

BlackBerry Java Development Environment

Versione: 4.7.0

Guida alle nozioni fondamentali

Indice

1	Informazioni su BlackBerry e programmazione dei dispositivi BlackBerry	5
	Principi di progettazione per i dispositivi BlackBerry	5
	Versioni e cicli di rilascio	6
	BlackBerry Java Development Environment	6
	API Java ME e Java per BlackBerry	7
	Supporto per le API Java standard	8
	Supporto delle estensioni API Java	8
	Soluzioni BlackBerry	9
	BlackBerry Enterprise Solution	9
	BlackBerry Internet Service	9
	BlackBerry MDS	10
2	Progettazione BlackBerry Java Application	11
	Applicazioni autonome	11
	Applicazioni con sincronizzazione desktop	11
	Applicazioni con accesso wireless, sincronizzazione wireless o avvisi wireless	11
	Applicazioni MIDlet	11
	Applicazioni CLDC	12
	Controllo delle API e firma codice	13
	Modellazione degli oggetti	13
	Multithreading	13
	Procedura consigliata: utilizzo del multithreading	13
	Procedure consigliate per la scrittura di una BlackBerry Java Application efficiente	14
	Procedura consigliata: scrittura di codice efficiente	14
	Procedura consigliata: utilizzo accorto degli oggetti	19
	Procedura consigliata: riduzione delle dimensioni del codice compilato	19
	Assistenza multilingua	21
	Procedura consigliata: archiviazione delle stringhe di testo nei file di risorse	21
	Supporto multimediale	22
	Supporto audio	22
	Supporto fotocamera	22
	Supporto video	22
3	Progettazione navigazione e interfaccia utente	25

Modalità di input e navigazione per l'utente del dispositivo BlackBerry.....	25
Rotellina e trackball a confronto.....	25
Creazione di un'interfaccia utente compatibile con le interfacce utenti BlackBerry standard.....	26
4 Gestione della memoria.....	27
Gestione della memoria.....	27
Memoria del dispositivo BlackBerry.....	27
Risorse principali da riservare.....	27
Procedura consigliata: riduzione della memoria utilizzata.....	28
Gestione della condizione di memoria in esaurimento.....	28
Individuazione della condizione di memoria in esaurimento su un dispositivo BlackBerry.....	28
Risparmio risorse.....	28
Procedura consigliata: selezione efficiente della struttura dati.....	28
Procedura consigliata: consolidazione degli oggetti in gruppi di oggetti.....	29
Garbage collection su un dispositivo BlackBerry.....	30
Garbage collection della RAM in un dispositivo BlackBerry.....	30
Garbage collection completa su un dispositivo BlackBerry.....	30
Garbage collection su un dispositivo BlackBerry BlackBerry.....	30
5 Gestione dei dati.....	31
Supporto API per l'archiviazione di dati nella memoria permanente.....	31
API di memorizzazione permanente.....	31
API del sistema di gestione dei record MIDP.....	31
API FileConnection.....	31
Archiviazione su supporti rimovibili.....	32
Accesso ai dati sulla scheda di memoria microSD.....	32
Backup e sincronizzazione dei dati.....	32
6 Trasporto di dati wireless.....	34
Gateway wireless.....	34
Utilizzo di BlackBerry Enterprise Server come gateway di rete.....	34
Utilizzo del gateway Internet del provider di servizi wireless.....	34
Opzioni alternative di trasporto dati.....	35
Utilizzo della posta elettronica per il trasporto di dati.....	35
Utilizzo di SMS per il trasporto di dati.....	35
Utilizzo della messaggistica PIN per il trasporto di dati.....	35

7	Integrazione delle applicazioni BlackBerry.....	36
	Aggiunta di voci di menu personalizzate.....	36
	Integrazione con le applicazioni BlackBerry Device Software.....	36
	Accesso ai dati della posta elettronica e dell'agenda elettronica.....	36
	Utilizzo di BlackBerry Messenger con una BlackBerry Application.....	36
	Utilizzo dei rilevatori per rispondere alle modifiche delle applicazioni.....	37
8	Considerazioni sulla protezione.....	38
	Crittografia dei dati e BlackBerry Application.....	38
	Crittografia dei dati nel trasporto.....	38
	Crittografia dei dati sul dispositivo BlackBerry.....	38
	Accesso alla memoria.....	38
	Autenticazione.....	39
	Criterio IT e autenticazione dei dispositivi BlackBerry.....	39
	Autenticazione delle applicazioni.....	39
	Autenticazione sul lato server.....	39
	API controllate e firma codice.....	39
	API BlackBerry con accesso controllato.....	40
	Supporto dei criteri IT.....	41
	Controllo delle applicazioni.....	41
	Crittografia file sulle schede microSD.....	41
	Crittografia di dati su una scheda di memoria microSD.....	41
	Utilizzo della scheda di memoria microSD con più di un dispositivo BlackBerry.....	42
	Criteri IT e la scheda di memoria microSD.....	42
9	Verifica di una BlackBerry Java Application.....	43
	Offuscamento di una BlackBerry Java Application.....	43
	Verifica preventiva di una BlackBerry Java Application.....	43
	Verifica delle applicazioni su un BlackBerry Smartphone Simulator.....	43
	Verifica delle applicazioni su un dispositivo BlackBerry.....	44
	Simulatori BlackBerry.....	44
	BlackBerry Smartphone Simulator.....	44
	BlackBerry MDS Simulator.....	44
	BlackBerry Email Simulator.....	45

10	Come rendere disponibili le applicazioni.....	46
	Distribuzione di applicazioni attraverso una connessione computer.....	46
	Distribuzione di un'applicazione da un computer.....	46
	Distribuzione di un'applicazione da una pagina Web.....	46
	Distribuzione di un'applicazione per la verifica.....	46
	Distribuzione di applicazioni sulla rete wireless.....	46
	Pull wireless (attivato dall'utente).....	47
	Push wireless (avviato dal server).....	47
11	Glossario.....	48
12	Note legali.....	51

Informazioni su BlackBerry e programmazione dei dispositivi BlackBerry

1

I dispositivi BlackBerry® forniscono un ambiente wireless Java® ME che supporta applicazioni client/server. Gli sviluppatori delle applicazioni possono creare una BlackBerry® Java Application che presenta interfacce utenti sofisticate per l'immissione di dati e per la ricerca e che supporta il multithreading, l'internazionalizzazione, la comunicazione di rete e l'archiviazione di dati locali. Le applicazioni possono comunicare con le reti mediante connessioni TCP e HTTP standard, a prescindere dalla rete wireless sottostante.

Gli sviluppatori possono creare anche una BlackBerry Java Application che si integra in maniera ideale con le principali applicazioni del dispositivo BlackBerry, come l'elenco dei messaggi, le applicazioni dell'agenda elettronica, il telefono e il browser, per un'esperienza utenti perfetta.

Principi di progettazione per i dispositivi BlackBerry

Le applicazioni progettate per i dispositivi BlackBerry® dovrebbero fornire il giusto equilibrio tra un'ottima esperienza utenti ed una durata prolungata della batteria. Quando si progetta la propria applicazione per il dispositivo BlackBerry, tenere in considerazione le differenze esistenti tra i dispositivi mobili e i computer. I dispositivi mobili

- hanno uno schermo dalle dimensioni più contenute, pertanto sono in grado di visualizzare un numero limitato di caratteri
- presentano processori dalle velocità inferiori
- utilizzano reti wireless con un periodo di latenza più lungo rispetto alle LAN standard
- hanno meno memoria disponibile
- presentano batterie dalla durata più breve
- visualizzano una schermata alla volta

Gli utenti dei dispositivi mobili utilizzano le applicazioni in maniera diversa da come lo farebbero in un computer. Nei dispositivi mobili, gli utenti si aspettano di trovare le informazioni rapidamente. Ad esempio, un sistema CRM può fornire un'enorme quantità di informazioni, ma gli utenti richiedono solo una piccola quantità di informazioni alla volta. L'interfaccia utente del dispositivo BlackBerry è progettata in modo che gli utenti possano eseguire facilmente le attività ed accedere alle informazioni in maniera rapida.

Quando si progettano le applicazioni per i dispositivi BlackBerry, tentare di rimanere il più possibile conforme alle altre applicazioni del dispositivo BlackBerry. Seguire le seguenti istruzioni:

- Se possibile, utilizzare o estendere i componenti dell'interfaccia utente esistenti in modo che l'applicazione possa ereditare il comportamento predefinito del componente.
- Seguire il modello di navigazione standard il più attentamente possibile in modo che gli utenti possano utilizzare in maniera completa la tastiera e la trackball.

- Eseguire tutte le azioni disponibili nel menu. Verificare che le azioni disponibili nel menu siano pertinenti al contesto attuale degli utenti.

Quando si progetta l'applicazione, seguire anche le linee guida seguenti:

- Concentrarsi sulla prossima attività degli utenti. Semplificare la selezione e la presentazione dei dati per visualizzare solo le informazioni di cui gli utenti hanno bisogno in qualsiasi un momento.
- Visualizzare le informazioni in modo tale da utilizzare lo schermo in maniera efficace.

Prima di progettare l'applicazione, considerare l'utilizzo delle applicazioni principali del dispositivo BlackBerry o di BlackBerry Smartphone Simulator per conoscere in maniera più approfondita il modello di navigazione e le procedure consigliate per progettare l'interfaccia utente dell'applicazione.

Versioni e cicli di rilascio

Tutti i dispositivi BlackBerry® includono una versione specifica di BlackBerry® Device Software e di BlackBerry® Java® Virtual Machine. Per determinare la versione di BlackBerry Device Software per un dispositivo BlackBerry, fare clic su **Informazioni su** nelle opzioni del dispositivo. È possibile aggiornare BlackBerry Device Software. Ad esempio, è possibile aggiornare un dispositivo BlackBerry alla versione 4.0 o 4.1 di BlackBerry Device Software.

Con il rilascio di ogni versione principale di BlackBerry Device Software e di BlackBerry JVM, Research In Motion include le API Java e la versione di BlackBerry® Java® Development Environment corrispondenti. La versione di BlackBerry Device Software determina la versione di BlackBerry® Integrated Development Environment da poter utilizzare per sviluppare le applicazioni. Ad esempio, RIM ha rilasciato contemporaneamente la versione 4.0 di BlackBerry Device Software e la versione 4.1 di BlackBerry JDE. La versione 4.0 di BlackBerry JDE include il supporto per le API introdotte nella versione 4.0 di BlackBerry Device Software e nella versione 4.0 di BlackBerry JVM. Le applicazioni create utilizzando la versione 4.0 di BlackBerry JDE funzionano solo sui dispositivi BlackBerry che eseguono la versione 4.0 o successiva di BlackBerry Device Software.

È possibile utilizzare i criteri seguenti per decidere quale versione di BlackBerry JDE utilizzare per sviluppare un'applicazione:

- Se l'applicazione non necessita di funzionalità hardware specifiche del dispositivo BlackBerry o delle estensioni API rilasciate di recente, utilizzare la versione 4.0 di BlackBerry JDE per sviluppare l'applicazione.
- Se l'applicazione è progettata per essere eseguita esclusivamente sullo smartphone BlackBerry® Pearl™ 8100, utilizzare la versione 4.2 o successiva di BlackBerry JDE.

BlackBerry Java Development Environment

BlackBerry® Java® Development Environment è un ambiente di sviluppo e simulazione completamente integrato per la creazione di una BlackBerry® Java Application per i dispositivi BlackBerry. Mediante BlackBerry JDE, gli sviluppatori possono creare applicazioni utilizzando il linguaggio di programmazione Java® ME e le API Java estese per BlackBerry.

BlackBerry Java Development Environment comprende gli strumenti di sviluppo seguenti:

- BlackBerry® Integrated Development Environment
- BlackBerry Smartphone Simulator
- API Java ME e API BlackBerry
- applicazioni campione

BlackBerry IDE include una suite completa di strumenti di modifica e debug ottimizzati per lo sviluppo di una BlackBerry Java Application. BlackBerry Smartphone Simulator fornisce un ambiente di tipo Windows® completo ed è progettato per simulare l'interazione tra le interfacce utenti e l'utente, le connessioni di rete, i servizi di posta elettronica e la sincronizzazione dei dati wireless.

Il pacchetto componenti di BlackBerry Java Development Environment comprende gli strumenti seguenti per lo sviluppo all'interno di IDE di terze parti, come NetBeans™™ o Eclipse®:

- RASP: è possibile utilizzare questo strumento per compilare file .java e .jar in file .cod che possono essere eseguiti in BlackBerry Smartphone Simulator o su un dispositivo BlackBerry.
- JavaLoader: è possibile utilizzare questo strumento per aggiungere o aggiornare un'applicazione su un dispositivo BlackBerry a scopo di verifica e per visualizzare informazioni sui file .cod dell'applicazione.
- BlackBerry® Signature Tool: è possibile utilizzare questo strumento per inviare richieste di firma codice a BlackBerry® Signing Authority Tool.
- Preverify Tool: è possibile utilizzare questo strumento per la verifica parziale delle classi prima di caricare la propria applicazione in un dispositivo BlackBerry.
- JDWP: è possibile utilizzare questo strumento per eseguire il debug delle applicazioni mediante ambienti di sviluppo integrati di terzi.

API Java ME e Java per BlackBerry

Java® ME è una piattaforma standard di settore che definisce serie comuni di API Java per diversi tipi di dispositivi wireless e incorporati. Un'applicazione Java ME su un dispositivo BlackBerry® viene eseguita su BlackBerry® Java® Virtual Machine che fornisce alle applicazioni tutti i servizi runtime ed esegue operazioni come allocazioni della memoria tipiche, controlli di sicurezza e garbage collection.

Lo standard MIDP Java ME risponde alla necessità delle API e di BlackBerry JVM di un dispositivo wireless limitato con un'interfaccia utente. Il dispositivo BlackBerry supporta lo standard MIDP Java ME definito come JSR 118. Lo standard MIDP Java ME fornisce una serie principale di API Java che qualsiasi dispositivo BlackBerry è in grado di supportare, a prescindere dal sistema operativo sottostante. Spesso gli sviluppatori possono creare un'applicazione Java utilizzando le API MIDP standard ed eseguendo l'applicazione in diversi tipi di dispositivi.

