



Rapport du TP6 : WorldOfECN

Auteur :

Otmane EL ALOI, Oussama TOUAOUSSA

Option :

GÉNIE INFORMATIQUE

Table des matières

1	Introduction	1
1.1	Chargement d'une partie	1
1.2	Sauvegarde d'une partie	3
1.3	Illustration du fonctionnement du jeu WOE	7
2	Conclusion :	10

Table des figures

1	Format d'une partie du jeu sauvegardée	1
2	Schéma illustrant la lecture d'un fichier par la classe <code>BufferedReader</code>	2
3	code exécuté pour charger une partie	2
4	Résultat du chargement d'une partie à partir d'un fichier texte	3
5	Schéma illustrant l'écriture d'un fichier par la classe <code>BufferedWriter</code>	4
6	Code pour sauvegarder un fichier à partir	5
7	Illustration de la demande de sauvegarde au milieu de la partie	5
8	Fichier de sauvegarde d'une partie	6
9	Illustration de la génération du nom de fichier de sauvegarde automatiquement	6
10	Partie jouée	8
11	Illustration du chargement d'une partie sauvegardée	9
12	Affichage du monde	9
13	Affichage du monde	10

1 Introduction

L'objectif de ce TP est en premier lieu, l'amélioration et la finition des résultats des TPs précédents. Il est aussi l'occasion d'ajouter des briques finales nécessaires pour un jeu fluide. C'est les fonctionnalités de sauvegarde et de chargement des parties jouées. Ce TP a aussi pour objectif la découverte des classes Java responsables de la gestion des flux de données textuelles. Tel que **BufferedReader** et **BufferedWriter**. Et finalement la classe **StringTokenizer** qui facilite le parcours des chaînes de caractère.

1.1 Chargement d'une partie

Le chargement d'une partie se fait à partir d'un fichier texte structuré comme suit :

```
|Largeur 25
|Hauteur 25
Guerrier grosBill 250 0 80 60 0 10 15 0 1 5 8 3
Mage merlin 50 50 20 25 85 80 2 20 7 2 15 22
Voleur arsene 75 0 60 80 0 25 15 0 1 5 5 7
Archer robin 75 0 50 60 0 25 10 0 10 5 20 11
Paysan peon 25 0 0 30 0 0 1 0 1 5 3 6
Loup 30 50 50 25 5 19 3
Lapin 4000 90 1 1000 1 10 10
Loup 80 75 30 45 5 2 13
Loup 30 30 30 20 10 5 14
Lapin 30 20 20 40 10 23 23
NuageToxique 50 5 20 5
Soin 25 18 18
Mana 50 4 4
Soin 100 4 20
Mana 100 7 1
Soin 48 1 19
Mana 24 0 0
```

FIGURE 1 – Format d'une partie du jeu sauvegardée

Pour lire ce fichier nous avons crée une instance d'objet de type **BufferedReader**. Cette dernière est une sous classe de la classe Java **Reader** utilisée pour simplifier la lecture de texte à

partir de flux d'entrée de caractères (character input stream) et optimiser les performances du programme. Ainsi le fonctionnement pourra être illustré comme suit :

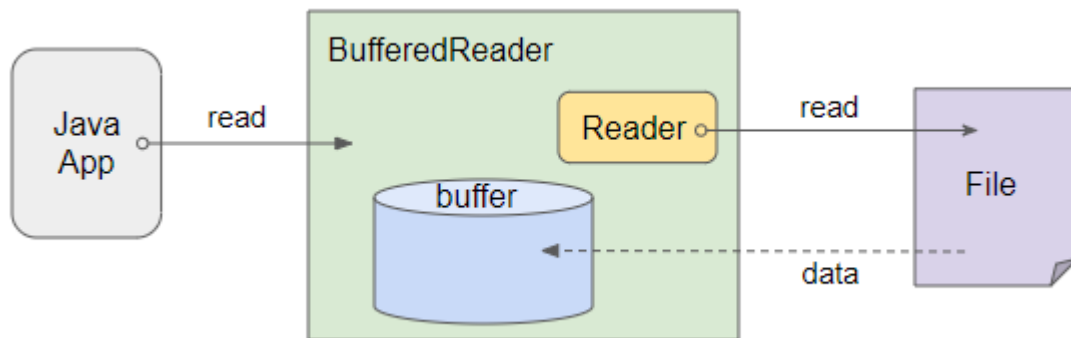


FIGURE 2 – Schéma illustrant la lecture d'un fichier par la classe `BufferedReader`

L'objet créé **`BufferedReader`** lit le fichier texte d'une partie sauvegardée et stocke un après l'autre les caractères et libère les positions lues dans un **buffer** qui est un tableau de taille par défaut (taille = 8192).

