Formation Comprendre la stack graphique sous Linux

Durée de la formation

4 demi-journées – 16 h

Langue

Transparents	Anglais
Présentation	Français
	Anglais

Formateur

Cette formation n'est plus proposée par Bootlin.

Contact

training@bootlin.com
+33 4 84 25 80 96

bootlin
bootlin.com

Formation en ligne

Public visé

Développeurs de systèmes multimédia utilisant le noyau Linux

Objectifs opérationnels

- Être capable de comprendre les bases de l'affichage graphique : représentation des images et des couleurs, affichage de pixels, opérations sur les pixels.
- Être capable de comprendre le matériel lié à l'affichage graphique : composants du *pipeline* graphique, matériel pour le rendu et l'affichage graphique.
- Avoir une compréhension claire des composants de la pile logicielle pour le graphique dans le noyau Linux et de leurs rôles : TTY, sous-systèmes *framebuffer* et DRM.
- Avoir une compréhension claire de la pile logicielle pour le graphique en espace utilisateur : DRM au niveau espace utilisateur, X.org, Wayland, OpenGL.

Prérequis

- **Expérience solide en programmation avec le langage C** : les participants doivent maîtriser l'utilisation de types de données et structures complexes, des pointeurs, pointeurs sur fonction et du pré-processeur C.
- **Expérience en développement bas-niveau sous Linux et avec les interfaces matérielles** : les participants doivent avoir une compréhension minimale de la gestion mémoire, de l'interaction avec des interfaces matérielles (registres, interruptions) et de l'interaction entre les applications user-space Linux et le noyau Linux (appels systèmes). Suivre la formation la formation Développement de pilotes de périphériques noyau Linux de Bootlin permet de remplir ce pré-requis.
- **Niveau minimal requis en anglais : B1**, d'après le *Common European Framework of References for Languages*, pour nos sessions animées en anglais. Voir la grille CEFR pour une auto-évaluation.

Méthodes pédagogiques

- Présentations animées par le formateur, par visioconférence. Les participants peuvent poser des questions à tout instant.
- Démonstrations pratiques réalisées par le formateur, par vidéo-conférence. Les participants peuvent poser des questions à tout instant.
- Messagerie instantanée pour questions entre les sessions (réponse sous 24h, hors week-end et jours fériés)
- Version électronique des supports de présentation, des instructions et des données de travaux pratiques. Les supports sont librement disponibles ici.

Modalités d'évaluation

Seuls les participants qui auront assisté à l'intégralité des journées de formation, et qui auront obtenu plus de 50% de réponses correctes à l'évaluation finale recevront une attestation individuelle de formation de la part de Bootlin.

Handicap

Les participants en situation de handicap qui ont des besoins spécifiques sont invités à nous contacter à l'adresse *training@bootlin.com* afin de discuter des adaptations nécessaires à la formation.

Équipement nécessaire

Équipement obligatoire :

- Ordinateur avec le système d'exploitation de votre choix, équipé du navigateur Google Chrome ou Chromium pour la conférence vidéo.
- Une webcam et un micro (de préférence un casque avec micro)
- Une connexion à Internet à haut débit

Optionnellement, si les participants souhaitant pouvoir reproduire par eux-mêmes les travaux pratiques, ils doivent acheter séparément la carte de développement et les accessoires associés, et devront diposer d'un PC avec une installation native d'Ubuntu Linux 24.04.

Programme de la formation

Demi-journée 1

Cours	Représentation des images et des couleurs	• Lumière, pixels et images • Échantillonage, domaine de fréquence, aliasing • Quantification et représentation des couleurs • Espaces colorimétriques et canaux, canal alpha • Sous-échantillonnage YUV et chroma • Plans de données de pixels, ordre d'analyse • Formats de pixels, codes FourCC codes, modificateurs *Introduction aux notions de base utilisées pour représenter les images en couleur.*
Cours	Dessin des pixels	• Accès aux données de pixels et itération • Concepts autour de la pixellisation • Dessin de rectangles • Dessin de gradients linéaires • Dessin de disques • Dessin de gradients circulaires • Dessin de lignes • Aliasing de lignes et de formes, dessin sub-pixel • Cercles et coordonnées polaires • Courbes paramétriques *Comment accéder aux données de pixels en mémoire et dessiner des formes simples.*
Cours	Opérations sur les pixels	• Copie de région • Alpha blending • Keying de couleur • Mise à l'échelle et interpolation • Filtrage linéaire et convolution • Filtres de floutage • Dithering *Notions de base autour du filtrage, avec des exemples d'utilisation très courants.*
Démo	Dessin et opérations	• Exemples de dessin de divers types de formes et de régions • Exemples d'opérations de base sur les pixels *Illustration des concepts présentés au fur et à mesure.*

Demi-journée 2

Cours	Vue d'ensembe des composants du pipeline et généralités	• Types d'implémentations de matériel graphique • Mémoire graphique et buffers • Pipelines graphiques • Vue d'ensemble du matériel d'affichage, de rendu et de vidéo *Cours du matériel impliqué dans les pipelines graphiques.*

