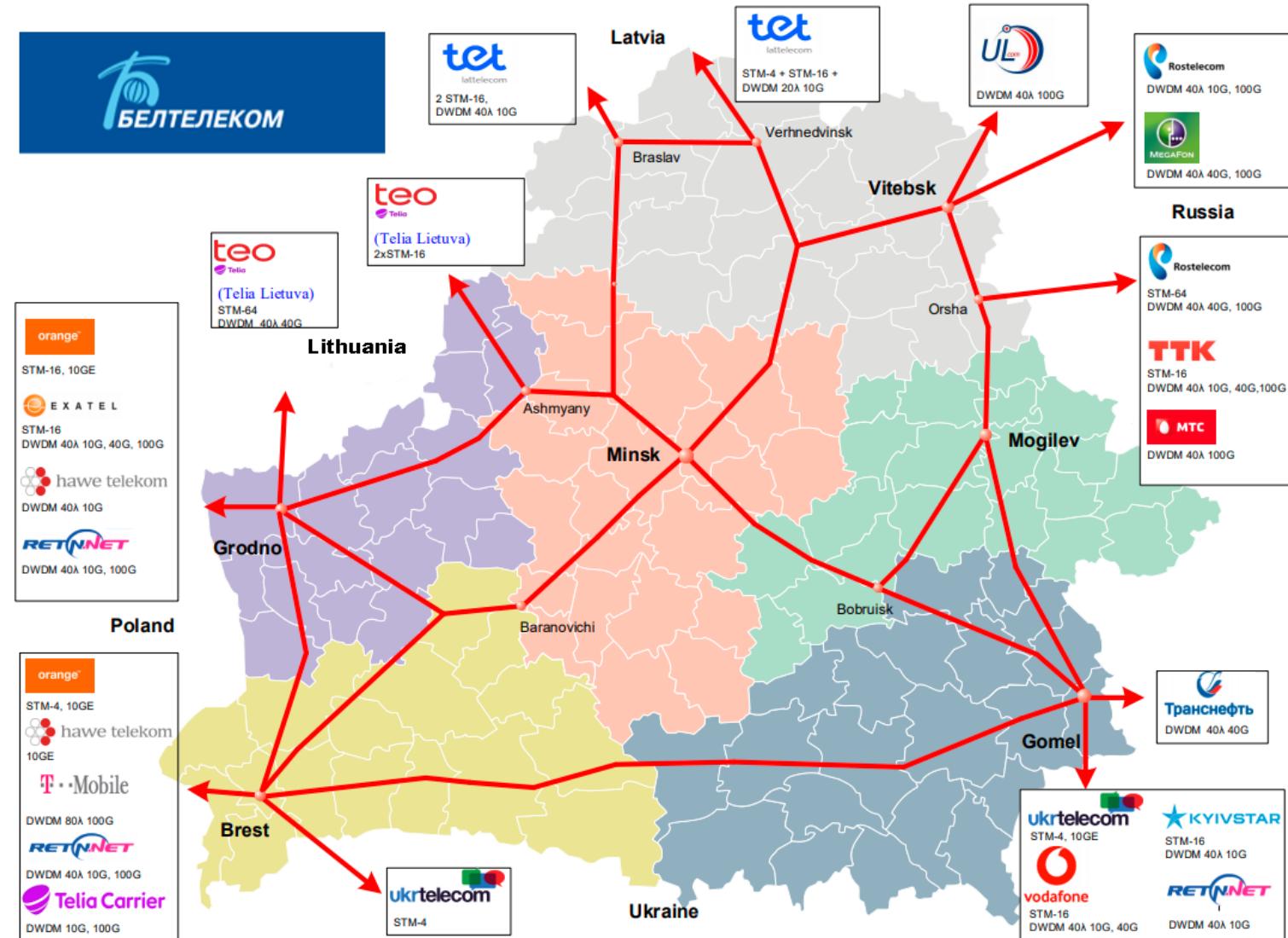
2.0.3.6d

The Verizon Global Network



Verizon 2010 [Verizon]

2.0.3.6e



Белтелеком 2020 [Белтелеком]

2.0.3.6f


Quick Links

[BGP Toolkit Home](#)

[BGP Prefix Report](#)

[BGP Peer Report](#)

[Exchange Report](#)

[Bogon Routes](#)

[World Report](#)

[Multi Origin Routes](#)

[DNS Report](#)

[Top Host Report](#)

[Internet Statistics](#)

[Looking Glass](#)

[Network Tools App](#)

[Free IPv6 Tunnel!](#)

[IPv6 Certification](#)

[IPv6 Progress](#)

[Going Native](#)

[Contact Us](#)

**HURRICANE ELECTRIC
INTERNET SERVICES**

Search

AS6697 Republican Unitary Telecommunication Enterprise Beltelecom

[AS Info](#)
[Graph v4](#)
[Graph v6](#)
[Prefixes v4](#)
[Prefixes v6](#)
[Peers v4](#)
[Peers v6](#)
[Whois](#)
[IRR](#)
[IX](#)

Company Website:

Country of Origin:

Internet Exchanges: 11

Prefixes Originated (all): 339
Prefixes Originated (v4): 328
Prefixes Originated (v6): 11

Prefixes Announced (all): 601
Prefixes Announced (v4): 584
Prefixes Announced (v6): 17

RPKI Originated Valid (all): 313
RPKI Originated Valid (v4): 304
RPKI Originated Valid (v6): 9

RPKI Originated Invalid (all): 1
RPKI Originated Invalid (v4): 1
RPKI Originated Invalid (v6): 0

BGP Peers Observed (all): 177
BGP Peers Observed (v4): 158
BGP Peers Observed (v6): 105

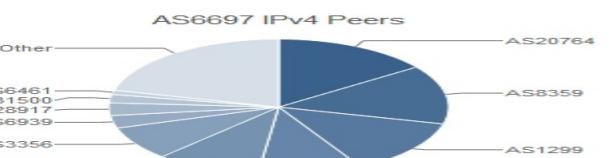
IPs Originated (v4): 1,067,520
AS Paths Observed (v4): 1,005
AS Paths Observed (v6): 857

Average AS Path Length (all): 3.473
Average AS Path Length (v4): 3.325
Average AS Path Length (v6): 3.646

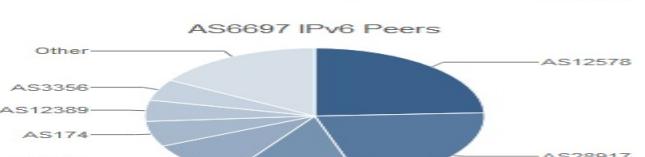
<http://www.beltelecom.by>

Belarus

**Preview
Unavailable**



| ASN | Name |
|---------|-------------------------------|
| AS20764 | CJSC RASCOM |
| AS8359 | MTS PJSC |
| AS1299 | Telia Company AB |
| AS12389 | PJSC Rostelecom |
| AS174 | Cogent Communications |
| AS3356 | Level 3 Parent, LLC |
| AS6939 | Hurricane Electric LLC |
| AS28917 | LLC "TRC FIORD" |
| AS31500 | Global Network Management Inc |
| AS6461 | Zayo Bandwidth |



| ASN | Name |
|---------|------------------------|
| AS12578 | SIA Tet |
| AS28917 | LLC "TRC FIORD" |
| AS6939 | Hurricane Electric LLC |
| AS1299 | Telia Company AB |
| AS174 | Cogent Communications |
| AS12389 | PJSC Rostelecom |
| AS3356 | Level 3 Parent, LLC |





AS 6697 (Beltelecom) (март 2021 г.) [Hurricane Electric]

