# 人人时代，无组织的组织力量

## 所涉及到的网站

Digg的首页类似报纸，有及时重要的新闻组成，根据用户的投票来辨别一个新闻的及时和重要程度

Flicker可以在上面进行照片分享，基本功能改变了群体性的旧法则，从“先集中后分享”到“先分享后集中”。

Myspace观众互动，

Boing Boing 网络文化博客

Meetup 社区，提供线下聚会。

IRC和处于同一频道的人交谈

## 所涉及到的软件

Wireshare 协议分析软件，监视网络通信，抓取数据包

Nmap 网络扫描软件，可以扫面任意ip范围测试ip地址是否有响应，也可以查看ip所开放的端口

Nagios 网络监控软件，设备短线时可以通知你。可以管理snmp协议。

SNMP 简单网络管理协议 可以用来查看网络上所有的设备，收集信息。开启snmp服务snmp-server community public ro，snmp-server contact yourname 设置联系方式，snmp-server location place 设置地点。Exit write memory。

Syslogd可以让设备向中央服务器报告自己的错误信息。命令如下：snmp-server community public ro 为日志增加时间，no loging console 停止向控制台输送日志，， no loging monitor 停止向监视器输送日志， loging 192.168.100.1 把syslogd服务的日志输送到指定的ip，loging waring 设置发送报警信息，exit write memory。

Snort可以检测异常的arp请求。Arpspoof可以制造arp数据包。

SAPS server agent plugin synchronization配置内存大小的工具

## 无组织的力量

各种各样的地方性变化全部来源一个深刻的根源：具备新能力的群体在形成，他们的工作无需遵循规则，客服了期限至有效性的传统桎梏，这就是无组织的social network（include social software, social media, social computing）

## 分享，创造新群体的基点

群体中人是复杂的，复杂的足以使群体难于形成和难于维持。传统机构之所以呈现我们看到的样子，大多是为了应对某种困难。新的社会化工具拜托其中的一些困难，似的新组织的形成成为可能，比如把简单的分享作为创造新群体的基点。

新的时代需要放宽对用户的监管，改为提供工具，让潜在的用户自我同步与协调。

合作是在往上一个阶梯，合作比单纯的共享要难，因为它涉及到改变个人的行为与他人的同步，而他人也在改变行为与你同步。

对于一个群体来说，群体越大。统一一个决定就会越困难，要采取集体行动，就必须要有足够大的愿景将大家绑在一起，尽管偶尔的决定不免使一些成员难以满意。

在群体与他人共享、合作、协调一致、是人的本能，但会受到交易成本的限制。

## 人人都是自媒体

大规模的业余化打破职业类别的界限。

## 先出版后过滤崭新的游戏规则

网络使得技术上的交互性成为可能，但是技术赐予的又被社会因素拿走了。在这种情况下名人的任何互动潜质都被压制，名人的两个条件：一是规模，必须受到数以千计或者更多的关注；二是不能及时反馈，不能针对每一个关注者予以反馈。

出版的大规模业余化使得过滤的大规模业余化势在必行。每个网页都是潜在的社区，他们受到感兴趣者的持续关注，而那些人也可能有兴趣彼此交谈，

## 当个性化动机遇上协同化生产

协同生产，即人们必须协同合作才能完成某件事情，分布式协同（distribute collaboration）最著名的案例是维基百科。在维基上读者和作者的角色可以随时穿越，一篇维基百科文章是一个过程而不是一个产品，因为他永远没有完成的一天。在大兴社会系统中会出现不均衡并且贡献者会呈现幂率分布—等级越高则越不均衡。

在互联网中人们为何想贡献？一是锻炼一下子，二是虚荣心，存在感，三是想做一件好事，所以就有了分享。

## 集体行动催生变革

集体行动即一群人整体行动，比协同生产更为复杂，新的社会也催生出多种活动。分享变的很容易。

在适当的情势下人们愿意甚至渴望集合起来改变世界，在这样的群体里，动机、活力和行动的才能都已具备了。社会化的工具并没有创造集体化的活动，他们只是消除了对于集体行动的阻碍。

