

Java Collection Framework

Em Java uma coleção é objeto que agrupa múltiplos elementos em uma unidade.

As coleções foram criadas como uma evolução dos arrays.

Um array é um grupo de variáveis (elementos ou componentes) que contém valores que são todos do mesmo tipo.

Para referenciar um elemento, é necessário especificar sua referência e o número da posição. O número de posição do elemento é chamado de índice.

Java Collection Framework

Collection é estrutura de dados que guarda elementos em grupo, provendo uma serie de métodos como: adicionar, remove, ordenar etc.

Arrays são muito estáticos, entretanto, de certo modo também pode ser chamados de coleção

JSF foi introduzido na versão 1.2 (Standard Template Library STL C++)

Java Collection Framework

nome do array (c)	array[0]	21
	array[1]	54
	array[2]	258
	array[3]	25
	array[4]	64
	array[5]	1
	array[6]	547
	array[7]	357
	array[8]	248
	array[9]	3
	array[10]	6
Índice	array[11]	254

Um Array em Java - Fonte: autor

Java Collection Framework

Coleções são usadas para armazenar, recuperar, manipular e comunicação de dados

Ex de uso: uma fila de banco, uma lista de compras, uma pilha de livros, um conjunto de elementos ...

Nas versões inferiores a 1.2, as coleções incluíam, Vector, Hashtable e Array, mas não possuíam um framework.

O framework de coleções (framework collection) é uma arquitetura unificada de representação e manipulação de coleção.

Interfaces - Interfaces permitem que as coleções sejam manipuladas independentemente de detalhes de representação.

Implementações - estruturas de dados reusáveis.

Algoritmos - métodos que realizam operações úteis, como pesquisas e ordenações nos objetos que implementam a coleção. Os algoritmos possuem comportamento polimórfico, isto é, o mesmo método pode ser implementado de diferentes formas usando as interfaces. Em suma, algoritmos possuem funcionalidade reutilizável.

Java Collection Framework

Alguns algoritmos do pacote `java.util.Collections` que fazem parte do Collection Framework.

- `sort` -- merge sort
- `shuffle` — randomly permutes the elements in a `List`.
- `binarySearch`
- `reverse/fill/copy/swap/addAll` (data manipulation)
- `Composition` — reverses the order of the elements in a `List`.
- `min`, `max`

Benefícios

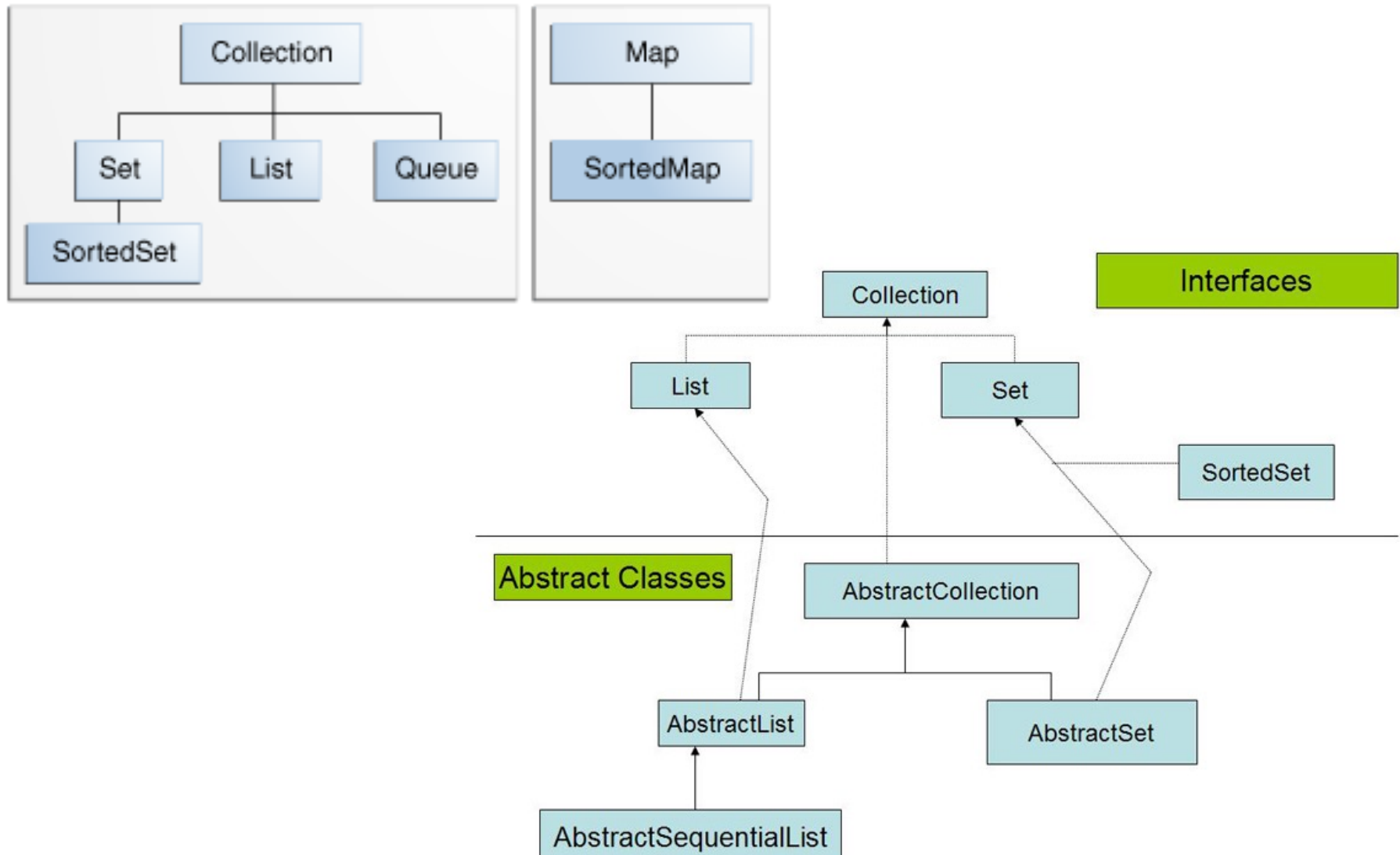
- Reduzir o esforço de programação
- Aumentar a velocidade e qualidade da programação
- Permitir interoperabilidade
- Reduzir o esforço para usar uma nova API.
- Reduzir o esforço para desenhar uma nova API.
- Promover a reusabilidade de código

