

How to get operating system version information (part 6 of 6)

Demo program, summary, references and feedback

Demo code

A demo program to accompany this article is available for download.

The demo includes the complete `UOSInfo` unit we developed in the course of the article, along with a simple application that displays information about the operating system hosting the demo application. The result of each public method of `TOSInfo` is displayed, along with the values of the original and extended Delphi RTL `Win32XXX` global variables.

System Information Unit

You may also wish to look at my *System Information Unit* that contains similar code to that presented in this article.

The code was developed using Delphi 7 Professional but should compile with Delphi 4 and later.

This source code is merely a proof of concept and is intended only to illustrate this article. It is not designed for use in its current form in finished applications. The code is provided on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. The source code is released under the same Creative Commons license as this article – see the bottom of this page for details. If you agree to all this then please download the code using the following link.

[Download the demo code](#)

Summary

In this article we examined how to get information about the underlying operating system.

We began by looking at the Windows API and Delphi support for operating system version information that is common to all platforms and started development of a static class – `TOSInfo` – that encapsulated this information.

It was observed that Windows provides extended information about NT operating systems from NT4 Service Pack 6 onwards. As a result the Delphi RTL support was extended to account for this information by defining a new set of global variables. `TOSInfo` was also modified to provide additional information based on this data.

Next it was discovered that additional information about NT 4 up to service pack 5 was available from the registry. We updated `TOSInfo` to access this information.

The final step in developing `TOSInfo` was to detect XP Media Center and Tablet editions using the `GetSystemMetrics` API call and to check for 64 bit Windows using the XP only `IsWow64Process` function.

Finally a demo program was made available that provides a full implementation of all the code developed in the article along with a front end that displays version information about the underlying operating system.

References

The following web pages were used while researching this article:

- ▶ *Windows Versions* by Philippe Majerus – (page no longer available).
- ▶ *Getting the System Version* (Microsoft® Developer Network Library)

Feedback

I hope you enjoyed this article and found it useful. If you have any observations, comments or have found any errors please *contact me*.

This article is copyright © Peter Johnson 2005-2006



Licensed under a *Creative Commons License*.

Copyright © Peter Johnson (DelphiDabbler) 2002-2020