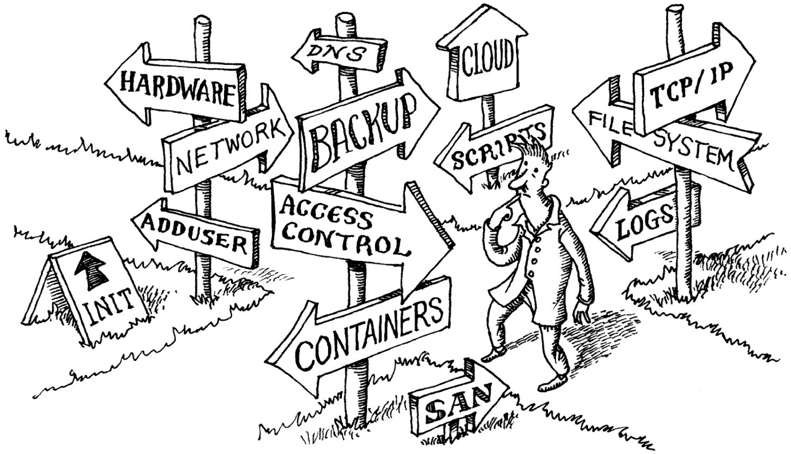
Introduction to the computer system administration

Topic 01: Fundamentals about *the system administrator*



Relevant Bibliography:

UNIX and Linux system administration handbook [Chapters 1, 31]

https://www.bsc.es

Course outline

- **Unit 1: Introduction to computer system administration**
  - Fundamentals about **the SYSTEM ADMINISTRATION** (the sysadmin)

- **Unit 2: Integration of global services** in enterprise environments I: **The INTRANET**
  - Computer systems for deploying **INFORMATION** and **VALIDATION services**
  - Computer systems for deploying "stub" *3party* **NETWORKING services**
  - Computer systems for deploying **STORAGE** and **RESOURCES sharing services**

- **Unit 3: Integration of global services** in enterprise environments II: **The INTERNET**
  - Computer systems for **WEB hosting service** management
  - Computer systems for **MAIL service** management

- **Unit 4: Configuration** & **Monitoring:**
  - Server and service **CONFIGURATION managers**
  - Server and service **MONITORING systems**

- **Unit 5: Introduction to the Data Centers**
  - Fundamentals about **DATA CENTER** design and operation

[Human organization around the *"Data Center"*]

- A "**Data Center**" is NOT only a real place which harbors computer systems
  - There are **people** too **(work teams)**
  - It's the "*office*" for system administrators
    - Although it can (fortunately) *remotely* work
      - **Remote tools for *on-promise* IT infrastructure**
      - **Private cloud platforms (Google, Amazon, Microsoft …)**
  - A good human organization is **essential** for a proper operation

- Its **structure** is usually strongly coupled to its *aims* and *purposes*:
  - In terms of **organizational aspects** specially**:**
    - **Infrastructure level**
    - **Staff level**
  - Types of DC, according to its *aims* and *purposes*:
    - DC for **cloud service and hyperscale computing:**
      - **Google, Amazon, Microsoft, Facebook …**
    - DC for **basic internet service and hosting** providers (ISPs):
      - **OVH, Telefónica, Arsys, CIC …**
    - DC for **supercomputing** – HPC/HTC services (I+D+i):
      - **BSC, CESGA, IFCA (UC), 3Mares (UC)**
    - DC for **specialized services**:
      - **Santander Bank "bunker", "small" DC for local/small businesses**

[Human organization around the *“Data Center”*]

- **Organic schema (**hierarchy**)** for the “*data center*”
  - Board of Trustees
    - Appointing the Director
    - Setting the general criteria for the operation
  - Executive Commission
    - Submitting the proposal for the annual action plan and budget to the Board of Governors
  - Management Boards
    - **General and associate director**
    - **Scientific Advisory Board**
    - Computer resources access committee
      - **Main task: defining the user profiles (clients) for the DC**
    - Area/department **directors/leaders**

example

  - **DC departments** and **work areas**:
    - **DC Operations area:**
      - **Infrastructure** department:
        - Cooling, power, security, *building* maintenance, etc.
      - **System engineering** department
        - **System administrators**
          - **System management**
          - **Monitor** everything (computers, networks, infrastructure) → **NOC**
          - **Security** experts
        - Hardware & software engineers
          - Hosting, storage, networking, *cloud,* etc.
      - **User support** department**:**
        - User support (*front line*)
        - Reception and redirection of **incidents**

