Deploying EDA Workloads On Public Cloud

Melvin Cardozo

Topics

- Cloud Motivations
- EDA compute workloads on Traditional, Cloud-First, Hybrid infrastructure
- Cloud Native Infrastructure and Terminology
- Customer Cloud Challenges
- Current Cloud Status For Synopsys Tools
- Case Study 1 - VCS
- Case Study 2 – PrimeTime, ICV2

Synopsys Confidential Information - Internal Use Only

Cloud Motivations

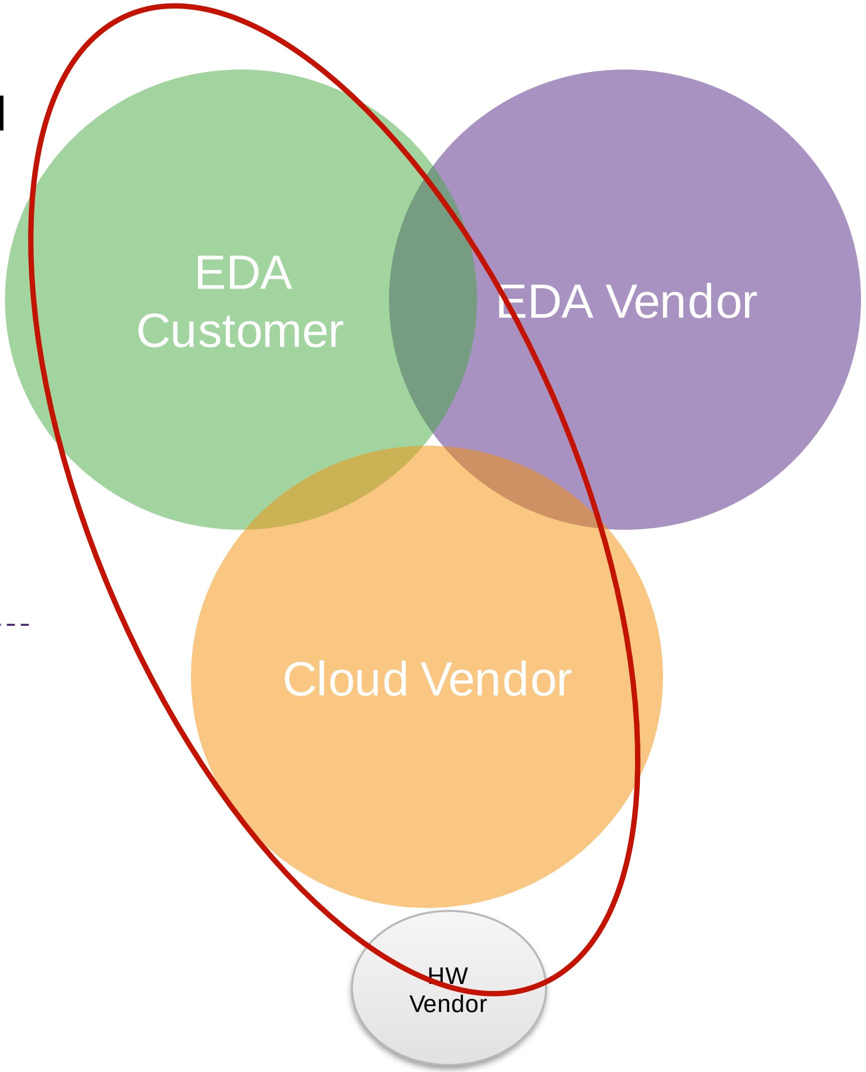
Synopsys Confidential Information - Internal Use Only

Cloud Participants in EDA

Motivation:
- Optimize the use of HW farm spend
- Use the best tools, methodology

New Motivation:
- Short term project-specific needs
- Improve TTR, QoR
- Managed Services
- New business (ARM, AMD, nvda, xlnx)

New Motivation:
- Make money from HW & managed services
- EDA compute is a big target (with specific infra requirements)



Motivation:
- Best / fastest engines
- Take advantage of h/w

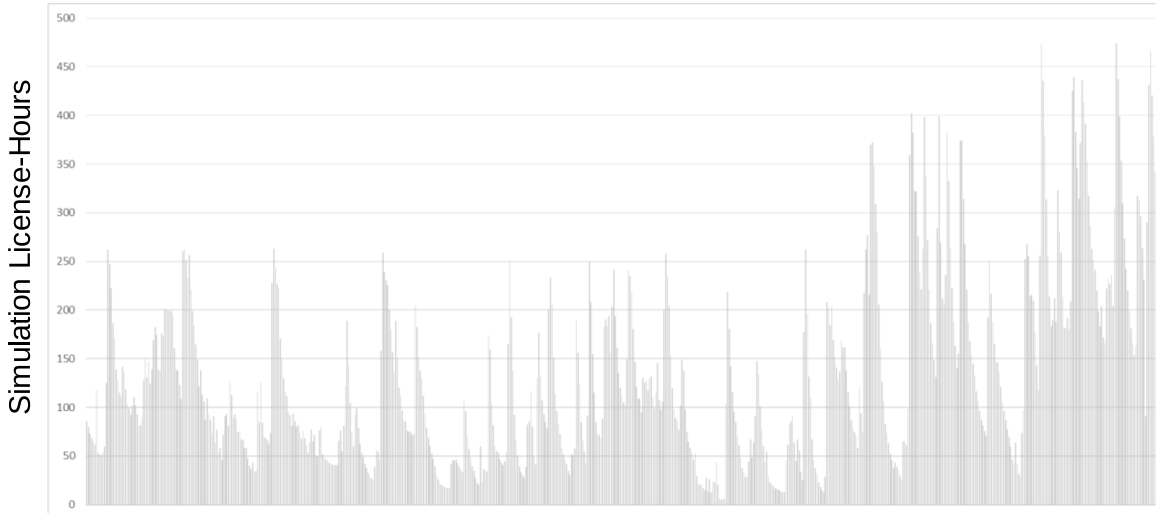
New Motivation:
- Use elastic & unlimited scale to improve product vs competition
- Make money off software in this segment
- Make money off special purpose hardware in this segment

Confidential – For Internal Use Only

Astera Labs: VCS on Amazon Cloud

Actual Usage with Hourly License Model

87,078 License-hours over 28 days



28 days with unlimited VCS licenses

Equivalent Usage with TSL Model

87,078 License-hours over 227 days

227 days with 16 VCS licenses

- Customer had only 16 VCS licenses available under TSL model
- Monitored real project use with “unlimited” cloud access to VCS – 87K license-hours used in 1 month
- Cloud access effectively delivered 8x faster TAT and 6+ months shift-left

Cloud can enable huge shift-left!

SYNOPSYS®

Traditional, Cloud-First & Hybrid Cloud

Synopsys Confidential Information - Internal Use Only

Verification on Cloud Segments

	Large Semi	Mega Disruptors	Startup Semi
Description	Traditional chip design companies with on-premise resources for EDA workloads	Companies which provide web services, products, platforms & infrastructure; moving into value added chip designs	Small, new fabless semiconductor companies looking to bring initial products to market quickly
Large, existing IT team and compute infra	Yes	Yes	No
Large, legacy EDA infrastructure	Yes	No	No
Existing compute infra designed for cloud-native workloads	No	Yes	No
Require "burst" access for peak or project use	Yes	No	No
CAPEX-constrained	No	No	Yes
Target cloud usage model	Hybrid for peak	Private	Public
EDA & related IT infrastructure spend	$$$	$$	$
Examples	*AMD*, Intel, Nvidia, Qualcomm, Samsung, Broadcom, *ARM, Xilinx*	Google, Amazon, Microsoft, Facebook, Baidu, Alibaba	Astera Labs, Groc, Cerebras, Esperanto, Sambanova, Graphcore

