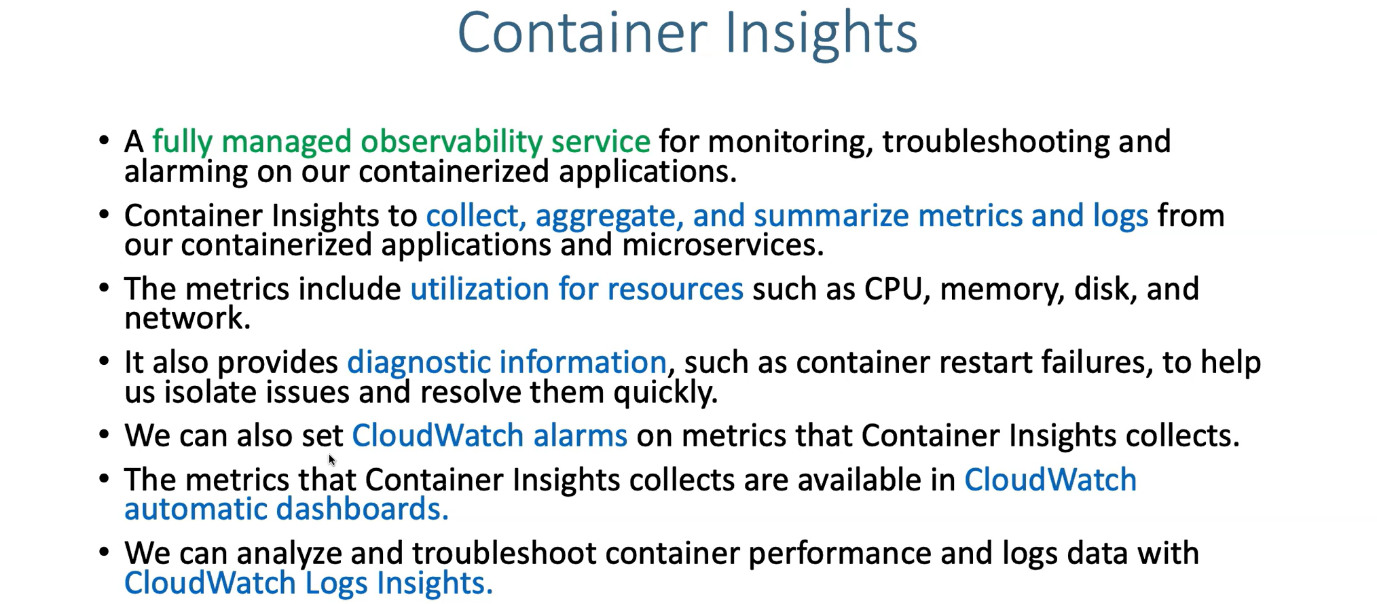
**01. CloudWatch Container Insights - Introduction**

--- We are going to look into cloud watch container insights feature as part of this section.

--- So, we are going to look into many clouds watch features and also a simple notification service as part of this.

--- what is this container insight and then what it is going to do?

**Container Insights**



--- Container Insights is a fully managed observability service for monitoring, troubleshooting and alarming on our containerized applications, container insights allow us to collect, aggregate and summarize metrics and logs from our containerised applications and also for micro services.

--- the metrics include utilization for resources such as C.P.U, memory disk and then network.

--- It also provides diagnostic information such as container restart failures so that you can consider

as one metric to help us isolate issues and resolve them quickly.

--- so, we can also set cloud watch alarms on metrics that container insights collects and the metrics that container insights collects are available in cloudwatch automatic dashboards.

