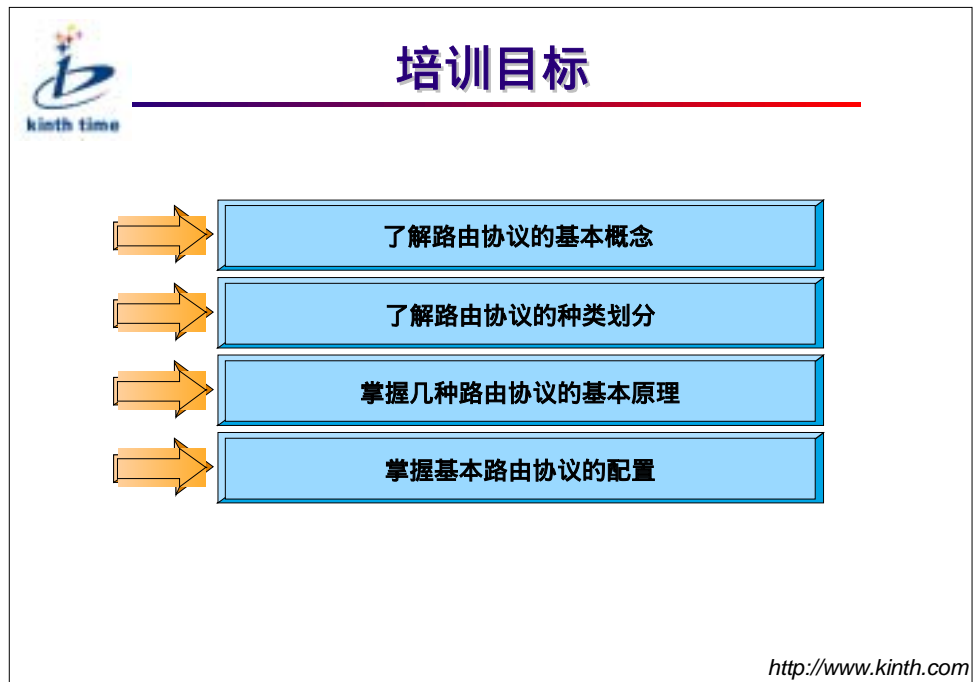


第九章 路由协议

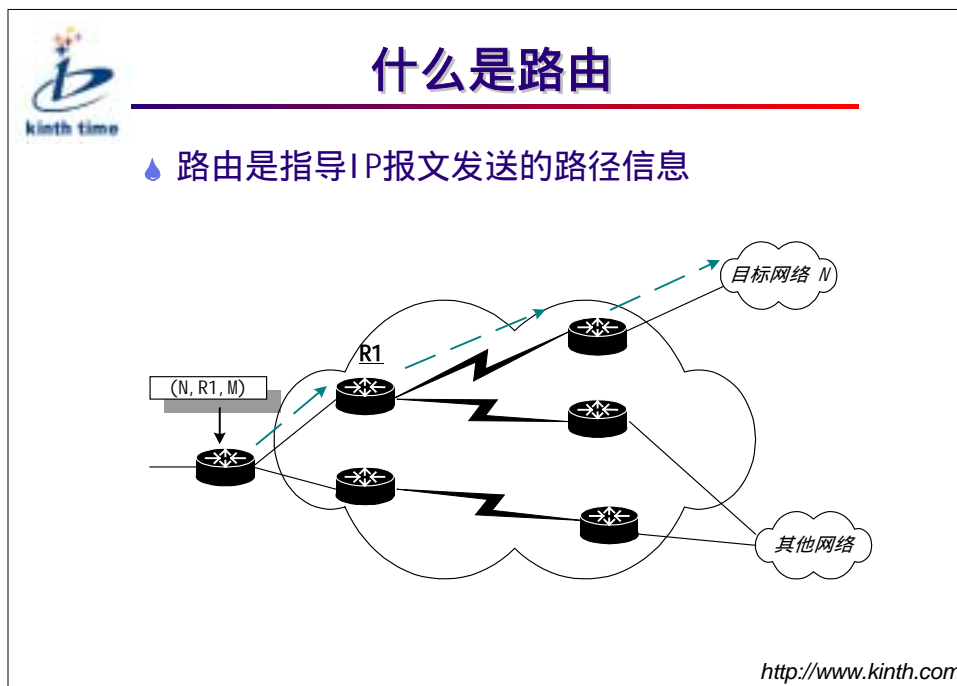
.1 培训目标



本章讲述路由协议及其配置，内容主要包括路由协议共通的基本概念和原理，并依据不同的原则对现有路由协议进行种类划分，以使读者对路由协议具有整体概念和基本了解。同时介绍了静态路由和几种基本的动态路由协议的原理和配置方法，以及一些典型配置的实例说明。

.2 路由的基本概念及算法

.2.1 什么是路由



在基于 TCP/IP 的网络中，所有数据的流向都是由 IP 地址来指定的，网络协议根据报文的目地地址将报文从适当的接口发送出去。而路由就是指导报文发送的路径信息。

就像实际上生活中交叉路口的路标一样，路由信息在网络路径的交叉点（路由器）上标明去往目标网络的正确途径，网络层协议可以根据报文的目的地查找到对应的路由信息，把报文按正确的途径发送出去。一般一条路由信息至少包含以下几方面内容：目标网络，用以配置报文的目地地址，进行路由选择；下一跳，指明路由的发送路径；Metric、路由权，标示路径的好坏，是进行路由选择的标准。

例如，在上图中路由器上有一条去往目标网络 N 的路由，下一跳是 R1。所有经过此路由器的去往目标网络 N 的报文都被转发到路由器 R1 上去，再重复这种路由过程，直到到达正确的目的地。

.2.2 路由的分类



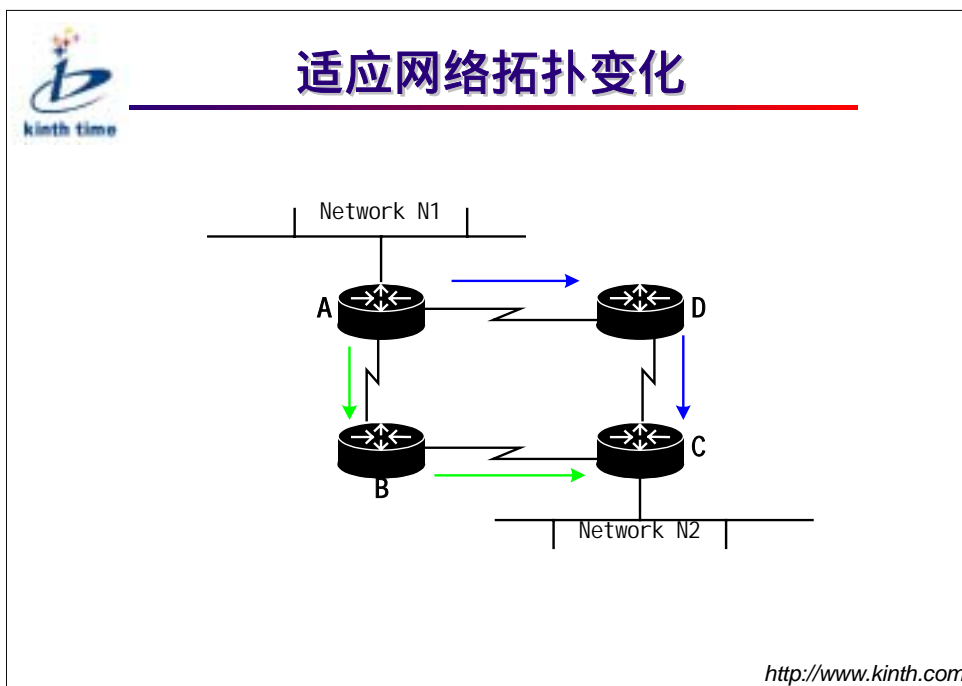
通常情况下,指导 IP 转发的路由信息可以通过如下三种不同的途径来获得:

静态路由 —— 由系统管理员手动配置的到目标网络的唯一路径,当网络结构发生变化时也必须由系统管理员手动的修改配置。但合理的使用静态路由可以改进网络的性能,为重要的应用保存带宽。

缺省路由 —— 由系统管理员手动配置的一种特殊路由,可以将所有找不到匹配路由的报文转发到指定的缺省网关。

动态路由 —— 由动态路由协议从其他路由器学到的到达目标网络的发送路径,可以根据网络结构的变化动态地更新路由信息。

.2.3 对网络拓扑变化的适应性



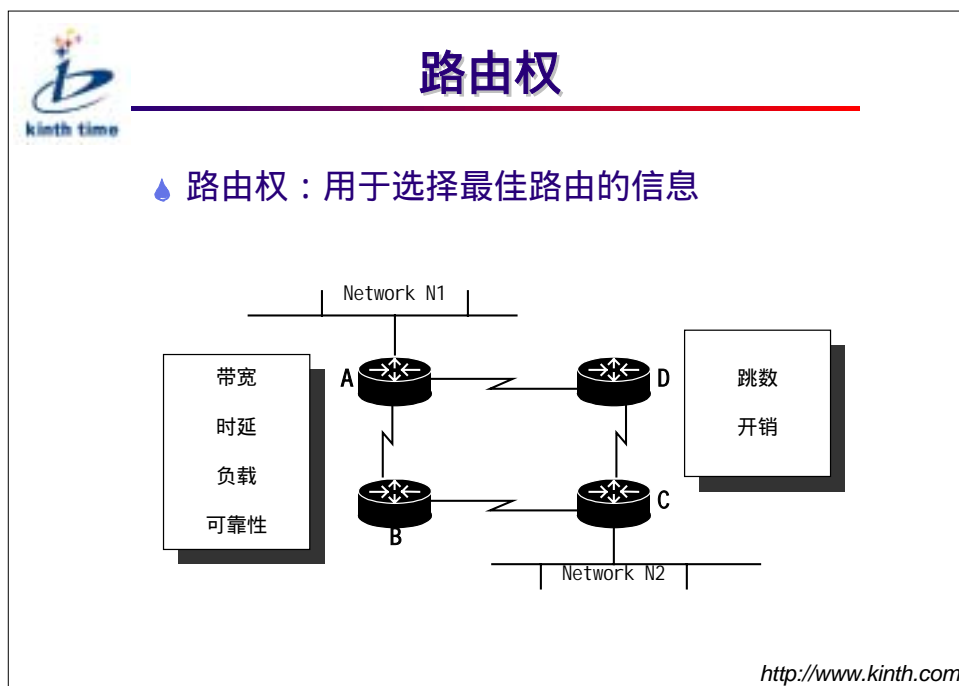
网络的配置不同决定了其对网络拓扑结构变化的适应能力，这取决于网络中是否使用动态路由协议。

静态路由信息可以指导报文的正常转发。再如图所示的网络中，为到达目标网络 N2，在路由器 A 上配置静态路由指向路由器 D，在路由器 D 上配置静态路由指向路由器 C，这样，从网络 N1 发往 N2 的报文就可以经过路由器 A、D、C 最后到达目标网络 N2。

但假如从路由器 A 到 D 的通路出现了问题，那么路由器 A 就不能根据静态路由的指示将报文发送到路由器 D 上去。如果想要保证网络的畅通，就必须由网管员手动配置一条经由路由器 B 的静态路由，这样，报文就可以经由路由器 A、B、C，最后到达目标网络。

如果网络中运行了某种动态路由协议（如 RIP 协议）情况就会有所不同。当经由路由器 D 的路由失效之后，路由器之间会通过动态路由协议的路由信息传递，自动的发现另外一条到达目标网络经由路由器 B 的路由，并修改路由表，指导报文进行正确的转发。

.2.4 路由权

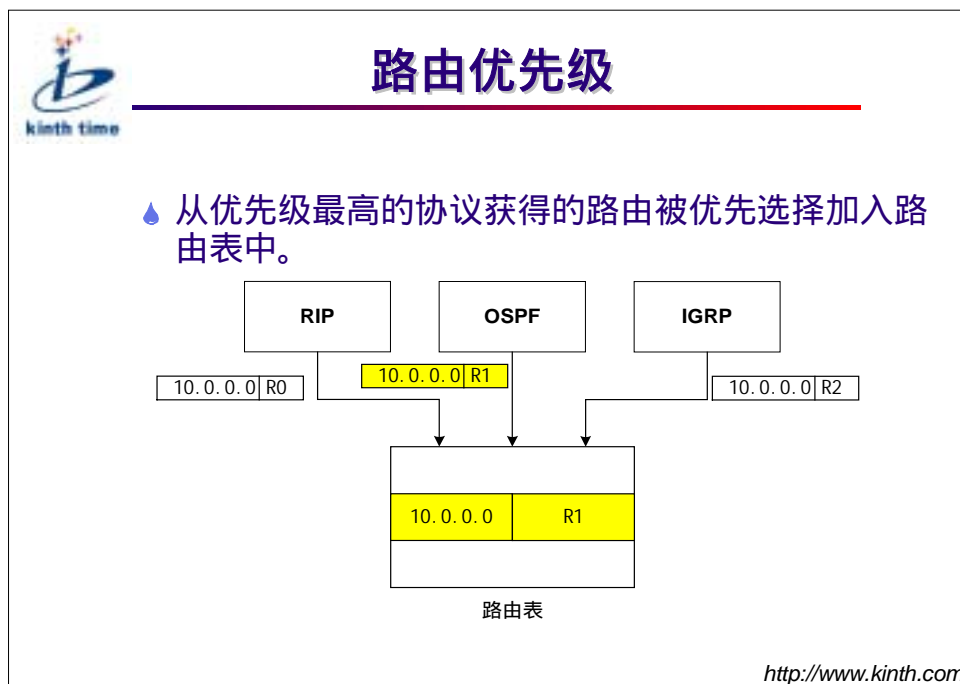


路由算法修改路由表的基本目的是将最好路由信息添加到路由表中，路由的好坏是由路由算法根据自己获得的路由信息计算出来的。对于每一条路由，路由算法产生一种权值来表示路由的好坏。通常情况下，这种权值越小，该路径越好。

