



TESTNETWORK OVERVIEW

TestNetwork is a platform for automating scalability, functionality and reliability tests of Internet-ready application software, including Web Services. This document describes the architecture, design choices, and example uses for TestNetwork.

June 10, 2003
(c) 2003 PushToTest. All rights reserved.
Approved for public distribution.

web www.PushToTest.com
email info@pushtotest.com
phone 408 374 7426

KEY CONCEPTS

PushToTest delivers technology and services to solve scalability, functionality and reliability problems in complex interoperating systems, especially Web Services.

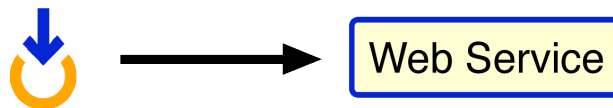
In a Web environment, users drive functions and exchange content using a variety of standard protocols (HTTP, HTTPS, SOAP, XML-RPC, SMTP, POP3) from any location, using any Internet software (browser, Internet enabled application, embedded devices) at any time. In a TestNetwork environment, intelligent test agents implement user behaviors and drive the system as users would. TestNetwork enables enterprises to deploy test agents anywhere within the system, anywhere on the company network or on the Internet, and in any desired volume to test for scalability, functionality, reliability and security.

As a result TestNetwork enables businesses to automate system tests to check for correctly operating functions, to handle increasing user loads, and continuously monitor systems to proof service levels.

UNDERSTANDING TESTMAKER AND TESTNETWORK

PushToTest publishes two solutions: TestMaker and TestNetwork.

TestMaker is a popular free open-source utility and framework that software developers, QA analysts and IT managers use to build and run intelligent test agents. Intelligent test agents implement the typical behavior of archetypal users: The agents click, read, and act like real users by issuing requests through standard protocols (HTTP, HTTPS, SOAP, XML-RPC, SMTP, POP3) directly to a service.



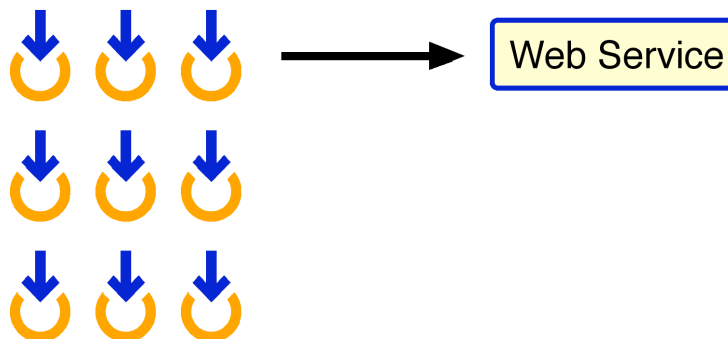
Sign-in
Place an order
Find an employee
Check status
Download HR documents
Send message

TestMaker offers the following choices to run test agents:

Run a single test agent	TestMaker checks a system for correct function.
Run multiple concurrent test agents	TestMaker checks a system for scalability.
Run test agents continuously over time	TestMaker monitors system availability and quality-of-service.

Many businesses need to run test agents that exceed TestMaker's abilities. They need to:

Problem	TestNetwork Solution
Run agents in greater scale than on a single machine.	For example, on a single 1 GHz Pentium system TestMaker runs out of bandwidth and resources at approximately 500 concurrently running test agents. With TestNetwork, 10,000 to 100,000 or more concurrently running test agents is possible.
Run test agents on multiple remote machines.	TestNetwork turns remote systems into TestNodes that remotely run test agents and report the results back to a central console.
Keep test agents running for a long duration.	TestNetwork's console/TestNode architecture turns test agents into mini-servers that are able to handle operations autonomously.



PushToTest meets these needs by publishing and selling TestNetwork, a commercial product.

BENEFITS

TestNetwork offers these benefits:

- Reduce costs of testing new software builds
- Increase speed of deployments
- Reduce costs to monitor quality of service
- Inexpensive solution to automate tests
- Single solution supports multiple protocol tests
- Easy effort to extend to support new and custom protocols
- Reduce costs by supporting multiple platforms (Java, C, C++, C#)
- Easy support of developers, QA analysts and IT managers with multiple levels of skills and experiences

FEATURES

TestNetwork offers these features:

Distributed client/server architecture

TestNetwork architecture uses proven client/server topology. The TestNetwork client software plugs into TestMaker as a master console. The console stages tests by identifying the available TestNodes, launching test agents on the TestNodes, and displaying the test results in graphs and charts. TestNetwork TestNode software runs on any Java enabled machine or device.

Loosely coupled Master/TestNode interface using SOAP and WSDL

TestNetwork is designed to support new protocols, new techniques and new infrastructure as the Web environment changes. The TestNetwork console communicates with TestNodes using Web Service protocols (SOAP and WSDL.) This allows TestNetwork to operate across networks and firewalls in your existing Web environment using HTTP protocols.

