

COMMENT NE PLUS AVOIR PEUR

DE VOS FICHIERS DE LOG

COMMENT NE PLUS AVOIR PEUR



DE VOS FICHIERS DE LOG

Qui suis-je ?

Virginie PAGEAUD (CASAVECCHIA)



- 🦄 **Customer Success Engineer @Quarkslab**
- 💳 10+ années dans le domaine du paiement par carte bancaire
- 🔧 Couteau suisse, touche à tout et curieuse
- 🧠 Amatrice d'énigmes et problèmes à résoudre
- 👉 Passionnée par la transmission de connaissances
- 📄 Blog : <https://virginie-blog.pageaud.net/>

Quarkslab

Securing every bit of your data



French cybersecurity company
founded in 2011



€ 8m fundraising in 2020



110+ employees
20% PhDs



324 conferences
221 Blogposts
46 academic articles

23 MISC articles
23 Public reports
58 CVE



Security audit & vulnerability research



Device & application protection



Offices in France and Argentina

Vous aussi ?





Benjamin Dauvissat - @bdauvissat@mastodon... @bdauvi... · 6 oct. ...

En réponse à [@ponceto91](#)

J'ai trouvé pire que l'absence de log.
Les logs hyper verbeux dans lesquels on ne trouve aucune information pertinente.



Virginie PAGEAUD
Quarkslab

devfest24
Dijon

Connaissez-vous ?



Virginie PAGEAUD
Quarkslab

Connaissez-vous ?

- Equipe d'exploitation
 -



Connaissez-vous ?

- Equipe d'exploitation
 - Exemple au quotidien



Disclaimer

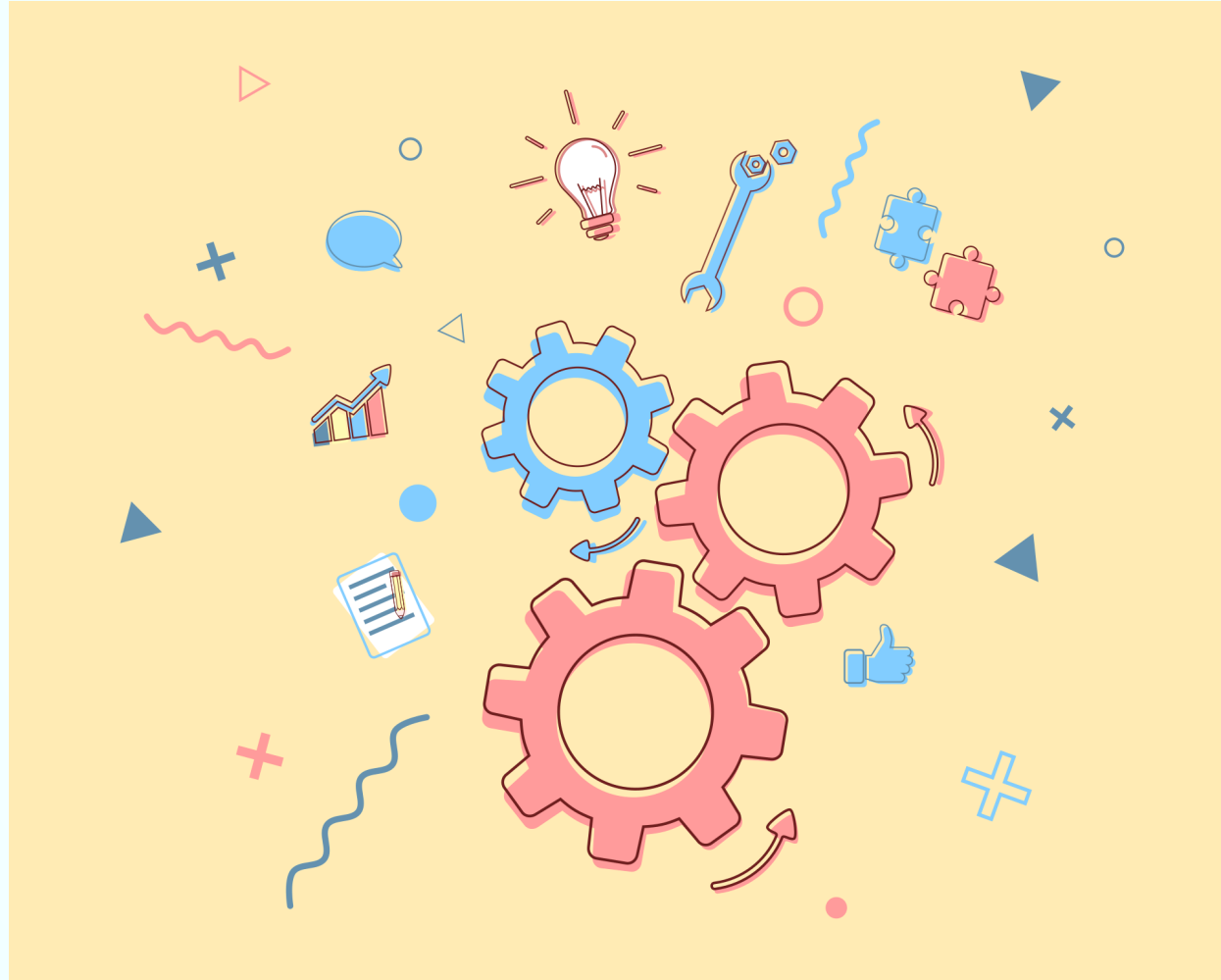


Virginie PAGEAUD
Quarkslab

A qui/quoi servent les traces ?



Réaliser le développement



Vous aussi vous l'avez déjà fait...

```
public List<Glycemie> importFichier() {  
    logger.debug("importFichier début -----");  
    /*  
    ...  
    */  
    logger.debug("toto");  
    /*  
    ...  
    */  
    logger.debug("importFichier fin -----");  
    return retour;  
}
```



Faciliter les tests et la correction d'anomalies



Faciliter les tests et la correction d'anomalies

```
//...  
catch (IOException e) {  
    logger.error("Une erreur est survenue lors du traitement du fichier "  
                + fichier.toString() + ".");  
    e.printStackTrace();  
}  
//...
```