A maioria das Coleções Java está no pacote `java.util` o pacote `java.util.concurrent` possui coleções que trabalham com concorrência

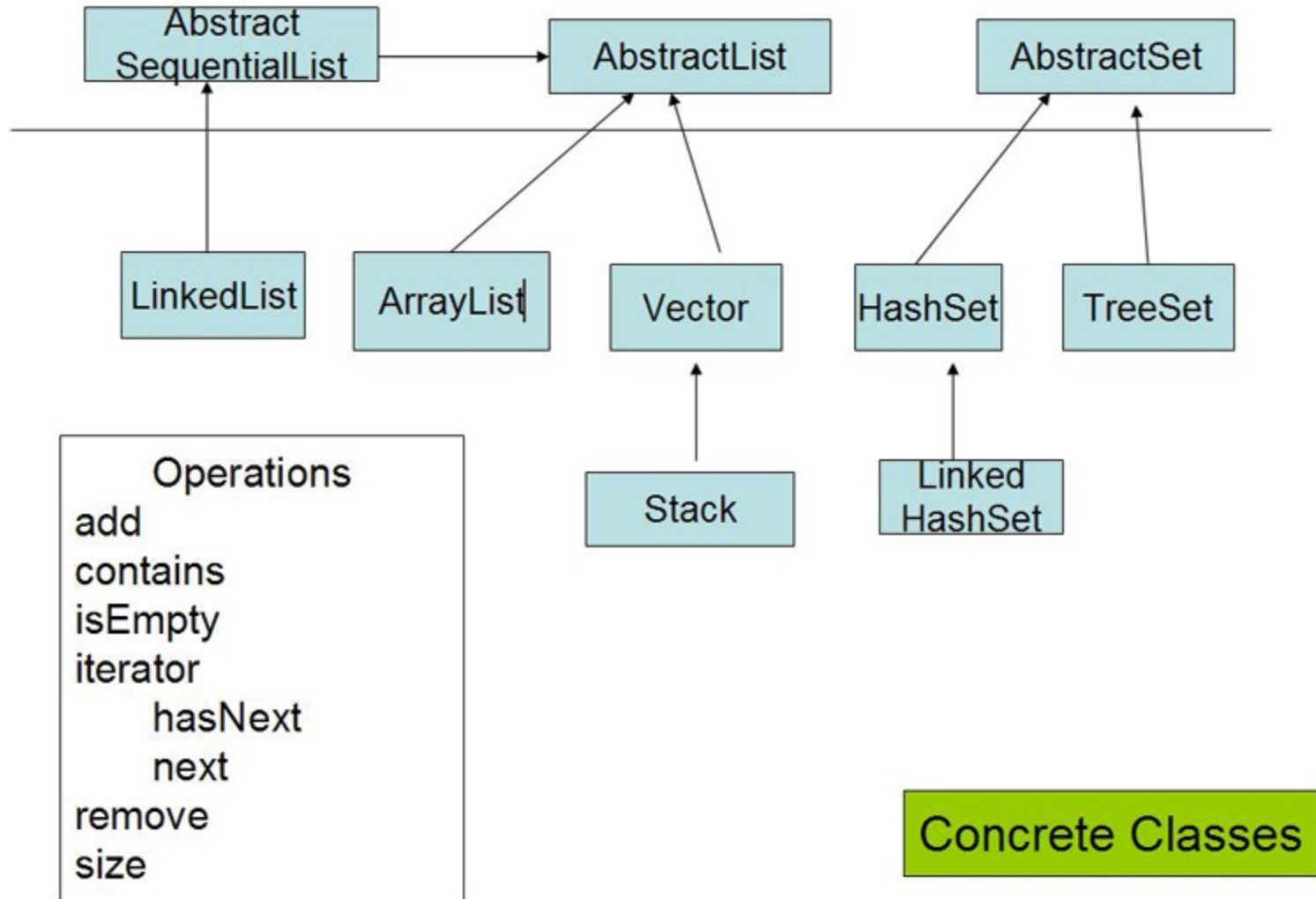
Java Collection Framework

Interfaces

