

Quattor 模板自动生成系统

翁云剑^{1,2}, 蒋卓明^{1,2}, 樊滑翔^{1,2}, 梁 栋^{1,2}, 孙功星¹

(1. 中国科学院高能物理所计算中心, 北京 100049; 2. 中国科学院研究生院, 北京 100049)

摘要: Quattor 软件启动节点机的 PXE 网卡远程安装节点机, 根据节点机的环境和要安装的软件包、服务等来实现自动安装节点机, 并通过用 PAN 语言编写以后缀名 tpl 结尾的模板文件来描述上述信息。针对一些 Quattor 系统管理员不熟悉模板编写的问题, 该文设计 Quattor 模板自动生成系统, 该系统能在远程使用支持 HTTPS (SSL 上的 HTTP) 协议的 Web 浏览器, 通过 Web 界面使 Quattor 自动生成模板, 可保证在安全性的前提下, 提供简单深入的远程模板自动生成。

关键词: Quattor 软件; PAN 语言; 模板; 自动生成系统

Quattor Template Auto-generation System

WENG Yun-jian^{1,2}, JIANG Zhuo-ming^{1,2}, FAN Hua-xiang^{1,2}, LIANG Dong^{1,2}, SUN Gong-xing¹

(1. Computing Center, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049;

2. Graduate University of Chinese Academy of Sciences, Beijing 100049)

【Abstract】 Quattor starts the node's PXE network card by long-distance installment. Automatic installment contains software package, service installment and so on. Even the node environment is installed. Templates which describes installment condition are used by PAN language compilations. Templates end with the suffix tpl. The system manager doesn't master Quattor. It is possible to run Quattor without knowing where to begin. However, the template automatic production system exactly solves this difficulty. It can support HTTPS in the long-distance by (on SSL HTTP) Web browser to make Quattor on the Web flat automatically produce templates. It is under the secure premise to use long-distance template automatic production.

【Key words】 Quattor; PAN language; template; auto-generation system

1 概述

从系统组成角度而言, 机群系统是由多台计算机组成的超级计算机。机群操作系统是机群系统中最基本的一个系统软件集, 它构建在节点操作系统基础之上; 但从最终用户角度而言, 机群系统由一台计算机组成, 即机群系统的构成对用户是透明的。KickStart 提供了 Linux 自动安装的基础。机群系统管理员的工作不仅要为机群节点安装操作系统, 还要为节点安装相应的应用软件, 根据需要启动相应的服务, 并对服务进行配置。对于数量不多的小型计算机机群系统而言, 手动配置是可行的, 但对于节点数量较多的机群, 手动配置则变得很困难, 因此, 需要一个机群自动配置的软件工具。

为满足北京正负电子对撞机的重大改造工程(BEPC)和北京谱仪(BES)的研制进展对计算和存储资源的需求, 高能所将构建大于 1 000 个 CPU 的机群系统和大于 5 PB 的海量存储系统, 并为机群的每个节点安装基本操作系统和网格中间件(如 LCG, glite)、软件包的更新、节点的配置等系统管理工作。

2 Quattor 介绍

Quattor^[1]起源于法国 CERN, 它是一种大规模的机群系统管理软件。Quattor 是回环结构如图 1 所示, 主要由 4 个部分组成: 配置库, 软件库, 自动安装部分和节点机。

配置库是节点机配置信息管理的核心, 主要用来保存目标节点机硬件和软件等所有相关配置信息, 如 CPU、内存、DISK、网络参数、内核参数、RPM 包等; 软件库用来保存目

标节点机所要安装的所有软件包, 包括操作系统和应用软件; 自动安装部分通过 PXE 网卡启动, 以广播的形式向 DHCP 服务器申请网络地址、TFTP 服务器的 IP 地址, 网络启动配置文件的名字等。从 TFTP 服务器上下载 PXELinux.0、网络安装启动 Linux 内核和根文件系统。通过 PXELinux 到 HTTP 服务器下载 ks.cfg 文件。根据 ks.cfg 去 HTTP 服务器下载 RPM 包。节点机执行安装, 根据 ks.cfg 执行最基本的系统安装; 配置库即是 Quattor 放置模板的配置库。系统管理员定义并导入各种自己编写的配置模板至配置库中。

