



Mini-projet

## M2.1 - Datamétrie

### Table des matières

<b>1 -CAHIER DES CHARGES DU SCRIPT DATAMÉTRIE .....</b>	<b>2</b>
1.1 -PRÉSENTATION .....	2
1.2 -EXEMPLE D'EMAIL DE DATAMÉTRIE .....	3
1.3 -TRAVAIL DEMANDÉ .....	4
1.3.1 -Composition des équipes .....	4
1.3.2 -Algorithme .....	4
1.3.3 -Décomposition en tâche élémentaire .....	4
1.3.4 -Codage des tâches élémentaires .....	4
1.3.5 -Fin de la première itération et itérations suivantes .....	4
1.3.6 -Fin du mini-projet .....	4
1.4 -RESSOURCES TECHNIQUES .....	4

# 1 - Cahier des charges du script Datamétrie

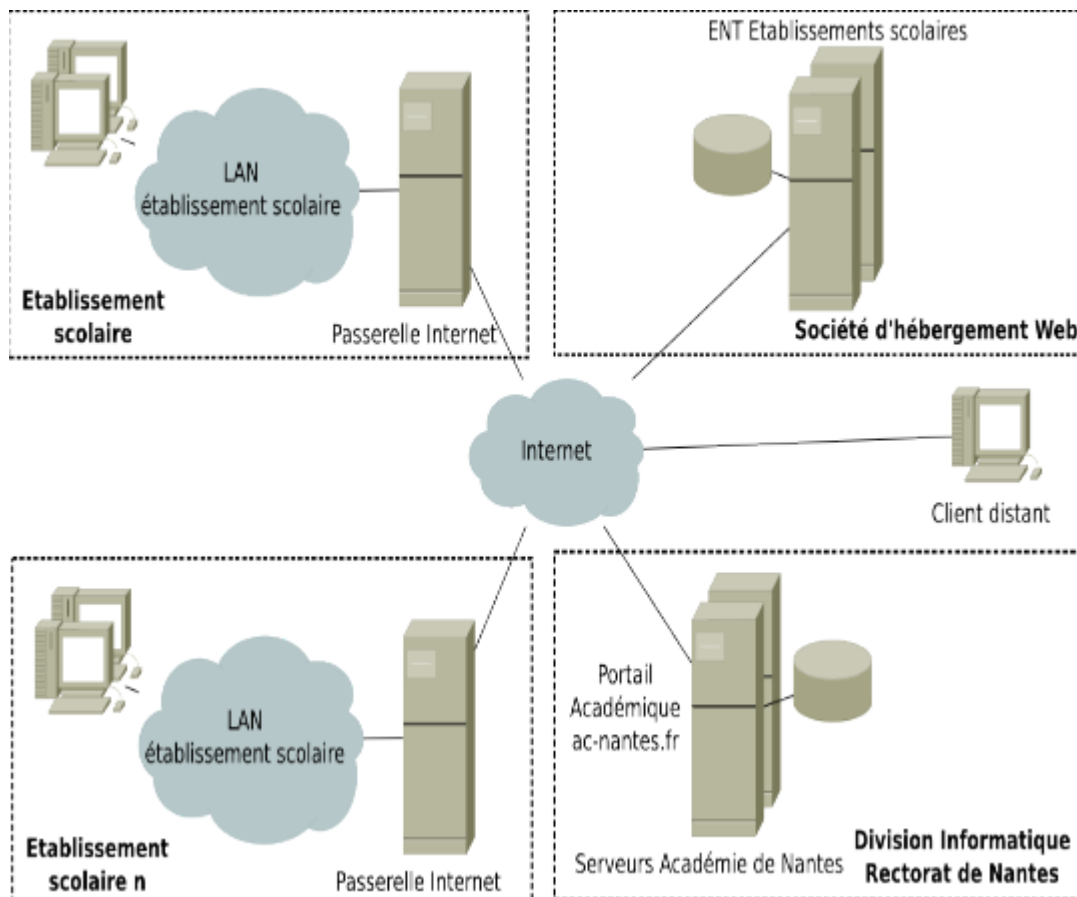
## 1.1 - Présentation

Le service DSI-Cinetic (Cellule d'innovation numérique pour l'enseignement et les TIC ) du Rectorat de Nantes souhaite être informé sur les temps de chargement des pages d'accueil de certains services Web hébergés dans le service informatique du Rectorat ou chez des prestataires de services (OVH, Oceanet, ...) à partir d'un panel d'établissements scolaires.

