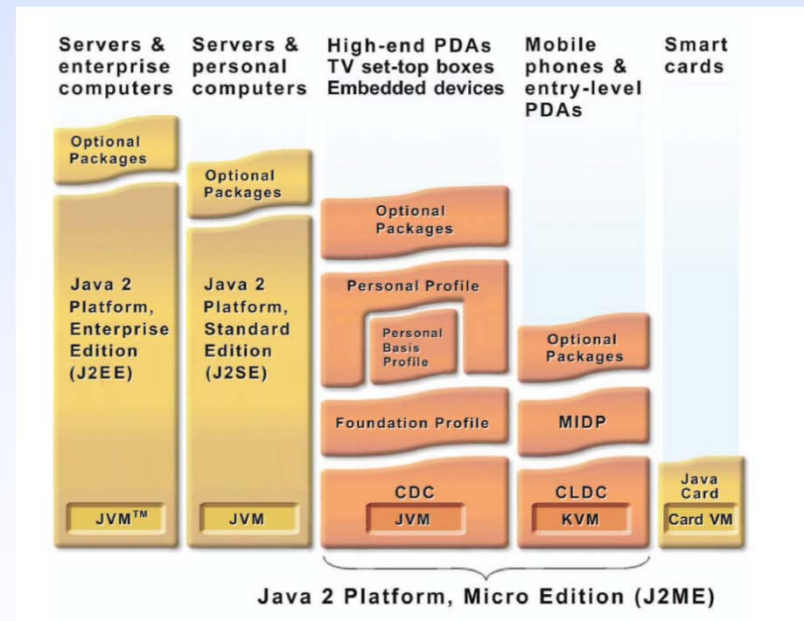


Skala der Java-Technologien

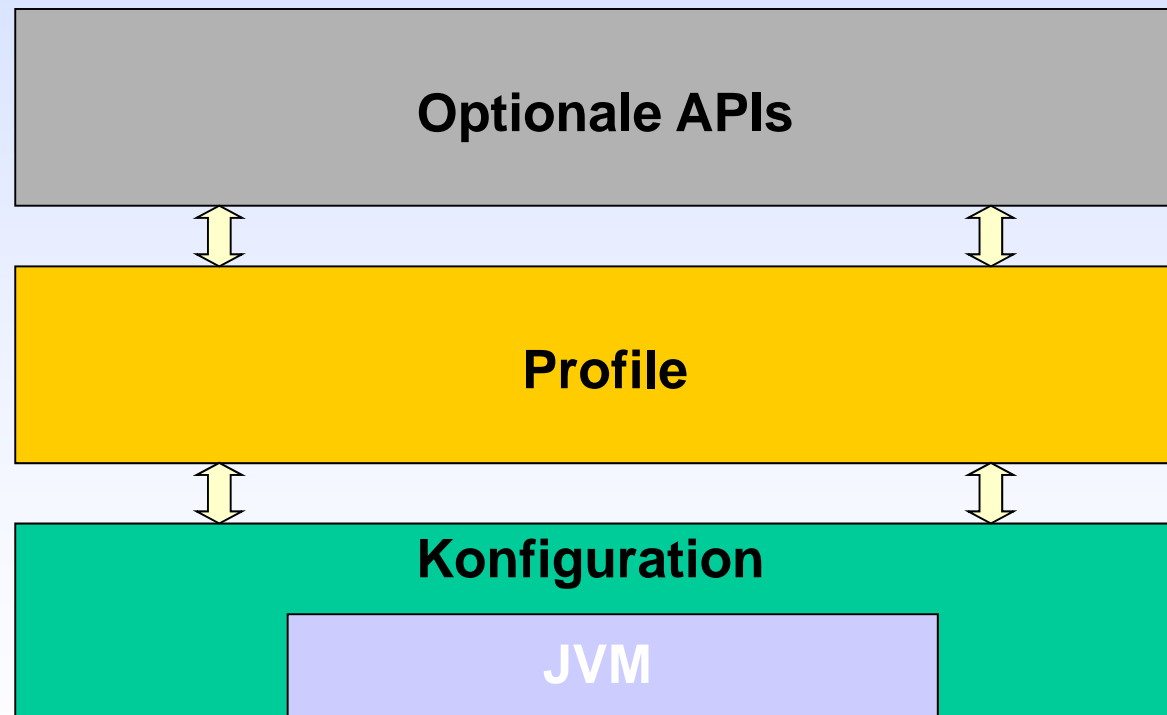
- Java 2 Enterprise Edition - J2EE für Server
- Java 2 Standard Edition - J2SE für PCs, Laptops, NCs und Workstations
- Java 2 Micro Edition - J2ME für Pager, Handys, PDAs, Set-Top-Boxen, etc.
- JavaCard für Chip-Karten



