
Padrão de Desenho Assassino em Série

Marcos Eduardo Bolelli Broinizi – 3286311
Peter Kreslins Junior – 3286401

Assassino em Série

Intenção

Define um assassino que se utiliza de algoritmos bem definidos para cometer crimes.

Também conhecido como

Serial Killer

Motivação

De um modo geral entende-se por assassino qualquer criminoso, homem ou mulher, que mate ao longo do tempo. Assassinos em série, por sua vez, são aqueles que, por estatísticas de pesquisadores, tem um mínimo de 4 mortes. Mais do que isso, eles matam seguindo um determinado padrão, seja através dos tipos de vítima selecionada, método ou motivo.

Eles matam na própria família, como mulheres que envenam os maridos ou pais que matam os filhos. Também podem agir numa cidade inteira, ou mesmo num estado, na caça de uma próxima vítima.

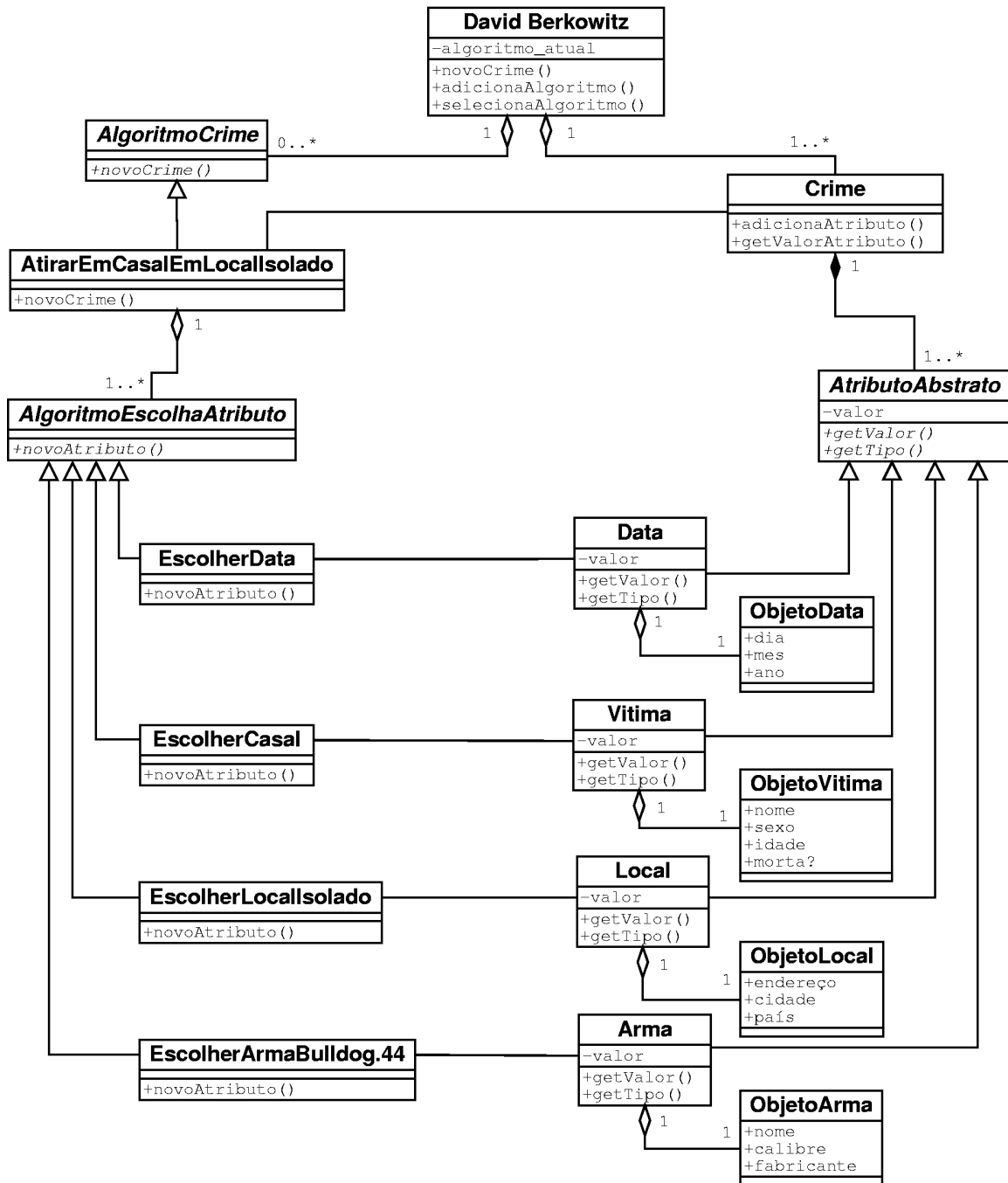
No entanto, o ponto mais importante de um assassino em série é o modo pelo qual ele liga as vítimas de acordo com um padrão muitas vezes bem definido. Assim, um assassino em série pode atacar grupos que dividem uma determinada característica, como prostitutas de uma mesma área ou homossexuais, por exemplo.

