2014 IEEE International Conference on Control System, Computing and Engineering, 28 - 30 November 2014, Penang, Malaysia

BMTutor Research Design: Malay Sentence Parse Tree Visualization

Yusnita binti Muhamad Noor[1] and Zulikha binti Jamaludin[2]
School of Computing
College of Arts and Sciences
Universiti Utara Malaysia
06010 Sintok, Kedah
(s92715@student.uum.edu.my[1], zulie@uum.edu.my[2])

***Abstract*—This paper discusses the research design for BMTutor. BMTutor is a prototype for visualizing Malay sentences that is combined with sentence checker, sentence correction and word attribute components. The purpose of BMTutor is to check sentence validation, provide sentence correction for invalid sentence used and produce parse tree visualization. The research design involved can be divided into four phases; categorizing sentence and produce repository (Phase 1), developing models and algorithms (Phase 2); development of a prototype (Phase 3); and prototype testing (Phase 4). To date, this system is the only one designed with the functions and characteristics as in BMTutor. There are two BM parsers to check the validity of simple BM sentences had been developed. Both parsers performed three phases in research design, namely 1) the collection of sentence or CFG, 2) develop a prototype, and 3) conduct evaluation. The phases involved are the basic method in developing a prototype. As a result of the lack of models and algorithms have been introduced in both parsers, the model and algorithm development phase is introduced in the design of BMTutor. Output from the development process shows that the prototype is able to provide sentence correction for all 15 invalid sentences and can produce parse tree visualizations for all 20 sentences used for prototype testing.**

***Keywords*—Malay sentence checker, Malay sentence correction, Parse tree visualization, Bahasa Melayu Tutor**

I. Introduction

In Malay language (BM), lack of knowledge in understanding grammatical sentences in the community especially among school students is nothing new [1,2]. Grammar issues among school students are due to the difficulty in understanding the grammatical structure of Malay sentences [3]. Moreover, school students face difficulty in understanding Malay sentence structure because they do not understand how to use the correct word class, and this difficulty will then affect their writing skill [4].

According to the BM syllabus for primary school student-Elementary School Integrated Curriculum (KBSR) and for secondary school student-Secondary School Integrated Curriculum (KBSM) as written in Kementerian Pendidikan Malaysia [5]) and mentioned in Ismail [6], students are taught about the rules of grammar and sentence structure since in primary school. The article by Abd. Talib [7] stated that learning grammar and sentence structure are more focused in secondary school. Students are taught the method of writing an essay that requires them to learn proper sentence structure as the goal of teaching in the secondary school is a long-term goal compared to the primary school. In addition, secondary school students also focused on mastering the language as a preparation for the higher level. However, there are some students that do not understand the grammar until they finished school. The students also faced a problem in forming a correct sentence and they cannot differentiate the type of word classes, which can be seen from a given essay writing [4, 6].

This paper highlights the research design for a prototype in visualizing BM sentence in parse tree namely BMTutor, which is combined with sentence checker, sentence correction and word attribute components. BMTutor is developed with the aim to help Malay speaker, especially secondary school students to explore and learn about the structure of BM sentence through learning about phrase structure formation and the word attribute through a computerized visualization method. Parse tree visualization is used in explaining a sentence structure because this method is also used by linguists in understanding the structure of a sentence. It is approved as a tool to increase the use of BM, as well as to help the community in using BM better, especially students.

However, as it can be seen today, the study of BM or explanation about BM sentence structure is done manually using paper-based method. If the explanation in understanding the sentence structure is computerized and combined with visualization techniques, hence this will become more attractive. As described in Almeida-Martinez, Urquiza-Fuentes and Velzquez-Iturbide [8], the use of computerized visualization techniques will enhance the students learning process, which makes the method as a part of their learning experience. Besides, students will have a better understanding in visual learning as the method will help students to create the mindset models of a concept [9].

Besides, the lack of emphasis researchers has given to process the Malay language (BM) has been described in Ab. Aziz [10], Bakar, Salaebing, Salleh, Rodzes and Ishak [11], and Ramli [12] in articles about computational linguistics and natural language in Malaysia. Thus, in making BM more progressive in computer-based processing, BMTutor is introduced. The method used in BMTutor can also contribute

978-1-4799-5686-9/14/$31.00 ©2014 IEEE

to other research-based natural language processing, such as research on semantics.

