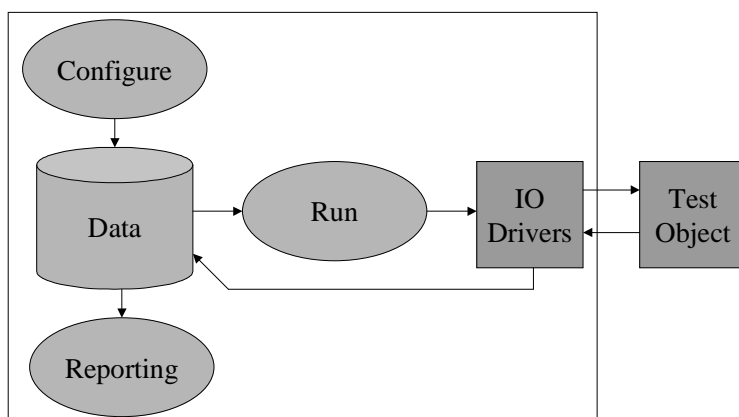


Performance Test Tool

What would you choose: To determine the bandwidth of your transaction system¹ in test mode - or have the customer finding it in real time? Wouldn't it be nice to have a test bench that could verify developed performance improvements objectively? And as a developer, wouldn't it be great to objectively test various implementation models, and recommend the best suited one based on real performance measurements? If you would like to determine and verify the capacity and performance of your transaction system, please do continue reading.

Plug-and-Play Configuration

The Performance Test Tool (PTT) is a flexible plug-and-play application based on the concept of a test engine and test IO drivers. You perform the simple task of configuring your test scenario, and you develop the test IO drivers that will interface with your test object, based on the skeleton ANSI C-code and the Application Programming Interfaces (APIs), that is supplied with PTT. **And - you are set to go.** You can now run the test and generate the reports in order to determine your bandwidth/capacity.



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

The above picture displays a typical flow of a test setup:

- First you build your IO drivers, based on the provided skeletons
- Then you configure your environment
- After that - run the defined test cases
- Lastly - generate the desired online reports

All you need to provide yourself - from a coding perspective - is pictured as green boxes: Your test object (which you already have) and the IO drivers in order to interface with your transaction application (usually 20-30 lines of C-code).

¹ Transaction system is defined as any system where there is “no control” of the request arrival time and the type of service that will be requested.

PTT has already been used successfully in various environments, examples are:

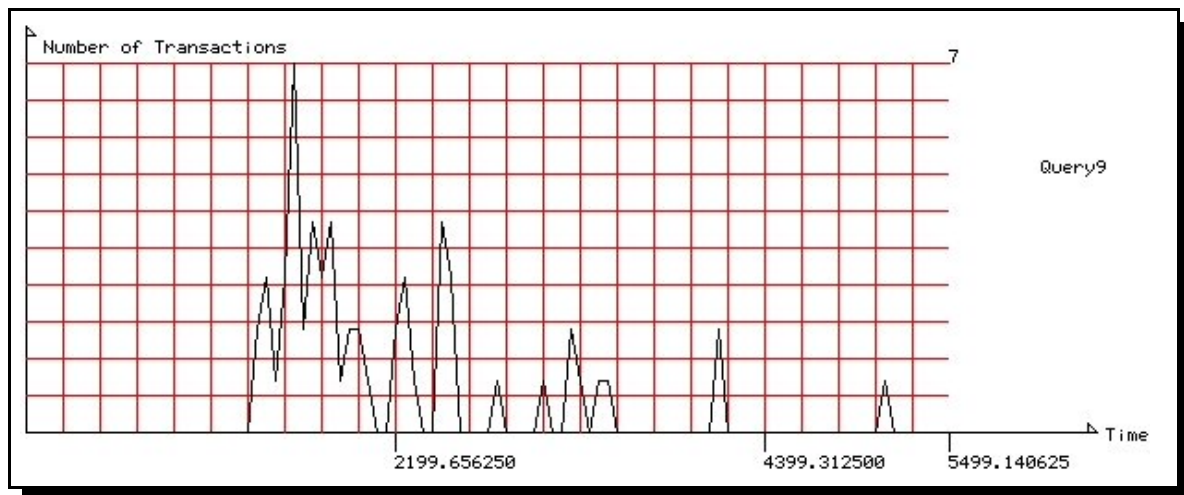
- IBM (www.ibm.com) has used it for testing the performance and capacity of a financial backend system.
- UBS (www.ubs.com) has used it for testing the performance of a datawarehouse.
- Currently PTT is being developed and tested for Web front-end Testing for a US based customer.
- WAP (Wireless Application Protocol) performance testing will also be carried out in a near future project.

