

COMPILE:

Interpréteur JS

Jonathan Soleillet
Selim Lakhdar



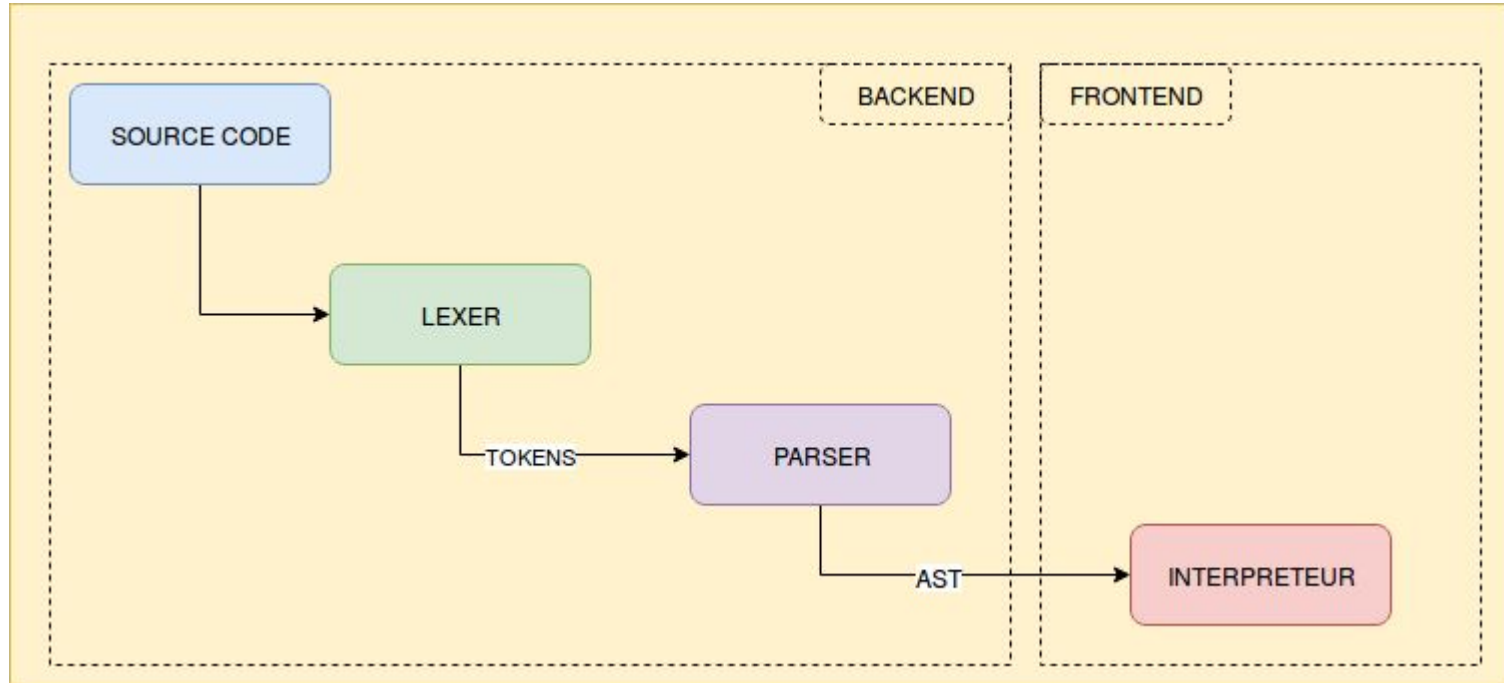
Plan

- Introduction
- Workflow de l'interprétation
- Interpréteur
- Benchmark
- Conclusion

Introduction

- Interpréteur JavaScript codé en C
- Les motivations :
 - Performance du langage C
 - Optimisation & Benchmark

Workflow de l'interprétation



Interpréteur (1/2)

- Les étapes :
 - Unserializer l'AST (Abstract Syntax Tree)
 - Evaluation de l'AST
- Stack :
 - Structure (tos, bos)
 - Fonctionnement
- Environnement d'interprétation :
 - Structure d'un environnement (hashmap, bos_local)
 - Environnement global & local

Interpréteur (2/2)

- Evaluation des statements (sans valeur de retour)
 - var, return, ifThenElse, while, branch
- Evaluation des expressions (avec valeur de retour)
 - fonctions, call, binaryOp, literal, postfix, prefix, ternary
- Evaluation des fonctions
 - Passage de paramètres
 - Récursion

Benchmark (gprof)

Each sample counts as 0.01 seconds.

% time	cumulative seconds	self seconds	calls	self ms/call	total ms/call	name
22.74	0.03	0.03	9711423	0.00	0.00	mlog
18.19	0.05	0.02	1578102	0.00	0.00	eval_litteral
18.19	0.07	0.02	485572	0.00	0.00	evaluate_statement
9.09	0.08	0.01	2913420	0.00	0.00	pop_stack
9.09	0.09	0.01	1092532	0.00	0.00	seek_in_env
9.09	0.10	0.01	1092531	0.00	0.00	eval_binaryop
9.09	0.11	0.01	242786	0.00	0.00	table_create

try1: with mlog

Each sample counts as 0.01 seconds.

