

# **Mz - "Managing z"**

**A systems management tool for z/VM and Linux**

Michael Maclsaac

VM Workshop

Indianapolis, IN

June 21, 2013

# Agenda

- **Introductions**
- **One question**
- **Why? What? Who? and How?**
- **Why Open source?**
- **Function provided**
- **Command line vs. Web interfaces**
- **Documentation**
- **One more question**
- **Demonstration**

# Introductions

- Who am I?
  - ▶ Michael MacIsaac
  - ▶ 26 years at IBM
  - ▶ Lots of different jobs
- Who are you?
  - ▶ Who has tried Mz?
  - ▶ To do systems management of z/VM and Linux
    - All *roll your own*?
    - Single encompassing tool?
    - Some tools, some *roll your own*?

# One question

- Q. Is there a lightweight, free, open source, powerful, fast, intuitive, solid, well-tested systems management tool for z/VM and Linux?

# One question

- Q. Is there a lightweight, free, open source, powerful, fast, intuitive, solid, well-tested systems management tool for z/VM and Linux?
- A. **Absolutely not!**

# One question

- Lightweight?
- Free?
- Open source?
- Powerful?
- Fast?
- Intuitive?
- Solid?
- Well-tested?

# Agenda

- ~~Introductions~~
- ~~One question~~
- Why? What? Who? and How?
- Why Open source?
- Function provided
- Command line vs. Web interfaces
- Documentation
- One more question
- Demonstration

# Why? What? Who? and How?

- To solve the business problem of *virtual server sprawl*
- To build the *foundation* before the *storefront*
  - ▶ 2 interns, summer of '08
- To prototype real-world requirements:
  - ▶ "No root login"
  - ▶ RPM history/reporting
  - ▶ OVF reference implementation
  - ▶ Device conflicts/reporting
  - ▶ Start and stop Linux (not `poweron`, `poweroff`)
  - ▶ z/VM health screen
- To help you the customer solve your IT needs and be successful

# Why? **What?** Who? and How?

- What is Mz?
  - ▶ A systems management tool on z that is:
    - agentless, daemonless, databaseless, stateless
  - ▶ A tool with commands of the form **mz<verb><object>**
    - Linux verbs: **mk** (make), **ls** (list), **rm** (remove), etc
    - Objects: **server**, **client**, **tree**, **appliance**, **monitordata**, etc.
  - ▶ A tool with the Linux file system as its database
  - ▶ A tool that crosses CECs and LPARs
  - ▶ Allow pings, copies and commands to all Linuxes in parallel
  - ▶ Command-line-centric, with a growing Web interface
  - ▶ Able to support 1st, 2nd and 3rd level Linux systems
  - ▶ A "poor man's" backup and monitoring tool

# What? (cont'd)

## CEC 1

LPAR 1 - z/VM	LPAR 2 - z/VM
Virtual Machine 1 Linux	Virtual Machine 4 Linux
Virtual Machine 2 Linux	Virtual Machine 5 Linux
Virtual Machine 3 Linux	Virtual Machine 6 Linux

## CEC 2

LPAR 3 - z/VM	LPAR 4 - z/VM
Virtual Machine 7 Linux	Virtual Machine 9 Linux
Virtual Machine 8 Linux	Virtual Machine 10 Linux

# What? (cont'd)

## CEC 1

LPAR 1 - z/VM	LPAR 2 - z/VM
<b>Virtual Machine 1 Administrative Linux</b>	<b>Virtual Machine 1 Administrative Linux</b>
Virtual Machine 2 Linux	Virtual Machine 5 Linux
Virtual Machine 3 Linux	Virtual Machine 6 Linux