Core collection interfaces are the foundation of the Java Collections Framework



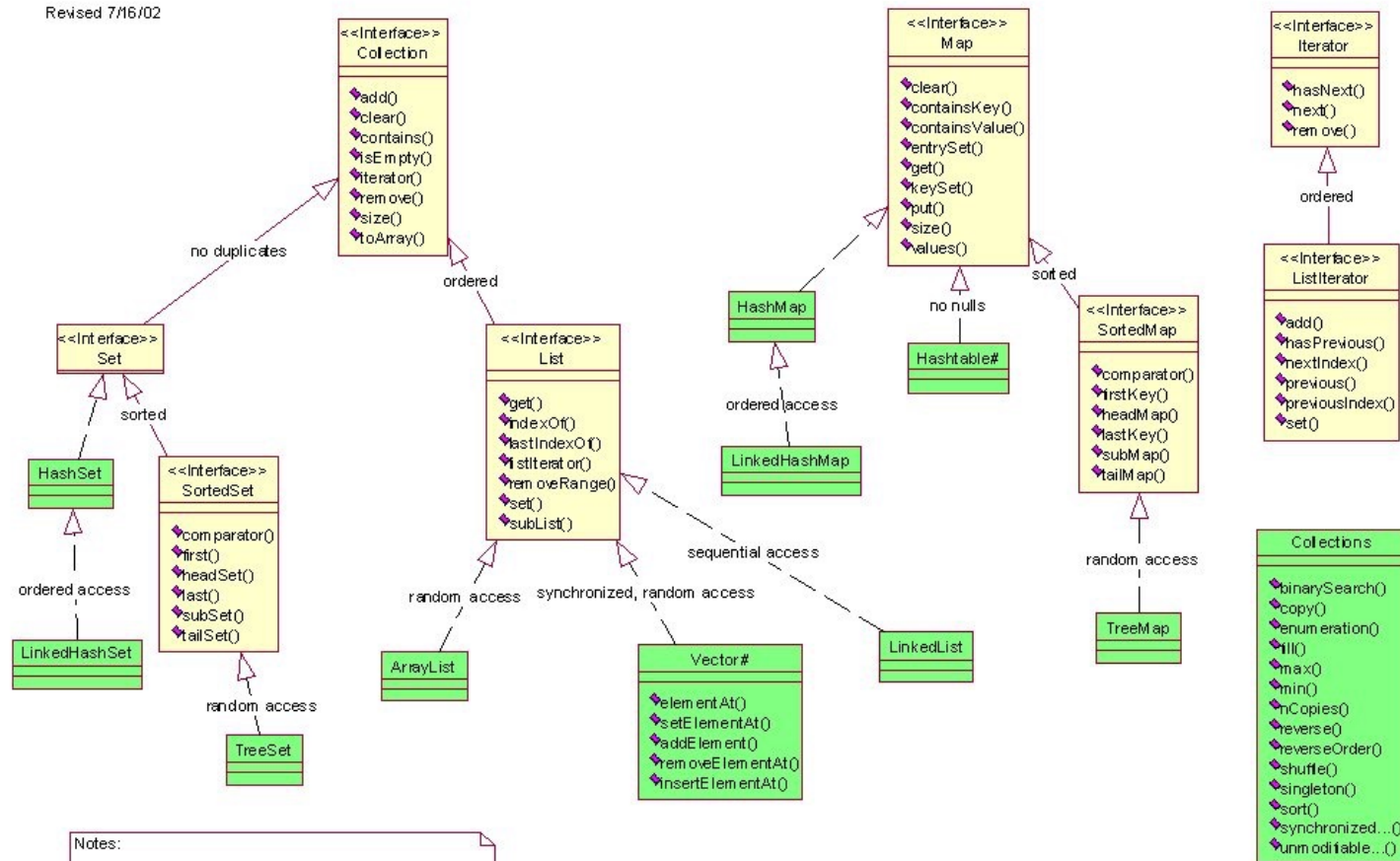
Java Collection Framework



Java Collection Framework

Java Collections Framework

Revised 7/16/02



Notes:

denotes class is a legacy class.