Cours	Matériel d'affichage	▪ Technologies d'affichage visuel : CRT, plasma, LCD, OLED, EPD ▪ Timings d'affichage, modes et EDID ▪ Interfaces d'affichage : VGA, DVI, HDMI, DP, LVDS, DSI, DP ▪ Bridges et transcodeurs *Cours du fonctionnement interne du matériel d'affichage.*
Cours	Spécificités du matériel de rendu	▪ Digital Signal Processors (DSPs) ▪ Accélérateurs matériels dédiés ▪ Graphics Processing Unit (GPUs) *Description de l'architecture du matériel de traitement et de rendu.*
Cours	Intégration système, mémoire et performance	▪ Intégration graphique et mémoire ▪ Mémoire partagée pour les graphiques ▪ Contraintes et performance de la mémoire graphique ▪ Soulager le processeur en utilisant du matériel dédié au graphisme ▪ Conseils pour les performances graphiques *Sujets autour de l'intégration système, la gestion de la mémoire et les performances.*

Demi-journée 3

Cours	Pile d'affichage	▪ Vue d'ensemble indépendante du système : noyau, affichage et rendu en espace utilisateur ▪ Vue d'ensemble de la partie dans le noyau Linux ▪ Vue d'ensemble de la partie bas niveau en espace utilisateur ▪ X Window et Wayland ▪ Bibliothèques graphiques haut niveau et environnements de bureau *Cours des composants nécessaires à un traitement graphique moderne, et comment ceux-ci sont répartis entre les espace noyau et utilisateur*
Cours	Aspects noyau TTY et device framebuffer	▪ Introduction au sous-système TTY de Linux ▪ Terminaux virtuels et graphiques ▪ Basculer entre terminaux virtuels et graphiques ▪ Vue d'ensemble de fbdev ▪ Opérations de base de fbdev ▪ Limitations de fbdev *Comment les TTYs interagissent avec les graphiques sous Linux et brève présentation de fbdev et pourquoi ce composant n'est plus recommandé*
Cours	DRM dans le noyau	▪ Devices DRM ▪ Identification et fonctionnalités des pilotes DRM ▪ Maître DRM, "magic authentification" ▪ Gestion de la mémoire des DRM ▪ API "dumb buffer" de DRM KMS ▪ Modificateurs et FourCCs dans DRM ▪ Détection des ressources dans DRM KMS ▪ Modes DRM KMS ▪ Gestion de framebuffer dans DRM KMS ▪ Ancien système de configuration de DRM KMS et échange de pages ▪ Notification d´évènements dans DRM ▪ Propriétés d'objets dans DRM KMS ▪ DRM KMS atomic ▪ Rendu DRM ▪ Partage mémoire sans copie (dma-buf) avec DRM Prime ▪ Barrières d'objets DRM sync ▪ Débug et documentation dans DRM *Une présentation complète de l'interface DRM.*
Démo	Aspects noyau	▪ Terminaux virtuels et TTYs dans Linux ▪ Configuration des modes DRM KMS ▪ Visite guidée d'un pilote DRM KMS ▪ Visite guidée d'un pilote de rendu DRM *Illustration du fonctionnement en espace noyau.*

Demi-journée 4

Cours	Aspects X Window en espace utilisateur	▪ Protocole X11 et son architecture ▪ Extensions au protocole X11 ▪ Architecture de Xorg et accélération ▪ Cours des pilotes Xorg ▪ Accélération X11 et OpenGL : GLX et DRI2 ▪ Utilisation de Xorg, intégration et configuration ▪ Principaux problèmes avec X11 ▪ Débug et documentation de Xorg *Cours des tous les aspects de X11 et Xorg.*
Cours	Aspects Wayland en espace utilisateur	▪ Vue d'ensemble et paradigmes de Wayland ▪ Protocole Wayland et son architecture ▪ Détails sur le coeur du protocole Wayland ▪ Protocoles supplémentaires de Wayland ▪ Interface asynchrone de Wayland ▪ Intégration OpenGL de Wayland ▪ Statut et adoption de Wayland ▪ Débug et documentation de Wayland *Une présentation approfondie de Wayland.*
Cours	Aspects Mesa 3D en espace utilisateur	▪ APIs de rendu 3D standardisées : OpenGL, OpenGL ES, EGL and Vulkan ▪ Vue d'ensemble de Mesa 3D ▪ Principaux détails d'implémentation de Mesa 3D ▪ Détails internes de Mesa 3D : Gallium 3D ▪ Détails internes de Mesa 3D : représentations intermédiaires ▪ Generic Buffer Management (GBM) dans Mesa 3D ▪ Point sur la prise en charge du matériel par Mesa 3D ▪ Mesa 3D comparée aux implémentations propriétaires ▪ Prise en charge du matériel par Mesa 3D : débug et documentation *Cours des APIs 3D et implémentation de Mesa 3D.*
Démo	Aspects en espace utilisateur	▪ Visite guidée du code de Xorg ▪ Visite guidée du coeur du compositeur Wayland ▪ Exemples de clients Wayland ▪ Visite guidée du code de Mesa ▪ Exemples OpenGL et EGL *Illustration des aspects en espace utilisateur et d'implémentations de clients et de serveurs.*