To date, there are two BM parsers to check the validity of simple BM sentences had been developed. Parser as in Abidin, Yong, Kasbon and Azman [13] aims to check the grammar of sentences and parser as in Abd. Rahim [15] aims to reduce the ambiguity problem in BM sentence structure. Both parsers performed three phases in research design, namely 1) the collection of sentence or CFG, 2) develop a prototype, and 3) conduct evaluation. The phases involved are the basic method in developing a prototype. Similarly, in this study, the same basic method is also performed. However, this study emphasizes the development phase of the model and algorithm in line with the main focus of the study. As a result of the lack of models and algorithms have been introduced in both parsers, the model and algorithm development phase is introduced in the design of BMTutor.

Next, Section 2 discusses the research design of BMTutor, Section 3 highlights the design methods used in the prototype development, Section 4 discusses about the result of the development process, and Section 5 presents the conclusion of the study.

II. Research Design for BMTutor

In the following Fig. 1, Phase 1 shows the categorizing sentence phase to produce a choice of sentences for the design of models and algorithms. The model and algorithm design phase involves designing models and algorithms such as the model for word attribute components, which are the combination of word class, word derivation, translation, image and sentence examples. Moreover, the purpose is to design an algorithm for sentence checker by performing search and analysis for BM syntax. Sentence correction algorithm is also needed in this phase for invalid sentence entered by the user according to the rules. In the third phase, a prototype is developed. The last phase is a reliability testing that assesses the reliability of the models and algorithms developed based on the descriptive analysis. The flow of activities involved in the development of BMTutor is shown in Fig. 1.

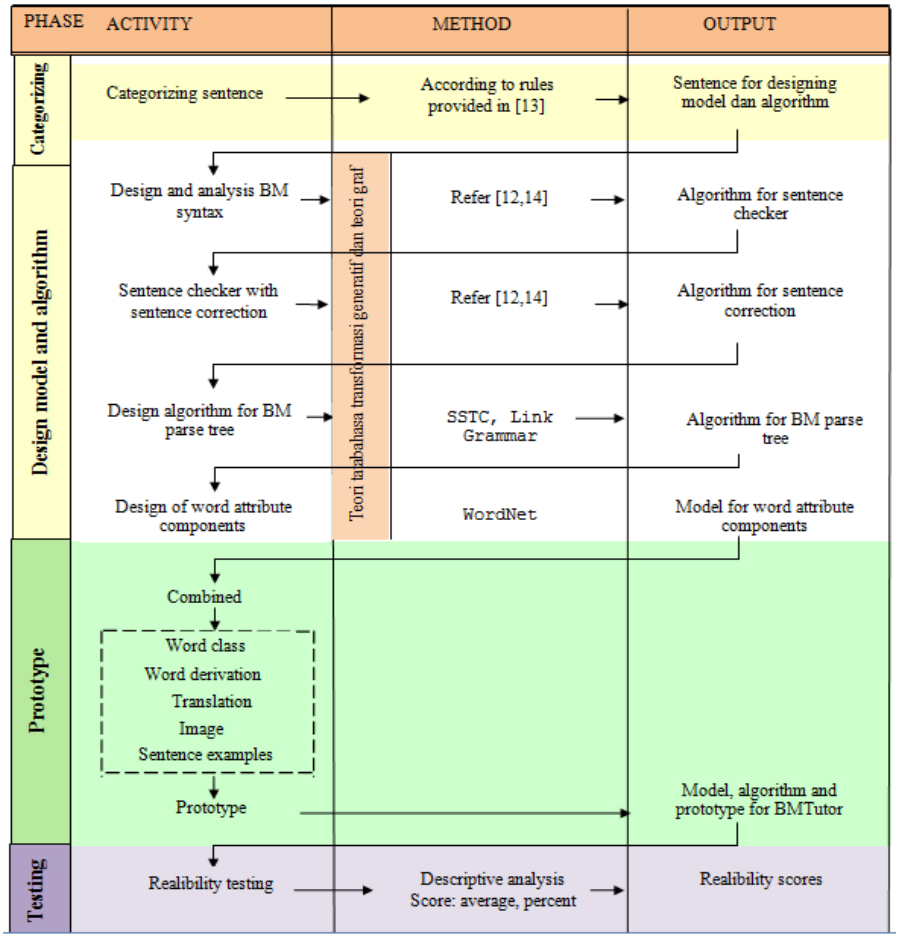


Fig. 1: Research design for BMTutor

Phase 1: Categorizing sentence and produce repository

The first phase was categorizing BM sentences within the scope of the study. 12 activities were involved, begin with the selection of sentences, collecting sentences, selection of a complete sentence, sentence separation, sketching the syntax tree, syntax verification, and ended with the preparation of the repository. The activities involved were divided into three processes. The first process was the collection of sentences to produce syntax rules and obtain words and sentences to be