90+% Utiliza
tion

SYNOPSYS®

EDA workloads: Traditional Architecture

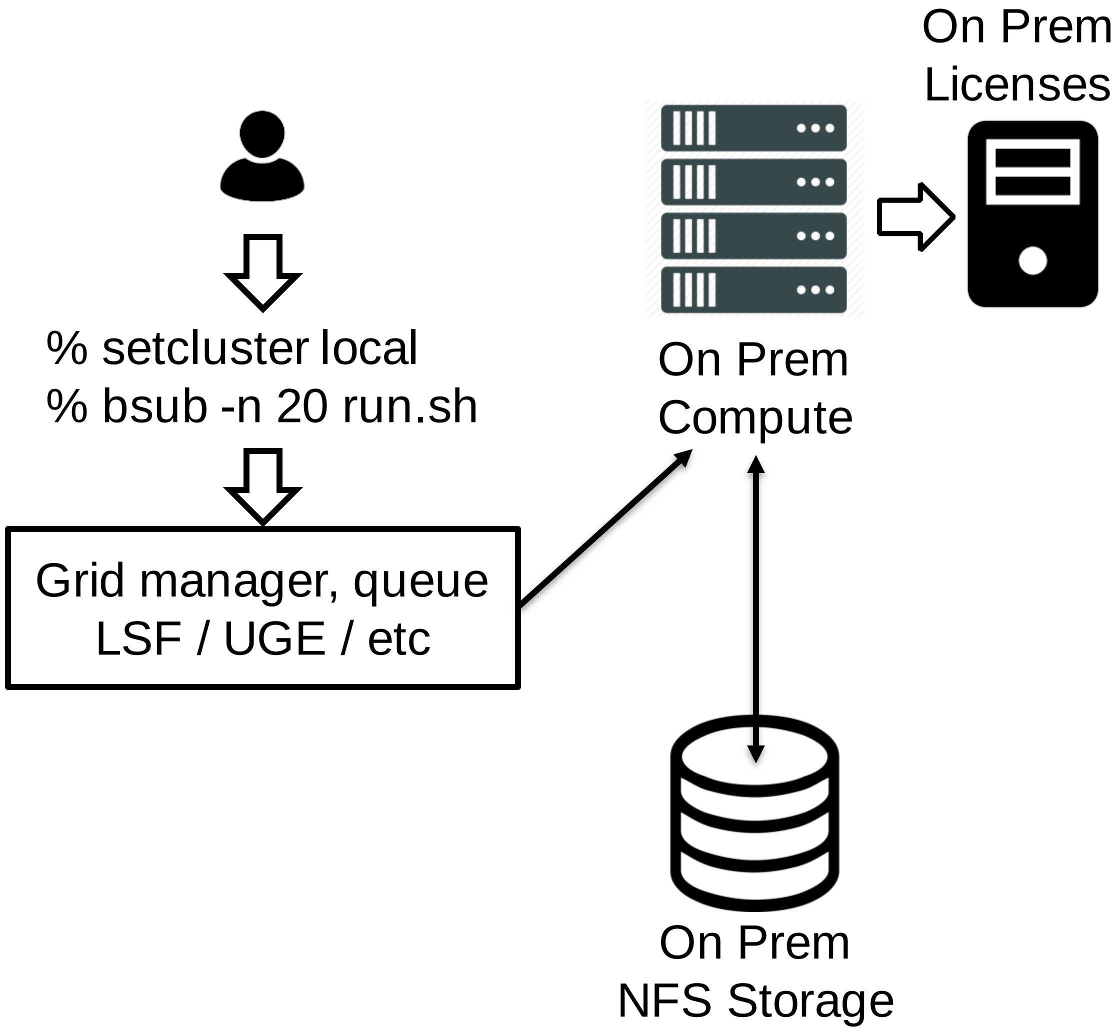
Users Access
VMs, Desktops, NFS

Tools and Workflows
NFS, multiple versions pre-installed

Grid software
Launch on pre-specified queues

Compute Layer
Bare Metal, Validated OS

Data and Storage Layer
NFS



Synopsys Confidential Information - Internal Use Only

EDA workloads: Cloud First Architecture

One of several options, not including security, authentication, authorization

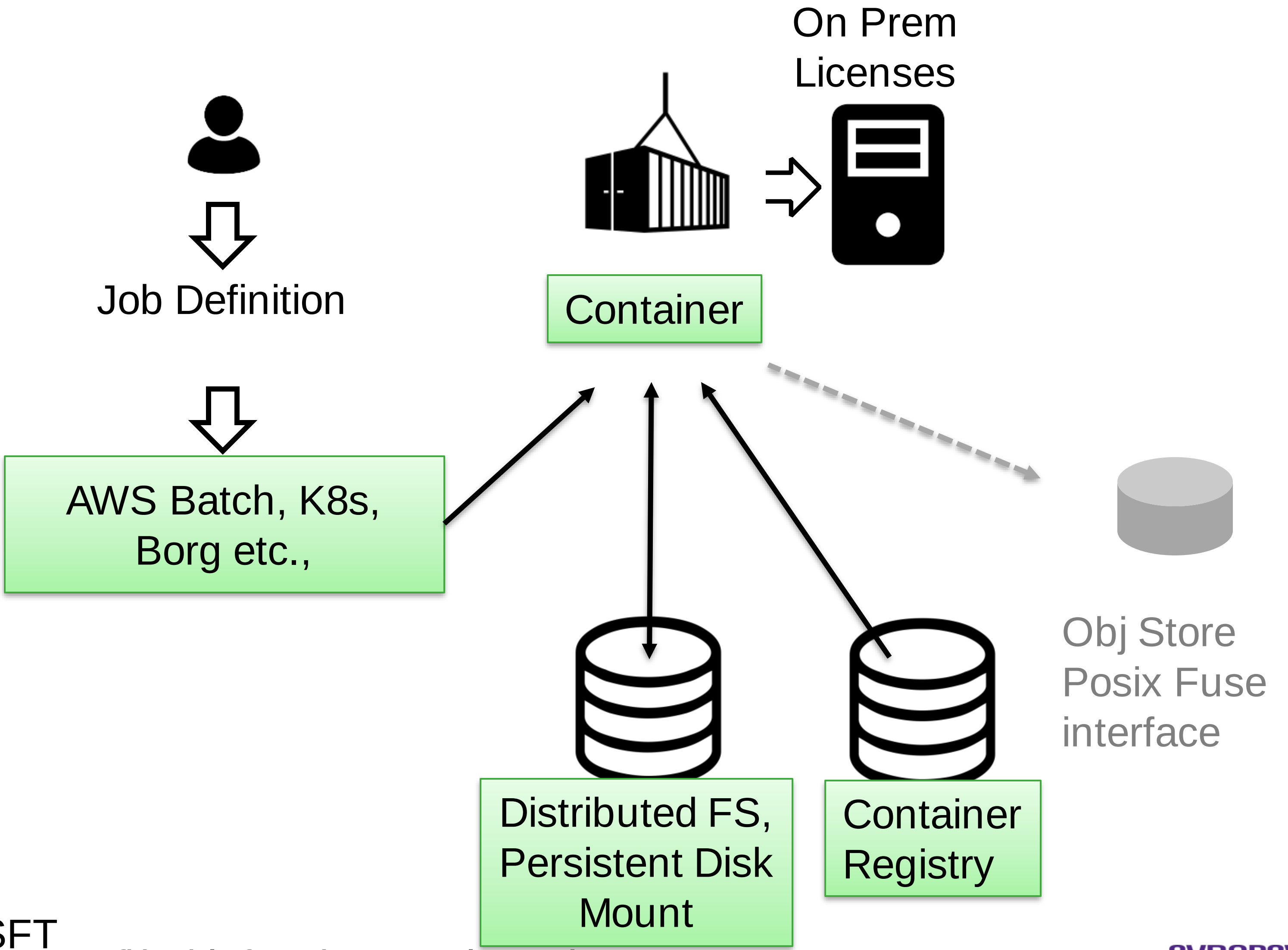
Users Access
VMs, Desktops, Distributed FS

Tools and Workflows
Local disk, Distributed FS, EDA Tools on read-only FS, containerization

Cloud Orchestration
K8s, Borg, AWS Batch, Azure Batch

Compute Layer
Containers, Contrainer registry, Orchestration

Data and Storage Layer
Distributed FS, Persistent Disk, Tools on Read-only FS, Obj Store



*: Discussions with AMZN, Google, MSFT

Synopsys Confidential Information - Internal Use Only

Hybrid cloud for traditional EDA users

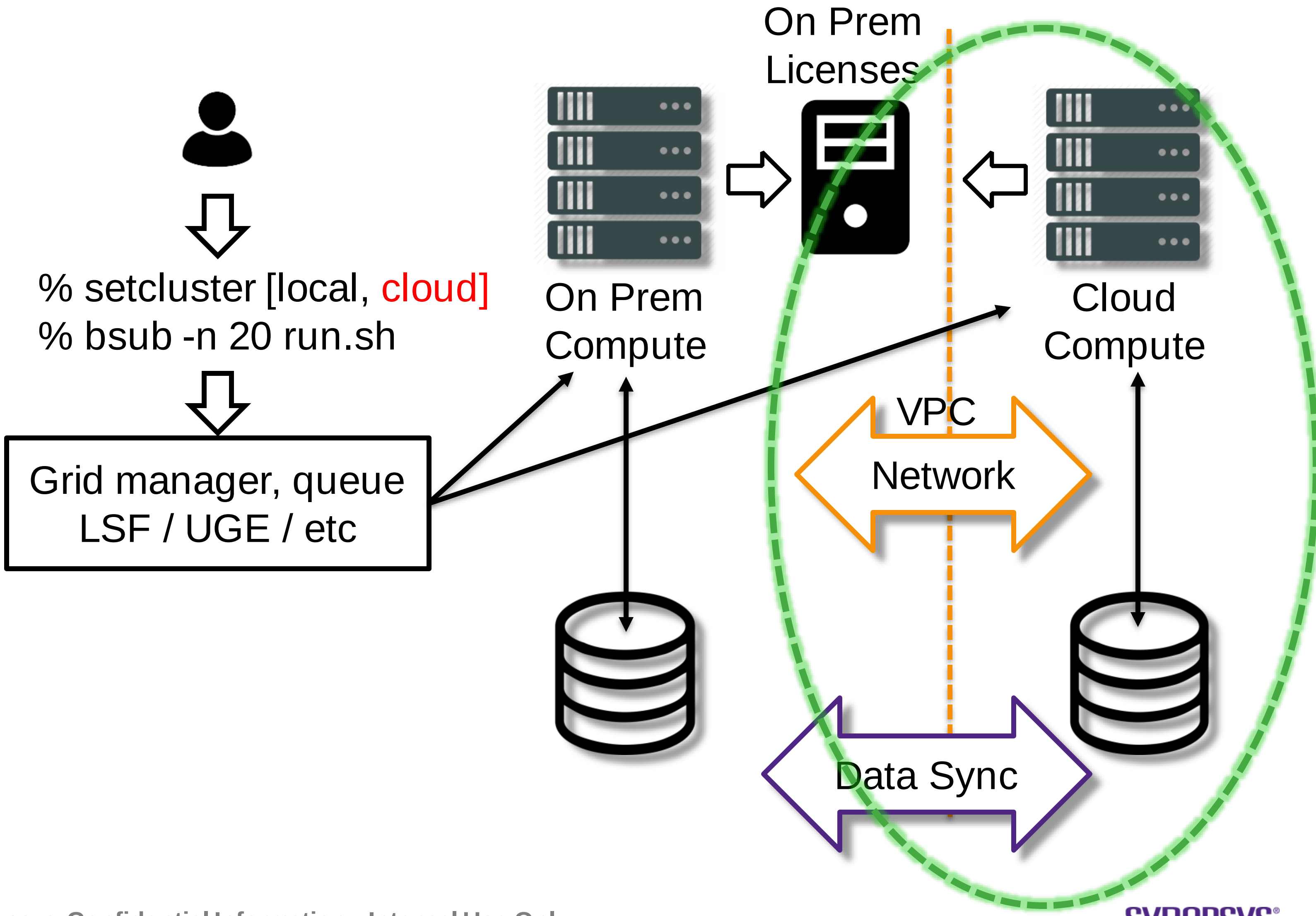
Users Access
VMs, Desktops, NFS

Tools and Workflows
NFS, Data sync with cloud, tools installed on cloud

Grid software
Launch on pre-specified queues

Compute Layer
Bare Metal, Validated OS

Data and Storage Layer
NFS



© 2018 Synopsys, Inc.

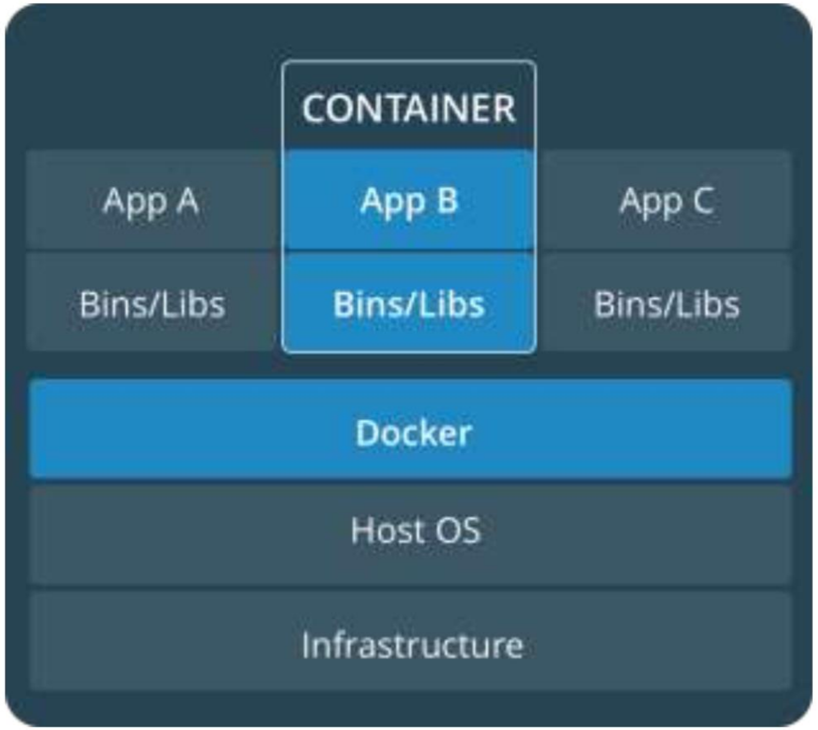
Cloud Native Infrastructure And Terminology