## 从更快的速度到更快的行动

随着越来越多的人采用了简单的社会化工具，随着这类的工具带来更快的沟通，群体的速度也在加快。多即不同，快也不同。快闪等新名词出现。

## 社会性困境的解决之道

任何时候，无论是与我们可以利用的还是能够利用我们的人交往，我们都同样面临着囚徒困境，不过我们常常却能相互信任并共同完成一些事情。

由于未来的影子，我可能在今天无从你的利益出发做事，即使让我承担一定的风险或者代价，但我所期望你会记得这一切并在明天给我回报。

社会资本：即运转良好的社区所具有的一系列而神秘而又关键的特征，简单的说就邻里互助，乡里互信。

城市的存在是因为人们喜欢互相邻近，不仅仅是简单的信息交换，而正是这个事实才产生社会资本。

在一个社会资本不断衰减的时代，如果有人能把参与社团变得更简单些，人就会采取行动参与它。

即使在一个媒体充斥的时代，人们仍然渴望真人接触。

社会生活的高度可见性和可搜索性意味着，想法相同的人们现在拥有了找到对方、聚集起来并相互合作的能力，而不受社会赞成和反对。

## 亲爱的小世界

大型社会网络变得紧密，更多的在于增加联系最广的那些人所能负担的联接数量，而不太会增加其普通成员的所能负担的联接数量。

## 开源的运动，没有代价的失败

连续性的系统倾向是人们对次等品产生了包容，

# 深入浅出networking

## 名词

NIC（Network Interface Card）：网络接口卡

ICMP 因特网报文控制协议 ping命令发送的包

Trancetoute 命令可以查看从发送到接收经过的包。

RIP （Routing Information Protocol）动态路由协议，让路由器自动更新路由表。

Dig 域名 +short 可以得到域名所对应的ip地址 +short表示过滤掉细节

Dig Ip地址 +short 可以得到ip所对应的的域名

DHCP （Dynamic Host Configuration Protocol）动态主机配置协议

NAT（Network Address Transaction） 网络地址解析转换

## 维修物理网路

CAT-5平常通用的非屏蔽双绞线，其中绿色电线负责接收数据，橙色电线负责发送数据。

纯色电线代表负极，花色电线代表正极。

光纤，单模光纤成本高，性能高可到14TB/S，多模光纤成本低，性能为10GB/S。

## 规划网络布局

## 查找故障的工具和办法

LAN网络分析仪可以检查、验证电缆，还能检测网路通信，协助定位网络问题，甚至还能处理无线网络问题，可以用来查找和定位光缆。

## 分析数据包

常用的编码方式有三种：不归零编码，不归零反向编码，曼彻斯特编码。曼彻斯特编码比不归零编码多了内置时钟信号，适合网络数据传输。

网卡收到待传送的消息后，先将消息转换成二进制数，即一系列的1和0，然后按照一定的编码规则对二进制数进行编码；最后将编码信号通过网络发送出去。网卡实现将机器cpu的不归零编码转换成曼彻斯特编码。

## 网络设备与网络通信

Tcp和udp，tcp传输数据较慢，开销比较大，但是不会丢包，而udp则传输速度较快，开销小，容易丢包。ICMP用来测试网络连接协议。交换机可以理解数据帧的内容，解读MAC地址。他是先把收到的数据帧存储下来解读MAC后根据MAC发送到目的端口。