% time	cumulative seconds	self seconds	calls	self ns/call	total ns/call	name
25.01	0.01	0.01	2913420	3.43	4.77	evaluate_expression
25.01	0.02	0.01	2913420	3.43	3.43	pop_stack
25.01	0.03	0.01	242786	41.21	73.27	eval_call
12.51	0.04	0.01	1092532	4.58	4.58	seek_in_env
12.51	0.04	0.01	242786	20.61	20.61	table_create
0.00	0.04	0.00	2913420	0.00	0.00	push_stack

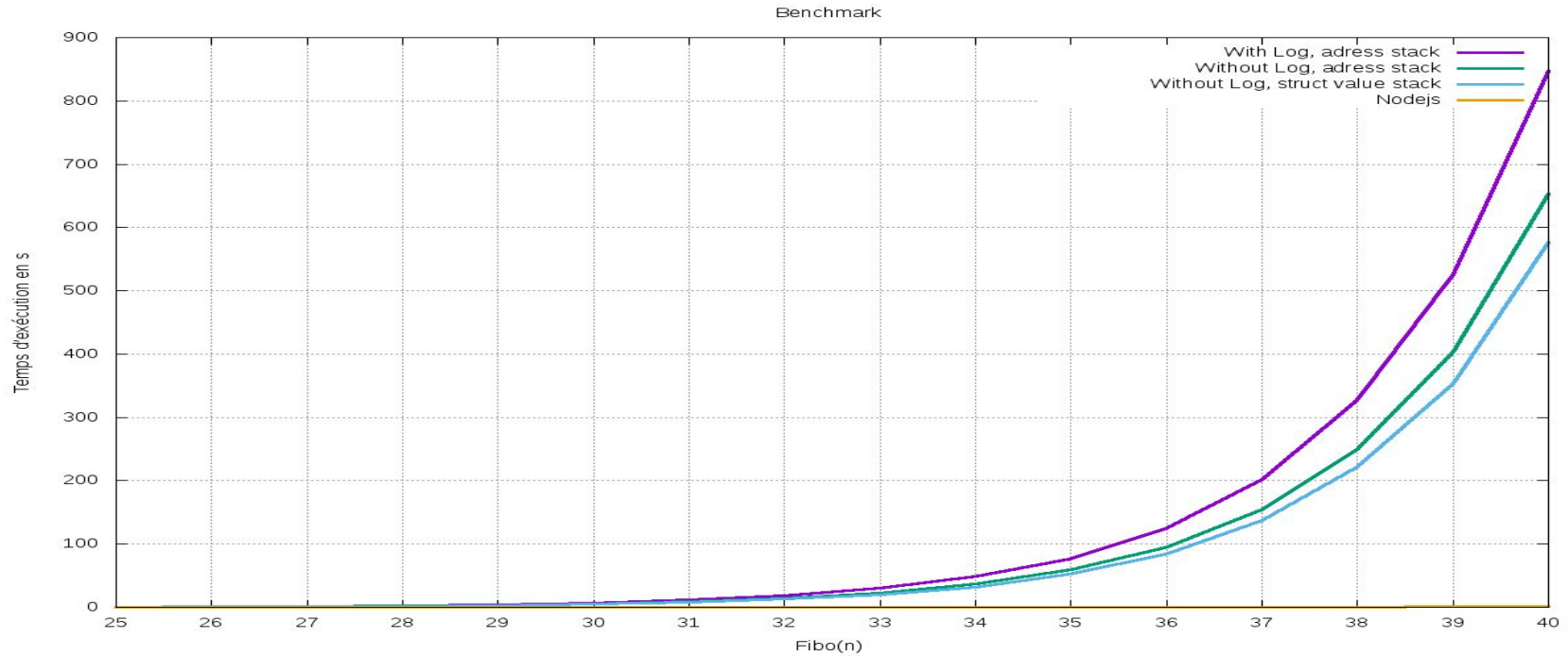
try2: without mlog

Each sample counts as 0.01 seconds.

% time	cumulative seconds	self seconds	calls	self ns/call	total ns/call	name
21.44	0.03	0.03	2913420	10.30	10.30	push_stack
17.87	0.06	0.03	2913420	8.59	19.51	evaluate_expression
14.29	0.08	0.02	2913420	6.87	6.87	pop_stack
14.29	0.10	0.02	1092531	18.32	42.36	eval_binaryop
7.15	0.11	0.01	1092532	9.16	18.32	seek_in_env
7.15	0.12	0.01	1092532	9.16	9.16	table_get
7.15	0.13	0.01	242786	41.21	124.78	eval_call
7.15	0.14	0.01	242786	41.21	41.21	table_create
3.57	0.14	0.01	485572	10.30	10.30	evaluate_statement

try3: without malloc & free

Benchmark



Conclusion

- Difficultés:
 - Evaluation
 - Création et gestion des environnements
 - Evaluation des appels de fonction (passage de paramètres et valeur de retour)
 - Gestion de fuite mémoire
- Bilan personnel
 - Approfondir notre compréhension sur la chaîne de compilation/interprétation
 - Approfondir nos connaissances sur l'optimisation du code C

Merci pour votre attention