



Maximizing your HPC cluster investment

Cray User Group

John Corne, Pre-Sales Engineer
23-May-2018

Agenda

- Bright Computing
- Bright Cluster Manager
 - What's new in 8.1
 - Workload Accounting and Reporting
- Bright for Data Science



Company Profile



We launched our software in 2009 because managing clustered IT infrastructure was **hard** and we thought it should be **easy**.

Nine years later, we are deployed in hundreds of IT environments worldwide, including several of the world's largest supercomputers.

About Bright & Cray

- Long history between Cray and Bright
 - Between ~2010 - ~2016: Bright used on cluster alongside XC systems for login nodes, storage nodes, data mover nodes
 - Since 2017: Bright standard on all CS systems
- The largest active Bright cluster is a Cray CS500 (almost 8500 nodes)
- Ambition still exists to expand from CS to XC series

Agenda

- Bright Computing
- **Bright Cluster Manager**
 - What's new in 8.1
 - Workload Accounting and Reporting
- Bright for Data Science

Bright Cluster Manager
saves time and money by
making it easy to deploy
and manage Linux clusters



Bright makes it easy to ...

Deploy

Compute, data,
storage clusters

Manage

Users, clusters,
and clouds

Monitor

From a single
pane of glass

... with a powerful, integrated, and intuitive
platform.



Easy to Deploy



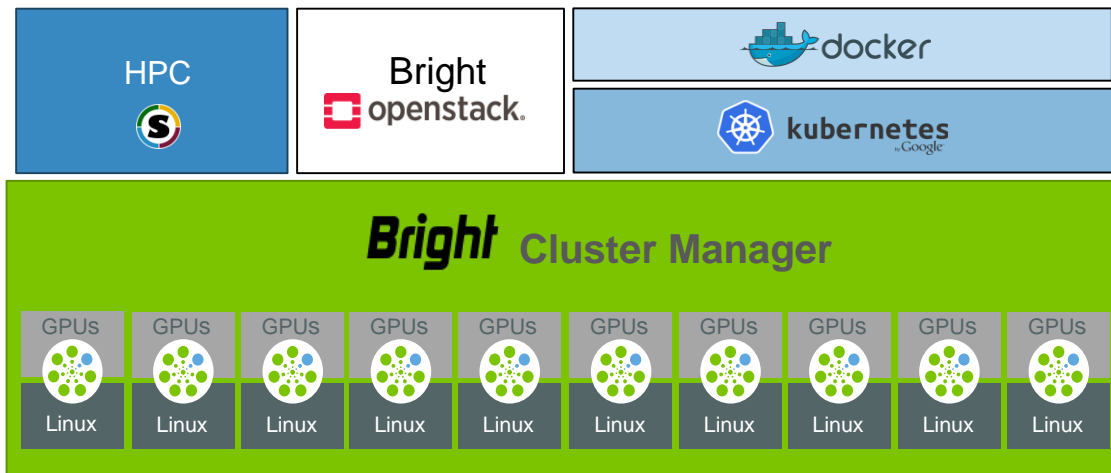
The screenshot shows the 'Bright Cluster Manager Installer' window. The title bar includes the product name and a language dropdown set to 'English(US)'. The main window title is 'Welcome to the Bright Cluster Manager Installer'. On the left is a navigation pane with a list of steps: Welcome (selected), License, Kernel Modules, Hardware Info, Nodes, Network Topology, Additional Networks, Networks, Nameservers, Network Interfaces, Subnet Managers, Installation Source, Workload Management, Disk Layout, Time Configuration, Cluster Access, Authentication, Console, and Summary. The main content area features a blue header with the 'Bright Cluster Manager ADVANCED EDITION' logo. Below this, the 'License Information' section displays the following details:

Version	8.2
Edition	Advanced
Name	Bright 8.2 Cluster
Organization	JCM
Unit	Development
Locality	San Jose
State	California
Country	US
Serial	6700
Valid from	17 May 2015
Valid until	08 Jan 2038
MAC address	??-??-??-??-??-??
Licensed nodes	3

