ASPOSE ®
The .NET & Java Component Publisher TM

Case Studies

Aspose Total.Net and Total Java

Bangladesh Election Commission – Nikosh Bangla Unicode Converter

Aspose.Total Implementation Case Study

By Mohammad Ashraful Anam, 17 July 2009

Product Background / Overview

Until Unicode arrived, many languages like Bangla did not have any standard implementation. Unicode solved the coding scheme for us and standardized most languages. But most countries like Bangladesh and India has already hundreds of products that let users write their native languages. How was this achieved? These language were coded in normal ASCII characters and a keyboard interface written to translate each character to its latin equivalent. Many fonts were developed as well. While these hack and slash schemes worked and were able to show and print Bangla correctly, each company had their own coding implementation. So each product was incompatible with another. A word document written with Product A and Font A1 would show gibberish when used with Product B and Font B1. This went one for more than 10 years. Unicode solved this problem, but what happens to the hundreds of document written in non standard ASCII encoding now that Unicode is here? We need to convert those documents to standard Unicode. Hence, "Nikosh Bangla Converter".

Requirements Scenario

It is quite easy to write a generic converter that will convert ASCII coded Bangla text to its Unicode equivalent. A simple text file containing hundreds of pages can be converted in less than a minute. The problem arises when we try to convert different MS Office documents. The MS Office Automation of PIA implementation was just too slow and it many cases it breaks the formatting. We tested the two leading currently available Bangla converters. The test document was a 32 page Bangla document containing text, image, table and header and footer. One of the converters took 4.5 minutes to convert and the other 56 minutes! Even at 4.5 minutes, it was too long. If a document takes 5 minutes on average to convert a single document, it would take just too much time to convert hundreds of documents. We needed something that would reduce this time to less than a minute and at the same time, preserve all the styles. Also, converter for only MS Word was available; there were absolutely no converters to convert MS Excel or PowerPoint documents.

Solution Implementation

I started looking for all the available solutions. At first I coded using Word Interop and as previously mentioned it was just too slow. I started looking for other available technologies. I found Aspose and Tx Text Converter. Both looked good. I gave Aspose a go because it looked easier and also it had Word, Excel and PowerPoint support which the other lacked. Then working over the weekend up to 3AM, I was finally ready with my first C# prototype converter (I found a supplied C# sample that provided me a good start). I tested it initially against a single page document and it produced good results. Finally I tested it against the complex 32 pages documents that I had tested with the other two products with and the results were very impressive. Not only did it convert very fast but also very accurately and without breaking any kind of formatting. After perfecting MS Word conversion technique we finally made converter for MS Excel and PowerPoint also.

Benefits

The chart below shows the performance comparison of our "Nikosh" implementation and the other converters available on the market. The speed increase is just amazing.

Sl	Page count	Words	Characters [With space]	Lines	Avro		Shabdik		Nikosh	
					Conversion Time	Accuracy	Conversion Time	Accuracy	Conversion Time	Accuracy
1	3	1,787	10,110	212	15 Min 3 Sec	96.77%	5 Min 16	98.38%	**0.25 Sec**	100%
2	45	10,412	66,970	2589	3 Hr 45 Min 35 Sec	97.41%	39 Min 21 Sec	96.32%	**5 Sec**	99.45%
3	42	15,997	1,025,71	2059	4 Hr 49 Min 15 Sec	94.78%	56 Min 35 Sec	96.45%	**9 Sec**	99.66%
4	82	8,972	76,691	6,007	3 Hr: 12 Min 17 Sec	92.21%	27 Min 05 Sec	97.28%	**4 Sec**	99.83%
5	30	15,268	102,189	1,255	4 Hr 35 Min 03 Sec	97.69%	1 Hr 02 Min 45 sec	96.96%	**8 Sec**	98.75%

After creating and distributing the converter for free for several months, we have finally come to a point where everyone is asking for a converter which will run in Linux. The best part is that Aspose has almost the same product lineup for Java and we have been able to reproduce our program for Linux using Aspose Total Java.