Ci-dessous le code exécuté pour charger une partie :

```

Voulez vous une partie aleatoire (=1) ou une partie existante(=0) ?
1
Choisir le type de votre joueur
Archer
Entrer un nom a votre joueur
warrior
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
0
Debut de deplacement
Deplacement par clavier: w=Haut, d=Droit, s=Bas, a=Gauche
d
Position du joueur :
[0,21]
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
1
Voulez-vous choisir un nom de la partie ? Oui = 1 ou Non = 0
1
Entrez un nom a votre partie
Part3
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
1
  
```

FIGURE 3 – code exécuté pour charger une partie

La figure ci-dessous montre le résultat du chargement :

```
Vous voulez une partie aleatoire (=1) ou une partie existante(=0) ?
0
Entrer le nom complet de la partie
-----Caracteristiques du jeu-----
Largeur 25
Hauteur 25
Guerrier grosBill 250 0 80 60 0 10 15 0 1 5 8 3
Mage merlin 50 0 20 25 85 80 2 0 7 2 15 22
Archer robin 75 0 50 60 0 25 10 0 10 5 20 11
Paysan peon 25 0 0 30 0 0 1 0 1 5 3 6
NuageToxique 50 5 25 50 20 5
Soin 25 18 18
Mana 50 4 4
Soin 100 4 20
Mana 100 7 1
Soin 48 1 19
Mana 24 0 0
Joueur bob 150 0 90 50 0 10 20 0 1 5 5 12 12
```

FIGURE 4 – Résultat du chargement d'une partie à partir d'un fichier texte

1.2 Sauvegarde d'une partie

De manière identique à la lecture d'un fichier texte en Java, basée sur l'utilisation de la classe **BufferedReader**, le mécanisme de sauvegarde a été basé sur l'utilisation de la classe **BufferedWriter**. Cette dernière enveloppe à son intérieur un objet **Writer**(figure :6), qui est responsable de l'écriture des données sur le fichier texte de sauvegarde.

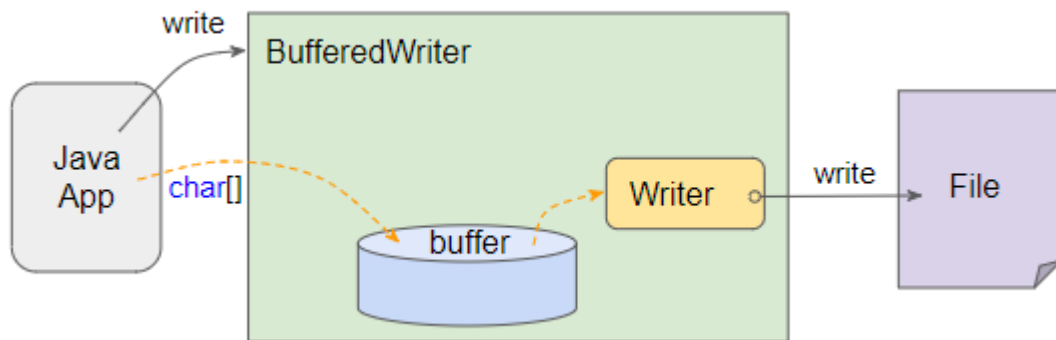


FIGURE 5 – Schéma illustrant l'écriture d'un fichier par la classe `BufferedWriter`

A l'aide de la méthode **`write()`** l'objet `BufferedWriter` écrit les données dans le buffer. lorsque ce dernier est plein, toutes les données dans buffer seront poussées dans `Writer` et libèrera buffer. Vers la fin la méthode **`BufferedWriter.flush()`** est utilisée pour libérer les données du buffer vers le writer. Les données sont également transférées de buffer vers `Writer` en appelant la méthode *`BufferedWriter.close()`*.

