Deployment Recommendations

The following topic outlines Parasoft's recommendations for a production-grade deployment of Parasoft Virtualize alongside Parasoft Data Repository and Parasoft Continuous Testing Platform (CTP). Sections in this topic include:

Assumptions

This document assumes that the deployment will include:

- Virtualize server—preferably deployed via the war file
- Parasoft Data Repository server
- Parasoft CTP and a supporting database (see Selecting a CTP Database for details)

Deployment Approaches

There are two deployment approaches: dynamic infrastructure (Docker image or Azure VM) or physical static infrastructure.

Dynamic infrastructures are designed for enabling dynamic, disposable test environments. This means that a test environment can be instantly provisioned from a golden template, used and dirtied, then simply destroyed. There is no need to share test environments or resources across teams or test phases; the exact environment you need is instantly spun up whenever you want it, then destroyed as soon as you're done with it. Dynamic infrastructures provide advanced flexibility for extreme automation. Moreover, when you need to scale (e.g., for performance testing), you can do that on demand.

With a physical static infrastructure, you have permanent (dedicated) servers. This is useful when long-term scaling is anticipated and hardware is designated ahead of time for heavy usage. Such an approach is designed for high availability and fault tolerance requirements. If you have such requirements and plan to configure a cluster of Parasoft Virtualize servers behind a load balancer, also see the recommendations at Setting Up a Cluster of Virtualize Servers Behind a Load Balancer.

Dynamic Infrastructure Recommendations

Dynamic infrastructures use either Docker Images or Microsoft Azure VMs.
For Docker, we recommend:

- Each image should contain a single Tomcat with CTP, Virtualize server (war file deployment), and a Data Repository server.
- The minimum size machine specs are 4 core and 16 GB RAM, but the recommendation is 8 cores, 32 GB RAM.
- We do not recommend more than 2-4 images on the same host OS because they will compete for resources (especially network resources). Having more than 4 images on the same host OS makes it difficult to diagnose any issues encountered.

For Azure, we recommend:

- A Virtual Machine size of DS2_V2 or bigger.
- An Azure Virtual Machine for Parasoft Service Virtualization (On-Demand or BYOL).

For AWS, we recommend:

- A Virtual Machine size of t2.medium or bigger.

Physical Static Infrastructure Recommendations

The following diagram shows the recommended architecture for a deployment with 3 Virtualize servers and CTP; note that the "Server n" icons in the lower right corner represent any number of additional servers (as appropriate for your environment).
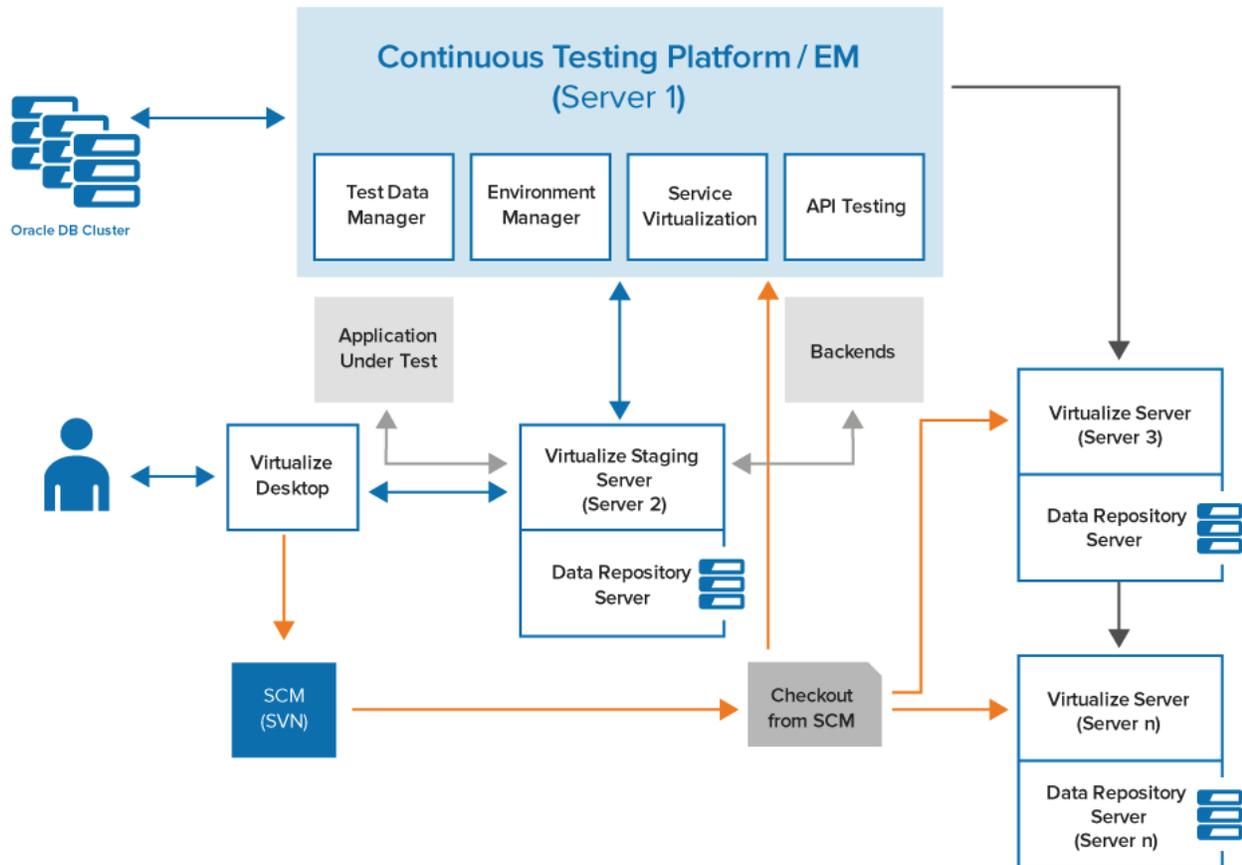
We recommend the following hardware for the Virtualize, Data Repository, and CTP server machines...