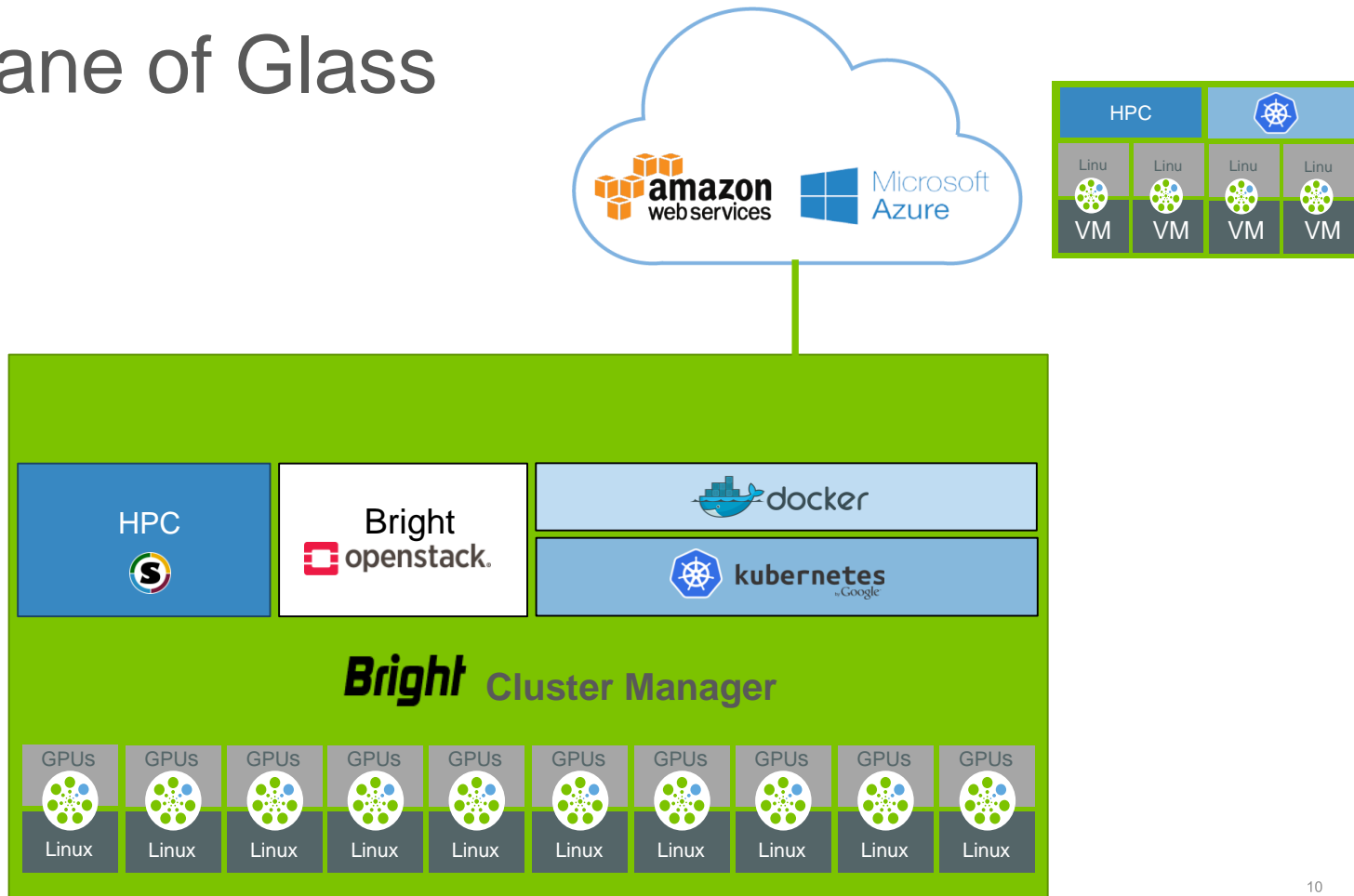
The 'Installation mode' section below has two radio buttons: 'Normal (recommended)' (which is selected) and 'Express'. At the bottom of the window are three buttons: 'Remote Installation', 'Cancel', and 'Continue'.



Flexible and Easy to Extend



Single Pane of Glass



Easy to Manage

The screenshot displays the Bright View interface for managing Physical Nodes. The main table lists nodes with the following columns: HOSTNAME, STATE, MAC, CATEGORY, IP, NETWORK, and OPTIONS. The 'OPTIONS' column for 'node01' is circled in red, and a red callout box highlights the 'Edit' button and the context menu that appears when it is clicked.

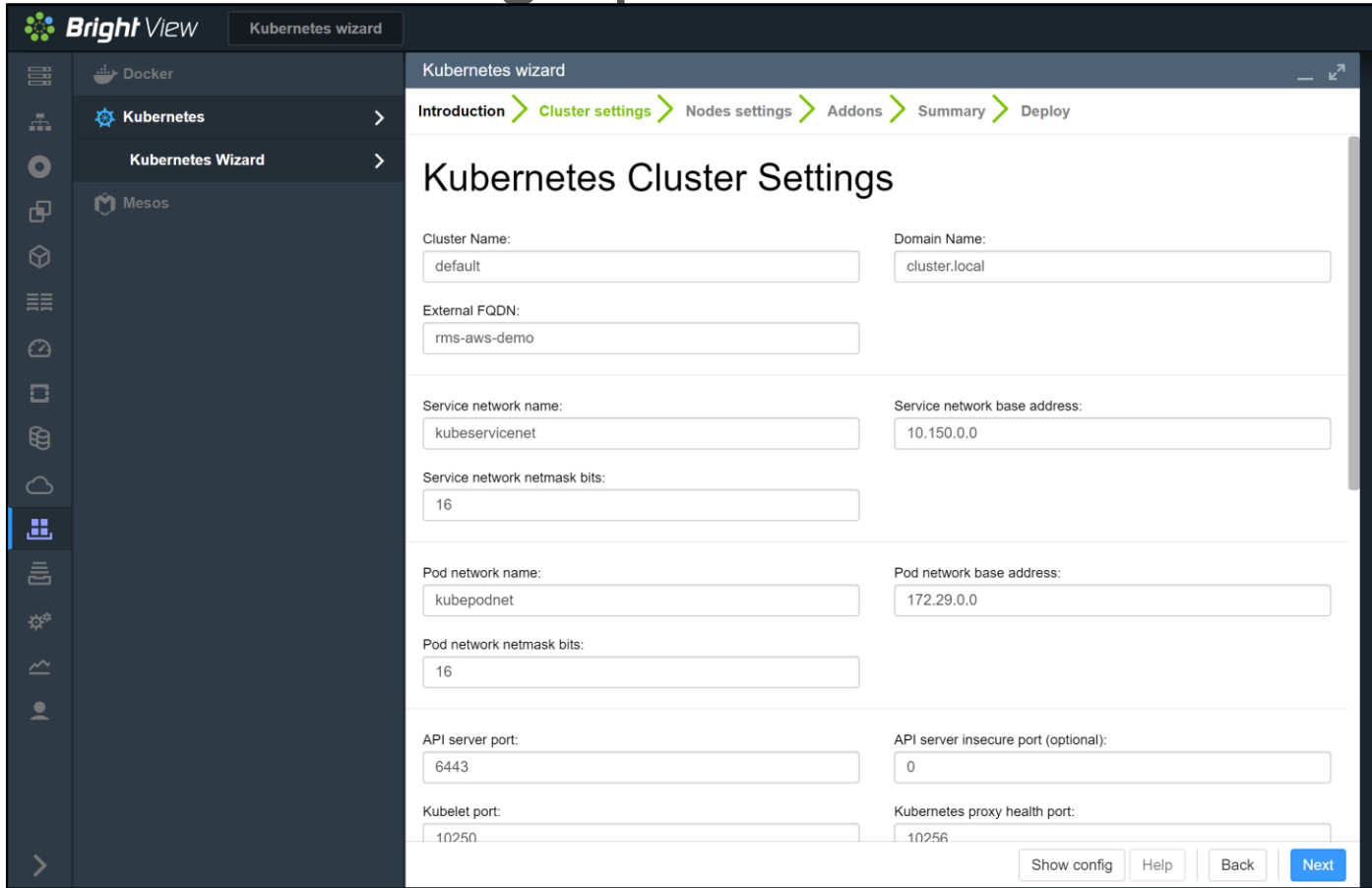
HOSTNAME	STATE	MAC	CATEGORY	IP	NETWORK	OPTIONS
<input checked="" type="checkbox"/> node01	↓	00:00:00:00:00:00	default	10.12.0.1	internalnet	Edit
<input type="checkbox"/> node02	↓	00:00:00:00:00:00	default	10.12.0.2	internalnet	Edit
<input type="checkbox"/> node03	↓	00:00:00:00:00:00	default	10.12.0.3	internalnet	Edit
<input type="checkbox"/> node04	↓	00:00:00:00:00:00	default	10.12.0.4	internalnet	Edit