Les données qui sont temporairement écrites dans buffer minimisent l'écriture continue sur la cible (par exemple les fichiers sur un disque dur), améliorant ainsi les performances du programme.

La code suivant est exécuté pour sauvegarder un fichier avec un nom proposé par l'utilisateur

Part3 :

```

World WoE = new World();
// Propositions de monde aleatoire ou partie existante :
Scanner sc = new Scanner(System.in);
System.out.println("Voulez vous une partie aleatoire (=1) ou une partie existante(=0) ?");
String choix = sc.next();
boolean verf = true;
while (verf){
    if (choix.equals("1")){
        // partie aleatoire
        int nbMaximalSouhaite = 10;
        WoE = new World(nbMaximalSouhaite);
        WoE.creeMondealea();
        WoE.creationJoueur();
        verf = false;
    }
}

```

FIGURE 6 – Code pour sauvegarder un fichier à partir

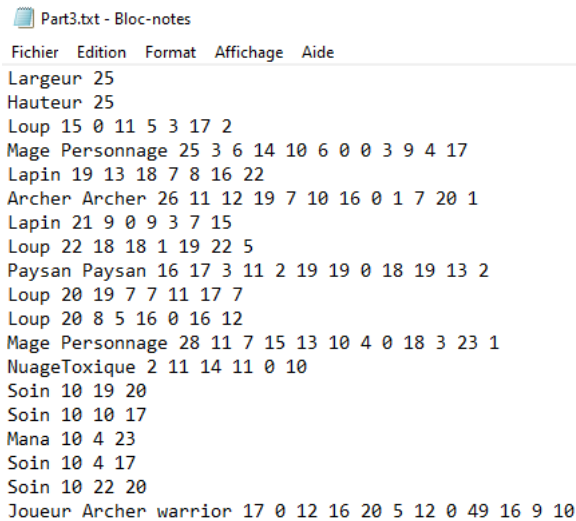
```

Voulez vous une partie aleatoire (=1) ou une partie existante(=0) ?
1
Choisir le type de votre joueur
Archer
Entrer un nom a votre joueur
warrior
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
0
Debut de deplacement
Deplacement par clavier: w=Haut, d=Droit, s=Bas, a=Gauche
d
Position du joueur :
[0,21]
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
1
Voulez-vous choisir un nom de la partie ? Oui = 1 ou Non = 0
1
Entrez un nom a votre partie
Part3
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
1

```

FIGURE 7 – Illustration de la demande de sauvegarde au milieu de la partie

Ci-dessous le résultat de l'exécution du code précédent :



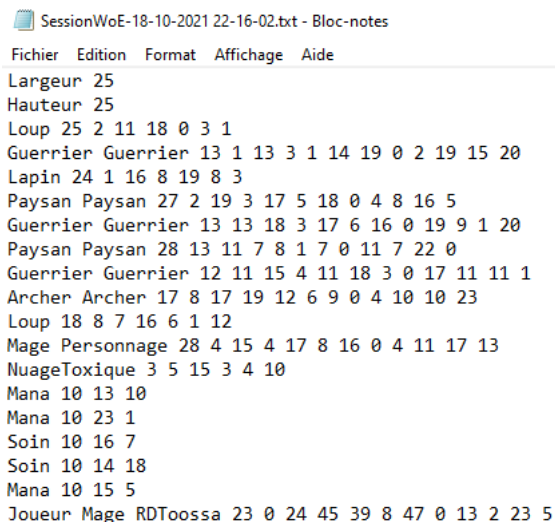
```

Fichier  Edition  Format  Affichage  Aide
Largeur 25
Hauteur 25
Loup 15 0 11 5 3 17 2
Mage Personnage 25 3 6 14 10 6 0 0 3 9 4 17
Lapin 19 13 18 7 8 16 22
Archer Archer 26 11 12 19 7 10 16 0 1 7 20 1
Lapin 21 9 0 9 3 7 15
Loup 22 18 18 1 19 22 5
Paysan Paysan 16 17 3 11 2 19 19 0 18 19 13 2
Loup 20 19 7 7 11 17 7
Loup 20 8 5 16 0 16 12
Mage Personnage 28 11 7 15 13 10 4 0 18 3 23 1
NuageToxique 2 11 14 11 0 10
Soin 10 19 20
Soin 10 10 17
Mana 10 4 23
Soin 10 4 17
Soin 10 22 20
Joueur Archer warrior 17 0 12 16 20 5 12 0 49 16 9 10

