

O Yocto e Buildroot são ferramentas desenvolvidas para a criação de distribuições customizadas do Linux. Estas duas ferramentas são as mais utilizadas e possuem algumas diferenças as quais vão ser apresentadas durante o resumo.

O Yocto é mais completo e flexível em comparação ao Buildroot e possui uma curva de aprendizado bem maior também. As características do Yocto é ter ser um projeto com código aberto com colaboração da comunidade em fornecer modelos, métodos e ferramentas para facilitar a criação de sistemas customizados baseados em Linux para produtos embarcados.

Ele usa Openembedded como sistema de construção e tem como saída dessa construção três componentes que são:

- \*Target SDK que é a coleção de bibliotecas e arquivos de cabeçalho que representam o software instalado;
- \*Target run-time binaries, este inclui um carregador de inicialização, módulos de kernel, kernel, imagem do sistema de arquivos raiz e outros arquivos para implementar o Linux;
- \*Package feed é uma coleção de pacotes de software disponíveis para instalação.

O Yocto tem a facilidade de ser expandido por meio de camadas que faz com que novas funcionalidades possam ser adicionadas por publicações independentes que não estavam disponíveis nas versões do projeto ou até para adicionar mais personalizações no sistema.

A indústria utiliza amplamente o projeto Yocto por ter grande apoio de varias empresas e uma comunidade grande de desenvolvedores.

Em meio a tanto de vantagens apresentadas, o projeto Yocto tem a maior desvantagem entre os dos projetos que é a curva de aprendizado que é muito íngreme o que exige um grande tempo de aprendizagem além de exigir um sistema mais “forte” para utilizar todas as ferramentas para a construção dos vários pacotes, kernel e outros componentes essenciais para a customização do Linux

O projeto Yocto exige um host de construção com uma CPU com arquitetura x86. O motivo de ele exigir um host de construção é ,sempre que possível, ele utiliza de processamento paralelo e para configurar o host de construção ,é preciso instalar pacotes de software adicionais .

Já com o Buildroot, em comparação ao Yocto, é “uma ferramenta mais simples, eficiente e fácil de usar para gerar sistemas Linux embarcados por meio de compilação cruzada”.

Ele foi pensado em ser simples e minimalista. Em outras palavras, ele vai desabilitar as configurações opcionais de tempo de compilação para todos os pacotes e por conta disso resulta em um sistema menor e mais enxuto. Caso precise de alguma configuração que não esteja habilitado, o desenvolvedor deverá habilitar esses recursos.

Este projeto é mais fechado e menos flexível visto que ele cria todos os componentes da fonte mas não oferece suporte ao gerenciamento desses pacotes no destino. As aplicações podem atualizar o sistema de arquivos de destino mas não aceita instalar novos pacotes dentro do sistema em execução .

As saídas de uma construção utilizando o projeto Buildroot são:

- \*O conjunto de ferramentas utilizado para construir todos os binários de destino;

\* A imagem do sistema de arquivos raiz e quaisquer outros arquivos necessários para implementar o Linux na plataforma;

\*O kernel, boot-loader e módulos de kernel apropriados para o hardware.

Em comparação ao projeto Yocto, o Buildroot é focado na simplicidade e por isso é mais fácil de aprender. O sistema de construção principal é pequeno e permite que o desenvolvedor compreenda o sistema durante a criação, mas não peca na capacidade de expansão para atender qualquer necessidade.

Como o Buildroot é simples e mais enxuto, ele cria as menores imagens e em menos tempo já prontas para o uso.

Como o projeto busca a simplicidade, o desenvolvedor deve fazer uma significativa customização para deixar o sistema o mais eficaz dentro da utilização. As configurações são salvas em um único arquivo e caso queira implementar o sistema customizado em várias plataformas deve-se fazer modificações a cada nova plataforma.

O maior problema desse projeto é que caso o sistema precise de uma mudança nos arquivos, ele vai precisar de uma reconstrução completa de todos os pacotes o que implica em mais tempo para as compilações.

#### Comparação

O Yocto teve uma evolução devido a grande comunidade que possui que incluíram melhorias nas compilações como redução do tempo e tamanho das imagens geradas. Por ele ser mais flexível e completo, ele dá uma maior maleabilidade do desenvolvedor

Já o Buildroot, por ser mais simples e de fácil aprendizagem em comparação ao Yocto, requer maior trabalho quando se fala de sincronização em várias plataformas.