- Exemples de services web du Rectorat de Nantes :
  - <http://www.ac-nantes.fr/>
  - <http://www.pedagogie.ac-nantes.fr/>
  - <http://b2i-44.ac-nantes.fr/>
  - ...
- Exemples de sites web d'établissements scolaires :
  - <http://www.appert44.org/>
  - ...

Ces mesures de temps de chargement permettront de mettre en évidence des problèmes comme :

- Accès internet perturbé dans certains établissements scolaires à certaines heures de la journée.
- Fournisseur d'accès moins performants que d'autres.
- Services du Rectorat difficiles d'accès à certaines heures de la journée.
- Etc.



Le programme sera réalisé sous la forme d'un script Shell implanté sur la passerelle d'accès internet de chaque lycée test. Dans une première version, le script réalisera les tâches suivantes toutes les 5 mn :

- Lancer un ping vers le serveur à tester et mesurer le temps de réponse du ping.
- Si le serveur est joignable, lancer le chargement de la page d'accueil du service web testé et mesurer le temps de chargement de la page d'accueil du service web testé.
- Les informations suivantes seront à mémoriser afin de fabriquer un rapport par 24H :

- la plus petite et la plus grande des grandeurs mesurées (ping et temps de chargement) ainsi que la datation de ces mesures.
- le nombre de tests fructueux et infructueux. Un test sera considéré comme fructueux si le temps de réponse du test est inférieur à un temps de réponse maximum fixé dans le script
- les moyennes des temps de réponse mesurés

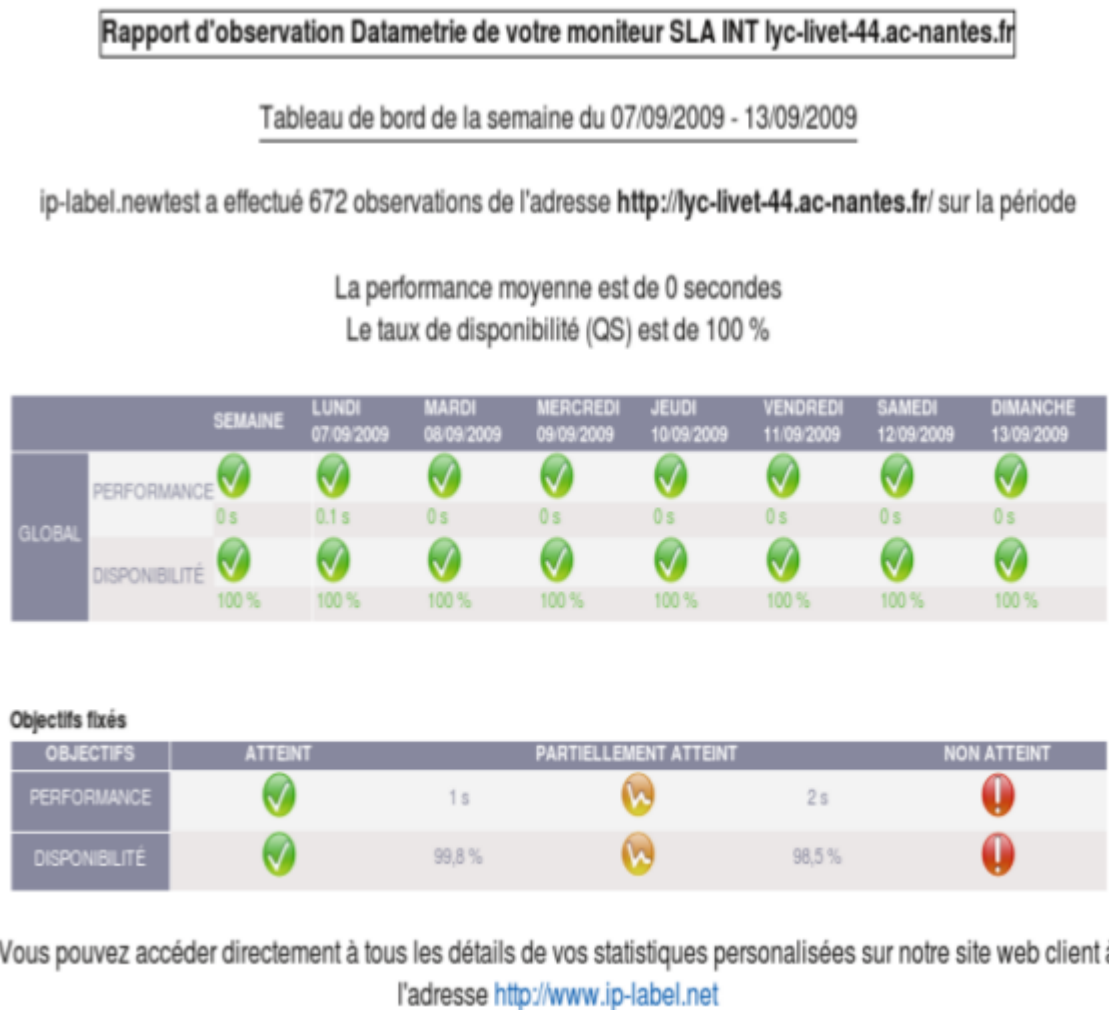
Toute les 24H, le script enverra un email récapitulatif aux responsables du service DSI-Cinetic avec les principales informations recueillies. On devra y retrouver aussi le nom du lycée, ainsi que le fournisseur d'accès Internet (si possible extrait de manière automatique).

