

MetaROUTER Overview & Configuration

Brian Vargyas

Baltic Networks

www.balticnetworks.com



Overview of this presentation

- MetaROUTER Overview
- MetaROUTER Limitations
- Different Interface types
- Real world application use
- How to make your own with OpenWRT
- How to access the RouterBOARD console port within MetaROUTER
- Demonstration of a running MetaROUTER

MetaROUTER Overview

- Scalable RouterOS within RouterOS on Atheros/PPC, x86 Platforms.
- Virtual environment allows to user to partition system into different administrative domains.
- Ability to run pre-compiled Atheros/PPC OpenWRT images
- Available since RouterOS 3.21 (4.0beta3 for PPC)

MetaROUTER Requirements

- RouterOS 3.28 recommended
- Any RB400 / 1000 & x86 Platform
- Each RouterOS instance requires at least 16MB Ram, 32MB Ram recommended
- NAND memory requirements minimal (For RouterOS only)

MetaROUTER Limitations & Faults

- Only 8 instances per RouterBOARD
- No CF or microSD devices can be used for running images
- No ability to export running virtual image back into a file (Feature Request 😊)
- OpenWRT on MetaROUTER won't properly shutdown when RouterOS reboots.
- Limited by available Ram (256MB 450G)
- No ability to monitor running states with Dude Server (Feature Request 😊)
- Host Router on occasion reboots with watchdog timeout error (V3.28)

MetaROUTER access to the world

- Console access through Winbox or CLI interface
- Two different types of virtual interfaces
- Dynamic interfaces used to connect to a bridge group you create
- Static interfaces are used to directly connect MetaROUTER instance to physical port on your host.

MetaROUTER Performance

- Performance is really good and virtualized RouterOS and OpenWRT perform very well
- Latency increased by a few milliseconds
- Expect to take a ~30% hit in packet performance through the MetaROUTER in real world applications
- Traffic through static interface slightly better than dynamic interface

