

# Performance Tools

## Performance Monitoring Tool

- *Do you have a transaction system, such as web servers, a data warehouse, transaction system using IMS, CICS, Amtrix, Tuxedo, etc. on which you have based your business?*
- *Do you know what the availability of your system is?*
- *Do you have a way of continuously detecting performance degradation in you system?*
- *Do have any way of altering your operations staff in case your service becomes unavailable.*

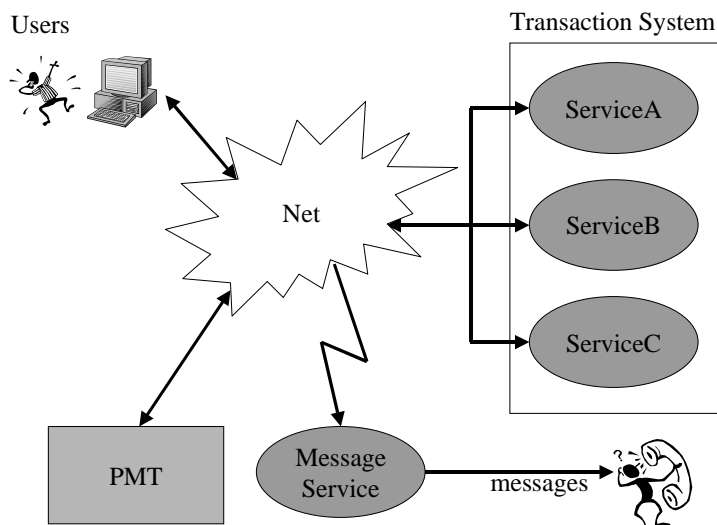
*If you don't - please read further.*

The Performance Monitoring Tool (**PMT**) is an application that continuously test your transaction services in a round robin fashion. It takes care of three major areas:

- Monitoring and alerting in case of failure
- Performance measurements
- Availability statistics

## Component overview

The typical scenario can be pictured like this:

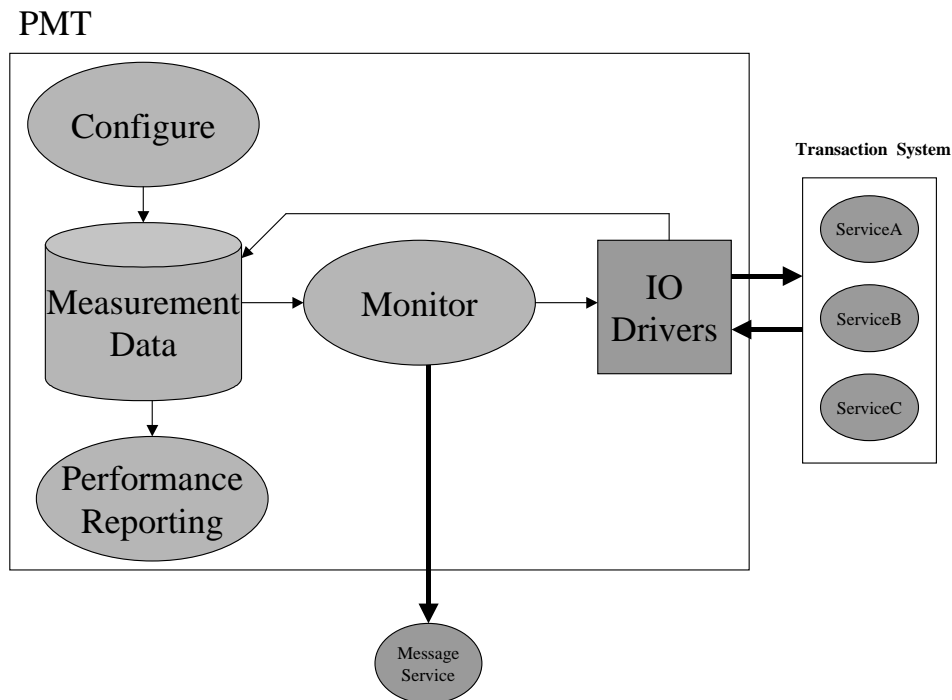


Users are connecting to your transaction services through the net (typically internet or intranet).

You want PMT to connect and test the services in the same fashion as your endusers do in order to measure end user availability and performance.

In case you have an internet transaction service, you could easily place the PMT application at a standalone monitoring machine externally on the web. You could then configure it to call the SMS Service component in case of failures - and let your operational staff be alerted.

If we take a closer look at the PMT application (at the left bottom area above), it can be broken down into:



Typically, you will go through the following tasks:

1. Building the IO drivers. These are modules that interface with your transaction system. These are typically around 20-30 lines of C-code and based on the provided source code skeletons, utilizing the PMT package APIs, which includes socket connectivity and external module execution.
2. You build your message service plugin and configure the filtering mechanism in the PMT application.
3. You start the monitoring task through the use of crontab or your preferred scheduler.

**And you are set to go.**

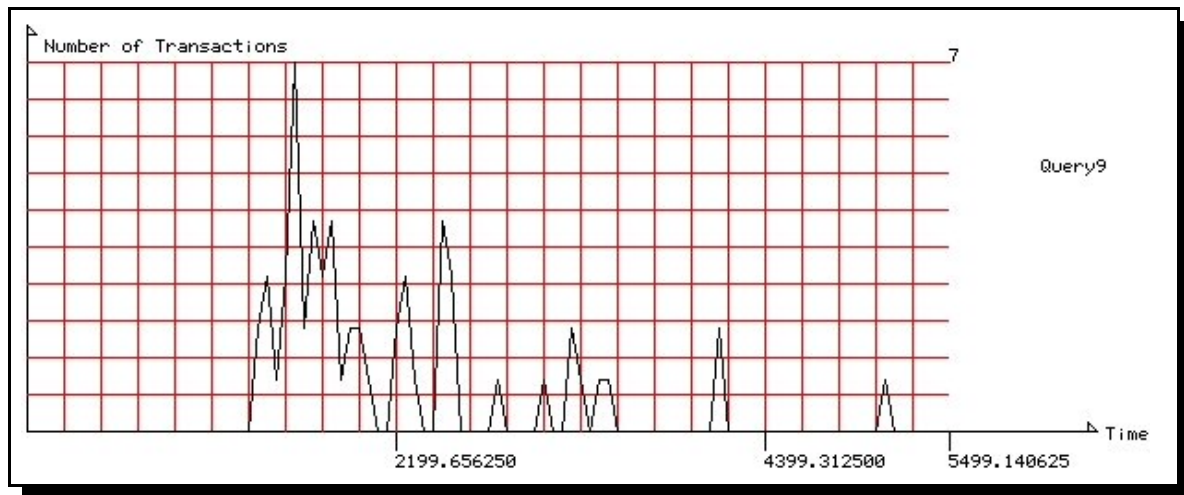
All interfacing to the PMT is carried out using the simple-to-use web front-end.

PMT will now collect and store all input and output from the transaction, along with elapse times, initiation time, etc. An error report is generated automatically based on your input and parsing in the IO driver. All this information is stored in the database and will trigger alerts based on the output and the behavior of the transaction. You are now capable of monitoring, detecting a degradation and calculate the availability of your system, all measured from and enduser (and your customers) point of view.