Supporto per le API Java standard

Il dispositivo BlackBerry® e BlackBerry® Java® Development Environment supportano lo standard MIDP Java® ME che fornisce il nucleo di API Java da poter utilizzare per sviluppare applicazioni del dispositivo wireless. Il dispositivo BlackBerry e BlackBerry® Java® Development Environment supportano anche le JSR seguenti:

- JSR 30: Connected Limited Device Configuration versione 1.0 (supportato sui dispositivi dotati di BlackBerry® Device Software versione 4.0 o precedente)
- JSR 37: Mobile Information Device Profile versione 1.0 (supportato sui dispositivi dotati di BlackBerry Device Software versione 4.0 o precedente)
- JSR 75: Portable Optional Packages per la piattaforma J2ME (PDAP) con supporto esclusivo per le API PIM e API FileConnection per Java ME (supportate sui dispositivi dotati di BlackBerry Device Software versione 4.2 o successiva)
- JSR 82: API Java per Bluetooth®
- JSR 118: Mobile Information Device Profile versione 2.0
- JSR 120: Wireless Messaging API (WMA) versione 1.1
- JSR 135: Mobile Media APIs (MM) versione 1.1
- JSR 139: Connected Limited Device Configuration versione 1.1
- JSR 172: J2ME Web Services
- JSR 177: Security and Trust Services API per J2ME (SATSA)
- JSR 179: Location API per Java ME
- JSR 185: Java Technology for the Wireless Industry (JTWI)
- JSR 205: Wireless Messaging API 2.0
- JSR 211: Content Handler API
- JSR 226: Scalable 2D Vector Graphics API per Java ME
- JSR 238: Mobile Internationalization API

Supporto delle estensioni API Java

I dispositivi BlackBerry® supportano le seguenti API Java® che non fanno parte delle definizioni JSR standard e che forniscono funzionalità migliori rispetto a quelle disponibili nelle librerie API MIDP standard.

API	Descrizione
API di interfaccia utente	È possibile utilizzare queste API per creare schermate, voci di menu e tutti i componenti dell'interfaccia utente.
API di memorizzazione dati permanente	È possibile utilizzare queste API per archiviare localmente dati personalizzati all'interno della propria applicazione.

API	Descrizione
API di rete e I/O	È possibile utilizzare queste API per stabilire le connessioni di rete e leggere o scrivere dati in un'applicazione sul lato server.
Rilevatori di eventi	È possibile utilizzare i rilevatori di eventi per rispondere a eventi avviati dall'utente del dispositivo BlackBerry o dal sistema su un dispositivo BlackBerry.
API di integrazione applicazioni	È possibile utilizzare queste API per l'integrazione con applicazioni di BlackBerry esistenti: posta elettronica, telefono, calendario, contatti, browser, fotocamera, lettore multimediale ed elenco delle attività.
Utilità aggiuntive	È possibile utilizzare queste API aggiuntive per la crittografia e la compressione dei dati, l'analisi XML, la connettività Bluetooth®, i servizi basati sulla posizione e altro ancora.

Soluzioni BlackBerry

Gli utenti del dispositivo BlackBerry® possono utilizzare BlackBerry® Enterprise Server o BlackBerry® Internet Service oppure avvalersi di entrambi sullo stesso dispositivo. È importante comprendere le differenze esistenti tra BlackBerry Enterprise Server e BlackBerry Internet Service e quali tipi di utenti si intendono supportare, poiché questo fattore può incidere sulle modalità di trasporto e di gestione della sincronizzazione dei dati.

BlackBerry Enterprise Solution

BlackBerry® Enterprise Server appartiene a BlackBerry® Enterprise Solution. BlackBerry Enterprise Server è protetto dal firewall aziendale e fornisce un gateway wireless per gli utenti del dispositivo BlackBerry per accedere alla posta elettronica aziendale e ai dati dell'agenda elettronica. BlackBerry Enterprise Server fornisce anche le seguenti funzionalità principali:

- crittografia e compressione dati
- utilità di gestione del dispositivo BlackBerry e di controllo
- provisioning delle applicazioni semplificato
- gateway autenticato per l'accesso a intranet da una BlackBerry® Java Application

BlackBerry Internet Service

Gli utenti del dispositivo BlackBerry® che non sono associati a un BlackBerry® Enterprise Server possono utilizzare BlackBerry® Internet Service. BlackBerry Internet Service è un servizio Internet e di posta elettronica per i dispositivi BlackBerry, progettato per fornire agli utenti il recapito automatico dei messaggi di posta elettronica, l'accesso wireless agli allegati dei messaggi di posta elettronica e l'accesso ai contenuti Internet.

BlackBerry Internet Service supporta la connettività diretta HTTP e TCP/IP a Internet da applicazioni BlackBerry® Java Application di terzi.

BlackBerry MDS

Per consentire ad una BlackBerry® Java Application di accedere alle risorse protette dal firewall aziendale, BlackBerry® Enterprise Server include BlackBerry® Mobile Data System. BlackBerry MDS fornisce proxy HTTP e TCP/IP per una BlackBerry Java Application, consentendo al dispositivo BlackBerry di comunicare con l'applicazione e con i server Web protetti dal firewall aziendale senza dover disporre di software VPN aggiuntivo. Le applicazioni che inviano dati mediante BlackBerry Enterprise Server come un gateway possono trarre vantaggio dalla connettività aziendale semplificata, dalla crittografia dati e dalla compressione dei dati e dall'indipendenza della rete aziendale offerte da BlackBerry® Enterprise Solution. Inoltre, BlackBerry MDS fornisce un'interfaccia aperta, consentendo alle applicazioni sul lato server protette dal firewall aziendale di trasferire contenuto alle applicazioni sui dispositivi BlackBerry.

Progettazione BlackBerry Java Application

2

Applicazioni autonome

È possibile utilizzare le API BlackBerry® per creare applicazioni autonome, come giochi e guide di riferimento che possono essere eseguite come applicazioni non in linea. Prima di compilare un'applicazione è possibile aggiungere i dati richiesti sulle risorse. Gli utenti del dispositivo BlackBerry possono installare l'applicazione tramite la rete wireless o utilizzando BlackBerry® Desktop Software. Una volta installata sul dispositivo BlackBerry, l'applicazione non richiede alcuna connessione alla rete wireless o al computer per poter essere eseguita.

Applicazioni con sincronizzazione desktop

È possibile utilizzare le API BlackBerry® per creare applicazioni dotate di funzionalità di sincronizzazione del desktop, come applicazioni della guida di riferimento e dell'agenda elettronica. L'utente connette il dispositivo BlackBerry a un computer per gestire e sincronizzare i dati che si trovano sul computer.

Research In Motion® non fornisce conduit HotSync® o altri moduli di sincronizzazione diretta del database. È necessario creare il codice di sincronizzazione e l'utente del dispositivo BlackBerry deve avviare manualmente il processo di sincronizzazione dei dati. Dopo aver installato l'applicazione sul dispositivo BlackBerry, l'utente del dispositivo BlackBerry deve sincronizzare manualmente le informazioni il dispositivo al computer mediante una connessione seriale, USB o Bluetooth®.

Applicazioni con accesso wireless, sincronizzazione wireless o avvisi wireless

È possibile utilizzare le API BlackBerry® per creare applicazioni che trasferiscono proattivamente i contenuti sulla rete wireless ai dispositivi BlackBerry negli ambienti che utilizzano BlackBerry® Enterprise Server. Una BlackBerry® Java Application per i dispositivi BlackBerry utilizza una connessione wireless a Internet o alla intranet aziendale per consentire agli utenti del dispositivo BlackBerry di accedere ad applicazioni e dati remoti. BlackBerry® Java® Development Environment fornisce API che possono essere utilizzate nelle applicazioni per stabilire connessioni di rete ai server su Internet o nella intranet aziendale.

Applicazioni MIDlet

Il modello delle applicazioni MIDlet appartiene alla descrizione MIDP. La classe principale di un MIDlet estende sempre la classe MIDlet e deve utilizzare i metodi per `startApp()`, `pauseApp()` e `destroyApp()`.

Vantaggi	Svantaggi
<ul style="list-style-type: none"> Le applicazioni possono essere spostate in altri dispositivi che supportano anche la descrizione MIDP. 	<ul style="list-style-type: none"> Le applicazioni possono utilizzare solo le API dell'interfaccia utente che esistono nella libreria <code>javax.microedition.lcdui</code>. Il modello presuppone che tutti i processi dell'applicazione terminino quando l'applicazione viene chiusa. Le applicazioni non possono essere avviate automaticamente in background quando si attiva il dispositivo.

Applicazioni CLDC

CLDC è una specifica di un framework per Java® ME. Un'applicazione CLDC può estendere la classe `Application` e si avvia con un metodo `main()` standard.

La maggior parte delle applicazioni di esempio incluse in BlackBerry® Java® Development Environment utilizzano API CLDC, BlackBerry e MIDP. Le funzionalità principali di BlackBerry® Device Software (compresi l'elenco dei messaggi, l'elenco di contatti, il calendario e il browser) utilizzano API CLDC e BlackBerry.

Vantaggi	Svantaggi
<ul style="list-style-type: none"> Le applicazioni possono essere trasferite in altre piattaforme. 	<ul style="list-style-type: none"> Le applicazioni non possono eseguire thread di background attivi dopo la loro chiusura. Le applicazioni non possono utilizzare API IPC per scambiare informazioni con altre applicazioni. Le applicazioni non possono accedere agli eventi del touchscreen su un dispositivo BlackBerry. Le applicazioni non possono utilizzare dati dall'accelerometro su un dispositivo BlackBerry. Le applicazioni non possono integrare BlackBerry® Browser o BlackBerry® Maps sull'interfaccia utente. Le applicazioni non possono modificare l'icona visualizzata sulla schermata iniziale. Le applicazioni non possono accedere agli eventi della trackball e della rotellina.

Controllo delle API e firma codice

Quando si sviluppa una BlackBerry® Java® Application per i dispositivi BlackBerry, è possibile utilizzare solo API Java pubbliche pubblicate e documentate nei documenti Javadoc™ in BlackBerry® Java® Development Environment. BlackBerry® Java® Virtual Machine sul dispositivo BlackBerry è progettata per proteggere i dati sottostanti e il sistema operativo, così le applicazioni non possono richiamare API non documentate o supportate o accedere a dati non esposti esplicitamente attraverso le API. Se si tenta di utilizzare API Java non esposte pubblicamente, l'applicazione riceve un messaggio di errore al momento dell'esecuzione.

Le API pubbliche sono aperte o firmate. Le API firmate espongono metodi di accesso ai dati utente del dispositivo BlackBerry o ad altre informazioni sul dispositivo considerate sensibili. È possibile utilizzare API firmate, ma è necessario richiedere e ricevere una serie di chiavi di firma codice da Research In Motion. È necessario firmare digitalmente l'applicazione prima di installarla su un dispositivo BlackBerry. La firma codice non certifica o approva un'applicazione; consente a RIM di identificare l'autore di un'applicazione che utilizza API sensibili, se l'applicazione è dannosa.

Per richiedere una serie di chiavi di firma codice, visitare <http://na.blackberry.com/eng/developers/javaappdev/codekeys.jsp>. La serie di chiavi di firma codice verrà consegnata tra 10 giorni circa.

Modellazione degli oggetti

Se si utilizza il modello di applicazione MIDlet o CLDC, è necessario adottare un approccio orientato a oggetti quando si progetta la propria applicazione per il dispositivo BlackBerry®. In un approccio orientato a oggetti, gli sviluppatori utilizzano degli oggetti per contenere il codice comune a un processo o a una funzione specifici. Ad esempio, uno sviluppatore può utilizzare oggetti separati per controllare l'attività della rete, l'archiviazione dei dati, l'elaborazione e la manipolazione dei dati e l'interazione dell'interfaccia utente. Quando si progettano le applicazioni, cominciare con un modello a oggetti valido.

Multithreading

Il sistema operativo BlackBerry® è un sistema operativo multithread: ciò significa che diversi applicazioni e processi possono essere eseguiti contemporaneamente sul dispositivo BlackBerry. Ad esempio, le applicazioni possono utilizzare thread in background per gestire attività con utilizzo intensivo del processore o comunicazioni di rete, in modo che non influiscano sul thread principale. Se un'applicazione crea thread in background e un utente del dispositivo BlackBerry chiude l'applicazione, i thread in background possono rimanere attivi.

Procedura consigliata: utilizzo del multithreading

Utilizzare in maniera efficace le funzionalità di multithreading del sistema operativo BlackBerry®. In particolare, creare sempre un nuovo thread per le connessioni di rete o per altre operazioni di lunga durata (più di un decimo di un secondo). Utilizzare i thread di background per i rilevatori o per altri processi che vengono eseguiti in background all'avvio dell'applicazione.

Procedure consigliate per la scrittura di una BlackBerry Java Application efficiente

Procedura consigliata: scrittura di codice efficiente

Per consentire ad una BlackBerry® Java® Application di utilizzare le risorse in modo efficiente, seguire le linee guida seguenti:

- Utilizzare variabili locali.
- Utilizzare la sintassi abbreviata per eseguire la valutazione delle condizioni booleane.
- Creare classi finali.
- Utilizzare un numero `int` invece di un `Long`.
- Evitare la garbage collection.
- Utilizzare variabili statiche per le stringhe.
- Evitare il costruttore `String(String)`.
- Utilizzare una `String` statica finale o non finale, in base ai requisiti dell'applicazione.
- Scrivere cicli efficienti.
- Ottimizzare la sottoespressioni.
- Ottimizzare le operazioni di divisione.
- Evitare `java.util.Enumeration`.
- Eseguire i cast mediante `instanceof`.
- Eseguire la valutazione delle condizioni mediante `instanceof`.
- Evitare di utilizzare `StringBuffer.append(StringBuffer)`.
- Evitare la restituzione `Null`.
- Evitare di passare un valore `Null` nei metodi.
- Prestare attenzione quando si passa un valore `Null` in un costruttore.
- Utilizzare `Long` per ottenere identificatori univoci.
- Chiudere le applicazioni correttamente.
- Stampare una traccia di stack.

Utilizzo delle variabili locali

Utilizzare le variabili locali quando possibile. L'accesso alle variabili locali è più efficiente dell'accesso ai membri delle classi.

Utilizzo della sintassi abbreviata per eseguire la valutazione delle condizioni booleane

Per eseguire la valutazione una condizione `Boolean`, utilizzare la sintassi abbreviata. Il codice compilato risultante è più breve.

Esempio di codice

```
return( boolean_expression );
```

Rendere le classi finali

Quando si creano librerie di codice, contrassegnare le classi come `final` se si è certi che gli sviluppatori non le estenderanno in futuro. La presenza della parola chiave `final` consente al compilatore di generare più codice efficiente.

