http://www-adele.imag.fr/users/Didier.Donsez/cours

Ramassage Miette
Garbage Collector

Didier DONSEZ
Université Joseph Fourier
PolyTech'Grenoble LIG/ADELE
Didier.Donsez@imag.fr, Didier.Donsez@ieee.org

Motivations

- Allocation dans le tas
  - en C, malloc()
  - en C++, new
- Problème de la désallocation
  - free, delete
  - quand supprimer les objets qui ne sont plus accessibles depuis une variable pointeur locale ou statique ?

Ramassage de Miettes

- Le développeur alloue de nouveaux objets
  mais ne gère pas la dessalocation
- Le GC recupère la place des objets inaccessibles

- Implémenté dans les langages
  - Smalltalk, Objective-C, CAML, Lisp, et Java

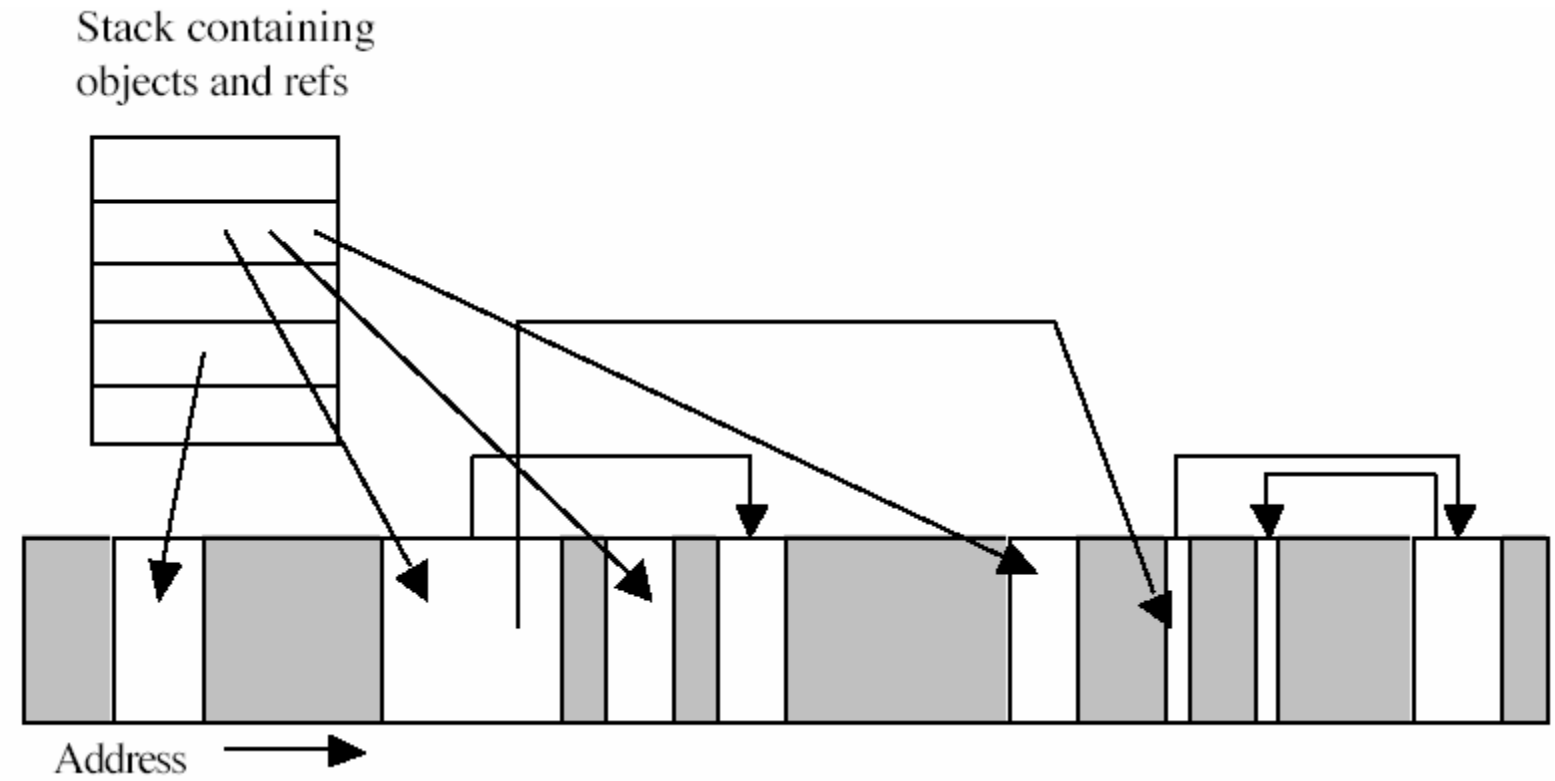
- Bibliothèque pour langages
  - C, C++

Terminologies de Base

- Mutateur
  - Le programme qui alloue des objets depuis le tas et les référence depuis les variables dans la pile et les variables globales (static).
- Glaneur
  - Remet les objets glanés dans le tas (heap)
- Synonymes
  - Ramasse miette, Glaneur de Cellule (GC)