路由权的计算可能基于路径某单一特性计算，也可能基于路径多种属性进行计算。有几种路径特性经常被用于权值计算，如下：

- ☞ 带宽 —— 链路的数据容量。例如，通常情况下 10M 以太网链路比 64K 出租线路要更好。
- ☞ 时延 —— 报文从到达目标网络所需要的时间。
- ☞ 负载 —— 处于活跃状态的网络资源数量。
- ☞ 可靠性 —— 每条数据链路的出错率。
- ☞ 跳数 —— 报文到目的地需要经过的网络数。
- ☞ 开销 —— 一种人为设定的值，通常由网络管理员根据带宽、线路价格或其他一些因素综合得出。

2.5 路由优先级

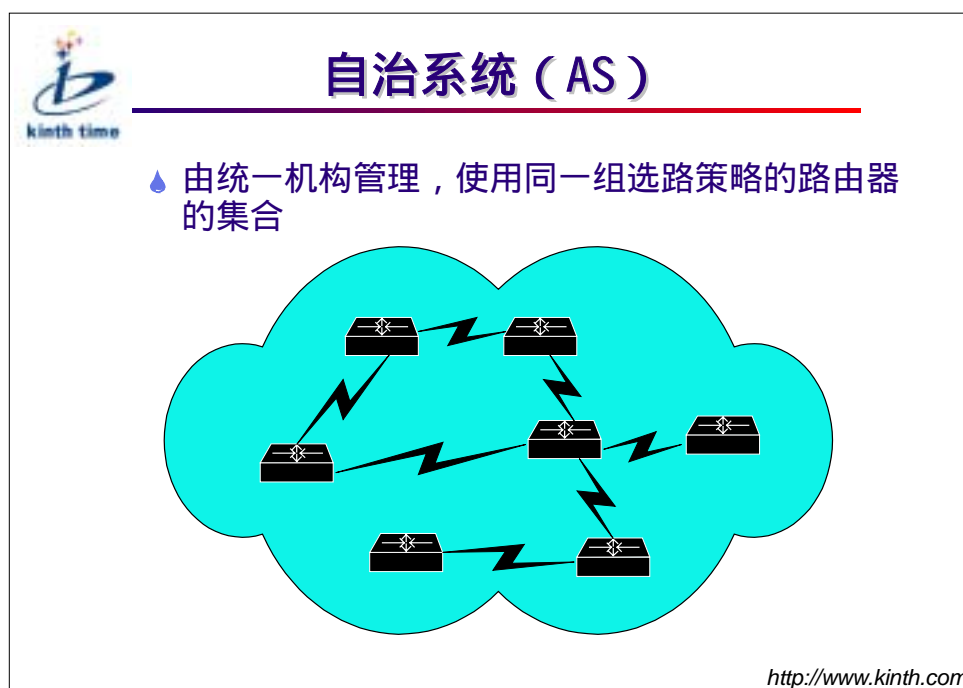


前面讲过，各个路由协议都有自己的标准来衡量路由的好坏（有的采用下一跳次数、有的采用带宽、有的采用时延，一般在路由数据中用度量 Metric 来量化），并且每个路由协议都试图将自己认为是最好的路由送到路由表中，这样我们就有可能从不同的协议得到到达同一目标网络的不同路由。尽管每个路由协议都给出了度量值，但是由于各个协议所采用度量值的含意不同，它们之间没有可比性。这就需要有种策略来决定使用哪一条路由。按照策略，判断最优的路由，我们才将它加入路由表，利用它来进行包的转发。

通常，我们使用路由优先级来判断不同路由协议所获得路由的好坏。每一种路由协议都由自己的优先级，当不同路由协议之间的路由发生冲突时，选择其中优先级最高的路由协议获得的路由。路由优先级是根据路由算法的优劣等因素得出的经验数值，也可以由网管员手动修改。

在上图中，三种路由协议 RIP、OSPF、IGRP 各自得到了一条到达目标网络 10.0.0.0 的路由。我们假定三种协议之间的路由优先级的次序是 OSPF > IGRP > RIP，则最终选定 OSPF 路由作为最优路由。

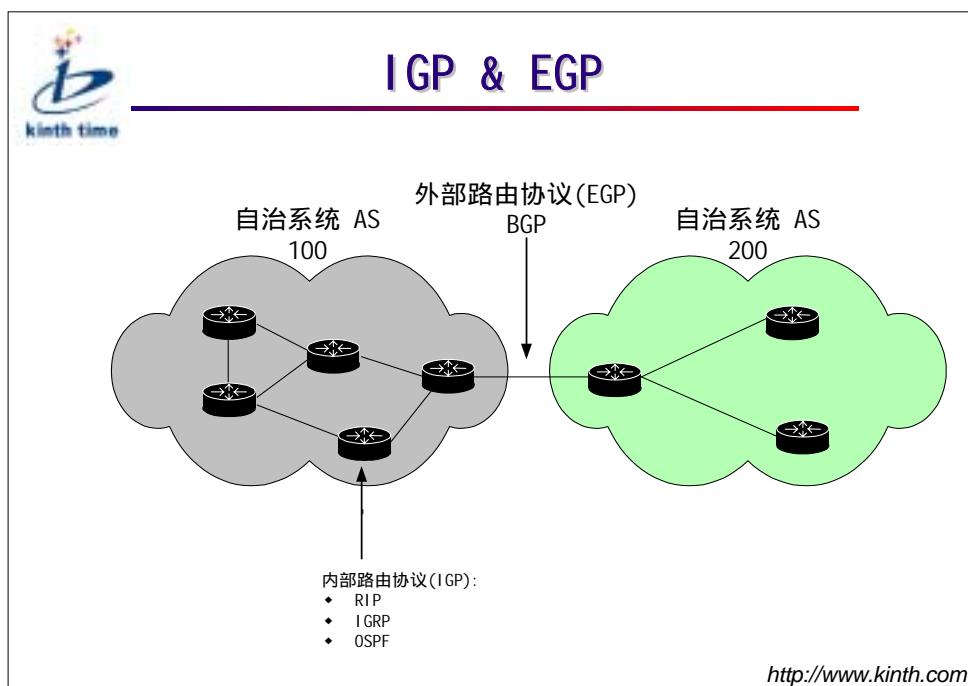
.2.6 自治系统



为了便于网络的管理，人为地将互联网划分成若干自治系统。每一个自治系统由一组在统一的机构管理下的路由器组成，整个系统对外呈现统一的路由机制，并被看成独立的网络组成单元。

自治系统由一个 16bit 的整数标示，这个整数被称作自治系统号。自治系统号是由 NIC (Network Information Center) 统一分配和管理的。

.2.7 IGP 和 EGP



从上一节所定义的自治系统我们可以把网络化分成若干区域，从而可以标示路由协议的作用范围。根据路由协议的不同作用范围，我们可以将路由协议划分成域内路由协议（IGP）和域间路由协议（EGP）。顾名思义，域内路由协议的作用范围被限制在自治系统内部，而域间路由协议适用于不同自治系统间的路由交换。

IGP 包括 RIP、IGRP、OSPF、IS-IS 等，而 EGP 目前只有 BGP 协议。

.2.8 按寻径算法划分路由协议



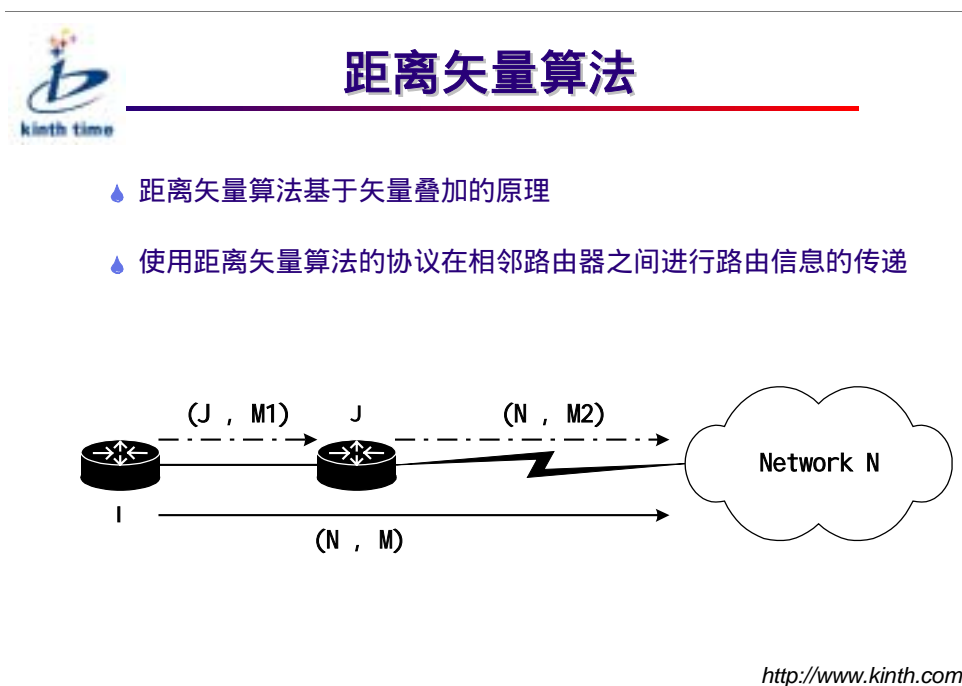
按寻径算法划分

- 距离矢量算法
 - RIP
 - IGRP
 - BGP
- 链路状态算法
 - OSPF
 - IS-IS

<http://www.kinth.com>

根据寻径算法，单播路由协议可分成距离矢量协议（Distance-Vector）和链接状态协议（Link-State）。距离矢量协议包括 RIP、IGRP、EIGRP、BGP，链接状态协议包括 OSPF、IS-IS。

.2.9 距离矢量算法



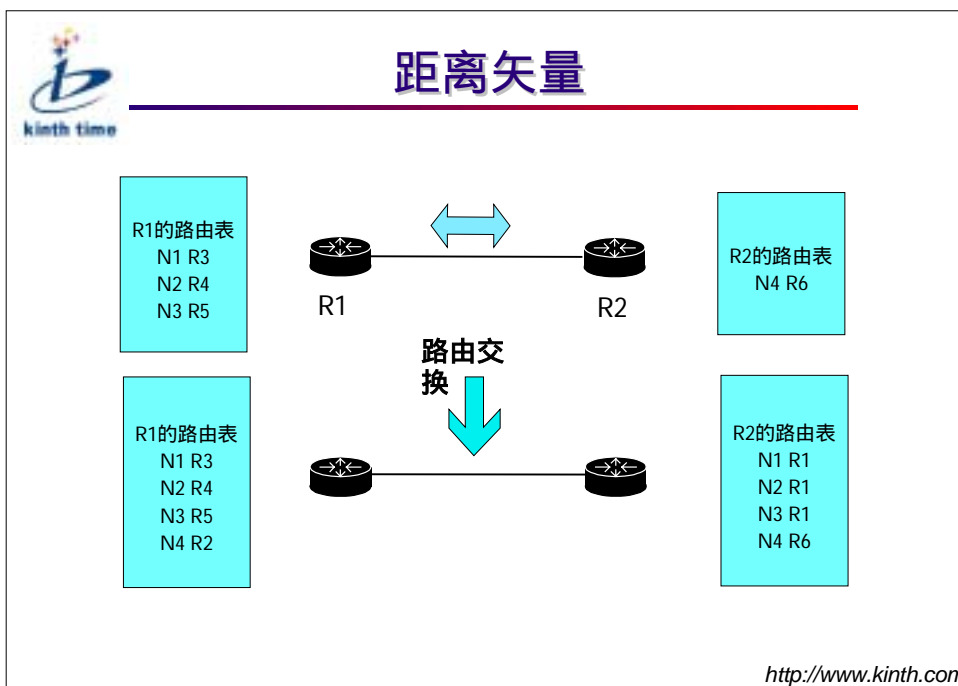
距离矢量算法是动态路由协议常用的一种路由算法，其基本原理就是运用矢量叠加的方式获取和计算路由信息。

所谓距离矢量即是将一条路由信息考虑成一个由目标和距离(用 Metric 来度量)组成的矢量，每一台路由器从其邻居处获得路由信息，并在每一条路由信息上叠加从自己到这个邻居的距离矢量，从而形成自己的路由信息。

在上图所示的例子中，路由器 I 从路由器 J 获得到达目标网络 N 的路由信息(N, M2)，其中 N 标示目标网络，M2 标示距离长短的 Metric 值。并且在这条矢量数据上叠加从 I 到 J 的距离矢量(J, M1)，形成从 I 到目标网络 N 的路由信息(N, M)，其中 $M = M1 + M2$ 。

这种过程发生在路由器的各个邻接方向上，通过这种方法路由器可以获得到达网络中目标网络的途径和距离，并从中选择最佳路径形成和维护自己的路由表。

2.10 距离矢量协议

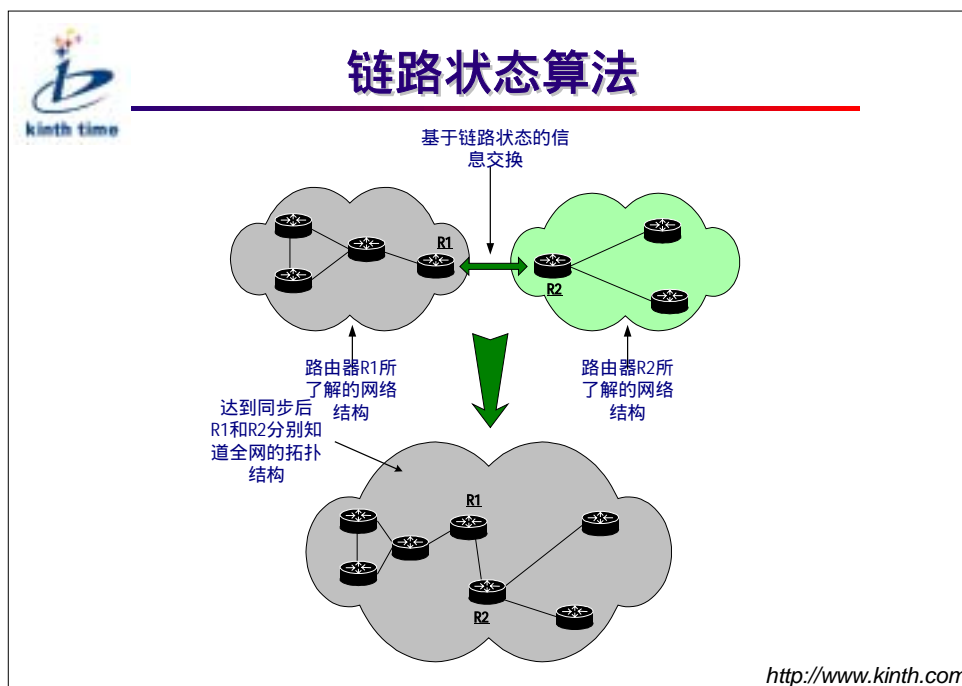


距离矢量协议直接传送各自的路由表信息。网络中的路由器从自己的邻居路由器得到路由信息，并将这些路由信息连同自己的本地路由信息发送给其他邻居，这样一级级的传递下去以达到全网同步。每个路由器都不了解整个网络拓扑，它们只知道与自己直接相连的网络情况，并根据从邻居得到的路由信息更新自己的路由表。