登陆进入交换机可以使用show mac-address命令来查看3分钟以内的mac地址表。

交换机只能根据数据帧的目的MAC地址转发数据帧，不能跨ip子网工作。

路由器可以跨ip子网工作。

交换机主要负责连接计算机，而路由器主要负责连接子网。

## 网络互连

交换路由器可以实现路由器和交换机的共同功能，交换路由器每个端口都含有一个专用处理器。

地址解析协议ARP 实现ip地址到mac地址间的转换。

## 路由协议

路由信息中C表示与端口直接相连的网段地址，D表示路由信息的来源，S表示路由手工录入的静态路由。

RIP协议会自动选择跳数最少的路由优先传输。RIP只适合小型网络动态路由，大型网络混合路由器一般采用OSPF，思科的单一路由种类选择EIGRP。

## 域名系统

域名[www.trustie.net](http://www.trustie.net) www是主机名，域名是trustie.net，顶级域名是.net

## 监控与检查故障

### 排查基本步骤

请用户演示问题所在，

检查物理设施

询问周围用户，区别个例还是所有

检查路由器交换机的工作状态

尝试注意ping检测各个网络

### 命令

多数交换机和路由器都提供show interface 命令，能比ping获取更多的端口网络信息。，路由器和交换机都支持ssh登陆的。

## 无线网络

无线网络把固定的ip分给用户，NAT可以为无线网提供更多的ip地址。

无线协议有时候需要设置兼容。

端口转发可以把需要的信息转发到指定端口。

## 网络安全

防止arp病毒攻击增强交换机的防御能力。

防止网络攻击仅仅靠交换机和路由器还远远不够，需要借助firewall。

防火墙有两种规则：静态包过滤和状态包过滤。静态包过滤依赖数据包的包头信息。

## 设计网络

流程

收集客户需求设计网络布局图列出硬件清单画逻辑结构网络图规划安装步骤安装

# VmvareVsphere 5虚拟数据中心构造指南

## 名词

DRP（Disaster Recovery Plan）灾难恢复计划

SaaS（Software as a Service）软件即服务，用户对底层基础架构没有控制权。用户可以直接使用在云中找到的运用程序。如：gmail，google app。

PaaS（Platform as a Service）平台即服务，用户使用服务商提供的语言，部署自己的程序。如 SpringSource、Microsoft Azure

IaaS（Infrastructure as a Service）基础架构即服务，用户自己部署处理器、内存、网络和存储资源。如EMC、VMware、Cicso、IBM、Amazon Web Services。

TCO（total cost of Ownership）总拥有成本

## 从服务虚拟化到云计算

服务器虚拟化是掩盖赢家设备物力资源的一个抽象层，位系统提供与实际形式不同的资源。

虚拟化必须要合理的管理，要建立VM的生命周期，恰当的管理资源分配，减少虚拟化蔓延的现象。

## Vsphere5 演变架构

Vcenter server：提供ESXi服务器的集中化管理。

ESXi虚拟化物理机。

VSphere Client：创建、管理、监控VM及ESCXI主机

VSphere web Access 提供web界面对虚拟机进行管理。

## 服务器和网路

## 高可用性和灾难性恢复

Vsphere HA高可用性；vsphere FT高容错性

# 鸟哥私房菜

## 安装系统

Ext3文件系统多了日志的记录，对于系统的恢复比较快速，

所有的文件系统分区需要挂载在/下面。Swap空间最好在物理内存的1.5-2倍之间。

SElinux（Security Enhanced Linux）是一个linux系统访问控制的细化设置。、

PKHunter 可以rootkit进行扫描，rootkit本身是一个木马或者蠕虫，会让主机一直向外发送数据包，结果会让你的带宽被吃掉，常见的有Nimda，Code Red