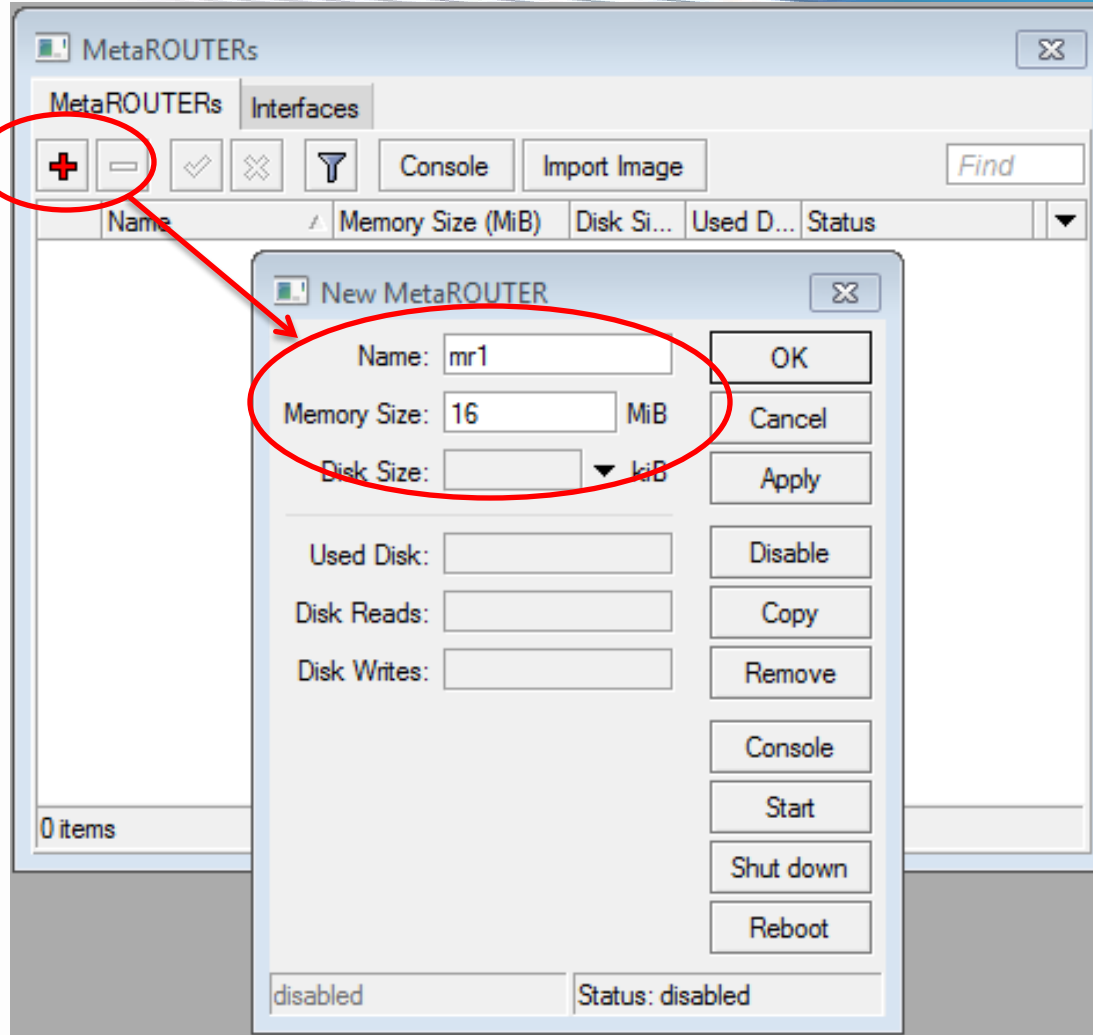
The MetaROUTER Winbox Interface

The image shows the RouterOS WinBox interface. On the left is a vertical menu with the following items: Interfaces, Wireless, Bridge, Mesh, PPP, IP, Routing, System, Queues, Files, Log, Radius, Tools, New Terminal, MetaROUTER (circled in red), Make Supout.tif, Manual, and Exit. A red arrow points from the MetaROUTER menu item to the