Modelando, por exemplo, o assassino em série David Berkowitz, também conhecido como "The son of Sam", responsável por instituir um reinado de terror na área da cidade de Nova York no período de 1976 até 1977, matando seis pessoas e ferindo outras sete, e está atualmente cumprindo sentença de 365 anos.

Neste exemplo (ver diagrama página X), a classe David Berkowitz possui um algoritmo de formação de um crime baseado no padrão que o deixou famoso. Os seus crimes possuem, além da data, um local escolhido dentre pontos isolados da cidade de New York frequentados por casais de namorados, que são as suas vítimas. Portanto a classe `AtirarEmCasalEmLocalIsolado` é composta por algoritmos que são responsáveis por escolher atributos, ou seja, data, casal, local e uma arma. Para isso existem as classes:

| | |
|-------------------------------------|---|
| <code>EscolherLocalIsolado:</code> | escolhe um Local isolado em New York. |
| <code>EscolherData:</code> | escolhe a Data do crime. |
| <code>EscolherCasal:</code> | escolhe um casal no local selecionado criando duas Vítimas. |
| <code>UtilizarArmaBulldog.44</code> | escolhe como Arma sua Bulldog.44. |

A classe David Berkowitz mantém referências para os crimes executados e em execução, para o algoritmo de crime. Já o algoritmo de crime mantém referências para os algoritmos de escolha de atributos, que por sua vez mantém referência para o respectivo tipo de atributo a ser criado. O atributo possui, também, uma referência para um objeto representativo do valor desse atributo. Como por exemplo no caso do atributo Arma que sempre mantém uma referência para o `ObjetoArma` com as propriedades `Nome = 'Bulldog'`, `Calibre = 0.44`.