Synopsys Confidential Information - Internal Use Only

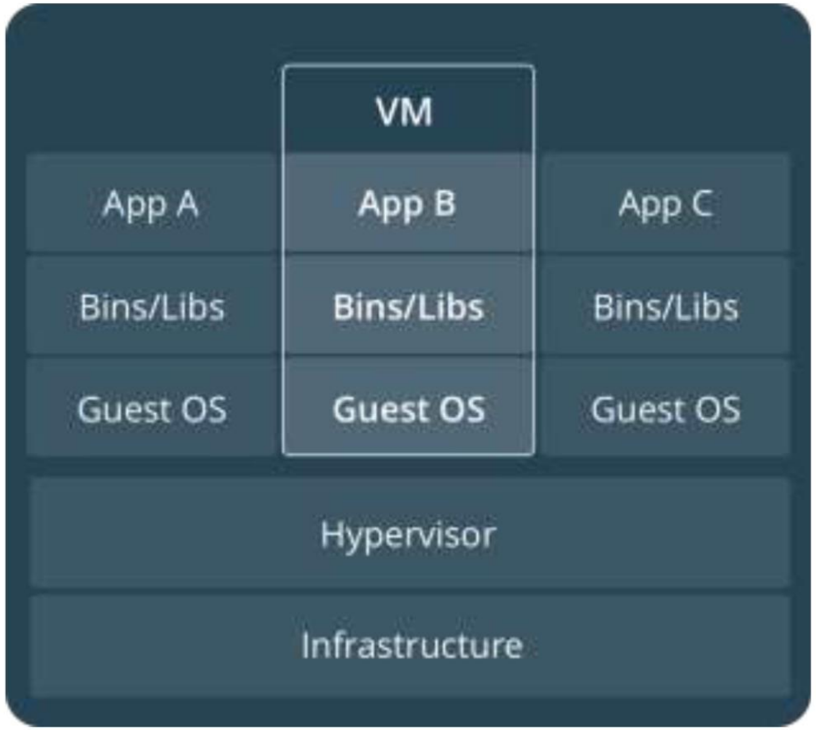
Amazon Cloud Offerings

- Analytics
- Application Integration
- AR & VR
- AWS Cost Management
- Blockchain
- Business Applications
- Compute
- Customer Engagement
- Database
- Developer Tools
- End User Computing
- Game Tech
- Internet of Things
- Machine Learning
- Management & Governance
- Media Services
- Migration & Transfer
- Mobile
- Networking & Content Delivery
- Robotics
- Satellite
- Security, Identity & Compliance
- Storage

Containers v/s VMs

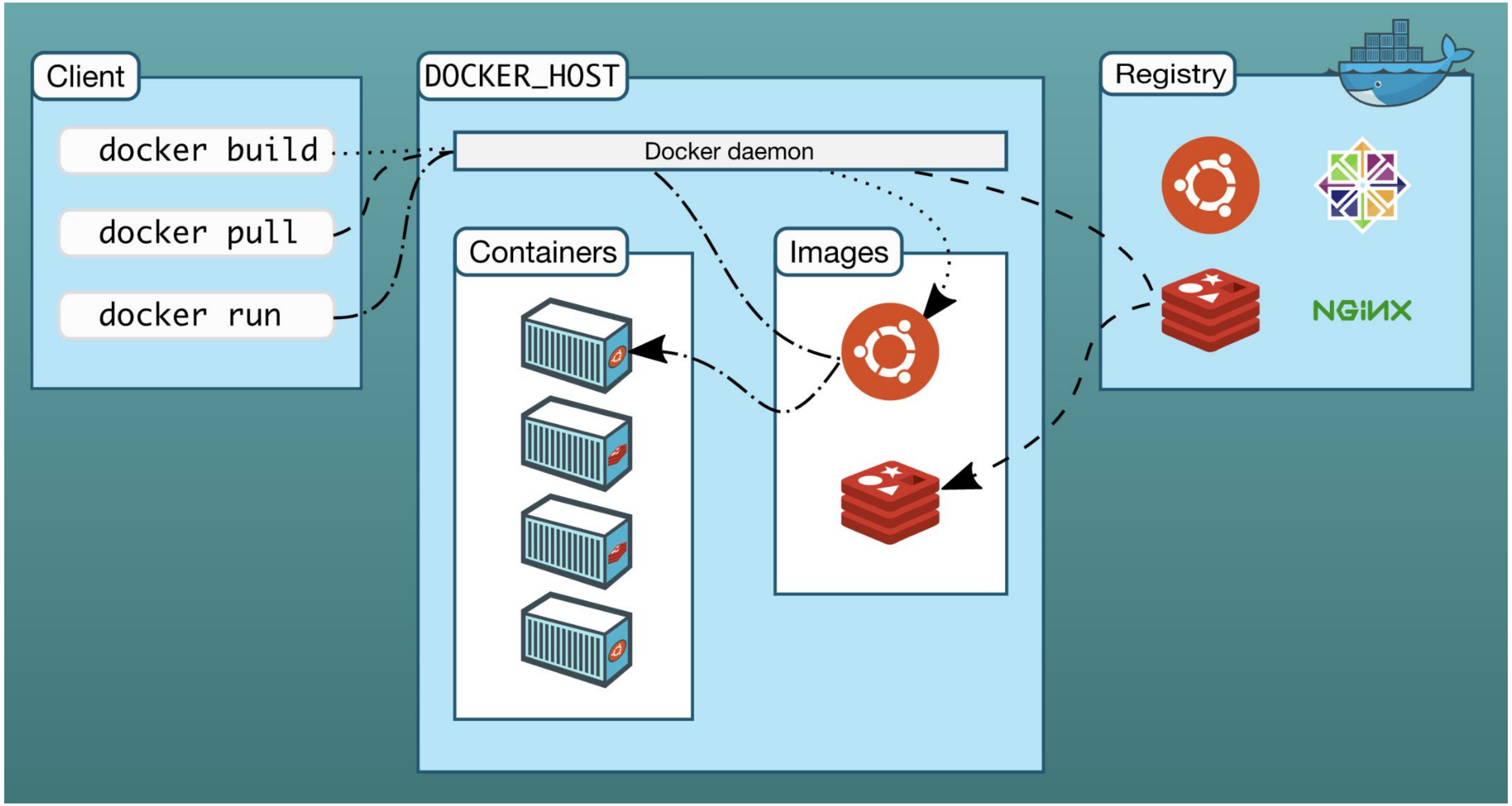


Containers are an app level construct



VMs are an infrastructure level construct to turn one machine into many servers

Docker Architecture



Docker Image v/s Container

- Docker Image
  - Docker images are **read-only** templates from which Docker containers are launched.
  - Each image consists of a series of **layers**.
  - Docker makes use of **Union file systems** to combine these **layers** into a single image.

- Docker Container
  - **Container** is built from an **image**.
  - A container consists of an operating system, user-added **files**, and **meta-data**.
  - That image tells Docker what the container **holds**, what process to **run** when the container is launched, and a variety of other **configuration** data.
  - The Docker image is **read-only**.
  - When Docker runs a container from an image, it adds a read-write layer on top of the image (using a union file system as we saw earlier) in which your application can then run.

© 2018 Synopsys, Inc.

Docker Image Registry

- Images are stored locally, but can be pulled from an Image registry
- The Registry is a server that stores and lets you distribute Docker images.

SYNOPSYS®

Dockerfile

```
FROM centos:7

ARG INSTALL_DIR

# Install the required dependencies
RUN set -ex \
    && yum makecache fast \
    && yum -y update \
    && yum -y install epel-release \
    && yum -y install mariadb-server \
    && yum clean all \
    && rm -rf /var/cache/yum

# Create
RUN set -
    && gr
    && us

# Create
RUN set -
    && mk
        $
        $
        /
```

```
# Install slurm
COPY $INSTALL_DIR/bin $INSTALL_DIR/bin
COPY $INSTALL_DIR/include $INSTALL_DIR/include
COPY $INSTALL_DIR/lib64 $INSTALL_DIR/lib64
COPY $INSTALL_DIR/sbin $INSTALL_DIR/sbin
COPY $INSTALL_DIR/share $INSTALL_DIR/share

# Install the config files
COPY $CONFIG_BASE/etc/slurm/slurmdbd.conf $INSTALL_DIR/etc/slurm/slurmdbd.conf

COPY $CONFIG_BASE/deploy/DATABASE/docker-entrypoint.sh /usr/local/bin/docker-entrypoint.sh
RUN sed -i -r "s#<INSTALL_DIR>#$INSTALL_DIR#g" /usr/local/bin/docker-entrypoint.sh
ENTRYPOINT ["/usr/local/bin/docker-entrypoint.sh"]

CMD ["slurmdbd"]
```

© 2018 Synopsys, Inc.

Docker Compose

```yaml
version: '3'

services:
  mysql:
    image: mysql
    hostname: mysql
    container_name: mysql
    environment:
      MYSQL_DATABASE: slurm_acct_db
      MYSQL_ROOT_PASSWORD: root
      MYSQL_USER: slurm
      MYSQL_PASSWORD: password
    volumes:
      - ./database:/var/lib/mysql
    ports:
      - 3306:3306
```

```yaml
  slurmdbd:
    image: slurmdbd:19.05
    command: [slurmdbd]
    container_name: slurmdbd
    hostname: slurmdbd
    volumes:
      - var_log_slurm:/var/log/slurm
    ports:
      - 6819:6819
    depends_on:
      - mysql

volumes:
  var_log_slurm:
```

Docker Build & Run

```
docker build -t slurmdbd:19.05 \
    --build-arg CONFIG_BASE=$CONFIG_BASE \
    --build-arg INSTALL_DIR=$INSTALL_DIR \
    --build-arg TARGET_INSTALL_DIR=$TARGET_INSTALL_DIR \
    --build-arg USER=$USER \
    --build-arg GROUP=$GROUP \
    --build-arg UID=$UID \
    --build-arg GID=$GID \
    -f Dockerfile \
    ../../..
```

```
docker volume create var_log_slurm
docker volume create var_lib_mysql
```

```
docker run -d \
    --name mysql \
    --hostname mysql \
    --env MYSQL_DATABASE=slurm_acct_db \
    --env MYSQL_ROOT_PASSWORD=root \
    --env MYSQL_USER=slurm \
    --env MYSQL_PASSWORD=password \
    --volume var_lib_mysql:/var/lib/mysql \
    --publish 3306:3306 \
    mysql:5.7

docker run -d \
    --name slurmdbd \
    --hostname slurmdbd \
    --volume var_log_slurm:/var/log/slurm \
    --publish 6819:6819 \
    slurmdbd:19.05
```

Object Store

Object Storage Systems Characteristics

- Data is stored as individual objects with a unique identifier
- Flat addressing scheme that allows for greater scalability
- Multi-tenant
- Usually software-based that runs on commodity hardware
- Capable of scaling to 100s of Petabytes
- Don’t use RAID but instead Replication and/or Erasure Coding
  - At PBs scale RAID has very long rebuild times
- Access over RESTful API over HTTP, which is a great fit for cloud and mobile applications
  - Amazon S3, Swift and CDMI API

Object Storage is a good fit for

- Unstructured data workloads
- Capacity requirements beyond 100s of Terabytes
- Distributed access to content
- Data archiving: documents, email, backups
- Storage for photos, videos, virtual machines images
- Need for granular security and multi-tenancy
- Need for automation, management, monitoring and reporting tools
- Non-high performance applications