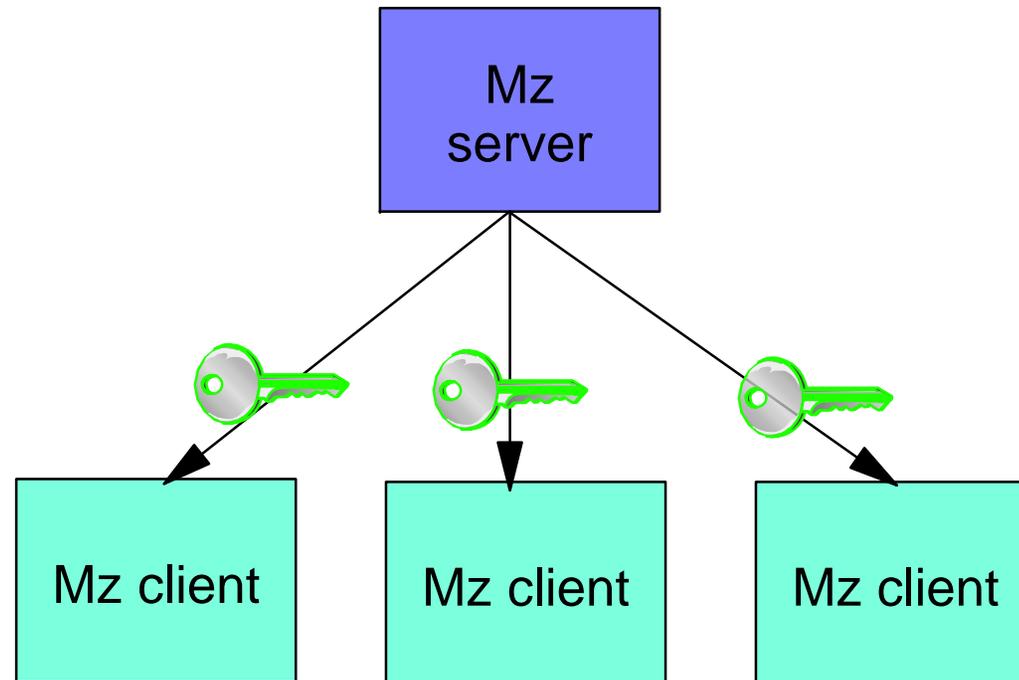
## CEC 2

LPAR 3 - z/VM	LPAR 4 - z/VM
<b>Virtual Machine 1 Administrative Linux</b>	<b>Virtual Machine 9 Administrative Linux</b>
Virtual Machine 8 Linux	Virtual Machine 10 Linux

# What? (cont'd)

An Mz "server"

CEC 1 LPAR 1 (z/VM)