Aplicabilidade

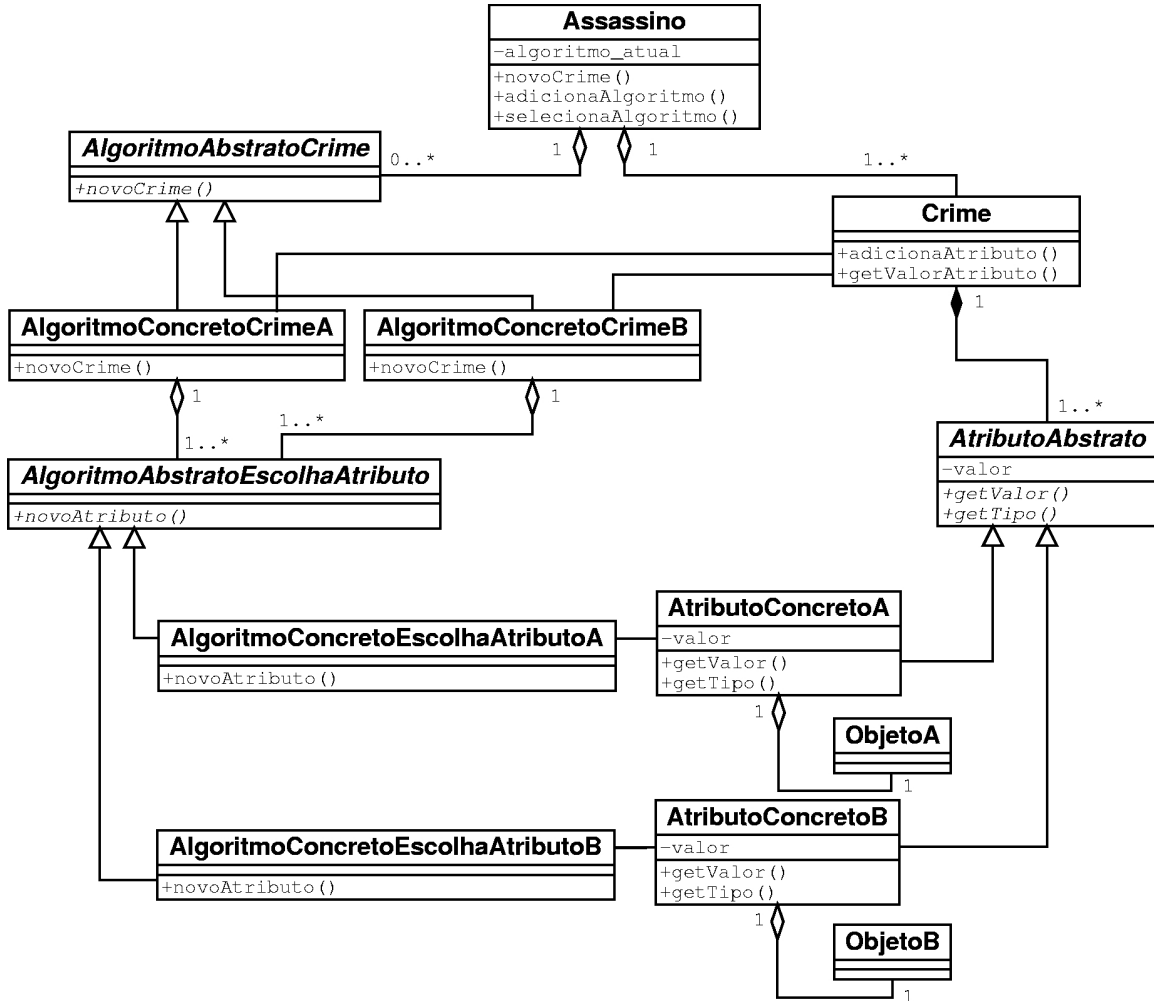
Use o padrão Assassino em Série quando:

O Assassino possui um padrão de comportamento para escolher os atributos inerentes ao crime. Ele pode assumir mais de um algoritmo em diferentes tempos. Os tipos de atributos do crime são determinados por esse algoritmo.

O padrão 'Assassino em Série' pode, também, ser utilizado de forma abstrata para modelar um criminoso que age de forma sequencial segundo um padrão, mesmo

não sendo um assassino. Ex.: terrorista que explode prédios em Manhattan de acordo com sua altura.

Estrutura



Participantes

Assassino(David Berkovitz)

Contém as características próprias de acordo com a implementação (ex: *nome, idade, nacionalidade*), as funcionalidades específicas do assassino (*Hanibal tem uma funcionalidade comerVítima por exemplo*), juntamente com um conjunto de algoritmos para execução de crimes e um outro conjunto de crimes criados à partir de algum dos algoritmos citados. Estabelece uma operação para adicionar novos algoritmos de execução de crime, uma operação para selecionar dentre os algoritmos contidos um para se tornar o algoritmo ativo e uma operação para criar um novo crime delegando essa responsabilidade para o algoritmo ativo.

AlgoritmoAbstratoCrime(*AlgoritmoCrime*)

Declara uma interface comum a todos os algoritmos de crime. O Assassino utiliza essa interface para executar o algoritmo esperando como retorno um novo crime de acordo com o algoritmo especificado.

AlgoritmoConcretoCrime(*AtirarEmCasalEmLocalIsolado*)

Implementa o algoritmo responsável por criar um novo crime. Possui um conjunto de algoritmos de escolha de atributos utilizados para criar os atributos do crime.

AlgoritmoAbstratoEscolhaAtributo(*AlgoritmoEscolhaAtributo*)

Declara uma interface comum a todos os algoritmos de escolha de atributos. O AlgoritmoConcretoCrime utiliza esta interface para executar o algoritmo esperando como retorno um novo atributo.

