

[Environment](#)[Cycles](#)[Testing](#)[WHQL](#)[Verifier](#)[Contact](#)

Users sometimes request information about the development and testing environment of the UPDD Software Suite. This document is aimed at satisfying these queries.

Our software is [event driven](#) meaning that a central reactor process (reacts) to events as they occur. This model allows for the central reactor to handle the complexities and synchronisation of thread and context processing and resource allocation to minimise the problems that can arise (at there worst - system crashes!).

### Source and configuration file control

All files associated with our drivers; source modules, configuration files, client configuration, customisation data and hardware technical information are held within a file and source control system called [Perforce](#). This software controls every aspect of file and source updates and forms the backbone of our file management system.

### Development environment

The following development tools are used in the development of UPDD

#### Windows Visual Studio 6.0

Microsoft Visual Studio is the main [Integrated Development Environment](#) (IDE) from Microsoft.

#### Windows DDK

The Windows Driver Kit is a software toolset from Microsoft that enables the development of device drivers for the Microsoft Windows platform.

#### QT 3.3.6 GUI development tool

[Qt](#) is a cross-platform application framework. Using Qt, you can write applications once and deploy them across many desktop and embedded operating systems without rewriting the source code.

#### ACE 5.4.6.2 IPC

The ACE Inter Process Communication development tool provides a rich set of components that perform common communication software tasks across a range of OS platforms.

The above tools form the backbone of the driver development. The resultant source code is compiled in other, non Windows desktop, OS environments such as Windows CE, Linux and Mac OS X.

### Development cycles

We implement 4 development cycles as follows:

Suffix	Meaning	Description
A	Alpha	Alpha for new development
B	Beta	Beta for bug fixes only
R	Release	Interim release – Customer requested release
P	Production	The latest stable release placed into production

The development cycle is implemented to ensure a good known version is available and yet cater for bug fixes and ongoing development. When a release is considered stable and beta testers have approved the software and want a production version the software is placed into production e.g. version 4.1.3P where P denotes a production release.

When a release is placed into production debug symbols are logged with our production server (to be used in the event of any crashes that result in system dumps being sent for analysis) and two development branches are created – Alpha to allow for the continued product development and Beta for bug fixes only. E.g. 4.1.6B and 4.1.8A. Should any issues be discovered in the production version that needs immediate attention then the fix will be applied to the Beta version leaving the Alpha cut to be used for ongoing development and bug fixes. Should a customer require a new, intermediate release the current beta will be cut into an interim release (e.g. 4.1.6R) and frozen. The beta release is still retained for further fixes as required.

Development and bug fixes will continue in the alpha version until such times as it is deemed stable and the new functionality is ready or required for release. At this point a production version is cut and the cycle repeats.

### Driver testing

As with most software it is difficult to test for every scenario, especially with driver software that can be affected by external and environmental factors. In addition, UPDD is a feature rich driver that is written to cater for a wide variety of system configurations. Consider that UPDD supports 100s of touch controllers, USB, serial and PS/2 interfaces, PnP, Power Management, many different OS, multi-monitor and multi touch, auto rotate (both software and hardware induced), local and eeprom based calibration, many different calibration algorithms, multiple languages and many touch related features. The driver ships as an installable module or in embedded component form for different OS. In Linux it caters for many distributions. Given this we believe it is literally impossible to test all scenarios We believe that to promote a 'test plan' may induce a feeling of well being that in practice would fall way short of the desire to have a fully tested driver at all times.

Taking this in to consideration our driver testing works as follows:

As stated earlier, we at all times have a fully tested production version available and a copy of the production driver waiting to address any issues that may arise with the production version. This allows us to concentrate and test any bug fixes building up to an interim release in the knowledge that we are working with a current 'known to be good' production version.

Ongoing changes that are made to alpha software will vary from release to release and may be quite major, say to cater for new OS requirements or minor, such as improvements to the GUI and during this time software may be delivered to test certain new features that will not have been fully tested and may cause issues, particularly if changes are being made to complex kernel functions such as PnP or Power Management issues.

At any one time we will have a number of customer beta testers reporting back their finding and at the same time we conduct internal tests. Our internal tests are performed on 2 reference systems, one running XP and one running Vista, both English language. We also run on-demand tests for other OS and language systems as requested.

Once a driver is placed in to production the change log is created to show all the changes made to the release and this can be viewed in the UPDD Console, About dialog, View change history.

#### **Windows Hardware Quality Lab test (WHQL)**

The WHQL tests are a set of Microsoft tests used to comprehensively test drivers and associated hardware. We have a WHQL test environment based on the WHQL DTK (Device Test Kit) hosted on Windows Server 2003 with Vista 32 and 64 bit systems as test clients.

These tests are run against approved hardware and UPDD kernel software to ensure the driver passes all WHQL tests. When run with approved hardware, the WHQL logs, recording all test phase passes, are submitted to Microsoft to generate an appropriate digital signature which, once embedded, allows us to distribute a signed driver for specific hardware.

These tests stress test the driver at many different levels and ensures that the driver is fully conformant to MS driver standards.

#### **Microsoft Driver verifier**

As part of our standard test procedure we also use the Microsoft Driver Verifier as described at <http://support.microsoft.com/kb/244617>.

Driver Verifier is included in Windows 2000, Windows XP, and Windows Server 2003 to promote stability and reliability and the tool is used to troubleshoot driver issues.

#### **Contact**

For further information or technical assistance please email the technical support team at [technical@touch-base.com](mailto:technical@touch-base.com)