## Analysis

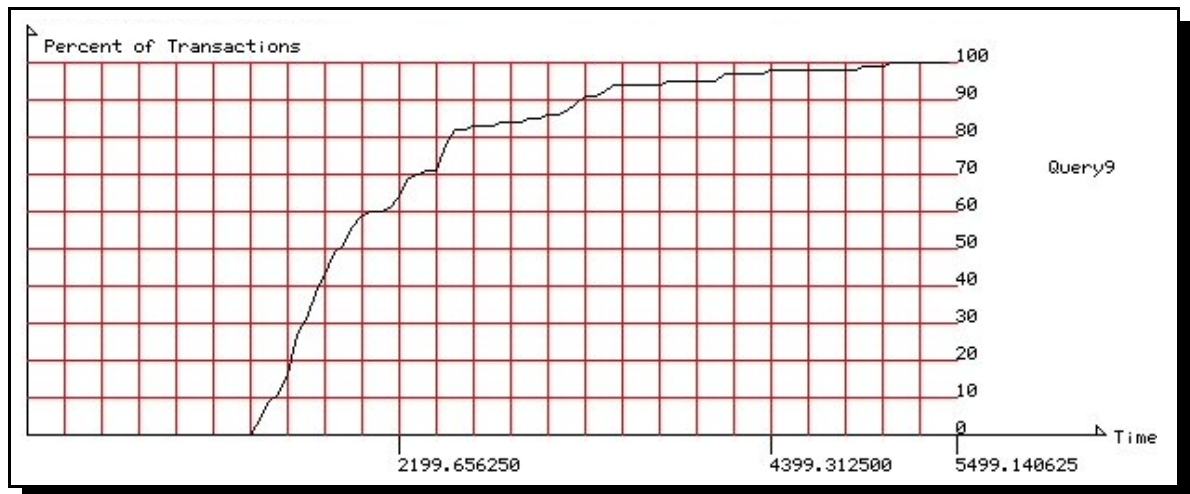
First, a bit of background information:

Usually, analysis tools for performance measuring usage will display the Probability Density Function (PDF), this is looking like below:



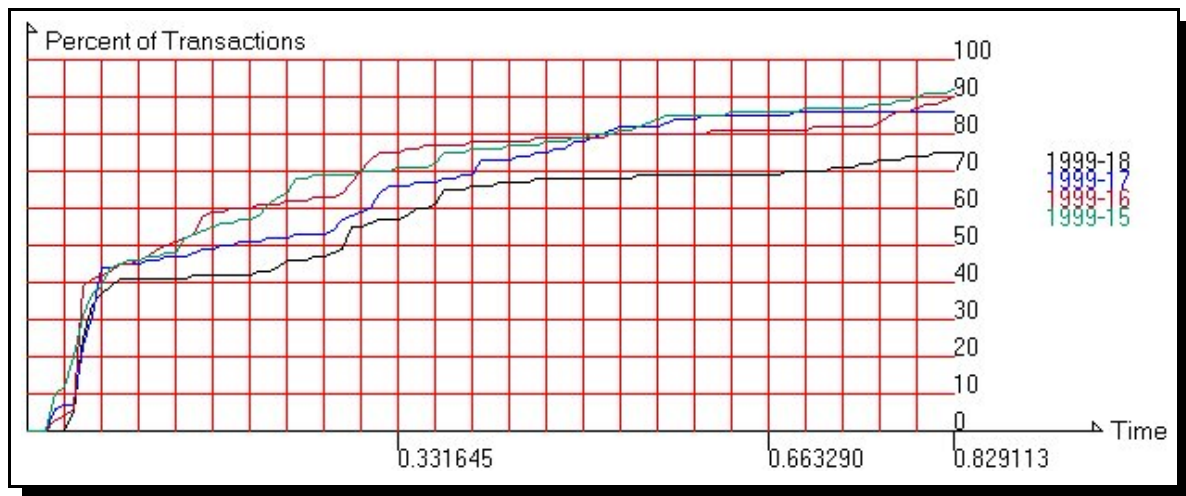
However - creating multiple PDF plots for comparison purposes, will (at best) result in a fuzzy picture, and it will be hard to make a clear judgment on whether or not performance has improved or degraded. The integral of the PDF called the Cumulative Distribution Function (CDF) is usually a much better way of comparing multiple runs.

A single CDF is displayed below (actually it is the integral of the above function):



The graphs simply displays the percentage of transactions (Y-axis) that have completed before the elapse time (X-axis).

Over time it has been proven that this is the best way to objectively compare performance measurement results. Therefore PMT supports a comparison report facility for multiple previous measured results (apart from also being able to generate the previous plots). This is pictured below:



### Platform requirements

PMT is installed on a Intel based PC with RedHat Linux 6.1 (free of charge) as the operating system. The Linux basis software package has a large share of the server market, where it has been proven to be a reliable and stable server platform. IBM Universal Database (UDB) is used as the database holding all information. License for UDB may have to be obtained seperately from IBM. DB2 for Linux is a superior database system, where pricing is very reasonable.