[Human organization around the *"Data Center"*]

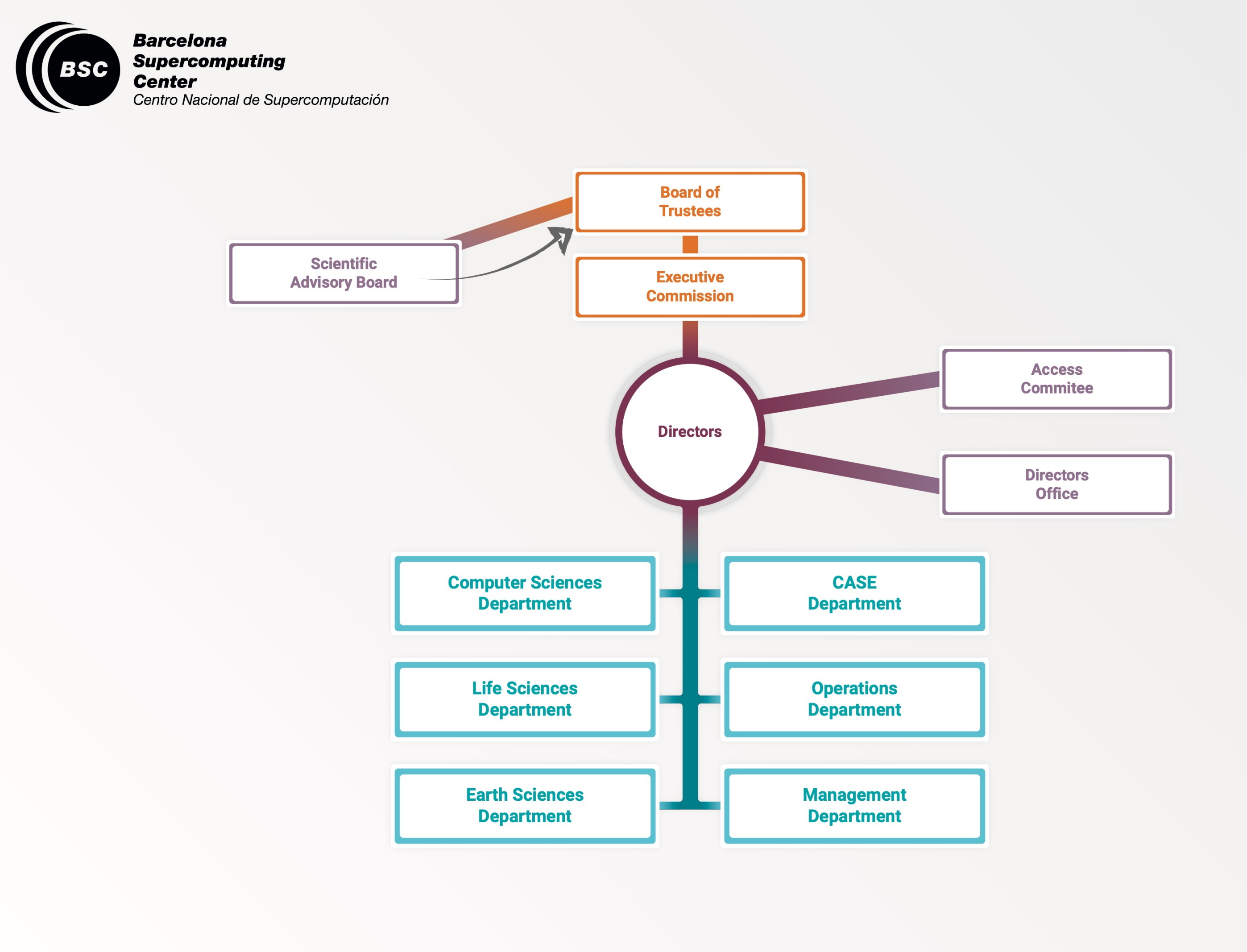
- **DC Development/operation areas:**
  - **Development** departments & groups **(Projects & apps → software)**
    - App analysts
    - Project managers
    - HPC
  - **Operation** departments:
    - According the business lines of the Data Center: ecommerce, ddbb, …