Java, C, C++, VB.NET Implementations

Currently, PushToTest ships TestMaker as a Java application. PushToTest intends to ship TestMaker .NET in 2003. The new .NET software will enable test agents to be written in C++, VB.NET, and C#.

Multiple Protocol Testing

TestNetwork intelligent test agents speak native protocols (HTTP, HTTPS, SOAP, SMTP, POP3, IMAP, XML-RPC) to drive a system. To the system it appears that a user with a browser, Internet-enabled application, email client or Web Service-enabled system is making a request. Plus, TestNetwork provides an easy framework for adding additional protocols and custom dialects of existing implemented protocols. PushToTest intends to add new protocols (ebXML, SAML, Liberty, JXTA, JMX, SNMP, Telnet) to TestNetwork over time.

Plug-in Architecture

Intelligent test agents use a scripting language (Jython) and the Test Object Oriented Library (TOOL) library of Java objects to implement user behavior. TestNetwork fully supports plug-in modules to add new features to the scripting language. TestNetwork also provides an extensible Java API to add new protocol handlers to TOOL.

Results Analysis

The TestNetwork console consolidates the results of agents running on TestNodes into a single log file. The results may be optionally displayed to a graphic display panel as the test progresses. Results logging is extensible and may be programmed for custom log results by using simple test agent script-level commands.

Friendly Environment for Developers, QA, IT

TestNetwork test agents are written using a scripting language (Jython) and library of Java objects to enable software developers, QA analysts and IT managers with a variety of programming experience to be immediately productive in writing and maintaining test agents. Most test agents use script-level code to define *user* behavior and the Test Object Oriented Library (TOOL) implements the protocol handlers.

Commercial and Open-Source solutions

Unlike the free open-source license for TestMaker, the TestNetwork product is a commercial solution requiring a license to be purchased. The license includes packaged software ready to run plus TestNetwork source code and the ability to compile bug fixes and improvements yourself. The commercial solution provides a stable revenue source from which PushToTest may viably continue to achieve its business objectives.

Commercial-grade Support.

TestNetwork software is backed by PushToTest technical support and PushToTest Global Services. Technical support agreements provide commercial-grade support to answer your questions by phone, in email, and in-person. PushToTest Global Services provides all the needed expertise, knowledge, tools and technicians to answer your most important questions about TestNetwork and your systems.

EXAMPLE USE SCENARIO

TestNetwork is typically used in one of these ways:

- Functionality test
- Scalability test
- Service level Monitoring

Let's look at how TestNetwork is applied in a service-level monitoring solution.

New software technologies, datacenter sizing, changing bandwidth needs, network bottlenecks and system failures can give any IT manager a really bad day. Automated monitoring solutions are an essential part of deploying high quality Web services. To meet your test automation needs PushToTest uses the TestMaker and TestNetwork technology to deploy End-To-End Monitor (ETEM) solutions.

Consider the needs of one IT director. His company manufactures chemicals that are sold by distributors. The chemical manufacturer uses Web services (UDDI/WSDL/SOAP) to publish their product catalog information to their distributors. The distributor system will receive the catalog data, add their own HTML formatting and pricing and present a catalog page to the end customer.

The system architecture uses the chemical manufacturer's existing database system in New Jersey, the Web Service is hosted in California and the distributor's systems are located in Texas.

The IT director has several needs:

- *Scalability and performance testing to evaluate the Web service software and the data center needs.* Using the TestNetwork software intelligent test agents check the entire system for scalability and performance by simulating thousands of concurrent requests. TestNetwork software enables a rack of 40 small servers to be a matrix of 40 TestNodes. The TestNodes are commanded by the TestNetwork console to run 100 concurrent test agents in each TestNode for a total of 4000 concurrent test agents.

- *Check that the Web service is quoting accurate information.* The intelligent test agents must interface directly with the company database system and check the Web service price quotes to make certain the pricing is accurate. The test agents running on each TestNode implement the behavior of the user and system. The test agent requests a price quote and validates the response. When an invalid response returns – or when no response comes – the test agent signals the failure to the TestNetwork console for action.
- *Proof that each of the service providers is meeting minimum service level commitments.* Intelligent agents monitor the database systems in New Jersey, connectivity to the distributor in Texas and the Web Service host in California. TestNetwork provides a single place to view system status and problem reports.

The combination of TestMaker and TestNetwork technology provide a powerful means to achieve automated Web service monitoring solutions today.

HOW TO BUY

Contact PushToTest today to buy TestNetwork. Please call (408) 374-7426 (USA, Pacific Time) or send email to sales@pushtotest.com. Further details are at www.PushToTest.com.