```

FIGURE 8 – Fichier de sauvegarde d'une partie

Dans le cas où l'utilisateur ne fournit pas le nom du fichier et pour assurer l'unicité des noms des fichiers. On affecte au nom du fichier la date et le temps de sauvegarde. Pour cela, on utilise les classes Java **LocalDateTime**, **DateTimeFormatter**. Comme illustré dans l'exemple d'une partie sauvegardée suivant :



```

Fichier  Edition  Format  Affichage  Aide
Largeur 25
Hauteur 25
Loup 25 2 11 18 0 3 1
Guerrier Guerrier 13 1 13 3 1 14 19 0 2 19 15 20
Lapin 24 1 16 8 19 8 3
Paysan Paysan 27 2 19 3 17 5 18 0 4 8 16 5
Guerrier Guerrier 13 13 18 3 17 6 16 0 19 9 1 20
Paysan Paysan 28 13 11 7 8 1 7 0 11 7 22 0
Guerrier Guerrier 12 11 15 4 11 18 3 0 17 11 11 1
Archer Archer 17 8 17 19 12 6 9 0 4 10 10 23
Loup 18 8 7 16 6 1 12
Mage Personnage 28 4 15 4 17 8 16 0 4 11 17 13
NuageToxique 3 5 15 3 4 10
Mana 10 13 10
Mana 10 23 1
Soin 10 16 7
Soin 10 14 18
Mana 10 15 5
Joueur Mage RDTtoossa 23 0 24 45 39 8 47 0 13 2 23 5

```

FIGURE 9 – Illustration de la génération du nom de fichier de sauvegarde automatiquement

1.3 Illustration du fonctionnement du jeu WOE

Pour illustrer le bon fonctionnement du jeu et notamment les fonctionnalités de sauvegarde et de chargement d'une partie de jeu.

La figure suivante montre l'historique d'une partie jouée :

```
Position du joueur :
[1,13]
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
0
Debut de la Tour 1
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
0
Debut de deplacement
Deplacement par clavier: w=Haut, d=Droit, s=Bas, a=Gauche
s
Position du joueur :
[2,13]
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
0
Debut de la Tour 2
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
1
Debut de combat
attaque ratée!
Attaque réussi avec un dégat de: 0
attaque ratée!
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
0
Debut de la Tour 3
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
1
Debut de combat
attaque ratée!
attaque ratée!
attaque ratée!
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
0
Debut de la Tour 4
Entrer 0 si vous voulez se deplacer ou 1 si vous voulez combattre
1
Debut de combat
Plus de flèches!Attaque réussi avec un dégat de: 0
attaque ratée!
attaque ratée!
Voulez-vous sauvegarder cette partie ? Oui = 1 ou Non = 0
1
Voulez-vous choisir un nom de la partie ? Oui = 1 ou Non = 0
1
Entrez un nom a votre partie
Part-5-Sauvegarde
Partie sauvegardee dans le fichier : Part-5-Sauvegarde.txt
Fin de la partie
```

FIGURE 10 – Partie jouée

Dans la figure précédente, on peut remarquer 2 fonctionnalités importantes :

- Possibilité de se déplacer en appuyant sur les touches du clavier. dans l'exemple ci-dessous, le joueur a choisi de se déplacer vers le bas en appuyant sur la touche "s"
- Possibilité de sauvegarder une partie à chaque tour du jeu. Un exemple de sauvegarde sous le nom "**Part-5-Sauvegarde**" est fournie.

La figure suivante montre le chargement de la partie précédente. On remarque qu'on charge le monde là où on l'a laissé.

```

Voulez vous une partie aleatoire (=1) ou une partie existante(=0) ?
0
Entrer le nom complet de la partie
Part-5-Sauvegarde.txt
-----Caracteristiques du jeu-----
Largeur 25
Hauteur 25
Loup 7 14 0 1 17 23 24
Archer Archer 11 4 0 8 1 2 18 0 7 16 11 12
Lapin 24 7 17 9 18 5 3
Loup 0 19 7 10 11 22 8
Lapin 27 5 4 15 13 16 21
Lapin 12 9 11 0 17 2 5
Paysan Paysan 24 10 10 5 7 12 16 0 14 17 18 8
Guerrier Guerrier 16 19 19 4 15 3 4 0 5 5 8 18
Mage Personnage 27 6 11 17 17 1 18 0 9 2 22 7
Loup 22 6 2 10 10 22 2
NuageToxique 0 12 11 14 10 10
Mana 10 19 16
Soin 10 24 9
Mana 10 13 9
Mana 10 22 1
Soin 10 12 7
Joueur WARRIOR 41 0 47 34 32 21 17 0 39 45 45 2 13
  
```

FIGURE 11 – Illustration du chargement d'une partie sauvegardée

Nous avons amélioré également l'affichage du monde du jeu. La figure suivante montre l'affichage du monde vers la fin de la partie précédente :

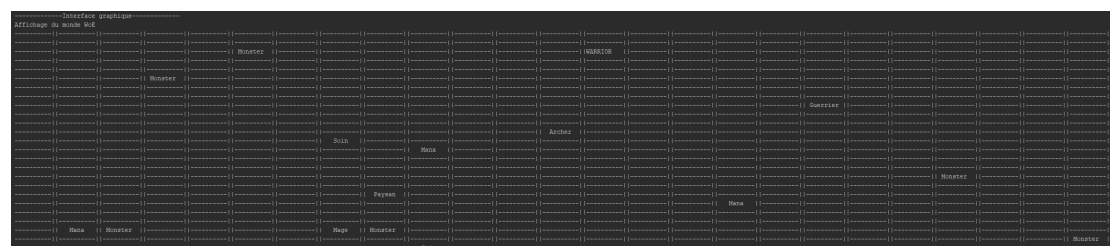


FIGURE 12 – Affichage du monde

Un autre exemple :

```
-----Interface graphique-----
Affichage du monde WoE
WARRIOR  ||Nourriture||Nourriture||Nourriture||-----||-----||-----||-----||-----|||
Nourriture|-----||Nourriture||Nourriture||Nourriture|| Monster ||Nourriture|-----||Nourriture||Nourriture|||
Mana      ||-----||-----||-----||-----||-----||Nourriture|-----||Nourriture||Nourriture|||
-----||Nourriture|-----||Nourriture|| Soin  ||-----||-----||-----|| Soin  ||Nourriture|||
-----||Nourriture||Nourriture||Nourriture||-----||Nourriture||Nourriture||-----||Nourriture|-----|||
Nourriture|Nourriture||Nourriture||Nourriture||-----||-----||-----||-----||-----||Nourriture|||
-----||Nourriture|-----||-----||-----||-----||-----||-----||-----||-----||Nourriture|||
Nourriture|-----||Nourriture||Nourriture||-----||Nourriture||-----||-----|| Mage  ||-----|||
-----||Nourriture|-----||-----||-----||Nourriture||Nourriture||Nourriture||Nourriture|| Mage  |||
Nourriture|-----||Nourriture||-----||Nourriture||-----||Nourriture|| Archer  ||Nourriture||Nourriture|||
```

FIGURE 13 – Affichage du monde

2 Conclusion :

Ce TP avait pour but de renforcer les connaissances de manipulation des fichiers (lecture et écriture) en utilisant des classes de Java performantes comme **BufferedReader**, **BufferedWriter** afin de sauvegarder une partie de jeu et la récupérer et continuer a jouer.

Le TP est aussi une initiation aux techniques d’affichage graphique en utilisant les librairies de java. Il a permis aussi de rappeler les notions vues précédemment (abstraction,...).