  - Garbage Collector

Techniques de base de GC

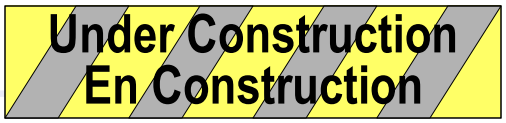
- Compteur de Référence
- Marquage et Nettoyer (Mark-and-Sweep)
- Compactage
- Copie
- Générationnel
- Incrémental



Compteurs de Référence

- Compteur de référence par objet
  - Chaque référence incrémente le compteur de référence
  - Chaque retrait de ref décremente le compteur
    - Affectation de la référence
  - Quand le compteur passe à zéro, l'objet est glanné
- Inconvénients : Cycle de références
  - Coût de l'affectation
  - Détection de cycle mort (dead cyclic data structure)
- Optimisation
  - Weizenbaum (1963)

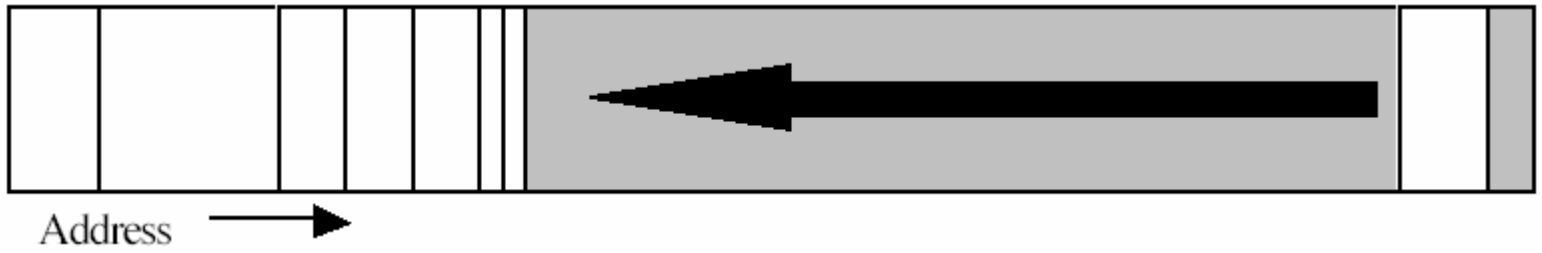
Mark and Sweep



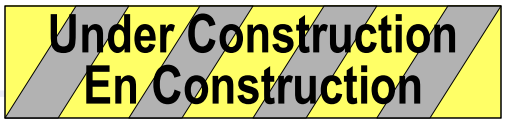
- Principe
  - Le glaneur parcourt tout le graphe d'objets depuis les références présentes dans la pile et les variables globales (static) et marque chaque objet.
  - La deuxième phase glane les objets non marqués
  - Défragmentation possible par compaction du tas
- Inconvénients
  - Blocage du mutateur pendant le glanage