## 首次登陆和在线求助

$：/etc/inittab这里面可以设置登录时选用桌面或者客户端，

$：LANG=en\_US修改语言 打开计算器：bc q可以结束一些特殊的模式

**$：netstat –a | grep \*查看网络联机状态**

**$：ps –aux 可以查看后台运行的程序**

$：sync将数据同步写入硬盘

$：shutdown –h now/+30 立马/30分钟后关机 -r重启

## Linux的文件权限和目录配置

FHS（Filesysterm Hierarchy Standard）。根据这个标准指出

/usr 放置软件

/etc 下放置配置文件

/opt 放置第三方软件

/boot 开机与内核文件

/var/mail 用户信箱

/var/run 程序相关

/var/spool/news 新闻组

/var/lock 程序相关

**$：uname –r 查看实际内核版本**

## Linux文件与目录管理

$：touch ./--test 可以创建特殊符号的文件

$：basename /etc/sysconfig/network 获得network这个最后的文件名

$：dirname /etc/sysconfig/network 获得/etc/sysyconfig这个目录名

$：cat –n filename 打印出行号

$：cat –b filename 去掉空白行号 然后打印输出

$：more/less filename 弹性查看文本信息

$：head/tail –n 行数 filename 显示头/尾多少行 默认10行

**$：od –t type(a默认字符输出；c利用ascii字符数出；d利用十进制输出；f利用浮点数输出；o利用八进制输出；x利用十六进制输出)解决二进制乱码.**

$：od –t oCc filename 以八进制列出存储值与ASCII的对照表

$：touch –a filename修改文件的访问时间 文件time

$：ls –l –h用M做单位显示目录和文件信息

**$：chattr +i fielname 给文件加上i属性 该文件不能被任何人更改和删除**

**Chattr +/- 添加/减去隐藏属性 i决定文件的修改 a保持文件的访问时间不被修改，C这个属性设置后将会自动将此文件进行压缩读取的时候会自动的解压缩，存储的时候会先压缩后存储。**

**$：lsattr查看隐藏属性**

**$ssh-keygen –b 1024**

## 常用的网络命令

$: route –n //查看路由状态

$: route add/del //增加/删除路由

$: ip link //查看与设备相关的关系

$: ip link show //显示本机所有端口

$: ip link set eth0 address/down //重设网卡地址、关闭物理设备、

**$: ping 两台主机间通信**

**$: traceroute 两主机间各节点分析**

**$: netstat –an //查看所有网络连接状态**

**$: netstat –tulnp //显示出目前已启动的网络服务**

**$: netstat –atunp //查看本机所有网络连接状态**

**$: telnet 远程连接，查看端口是否正常启动**

**$: links http ://www.baidu.com //查看网页**

**$ links –dump http://www.baidu.com>baidu.html //把首页内容抓取到本地**

**$ tcpdump 监听软件**

**$ tcpdump - i eth0 –nn port 22 取出22的数据连接包**

**$ netstat –tunl //查看正在监听的网络服务**

$ netstat –tun //列出已连接的网络连接状态

$ nmap –sTU localhost //查看本机的tcp/udp端口

$ nmap –sP 10.107.17.0/24 通过icmp数据监测，分析局域网内有几台机器是主动的

$ nmap

# GERT安全指南

## 相关工具

SATAN（System Administrator Tool for Analyzing Network）：系统管理员分析网络工具，ftp;//ftp.porcupine.org/pub/security/

