# 📘 Intégration des Logs dans le Projet

## 🔧 Système de journalisation (`log.c` / `log.h`)

### 📌 Objectif

Le système de journalisation (logging) a été intégré pour :  
- Tracer les événements clés du programme.  
- Faciliter le débogage.  
- Comprendre les étapes de traitement du réseau de neurones et de l’interface SDL.

### 📁 `log.h` – Interface des fonctions de log

Ce fichier contient :  
- L’énumération des niveaux de log (`LOG\_LEVEL\_INFO`, `LOG\_LEVEL\_WARNING`, `LOG\_LEVEL\_ERROR`).  
- La fonction générique `log\_message(level, format, ...)` pour afficher des logs formatés.  
- Des fonctions utilitaires pour centraliser les messages liés à certaines actions :  
 • `log\_colorier\_once(const char \*mode)` – loggue une seule fois le coloriage (R/B ou RGB).  
 • `log\_generation\_spirale(const char \*type)` – loggue la génération de spirales.  
 • `log\_save\_success()` / `log\_save\_failure()` – logs liés à la sauvegarde.  
 • `log\_load\_success()` / `log\_load\_failure()` – logs liés au chargement.  
 • `log\_choix\_train\_load()` – loggue l’ouverture de la fenêtre "Train/Load".

### ⚙️ `log.c` – Implémentation centralisée

Ce fichier implémente :  
1. `log\_message()` : utilise `va\_list` pour formater un message avec horodatage et niveau.  
2. Fonctions `log\_\*\_once()` : utilisent `static int deja\_fait = 0` pour n’afficher le message qu’une seule fois.  
3. Tous les logs sont centralisés pour une traçabilité uniforme.

### 🔌 Intégration dans le projet

|  |  |  |
| --- | --- | --- |
| Contexte | Fichier | Log appelé |
| Propagation dans le réseau | neural\_network.c | log\_propagation\_once() |
| Backpropagation | neural\_network.c | log\_retropropagation() |
| Colorier écran (R/B ou RGB) | data.c | log\_colorier\_once("R/B") / ("RGB") |
| Génération de spirales | data.c | log\_generation\_spirale("R/B") / ("multi-classes") |
| Sauvegarde du réseau | save.c | log\_save\_success() / log\_save\_failure() |
| Chargement d’un fichier | save.c | log\_load\_success() / log\_load\_failure() |
| Affichage Train ou Load | ui\_manager.c | log\_choix\_train\_load() |

### ✅ Résultat

- Les logs permettent de comprendre l’état du programme sans ouvrir GDB.  
- Le système est minimaliste, centralisé et facilement réutilisable.  
- Il améliore significativement la traçabilité des actions utilisateur et du réseau.