



Pourquoi il n'est pas si simple d'envoyer une donnée ?

[Mise à jour le 16/8/2022]

- **Source** : Mooc Fun "Programmer l'internet des objets"
- **Vidéo** sur YouTube: [Qu'est-ce que La sérialisation des données ?](#)

Envoyer une donnée sur un réseau n'est pas aussi simple que l'on croit. (voir la **vidéo**)

Il faut faire la différence entre le format utilisé pour stocker des données dans la mémoire de l'ordinateur et celui employé pour l'envoyer à une autre machine. En effet, **chaque machine à sa propre représentation** souvent liée aux capacités de son processeur. Cela est **surtout vrai pour les nombres**. Ils peuvent être stockés sur un nombre de bits plus ou moins important ou peuvent être représentés en mémoire de manière optimisée pour accélérer leur traitement.

En revanche, la représentation des **chaînes de caractères (non accentués)** est relativement uniforme, car elle se base sur le code **ASCII** qui est le même pour tous les ordinateurs.

Un texte de base est facilement compréhensible par toutes les machines.

Une solution serait donc de n'utiliser que des chaînes de caractères. Par exemple, si l'on veut envoyer l'entier ayant pour valeur **123**, il existe plusieurs représentations possibles :

- envoyer une chaîne de caractères **"123"** contenant les chiffres du nombre ;
- envoyer la valeur binaire **1111011**.

On voit que juste pour transmettre une simple valeur stockée dans la mémoire d'un ordinateur, il existe plusieurs options et évidemment pour que cette valeur soit interprétée de la bonne façon, **il faut que les deux extrémités se soient mises d'accord sur une représentation**.

ATTENTION

Quand on veut transmettre plusieurs valeurs, c'est-à-dire quand on a des données structurées, d'autres problèmes surviennent.

Par exemple :

- Quelle est la taille des blocs que l'on va transmettre ?
- Comment indiquer la fin de la transmission ?
- Pour une chaîne de caractères, comment indiquer qu'elle se termine ?

Autre exemple : si l'on veut transmettre "12" puis "3", comment faire pour que l'autre extrémité ne comprenne pas "123" ?

Pour que la transmission se fasse correctement, il faut que l'émetteur et le récepteur adoptent les mêmes conventions.

Quand il s'agit d'un ensemble de données, il faut être capable de les séparer. Avec les **tableurs**, une première méthode est possible avec la **notation CSV** (pour **Comma Separated Values**). Comme son nom l'indique, les valeurs sont séparées par des virgules. Les valeurs sont représentées par des chaînes de caractères. Les textes sont différenciés des valeurs numériques, par l'utilisation de guillemets. Ainsi, 123 sera interprété comme un nombre et "123" comme un texte.



Si cette représentation est adaptée aux tableurs, elle est relativement pauvre, car elle ne permet de représenter que des valeurs sur des lignes et des colonnes.

Pour les usages du Web, il a fallu trouver un format plus souple permettant de représenter des structures de données complexes. Évidemment, comme rien n'est simple, il en existe plusieurs et les applications échangeant des données devront utiliser le même.



Conclusion : on voit que **l'envoi de la chaîne de caractères ne suffit pas**, il faut la **formater** pour que le récepteur puisse trouver le type de la donnée transmise, qu'un nombre ne soit pas interprété comme une chaîne de caractères, qu'une chaîne de caractères reste une chaîne de caractères même si elle ne contient que des chiffres.

From:

<https://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<https://webge.fr/dokuwiki/doku.php?id=reseaux:generalites:envoidonnee>

Last update: **2022/09/07 19:25**