Aider l'exploitation

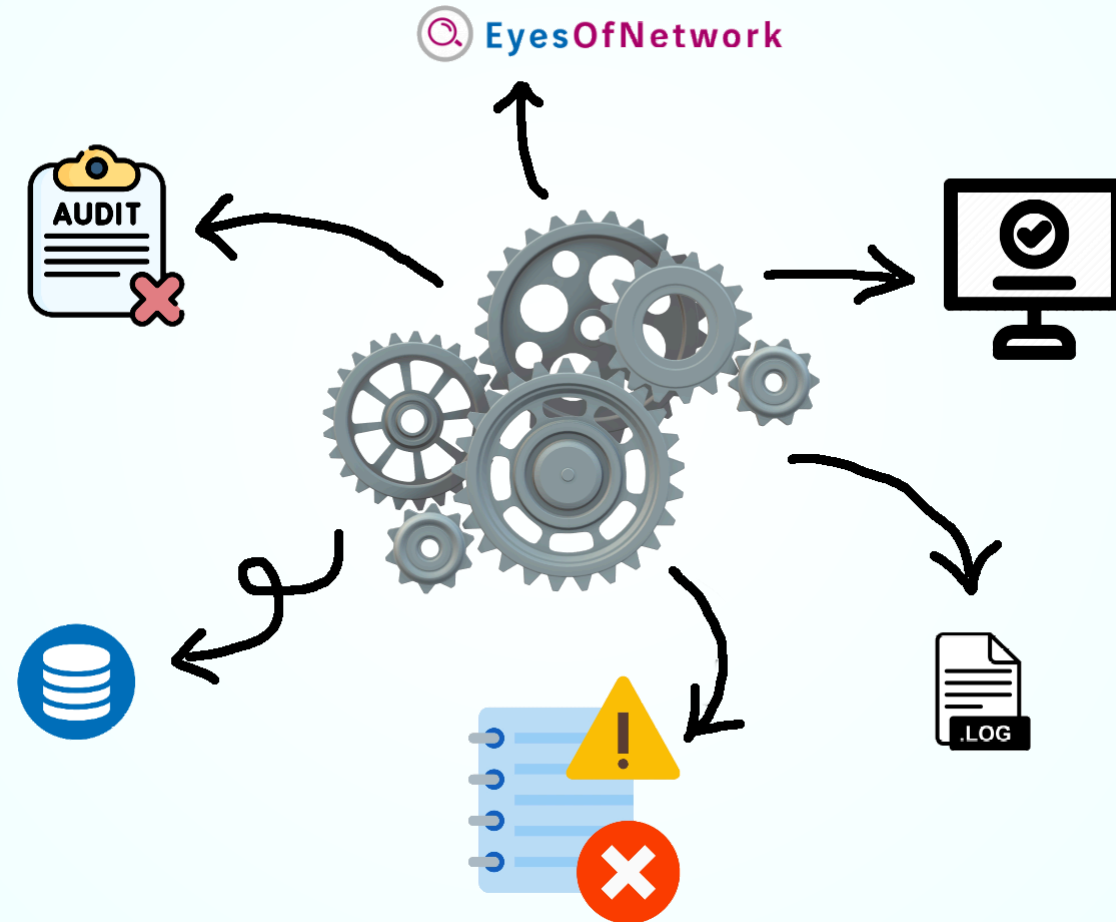


Aider l'exploitation

```
if (!repertoire.isDirectory()) {  
    logger.error("\"" + repertoire.toString() + "\"  
        + " n'est pas un répertoire. Arrêt du traitement. Veuillez vérifier la configuration.");  
  
    return retour;  
}
```



Diverses cibles



Un exemple de système de supervision - Nagios en vue liste

Nagios Current Network Status
Last Updated: Tue Jun 9 15:56:07 CDT 2015
Updated every 50 seconds
Nagios Core™ 4.0.8 - www.nagios.org
Logged in as nagiosadmin

Host Status Totals
Up: 41, Down: 9, Unreachable: 0, Pending: 1
All Problems: 9, All Types: 51

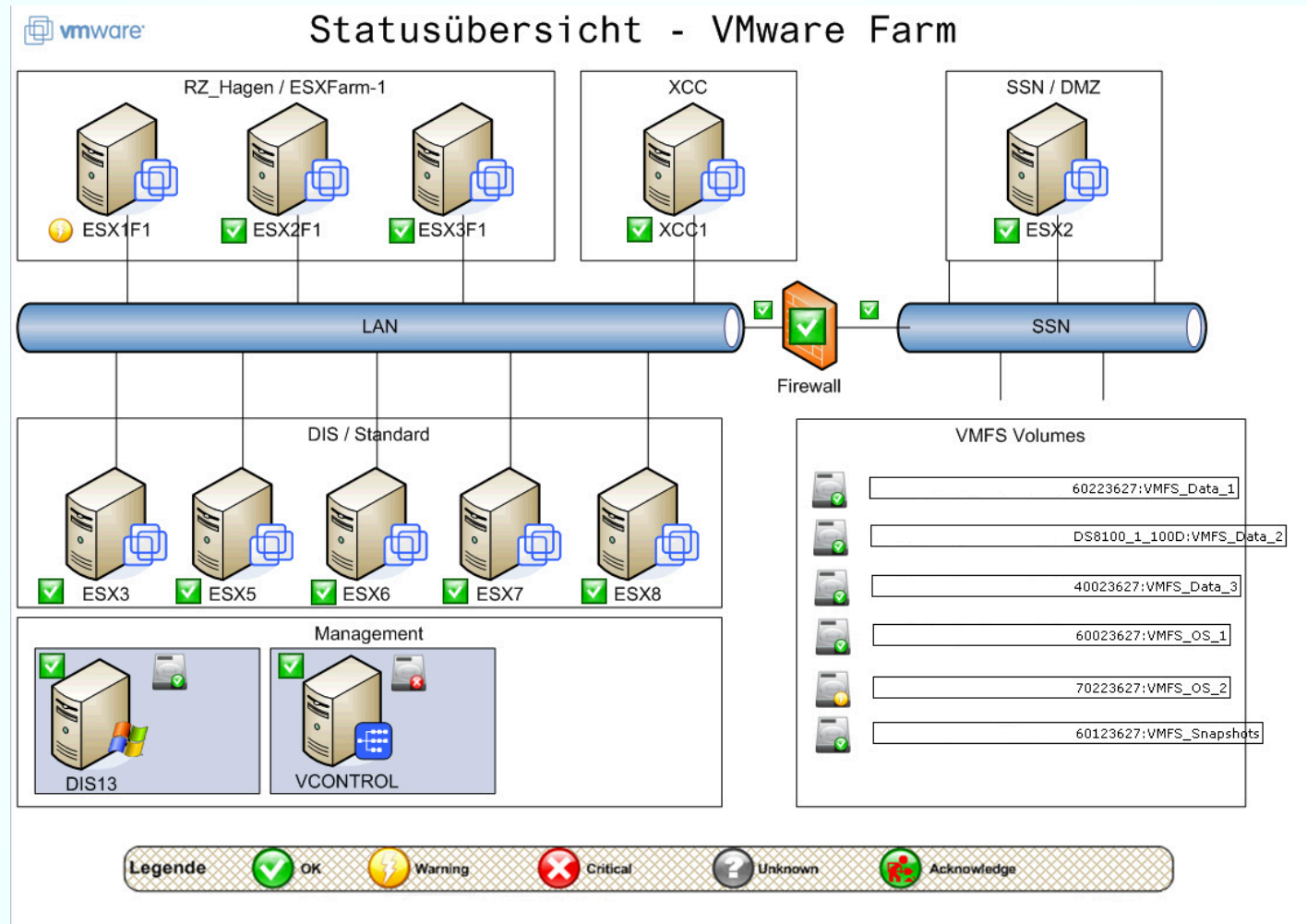
Service Status Totals
Ok: 338, Warning: 11, Unknown: 9, Critical: 141, Pending: 1
All Problems: 161, All Types: 500