J2ME

Modular aufgebaut

Java Community Process(JCP) und Java Specification Requests(JSRs).



Konfigurationen

- Eine Konfiguration bestimmt die Basisfunktionalität in Form von Klassenbibliotheken und der dazu gehörigen VM.
- Die Zuordnung von Geräten zu einzelnen Konfigurationsgruppen erfolgt anhand ihrer Leistungsfähigkeit (Prozessor und Speichergrösse).
- Im Moment existieren zwei verschiedene Konfigurationen:
 - Connected Device Configuration (CDC).
 - Connected Limited Device Configuration (CLDC).

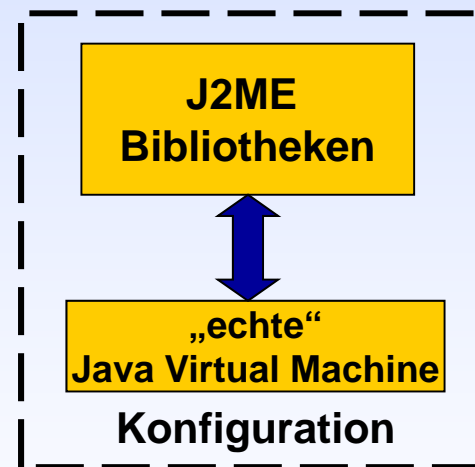
Konfiguration	(Massen-) Speicher	Netzwerk- verbindung	Band- breite	TCP/ IP	Energie- versorgung
CDC	2-4 MB	persistent	hoch	ja	fest
CLDC	ab 128 kB	flüchtig	gering	nein	Batterie

Connected Limited Device Configuration (CLDC)

- CLDC wendet sich an die kleinsten mobile Geräte: Handys und PDA
- Im Moment gibt es zwei Versionen:
 - CLDC 1.0 (JSR 30)
 - ab 32 kb flüchtiger Speicher
 - ab 128 kb nicht flüchtiger Speicher
 - 16 oder 32 Bit Prozessor
 - CLDC 1.1 (JSR 139)
 - ab 32 kb flüchtiger Speicher
 - ab 160 kb nicht flüchtiger Speicher
 - 16 oder 32 Bit Prozessor
- Beide Versionen arbeiten mit einer „angepassten“ Virtual Machine
KVM, „K“ steht für Kilo, da die Grösse 10 kB beträgt.
- Neben KVM, bietet CLDC auch eine Reihe von Basisklassen an,
wie z.B. Ein/Ausgabe Operationen,
Netzwerkprogrammierung u.s.w.

Connected Device Configuration (CDC)

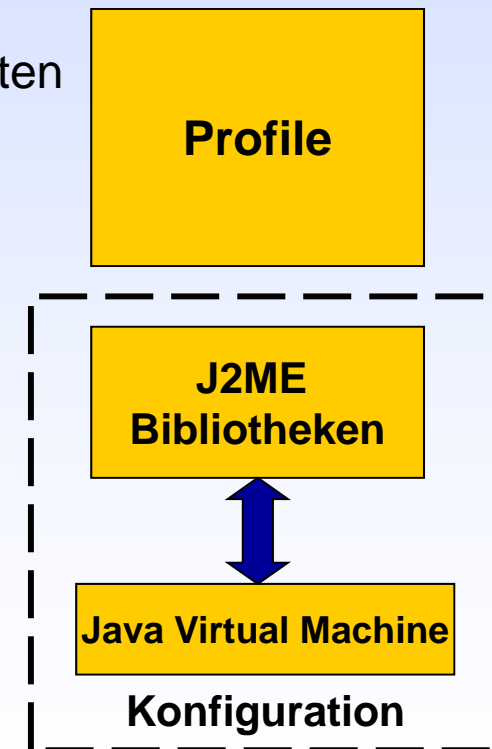
- Für wesentlich leistungsfähigere Geräte gedacht, wie High-End PDAs, TV Set-Top Boxen u.s.w.
- Unterschied zu CLDC ist, dass hier eine „echte“ JVM genutzt wird.