Consumer Activity
(Events, GPS, WiFi)

Device Tracking and Logs
(Event, Configuration, Usage, Performance,)

Social media

Object Store APIs

Container/Bucket

GET /v1/ `{account}` / `{container}`
Show container details and list objects

PUT /v1/ `{account}` / `{container}`
Create container

POST /v1/ `{account}` / `{container}`
Create, update, or delete container metadata

HEAD /v1/ `{account}` / `{container}`
Show container metadata

DELETE /v1/ `{account}` / `{container}`
Delete container

Objects

GET /v1/ `{account}` / `{container}` / `{object}`
Get object content and metadata

PUT /v1/ `{account}` / `{container}` / `{object}`
Create or replace object

COPY /v1/ `{account}` / `{container}` / `{object}`
Copy object

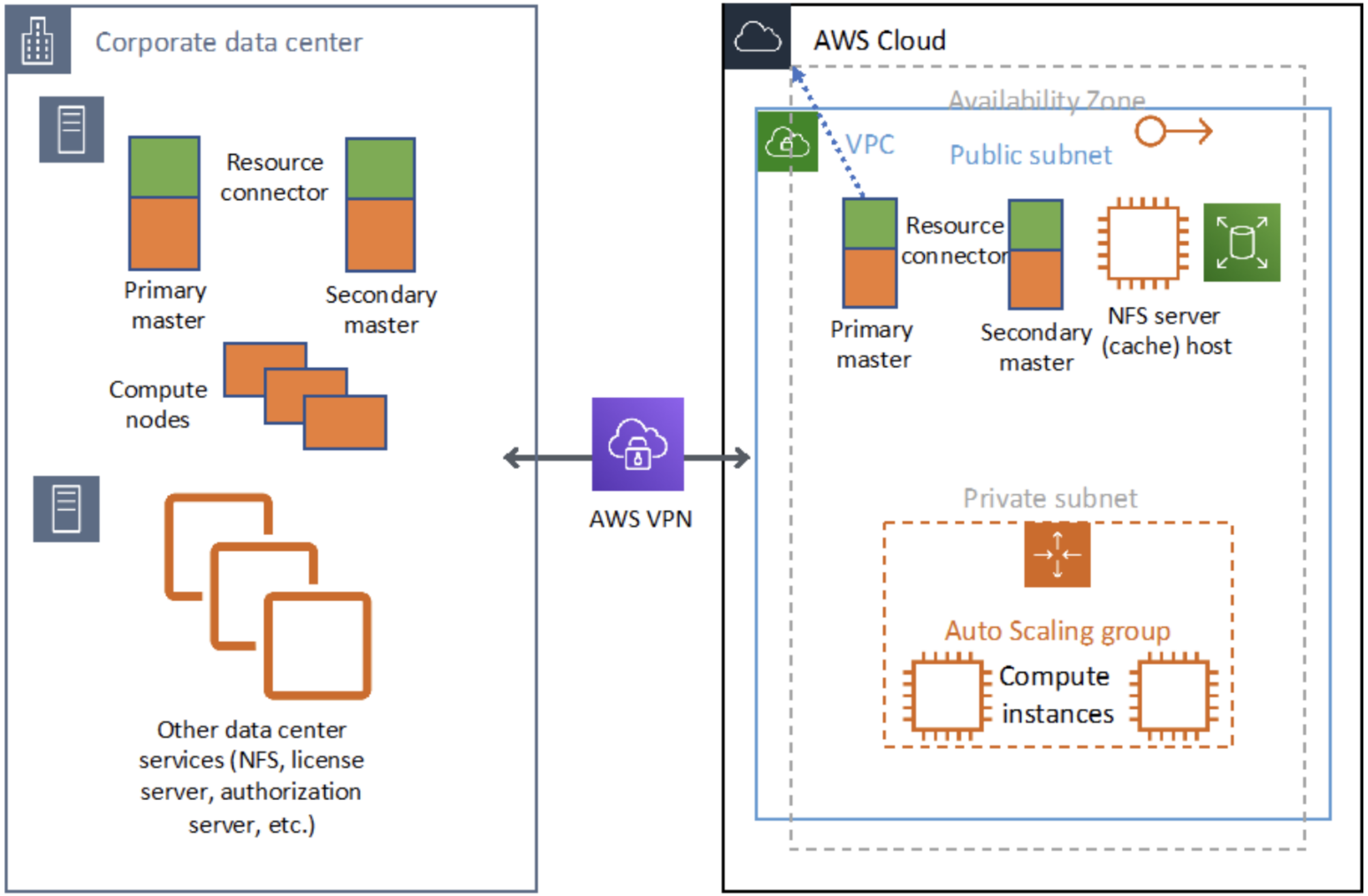
DELETE /v1/ `{account}` / `{container}` / `{object}`
Delete object

HEAD /v1/ `{account}` / `{container}` / `{object}`
Show object metadata

POST /v1/ `{account}` / `{container}` / `{object}`
Create or update object metadata

Batch Schedulers

LSF Connect


Corporate data center
Resource connector
Primary master
Secondary master
Compute nodes
Other data center services (NFS, license server, authorization server, etc.)
AWS VPN
AWS Cloud
Availability Zone
VPC
Public subnet
Resource connector
Primary master
Secondary master
NFS server (cache) host
Private subnet
Auto Scaling group
Compute instances

Cloud Pricing - AWS

<50% Utilization

Instance Types	Cores	Memory (MB)	Disk	Description	OnDemand Price	Spot Price
t3.nano – t3.2xlarge	1 - 4	0.5 - 32	EBS	0.5GB/core	$0.005 - $0.332	32%
c5.large – cg.24xlarge	1 - 48	4 - 92	EBS	4GB/core	$0.085 – $4.08	38%
c5d.large – c5d.18xlarge	1 - 36	4 - 144	50G – 1.8T	4GB/core, SSD	$0.096 - $3.456	33%
m5.large – m5.24xlarge	1 - 48	8 - 345	EBS	8GB/core	$0.096 - $4.608	35%
m5d.large – m5d.24xlarge	1 - 48	8 - 345	75G – 3.6T	8GB/core, SSD	$0.113 - $5.424	30%
r5.large – r5.24xlarge	1 - 48	16 - 768	EBS	16GB/core	$0.126 - $6.048	28%
r5d.large – r5d.24xlarge	1 - 48	16 - 768	75G – 3.6T	16GB/core	$0.144 - $6.912	25%
i3.large – i3.16xlarge	1 - 32	16 - 512	475 – 15T	16GB/core, SSD	$0.156 - $4.992	30%
f1.2xlarge – f1.16xlarge	4 - 32	122 - 976	470 – 4T	1 – 8 FPGAs	$1.65 - $13.20	30%
a1.medium – a1.4xlarge	0.5 - 8	2 - 32	EBS	4GB/core, ARM	$0.025 - $0.408	33%
g3.4xlarge - g3.16xlarge	8 - 32	47 - 188	EBS	6GB/core, GPU	$1.14 - $4.56	30%
p3.2xlarge – p3.16xlarge	8 - 64	26 - 188	EBS	3GB/core, GPU	$3.06 - $24.48	30%

© 2018 Synopsys, Inc.

AWS EC2 Pricing Models

Instance purchasing option	Risk	Cost	Features
On-demand	Low	High	• Pay, by the second, for the instances that you launch.
Reserved	Low	Medium	• Dedicated compute, paid for up-front
Spot	High	Low	• Spare compute at steep discounts • Spot Instances can be interrupted by EC2 with two minutes of notification when EC2 needs the capacity back.

© 2018 Synopsys, Inc.

AWS WhitePaper

- Optimizing Electronic Design Automation (EDA) Workflows on AWS

  https://d1.awsstatic.com/whitepapers/optimizing-electronic-design-automation-eda-workflows-on-aws.pdf

Customer Cloud Motivations & Challenges

Synopsys Confidential Information - Internal Use Only

Cloud Challenges For EDA

Cloud is more expensive than on-premise dedicated compute. Network data *egress* is expensive
Spot instances on cloud comes close (*but still higher*) than on-premise.
Cost management, control and allocation is absolutely necessary.

Scalability in HW enables shift-left. EDA license models are the gating factor in flexibility.
Astera (startup) was a proof point for shift left with metered licensing, with win-win for Synopsys.

Data transfer to and from cloud is a bottleneck and cost in hybrid and data-intensive flows.
Data management and sync with cloud is a work in progress.

Grid scheduler issues for ephemeral instances and / or workload prediction strategy.
Data analysis and debug can become bottleneck.

For **startups, cloud is the preferred** solution. Looking for help and qualified flows.
Cloud setup still takes too much time (weeks) to start, with too many options.

Cloud security not mentioned as **a primary issue**. That seems past us.

© 2016 Synopsys, Inc.

Scale

Customer Needs

- HPC Scaling
  - Scale a single application run across hundreds' of cores (distributed system)
    - To reduce time
    - Handle large designs
  - Elastic scaling
- HTC Scaling
  - Scaling for peak usage to reduce overall turnaround time
- Workflow management
- Non-standard hardware
- Emulation hardware Scaling

Product Needs

- Products that can elastically scale their resource requirements with minimal communication overhead
- Handle non-standard hardware for cost/performance benefits
  - AMD
  - ARM
  - Cray (Azure)
  - ENI (Enhanced network interface for low latency communication)
- Host ZeBu/HAPS on the cloud

Compute Cost

Customer Needs

- Cloud vendors are pushing EDA towards spot/preemptible instances to bring costs in line with on-premise costs
- Spot/Preemptible instances can be terminated with a very short notice: 30-120s
- Spot/Preemptible instances availability and pricing is dynamic and different for different instance types
- Management interface for putting cost restrictions

Product Needs

- Products can support checkpoint/restore
  - One short
  - Periodic saves
- Products can change requirements dynamically based on available compute resources
  - CPU
  - Memory
- Products can be compiled against a specific target

Hybrid File Access

Customer Needs

- Jobs could run either on-premise or on the cloud depending on hardware availability
  - Typically not known till the application is ready to run
- Keeping data in-sync between on-premise and cloud is a challenge
- Cost of transferring data from the cloud to on-premise is a big factor in determining what can be run on the cloud

Cloud Vendors Provide

- Cloud hosted NetApp
  - Seamless sync between on-prem and cloud
  - No transfer cost (both directions)

Product Needs

- Products can generate executables that can run both on-premise and the cloud
- Packing all the job's dependencies into a container significantly reduces the data sync problem
- Ability to create summarized data and merge sets of summarized data
- Debug/Analysis tools that can be run remotely (close to the data)
- Co-locate emulation hardware on the public cloud

Startups

Customer Needs

- Infrastructure setup is a big issue
  - No dedicated IT expertise
- EDA tool setup is an overhead

Cloud Vendors Provide

- Documents on how to setup an environment
  - Too many options for each requirement
- Visualization and Management
  - e.g Nice DCV

Product Needs

- Startup kit for bringing up a verification optimized infrastructure on each cloud provider
- Prebuilt environments (or images) on each cloud provider with our EDA tool bundles pre-installed
  - Execution Manager
  - ML Platform
  - License Server
  - Different Tool Versions

© 2018 Synopsys, Inc.