距离矢量协议无论是实现还是管理都比较简单，但是它的收敛速度慢，报文量大，占用较多网络开销，并且为避免路由环路得做各种特殊处理。

目前基于距离矢量算法的协议包括 RIP、IGRP、EIGRP、BGP。其中 BGP 是距离矢量协议变种，它是一种路径矢量协议。

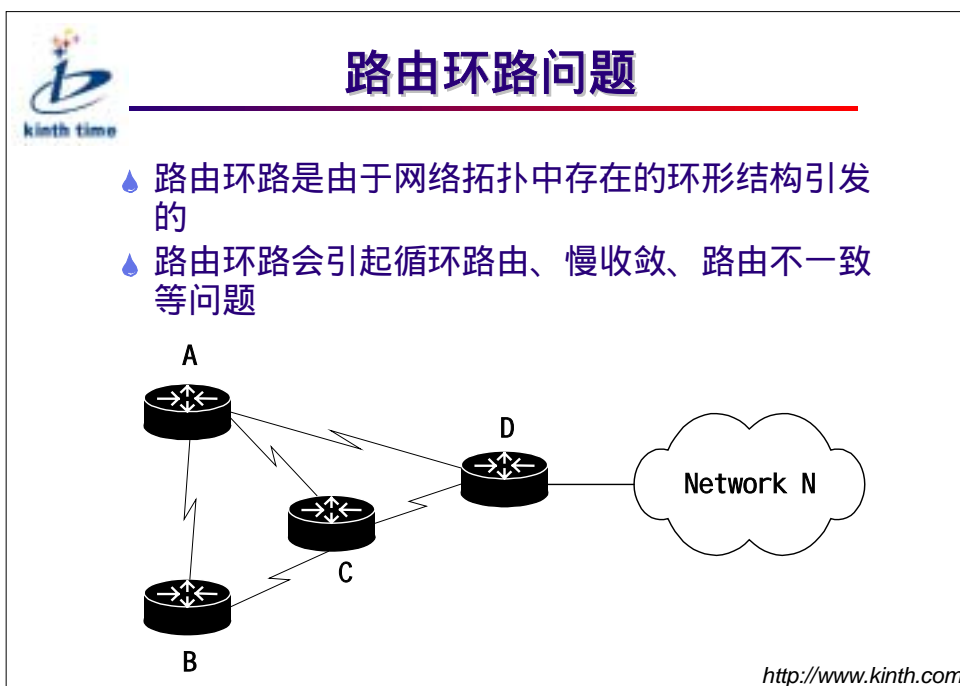
.2.11 链路状态协议



链路状态协议传送路由器之间的连接状态，每个路由器将自己所知道的链路状态通知其他路由器。这样网络中的路由器都知道整个网络拓扑结构，路由根据 SPF（Shortest Path First）算法得出。基于链路状态算法的协议结构复杂，难于管理。但由于每一台路由器都了解全网的拓扑结构，所以不用担心路由环路的影响，同时它的收敛速度快，需要传递的信息量少，可以节省网络带宽。

典型的基于链路状态算法的协议有 OSPF 和 IS-IS。

2.12 路由环路问题



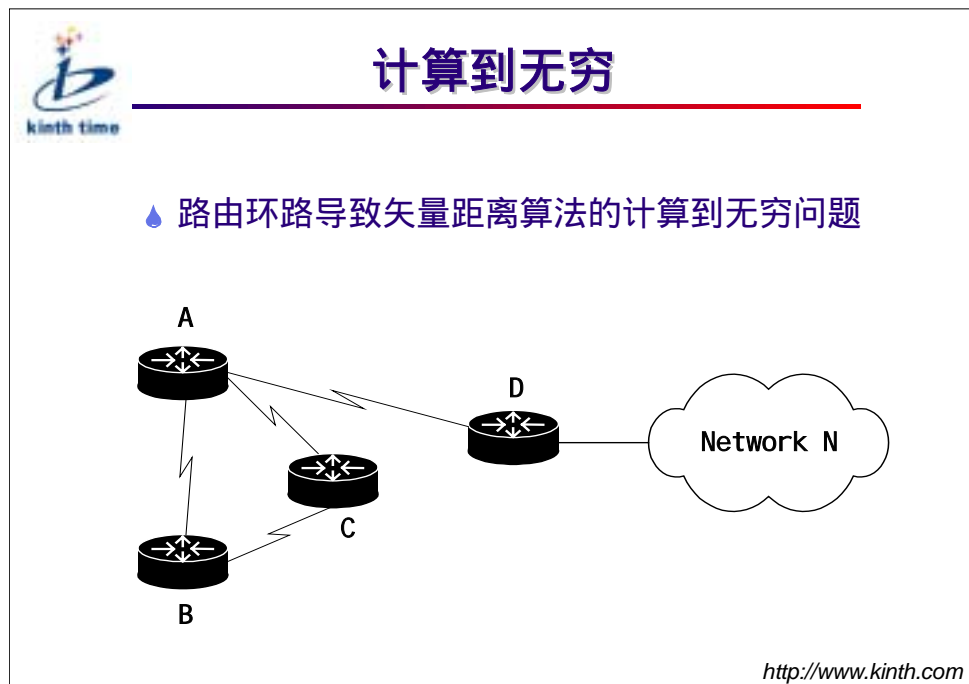
路由环路问题是当网络拓扑结构发生变化，由于网络中存在的环状结构所引发的。上图是一个简单的例子。

如图所示，路由器 D 与目标网络 N 直接相连，而路由器 A、B、C 之间组成了一个环形网络，连接路由器的路径旁边的数字显示了路径对应的权值。这样在路由器 A 上就有一条经过路由器 D 的去往目标网络的路由，权值为 1，而路由器 B、C 上分别有一条经过路由器 A 的去往目标网络的路由，权值为 2。假设，路由器 A 与 D 之间的通信出现了问题，这时候就有可能产生路由环路问题，并导致慢收敛。为了简单起见，我们假设所有的路由器都同时发送路由更新报文，下表显示了路由器 A、B、C 上到达目标网络的路由随时间的变化。

时间 T	——					
路由器A	不可达	(C, 3)	(C, 4)	...	(C, 10)	(C, 11)
路由器B	(A, 2)	(C, 3)	(C, 4)	...	(C, 10)	(C, 11)
路由器C	(A, 2)	(B, 3)	(B, 4)	...	(A, 10)	(D, 10)

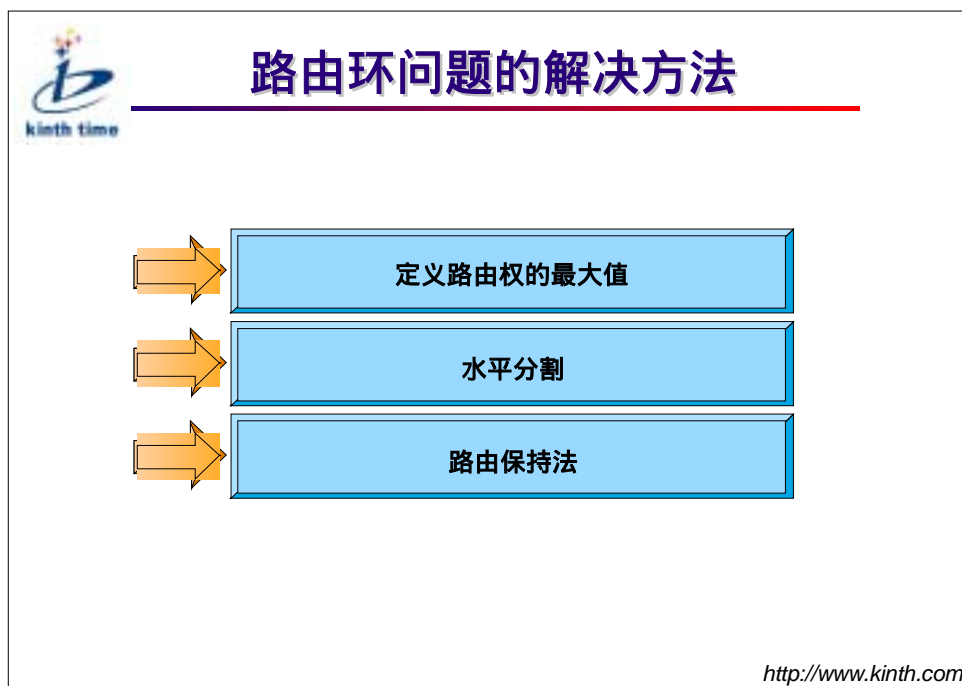
虽然算法最终收敛，但在漫长的收敛过程中在各个路由器上的路由并不能反映网络的真实结构，A、B、C 三台路由器之间相互欺骗，把错误的路由信息在网络中相互传递。

.1.1 计算到无穷



仍然考虑如前面的网络结构，这里我们假设路由器 C 与 D 之间没有直接的通路相连，这时再断掉路由器 A 与 D 之间的连接，循环路由就会持续进行下去，错误的路由在路由环中一直传播下去。在前面所述的例子中，这种循环最终会停下来是因为路由器 C 得到了一条更好的路由 (D, 10)，从而中止了路由循环的继续传播。现在这个中止条件没有了，路由信息的循环累加就会不停的继续下去，直到路由权值累加到无穷大。这就被称为计算到无穷问题。

1.2 解决路由环问题的几种方法




为解决路由环问题首先要设定一个最大值作为路由权的无穷大值，这个数值通常要根据协议的路由权值的计算方法而定。比如在 RIP 中以跳数来作为路由权的度量，它的最大值就是 16，也就是说如果某条路由的 Metric 值为 16 就表示这条路由不可达。

最大值的设定只能解决无限循环的问题，而并不能解决慢收敛问题。路由环路产生的一个重要原因就是不正确的路由信息通过获得这条信息的接口再发送回去，替代了新的正确的路由，这也就导致了错误路由信息的循环往复。如在我们前面提到的例子中，正是因为路由器 C 将从路由器 A 上得到的路由信息有发送回路由器 A，才会导致路由的循环依赖。由此我们得到一个解决路由环路的方法：水平分割。水平分割就是从某个接口接收到的路由信息不再从这个接口发送回去，从而避免错误的路由信息被使用。

另一个方法就是路由保持法，也就是将路由的不可达状态保持一段时间，在这段时间内不对这条路由作任何修改，直到这条路由的不可达状态被尽可能的扩散出去。这样也可以防止错误路由的传播。

.1.3 小结



小结


以上部分讲述了如下问题

- ◆ 路由及路由协议的基本概念
- ◆ 路由协议的不同分类
- ◆ 路由协议算法中存在的缺陷以及相应的解决方案

<http://www.kinth.com>

.2 路由的基本配置方法

.2.1 配置静态路由



静态路由配置

💧 静态路由的配置命令和命令模式

```
Quidway (config) #  
ip route <ip_address> [ <mask> | <masklen> ]  
<interface_name> | <gateway_address>  
[ preference <preference_value> | reject | blackhole ]
```

<http://www.kinth.com>

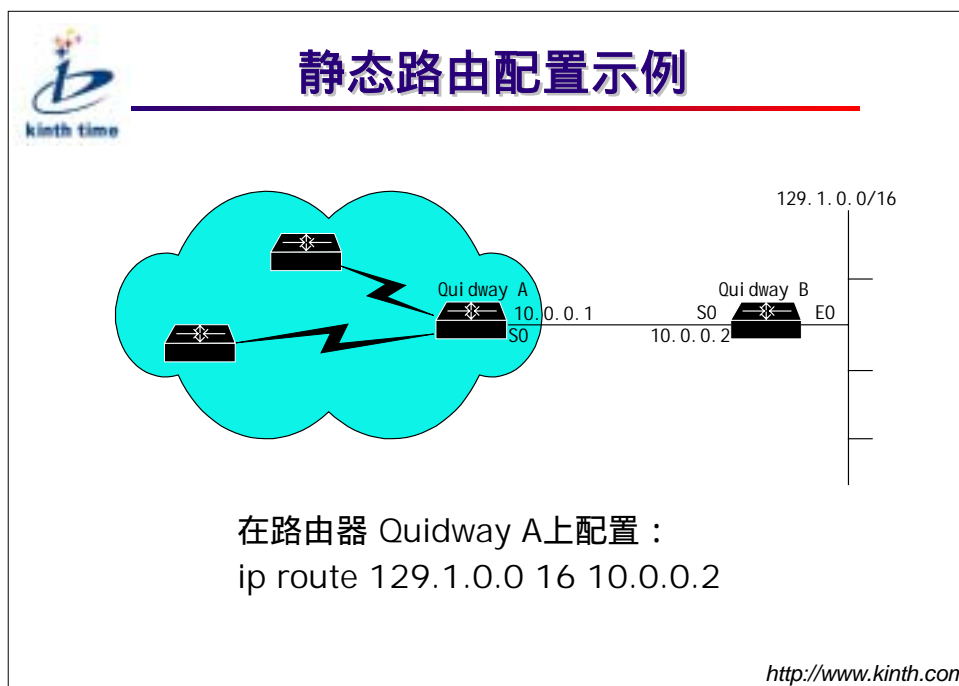
用户可以在全局配置模式下通过命令 `ip route` 来配置一条静态路由。命令的参数说明如下：