example

- **DC I+D or research area:**
  - **According to each area of knowledge**
    - **Computer science**, **Earth sciences**, **biology**, **engineering**, **human sciences**

- **DC Administration area:**
  - **Training and Learning:**
    - Technical training for users and DC staff
    - e-learning
  - **Communications and disclosure**
  - **Marketing and Sales**
  - **…**
  - **Human Resources:**
    - Staff management, purchases, agreements, etc.
  - **Finances and Accounting**

Human organization in the BSC: case of use



example

Barcelona Supercomputing Center, 2019

[The system administrator (sysadmin)]

- Definition: This is the worker responsible for the **deployment**, **start up**, **administration** and **proper operation** of a computer or computer environment (IT)
  - The system administrators must be able to deploy **policies**, take **actions** and define **rules** for making sure that systems are:
    - Usable
    - Reliable
    - Safe
    - Efficient
  - **Use technology** to accelerate and support (help) the enterprise's mission.
    - Remember …: **Enterprise needs drive IT activities, not the other way around**.

- There is not really a "super sysadmin" or unique system administrator
  - The admin usually becomes specialized as part of **work team**
    - It is more than a group of technical folks
  - System administrators' **roles**:
    - **Computer system** administrator (*system manager*):
      - **Operating systems (UNIX, Linux, MS …)**
      - **Virtualization and "*Cloud Computing*"**
      - **Computing systems:**
        - » by and large
        - » specialized
          - Banking, Heavy Industry, Robotic, **Research** (**HPC**) …
    - **Database** administrator
    - **Networking** administrator
    - **Security** systems administrator

Essential tasks, duties and responsibilities (sysadmins team)

- Facility **infrastructure support**
  - Monitor facility equipment:
    - **Cooling** systems, **power** *feeds* and **power** *distribution* systems, **security** devices, etc.
  - Deploy control mechanisms for a good operation
    - **DCIM** systems

- Monitoring the use and state of the computer systems:
  - Know what is happening (at all times) with systems:
    - **Hardware**:
      - **Networking + Storage + Computers**
    - **Middleware**:
      - **Virtualization infrastructures ("*cloud computing*")**
    - **Software**:
      - **OS, services, user apps**
  - System accounting:
    - Servers(hw), OS and services
  - Perform periodic performance reporting to support capacity planning

Essential tasks, duties and responsibilities (sysadmins team)

- Hardware **support**
  - Administrators who work with physical hardware (**on-site staff**)
    - Not cloud or hosted systems
  - Installation and configuration of:
    - Racks, servers, store devices, backup systems (drivers and robots)
    - Networking drivers and devices (routers, switches, computer network devices)
  - Apply patches and update *firmwares*
  - Deploy control mechanisms for a good operation
- Software **support**
  - Installation and configuration of **operating systems**
  - Installation and configuration of "***cloud***" (**virtualization**) **environments**
  - Installation and management of **storage systems** (logical structures, file systems …)
  - Installation and management of **services and user apps**
  - Customization (and *tuning*) of systems to requirements
  - **Updating** all the software
  - System **Accounting**
- Security **support**
  - Deploy **security polices**
  - Design and development of **contingency actions** and well-defined protocols
  - Deploy and use **security control tools**
    - Physical security, security in networking, premises security (servers and services)
  - Deploy and oversee "**backups**"
- Tuning performance

Essential tasks, duties and responsibilities (sysadmins team)