MetaROUTERs window. The MetaROUTERs window has two tabs: MetaROUTERs and Interfaces. The MetaROUTERs tab is active and contains a table with the following data:

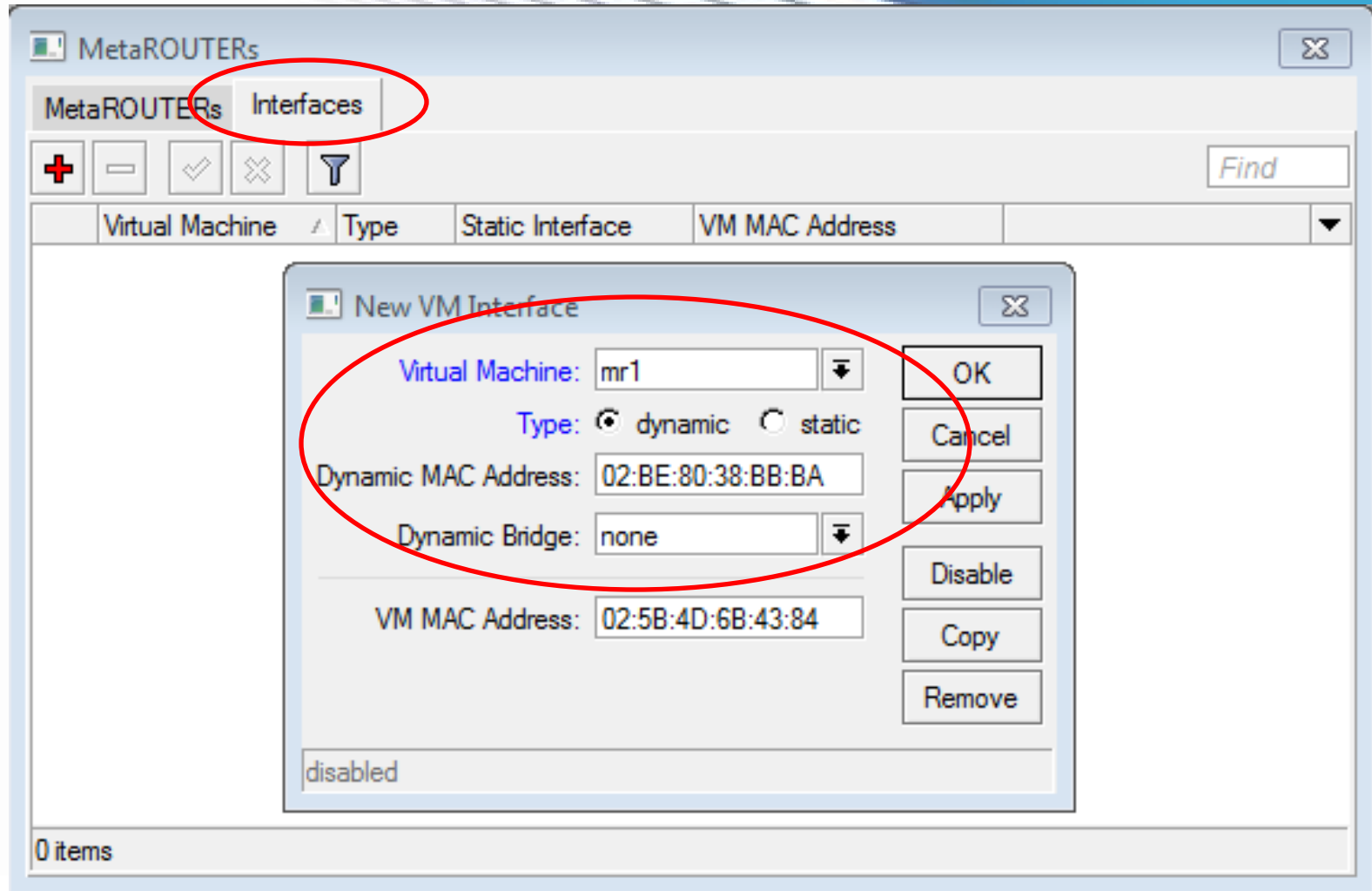
Name	Memory Size (MiB)	Disk Size (kiB)	Used Disk (kiB)	Status
mr1	16		195	running
mr2	16		195	running
mr3	16		195	running
mr4	16		195	running
mr5	16		195	running
mr6	16		195	running
mr7	16		195	running