<code>ip_address</code>	目标网络的网络地址
<code>mask</code>	目标网络的子网掩码
<code>masklen</code>	目标网络的掩码长度
<code>interface_name</code>	指定去往目标网络的报文的发送接口
<code>Gateway_addr</code>	指定去往目标网络的报文经由的下一条地址
<code>preference_value</code>	静态路由加到核心路由表中的优先级

静态路由允许网管员手工配置路由表，但不能够动态的反映网络的变化，因此，此方法对保证网络不间断运行存在一定的局限性。

而在网络结构相对稳定的网络中使用静态路由就可以减少路由选择问题，并节省网络开销。同时，使用静态路由还可以实现负载平衡和路由备份功能等特殊应用。

.2.2 静态路由配置举例



在这个例子中使用 IP ROUTE 命令来配置静态路由：

```
ip route 129.1.0.0 16 10.0.0.2
```

命令

```
ip route
```

```
16
```

```
10.0.0.2
```

说明

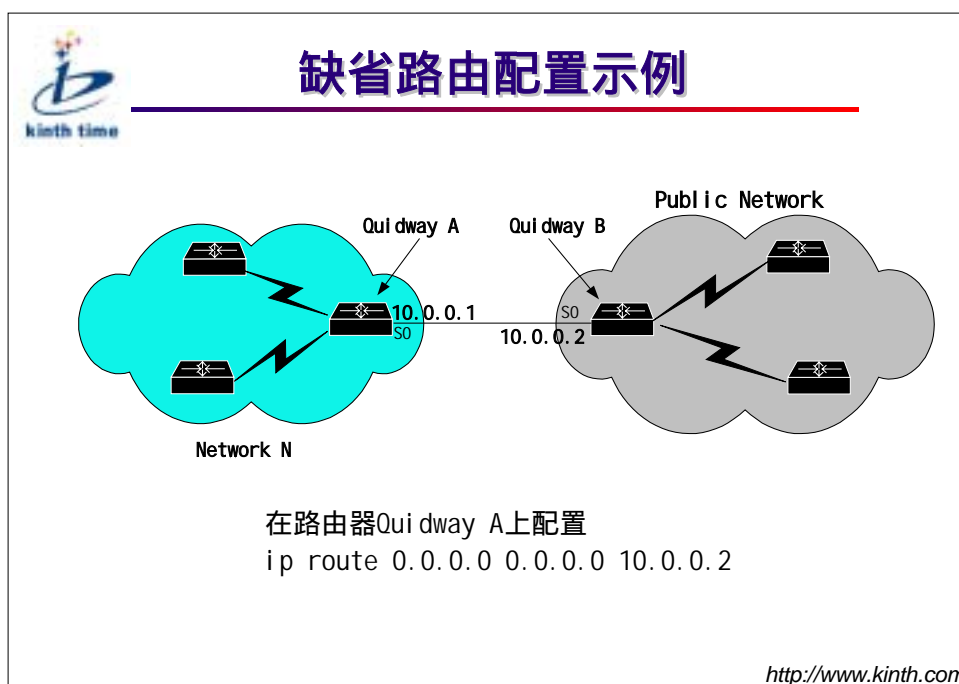
配置一条到达目标网络 (129.1.0.0) 的静态路由。

目标网络的掩码长度，也可以用点分法表示，如 255.255.0.0 。

去往目标网络的报文所经由的下一个路由器(下一跳)的 IP 地址。

通过这个配置在路由器 Quidway A 上配置到目标网络网络 129.1.0.0/16 的静态路由，此路径经过路由器 Quidway B，目标网络与路由器 Quidway B 的以太网口相连。


.2.3 缺省路由的配置及举例



在本例中，网络 N 只有一个到公网的出口，就是通过路由器 Quidway B。于是可以通过配置缺省路由使得从网络 N 内可以访问公网内的所有网络，而不必逐个网络的配置静态路由。

缺省路由的配置也使用命令 `ip route`，并且命令的格式和参数都相同，但与普通静态路由的配置不同的是，缺省路由的目标网络的地址和掩码必须全部为零。

.2.4 IP 路由配置任务



IP路由配置任务

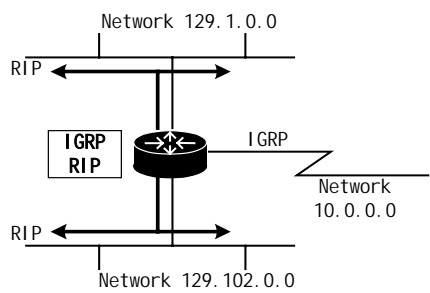
全局配置

- 选择路由协议
- 指定工作网络

接口配置

- 指定接口地址和子网掩码

```
router rip
network 129.1.0.0
network 129.102.0.0
router igrp
network 10.0.0.0
```



<http://www.kinth.com>

路由器上配置动态路由协议包括全局配置和接口配置两个方面。

在全局配置下：

选择一种路由协议，如 RIP、IGRP、OSPF 等。启动对应路由协议的处理进程。

☞ 使用全局配置命令 `router protocol [keyword]` 进行配置。

命令及参数

`router protocol`

说明

指定所选择的协议，如 RIP、IGRP、EIGRP、OSPF、BGP 等

`keyword`

进程 ID 或自治系统号，某些需要指定自治系统号协议如 BGP 要使用此参数

☞ 指定工作网络，需要注意的是，此时不能同时指定子网掩码。


命令及参数

`network network-number`

说明

指定一个直接相连的网络在接口上需要配置 IP 地址和对应网络的子网掩码。

.2.5 小结



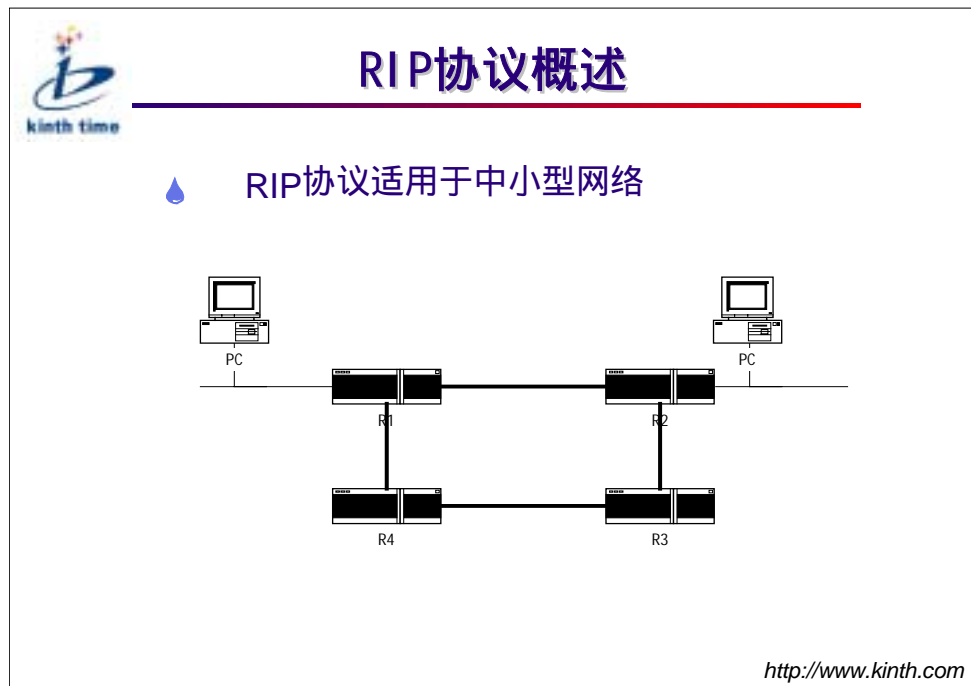
小结

- ◆ 静态路由是由网管员手动配置的路由信息
- ◆ 缺省路由是一种特殊的静态路由
- ◆ 配置路由协议、启动动态路由的方法

<http://www.kinth.com>

.3 RIP 协议及配置

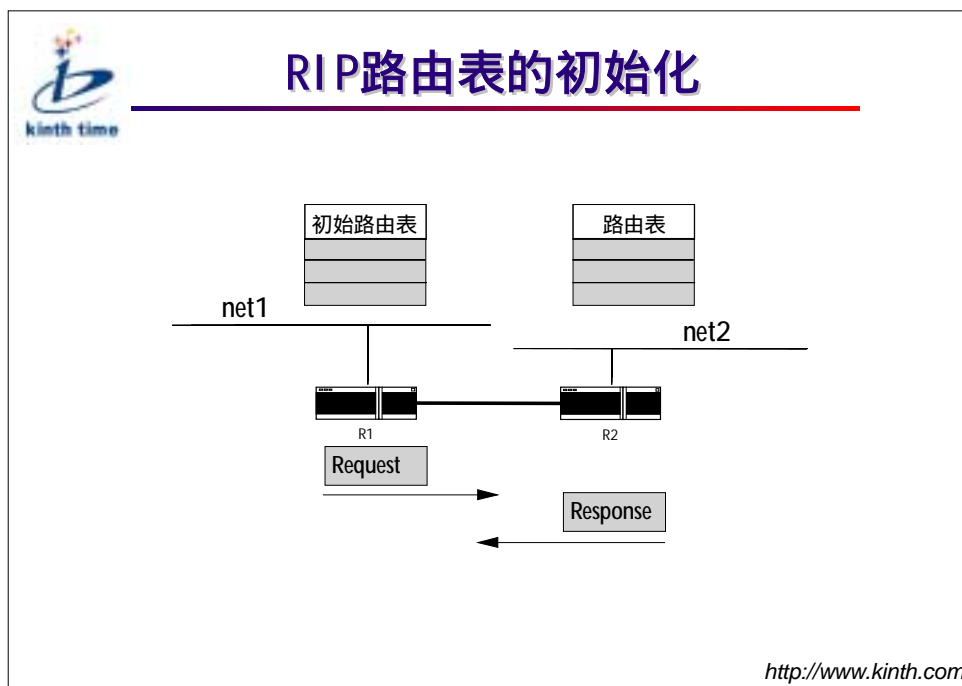
.3.1 RIP 协议概述



RIP 协议要点：

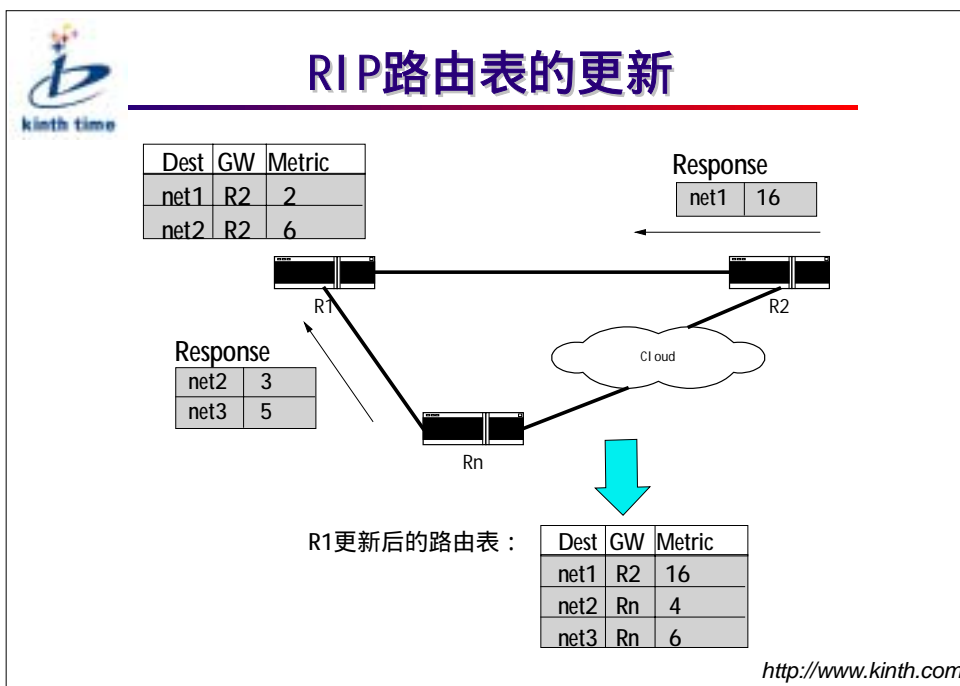
- ☞ RIP 协议基于距离向量算法，属于内部网关协议；
- ☞ RIP 协议以到达目的地址所经过的路由器个数（跳数）为衡量路由好坏的度量值，最大跳数为 15；
- ☞ RIP version 1 不支持子网掩码，version 2 支持变长掩码；
- ☞ RIP 协议适用于基于 IP 的中小型网络。

.3.2 RIP 路由表的初始化



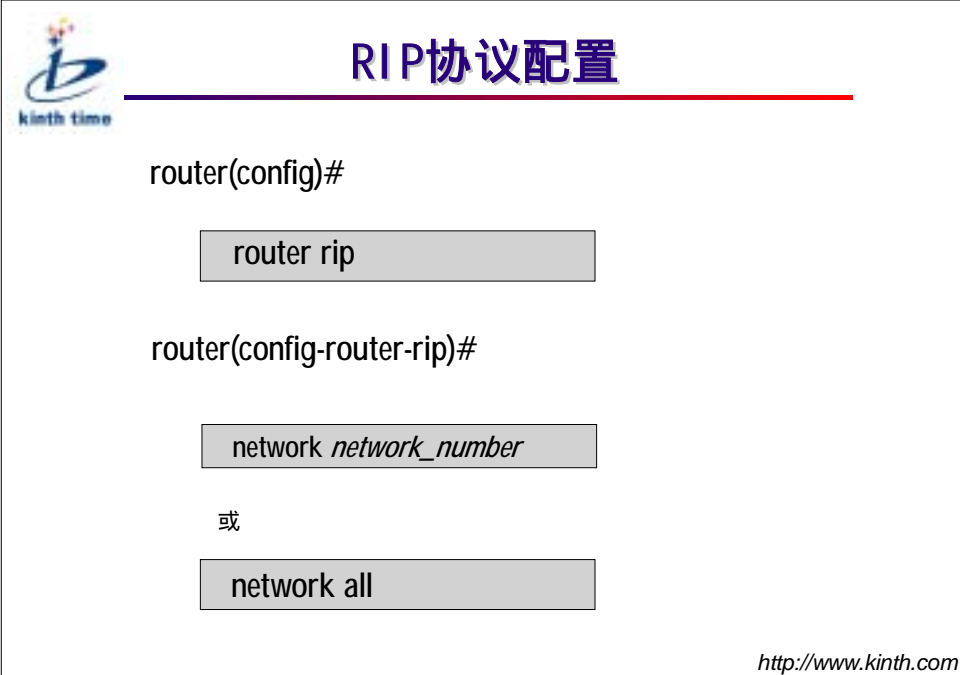
- ☞ RIP 启动时的初始路由表仅包含本路由器的一些接口路由。
- ☞ RIP 协议启动后向各接口广播一个 Request 报文。
- ☞ 邻居路由器的 RIP 协议从某接口收到 Request 报文后，根据自己的路由表，形成 Response 报文向该接口对应的网络广播。
- ☞ RIP 接收邻居路由器回复的包含邻居路由器路由表的 Response 报文，形成自己的路由表。