Per impostazione predefinita, il compilatore di BlackBerry® Java® Development Environment contrassegna come `final` tutte le classi che non vengono estese in un file `.cod` dell'applicazione.

Utilizzo di un numero `int` invece di un `long`

In Java®, un `long` è un numero intero a 64 bit. Perché i dispositivi BlackBerry® utilizzano un processore a 32 bit, le operazioni possono essere eseguite da due a quattro volte più velocemente se si utilizza un numero `int` invece di un `long`.

Come evitare la garbage collection

Evitare di richiamare `System.gc()` per eseguire un'operazione di garbage collection poiché potrebbe richiedere troppo tempo nei dispositivi BlackBerry® la cui memoria disponibile è limitata. Lasciare che BlackBerry® Java® Virtual Machine esegua l'operazione di garbage collection.

Utilizzo delle variabili statiche per le stringhe

Quando si definiscono i campi statici (chiamati anche campi classe) di un tipo `String`, è possibile aumentare la velocità dell'applicazione mediante le variabili statiche (non `final`) invece delle costanti (`final`). Il contrario vale per i tipi di dati primitivi, come `int`.

Ad esempio, si desidera creare un oggetto `String` come illustrato di seguito:

```
private static final String x="example";
```

Per questa costante statica (contraddistinta dalla parola chiave `final`), ogni volta che si utilizza la costante, un'istanza temporanea `String` viene creata. Il compilatore elimina "x" e lo sostituisce con la stringa "example" nel bytecode, in modo che BlackBerry® Java® Virtual Machine esegua una ricerca della tabella hash ogni volta che si fa riferimento a "x".

Al contrario, per una variabile statica (senza la parola chiave `final`), la `String` viene creata una volta sola. BlackBerry JVM esegue la ricerca della tabella hash solo quando inizializza "x", rendendo l'accesso più veloce.

```
private static String x="example";
```

È possibile utilizzare costanti pubbliche (vale a dire, i campi `final`), ma è necessario contrassegnare le variabili come private.

Come evitare il costruttore `String(String)`

In una BlackBerry® Java Application, ogni stringa tra virgolette è un'istanza della classe `java.lang.String`. Creare una stringa senza utilizzare il costruttore `java.lang.String(String)`.

Esempio di codice

```
String str = "abc"; String str = "found " + n + " items";
```

Scrittura di cicli efficienti

Se è probabile il proprio contenitore includa più di un elemento, assegnare le dimensioni a una variabile locale.

Se l'ordine con cui si ripetono le voci non è importante, è possibile eseguire una ripetizione all'indietro per evitare ulteriori variabili locali sullo stack ed eseguire il confronto più velocemente.

Esempio di codice

```
int size = vector.size(); for( int i = 0; i < size; ++i ) { ... } for( int i = vector.size() - 1; i >= 0; --i ) { ... }
```

Ottimizzazione delle sottoespressioni

Se si utilizza la stessa espressione due volte, adottare una variabile locale.

Esempio di codice

```
int tmp = i+1; one( tmp ); two( tmp );
```

Ottimizzazione delle operazioni di divisione

Le operazioni di divisione possono essere lente sui dispositivi BlackBerry® perché il processore non dispone di istruzioni per la divisione dell'hardware.

Quando il codice divide un numero positivo per due, utilizzare il tasto MAIUSC di destra di un'unità (`>> 1`). Utilizzare il tasto MAIUSC di destra (`>>`) solo quando si è certi che si sta lavorando con un valore positivo.

Esempio di codice

```
int = width >> 1;
```

Come evitare java.util.Enumeration

Evitare l'utilizzo di oggetti `java.util.Enumeration` a meno che non si desideri nascondere dei dati (in altre parole, per restituire un'enumerazione dei dati anziché i dati stessi). La richiesta di un vettore o di una tabella hash per un oggetto `Enumeration` è lenta e crea informazioni superflue. Se un altro thread potrebbe modificare il vettore, sincronizzare l'iterazione. Java® SE utilizza un oggetto `Iterator` per operazioni simili, ma gli oggetti `Iterator` non sono disponibili in Java® ME.

Esempio di codice

```
for( int i = v.size() - 1; i >=0; --i ) { o = v.elementAt( i ); ... } synchronized( v )
{ for( int i = v.size() - 1; i >=0; --i ) { o = v.elementAt( i ); ... } }
```

Esecuzione di cast mediante instanceof

Utilizzare `instanceof` per verificare il completamento di un cast.

Esempio di codice

```
if( x instanceof String ) { (String)x.whatever(); } else { ... } x
```

Valutazione delle condizioni mediante instanceof

Per produrre codice più piccolo e più veloce, se si esegue la valutazione di una condizione mediante `instanceof`, non eseguire la valutazione esplicita se la variabile è nulla.

Esempio di codice

```
if( e instanceof ExampleClass ) { ... } if( ! ( e instanceof ExampleClass ) ) { ... }
```

Come evitare StringBuffer.append (StringBuffer)

Per aggiungere un buffer `String` ad un altro, una BlackBerry® Java Application dovrebbe utilizzare `net.rim.device.api.util.StringUtilities.append (StringBuffer dst, StringBuffer src[, int offset, int length])`.

Esempio di codice

```
public synchronized StringBuffer append(Object obj) { if (obj instanceof StringBuffer)
{ StringBuffer sb = (StringBuffer)obj; net.rim.device.api.util.StringUtilities.append
( this, sb, 0, sb ) return this; } return append(String.valueOf(obj)); }
```

Come evitare la restituzione Null

Se si scrive un metodo pubblico che restituisce un oggetto, il metodo dovrebbe restituire Null solo in presenza delle condizioni seguenti:

- L'applicazione prevede la comparsa di un valore Null durante il suo normale funzionamento.
- Il parametro Javadoc™ `@return` per il metodo afferma che Null è un valore restituito possibile.

Se l'applicazione non prevede un valore restituito Null, il metodo dovrebbe inviare a un'eccezione appropriata che forza il chiamante del metodo ad affrontare esplicitamente il problema. È possibile che il chiamante del metodo non debba verificare l'eventuale presenza di un valore restituito Null a meno che non invii egli stesso un'eccezione Null.

Come evitare di passare un valore Null nei metodi

Non passare parametri Null in un metodo API a meno che i materiali di riferimento API per BlackBerry® Java® Development Environment confermino esplicitamente che il metodo li supporta.

Prestare attenzione quando si passa un valore Null ad un costruttore

Per evitare ambiguità quando si passa un valore Null ad un costruttore, eseguire il cast di Null sull'oggetto appropriato.

Se una classe ha almeno due costruttori, il passaggio di un parametro Null potrebbe non identificare univocamente il costruttore da utilizzare. Di conseguenza, il compilatore segnala un errore.

Quando si esegue il cast di un valore Null sull'oggetto appropriato, specificare quale costruttore deve essere utilizzato dal compilatore. Questa procedura fornisce anche compatibilità con le versioni future se versioni successive dell'API aggiungono nuovi costruttori.

Esempio di codice

```
new someObject ((someObject)null );
```

Utilizzo dei valori long per gli identificatori univoci

Utilizzare un identificatore `Long` anziché un identificatore `String` per le costanti univoche, come GUID, chiavi tabella hash e identificatori di stato o contesto.

Affinché gli identificatori rimangano univoci all'interno di una BlackBerry® Java Application, utilizzare le chiavi generate da un'applicazione sulla base dell'hash di una `String`. Nella `String` di input, includere informazioni sufficienti per rendere l'identificatore univoco. Ad esempio, utilizzare un nome pacchetto completamente qualificato come `com.rim.samples.docs.helloworld`.

Chiusura corretta delle applicazioni

Prima di richiamare `System.exit(int status)`, eseguire tutte le operazioni di pulizia necessarie, ad esempio rimuovendo gli oggetti dall'archivio runtime di cui le applicazioni non hanno più bisogno.

La stampa di una traccia di stack

Quando si esegue il debug dell'applicazione, intercettare un'istanza `Throwable` per visualizzare la traccia di stack.

Esempio di codice

```
catch (Throwable t) { t.printStackTrace(); }
```

Procedura consigliata: utilizzo accorto degli oggetti

Per consentire a una BlackBerry® Java Application di utilizzare in modo efficiente le risorse di memoria, prendere in considerazione le domande seguenti:

- Date le dimensioni di un'applicazione, tutti gli oggetti sono necessari?
- L'applicazione è in grado di archiviare oggetti che rappresentano delle primitive, come `Long`, `Integer`, e `Boolean`, come primitive anziché come oggetti?
- Tutti gli oggetti archiviati sono necessari?
- Qualsiasi istanze `Vector` e `Hashtable` esistono? Queste istanze sono necessarie? Se sì, quanti handle di `Object` non vengono utilizzati in `Vector` o in `Hashtable` perché le dimensioni iniziali sono superiori a quelle richieste?
- Quanti `Objects` vengono creati dall'applicazione e poi eliminati? In altre parole, quanti `Objects` con scopo specifico vengono creati dall'applicazione?

Procedura consigliata: riduzione delle dimensioni del codice compilato

Per ridurre le dimensioni del codice compilato, seguire le linee guida seguenti:

- Impostare un accesso appropriato.
- Evitare la creazione di interfacce.
- Utilizzare classi statiche interne.
- Evitare un'inizializzazione dei campi non necessaria
- Importare classi singole.

Impostazione dell'accesso appropriato

Quando si creano librerie di codice, è possibile ridurre considerevolmente le dimensioni del codice compilato utilizzando i modificatori di accesso appropriati per i campi e i metodi. Dichiarare i campi come privati quando possibile. Oltre ad essere una buona abitudine per la codifica, consente al compilatore di ottimizzare il file .cod. Quando possibile, utilizzare l'accesso predefinito (`pacchetto`) anziché l'accesso pubblico (omettere le parole chiave `pubbliche` e `protette`).

Come evitare la creazione di interfacce

Quando si creano librerie API, evitare di creare interfacce a meno che non si prevedano implementazioni multiple dell'API. Le interfacce producono un codice di maggiori dimensioni e più lento.

Utilizzo delle classi statiche interne

Quando si utilizza una classe interna per nascondere una classe all'interno di un'altra, ma la classe interna non fa riferimento all'oggetto della classe esterna, definire statica la classe interna. Questa azione evita la creazione di un riferimento alla classe esterna.

Se si utilizza una classe interna per l'ambito del nome, renderla statica.

Esempio di codice

```
class outer { static class inner { ... } }
```

Come evitare un'inizializzazione dei campi non necessaria

Se possibile, consentire ai campi di eseguire l'inizializzazione automatica come indicato di seguito:

- i riferimenti agli oggetti sono inizializzati su `Null`
- `int`, `byte` o `long` è inizializzato su `0`
- il valore booleano è inizializzato su `false`

Le variabili locali vanno inizializzate esplicitamente in un metodo.

Esempio di codice

```
class BetterExample { private int fieldsCount; private Field _fieldWithFocus; private boolean _validLayout; private boolean _validLayout; }
```

Importazione di classi singole

Una BlackBerry® Java Application che utilizza solo un numero limitato di classi di un pacchetto dovrebbe importare solo classi singole.

Esempio di codice

```
import net.rim.blackberry.api.browser.Browser;
```

Assistenza multilingua

BlackBerry® Integrated Development Environment include un meccanismo per la creazione di risorse stringa. L'API di localizzazione appartiene al pacchetto `net.rim.device.api.i18n`. Le applicazioni MIDP non supportano la localizzazione.

BlackBerry Integrated Development Environment archivia le risorse per una lingua in un oggetto `ResourceBundle`. Un oggetto `ResourceBundleFamily` contiene una raccolta di `ResourceBundles` che raggruppa le risorse per un'applicazione. L'applicazione cambia la lingua, in base alle impostazioni dell'utente del dispositivo BlackBerry, senza richiedere i nuovi bundle di risorse.

È possibile utilizzare BlackBerry Integrated Development Environment per compilare tutti i bundle di risorse in file `.cod` compilati separatamente. È possibile caricare il file `.cod` appropriato nei dispositivi BlackBerry insieme agli altri file `.cod` per l'applicazione.

Le risorse sono organizzate secondo una gerarchia basata sull'eredità. Se una stringa non è definita in una lingua, viene utilizzata una stringa appartenente alla lingua successiva più vicina.

Procedura consigliata: archiviazione delle stringhe di testo nei file di risorse

Anziché utilizzare il testo nel codice sorgente, progettare le applicazioni in modo tale che utilizzino i file di risorse per la localizzazione (adattamento ai linguaggi specifici e alle regioni).

Seguire le seguenti istruzioni:

- Archiviare le stringhe di testo per ogni lingua in un unico file di risorse.
- Nel proprio codice sorgente, avvalersi di identificatori univoci per utilizzare i file di risorse appropriati.
- Progettare l'applicazione affinché recuperi dinamicamente il file di risorse appropriato da visualizzare in base alla lingua dell'utente del dispositivo BlackBerry.

Supporto multimediale

Supporto audio

È possibile creare una BlackBerry® Java Application che funziona con i formati audio supportati da un dispositivo BlackBerry. Il tipo di formato audio supportato dal dispositivo BlackBerry dipende dal numero del modello del dispositivo BlackBerry.

Per ulteriori informazioni sul supporto audio in un dispositivo BlackBerry, visitare www.blackberry.com/developers.

Supporto fotocamera

Su un dispositivo BlackBerry® che include una fotocamera, quando un utente scatta una foto, il dispositivo BlackBerry archivia l'immagine nel file system all'interno del dispositivo. Un'applicazione del dispositivo BlackBerry può accedere alle immagini mediante l'API `FileConnection` per Java ME disponibile nella versione 4.2 o successiva di BlackBerry® Java® Development Environment. L'applicazione del dispositivo BlackBerry può richiamare l'applicazione fotocamera e rilevare gli eventi quando le immagini vengono aggiunte al file system.

Su un dispositivo BlackBerry dotato della versione 4.7 o successiva di BlackBerry® Desktop Software, è possibile consentire ad un utente del dispositivo BlackBerry di inviare un'immagine ad un'applicazione del dispositivo BlackBerry. Se un utente del dispositivo BlackBerry utilizza l'applicazione fotocamera per scattare una foto o per accedere ad un'immagine mediante la funzione di esplora file, BlackBerry Device Software aggiunge una voce di menu all'applicazione fotocamera o visualizza una finestra di dialogo in esplora file. Quando un utente del dispositivo BlackBerry seleziona la voce di menu o la finestra di dialogo, BlackBerry Device Software invia una richiesta a un'applicazione del dispositivo BlackBerry che funge da gestore dei contenuti per l'immagine. L'applicazione del dispositivo BlackBerry può accettare la richiesta ed aprire l'immagine.