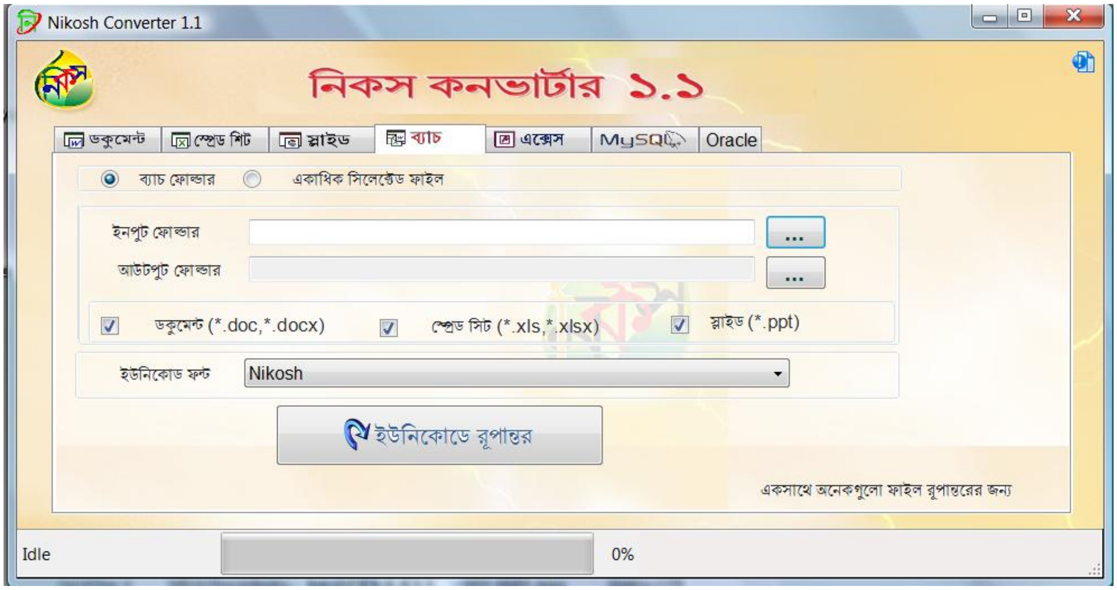
Future Implementations

Now that we are able to convert all MS Office legacy documents, we would try to squeeze even more performance out of Aspose products and make our converter even faster.

Conclusion

I was really amazed that I was able to produce a fully working product in less than two days work. And the speed at which I was able to perform all MS Office document operation was just amazing! You just can’t go wrong with this product!!!

Splash Screen



● ব্যাচ ফোল্ডার ○ একাধিক সিলেক্টেড ফাইল

ইনপুট ফোল্ডার

আউটপুট ফোল্ডার

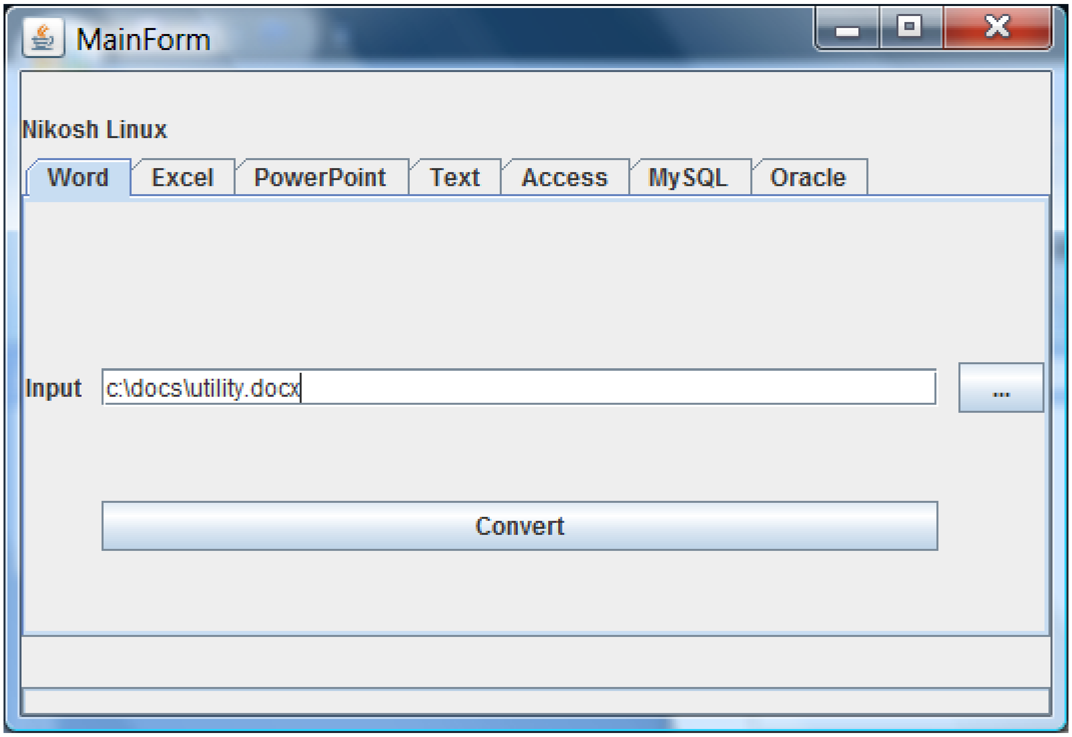
☑ ডকুমেন্ট (*.doc,*.docx) ☑ স্প্রেড সিট (*.xls,*.xlsx) ☑ স্লাইড (*.ppt)

ইউনিকোড ফন্ট: Nikosh

একসাথে অনেকগুলো ফাইল রূপান্তরের জন্য

Idle 0%

Batch Conversion Screen (.NET version)



Input c:\docs\utility.docx

MS Word document conversion screen (Java version)