3.3 RIP 路由的更新



- ☞ RIP 协议以 30 秒为周期用 Response 报文广播自己的路由表。
- ☞ 收到邻居发送而来的 Response 报文后，RIP 协议计算报文中的路由项的度量值，比较其与本地路由表路由项度量值的差别，更新自己的路由表。
- ☞ 报文中路由项度量值的计算： $metric' = \text{MIN}(metric + cost, 16)$ ，metric 为报文中携带的度量值信息，cost 为接收报文的网络的度量值开销，缺省为 1 (1 跳)，16 代表不可达。
- ☞ RIP 路由表的更新原则：
 - 对本路由表中已有的路由项，当发送报文的网关相同时，不论度量值增大或是减少，都更新该路由项（度量值相同时只将其老化定时器清零）；
 - 对本路由表中已有的路由项，当发送报文的网关不同时，只在度量值减少时，更新该路由项；
 - 对本路由表中不存在的路由项，在度量值小于不可达（16）时，在路由表中增加该路由项；
- ☞ 路由表中的每一路由项都对应一老化定时器，当路由项在 180 秒内没有任何更新时，定时器超时，该路由项的度量值变为不可达（16）。
- ☞ 某路由项的度量值变为不可达后，以该度量值在 Response 报文中发布四次（120 秒），之后从路由表中清除。

.3.4 RIP 协议配置



The diagram illustrates the configuration steps for RIP. It starts with the 'router(config)#' prompt. A grey box contains the command 'router rip'. This leads to the 'router(config-router-rip)#' prompt. Two options are shown in grey boxes: 'network network_number' and 'network all', separated by the Chinese character '或' (or). The URL 'http://www.kinth.com' is at the bottom right.

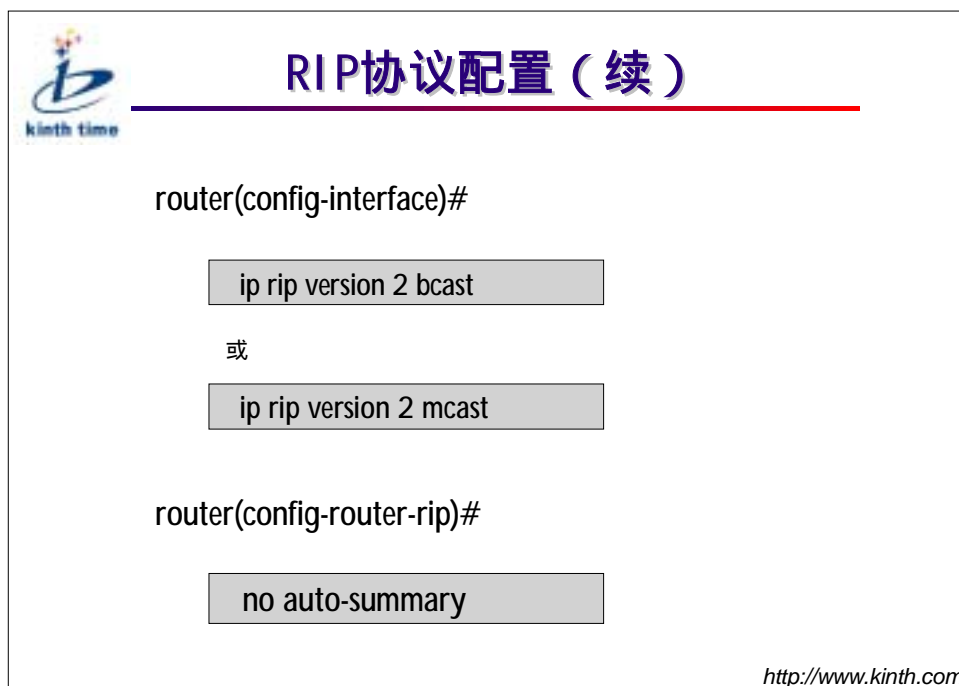
RIP协议配置

```
router(config)#  
router rip  
router(config-router-rip)#  
network network_number  
或  
network all
```

<http://www.kinth.com>

- 在全局配置模式下用 `router rip` 命令启动 RIP 协议并进入 RIP 协议配置模式。
- 在 RIP 协议配置模式下用 `network network_number` 命令在某一网段对应的接口上使能 RIP 协议。
- `network all` 命令在路由器的所有接口上使能 RIP 协议。
- 这种配置下 RIP 协议在接口上广播 version 1 类型的报文，RIP V1 不发布子网信息。

RIP 协议配置（续）



The image shows a slide titled "RIP协议配置（续）" (RIP Protocol Configuration (Continued)). It contains the following configuration commands:

```
router(config-interface)#  
    ip rip version 2 bcst  
或  
    ip rip version 2 mcast  
  
router(config-router-rip)#  
    no auto-summary
```

The slide also includes a logo for "kinth time" and a URL <http://www.kinth.com> at the bottom right.

☞ 在接口上使能 RIP version 2

在接口配置模式下使能广播方式的 RIP V2 (bcst) 或多播方式的 RIP V2 (mcast)；

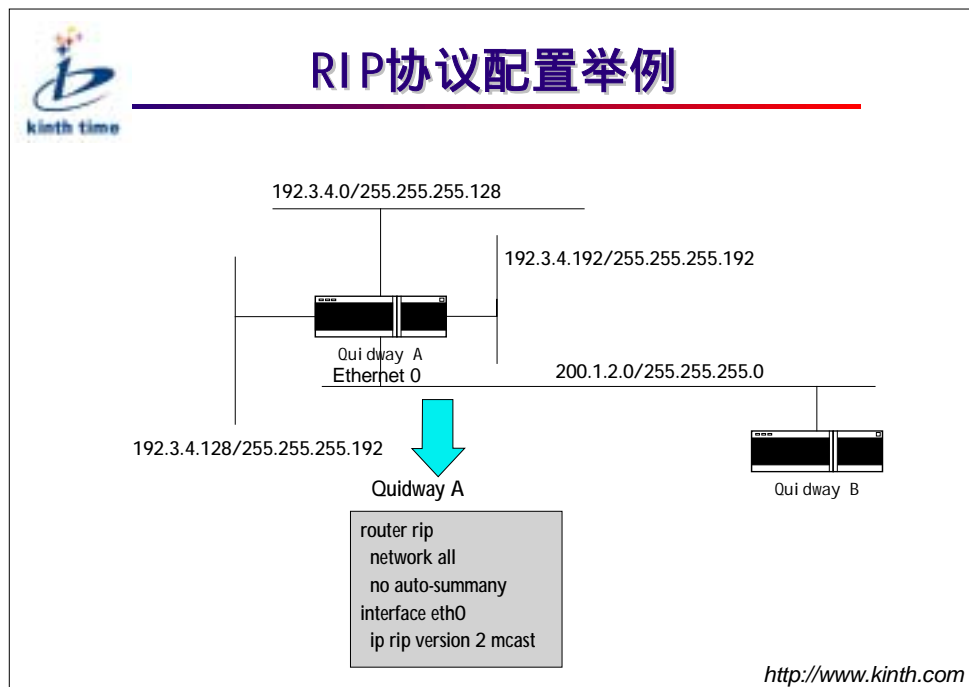
RIP 协议缺省进行路由聚合，在 RIP 协议配置模式下取消 RIP 的自动聚合功能，使其发布子网信息。

☞ RIP V2 广播方式与多播方式

RIP V2 的广播方式以广播地址 (255.255.255.255) 周期发布 RIP V2 报文，RIP V2 的多播方式以多播地址 (224.0.0.9) 周期发布 RIP V2 报文；RIP V2 缺省使用多播方式，以减少周期发布的 RIP 报文对不监听 RIP 信息的主机的影响；

RIP V2 的广播方式是 RIP V1 与 RIP V2 之间的兼容方式，以广播方式发布的 RIP V2 报文可以被 RIP V1 路由器和 RIP V2 路由器 (广播方式或多播方式) 接收，同时运行在广播方式的 RIP V2 路由器可以接收 RIP V1 的广播报文和 RIP V2 的广播或多播报文。

.3.5 RIP 配置举例



- 在全局配置模式下启动 RIP 协议。
- 在 RIP 协议配置模式下使能接口，并禁止 RIP 协议的路由聚合功能。
- 在接口配置模式下使能多播方式的 RIP V2 以发布子网信息。

.3.6 显示 RIP 协议配置信息



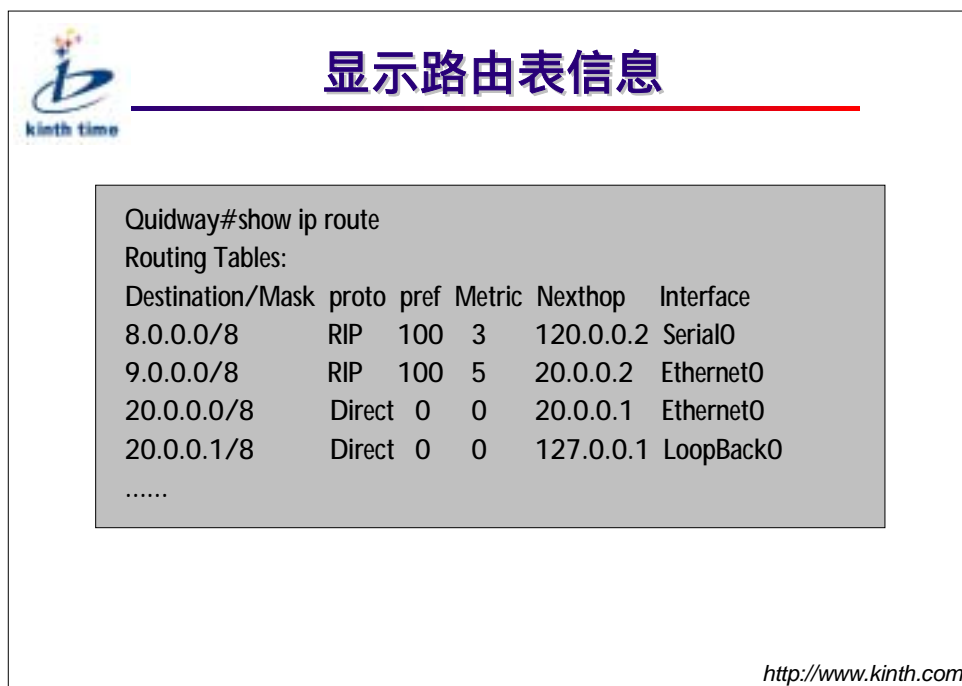
显示RIP协议配置信息

```
Quidway#show ip rip
RIP is turning on
  default-metric : 16
  no neighbor
  network : 20.0.0.0
           120.0.0.0
  auto-summary is on   preference : 100
  redistribute static metric : 2
```

<http://www.kinth.com>

- ☞ 显示当前 RIP 协议的运行状态：
- 缺省路由权为 16；
- 没有指定定点传送地址；
- 在 20.0.0.0 与 120.0.0.0 网段上使能 RIP 协议；
- 自动聚合路由；
- RIP 路由的 reference 为 100；
- 引入静态路由，并设置其度量值为 2。

.3.7 显示路由表信息



显示路由表信息

```
Quidway#show ip route
Routing Tables:
Destination/Mask proto pref Metric Nexthop Interface
8.0.0.0/8        RIP    100  3    120.0.0.2 Serial0
9.0.0.0/8        RIP    100  5    20.0.0.2 Ethernet0
20.0.0.0/8       Direct 0    0    20.0.0.1 Ethernet0
20.0.0.1/8       Direct 0    0    127.0.0.1 LoopBack0
.....
```


<http://www.kinth.com>

☞ 显示当前的路由表信息，其中有两条 RIP 路由：

RIP 路由 1：目的地址 8.0.0.0/8，下一跳为 120.0.0.2，度量值为 3；

RIP 路由 2：目的地址 9.0.0.0/8，下一跳为 20.0.0.2，度量值为 5。

.3.8 RIP 协议的 debug 信息



RIP协议的debug信息

```
Quidway#debug ip rip packet

Rip packet debugging is on
Quidway#MON
Rip : receive Response from 120.0.0.2
packet : vers 1, cmd Response, length 24
        dest 110.0.0.0, Metric 1
Rip : send from 20.0.0.1 to 255.255.255.255
packet : vers 1, cmd Response, length 44
        dest 110.0.0.0, Metric 2
        dest 120.0.0.0, Metric 1
Rip : send from 120.0.0.1 to 255.255.255.255
packet : vers 1, cmd Response, length 24
        dest 20.0.0.0, Metric 1
```