Affinché un'applicazione del dispositivo BlackBerry sia in grado di ricevere un'immagine, l'applicazione deve implementare l'interfaccia `javax.microedition.content.ContentHandler` per effettuare la registrazione come gestore dei contenuti per un tipo di immagine come `.jpg`. La registrazione come gestore dei contenuti associa l'applicazione del dispositivo BlackBerry al tipo di immagine e consente ad altre applicazioni del dispositivo BlackBerry di inviare questi file all'applicazione del dispositivo BlackBerry. Per ulteriori informazioni sull'utilizzo dell'interfaccia `ContentHandler`, visitare <http://jcp.org/en/jsr/detail?id=211>.

Supporto video

È possibile creare un'applicazione del dispositivo BlackBerry® in grado di utilizzare un oggetto `javax.microedition.media.Player` e le sottoclassi della classe `javax.microedition.media.Control` per riprodurre un video. Un'applicazione del dispositivo BlackBerry richiama uno dei metodi

`net.rim.device.api.ui.Manager.createPlayer()` per recuperare un oggetto `Letttore` e un video da fonti diverse, ad esempio un file o un flusso. Un oggetto `Letttore` consente a un BlackBerry di utilizzare l'applicazione come un lettore multimediale ed eseguire azioni come riprodurre e interrompere un video.

L'applicazione del dispositivo BlackBerry richiama `Player.realize()` per consentire all'applicazione di recuperare le risorse richieste, come la posizione di un file video, per riprodurre un video. L'applicazione del dispositivo BlackBerry richiama `Player.prefetch()` per consentire all'applicazione di preparare la riproduzione di un video. L'applicazione del dispositivo BlackBerry può richiamare `Player.getControls()` per recuperare un array di oggetti `Controllo` che rappresentano le operazioni o le funzioni supportate dal `Letttore`. Ad esempio, un oggetto

`javax.microedition.media.control.VolumeControl` consente ad un'applicazione del dispositivo BlackBerry di controllare il volume utilizzato per riprodurre un video. Un oggetto `javax.microedition.media.control.VolumeControl` consente ad un'applicazione del dispositivo BlackBerry di controllare le modalità di riproduzione di un video.

Dopo che l'applicazione del dispositivo BlackBerry recupera le informazioni sugli oggetti `javax.microedition.media.Control` supportati, essa può richiamare `Player.getControl()`, utilizzando come parametro una stringa con il nome del controllo, per recuperare un oggetto `Controllo` specifico. Ad esempio, `VideoControlmyVC=(VideoControl) Player.getControl("<VideoControl>")`. L'applicazione del dispositivo BlackBerry può impostare le proprietà dell'oggetto `Controllo` richiamandone i metodi. L'applicazione del dispositivo BlackBerry richiama `Player.start()` per avviare la riproduzione di un video e richiama `Player.close()` per arrestare la riproduzione di un video.

È possibile creare un'applicazione del dispositivo BlackBerry in grado di utilizzare il mirino della fotocamera per registrare un video in un file `.3gp` o per eseguire lo streaming del video in un oggetto `java.io.ByteArrayOutputStream`. L'applicazione del dispositivo BlackBerry richiama `Manager.createPlayer("capture://video")` per recuperare un oggetto `Letttore` che supporta le interfacce `VideoControl` e `RecordControl`.

L'applicazione del dispositivo BlackBerry richiama `Player.getControl("RecordControl")` ed esegue il cast del valore restituito come un oggetto `RecordControl`. Un oggetto `RecordControl` consente ad un'applicazione del dispositivo BlackBerry di controllare le modalità di registrazione del video. Richiamando `RecordControl.setRecordLocation("file:///<filename>")` si imposta la posizione del file `.3gp`. Richiamando `RecordControl.setRecordStream(ByteArrayOutputStreamoutput)` si imposta il `ByteArrayOutputStream` sul quale eseguire lo streaming. L'applicazione del dispositivo BlackBerry richiama `RecordControl.startRecord()` seguito da `Player.start()` per avviare la registrazione di un video. L'applicazione del dispositivo BlackBerry richiama `RecordControl.commit()` seguito da `Player.stop()` per salvare il video e arrestare la registrazione.

È possibile creare un'applicazione del dispositivo BlackBerry in grado di eseguire il video in BlackBerry® Browser. L'applicazione del dispositivo BlackBerry richiama `net.rim.blackberry.api.browser.Browser.getDefaultSession()` per recuperare un oggetto `net.rim.blackberry.api.browser.BrowserSession`. L'oggetto `BrowserSession` consente all'applicazione del dispositivo BlackBerry di interagire con BlackBerry Browser. L'applicazione del dispositivo BlackBerry richiama `BrowserSession.displayPage()` utilizzando come parametro una `String` dal formato "file:///<locationOfVideo>", dove <locationOfVideo> rappresenta la posizione del video. `BrowserSession.displayPage()` visualizza un video in BlackBerry Browser.

È possibile creare un'applicazione del dispositivo BlackBerry in grado di eseguire un video che richiede licenze di gestione dei diritti digitali. L'applicazione del dispositivo BlackBerry utilizza la classe `net.rim.device.api.drm.DRMManager` per installare le licenze DRM per un video. Un dispositivo BlackBerry ha un oggetto `DRMManager`. Richiamando `DRMManager.getLicenseChallenge(String contentURL, String[] permissions)` si recupera un array di byte contenente informazioni sulle licenze DRM che l'applicazione del dispositivo BlackBerry deve ottenere ed installare per eseguire un video. Richiamando `DRMManager.installRights(String contentURL, byte[] rightsData, DRMRightsInstallCallback callback)` si installano le licenze per un video.

È possibile consentire ad un utente del dispositivo BlackBerry di inviare un video ad un'applicazione del dispositivo BlackBerry. Affinché l'applicazione del dispositivo BlackBerry riceva ed utilizzi il file video, l'applicazione deve implementare l'interfaccia `javax.microedition.content.ContentHandler` per effettuare la registrazione come gestore dei contenuti. Quando un utente del dispositivo BlackBerry accede a un file video mediante la fotocamera o la funzione esplora file, viene visualizzata una voce di menu che consente all'utente del dispositivo BlackBerry di inviare il file video alle applicazioni del dispositivo BlackBerry che sono gestori dei contenuti.

Progettazione navigazione e interfaccia utente

3

Modalità di input e navigazione per l'utente del dispositivo BlackBerry

Per le attività di input e navigazione i dispositivi BlackBerry® includono una tastiera, una rotellina o una trackball e un tasto Esc.

Una BlackBerry® Java® Application dovrebbe utilizzare il modello di input e navigazione seguente il più attentamente possibile.

- Facendo clic con la rotellina o con la trackball si richiama generalmente un menu.
- Con il tasto ESC è possibile annullare azioni o tornare alla schermata precedente. Premendo il tasto ESC varie volte è possibile tornare alla schermata iniziale. Tenendo premuto il tasto ESC è possibile chiudere il browser o l'applicazione multimediale.

Per impostazione predefinita, gli oggetti dello schermo BlackBerry forniscono questa funzionalità senza alcuna personalizzazione; tuttavia, è necessario aggiungere alcune voci di menu e logica aggiuntiva dell'interfaccia utente e di navigazione.

Rotellina e trackball a confronto

Sensibilità della trackball

La sensibilità della trackball fa riferimento alla quantità di movimento necessaria affinché il sistema identifichi il movimento come un evento di navigazione e lo invii allo strato software. L'hardware del dispositivo BlackBerry® misura il movimento fisico della trackball mediante unità chiamate tick. Quando il numero di tick lungo un asse supera la soglia del sistema o di una BlackBerry® Java® Application, un evento di navigazione lungo tale asse viene inviato allo strato software e il sistema azzeri il conteggio di tick. Inoltre, i conteggi dei tick vengono azzerati dopo un determinato periodo di inattività.

È possibile utilizzare l'API della trackball per impostarne la sensibilità. Una sensibilità elevata corrisponde ad una soglia di tick inferiore, ciò significa che piccoli movimenti della trackball avviano eventi di navigazione. Viceversa, una sensibilità ridotta corrisponde ad una soglia di tick superiore, ciò significa che per generare eventi di navigazione sono necessari movimenti della trackball più grandi.

Movimento della trackball

È possibile utilizzare l'API della trackball per filtrare i dati relativi al movimento della trackball che l'hardware del dispositivo BlackBerry® invia allo strato software. L'API della trackball può filtrare il movimento "rumore" o i movimenti superflui.

È possibile utilizzare l'API della trackball anche per modificare le impostazioni, ad esempio l'accelerazione del movimento della trackball. Se si aumenta l'impostazione relativa all'accelerazione del movimento della trackball, lo strato software potrebbe considerare la velocità della trackball superiore a quella rilevata dall'hardware del dispositivo BlackBerry, finché l'utente ruota ininterrottamente la trackball. La sensibilità della trackball aumenta temporaneamente quando l'utente ruota la trackball senza pause.

Rotellina

I dispositivi BlackBerry® precedenti ai modelli BlackBerry® Pearl™ 8100 Series non includono una trackball. Presentano invece una rotellina sul lato destro del dispositivo. La rotellina rappresenta il comando principale per la navigazione dell'utente.

Gli utenti possono

- ruotare la rotellina per spostare il cursore verticalmente
- ruotare la rotellina tenendo premuto il tasto Alt per spostare il cursore orizzontalmente
- fare clic sulla rotellina per selezionare gli oggetti o aprire il menu

Istruzioni

- Esercitarsi con i modelli di navigazione mediante rotellina e trackball ed accertarsi che la propria applicazione funzioni correttamente con entrambi.

Creazione di un'interfaccia utente compatibile con le interfacce utenti BlackBerry standard

È possibile utilizzare le API MIDP standard e le API dell'interfaccia utente BlackBerry® per creare le interfacce utenti di BlackBerry® Java® Application.

Le API dell'interfaccia utente BlackBerry rappresentano una libreria di componenti dell'interfaccia utente progettati per fornire layout e comportamenti predefiniti compatibili con le applicazioni di BlackBerry® Device Software.

- I componenti del campo offrono elementi UI standard per selezione data, opzioni, caselle di selezione, elenchi, campi di testo, etichette e comandi della barra di avanzamento.
- I gestori di layout forniscono un'applicazione in grado di disporre i componenti nello schermo di un dispositivo BlackBerry secondo la modalità standard, ad esempio orizzontalmente, verticalmente o in modalità di scorrimento da sinistra a destra.

È possibile utilizzare le API dell'interfaccia utente BlackBerry per creare interfacce utenti che includono tabelle, griglie ed altre funzionalità speciali. BlackBerry® Java® Development Environment utilizza un modello degli eventi Java standard per ricevere e rispondere a tipi di eventi specifici. Le applicazioni possono ricevere e rispondere agli eventi dell'utente del dispositivo BlackBerry, come quando l'utente fa clic sulla rotellina o sulla trackball o quando digita sulla tastiera. Lo stesso vale per gli eventi di sistema, come gli avvisi globali, i cambiamenti dell'orologio in tempo reale e le connessioni mediante la porta USB.

Gestione della memoria

4

Gestione della memoria

BlackBerry® Java® Virtual Machine gestisce l'utilizzo della memoria sul dispositivo BlackBerry. BlackBerry JVM assegna la memoria, esegue le garbage collection e sposta automaticamente i dati tra la SRAM e la memoria flash. BlackBerry JVM deve anche dividere la memoria disponibile tra le applicazioni del dispositivo BlackBerry e BlackBerry® Java Application. La capacità della memoria rappresenta l'importo totale di memoria disponibile, superiore alla memoria di lavoro disponibile quando tutte le applicazioni e i dati delle applicazioni associati sono presenti sul dispositivo BlackBerry.

Memoria del dispositivo BlackBerry

I dispositivi BlackBerry® includono i tipi seguenti di memoria:

Memoria	Descrizione
flash	Il sistema operativo BlackBerry e tutti i moduli applicazioni sono archiviati nella memoria flash. Quando un utente del dispositivo BlackBerry attiva il dispositivo BlackBerry, il sistema operativo principale e i moduli BlackBerry® Java Application utilizzano da 10 MB a 15 MB di memoria flash, in base alla versione. La memoria flash può archiviare i messaggi di posta elettronica dell'utente del dispositivo BlackBerry, i dati dell'agenda elettronica e altre informazioni personali, come pure i dati che una BlackBerry Java Application archivia nella memoria.
SRAM	La SRAM controlla gli oggetti dati transitori e i processi runtime.
Scheda di memoria espandibile microSD	La scheda microSD archivia file multimediali, documenti e dati da una BlackBerry Java Application.

Risorse principali da riservare

- Memoria flash: lo spazio per l'archiviazione permanente disponibile sul dispositivo BlackBerry® è una quantità prestabilita di memoria flash, in genere compresa tra 8 MB a 64 MB.
- Handle di oggetti permanenti: gli handle assegnati ad ogni oggetto permanente sono utilizzati solo dagli oggetti permanenti. La quantità di memoria flash presente sul dispositivo BlackBerry determina il numero corretto di handle di oggetti permanenti nel sistema.
- Handle di oggetto: ogni oggetto e ogni array di primitive ha un handle di oggetto a cui è associato. La quantità di memoria flash sul dispositivo BlackBerry determina il numero corretto di handle di oggetto nel sistema.

Procedura consigliata: riduzione della memoria utilizzata

Per ridurre la memoria runtime, seguire le linee guida seguenti:

- Utilizzare i tipi primitivi (come `int` o `Boolean`) invece degli oggetti (come `String` o `Integer`).
- Non dipendere completamente dal Garbage Collector.
- Evitare di creare molti oggetti rapidamente.
- Impostare i riferimenti all'oggetto su `Null` quando si è terminato di utilizzarli.
- Riutilizzare gli oggetti il più possibile.
- Spostare nel server le attività di elaborazione complessa. Ad esempio, è possibile filtrare o ordinare i dati prima di inviarli al dispositivo BlackBerry®.

Gestione della condizione di memoria in esaurimento

Il gestore di memoria in esaurimento gestisce le risorse di memoria del dispositivo BlackBerry® quando esse scendono al di sotto di una determinata soglia. Il gestore di memoria in esaurimento tenta di liberare memoria in uso al fine di aumentare la memoria disponibile sul dispositivo BlackBerry. Tutte le applicazioni, comprese le BlackBerry® Java Application, dovrebbero collaborare con il gestore di memoria in esaurimento per liberare quanta memoria possibile quando le risorse del dispositivo BlackBerry sono in esaurimento.