It proves that the combination of a test engine, interface drivers and your test object **works**.

PTT replicates real end user load, i.e. user service requests will randomly arrive over time, and services will be randomly requested. PTT lets you specify the services, the percentage split, the number of test-transactions executed and the sample elapse time. Within your specified time frame, it will randomly “slot in” the individual transaction initiation time. Furthermore, it will randomly assign a service to the transaction execution time. This directly resembles the end user load. Of course, you can then map the capacity to your own actual number of end users.

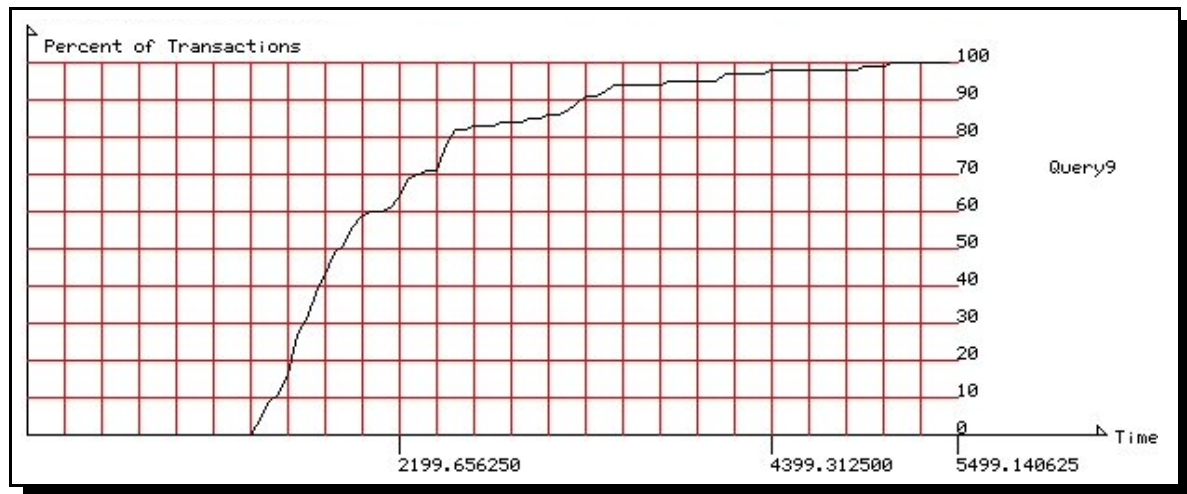
It collects and stores all input and output from the transaction, along with elapse time, real starting time compared with the requested start up time, etc. An error report is generated automatically based on your input and parsing in the IO driver. All this information is stored in the database.

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



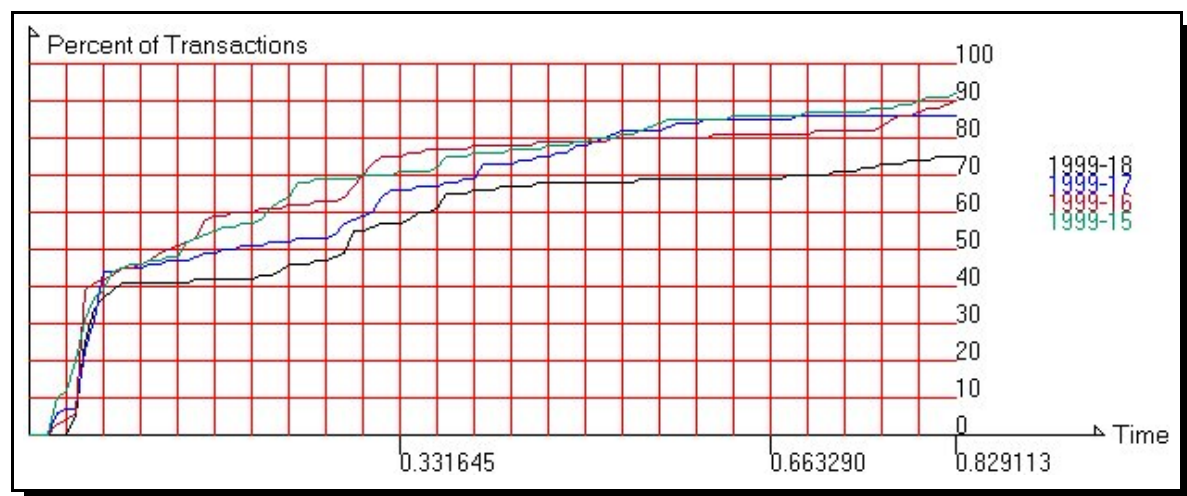
However - creating multiple PDF plots for comparison purposes, will (at best) result in a fuzzy picture, and it will be hard to make a clear judgment on whether or not performance has improved or degraded. The integral of the PDF called the Cumulative Distribution Function (CDF) is usually a much better 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 PTT supports a comparison report facility, which is able to generate multiple plots of the measured results for various runs. This is pictured below:



Platform requirements

PTT 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 separately from IBM. DB2 for Linux is a superior database system, where pricing is very reasonable.

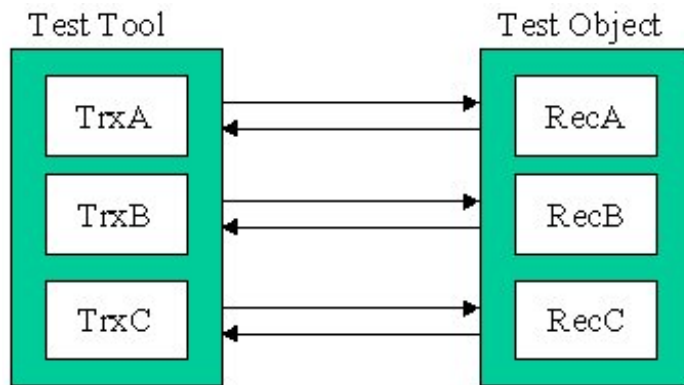
All tests will run directly off the PC and towards any platform where a TCP/IP connection can be made.

All in all PTT is a powerful low cost test bench, with generic testing capabilities.

Testing Modes

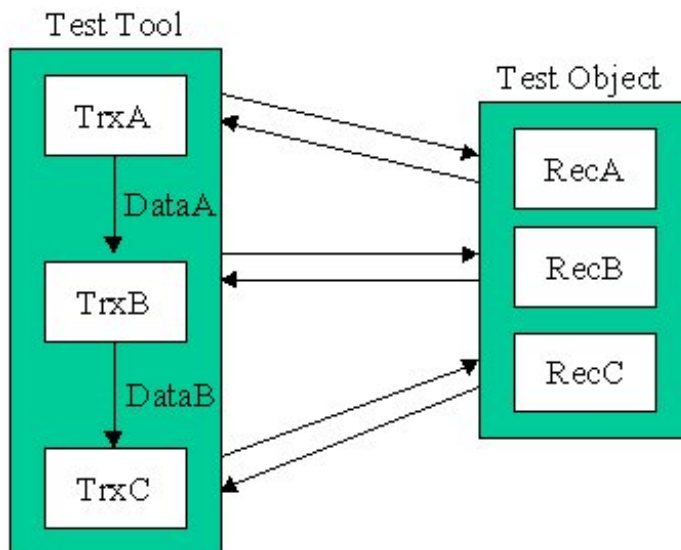
PTT supports two modes of transaction testing:

- 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 service requests. The output of a previous service request should be stored and be reused at a later point in time. This resembles an online web dialog where user selections is being carried forward.



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.

Load levels

PTT has been measured to >20.000 transactions an hour on a Pentium II processor with 380 MB RAM, with 4 services with elapse times from 3-13 seconds. Scalability can be obtained by adding multiple CPUs (SMP) and/or more memory (RAM).

PTT has brought a multinode SMP AIX SP system to a halt, just based on load. Furthermore whole clusters of AIX SP nodes and MVS configurations has also been overloaded.

So - PTT is the perfect choice for your needs, if you are looking for **the** test bench for your business transaction system.

“It is better to know the capacity of your system, rather than letting your customers determine it for you”.

Purchasing

Usually HITCON will provide 1 week of consultancy included with PTT in order to help you install the application, build the first couple of IO drivers, setup your first test cases and validate the first results. All which should make you feel comfortable using PTT and make you perfectly able to carry out new test cases and new setups.

Pricing

You will receive a single test license which supports installing PTT on a single PC in your network, along with one week worth of consulting. Product support will also be available for an additional two hours. However, errors caused by the clients misuse of the product will be invoiced 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
Phone: +45 48 26 48 14

Please use email as the preferred form of communication.