- Access control
  - **User/group account** provisioning
  - Define **user/group *profiles***
  - Help users in their daily tasks with computers
    - **User support**
  - Elaboration of **user manuals and tutorials**
- Incident management (detection and resolution)
  - By themselves or third-party maintenance tools/companies
    - Hardware, software and services
- Documentation management and maintenance
  - About operating systems, services, DC infrastructure, protocols, user manuals, etc.
- **And … :**
  - Planning of **maintenance contracts** and **new equipment purchases**, according to each moment …
  - Working side by side with vendors & manufacturers

- **Actually … "fire fighting":** →
  - **Troubleshooting!!**

"…although most of the time they spend their time creating scripts to automate tasks"

Evi Nemeth. 😬

Good practices

- **Do not improvise**:
  - Everything you are going to do in the system must be **planned**.
    - Always **plan** ahead.
- **Reversible actions:**
  - Every action in our system must be reversible.
  - You must provide "**undo**" procedures.
- **Incremental actions**:
  - You should apply the changes **gradually**, step by step.
- **"First test then do"**:
  - **Test**, **test** and **test** before making any changes to the system
    - Especially if it's a system running in a *critical operation environment*
  - Very useful and <u>necessary</u>:
    - **Use lab environments** (develop and test)
    - Virtualization/container environments

[More job descriptions … from USENIX]

- **Management-Level Job descriptions (individual)**
  - **Level I:** *Novice* System Administrator
  - **Level II:** *Junior* System Administrator
  - **Level III:** *Intermediate/Advanced* System Administrator
  - **Level IV:** *Senior* System Administrator
- **Management-Level Job descriptions (teams)**
  - **Management Level I:** *Technical* Lead
  - **Management Level II:** *System Administration* Manager
  - **Management Level III:** *IT* Director
  - **Management Level IV:** *Chief* Information Officer

https://www.usenix.org/system/files/lisa/books/usenix_22_jobs3rd_core.pdf

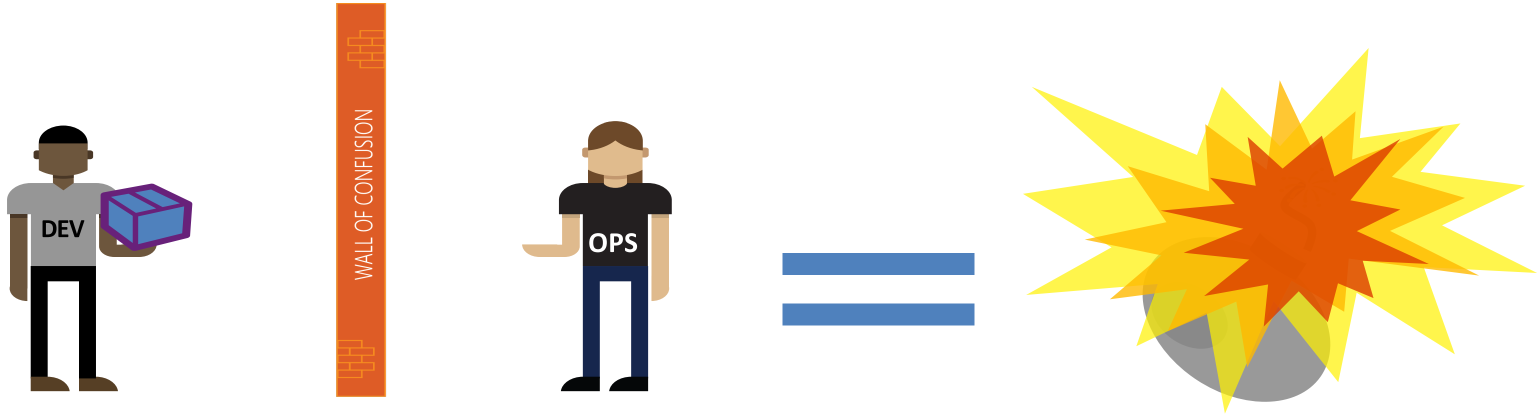
Good practices

- **Have knowledge about what is going to be done**:
  - Foresee the consequences of your changes (as far as possible)
  - **Get ready** (studying) before any action is taken
- **Automate tasks:**
  - Backups
  - Regular monitoring
  - User account maintenance
  - Disk space maintenance
  - **Routine tasks**…
- **Document everything**:
  - Policies, process, changes, **incidents**…
- **Report** (to users) whenever possible:
  - System power off, application of patches and updates, etc.

