Continuous Delivery(CD) is a software engineering approach in which teams keep producing valuable software in short cycles and ensure that the software can be reliably released at anytime (Chen,2015a).

Companies that practice CD have reported huge benefits ,such as significant improvements in time-to-market, customer satisfaction, product quality , release reliability , productivity and efficiency, and the ability to build the right product through rapid experiments (Chen,2015a;Leppanenetal.,2015)

Characteristic :-

1. Valuable software

Developing valuable software is a goal that has long been on the Agile manifesto(Becketal.,2001). However , it Is not an easy goal to achieve .Before adopting CD ,some of our teams had been using an Agile method called Kanban(Anderson,2010); however, due to delivery problems , we still had situations where a team had completed a feature but could not deliver it to production to obtain users’ feedback. Consequently , they built additional functionalities on top of that feature, simply assuming it was useful .Unfortunately, when they finally delivered the software to the users, they found out that the feature was not what the users needed.Even worse , by that point , significant effort had been spent on the feature and the additional functionalities. An important objective of our CD implementation was to alleviate this problem . We want teams to build valuable software rather than spend time on features that users do not need.

1. Short cycles
2. Releasable at anytime
3. Reliable releases

A term very similar to CD exists: Continuous Deployment .In

Academia ,people tend to use it interchangeably with Continuous Delivery(Rodríguezetal.,2016).

However , many practitioners tend to clearly distinguish these two terms. The distinction is that under Continuous Deployment

We deploy any change to production that passes a series of tests.

In contrast, under Continuous Delivery , we ensure that the software can be reliably released at anytime, but it is up to a human to decide when to release. In both Continuous Deployment

And Continuous Delivery , deployment to production itself is automated .The difference lies in the trigger for making the deployment . One is triggered automatically , while the other is triggered

By a human .According to this distinction , Continuous Delivery is

Compatible with a wide range of scenarios , but Continuous Deployment is suitable only under special conditions(O’Dell and Skelton,2016). As do most companies ,we mainly use Continuous Delivery.

Challenge of making the full coverage test , prove the quality of the code , pipelines abort notify the developer

If the updated code passes all the stages,it can be released to production with the click of a button(Chen,2015a).