Service Status Details For All Hosts
Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
ScottsServer	Page File Usage	OK	06-09-2015 15:54:55	36d 4h 35m 32s	1/5	Paging File usage is 0.00 %
	Ping	OK	06-09-2015 15:55:30	4d 6h 4m 35s	1/5	OK - 192.168.5.15: rta 1.011ms, lost 0%
	SQL Server	CRITICAL	06-09-2015 15:51:43	36d 4h 31m 32s	5/5	MSSQLSERVER: Not found
	Server Work Queues	OK	06-09-2015 15:51:41	26d 4h 13m 36s	1/5	Current work queue (an indication of processing load) is 0
	Uptime	OK	06-09-2015 15:51:55	4d 6h 2m 8s	1/5	System Uptime - 4 day(s) 6 hour(s) 0 minute(s)
abc.com	DNS IP Match	OK	06-09-2015 15:52:18	26d 15h 15m 11s	1/5	DNS OK: 0.007 seconds response time: abc.com returns 199.181.132.250
	DNS Resolution	OK	06-09-2015 15:51:08	26d 15h 16m 25s	1/5	DNS OK: 0.011 seconds response time: abc.com returns 199.181.132.250
	HTTP	OK	06-09-2015 15:55:42	5d 14h 11m 15s	1/5	HTTP OK: HTTP/1.1 301 Moved Permanently - 428 bytes in 0.144 second response time
	Ping	OK	06-09-2015 15:54:58	4d 21h 21m 47s	1/5	OK - abc.com: rta 57.127ms, lost 0%
	Web Page Content	CRITICAL	06-09-2015 15:55:39	39d 21h 17m 29s	5/5	HTTP CRITICAL: HTTP/1.1 301 Moved Permanently - string 'ABC' not found on http://abc.com/80/ - 504 bytes in 0.123 second response time
conference.nagios.local	NagIOS	CRITICAL	06-09-2015 15:51:08	83d 0h 51m 18s	5/5	connect to address 192.168.5.10 and port 139: No route to host
	Ping	CRITICAL	06-09-2015 15:51:08	83d 0h 50m 45s	5/5	CRITICAL - 192.168.5.10: Host unreachable @ 192.168.5.60: rta nan, lost 100%
	RDP	CRITICAL	06-09-2015 15:54:20	83d 0h 53m 4s	5/5	connect to address 192.168.5.10 and port 3389: No route to host
exchange.nagios.org	/ Disk Usage	OK	06-09-2015 15:53:39	0d 1h 56m 46s	1/5	DISK OK - free space: / 72915 MB (92% inode=99%)
	Apache Web Server	OK	06-09-2015 15:54:26	0d 1h 56m 0s	1/5	httpd (pid 2296) is running...
	CPU Stats	OK	06-09-2015 15:54:50	3d 4h 17m 5s	1/5	CPU STATISTICS OK: user=5.11% system=2.60% iowait=0.00% idle=92.22%
	Cron Scheduling Daemon	OK	06-09-2015 15:54:26	0d 1h 55m 57s	1/5	crond (pid 2296) is running...
	DNS IP Match	OK	06-09-2015 15:51:26	19d 18h 48m 9s	1/5	DNS OK: 0.279 seconds response time: exchange.nagios.org returns 66.228.58.94
	DNS Resolution	OK	06-09-2015 15:51:11	4d 7h 11h 59m 59s	1/5	DNS OK: 0.015 seconds response time: exchange.nagios.org returns 66.228.58.94
	HTTP	OK	06-09-2015 15:54:58	6d 23h 52m 1s	1/5	HTTP OK: HTTP/1.1 301 Moved Permanently - 572 bytes in 0.085 second response time
	Load	OK	06-09-2015 15:54:05	8d 1h 35m 37s	1/5	OK - load average: 0.15, 0.20, 0.22
	Memory Usage	OK	06-09-2015 15:53:54	0d 1h 56m 30s	1/5	OK - 7824 / 8107 MB (94%) Free Memory: Used: 483 MB, Shared: 0 MB, Buffers: 21 MB, Cached: 228 MB
	MySQL Server	OK	06-09-2015 15:53:40	3d 23h 15m 19s	1/5	mysqld (pid 2127) is running...
	Open Files	OK	06-09-2015 15:53:33	1d 16h 8m 41s	1/5	OK: 1184 open files (0% of max: 830107)
	Ping	OK	06-09-2015 15:54:14	0d 1h 56m 11s	1/5	OK - exchange.nagios.org: rta 40.931ms, lost 0%
	SSH Server	OK	06-09-2015 15:54:36	0d 0h 56m 12s	1/5	sshd (pid 1961) is running...
	Swap Usage	OK	06-09-2015 15:55:48	1d 4h 20m 34s	1/5	SWAP OK - 100% free (255 MB out of 255 MB)
	System Logging Daemon	OK	06-09-2015 15:53:60	0d 1h 56m 35s	1/5	rsyslogd (pid 1653) is running...
	Total Processes	OK	06-09-2015 15:51:54	0d 0h 4m 13s	1/5	PROCS OK: 145 processes
	Users	OK	06-09-2015 15:53:58	26d 15h 14m 20s	1/5	USERS OK - 1 users currently logged in
	Yum Updates	WARNING	06-09-2015 15:52:22	0d 1h 53m 6s	5/5	YUM WARNING: O/S requires an update.
freewad	Ping	OK	06-09-2015 15:54:37	206d 3h 56m 47s	1/5	OK - 192.168.5.1: rta 2.772ms, lost 0%
gateway.nagios.local	HTTPS	OK	06-09-2015 15:54:32	85d 3h 20m 49s	1/5	TCP OK - 0.002 second response time on 192.168.5.1 port 443
	Ping	OK	06-09-2015 15:54:26	85d 3h 20m 43s	1/5	OK - 192.168.5.1: rta 2.250ms, lost 0%
	Telnet	OK	06-09-2015 15:54:40	85d 3h 21m 9s	1/5	TCP OK - 0.002 second response time on 192.168.5.1 port 23
	bgrou0 Bandwidth	OK	06-09-2015 15:53:53	0d 11h 57m 16s	1/5	OK - Current BW in: 49Mbps Out: 6.70Mbps
	bgrou0 Status	OK	06-09-2015 15:51:58	85d 3h 21m 37s	1/5	OK: Interface bgrou0 (index 9) is up.
	bgrou1 Bandwidth	OK	06-09-2015 15:55:42	286d 17h 2m 4s	1/5	OK - Current BW in: 0Mbps Out: 0Mbps
	bgrou1 Status	CRITICAL	06-09-2015 15:51:45	85d 3h 17m 6s	5/5	CRITICAL: Interface bgrou1 (index 10) is down.
	bgrou2 Bandwidth	OK	06-09-2015 15:53:17	286d 17h 1m 10s	1/5	OK - Current BW in: 0Mbps Out: 0Mbps
	bgrou2 Status	WARNING	06-09-2015 15:51:25	265d 22h 47m 42s	5/5	WARNING: Interface bgrou2 (index 11) is administratively down.
	bgrou3 Bandwidth	OK	06-09-2015 15:51:09	18d 18h 48m 9s	1/5	OK - Current BW in: 0Mbps Out: 0Mbps
	bgrou3 Status	WARNING	06-09-2015 15:52:59	18d 18h 44m 15s	5/5	WARNING: Interface bgrou3 (index 12) is administratively down.



Un exemple de système de supervision - Nagvis



Comment choisir le niveau d'une trace ?



Niveaux couramment utilisés

- VERBOSE
- DEBUG
- INFO
- WARNING
- ERROR



VERBOSE / DEBUG



VERBOSE / DEBUG - EXEMPLE

```
[date] [DEBUG ] Creating shared instance of singleton bean  
            'propertySourcesPlaceholderConfigurer'  
[date] [DEBUG ] Creating shared instance of singleton bean  
            'org.springframework.context.event.internalEventListenerProcessor'  
[date] [DEBUG ] Creating shared instance of singleton bean  
            'org.springframework.context.event.internalEventListenerFactory'  
[date] [DEBUG ] Creating shared instance of singleton bean  
            'org.springframework.context.annotation.internalAutowiredAnnotationProcessor'
```



INFO



Virginie PAGEAUD
Quarkslab

INFO

-  Tâches métier



INFO

-  Tâches métier
-  Intelligible pour l'exploitant·e



INFO - EXEMPLE

```
[date] [INFO ] Traitement du fichier  
          "D:\Documents\GitHub\glycescan\test_files\datafile_03.txt."  
[date] [INFO ] Déplacement du fichier  
          "D:\Documents\GitHub\glycescan\test_files\datafile_03.txt"  
          vers le répertoire de sauvegarde  
          "D:\Documents\GitHub\glycescan\test_files\traite".  
[date] [INFO ] Déplacement réussi.
```



WARNING



Virginie PAGEAUD
Quarkslab

WARNING

-  Alerte



WARNING

- ⚠️ Alerte
- 😬 Fonctionnement dégradé



WARNING

- ⚠️ Alerte
- 😬 Fonctionnement dégradé
- 👁️ Surveillance



ERROR



Virginie PAGEAUD
Quarkslab

devfest 24
Dijon

ERROR

- 🔥 Alerte grave



ERROR

- 🔥 Alerte grave
- ⚠️ Fonctionnement en péril



ERROR

- 🔥 Alerte grave
- ⚠️ Fonctionnement en péril
- 🛠️ Action nécessaire



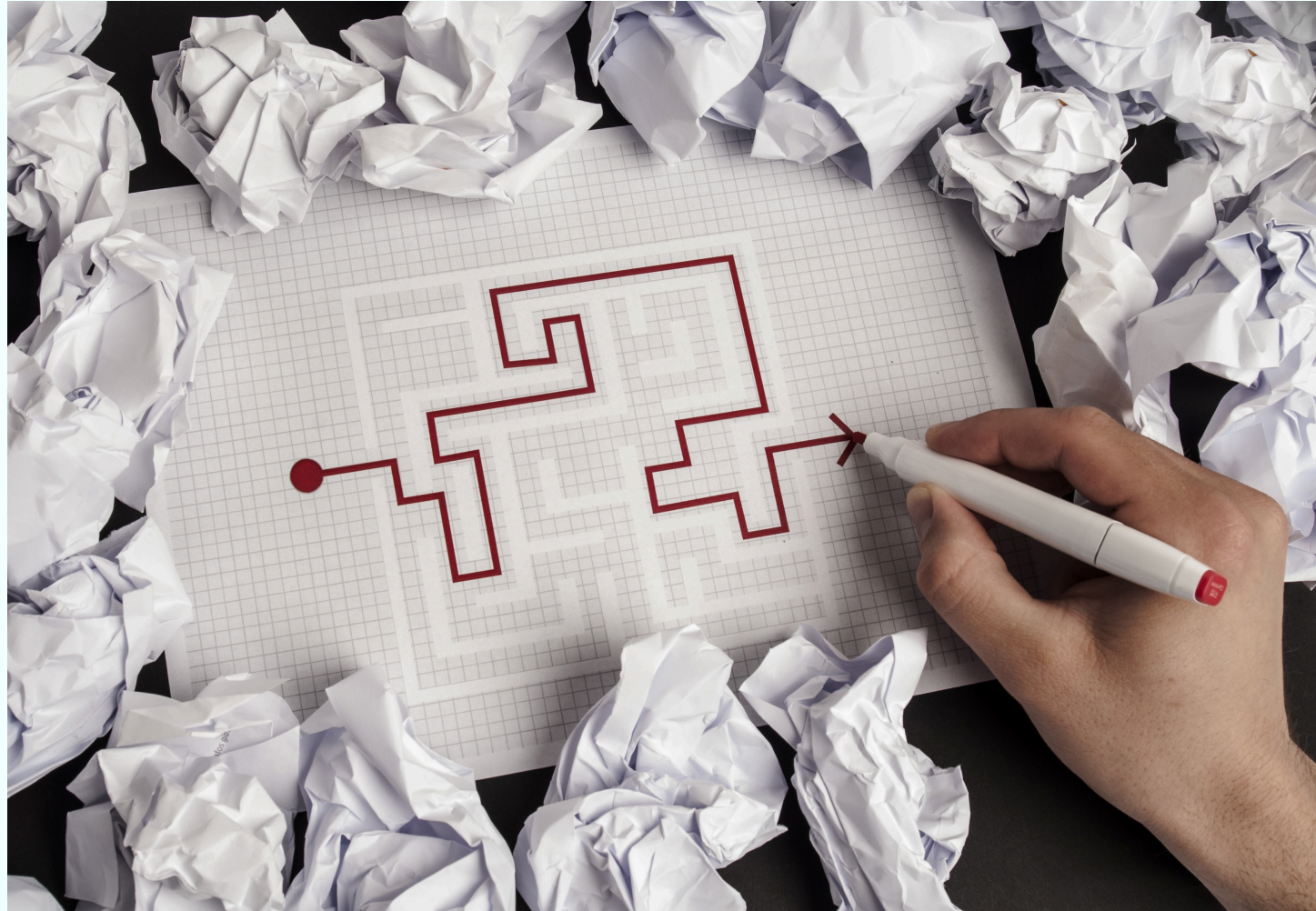
ERROR - EXEMPLE

```
2024-12-06 11:30:00.017 [ERROR ] "D:/bad_directory" n'est pas un répertoire.  
Arrêt du traitement. Veuillez vérifier la configuration.
```



Quelles informations indiquer ?





En C/C++ utilisez les constantes préprocesseur et compilateur !

```
// File HelloThere.cpp
#include <iostream>

void general_kenobi()
{
    printf("WARN   : %s:%s\r\n\r\n",__FILE__,__func__);
    return;
}

int main(int argc, char *argv[])
{
    printf("\r\n");
    printf("INFO   : Quelques pistes pour soigner ses traces.\r\n");
    printf("DEBUG  : %s(%d).\r\n",__FILE__,__LINE__);
    printf("DEBUG  : %s\r\n",__PRETTY_FUNCTION__);
    // printf("DEBUG : %s\r\n",__FUNCSIG__); // Visual C++
    general_kenobi();
    return 0;
}
```




```
PS D:\Documents\GitHub\talks\HelloWorld> .\HelloThere.exe
```

```
INFO : Quelques pistes pour soigner ses traces.
```

```
DEBUG : D:\Documents\GitHub\talks\HelloWorld\HelloThere.cpp(13).
```

```
DEBUG : int main(int, char**)
```

```
WARN : D:\Documents\GitHub\talks\HelloWorld\HelloThere.cpp:general_kenobi
```

```
PS D:\Documents\GitHub\talks\HelloWorld>
```



Plus en détail



Virginie PAGEAUD
Quarkslab

Plus en détail

- Objet métier, identifiant (numéro de commande, nom de fichier...)



Plus en détail

- Objet métier, identifiant (numéro de commande, nom de fichier...)
- Écart valeur reçue / attendue



Plus en détail

- Objet métier, identifiant (numéro de commande, nom de fichier...)
- Écart valeur reçue / attendue
- Traitement métier en cours



Plus en détail

- Objet métier, identifiant (numéro de commande, nom de fichier...)
- Écart valeur reçue / attendue
- Traitement métier en cours
- Actions possibles
→ dans les traces ou en annexe



Plus en détail - Exemple

```
[date] [ERROR ] Champ n°3 incorrect : "2024/12/06"  
n'est pas une date valide. Format attendu "dd/MM/yyyy".  
[date] [ERROR ] Veuillez corriger le fichier  
"mon_fichier.txt" et réessayer.
```



Et la réglementation dans tout ça ?



Et la réglementation dans tout ça ?

- PCI-DSS (Payment Card Industry Data Security Standard)



Et la réglementation dans tout ça ?

- PCI-DSS (Payment Card Industry Data Security Standard)
- CNIL (Commission Nationale de l'Informatique et des Libertés)



Et la réglementation dans tout ça ?

- PCI-DSS (Payment Card Industry Data Security Standard)
- CNIL (Commission Nationale de l'Informatique et des Libertés)
 - RGPD (Règlement général sur la protection des données)



Et la réglementation dans tout ça ?

- PCI-DSS (Payment Card Industry Data Security Standard)
- CNIL (Commission Nationale de l'Informatique et des Libertés)
 - RGPD (Règlement général sur la protection des données)
- CRA (Cyber Resilience Act)



Un exemple dans du vrai code ?

Voyons un outil de lecture de fichier, codé en Java.

(Merci à Benjamin Dauvissat - Twitter @bdauvissat)



```

1 package net.benji.glycescan.service.impl;
2
3 // import des dépendances
4
5 @Service("lectureFichierGlycemie")
6 public class LectureFichierGlycemieImpl implements ILectureFichierGlycemie {
7
8     private static final Logger logger = LogManager.getLogger(LectureFichierGlycemieImpl.class);
9
10    @Value("${local.path.glycemie}")
11    private String path;
12
13    @Value("${local.path.glycemie.done}")
14    private String done;
15
16    @Override
17    public List<Glycemie> importFichier() {
18
19        logger.debug("importFichier début -----");
20
21        List<Glycemie> retour = new ArrayList<>();
22
23        File repertoire = new File(path);
24
25        logger.debug("Vérification que le chemin fourni est bien un répertoire");
26        if (!repertoire.isDirectory()) {
27            logger.error("\\" + repertoire.toString() + "\"
28                + " n'est pas un répertoire. Arrêt du traitement.");
29
30            return retour;
31        }

```



```

3 // import des dépendances
4
5 @Service("lectureFichierGlycemie")
6 public class LectureFichierGlycemieImpl implements ILectureFichierGlycemie {
7
8     private static final Logger logger = LogManager.getLogger(LectureFichierGlycemieImpl.class);
9
10    @Value("${local.path.glycemie}")
11    private String path;
12
13    @Value("${local.path.glycemie.done}")
14    private String done;
15
16    @Override
17    public List<Glycemie> importFichier() {
18
19        logger.debug("importFichier début -----");
20
21        List<Glycemie> retour = new ArrayList<>();
22
23        File repertoire = new File(path);
24
25        logger.debug("Vérification que le chemin fourni est bien un répertoire");
26        if (!repertoire.isDirectory()) {
27            logger.error("\"" + repertoire.toString() + "\"
28                + " n'est pas un répertoire. Arrêt du traitement.");
29
30            return retour;
31        }
32
33        List<File> fichiersDuRepertoire = Arrays.asList(Objects.requireNonNull(repertoire.listFiles()));

```



```

12
13     @Value("${local.path.glycemie.done}")
14     private String done;
15
16     @Override
17     public List<Glycemie> importFichier() {
18
19         logger.debug("importFichier début -----");
20
21         List<Glycemie> retour = new ArrayList<>();
22
23         File repertoire = new File(path);
24
25         logger.debug("Vérification que le chemin fourni est bien un répertoire");
26         if (!repertoire.isDirectory()) {
27             logger.error("\"" + repertoire.toString() + "\"
28                 + " n'est pas un répertoire. Arrêt du traitement.");
29
30             return retour;
31         }
32
33         List<File> fichiersDuRepertoire = Arrays.asList(Objects.requireNonNull(repertoire.listFiles()));
34
35         logger.debug("Vérification de la présence de fichiers à traiter");
36         if (CollectionUtils.isEmpty(fichiersDuRepertoire)) {
37             logger.info("Aucun fichier à traiter dans"
38                 + "\"" + path.toString() + "\".");
39
40             return retour;
41         }
42
43         logger.debug("Traitement des fichiers.");

```




```

23     File repertoire = new File(path);
24
25     logger.debug("Vérification que le chemin fourni est bien un répertoire");
26     if (!repertoire.isDirectory()) {
27         logger.error("\"\" + repertoire.toString() + "\""
28             + " n'est pas un répertoire. Arrêt du traitement.");
29
30         return retour;
31     }
32
33     List<File> fichiersDuRepertoire = Arrays.asList(Objects.requireNonNull(repertoire.listFiles()));
34
35     logger.debug("Vérification de la présence de fichiers à traiter");
36     if (CollectionUtils.isEmpty(fichiersDuRepertoire)) {
37         logger.info("Aucun fichier à traiter dans"
38             + "\"\" + path.toString() + "\".");
39
40         return retour;
41     }
42
43     logger.debug("Traitement des fichiers.");
44     for (File fichier:fichiersDuRepertoire) {
45         logger.debug("Traitement de l'élément "
46             + "\"\" + fichier.getName() + "\".");
47
48         retour.addAll(lectureFichier(fichier));
49     }
50
51     logger.debug("importFichier fin -----");
52     return retour;
53 }

```



```

29
30     return retour;
31 }
32
33 List<File> fichiersDuRepertoire = Arrays.asList(Objects.requireNonNull(repertoire.listFiles()));
34
35 logger.debug("Vérification de la présence de fichiers à traiter");
36 if (CollectionUtils.isEmpty(fichiersDuRepertoire)) {
37     logger.info("Aucun fichier à traiter dans "
38         + "\"" + path.toString() + "\".");
39
40     return retour;
41 }
42
43 logger.debug("Traitement des fichiers.");
44 for (File fichier:fichiersDuRepertoire) {
45     logger.debug("Traitement de l'élément "
46         + "\"" + fichier.getName() + "\".");
47
48     retour.addAll(lectureFichier(fichier));
49 }
50
51 logger.debug("importFichier fin -----");
52 return retour;
53 }
54
55 private List<Glycemie> lectureFichier(File fichier) {
56
57     logger.debug("lectureFichier début -----");
58
59     List<Glycemie> lignesDuFichier = new ArrayList<>();
60

```



```

37         logger.info("Aucun fichier à traiter dans"
38                     + "\"" + path.toString() + "\".");
39
40         return retour;
41     }
42
43     logger.debug("Traitement des fichiers.");
44     for (File fichier:fichiersDuRepertoire) {
45         logger.debug("Traitement de l'élément "
46                     + "\"" + fichier.getName() + "\".");
47
48         retour.addAll(lectureFichier(fichier));
49     }
50
51     logger.debug("importFichier fin -----");
52     return retour;
53 }
54
55 private List<Glycemie> lectureFichier(File fichier) {
56
57     logger.debug("lectureFichier début -----");
58
59     List<Glycemie> lignesDuFichier = new ArrayList<>();
60
61     logger.debug("Vérification qu'il s'agit bien d'un fichier.");
62
63     if (fichier.isDirectory()) {
64         logger.info("\"" + fichier.toString() + "\""
65                 + " est un répertoire. Passage à l'élément suivant de la liste.");
66
67         return lignesDuFichier;

```



```

41     }
42
43     logger.debug("Traitement des fichiers.");
44     for (File fichier:fichiersDuRepertoire) {
45         logger.debug("Traitement de l'élément "
46             + "\"" + fichier.getName() + "\".");
47
48         retour.addAll(lectureFichier(fichier));
49     }
50
51     logger.debug("importFichier fin -----");
52     return retour;
53 }
54
55 private List<Glycemie> lectureFichier(File fichier) {
56
57     logger.debug("lectureFichier début -----");
58
59     List<Glycemie> lignesDuFichier = new ArrayList<>();
60
61     logger.debug("Vérification qu'il s'agit bien d'un fichier.");
62
63     if (fichier.isDirectory()) {
64         logger.info("\"" + fichier.toString() + "\""
65             + " est un répertoire. Passage à l'élément suivant de la liste.");
66
67         return lignesDuFichier;
68     }
69
70     if (!fichier.isFile()) {
71         logger.error("\"" + fichier.toString() + "\"")

```



```

48     retour.addAll(lectureFichier(fichier));
49 }
50
51     logger.debug("importFichier fin -----");
52     return retour;
53 }
54
55 private List<Glycemie> lectureFichier(File fichier) {
56     logger.debug("lectureFichier début -----");
57
58     List<Glycemie> lignesDuFichier = new ArrayList<>();
59
60     logger.debug("Vérification qu'il s'agit bien d'un fichier.");
61
62     if (fichier.isDirectory()) {
63         logger.info("\\" + fichier.toString() + "\"
64             + " est un répertoire. Passage à l'élément suivant de la liste.");
65
66         return lignesDuFichier;
67     }
68
69     if (!fichier.isFile()) {
70         logger.error("\\" + fichier.toString() + "\"
71             + " n'est pas un fichier. Passage à l'élément suivant de la liste.");
72
73         return lignesDuFichier;
74     }
75 }
76
77 try {
78     logger.info("Traitement du fichier " + fichier.toString() + ".");

```



```

56     logger.debug("lectureFichier début -----");
57
58     List<Glycemie> lignesDuFichier = new ArrayList<>();
59
60     logger.debug("Vérification qu'il s'agit bien d'un fichier.");
61
62     if (fichier.isDirectory()) {
63         logger.info("\"" + fichier.toString() + "\""
64             + " est un répertoire. Passage à l'élément suivant de la liste.");
65
66         return lignesDuFichier;
67     }
68
69     if (!fichier.isFile()) {
70         logger.error("\"" + fichier.toString() + "\""
71             + " n'est pas un fichier. Passage à l'élément suivant de la liste.");
72
73         return lignesDuFichier;
74     }
75
76     try {
77         logger.info("Traitement du fichier " + fichier.toString() + ".");
78         FileInputStream fis = new FileInputStream(fichier);
79
80         InputStreamReader isr = new InputStreamReader(fis, StandardCharsets.UTF_8);
81         BufferedReader br = new BufferedReader(isr);
82
83         String line;
84         int numLigne = 0;
85
86

```



```

62
63     if (fichier.isDirectory()) {
64         logger.info "\"" + fichier.toString() + "\""
65             + " est un répertoire. Passage à l'élément suivant de la liste.");
66
67         return lignesDuFichier;
68     }
69
70     if (!fichier.isFile()) {
71         logger.error "\"" + fichier.toString() + "\""
72             + " n'est pas un fichier. Passage à l'élément suivant de la liste.");
73
74         return lignesDuFichier;
75     }
76
77     try {
78         logger.info("Traitement du fichier " + fichier.toString() + ".");
79         FileInputStream fis = new FileInputStream(fichier);
80
81         InputStreamReader isr = new InputStreamReader(fis, StandardCharsets.UTF_8);
82         BufferedReader br = new BufferedReader(isr);
83
84         String line;
85         int numLigne = 0;
86
87         while ( ( line = br.readLine() ) != null )
88         {
89             String[] ligne = line.split("\t", -1);
90             numLigne++;
91
92             logger.debug("Ligne non vide, appel à transformeLigne.");
93             GluegunRecord transformeLigne(ligne, numLigne);

