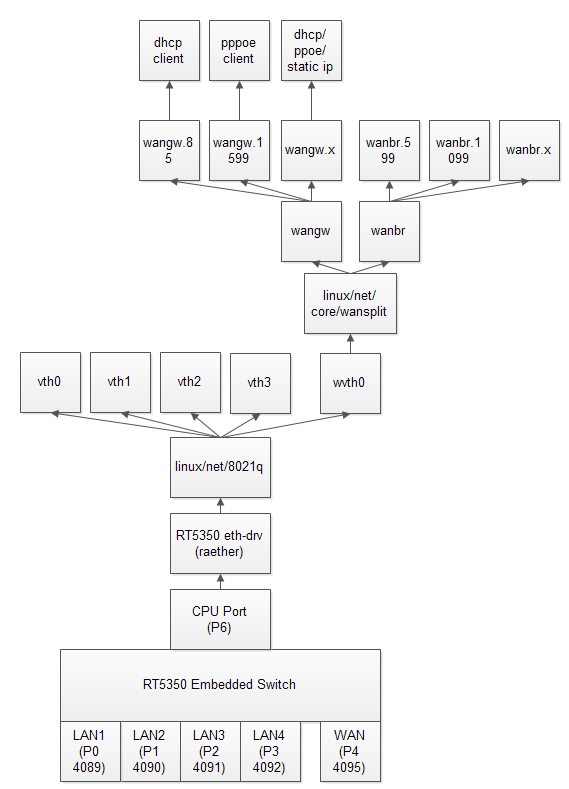
T5214EWL-OM LAN WAN口区分及报文转发流程

T5214EWL-OM开发过程中，LAN/WAN口区分和报文转发流程进行过一次调整，主要是为减少冗余流程，提升L2/L3转发性能，此外修改实施完成后总体代码流程更简单清晰。

【T5214EWL-OM L2/L3性能优化前】



主要处理流程：

* RT5350内置交换芯片：

通过保留的4089~4092,4095,把各个LAN/WAN口都区分出来。主要配置过程是在rcS脚本中的switch reg w e4 40041f, vconfig set\_vlan\_sw 1, vconfig add eth2.40XX命令完成。

* Raether以太网驱动:没做太多事情，主要是调整了一下上去报文的SPECIAL\_TAG。
* 协议栈8021q流程:从上来LAN1~LAN4/WAN口接收到的报文的vlantag都是不同的，通过8021q中的vlanid判断，把报文投递给vth0~vth3和wvth0。
* 协议栈wansplit模块：如果是wvth0接口的报文,如果报文的目的IP地址就是WM800网关模块的MAC地址，那么报文就投递给wangw接口，否则就投入给wanbr接口。如果是广播或组播报文，那么copy一份后向两个接口都投递一份。
* 协议栈8021q流程：从wangw到wangw.X,从wanbr到wanbr.X这个属于标准的8021q流程(通过用户态工具vconfig来配置)，不再赘述。

