



~~J2ME~~

Introduzione allo sviluppo di applicazioni per dispositivi mobili

Stefano Sanna

gerdavax@tiscali.it

gerda@crs4.it



Java ME

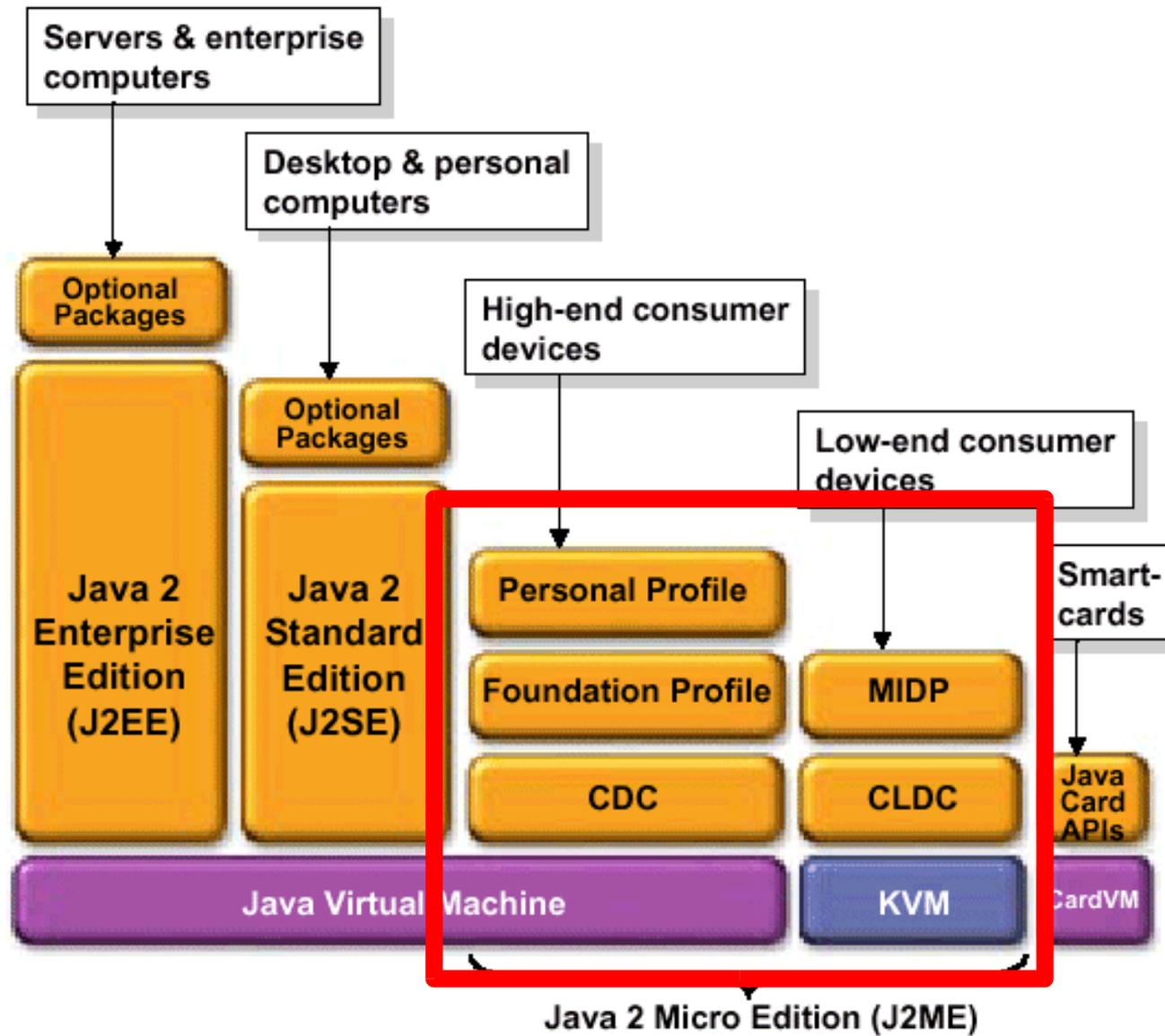
Introduzione allo sviluppo di applicazioni per dispositivi mobili

Stefano Sanna

gerdavax@tiscali.it

gerda@crs4.it

La piattaforma Java...