- Die zwei Versionen werden in JSR-36 (CDC 1.0) und JSR-218 (CDC 1.1) spezifiziert

Profile

- Profilen erweitern die Konfigurationen durch zusätzliche Klassen, welche auf den Konfigurationen aufsetzen und diese nutzen.
- Somit wird die Möglichkeit erreicht, die erweiterten Fähigkeiten eines Gerätes zu nutzen, wie z.B.:
 - Grafische Oberflächen
 - Audiofunktionen
 - oder erweiterte Netzwerkfunktionen
- Jedes Profil gehört immer zu einer bestimmten Konfiguration.
- Der wichtigste CLDC Vertreter ist das MIDP.



Optionale APIs

- durch den Einsatz von optionalen Paketen kann die Laufzeitumgebung um einzelnen Funktionen erweitert werden.
- es existieren eine grosse Anzahl optionaler Pakete wie z.B.
 - Mobile 3D Graphics (JSR-184)
 - J2ME Web Services APIs (JSR-172)
 - Bluetooth API (JSR-82)
 - ...

Mobile Information Device Profile (MIDP)

- Profil mit der grössten Bedeutung, welches zum CLDC gehört.
- Jedes javafähige Mobiltelefon benutzt das MIDP.
- Folgende Versionen:
 - MIDP 1.0.4.(JSR-37)
 - MIDP 2.0 (JSR-118)
 - MIDP 3.0 ist bereits als (JSR-271) im Bearbeitung

Hardwareanforderungen

- Bildschirm:
 - Bildschirmgrösse 96 x 54
 - Farbtiefe 1 Bit
 - Pixel Form (*aspect ratio*) ca. 1:1
- Eingabe:
 - „*one-hand*“ Tastatur
 - „*two-hand*“ Tastatur
 - Touchscreen
- Speicher:
 - 264 kB ROM
 - 128 kB RAM

Softwareanforderungen

- Ein Minimalkernel zur Verwaltung der Hardware (z.B. Unterbrechungen, Ausnahmen,...)
- Ein Mechanismus zum Lesen und Schreiben.
- Lesen und Schreiben auf die Netzwerkverbindung.
- Timer.
- Abfrage der Benutzereingabe.
- Mechanismus zur Verwaltung des Anwendungslebenszyklus des Geräts (Installation, Ausführung, Deinstallation)

