

Moodle (traitement recommandé - depuis la table des logs)

- Travail côté moodle
 - Récupération des cours et des noms de modules
 - Mise en place d'une table des logs temporaire (facultatif)
- Intégration dans Agimus
 - Envoi des données dans elasticsearch
 - Création du nouveau tableau de bord
- Exemple de mise en pratique
 - Université de La Réunion

Après quelques tests de traitement des logs apache de moodle, il nous a semblé plus pertinent de travailler directement sur la base des logs de moodle.

Nous en profitons pour extraire davantage d'informations ce qui rend les indicateurs plus parlants. Il est nécessaire pour intégrer cette solution de travailler côté moodle mais elle présente l'avantage de ne pas être intrusive et de n'utiliser que la consultation de la base moodle.

Travail côté moodle



Si vous ne souhaitez pas effectuer de traitement côté moodle, vous pouvez regarder la solution mise en place par l'université de La Réunion et disponible en bas de cette page.

Récupération des cours et des noms de modules

La récupération des noms de modules n'est à faire qu'une seule fois. Le script python [liste_modules.py](#) (à paramétrer) permet de générer un fichier yaml (moodle-activites.yaml) qui sera utilisé dans le traitement logstash pour associer un nom lisible à un module.

Vous pouvez compléter/modifier la liste générée, y compris en ajoutant le wildcard *. Exemple :

Tous les modules booktool renverront la valeur "Livre" et les tool_ seront des outils admin

```
"booktool_*": "Livre"
"tool_*": "Outil admin"
```

Le traitement des cours est fait quotidiennement afin de conserver également une évolution de la plate-forme.

Le script python [liste_cours.py](#) (à paramétrer) permet de générer la liste des cours de moodle que vous enverrez ensuite sur le serveur agimus. Chaque ligne du fichier correspond à un cours et aura le format suivant :

```
[cours:id_cours];[nom_cours:nom_cours];[cat:id_cat_parente];[enseignants:logins_enseignants];[activites:
id_act_1=type_act_1|id_act_2=type_act_2|id_act_3=type_act_1|id_act_4=type_act_1|id_act_5=type_act_2];
[type_activites:type_act_1=3|type_act_2=2];[actif:1];[niv2:id_cat_niv2];[niv1:id_cat_niv_1]
```

Exemple :

```
[cours:16];[nom_cours:Agimus facile];[cat:77];[enseignants:enseignant1|enseignant2];[activites:
315=Forum|66789=Fichier|66791=Fichier|66792=Fichier|66795=Fichier|66796=Fichier|218286=URL|232492=Fichier|240718
=Fichier|291614=Fichier|294869=Fichier|295067=Fichier|298778=Fichier|298779=Fichier|302150=Fichier|312374=Fichie
r|314361=Fichier|315952=Fichier|317008=Fichier|317029=Fichier|318624=Fichier|321352=Fichier];[type_activites:
Forum=1|Fichier=20|URL=1];[actif:1];[niv2:1309];[niv1:75]
```

Lorsque vous exécuterez ce script et enverrez le résultat sur le serveur Agimus, prenez garde à ne pas exécuter le traitement logstash avant que l'envoi ne soit fait.

Mise en place d'une table des logs temporaire (facultatif)



Si vous n'utilisez pas de table temporaire, vous pouvez supprimer l'appel au script `purge_db_logs.py` qui nettoie cette table temporaire afin d'accélérer les traitements.

Cette étape est facultative mais a le mérite de faciliter le traitement des logs. Nous utilisons la possibilité offerte par moodle de stocker une partie des logs dans une base externe en plus de la base habituelle. Ceci va nous permettre de traiter quotidiennement la totalité des logs depuis notre dernier traitement. Il suffira de supprimer les logs de cette nouvelle base à la fin du traitement logstash.

Pour mettre en place cette nouvelle base, connectez-vous à la page de gestion des stockages d'historiques (moodle.univ.fr/admin/settings.php?section=managelogging)

Gérer les stockages d'historiques

Stockages d'historiques disponibles

Nom	Rapports supportés	Version	Activer	Vers le haut/Vers le bas	Paramètres	Désinstaller
Historiques standards	Historiques, Historiques en direct, Activités du cours, Participation au cours, Statistiques	2014111000		↓	Paramètres	Désinstaller
Historique dans base de données externe	Historiques, Historiques en direct	2014111000		↑	Paramètres	Désinstaller
Historiques obsolètes	Historiques, Historiques en direct, Activités du cours, Participation au cours, Statistiques	2014111000			Paramètres	Désinstaller

Veuillez activer tous les plugins nécessaires et les placer dans l'ordre adéquat.
Les modifications dans la table ci-dessus sont enregistrées automatiquement.