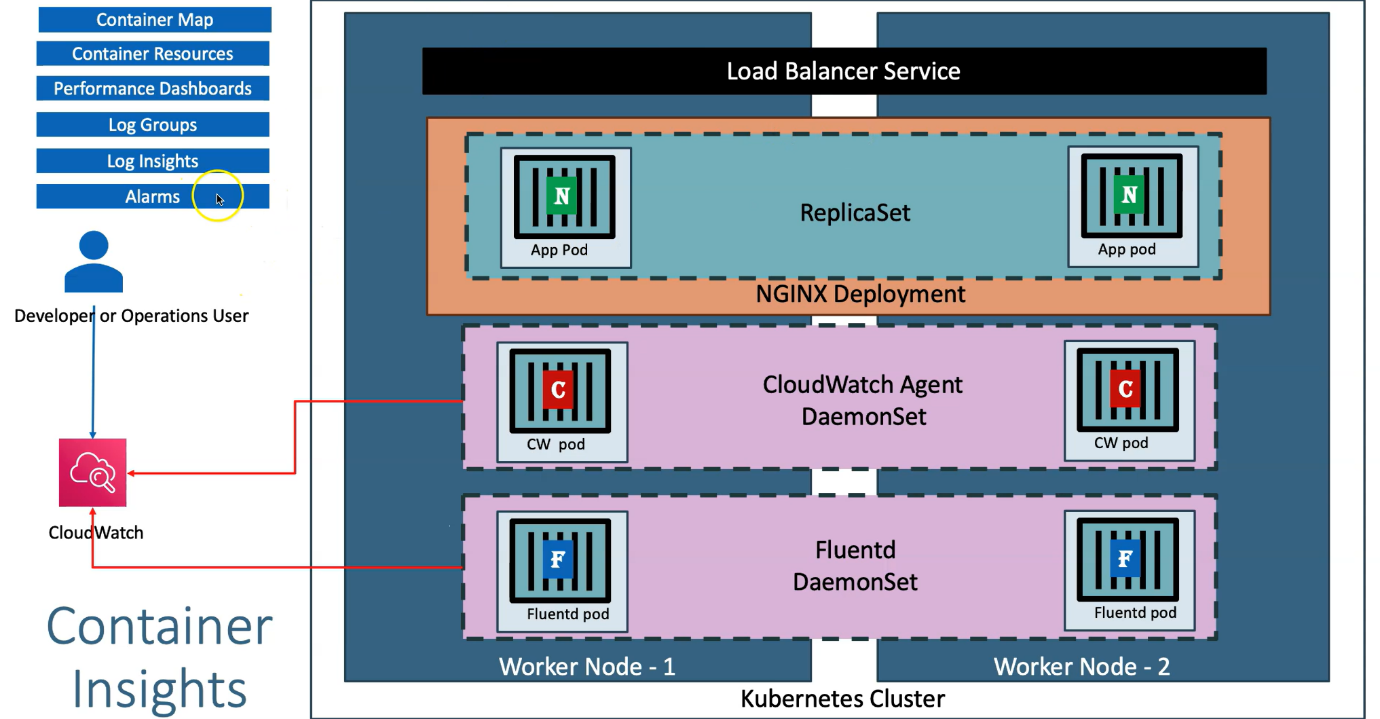
--- we can analyse and troubleshoot container performance and logs data with cloud watch log insights.

--- if you see from cloud watch perspective. on a very high level, it has covered a maximum ground, even the newest and latest container insights feature.

--- you can configure cloud watch alarms inside container insights and you can also have cloud watch

automatic dashboards for the metrics information and you can also have container insights is collecting the logs, using logs we can create custom dashboards for us.

**Container insights (what we are going to deploy)**



--- in kubernetes cluster, let's consider you have 2 worker nodes. Worker node -1 and worker node-2.

--- you're going to first deploy a fluentd deamonset. we already discussed about what is deamonset and what are the features it provides during our aws x-ray section.

--- Deamonset in very simple terms is whenever you configure your application as a deamonset, it

is going to deploy one pod per worker node.

--- now we have the fluentd pod is run and running as a deamonset in worker node-1 and worker node-2

--- what is this fluentd pod going to do?

--- it is going to collect the application logs, which is nothing but the container logs and also the

kubernetes cluster logs, application logs, performance logs and all these things. it is going to push to the cloud watch.

--- We are going to analyse them and then create our custom dashboard screens.

--- next thing is cloudwatch Agent deamonset. what is this going to do?

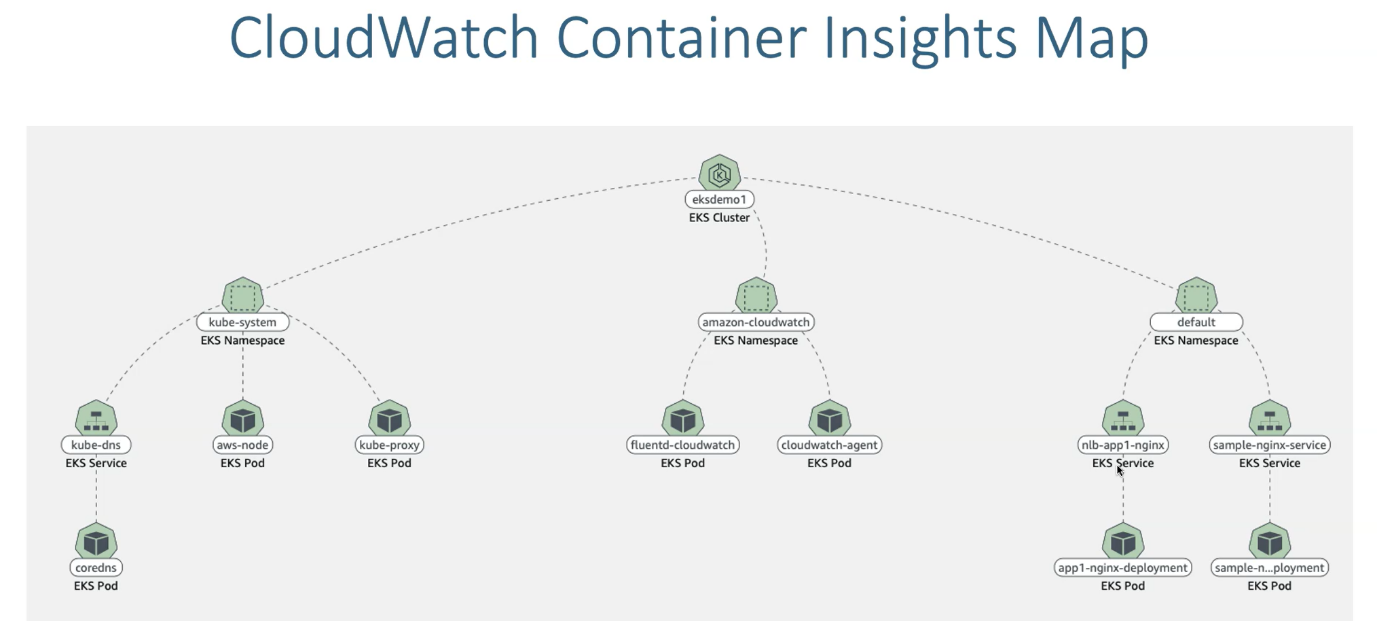
--- one more thing, we are going to deploy. this one will collect the metrics information of our applications, cluster, worker nodes, EC2 instances everything. 134 types of cloudwatch container insights metrics that is going to get collected as part of this cloudwatch agent.

--- next is, if we deploy our application on this cluster. when these things are running, they are going to collect the necessary information and then push it to the cloudwatch service. now the information is there in cloudwatch service.

--- who need to use it…? Either developer or operations user can go to cloudwatch management console and then start looking and **analysing** the data.

--- The core feature is, it provides a container map. So, it is going to be a very nice future. it is going to give a complete picture of all the objects present in our kubernetes cluster like pods, services, nodes, everything.

**Cloudwatch container insights map**



--- So, if you see here, eksdemo1, which is nothing but a cluster.

**Automatic performance dashboards**

