A open source way:
From Cloud Native to Edge Native

Zefeng Wang, Huawei
Zhengguang Ou, Huawei
Agenda

• What is the challenge of Edge?

• Why we need Cloud Native in Edge?

• What is CNCF KubeEdge?

• How Cooperate CNCF and Eclipse in Edge?

• Q&A
# What is the challenges of Edge?

## Edge Computing Challenges:

<table>
<thead>
<tr>
<th>App &amp; App mngt</th>
<th>Network Connection</th>
<th>Deployment &amp;Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Returning from cow to pet</td>
<td>Huge edge node offline or online Simultaneously</td>
<td>High deployment cost</td>
</tr>
<tr>
<td>Strong role-based for edge node</td>
<td>Unstable network</td>
<td>Requirements of good IT skills</td>
</tr>
<tr>
<td>Requirement of lightweight</td>
<td>The network topology is complex and needs to be layered.</td>
<td>Bad environment</td>
</tr>
<tr>
<td>Half-state application</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0-trusted principle</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- High deployment cost
- Requirements of good IT skills
- Bad environment
What is Cloud Native?

**Core Components**
- Physical Service ➔ Virtual Machine ➔ Buildpacks ➔ Container

**Immutability**
- From pet to cow

**Isolation**
- Lightweight, and more faster boot

**Vendor**
- From a single closed-source vendor to multiple open-source vendors
Why we need Cloud Native in Edge?

<table>
<thead>
<tr>
<th>App Lifecycle</th>
<th>Embedded Software Edge Today</th>
<th>Cloud Native Approach Edge Tomorrow</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Design</strong></td>
<td>Hardware Dependent</td>
<td>Platform Independent</td>
</tr>
<tr>
<td></td>
<td>Waterfall (gate based)</td>
<td>Agile (iterative)</td>
</tr>
<tr>
<td></td>
<td>Monolithic</td>
<td></td>
</tr>
<tr>
<td><strong>Develop</strong></td>
<td>Reinvent wheel (eg many RTOSes)</td>
<td>Re-use centric</td>
</tr>
<tr>
<td></td>
<td>Legacy programming models (C, ASM)</td>
<td>Modern small footprint programming (Go, Node.js)</td>
</tr>
<tr>
<td></td>
<td>Proprietary, expensive dev tools &amp; closed-source</td>
<td>Open source, open development</td>
</tr>
<tr>
<td><strong>Deploy</strong></td>
<td>&quot;Burn-in&quot; a the factory with testing prior to shipping</td>
<td>Instant app deployment when and where needed; Zero-touch full stack deployment</td>
</tr>
<tr>
<td></td>
<td>Update in field as little as possible</td>
<td>As many times as needed at the moment it is available</td>
</tr>
<tr>
<td></td>
<td>High risk for upgrades, one way, Manual Disaster Recovery</td>
<td>Roll-back ability; Automated verification and disaster recovery</td>
</tr>
<tr>
<td><strong>Operate</strong></td>
<td>One at a time management</td>
<td>At scale management with &quot;intended state&quot; model</td>
</tr>
<tr>
<td></td>
<td>Local focused (console/serial/shell)</td>
<td>Network/remote focused, SD-WAN overlay integrated with apps</td>
</tr>
<tr>
<td></td>
<td>Insecure due to being air-gapped (not connected) historically</td>
<td>&quot;Ground to Cloud&quot; defense in layers with PKI cryptography;</td>
</tr>
<tr>
<td><strong>Analyze</strong></td>
<td>Not easy to get statistics/diags out of device</td>
<td>Continuous statistics, diags and analysis</td>
</tr>
<tr>
<td></td>
<td>Distributed and &quot;ships in the night&quot;</td>
<td>Centralized and aggregated</td>
</tr>
<tr>
<td></td>
<td>No feedback loop / lack of automated analytics</td>
<td>Machine learning and Insight driven</td>
</tr>
</tbody>
</table>
Building Edge platform With Kubernetes

• **Gaps**
  - Limited resource at Edge
    • Not enough for vanilla K8s, even just a Kubelet.
  - Unstable network
    • Private network, limited bandwidth, latency, etc.
  - Need autonomy at Edge
    • Edge may get offline/disconnected often
    • Should not evict/migrate applications when disconnected
  - Device management at Edge
    • Quite different requirement with current device plugin

• **Benefits**
  - Containerized Application
    • Build once, run anywhere
    • Lightweight base image
  - General application abstraction
    • Already become standard
    • Same experience across cloud and edge
  - Extendable Architecture
    • Extendable API machinery
    • Easy to add customized components
What is KubeEdge?

- **KubeEdge** is built upon Kubernetes and provides fundamental infrastructure support for network, application deployment and metadata synchronization between cloud and edge.

- **KubeEdge provides**
  - **Seamless Cloud-Edge Communication** for both metadata and data
  - **Edge Autonomy**: Autonomous operation of Edge even during disconnection from cloud.
  - **Low Resource Ready**: KubeEdge can work in constrained resource situations (low memory, low bandwidth, low compute)
  - **Simplified Device Communication**: Easy communication between application and devices for IOT and IIOT

- Entered CNCF sandbox in March 2019
- Stars: 400+, Forks: **600+**, member organizations: **25+**
- Cross community collaboration
  - K8s iot edge WG
  - Akraino
    - ELIOT BP Family
    - KubeEdge Edge Service BP
KubeEdge Architecture – Cloud

- **EdgeController**
  - Shadow management for nodes, applications at edge
- **Device API/DeviceController**
  - IOT/Edge device modeling
  - Shadow management for devices at edge
- **CSI Driver**
  - Hook storage provisioning etc. to edge
- **Admission Webhook**
  - Extended API validation
  - Best practice enforcement
KubeEdge Architecture – Edge

- **EdgeHub**
  - Messaging over WebSocket
- **MetaManager**
  - Node level metadata persistence
- **Edged**
  - Kubelet-lite
  - Pod lifecycle management etc.
- **DeviceTwin**
  - Sync device status between cloud, edge node and device
- **EventBus**
  - MQTT client
- **ServiceBus**
  - HTTP client
How Collaboration CNCF KubeEdge and Eclipse in Cloud2Edge

CloudHub

- kubectl
- k8s

EdgeHub

- WebSocket (Default)
- QUIC (Alternate)

On cloud

At edge

Devices

Data Cleaning

mosquitto

Video analysis (AI Inference)

Container (docker/containerd/runc)

X86/ARM/GPU/NPU

Kubelet

ditto

hawkBit

neoSCADA

AI Training

3-party APP

Container

Device 1

Device 2

Device 3

Device 4
KubeEdge Resources

- Website: https://kubeedge.io
- Github: https://github.com/kubeedge/
- Slack channel: https://kubeedge.slack.com
- Mailing List: https://groups.google.com/forum/#!forum/kubeedge
- Bi-weekly community meeting: https://zoom.us/my/kubeedge
- Twitter: https://twitter.com/KubeEdge
- Documentation: https://docs.kubeedge.io/en/latest/
Thanks