```



```

77     try {
78         logger.info("Traitement du fichier " + fichier.toString() + ".");
79         FileInputStream fis = new FileInputStream(fichier);
80
81         InputStreamReader isr = new InputStreamReader(fis, StandardCharsets.UTF_8);
82         BufferedReader br = new BufferedReader(isr);
83
84         String line;
85         int numLigne = 0;
86
87         while ( ( line = br.readLine() ) != null )
88         {
89             String[] ligne = line.split("\t", -1);
90             numLigne++;
91
92             logger.debug("Ligne non vide, appel à transformeLigne.");
93             Glycemie record = transformeLigne(ligne, numLigne);
94
95             if (record != null) {
96                 lignesDuFichier.add(record);
97                 logger.debug("Décodage de la ligne effectué, ajout du record à lignesDuFichier.");
98             }
99             else {
100                 logger.error("Erreur de décodage de la ligne " + numLigne + " du fichier " + fichier.toString() +
101                     " : ");
102                 logger.error("-" + line + "-");
103             }
104         }
105
106         fis.close();
107
108     } catch (IOException e) {

```




```

86
87     while ( ( line = br.readLine() ) != null )
88     {
89         String[] ligne = line.split("\t", -1);
90         numLigne++;
91
92         logger.debug("Ligne non vide, appel à transformeLigne.");
93         Glycemie record = transformeLigne(ligne, numLigne);
94
95         if (record != null) {
96             lignesDuFichier.add(record);
97             logger.debug("Décodage de la ligne effectué, ajout du record à lignesDuFichier.");
98         }
99         else {
100             logger.error("Erreur de décodage de la ligne " + numLigne + " du fichier " + fichier.toString() +
101                 logger.error("Contenu en erreur : ");
102                 logger.error("-" + line + "-");
103         }
104     }
105
106     fis.close();
107
108 } catch (IOException e) {
109     logger.error("Une erreur est survenue lors du traitement du fichier "
110         + fichier.toString() + ".");
111
112     e.printStackTrace();
113 } finally {
114     // Déplacement du fichier
115     File destination = new File(path + "/" + done);
116     boolean existeDone = destination.exists();

```



```

95         if (record != null) {
96             lignesDuFichier.add(record);
97             logger.debug("Décodage de la ligne effectué, ajout du record à lignesDuFichier.");
98         }
99         else {
100             logger.error("Erreur de décodage de la ligne " + numLigne + " du fichier " + fichier.toString() +
101                 " ");
102             logger.error("Contenu en erreur : ");
103             logger.error("-" + line + "-");
104         }
105     }
106     fis.close();
107
108 } catch (IOException e) {
109     logger.error("Une erreur est survenue lors du traitement du fichier "
110         + fichier.toString() + ".");
111
112     e.printStackTrace();
113 } finally {
114     // Déplacement du fichier
115     File destination = new File(path + "/" + done);
116     boolean existeDone = destination.exists();
117
118     if (!existeDone) {
119         logger.debug("Création du répertoire de destination.");
120         existeDone = destination.mkdir();
121     }
122
123     if (existeDone) {
124         boolean move = fichier.renameTo(new File(destination + "/" + fichier.getName()));
125
126         if (move) {

```



```

104     }
105
106     fis.close();
107
108     } catch (IOException e) {
109         logger.error("Une erreur est survenue lors du traitement du fichier "
110             + fichier.toString() + ".");
111
112         e.printStackTrace();
113     } finally {
114         // Déplacement du fichier
115         File destination = new File(path + "/" + done);
116         boolean existeDone = destination.exists();
117
118         if (!existeDone) {
119             logger.debug("Création du répertoire de destination.");
120             existeDone = destination.mkdir();
121         }
122
123         if (existeDone) {
124             boolean move = fichier.renameTo(new File(destination + "/" + fichier.getName()));
125
126             if (move) {
127                 logger.info("Déplacement du fichier "
128                     + "\"" + fichier.toString() + "\""
129                     + " vers le répertoire de sauvegarde "
130                     + "\"" + destination.toString() + "\""
131                     + " réussi");
132             }
133             else {
134                 logger.error("Échec du déplacement du fichier "
135                     + "\"" + fichier.toString() + "\"");

```



```

114 // Déplacement du fichier
115 File destination = new File(path + "/" + done);
116 boolean existeDone = destination.exists();
117
118 if (!existeDone) {
119     logger.debug("Création du répertoire de destination.");
120     existeDone = destination.mkdir();
121 }
122
123 if (existeDone) {
124     boolean move = fichier.renameTo(new File(destination + "/" + fichier.getName()));
125
126     if (move) {
127         logger.info("Déplacement du fichier "
128             + "\"" + fichier.toString() + "\""
129             + " vers le répertoire de sauvegarde "
130             + "\"" + destination.toString() + "\""
131             + " réussi");
132     }
133     else {
134         logger.error("Échec du déplacement du fichier "
135             + "\"" + fichier.toString() + "\""
136             + " vers le répertoire de sauvegarde "
137             + "\"" + destination.toString() + "\"" + ".");
138     }
139 }
140 }
141 logger.debug("lectureFichier fin -----");
142 return lignesDuFichier;
143 }
144 }

```



```

114 // Déplacement du fichier
115 File destination = new File(path + "/" + done);
116 boolean existeDone = destination.exists();
117
118 if (!existeDone) {
119     logger.debug("Création du répertoire de destination.");
120     existeDone = destination.mkdir();
121 }
122
123 if (existeDone) {
124     boolean move = fichier.renameTo(new File(destination + "/" + fichier.getName()));
125
126     if (move) {
127         logger.info("Déplacement du fichier "
128             + "\"" + fichier.toString() + "\""
129             + " vers le répertoire de sauvegarde "
130             + "\"" + destination.toString() + "\""
131             + " réussi");
132     }
133     else {
134         logger.error("Échec du déplacement du fichier "
135             + "\"" + fichier.toString() + "\""
136             + " vers le répertoire de sauvegarde "
137             + "\"" + destination.toString() + "\"" + ".");
138     }
139 }
140 }
141 logger.debug("lectureFichier fin -----");
142 return lignesDuFichier;
143 }
144 }

