

ARQUITECTURA DE SOFTWARE I: PATRON PROXY

**INF-552**



25 DE NOVIEMBRE deL 2019

Luffi

[Dirección de la compañía]

**ALUMNA: DORIS VIVEL BALDERRAMA TORRICO 213036827**

**ING: JOSUE VEIZAGA**

**Contenido**

[1. Introduccion 2](#_Toc25538970)

[2. Tipo de Patrón 2](#_Toc25538971)

[3. Diagrama genérico Proxy 2](#_Toc25538972)

[4. Diagrama de clases dinámicas 3](#_Toc25538973)

[4.1. Patrón Proxy: 3](#_Toc25538974)

# Introduccion

El patrón Proxy se utiliza como intermediario para acceder a un objeto, permitiendo controlar el acceso a él. Para ello obliga que las llamadas a un objeto ocurran indirectamente a través de un objeto proxy, que actúa como un sustituto del objeto original, delegando luego las llamadas a los métodos de los objetos respectivos.

El patrón Proxy debe ser utilizado cuando:

* Se necesite retrasar el coste de crear e inicializar un objeto hasta que es realmente necesario.
* Se necesita una referencia a un objeto más flexible o sofisticada que un puntero.

Existen diferentes tipos de proxy:

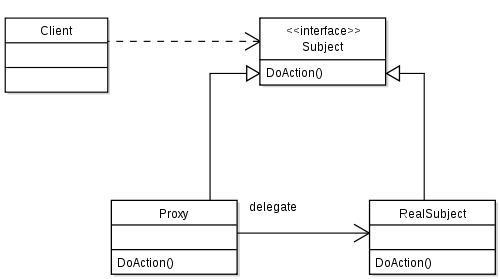
* **Proxy remoto**: proporciona un representante local de un objeto situado en otro espacios de direcciones (en otro dispositivo conectado en red).
* **Proxy virtual**: usados para crear objetos costosos sólo cuando se soliciten.
* **Proxy de protección**: permiten controlar el acceso a un objeto cuando es accesible o no, dependiendo de determinados permisos.
* **Referencia inteligente**: un sustito de un puntero, que realiza operaciones adicionales en el momento de accederse al objeto.

# Tipo de Patrón

**Tipo:** Estructural

**Propósito:** proporcionar un objeto que haga de intermediario (proxy) de otro, para controlar el acceso a él

# Diagrama genérico Proxy



# Diagrama de clases dinámicas

## Patrón Proxy:

