Document Number and Revision: 923-3734 Rev03

Test Infrastructure Overview

Overview

This document provides a high-level overview of the test infrastructure used to implement the test environment and to support the execution of test processes on manufactured product. The document highlights the roles and responsibilities of the supplier when hosting and administering the test infrastructure.

Audience

External supplier's IT, test and manufacturing engineering departments.

TABLE OF CONTENTS

Test Infrastructure Overview

The test infrastructure implements the environment that is used to support the execution of test processes on manufactured hardware. The hardware includes servers, storage and switches. Test processes include board, system and integrated solution level tests. In addition to ensuring that product has been assembled correctly, processes include: firmware updates; recording configuration details; and the installation of operating system and application software. Test processes are applicable during NPI (new product introduction), mainline manufacturing, remanufacturing and service repairs. The test environment is applicable in both manufacturing and laboratory settings.

This document provides a high-level overview of the test infrastructure used to implement the test environment and to support the execution of test processes on manufactured product. The document highlights the roles and responsibilities of the supplier when hosting and administering the test infrastructure.

Both the test infrastructure and the supported test processes are subject to full life-cycle change and configuration management policies; uncontrolled change is difficult to achieve and is detectable.

NAMING CONVENTIONS (INITIAL)

Every entity in the test environment has a structured identity. There is a well-defined sequence of transformations used to obtain these.

Server hostnames

All physical and virtual servers have hostnames that have the same structure composed of three hyphen separated components:

<prefix>-<type>-<index>

The prefix is further subdivided into two parts:

<supplier><site>

A two-letter supplier ID; and a two-letter site ID. The site ID usually represents the city where the manufacturing facility is located.

The type reflects the primary purpose of the server and the index is used to distinguish between instances of the same type of server at the same site. The index has leading zeroes applied to ensure that all the indexes used at a site are the same length. Typically, on each manufacturing site, there can be either two instances of some server types or a large number of other types; determined by the overall capacity required to meet the product volumes forecast.

All hostnames only contain lower-case letters, digits and hyphens: the latter makes parsing the hostname trivial and allows other test infrastructure component names to be easily derived. Server hostnames are globally unique and therefore all names and IDs derived from them are too.

Test domain IDs

A manufacturing site is divided into test domains. Their ID is formed from the prefix and index of the primary server associated with that domain. The prefix is converted to upper-case letters:

<PREFIX><index>

A test domain is an administrative unit of control within a manufacturing site. Any test domain should be able to support any type of product within any test operation. (There may be restrictions for some types of product that have particular requirements.) A test domain can be designated for: mainline production; engineering development; failure analysis; ORT; and PPA (if required). Each of these require to be performed in clearly defined areas on the manufacturing shop floor.

A test domain is also a soft entity: it contains test environment software and their configurations.

NETWORK ARCHITECTURE

In the test environment, there are three levels of networking and two levels of servers. A server is connected to two levels of network; with all servers connected to a shared network in the middle level.

Servers

The test infrastructure contains a number of physical servers. These do not provide services directly; rather it is the virtualised servers that they host that provide all the services necessary to execute test processes.

Top-level servers

There are at least two top-level physical servers that host the same set of virtualised servers; the services that they host are mirrored. Test processes can access any service instance using fail-over and load balancing.

Server room

The top-level servers are mounted in secure cabinets within a secure server room.

UPS

The top-level servers must be powered by a UPS and/or generator.

Test domain servers

A single physical server may host a sufficient number of virtualised servers to serve multiple test domains.

Secure cabinets

There are cable length restrictions on the uplinks between test domain servers and the network switches to which the product instances are connected. Therefore the test domain servers need to reside on the manufacturing shop floor. They must be mounted in secure cabinets.

UPS

The test domain servers must be powered by at least a UPS.

NETWORKS

There are three levels of networking: one that product being tested is connected to; one that interfaces to the internet; and one in the middle that allows all server types to transfer resources and information between them.

Test domain networks

Each test domain is implemented by one or more private (non-routing) networks that connect product to be tested to the test domain services. Each of these networks has a unique network address within the manufacturing site.

A network will be implemented using at least one network switch to which the test domain servers are connected. The network can contain additional different types of network switch that use different connector types and operate at different transfer speeds as the product requirements dictate. The network can also contain serial connectivity devices and power distribution units (PDUs) that allow test processes to control the power to test locations.

It is possible for a test domain to contain additional types of test support equipment that can be connected to the network or by a serial connection.

The network switch and serial devices must not be connected to the same PDUs as the product instances.

Factory network

All the top-level servers and the test domain servers are connected to a common network. This is also a non-routing network.

On this network, a range of IP addresses is reserved to allow the supplier to connect test operator monitor stations. A network connection to the shop-floor control system (interface) is also possible to support the delivery-in of product information (to start a test operation) and the delivery-out of test results (to complete a test operation). Test operators have read-only access to (detailed) test results using a web-browser.

It is possible for this network to span across multiple sites for the same supplier. This allows service mastership to reside at one site only.

Internet connection

The top-level servers are also connected to an internet facing VPN device. This is to allow: resources to be downloaded from parent repositories; test results to be transferred to the centralised test data-warehouse; login access from Oracle; and access to in-process test results using a reverse-proxy service.

At least 14 IP addresses must be made available to connect the top-level physical and virtual servers to the outward facing network to allow resources to be downloaded, test results to be pulled, user login and web-browser access to the test infrastructure.

TEST CELLS

A test cell is the physical manifestation of a test domain. It contains the shelves used to host product while they are being tested. An important feature of a cell is the management of the cables used to connect product to the network, serial and power devices.

A cell has a geometry: this is the number of bays (vertical spaces) each with shelves (horizontal spaces) that form test locations. The notation to express this succinctly has the form:

CG<#*bays* >x<#*shelves*>

A test domain will overlay the cell geometry with a much larger virtual cell geometry. This reduces development and maintenance time (and cost) for sites where a variety of cell geometries are required. Therefore multiple physical cell geometries can be accommodated within a single virtual cell geometry. Test domains make multiple uses of sparse matrix concepts to achieve cost savings across a number of different features.

Test domains use a number of configuration files where the entries are keyed by test location names. Within the source repository these configuration files are represented using templates. A package is built containing instantiated versions of the template files that are associated with each test domain. The package is built and released, associated with the test infrastructure and automatically deployed to each test domain. It is the supplier's responsibility to maintain the source code, along with the package build, release and association with the test infrastructure.

Racks

Most test cells are implemented using racking. Each bay contains a rack and a test location is realised by a shelf within the rack. Shelves are usually removable. There must be sufficient clearance to allow the product to be inserted and removed without the possibility of damage.

Benches

A few test cells (particularly for failure analysis) can be implemented on benches. This means that a bay may actually occupy a horizontal space. The problem with this arrangement is keeping the cable lengths within their maximum length limits.

Cable management

Cable management is a vital part of the test cell design. It must keep power and data cable separate (as much as possible). Allow test operator access to (dis)connect cable to (/from) the product. Route cables to keep them within the maximum length limit and ensure that cable turns are not too tight. The design should also allow cables to be easily replaced (cable connectors and cable ends are subject to wear and tear because of the frequency of connections being made and removed). Cable connectors must not be used.

Networked equipment

The (networked) equipment must reside within the test cell to ensure that the maximum cable lengths are not exceeded. But they must also be positioned in such a way as to prevent the test operators accidentally dislodging cable connections. It is useful for the LED status indicator lights to be easily seen to help analysis any networking issues that occur because of defective cables, connections or test product faults.

Cell design specification

The supplier will be provided with a copy of a version controlled cell design specification document that describes the complete test infrastructure. The document includes:

- Equipment types used;
- Test domain names;
- Geometry of each cell;
- Networked equipment associated with a set of bays or set of test
- locations;
- Networked equipment power and connection types;
- Networked equipment IP addresses;
- Device to test location port maps;
- Cable details:
  - Type specification;
  - Maximum length;
  - Cable colour;
  - Connector type(s);
  - Wire to pin mapping (as required);
  - Labelling and other visual identification;
  - Test location shelf aspect (front or rear).

The supplier is free to implement the test cell as they see fit: provided the requirements detailed in the design document are adhered to.

ACCESS PRIVILEGES

User management is a key feature of the test environment. There are strict policies and procedures applied to control who can do what, where and when.

User names

All users have their own login account to access the test infrastructure. The use of shared accounts and sharing accounts is strictly prohibited. User accounts are never removed: as users create resources and their activities are recorded in log files. When an account is no longer required then it is locked to ensure that resource ownership is correctly preserved.

E-mail addresses, organisation and company details are stored as a comma-separated list with the GCOS field within the password database.

Passwords are subject to maximum age, length restrictions, word analysis and re-use history policies. Accounts that have been idle for a period of time will be automatically locked.

Home directories

Each user has their own home directory. This is mirrored hourly, daily and monthly on a rotational basis, and permanently yearly from the master server to a mirror. The files can be accessed using a web-browser to navigate to the required version.

Group membership

Primary and secondary group memberships are used to control which privileges a user can be granted and to what level.

Primary group

A user's primary group controls which privileges they can be granted and to what level. Note that all users essentially have no privileges within the test infrastructure by default.

Secondary group

Some users can be secondary members of a group. This typically provides them with minimal access to the applications that manage test processes and the resources used to implement and support them.

Production domain access

Most users do not get login access to production test domains. However, it may be necessary to grant access, for a limited period of time, to support failure analysis activities or to support the development of test processes for product types that can only be supported in particular test locations (the product may have physical size and/or weight restrictions and/or require particular networking and power connection requirements).

The supplier has ownership of the policy file that grants access permissions to test domains. This policy file originates in a source repository, and is released and deployed as a package to the server that hosts the user management master service.

Activity monitoring

Access to the test environment is strictly controlled. The privileges and actions of all users is monitored and reported on to ensure the integrity of test processes is assured.

Access report

A periodic report is generated for all users that contains the details of the locked status of their account, the last time that they accessed the test infrastructure, group memberships, home directory usage and the test domains to which they have been granted login access to.

Test domains

The date/time, current working directory and command-lines are logged for each user on all test domains regardless if the domain is designated for production or non-production use.

RELEASES

Releases are immutable entities that each have a distinct identity and defined composition. Releases are built from a set of co-operating packages:

$$\sum \mathit{file} \;==\; \sum \mathit{package} \;==\; \mathit{release}$$

All test process content is deployed within a release. But much of the test infrastructure content is also deployed within a release.

Documentation is an integral part of a release. When the release is deployed a top-level portal is created to allow all the documentation to be accessed using a web-browser. Documentation includes reference manuals, user guides and (accumulated) release notes.

Server releases

All packages used to implement the test process execution and environment are deployed together. There are no package updates. The version and integrity of the server release is included in the detailed test process results.

Previous server releases are not removed (at least until the server is completely re-installed). In the event that a new release cannot be used for manufacturing, then it is possible for a retired server release to bere-activated without the need to re-deploy it.

Server release schedule

The server release is announced on the last Friday in the second month (July, October, January and April) in each quarter. The updated resource files are downloaded to the resource repository on each manufacturing site.

The release deployment instructions are emailed to the test infrastructure support team towards the end of the last month in the quarter. The instructions include the standard procedure to deploy and then activate the release to the (virtualised) servers that host the services that implement the test environment; but the instructions may also include preparation for future enhancements to the environment.

Each supplier must provide an email group (alias) address to which the release announcement and deployment instructions are sent to.

It is expected that the release is deployed and then activated from the start of the next quarter. Depending on the demands of manufacturing or product development activity, it may take some time to complete this on all servers within a site. No updates are permitted in the last month in each quarter. Supplier staff must plan the update: schedule server downtime, apply the update and then resume manufacturing activities.

Test releases

Test releases contain all the software resources (executable, library, configuration, data and other resource files) are deployed dynamically on-demand to each test domain when and where they are required. There is no manufacturing downtime required. These release are automatically removed when they have not been used for at least three months.

Test releases are deployed using an optimised approach to minimise the amount of file-system capacity consumed and the total amount of time required to complete the deployment. Usually many (if not all) of the product instances in test will use the same release on a production test domain. However, it is also possible that every product instance is using a different release; this could be possible in development test domains.

Within a test domain, a unit is guaranteed to use the same release until it completes a test operation. The test release version and integrity are included in the detailed test process results.

A release descriptor can be published by any user that has the necessary group membership.

Each release is accessed through a zone; these contain the rule(s) that determine which product type (along other optional selection criteria) is to use which release.

Production zones

A production zone can only be published by a user who is a member of the supplier's staff. There are policies that determine which users have this privilege. The granting (or revoking) of production publishing privileges is owned by appointed supplier staff.

It is the supplier's responsibility to ensure that the Oracle test and product engineers have created a release that is suitable for manufacturing test. The latter must provide proof of this. Further, the supplier can perform their own validation tests on a release which is a candidate for production.

Non-production zones

A non-production zone is published by a user (Oracle or supplier staff) that has been granted the necessary privileges that the zone's selection rules must satisfy. This usually means creating selection criteria using serial number(s) and/or test location name(s).

Non-production zone privileges are granted and revoked by the responsible appointed members of the supplier's staff.

TEST ARCHITECTURE

All the (software) components that collectively implement the test environment, test system and test processes are classified according to an organisational architecture. The classifications are used in the naming the packages that are created, and they are also used to structure the repository used to store them and make them available for deployment within releases.

DELIVERY-IN/OUT

The delivery-in/out system interfaces the supplier's shop floor control system to the test system to start and complete test operations when product data is transferred in and test results out.

