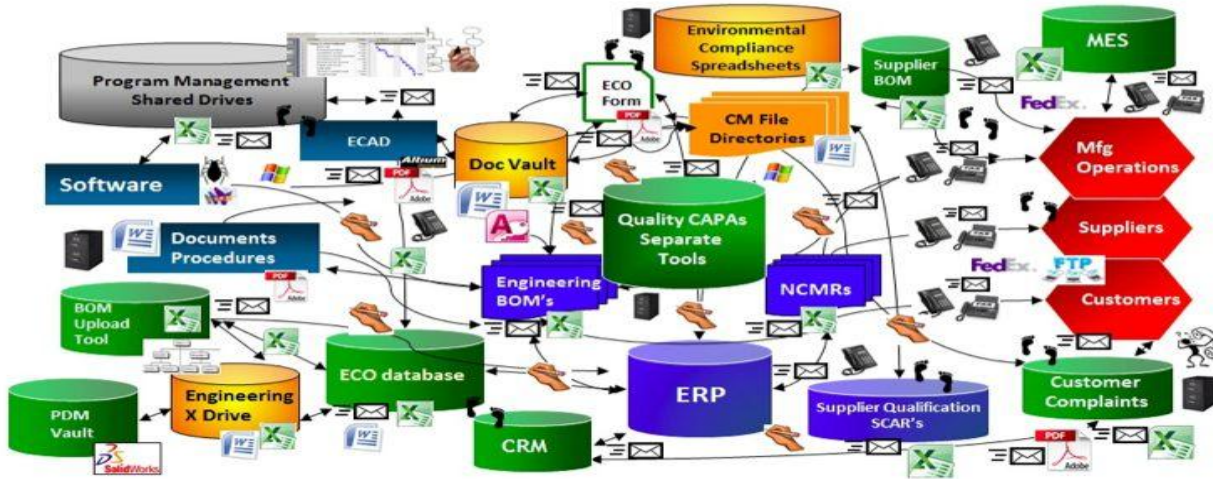


IT or Application or Test Environments Management and Terminologies.



IT or Software Application Environments Management is probably one of the most misunderstood concepts in terms of the part it plays and its huge significance across Quality Assurance as a whole, its contribution to reduced spending cross the overall software development cycle across Enterprise Organisations and its contribution to the overall success of an IT Project or Program.

Most Program, Project and Test Managers in running Projects sooner or later come across IT Environment related issues that can severely delay the delivery of their Project. Here is an attempt at defining IT or Application Environment Management or Test Environment Management (TEM) in its entirety.

IT or Application Environment Management or TEM service clearly contributes to improving the overall quality of IT Software development and support through the lifecycle and encompasses a set of best practices proposed to provide an effective, end to end management service for test software beds/platforms or environments or development beds/environments.

The software test bed or development environment could consist of a client server application, Relational Database Management System (RDBMS), middleware, interfaces, daemons, customised processes (written in any software programming language), FTP utilities etc.

Functional test phases such as Unit, Integration, Acceptance, all manner of Performance or Non Functional Testing and development phases all require IT or Application Environments.



The primary clients of an IT Environments Management Function are the Software Project and Test teams.

An end to end IT Environments Management service will cater for the following;

- Manage the creation, build, upgrade and support for all test and development Application Environments.
- Clearly defining auditable processes of allocating Application or IT Environments, multiple bookings or shared usage, code upgrades, service level agreement, support, decommissioning and re-allocation.
- Manage data refreshes, collating test data and assist in the anonymising of production or other sensitive data if necessary.
- Supply, provision and manage all Application or IT Environment Requirements from the Project and Test teams all through the software development cycle of a project.
- Assist the Project in establishing their application environment requirements, provide expert knowledge on an IT environment's set up, connectivity and serve as a guide to the projects in using the environment in the most efficient manner.
- Review and contribute to the Project Initiation Document (PID) ensuring that the IT Environments Management function and its deliverables are clearly defined and captured.
- Create and maintain project plans to assist in managing all activities required to successfully carry out major code upgrades to all Application Environments.
- Provide reports on usage/utilisation, availability, forward planning and schedules.