[The DevOps philosophy]

- The development, delivery and evolution (*life cycle*) of IT **apps and services** depends on the IT work teams
- **IT** organization = **developers** + **operators** (systems administrators)
  - **Dev**: Programmers, app analysts, app owners, project managers
    - Specialists to push software products forward with new features and enhancements.
  - **Ops**: System and network administrators, security experts and monitors, data center staff, databases admins …
    - Change-resistant engineers to provide 24X7 management of a production env.

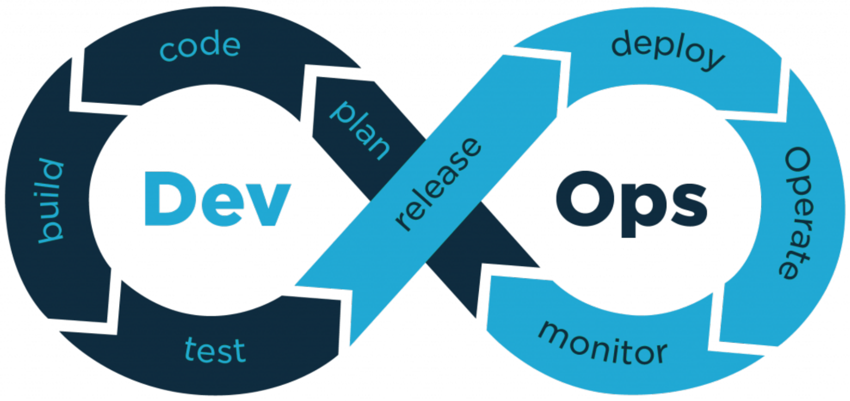
isolation work



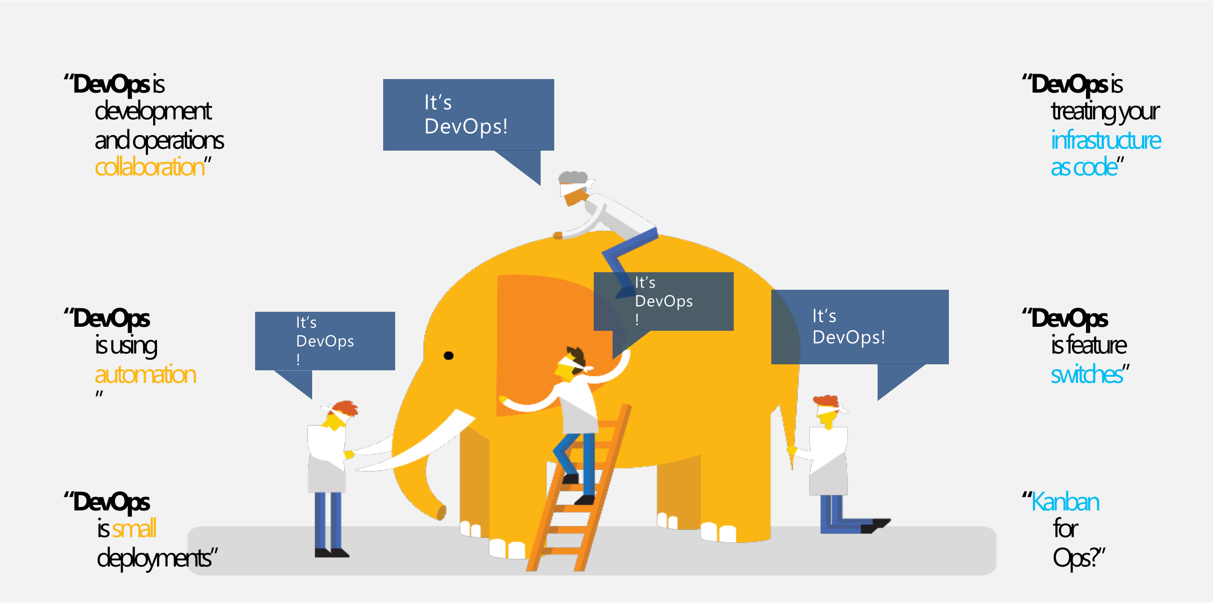
- **DevOps** is a **work philosophy** to mingle everyone (operation/development engineer) in the belief that working together as a collaborative team is more efficient
- **DevOps** (from early 2000s) comes to **improve** this relationship.