Tripwire是unix系统上用于检查文件系统完整性的程序

Spar显示进程统计记录

Tcpdump网络监视工具

Snort网络轻型入侵检测系统

# 中国智慧

## 周易的启示

周易最重要的意义在于提供了一种世界观和方法论。

变化是吉利的，有前途的；不变是凶的。长久是因为变通，变通是因为穷极。所以逆境中要有信心，顺境中要居安思危，夹着尾巴做人。

## 中庸的原则

中庸不是老好人；老好人就好像麦田里的稗子，似是而非。就好像癌细胞占着位置不干活。癌细胞多了人就死了。

中庸不是和稀泥，实事求是。

中庸不是没有原则，无原则的交往就是勾结。

实际上中庸不仅是讲原则，而且中庸本身就是原则。

中就是不走极端，中庸就是不唱高调，所谓不走极端就是：既不缺位，也不越位；既不过头，也不掉队。

如果我们觉得是我们应该做同时也能够做的就去做。

中庸一定是常人之道；一定是适中之道；中庸一定是可行之道。

# 博弈论战略分析入门

## 冲突战略与博弈

# 虚拟化技术详解

## 虚拟化与云计算

虚拟化分为：

平台虚拟化，针对服务器操作系统进行虚拟化、

资源虚拟化，针对存储和网络进行虚拟化

应用程序虚拟化，为应用程序提供一个虚拟的运行环境

表示层虚拟化，服务器仅向用户提供表示层

云计算的特点：

基于互联网，通过网络连接服务器资源

按需服务，资源可以动态伸缩

资源池化，统一资源，用户不用关心型号

## Vmvare虚拟化

磁盘制备：

精简制备，精简制备随着写入空间不断变大。

后制备延迟置零，先分配给虚拟不擦除物理媒介上的数据在第一次写磁盘的时候清除磁盘。

后置备置零，创建时就清除以前的数据。

Clone快速复制产生新一模一样到的虚拟机。

虚拟机部署常用手段，克隆技术，模板技术ovf，自动封装技术。

自定义规范会预先设置可以避免重复的进入虚拟机进行设置

Vsphere web client可以提供网页管理虚拟机

## Vsphere数据存储

采用sas或者sata协议连接外部存储。

添加外置的nfs存储可以实现数据中心的建设

## Vsphere Vmotion

Vmotion迁移在不关闭电源的情况下迁移虚拟机，前提是需要共享文件

## Vmware vCenter server

他可以把本地运行的系统和远程运行的系统，各种其他虚拟机变成vsphere虚拟机。

# 黑客与画家

## 垃圾邮件过滤机制

贝叶斯推断可以防止垃圾邮件

贝叶斯过滤器可以根据新收到的邮件不断的调整，收到的垃圾邮件越多，他的准确率越高。

# 代码整洁之道

## 前言

衡量代码的标准WTF/min

## 整洁代码

混乱代码会把生产力耗尽

## 有意义的命名

变量，方法的命名必须要有一定的意义。

使用能够读的出来的名称

使用便于搜索的名称

不要使用了l作为单个字母变量实在是难于理解

类名应该是名词或者名字短语

使用类，方法构成语境

封装方法很重要，对于不利于理解的建议细化一个新的方法

## 函数

函数是要尽可能的短小

# Ruby元编程

## 元编程

### 引言

元编程时编写在运行时操纵语言构件的代码

Ruby的对象是元编程的土壤

方法查找就是Ruby查找并执行方法的过程，方法查找使得可以在运行时创建方法，插入方法和调用方法甚至调用一个不存在的方法

代码块告诉我们如何打包将在未来执行的代码

### 对象模型

类中定义的都是实例方法

# rails c 进入console模式 rials –sandbox 进入沙盒模式

# irb>Project.instance\_methods.grep /^cre/ 查找以cre开头的实例方法

对象的方法是对象自身的类，对象中的实例方法才是调用对象的方法

类自身也是对象

>Module.nesting可以获取当前常量的路径

单元测试可以很好地帮我们发现冲突的类名

调用一个方法时ruby会做两件事

1. 找到这个方法，方法查找
2. 执行这个方法，通过ruby的self

方法查找会沿着祖先类（典型的文件目录结构）向上查找

### 方法

采用send方法可以在代码执行的最后一刻决定调用哪个方法，该技术成为动态派发，send方法可以调用任何方法，甚至是私有方法

动态方法slef

幽灵方法（Ghost method）method missing 可以解决大量相似方法的定义。

### 代码块

在调用一个方法时定义一个块，块会直接传递给这个方法，然后方法通过用yield回调块

# LinuxBook

## ln — 创建链接

ln 命令即可创建硬链接，也可以创建符号链接。可以用其中一种方法来使用它：

ln file link

创建硬链接，和：

ln -s item link

## 命令行编辑

Bash 使用了一个名为 Readline 的库（共享的线程集合，可以被不同的程序使用）， 来实现命令行编辑。我们已经看到一些例子。我们知道，例如，箭头按键可以移动鼠标， 此外还有许多特性。想想这些额外的工具，我们可以在工作中使用。学会所有的特性 并不重要，但许多特性非常有帮助。选择自己需要的特性。