Current Cloud Status For Synopsys Tools

Synopsys Confidential Information - Internal Use Only

Becoming Cloud Native

	Feature	
Ready	OS Compatibility	Doe the tool work without issues on cloud vendor OS (on a supported QSC machine)?
	License/Metering access	Can we host the license keys / checkout on Cloud ?
	Scale	How much compute capacity can the tool use ?
	IO dependency	How much is the tool dependent on NFS ?
Optimized	Distributed computing	Does the tool support usage of large number of cloud nodes ?
	Cloud data storage models	Does the tool support cloud based data storage (like Key-Value storage, not just NFS)?
	Cloud job scheduling	Does the tool run under schedulers favored by cloud vendors - like AWS Batch, PBS ?
	Flow/Data optimization	Is data transfer optimized ? Do tools work in coordination on cloud? Do the runs support burst/hybrid modes of running?
Native	Cloud native packaging	Does the tool use cloud Container technology to bypass Cloud OS compatibility issues ?
	Cloud native data usage	Does the tool support cloud data primitives (like S3, Object Storage) ?
	Application checkpointing	Does the tool work well when jobs are halted and resumed at will ?
	Cloud HPC support	Can the tool make use of high performance components on cloud (Infiniband, Lustre) ?
	Elasticity	Can the tool dynamically scale its demand and throughput using more/less capacity ?
	Application Security	Does the tool support cloud native security like data encryption and application hardening ?
	License scaling	Does the tool optimize checkouts for cloud scale? Does it allow metering in the order of hours?

© 2018 Synopsys, Inc.

VG – Cloud Matrix

● Complete. ⊗ Not Available
N/A Not Applicable
☐ In progress / partially tested
◔ Low ◕ High Dependency Levels

	Feature	VCS/VIP	Static	Formal	Verdi	Wattson	ZeBu	HAPS	Hawk	FPGA	VP
Ready	OS Compatibility	●	●	●	●	●	●	●	●	●	●
	License/Metering Access	●	●	●	◕	●	●	●	●	●	◕
	Scale	x,000,000	x,00	x000	x,000,000	☐	x,000	x,00	x,00	x,000	●
	IO dependency	◕	◕	◕	◕	◔	◕	◕	◕	◕	◔
Optimized	Distributed computing	●	☐	●	◔	●	●	●	●	●	N/A
	Cloud data storage models	◕	◕	⊗	☐	●	⊗	⊗	◕	⊗	⊗
	Cloud job scheduling	●	●	●	⊗	☐	●	●	●	⊗	⊗
Native	Cloud native packaging										
	• Singularity / DP (in-tool)	●	☐	⊗	⊗	☐	N/A	N/A	●	⊗	⊗
	• Docker / Kubernetes	●	☐	⊗	⊗	☐	N/A	N/A	●	⊗	☐
	Cloud native data usage	⊗	⊗	⊗	☐	●	⊗	⊗	⊗	⊗	⊗
	Application checkpointing	●	●	⊗	⊗	☐	☐	☐	●	☐	⊗
	Cloud HPC support	●	⊗	⊗	⊗	●	⊗	⊗	●	⊗	N/A
	Elasticity	●	⊗	⊗	◔	☐	●	●	●	●	◕

SNPS - Cloud Matrix

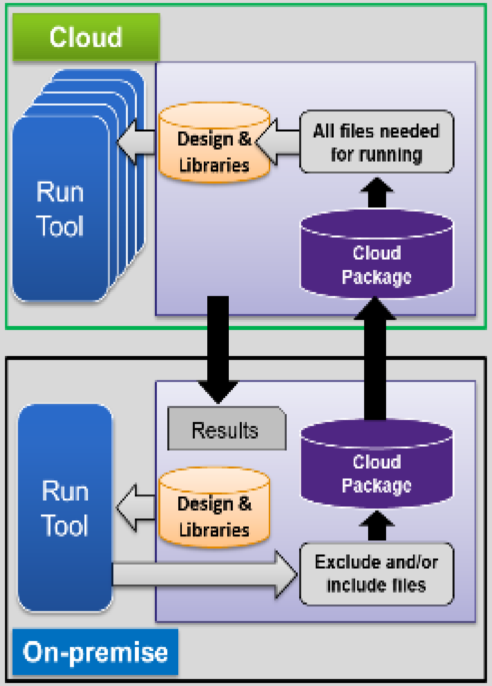
● Complete. ⊗ Not Available
N/A Not Applicable
☐ In progress / partially tested
◔ Low ◕ High Dependency Levels

	Feature	ICV	Primetime	StarRC	Tetramax	VCS	SiS	Proteus	CATS
Ready	OS Compatibility	●	●	●	●	●	●	●	●
	License Accessibility	●	●	●	●	●	●	●	●
	Scale	x,00	x,00	x,00	x,000	x,000,000	x,00,000	x0,000	x0,000
	NFS Dependency	◕	◕	◕	◕	◕	◐	◐	◐
Optimized	CDPL Integration	⊗	●	●	⊗	N/A	●	◔	●
	CDSL Integration	⊗	●	⊗	⊗	N/A	●	◔	●
	Alternate DRM Support	●	●	●	●	●	●	●	●
Native	Containers								
	• Singularity / DP (in-tool)	●	●	●	●	●	⊗	⊗	⊗
	• Docker / Kubernetes	⊗	⊗	⊗	⊗	●	⊗	⊗	⊗
	Object Storage	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
	Checkpointing	⊗	⊗	⊗	⊗	●	⊗	●	⊗
	HPC Infrastructure	●	●	●	●	●	●	●	●
	Dynamic Scale	●	●	◕	◕	●	●	◕	◕

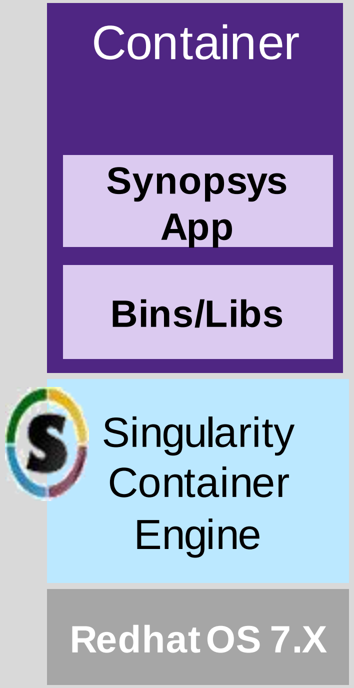
© 2018 Synopsys, Inc.

Cloud Building Blocks

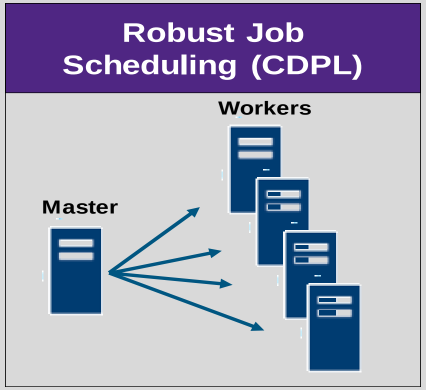
Cloud Transfer Packager

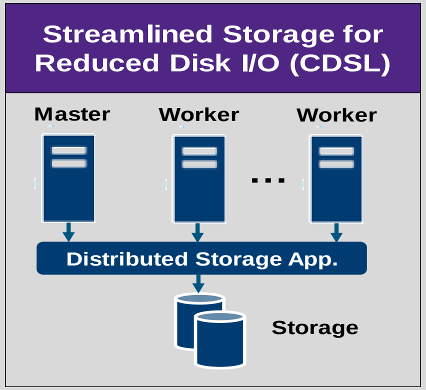


Cloud Tool Containers



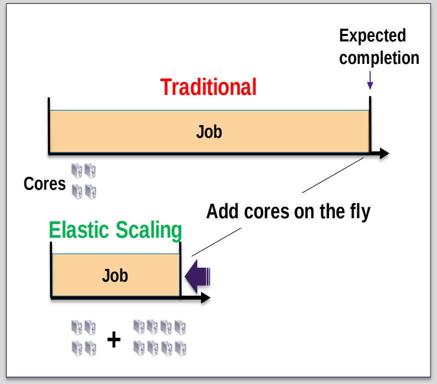
Distributed-Workload Manager



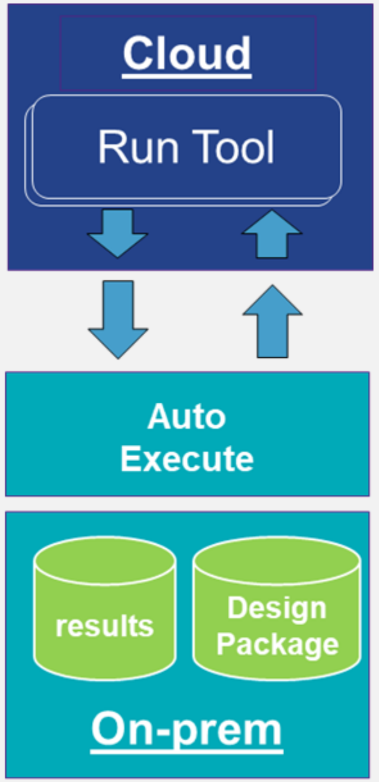


Elastic Scalability

Adds Cores on the fly



Hybrid Connector -*$cloud*



© 2018 Synopsys, Inc.

SCTP : Synopsys Cloud Transfer Packager

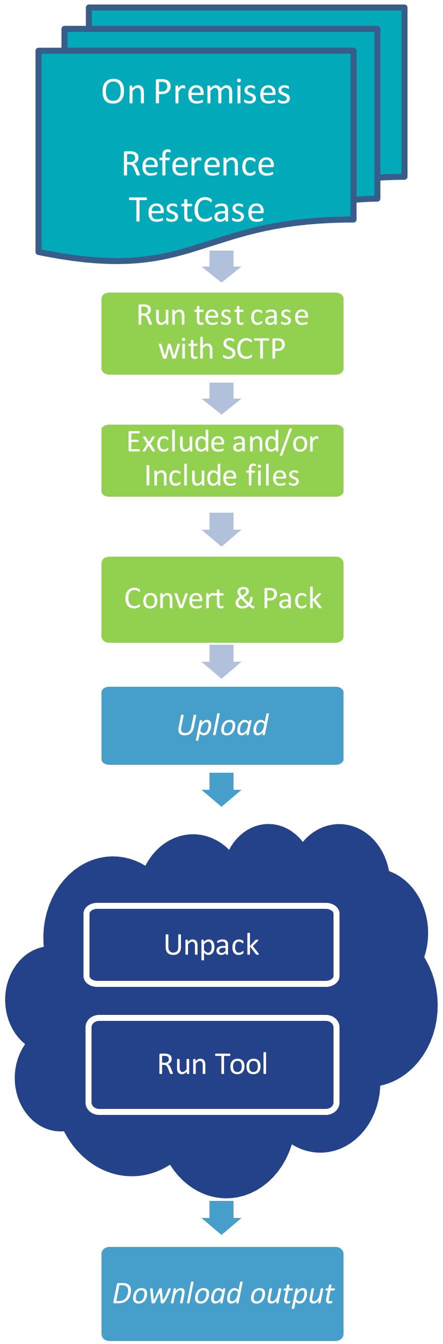
Product Integration Guide : https://sweportal/sde/sctp/Pages/Product-Integration.aspx

