

Projet Progsys

Jérôme Skoda & Joaquim Lefranc



Prérequis Logiciels

- **cmake**
- **gnuplot**



Les différents fichiers

- **bin/bench** : Executables utilisés pour la génération de graph
- **bin/test** : Tout les tests
- **script/bench** :
 - **generate-data.sh** : Generation des données pour les graphs
 - **generate-graph.sh** : Generation des graphiques (gnuplot)
 - **gnuplot.in** : Configuration de gnuplot
 - **output/** : Emplacement des résultats
- **script/test** :
 - **test.sh** : Script de test
 - **output/*.expected** : Output attendu des programmes de test
 - **output/*.out** : Output lors de la dernière exécution
- **src/** : Emplacement des sources
- **Makefile** : Règles de compilation
- **CMakeLists.txt** : Informations pour CMake



Compilation et exécution

Nettoyage et compilation :

```
$ make mrproper  
$ make clean  
$ make
```

Lancement des tests :

```
$ make test
```

Génération graphique :

```
$ make bench-data  
$ make bench-graph
```



- 7 fonctions implémentées

- Benchmarks

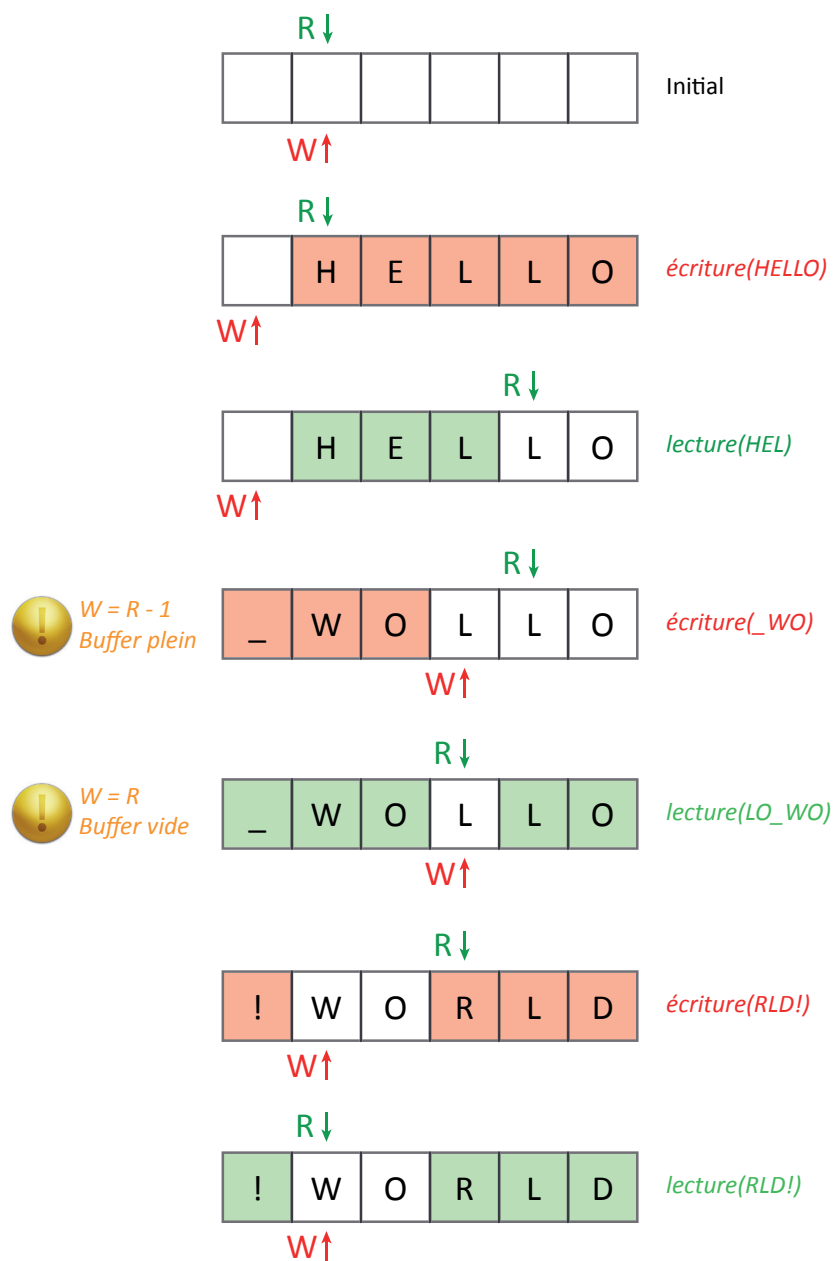
- Gestion des erreurs

- *R/W non bloquantes*

- Lecture / Ecriture par lots

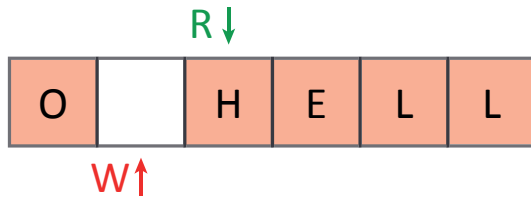
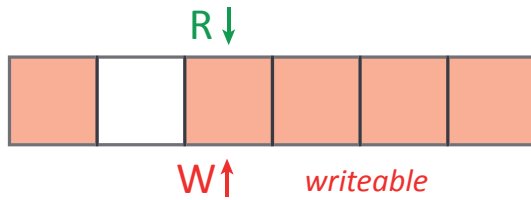
Fonctionnement