注意：下面一些按键组合（尤其使用 Alt 键的组合），可能会被 GUI 拦截来触发其它的功能。 当使用虚拟控制台时，所有的按键组合都应该正确地工作。

## 移动光标

下表列出了移动光标所使用的按键：

|  |  |
| --- | --- |
| *表9-1: 光标移动命令* | |
| 按键 | 行动 |
| Ctrl-a | 移动光标到行首。 |
| Ctrl-e | 移动光标到行尾。 |
| Ctrl-f | 光标前移一个字符；和右箭头作用一样。 |
| Ctrl-b | 光标后移一个字符；和左箭头作用一样。 |
| Alt-f | 光标前移一个字。 |
| Alt-b | 光标后移一个字。 |
| Ctrl-l | 清空屏幕，移动光标到左上角。clear 命令完成同样的工作。 |

## 修改文本

表9－2列出了键盘命令，这些命令用来在命令行中编辑字符。

|  |  |
| --- | --- |
| *表9-2: 文本编辑命令* | |
| 按键 | 行动 |
| Ctrl-d | 删除光标位置的字符。 |
| Ctrl-t | 光标位置的字符和光标前面的字符互换位置。 |
| Alt-t | 光标位置的字和其前面的字互换位置。 |
| Alt-l | 把从光标位置到字尾的字符转换成小写字母。 |
| Alt-u | 把从光标位置到字尾的字符转换成大写字母。 |

**剪切和粘贴文本**

Readline 的文档使用术语 killing 和 yanking 来指我们平常所说的剪切和粘贴。 剪切下来的本文被存储在一个叫做剪切环(kill-ring)的缓冲区中。

|  |  |  |
| --- | --- | --- |
| 表9-3: 剪切和粘贴命令 | | |
| 按键 | | 行动 |
| Ctrl-k | | 剪切从光标位置到行尾的文本。 |
| Ctrl-u | | 剪切从光标位置到行首的文本。 |
| Alt-d | | 剪切从光标位置到词尾的文本。 |
| Alt-Backspace | | 剪切从光标位置到词头的文本。如果光标在一个单词的开头，剪切前一个单词。 |
| Ctrl-y | | 把剪切环中的文本粘贴到光标位置。 |
| *表10-1: 文件类型* | | |
| 属性 | 文件类型 | |
| - | 一个普通文件 | |
| d | 一个目录 | |
| l | 一个符号链接。注意对于符号链接文件，剩余的文件属性总是 "rwxrwxrwx"，而且都是 虚拟值。真正的文件属性是指符号链接所指向的文件的属性。 | |
| c | 一个字符设备文件。这种文件类型是指按照字节流，来处理数据的设备。 比如说终端机，或者调制解调器 | |
| b | 一个块设备文件。这种文件类型是指按照数据块，来处理数据的设备，例如一个硬盘，或者 CD-ROM 盘。 | |

### locate – 查找文件的简单方法

这个 locate 程序快速搜索路径名数据库，并且输出每个与给定字符串相匹配的文件名。比如说， 例如，我们想要找到所有名字以“zip”开头的程序。因为我们正在查找程序，可以假定包含 匹配程序的目录以”bin/”结尾。因此，我们试着以这种方式使用 locate 命令，来找到我们的文件：

[me@linuxbox ~]$ locate bin/zip

### 文本处理

* cat – 连接文件并且打印到标准输出
* sort – 给文本行排序
* uniq – 报告或者省略重复行
* cut – 从每行中删除文本区域
* paste – 合并文件文本行
* join – 基于某个共享字段来联合两个文件的文本行
* comm – 逐行比较两个有序的文件
* diff – 逐行比较文件
* patch – 给原始文件打补丁
* tr – 翻译或删除字符
* sed – 用于筛选和转换文本的流编辑器
* aspell – 交互式拼写检查器

# Git 指南

## help.autocorrect

该配置项只在 Git 1.6.1及以上版本有效，假如你在Git 1.6中错打了一条命令，会显

示：

$ git com

git: 'com' is not a git-command. See 'git --help'.