Fonte:
Sun Microsystems



- Virtual Machine
 - E' il motore Java che fornisce la prima astrazione rispetto all'hardware del dispositivo
- Configurazione
 - Rappresenta il set di classi comune ad una famiglia di dispositivi caratterizzati da potenza hardware simile. Non definisce una GUI!
- Profilo
 - Rappresenta la specializzazione della Configurazione su una tipologia di prodotto. Definisce la GUI (se esiste...)

- **CDC** (palmari, set-top box, navigatori):
 - Foundation Profile
 - Personal Basis Profile
 - **Personal Profile**



- **CLDC** (cellulari, pager, palmari):
 - **Mobile Information Device Profile (MIDP)**
 - Information Module Profile (IMP)





- E' il profilo per eccellenza in ambito telefonico:
 - Definisce un set minimo ma funzionale per la creazione di interfacce grafiche per display mouse- & pen- less
 - Offre connettività wireless e un servizio base di persistenza locale
 - E' sufficientemente leggero e astratto da poter essere portato su una grande varietà di dispositivi e di piattaforme
 - E' continuamente aggiornato e arricchito da librerie opzionali

MIDP 1.0



- Rilasciata a settembre 2001
- Funzionalità:
 - Interfacce grafiche elementari
 - Connessioni HTTP
 - Persistenza dei dati sul dispositivo
- Principali limiti:
 - Nessuna funzione multimediale
 - Nessun supporto a connessioni sicure
 - Mancanza di componenti personalizzabili



Nokia 7650
Uno dei primi cellulari J2ME

MIDP 2.0



- Rilasciata a fine 2002
- Miglioramenti:
 - Possibilità di definire componenti grafici personalizzati
 - Maggior numero di protocolli di rete supportati
 - Funzionalità di base per gestione file audio
 - Gestione connessioni HTTP sicure
 - API avanzate per lo sviluppo di videogiochi e Push Registry



Nokia 6600
Uno dei primi cellulari MIDP 2.0

Il futuro: MIDP 3.0



- Il Java Community Process ha iniziato a lavorare al JSR 271, nella cui agenda figurano:
 - Multitasking e background MIDlets
 - Potenziamento GUI e uso del secondo display
 - Internazionalizzazione più potente
 - Supporto per diversi tipi di reti (Wi-Fi, telefonica...)
 - Standardizzazione delle interfacce per l'installazione delle applicazioni
- Se ne parla nel 2006...! :-)

Differenze rispetto a Java SE



- A parte le specificità dello sviluppo in ambito mobile, queste sono alcune delle differenze significative tra Java SE e Java ME:
 - Managed Application Model
 - Processo di sviluppo e deployment
 - Libreria di classi ridotta (java.lang, java.util...)
 - Generic Connection Framework
 - LCDUI
 - Modularità spinta

Libreria di classi ridotta (cenni)



- Multithreading, I/O e collection semplificati
- Mancano:
 - Serializzazione
 - Java Native Interface
 - Reflection
 - Dynamic class loading remoto (solo dal Jar)
 - Aritmetica floating point (CLDC 1.0)
 - ... mancano StringTokenizer, BufferedReader...

La differenza principale...