Individuazione della condizione di memoria in esaurimento su un dispositivo BlackBerry

Le condizioni seguenti possono indurre il gestore di memoria in esaurimento a tentare di liberare le risorse di memoria:

- La quantità di memoria flash disponibile nel dispositivo BlackBerry® scende al di sotto di una determinata soglia. La soglia della memoria flash dipende dalla quantità di RAM libera nel sistema. La soglia della memoria flash varia da 400 KB e 800 KB.
- Il numero di handle di oggetti permanenti che sono disponibili nel dispositivo BlackBerry scende al di sotto dei 1000 handle di oggetti permanenti.
- Il numero di handle di oggetti che sono disponibili nel dispositivo BlackBerry scende al di sotto dei 1000 handle di oggetti.

Risparmio risorse

Procedura consigliata: selezione efficiente della struttura dati

La selezione della struttura dati definisce quanti oggetti vengono gestiti da una BlackBerry® Java Application e quanta memoria flash essa utilizza. Una selezione errata della struttura dati può causare il consumo delle risorse principali senza migliorare il funzionamento di BlackBerry Java Application o l'esperienza dell'utente del dispositivo BlackBerry.

Seguire le seguenti istruzioni:

- La struttura dati dovrebbe comprendere il numero minimo possibile di oggetti, soprattutto quando si utilizzano oggetti di alto livello come un `Vector` o un `HashTable`. Queste classi forniscono funzionalità rilevanti ma non rappresentano meccanismi di archiviazione efficienti, pertanto si dovrebbe evitare di utilizzarli nell'archivio permanente.
- Se possibile, utilizzare le primitive anziché gli oggetti, poiché esse riducono il numero di handle di oggetti in uso sul dispositivo BlackBerry. Un array di primitive è un oggetto e utilizza un handle di oggetto.
- Gli oggetti `String` sono efficienti come gli array di byte. Un oggetto `String` utilizza solo un handle di oggetto ed è equivalente se la propria applicazione archivia tutti i caratteri come un byte. In altre parole, il valore di ogni carattere è uguale o inferiore al valore decimale di 255. Se la propria applicazione non è in grado di archiviare i caratteri come un byte, è possibile archivarli come una `String` poiché questa operazione equivale all'archiviazione di una matrice di caratteri.

Procedura consigliata: consolidazione degli oggetti in gruppi di oggetti

Uno degli errori più comuni segnalato dagli sviluppatori delle applicazioni è un esaurimento degli handle di oggetti permanenti. La quantità di memoria flash sul dispositivo BlackBerry® determina il numero corretto di handle di oggetti permanenti disponibili nel sistema. In base alla selezione della struttura dei dati, i record archiviati possono esaurire rapidamente il numero di handle di oggetti permanenti. Un oggetto permanente utilizza un handle di oggetti permanenti e un handle di oggetto. Un oggetto transitorio utilizza solo un handle di oggetto.

Ad esempio, un record contenente dieci campi `String` che rappresentano voci come un nome, un numero di telefono e un indirizzo, utilizza 11 handle di oggetti permanenti, uno per l'oggetto record ed uno per ogni `String`. Se una BlackBerry® Java Application presenta 3000 record, l'applicazione utilizza 33.000 handle di oggetti permanenti, superando il numero di handle di oggetti permanenti disponibili su un dispositivo BlackBerry con una memoria flash da 16 MB.

È possibile utilizzare la classe `net.rim.device.api.system.ObjectGroup` per consolidare gli handle di oggetto per un oggetto in un gruppo. Utilizzando l'esempio contenuto nel paragrafo precedente, se si raggruppa il record, esso utilizza un handle di oggetti permanenti anziché 11. L'oggetto gestisce i campi `String` da consolidare all'interno dell'handle di oggetti record.

Quando si consolidano gli handle di oggetto in un gruppo, l'handle di oggetto è di sola lettura. Prima di poter modificare l'oggetto, è necessario separarlo dal gruppo. Dopo aver apportato le modifiche, raggruppare di nuovo l'oggetto. Se si tenta di modificare un oggetto raggruppato senza prima separarlo dal gruppo, viene eseguito un `ObjectGroupReadOnlyException`.

La separazione di un oggetto da un gruppo ha un impatto sulle prestazioni. Il sistema crea una copia dell'oggetto raggruppato e assegna gli handle a ciascuno degli oggetti contenuti nel gruppo. Pertanto gli oggetti dovrebbero essere separati dal gruppo solo quando necessario.

Garbage collection su un dispositivo BlackBerry

Garbage collection della RAM in un dispositivo BlackBerry

BlackBerry® Java® Virtual Machine avvia un'operazione di garbage collection della RAM solo quando BlackBerry JVM non è in grado di allocare un oggetto a causa della mancanza di spazio nella RAM. Per eseguire l'operazione di garbage collection della RAM sono necessari in genere dai 500 ai 600 millisecondi. L'operazione di garbage collection rimuove tutte le variabili allocate di recente per le quali non esistono più riferimenti nella RAM. Per assicurarsi che la mancanza di riferimenti all'interno della RAM sia una condizione sufficiente per rimuovere l'oggetto, un'operazione di garbage collection della RAM può essere eseguita solo quando gli oggetti non sono stati estratti nella memoria flash.

Garbage collection completa su un dispositivo BlackBerry

L'operazione di garbage collection completa dura in media 1 secondo e dovrebbe essere completata in meno di 2 secondi. La garbage collection completa esegue le azioni seguenti:

- Esegue un'operazione di garbage collection della RAM.
- Contrassegna gli oggetti nella memoria flash per i quali non esiste più un riferimento o che non sono più archiviati.
- Rilascia gli handle di oggetti non permanenti nella RAM e nella memoria flash.

Il sistema potrebbe avviare una garbage collection completa nelle situazioni seguenti:

- BlackBerry® Java® Virtual Machine non è in grado di assegnare un oggetto a causa di una mancanza di spazio nella RAM.
- Un processo sta per superare le dimensioni heap attualmente assegnate.
- BlackBerry JVM non è in grado di assegnare un nuovo oggetto perché gli handle di oggetto non sono disponibili.
- Il dispositivo BlackBerry è inattivo.

Garbage collection su un dispositivo BlackBerry BlackBerry

La garbage collection non avviene ogni volta che il dispositivo BlackBerry® è inattivo. Accade solo quando il sistema considera vantaggioso eseguire un'operazione di garbage collection per ottimizzare le prestazioni del sistema e aumentare le prestazioni della batteria.

Per migliorare le prestazioni senza compromettere l'esperienza dell'utente del dispositivo BlackBerry, il sistema tenta di eseguire le operazioni di garbage collection seguenti quando il dispositivo BlackBerry è inattivo:

- Una garbage collection completa può verificarsi quando il dispositivo BlackBerry è inattivo per un periodo di tempo relativamente breve.
- Una garbage collection completa può verificarsi quando il dispositivo BlackBerry è inattivo per un periodo di tempo significativo.

Gestione dei dati

5

Il dispositivo BlackBerry® fornisce API per archiviare i dati nella memoria permanente del dispositivo BlackBerry. Le API di memorizzazione permanente BlackBerry e le API MIDP RMS (supporto per JSR 37 e JSR 118) sono disponibili in tutti i dispositivi BlackBerry basati su Java®. Un dispositivo BlackBerry che esegue la versione 4.2 o successiva di BlackBerry® Device Software fornisce un file system tradizionale e il supporto per l'archiviazione del contenuto direttamente nel file system utilizzando le API JSR 75. Sia mediante le API di memorizzazione permanente BlackBerry sia mediante le API MIDP RMS, è possibile archiviare i dati in maniera permanente nella memoria flash. I dati rimangono archiviati anche se si rimuove la batteria dal dispositivo BlackBerry.

Supporto API per l'archiviazione di dati nella memoria permanente

API di memorizzazione permanente

Le API BlackBerry® di memorizzazione permanente sono progettate per fornire un'interfaccia di archiviazione dei dati flessibile e robusta. Grazie alle API BlackBerry di memorizzazione permanente, è possibile salvare interi oggetti Java® nella memoria senza dover serializzare prima i dati. Quando si avvia l'applicazione, è possibile recuperare l'oggetto Java dalla memoria ed elaborare le informazioni.

Le API BlackBerry di memorizzazione permanente non forniscono un modello di database relazionale. È necessario creare un modello a oggetti efficace e gestire le relazioni tra gli oggetti, se necessario, mediante gli indici e le tabelle hash.

API del sistema di gestione dei record MIDP

Le API RMS forniscono un semplice sistema di gestione dei record che consente di creare un oggetto di archivio dati e memorizzare una serie di record all'interno di tale oggetto. Ogni record è un array di byte, pertanto è necessario innanzitutto serializzare i dati in un formato array di byte prima di archivarli localmente. Le API RMS non forniscono alcuna indicizzazione integrata o relazioni tra i record. Le dimensioni massime per un singolo archivio dati RMS sono 64 KB. Un'applicazione può creare archivi dati RMS multipli per memorizzare maggiori quantità di dati. Le API RMS appartengono alla descrizione MIDP standard, pertanto tutti i dispositivi in grado di supportare MIDP possono supportare anche le API RMS.

API FileConnection

Le API FileConnection forniscono un file system tradizionale e un supporto per archiviare i dati direttamente sul file system nel dispositivo BlackBerry® o in una scheda microSD. È possibile visualizzare i dati nel file system e spostarli in un computer tramite Windows®.

Archiviazione su supporti rimovibili

Accesso ai dati sulla scheda di memoria microSD

Il pacchetto `javax.microedition.io.file` supporta le API `FileConnection` JSR 75 e viene utilizzato nelle applicazioni per l'accesso al file system della scheda di memoria microSD. Inoltre, è possibile implementare l'interfaccia `FileConnection` per accedere alle suonerie del dispositivo BlackBerry® e alle immagini della fotocamera.

Classe o interfaccia	Descrizione
<code>ConnectionClosedException</code>	Questa eccezione viene eseguita quando un'applicazione richiama un metodo su una connessione file chiusa.
<code>FileConnection</code>	Un'applicazione può utilizzare questa API per accedere ai file o alle directory.
<code>FileSystemListener</code>	Un'applicazione può utilizzare questa API per ricevere notifiche di stato quando l'applicazione aggiunge o rimuove un file system principale.
<code>FileSystemRegistry</code>	Un'applicazione può utilizzare questa API come un registro centrale per i rilevatori di file system che individuano file system aggiunti o rimossi.
<code>IllegalModeException</code>	Questa eccezione viene eseguita quando un metodo richiede una modalità di protezione specifica (ad esempio, di LETTURA o di SCRITTURA) e la connessione aperta non si trova in tale modalità.

Backup e sincronizzazione dei dati

BlackBerry® Desktop Manager fornisce uno strumento di backup e ripristino che un utente del dispositivo BlackBerry può utilizzare per salvare i dati del dispositivo BlackBerry su un file del computer e per ripristinare dati nel dispositivo BlackBerry.

Quando un'applicazione utilizza l'API di sincronizzazione, BlackBerry Desktop Manager può eseguire il backup e ripristinare il database dell'applicazione contemporaneamente ad altri database del dispositivo BlackBerry. È possibile utilizzare l'API di sincronizzazione per creare archivi di dati o per inserire i dati nei database dell'applicazione la prima volta che il dispositivo BlackBerry si connette al computer dell'utente del dispositivo BlackBerry.

Per sincronizzare i dati nelle origini dati remote, è necessario creare la logica di sincronizzazione nella propria BlackBerry Java® Application. La maggior parte delle applicazioni invia i dati a un'applicazione sul lato server mediante i protocolli standard HTTP o TCP/IP sulla rete wireless e attraverso Internet o l'intranet aziendale. È possibile utilizzare API XML per generare ed analizzare i dati formattati in XML da inviare e ricevere sulla rete wireless. Tuttavia, le applicazioni sul lato client e sul lato server devono essere in grado di leggere e scrivere correttamente i dati e riconoscere il completamento della trasmissione.

Una BlackBerry Java Application può connettersi ad un'applicazione basata su computer per inviare i dati tramite una connessione USB utilizzando le API di sincronizzazione desktop BlackBerry e BlackBerry Desktop Manager. In questo caso, è necessario creare un'applicazione per Windows® che sia in grado di leggere i dati dal client mediante un'attività aggiuntiva per BlackBerry Desktop Manager. L'utente del dispositivo BlackBerry deve eseguire manualmente la sincronizzazione eseguendo il componente aggiuntivo BlackBerry Desktop Manager che chiede all'applicazione sul dispositivo BlackBerry di inviare i dati all'applicazione del computer. Inoltre, è possibile scrivere dati nell'applicazione del computer mediante i protocolli USB nativi.

Trasporto di dati wireless

6

Gateway wireless

Le applicazioni Java® per i dispositivi BlackBerry® possono utilizzare protocolli socket standard HTTP, HTTPS e TCP per stabilire connessioni sulla rete wireless. Quando un'applicazione stabilisce una connessione sulla rete wireless, può utilizzare uno dei due gateway wireless per creare una connessione proxy a Internet o all'intranet aziendale. È possibile progettare la propria applicazione basandola sul gateway predefinito disponibile per l'utente del dispositivo BlackBerry oppure personalizzare il proprio codice per selezionare un gateway preferito. Progettare la propria applicazione per selezionare esplicitamente il gateway preferito per la connessione ed utilizzare il gateway predefinito se il metodo preferito non è disponibile. Ciò potrebbe ridurre il numero dei problemi relativi alla connessione di rete segnalati dagli utenti e consentire all'applicazione di utilizzare un modello di connettività uniforme in tutti i tipi di rete e operatori wireless.

Utilizzo di BlackBerry Enterprise Server come gateway di rete

Quando si utilizza BlackBerry® Enterprise Server come gateway di rete, tutto il traffico tra la propria applicazione e BlackBerry Enterprise Server è crittografato mediante la crittografia AES o Triple DES. Poiché BlackBerry Enterprise Server è protetto dal firewall dell'azienda e fornisce una crittografia dati integrata, le applicazioni possono comunicare con i server Web e i server delle applicazioni che si trovano sull'intranet dell'azienda. Il componente BlackBerry® Mobile Data System di BlackBerry Enterprise Server comprende BlackBerry MDS Services, che fornisce un servizio proxy HTTP e TCP/IP per consentire alla BlackBerry® Java® Application di utilizzarlo come gateway protetto per gestire le connessioni HTTP e TCP/IP all'intranet.