AlgoritmoConcretoEscolhaAtributo (*EscolherData, EscolherCasal, EscolherLocalIsolado, UtilizarAmaBuldog.44*)

Implementa algoritmo responsável por criar um novo AtributoConcreto.

Crime (*Crime*)

Contem as características do crime, ou seja, um conjunto de atributos e seus respectivos valores para o crime em questão. Provê uma operação que permite adicionar novos atributos ao crime e recuperar o valor de um atributo.

AtributoAbstrato (*Atributo*)

Declara a interface comum a todos os atributos, estabelecendo que cada um deve conter um valor, disponibilizar uma operação para se obter este valor e outra para indicar que atributo este representa (*vide consequências*).

AtributoConcreto (*Data, Vítima, Local, Arma*)

Funciona como um encapsulador de um objeto, implementando a interface definida por AtributoAbstrato, cada AtributoConcreto contém como valor uma instância de um ObjetoConcreto.

ObjetoConcreto (*ObjetoData, ObjetoVítima, ObjetoLocal, ObjetoArma*)

Objeto do sistema cuja instância que representa uma valor assumido por um atributo em um crime. Estes objetos possuem também funcionalidades próprias, normalmente inerentes à aplicação.

Colaborações

- Assassino, AlgoritmoConcretoCrime e AlgoritmoConcretoEscolhaAtributo interagem para criar um novo Crime, estabelecendo seus AtributosConcretos. O assassino pode passar toda a informação necessária ao algoritmo de crime quando este é chamado, ou o Assassino pode se passar como argumento para o algoritmo. A informação necessária para o algoritmo de escolha de atributo pode ser passada pelo algoritmo de crime, o que pode incluir o crime parcialmente preenchido com outros atributos e, ou, as informações do Assassino.
- AlgoritmoConcretoEscolhaAtributo necessita, como seu nome indica, opções para serem escolhidas, onde encontrar tais informações e como elas são acessadas através das chamadas é deixada em aberto (*várias são as forma possíveis: o algoritmo sabe desde sua criação todas as possíveis opções, ou existe uma representação do conjunto de elementos que compreendem o mini-mundo do Assassino que pode crescer de acordo com a interação do Assassino, ex: mudar-se faz com que ele conheça novos locais, ou ele compra uma nova arma aumentando suas opções, entre diversas, além de muitas outras formas de representar estas opções*). É importante ressaltar que isto reflita as opções do Assassino no momento de criação do crime.

Conseqüências

- *Abstração na criação de um crime.* A responsabilidade de criar um crime transfere-se do Assassino para seu padrão de assassinato, tornando mais fácil isolar, por exemplo, assassinos com padrões muito semelhantes mesmo que possuam características díspares.
- *"Copy-cat".* Permite reutilização de um mesmo algoritmo para mais de um Assassino, deixando clara a relação de cópia do padrão de assassinato por parte de um novo Assassino.
- *Composição por algoritmos de escolha de atributos.* Permite a reutilização de algoritmos de escolha por diversos algoritmos de crime, facilitando verificar a evolução dos padrões dentro de um mesmo assassino, e a ligação entre padrões de assassinos diferentes através das mesmas escolhas de um mesmo atributo.
- *Possibilidade de alteração entre mais de um algoritmo de assassinato.* É fácil e simples criar um novo algoritmo de crime e o associar à um Assassino pré-existente, sem perder seu rastro nem o algoritmo de assassinato inicial. Permitindo, inclusive, que os algoritmos criem crimes com atributos diferentes.
- *Dois níveis de abstração e delegação.* O uso do algoritmo de crime composto por algoritmos de escolha de atributos leva a sucessivas delegações que , além de criar atributos abstratos encapsulando os reais objetos.
- *Representação complexa, porém mais versátil.* É fácil elaborar uma forma mais estática e simples, abrindo mão de múltiplos algoritmos, estabelecendo um conjunto bem definido de atributos, evitando os *dois níveis de abstração*, porém esta forma não seria capaz de descrever uma boa parte de assassinos em série que alteram seus padrões, inclusive os atributos, dessa forma perdendo seu rastro, como se houvesse um novo assassino sendo criado, quando na verdade ele é o mesmo.

Exemplo de forma alternativa

