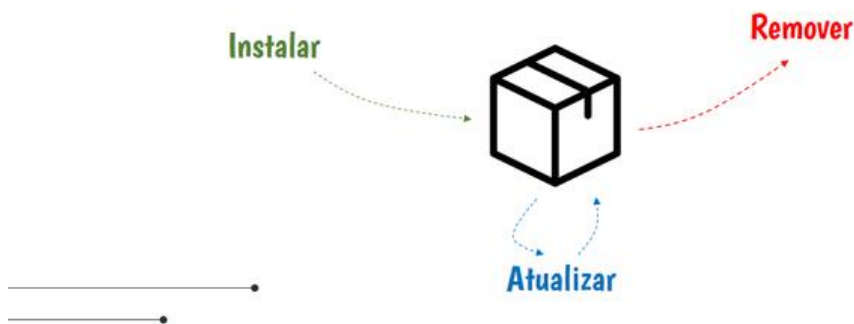


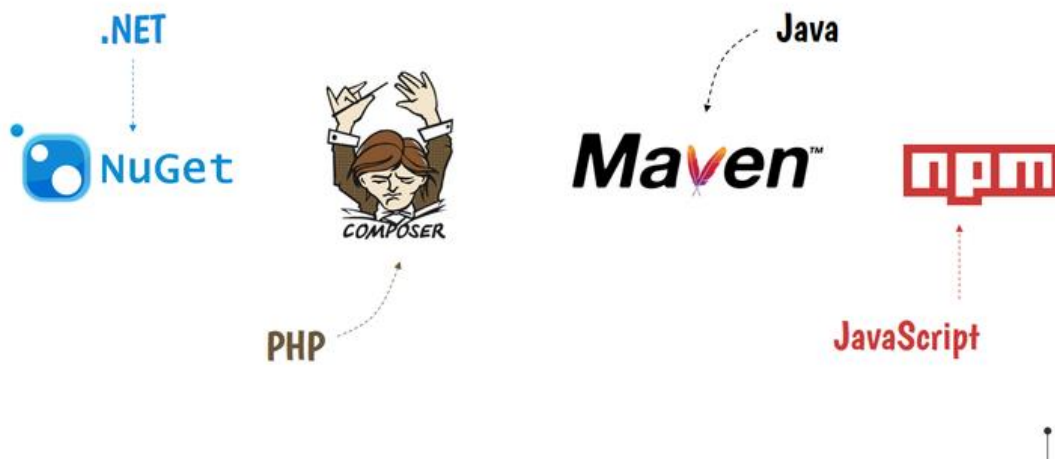
## O que é NuGet

Gerenciador de pacotes/dependências do ambiente de programação .NET.



NuGet é o gerenciador de pacotes do ecossistema .NET. Por meio dele podemos instalar, atualizar e remover dependências externas do nosso projeto." title="NuGet é o gerenciador de pacotes do ecossistema .NET. Por meio dele podemos instalar, atualizar e remover dependências externas do nosso projeto."

## O que é NuGet



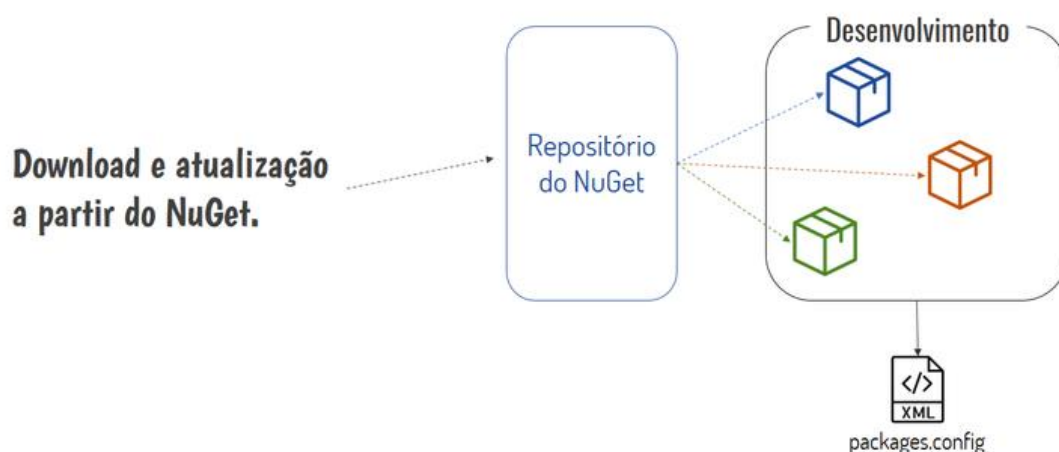
O NuGet está para o .NET assim como o Composer está para o PHP, o Maven para o Java e o NPM para o JavaScript/Node.js" title="O NuGet está para o .NET assim como o Composer está para o PHP, o Maven para o Java e o NPM para o JavaScript/Node.js"

## Gerenciando pacotes **sem** o NuGet



Quando não usamos o NuGet é preciso fazer download de cada biblioteca (DLL) no site do seu desenvolvedor. Quando uma atualização é lançada, precisamos estar atentos para baixar a nova versão e incluí-la no projeto manualmente." title="Quando não usamos o NuGet é preciso fazer download de cada biblioteca (DLL) no site do seu desenvolvedor. Quando uma atualização é lançada, precisamos estar atentos para baixar a nova versão e incluí-la no projeto manualmente."

## Gerenciando pacotes **com** o NuGet



Ao usar o NuGet todas as bibliotecas ficam centralizadas em um repositório, a partir do qual podemos instalar os pacotes e obter suas versões mais recentes. Tudo isso é feito a partir da ferramenta integrada ao Visual Studio." title="Ao usar o NuGet todas as bibliotecas ficam centralizadas em um repositório, a partir do qual podemos instalar os pacotes e obter suas versões mais recentes. Tudo isso é feito a partir da ferramenta integrada ao Visual Studio."

centralizadas em um repositório, a partir do qual podemos instalar os pacotes e obter suas versões mais recentes. Tudo isso é feito a partir da ferramenta integrada ao Visual Studio.

Com frequência, durante o desenvolvimento das nossas aplicações, precisamos instalar pacotes de terceiros para realizar as mais diversas tarefas, como acesso a bancos de dados, validações, etc.

Antes do NuGet esse procedimento era feito de forma manual, ou seja, precisávamos realizar nos sites dos desenvolvedores o download das bibliotecas e incluí-las em nossos projetos. Quando uma nova versão era lançada, era preciso baixá-la novamente e substituir em todos os locais em que ela estivesse sendo usada.

Com o NuGet esse procedimento é simplificado. Todas as bibliotecas ficam centralizadas no repositório nuget.org e por meio de uma ferramenta integrada ao Visual Studio é possível instalar, atualizar e remover os pacotes de forma bastante prática.

Quando instalamos um pacote em nossa aplicação o NuGet o registra no arquivo packages.config, semelhante ao pom.xml do Maven, composer.json do Composer e package.json do NPM. A partir desse arquivo podemos reinstalar com facilidade todos os pacotes posteriormente.

Abaixo temos um exemplo de arquivo packages.config contendo o pacote do Entity Framework:

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="EntityFramework"
    version="6.1.3"
    targetFramework="net461" />
</packages>
```

Graças a esse mecanismo não é preciso enviar a pasta packages para todos os desenvolvedores da equipe ou para o repositório de controle de versão. Basta enviar o código do projeto e ao abri-lo no Visual Studio o NuGet poderá restaurar todos os pacotes dos quais o projeto depende.