used in the prototype development. The second process was the syntax rules verification by BM experts to ensure that the rules used are correct. The rules used in the prototype development are important to determine the reliability of the developed algorithm and model. The third process was the process of collecting repository containing all of the words, sentences and syntax rules.

Phase 2: Developing models and algorithms

Referring to the literature review involved as shown in Fig. 1 (method category), the models and algorithms involved in previous studies served as a guide. Some of the processes involved in the production of models and algorithms were sketching an interface, designing models, generating a process of analyzing the sentence and giving sentence correction, as well as producing algorithms.

Phase 3: Development of a prototype

By referring to the models and algorithms, a prototype was developed to ensure the development of the models and algorithms fulfilled the parse tree visualization design. The prototype was developed using the Python programming. The program is commonly used by other researchers to analyze the structure of sentences.

Phase 4: Prototype testing

The reliability of the developed models and algorithms was tested based on the score value obtained. This score was obtained using descriptive analysis of the average, middle (median) and the percentage.

The activities involved in the research design of this study are shown in Table 1:

TABLE I. ACTIVITIES INVOLVED IN THE DEVELOPMENT OF BMTUTOR

Phase	Phase 1	Phase 2	Phase 3	Phase 4
Activity	• Choosing types of sentences • Collecting sentences according to the research scope • Choosing complete sentences within the scope • Separating sentences according to the number of words • Sketching parse tree • Verifying syntax • Collecting CFG (context-free grammar) • Categorization • Collecting valid CFG • Collecting word attribute components • Creating repository for word • Creating repository for CFG	• Sketching interface • Developing model for word attribute components • Developing model for sentence parse tree • Developing an overall model • Developing a process flow for sentence analysis and recommendation • Developing pseudo code • Developing BM sentence checker algorithm • Developing an algorithm for BM sentence structure correction • Developing an algorithm for parse tree visualization • Developing an overall algorithm	• Designing interface • Creating prototype programming codes	• Reliability testing

III. BMTUTOR DESIGN METHODS

Research design methods as explained can produce a prototype known as BMTutor. The methods involved in the development of BMTutor are shown in Fig. 2. It can be divided into three processes; parsing, suggestion and parse tree processes. After receiving the input, the parsing process is started by tokenizing the sentence into words until the rules matched. The suggestion process plays its role when there is a problem in matching the syntax structure. Lastly, the parse tree is produced for correct grammatical sentence, as well as the word attributes page and the parse tree visualization for the sentence example.

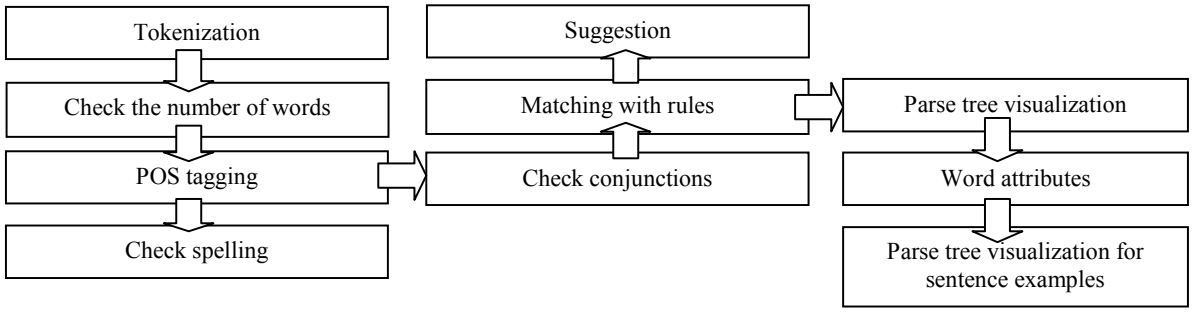


Fig. 2: Methods in visualizing Malay sentence parse tree

The processes involved in Fig. 2 are described as below:

1. Sentence is tokenized into words.
2. Sentence conditions are reviewed to ensure that the input sentence is more than one word.
3. Each word is matched with the appropriate word class as provided in the repository.
4. For unmatched words with appropriate word class, the spell-checking process checks if the word is categorised under special noun words, date, number, or address. If the spell-checking process still cannot assign the appropriate word class, an error message will be displayed. Next, the sentence condition is checked to determine that the sentence is not a compound sentence by checking the conjunctions. If there is a conjunction in the word lists, the system produces a warning message because only basic sentence will be processed.
5. The determination of the validity of the input sentence is conducted by matching the structure of the input sentence or the order of its word classes with context-free grammar (CFG). The CFG produced as in Karim, Onn, Musa and Mahmood [16] is used as a reference. When the matching succeeds, the steps proceed by displaying the output. Otherwise, the proposed method of sentence correction will be carried out.
6. In the suggestion stage, similar CFG for an input sentence is searched according to the order of word classes listed for input and CFG. Only one similar CFG is taken. Replacement is done by changing the position of the words (the input sentence) according to the word class order in CFG that has been retrieved. Hence, the proposed sentence is displayed to the user.
7. For the correct sentence determined by the parser, the prototype displays an output in CFG (order of CFG) and a parse tree visualization. Each node in the parse tree has a link. The node for the abbreviation word classes shows the meaning of the word class, and the nodes for words in the input sentence are linked to a word attributes page.
8. The word attributes page displays the attributes for the selected word. The page consists of the word class, word derivation, translation of word in English, an image, and a list of sentence examples. Each sentence example has a link to the new parse tree visualization page.

IV. Output from Prototype Development

To test the prototype developed, a total of 50 types of sentences were selected randomly from the Malay textbook for Form 1 student to collect the rules involved. The rules were obtained by making paper-based parse tree sketch of each sentence. Testing results are evaluated according to the methods used by Abidin, Yong, Kasbon and Azman [13] and Abd. Rahim [15] that depends on the rules and comparisons made with the sketch of the parse tree that was made during the categorizing phase.

a) Test invalid sentence

The prototype was tested by using 15 invalid sentences. The position of the words for the collected sentences was changed to produce invalid sentences. The results obtained indicated that the prototype can produce a good output with the correct sentence correction for all 15 sentences. This study does not focus on semantic aspects. A sentence with invalid use of semantic element is considered a correct sentence if it follows the rules provided. The results are shown in Table 2.

TABLE 2. Suggested sentences produced to the users for invalid sentences entered

No.	Input sentence	Sentence correction	Rules for sentence correction	Right (✓), wrong (✗)	Target sentence	Rules for target sentence
1.	*di sana baginda*	*baginda di sana*	KN KSN KN	✓	*baginda di sana*	KN KSN KN
2.	*di baginda sana*	*sana di baginda*	KN KSN KN	✓	*baginda di sana*	KN KSN KN
3.	*baginda sana di*	*sana di baginda*	KN KSN KN	✓	*baginda di sana*	KN KSN KN
4.	*ditangkap penjahat*	*penjahat ditangkap*	KN KK	✓	*penjahat ditangkap*	KN KK
5.	*membantu kami sedia*	*kami sedia membantu*	KN KK KK	✓	*kami sedia membantu*	KN KK KK
6.	*membantu sedia kami*	*kami sedia membantu*	KN KK KK	✓	*kami sedia membantu*	KN KK KK
7.	*saya ayah dengan*	*ayah dengan saya*	KN KSN KN	✓	*saya dengan ayah*	KN KSN KN
8.	*dengan saya ayah*	*ayah dengan saya*	KN KSN KN	✓	*saya dengan ayah*	KN KSN KN

9.	*itu makmal saiz*	*saiz makmal itu*	KN KN PENT	✓	*saiz makmal itu*	KN KN PENT
10.	*di sini baru saya*	*saya di sini baru*	KN KSN KN KA	✓	*saya baru di sini*	KN KA KSN KN
11.	*sini saya di*	*saya di sini*	KN KSN KN	✓	*saya di sini*	KN KSN KN
12.	*saya sini di*	*saya di sini*	KN KSN KN	✓	*saya di sini*	KN KSN KN
13.	*di saya sini*	*sini di saya*	KN KSN KN	✓	*saya di sini*	KN KSN KN
14.	*berpengalaman Mahzan*	*mahzan berpengalaman*	KN KK	✓	*mahzan berpengalaman*	KN KK
15.	*berpengalaman Datuk Mahzan*	*datuk mahzan berpengalaman*	GEL KN KK	✓	*datuk mahzan berpengalaman*	GEL KN KK

The abbreviations used in Table 2 are shown in Table 3.