At the bottom of the window, it says "7 items".

Creating a MetaROUTER



Dynamic Interface Creation



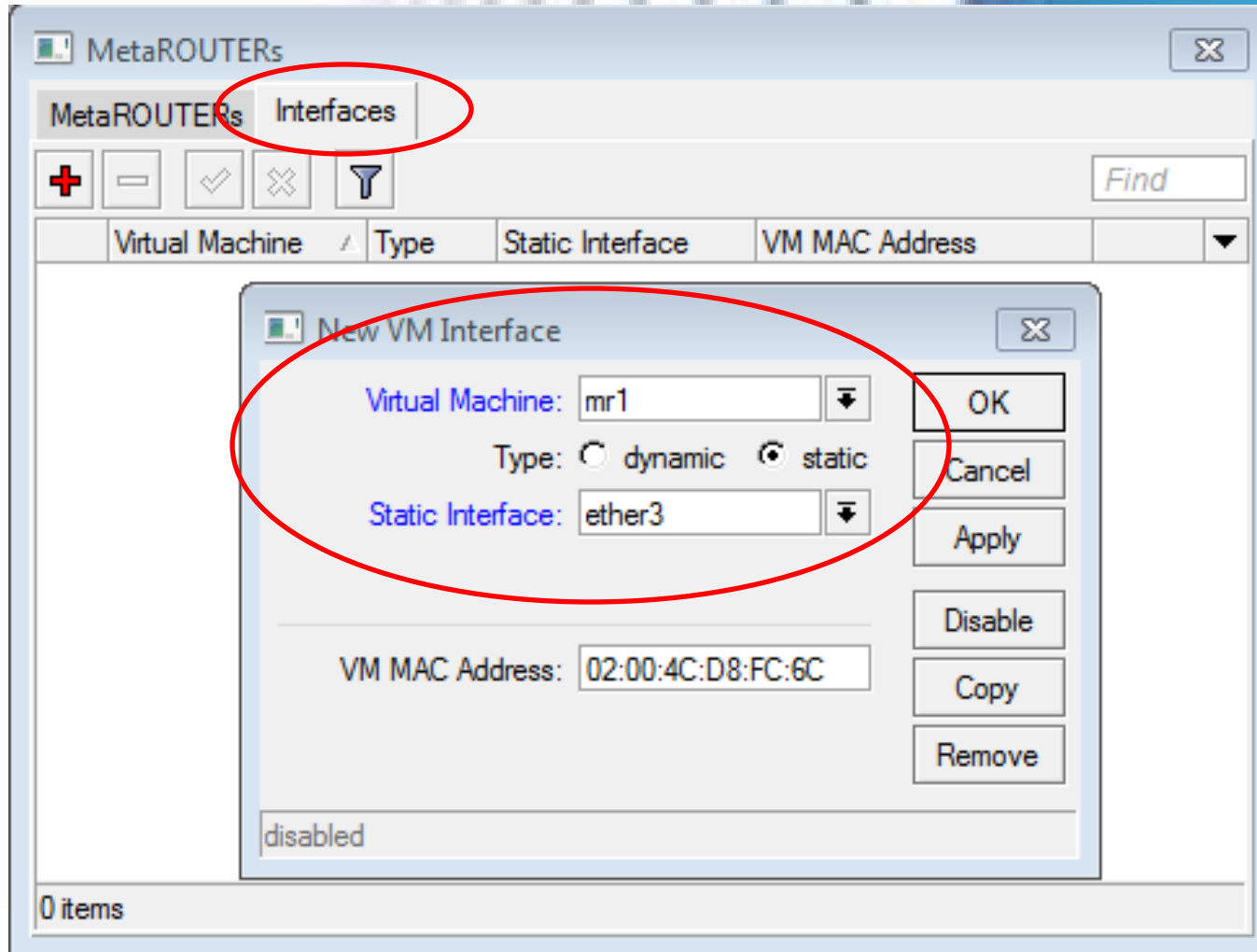
Dynamic VIF Interface

The image shows two configuration windows from a network management tool. The top window, titled 'MetaROUTERS', displays a table of interfaces for the 'mr1' virtual machine. The 'Type' column for the interface is set to 'dynamic', which is circled in red. The 'VM MAC Address' is listed as '02:5E:C1:1C:81:6C'. The bottom window, titled 'Bridge', shows a table of bridge ports. The 'vif1' interface is listed as a 'designated port' with a priority of 80 and a path cost of 10, also circled in red. The 'Bridge-Inside' interface is listed as a 'root port' with a priority of 80 and a path cost of 10. The status 'D' is visible next to 'vif1'.

Virtual Machine	Type	Static Interface	VM MAC Address
mr1	dynamic		02:5E:C1:1C:81:6C

Interface	Bridge	Priority (h...)	Path Cost	Horizon	Role	Root Pat...
↑↑ Inside	Bridge-Inside	80	10		root port	14
D ↑↑ vif1	Bridge-Inside	80	10		designated port	