2.0.3.6g

Country Info

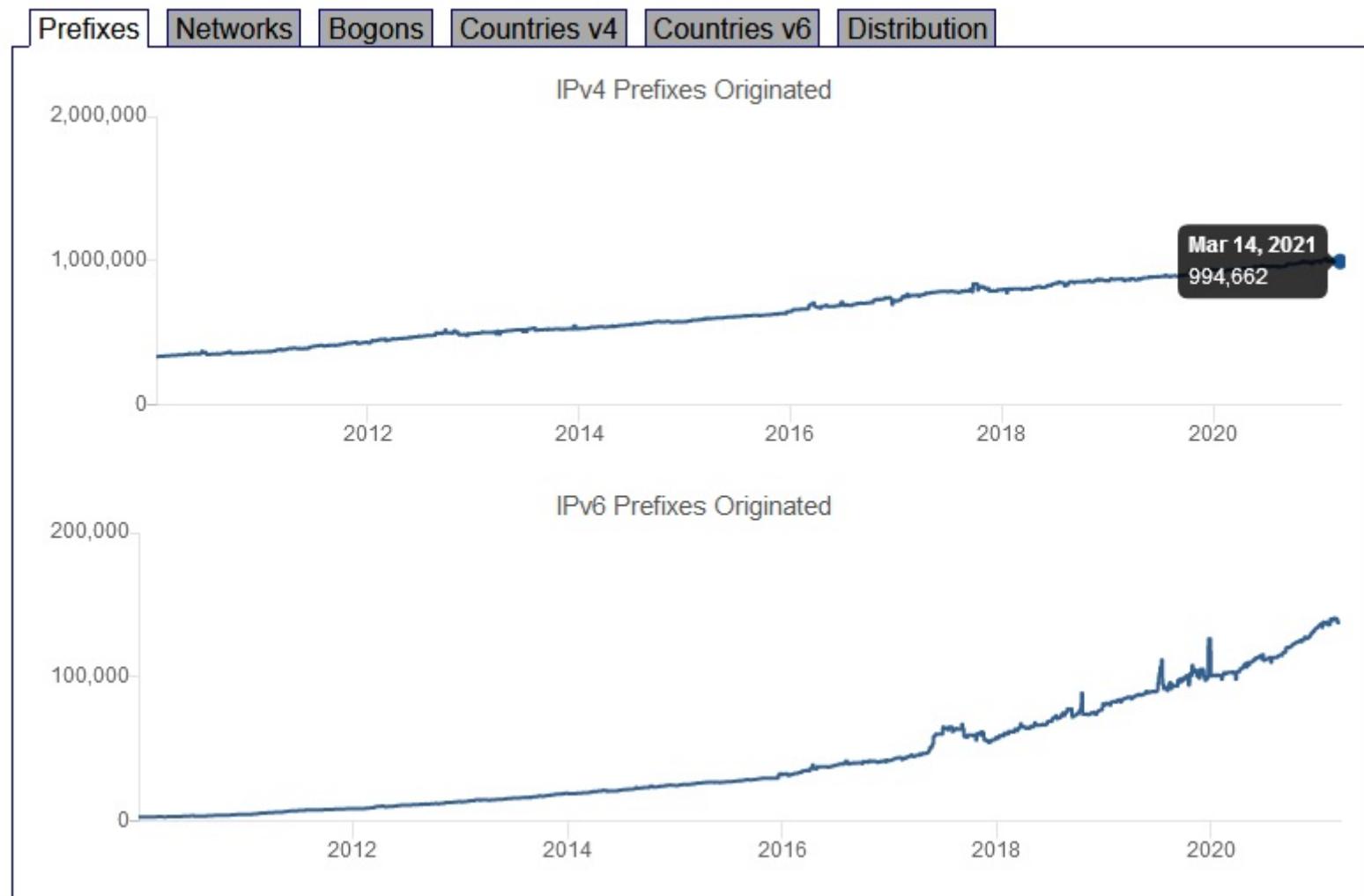
Networks: Belarus

| ASN | Name | Adjacencies v4 | Routes v4 | Adjacencies v6 | Routes v6 |
|--------------------------|---|----------------|-----------|----------------|-----------|
| AS6697 | Republican Unitary Telecommunication Enterprise Beltelecom | 158 | 584 | 105 | 17 |
| AS60280 | Republican Unitary Enterprise "National Traffic Exchange Center" | 129 | 584 | 7 | 57 |
| AS60330 | Belarusian Cloud Technologies JLLC | 34 | 327 | 14 | 43 |
| AS12406 | Business Network Ltd | 33 | 148 | 13 | 25 |
| AS42772 | Unitary enterprise A1 | 13 | 206 | 1 | 5 |
| AS25106 | Mobile TeleSystems JLLC | 11 | 165 | 3 | 21 |
| AS21274 | State Scientific Enterprise 'United Institute of Informatics Problems of National Academy of Sciences of Belarus' (UIIP NASB) | 10 | 20 | 3 | 1 |
| AS28849 | JSC GLOBALONEBEL | 6 | 15 | 0 | 0 |
| AS56740 | DataHata Ltd | 5 | 27 | 3 | 5 |
| AS205820 | Unitary enterprise A1 | 5 | 5 | 3 | 1 |
| AS33973 | Epam Systems FLLC | 4 | 6 | 0 | 0 |
| AS209974 | ITGLOBALCOM BEL LLC | 4 | 10 | 2 | 1 |
| AS31143 | COSMOS TV JLLC | 3 | 11 | 1 | 1 |
| AS21305 | IP TelCom | 3 | 32 | 1 | 1 |
| AS207587 | FE IBA IT Park | 3 | 2 | 2 | 1 |
| AS62170 | JSC "BPS-Sberbank" | 2 | 1 | 0 | 0 |
| AS61277 | ScienceSoft CJSC | 2 | 1 | 0 | 0 |
| AS60744 | A.V. Luikov Heat and Mass Transfer Institute of the National Academy of Sciences of Belarus | 2 | 3 | 0 | 0 |
| AS60367 | "Seti Plus" Ltd. | 2 | 6 | 0 | 0 |
| AS5498 | Institution Central Information and Analytical Center at the Ministry of Education of Belarus | 2 | 4 | 0 | 0 |
| AS51276 | Limited Responsibility Society "KomProektServis" | 2 | 2 | 0 | 0 |

Academy of Public Administration

Беларуские автономные системы (фрагмент) (март 2021 г.) [Hurricane Electric]

2.0.3.6h



Некоторая статистика по автономным системам IPv4 и IPv6 [Hurricane Electric]

2.0.3.7

Основные надгосударственные организации, управляющие Internet:

1. ISOC (Internet SOCIETY) -- Internet-сообщество (главная организация).
2. IAB (Internet Architecture Board) -- совет по архитектуре Internet.
3. IETF (Internet Engineering Task Force) -- рабочая группа по Internet-инженерии.
4. IGF (Internet Governance Forum) -- форум по управлению Internet.
5. IRTF (Internet Research task Force) -- рабочая группа по исследованию Internet.
6. ICANN (Internet Corporation for Assigned Names and Numbers) -- корпорация по назначению названий и номеров в Internet.

2.0.3.8

WHO RUNS THE INTERNET?

NO ONE PERSON, COMPANY, ORGANIZATION OR GOVERNMENT RUNS THE INTERNET.

The Internet itself is a globally distributed computer network comprised of many voluntarily interconnected autonomous networks. Similarly, its governance is conducted by a decentralized and international multi-stakeholder network of interconnected autonomous groups drawing from civil society, the private sector, governments, the academic and research communities, and national and international organizations. They work cooperatively from their respective roles to create shared policies and standards that maintain the Internet's global interoperability for the public good.

WHO IS INVOLVED:

IAB ACP SR
INTERNET ARCHITECTURE BOARD
Oversees the technical and engineering development of the IETF and IRTF.
www.iab.org

ICANN COP YV
INTERNET CORPORATION FOR ASSIGNED NAMES AND NUMBERS
Coordinates the Internet's systems of unique identifiers: IP addresses, protocol parameter registries, top-level domain space (DNS root zone).
www.icann.org

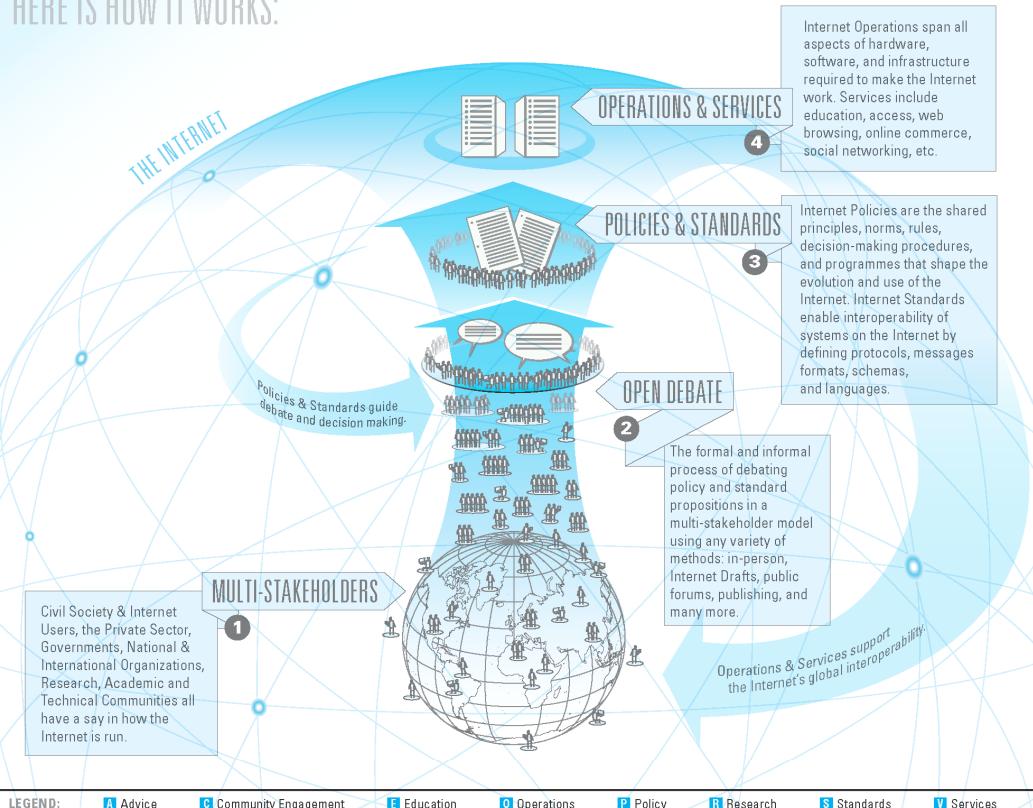
IETF C P S
INTERNET ENGINEERING TASK FORCE
Develops and promotes a wide range of Internet standards dealing in particular with standards of the Internet protocol suite. Their technical documents influence the way people design, use, and manage the Internet.
www.ietf.org

IGF A CP
INTERNET GOVERNANCE FORUM
A multi-stakeholder open forum for debate on issues related to Internet governance.
www.intgovforum.org

IRTF R
INTERNET RESEARCH TASK FORCE
Promotes research of the evolution of the Internet by creating focused, long-term research groups working on topics related to Internet protocols, applications, architecture and technology.
www.irtf.org

GOVERNMENTS AND INTER-GOVERNMENTAL ORGANIZATIONS C P
Develop laws, regulations and policies applicable to the Internet within their jurisdictions; participants in multilateral and multi-stakeholder regional and international fora on Internet governance.

