**O que é inferno das dependências?**

Inferno das dependências (dependency hell), ocorre principalmente quando as aplicações dependem de versões especificas, principalmente de aplicações terceiras, mas é implantada com versão diferente das que são requeridas. Quando o sistema se torna cada vez maior (com muitos pacotes), transformando-o em um verdadeiro pesadelo, devido ao número de dependências.

Existindo especificações pouco detalhadas de dependências, você tornara a versão “promíscua”, ou seja, assumira a compatibilidades com novas versões mais do que é aconselhável. Ou, caso existam especificações muito detalhadas de dependências ou muito engessadas, existe um grande risco de bloquearmos a versão (impedindo que tenham seus pacotes atualizados sem que haja uma nova liberação de versão).

Então o inferno das dependências é onde nosso sistema está se algumas das duas situações acima (“promiscuidade” ou bloqueio) impeçam-no de ser atualizado facilmente e principalmente de modo seguro.

Um exemplo bem conhecido de inferno das dependências eram as versões antigas do Windows (como por exemplo o Windows 98), que possuíam todas as bibliotecas compartilhadas, em DLLs, armazenadas no diretório System32, sem que existisse quaisquer tipos de versionamentos, ou seja, novas versões eram obrigadas a substituírem integralmente as versões presentes.

Isso impossibilitava que aplicações dependessem de versões diferentes da DLL presente na biblioteca.

Com implementação de conceito assemblies, introduzido no .NET, tornou possível o versionamento das bibliotecas de DLL, distinguindo-as (mesmo se tiverem o mesmo nome) através de atribuição de número. Com isso novas aplicações podem utilizar DLLs instaladas com versões mais recentes, sem a necessidade de substituição das anteriores.

Inferno de Dependências no Caso Jarbas:

Jarbas é um componente de um projeto, desenvolvido por Eduardo Cuducos, chamado Serenata de Amor. Este projeto consiste em robôs de buscas que utilizando inteligência artificial para investigar, ou como o próprio idealizador sugere, passar um pente fino nos gastos com a chamada verba indenizatória, na qual todos os nossos deputados têm direitos durante o mandato, podendo-a utilizar sem licitação.

A ideia é utilizar a data science e machine learning, para analisar de forma ágil, fácil e transparente todos os gastos com esta verba, cruzando-os com informações como CNPJ e conteúdo dos gastos. O projeto é totalmente escrito em código aberto e está disponível no GitHub.

O inferno das dependências ocorreu durante a atualização automática de um pacote, o reprint, que foi da versão 0.3.0 para a 0.4.0, ocasionando um erro ao acessar o banco de dados, acesso esse que já funcionará.

Devido ao versionamento foi possível, após efetuar debugging, identificar qual componentes estava causando a quebra do Jarbas e para confirmação foi reinstalada a versão 0.3.0.

O Pyup é um robô que acompanha as atualizações das dependências do Pyton, enviando solicitações automáticas para o repositório, permitindo que sejam gerenciadas suas dependências em um local central com o restante do seu código. Cirando um pull request para cada atualização, onde poderá discutir com a equipe e fazer alterações necessárias, contribuindo para o controle de gerenciamento de mudança.

Algumas alternativas para o Pyup são: Bibliotecas.io, David, requer.io e dependencies.io

Alexandre do Carmo – R.A. 618107356

Felipe Gabriel dos Santos – R.A. 618106864

Jefferson Felipe Maia de Souza – R.A. 618106790