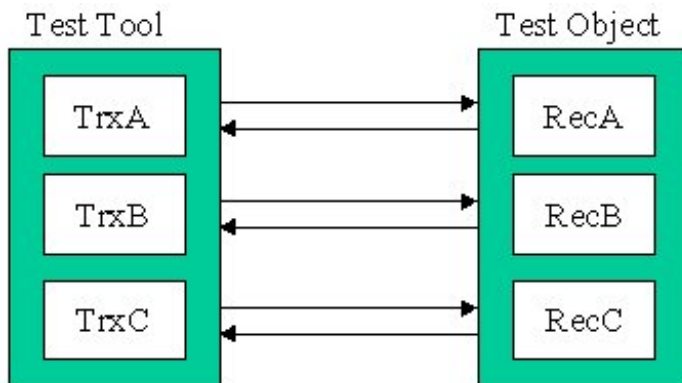
All transaction tests will run directly off the PC and towards any platform where a TCP/IP connection can be made.

**All in all PMT is a powerful low cost monitoring system, with measurement and alerting capabilities.**

## Testing Modes

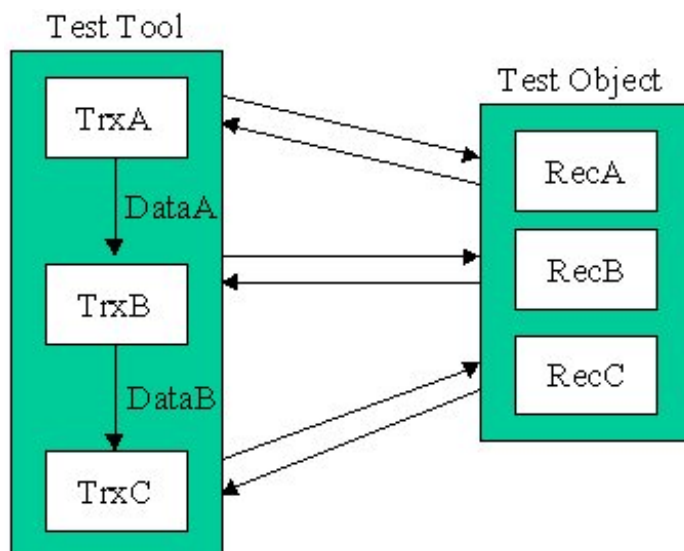
PMT supports two models of transaction testing:

- Independent service requests. This is resembling web pages where no information is carried forward, i.e. the user is simply making simple requests by following hyper links - as an example.



The request (TrxA) requests a single receiver (RecA) and likewise for the others.

- Dependent services. The output of a previous service request should be stored and be reused at a later point in time.



Here the requester (TrxA) request a service at the receiver (RecA) and stores the output (DataA) for processing with the successor (TrxB) that take the data as input and requests the service at the receiver (RecB), etc.

***“It is better to know the availability of your system than having the customers calculating determine it for you”.***

## **Purchasing**

Usually HITCON will provide 1 weeks of consultancy with PMT in order to help you install the application, build the first couple of IO drivers, setup your first monitoring scenarios and validate the first results. All which should make you feel comfortable using PMT and make you perfectly able to carry out new configurations and monitor new services.

## **Pricing**

You will receive a single monitor license which supports installing PMT on a single PC in your network, along with a weeks worth of consulting. Product support will also be available for an additional two hours. However, errors cause by the clients misuse of the product will be invoice based on an hourly fee.

Currency	Amount
USD	13,000
DKK	100,000
GBP	8,200
CHF	21,000

Prices does not include travel and expenses.

## **Contact**

HITCON  
Ole Holmskov  
Broedskovparken 5  
3400 Hilleroed  
Denmark

Email: [info@hitcon.dk](mailto:info@hitcon.dk)  
Phone: +45 48 26 48 14

Please use email as the preferred form of communication.