The context menu for the 'Edit' button includes the following options:

- Edit
- Revert
- Clone
- Delete
- Power
 - On
 - Off
 - Reset
- OS
- Workload
- Software image



Wizards for setting up



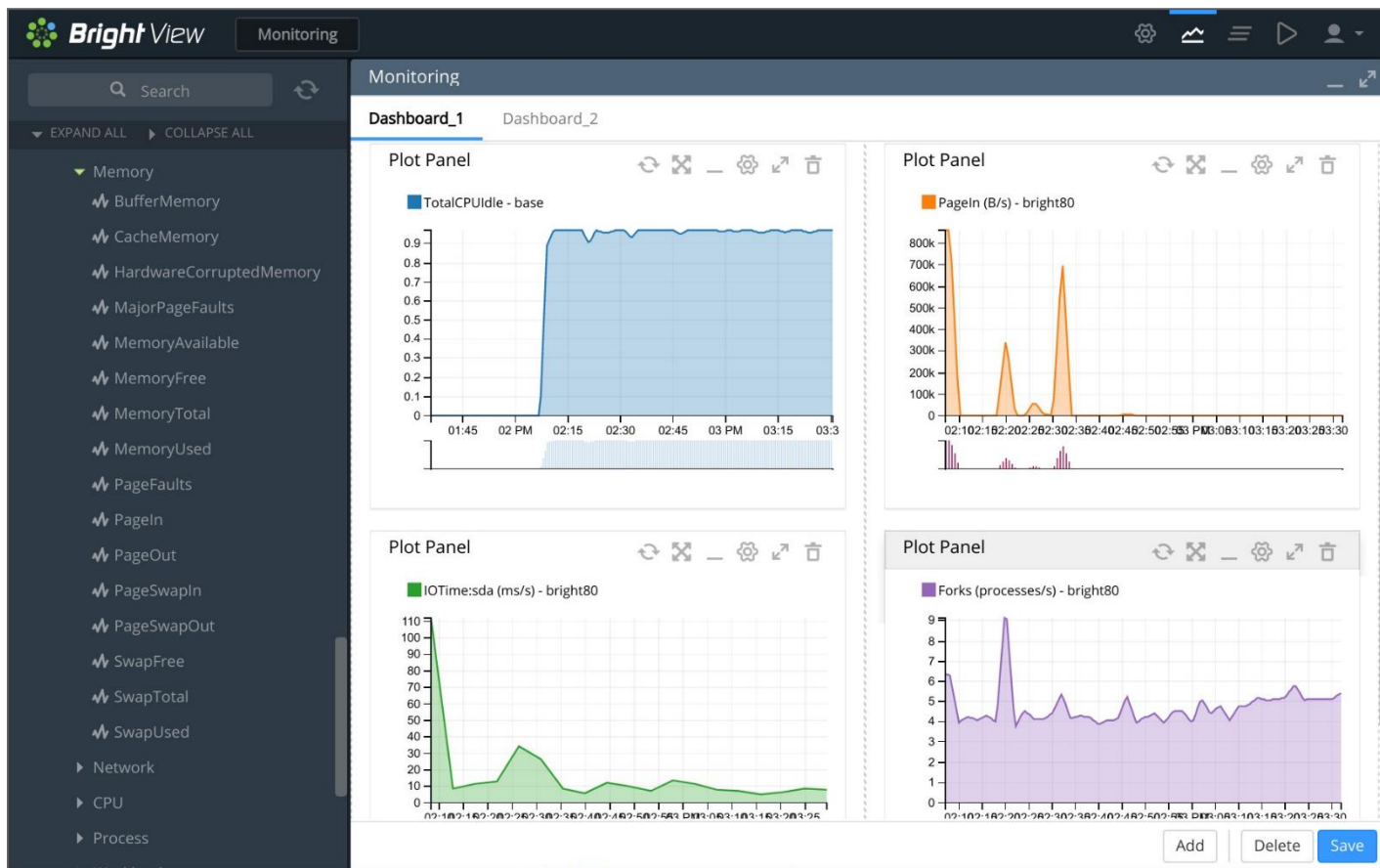
The screenshot displays the 'Kubernetes wizard' interface in BrightView. The left sidebar contains navigation options for Docker, Kubernetes, Kubernetes Wizard, and Mesos. The main content area is titled 'Kubernetes Cluster Settings' and features a breadcrumb trail: Introduction > Cluster settings > Nodes settings > Addons > Summary > Deploy. The settings are organized into several sections, each with a label and an input field:

- Cluster Name:** default
- Domain Name:** cluster.local
- External FQDN:** rms-aws-demo
- Service network name:** kubeservicenet
- Service network base address:** 10.150.0.0
- Service network netmask bits:** 16
- Pod network name:** kubepodnet
- Pod network base address:** 172.29.0.0
- Pod network netmask bits:** 16
- API server port:** 6443
- API server insecure port (optional):** 0
- Kubelet port:** 10250
- Kubernetes proxy health port:** 10256

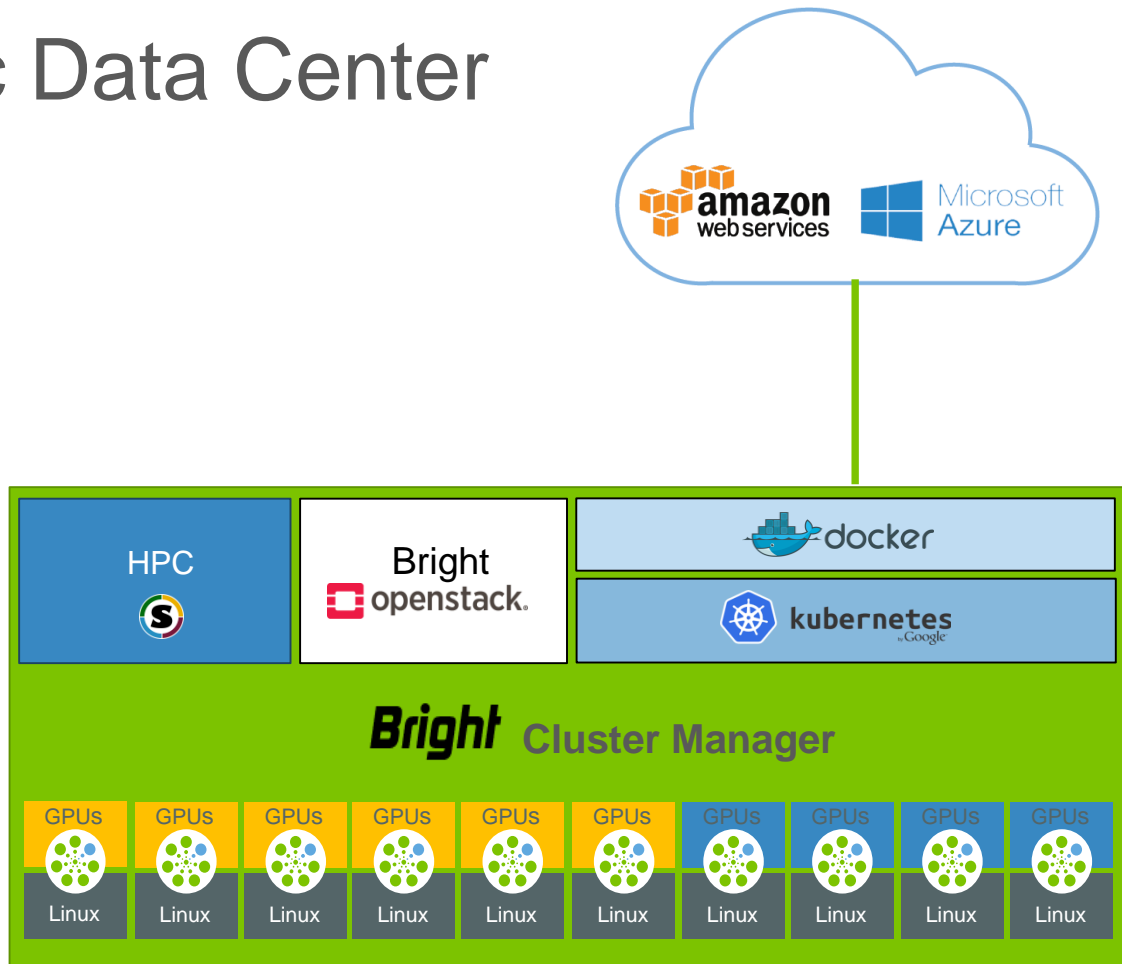
At the bottom right, there are four buttons: 'Show config', 'Help', 'Back', and 'Next'.