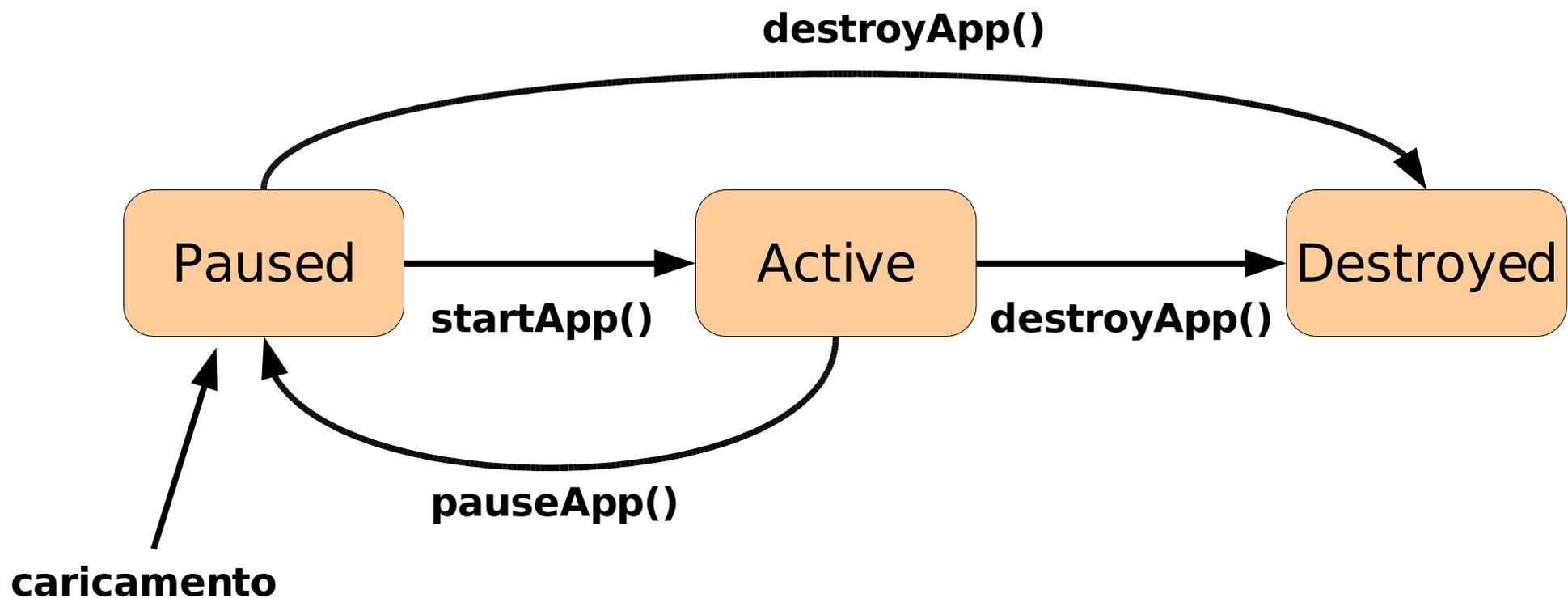


- Le applicazioni girano su **dispositivi** molto diversi tra loro (non esiste una piattaforma di riferimento!) e caratterizzati da differenti:
 - Prestazioni e architettura dell'hardware
 - Sistemi operativi
 - Interfacce utente (es, Soft Button...)
 - Comportamento della Java Virtual Machine... e... interpretazioni della specifica
 - Utenti target (consumer, professional, business)

MIDlet e ciclo di vita



- L'applicazione definita dal profilo MIDP è la MIDlet. Questo il suo ciclo di vita:



MIDlet



```
import javax.microedition.midlet.*;

public class MIDletSkeleton extends MIDlet {

    public void startApp() {    }

    public void pauseApp() {    }

    public void destroyApp(boolean unconditional) {    }

}
```

Java Application Descriptor



- Contiene le informazioni sull'applicazione, mostrate all'utente prima che si proceda con il download e l'installazione del software
- Viene generato automaticamente dai tool di sviluppo
- Attenzione: l'indicazione della dimensione del file JAR **deve essere esatta**; in caso contrario alcuni dispositivi rifiutano il caricamento dell'applicazione

Esempio di file JAD



MIDlet-Version: 1.0.0

MIDlet-Vendor: Stefano Sanna

MIDlet-Jar-URL: <http://www.gerdavax.it/jpizza.jar>

MicroEdition-Configuration: CLDC-1.0

MicroEdition-Profile: MIDP-1.0

MIDlet-1: JPizza, , jpizza.JPizza

MIDlet-Jar-Size: 10819

MIDlet-Name: Jpizza

...