Puis dans Paramètres, vous renseignerez les paramètres d'accès à cette base ainsi que les filtres suivants :

Filtrer les historiques

Activer des filtres qui empêchent certaines actions d'être consignées dans les historiques.

Enregistrer les actions des utilisateurs anonymes

☐ Défaut : Non

logstore_database | logguests

Inclure les interactions de ces types

- ☒ Enseignement
- ☒ Participation
- ☒ Autre

Défaut : Enseignement, Participation, Autre

logstore_database | includelevels

Inclure les actions de ces types

- ☒ Créer
- ☒ Lire
- ☒ Modifier
- ☒ Supprimer

Défaut : Créer, Lire, Modifier, Supprimer

logstore_database | includeactions

Intégration dans Agimus

Envoi des données dans elasticsearch

1. Logstash va devoir se connecter à la base de données de moodle pour récupérer ses informations. Pour cela, nous utilisons le [plugin input jdbc](#). Pour qu'il puisse fonctionner il faut télécharger le driver correspondant à votre installation et paramétrer correctement les configurations logstash correspondantes. Un exemple est donné dans les fichiers sur github.
2. Il faut exécuter le script de récupération des catégories moodle. Vous pouvez l'intégrer dans le traitement quotidien ou l'exécuter ponctuellement. L'important est que le fichier résultat (logstash/maps/moodle-activites.yml) soit présent

```
logstash -f logstash/gen_mappings/moodle_categories/
```

3. Ajouter le nouveau template elasticsearch

template moodle pour elasticsearch

```
PUT _template/moodledb
{
  "order" : 2,
  "index_patterns" : [
    "ag-moodledb-*"
  ],
  "settings" : { },
  "mappings" : {
    "properties" : {
      "mdl_courseid" : {
        "type" : "integer"
      },
      "mdl_id" : {
        "type" : "integer"
      },
      "mdl_contextinstanceid" : {
        "type" : "integer"
      },
      "mdl_objectid" : {
        "type" : "integer"
      },
      "mdl_contextid" : {
        "type" : "integer"
      },
      "mdl_actif" : {
        "type" : "boolean"
      }
    }
  },
  "aliases" : { }
}
```

4. Vous trouverez l'appel à ce traitement quotidien dans le fichier de batch [daily_batch.sh](#)

Création du nouveau tableau de bord

Le nouveau tableaux de bord est disponible sur le github : <https://github.com/EsupPortail/agimus-ng/blob/master/kibana/dashboard/Moodle.json>

Il faut également importer les visualisations associées disponibles ici : <https://github.com/EsupPortail/agimus-ng/tree/master/kibana/visualization>

Exemple de mise en pratique

Université de La Réunion

L'université de La Réunion a également intégré un tableau de bord moodle basé sur la table des logs. Vous pouvez en voir le rendu sur la page [5 - Les tableaux de bord disponibles](#)

L'approche est légèrement différente de celle proposée ci-dessus car il n'y a pas de traitement sur le serveur moodle. Il suffit de permettre un accès à la base de données par logstash.

Si vous souhaitez vous en inspirer pour créer votre propre tableau de bord, vous pouvez utiliser les documents en libre accès sur l'espace github du projet :

Configurations logstash :

- <https://github.com/EsupPortail/agimus-ng/blob/master/logstash/logstash-moodle-la-reunion-metadata.conf> (récupération des méta-données des utilisateurs, cours, catégories, forum,quiz)
- <https://github.com/EsupPortail/agimus-ng/blob/master/logstash/logstash-moodle-la-reunion-logtable.conf> (traitement des logs)

Configurations kibana :

- Dashboard : <https://github.com/EsupPortail/agimus-ng/blob/master/kibana/dashboard/La-reunion/Moodle2015-Log-Table.json>
- Visualisations : <https://github.com/EsupPortail/agimus-ng/tree/master/kibana/visualization/La-reunion>