A. Tytko, J. Kowalski
AGH University of Science and Technology,
Faculty of Mechanical Engineering and Robotics Department of Rope Transport

STEERING MECHANICAL DEVICES BY COMPUTER OPERATING SYSTEMS

Methods which uses mobile access to manage work of that devices. Computer operating systems may be used to control and steer mechanical devices (ie. Embedded Linux/*BSD/Unix). Data collected during monitoring process may be estimated by interpolation process (ie. in batch mode programs like Octave).

Introduction. The Customers expect easy to use, automatic and reliable devices or applications. The applications or devices designed by constructors use new tested inventions and technologies to provide better way of their work. Internet and computer science and technology devices and applications develop surprisingly fast. The computer systems whose are working on-line connected to network have global tasks and have to work hard without break. Model of operating system which is used in personal computers may be expanded on peripheral devices and may be use to remote control device. There are a few developer open source project for microprocessors and microcontrollers operating systems. Microcontrollers work in devices like palmtops, ip phones, access points, ip cameras (ie. Motorola, ARM) Name of their operating system has sufix embbeded (*niux/windows). It is difficult to find some diagnostic services without computer analysys (ie. Car service station. Application runs on x86 and is connected with car computer controller via RS232 – and verifying ignition). To guarantee quality of service it is needed to check state of system elements. Nowadays often combined environment for devices is computer network and often connected devices are minicomputers. In this paper are presented methods which uses mobile access to manage work of that devices.

The Linux distribution. When we are building the linux distribution, we have to consider potential appliance and forced restrictions. Main components of the distribution, which we have to prepare are: file system (devices, binary, library, scripts), kernel and boot loader. When we do not want compile glibc, we can download complied and linked asmutils and libc from net. To compile files for other architecture we may use crosscompiler.

```
$make ARCH=arm CROSS_COMPILE=arm-linux- zImage
```

Fig 1. Linux kernel compilation for arm

System can be started loading kernel and data from local disk, flash memory or remote boot device from Etherboot or PXE using TFTP, NFS or HTTP.

Controllers and microcontollers. Devices - computers which process monitoring data may be more or less complex, According to their role, we can use mini-itx structure or some microcontrollers:

Motorola - DragonBall, ColdFire, ADI, Blackfin, ETRAX, mc68360 QUICC, ARM7TDMI, ARM9, MC68EN302, Intel i960, PRISMA, ATMEL ARM7TDMI.

Fig 2. Pentium 4 Mini ITX board with intel 845GV chipset

Fig 3. Acunia "Zingu" - this 2.7 x 3.6 in. SBC is based on an Intel Xscale i80200 processor with up to 850 MIPS performance. Includes up to 128MB SDRAM and 32MB Flash, plus built-in controllers for video, UART, AC97 Audio codec, PCMCIA, and I2C. Power consumption is under 2.5W

Monitoring devices and services. We assume monitoring data transfer through network. Monitoring system can be placed in one room, building or different mainland (replication of data base systems). The infrastructure of computer diagnostic system consist of: uninterruptible power supply, network - switch, routers, controllers of devices (minicomputers sharing services). All devices are connected to network monitoring environment. Monitoring gadgets serves remote access, control and supervise.

Collecting monitoring data. Monitoring system structure is designed in client-server mode.

Client of monitoring system locally or remotely call over and gather data from all monitoring points (clients). Monitoring data are compared to their norm factors. General system structure is presented on figure 4. The client send data to central database.

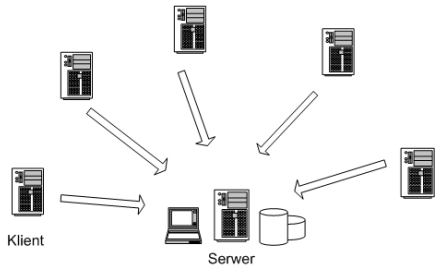

Fig 4. The structure of monitoring system

Presentation Results. Server of the monitoring system provides verification list of each parameters. The console shows which parameter (point of system) need check or adjustment.