Virtualize

- CPU: 8 core
- RAM: 32 GB or higher
- Dedicated disk space: 300 GB or higher

Data Repository

- CPU: 4 core
- RAM: 32 GB or higher
- Dedicated disk space: 300 GB or higher

Always Separate Virtualize and Data Repository if Possible

If possible, separate Virtualize and Data Repository. This is important for "future proofing" your deployment. This becomes especially important:

- When Virtualize and Data Repository eventually compete for resources.
- If bad data repository happens to bring down the machine hosting Data Repository. If Virtualize and Data Repository are separated, this Data Repository failure won't take the Virtualize server down with it.

CTP

- CPU: 4 core
- RAM: 8 GB
- Dedicated disk space: 150 GB

Notes

- Since the database for CTP will eventually host multiple teams' artifacts as binaries, it will grow quite large.
- On Virtualize server, the workspaces typically account for the greatest storage allocation. All artifacts and data (including data snapshots) are stored here prior to checking in to source control.
- With Virtualize server, strong horsepower is essential for achieving optimal performance.
- With Virtualize server and Data Repository server machines, increasing the amount of RAM should boost performance—especially with Data Repository memory mapping.
- Meeting or exceeding the storage requirements for Data Repository servers is critical. You will be storing multiple versions of the same service and database data—from multiple teams, across multiple environments.

Operating System

We recommend Linux over Windows for Parasoft deployments because:

- Administrative tasks (such as startup, restart, logging, debugging) are easier. Also, if you want to run an application "behind the scenes" as a service, you can use nohup/ init.d/cron if you have Linux. Windows does not provide an equally simple solution.
- Permissions are easier to manage. A service account can be created so that only an admin group can view/modify files such as startup scripts, server.xml, etc.
- Performance is better. With Windows, the core OS consumes a considerable amount of performance. With Linux, the machine can be stripped down to bare bones—allowing Virtualize to run in its own space.
- Automation efforts are simplified thanks to tools like ssh and scp. Parasoft offers a package of scripts for these tasks on Linux.

Selecting a CTP Database

CTP supports Oracle, HyperSQL, and MySQL, but we strongly recommend Oracle or HyperSQL over MySQL; in a nutshell, our recommendation is

```
Oracle >= HyperSQL > MySQL
```

Oracle and HSQLDB perform equally well, but MySQL is difficult and challenging to troubleshoot. If you do not plan on clustering the CTP database and you have sufficient space, we recommend HyperSQL.

Starting Up Cloud-based Dynamic Infrastructure Deployments

When VMs are deployed in the cloud through cloud service providers, such as Microsoft Azure and Amazon AWS, machine IDs may change as the VM is shut down and restarted. Use the following flag when starting Parasoft products to ensure that the machine ID remains stable when VMs are restarted on cloud platforms:

```
-Dparasoft.cloudvm=true
```