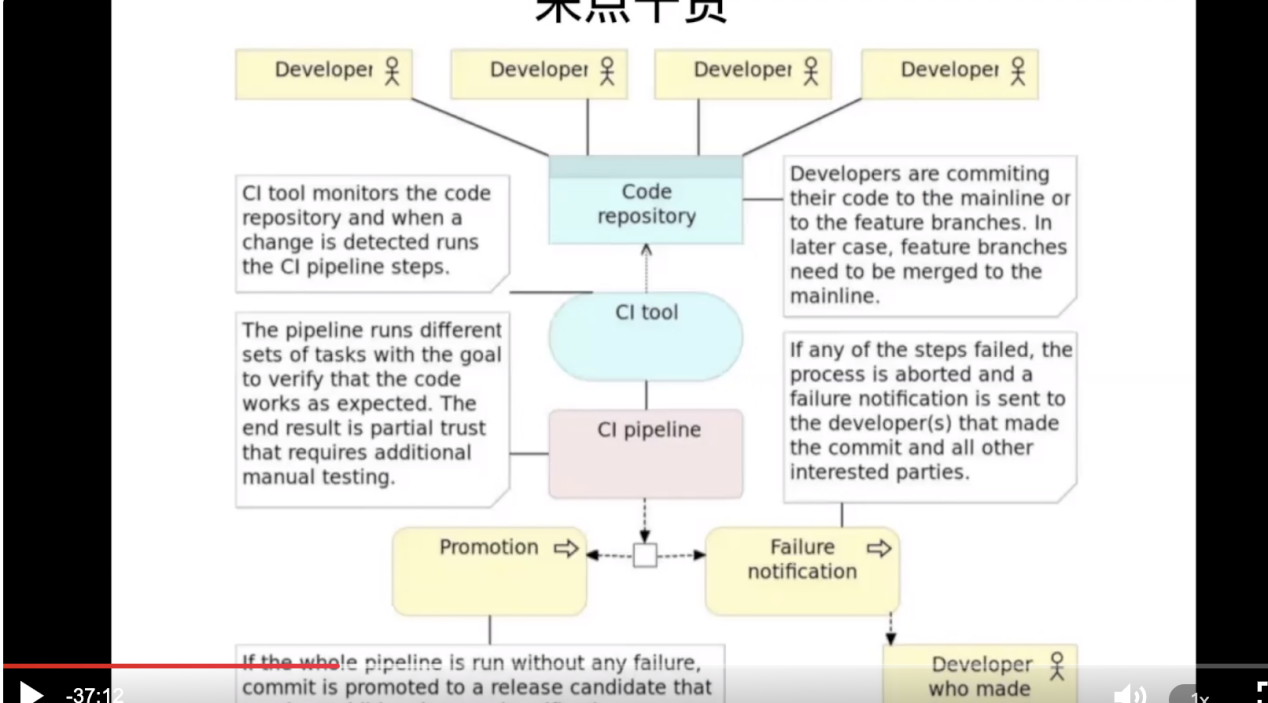
## CI continuous integration 持续集成

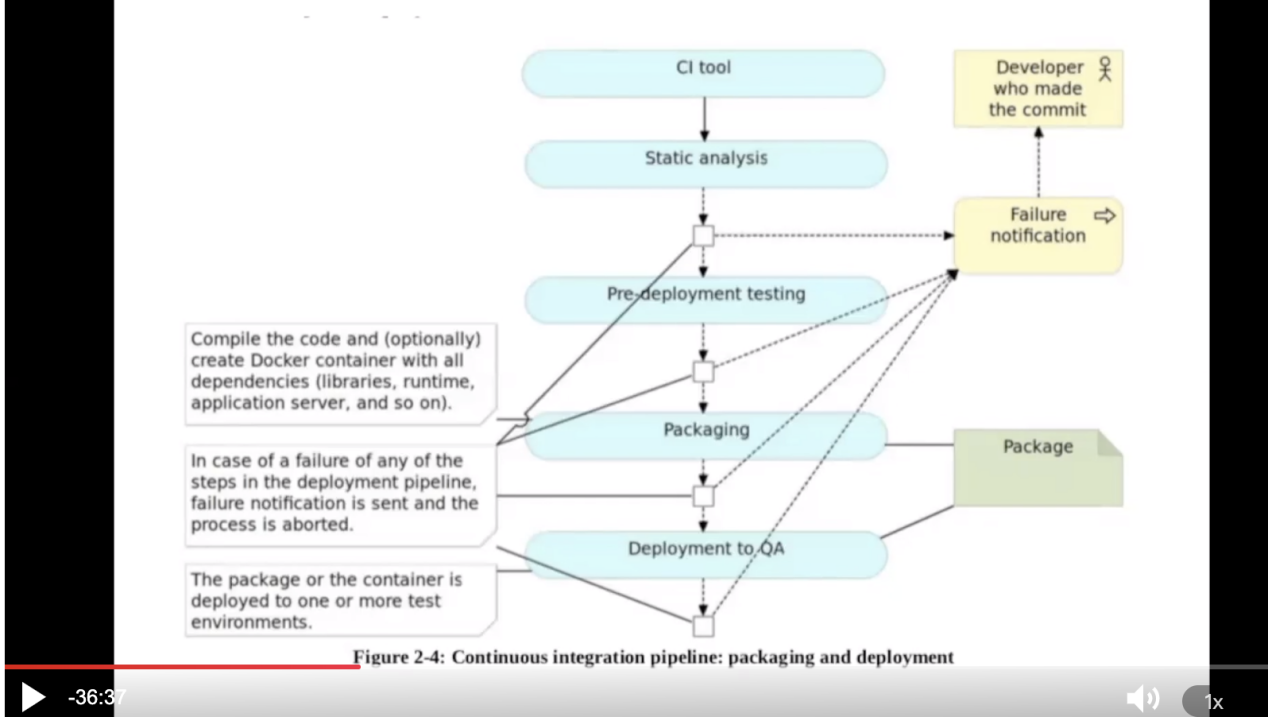
citool就是jekines

就是如果你有很多分支的话，那么当大家都开发完之后，就要把所有的分支合并到主干分支，但是这个最后的合并工作灰常困难，有了冲突会很难处理，CI就是每天都往主干里面合。

用来解决集成地狱



无非就是有一个citool，通常都是jeskains这个东西，你往代码库提交代码的时候，就被这个citool发现了，就会触发后面一系列流程，比如编译构建啊，如果通过了就通过了，如果没有通过就会给你发消息告诉你ci失败了，他会找到这个commit属于谁的。



Ci pinple的内容

静态代码检查 static analysis

部署前检查 pre deployment testing

打包 并且做成镜像 packaging

部署到测试环境，通知QA deployment QA

QA拿到后可能再做些手工验证

部署之后的测试，看看启动起来没有啊，基本业务可不可以啊 post-develoyment testing

经过上面的ci pipeline的操作，就开始把打出来的包分发到各个节点部署就是CD了。

## CD continuous delivery 持续交付

就是持续把我们那比CI好的版本发布给需要的人，让需要的人那里拿到的一直都最新的，这样的话，大家部署的版本就是可以统一的最新版本了。

用户解决发布噩梦

## CD continuous deployment 持续部署

就是发布出去了你要控制质量，发布到各个节点后要确保正确，所以还要对线上环境进行测试，保证发布质量。

## VUCA

需求的四个特点

易变 不确定 复杂 模糊性

软件研发很难找到best解决方式，只能找到一个还可以的。