

# Spaced Repetition System

11 novembre 2013

# SRS en 1 schema

(boites de plus en plus grandes)

# Pourquoi utiliser le SRS ?

## Problème == le cerveau

- Apprendre 1000 définitions prend un temps fou
- L'apprentissage passif ne garantit pas une mémoire à long terme

## Solution == SRS

- La répétition en apprentissage actif assure (à priori) des souvenirs plus solides
- L'espacement permet de réduire le nombre de sollicitations du cerveau

# Principes

## Apprendre $\neq$ comprendre

On suppose que ce qui est à retenir est déjà compris (ou n'a pas besoin d'être compris)

## Questions simples

- On ne demande qu'une information à la fois
- La question est concise

# Idées reçues

- *Memory is infinite*
- *The more you repeat the better*
- *Learning by doing is the best*

→ Il faut répéter régulièrement et intelligemment

# Algorithme SR

Il existe plusieurs versions de l'algorithme (SM-O à SM-8...+) de plus en plus optimisées. La démarche reste la même : Pour un ensemble de questions, initialiser l'intervalle de répétition de chaque question au minimum Puis pour chaque question posée,

- Recueillir la difficulté de la réponse selon l'utilisateur (sur une échelle de 1 à n...)
- Mettre à jour l'intervalle de répétition en fonction de la difficulté, du temps de réponse...

# Implémentation du SM2

```
public static void calculateInterval(Card card) {  
    if (card.getEFactor() < 3) {  
        card.setCount(1);  
    }  
    int count = card.getCount();  
    int interval = 1;  
    if (count == 2) {  
        interval = 6;  
    } else if (count > 2) {  
        interval = Math.round(card.getInterval()  
* card.getEFactor());  
    }  
    card.setInterval(interval);  
}
```