Did you mean this?

commit

如果你把help.autocorrect设置成1（译注：启动自动修正），那么在只有一个命令被模

糊匹配到的情况下，Git 会自动运行该命令。

## Git 比较二进制文件（doc，pdf）

比较二进制文件

在Git 1.6及以上版本中，你能利用 Git 属性来有效地比较二进制文件。可以设置 Git

把二进制数据转换成文本格式，用通常的diff来比较。

这个特性很酷，而且鲜为人知，因此我会结合实例来讲解。首先，要解决的是最令人头疼

的问题：对Word文档进行版本控制。很多人对Word文档又恨又爱，如果想对其进行版本控

制，你可以把文件加入到 Git 库中，每次修改后提交即可。但这样做没有一点实际意义，

因为运行git diff命令后，你只能得到如下的结果：

$ git diff

diff --git a/chapter1.doc b/chapter1.doc

index 88839c4..4afcb7c 100644

Binary files a/chapter1.doc and b/chapter1.doc differ

你不能直接比较两个不同版本的Word文件，除非进行手动扫描，不是吗？ Git 属性能很

好地解决此问题，把下面的行加到.gitattributes文件：

\*.doc diff=word

当你要看比较结果时，如果文件扩展名是“doc”，Git 调用“word”过滤器。什么

是“word”过滤器呢？其实就是 Git 使用strings 程序，把Word文档转换成可读的文本

文件，之后再进行比较：

$ git config diff.word.textconv strings

现在如果在两个快照之间比较以.doc结尾的文件，Git 对这些文件运用“word”过滤器，

在比较前把Word文件转换成文本文件。

你还能用这个方法比较图像文件。当比较时，对JPEG文件运用一个过滤器，它能提炼出

EXIF信息 — 大部分图像格式使用的元数据。如果你下载并安装了exiftool程序，可以用它

参照元数据把图像转换成文本。比较的不同结果将会用文本向你展示：

$ echo '\*.png diff=exif' >> .gitattributes

$ git config diff.exif.textconv exiftool

如果在项目中替换了一个图像文件，运行git diff命令的结果如下：

你会发现文件的尺寸大小发生了改变。

## 核弹级选项: filter-branch

如果你想用脚本的方式修改大量的提交，还有一个重写历史的选项可以用——例如，全局

性地修改电子邮件地址或者将一个文件从所有提交中删除。这个命令是filter-branch，这

个会大面积地修改你的历史，所以你很有可能不该去用它，除非你的项目尚未公开，没有其

他人在你准备修改的提交的基础上工作。尽管如此，这个可以非常有用。你会学习一些常见

用法，借此对它的能力有所认识。

全局性地更换电子邮件地址

另一个常见的案例是你在开始时忘了运行git config来设置你的姓名和电子邮件地址，也

许你想开源一个项目，把你所有的工作电子邮件地址修改为个人地址。无论哪种情况你都可

以用filter-branch来更换多次提交里的电子邮件地址。你必须小心一些，只改变属于你的

电子邮件地址，所以你使用--commit-filter：

$ git filter-branch --commit-filter '

if [ "$GIT\_AUTHOR\_EMAIL" = "schacon@localhost" ];

then

GIT\_AUTHOR\_NAME="Scott Chacon";

GIT\_AUTHOR\_EMAIL="schacon@example.com";

git commit-tree "$@";

else

git commit-tree "$@";

fi' HEAD