﻿This is the "roadmap" posted in the mailing list, augmented by

comments from the mailing list and the irc chat.

Anybody is welcome to work on any of these issues. Some of

these items are rather simple and can be implemented by single

individuals. Other items are quite complex and development needs

to be coordinated. So, if you want to contribute, please drop

us a note in the mailing list, so you can get help or exchange

ideas.

Christophe Bothamy.

0. Donations

Source Forge recently set up a donation system for hosted projects.

Should we accept donations ? What could we do with the money ?

- give to EFF, FSF or other

- fund Kevin to continue the work on plex86 so we can use it

- bounties for somebody write optimized win9x/NT/XFree/linux/\*BSD

drivers for our vga/net/ide cards

- other ?

Status in Bochs 2.6.5:

No decisions about this yet.

1. Speed

Speed (well lack of) is one of the biggest criticism made by users

who'd like to see Bochs run as fast as Virtual PC.

Paths we can explore to get more speed :

1.1 virtualization : plex86

1.2 dynamic translation : qemu

Status:

Some work has been done for Bochs 2.5 and 2.6 but still long way is ahead.

2 multithreading. Conn Clark wrote :

Threading might be nice too, for those of us who have SMP/SMT machines.

I have a patch from Mathis (who hangs out on the IRC channel all the

time) that puts the video card interface in its own thread. It has

troubles though that I have not resolved. It may also be easier to debug

a threaded peripheral.

I also think that it might be possible to thread a chunk of the CPU

emulation to improve performance on a SMP/SMT machine. Specifically

write\_virtual\_dword, write\_virtual\_word, write\_virtual\_byte, etc...

might just be able to be threaded. I think the threading overhead might

be less than the protection and address translation code. We would have

to try it to find out. I'm also sure there can be some nasty hurdles to

overcome.

Status:

Third party group started a para-Bochs project exactly to reach above goals,

some beta version is already released.

The home page of the project: http://grid.hust.edu.cn/cluster/VirtualMachine/main.html

3. Plugin architecture

3.1 The plugin architecture can be reworked if we want to support

multiple similar devices like serial, net or vga cards.

We currently have two "types" of plugins: "core" and "optional".

Maybe we could add "classes" of plugins. The current version of

Bochs supports the classes "display\_library" and "io\_device".

New classes can be "config\_interface", "net\_lowlevel" and

"sound\_lowlevel"

3.2 Stanislav wrote :

Plugin architecture should be rewritten like real plugin architecture s.t.

Bochs VGA plugin for example will be real plugin. I mean that replacement

of plugin dll in already compiled Bochs will replace Bochs VGA card and

the new card will be detected automatically.

This will allow for example developing of plugins separately from Bochs.

3.3 Michael Brown wrote :

If the configuration interface is to be reworked, could we also make it so

that plugins are self-contained, rather than needing to pollute config.cc

with code for defining and parsing plugin-specific options

Status:

Some of the basic work is done now: The config parameter handling has

been rewritten to a parameter tree and user-defined bochsrc options are now

supported. For most of the optional plugins the config parameter creation and

bochsrc parsing has been moved to the plugin code. Unknown bochsrc options are

now treated as plugin names and Bochs tries to load them.

4. PCI host<->guest proxy

Being able to use a real pci device from inside Bochs would be a

great feature of Bochs. It would ease reverse engineering of non

documented cards, or one could even use a real spare vga card.

Frank Cornellis has done a great job on this subject, and we began

integrating his changes.

Status:

The pcidev device is present in SVN and it has been updated for the new PCI

infrastructure, but the new code is untested yet.

5. VGA

For SVGA emulation we have Bochs VBE and the Cirrus adapter. We should have

a look at the voodoo3 (specs http://v3tv.sourceforge.net/docs.php).

Status:

Voodoo1 emulation has been added in Bochs 2.6.1, initial Voodoo2 support is

present in Bochs 2.6.5, but the performance is still low.

6. Random thoughts on disk emulation improvements :

6.1 support more disk image types

6.2 compressed disk image support

Status:

VPC disk image support has been added in Bochs 2.6. The bximage utility has been

rewritten in C++ for image creation, conversion and resize in Bochs 2.6.5.

7. net

7.1 bootable ethernet rom ?

7.2 user mode networking ?

see etherboot, Micheal Brown wrote :

This already works; you can build an Etherboot rom image with the pnic

driver, specify it as an option ROM in bochsrc and it will boot. I'm

using this extensively at the moment in Etherboot development.