Exemple complet :





$W = R$
Conduit vide



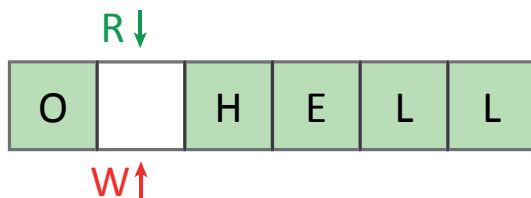
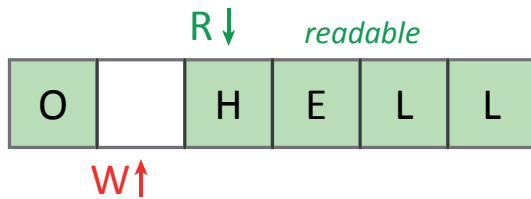
écriture(HELLO_TOTO)



Lors d'une écriture, W ne déplace
jamais et ne rejoint jamais R .

Donc :

Taille du buffer = Capacité + 1

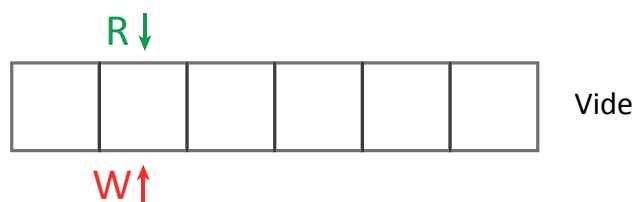
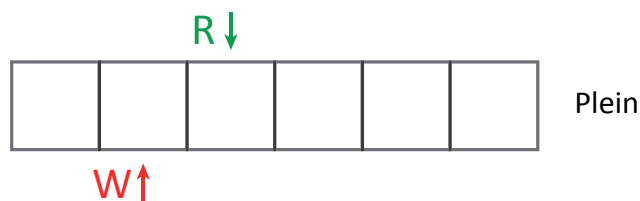
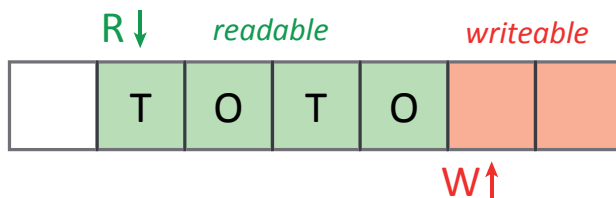


lecture(HELLO)



R en revanche peut rejoindre W après
une lecture, mais ne peut jamais le
dépasser.

Quelques cas :





Tests disponibles

- **Test 1** : Simple écriture et lecture dans un conduit anonyme
- **Test 2** : Simple écriture et lecture dans un conduit utilisant un fichier
- **Test 3** : Simple écriture et lecture de vecteur dans un conduit anonyme
- **Test 4** : test EoF sur read (return 0)
- **Test 5** : test Eof sur write (error)
- **Test 6** : test de fork sur conduit anonyme
- **Test 7** : test de fork sur conduit nommé



Benchmarks

Configuration matériel :

- Intel Core i7-4790K @ 4.00Ghz
- 16Go RAM @ 1600Mhz CAS9

#Bytes	Conduct (s)	Pipe (s)	Socket (s)
1	0.000211	0.000153	0.000146
10	0.000196	0.000130	0.000143
10 ²	0.000191	0.000134	0.000143
10 ³	0.000202	0.000147	0.000165
10 ⁴	0.000172	0.000153	0.000148
10 ⁵	0.000189	0.000178	0.000190
10 ⁶	0.000469	0.000655	0.000618
10 ⁷	0.001884	0.005612	0.004523
10 ⁸	0.0154550	0.051857	0.044170
10 ⁹	0.149838	0.529052	0.434956
10 ¹⁰	1.567240	5.793963	4.426537

Courbes constatées :

- Conduct : $2.1E-4 + (x - 3E5) / 70E8$
- Pipe : $3.1E-4 + (x - 3E5) / 17.7E8$
- Socket : $2.7E-4 + (x - 3E5) / 22.4E8$

Bench Conduct/Pipe/Socket