HERE IS HOW IT WORKS:



WHO IS INVOLVED:

ISO 3166 MA S
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, MAINTENANCE AGENCY
Defines names and postal codes of countries, dependent territories, special areas of geographic significance.
www.iso.org/iso/country_codes.htm

ISOC C E PY
INTERNET SOCIETY
Assure the open development, evolution and use of the Internet for the benefit of all people throughout the world. Currently ISOC has over 90 chapters in around 80 countries.
www.internet-society.org

RIRs O P V
5 REGIONAL INTERNET REGISTRIES
Manage the allocation and registration of Internet number resources, such as IP addresses, within geographic regions of the world.
www.afrinic.net
www.apnic.net
www.arin.net
www.lacnic.net
wwwripe.net
Africa
Asia Pacific
Canada & United States
Latin America & Caribbean
Europe, the Middle East & parts of Central Asia

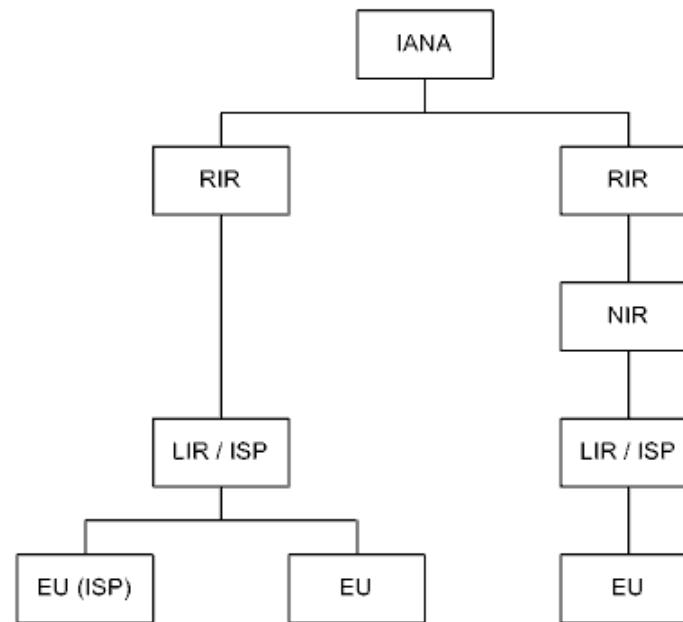
W3C S
WORLD WIDE WEB CONSORTIUM
Create standards for the world wide web that enable an Open Web Platform, for example, by focusing on issues of accessibility, internationalization, and mobile web solutions.
www.w3.org

INTERNET NETWORK OPERATORS' GROUPS A O V
Discuss and influence matters related to Internet operations and regulation within informal fora made up of Internet Service Providers (ISPs), Internet Exchange Points (IXPs), and others.

2.0.3.9

Распределение и регистрация адресов и номеров (например, номеров ASes) находится в ведении IANA (Internet Assigned Numbers Authority) под контролем ICANN.

IPv4- и IPv6-адреса выделяют блочно с последовательным делегированием прав следующим образом.



Где: IR -- Internet Registry, RIR -- Regional IR, NIR -- National IR, LIR -- Local IR, EU -- End User.

2.0.3.10

В современной сети Internet выделяют пять регионов.



Где:

AfriNIC -- Africa Network Information Center,

APNIC -- Asia Pacific Network Information Center,

ARIN -- American Registry for Internet Numbers,

LACNIC -- Latin American and Caribbean Network Information Center,

RIPE NCC -- Reseaux IP Europeens Network Coordination Centre.

[IANA]

2.0.4.1

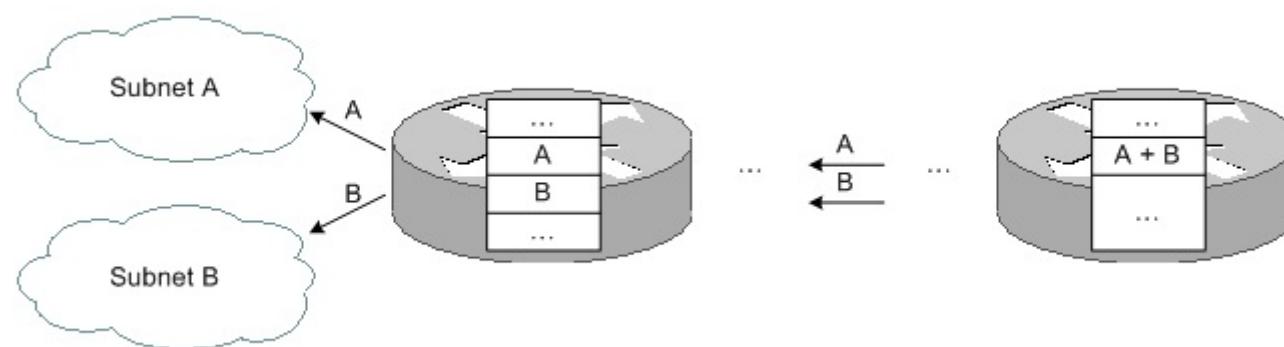
Суть всех протоколов динамической маршрутизации заключается в реализации тех или иных алгоритмов обмена маршрутами к подсетям, с целями как оптимизации трафика, так и вообще нахождения абонентов.

Обмен происходит именно маршрутами к подсетям.

Основной смысл разбиения на подсети состоит в упрощении таблиц маршрутизации.

Вместо того чтобы отслеживать станции и направлять пакет каждой из них «персонально», пакет направляется сразу в подсеть.

Также упрощение достигается за счет *агрегации маршрутов* (route aggregation) -- получение более общего маршрута из отдельных маршрутов к нескольким подсетям, если направления к этим подсетям совпадают.



2.0.4.2

Реально агрегация происходит путем *суммирования маршрутов* (route summarization). Суммирование может быть:

1. «Ручным» (manual) -- выполняется администратором (причем обычно выполняется оптимально).
2. Автоматическим (auto) -- выполняется реализацией протокола динамической маршрутизации (причем обычно выполняется «округление» до ближайшей большей подсети стандартного размера, что не всегда правильно).

2.0.4.3

С каким маршрутом оптимально суммируется маршрут к подсети 192.168.11.64?

Какой маршрут получается в результате?

2.0.4.4

Как на уровне базовых магистралей, так и в пределах AS, допускается одновременное применение нескольких протоколов динамической маршрутизации.

Шлюзы в пределах ASes называют *внутренними* (interior), а шлюзы, через которые ASes подключены к базовым магистралям -- *внешними* (exterior).

Соответственно, протоколы для внутренних шлюзов называют IGP^s (Interior Gateway Protocols), а для внешних -- EGP^s (Exterior Gateway Protocols).

2.0.4.5

За относительно долгую историю КС было придумано много алгоритмов маршрутизации, но практически все реально эксплуатируемые в IP-сетях протоколы динамической маршрутизации относят к группе адаптивных, причем двух типов:

1. Distance Vector Algorithms (DVAs) -- алгоритмы, основанные на анализе векторов расстояний.
2. Link State Algorithms (LSAs) -- алгоритмы, основанные на анализе состояния связей.

DVAs при выборе маршрутов оценивают расстояние до подсетей. Касательно пересылки пакетов, расстояние в КС принято измерять в холах. Один хол (hop) -- это изначальная передача либо одна последующая ретрансляция пакета. Базовым является алгоритм Беллмана-Форда (Bellman-Ford).

LSAs при выборе маршрутов оценивают состояние связей, то есть каналов. Классическим примером состояния канала является его пропускная способность. Базовым является алгоритм Дийкстры (Dijkstra).

2.0.4.6

Поддержку подсетей нестандартного размера при IP-маршрутизации называют бесклассовой междоменной маршрутизацией -- Classless Inter-Domain Routing (CIDR) (RFC 4632).

В случае бесклассового протокола, для учета нестандартных размеров подсетей при передаче IP-адресов подсетей дополнительно передаются и маски.

2.0.4.7

| | IGP | | EGP | |
|-------------------|--|--------------------------|--------|-----|
| | DVA | LSA | DVA | LSA |
| IPv4 Classful | RIPv1 EIGRP (Cisco) | – | – | EGP |
| IPv4 Classless | RIPv2 EIGRP (Cisco) | OSPFv2 IS-IS | BGPv4 | – |
| IPv6 | RIPng EIGRP for IPv6 (Cisco) | OSPFv3 IS-IS for IPv6 | BGPv4+ | – |

Cisco включает все EGP-протоколы в отдельную группу под названием Path Vector.

(Практически выведенные из эксплуатации протоколы зачеркнуты.)

(Касательно проприетарных протоколов в скобках указана компания-разработчик.)

Классификация основных протоколов динамической маршрутизации

2.0.4.8

Наиболее сложные протоколы динамической маршрутизации состоят из комплексов подпротоколов.

Классическим примером вспомогательного подпротокола является HELLO (есть разные вариации) -- позволяет устанавливать соседские отношения.

2.0.5.1

Последовательность действий при передаче пакета в некоторой подсети (**опуская** коллизии, оконный механизм, DNS) заключается в следующем:

1. Пакет с известным IP-адресом назначения в заголовке передается на уровень MAC (например, Ethernet) и выполняется инкапсуляция.
2. В нормальной ситуации ядро сетевой ОС хранит таблицу соответствия MAC и IP-адресов. Если MAC-адрес назначения не известен, то для его восстановления используется протокол ARP (RFC 826).
3. Если пакет (теперь уже кадр) предназначен станции из текущей подсети, то, после передачи сетевым интерфейсом станции-передатчика, он будет сразу принят всеми станциями подсети.
4. Причем только на станции-абоненте, на основании анализа MAC-адреса назначения, кадр будет распознан как свой и его содержимое будет передано на уровень IP для дальнейшей обработки. Остальными станциями кадр будет отброшен.
5. Если пакет предназначен станции из другой (в том числе соседней) подсети, то он будет передан, согласно таблице маршрутизации, соответствующему шлюзу с использованием MAC-адреса этого шлюза (возможно после привлечения ARP).

2.0.5.2

Если по каким-либо причинам необходимо принимать и обрабатывать все кадры, то включается специальный режим работы сетевого интерфейса --
promiscuous.

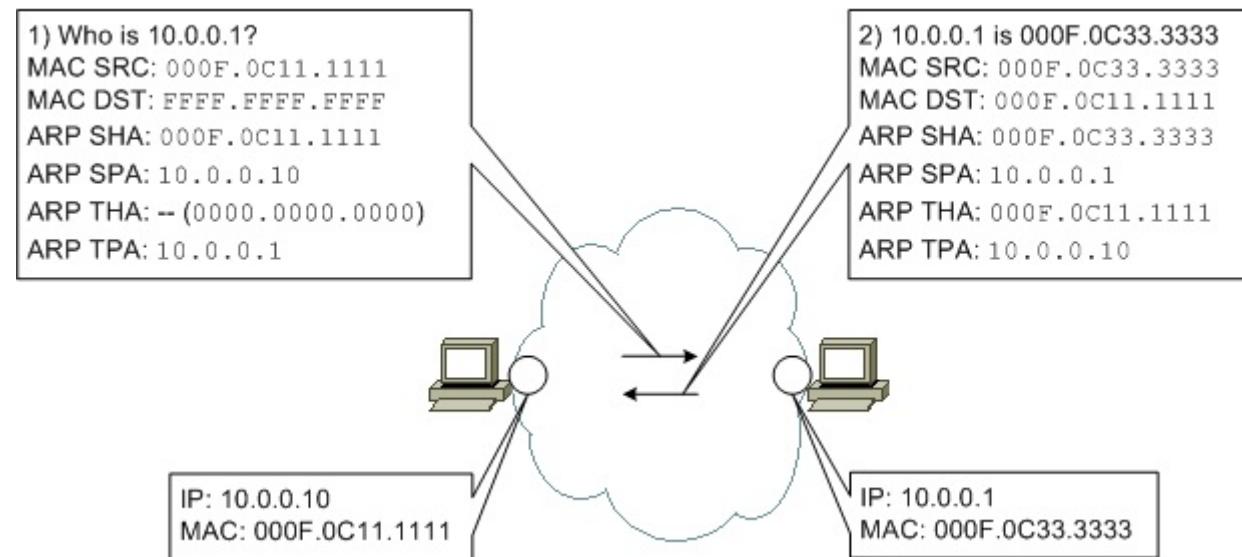
2.0.5.3

Строки ARP-таблицы могут быть:

1. Статическими (static) -- вносятся администратором (изредка) и, как правило, хранятся до перезагрузки или «ручного» удаления.
2. Динамическими (dynamic) -- вносятся ОС автоматически и, как правило, удаляются по таймеру.