```



```

114 // Déplacement du fichier
115 File destination = new File(path + "/" + done);
116 boolean existeDone = destination.exists();
117
118 if (!existeDone) {
119     logger.debug("Création du répertoire de destination.");
120     existeDone = destination.mkdir();
121 }
122
123 if (existeDone) {
124     boolean move = fichier.renameTo(new File(destination + "/" + fichier.getName()));
125
126     if (move) {
127         logger.info("Déplacement du fichier "
128             + "\"" + fichier.toString() + "\""
129             + " vers le répertoire de sauvegarde "
130             + "\"" + destination.toString() + "\""
131             + " réussi");
132     }
133     else {
134         logger.error("Échec du déplacement du fichier "
135             + "\"" + fichier.toString() + "\""
136             + " vers le répertoire de sauvegarde "
137             + "\"" + destination.toString() + "\"" + ".");
138     }
139 }
140 }
141 logger.debug("lectureFichier fin -----");
142 return lignesDuFichier;
143 }
144 }

```



Exemple d'output - Niveau debug

```
[date] [INFO ] [main] [n.b.g.GlycescanApplication(61)] - Started GlycescanApplication in 3.32 seconds (JVM running for 5
[date] [DEBUG ] [...] - importFichier début -----
[date] [DEBUG ] [...] - Vérification que le chemin fourni est bien un répertoire
[date] [DEBUG ] [...] - Vérification de la présence de fichiers à traiter
[date] [DEBUG ] [...] - Traitement des fichiers.
[date] [DEBUG ] [...] - Traitement de l'élément "datafile_NOK.txt".
[date] [DEBUG ] [...] - lectureFichier début -----
[date] [DEBUG ] [...] - Vérification qu'il s'agit bien d'un fichier.
[date] [INFO ] [...] - Traitement du fichier D:\Documents\GitHub\glycescan\test_files\input\datafile_NOK.txt.
[date] [DEBUG ] [...] - Ligne non vide, appel à transformeLigne.
[date] [ERROR ] [...] - Erreur de décodage de la ligne 1 du fichier D:\Documents\GitHub\glycescan\test_files\input\datafile
[date] [ERROR ] [...] - Contenu en erreur :
[date] [ERROR ] [...] - -5 2024/12/05 16:10 1 3.2 3.1 4.8 1.2 6 125 4.1 Super commentaire 1.0 0.5 3.4 3.1 6 0.9 4-
[date] [INFO ] [...] - Déplacement du fichier "D:\Documents\GitHub\glycescan\test_files\input\datafile_NOK.txt" vers le r
[date] [DEBUG ] [...] - lectureFichier fin -----
[date] [DEBUG ] [...] - Traitement de l'élément "datafile_OK.txt".
[date] [DEBUG ] [...] - lectureFichier début -----
[date] [DEBUG ] [...] - Vérification qu'il s'agit bien d'un fichier.
[date] [INFO ] [...] - Traitement du fichier D:\Documents\GitHub\glycescan\test_files\input\datafile_OK.txt.
[date] [DEBUG ] [...] - Ligne non vide, appel à transformeLigne.
[date] [DEBUG ] [...] - Décodage de la ligne effectué, ajout du record à lignesDuFichier.
[date] [INFO ] [...] - Déplacement du fichier "D:\Documents\GitHub\glycescan\test_files\input\datafile_OK.txt" vers le ré
[date] [DEBUG ] [...] - lectureFichier fin -----
[date] [DEBUG ] [...] - importFichier fin -----
```



Exemple d'output - Niveau info

```
[date] [INFO ] [main] [n.b.g.GlycescanApplication(61)] - Started GlycescanApplication in 3.32 seconds (JVM running for 5
[date] [INFO ] [...] - Traitement du fichier D:\Documents\GitHub\glycescan\test_files\input\datafile_NOK.txt.
[date] [ERROR] [...] - Erreur de décodage de la ligne 1 du fichier D:\Documents\GitHub\glycescan\test_files\input\datafil
[date] [ERROR] [...] - Contenu en erreur :
[date] [ERROR] [...] - -5 2024/12/05 16:10 1 3.2 3.1 4.8 1.2 6 125 4.1 Super commentaire 1.0 0.5 3.4 3.1 6 0.9 4- vers
[date] [INFO ] [...] - Traitement du fichier D:\Documents\GitHub\glycescan\test_files\input\datafile_OK.txt.
[date] [INFO ] [...] - Déplacement du fichier "D:\Documents\GitHub\glycescan\test_files\input\datafile_OK.txt" vers le ré
```



Exemple d'output - Niveau error

```
[date] [ERROR ] [main] [n.b.g.GlycescanApplication(61)] - Erreur de décodage de la ligne 1 du fichier D:\Documents\GitHub\  
[date] [ERROR ] [...] - Contenu en erreur :  
[date] [ERROR ] [...] - -5 2024/12/05 16:10 1 3.2 3.1 4.8 1.2 6 125 4.1 Super commentaire 1.0 0.5 3.4 3.1 6 0.9 4-
```



Synthèse



Virginie PAGEAUD
Quarkslab

Synthèse

- DEBUG, VERBOSE en développement 🧑💻






Synthèse

- DEBUG, VERBOSE en développement 
- INFO, WARNING, ERROR en exploitation 



Synthèse

- DEBUG, VERBOSE en développement 
- INFO, WARNING, ERROR en exploitation 
- WARNING si sans impact immédiat 



Synthèse

- DEBUG, VERBOSE en développement 🧑💻
- INFO, WARNING, ERROR en exploitation 🏭
- WARNING si sans impact immédiat ⚠️
- ERROR 🔥 si action nécessaire pour remédiation 🚒



Synthèse

- DEBUG, VERBOSE en développement 🧑💻
- INFO, WARNING, ERROR en exploitation 🏭
- WARNING si sans impact immédiat ⚠️
- ERROR 🔥 si action nécessaire pour remédiation 🚒
- penser aux constantes magiques du langage ✨



Synthèse

- DEBUG, VERBOSE en développement 🧑💻
- INFO, WARNING, ERROR en exploitation 🏭
- WARNING si sans impact immédiat ⚠️
- ERROR 🔥 si action nécessaire pour remédiation 🚒
- penser aux constantes magiques du langage ✨
- identifier les objets et traitements métier 🛠️



Synthèse

- DEBUG, VERBOSE en développement 🧑💻
- INFO, WARNING, ERROR en exploitation 🏭
- WARNING si sans impact immédiat ⚠️
- ERROR 🔥 si action nécessaire pour remédiation 🚒
- penser aux constantes magiques du langage ✨
- identifier les objets et traitements métier 🛠️
- inclure les logs dans la definition of done 📄



Conclusion

Profitons de notre métier pour en apprendre d'autres !





Virginie PAGEAUD
Quarkslab



Virginie PAGEAUD
Quarkslab

Me contacter

LinkedIn [Virginie \(CASAVECCHIA\) PAGEAUD](#)

Bluesky [@la-fee-dragee.bsky.social](#)

Podcast PunkinDev
de Sylvain Coudert



Feedback :