TABLE 3. ABBREVIATIONS AND THEIR MEANING

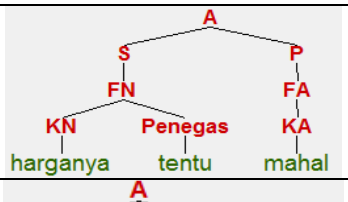
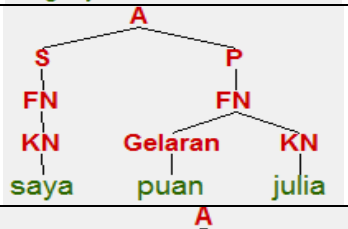
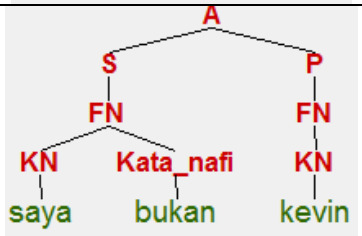
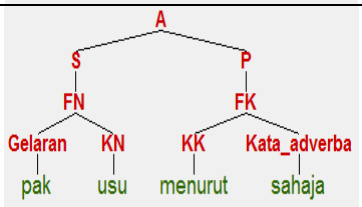
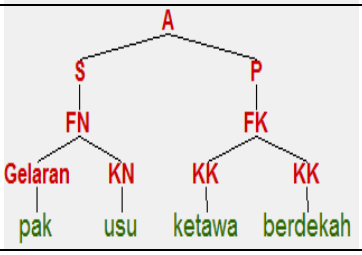
Abbreviation	Meaning
KN	Noun
KSN	Prepositional
KK	Verb
KA	Adjective
PENT	Determinant

For the sentences number 2, 3, 7, 8, 10, and 13, if compared with the sentence examples that it should produce, it produces sentences that differ in order of the words. However, in terms of the sequence of word classes belonging to each sentence suggestion, the sentences meet the rules, thus they are considered correct. Nine of these sentences propose sentence correction according to the target sentence, while the other six produce sentence corrections that are different from the target sentence. The weakness of these sentences is that it can confuse users if the word substitution produces incorrect sentence structure especially in terms of the meaning because the semantic aspect is not considered.

b) Parse tree visualization

20 testing sentences were randomly picked from the 50 collected sentences. The purpose of these testing sentences is to ensure that the prototype is able to analyse a sentence by producing a parse tree if the sentence entered is classified valid. The testing showed that the prototype can produce a good output, and all the sentences entered were successfully analyzed. As an example, Table 4 shows 5 out of 20 parse tree visualizations done by the prototype.

TABLE 4. PARSE TREE VISUALIZATION

No.	Test sentence	Parse Tree Visualization
1	*Harganya tentu mahal*	
2	*Saya puan Julia*	
3	*Saya bukan Kevin*	
4	*Pak Usu menurut sahaja*	
5	*Pak Usu ketawa berdekah*	

The abbreviations used in Table 4 are shown in Table 5.

TABLE 5. ABBREVIATIONS AND THEIR MEANING

Abbreviation	Meaning
A	Sentence
S	Subject
P	Predicate
FN	Noun phrase
FA	Adjective phrase
FK	Verb phrase
KN	Noun
KK	Verb
KA	Adjective

V. CONCLUSION

Research in the language field has grown in popularity among researchers from various countries. Therefore, the design method used in the development of BMTutor can be used to undertake studies related to language processing for any language. BMTutor can be introduced in other language-based studies. The tool will be useful as a sub-tool needed for semantic processing, machine translation and others.