Строки с постоянными (persistent) соответствиями сохраняются после перезагрузки.

2.0.5.4

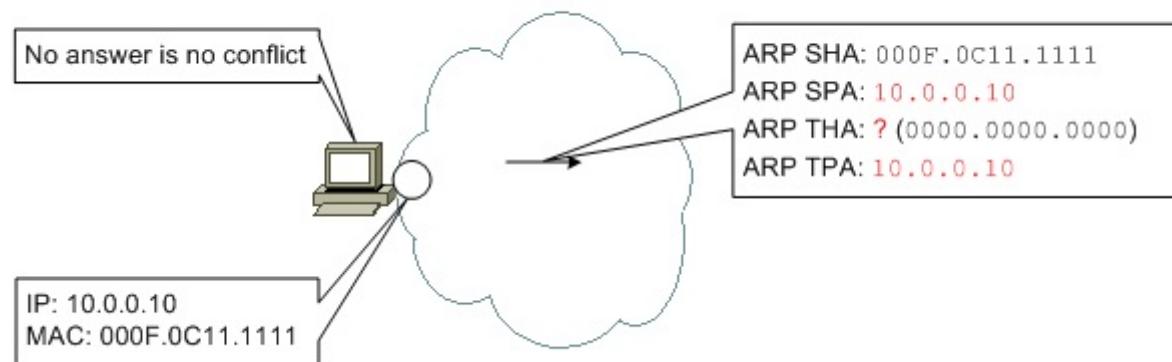


А при обновлении строк ARP-таблицы в реализациях как правило используются уже юникаст-запросы.

ARP

2.0.5.5

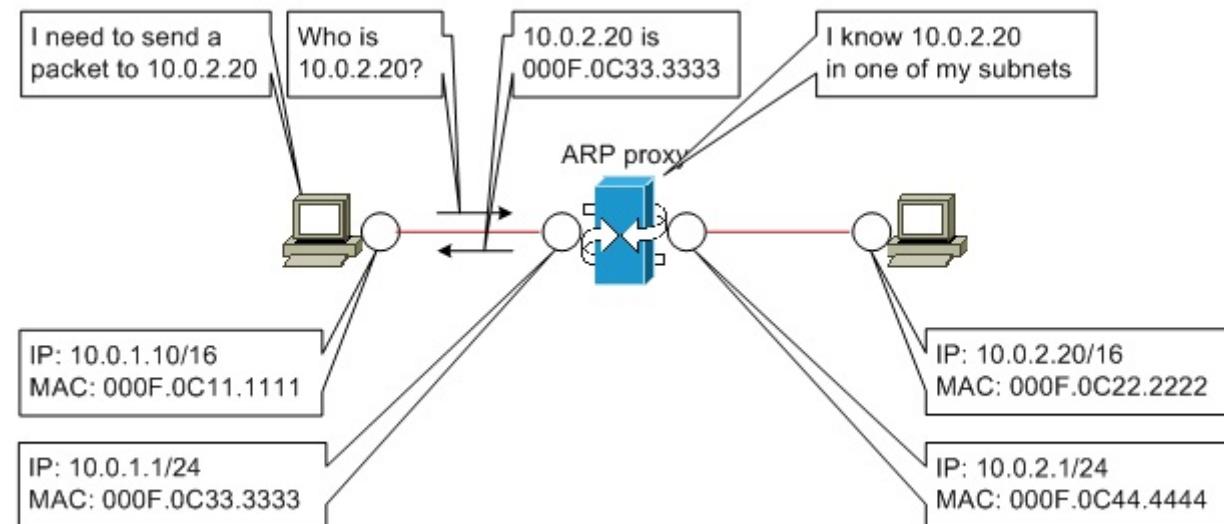
ARP-сообщения инкапсулируются в Ethernet-кадры (Length/Type = 0806h). В дополнение к запросу и ответу, предусмотрен еще один вид ARP-сообщений -- ARP probe или, по-другому, gratuitous ARP request (RFC 5227) (Op = 1, как у обычного запроса). Что позволяет, например, при загрузке ОС, обнаружить конфликты IP-адресов и параллельно оповестить все станции в подсети о «возникновении» у сетевого интерфейса нового IP-адреса (согласно RFC эти шаги должны следовать друг за другом, но на практике обычно совмещены). «Исчезновение» IP-адреса, например, при нормальном завершении работы ОС, не анонсируется.



2.0.5.6

ARP proxy (чаще proxy ARP) (RFC 1027) в связке с directed broadcast forwarding позволяет организовать прозрачный шлюз (transparent gateway).

Включение ARP proxy разрешает шлюзу отвечать на ARP-запрос из одной своей подсети в отношении IP-адреса из другой своей подсети (подставлять свой MAC-адрес). Такой запрос может возникнуть только если запрашивающая станция считает что запрашиваемая станция находится в той же подсети. Это возможно в специфичных топологиях (не совсем правильных, но массово применявшимся на ранних этапах развития КС).



2.0.6.1

Сегменты с топологией point-to-point и связанные с ними технологии канального уровня в контексте маршрутизации, в сравнении с multi-access, обладают двумя особенностями.

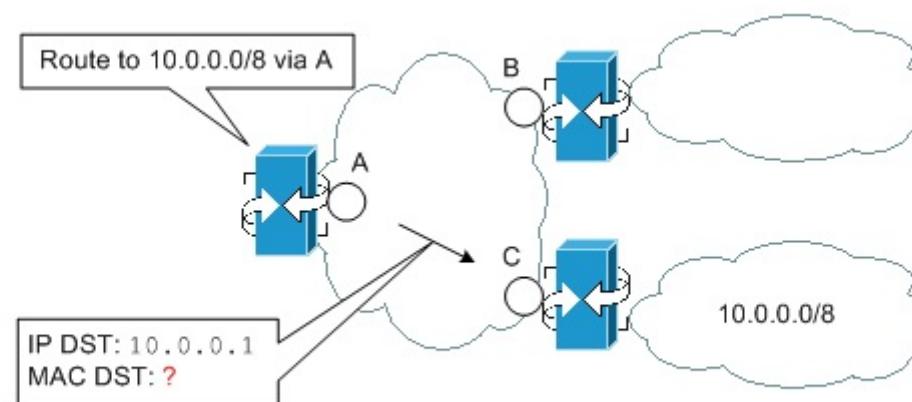
Во-первых, поскольку физически связаны только два устройства, MAC-адреса не нужны и не реализуются. Следовательно, по понятным причинам, нет необходимости в ARP.

Во-вторых, поскольку направление однозначно, при вводе добавляющей маршрут команды вместо указания IP-адреса шлюза (как это принято «канонически») можно указать выходной интерфейс -- синтаксически во всех основных ОС это допустимо.

2.0.6.2

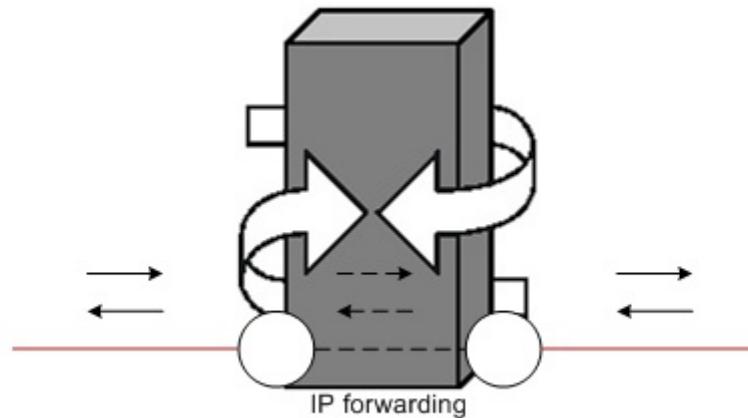
Следует отметить, что при вводе маршрута применительно к топологиям multi-access не рекомендуется указывать вместо IP-адреса шлюза выходной интерфейс.

Если попытаться это сделать, то у передающего пакет шлюза не будет никакой информации о том, какой шлюз должен этот пакет принять. Передающий шлюз сможет «зацепиться» только за IP-адрес назначения в пакете. Следовательно, требуется специальная поддержка в ОС, плюс не обойтись без ARP proxy.



2.0.7.1

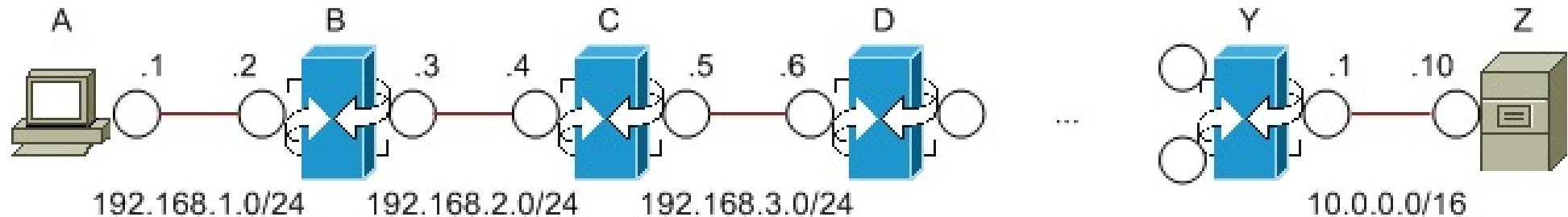
Для того чтобы обеспечить передачу транзитных пакетов между подсетями через шлюз на нем должен быть разрешен IP forwarding.