...

Deployment



- L'installazione di una applicazione sul dispositivo può essere effettuata:
 - Over-The-Air (OTA): attraverso connessione HTTP a server web
 - Sincronizzazione: attraverso apposita utility fornita dal produttore del terminale
 - OBEX Push: attraverso oggetti inviati attraverso Bluetooth (non supportato da tutti i cellulari)

Deployment OTA

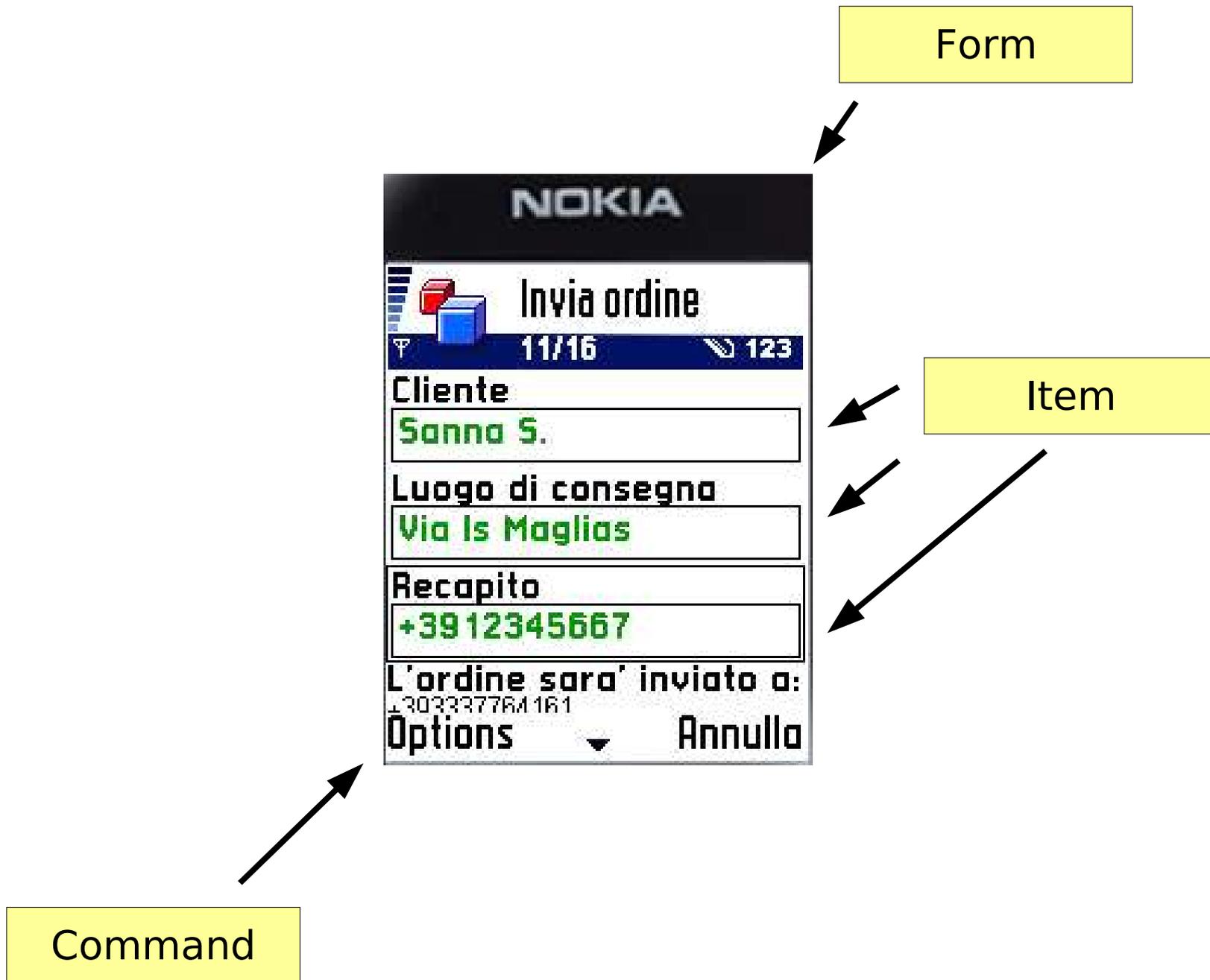
Affinché l'Application Manager del dispositivo riconosca gli oggetti restituiti dal server, questi devono essere qualificati con i mime-type:

- application/java-archive (JAR)
- text/vnd.sun.j2me.app-descriptor (JAD)

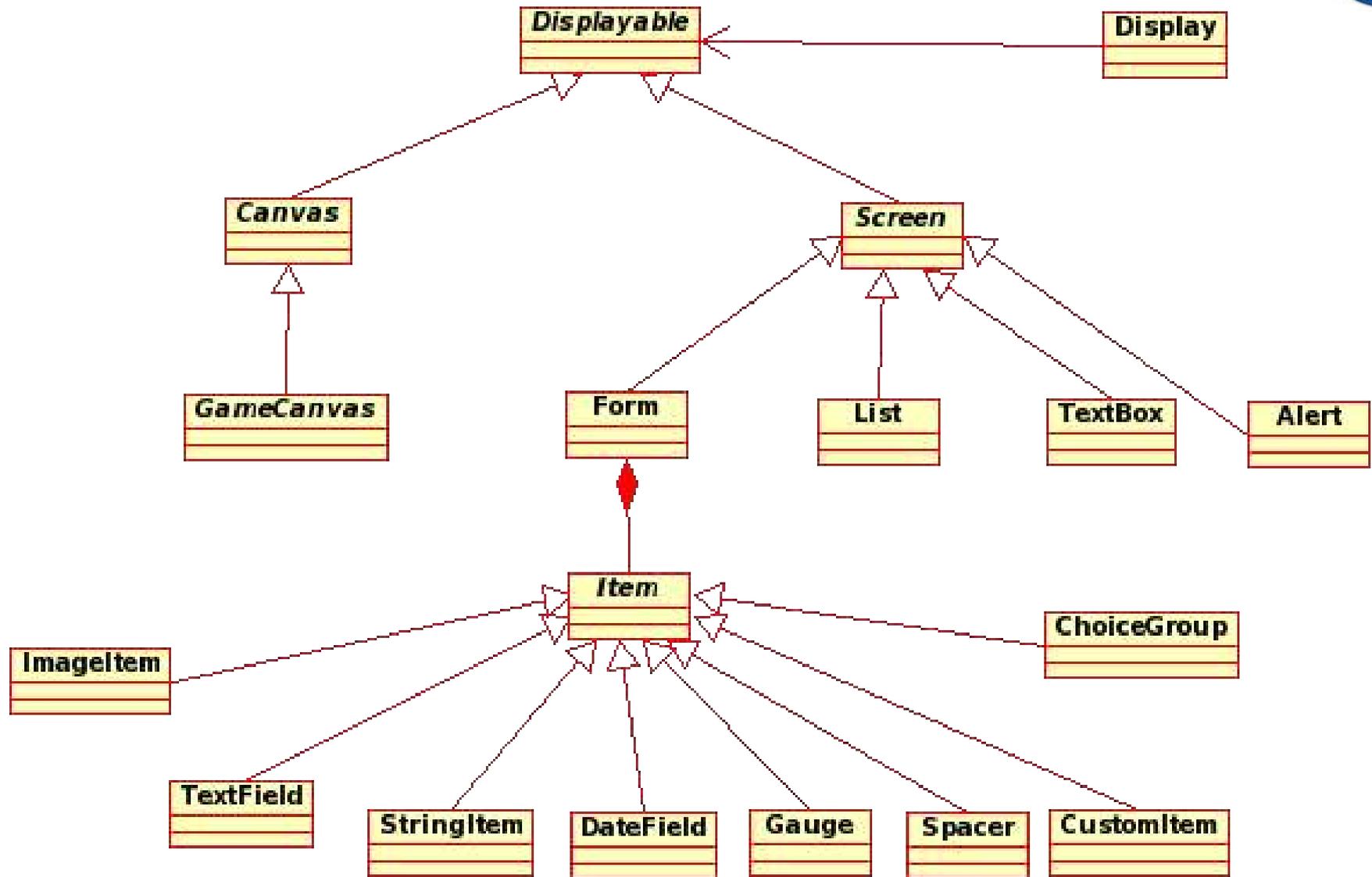


- Il modello di componenti per le GUI su Java ME sono sostanzialmente diverse da quelle per Java SE... perché diversi sono i dispositivi coinvolti!
- Particolarità:
 - Assenza di finestre flottanti
 - Visualizzazione non bloccante
 - Comandi condivisi
 - Maschere di input native