DEFINITION : Create & Xfer packages to & from premises to cloud

/depot/tools/sctp/sctp

Key features

- ✓ Standalone / Product independent
- ✓ Generate/Pack & Unpack full data dependency tree*
  - Multithreaded execution
- ✓ Maintains environmental namespace on-prem to cloud
- ✓ MT & Distributed support
- ✓ Reduces upload files sizes
  - Comprehensive exclusion rule set
- ✓ Full Incremental functionality / overlay
  - Upload & download
- ✓ Package creation for Hybrid model execution (-cloud)

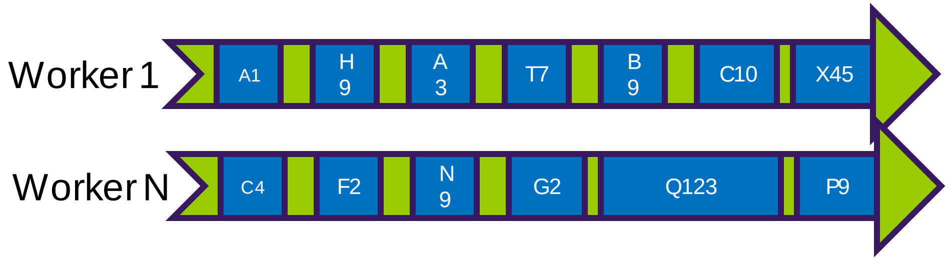


*requires tool run

CDPL: build complex distributed flows in our tools

Simple, uncoupled operations

- Highly parallel operations

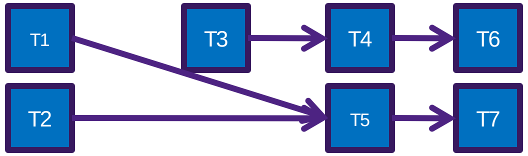


Simple, time dimension flows

- Single-lane sequenced fast operations

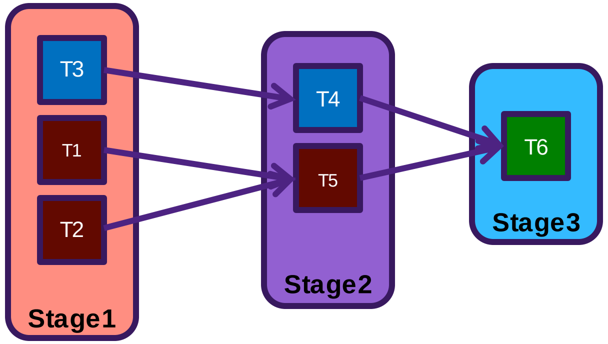
Domain specific tasks

- Operations with dynamic dependencies



Phases, Checkpoints

- Batched/staged operations

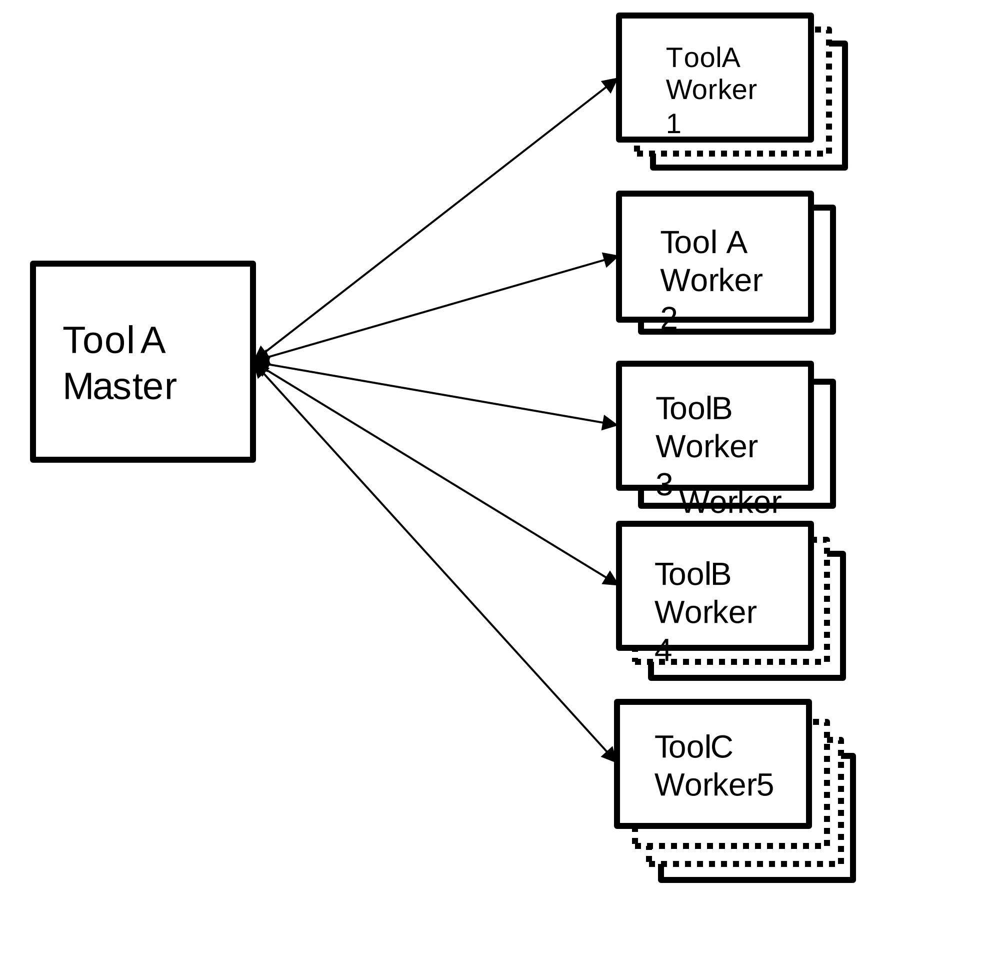


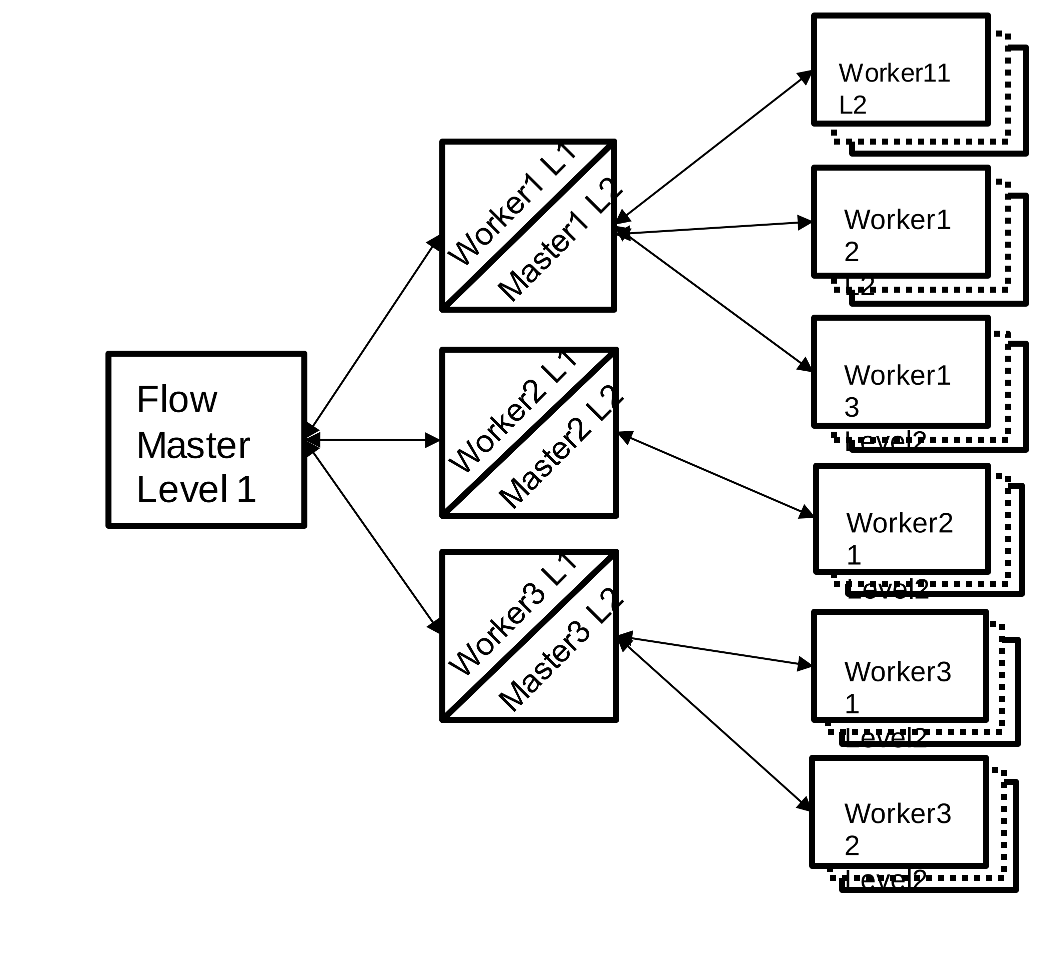
© 2018 Synopsys, Inc.

Examples of applications (today, and in the works)

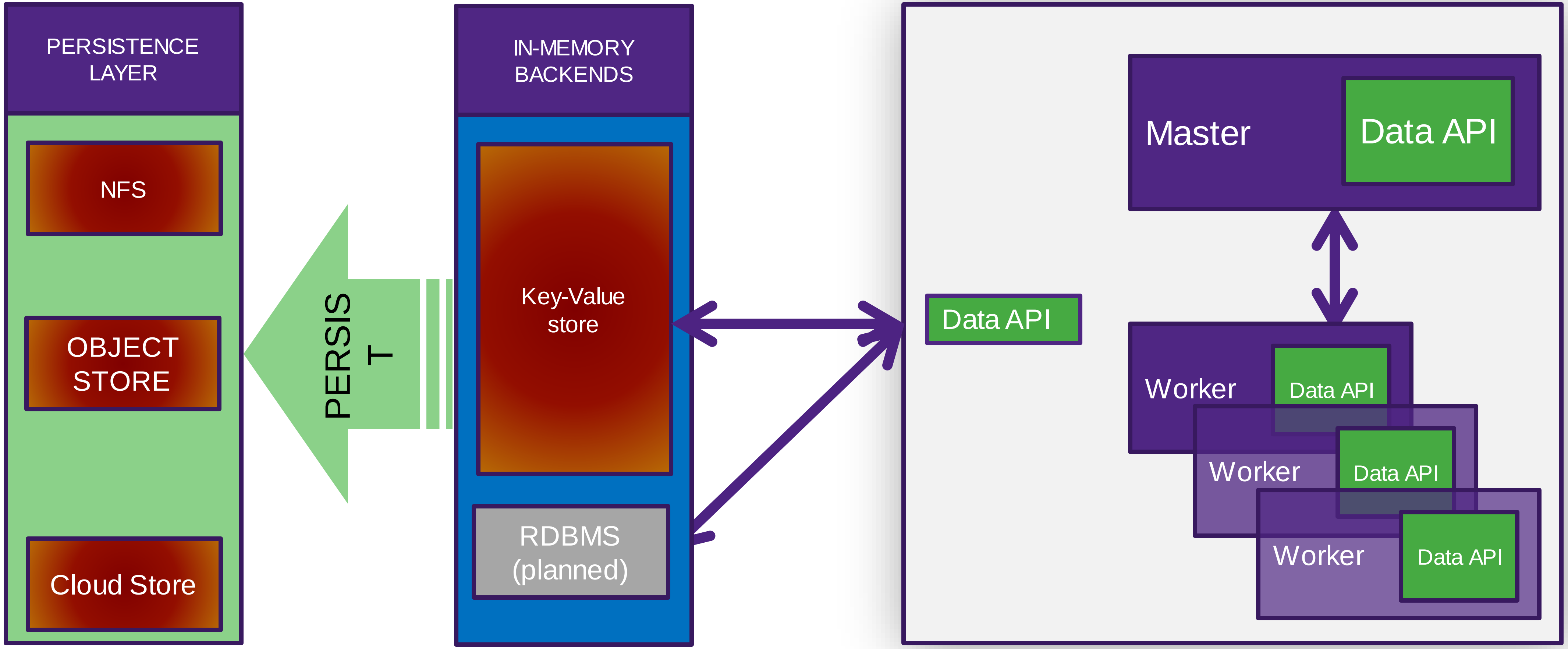
Large and Complex DP possibilities

Single-product/Multi-product flow	Single product, very large scale flow (WIP)