Test unit

It is often perceived that a test unit is the product hardware. But a test unit is actually the data-structure that represents that hardware. The test system operates on data. (A unit does not actually need to be physically present to allow a test process to be executed using its data.)

A unit's data-structure contains some high-level data such as its serial number, part number and product family. This needs to be augmented with the test operation name and test location name to allow a test process to be selected and executed. The data-structure also contains the unit's BOM (bill of materials): this contains the serial numbers, part numbers, description and other attributes of each component that was assembled to create the product instance. The BOM can also contain references to software components such a firmware levels and customer software images.

Delivery-in

Delivery-in is the process of starting a test operation by the transfer of product data from the shop-floor control system to the test system. The supplier can either implement their own delivery-in system or provide an interface to their shop-floor system to allow the provided system to interact with it.

There is a data-type definition used by the test system that details how the data is to be structured. There is a standard set of data names that should be used; but the test system is capable of being configured to translate the supplier's names into those used by the test system. The supplier is responsible for configuring these translations.

The interface must prevent units from being restarted in a test operation without completing an existing operation.

Delivery-out

Delivery-out is the process of completing a test operation by the transfer of test results from the test system to the shop-floor control system. The test system returns the results as a selection of <*name*>=<*value*> pairs. The supplier can augment the configuration to select additional values and/or rename them.

Again, the supplier can provide their own delivery-out system, or use the provided system. When using the latter, the appropriate interface must exist to support this. When a test operation has been completed, then the unit is released to allow another (test) operation to be started.

NAMING CONVENTIONS (CONTINUED)

Server hostnames determine test domain IDs. The latter in turn determine test location names, and these are used to form test client hostnames. IP addresses are assigned for each of the latter.

Test locations

Each test domain contains a set of test locations. Product instances to be tested each need to be associated with one. So a test location provides a physical space in which an instance resides; but it also provides the resources required to test that instance. Resources include both hardware and software. Hardware resources include power, network and serial connections. Software resources include hostnames and IP addresses, but also the access to control power and establish serial connections.

Test location names are formed by extending the domain ID.

Bay ID

A test location resides within a vertical space called a bay. Each bay within a test domain is identified by a single upper-case letter.

A hyphen is used to separate the domain ID from the bay ID:

<PREFIX><index>-<BAY>

Shelf index

A bay is divided into shelves. These are two-digit indexes and are numbered from top to bottom within the bay. The index is appended directly onto the bay ID:

<PREFIX><index>-<BAY><shelfindex>

To support some types of product, some physical shelves do not need to exist. Therefore, it is possible for multiple locations to occupy the same physical space.

Slots

Some test processes work on collections of sub-assemblies. These are not tested individually. Rather, the test process performs a virtual assembly on the collection to create a functionally complete entity. This entity resides in a test location, but each sub-assembly is associated with a slot. Each slot is formed by extending the base test location name with a unique suffix; the chassis slot location is usually used for this purpose. This is required to associate each sub-assembly with a unique test location name.

A slot ID is appended directly onto the test location name:

<PREFIX><index>-<BAY><shelfindex><slot>

There are no resources associated with a test location slot. The execution of the test process is still associated with the base test location name.

Virtual test locations

Virtual disassembly is possible for entities that contain multiple functional entities; each requiring hostnames and IP addresses to allow them to be tested. The base test location name can be extended with a unique suffix that reflects the location of the entity within its parent.

<PREFIX><index>-<BAY><shelfindex>-<relativelocation>

Test client hostnames

Each functional entity has access to multiple client hostnames and their IP addresses. Client hostnames are derived from the test location name (converted to lower-case letters) and appended with a numerical index (separated by a hyphen) to create a unique hostname:

<prefix><index>-<bay><shelfindex>-<clientindex>

Test processes can further append a suffix that represents the purpose or instance of the entity represented by the hostname:

<prefix><index>-<bay><shelfindex>-<clientindex>-<function>

IP addresses

Each test client is assigned a set of IP addresses that are associated with the hostname set. Client IP addresses are unique within a manufacturing site. Test processes are responsible for adding and removing entries from the local hosts database.

DOCUMENT HISTORY

REV	DATE	DESCRIPTION OF CHANGE	
A	12 Nov 2009	Initial Release	
02	22 Sep 2014	Update from Sun to Oracle, removed Webdocs references, updated all links in Reference Document and Records table	
03	14 Feb 2022	Complete update to all content removing obsolete applications and tools, using the current corporate template	

- When Document Template is complete, email source file to eso_business_docs_us_grp@oracle.com
- All hard copies of this document are uncontrolled and are to be used for reference only.
- For questions or comments about this document, please send an email to: eso_business_docs_us_grp@oracle.com