Se la propria applicazione si connette a Internet, è possibile utilizzare BlackBerry Enterprise Server come gateway. Protette firewall dell'azienda, le richieste di rete vengono indirizzate a BlackBerry Enterprise Server che esegue la richiesta alla rete Internet attraverso il firewall aziendale. Gli amministratori possono impostare un criterio IT per garantire che BlackBerry Enterprise Server rappresenti il gateway per tutto il traffico della rete wireless, compreso il traffico destinato a Internet.

Se la propria applicazione si connette a Internet, è possibile utilizzare BlackBerry® Internet Service o il gateway Internet del provider del server wireless per gestire le connessioni.

Utilizzo del gateway Internet del provider di servizi wireless

La maggior parte dei provider di servizi wireless fornisce un gateway Internet in grado di offrire connettività TCP/IP diretta a Internet. Alcuni forniscono anche un gateway WAP che consente di stabilire connessioni HTTP su protocollo WAP. Una BlackBerry® Java® Application può utilizzare questi gateway per stabilire una connessione a Internet. Se la propria applicazione è destinata ad utenti del dispositivo BlackBerry che si trovano su una rete wireless specifica, l'utilizzo di gateway Internet del provider di servizi wireless può spesso produrre buoni risultati. Se la propria applicazione è destinata ad utenti del dispositivo

BlackBerry che si trovano su diverse reti wireless, può essere difficile verificare la propria applicazione mediante gateway Internet diversi e garantire un'esperienza affidabile. Potrebbe essere utile utilizzare BlackBerry® Internet Service e il gateway Internet del provider di servizi wireless come tipo di connessione predefinito se BlackBerry Internet Service non è disponibile.

Opzioni alternative di trasporto dati

Utilizzo della posta elettronica per il trasporto di dati

È possibile utilizzare le API BlackBerry® per creare una BlackBerry Java® Application che usa la posta elettronica come meccanismo di trasporto per i dati da inviare e ricevere. La posta elettronica può rappresentare una modalità efficace di distribuire proattivamente il contenuto agli utenti del dispositivo BlackBerry se i modelli push tradizionali non sono disponibili. Una BlackBerry Java Application può utilizzare le API BlackBerry per inviare messaggi di posta elettronica e per rilevare messaggi di posta elettronica in entrata. Una BlackBerry Java Application può anche accedere ai dettagli e alle intestazioni dei messaggi di posta elettronica archiviati localmente sul dispositivo BlackBerry e registrare i rilevatori per i cambiamenti nello stato di un messaggio di posta elettronica.

Utilizzo di SMS per il trasporto di dati

È possibile utilizzare le API BlackBerry® per creare applicazioni che si avvalgono degli SMS come metodo di trasporto per l'invio e la ricezione di dati. Le API BlackBerry consentono ad un'applicazione di inviare messaggi SMS e di rilevare messaggi SMS in entrata.

Utilizzo della messaggistica PIN per il trasporto di dati

La messaggistica PIN utilizza il canale dati anziché il canale vocale e consente selezionare il dispositivo BlackBerry® di destinazione in base al suo numero PIN univoco. La messaggistica PIN può essere utilizzata solo per l'invio di dati da un dispositivo BlackBerry a un altro. Inoltre, le API BlackBerry possono consentire a un'applicazione di inviare e ricevere messaggi PIN BlackBerry in maniera programmata. La messaggistica PIN può rappresentare una modalità efficace per implementare le applicazioni PIN destinate esclusivamente agli utenti del dispositivo BlackBerry.

Integrazione delle applicazioni BlackBerry

7

Aggiunta di voci di menu personalizzate

Una BlackBerry® Java® Application può aggiungere voci di menu personalizzate al menu del dispositivo BlackBerry per la posta elettronica, l'agenda elettronica e le applicazioni del telefono. Quando un utente del dispositivo BlackBerry seleziona la voce di menu personalizzata, BlackBerry Java Application viene avviata con un riferimento all'oggetto selezionato dall'utente del dispositivo BlackBerry. Ad esempio, una BlackBerry Java Application può aggiungere una voce di menu chiamata Visualizza posizione del mittente all'applicazione di posta elettronica. Quando l'utente del dispositivo BlackBerry seleziona la voce di menu, BlackBerry Java Application viene avviata con un riferimento all'oggetto di posta elettronica che è attualmente evidenziato o che viene aperto dall'utente. BlackBerry Java Application utilizza l'indirizzo di posta elettronica del mittente per determinarne la posizione recuperando l'indirizzo di posta elettronica dall'elenco contatti o i dati da un server remoto, quindi compare in primo piano visualizzando una mappa.

Integrazione con le applicazioni BlackBerry Device Software

Per eseguire un'azione o per visualizzare alcune informazioni, un'applicazione del dispositivo BlackBerry® può richiamare un'applicazione BlackBerry® Device Software come la posta elettronica, l'agenda elettronica, il telefono, il browser e le applicazioni della fotocamera. Gli esempi seguenti dimostrano come un'applicazione del dispositivo BlackBerry può richiamare un'applicazione BlackBerry Device Software:

- L'applicazione del dispositivo BlackBerry richiama il calendario per visualizzare una data o una voce del calendario specifiche.
- L'applicazione del dispositivo BlackBerry richiama la rubrica per visualizzare un contatto specifico.
- L'applicazione del dispositivo BlackBerry richiama il browser per aprire un indirizzo Web specifico.
- L'applicazione del dispositivo BlackBerry richiama il telefono per comporre un numero specifico.

Accesso ai dati della posta elettronica e dell'agenda elettronica

Una BlackBerry® Java® Application può utilizzare la API BlackBerry per accedere ai dettagli dei messaggi di posta elettronica, dei contatti, degli eventi del calendario, delle attività e dei registri delle chiamate memorizzati dal dispositivo BlackBerry. Una BlackBerry Java Application può leggere e aggiornare le informazioni e creare nuove voci.

Utilizzo di BlackBerry Messenger con una BlackBerry Application

È possibile integrare una BlackBerry® Java® Application con un'applicazione BlackBerry® Messenger. Un'operazione di questo tipo può essere utile se si crea un'applicazione, ad esempio un gioco basato su turni per il dispositivo BlackBerry.

Per creare una BlackBerry Java Application che si integra con un'applicazione di BlackBerry Messenger, è possibile utilizzare le classi contenute nel pacchetto `net.rim.blackberry.api.blackberrymessenger`. Per ulteriori informazioni sulle modalità di utilizzo della classe `BlackBerryMessenger`, consultare i materiali di riferimento sulle API BlackBerry.

Utilizzo dei rilevatori per rispondere alle modifiche delle applicazioni

Una BlackBerry® Java® Application può registrare i rilevatori delle modifiche sugli archivi dei dati della posta elettronica e dell'agenda elettronica e nell'applicazione del telefono. I rilevatori consentono alla BlackBerry Java Application di intervenire immediatamente quando l'utente del dispositivo BlackBerry esegue un evento locale. È possibile utilizzare i rilevatori dei dati della posta elettronica e dell'agenda elettronica per inviare una notifica a una BlackBerry Java Application quando giungono nuove voci o quando l'utente del dispositivo BlackBerry apporta dei cambiamenti, ad esempio aggiunge, elimina o aggiorna i dati esistenti. È possibile utilizzare i rilevatori del telefono per individuare le azioni relative alle chiamate, come l'avvio o la conclusione.

Considerazioni sulla protezione

8

Crittografia dei dati e BlackBerry Application

Crittografia dei dati nel trasporto

Se si utilizza BlackBerry® Enterprise Server come gateway di rete per la propria applicazione, BlackBerry Enterprise Server crittografa i dati mediante la crittografia AES o TripleDES in tutti i punti della connessione tra il dispositivo BlackBerry e BlackBerry Enterprise Server protetti dal firewall dell'azienda. Se si richiede che i dati vengano crittografati oltre BlackBerry Enterprise Server e il server di destinazione, è possibile utilizzare il protocollo HTTPS e la crittografia SSL/TLS.

Se la propria applicazione utilizza BlackBerry® Internet Service o il gateway di Internet del provider di servizi wireless, il traffico dati non è crittografato. Se gli utenti del dispositivo BlackBerry lo preferiscono, è possibile utilizzare HTTPS per crittografare i dati o le API Java® per applicare la propria crittografia a chiave simmetrica o a chiave pubblica.

Crittografia dei dati sul dispositivo BlackBerry

Gli amministratori possono impostare un criterio IT per garantire che tutti i dati utente del dispositivo BlackBerry® archiviati nelle applicazioni siano crittografati localmente nella memoria flash. È possibile creare una BlackBerry Java® Application che utilizza le API per registrare i dati in modo che il servizio di crittografia crittografi i dati con la stessa chiave di protezione prima di archivarli nella memoria flash.

Accesso alla memoria

BlackBerry® Java® Development Environment è progettato per impedire alle applicazioni di causare inavvertitamente o in maniera intenzionale problemi nelle altre applicazioni o sul dispositivo BlackBerry. Le applicazioni BlackBerry possono scrivere solo sulla memoria del dispositivo BlackBerry utilizzata da BlackBerry® Java® Virtual Machine; esse non possono accedere alla memoria virtuale o permanente di altre applicazioni (a meno che non ricevano esplicitamente l'autorizzazione a farlo). Una BlackBerry® Java Application può accedere solo all'archiviazione permanente o ai dati utente oppure comunicare con le altre applicazioni mediante API BlackBerry specifiche. Research In Motion deve firmare digitalmente una BlackBerry Java Application che utilizza queste API BlackBerry, al fine di fornire un audit trail delle applicazioni che si avvalgono di API sensibili.

Autenticazione

Criterio IT e autenticazione dei dispositivi BlackBerry

Gli utenti del dispositivo BlackBerry® possono impostare una password per i propri dispositivi. Quando la password del dispositivo è attiva, gli utenti del dispositivo BlackBerry devono fornire la password per accedere ai dati e alle applicazioni. Utilizzare la password del dispositivo è una buona abitudine per limitare l'accesso alla BlackBerry Java® Application sul dispositivo BlackBerry.

Gli amministratori possono utilizzare i criteri IT forniti in BlackBerry® Enterprise Server per verificare i dispositivi BlackBerry dell'azienda siano protetti da password. Inoltre, gli amministratori possono utilizzare i criteri IT in modalità remota per bloccare un dispositivo BlackBerry, modificare la password o rimuovere tutti i dati.

Autenticazione delle applicazioni

Per le applicazioni in cui le funzioni di protezione sono critiche, si potrebbe voler fornire una schermata di accesso che richiede all'utente del dispositivo BlackBerry® di effettuare l'accesso all'applicazione prima di utilizzarla. Le classi delle interfacce utenti forniscono semplici campi della password che nascondono le immissioni di testo mediante asterischi. Le schermate di accesso potrebbero avere un impatto negativo sull'esperienza dell'utente del dispositivo BlackBerry e se l'utente imposta una password per proteggere il dispositivo, l'applicazione potrebbe non presentare una schermata di accesso.

Autenticazione sul lato server

Se la propria applicazione si connette ad un'applicazione su un server oppure a Internet o ad un'intranet, si potrebbe voler includere funzionalità di autenticazione aggiuntive quando gli utenti del dispositivo BlackBerry® effettuano l'accesso al server. La maggior parte delle applicazioni che richiedono l'autenticazione utente si basano sull'autenticazione HTTP Basic che utilizza una semplice combinazione di nome utente e password. È possibile utilizzare l'autenticazione HTTP Basic aggiungendo le intestazioni HTTP corrette quando si apre la connessione HTTP. Inoltre, è possibile aggiungere un'autenticazione più avanzata utilizzando i certificati; tuttavia, la maggior parte delle applicazioni non la richiede.

API controllate e firma codice

Research In Motion tiene traccia dell'utilizzo di API sensibili per ragioni di sicurezza e di controllo delle esportazioni. Nella documentazione di riferimento sulle API per BlackBerry® Java® Development Environment, RIM identifica una classe o un metodo controllati mediante un'icona di blocco o una nota firmata. Per utilizzare le classi o i metodi controllati nelle proprie applicazioni e prima di poter installare i file .cod dell'applicazione sul dispositivo BlackBerry, è necessario firmare l'applicazione mediante una chiave o una firma proveniente da RIM. Altre funzionalità, ad esempio la capacità di esecuzione all'avvio dell'applicazione, potrebbero richiedere la firma delle applicazioni.

Mentre il processo di registrazione RIM copre l'utilizzo della maggior parte delle API controllate, alcune classi di crittografia collegate alla crittografia delle chiavi pubbliche e private contengono la tecnologia fornita da Certicom™. Per utilizzare queste classi, è necessario effettuare la registrazione ed ottenere una licenza direttamente da Certicom. Il processo di registrazione RIM non include l'utilizzo di classi Certicom.

Per verificare ed eseguire il debug del proprio codice prima di ricevere le firme codice, è possibile utilizzare BlackBerry Smartphone Simulator. È necessario firmare l'applicazione prima di installarla sui dispositivi BlackBerry. Il codice effettivo non viene inviato a RIM. È possibile utilizzare BlackBerry® Signature Tool per inviare un hash SHA-1 del proprio file di codice in modo che il sistema di autorizzazione firma possa generare la firma necessaria.

Per ulteriori informazioni su come effettuare la registrazione ed ottenere le firme codice, vedere *Guida per l'amministratore di BlackBerry Signing Authority Tool basata su password*. Per ulteriori informazioni sulla registrazione e l'utilizzo delle classi, visitare www.blackberry.com/developers.

API BlackBerry con accesso controllato

Con BlackBerry® Smartphone Simulator è possibile eseguire applicazioni che utilizzano API controllate senza firme codice; tuttavia, è necessario ottenere tali firme da Research In Motion prima di poter installare tali applicazioni sui dispositivi BlackBerry.

È possibile utilizzare le categorie seguenti di API controllate RIM:

- API runtime
- API BlackBerry® Application
- API crittografica BlackBerry

Se si utilizza uno dei pacchetti API BlackBerry seguenti, l'applicazione richiede le firme codice prima di poterla installare su un dispositivo BlackBerry:

- `net.rim.blackberry.api.browser`
- `net.rim.blackberry.api.invoke`
- `net.rim.blackberry.api.mail`
- `net.rim.blackberry.api.mail`
- `net.rim.blackberry.api.menuitem`
- `net.rim.blackberry.api.options`
- `net.rim.blackberry.api.pdap`
- `net.rim.blackberry.api.phone`
- `net.rim.blackberry.api.phone.phonelogs`
- `net.rim.device.api.browser.field`
- `net.rim.device.api.browser.plugin`
- `net.rim.device.api.crypto.*`
- `net.rim.device.api.io.http`
- `net.rim.device.api.notification`

- `net.rim.device.api.servicebook`
- `net.rim.device.api.synchronization`
- `net.rim.device.api.system`

Per ulteriori informazioni sulle API controllate RIM, consultare la documentazione di riferimento sulle API BlackBerry.