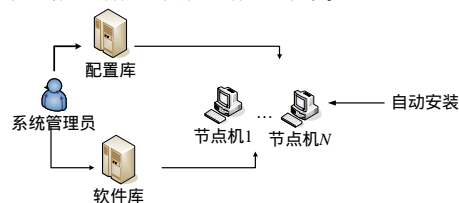


图 1 Quattor 体系结构

Quattor 的特点在于它可实现目标节点机的完全自动安

基金项目: 国家自然科学基金资助项目“国家科研网络综合平台下的高能物理网格试验床系统的建设”(90412017); 中国科学院创新基金资助项目“网格计算在高性能物理科学中的应用”(U-512)

作者简介: 翁云剑(1982-), 女, 硕士研究生, 主研方向: 计算机网络, 计算机机群; 蒋卓明、樊滑翔, 博士研究生; 梁 栋, 硕士研究生; 孙功星, 研究员

收稿日期: 2008-05-10 **E-mail:** wengyunjian@ihep.ac.cn

装、配置和管理。对于机群系统而言,假设手动给每个节点安装一个基本的 linux 操作系统,耗时 1 h,当节点数为 100 时,传统手动安装、使用 Quattor 安装和使用 Web 设计的 Quattor 在安装效益比较如表 1 所示。

表 1 机群环境下的 3 种方式节点机安装效益比较

安装方式	总耗时	管理员经验
手动安装	1 h×100 = 4 个工作日	一般的系统 管理员
Quattor	1 h + Quattor 使用(包括模板编写等)5 h = 6 h	很熟练 Quattor 的系 统管理员
Web 下的 Quattor	1 h + Web 下的 Quattor (模板自动生成系 统)2h = 3 h	一般的系统 管理员

由表 1 可见,手动安装与配置大量(如 100 个)节点较易出错。如将配置文件写好一份,由系统自动分给 100 个节点安装,将减小出错率和管理员花费时间。模板自动生成系统可使庞大的 Quattor 管理软件简化。

3 Quattor 模板自动生成系统

针对配置库的模板,本文设计一个模板自动生成系统。用 PAN 语言编写通用模块,按继承关系组成树状结构。用 perl 语言开发 Web CGI 程序。配置库里的内容主要包括 3 个部分:硬件部分(CPU、硬盘、网卡、内存、节点位置),软件部分(安装的软件列表如 OS 的基本组件、应用软件、网络中间件的软件包)和服务部分(Grub、NFS 等)。

通用网关接口(Common Gateway Interface, CGI)^[2]是 Web 服务器主机提供信息服务的标准接口,通过此类接口, Web 服务器能执行应用程序并输出。一般而言,CGI 的功能是在超文本文档与服务器应用程序之间传递信息。如果没有 CGI, Web 服务器只能提供静态文本或连接到其他服务器。CGI 程序可用任何语言来编写,如 Perl, C/C++, Visual Basic 等。由于 perl 是解释性语言,启动 Apache 服务器时,还要同时启动 perl 解释器,这样会降低服务器的运行速度。

3.1 PAN 语言介绍

PAN 语言是 Quattor 工具包的特色之一,是一种高层次的描述性语言,用来描述整个系统的配置,PAN 语言通过抽象和继承来共享一些通用信息,以减少重复。

PAN 语言将系统的配置参数按树状进行编排:

```
/hardware/memory/size=256          #内存大小
/hardware/cpus/0/vendor=GenuineIntel #cpu 供应商
```

其中,表达式等号右边为属性,由字符串或数字等简单值组成。等号左边为资源,是系统的一个组件,按路径的方式进行编辑。路径可分为相对路径(如 size)、绝对路径(如 /hardware/memory/size)和外部路径(如 //pc-wyj/hardware/memory/size)。