После включения IP forwarding, каждый пакет, принятый одним из сетевых интерфейсов, может быть ретранслирован другими, то есть станция работает собственно как шлюзовая.

Доступность своих интерфейсов IP forwarding не затрагивает, то есть каждый интерфейс достижим через любой другой вне зависимости от состояния IP forwarding.

2.0.8.1



| Hop | Node | Destination | Gateway | Interface |
|-----|------|-------------|---------|-----------|
| ... | ... | ... | ... | ... |
| ... | ... | ... | ... | ... |
| ... | ... | ... | ... | ... |

Построчно заполните ключевые поля маршрутов, по которым может быть передан пакет от клиента к серверу, а также ответный пакет, с указанием узлов к которым эти маршруты относятся.

Подумайте о MAC-адресах.

Задача у доски

2.0.9.1

Любая сетевая ОС, в том числе Windows и Linux, после установки уже имеет базовую таблицу маршрутизации, сформированную на основе введенных IP-параметров.

При этом в таблицу автоматически вносятся именно маршруты к своим подсетям -- при переходе соответствующих сетевых интерфейсов в работоспособное состояние.

В дальнейшем таблицу бывает нужно лишь корректировать.

2.0.9.2

Чтобы просмотреть текущую таблицу маршрутизации в Windows используют команду `route` с аргументом `print`.

Традиционная команда Linux: `netstat -r (-nr)`.

Следует отметить, что часто в реализациях при выводе на экран таблицы маршрутизации отображаются не все маршруты.

2.0.9.3

```
C:\Users\Administrator>route print
```

```
=====
Interface List
  11...00 22 4d 80 c4 27 .....Intel(R) 82579LM Gigabit Network Connection
    1.....Software Loopback Interface 1
...
=====
```

```
IPv4 Route Table
```

```
=====
Active Routes:
Network Destination      Netmask        Gateway       Interface Metric
          0.0.0.0        0.0.0.0      192.168.11.193  192.168.11.221    266
          127.0.0.0      255.0.0.0        On-link        127.0.0.1    306
          127.0.0.1  255.255.255.255        On-link        127.0.0.1    306
        127.255.255.255  255.255.255.255        On-link        127.0.0.1    306
          192.168.11.192  255.255.255.224        On-link        192.168.11.221    266
        192.168.11.221  255.255.255.255        On-link        192.168.11.221    266
        192.168.11.223  255.255.255.255        On-link        192.168.11.221    266
          224.0.0.0        240.0.0.0        On-link        127.0.0.1    306
          224.0.0.0        240.0.0.0        On-link        192.168.11.221    266
        255.255.255.255  255.255.255.255        On-link        127.0.0.1    306
        255.255.255.255  255.255.255.255        On-link        192.168.11.221    266
=====
Persistent Routes:
```

| Network Address | Netmask | Gateway Address | Metric |
|-----------------|---------|-----------------|---------|
| 0.0.0.0 | 0.0.0.0 | 192.168.11.193 | Default |

```
=====
...
```

Пример таблицы маршрутизации Windows на пользовательской станции

2.0.9.4

```
#netstat -r #Вывести на экран таблицу маршрутизации ядра
```

Kernel IP routing table

| Destination | Gateway | Genmask | Flags | MSS | window | irtt | Iface |
|-----------------------|--------------|------------------------|-----------|-----|--------|------|-------------|
| <u>192.168.11.160</u> | * | <u>255.255.255.240</u> | U | 0 | 0 | 0 | <u>eth0</u> |
| <u>192.168.11.0</u> | * | <u>255.255.255.128</u> | U | 0 | 0 | 0 | <u>eth1</u> |
| 169.254.0.0 | * | 255.255.0.0 | U | 0 | 0 | 0 | <u>eth1</u> |
| <u>127.0.0.0</u> | * | <u>255.0.0.0</u> | U | 0 | 0 | 0 | <u>lo</u> |
| <u>default</u> | 192.168.11.1 | <u>0.0.0.0</u> | <u>UG</u> | 0 | 0 | 0 | <u>eth1</u> |

#

#Флаги: U -- route is Up, G -- use Gateway

#MSS, window, irtt -- параметры TCP (устарело)

#

```
#netstat -nr #Адреса отображать в цифровой форме (не делать DNS-запросы)
```

| Destination | Gateway | Genmask | Flags | MSS | window | irtt | Iface |
|----------------|----------------|-----------------|-----------|-----|--------|------|-------------|
| 192.168.11.160 | <u>0.0.0.0</u> | 255.255.255.240 | U | 0 | 0 | 0 | <u>eth0</u> |
| 192.168.11.0 | 0.0.0.0 | 255.255.255.128 | U | 0 | 0 | 0 | <u>eth1</u> |
| 169.254.0.0 | 0.0.0.0 | 255.255.0.0 | U | 0 | 0 | 0 | <u>eth1</u> |
| 127.0.0.0 | 0.0.0.0 | 255.0.0.0 | U | 0 | 0 | 0 | <u>lo</u> |
| <u>0.0.0.0</u> | 192.168.11.1 | 0.0.0.0 | <u>UG</u> | 0 | 0 | 0 | <u>eth1</u> |

Пример таблицы маршрутизации Linux на шлюзе

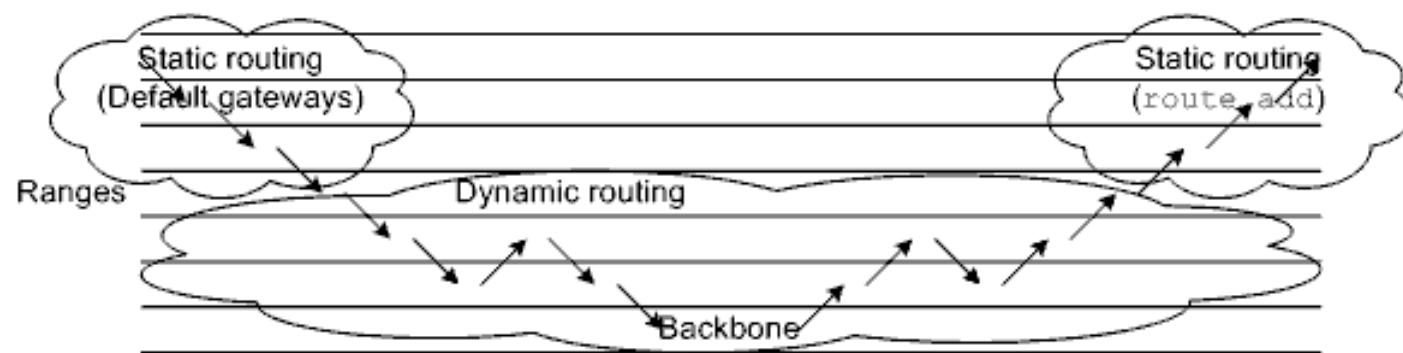
2.0.9.5

В типовом случае, на примере взаимодействия двух максимально удаленных абонентов, на границе СПД применяют статическую маршрутизацию.

Механизм шлюзов по умолчанию позволяет «спустить» пакет ближе к базовым магистралям в зону динамической маршрутизации высокоранговых подсетей.

Чтобы обеспечить «подъем» пакетов к абоненту, а также подъем ответных пакетов в зоне статической маршрутизации необходимо прописывать маршруты к соответствующим подсетям.

При необходимости, всегда существует возможность задать нужный маршрут «вручную».



2.0.9.6

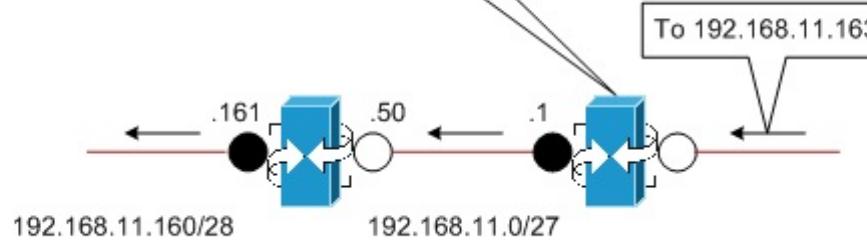
Чтобы добавить статический маршрут в таблицу маршрутизации ядра, и в Windows, и в Linux, используют команду `route` с аргументом `add`.

Удалить в Windows: `route delete`.

Удалить в Linux: `route del`.

2.0.9.7

```
Windows: route add 192.168.11.160 mask 255.255.255.240 192.168.11.50  
Linux: route add -net 192.168.11.160 netmask 255.255.255.240 gw 192.168.11.50
```



Пример добавления статического маршрута

2.0.9.8

Постоянство вводимого статического маршрута в Windows достигают за счет аргумента -р.

2.0.9.9

Постоянство статических маршрутов в Linux обеспечивают несколькими способами с возможностью комбинирования этих способов (зависит от дистрибутива). Маршруты могут «привязываться» к конкретным сетевым интерфейсам (но необязательно использовать их).