Un semplice Form



LCDUI



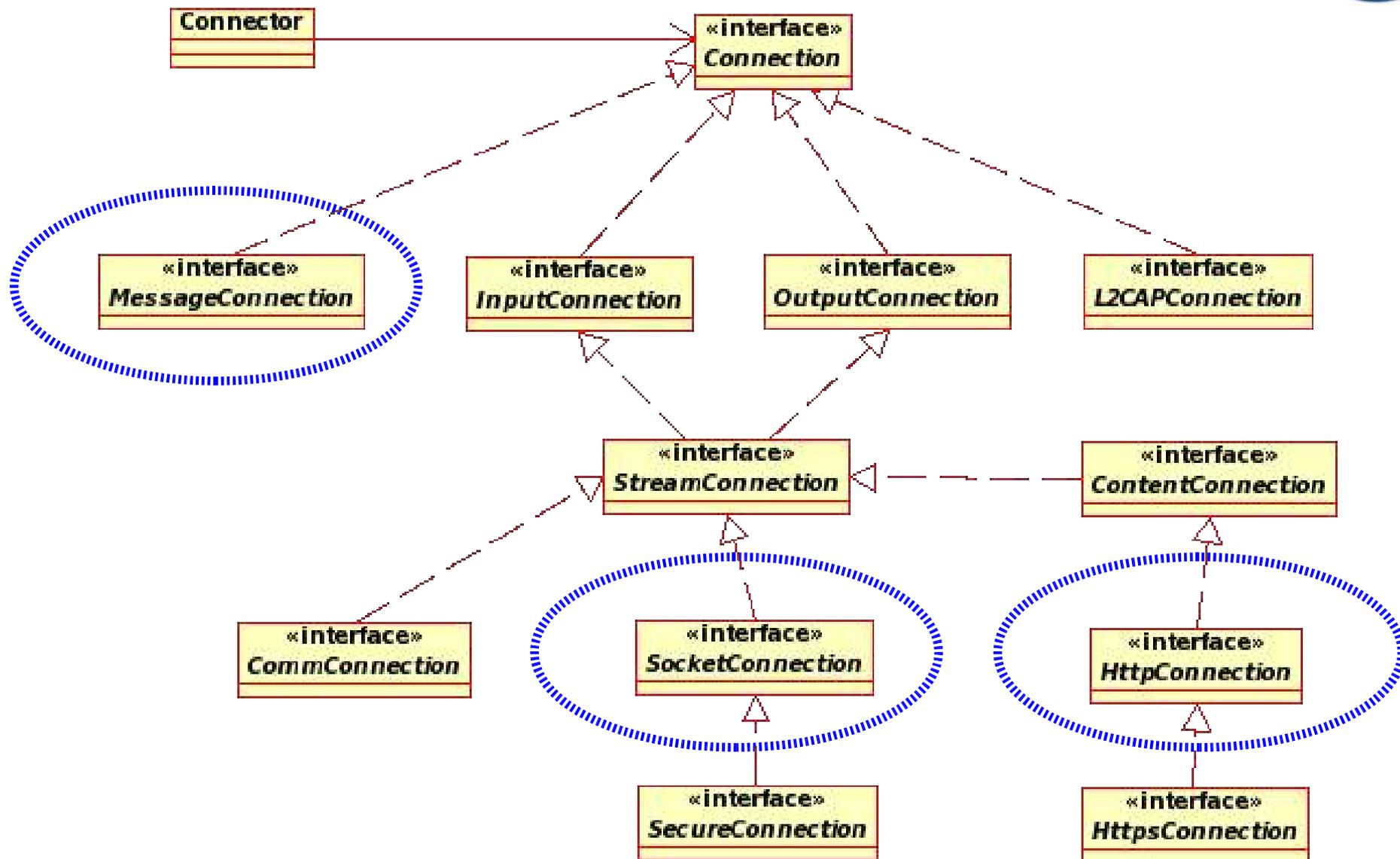
Generic Connection Framework



- La varietà mezzi di comunicazione ha portato alla definizione di un framework astratto per la gestione delle connessioni remote
- La factory *Connector* fornisce *Connection* specializzate per ciascun protocollo supportato, mentre I/O Stream e relativi Adapter/Decorator forniscono gli strumenti di lettura, scrittura e conversione dei dati

```
Connector.open( . . . )
```

Generic Connection Framework



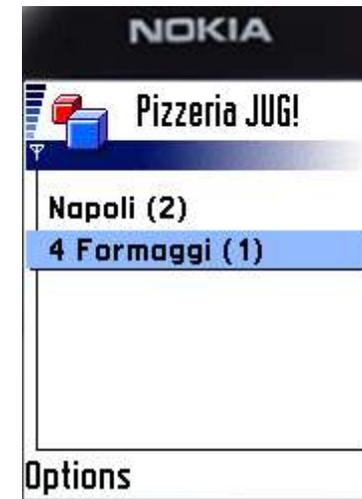
Una semplice MIDlet: JPizza!



- Implementiamo una semplice MIDlet che permette all'utente di effettuare, con il proprio cellulare, l'ordine di una pizza a domicilio:
 - Scegliamo le pizze (tipo e quantità)
 - Indichiamo il nostro indirizzo e recapito telefonico
 - Usiamo gli SMS come mezzo di trasporto



JPizza



JPizza: altro che MicroEdition!



- Comunicazione:
 - Interfaccia web service per la ricezione del menù aggiornato (pizza con asparagi!) e l'invio dell'ordine
- GUI:
 - possibilità di specificare modifiche agli ingredienti di ciascuna pizza (doppia mozzarella, niente origano...)
- Storage:
 - Possibilità di memorizzare le pizze più gradite (dopo la cena...)

API opzionali



- CLDC e MIDP permettono di costruire applicazioni “elementari”: la vera ricchezza è nelle API opzionali:
 - Wireless Messaging API (JSR 120 & 205)
 - Mobile Media API (JSR 135)
 - Mobile 3D Graphics (JSR 184)
 - Bluetooth API (JSR 82)
 - FileConnection & PIM API (JSR 75)
 - Web Services API (JSR 172)
 - Location API (JSR 179)
 - ... molte altre in arrivo! :-)

Ambienti di sviluppo



- Sun J2ME Wireless Toolkit
<http://java.sun.com/j2me>
- Netbeans + Mobility Pack
<http://www.netbeans.org>
- Eclipse + EclipseME
<http://www.eclipseme.org>
- Nokia Developer's Suite for J2ME
<http://www.forum.nokia.com>
- Sony-Ericsson J2ME SDK
<http://developer.sonyericsson.com>
- Motorola J2ME SDK
<http://www.motocoder.com>