<http://www.kinth.com>

☞ debug ip rip packet 打开 RIP 协议的调试开关：

RIP 协议从 120.0.0.2 接收到一条目的地址为 110.0.0.0 的路由信息，度量值为 1；

RIP 协议向 20.0.0.1 与 120.0.0.1 分别发送路由更新信息，分别包含两条路由信息和一条路由信息。

.3.9 小结



小结

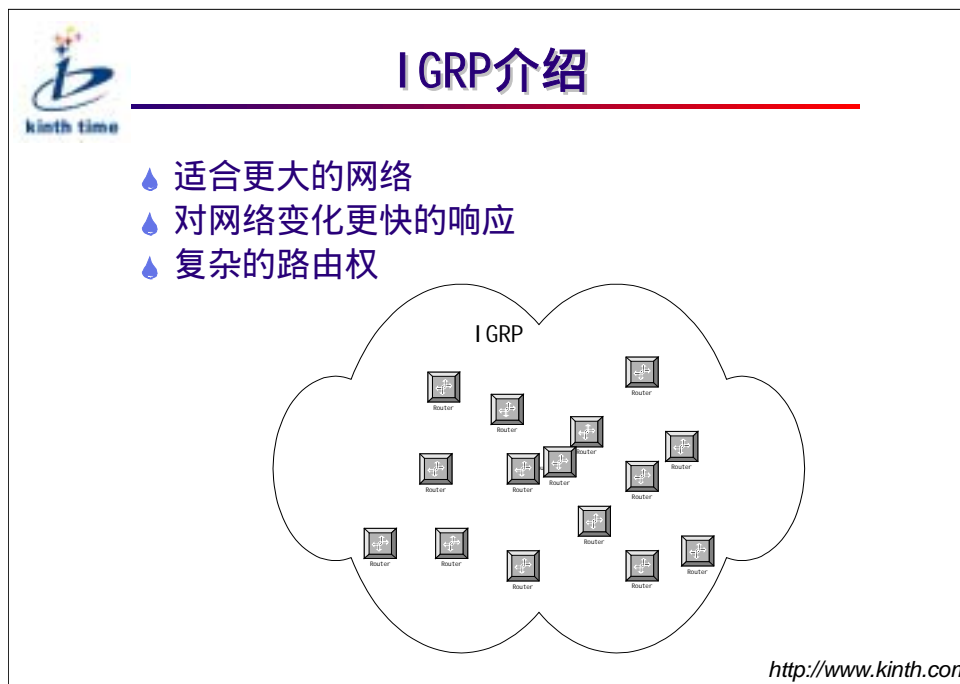
本节主要讲述了以下内容：

- ◆ RIP协议的概述
- ◆ RIP路由表的建立和更新
- ◆ RIP协议的配置及举例
- ◆ RIP协议信息的监控

<http://www.kinth.com>

.4 IGRP 协议及配置

.4.1 IGRP 简介

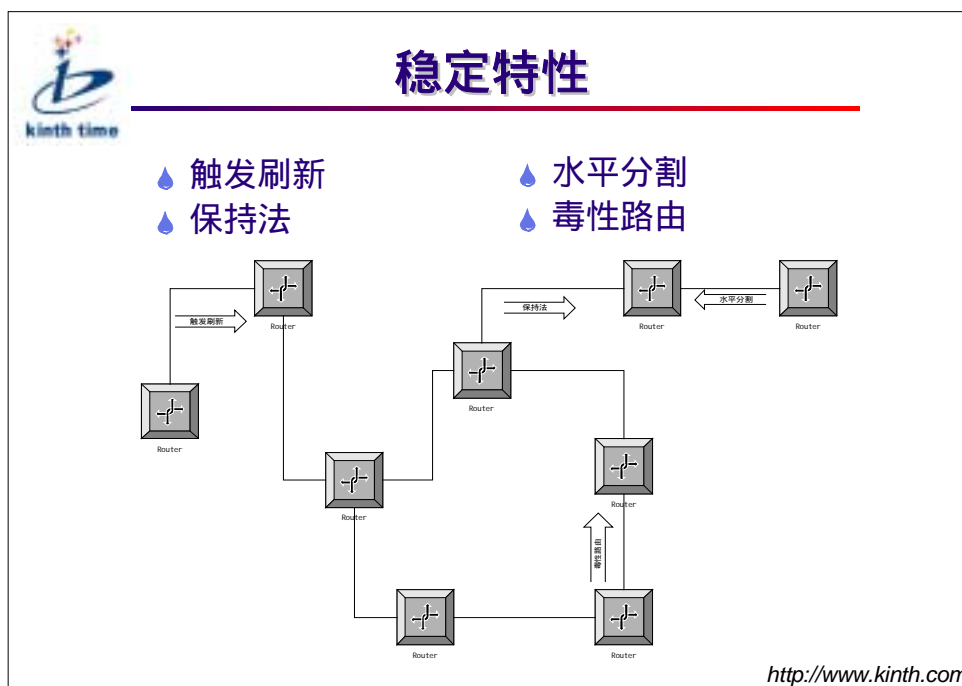


IGRP 是一个基于 D-V (Distance vector) 算法的路由协议，运行 IGRP 的路由器通过和相邻路由器之间相互交换路由信息来建立路由表。IGRP 是从 RIP 基础之上发展而来的。它比较 RIP 而言，主要有以下几点改进：

- ☞ IGRP 路由的跳数不再受 16 跳的限制，同时在路由更新上引入新的特性，使得 IGRP 协议适用于更大的网络；
- ☞ 引入了触发刷新、路由保持、水平分割和毒性路由等机制，使得 IGRP 对网络变化有着较快的响应的速度，并且在拓扑结构改变后仍然能够保持稳定。
- ☞ 在 Metric 值的范围和计算上有了很大的改进，使得路由的选择更加准确，同时使路由的选择可以适应不同的服务类型。

运行 IGRP 协议的路由器通过广播地址向相邻的路由器周期性的发送自己的路由表，同时当它收到相邻路由器发送的路由表后，根据收到的路由表增加、删除、修改本地的路由表，以达到全局路由的一致性。

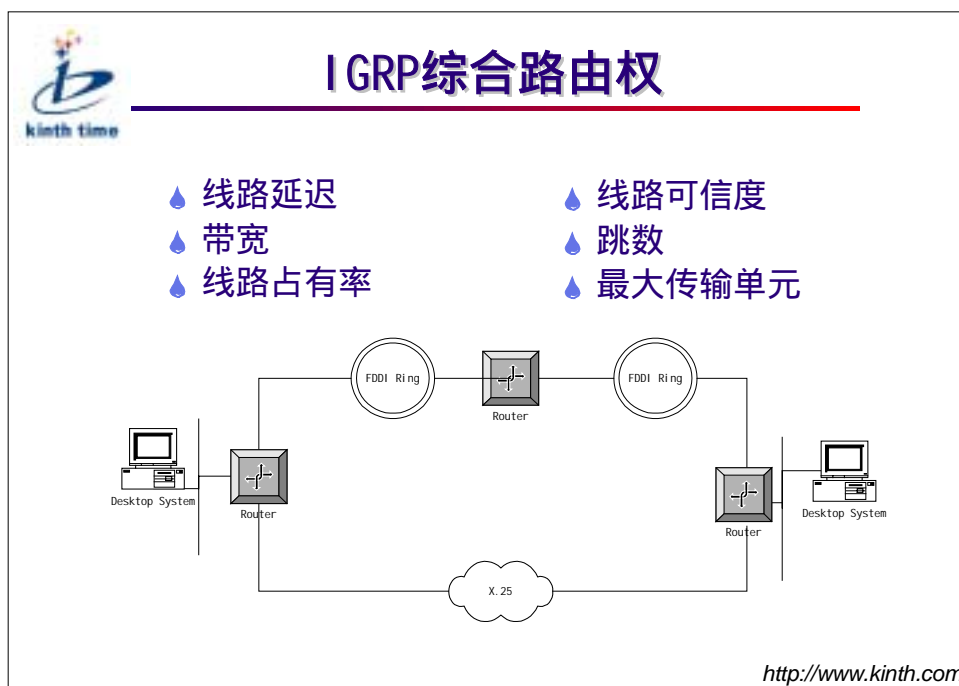
4.2 稳定特性



动态路由协议的基本功能是当网络中的路由发生改变时，将此改变迅速有效的传递到网络中的每一台路由器。同时，由于网络传递的不可靠、时延等各种偶然因素的存在，可能造成路由信息的反复变化，从而导致网络的不稳定。IGRP 协议引入了引入了触发刷新、路由保持、水平分割和毒性路由等机制，较为有效的解决了这些问题：

- ☞ 触发刷新：当路由发生改变，立即将新发生改变的路由送出，而不必等到下一次的周期性刷新，从而使得最新的路由信息很快地传送到网络中的各个路由器；
- ☞ 路由保持：路由保持是指当一条路径被删除后，此路由在一定的时间内要以不可达发送，在此段时间内即使有可达路径的报文,也丢弃不理。这样做可以使不可达路由信息在不可靠传送的情况可以最大限度的发送出去，而不会丢失和引起网络波动；
- ☞ 水平分割：水平分割规定不能将从某一网关送来的路由信息再送回此网关。即它如果要发送刷新报文给相邻网关 A,那么必须把路由中 A 送来的信息全部去掉，这样可以有效地避免相邻网关中环路的形成；
- ☞ 毒性路由：毒性路由是指如果一条路由的刷新使它的路由权的增长率大于某一比率，则此路由必须删除，并使其处于 Holddown 状态。这样做可以免在网络中形成更大的环路。

.4.3 综合路由权



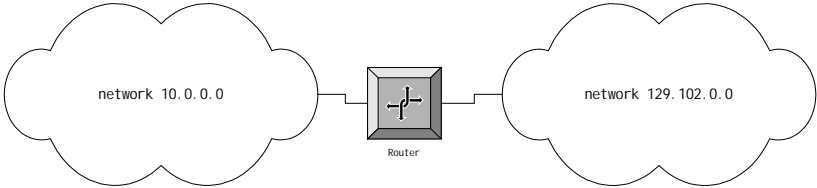
路由权是路由协议在计算路由时的主要依据，所以路由权的定义对路由的选择有着重要的影响。网络结构千变万化，单纯的跳数根本无法反映实际的网络结构，所以 IGRP 协议使用综合路由权，使得 IGRP 协议对网络路径的计算更加准确。IGRP 协议的综合路由权包括如下内容：

- ☞ 带宽：网络的带宽，单位 kbytes/s，范围 0~16777215；
- ☞ 时延：网络的时延，每单位代表 10 微秒，范围 1~4294967295；
- ☞ 信道可信度：网络传输的可靠性，范围 1~255，这里 255 代表 100% 可信；
- ☞ 信道占用率：网络的当前占用率，范围 1~255，这里 255 代表 100% 被占用；
- ☞ 最大传输单元：接口的最大传输单元，单位字节，范围 1~65535；
- ☞ 跳数：路径每经过一台路由器为一跳。

在实际计算路由权时，通常情况下不考虑信道可信度和占用率，最大传输单元根据实际接口特性获得，以下列举几个典型网络的带宽和时延：

- 卫星传输：时延 2,000,000 ms，带宽 500Mbit；
- 10M 以太网：时延 1,000 ms，带宽 10,000Kbps；
- 64K 专线：时延 20,000 ms，带宽 64Kbps。

.4.4 IGRP 的配置



IGRP的配置

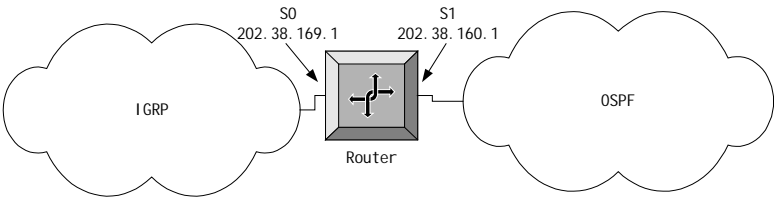
```
Quidway(config)#router igrp
Quidway(config-router-igrp)#asystem 10
Quidway(config-router-igrp)#network 10.0.0.0
Quidway(config-router-igrp)#network 129.102.0.0
```

<http://www.kinth.com>

IGRP 协议的配置很简单，主要有以下几个步骤：

- ☞ router igrp 命令启动 IGRP 协议进程；
- ☞ asystem 10 命令配置 IGRP 的自治系统号，此自治系统号要求和对端路由器的自治系统号保持一致；
- ☞ network 10.0.0.0 和 network 129.102.0.0 命令分别在相应的网络范围内的接口上使能 IGRP 协议。

.4.5 引入其他协议路由



引入路由其他协议路由

```
Quidway(config)# router igrp
Quidway(config-router-igrp)# network 202.38.169.0
Quidway(config-router-igrp)# default-metric 1000 100 250 100 1500
Quidway(config-router-igrp)# redistribute ospf
```

<http://www.kinth.com>


路由器一般可以支持多种路由协议，各种路由协议之间可以通过互相引用来共享彼此的路由信息。

IGRP 协议在引入其他协议路由时可以设定引入路由的路由权，如果没有设定，则必需使用 `default-metric` 命令设定缺省路由权，没有设定引入路由权的引入路由协议会使用缺省路由权作为它的路由权。注意：缺省路由权的缺省值为不可达，所以引入路由时一定要设定引入路由权或设定缺省路由权。

`Default-metric` 命令的配置表示：路径的带宽 1000kb/s，拓扑延迟 1000 微秒，路径的可信度 98%，路径的通道占用率 39%，最大传输单元 1500 字节。

注意路由器上需配置了 OSPF 协议。

.4.6 IGRP 协议的监控和调试



IGRP协议的监控和调试

- 💧 显示IGRP协议的当前配置情况

```
Quidway# show ip igrp
```

- 💧 在当前终端显示IGRP的调试信息

```
Quidway#debug ip igrp packet
Quidway#monitor
```

<http://www.kinth.com>

对 IGRP 协议的监控和调试主要使用以下命令：

show ip igrp 命令可以显示当前 IGRP 协议中各个配置项的值，包括缺省配置的参数值。

debug ip igrp packet 命令打开 IGRP 协议的调试开关，可以看到 IGRP 协议的收发报文情况。下面是一个 IGRP 的典型收发报文的示例：

Quidway 路由器收发对端路由器收到的报文：

```
IGRP:receive update from 12.0.0.4 (Ethernet0)
packet:vers 1, edition 1, as 1, interior 0, system 1, exterior 0, length 26
       dest 98.0.0.0 , metric 180571,hop 0
```