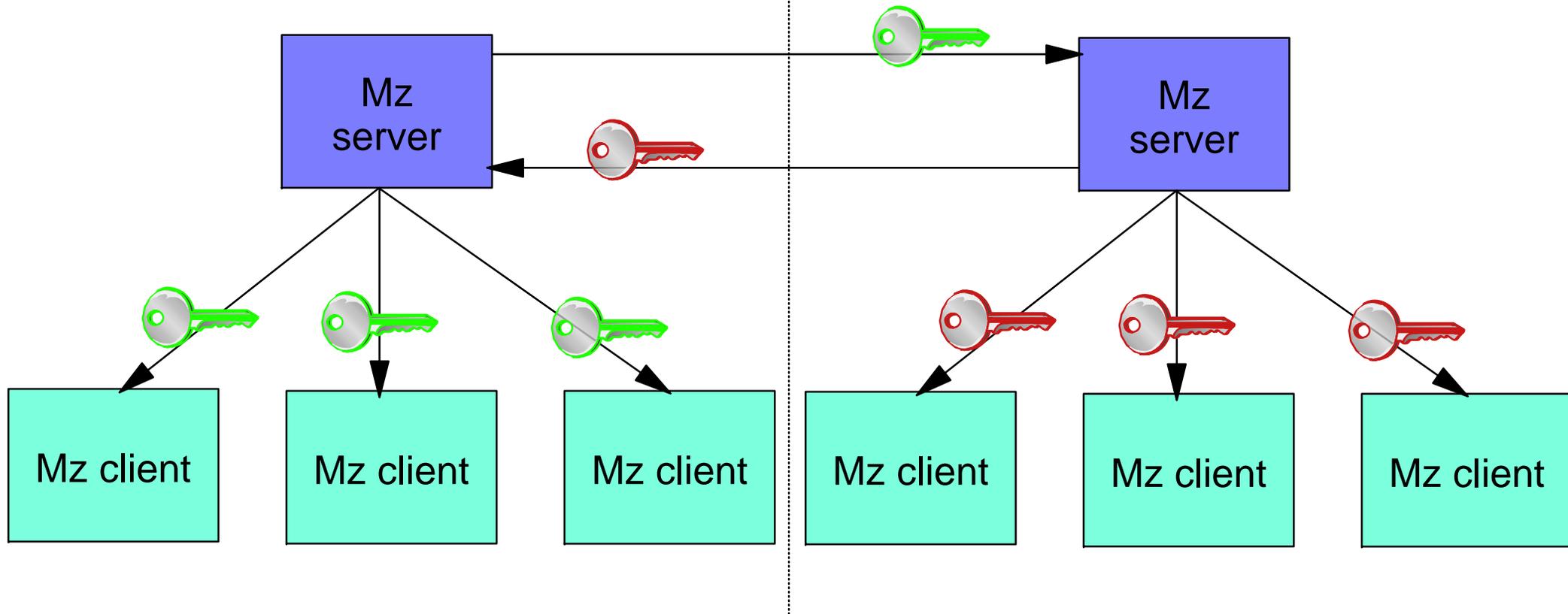


# What? (cont'd)

An Mz "cluster"

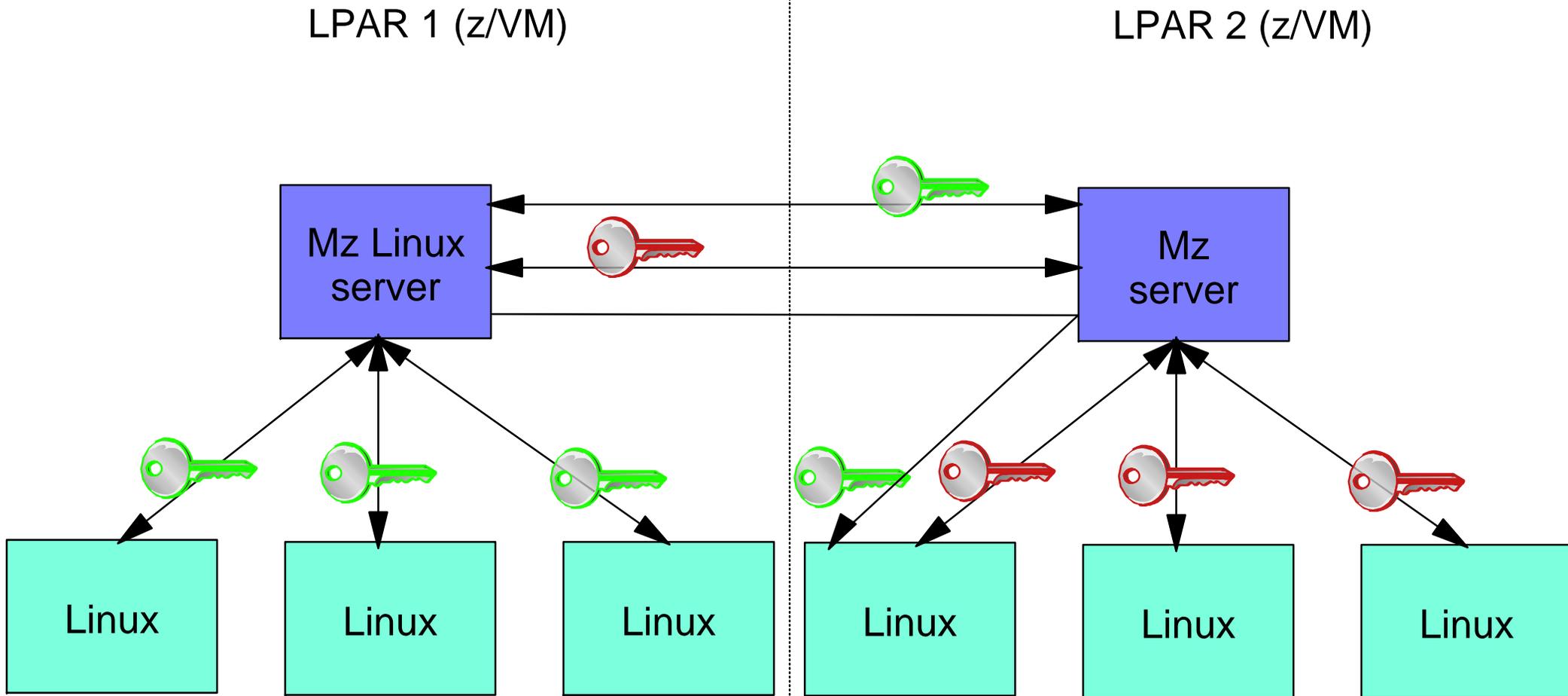
CEC 1 LPAR 1 (z/VM)