Static Interface Creation



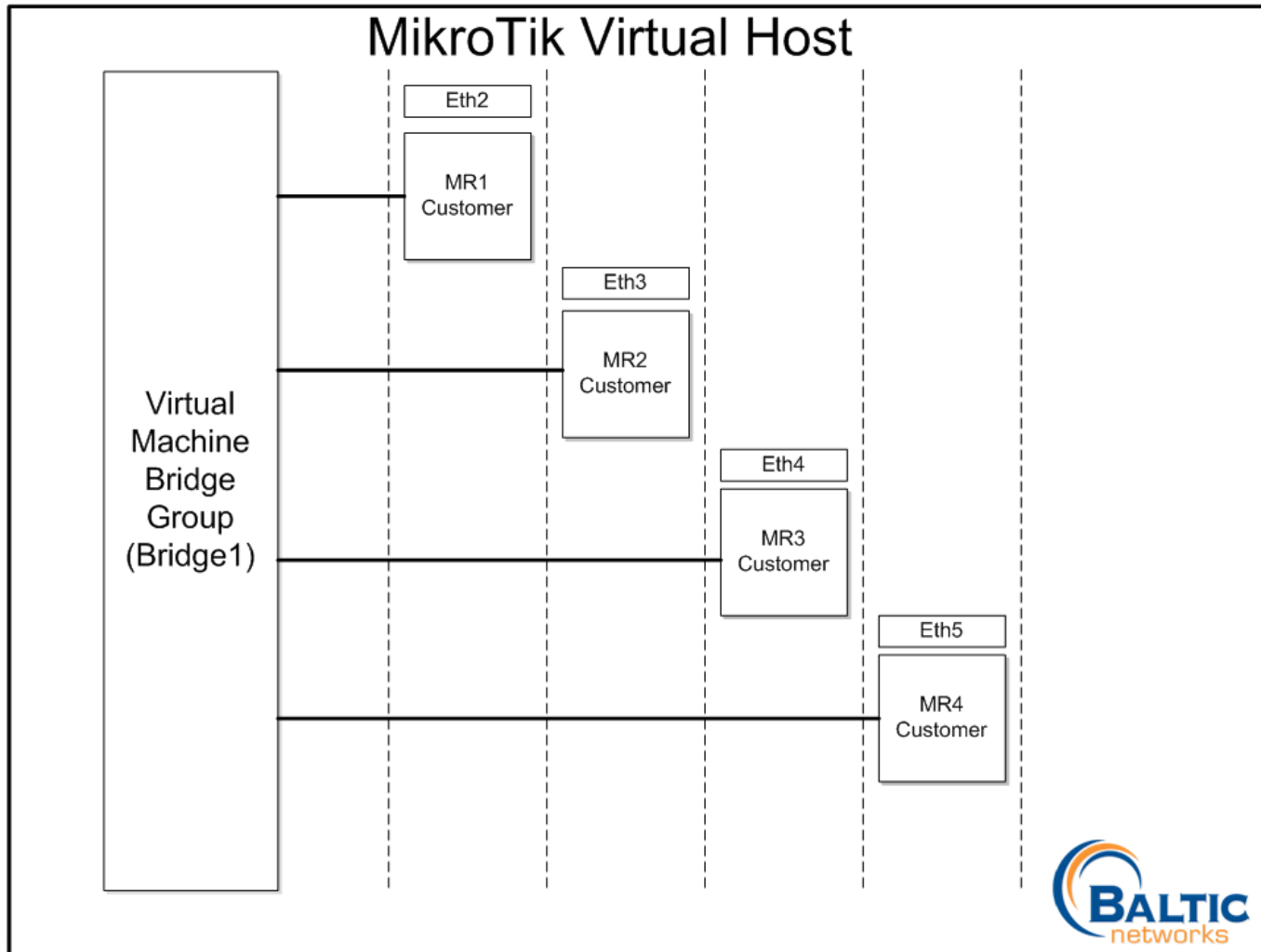
Console Access

The screenshot shows the MetaROUTERS management interface. The 'MetaROUTERS' tab is active, displaying a table of devices. A red circle highlights the 'Console' button in the toolbar, with a red arrow pointing to the console window below. The console window shows the MikroTik login prompt.

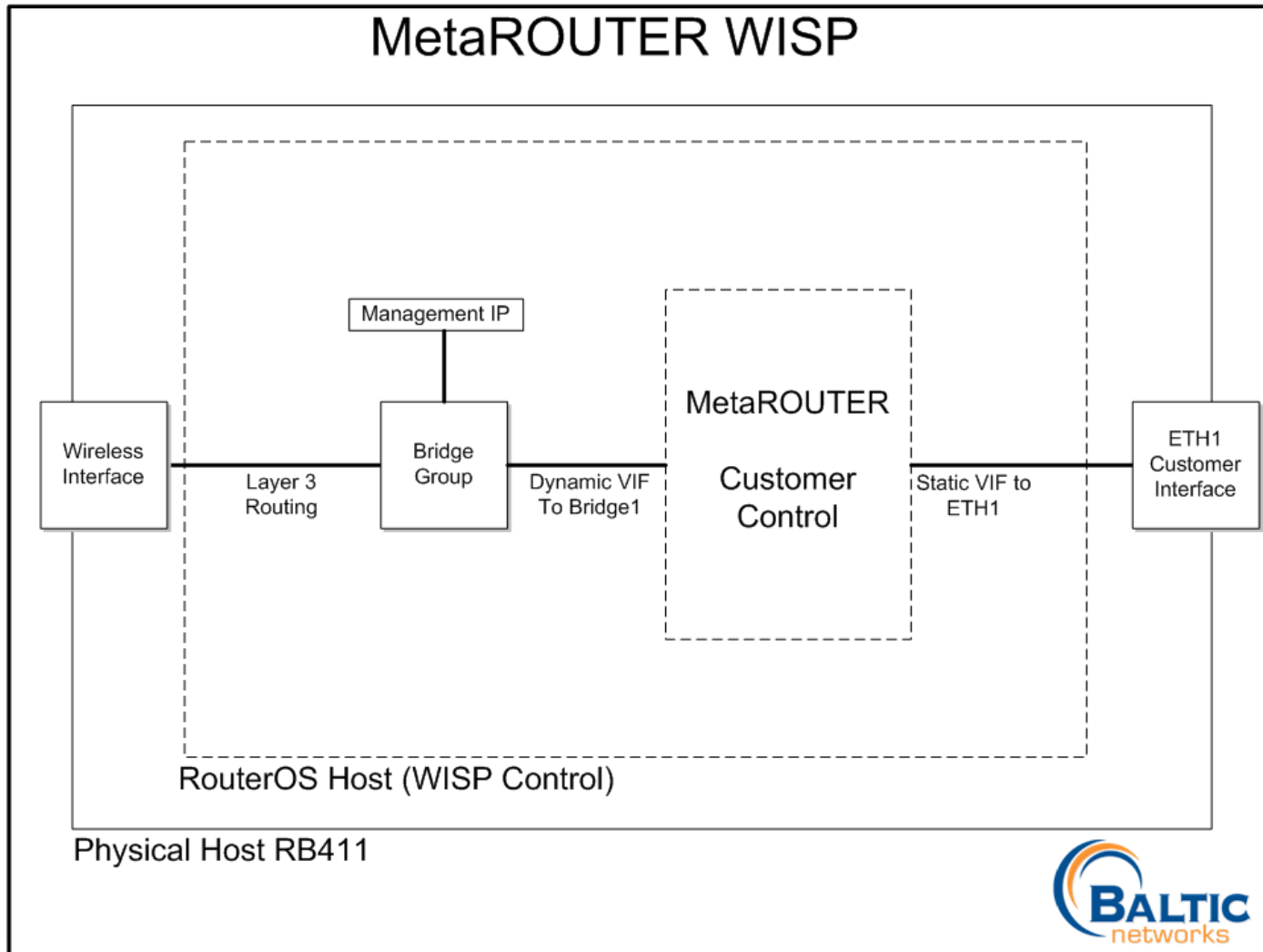
Name	Memory Size (MiB)	Disk Size (kiB)	Used Disk (kiB)	Status
mr1	16		195	running

```
MikroTik 3.27
MikroTik Login: █
```

