## Akraino Feature Project – Snappy (Backup and Restore)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Companies Participating / Committers</th>
<th>Requested Release / Timeline</th>
<th>Informational</th>
</tr>
</thead>
<tbody>
<tr>
<td>Snappy – Backup and Restore</td>
<td>An extensible backup and restore framework for use with cloud storage systems. For use with multiple blueprints.</td>
<td>AT&amp;T</td>
<td>R2 for Telco Appliance</td>
<td>Initial Blueprints that will use Snappy: Telco Appliance, Network Cloud</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>See next slides for additional details</td>
</tr>
</tbody>
</table>

AT&T Committers: Pingkai Liu (pl869j@att.com)  
Josh Auzins (ja7858@att.com)  
Don Henderson (dh2682@att.com)
Why is Snappy needed?

The backup/restore challenge does not automatically disappear in the cloud. In fact, it is even more challenging!

- **Heterogeneous storage backend requires backup/restore system to be easily extendable**
- **Virtualization poses a challenge because of the difficulty to talk to storage I/O directly**
- **Backup/Restore Admin has much less knowledge and control on tenant self hosted applications**
Snappy: 3 Main Components

• **Frontend**: receives jobs
  • Interface to the outside world

• **Database**: stores jobs
  • Past, current and future

• **Core**: runs jobs
  • Plug-ins for different types of sources and targets

“job” = backup or restore
How Snappy Can Help

Heterogeneous storage backend requires backup/restore system to be easily extendable

Plug-ins can be added for new storage systems

Virtualization poses a challenge because of the difficulty to talk to storage I/O directly

Required parameters for virtualization layer specified in setup

Backup/Restore Admin has much less knowledge and control on tenant self hosted applications

Snappy provides flexible ways of doing backups (both agentless and agent based)
Snappy Implementation – Kubernetes Example

Each broker runs:
- Exchange core
- Plug-ins

Backup Sources:
- LVM/SCSI
- Ceph Block Pool
- Pure All-Flash

Backup Targets:
- GlusterFS
- S3 Object Storage
- Openstack Swift