系统管理员用 PAN 语言编写好配置模板后,上传到配置库,如语法正确,则被 PANC 专用编译器编译成 XML 文件,存放在配置文库,供节点下载。

3.2 系统的功能设计

Quattor 模板自动生成系统基于 Quattor 的模板自动产生,应用于 scientificLinux 操作系统环境。后台是用 Perl 写的 CGI,前台是 html 编写的 Web 界面,通过表单的形式调用后台 CGI 来实现自动产生对应的 PAN 语言写的配置模板文件(如

图 2 所示),主要可分为模板生成、模板管理、模板自动生成文档。



图 2 Quattor 模板自动产生系统的主界面

3.2.1 模板生成模块

模板生成包括如下 4 个部分:

(1)节点部分,负责产生各个节点机的配置模板。查询机群环境下有哪些节点机。设置各节点机的配置信息。

(2)硬件部分,负责产生对硬件环境描述和配置的模板。管理员根据需要的硬件环境,对节点机进行分区大小、内存设置等。

(3)软件部分,负责生辰将要安装的软件包。管理员可以查看软件库中软件包列表,并可对某个软件包进行增加、删除和更新操作。

(4)服务部分,将要被需求的一些服务产生的模板。

3.2.2 模板管理模块

模板管理主要包括上传单个或批量模板到配置库,删除配置库中的模板,查询某个模板是否存在,从配置库下载模板到本地工作目录。模板管理可分为 cdbop 和 namespace。cdbop 部分将已产生的模板提交到配置库,增加、删除、升级模板;namespaces 部分对产生或提交的模板进行目录名管理。配置模板按逻辑关系组成层次结构,称为名字空间(namespaces),与文件系统类似,可将模板分类管理,使同一个模板可存在于不同名字空间。通过界面可完成的功能是名字空间管理,包括对名字空间的操作有增加、删除、修改显示系统中已存在的名字空间。

3.3 系统的设计实现及部分代码

配置库里的模板是以 PAN 语言编写的模板形式的后缀名为 .tpl 文件,如 profile_wyj.tpl:

```
include pro_hardware_asus_terminator_p4_533a;
# network settings of the node
# ethernet address - needed for PXE installs
"/hardware/cards/nic/eth0/hwaddr" = "00:0D:56:2C:D7:30";
#节点机 MAC 地址
# host name
"/system/network/hostname" = "wyj"; #节点机主机名
# IP address
"/system/network/interfaces/eth0/ip" = "192.168.25.19";
#节点机 IP 地址
```

模板 profile_wyj.tpl 主要定义节点的网络参数与继承的组件模板,具有 1 个 wyj 节点机的 MAC 地址、主机名和 IP 地址等信息。PXE 网卡启动后,通过上述信息,即可通过广播识别 wyj 节点机。

模板自动生成系统实现了从依赖于命令行 Quattor 配置库的模板生成过渡到 Web 平台的模板自动生成系统。管理员只要提供相应的信息，系统即能自动产生模板。然后由系统管理员将产生模板导入到配置库中(如图 3 所示)。



图 3 节点机模板的配置

首先在表单中输入将要产生模板的名称，后台模板会自动产生一个以后缀名tpl的模板空白文件。然后在表单中，输入节点机的MAC地址、主机名和IP地址信息。再对文件句柄进行操作，将节点机的信息写入对应的模板空白文件。写入完毕，打印输出节点机信息^[3]。操作完毕之后，关闭文件句柄，程序结束。CGI代码如下：

```
#!/usr/bin/perl
use strict;
use CGI qw(:standard);
$|=1;
print header;
my $FILEEXT;
$FILEEXT=param('textfield'),"<p>";
my $nodefile;
$nodefile="profile_$FILEEXT.tpl";
    #表单中节点机配置模板名称
print "$nodefile\n";
my $hwaddr=param('hwaddr');#表单中 MAC 地址
```

(上接第 257 页)

用于数据库访问的应用程序编程接口。对于数据库API，使用结构化查询语言(SQL)作为其数据库访问语言。ODBC是为最大的互用性而设计的，即 1 个应用程序用相同的源代码访问不同的数据库管理系统(DBMS)的能力。数据库应用程序在 ODBC接口中调用函数，ODBC接口在驱动程序针对数据库的模块中实现。驱动程序使应用程序从具体的数据库调用中隔离开来，这与打印机驱动程序将字处理程序与具体的打印机命令隔离开的方式一致^[4]。