2.0.9.10

```
/etc/sysconfig/network-scripts/route-eth0:
```

```
GATEWAY0=192.168.11.1
```

```
NETMASK0=255.240.0.0
```

```
ADDRESS0=172.16.0.0
```

```
GATEWAY1=192.168.11.1
```

```
NETMASK1=255.0.0.0
```

```
ADDRESS1=10.0.0.0
```

```
/etc/sysconfig/network-scripts/route-eth0:
```

```
172.16.0.0/12 via 192.168.11.1 dev eth0
```

```
10.0.0.0/8 via 192.168.11.1 dev eth0
```

```
/etc/sysconfig/static-routes:
```

```
any net 172.16.0.0 netmask 255.240.0.0 gw 192.168.11.1 eth0
```

```
any net 10.0.0.0 netmask 255.0.0.0 gw 192.168.11.1 eth0
```

Пример постоянных маршрутов (ветвь Red Hat)

2.0.9.11

/etc/sysconfig/network/routes:

| | | | |
|------------|--------------|-------------|------|
| 172.16.0.0 | 192.168.11.1 | 255.240.0.0 | eth0 |
| 10.0.0.0 | 192.168.11.1 | 255.0.0.0 | eth0 |

/etc/sysconfig/network/ifroute-eth0:

| | | | |
|------------|--------------|-------------|------|
| 172.16.0.0 | 192.168.11.1 | 255.240.0.0 | eth0 |
| 10.0.0.0 | 192.168.11.1 | 255.0.0.0 | eth0 |

Пример постоянных маршрутов (ветвь SUSE)

2.0.9.12

/etc/network/interfaces:

```
auto eth0
iface eth0 inet static
    address 192.168.11.2
    netmask 255.255.255.0
    gateway 192.168.11.1
    up route add -net 172.16.0.0 netmask 255.255.240.0 gw 192.168.11.1
    up route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.11.1
    down route del -net 172.16.0.0 netmask 255.255.240.0 gw 192.168.11.1
    down route del -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.11.1
```

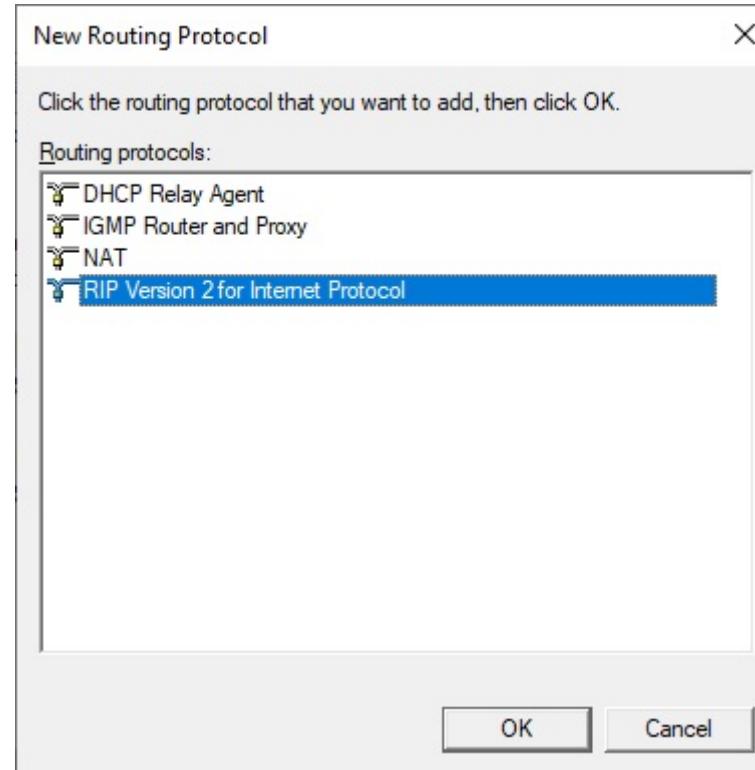
Пример постоянных маршрутов (ветвь Debian)

2.0.9.13

Поддержка динамической маршрутизации в Windows сильно ограничена (поддерживаются часть функционала RIP и часть функционала OSPF).

Вне пределов организаций Windows-шлюзы практически не **найти** (хотя ради справедливости следует отметить, что и Linux там не доминирует).

2.0.9.14



Пример поддержки динамической маршрутизации в Windows

2.0.9.15

При необходимости введения сравнительно большого количества статических маршрутов или при переходе к простейшей динамической маршрутизации (RIP) в Linux можно задействовать демон `routed`. При этом статические маршруты помещают в стандартный конфигурационный файл `/etc/gateways`.

После настройки и запуска сервиса в течение некоторого времени сформируется таблица маршрутизации, которая затем может изменяться.

2.0.9.16

```
net 0.0.0.0 gateway 10.0.0.1 metric 1 active
```

Пример маршрута из файла gateways

2.0.9.17

Для обеспечения полноценной поддержки динамической маршрутизации в последнее время вместо традиционного демона `gated` обычно применяют пакет Quagga.

Кроме «корневого» демона под названием `zebra`, Quagga включает набор демонов, каждый из которых соответствует определенному протоколу маршрутизации (`ripd`, `ospfd`, `bgpd` и другие). Quagga предоставляет интерфейс управления, подобный интерфейсу Cisco IOS.

2.0.9.18

```
traceoptions
    "/var/log/gated.log"
    size 1000K
    files 2
    general
    route
;
options
    noresolv;
;
interfaces {
    define 200.200.233.2 broadcast 200.200.233.15 netmask 255.255.255.240;
    define 200.200.233.193 broadcast 200.200.233.207 netmask 255.255.255.240;
};
rip yes {
    broadcast;
    interface lo noripin noripout;
    interface 200.200.233.2 ripin ripout version 1;
};
static{
    default gateway 200.200.233.1 interface 200.200.233.2 retain;
};
import proto rip {all; default restrict;};
export proto rip interface 200.200.233.2 {
    proto direct {all metric 1;};
};
```

Пример файла /etc/gated.conf

2.0.9.19

Quagga Routing Suite

Quagga Routing Software Suite, GPL licensed

About Quagga

Quagga is a routing software suite, providing implementations of OSPFv2, OSPFv3, RIP v1 and v2, RIPng and BGP-4 for Unix platforms, particularly FreeBSD, Linux, Solaris and NetBSD. Quagga is a fork of GNU Zebra which was developed by Kunihiro Ishiguro.

The Quagga architecture consists of a core daemon, `zebra`, which acts as an abstraction layer to the underlying Unix kernel and presents the Zserv API over a Unix or TCP stream to Quagga clients. It is these Zserv clients which typically implement a routing protocol and communicate routing updates to the `zebra` daemon. Existing Zserv implementations are:

| | IPv4 | IPv6 | Description |
|--------------------|---------------------|------|--|
| <code>zebra</code> | | | - kernel interface, static routes, zserv server |
| <code>ripd</code> | <code>ripngd</code> | | - RIPv1/RIPv2 for IPv4 and RIPng for IPv6 |
| <code>ospfd</code> | <code>ospf6d</code> | | - OSPFv2 and OSPFv3 |
| <code>bgpd</code> | | | - BGPv4+ (including address family support for multicast and IPv6) |
| <code>isisd</code> | | | - IS-IS with support for IPv4 and IPv6 |

Quagga daemons are each configurable via a network accessible CLI (called a 'vty'). The CLI follows a style similar to that of other routing software. There is an additional tool included with Quagga called 'vtysh', which acts as a single cohesive front-end to all the daemons, allowing one to administer nearly all aspects of the various Quagga daemons in one place.

Please see the [Documentation](#) for further detailed information. Community support is also available via the [mailing lists](#).

Similar projects

Other community driven, free software / open-source routing projects include:

- The [BIRD Internet Routing Daemon Project](#), supported by [cz.nic](#).
- The [OpenBGPd](#) and [OpenOSPFd](#) project, primarily developed for [OpenBSD](#).

\$Id: 674f384c825ae0fc27c39217401e054c48ad7fde \$
maintainers
"Quagga" is a trademark of Paul Jakma.

Quagga Project Site

2.0.9.20

```
hostname bgpd
password zebra
router bgp 65002
bgp router-id 10.1.2.8
network 172.16.100.0/24
network 172.16.102.0/24
network 172.16.103.0/24
neighbor 10.1.2.4 remote-as 65001
neighbor 172.16.102.2 remote-as 65003
neighbor 172.16.103.2 remote-as 65004
log file /var/log/quagga/bgpd.log
log stdout
```

Пример файла /etc/quagga/bgpd.conf

2.0.9.21

IP forwarding, и в Windows, и в Linux, по умолчанию выключен (и на серверах).

2.0.9.22

Сервис (должен быть запущен):

Routing and Remote Access

Ключ реестра (должен быть равен 1):

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\IPEnableRouter

Способы включения IP forwarding в Windows

2.0.9.23

Файловая система /proc:

```
#echo "1" > /proc/sys/net/ipv4/ip_forward
```

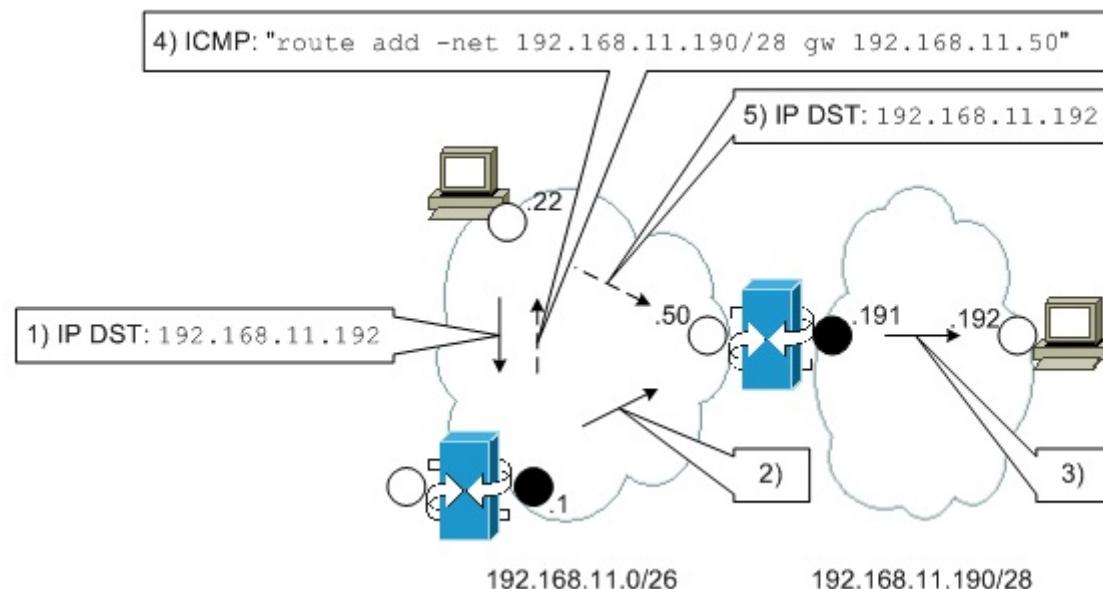
Конфигурационный файл /etc/sysctl.conf:

```
...
net.ipv4.ip_forward = 1
...
```

Способы включения IP forwarding в Linux

2.0.9.24

Даже при использовании только статической маршрутизации, практически во всех реализациях заложен механизм простейшей оптимизации -- ICMP redirects.