T5214EWL-OM的rcS脚本文件：

#######BEGIN:create vth0~vthX there#####

ifconfig eth2 up

sleep 3

switch reg w e4 40041f //LAN1~LAN4,WAN口启用DoubleTag功能

//启用CPU\_TPID\_EN和SPECIAL\_TAG\_EN

vconfig set\_vlan\_sw 1 //执行后会调用内核vlan.c中我们写的ra3052\_vlan\_init函数

//做的主要事情就是设置交换芯片的PVID，vlan group，是否untag等

vconfig add eth2.4089 //vlan模块中有特殊处理eth2.4089~4092自动被重命名为vth0~vth3

vconfig add eth2.4090

vconfig add eth2.4091

vconfig add eth2.4092

ifconfig vth0 up

ifconfig vth1 up

ifconfig vth2 up

ifconfig vth3 up

#######END :create vth0~vthX there#####

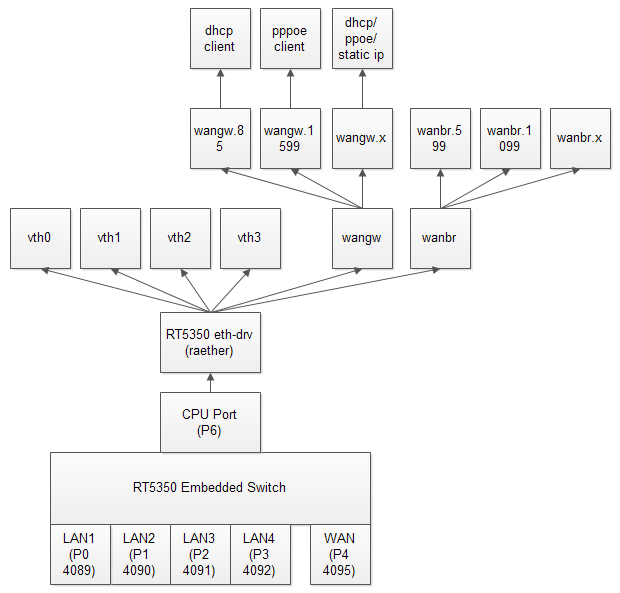
#######BEGIN:create wvth0 there#####

vconfig add eth2 4095 //vlan模块中有特殊处理eth2.4095自动被重命名为wvth0

ifconfig wvth0 up

#######END :create wvth0 there#####

【T5214EWL-OM L2/L3性能优化后】



主要处理流程：

* RT5350内置交换芯片：

没有变更，变动的仅仅是rcS脚本中直接用switch命令替换了命令。

* Raether以太网驱动:1.从LAN1~LAN4上来的报文直接投递给vth0~vth3

2.从WAN上来的报文通过判断目的MAC地址,投递给wangw和wanbr。

* 协议栈8021q流程：从wangw到wangw.X,从wanbr到wanbr.X这个属于标准的8021q流程(通过用户态工具vconfig来配置)，不再赘述。

显然少走了两个大流程，所以性能有明显和代码流程更加简单：LANx/WAN口上来的报文少走了一次8021q的流程，WAN口上来的报文少走了一次WANSPLIT模块。

T5214EWL-OM的rcS脚本文件：

#######BEGIN:create vth0~vthX there#####

ifconfig eth2 up

sleep 3

switch reg w e4 40041f //性能优化前后本行无变化,下面的switch reg和switch vlan set命令

//其实做的事情就是优化前vconfig set\_vlan\_sw 1命令做的事情

//之前用C写的很麻烦不灵活，所以这里直接用switch命令搞定了。

#set p0,p1,p2,p3,p4(LLLLW) to VLAN\_EG\_UNTAGGED,set p6(cpu-port) to VLAN\_EG\_TAGGED

switch reg w 98 7f3f //把LAN1~LAN4设置为出方向时拨vlan tag，CPU口出方向时留vlan tag

#set port pvid:p0 4089,p1 4090,p2 4091,p3 4092,p4 4095,p6 4093

switch reg w 40 ffaff9 //设置LAN1 PVID为4089，设置LAN2 PVID为4090

switch reg w 44 ffcffb //设置LAN3PVID为4091，设置LAN4 PVID为4092

switch reg w 48 1fff //设置 WAN PVID为4095

switch reg w 4c 20ffd//设置CPU口PVID为4093

switch vlan set 10 4095 0000101 //创建vlan组4095，成员端口为WAN口和CPU口

switch vlan set 11 4092 0001001 //创建vlan组4092，成员端口为LAN4口和CPU口

switch vlan set 12 4091 0010001 //创建vlan组4091，成员端口为LAN3口和CPU口

switch vlan set 13 4090 0100001 //创建vlan组4090，成员端口为LAN2口和CPU口

switch vlan set 14 4089 1000001 //创建vlan组4089，成员端口为LAN1口和CPU口

switch vlan set 15 4093 1111111 //创建vlan组4093，成员端口为LAN1~4及CPU口

ifconfig vth0 up

ifconfig vth1 up

ifconfig vth2 up

ifconfig vth3 up

######BEGIN:process interfaces and bridges here######