IT or Test Environment Management is clearly an emerging area which has arisen due to the following reasons:

- The increased Test and Development Environment requirements for many companies who have several software projects running at any one time.
- The increased levels of interfacing and connectivity between several systems in most organisations also known as spaghetti. For example in some companies more than thirty systems are interfaced or connected with each other exchanging files and data flows etc and has meant that any changes to one system most times could require a change to many others and requiring large numbers of test and development IT environments.
- Increased awareness and more commitment to carrying out rigorous software testing especially with more companies opting to use the Prince 2 methodology and ITIL Framework

A typical IT Environments Management Tool such as [Omnium Light](#) should be able to provide the following services:

Capture Environments bookings and allocation; manage or de-conflict multiple bookings and reoccurring bookings.

Provide reporting on usage, availability, interconnectivity or interfacing environments, utilisation and conflict reporting etc.

It must also serve as a repository of all information (CMDB) on IT Environments to include Host Server names, Hardware Type, Operating System, IP Address and Interfaces if any, code levels and versions.

The ideal background for Application Environments Management personnel could be Software Development, Application or Technical Support, Infrastructure Project Management, Configuration and Release Management etc but must be exposed to at least the ITIL Framework, Client? Server development, System Architecture/Design, Networks, TCP/IP and messaging systems etc.

Terminologies Defined and Explained:

Application or IT Environment or Test Environment Management (TEM)? A single test bed or development platform instance of a software application or system that can also be used for all



manner of functional and non functional testing or could be the production instance (production environment). It could also be large, medium or small which normally refers to the size of data the RDBMS will be holding depending on the type of testing it is required for.

Integrated Application environments (also known as stripes): More than one Application or IT or Test environment connected to each other also communicating with each other and exchanging files and data flows. Connections could be via Microsoft ODBC, via FTP, TCP/IP, daemons, middleware, defined interfaces, database links, more recently via API's etc.

Anonimising or scrambling of data: Anonimising of data refers to the manipulation or transformation of production data held in the RDBMS such as Oracle, SQL Server, Sybase, Microsoft Access, DB2 etc to be used in a test or development Application Environment ensuring that for example real names, addresses, date of birth, bank account details and other sensitive information or data is transformed to dummy data.

The data is transformed whilst still maintaining its defining characteristics in a Relational Database Management System table such as character length (Char 25 or Varchar 50) etc to ensure its usage in testing or development is not compromised and that the integrity is maintained. For example a valid name such as John Smith defined as Char 10 will now be updated in the table to become possibly a unique character string XXXXXYYYYY (comprising of ten characters including the space between John and Smith).

Pipe cleaning: Pipe cleaning caters for the all the activities required to be carried out before a test or development environment is handed over to the Test or Project team and includes disk clear down, archiving and purging logs, importing test data, killing off rogue processes, resetting passwords, changing environment settings, end to end connectivity or integration tests to make sure everything is working okay.

Depending on the complexity of the system a checklist of activities may be required and ticked off capturing all the checks and tests that have been completed on an environment or an integrated suite of environments prior to its hand over to a Project or Test team.

Smoke Test: A smoke test describes an initial end to end test of all the integrated or even stand alone environments very possibly using dummy data and carried out by the support teams who have created or built the environment or by the test team when the environment is handed over.

Stubbing an environment – Since IT or Application Environments can be quite expensive to build or provision depending on its complexity and level of integration. For Unit Testing, FAT and possibly other forms of testing, a decision can be made to provide a stubbed environment which



could simply be a folder or directory that data is passed to during testing instead of having a fully fledged instance of an IT or Application Environment particularly if there have not been any changes to the specific application.

Valentine Wats – CTO & Founder-TEMS INC (www.temscorp.com)