2.0.9.25

ICMP redirects, и в Windows, и в Linux, по умолчанию включены.

2.0.9.26

Ключ реестра (должен быть равен 0):

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\EnableICMPRedirects

Выключение ICMP redirects в Windows

2.0.9.27

```
#echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects  
#echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
```

Выключение ICMP redirects в Linux

2.0.9.28

Для работы с ARP-таблицами, и в Windows, и в Linux, используют команду arp.

Чтобы сделать вводимые соответствия IP- и MAC-адресов постоянными, в Windows нужно использовать соответствующий вариант комплексной команды netsh, а в Linux -- отредактировать стандартный конфигурационный файл /etc/ethers.

2.0.9.29

```
C:\Users\Administrator>arp -a ; Вывести на экран ARP-таблицу
```

```
Interface: 192.168.11.214 --- 0xb
  Internet Address        Physical Address          Type
  192.168.11.193          b8-38-61-81-10-ca    dynamic
  192.168.11.223          ff-ff-ff-ff-ff-ff-ff  static
  224.0.0.22               01-00-5e-00-00-16    static
  224.0.0.252              01-00-5e-00-00-fc    static
  255.255.255.255         ff-ff-ff-ff-ff-ff    static
```

```
C:\Users\Administrator>arp -s 10.0.0.10 00-AA-00-4F-2A-9C
```

```
C:\Users\Administrator>netsh -c interface ipv4 add neighbors "Local Area
Connection" "192.168.10.10" "00-1d-71-83-6c-00" store=persistent
```

2.0.9.30

```
#arp
```

| Address | HWtype | HWaddress | Flags | Mask | Iface |
|----------------|--------|-------------------|-------|------|-------|
| 192.168.11.216 | ether | 00:27:0e:1f:a0:b9 | C | | eth1 |
| 192.168.11.193 | ether | b8:38:61:81:10:ca | C | | eth1 |
| 192.168.11.214 | ether | 00:27:0e:1f:a0:e2 | C | | eth1 |

```
#arp -s 10.0.0.20 00:0c:29:c0:94:bf
```

2.0.9.31

ARP proxy при правильном конфигурировании никогда не требуется (особенно при наличии перемещаемых станций).

В Windows вообще нет способов включения ARP proxy.

2.0.9.32

```
#echo 1 > /proc/sys/net/ipv4/conf/eth0/proxy_arp
```

Принудительное включение ARP proxy в Linux применительно к eth0

2.0.9.33

Для того чтобы проследить путь к какому-либо узлу, в Windows используют команду tracert, в Linux -- traceroute.

2.0.9.34

```
C:\Users\Administrator>tracert 192.168.251.1
```

Tracing route to proxy1.bsuir.by [192.168.251.1]
over a maximum of 30 hops:

| | | | | |
|---|-----------------|-----------------|-----------------|---------------------------------|
| 1 | <u><1 ms</u> | <u><1 ms</u> | <u><1 ms</u> | 192.168.11.193 ;Нормально |
| 2 | <u><1 ms</u> | <1 ms | <1 ms | 192.168.11.29 |
| 3 | <u><1 ms</u> | <1 ms | <1 ms | st16.bsuir.by [172.16.0.1] |
| 4 | <u><1 ms</u> | <1 ms | <1 ms | proxy1.bsuir.by [192.168.251.1] |

Trace complete.

```
C:\Users\Administrator>tracert -d proxy1.bsuir.by ;Не делать DNS-запросы
```

Tracing route to proxy1.bsuir.by [172.20.0.1]
over a maximum of 30 hops:

| | | | | |
|---|-------|-------|-------|-------------------|
| 1 | <1 ms | <1 ms | <1 ms | 192.168.11.193 |
| 2 | <1 ms | <1 ms | <1 ms | 192.168.11.2 |
| 3 | <1 ms | <1 ms | <1 ms | <u>172.16.0.1</u> |
| 4 | 1 ms | <1 ms | <1 ms | 172.20.0.1 |

Trace complete.

2.0.9.35

```
#traceroute 192.168.251.1

traceroute to 192.168.251.1 (192.168.251.1), 30 hops max, 60 byte packets
 1  192.168.11.193 (192.168.11.193)  0.585 ms  1.290 ms  1.580 ms
 2  192.168.11.29 (192.168.11.29)  1.558 ms  1.688 ms  1.800 ms
 3  172.16.0.1 (172.16.0.1)  0.586 ms  0.651 ms  0.748 ms
 4  192.168.251.1 (192.168.251.1)  0.814 ms  0.908 ms  0.971 ms
```

2.0.10.1

Таблица маршрутизации IOS имеет достаточно сложную иерархическую структуру.

В первую очередь маршруты делят на:

1. Directly connected равно Connected (код C) -- маршруты к своим подсетям (а не «к подключенными интерфейсам») (особо выделяемые статические маршруты, которые вносятся автоматически при административном включении соответствующих интерфейсов).

2. Static (код S) -- статические (собственно статические маршруты, которые вносят «вручную»).

3. Dynamic (коды R, B, O и другие) -- динамические (автоматически вносятся процессами динамической маршрутизации).

Новшество таблиц маршрутизации IOS 15.X (а также таблиц маршрутизации IPv6):

+4. Local (код L) -- локальные или, в данном контексте, маршруты к своим сетевым интерфейсам (еще одни особо выделяемые статические маршруты, которые также вносятся автоматически при административном включении соответствующих интерфейсов, и которые в таблицах маршрутизации 12.X явно не присутствуют).

2.0.10.2

В иерархии маршрутов выделяют два уровня:

1. L1 -- маршруты к стандартным подсетям и подсетям, больших чем стандартные.
2. L2 -- маршруты к подсетям, меньшим чем стандартные, и к сетевым интерфейсам.

С другой стороны, маршруты в иерархии можно рассматривать как:

1. Parent -- родительские.
2. Child -- дочерние.

Иерархия необходима для ускорения обработки таблицы маршрутизации. Сначала просматриваются маршруты первого уровня. В случае **попадания** происходит переход к просмотру соответствующих маршрутов второго уровня.

Маршруты первого уровня:

1. Default route -- маршрут (**маршруты**) по умолчанию.
2. Supernet routes -- маршруты к подсетям, больших чем стандартные.
3. Network routes -- маршруты к стандартным подсетям.

2.0.10.3

При выборе маршрута для передачи пакета из имеющихся в таблице маршрутизации (равно как и выборе маршрута для внесения в таблицу маршрутизации при динамической маршрутизации) оцениваются:

1. Prefix **length** -- длина префикса -- чем больше, тем маршрут приоритетнее.
2. Administrative **distance** (по-другому, **external administrative distance**) -- административная дистанция -- «старшая» часть (указывается в квадратных скобках до слеша) -- чем меньше, тем маршрут приоритетнее.
3. Metric (по-другому, **internal administrative distance**) -- **метрика** -- «младшая» часть (указывается в квадратных скобках после слеша) -- в разных протоколах высчитывается по-разному -- **так же чем меньше, тем маршрут приоритетнее.**

2.0.10.4а

| Административная дистанция | Маршрут |
|-------------------------------|---|
| 0 | Connected |
| 1 | Static |
| 3 | DMNR (Dynamic Mobile Network Routing) |
| 5 | EIGRP summary |
| 20 | BGP external |
| 90 | EIGRP internal |
| 100 | IGRP |
| 110 | OSPF |
| 115 | IS-IS |
| 120 | RIP |
| 140 | EGP |
| 160 | ODR (On-Demand Routing) |
| 170 | EIGRP external |
| 200 | BGP internal |
| 250 | NHRP (Next Hop Resolution Protocol) |
| 254 | Маршрут по умолчанию, полученный посредством DHCP |
| 255 | Недоверительный маршрут |

Значения предустановленных административных дистанций Cisco по умолчанию

2.0.10.4b

Административную дистанцию статических и динамических маршрутов можно корректировать в диапазоне от 0 до 255.

Статические маршруты с заведомо большей административной дистанцией чем у динамических (например, 254), создаваемые как резервные, называют плавающими (*floating*).

Маршруты с административной дистанцией 255 (специально созданные или полученные от других маршрутизаторов) в таблицу маршрутизации IOS не вносятся.

2.0.10.5

Любой маршрут зависит от состояния выходного интерфейса (в нем). Скажем, временное административное выключение интерфейса приводит к временному изъятию (из таблицы маршрутизации) маршрутов, действующих данный интерфейс.

2.0.10.6

Касательно Cisco IOS, фундаментальное различие между полноклассостью и бесклассостью проявляется и в процессе выбора маршрутов.

Следует различать два алгоритма поведения, причем **любое** поведение не влияет на передачу маршрутов (не нужно путать с разницей между полноклассовыми и бесклассовыми протоколами динамической маршрутизации).

При полноклассовом выборе маршрутов (classful routing behavior), в отличие от бесклассового (classless routing behavior), после «захода» к L2-маршрутам возврата к L1-маршрутам уже не происходит. В современных версиях IOS по умолчанию применяется бесклассовый подход.

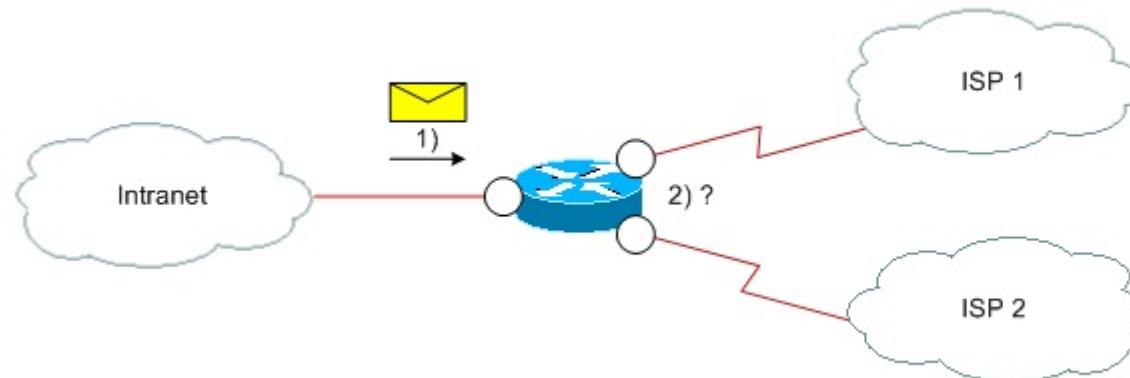
2.0.10.7

IOS поддерживает эквивалентную и неэквивалентную балансировку нагрузки при маршрутизации (и в отношении маршрутов по умолчанию).