本系统中数据库接口部分的流程如下：

(1)配置 ODBC 数据源，选择“SCATS_DATA”数据库为数据源，并将数据源命名为“ODBC_SCATS_DATA”。

(2)连接至数据源，关键代码如下：

```
SQLConnectA(m_hDBCConnection, (SQLCHAR*)pDriverName,
            SQL_NTS, (SQLCHAR*)pUserName, SQL_NTS,
            (SQLCHAR*)pPassword, SQL_NTS);
```

(3)生成 SQL 语句，关键代码如下：

```
sprintf(strSql,
"Insert into TBL_RTDETECTOR5MIN(FSTR_RECTIME, FSTR_
DETECTORID,FINT_FLOW) values ('%s', '%s', '%d') ",
pSavingData->m_StartTime,
pSavingData->m_DetectorID,
pSavingData->m_nFlow);
```

(4)使用准备执行 SQLPrepare，提高 ODBC 处理的效率。

```
my $hostname=param('hostname');#表单中主机名称
my $ipaddr=param('ipaddr'); #表单中 IP 地址
my $line;
open(INFILE,"</home/node/profile_mynode.tpl");
open(OUTFILE,">/home/node/$nodefile");
select(OUTFILE);
while($line=<INFILE>)
{$line=~s/00:FF:00:FF:00:FF/$hwaddr/g;
 #产生节点机的模板中 MAC 地址
$line=~s/mynode/$hostname/g;
 #产生节点机的模板主机名称
$line=~s/10.0.0.1/$ipaddr/g;
 #产生节点机的模板 IP 地址
print $line;}
close(INFILE);
close(OUTFILE);
```

4 结束语

Quattor 模板自动生成系统使大规模机群系统的自动安装变得更快、更方便。管理员无须手动书写配置模板，即可自动生成配置模板，在很大程度上减少了管理员的工作量和出错率。下一步的研究方向是完善该系统，如节点的硬件部分的信息由操作系统安装时自动识别，并添加到配置模板中等。

参考文献

- [1] System Administration Tool Suite[EB/OL]. (2007-05-21). <http://quattor.web.cern.ch/quattor>.
- [2] Meltzer K, Michalski B. Writing CGI Application with Perl[M]. New York, USA: Addison-wesley Professional, 2001.
- [3] 许叔云, 薛凤武. Perl 网络编程实例[M]. 北京: 中国电力出版社, 2002.

(5)执行 SQL 语句，向数据库表中执行插入记录操作，关键代码如下：

```
SQLExecDirectA(m_hCommandStmt,(SQLCHAR*)pSql,
SQL_NTS);
```

(6)解除与数据源的连接。应用程序调用 SQLFreeHandle 释放所有的语句句柄，调用 SQLDisconnect 断开与数据库的连接，再调用 SQLFreeHandle 释放连接句柄。

5 结束语

本文介绍了 SCATS 数据采集系统各部分的设计，该方案在实际应用中运行良好。目前采集的交通数据主要包括交通流量和路口相位数据，下一步的工作是进一步完善系统，以采集更多的交通信息，并优化程序的性能。

参考文献

- [1] 郑纲, 丁克良, 汪红星. SCATS 与车载 GPS 交通信息采集和融合[C]//现代化道路交通管理发展与展望——第八届多国城市交通学术会议论文集. 北京: 农业出版社, 2003.
- [2] Akhter S, Roberts J. 多核程序设计技术: 通过软件多线程提升性能[M]. 李宝峰, 译. 北京: 电子工业出版社, 2007.
- [3] 王艳平, 张越. Windows 网络与通信程序设计[M]. 北京: 人民邮电出版社, 2006.
- [4] Microsoft Inc.. Microsoft ODBC 3.0 程序员参考及 SDK 指南[M]. 希望图书创作室, 译. 北京: 北京希望电子出版社, 1999.