Although in BM, there have been two parsers [13, 15] were developed to check the validity of a sentence, however, components required in BMTutor are not included. Besides, the research design is the basic design in the development of a prototype, which are 1) collecting sentences or CFG, 2) prototype development, and 3) evaluation. Both of these studies did not focus on the models and algorithms for the prototype development. Thus, in developing BMTutor, model and algorithm design phase is added.

REFERENCES

[1] Z. Ahmad and N. H. Jalaluddin, "Incorporating structural diversity in the Malay grammar." *GEMA Online™ Journal of Language Studies*, 12(1), Special Section, 17-34. 2012.

[2] N. H. Jalaluddin, J. Kasdan and Z. Ahmad, "Sosiokognitif pelajar remaja terhadap Bahasa Melayu." *GEMA Online™ Journal of Language Studies*, 10(3), 67-87. 2010.

[3] A. C. Bagavathy," Mengatasi Kelemahan Murid Menguasai Aspek Tatabahasa Dalam Bahasa Melayu Melalui Cara Permainan Bahasa," *Prosiding seminar penyelidikan pendidikan IPBA, 2005*, 50-58.

[4] R. Daing Melebek, "Perubahan struktur kata tunggal Bahasa Melayu mengikut aliran," PhD. Thesis, Universiti Putra Malaysia, 2004.

[5] Kementerian Pendidikan Malaysia. (2003). *Kurikulum bersepadu sekolah rendah, Sukatan pelajaran, Bahasa Melayu* [Online]. Retrieved January 15, 2010, Available:http://www.moe.gov.my/bpk/sp_hsp/bm/kbsr/sp_bm_kbsr.pdf

[6] N. Ismail. (2003). *Budaya bangau oh bangau dalam Bahasa Melayu* [Online]. Retrieved January 18, 2010, Available: http://www.oocities.com/pendidikmy/berita/berita42003.html

[7] A. A. Abd. Talib. *Pedagogi Bahasa Melayu, prinsip, kaedah, dan teknik*, Utusan Publications & Distributors Sdn. Bhd.: Kuala Lumpur, 2000.

[8] F. J. Almeida-Martinez, J. Urquiza-Fuentes, and A. Velzquez-Iturbide, "Visualization of Syntax Trees for Language Processing Courses." *Journal of Universal Computer Science*, vol. 15(7), pp. 1546-1561. 2009.

[9] J. Bergin, K. Brodlie, M. Goldweber, R. Jimenez-Peris, S. Khuri, M. Patiho-Mattnez, M. McNally, T. Naps, S. Rodger, and J. Wilson, "An overview of visualization: its use and design: report of the Working Group on Visualization," *ITiCSE '96 Proceedings of the 1st conference on Integrating technology into computer science education*, 1996, USA, 192-200.

[10] M. J. Ab Aziz. (2007). Pengkomputeran Linguistik Bahasa Malaysia [Online]. Retrieved Dec 28, 2010, Available: http://www.ftsm.ukm.my/programming/prosiding-atur07/08-Juzaiddin.pdf

[11] N. A. Bakar, M. Salaebing, S. Salleh, N. M. Rodzes, and F. M. Ishak. *Penggunaan komputer dalam pengajaran bahasa*. Retrieved Dec 28, 2010, available http://202.28.66.7/smuhammad/pdf/Penggunaan%20Komputer%20dlm%20pengajaran%20bahasa.pdf. 2006.

[12] S. Ramli, "Reka bentuk dan implementasi suatu penghurai bahasa Melayu menggunakan sistem logik selari," M.S. thesis, Universiti Putra Malaysia, Selangor, 2002.

[13] A. I. Zainal Abidin, S. P. Yong, R. Kasbon, and H. Azman. "Utilizing top-down parsing technique in the development of a Malay language sentence parser, "in *Proc. of the 2nd International Conference on Informatics*, 2007, pp. 128-134.

[14] R. Abdul Latif, "Penyemak Sintaksis Ayat Bahasa Malaysia," M.S. thesis, Universiti Kebangsaan Malaysia, Bangi, 1995.

[15] N. F. Abd. Rahim, "A statistical parser to reduce structural ambiguity in Malay grammar rules," M.S. Thesis, Universiti Malaya, Kuala Lumpur, 2011.

[16] N. S. Karim, F. M. Onn, H. Musa, and A. H. Mahmood, *Tatabahasa Dewan Edisi Ketiga*, Dewan Bahasa dan Pustaka: Kuala Lumpur, 2009.