Multi-Tenant Applications



WISP Applications



Make your own MetaROUTER

- Import Image feature allows you to import a special built OpenWRT kernel running busybox shell.
- Basic buildroot image only 2-3MB in size
- Lots of packages available, including Asterisk, SQUID, Apache, PHP, etc.
- Opens up a whole world for building embedded applications on top of RouterOS

Steps to run OpenWRT

- Copy .tar or .gz OpenWRT image into files
- Import file using “Import Image” in MetaROUTER interface
- Wait for import to complete and go into running state.
- Click on running image and then select “Console”
- Hit return a couple times...

What you should see

The screenshot displays two windows from a network management application. The top window, titled 'MetaROUTERS', shows a table of routers. The bottom window, titled 'MetaROUTER asteriskdemo', shows the output of a terminal session.

MetaROUTERS Table:

Name	Mem...	Disk Siz...	Used D...	Status
asteriskdemo	32		12527	running

MetaROUTER asteriskdemo Terminal Output:

```
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
PPP generic driver version 2.4.2
ip_tables: (C) 2000-2006 Netfilter Core Team
nf_conntrack version 0.5.0 (1024 buckets, 4096 max)
CONFIG_NF_CT_ACCT is deprecated and will be removed soon. Please use
nf_conntrack.acct=1 kernel parameter, acct=1 nf_conntrack module option or
sysctl net.netfilter.nf_conntrack_acct=1 to enable it.

BusyBox v1.13.4 (2009-08-05 01:14:27 CEST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

|_| W I R E L E S S   F R E E D O M
KAMIKAZE (bleeding edge, r16988) -----
* 10 oz Vodka      Shake well with ice and strain
* 10 oz Triple sec mixture into 10 shot glasses.
* 10 oz lime juice Salute!
-----

root@METARouter:/#
```

Getting OpenWRT on the VIF!

- Use the following commands in OpenWRT:

```
uci set network.lan.ipaddr=X.X.X.X
uci set network.lan.netmask=X.X.X.X
uci set network.lan.gateway=X.X.X.X
uci set network.lan.dns=X.X.X.X
uci set system.@system[0].hostname=NAME
uci commit
/etc/init.d/dnsmasq stop
/etc/init.d/dnsmasq disable
reboot
```

OpenWRT Serial Port Access

- Create a loopback bridge group
- Add dynamic interface in MetaROUTER to your OpenWRT image
- Disable console on Host (/system console disable 0)
- Add (/port remote-access) device with IP address on your bridge LAN
- Have OpenWRT application create a TCP socket to your serial port IP address on Host

Online Documentation

- Lots of information on MikroTik Wiki
<http://wiki.mikrotik.com/wiki/Metarouter>
- MikroTik Message Forum
<http://forum.mikrotik.com> (Click on MetaRouter and Xen Forum link)
- OpenWRT (<http://www.openwrt.org>)
- Baltic Networks (www.balticnetworks.com)

Let's do it again, Live this time!

- Going to enable MetaROUTER on a RB450G (680Mhz Atheros, 256MB Ram, 512MB Flash)
- Using only 16MB of RAM for full RouterOS implementation
- Dynamic Interface for Inside Network Access
- Static Interface for Outside WAN Connection