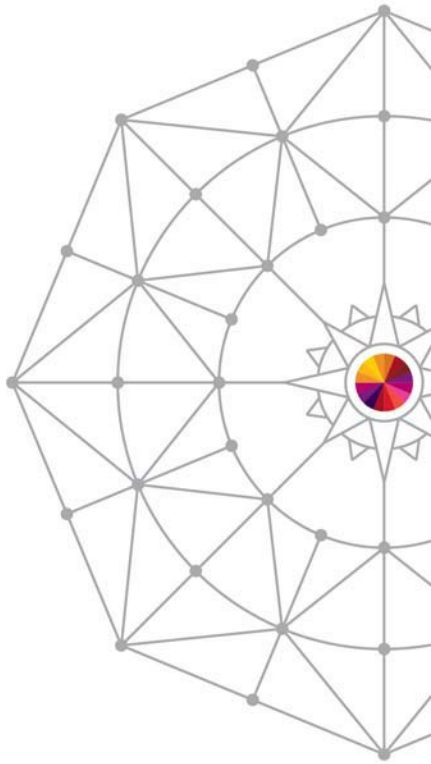
Linux bootloaders on System z

Ihno Krumreich
SUSE

August 7th, 2014
Session 15694



Insert Custom Session QR if Desired.

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides education, professional networking and industry influence.

Agenda

- Functions of a Bootloder
- Types of bootloaders
- zIPL
- Grub 2
- Grub 2 s390x implementation

Functions of a Bootloader

Functions of a bootloader

- Load kernel / initrd to memory and start kernel
- Offer one or more systems to choose from
- Offer the possibility to set / delete parameters to the kernel

Types of Bootloaders

How to load Kernel and initrd

Two possibilities:

- Type 1: Bootloader knows the blocks on the disk where kernel / initrd are located.
  At boot time these blocks are copied to the memory and the kernel is started.
- Type 2: Bootloader can read and understand the filesystem where kernel / initrd are located.
  At boot time the bootloader reads the filesystem and loads the files specified in the config.

Todays represantatives of the two types

- Type 1
  - zIPL
  - lilo (Linux loader)
- Type 2
  - grub
  - grub 2

IPL Process for Linux on System Z

Linux Initial Program Load (zIPL)

- Linux Bootloader for System Z
- Configured with /etc/zipl.conf
- Bootloader written using /sbin/zipl

zIPL Configuration file (/etc/zipl.conf)

```
[defaultboot]
defaultmenu = menu

[SLES11_SP3]
    image = /boot/image-3.0.76-0.11-default
    target = /boot/zipl
    ramdisk = /boot/initrd-3.0.76-0.11-default,0x2000000
    parameters = "root=/dev/dasda2 hvc_iucv=8 TERM=dumb
resume=/dev/dasda1"

:menu
    default = 1
    prompt = 1
    target = /boot/zipl
    timeout = 10
    1 = SLES11_SP3
```

zIPL Operation

- Multi-stage operation
  - Stage 0 – 24 bytes – 2 CCWs to load and TIC to Stage1
  - Stage 1 – 104 byte channel program to load Stage 1b
  - Stage 1b – 1k channel program to load Stage 2
  - Stage 2 – 8k maximum menu and kernel loader
  - Stage 3 – Kernel parameter parser and execution

zIPL Limitations - Devices

- Limited device capabilities
  - Restricted to specific Channel I/O device types
    - DASD
    - SCSI
    - Tape
  - Boot from the Network? Nope.
  - Device-mapper tends to be finicky

zIPL Limitations – File Systems

- Limited file system support
  - Only ext2, ext3, ext4, xfs
  - Btrfs not supported

Grub 2

GRUB (GRand Unified Bootloader)

- Loads itself
  - This also requires multiple steps. (similar way zIPL does it)
- Reads /etc/grub2/grub.cfg
  - This requires that all supported filesystems are compiled in
- Offfers a selection menu to the user
- Boots the selected kernel (either via timeout or user selection)