Compactage

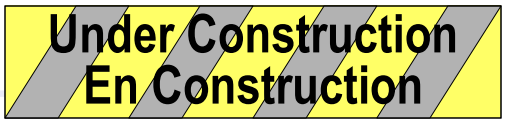
Under Construction
En Construction



Copie



Générationnel



- Principe
  - objects have differing lifetime characteristics
    - some live very short lives, some live very long lives

Incrémental

Under Construction
En Construction

- Motivation
  - Limite la durée du blocage du mutateur
- Principe

Techniques Avancées de GC

- Temps réel / critique
  - Un mutateur TR ne peut pas bloquer/rétarder au délà d'un délai borné
- Distribution
  - multi-sites (RMI DGC)
    - *Garbage Collecting the Internet: A Survey of Distributed Garbage Collection, ACM Computing Surveys, Vol. 30, No. 3, September 1998*
- Transactionnel
  - un mutateur peut avorter et défaire ses modifications
  - plusieurs mutateurs concurrents
- Hiérarchie de Mémoires
  - Primaire, Secondaire,Ternaire
    - *Garbage Collection for a Client-Server Persistent Object Store, ACM Transactions on Computer Systems, Vol. 17, No. 3, August 1999.*

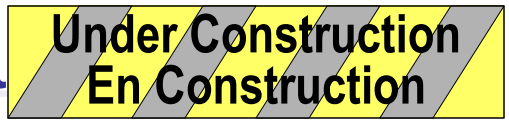
GC et Java

- Par défaut : Mark and Sweep
- –Xincgc
  - Active le GC incrémental
    - évite les pauses
    - mais diminue les performances du GC de 10%
- -Xnoclassgc
  - Supprime le GC des classes
- -verbose:gc -Xloggc:file
  - Rapporte les événements du GC
- -Xmsn , -Xmxn
  - Tailles minimum et maximum du pool d'allocation mémoire
- A lire
  - http://72.5.124.55/j2se/reference/whitepapers/memorymanagement_whitepaper.pdf
- A visiter
  - Le source de la VM Waba

GC et Java 1.5

- Package java.lang.ref
  - Provides reference-object classes, which support a limited degree of interaction with the garbage collector.
  - A program may use a reference object to maintain a reference to some other object in such a way that the latter object may still be reclaimed by the collector.
  - A program may also arrange to be notified some time after the collector has determined that the reachability of a given object has changed.
- Classes
  - Reference<T> Abstract base class.
  - PhantomReference<T>
    - are enqueued after the collector determines that their referents may otherwise be reclaimed.
  - SoftReference<T>
    - are cleared at the discretion of the garbage collector in response to memory demand.
  - WeakReference<T>
    - do not prevent their referents from being made finalizable, finalized, and then reclaimed.
  - ReferenceQueue<T>
    - registered reference objects are appended by the garbage collector after the appropriate reachability changes are detected.

GC et RTJ (Real-Time Ja



- A FAIRE

GC et .NET/CLR

- .NET et SSCLI
  - Générationnel Adaptatif (2 générations)
  - Voir le livre *http://msdn.microsoft.com/net/sscli chapitre 7*
- Compact .NET
  - Mark-and-Sweep simple
- Remarque
  - Blocs d'instruction unsafe { }

Bibliographie

- Wilson Etat de l ’art en centralisé
  - WILSON, P. R., JOHNSTONE, M. S., NEELY, M., AND BOLES, D. 1995. Dynamic storage allocation: A survey and critical review. In *Proceedings of the International Workshop on Memory Management* (Kinross, UK), LNCS 986, Springer-Verlag, New York, 1–116.
- Shapiro
  - Larchant et Perdis
- Amsaleg
  - Transactionnel
- Thevenin
- Richard Jones, Rafael Lins "Garbage Collection" page
  - http://www.cs.ukc.ac.uk/people/staff/rej/gc.html
  - Bibliographie et résumé sur le GC.
  - Richard Jones and Rafael Lins, Garbage Collection, Algorithms for Automatic Dynamic Memory Management