Il ne peut y avoir qu'une seule instance du script à un instant donné.

Le script devra utiliser le minimum de ressources systèmes (CPU et mémoire).

## 1.2 - Exemple d'email de datamétrie

Le service DSI-Cinetic est déjà abonné à un service de datamétrie et reçoit régulièrement par mails des rapports comme celui-ci :



On se servira de cet exemple pour le style de rapport à générer.

## 1.3 - Travail demandé

Le travail se fera par équipe de 3 ou 4 développeurs sur 15 heures environ.

L'évaluation sera faite sous la forme d'une soutenance orale (30 mn par équipe) pendant laquelle le travail réalisé sera présenté (diaporama) devant le groupe et un jury d'enseignants.

### 1.3.1 - Composition des équipes

- Proposez la composition des équipes.
- Les équipes sont concurrentes et ne devront pas communiquer leurs résultats aux autres équipes.

### **1.3.2 - Algorithme**

- Par équipe, proposez un algorithme complet du script Datamétrie prenant soin de découper la problématique en un enchaînement de fonctions élémentaires.
- Vous utiliserez au maximum des structures algorithmiques simples (SI ALORS SINON, TANT QUE).
- Vous pouvez commencer avec une version simplifiée de l'algorithme demandé au final.

### **1.3.3 - Décomposition en tâche élémentaire**

- Il est primordial de commencer le travail par un découpage soigneux en tâches élémentaires et éventuellement pour une tâche complexe, il est bon de prévoir une ou plusieurs étapes intermédiaires (ex. envoyer un email au format HTML = envoyer un mail au format texte + enrichir le mail texte de balises HTML).
- Ces tâches élémentaires doivent correspondre à des fonctions du script à développer.
- Les tâches élémentaires doivent être les plus indépendantes possible des autres tâches.
- Numérotez et classez par ordre de priorité l'ensemble des tâches découvertes lors de l'analyse de l'algorithme.
- Chaque tâche élémentaire sera écrite sur un post'it.
- Faites un choix de n tâches élémentaires permettant d'obtenir une première version fonctionnelle du script. Ces tâches seront à coder en priorité lors d'une première itération de codage (sprint).

### **1.3.4 - Codage des tâches élémentaires**

- Pour la première itération, chaque stagiaire choisit une tâche élémentaire parmi celle à réaliser pendant l'itération.
- Chaque stagiaire code et teste le script associé à la tâche.
- Un développeur n'a le droit de reprendre une tâche (post'it) que lorsque la précédente est testée et validée par l'équipe.

### **1.3.5 - Fin de la première itération et itérations suivantes**

- A la fin de la première itération, une première version du script devra être opérationnelle, même si cette version ne réalise pas l'ensemble des fonctions demandées dans le cahier des charges.
- Cette version sera présentée au formateur pour une première démonstration.
- Si la démonstration est concluante, l'équipe proposera la liste des fonctionnalités à inclure dans la version de l'itération suivante.
- Un cycle de codage reprend alors.

### **1.3.6 - Fin du mini-projet**

Après plusieurs itérations, une fois que le script de Datametrie possède suffisamment de fonctionnalités, l'équipe se consacre à la rédaction du diaporama résumant les principales étapes du projet (10-15 mn maxi par équipe).

## **1.4 - Ressources techniques**

- Commandes shell utiles pour le script :
  - ping
  - wget
  - date
  - mail
  - awk
  - sleep
  - bc
  - time