Grub 2 on System Z

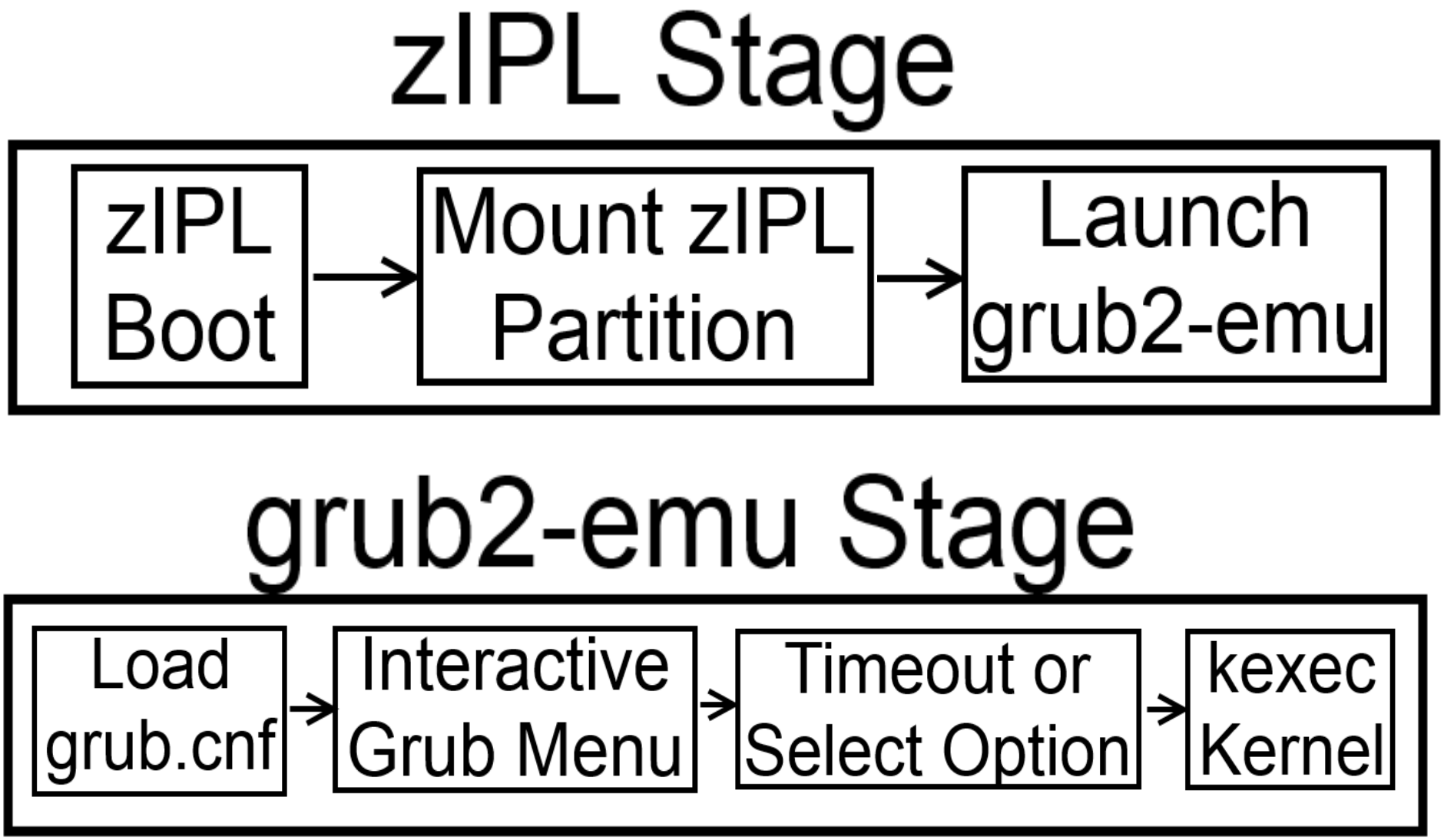
Challenges of Grub 2 on System Z

- Large Binary Size
  - 1.3MB cannot be easily loaded via CCW
- Unique System Z Drivers
  - DASD, zFCP, QETH all should be cleanly implemented
  - Maintenance a major factor
- Unique Display Configuration
  - Both 3270 and 3215 need to be supported

Solution : grub2-emu

- User-mode grub2
  - Relatively little modification necessary
- Requires a linux kernel to be booted
  - All supported s390 drivers available
  - udev resolves devices
  - Still uses zIPL, but in a “Stage 4” role
- Full filesystem support
  - BTRFS snapshots
  - Unusual device-mapper configurations
- Two stage boot

Grub 2 boot process



How to boot another kernel?

- kexec
  - Kernel-supported reboot
  - Safe shutdown and restart of devices
- Implementation of kexec in Grub 2
    - New feature for grub2-emu
  - Options for kernel and initrd are used and verified
  - Only available to root user

Configuration : grub2

- Special grub2 boot partition
  - Normally mounted to /boot/zipl
- Contains grub2 kernel and initrd
- Contains grub2 config file
  - Allows for dynamic configuration
  - No more need to run zipl when kernel configuration changes

Pros and Cons

zIPL

- Pro
  - Small bootloader
  - No hazle / maintenance with filesystemcode
    (but filesystem is not allowed to relocate datablocks)
- Con
  - Every time the kernel / initrd is updated the bootloader must be rewritten
  - Every time the configuration is changed the bootloader must be rewritten
  - What was not configured at install time is not doable

Grub2

- Pro
  - Change of the configuration is just the editing of a file
  - Update of kernel / initrd is just an update of the file.
- Con
  - Big bootloader
  - The read part of the filesystemcode must be included and maintained

Why we did it

There where two main reasons:

- The snapshot technology of btrfs could only be easy used by a bootloader that read files
- From an administrators point of view
  - We have grub on all architectures where SLES runs on
  - This simplyfies the knowledge the administrator and managing tools need to have.
  - Simplyfies tools that automate system administration

Evaluate

SHARE
Educate · Network · Influence

SHARE
in Pittsburgh 2014

Corporate Headquarters
Maxfeldstrasse 5
90409 Nuremberg
Germany

+49 911 740 53 0 (Worldwide)
www.suse.com

Join us on:
www.opensuse.org