CEC 2, LPAR 1 (z/VM)



# What? (cont'd)

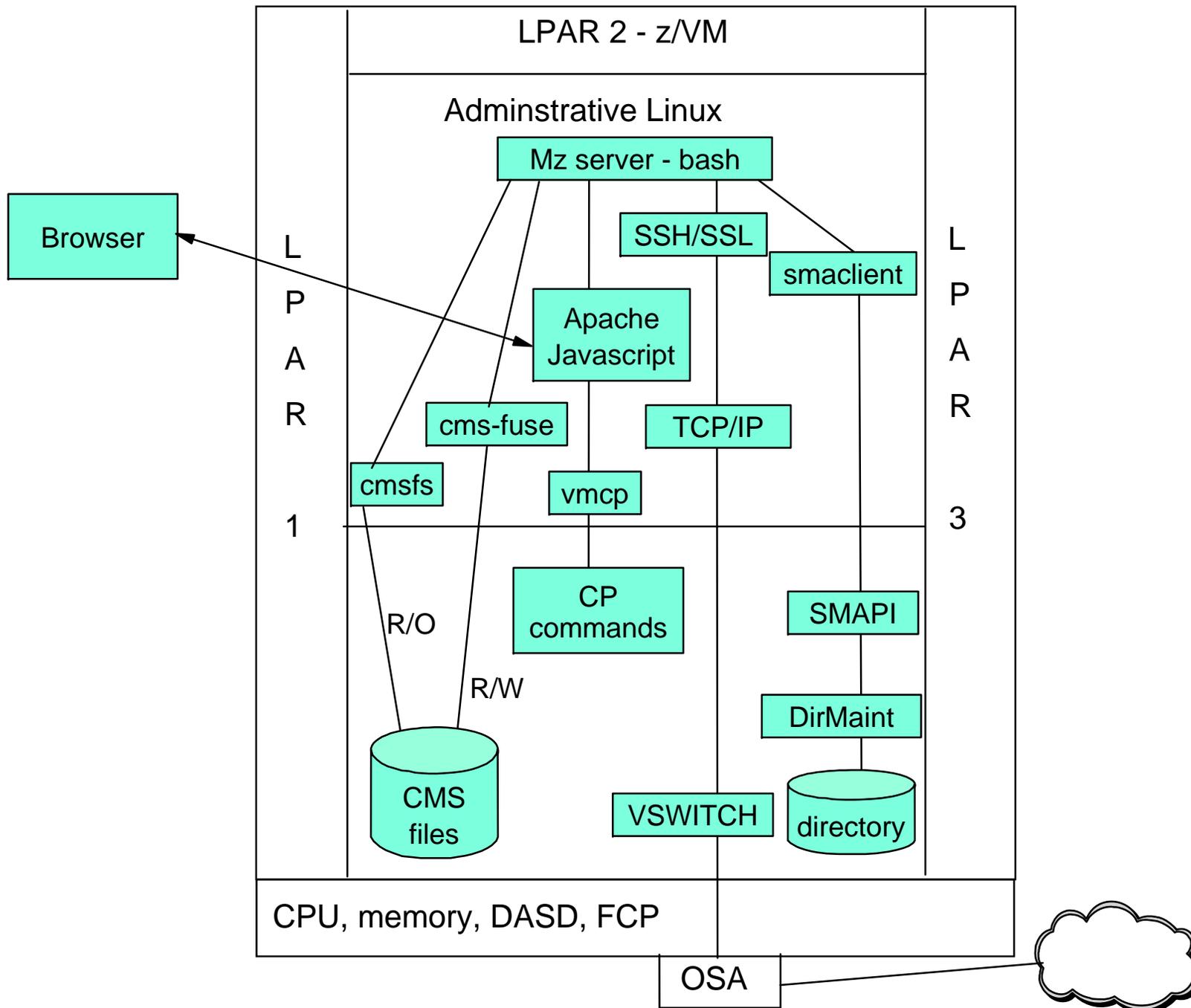
## Cross-LPAR/CEC key exchange



# Why? What? **Who?** and How?

- Coders
  - ▶ Myself
  - ▶ Marian Gasparovic
  - ▶ Two others (mentioned in the PDF)
- Collaborators
  - ▶ Carlos Ordonez
  - ▶ Alan Altmark
- Supporters
  - ▶ many (mentioned in the PDF)

# Why? What? Who? and How?



# Agenda

- ~~Introductions~~
- ~~One question~~
- ~~Why? What? Who? and How?~~
- Why Open source?
- Function provided
- Command line vs. Web interfaces
- Documentation
- One more question
- Demonstration

# Why open source?

- It's the best software development model
  - ▶ Linux is the only cross-IBM-platform operating system (QED)
- To be able to "Release early, release often"
- To enable community contribution
- To be sure it's the best model:
  - ▶ Document reasonably well
  - ▶ Don't put out crap code
  - ▶ Don't abandon and leave dead projects

# Function provided

- Captures z/VM and Linux hierarchy cross-enterprise
- Command line interface
  - ▶ Many **mz-verb-object** commands
- Web interface
  - ▶ **mzdevices**: show system devices in a table
  - ▶ **mztable**: show Linux, z/VM systems in a table
  - ▶ **mztree**: show a hierarchy of the tree
  - ▶ **mzhelp**: show a help screen
- **Description** and **owner** fields for all z/VM & Linuxes
