



**Universitat  
de Lleida**

# **PRACTICA 1**

---

Enric Guitart

Departament d'Informàtica i Enginyeria Industrial  
Escola Politècnica Superior  
Universitat de Lleida

Introducció

Escenari

Client

Servidor

Programari

Opcions programari

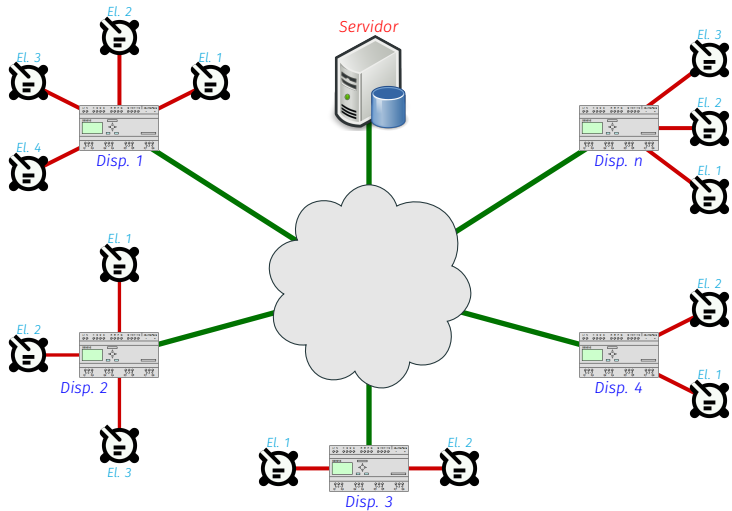
Test



- Objectius
  - Aprendre a programar aplicacions de xarxes
  - Entendre el model client-servidor
  - Aprendre a programar i dissenyar un protocol de comunicacions
- Arquitectura
  - Model client-servidor
- Implementació
  - Comunicacions per sistema de lectura i emmagatzematge de dades fabricació



# ESCENARI



- Dispositiu que proporciona informació de processos de fabricació

## *Estructura*



- Dispositiu que proporciona informació de processos de fabricació

## *Estructura*



- Dispositiu que proporciona informació de processos de fabricació

## *Estructura*



- Dispositiu que proporciona informació de processos de fabricació

## *Estructura*





- Dispositiu que proporciona informació de processos de fabricació

## *Estructura*



- Equip d'emmagatzemament i control

*Estructura*



- Equip d'emmagatzemament i control

*Estructura*



- Equip d'emmagatzemament i control

*Estructura*



- Equip d'emmagatzemament i control

*Estructura*



- Equip d'emmagatzemament i control

*Estructura*



- **client:** Implementació del client
- **client.cfg:** Arxiu paràmetres dispositiu
- **client1.cfg:** Arxiu paràmetres dispositiu
- **client2.cfg:** Arxiu paràmetres dispositiu
- **client3.cfg:** Arxiu paràmetres dispositiu
- **client4.cfg:** Arxiu paràmetres dispositiu no autoritzat
- **server:** Implementació del servidor
- **server.cfg:** Arxiu paràmetres servidor
- **bbdd\_dev.dat:** Arxiu de dispositius autoritzats en el sistema



- Tant en el servidor com en el client accepten els següents paràmetres en la línia de comandes:
  - **-c** *nom-arxiu*: especificar arxiu de paràmetres del dispositiu
  - **-d** *n*: especificar un nivell de *debug*.
    - 9: Nivell màxim, mostra tots els events de l'evolució del programa
  - **-l**: fer log en arxiu **server.log** o **client.log** (es fa amb -d 9).
  - **-t** *n*: endegar test de l'altra entitat
- Servidor:
  - **-u** *nom-arxiu*: especificar nom arxiu dispositius autoritzats (si és diferent a **bbdd\_dev.dat**)





# TEST DEL CLIENT (1)

## S'ha d'especificar el test en el servidor (-t n)

- -t 1: Envia [REG\_NACK] als tres primers paquets [REG\_REQ].
- -t 2: Envia [REG\_REJ] als tres primers paquets [REG\_REQ].
- -t 3: Envia [INFO\_ACK] amb Id. incorrecta en el primer procés de subscripció.
- -t 4: Envia [INFO\_ACK] amb Id. com. incorrecte procés 1 de subscripció.
- -t 5: Envia [INFO\_NACK] als dos primers paquets [REG\_INFO].
- -t 6: No s'envia resposta al primer paquet [REG\_INFO].
- -t 7: No s'envia resposta als paquets [ALIVE] rebuts.
- -t 8: S'envia [ALIVE\_REJ] a partir del tercer [ALIVE] rebut.
- -t 9: S'envia [ALIVE] amb Id. com. incorrecte als paquets [ALIVE] rebuts.



## TEST DEL CLIENT (2)

### S'ha d'especificar el test en el servidor (-t n)

- -t 10: S'envia [ALIVE] amb Id. client erroni des de tercer [ALIVE] rebut.
- -t 11: No s'envia resposta als paquets [ALIVE] 4, 5, 8, 9, 13, 14, 15, 16, 17 i 18.
- -t 12: S'envia la resposta a [SEND\_DATA] amb Id. com. incorrecte.
- -t 13: S'envia la resposta a [SEND\_DATA] amb nom element incorrecte.
- -t 14: S'envia la resposta a [SEND\_DATA] amb Info incorrecte.
- -t 15: No s'envia resposta a [SEND\_DATA].
- -t 16: S'envia Id del servidor errònia en els paquets [SET\_DATA] i/o [GET\_DATA].
- -t 17: En paquets [SET\_DATA] i/o [GET\_DATA] s'intercanvia tipus d'element ( $I \leftrightarrow O$ ).
- -t 18: S'envia Id del dispositiu errònia en els paquets [SET\_DATA] i/o [GET\_DATA].



## S'ha d'especificar el test en el client (-t n)

- -t 1: S'envien dos primers paquets [REG\_REQ] amb dades incorrectes:  
Paquet 1: `id` erroni                      Paquet 2: `Id. com.` erroni.
- -t 2: S'envien els tres primers paquets [REG\_REQ] amb el camp de dades no buit.
- -t 3: En dos primers registres s'envia el paquet [REG\_INFO] erroni:  
Paquet 1: `Id. com.` erroni                      Paquet 2: camp `data` buit.
- -t 4: No s'envia paquet [REG\_INFO] en primer procés de registre.
- -t 5: No s'envinen [ALIVE] finalitzat el procés de registre.
- -t 6: S'envia tercer [ALIVE] del primer procés de registre amb camp de dades no buit.



## TEST DEL SERVIDOR (2)

### S'ha d'especificar el test en el client (-t n)

- -t 7: S'envia quart [ALIVE] dels dos primers processos de reg. amb camp Id. com. erroni.
- -t 8: No s'envien els paquets [ALIVE] 3, 5, 7, 10, 11, 15, 15, 16 i 17.
- -t 9: No s'envia [SEND\_DATA] un cop establerta la connexió TCP.
- -t 10: S'envia [SEND\_DATA] amb Id. dispositiu incorrecte.
- -t 11: S'envia [SEND\_DATA] amb Id. com. incorrecte.
- -t 12: S'envia [SEND\_DATA] amb nom element incorrecte.
- -t 13: Resposta a paquets [SET/GET\_DATA] amb Id. dispositiu incorrecte.
- -t 14: Resposta a paquets [SET/GET\_DATA] amb nom element incorrecte.
- -t 15: No resposta als paquets [SET\_DATA] i/o [GET\_DATA].