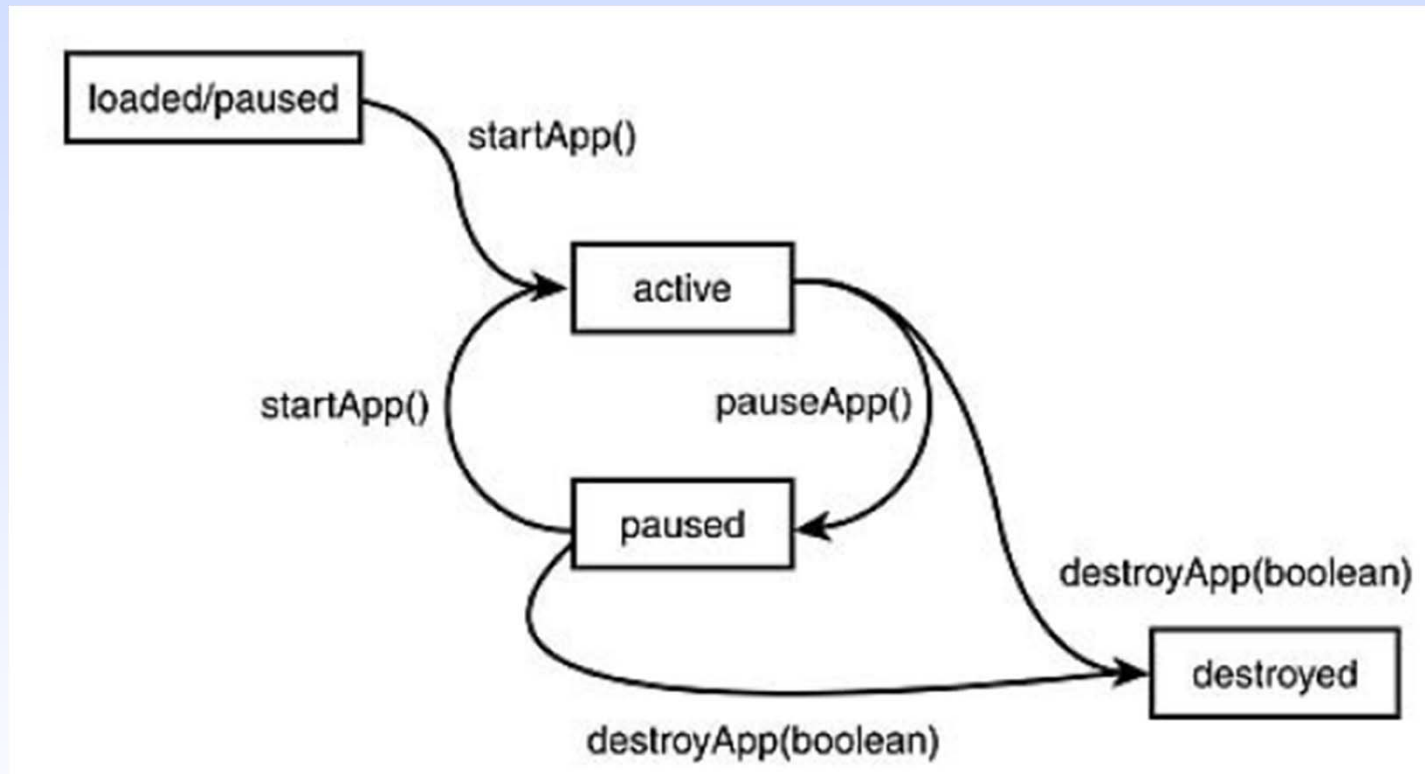
Anwendungen auf Basis des MIDP - MIDlets

- Die Anwendungen, die auf MIDP-unterstützenden Geräten laufen heißen MIDlets.
- Die Ausführung der MIDlets wird durch den Java Application Manager (JAM) kontrolliert.
- Die Anwendung muss, wie bei Applets, die folgenden Methoden implementieren, damit der JAM die kontrollieren kann.
 - `startApp() ;`
 - `pauseApp() ;`
 - `destroyApp() ;`
- Die MIDlets sollen auf verschiedene Geräte übertragbar sein.