Anche su Linux! :-)



The screenshot displays the NetBeans IDE 4.1 environment. The left pane shows a virtual mobile phone with a screen displaying the 'Alert Options' menu. The main workspace is the 'Flow Designer: MIDP-2.0', which visualizes the application's logic flow. It starts at a 'Mobile Device' icon, leading to a 'Start Point' which connects to the 'mainForm [Form]'. From 'mainForm', the flow goes to an 'alert [Alert]' icon, then to a 'Dismiss' point, and finally to a 'list1 [List]' icon. The 'list1 [List]' has a 'Back' button. The right pane, 'Inspector', shows the component hierarchy: typeGroup, timeoutGroup, optionsGroup, alert [Alert], list1 [List], Assigned Command, and backCommand. The bottom status bar indicates 'Building build.xml (run)...' and the system tray shows a clock at 18:06.

Due ottime letture

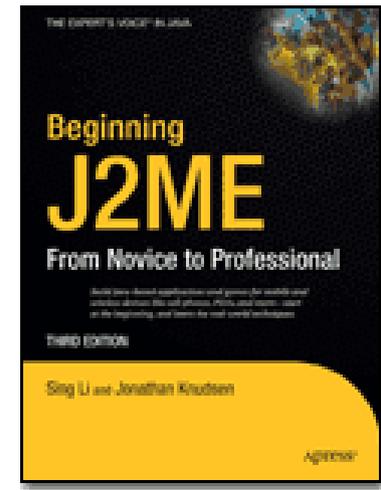


- **Beginning J2ME**

Jonathan Knudsen, Sing Li

APress, 2005

ISBN: 1-59059-479-7



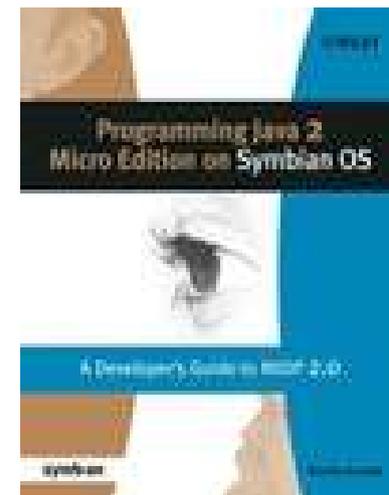
- **Programming Java 2 Micro Edition for Symbian OS:**

A developer's guide to MIDP 2.0

Martin De Jode

Wiley, 2004

ISBN: 0-470-09223-8



Conclusioni: il presente...



- Java ME è una piattaforma matura ed affidabile per lo sviluppo di applicazioni mobile-centric di classe business
- Il vastissimo parco installato rende MIDP leader indiscusso nella telefonia mobile
- La ricchezza della API e la complessità del processo di porting/testing su diverse classi di terminali richiede una forte specializzazione degli sviluppatori
- Il 2006 porterà uno straordinario potenziamento della piattaforma

... e il (bellissimo!) futuro...



- Nokia ha presentato N91, un terminale con caratteristiche decisamente interessanti:
 - GSM, GPRS, EDGE, WCDMA
 - SymbianOS 9
 - HD 4GB
 - Fotocamera 2MPixel
 - J2ME con MIDP 2.0, CLDC 1.1, WMA, MMA, M3G, BTA, FC&PIM, WSA, LocationAPI...

Grazie... :-)



Java ME

Introduzione allo sviluppo di applicazioni per dispositivi mobili

(Versione 1.0)

(C) 2005 Stefano Sanna (gerdavax@tiscali.it)

è garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della Licenza per Documentazione Libera GNU, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation. Una copia della licenza in lingua italiana è disponibile presso:

<http://www.softwarelibero.it/gnudoc/fdl.it.html>

Realizzato in ambiente Linux con OpenOffice 1.1.2

Tutti i marchi commerciali sono di proprietà dei rispettivi titolari e sono stati citati in questa presentazione a solo scopo illustrativo.