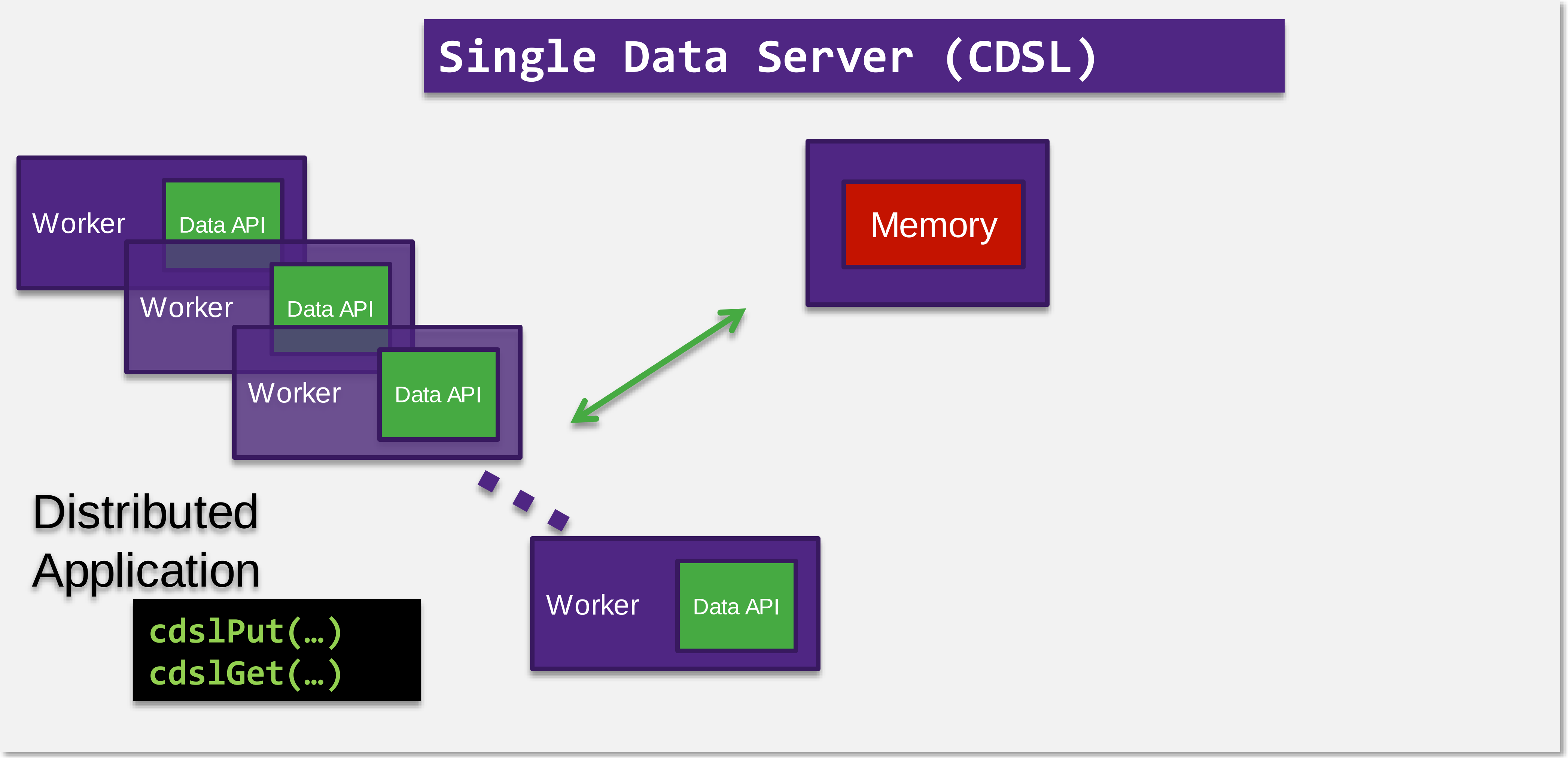
CDSL – providing a data management backplane



© 2018 Synopsys, Inc.

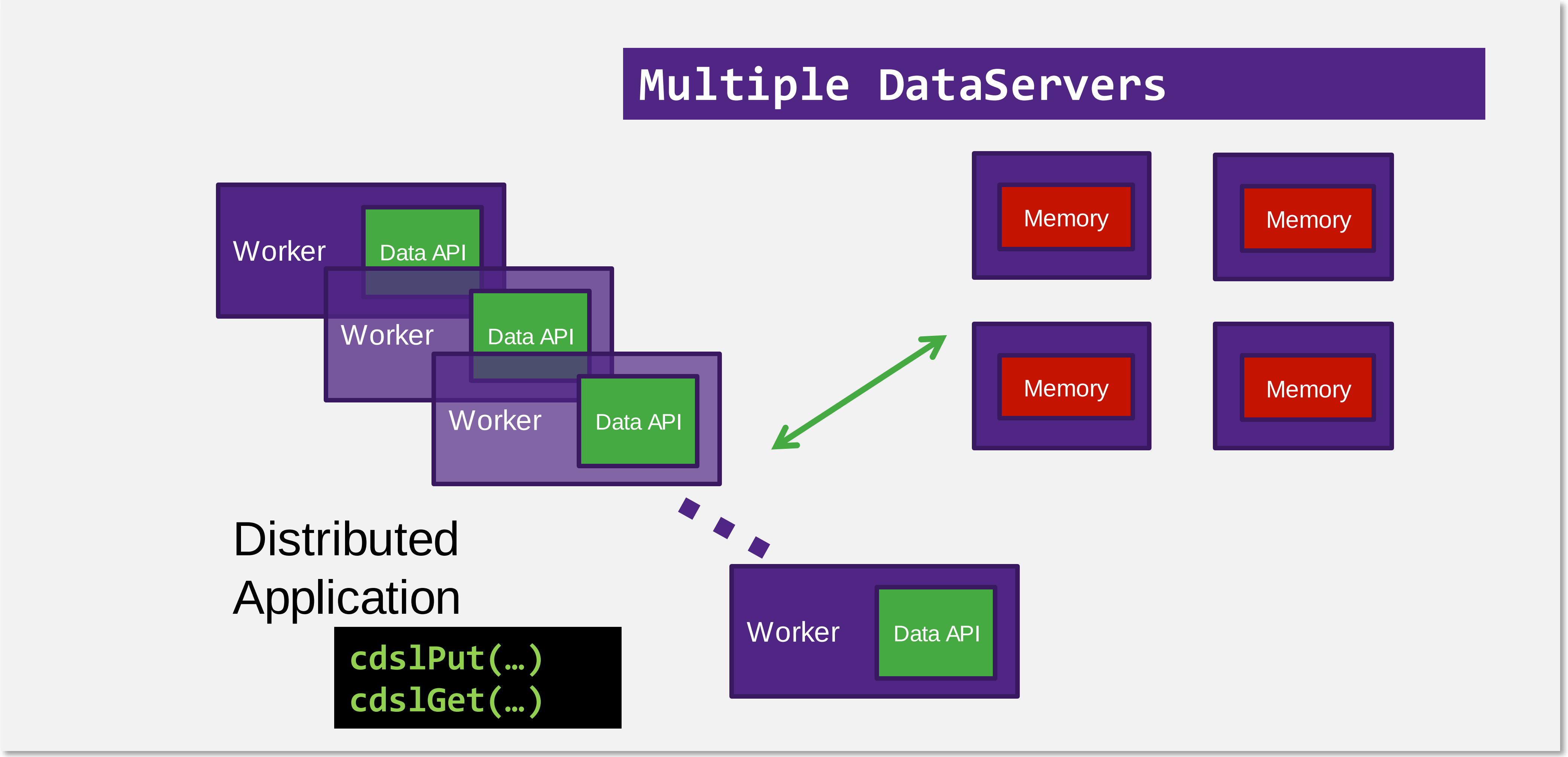
CDSL use model: single instance

Start and use one in-memory Key-Value Store



Horizontally scaled CDSL Data Servers

Shard data across a pool of in-memory Key-Value Stores

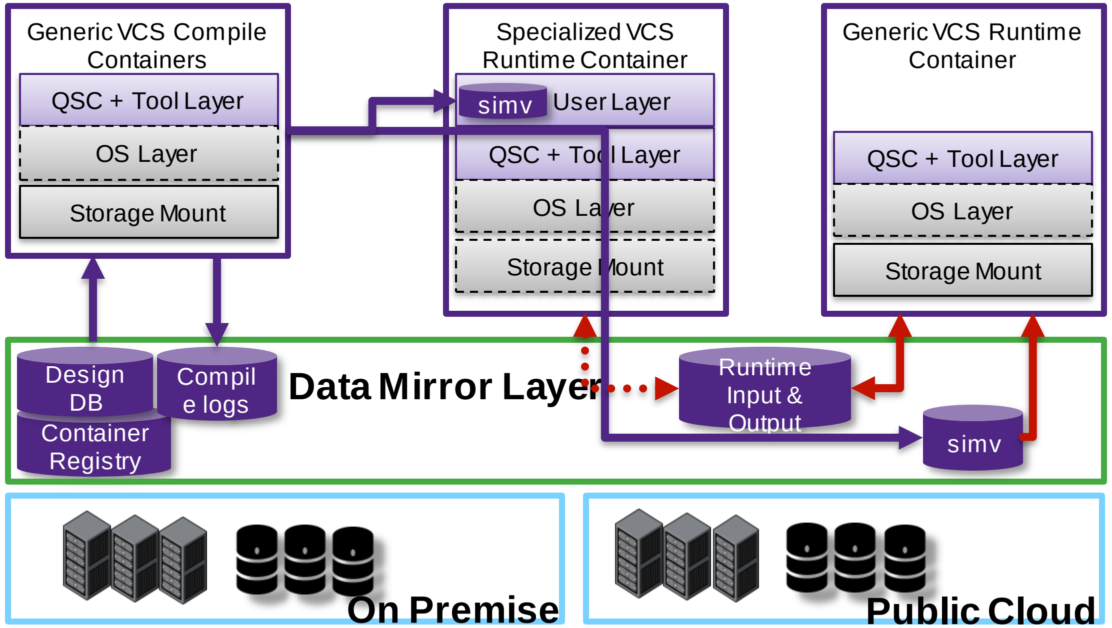


© 2018 Synopsys, Inc.