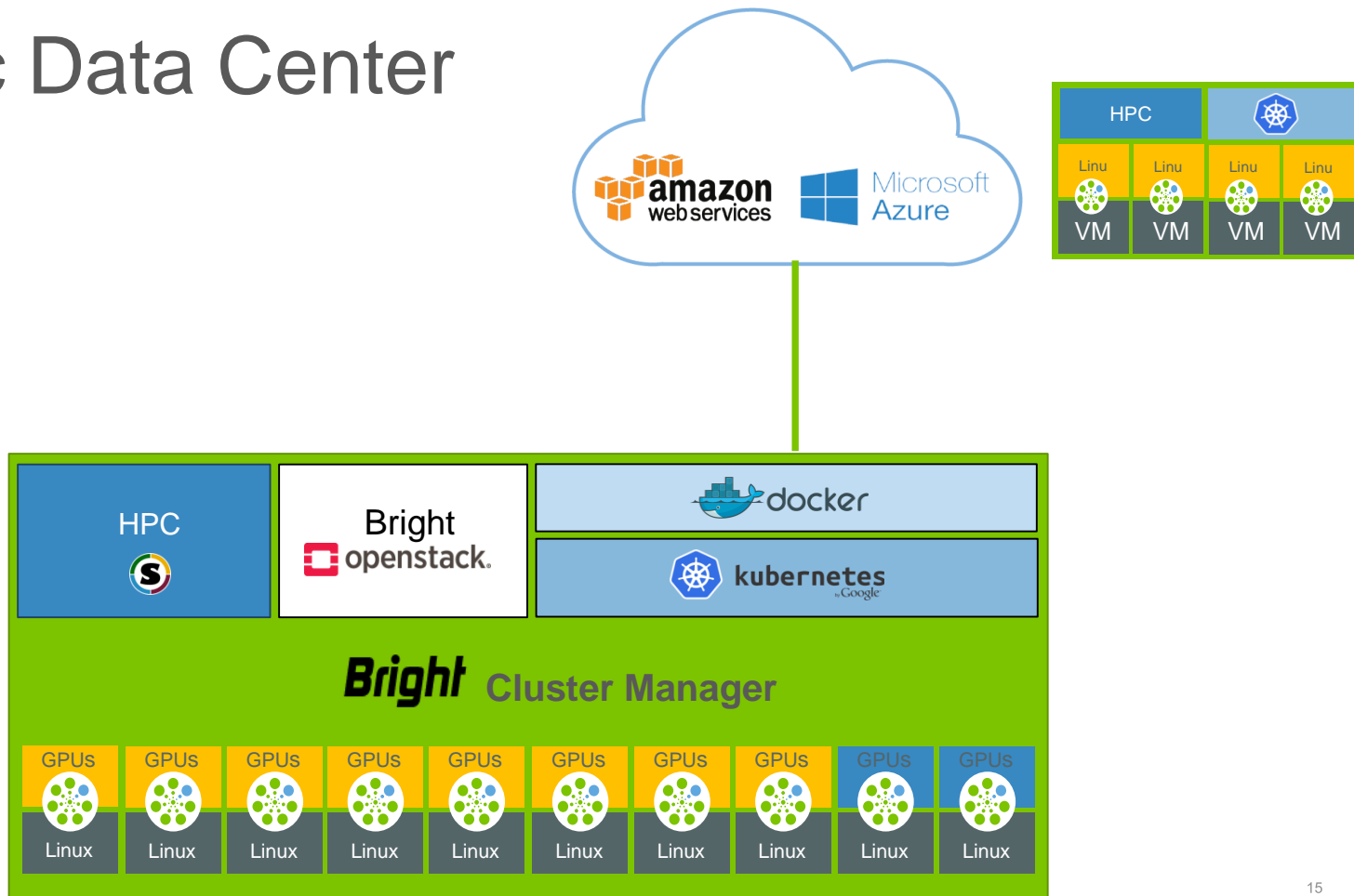
Easy to Monitor



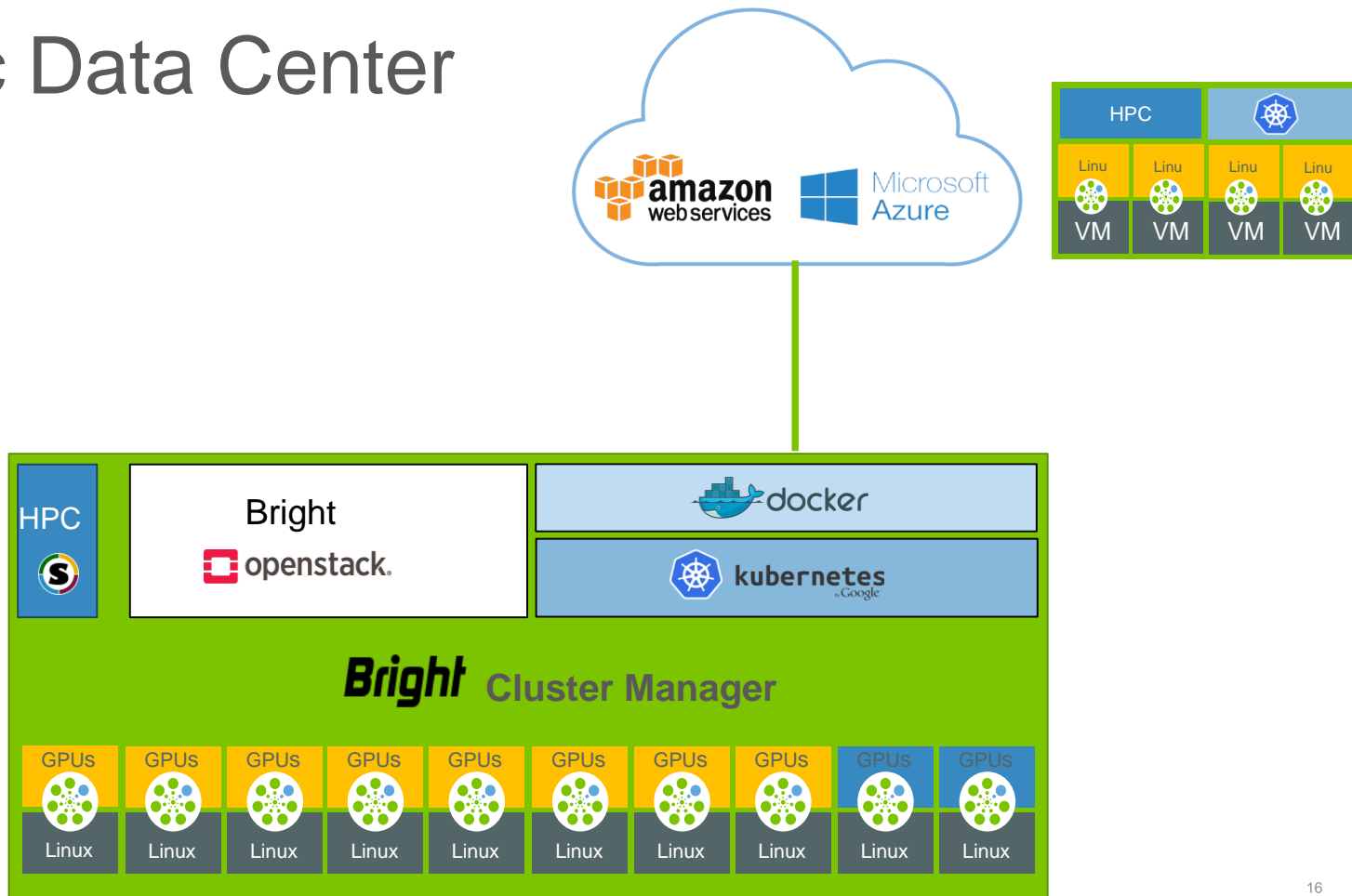
Dynamic Data Center