Quidway 路由器发送更新报文：

```
IGRP: send update 12.0.0.1 to 255.255.255.255 (ethernet0)
packet: vers 1, edition 8, as 1, interior 0, system 2,exterior 0,length 40
       dest 30.0.0.0 , metric 1041700,hop 0
       dest 20.0.0.0 , metric 1041700,hop 0
```

.4.7 小结



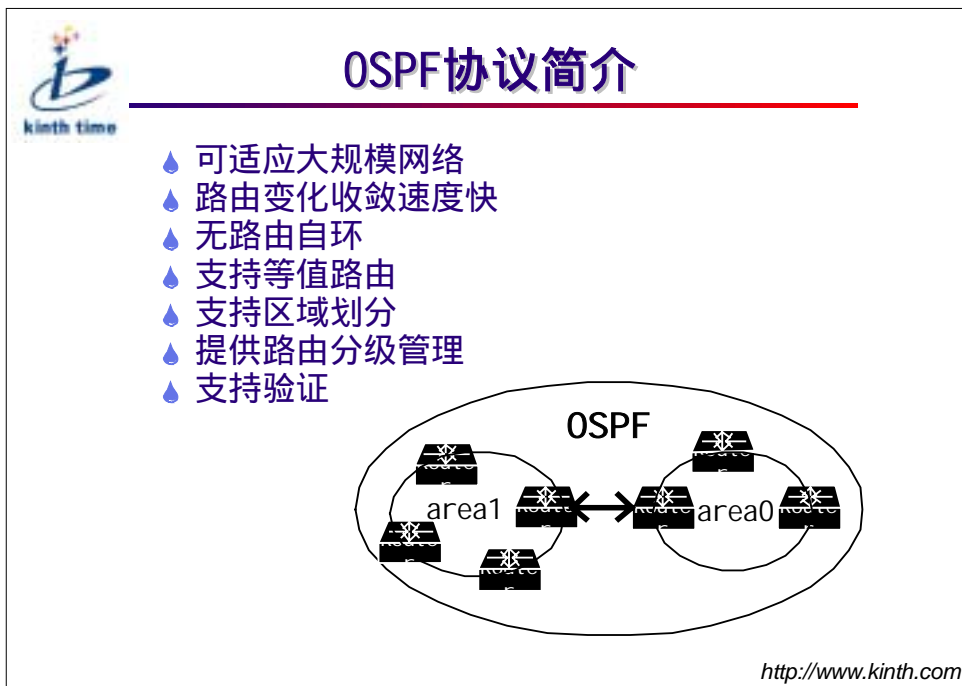
小结

- 💧 IGRP协议是一个增强型V-D算法的路由协议
- 💧 IGRP协议具有以下特点:
 - 适应更大的网络
 - 更加稳定的特性
 - 准确的路径计算

<http://www.kinth.com>

.5 OSPF 协议及配置

.5.1 OSPF 协议概述



OSPF 是 Open Shortest Path First (即“开放最短路由优先协议”)的缩写。它是 IETF 组织开发的一个基于链路状态的自治系统内部路由协议。在 IP 网络上，它通过收集和传递自治系统的链路状态来动态地发现并传播路由。

☞ 适应范围 —— OSPF 支持各种规模的网络，最多可支持几百台路由器。

☞ 快速收敛 —— 如果网络的拓扑结构发生变化，OSPF 立即发送更新报文，使这一变化在自治系统中同步。

☞ 无自环 —— 由于 OSPF 通过收集到的链路状态用最小生成树算法计算路由，故从算法本身保证了不会生成自环路由。

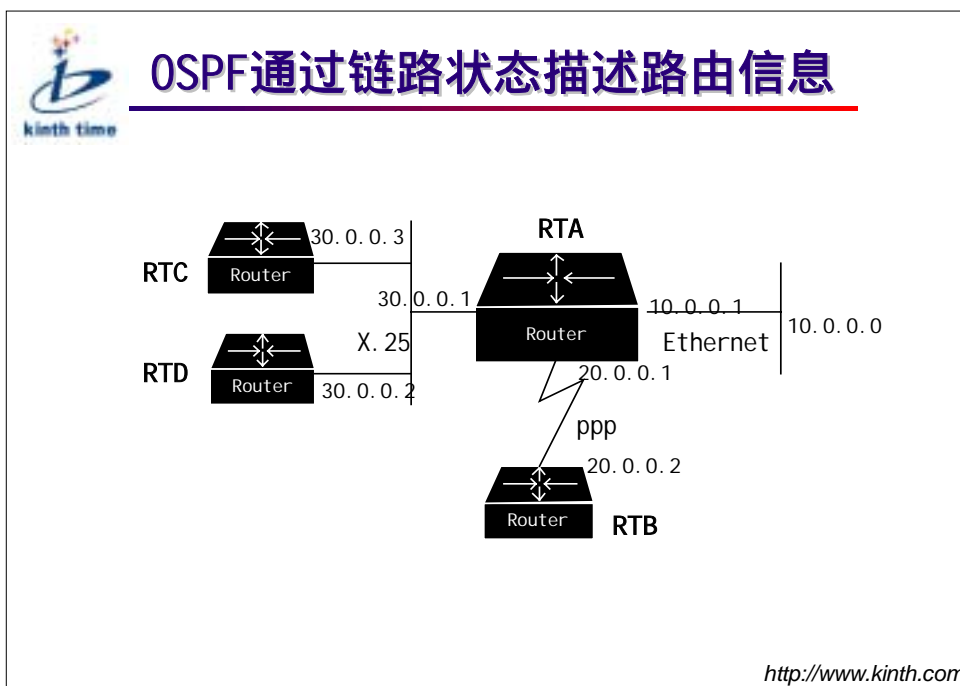
☞ 区域划分 —— OSPF 协议允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，从而减少了占用网络的带宽。

☞ 等值路由 —— OSPF 支持到同一目的地址的最多三条等值路由。

☞ 路由分级 —— OSPF 使用 4 类不同的路由，按优先顺序来说分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。

☞ 支持验证 —— 它支持基于接口的报文验证以保证路由计算的安全性。

5.2 链路状态

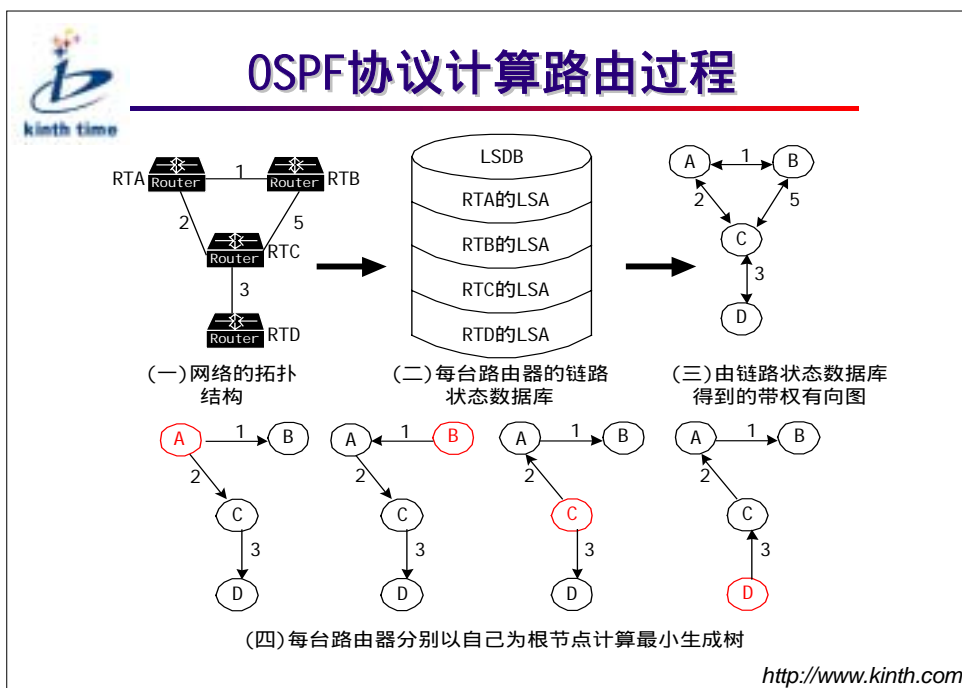


上图中 RTA 通过 PPP 协议与另一台路由器 RTB 直接相连，通过一个 X.25 网络与 RTC 和 RTD 相连，并且 RTA 连接着一个局域网。RTA 通过如下的一条 LSA（链路状态广播）来描述周边网络的拓扑结构。

```

连接数目 = 3 ; 本路由器一共有三个连接
/* 对 X.25 网络的描述*/
连接标识 = 30.0.0.3 ; 本网段中某台路由器的 IP 地址.
连接数据 = 30.0.0.1 ; RTA 连接到本网段的接口的 IP 地址
连接类型 = 2 ; 连接的类型是一个转换网段（网段中
还有其它 路由器）
连接花费 = 1 ; 从 30.0.0.1 接口发送报文的花
费值
/* 对 Ethernet 的描述*/
连接标识 = 10.0.0.0 ; 本网段的地址
连接数据 = 0xff000000 ; 本网段的掩码
连接类型 = 3 ; 连接的类型是一个末端网段（网段中没有
其它 路由器）
连接花费 = 2 ; 从 10.0.0.1 接口发送报文的花费
值
/* 对 ppp 的描述*/
连接标识= 20.0.0.2 ; 邻接点 RTB 的路由器标识(router id)
连接数据= 20.0.0.2 ; 邻接点 RTB 的 IP 地址
连接类型= 1 ; 连接的类型是另一台路由器
连接花费= 8 ; 从 20.0.0.1 接口发送报文的花费值
    
```


5.3 计算路由



上图中描述了通过 OSPF 协议计算路由的过程。

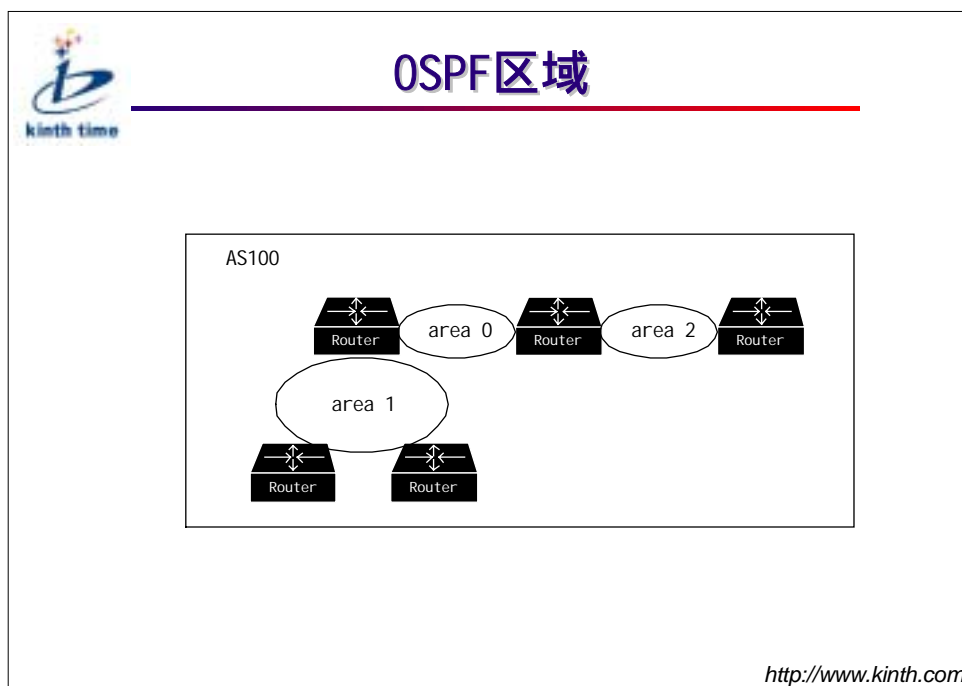
(一) 由四台路由器组成的网络，连线旁边的数字表示从一台路由器到另一台路由器所需要的花费。为简化问题，我们假定两台路由器相互之间发送报文所需花费是相同的。

(二) 每台路由器都根据自己周围的网络拓扑结构生成一条 LSA (链路状态广播)，并通过相互之间发送协议报文将这条 LSA 发送给网络中其它的所有路由器。这样每台路由器都收到了其它路由器的 LSA，所有的 LSA 放在一起称作 LSDB (链路状态数据库)。显然，4 台路由器的 LSDB 都是相同的。

(三) 由于一条 LSA 是对一台路由器周围网络拓扑结构的描述，那么 LSDB 则是对整个网络的拓扑结构的描述。路由器很容易将 LSDB 转换成一张带权的有向图，这张图便是对整个网络拓扑结构的真实反映。显然，4 台路由器得到的是一张完全相同的图。

(四) 接下来每台路由器在图中以自己为根节点，使用相应的算法计算出一棵最小生成树，由这棵树得到了到网络中各个节点的路由表。显然，4 台路由器各自得到的路由表是不同的。这样每台路由器都计算出了到其它路由器的路由。

.5.4 区域划分



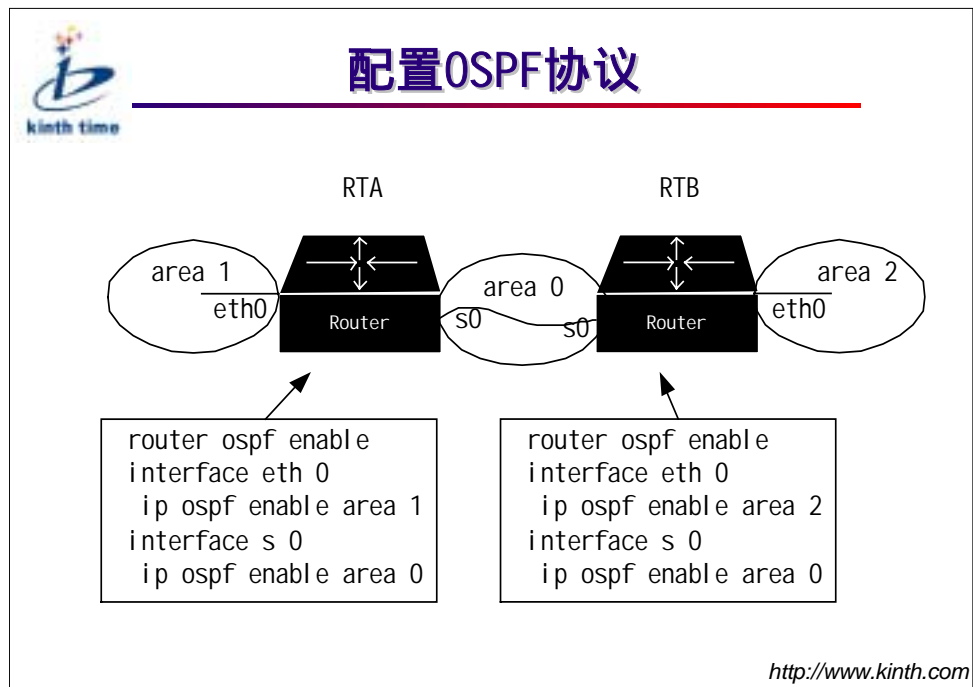
OSPF 协议允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，从而减少了占用网络的带宽。

