

# **PACMAN: EL MOTOR DE GESTIÓN DE SOFTWARE EN ARCH LINUX**

## **Integrantes:**

Saldaña Navarrete Andrea

Salgado Cruz Emiliano Roman

## **Asignatura:**

Sistemas Operativos

Grupo 6

# ¿QUÉ ES ARCH LINUX?

Distribución  
rolling-release.

Filosofía KISS.

Dirigida a  
usuarios  
experimentados.

Publicado bajo  
la GPL.

# **SISTEMA *ROLLING-RELEASE***

---

Los paquetes en los repositorios siempre están en su última versión estable.

---

No hay versiones fijas del sistema operativo.





**¿QUÉ ES UN  
GESTOR DE  
PAQUETES?**

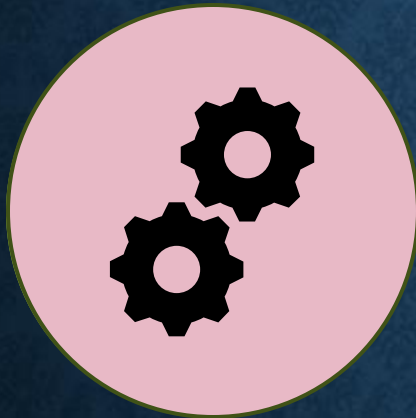
# PRIMERO, ¿QUÉ ES UN PAQUETE?

Es un archivo comprimido que contiene:

- Binarios o ejecutables del software.
- Scripts para su configuración.
- Metadatos (nombre, versión, dependencias).



# AHORA SÍ, LOS GESTORES DE PAQUETES



FUNCIONES: INSTALAR, ACTUALIZAR,  
ELIMINAR Y CONFIGURAR SOFTWARE  
DE FORMA AUTOMÁTICA



CLAVE EN SISTEMAS LINUX.





# ¿QUÉ ES PACMAN?

- Gestor predeterminado en Arch Linux.
- Conocido por su simplicidad, velocidad y enfoque minimalista.

– S

– R

– Q





# CARACTERÍSTICAS PRINCIPALES



# **DISEÑO MINIMALISTA Y EFICIENTE**

---

Escrito en lenguaje C.

---

Utiliza scripts en Bash  
para operaciones.

---

Comandos cortos y  
consistentes.

# MANEJO DE DEPENDENCIAS

Instala solo lo necesario.



Soporte para  
dependencias  
opcionales (`--asdep`).



# DE FORMA MÁS TÉCNICA...

1. El gestor identifica las relaciones entre paquetes.
2. Construye un grafo dirigido de dependencias.
  - Paquetes como vértices.
  - Relaciones de dependencia como aristas.
3. Detecta posibles ciclos para evitar conflictos durante la instalación.

# CÓDIGO FUENTE Y DEPENDENCIAS

- Archivo deps.c del repositorio oficial de Pacman
- Funciones clave:  
\_alpm\_warn\_dep\_cycle,  
\_alpm\_sortbydeps.
- Evita conflictos y asegura orden correcto.

## deps.c

```
101  /* Convert a list of alpm_pkg_t * to a graph structure,
102  * with an edge for each dependency.
103  * Returns a list of vertices (one vertex = one package)
104  * (used by alpm_sortbydeps)
105  */
106  static alpm_list_t *dep_graph_init(alpm_handle_t *handle,
107  alpm_list_t *targets, alpm_list_t *ignore)
108  {
109  alpm_list_t *i, *j;
110  alpm_list_t *vertices = NULL;
111  alpm_list_t *localpkgs = alpm_list_diff(
112  | alpm_db_get_pkgcache(handle->db_local), targets, _alpm_pkg_cmp);
113  |
114  /* We create the vertices */
115  for(i = targets; i; i = i->next) {
116  alpm_graph_t *vertex = _alpm_graph_new();
117  vertex->data = (void *)i->data;
118  vertices = alpm_list_add(vertices, vertex);
119  }
120  |
121  /* We compute the edges */
122  for(i = vertices; i; i = i->next) {
123  alpm_graph_t *vertex_i = i->data;
124  alpm_pkg_t *p_i = vertex_i->data;
125  for(j = vertices; j; j = j->next) {
126  alpm_graph_t *vertex_j = j->data;
127  alpm_pkg_t *p_j = vertex_j->data;
128  if(_alpm_pkg_depends_on(p_i, p_j)) {
129  vertex_i->children =
130  | alpm_list_add(vertex_i->children, vertex_j);
131  }
132  }
133  }
134  |
135  return vertices;
136  }
137  |
```

# REPOSITARIOS

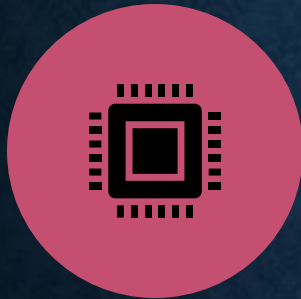
Repos oficiales:  
Core, Extra,  
Community.

AUR: scripts de  
la comunidad.

Riesgo:  
responsabilidad  
del usuario.



# BASE DE DATOS DE PACMAN



UBICACIÓN:  
/VAR/LIB/PACMAN.



ARCHIVOS EN TEXTO  
PLANO.



TRANSPARENCIA,  
SIN *HIDDEN LOCKS*.

# INSTALACIÓN DESDE CÓDIGO FUENTE

- Soporte para PKGBUILD.
  - *Scripts* de Bash con información necesaria para descargar, compilar y empaquetar software desde su código fuente.
  - Contienen:
    - Metadatos del paquete.
    - Dependencias.
    - Orígenes del código fuente.
    - Funciones de construcción.
    - Opcional: parches personalizados, variables útiles, funciones adicionales.



# UTILIDAD DE LOS PKGBUILD

- Generan paquetes con extensión `.pkg.tar.zst`
- Se crean al ejecutar `makepkg`
- Se compone de:
  - Archivos del software instalado.
  - Metadatos del paquete.
  - *Scripts* opcionales.







# COMPARACIÓN CON OTROS GESTORES

# **VELOCIDAD DE INSTALACIÓN DE PAQUETES**

```
PS C:\Users\emili> Measure-Command {  
>>     winget install Neovim.Neovim --accept-package-agreemen  
ts --silent  
>> }
```

```
Days           : 0  
Hours          : 0  
Minutes        : 0  
Seconds        : 18  
Milliseconds    : 695  
Ticks          : 186959778  
TotalDays      : 0.000216388631944444  
TotalHours     : 0.00519332716666667  
TotalMinutes   : 0.31159963  
TotalSeconds   : 18.6959778  
TotalMilliseconds : 18695.9778
```

# WINGET

Instalación de neovim



```
real    4m2.862s
user    0m9.929s
sys     0m3.556s
```

## DNF

Instalación de neovim

```
real    1m29.238s
user    0m1.372s
sys     0m0.877s
```

# PACMAN

Instalación de neovim

# VENTAJAS Y DESVENTAJAS DE PACMAN

**Ventajas:** rápido,  
resuelve  
dependencias.

**Desventajas:** curva  
de aprendizaje,  
menos paquetes  
precompilados.