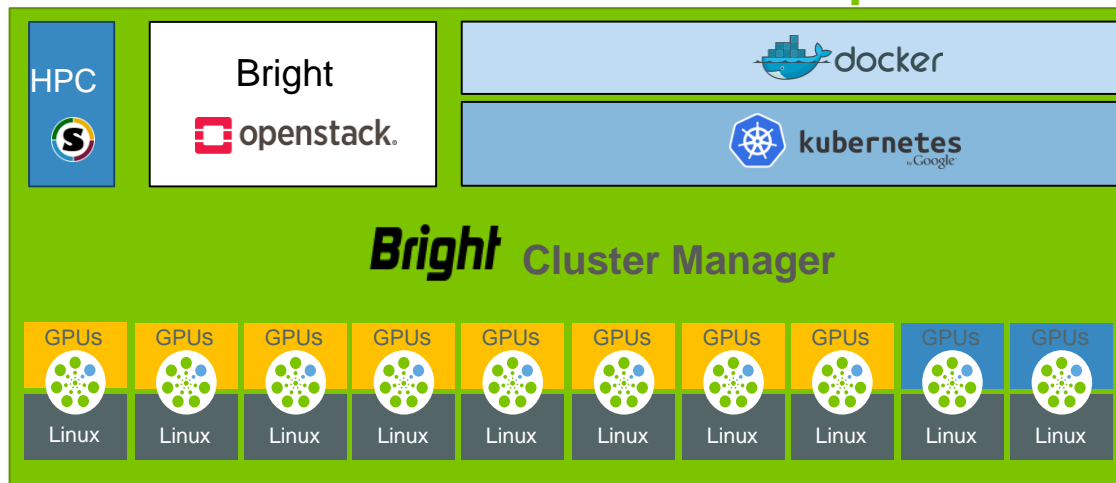
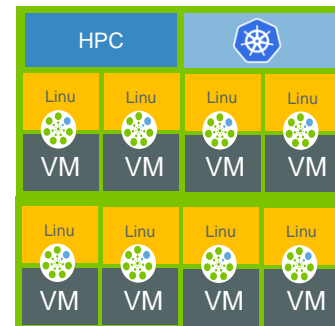
Dynamic Data Center