本图中在 AS100 内运行 OSPF 协议，自治系统被划分为三个不同的区域，分别用不同的区域号 (AREA ID) 来标识。其中区域号为 0 的区域被称作“骨干区域”。

注意：

如果自治系统被划分成一个以上的区域，则必须有一个区域是骨干区域，并且保证其它区域与骨干区域直接相连或逻辑上相连，且骨干区域自身也必须是连通的。

.5.5 OSPF 协议配置



本例中 RTA 在两个接口上配置 OSPF 协议，以太网配置为区域 1，串口 S0 配置成区域 0。

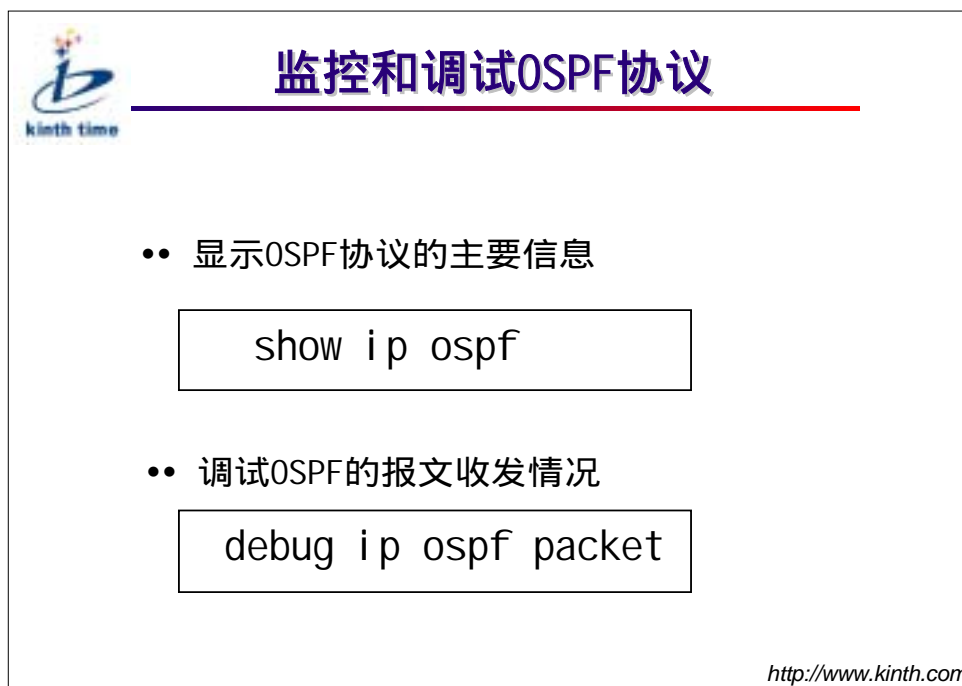
命令

```
router ospf enable
ip ospf enable area 0
```

含义

启动 OSPF 协议
指定本接口运行的区域号

.5.6 调试和监控



监控和调试OSPF协议

- 显示OSPF协议的主要信息

```
show ip ospf
```

- 调试OSPF的报文收发情况

```
debug ip ospf packet
```

<http://www.kinth.com>

用以上两条命令监控和调试 OSPF 协议。

show ip ospf 命令可以查看当前路由器配置 OSPF 的情况 :路由器的标识 (router id), 区域状态, 接口状态, 引入的外部路由情况等。

debug ip ospf packet 可以监控 OSPF 协议收发报文的情况,并打印出报文内容。

.5.7 小结




OSPF小结

- OSPF是基于链路状态的协议
- OSPF通过最小生成树来产生路由
- OSPF将整个网络化分成不同的区域

<http://www.kinth.com>

.6 BGP 协议及配置

.6.1 BGP 简介




BGP简介

- ◆ 边界路由协议(Border Gateway Protocol)
- ◆ 自治系统之间传播路由的动态路由协议的标准
- ◆ 路由信息记录了它经过的自治系统，是一种向量路由，保证了无环路
- ◆ 外部路由协议，与内部路由协议协同工作
- ◆ 支持无类别域间路由(CIDR)

<http://www.kinth.com>

BGP 的最初版本在 1989 年提出，发展到 1993 年开始开发的 BGP4，它是自治系统之间的事实上的路由协议的标准。边界是指自治系统的边界。它是一种外部路由协议，与 OSPF、RIP 等内部路由协议不同，其着眼点不在于发现和计算路由，而在于控制路由的传播和选择最好的路由。因此，BGP 不是单独工作的，它同内部路由协议协同，内部路由协议（如 OSPF）在自治系统内工作，而 BGP 在自治系统之间工作。它是第一个支持 CIDR 的路由协议，通过路由聚合可以有效的抑制因特网上路由的爆炸性增长。通过携带 AS 路径信息，它可以彻底解决路由循环问题。

.6.2 BGP 的特点



BGP的特点


- ◆ **简明**
对网络拓扑无限制，只有四种报文
- ◆ **可靠**
使用TCP作为传输协议
- ◆ **有效**
发送增量路由，而非周期性广播
- ◆ **灵活**
路由携带属性，具有丰富的路由策略，控制路由在入口和出口处的选择

<http://www.kinth.com>


BGP 协议看起来很简单，它并不需要规划网络的拓扑，事实上，Internet 并不是从上而下由某个组织建立起来的，而是一些网络自下而上互相连接而成的，每个这样的网络称为一个自治系统。

而由于政治的、经济的原因，每个自治系统希望对路由进行过滤、选择和控制，因此，BGP 路由携带了丰富的属性，由 BGP 的路由策略来使用，正是这一特性使得 BGP 是如此简明而又如此灵活和强大，它还使得 BGP 便于扩展，以支持因特网新的发展。BGP 协议使用 TCP 作为其传输层协议，不仅提高了协议的可靠性，而且使得发送增量路由成为可能，这就大大减少了 BGP 传播路由所占用的带宽，适用于在 Internet 上传播大量的路由信息。

.6.3 BGP 的适用范围




如何选择使用BGP



使用BGP

- ◆ 一般用于ISP之间
- ◆ 同两个或多个ISP连接
- ◆ 为客户提供部分或完全的Internet路由



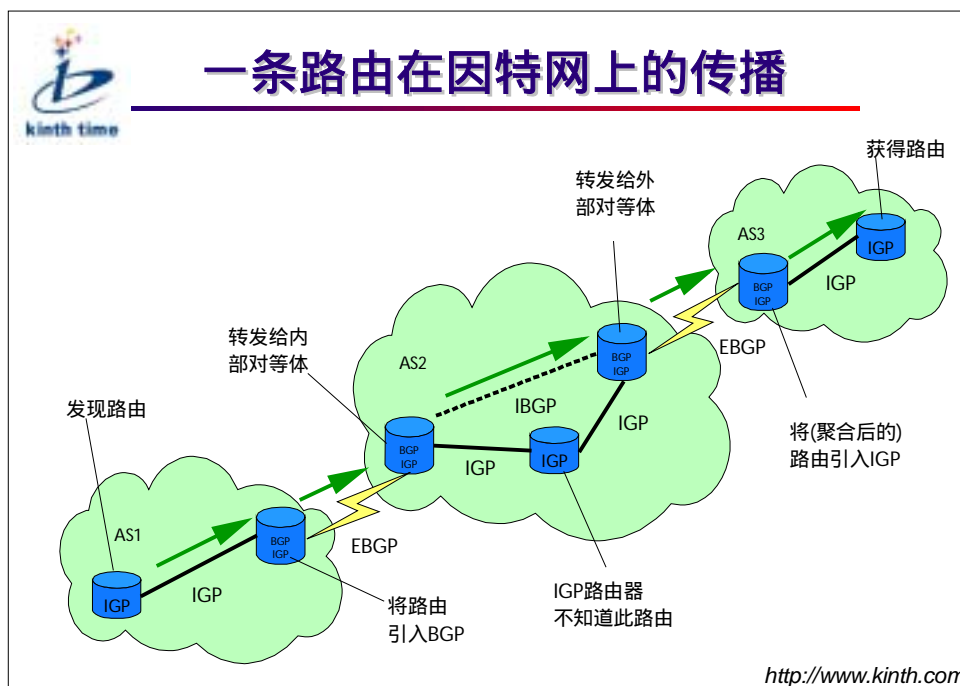
不使用BGP

- ◆ 只同一个ISP相连
- ◆ 不向客户提供Internet路由服务
- ◆ 使用默认路由时
- ◆ 典型的局域网，Intranet

<http://www.kinth.com>


正如前面所述的，BGP 用在自治系统之间。一般来说，在 ISP 之间才需要使用 BGP，在这时，你要同多个 ISP 连接，需要在多个相同目的地的路由之间进行选择，并且为客户提供 Internet 路由。如果你只是 ISP 的客户，同一个 ISP 连接，最简单地，可以使用一个默认路由来指向 ISP，并不需要使用 BGP。对于大多数的局域网和 Intranet 来说，BGP 是一种奢侈品，只有你准备同多个 ISP 连接或成为一个 ISP 时，才使用 BGP。

.6.4 BGP 路由的传播



一条路由在一般情况下是从自治系统内部产生的，它由某种内部路由协议发现和计算，传递到自治系统的边界，由自治系统边界路由器（ASBR）通过 BGP 传播到其它自治系统中，这种连接称为 EBGP，两个 ASBR 互称对等体。路由在传播过程中可能会经过若干个自治系统，这些自治系统称为过渡自治系统。若这个自治系统有多个边界路由器，这些路由器之间也运行 BGP 来交换路由信息，称为 IBGP。这时内部的路由器并不需要知道这些外部路由，它们只需要在边界路由器之间维护 IP 连通性。路由到达自治系统边界后，若内部路由器需要知道这些外部路由，ASBR 可以将路由引入内部路由协议。外部路由的数量是很大的，通常会超出内部路由器的处理能力，因此引入外部路由时一般需要过滤或聚合，以减少路由的数量，极端的情况是使用默认路由。

.6.5 小结




BGP小结

- ◆ BGP4是自治系统之间传播路由的动态路由协议的事实标准
- ◆ BGP和OSPF结合是大型网络的主流路由协议
- ◆ BGP为路由提供丰富灵活的控制手段
- ◆ 在必要时使用BGP

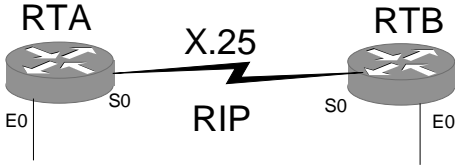
<http://www.kinth.com>

.7 配置实例

.7.1 组网介绍



配置实例



RTA :

E0 : 10.10.10.1 255.255.255.0

S0 : 12.12.12.1 255.255.255.0

36115193 (121地址)

RTB :

E0 : 20.20.20.1 255.255.255.0

S0 : 12.12.12.2 255.255.255.0

36115196 (121地址)

要求 :

走RIP协议，让两边的E0口互通

<http://www.kinth.com>

.7.2 数据配置

配置实例（续）

RTA路由器的配置：

```
RTA#config
RTA(config)#router rip
RTA(config)#int e 0
RTA(config-if-Ethernet0)#ip address 10.10.10.1 255.255.255.0
RTA(config-if-Ethernet0)#int s 0
RTA(config-if-Serial0)#ip address 12.12.12.1 255.255.255.0
RTA(config-if-Serial0)#encap x25
RTA(config-if-Serial0)#x25 address 36115193
RTA(config-if-Serial0)#x25 map ip 12.12.12.2 36115196 broadcast
RTA(config-if-Serial0)#exit
RTA(config)#exit
RTA#write
```

<http://www.kinth.com>

在 X25 的地址映射配置中，最后加了一个 broadcast 参数，目的是让 X25 能承载 IP 广播包。因为 RIP 协议是通过广播包发送路由信息的，所以如果不加这个参数的话，两台路由器之间交换不了路由信息。

数据配置（续）

配置实例（续）


RTB路由器的配置：

```
RTB#config
RTB(config)#router rip
RTB(config)#int e 0
RTB(config-if-Ethernet0)#ip address 20.20.20.1 255.255.255.0
RTB(config-if-Ethernet0)#int s 0
RTB(config-if-Serial0)#ip address 12.12.12.2 255.255.255.0
RTB(config-if-Serial0)#encap x25
RTB(config-if-Serial0)#x25 address 36115196
RTB(config-if-Serial0)#x25 map ip 12.12.12.1 36115193 broadcast
RTB(config-if-Serial0)#exit
RTB(config)#exit
RTB#write
```

<http://www.kinth.com>

RTB路由器的配置与RTA路由器的配置类似。需要注意的是在X25的地址映射配置中，不要忘记加上broadcast参数。

.8 总结



路由协议总结

- ◆ 路由及路由协议的基本概念
- ◆ 路由器上可以配置一种或多种路由协议
- ◆ 介绍了四种动态路由协议及其配置
 - RIP
 - IGRP
 - OSPF
 - BGP

<http://www.kinth.com>

.9 本章重点



本章重点

- ◆ 对路由这个概念的把握和理解
- ◆ 掌握几种常用路由协议的基本思想
- ◆ 路由的配置

<http://www.kinth.com>

路由是数据网络基本而又重要的一个概念，必须理解和掌握。距离矢量和链路状态是动态路由协议两种基本的寻径算法。