[The DevOps philosophy]

- **DevOps** is a mindset, a cultural shift, where IT teams adopt *new ways of working*:
  - **Collaborative work and shared development**
  - Automation
  - Continuous Improvement
  - Customer-centric action
  - Create with the end in mind
- Target:
  - **Improve the software qualty**
  - **Accelerate the software delivery**
- DevOps in Not a:
  - tool
  - person
  - tittle
  - team
- It is **C.L.A.M.S**:
  - **C**ulture
  - **L**ean
  - **A**utomation
  - **M**easurement
  - **S**haring



See chapter 31 of Bibliography ref. 2


"DevOps is development and operations collaboration"
It's DevOps!
"DevOps is treating your infrastructure as code"
"DevOps is using automation"
It's DevOps !
It's DevOps !
It's DevOps!
"DevOps is feature switches"
"DevOps is small deployments"
"Kanban for Ops?"

[The DevOps tools (some of them)]

- **Ticketing** and task management tools
  - Mantis, OSTicket, ServiceDesk
- **Version** Control tools
  - GitHub, **GitLab**, Bitbucket
- **Container** Management tools
  - **Docker**, Kubernetes, Mesos
- Application **Performance Monitoring** tools
  - **Prometheus**, Dynatrace
- **Deployment** & Server **Monitoring** tools
  - Splunk, Datadog, Sensu

- **Configuration** Management tools
  - **Ansible**, Chef, puppet
- **CI/Deployment** Automation tools
  - Bamboo, Jenkins
- **Test** Automation tools
  - Test.ai, Ranorex, Selenium
- **Codeless Test** Automation tools
  - **AccelQ, Appvance, Testim.io**

https://www.qentelli.com/thought-leadership/insights/devops-tools

[Ticketing and Task management systems]

- **It lies at the heart of every functioning IT group:**
  - Enhancement **requests**
  - **Issues** management
  - **Bug** control & documentation
- It helps staff avoid two common pitfalls:
  - **Tasks unsolved** because everyone thinks they are taken care of by someone else
  - **Duplication** of effort when many people work on the same problem
- Common function:
  - Accept requests and track them (using several interfaces)
  - **Managers:**
    - They can assign tickets to groups o individual staff members and manage a lot of information about request resolutions:
      - **Open/close tickets**
      - **Average time to close**
      - **Productivity**
      - **Workload resolution**
  - **Technical staff:**
    - They can query the system to see the **pending issues**
    - Use and manage issue resolution **documentation**
  - **End users:**
    - **Notify** an issue through a request … (no phone, no mail …)
    - They can find out the **status** of their requests

[Ticketing and Task management systems]

- Ticket ownership
  - Every task (ticket) should have a single, well-defined owner
  - Responsibility for taking sure tasks get done.
- User acceptance of ticketing systems
  - Users willing to wait a reasonable time to revolve a request, but nor **willing to be ignored**
  - Sysadmin must take a time to make sure the understand users are requesting and **keep users informed**

- **Sample ticketing systems**
  - **Open source:**
    - Double choco Latte
    - Mantis
    - OSTicket
    - Bugzilla
  - **Commercial**:
    - EMC Ionix
    - HEAT
    - ServiceDesk

[Ticketing and Task management systems]