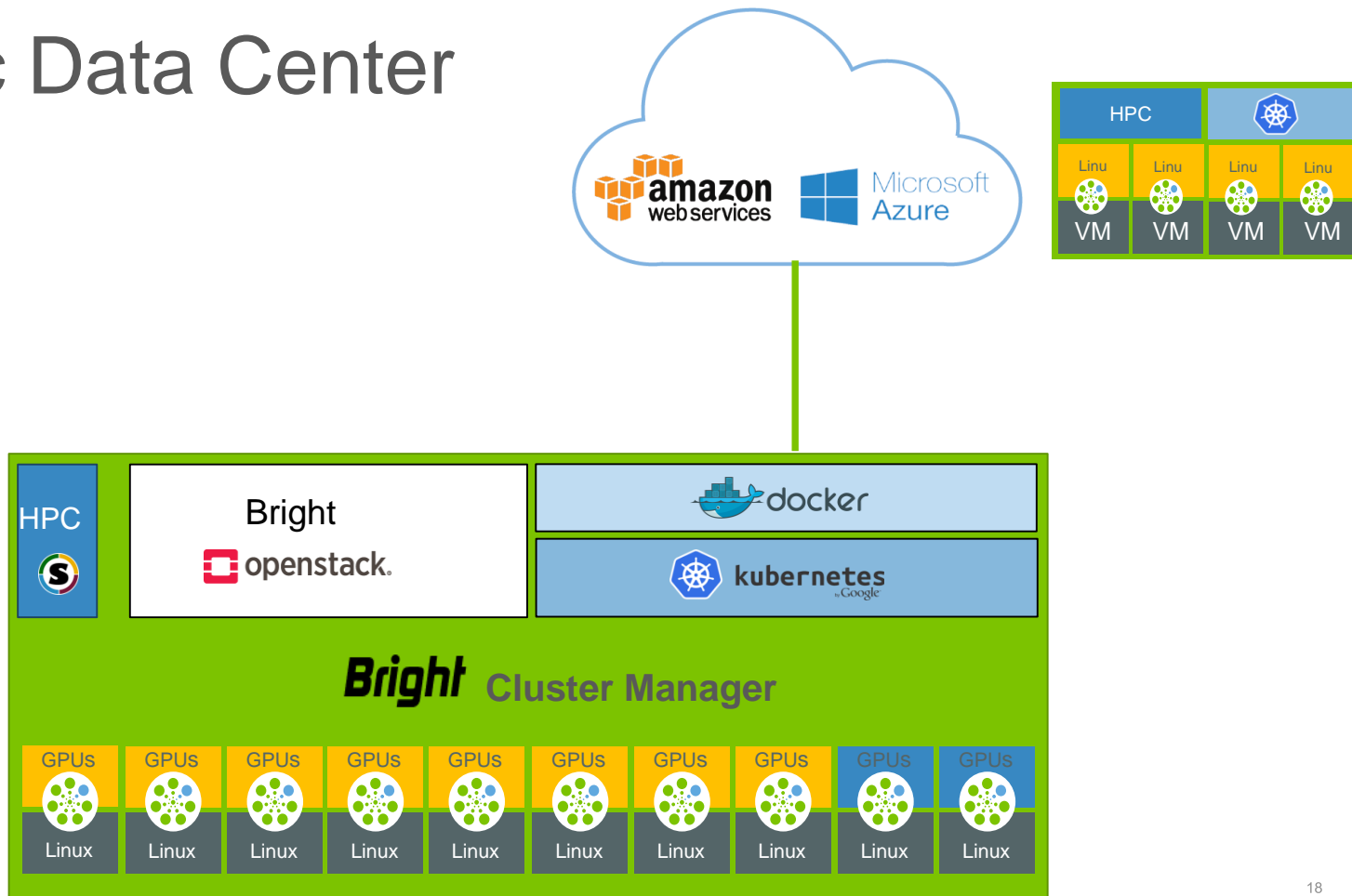
Dynamic Data Center



Dynamic Data Center



Dynamic Data Center



Includes.....

- Integration with WorkLoad Managers
 - SLURM, PBS Pro, Torque, Moab, Maui, Univa Grid Engine etc
- GPU Support
 - NVIDIA
 - AMD
 - Integration with WorkLoad Managers
- CUDA and OpenCL libraries
- Hundreds of popular HPC libraries



Agenda

- Bright Computing
- Bright Cluster Manager
 - What's new in 8.1
 - Workload Accounting and Reporting
- Bright for Data Science



What's new in 8.1

- Workload management accounting & reporting
 - Job based metrics
- Bursting to OpenStack
 - Bright OpenStack supported
 - Select OpenStack public clouds supported
 - And still AWS and Azure
- OpenStack Pike integration
- AMD GPU support
 - GPU settings, metrics, healthchecks, clean software stack deployment



Smaller 8.1 Features and Changes

- Lightweight CMDaemon
 - Implements monitoring API
 - Portable (100% Python code; Python 2.7.5+ required)
- Kubernetes 1.9.2
 - Ncurses & Bright View wizard
- Docker 1.12.6
 - Ncurses & Bright View wizard
- Ceph Luminous
 - Ncurses & Bright View wizard



Agenda

- Bright Computing
- Bright Cluster Manager
 - What's new in 8.1
 - Workload Accounting and Reporting
- Bright for Data Science



What do we want to know?

- Who is using the resources?
- Who is using them poorly?
- How was the system behaving in a particular moment in the past?
- Are all the components of the system being used?
 - Are the jobs CPU bound or I/O bound?
 - Do we have enough network bandwidth?
 - What is the utilization on GPUs?



How do we do it?

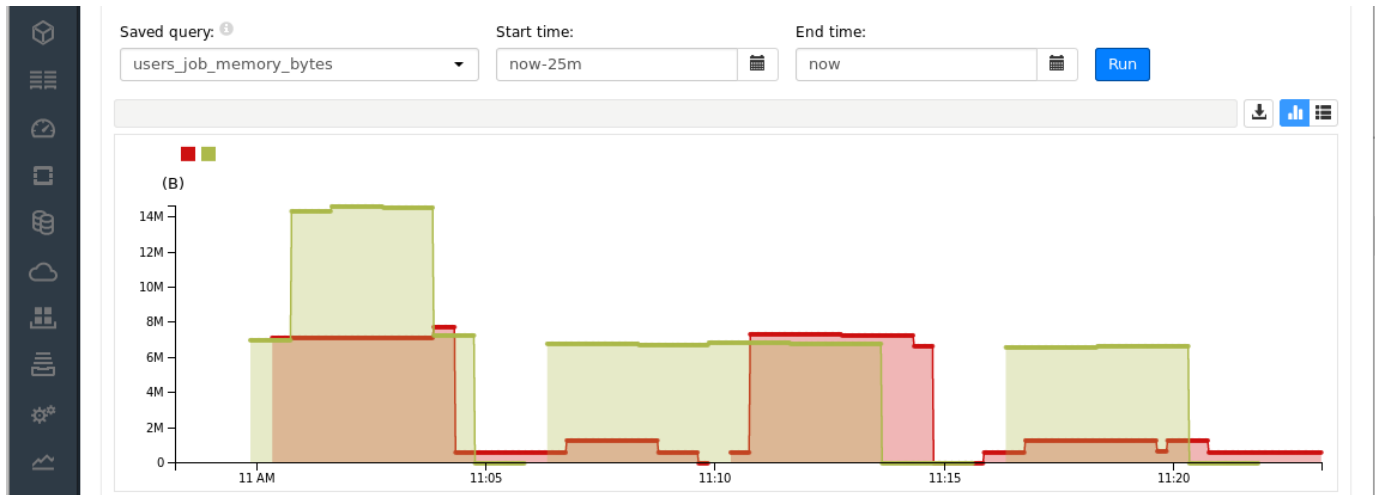
- PromQL is a functional expression language
- Allows to select and aggregate time series data in real time
- Features
 - Labels selection
 - Arithmetic and comparison operators
 - Aggregation
 - Joins
 - Statistical functions
 - Sorting
 - etc...