Differences between Java and Java2:

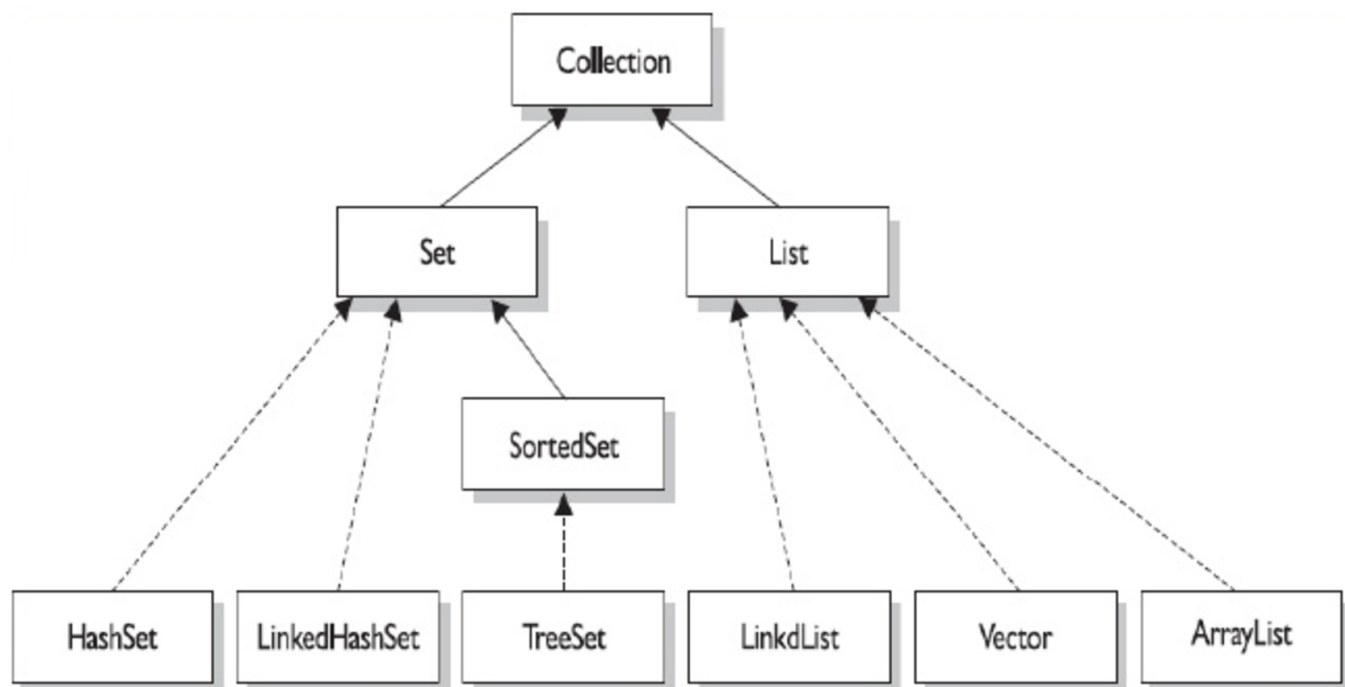
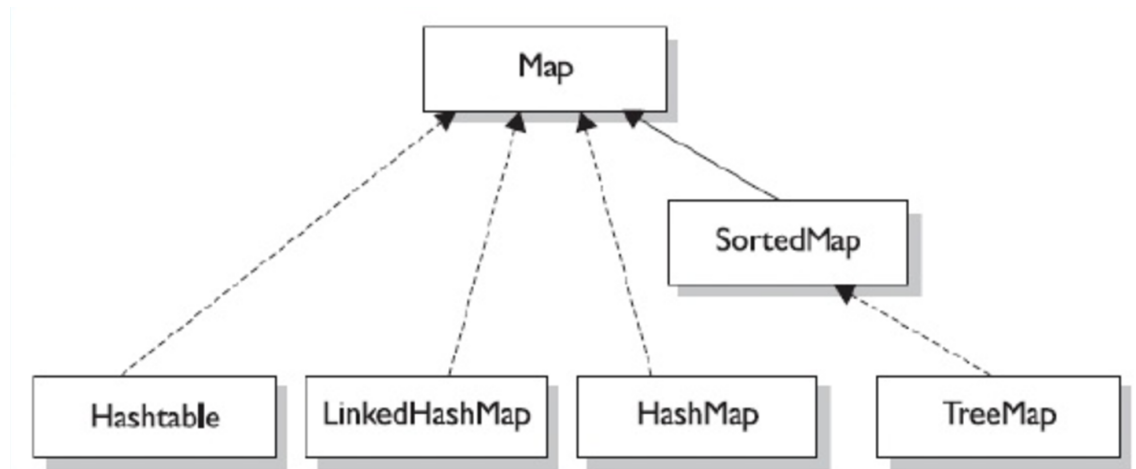
* Non-legacy collections are not synchronized.

* Iterator modifications change the backing collection.