Маршруты, которые могут быть применены для фактической передачи пакетов называют актуальными (ultimate). Их признаком является наличие шлюза или выходного сетевого интерфейса.

2.0.10.8

Расширение PBR (Policy-Based Routing) позволяет осуществлять маршрутизацию не только исходя из адреса назначения, а и с учетом множества других критериев (например, назначать шлюз исходя из адреса источника).



2.0.10.9

Расширение VRF (Virtual Routing and Forwarding) позволяет в дополнение к собственно маршрутизатору на базе IOS создать виртуальные маршрутизаторы с изолированными таблицами маршрутизации. При этом, к отдельно взятой сущности VRF необходимо привязать соответствующие интерфейсы или подинтерфейсы.



2.0.10.10

Для просмотра текущей таблицы маршрутизации IOS используют команду `show ip route`.

При этом видна иерархия маршрутов. Наряду с уровнями, если маски дочерних подсетей разные, то родительская подсеть показывается как `variably subnetted`, иначе, просто как `subnetted`.

2.0.10.11а

```
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      + - replicated route, % - next hop override
```

Gateway of last resort is 0.0.0.0 to network 0.0.0.0 !Все нули так как через se

```
    172.16.0.0/28 is subnetted, 1 subnets !Относится к 172.16.41.0
O      172.16.41.0 [110/11] via 204.120.160.1, 1d20h, GigabitEthernet0/0
    204.120.160.0/24 is variably subnetted, 4 subnets, 2 masks
C        204.120.160.0/29 is directly connected, GigabitEthernet0/0
L        204.120.160.2/32 is directly connected, GigabitEthernet0/0
C        204.120.160.168/30 is directly connected, Serial0/0/0
L        204.120.160.169/32 is directly connected, Serial0/0/0
    192.168.255.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.255.0/29 is directly connected, GigabitEthernet0/0
L        192.168.255.1/32 is directly connected, GigabitEthernet0/0
    63.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        63.166.59.188/30 is directly connected, Vlan10
L        63.166.59.190/32 is directly connected, Vlan10
    208.35.255.0/24 is variably subnetted, 3 subnets, 2 masks
O        208.35.255.0/28 [110/11] via 204.120.160.1, 1d20, GigabitEthernet0/0
O        208.35.255.48/28 [110/11] via 204.120.160.1, 1d20h, GigabitEthernet0/0
O        208.35.255.64/26 [110/11] via 204.120.160.1, 1d20h, GigabitEthernet0/0
    75.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        75.160.162.64/26 is directly connected, GigabitEthernet0/1
L        75.160.162.65/32 is directly connected, GigabitEthernet0/1
S*       0.0.0.0/0 is directly connected, Serial0/0/0
```

2.0.10.11b

```
!  
!21 маршрут всего  
!7 маршрутов первого уровня  
!14 маршрутов второго уровня  
!6 родительских маршрутов  
!14 дочерних маршрутов  
!15 актуальных маршрутов  
!Разница между полноклассовостью и безклассовостью проявится при маршрутизации  
!пакета с IP-адресом назначения например 208.35.255.33  
!
```

2.0.10.12

Gateway of last resort -- так часто называют шлюз по умолчанию.

Кроме упомянутых выше буквенных кодов маршрутов, в таблице маршрутизации можно увидеть другие.

2.0.10.13

```
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override, p - overrides from PfR
```

Gateway of last resort is not set

Пример другой, наиболее полной, «шапки» таблицы

2.0.10.14а

Буквы, не относящиеся к протоколам динамической маршрутизации:

М -- mobile -- маршрут внесен подсистемой LAM (Local Area Mobility) (расширение ARP для поддержки перемещаемых устройств).

У -- per-user static route -- статический маршрут получен по протоколу RADIUS либо TACACS с AAA-сервера (клиент подключен посредством топологии «точка-к-точке»).

о -- ODR -- маршрут получен по протоколу ODR (расширение CDP для топологии «ступица со спицами»).

Р -- periodic downloaded static route -- статический маршрут получен по протоколу RADIUS либо TACACS с AAA-сервера.

н -- NHRP -- маршрут получен по протоколу NHRP (расширение ARP для иерархических NBMA-топологий).

л -- LISP -- маршрут получен по протоколу LISP (Locator/ID Separation Protocol) (альтернатива и в то же время надстройка над IP).

а -- application route -- маршрут создан приложением onePK (доступно на некоторых платформах).

2.0.10.14b

Символы, дополняющие буквы:

- * -- candidate default -- при балансировке нагрузки маршрут будет выбран для передачи очередного пакета.
- + -- replicated route -- маршрут импортирован из другой сущности VRF.
- % -- next hop override -- шлюз в маршруте заменен протоколом NHRP.
- p -- overrides from PfR -- маршрут изменен фреймворком PfR (Performance Routing) (оптимизация).

2.0.10.15

Для внесения статического маршрута в таблицу маршрутизации используют команду `ip route`.

Отключение бесклассового выбора маршрутов, то есть включение полноклассового, осуществляют командой по `ip classless`.

Функционал IP forwarding по умолчанию включен и может быть выключен командой по `ip routing`.

Функционал ICMP redirects по умолчанию включен и может быть выключен командой по `ip redirects`.

2.0.10.16

```
Router(config)#ip route 192.168.11.160 255.255.255.240 192.168.11.50
```

```
Router(config)#ip route 0.0.0.0 0.0.0.0 192.168.11.1 !Маршрут по умолчанию  
!(команда ip default-gateway предназначена для управляемых коммутаторов)
```

```
Router(config)#no ip classless
```

```
Router(config)#no ip routing
```

```
Router(config-if)#no ip redirects
```

Команды IOS

2.0.10.17а

Для просмотра ARP-таблицы **используют** команду `show ip arp`.

Кроме просто ARP proxy, Cisco различает local ARP proxy.

При ARP proxy задействуются сетевые интерфейсы из разных подсетей, как и было описано ранее.

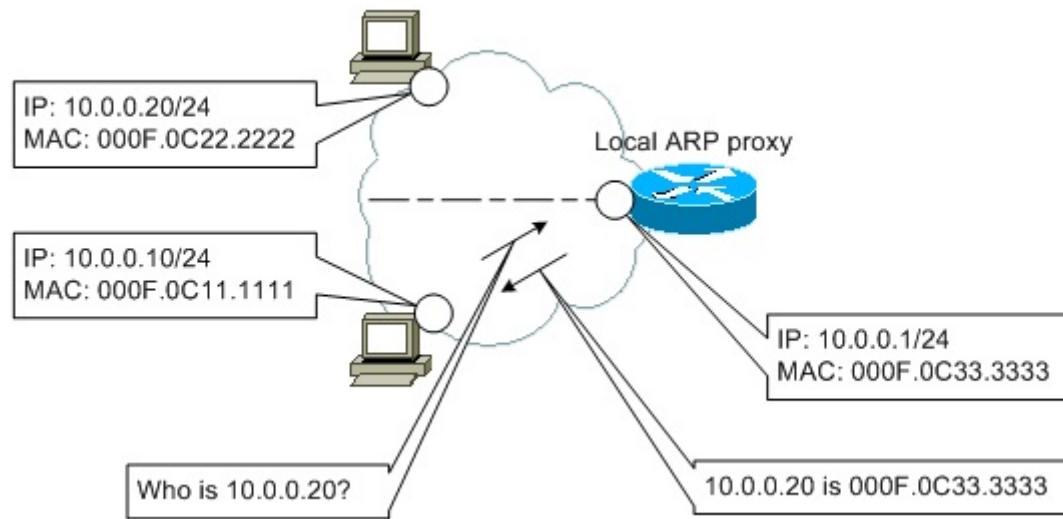
При local ARP proxy задействуется только один сетевой интерфейс маршрутизатора (прозрачного шлюза). Такое востребовано когда этот сетевой интерфейс находится в одной подсети с запрашивающей и запрашиваемой станциями, но запрашиваемая станция по каким-либо административным или технологическим причинам сама ответить не может (например, чтобы весь трафик между станциями прозрачно пропускать через маршрутизатор).

Первый функционал по умолчанию включен, а второй -- выключен. Конфигурирование осуществляют командами `ip proxy-arp` и `ip local-proxy-arp` соответственно.

ARP proxy (не local ARP proxy) можно выключить и глобально: `ip arp proxy disable`.

(Командой `ip directed-broadcast` можно включить по умолчанию выключенный функционал `directed broadcast forwarding`.)

2.0.10.17b



Cisco local ARP proxy

2.0.10.18

```
Router#show ip arp
```

| Protocol | Address | Age (min) | Hardware Addr | Type | Interface |
|----------|---------------|-----------|----------------|------|--------------------|
| Internet | 80.94.160.193 | 0 | 0022.5517.e002 | ARPA | GigabitEthernet0/0 |
| Internet | 80.94.160.200 | - | a80c.0d99.757b | ARPA | GigabitEthernet0/0 |
| Internet | 80.94.160.206 | 59 | 0017.9a3a.cd1f | ARPA | GigabitEthernet0/0 |
| Internet | 172.16.0.1 | 0 | 0050.569f.9ebb | ARPA | GigabitEthernet0/1 |
| Internet | 172.16.0.8 | 0 | 0050.569f.984e | ARPA | GigabitEthernet0/1 |
| Internet | 172.16.0.12 | 148 | 94de.8070.463f | ARPA | GigabitEthernet0/1 |
| ... | | | | | |

```
Router(config-if)#no ip proxy-arp
```