Examples (I)

Memory usage by users

- Aggregate memory metrics by user
- Plot them over a period of time



Examples (II)

Current jobs' waiting time

job_id	job_name	user	group	job_waiting_time
7	pi	bob	dev	69034 s
6	my_mpi_job	mike	ds	360 s
15	pi	bob	dev	10 s

- Single metric
- Take the last value
- Sorting
- Show them in a table



Examples (III)

CPU wall clock time used over the last week by account

account	account_cpu_time
Drilling	15300 CPU s
Seismic	360 CPU s
Modeling	369034 CPU s

- Aggregating over time
- Grouping by account



Examples (IV)

Power consumption of Bob's jobs over the last week

power_usage

231 kWh

- Aggregation over time
- Filtering by a particular user
- Single number as a result



Agenda

- Bright Computing
- Bright Cluster Manager
 - What's new in 8.1
 - Workload Accounting and Reporting
- **Bright for Data Science**



Bright for Data Science
makes it easy to use a
Bright cluster for AI



Without Bright

- Not installable from OS repos
- Time-consuming, manual installation of deep learning libraries and frameworks
- 60+ top-level dependencies must be satisfied
- Versions must work together

This [solution] will be a powerful productivity multiplier for customers because these software modules take days to download and install if using the open source repositories.

– a Bright user



With Bright

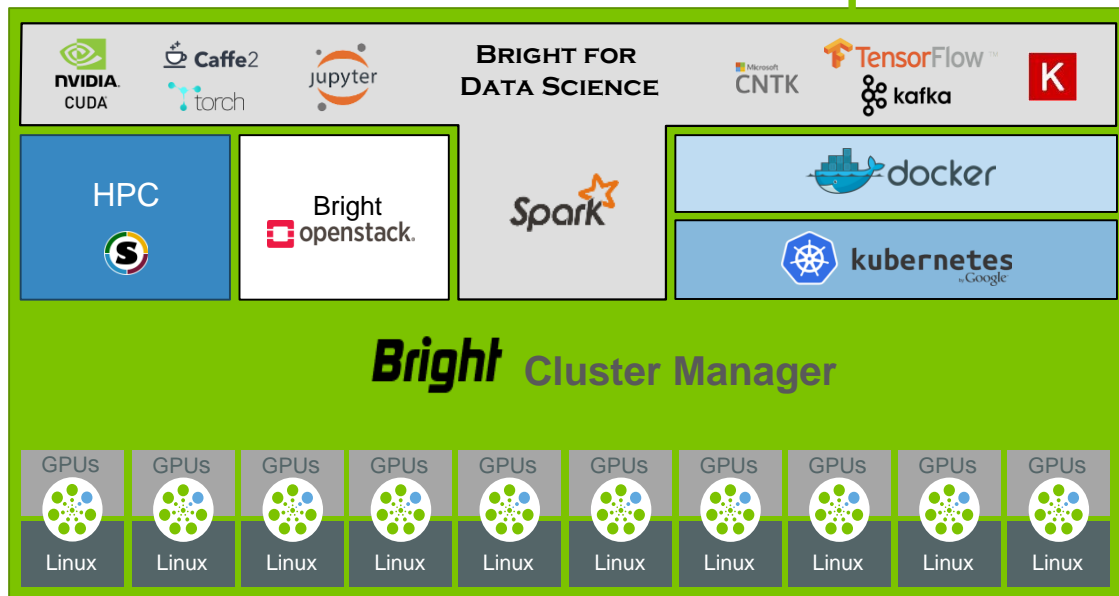
```
# yum install tensorflow cm-jupyterhub
```

```
# yum --installroot=/cm/images/ai-image \  
install cm-ml-distdeps
```

- 1st command installs frameworks into a shared directory on the head node. It is immediately available on every node in cluster.
- Yum installs all dependencies for tensorflow and cm-jupyterhub, and all the Python dependencies
- 2nd command installs all library dependencies into ai-image



Bright for Data Science



DS Ecosystem



Natural Language Analysis with Python NLTK



OpenBLAS



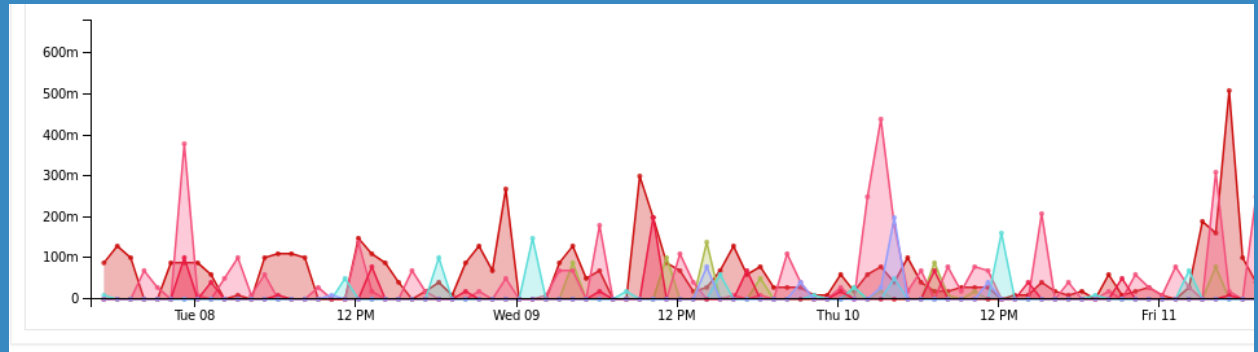
Intel MKL-DNN



theano



Bright offers a complete platform to get insights on your infrastructure...



Thank you

<https://www.brightcomputing.com>