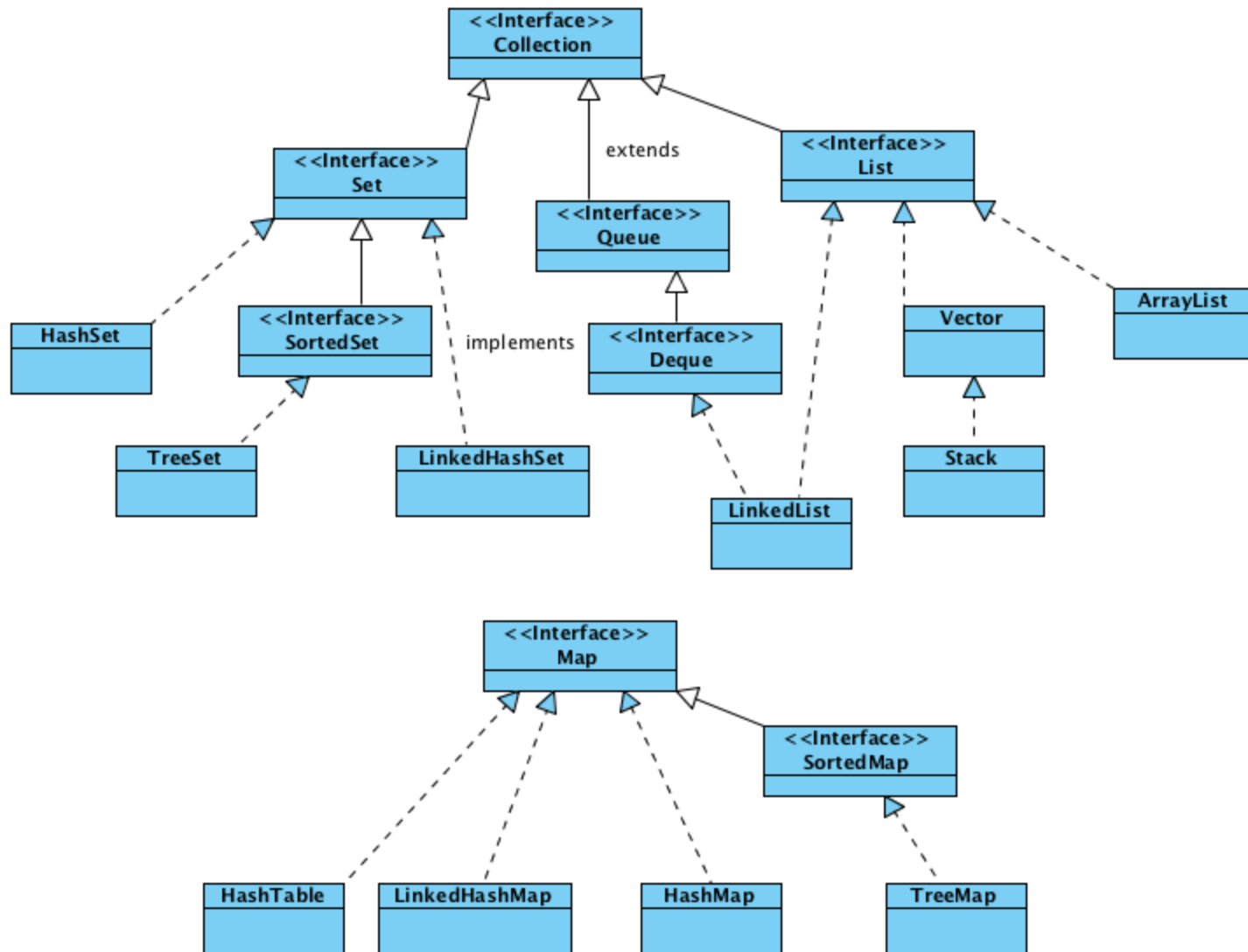
Not all classes, interfaces, or methods are shown.

(C) 2002, McWrest Corp., www.mcwrestcorp.com
May be freely copied or adapted.