- Capture and deploy with OVF
- *No-root* SSH support
- Monitoring
- Shared devices

# Agenda

- ~~Introductions~~
- ~~One question~~
- ~~Why? What? Who? and How?~~
- ~~Why Open source?~~
- ~~Function provided~~
- **Command line vs. Web interfaces**
- **Documentation**
- **One more question**
- **Demonstration**

# Command line vs. Web interfaces

- CLI is
  - ▶ function-centric for the sysadmin
- Web interface is
  - ▶ R/O except **Description** and **Owner** fields
  - ▶ Richer in drill down capabilities?

# Documentation

- One manual as a PDF (~60 pages)
- Help flags
- CLI help command
- Web help page
- No man pages (yet)

# One more question

- **Q.** Is or will Mz be cross-platform?
- **A.** No, possibly
- **Potter's rule of systems management:**
  - ▶ "The temptation in systems management is to try to abstract function and code across platforms. Resist that temptation - it is better to drill down into a platform-specifics sooner rather than later."
- **However, `/var/lib/mz/systems/` could be**
  - ▶ `/var/lib/mz/systemz/`
  - ▶ `/var/lib/mz/systemp/`
  - ▶ `/var/lib/mz/systemx/`
- **mzlstree** could also be **mplstree** and **mxlstree**
  - ▶ (some day, but I'm not coding it :))

# Demonstration

- Network dependent...

# System z hierarchy (cont'd)