Supporto dei criteri IT

È possibile utilizzare le API contenute nel pacchetto `net.rim.device.api.itpolicy` per accedere alle informazioni sui criteri IT sul dispositivo BlackBerry® e modificare il comportamento o il funzionamento di un'applicazione del dispositivo BlackBerry. Ogni voce del criterio IT è composta da un nome, una descrizione e un valore. Il valore può essere una stringa, un numero intero o un valore booleano.

Le impostazioni del criterio IT del dispositivo BlackBerry vengono automaticamente sincronizzate e aggiornate sulla rete wireless alla versione 3.6 o successiva di BlackBerry® Device Software. Nelle versioni precedenti di BlackBerry Device Software, le impostazioni del criterio IT vengono aggiornate quando l'utente del dispositivo BlackBerry sincronizza il dispositivo con il computer.

Per ulteriori informazioni sui criteri IT, consultare la *Guida di riferimento ai criteri di BlackBerry Enterprise Server*.

Controllo delle applicazioni

Le regole dei criteri di controllo delle applicazioni di BlackBerry® Enterprise Server sono state concepite per impedire l'installazione di specifiche applicazioni sul dispositivo BlackBerry e per limitarne le autorizzazioni. Ad esempio, gli amministratori possono utilizzare il criterio di controllo delle applicazioni per assicurarsi che un gioco presente sul dispositivo BlackBerry non possa accedere all'applicazione del telefono.

L'amministratore può applicare i criteri di controllo delle applicazioni solo quando il dispositivo BlackBerry è associato a un BlackBerry Enterprise Server. Se l'amministratore o un utente del dispositivo BlackBerry impedisce all'applicazione di accedere a una delle aree protette, l'applicazione esegue un `ControlledAccessException`.

Crittografia file sulle schede microSD

Crittografia di dati su una scheda di memoria microSD

Quando una BlackBerry® Java® Application accede a un file sulla scheda di memoria microSD, il file viene decrittografato e spostato nella memoria principale affinché un'applicazione possa leggerlo. Affinché una BlackBerry Java Application possa accedere a un file protetto da password, il dispositivo BlackBerry non deve essere bloccato. I file crittografati hanno un'estensione `.rem` e non possono essere decrittografati su piattaforme non BlackBerry.

Se la NVRAM è stata rimossa e la scheda di memoria microSD è bloccata mediante una chiave del dispositivo BlackBerry, i dati sulla scheda di memoria microSD non sono più accessibili. Per rimuovere i dati non accessibili, avviare il dispositivo BlackBerry e rimuovere tutti i file multimediali crittografati.

Il dispositivo BlackBerry utilizza una chiave principale archiviata sulla scheda di memoria microSD per crittografare i file multimediali del dispositivo BlackBerry. La chiave principale impedisce al dispositivo BlackBerry di dover decrittografare o crittografare di nuovo tutti i file multimediali quando si disattiva la crittografia o si cambia la password.

Utilizzo della scheda di memoria microSD con più di un dispositivo BlackBerry

Se l'utente del dispositivo BlackBerry® sposta la scheda di memoria microSD in un dispositivo BlackBerry che non utilizza una password del dispositivo oppure una password che non decrittografa correttamente la chiave principale della scheda di memoria microSD, il dispositivo BlackBerry richiede all'utente di inserire la password della scheda di memoria microSD. Se il dispositivo BlackBerry presente una password, l'utente del dispositivo BlackBerry può utilizzare la richiesta per modificare la password della scheda di memoria microSD nella password del dispositivo BlackBerry.

Criteri IT e la scheda di memoria microSD

È possibile applicare i dati del criterio IT Crittografia scritti nella scheda di memoria microSD a tutti i file nuovi o modificati che vengono archiviati nella scheda di memoria microSD. Solo i file che vengono archiviati nella scheda di memoria microSD dopo che un amministratore imposta il criterio IT sono crittografati. Ad eccezione dei file multimediali, tutto il contenuto è crittografato.

Verifica di una BlackBerry Java Application

9

Offuscamento di una BlackBerry Java Application

Il compilatore per BlackBerry® Java® Development Environment BlackBerry® JDE Plug-in per Eclipse® è impostato per ridurre al minimo le dimensioni dell'applicazione. Il compilatore produce un file .cod che fornisce servizi simili a quelli offerti dai pacchetti di offuscamento per ridurre le dimensioni del file .cod. Ad esempio, BlackBerry Java Development Environment rimuove le informazioni seguenti da un file .cod:

- tutte le informazioni di debug
- nomi delle variabili locali
- numeri delle righe di origine
- metodi privati e nomi dei membri

In genere, per le proprie applicazioni non è necessario un ulteriore offuscamento oltre a quello già fornito per impostazione predefinita da BlackBerry JDE. Infatti, Research In Motion non offre alcun offuscamento aggiuntivo dei propri prodotti.

BlackBerry JDE non integra il supporto per l'offuscamento mediante gli strumenti di terzi. È necessario includere una procedura della riga di comando per offuscare i file .cod per l'utilizzo sui dispositivi BlackBerry.

Verifica preventiva di una BlackBerry Java Application

Quando si esegue una verifica preventiva delle classi, si riduce la quantità di elaborazione che il dispositivo BlackBerry® deve eseguire quando si installa l'applicazione. Per una verifica parziale delle classi prima di installare l'applicazione su un dispositivo BlackBerry, è possibile utilizzare lo strumento di verifica preventiva, disponibile con BlackBerry® Java® Development Environment. È possibile utilizzare BlackBerry Smartphone Simulator per la verifica preventiva dei file .cod.

Per ulteriori informazioni sull'utilizzo di BlackBerry Smartphone Simulator, vedere il *Manuale dell'utente di BlackBerry Smartphone Simulator*. Per ulteriori informazioni sull'utilizzo dello strumento di verifica preventiva, vedere la *Guida a BlackBerry Java Application per sviluppatori*.

Verifica delle applicazioni su un BlackBerry Smartphone Simulator

Dopo aver sviluppato e compilato un'applicazione, è possibile testarla sul dispositivo BlackBerry®. Il primo passaggio consiste nell'impostare BlackBerry® Java® Development Environment per l'utilizzo di un BlackBerry® Smartphone Simulator. BlackBerry Smartphone Simulator esegue lo stesso codice Java dei dispositivi BlackBerry, fornendo così un ambiente adatto per verificare con precisione il funzionamento delle applicazioni in un dispositivo BlackBerry. BlackBerry JDE include le versioni attuali di BlackBerry Smartphone Simulator. Per scaricare altre versioni di BlackBerry Smartphone Simulator, visitare www.blackberry.com/developers/index.shtml.

Verifica delle applicazioni su un dispositivo BlackBerry

Dopo aver eseguito la verifica dell'applicazione in BlackBerry® Smartphone Simulator, è possibile installare l'applicazione su un dispositivo BlackBerry. Se l'applicazione utilizza API firmate, potrebbero essere necessarie le chiavi di firma codice. Dopo aver installato l'applicazione sul dispositivo BlackBerry, è possibile aprire l'applicazione e verificarne la funzionalità e le prestazioni.

Ai fini di debug, è possibile associare il dispositivo a BlackBerry® Integrated Development Environment ed utilizzare lo strumento di debug per spostarsi tra il codice dell'applicazione. BlackBerry IDE può essere utile se si tenta di identificare una rete o un problema Bluetooth® o altri problemi difficili da simulare.

Simulatori BlackBerry

BlackBerry Smartphone Simulator

BlackBerry® Smartphone Simulator è progettato per emulare un'esperienza BlackBerry senza utilizzare un dispositivo BlackBerry reale. BlackBerry Smartphone Simulator è un'applicazione da installare sul proprio computer per visualizzare un'immagine del modello del dispositivo BlackBerry di propria scelta. Le caratteristiche e l'aspetto dell'immagine sono le stesse di un dispositivo BlackBerry reale. Le funzionalità offrono all'utente la stessa interazione di un dispositivo BlackBerry reale (rotellina, trackball e tastiera comprese); anche le applicazioni, le funzionalità, i messaggi di posta elettronica, il telefono e l'esplorazione Internet coincidono.

BlackBerry Smartphone Simulator funge anche da piattaforma per l'esecuzione delle applicazioni. Le applicazioni possono anche stabilire connessioni di rete, archiviare dati e gestire messaggi di posta elettronica. BlackBerry Smartphone Simulator include applicazioni del dispositivo BlackBerry tipicamente disponibili sui dispositivi BlackBerry e consente di installare e verificare le proprie applicazioni. Grazie a BlackBerry Smartphone Simulator è possibile simulare e verificare diversi cambiamenti di stato e connettività.

Quando si utilizza BlackBerry Smartphone Simulator per eseguire una verifica, potrebbe essere necessario dover simulare servizi BlackBerry aggiuntivi. BlackBerry® MDS Simulator e BlackBerry® Email Simulator sono disponibili per effettuare questa operazione.

Per ottenere BlackBerry Smartphone Simulator, visitare www.blackberry.com/developers e scaricare BlackBerry® Java® Development Environment o il pacchetto componenti di BlackBerry Java Development Environment.

BlackBerry MDS Simulator

BlackBerry® MDS Simulator è progettato per simulare il componente BlackBerry MDS Connection Service di BlackBerry® Enterprise Server. Quando si utilizza BlackBerry Smartphone Simulator con BlackBerry MDS Simulator è possibile verificare la rete, trasferire applicazioni HTTP e browser progettate per l'utilizzo con un BlackBerry Enterprise Server. Per ottenere BlackBerry MDS Simulator, visitare www.blackberry.com/developers e scaricare BlackBerry® Email and MDS Services Simulator Package.

BlackBerry Email Simulator

BlackBerry® Email Simulator è progettato per inviare e ricevere messaggi tra BlackBerry® Smartphone Simulator e un'altra applicazione di messaggistica, come Microsoft® Outlook® o i server POP3 e SMTP. Non è necessario disporre di un BlackBerry® Enterprise Server. Per ottenere BlackBerry Email Simulator, visitare www.blackberry.com/developers e scaricare BlackBerry® Email and MDS Services Simulator Package.

Come rendere disponibili le applicazioni

10

Distribuzione di applicazioni attraverso una connessione computer

Distribuzione di un'applicazione da un computer

È possibile utilizzare lo strumento caricatore applicazioni di BlackBerry® Desktop Manager per installare le applicazioni sul dispositivo BlackBerry. Lo strumento caricatore applicazioni può fornire agli utenti un metodo semplice per scaricare le applicazioni dal computer sul proprio dispositivo BlackBerry.

Distribuzione di un'applicazione da una pagina Web

È possibile utilizzare BlackBerry® Application Web Loader per pubblicare la propria applicazione compilata su un sito Web. Gli utenti possono utilizzare Windows® Internet Explorer® sul proprio computer per visitare la pagina Web e installare l'applicazione sul proprio dispositivo BlackBerry. Quando gli utenti del dispositivo BlackBerry visitano la pagina Web, BlackBerry Application Web Loader chiede loro di connettere i dispositivi alla porta USB. Gli utenti possono quindi installare l'applicazione utilizzando un controllo ActiveX®. BlackBerry Application Web Loader rappresenta una semplice soluzione che gli utenti del dispositivo BlackBerry possono utilizzare per installare le applicazioni dai propri computer senza eseguire BlackBerry® Desktop Manager.

Distribuzione di un'applicazione per la verifica

BlackBerry® Java® Development Environment include uno strumento della riga di comando chiamato JavaLoader che si trova nella cartella Cestino all'interno della cartella BlackBerry JDE. È possibile utilizzare lo strumento JavaLoader per installare e rimuovere in maniera rapida e diretta i file compilati dell'applicazione sul dispositivo BlackBerry attraverso la porta USB. Non sono necessari file descrittivi o pagine Web. Lo strumento JavaLoader può essere utile quando si installa e si rimuove frequentemente la propria applicazione nelle fasi di verifica e sviluppo; tuttavia, JavaLoader non è progettato per essere utilizzato dagli utenti del dispositivo BlackBerry.

Distribuzione di applicazioni sulla rete wireless

È possibile distribuire le proprie applicazioni sulla rete wireless per garantire agli utenti del dispositivo BlackBerry® un'esperienza migliore e per semplificare la distribuzione delle applicazioni ad un gruppo di persone numeroso poiché non è necessario disporre di un'applicazione del computer. Un utente del dispositivo BlackBerry può installare le applicazioni sulla rete wireless.

Pull wireless (attivato dall'utente)

È possibile pubblicare applicazioni compilate su un sito Web pubblico o privato. Gli utenti del dispositivo BlackBerry® possono visitare il sito Web per scaricare le applicazioni sulla rete wireless mediante il browser presente sui dispositivi BlackBerry. Il browser richiede agli utenti di installare l'applicazione, quindi l'applicazione effettua il download sulla rete wireless e l'installazione sul dispositivo BlackBerry.

Push wireless (avviato dal server)

Nell'ambiente BlackBerry® Enterprise Server, l'amministratore può trasferire applicazioni agli utenti del dispositivo BlackBerry sulla rete wireless per l'installazione obbligatoria. L'amministratore crea un nuovo criterio e specifica che il dispositivo BlackBerry richiede l'applicazione. L'applicazione è trasmessa agli utenti senza alcun intervento da parte dell'utente. Questo approccio potrebbe essere utile alle aziende quando desiderano inviare nuove applicazioni ad un numero elevato di utenti del dispositivo BlackBerry.

Glossario

11

AES

Standard di crittografia avanzata

API

Interfaccia di programmazione applicazioni

BlackBerry MDS

BlackBerry® Mobile Data System

CLDC

Connected Limited Device Configuration (Configurazione di dispositivi limitati e connessi)

CRM

customer relationship management (gestione dei rapporti con il cliente)

GUID

GUID (Globally Unique Identifier) è un numero di identificazione univoco creato dalle applicazioni Microsoft per identificare o tenere traccia di un'entità specifica, ad esempio un componente, un'applicazione, un file, una voce di database o un utente.