Grundstruktur

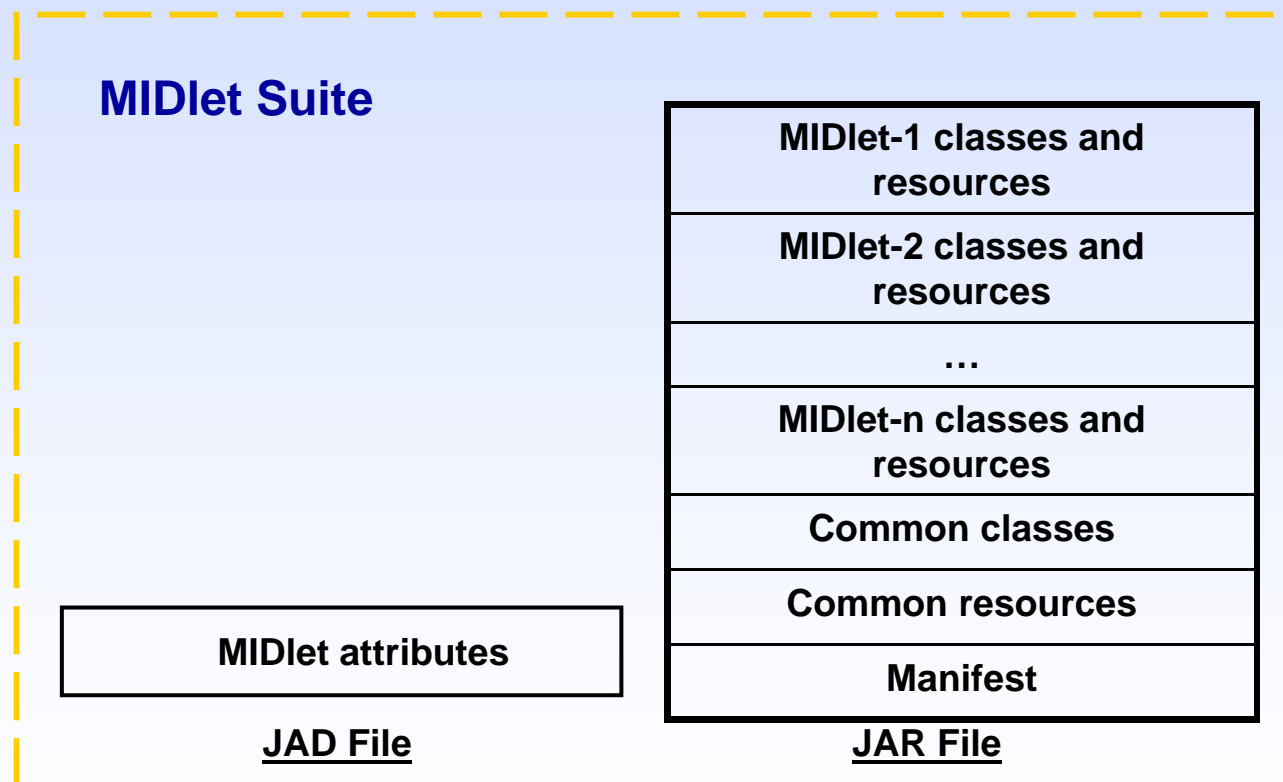
```
public class MIDletGrundstruktur extends MIDlet {  
    public MIDletGrundstruktur() {  
        ...  
    }  
    protected void startApp() {  
        ...  
    }  
    protected void pauseApp() {  
        ...  
    }  
    protected void destroyApp(boolean unconditional)  
    {  
        ...  
    }  
}
```

Zustände



MIDlet Suites

- Die zu einer MIDlet-Anwendung gehörenden Klassen und Ressourcen werden in einer MIDlet-Suite zusammengestellt.



MIDlet Suites

- Das Java Archive (JAR).
 - Die Anwendungsklasse.
 - Klassen, die von mehrere Anwendungen genutzt werden.
 - Ressourcendateien, z.B. Bilder, Hilfstexte u.s.w.
 - Eine **Manifest**-Datei.

Attribut Name		Verwendung
MIDlet-Name	↔	Name der MIDlet Suite
MIDlet-Version	↔	Versionsnummer
MIDlet-Vendor	↔	Herstellername

MIDlet Suites

□ Das Java Application Descriptor (JAD).

■ Pflichtattribute der JAD Datei.

Attribut Name		Verwendung
MIDlet-Name	↔	Name der MIDlet Suite
MIDlet-Version	↔	Versionsnummer
MIDlet-Vendor	↔	Herstellername
MIDlet-Jar-URL	↔	Adresse der Archivedatei
MIDlet-Jar-Size	↔	Größe des JavaArchivs

■ Sonstige Pflichtattribute der JAD Datei.

