



Tout ce qui est indispensable
pour réaliser directement
des **applets** réellement
professionnelles

Nouveau

Le développeur JavaTM2 Mise en œuvre et solutions Les Applets

- Exploitez Java 2 avec **tous** les navigateurs existants
- Du son en temps réel
- Des images affichées instantanément

SUR LE CD-ROM GRATUIT



- Le SDK Java 2 y compris le JDK 1.2.2
- Le JDK 1.1.8, Le JDK 1.0.2, Le JSDK
- HotJava 3.0, UltraEdit 32 (shareware)
- WebSpeed Optimizer, CoolEdit (shareware)
- JPEG Optimizer, Winzip (shareware)
- La version électronique indexée du livre au format PDF
- Acrobat Reader, etc.

www.editions-eyrolles.com

Antoine Mirecourt et Pierre-Yves Saumont

PRESENT
SUR LE
WEB



OSMAN EYROLLES MULTIMEDIA

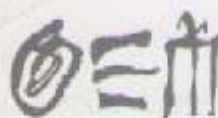
2-005-11-1

2-005-11-1



Le développeur Java2 Mise en œuvre et solutions Les Applets

Antoine Mirecourt
Pierre-Yves Saumont



OSMAN EYROLLES MULTIMEDIA



Sommaire

Introduction	xvii
La théorie et la pratique	xvii
Lassés des exemples dignes du jardin d'enfants ?	xix
Un investissement rentable	xx
Les droits d'utilisation	xxi
Un cas de problème	xxii
Le CD-ROM	xxii
De quoi parlons-nous ?	xxiii
Conventions d'écriture	xxv
Les identificateurs	xxvi
L'indentation	xxviii
Les accolades	xxix
Les blocs d'une seule ligne	xxx
Les parenthèses	xxxi
Les commentaires	xxxi

De l'ingéniosité des solutions proposées	xxxii
Ce que vous ne trouverez pas dans ce livre	xxxii
Ce que vous êtes censé savoir avant de commencer	xxxiii
Chapitre 1 : L'environnement de programmation	1
Le SDK	2
Configuration	2
Installation des versions précédentes	3
Installation de la documentation	5
Éditeur ou IDE	5
Le(s) navigateur(s)	8
Un serveur HTTP	16
Autres outils	20
Les obfuscators	20
Les décompilateurs	24
Les programmes de traitement des ressources	25
Résumé	26
Chapitre 2 : La stratégie et le cahier des charges	27
Variabilité de l'environnement	28
Les extensions	29
Les spécificités de chaque environnement	31
Différences d'implémentation	32
Les bugs	37
Les ressources disponibles	41
Définir la cible	43
Le choix d'un modèle de couleurs	47
La bande passante	49
Pérennité du modèle	51
Optimisation des sources	51
Quelle version de Java employer ?	54

Précautions à prendre pour tester les applets	56
Résumé	57
Chapitre 3 : Conception d'une applet	59
La méthode d'apprentissage	61
Structure de l'applet	63
L'écran d'introduction	63
Les questions	64
Les réponses	66
Le résultat	67
Les composants nécessaires	67
Les composants visuels	68
Les composants non visuels	70
Le modèle fonctionnel de l'applet	72
Chargement statique des séries	73
Chargement dynamique des séries	73
Le préchargement contrôlé	73
Modèle fonctionnel, suite	74
Préchargement des images	75
Mise en page des questions	76
Layout manager or not layout manager	81
Mise en page statique	82
Layout manager	82
Résumé	83
Chapitre 4 : Les fondations de notre applet	85
Un document HTML	86
Création de l'applet	91
Définition des constantes	91
Les couleurs	92
Les polices de caractères	96

Les variables	98
La méthode paint()	102
Les coordonnées des pixels	103
Résumé	110
Chapitre 5 : Les composants graphiques personnalisés	111
Un composant pour afficher des images	112
Spécifications	112
Un composant pour afficher du texte en relief	123
Optimisation	130
Moyens globaux	130
Moyens internes	131
Moyens externes	132
Moyens spécifiques	133
Optimisation de la classe Label3D	133
Optimiser la maintenabilité	135
Résumé	142
Chapitre 6 : Communication entre objets : les événements	143
Référence à la classe contenant	146
Utiliser des classes internes	147
Passer une référence à la classe contenant	148
Utilisation d'une classe anonyme	153
Les classes internes en Java 1.0	157
Utiliser les événements	158
Postage des événements	160
Événements et méta-événements	161
Circulation des événements, modèle Java 1.0	162
Principe fonctionnel d'un bouton selon le modèle 1.0	164
Les événements en Java 1.1 et Java 2	165
Circulation des événements en Java 1.1 et Java 2	166
Résumé	169

Chapitre 7 : Création d'un composant actif	171
L'aspect graphique	172
Implémenter le modèle 1.1	179
Recevoir les événements souris	179
Une autre façon de procéder	184
Fiabiliser la diffusion des événements	190
Implémenter le modèle 1.0	191
Résumé	211
Chapitre 8 : Des contrôles graphiques	213
Création d'images	214
Utilisation de ressources externes	214
Un composant de type glissière	217
Création d'images bitmap en Java	222
Création du modèle colorimétrique	222
Création du tableau de données	224
Création de l'image	226
Des boutons avec des icônes	240
Résumé	251
Chapitre 9 : Mise en page automatique du texte	253
Les membres communs à toutes les questions	254
Les questions	256
Mise en page des questions	259
Modification de la classe principale	267
Disposition automatique des éléments	270
Tourner les pages	276
Affichage du résultat	283
Afficher les explications : le texte multilignes	285
Utilisation de l'étiquette multilignes	290