In the Etherboot project's CVS, in the contrib/bochs directory, you can

find a working bochsrc file and an up-to-date README with step-by-step

instructions on getting this working.

Status:

The pnic device is present in SVN, but the status is unknown. PCI boot ROM support

has been added for Bochs 2.6. Built-in 'slirp' support for user mode networking

has been added in Bochs 2.6.5.

8. Bios

8.1 add "jump table placeholder" and log missing function calls in the bios.

Check completness with Ralf Brown interrupt list.

Status:

Not done yet.

8.2 use Coreboot or SeaBios as possible alternatives/extensions to

Bochs Bios ROM we have.

Status:

Starting from Bochs 2.5 SeaBIOS is usable.

9. LGPL VGABios

9.1 Video parameters table

There is a very nice parameter table in 3dfx banshee document

http://www2.lm-sensors.nu/~lm78/pdfs/Banshee\_2d\_spec.PDF

see also http://www.xyzzy.claranet.de/dos/vgacrt.c

Status:

Version 0.7a of the LGPL'd VGABIOS has minimal support for the video

parameter table.

10. Optimized Guest drivers still needed : VGA, IDE, NET

We have a specific VGA driver for winNT/2K, but still

lack drivers for other OSes.

Status:

Not done yet.

11. USB support

Ben Lunt has been working on USB support. The USB mouse and keypad code

is present in Bochs and almost stable. USB flash disk support has been

started and the runtime device change support should be completed.

Status:

OHCI and UHCI host controller and 7 devices are known to work in Bochs.

USB xHCI support is present, but needs more testing.

12. Config file

Benjamen R. Meyer wrote :

I think we should rework the .bochsrc file to be more standard across all

devices. I like how the USB configuration is done in it, and think we should

put something similar together for everything else. In other words, create

something that can be easily used for everything, and make it easier to

configure in the process.

From what I can tell right now, most of the configuration lines are randomly

thrown together as each gets implemented or added, instead of having

something that is based on a standard approach to the configuration.

The result should be something that would be able to easily auto-configured

by another program (a configuration editor?) with minimal changes necessary

when new devices/features are added.

Status: Some work to unify parsing and saving config options has been done

in Bochs 2.6.1.

13. lowlevel serial support for Windows.

Volker has been working on this.

Status:

Not yet complete (transmit works, receive is losing data).

14. Parallel port

Conn Clark wrote :

I would like to see better parallel port support so I can use a dongle.

This is something I would find very useful as it would mean I wouldn't

have to boot back into windows ever again. I also recognize that this

may require a kernel module be written, which is beyond my current

skills. I know others will find this useful as I have had to tell a

few people that their parallel port driven peripherals that require a

bidirectional parallel port won't work.

Status:

Not done yet.

15. Guest-To-Host Communication

Try to adapt VirtualBox guest-to-host communication methods into Bochs.

Having VirtualBox Shared Folders or VNAT support in Bochs could very

simplify its usage.

16. Patches / Bug reports

There are dozens of patches floating around. Some are outdated,

don't apply cleanly, are obsolete/unneeded. We could try to do

some clean-up, and keep only relevant ones.

We should also clean up the SF bug tracker. Some bugreports are

very old and we asked for more information with no response.

Status:

There is some progress, but still a lot of work to do.

17. Positions

If you want to help without coding, here are available positions :

19.1 Webmaster : update website (Jan Bruun Andersen offered to help)

19.2 patch coordinator : look at incoming patches (sourceforge and

mailing list) and upload / update in the SVN patches directory.

19.3 platform maintainers for macos / win32

19.4 disk image maintainer : create and maintain our collection

of disk images. Usually, only the configuration file needs to be

updated, and old bios files have to be removed. Some packages

still contain very old bios files, they should definitely have

to be removed.

Status:

More active developers are needed to do the things described above.

18. Bochs demo cd/dvd

With version 2.1, it is now technically possible to use disk images

on a read-only media, with a journal files on a read/write media.

It would be great to create a demo cd/dvd with executables for

supported platforms, configuration files and read-only disk

images, the journal files would be written in a temporary

directory on the harddisk.

Status:

Not done yet.

19. Other CPU architectures : arm, ppc

This has been asked in the mailing list. I'm not really

interested, but other people might be. Should we propose to

host the new CPUs code in our source tree, or should we let

people fork ?

Status:

Not done yet.