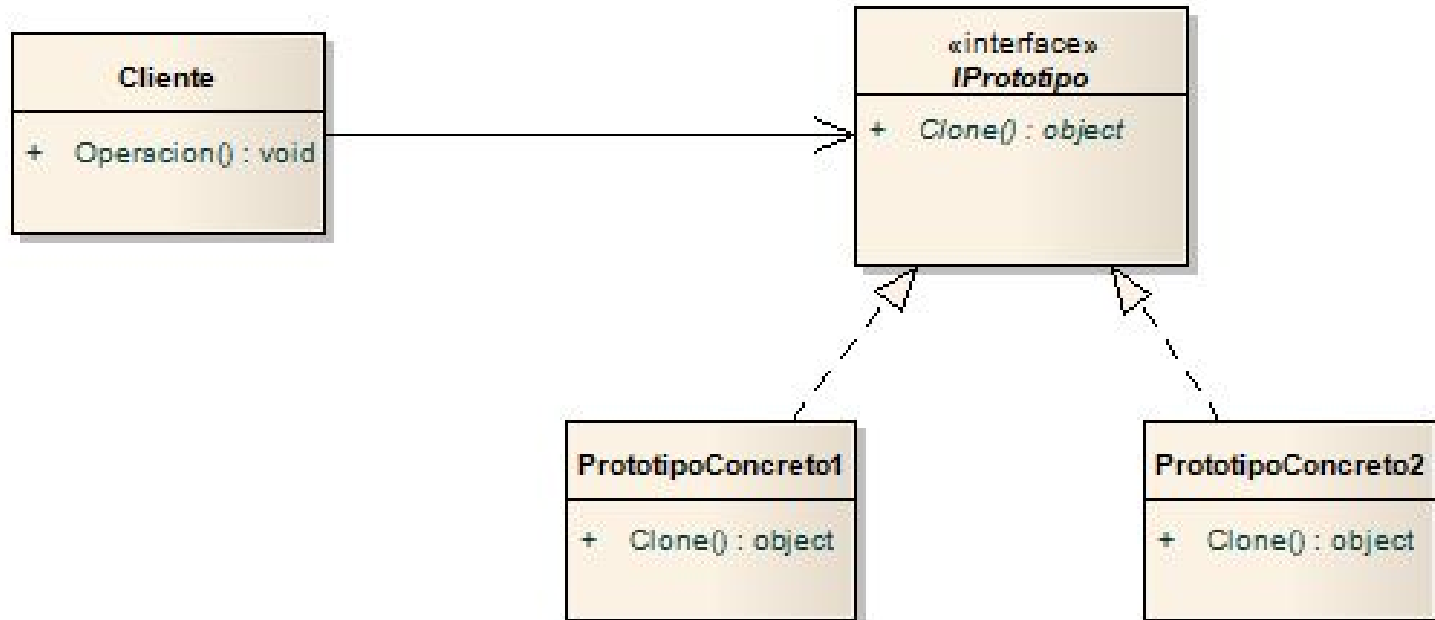


Patrón Prototype

Angel Tomas



Modelo Patrón Prototype





Cuando Usar

- cuando el costo de generar una instancia más alto en comparación de realizar una copia de una instancia ya existente.
- cuando sea necesario “salvar” el estado de un objeto en un determinado momento realizando una copia del mismo, Ej. clonación de figuras



sobre el proyecto:

- Descripción del Caso: El aplicativo se trata de una linea de produccion de mesas en una carpintería, donde se produce una mesa genérica que posteriormente se le personaliza en sus atributos.
- Está implementado en nodejs:
- Usa base de datos mongodb
- Ejecución: Para ejecutar el proyecto se requiere:
 - Levantar el servicio de base de datos mongo
 - Ejecutar en la raíz del proyecto(instalación de librerías): npm install
 - Desplegar el proyecto:node app.js

Uso del patrón: método clone

```
var clone = require('clone');
var models = require('./models'),

Schema = models.Schema;

var mesaSchema = new Schema({
  color:{type: String,default:'Marron'},
  forma:{type: String,default:'Rectangular'},
  esArmable:{type: String,default:'No'},
  esDesplegable:{type: String,default:'No'}
});

mesaSchema.methods = {
  clone : function(){
    //compia simple
    //alternativo
    //return clone(this);

    return {
      id:this.id,
      color:this.color,
      forma:this.forma,
      esArmable:this.esArmable,
      esDesplegable:this.esDesplegable,
      clone:this.clone};
  }
}

var Mesa = models.model('Mesa',mesaSchema,'mesa');
module.exports = Mesa;
```

Probando los servicios Rest

The screenshot displays a REST client interface with a sidebar on the left and a main panel on the right. The sidebar has two tabs: 'History' (selected) and 'Collections'. Under 'History', there is a 'Clear all' button and a list of requests under the 'Today' section: a GET request to 'http://localhost:3000/lista' and two POST requests to 'http://localhost:3000/savecatalogo'. The main panel shows the details of the selected GET request. It includes a 'Type' dropdown set to 'No Auth' and tabs for 'Body', 'Cookies', 'Headers (6)', and 'Tests'. The 'Body' tab is active, showing a JSON response in 'Pretty' format. The response is a JSON object with 'success' set to true and 'mesas' as an array of three table objects. Each object contains fields for 'id', 'color', 'forma', 'esArmable', and 'esDesplegable'.

```
1 {
2   "success": true,
3   "mesas": [
4     {
5       "_id": "59d99bdb54987454226a6ea8",
6       "esDesplegable": "No",
7       "esArmable": "No",
8       "forma": "Rectangular",
9       "color": "Marron"
10    },
11    {
12      "id": "59d99bdb54987454226a6ea8",
13      "color": "azul",
14      "forma": "borde redondo",
15      "esArmable": "totalmente",
16      "esDesplegable": "solo las patas"
17    },
18    {
19      "id": "59d99bdb54987454226a6ea8",
20      "color": "verde",
21      "forma": "sin borde",
22      "esArmable": "parcialmente",
23      "esDesplegable": "no"
24    }
25  ]
26 }
```



referencias

<http://java-white-box.blogspot.pe/2014/10/patrones-de-diseno-patron-prototype-que.html>

<https://danielggarcia.wordpress.com/2014/02/24/patrones-de-creacion-iii-patron-prototype/>