Afficher les erreurs	293
Sélection des séries	297
Résumé	299
Chapitre 10 : Optimiser le chargement des images	301
Optimiser les images	302
Optimiser le chargement des images	306
Création d'un thread	309
Contrôler le chargement des images	311
Arrêt du thread	313
Ajouter une boîte de dialogue	315
Améliorations	320
Afficher la progression	320
Résumé	329
Chapitre 11 : Utiliser les threads pour animer les applets	331
L'animation	332
Principe du clignotement	333
Interaction avec le fond	334
Pourquoi pas des GIF animés ?	334
Problèmes des images JPEG	335
Un composant pour afficher les clignotants	335
Stratégie	336
Synchroniser les clignotants	342
Un serveur indépendant	348
Un séquenceur d'animation	354
Exploiter le clignotement	361
Résumé	377

Chapitre 12 : Automatiser l'applet	379
Modification de la classe principale	380
Modification de la classe Blinker	382
Création d'une interface pour les clients	386
Modification de la classe ImageBlinker	386
Modification de la classe Serie	387
Modification de la classe ConfirmDialog	398
Modification de la classe Button3D	398
Résumé	399
Chapitre 13 : Jouer des clips audio	401
Jouer un clip audio	402
Problèmes liés à l'utilisation des clips	408
Les packages Sun	409
La classe AppletAudioClip	410
La classe AudioData	411
La classe AudioPlayer	412
Les données audio	413
Aperçu des méthodes de compression	416
Utilisation d'une échelle logarithmique	418
Décalage de l'échelle	418
Codage des silences	419
Fréquence d'échantillonnage variable	420
Modélisation d'un type de son	421
Les différents formats de fichiers audio	423
Comment Java lit les données audio	424
Le choix d'un format	424
Les formats GSM	425
Les "formats de fichiers" GSM	426
Résumé	427

Chapitre 14 : Lire les fichiers Wave	429
Lire un fichier audio encodé MuLaw	430
Remarques	436
Pourquoi utiliser une classe interne ?	436
Pourquoi utiliser la méthode <code>readUnsignedByte()</code> ?	437
Les aventures de la méthode <code>read()</code>	437
Lire un fichier Wave encodé MuLaw	441
Le format Wave	442
L'en-tête RIFF	442
Le bloc Format	443
Les autres blocs	444
Le bloc data	444
Lecture brute d'un fichier Wave	444
Lecture d'un fichier Wave avec contrôle du format	449
Résumé	457
Chapitre 15 : Convertir les formats audio	459
Les données audio brutes	460
La qualité du son échantillonné	461
Influence du type de son	461
Influence du type de mesure	461
La qualité subjective	462
Quelle qualité utiliser ?	463
Un encodeur MuLaw	464
Utilisation de la classe MuLaw	470
Contrôler le clip	481
Voie à suivre pour d'autres types de conversions	496
Résumé	497
Chapitre 16 : Un décodeur GSM 6.10	499
Principe du codage GSM	500

Le décodage	502
Réalisation du décodeur	503
Le décodeur	510
Décodage des fichiers WAVE#49	520
Le streaming	538
Écouter le son pendant le décodage	540
Écouter le son pendant son chargement	541
Mieux contrôler le déroulement du clip	543
Revenir en arrière d'une valeur fixe	544
Connaître la longueur totale et la longueur restante	546
Positionner le clip à un point quelconque	552
Le streaming des fichiers longs	553
Optimisation de la mémoire	556
Résumé	569
Chapitre 17 : Mise en œuvre du décodeur GSM	571
Les classes SoundSystem et Controleur	572
Modification de l'applet	580
Modification des questions	584
Résumé	587
Chapitre 18 : Optimiser les applets	589
Tester en situation réelle	590
WebSpeed Optimizer	590
Installation de WebSpeed Optimizer	591
Utilisation de WebSpeed Optimizer	591
Le chargement dynamique des classes	594
Précharger les classes	603
Modification de la classe imageCanvas	603
L'interface imageClient	605
Modification de l'applet	605

Résultat	608
Précharger les éléments dans d'autres pages Web et préinitialiser Java	608
Mise en mémoire cache et URL	616
Utilisation des fichiers d'archives	617
Les formats d'archives	618
Utilisation des archives	619
Bénéfice de l'archivage	626
Optimiser le code pour les performances	626
Optimiser le code pour le volume	627
Résumé	629
Chapitre 19 : Optimiser l'affichage	631
Les problèmes de compatibilité de l'affichage	632
Problèmes de rendu des couleurs	635
Optimiser les animations	636
Résumé	639
Chapitre 20 : Java 2 avec tous les navigateurs	641
Fonction du plug-in Java 2	642
Où télécharger le plug-in ?	642
Comment télécharger le plug-in ?	643
Inconvénient du plug-in Java 2	643
Les classes propriétaires	644
Principe de l'utilisation du plug-in Java 2	644
Utilisation du plug-in avec les navigateurs Microsoft	645
Utilisation du plug-in avec les navigateurs Netscape	647
Utiliser le plug-in avec les navigateurs Netscape et Microsoft	648
Utiliser le plug-in avec tous les navigateurs	650
Résumé	654

Chapitre 21 : L'environnement des applets	657
Afficher des messages dans la barre d'état	658
Afficher des documents	659
Communiquer avec les autres applets de la page	660
Exemple de communication entre applets	660
Résumé	670
Chapitre 22 : HTML et les applets	671
Les éléments et les balises	672
Les attributs	673
Les éléments concernant les applets	674
Les attributs communs à tous les éléments	674
<APPLET> (applet Java)	678
<EMBED> (objet enchâssé)	684
<NOEMBED> (élément de remplacement)	689
<OBJECT> (objet enchâssé)	691
Index	701