

# Time Plan about implementation of Hyphenation support in AbiWord

Chen Xiajian

Date	Plan to do
Week 1	<p>1 Read more about Enchant and get familiar with the use of Enchant in AbiWord (this step will make an abstraction of the hyphenation algorithm)</p> <p>2 Search and find more libraries which can be used to determine the breakpoints within words. Includes:</p> <ul style="list-style-type: none"> <li>➤ Some TeX formatting system</li> <li>➤ Open Source libraries</li> <li>➤ If possible pull out just the code we need from some package.</li> <li>➤ Care about the language factors(besides English)</li> </ul> <p>3 Learn more about Hyphenation algorithm</p> <p><b>Deliverable 1:</b> Make a summary about Enchant and the open source hyphenation libraries Choose one most suitable for us</p>
Week 2	<p><b>(continue and finish)</b></p> <p>1 Some programming test about Open Source hyphenation libraries</p> <p>2 Make a code framework about where to add code(to make code more flexible)</p> <p>3 <b>Design the data structures of Hyphenation</b></p> <p>4 Building the <b>hyphenationChecker</b> using open source library in Enchant hyphenationChecker is the name I define in Enchant to deal with hyphenation</p> <p><b>Deliverable 2:</b> 1 Summary Document 2 Patch about data structures of Hyphenation 3 Make an <b>patch</b> about hyphenationChecker</p>
Week 3	<p>1 Fix possibly remaining bugs to have a good foundation for the further.</p> <p>2 Read source code and place hyphenation points called from the Enchant The codes in src/text/fmt/xp/*.h,cpp</p> <ul style="list-style-type: none"> <li>➤ fb_LineBreaker.h,cpp and</li> <li>➤ fb_Alignment.h,cpp</li> </ul>

	<b>Deliverable 3:</b> 1 Make an patch about implementation of hyphenation points 2 Fix remaining bugs
Week 4	1 Draw the hyphenation characters in AbiWord 2 Find more good dictionary to have a good coverage of different language and words  <b>Deliverable 4:</b> 1 Make an patch about preliminary results 2 Make a summary of dictionary
Week 5	<b>(continue and finish)</b> According to Martin, it seems that it's a difficult task. Time is needed. 1 Have more tests 2 Again, Fix possibly remaining bugs 3 Prepare some document that Google required in Mid Evaluation  <b>Deliverable 5:</b> 1 Fix remaining bugs 2 Some document that Google require
July 15	Mid Evaluation
Week 6	1 Care about some special languages and special words  <b>Deliverable 6:</b> 1 Have an more good result to support more language and words
Week 7	Rejoin hyphenated words as they get moved via the users editing. Codes: <code>src/text/fmt/xp/fl_BlockLayout.cpp</code> More tests  <b>Deliverable 7:</b> Patch about implementation of this function More tests
Week 8	1 deal with the result: misspell words are correctly handled after hyphenation. 2 Test and Refactor  <b>Deliverable 8:</b> 1 Patch about implementation of this function 2 better version

<b>Week 9</b>	<b>(continue and finish)</b> 1 Fix new or remaining bugs 2 Some documents what GsoC needed
<b>Week 10</b>	<b>(continue and finish)</b> <b>Time for emergency</b>
<b>August 16</b>	Suggested 'pencils down' date. Take to scrub code, write tests, improve documentation, etc.
<b>August 26</b>	End