Attribut Name		Verwendung
MIDlet-<n> for each MIDlet	↔	Eintrag pro Applikation
Microedition-Profile	↔	Name des Profils
Microedition-Configuration	↔	Name der Konfiguration

MIDP Bibliotheken

- Die CLDC Konfiguration wird durch Packages erweitert:
 - **javax.microedition.midlet**
 - Für den Lebenszyklus
 - **javax.microedition.lcdui**
 - Für Benutzerinteraktionen
 - **javax.microedition.media**
 - Für Multimedia
 - **javax.microedition.io**
 - Für Netzwerkverbindungen

Bildschirmmasken - form

Anlegen der form-Instanzen

```
Form form;  
form = new Form(„Titel“);
```

Maskenelemente hinzufügen

```
form.append( <Maskenelemente> );
```

Anzeigen

```
display.setCurrent(form);
```

Komponenten

StringItem - Namensgebender String

ImageItem - Verweis auf Grafik

CustomItem - Abstrakte Superklasse für eigene Klassen

Spacer - Trennelement

ChoiceGroup - Auswahlgruppe

TextField - Editierbares Texteingabefeld

DateField - Editierbares Datumsfeld

Gauge - Fortschrittsanzeige

TextField - Befehle

- `delete(int offset, int length)`
- `getCaretPosition()`
- `getChars(char[] data)`
- `getConstraints()`
- `getMaxSize()`
- `getString()`
- `insert(char[] data, int offset, int length, int position)`
- `insert(String src, int position)`
- `setChars(char[] data, int offset, int length)`
- `setConstraints(int constraints)`
- `setMaxSize(int maxSize)`
- `setString(String text)`
- `size()`

TextField - Eingabemöglichkeiten

ANY- Beliebig

NUMERIC - ganze Zahlen

DECIMAL – Ziffern, „.“ und „-“

EMAILADDR

PHONENUMBER

URL

CommandListener

Softbuttons zur Steuerung der Abläufe:

Anlegen der Command-Instanzen

```
Command command1;  
Command command2;  
command1= new Command(„Titel1“, <Command.TYP>,  
<position>);  
command2 = new Command(„Titel2“, <Command.TYP>,  
<position>);
```

Der Ausgabemaske hinzufügen

```
form.addCommand(command1);  
form.addCommand(command2);  
form.setCommandListener(this);
```

Command-Typen

SCREEN - Bildschirmbezogene Kommandos, z. B.
Laden, Speichern, Öffnen

BACK - Rückkehr zum vorherigen Bildschirm

CANCEL - Dialogabbruch

EXIT - Programmabbruch

HELP - Hilfe anfordern

ITEM - Elemente des Bildschirms, z. B. Auswahlliste

OK - Bestätigung

STOP - Abbruch der laufenden Aktion

J2ME Wireless Toolkit

- Tool, mit dem MIDlets erstellt, getestet und zu einer MIDlet-Suite gepackt werden können.
- In Verbindung mit dem eclipse-PlugIn „EclipseME“ können J2ME-Projekte entwickelt werden

Aufgabe

J2ME-Projekt

Literatur-Hinweise (1)

- *J. Sanchez, JAVA Programming for Engineers, CRC-Press*
- *Ken Arnold, James Gosling und David Holmes: Die Programmiersprache Java, Verlag Addison-Wesley Deutschland*
- *Mary Campione und Kathy Walrath: The Java Tutorial – 3 Bände, Verlag Addison-Wesley*
- *BR-Alpha, Java-Kompaktkurs, www.bw.fh-deggendorf.de/kurse*
- *RRZN (Hg.): Java 2 - Begleitmaterial zu Vorlesungen/Kursen*

Literatur-Hinweise (2)

- *Judy Bishop*: Java lernen, Verlag Addison-Wesley Deutschland
- *Fritz Jobst*: Programmieren in Java, Hanser-Verlag
- *Martin Schader und Lars Schmidt-Thieme*: Java – Eine Einführung, Springer-Verlag
- D. Ratz, Grundkurs Programmieren in Java, Hanser-Verlag
- R. Oechsle, Parallele Programmierung mit Java Threads, Fachbuchverlag Leipzig
- Breymann, Mosemann, JavaME, Hanser Verlag

Ressourcen im Web

Wichtige „Einstiegs-URLs“

- <http://www.sun.com>
- <http://java.sun.com>
- <http://developer.java.sun.com>
- <http://forte.sun.com/ffj>
- <http://www.gamelan.com>
- <http://www.javaworld.com>