All data are stored in database, therefore they are easy to obtain or to make report. There is possible to create plot of factor in some period of time. The data can be process in external programs through scripts or plugins. Along of data approximation you can foresee factor behavior. You can use external PERL or OCTAVE programs.

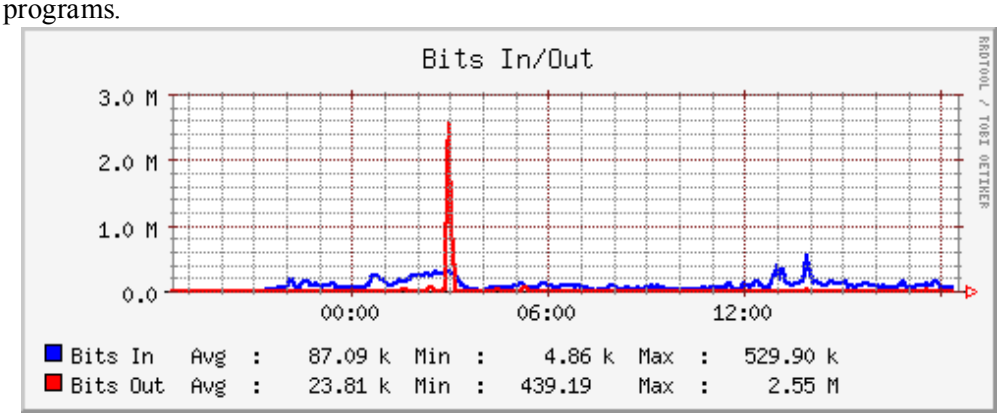

Fig 5. Plot shows bandwidth speed of internet connection

Accident reaction. The difference between real measured and ideal work factors or case of problem allows to estimate accident importance. There are some level of accident importance: ie. critical, high,

average, low. Each level and sort of accident, when it happens, force do some action (send information or stop/restart service).

Obtaining or setting factors of the service. Computer network standard has many benefit way to solve some kinds of monitoring and manage problems ie. SNMP (Simple Network Management Protocol) or remote services which can be applied to devices controlled by microprocessors with network access.

Applications which are installed on computers or controllers can through the net service, obtain information about results of external devices work or manage their work. Network is excellent environment to transport information. You can use copper wires or wireless technology and transport data on long or short distance. You can also encode transported data.

```
Snmpwalk -v 2c -c public device address service_id
```

Fig 6. List of value current factors (result of service) of some devices

Another monitoring tool. Problem of monitoring computer and telecommunication networks is usual and generally known. There are several significant tools to monitor devices and services.

- Hp Open View
- Open NMS
- NetSaint (Nagios)
- BMC Patrol

Conclusion. Computer systems are connected with wide network across switches and routers shares and provides resources. Their role is endless work. Services works on servers – hardware verified by operating system. Server are build redundant (disks, cpus, fans etc.) but it is also needed to monitor their work and in good time replace damaged element.

Hardware information of devices is accumulated and controlled by the operating system. Control of applications resolves to track their work logs. It is easier to do examine computer services rather then mechanical work.

Computer services like SMTP, WWW or DNS may be easy tested by checking the result of their request.

Industrial equipment which perform mechanical work may be controlled by dedicated electronic steering structure which locally inspect their state and work. In some cases we can not perform entire test of service or device. To solve that problem we have to monitor time of use all working parts.

1. Linux for mips http://www.linux-mips.org/ 2. Port for Linux operating system to run on systems using Motorola's 68020, 68030, 68040 and 68060 microprocessors http://www.linux-m68k.org/ 3. Linux for all ARM based machines http://www.arm.linux.org.uk/ - 4. Tiny SBCs for Embedded Linux based projects http://www.linuxdevices.com/articles/AT8498487406.htm, http://www.arcturusnetworks.com/coldfire5282.shtml. 5. Embedded Linux/Microcontroller Project http://www.uclinux.org/. 6. Nagios http://www.nagios.org/ 7. HP Open View http://www.openview.hp.com/. 8. Open NMS http://freshmeat.net/projects/opennms/. 9. MRTG http://people.ee.ethz.ch/~oetiker/webtools/mrtg/