Case Study 1 - VCS

Synopsys Confidential Information - Internal Use Only

VCS – Data Consistency / Transfer



- Future Work
  - Eliminate NFS bottlenecks on cloud
    - Read only partitions backed by object store with a Posix interface
  - Sync of summarized ML data on cloud and on-prem
  - Distributed constraints caching

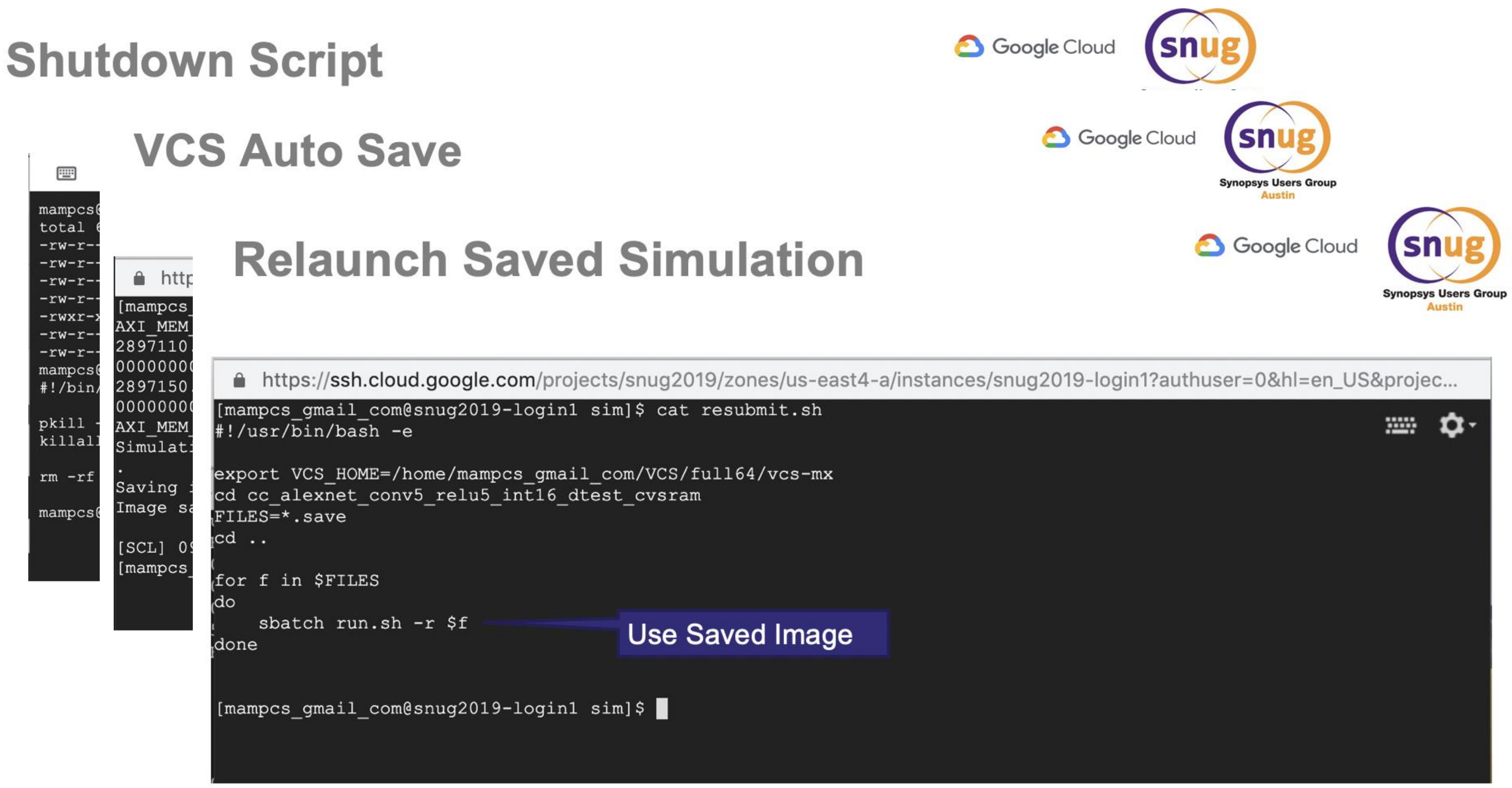
VCS Container Use Model

- vcs –container=singularity[:path to base image] –container_include=<file name>
  - If path to base image is not specified, default image is used from the vcs installation
  - File name is the path to a file that contains a list of directories/files to the copied into the container image

- vcs –container=docker:[from_registry/]image_name[:version] –container_include=<file name> -container_publish=to_registry
  - from_registry is the details of the registry where the base container image is available
  - to_registry is the details of the registry when the built image should be published
  - File name is the path to a file that contains a list of directories/files to the copied into the container image

© 2018 Synopsys, Inc.

VCS - Spot Instances



- Future Work
  - Support for other cloud providers
  - Tighter integration with schedulers for restart
    - LSF
    - Univa
  - Periodic saves
  - Support for heterogeneous instance types
  - Selection of spot instances based on spot pricing and simulation job requirements

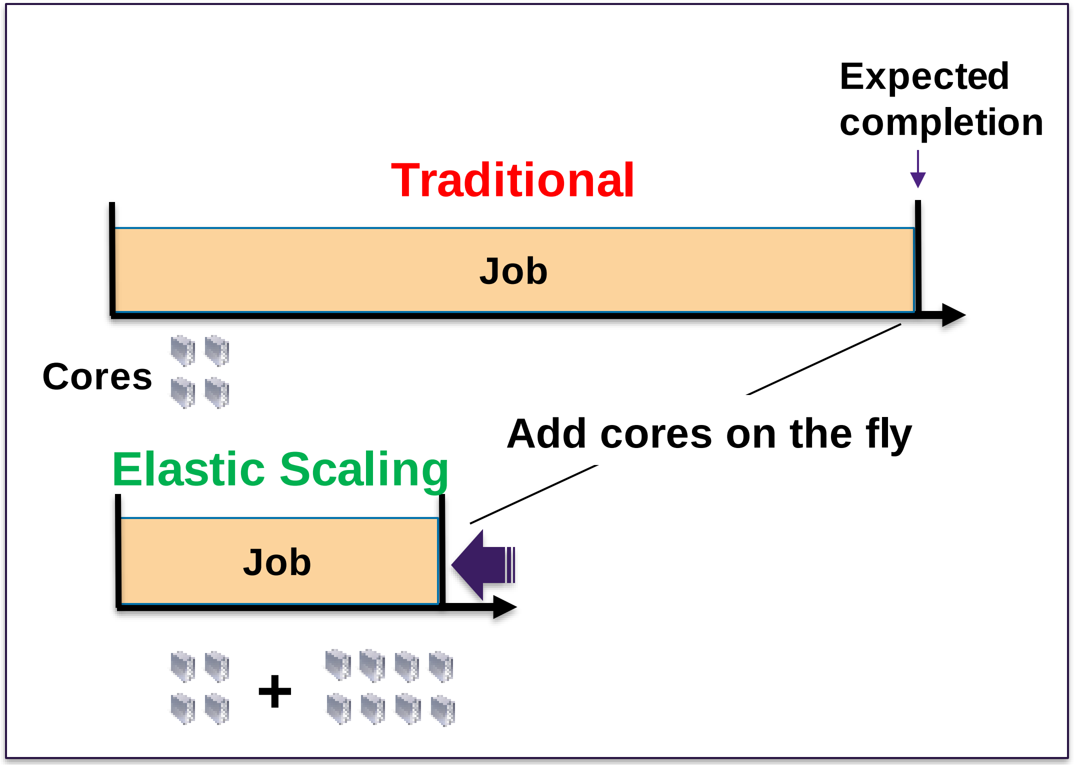
© 2018 Synopsys, Inc.

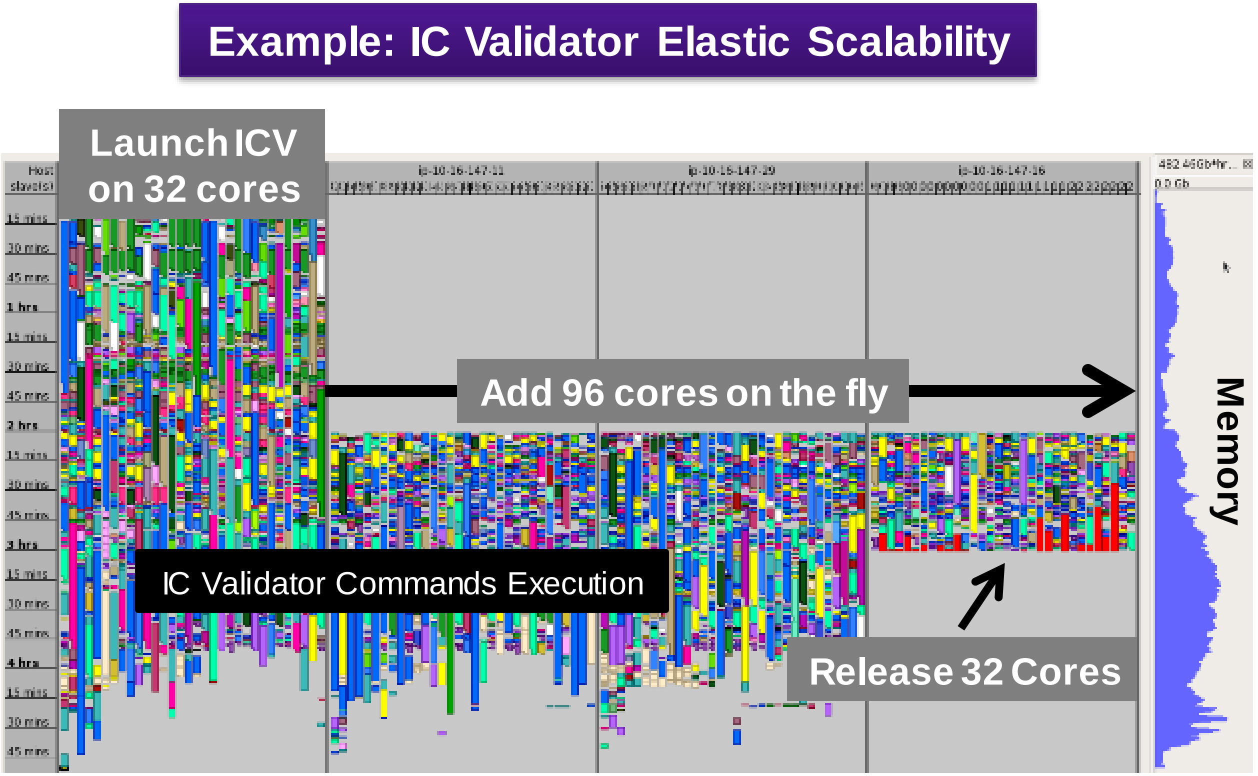
Case Study 2 – ICV

Synopsys Confidential Information - Internal Use Only

ICV - Elastic Scalability Technology for the Cloud

Add cores on the fly to a running job





Synopsys Confidential Information