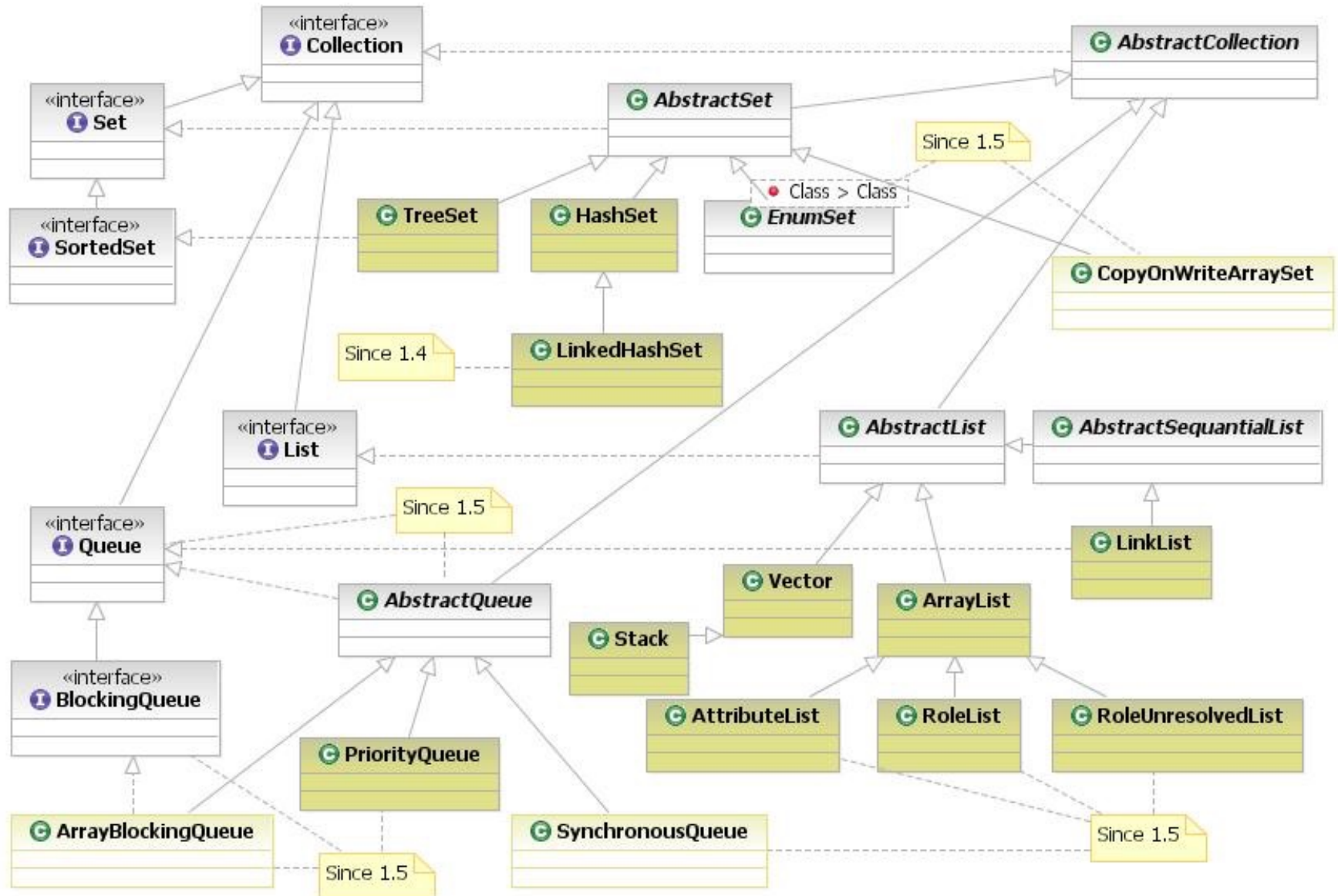
Java Collection Framework



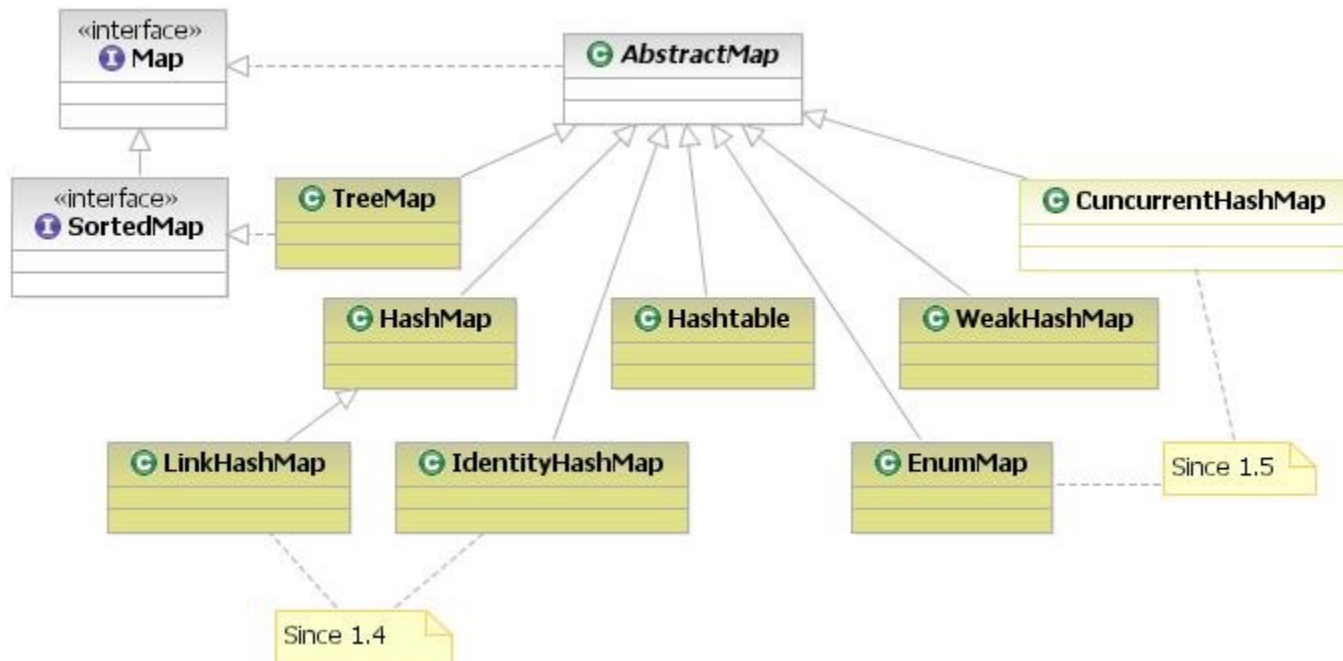
Java Collection Framework



Java Collection Framework

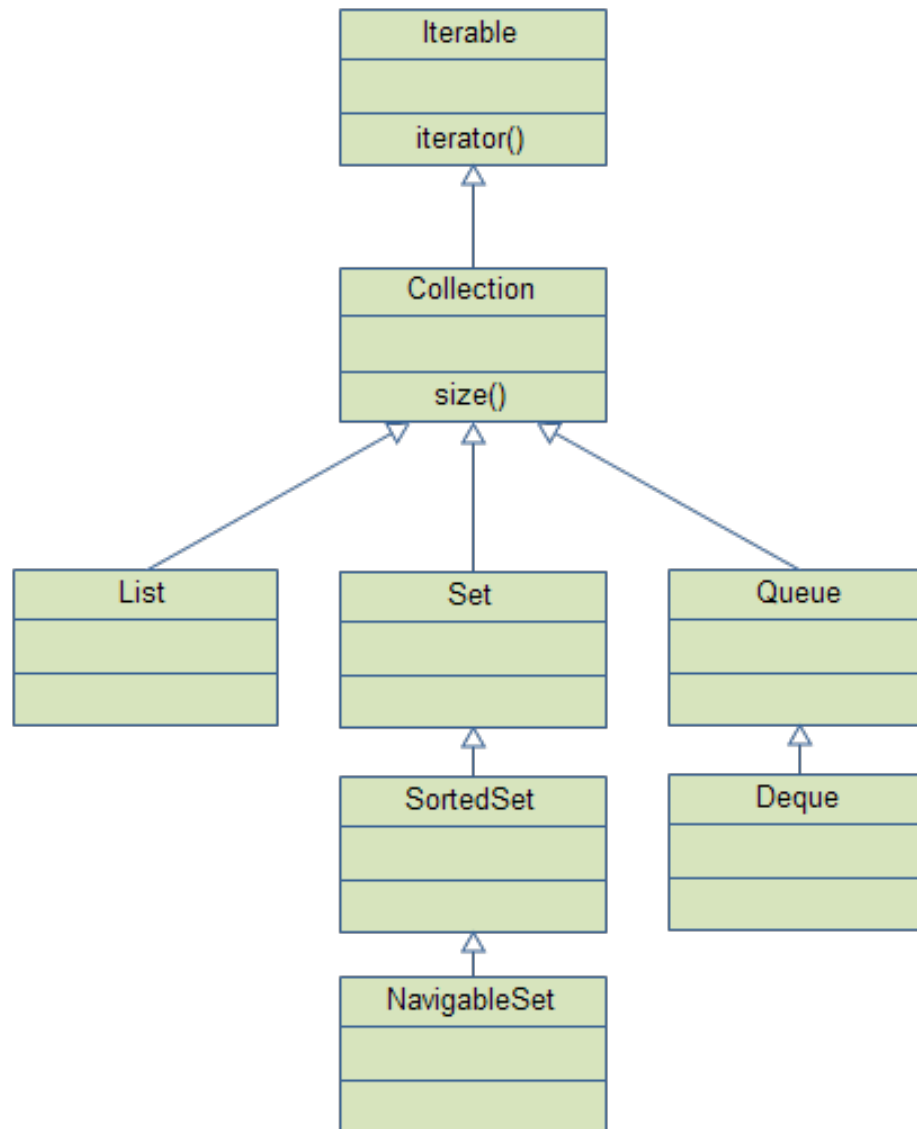


Java Collection Framework



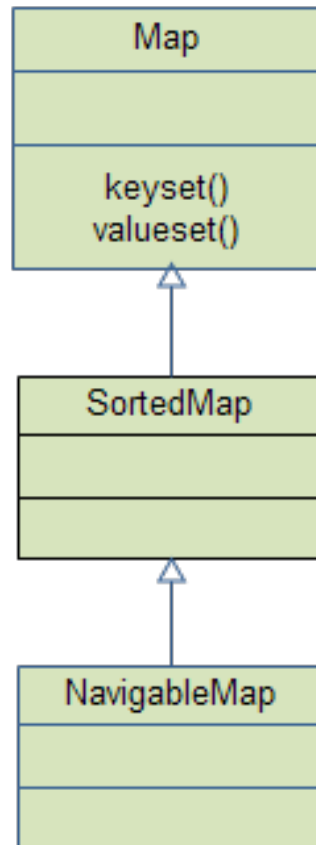
Java Collection Framework

Resumo da hierarquia de Interfaces

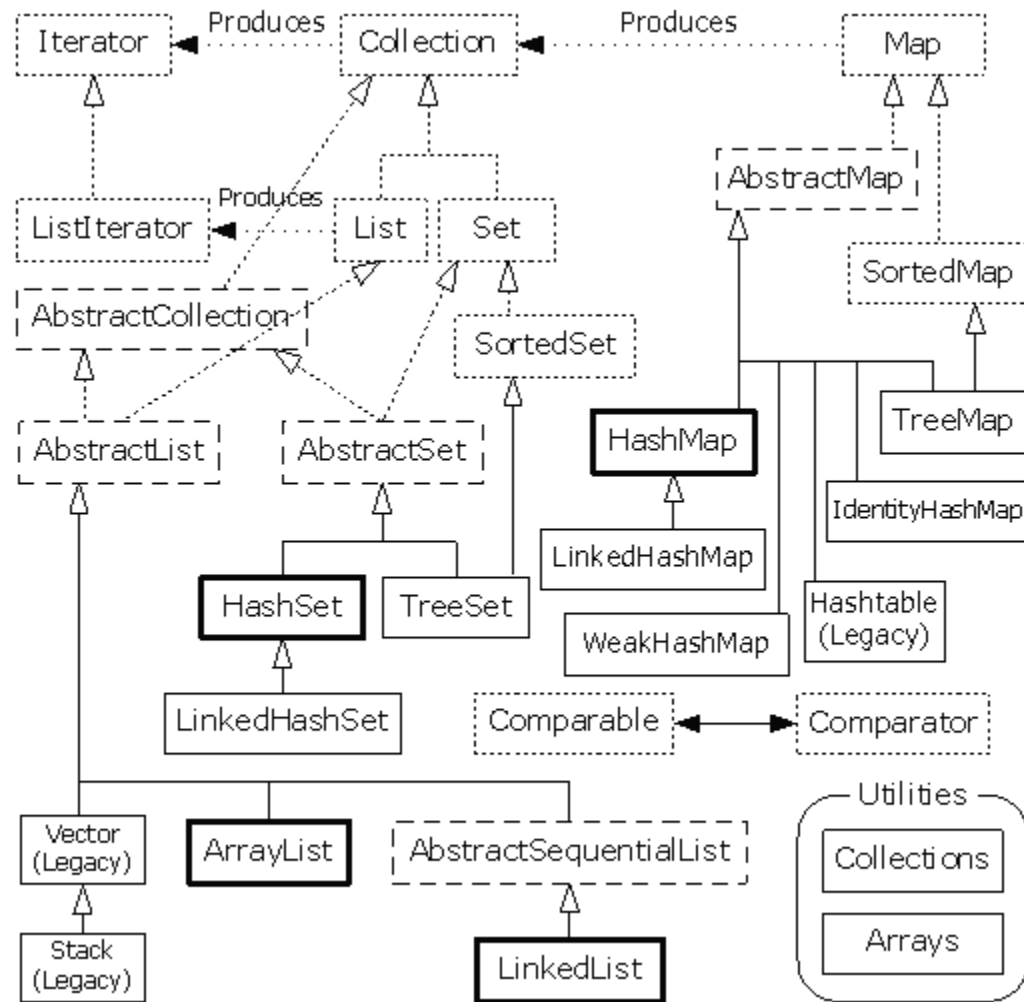


Java Collection Framework

Map interface hierarchy:



Java Collection Framework



Java Collection Framework

Set não pode conter elementos duplicados
Elementos inseridos (put) numa Set devem sobrescrever o método equals() para estabelecer a unicidade.

comparar performance entre Vector e ArrayList

Vector é thread safe arrayList nah

Queue -> LinkedList (FIFO) principal implementação.
So se pode acessar o primeiro ou ultimo elemento
peekFirst() 1st element
PeekLast()
Poll ...

Java Collection Framework

Map

Usada para armazenar um par chave/valor.

Exemplo uma lista telefônica, nome=chave, valor=número
Não estende Collection mas provê métodos semelhantes.

put, get, remove,
containsKey(), containsValue(),

```
Set<String> keys = agenda.keySet();  
for (String s: keys){  
    System.out.println(s: “: ” agenda.get(s));  
}
```

```
Set<Map,Entry<String,String>> values = agenda.entrySet();  
for (Map.Entry<String, String> e: values){  
    System.out.println(e.getKey() + “:” + e.getValues());  
}
```

Java Collection Framework

Serialização

Leitura e Gravação de arquivos

```
FileOutputStream fos = new FileOutputStream(f, true);  
ObjectOutputStream objGravar = new ObjectOutputStream(fos);  
objGravar.writeObject(maria);
```

```
FileInputStream fis = new FileInputStream(f);  
ObjectInputStream objLeitura = new ObjectInputStream(fis);  
System.out.println(objLeitura.readObject());
```