- When a request is assigned by the ticket system by manager(s), the sysadmin asigned *gets down to work*:
- **Procedure**:
  - **Step 1:** Thoroughly **analyze** the issue - *reproduce*
    - Run daemons/process in **debug level** (*foreground*)
    - **READ** outputs **carefully** from console and **system log files**
  - **Step 2**: Documentation:
    - First level of information:
      - **Have a look at system reference manuals ($ `man`)**
      - **Check the latest bug list published about it.**
    - Second level of information:
      - **Massive "*online*" sources:**
        - » Google
        - » Blogs and Forums
          - https://stackoverflow.com
          - https://serverfault.com
    - Third level of information:
      - **Specialized: (manuals and tutorials)**
        - » RFCs (Request for Comments documents)
        - » Linux Documentation Project (tldp.org)
        - » Technical manuals and developer "*howto's*"
      - **Generic: essential books**
        - » Bibliography of this course
        - » O'Reilly series
          - From "Unix in a Nut-shell" to …
  - **Step 3:** Elaborate a solution plan and run it
    - Keep daemons/process in **debug level** (*foreground*)
  - **Step 4:** If it fails, go to **step 2**

1	User-level general commands and apps
2	System calls and kernel error codes
3	Library functions and calls
4	Device drivers and network protocols
5	Standard file formats and conventions
6	Games and demostrations
7	Miscellanea
8	System administration commands and daemons

```
$ man [-options] [command]
        -a: display all the manual pages
        that match "command", not just the
        first.
        -K: search for the specified string
        in all man pages.
        <n>: secction

$ man ls
$ man 3 gethostbyname
```

"When stuck, Google!!!"
Apply filters in searches (distributions, versions …)

[Documentation resources maintenance]

- An important "transversal" task for sysadmin is to maintain a **"good" local documentation**
  - Doc. reduces the likelihood of a single point of failure
  - Doc. aids reproducibility
  - Doc. save time (*)
  - Doc. enhances the intelligibility and understanding of the systems
- Documentation should be kept in a **well-defined** spot
  - Simple **knowledge management systems**:
    - Wikis
    - Blogs (CMS)
  - **Infrastructure as a code:**
    - Use new configuration managers
      - **Ansible, puppet …**
    - Kept in the form of **configuration definitions**
      - **Stored and tracked using version control system as GIT**

  - With a single location
  - Easier search function
- If you must doc. elements **manually** → on short and light weight documents
  - If you need more details somewhere → additional one-page document
- A good documentation is:
  - Concise, relevant and unpolished.
  - **Cut to the chase!**
  - **No redundant**
- Integrate documentation into processes:
  - **Comments** in configuration files
  - Infrastructure as code (Puppet, Ansible …)
- Try to keep documentation **updated**

[Documentation samples]

- System administrator *on-line* **bibliography**:
  - Webs about **system administration**
    - ✓ http://systemadmin.es/
    - ✓ http://www.linuxfoundation.org/
    - ✓ https://www.usenix.org
    - ✓ https://linux.slashdot.org
  - Webs about **computer system security**
    - ✓http://www.securityfocus.com/archive
  - Webs about "**data centers**"
    - ✓http://www.datacentremanagement.com/
    - ✓http://whitepapers.datacenterknowledge.com/
    - ✓http://www.datacenterjournal.com/category/facilities/
  - **Press** and research papers & conferences
    - ✓ http://www.admin-magazine.com
    - ✓ http://www.linuxpromagazine.com

[Documentation samples]

- System administrator *on-line* **bibliography**:
  - Webs about **system administration**
    - ✓ http://systemadmin.es/
    - ✓ http://www.linuxfoundation.org/
    - ✓ https://www.usenix.org
    - ✓ https://linux.slashdot.org
  - Webs about **computer system security**
    - ✓ http://www.securityfocus.com/archive
  - Webs about "**data centers**"
    - ✓ http://www.datacentremanagement.com/
    - ✓ http://whitepapers.datacenterknowledge.com/
    - ✓ http://www.datacenterjournal.com/category/facilities/
  - **Press** and research papers & conferences
    - ✓ http://www.admin-magazine.com
    - ✓ http://www.linuxpromagazine.com

More details in: **Unix and Linux System Administration Handbook**, *chapters 1, 31 and*
A Brief History of System Administration.