HTTP

Protocollo di trasferimento Hypertext

HTTPS

Protocollo di trasferimento Hypertext su layer di socket sicuri

I/O

input/output

IMAP

Internet Message Access Protocol

IPC

Comunicazione interprocesso

JDWP

Java® Debug Wire Protocol

JSR

Java® Specification Request

JVM

Java® Virtual Machine (Macchina virtuale Java®)

JTWI

Java® Technology for the Wireless Industry

JVM

Java® Virtual Machine (Macchina virtuale Java®)

MIDP

Mobile Information Device Profile

NVRAM

Memoria ad accesso casuale non volatile

PDAP

Pacchetti PDA opzionali per la piattaforma J2ME

PIM

Personal Information Management

PIN

Personal Identification Number (PIN)

POP

Post Office Protocol

RAPC

Compilatore del programma dell'applicazione RIM

RMS

Record Management System (sistema di gestione dei record)

SHA

Secure Hash Algorithm (Algoritmo hash protetto)

SMS

Short Message Service (Servizio SMS)

SMTP

Simple Mail Transfer Protocol (protocollo di trasferimento per la posta elettronica)

SRAM

memoria ad accesso casuale statica

SSL

Secure Sockets Layer

TCP

Transmission Control Protocol (Protocollo per il controllo della trasmissione)

TLS

Transport Layer Security (Protezione del livello trasporto)

Triple DES

Triple Data Encryption Standard (crittografia Triple DES)

VPN

Virtual Private Network

WAP

Wireless Application Protocol (Protocollo per applicazioni wireless)

XML

Extensible Markup Language

Note legali

12

©2009 Research In Motion Limited. Tutti i diritti riservati. BlackBerry®, RIM®, Research In Motion®, SureType®, SurePress™ e i relativi marchi, nomi e loghi sono di proprietà di Research In Motion Limited e sono registrati e/o utilizzati negli Stati Uniti e in altri Paesi.

Bluetooth è un marchio di Bluetooth SIG. Certicom è un marchio di Certicom Corporation. Eclipse è un marchio di Eclipse Foundation, Inc. Java e Javadoc sono marchi di Sun Microsystems, Inc. HotSync è un marchio di PalmSource, Inc. Microsoft, ActiveX, Internet Explorer e Windows sono marchi registrati di Microsoft Corporation. SQLite è un marchio di Hipp, Wyrick & Company, Inc. Tutti gli altri marchi sono di proprietà dei rispettivi titolari.

Lo smartphone BlackBerry e altri dispositivi e/o il software associato sono protetti da copyright, trattati internazionali e brevetti vari, inclusi uno o più dei seguenti brevetti USA: 6.278.442, 6.271.605, 6.219.694, 6.075.470, 6.073.318, D445.428, D433.460 e D416.256. Altri brevetti sono registrati o in attesa di registrazione negli Stati Uniti e in altri Paesi. Per un elenco aggiornato dei brevetti RIM, visitare il sito Web all'indirizzo: www.rim.com/patents.

La presente documentazione, compresa tutta la documentazione ivi allegata come riferimento, ad esempio la documentazione fornita o resa disponibile su www.blackberry.com/go/docs, viene fornita o resa accessibile "COSÌ COM'È" e "COME DISPONIBILE" e senza alcuna condizione, approvazione, garanzia, o dichiarazione di alcun tipo da Research In Motion Limited e le relative società affiliate ("RIM") e RIM non si assume alcuna responsabilità per errori tipografici, tecnici o per altre imprecisioni, errori o omissioni presenti nella documentazione. Al fine di tutelare le informazioni proprietarie e riservate e/o i segreti commerciali di RIM, è possibile che alcuni aspetti delle tecnologie di RIM vengano descritti in modo generico nella presente documentazione. RIM si riserva il diritto di modificare periodicamente le informazioni contenute nella presente documentazione, senza alcun obbligo di distribuzione agli utenti di modifiche, aggiornamenti, miglioramenti o aggiunte al documento in tempo utile.

La presente documentazione potrebbe contenere riferimenti a fonti di informazioni, hardware o software, prodotti o servizi di terzi, compresi componenti e contenuti protetti da copyright e/o siti Web di terzi (collettivamente denominati "Prodotti e servizi di terzi"). RIM non controlla né è responsabile per eventuali Prodotti e servizi di terzi, inclusi, senza alcuna limitazione, il contenuto, l'accuratezza, la conformità al copyright, le prestazioni, la compatibilità, l'affidabilità, la legalità, la decenza, i collegamenti o qualsiasi altro aspetto dei Prodotti e servizi di terzi. La presenza di riferimenti a Prodotti e servizi di terzi nella presente documentazione non implica in nessun caso l'approvazione da parte di RIM dei Prodotti e servizi di terzi o dei terzi.

A MENO CHE LE LEGGI APPLICABILI NELLA GIURISDIZIONE VIGENTE LO PROIBISCANO, CON IL PRESENTE DOCUMENTO SI ESCLUDONO TUTTE LE CONDIZIONI, APPROVAZIONI, GARANZIE, DICHIARAZIONI IMPLICITE O ESPLICITE DI OGNI GENERE, COMPRESSE, SENZA ALCUNA LIMITAZIONE, LE CONDIZIONI, APPROVAZIONI, GARANZIE, DICHIARAZIONI O GARANZIE DI DURATA, IDONEITÀ A UN PARTICOLARE SCOPO O USO, COMMERCIALITÀ, QUALITÀ, NON VIOLAZIONE, QUALITÀ SODDISFACENTE O TITOLO, OVVERO DERIVANTI DA STATUTI O USANZE COMMERCIALI O USI DI COMMERCIO, O ANCORA CORRELATE ALLA DOCUMENTAZIONE O ALL'USO DI QUESTA, OVVERO ALLE PRESTAZIONI O ALLE MANCATE PRESTAZIONI DI SOFTWARE, HARDWARE, SERVIZI O QUALSIASI TIPO DI PRODOTTI E SERVIZI DI TERZI IVI NOMINATI. L'UTENTE POTREBBE INOLTRE GODERE DI ULTERIORI DIRITTI A SECONDA DELLO STATO O DELLA PROVINCIA DI APPARTENENZA. È POSSIBILE CHE ALCUNE GIURISDIZIONI NON CONSENTANO L'ESCLUSIONE O LA LIMITAZIONE DI GARANZIE E CONDIZIONI IMPLICITE. NEI LIMITI CONSENTITI DALLA LEGGE, TUTTE LE GARANZIE E CONDIZIONI IMPLICITE IN RELAZIONE ALLA PRESENTE DOCUMENTAZIONE, CONSIDERANDO CHE LA LORO ESCLUSIONE NON È CONCESSA, IN

BASE A QUANTO SPECIFICATO PRIMA, MENTRE LA LORO LIMITAZIONE È CONCESSA, VENGONO QUI LIMITATE A NOVANTA (90) GIORNI A PARTIRE DALLA DATA DI ACQUISTO DELLA DOCUMENTAZIONE O L'ELEMENTO OGGETTO DELLA LAMENTELA.

ENTRO I LIMITI MASSIMI CONSENTITI DALLE LEGGI APPLICABILI NELLA GIURISDIZIONE VIGENTE, RIM NON PUÒ ESSERE IN NESSUN CASO RITENUTA RESPONSABILE DI QUALSIASI TIPO DI DANNI RELATIVI ALLA PRESENTE DOCUMENTAZIONE O ALL'USO DI QUESTA, OVVERO ALLE PRESTAZIONI O MANCATE PRESTAZIONI DI QUALSIASI SOFTWARE, HARDWARE, SERVIZIO O QUALSIASI DEI PRODOTTI E SERVIZI DI TERZI DI CUI SOPRA, COMPRESI, SENZA LIMITAZIONE, I DANNI SEGUENTI: DIRETTI, CONSEQUENZIALI, ESEMPLARI, INCIDENTALI, INDIRETTI, SPECIALI, PUNITIVI O AGGRAVATI, DANNI DERIVANTI DALLA PERDITA DI RICAVI O PROFITTI, MANCATA REALIZZAZIONE DEI GUADAGNI PREVISTI, INTERRUZIONE DELL'ATTIVITÀ COMMERCIALE, PERDITA DI INFORMAZIONI COMMERCIALI, PERDITA DI OPPORTUNITÀ COMMERCIALI OVVERO DANNEGGIAMENTO O PERDITA DI DATI, MANCATA TRASMISSIONE O RICEZIONE DI DATI, PROBLEMI ASSOCIATI A QUALSIASI DELLE APPLICAZIONI UTILIZZATE CON I PRODOTTI O I SERVIZI RIM, COSTI DERIVANTI DA TEMPI MORTI, MANCATO USO DEI PRODOTTI O DEI SERVIZI RIM O DI PARTI DI ESSI OVVERO DI QUALSIASI SERVIZIO DI DISPONIBILITÀ, COSTO DI SERVIZI SOSTITUTIVI, COSTI DI COPERTURA, STRUTTURE O SERVIZI, COSTI DI CAPITALE OVVERO ALTRE PERDITE DI CARATTERE PECUNIARIO DI TALE TIPO, SIA NEL CASO IN CUI TALI DANNI SIANO PREVISTI CHE NEL CASO IN CUI NON LO SIANO, E ANCHE QUALORA RIM SIA STATA AVVISATA DELLA POSSIBILITÀ DEL VERIFICARSI DI TALI DANNI.

NEI LIMITI MASSIMI CONSENTITI DALLE LEGGI APPLICABILI NELLA GIURISDIZIONE VIGENTE, RIM NON HA NESSUN ALTRO OBBLIGO, DOVERE O RESPONSABILITÀ RELATIVAMENTE ALL'ADEMPIMENTO DEL CONTRATTO, NEGLIGENZA O ALTRO NEI CONFRONTI DELL'UTENTE, COMPRESA LA RESPONSABILITÀ PER NEGLIGENZA O COMPATIBILITÀ.

LE LIMITAZIONI, ESCLUSIONI ED ESCLUSIONI DI RESPONSABILITÀ RIPORTATE NEL PRESENTE DOCUMENTO SI APPLICANO: (A) INDIPENDENTEMENTE DALLA NATURA DELLA CAUSA DELL'AZIONE, DELLA RICHIESTA O DELL'AZIONE INTRAPRESA DALL'UTENTE, COMPRESA, SENZA LIMITAZIONE, LA VIOLAZIONE DEL CONTRATTO, LA NEGLIGENZA, LE AZIONI LESIVE, LA COMPATIBILITÀ O QUALSIASI ALTRO DANNO CONTEMPLATO DALLA LEGGE E RIMARRANNO IN VIGORE IN SEGUITO A VIOLAZIONI DI FONDO O AL MANCATO ADEMPIMENTO DELLO SCOPO ALLA BASE DEL PRESENTE CONTRATTO O DI QUALSIASI CLAUSOLA IN ESSO PRESENTE; E (B) A RIM E ALLE RELATIVE SOCIETÀ AFFILIATE, AI SUCCESSORI DI QUESTE, AI CESSIONARI, AGLI AGENTI, AI FORNITORI (IVI COMPRESI I PROVIDER DI SERVIZI), AI RIVENDITORI AUTORIZZATI RIM (IVI COMPRESI I PROVIDER DI SERVIZI) E AI RISPETTIVI DIRETTORI, DIPENDENTI E COLLABORATORI INDIPENDENTI.

OLTRE ALLE LIMITAZIONI ED ESCLUSIONI DI CUI SOPRA, IN NESSUN CASO SUI DIRETTORI, I DIPENDENTI, GLI AGENTI, I RIVENDITORI, I FORNITORI, I COLLABORATORI INDIPENDENTI DI RIM O TUTTI GLI AFFILIATI DI RIM RICADRÀ LA RESPONSABILITÀ DERIVANTE DA O CORRELATA ALLA PRESENTE DOCUMENTAZIONE.

Prima della sottoscrizione, dell'installazione o dell'utilizzo di Prodotti e servizi di terzi, è responsabilità dell'utente assicurarsi che il provider di servizi garantisca il supporto di tutte le relative funzioni. Alcuni provider di servizi potrebbero non offrire la funzionalità di esplorazione in Internet con un abbonamento a BlackBerry® Internet Service. Verificare con il provider di servizi la disponibilità, le disposizioni di roaming, i piani di servizio e le funzionalità. Per l'installazione o l'utilizzo di Prodotti e servizi di terzi con prodotti e servizi RIM potrebbero essere necessari uno o più brevetti, marchi, copyright o licenze di altro tipo al fine di evitare violazioni dei diritti di terzi. L'utilizzo di Prodotti e servizi di terzi e l'eventuale necessità di licenze di terzi sono responsabilità esclusive dell'utente. Nel caso in cui sia necessario, l'utente è responsabile del relativo acquisto. L'utente non deve installare o utilizzare Prodotti e servizi di terzi fino all'acquisto delle relative licenze richieste. Tutti i Prodotti e servizi di terzi forniti con i prodotti e i servizi RIM hanno lo scopo di offrire maggiore comodità all'utente e vengono forniti "COSÌ COME SONO", senza condizioni, approvazioni, garanzie, dichiarazioni espresse o implicite di qualsiasi tipo da parte di RIM. RIM non si assume nessun tipo di

responsabilità a tale riguardo. L'utilizzo da parte dell'utente di Prodotti e servizi di terzi deve essere sottoposto e soggetto all'accettazione da parte dell'utente stesso dei termini di licenze indipendenti e altri contratti ivi applicabili con terzi, salvo quanto espressamente coperto da eventuali licenze o altri contratti stipulati con RIM.

Alcune funzionalità illustrate nella presente documentazione richiedono una versione minima del software BlackBerry® Enterprise Server, BlackBerry® Desktop Software, e/o BlackBerry® Device Software.

I termini di utilizzo di tutti i prodotti o servizi RIM vengono definiti in una licenza o in un altro contratto indipendente stipulato con RIM ivi applicabile. **NESSUNA AFFERMAZIONE CONTENUTA NELLA PRESENTE DOCUMENTAZIONE PUÒ SOSTITUIRE CONTRATTI O GARANZIE SCRITTE ESPLICITE FORNITE DA RIM PER PARTI DI PRODOTTI O SERVIZI RIM DIVERSI DA QUELLI INDICATI NELLA PRESENTE DOCUMENTAZIONE.**

Research In Motion Limited
295 Phillip Street
Waterloo, ON N2L 3W8
Canada

Research In Motion UK Limited
Centrum House
36 Station Road
Egham, Surrey TW20 9LF
Regno Unito

Pubblicato in Canada