



CATALOGUE DES FORMATIONS LINUX

Septembre 2011



Site web : <http://www.eurogiciel.fr>
Contact : formations@eurogiciel.fr

Tél. : 02 28 23 21 00

Fax : 02 51 13 94 38

www.eurogiciel.fr



Sommaire

1.	QUI SOMMES-NOUS ?	3
2.	EUROGICIEL ET LINUX	5
3.	LA FORMATION LINUX CHEZ EUROGICIEL.....	6
	FORMATION « LINUX NOYAU ET DRIVERS » - REF : LINUX_01	7
	FORMATION « LINUX EMBARQUE » - REF : LINUX_02	8
	FORMATION « LINUX TEMPS REEL » - REF : LINUX_03	9
	FORMATION « LINUX SYSTEME » - REF : LINUX_06	10
	FORMATION « LINUX EMBARQUE » - REF : LINUX_06_COMP	11
	FORMATION « LINUX POUR LES MANAGERS » - REF : LINUX_07	12
	FORMATION « AUTOTOOLS » - REF : LINUX_8	13
	FORMATION « BASCULER DU C AU C++ » - REF : LINUX_9	14
4.	FORMATIONS A VENIR COURANT 2011	15
5.	PLAN D'ACCES A NOS CENTRES DE FORMATION.....	16
6.	CONDITIONS GENERALES.....	20
7.	BULLETIN D'INSCRIPTION	25



1. Qui sommes-nous ?

EUROGICIEL, société de 850 collaborateurs est présente dans les secteurs :

- Aéronautique, Spatial et Défense,
- Télécoms, Automobile, Transport,
- Tertiaire (Banque, Assurance, Administration...)

Le Groupe est composé de 3 filiales métier ayant des spécificités métier complémentaires :

- **ETOP International** : Management de Projets et Formation
- **EQUERT** : Management de la Qualité
- **EUROGICIEL Ingénierie** : Ingénierie Système & Logiciel

Le groupe EUROGICIEL est présent dans tous les domaines en tant que :

Société de Services en Accompagnement de Projets

EUROGICIEL se définit, sur l'ensemble de ses prestations auprès de ses clients en informatique, comme équipementier logiciel :

- Le développement **d'applications forfaitaires** constitue l'une des **prestations majeures** du groupe EUROGICIEL. Nous réalisons des forfaits importants en termes de charges, d'équipes et d'enjeux stratégiques.
- Au-delà de nos compétences reconnues sur les outils utilisés dans le domaine de l'**embarqué critique**, notre **référentiel méthodologique**, issu de la capitalisation de nos expériences et d'investissements réalisés depuis de nombreuses années, constitue un atout concurrentiel garantissant non seulement **la qualité des produits livrés mais aussi la réussite de nos projets**.
- L'intégralité du processus de réalisation de nos projets au forfait est certifiée **ISO 9001** depuis **1996**.
- Depuis février 2005, ce sont trois de nos métiers, **EUROGICIEL Ingénierie, ETOP INTERNATIONAL et EQUERT** qui sont certifiés **ISO9001 V2000** par le BVQI et ce, pour l'intégralité du Système d'Organisation.
- La **Direction d'EUROGICIEL** s'engage fortement dans le bon déroulement et la réussite des projets.

Eurogiciel est présent sur le territoire français et européen :

- Bordeaux, Nantes, Marseille, Paris, Rennes, Sophia-Antipolis, Toulouse, Valence
- Allemagne (Munich), Grande-Bretagne (Filton), Tunisie (Tunis)





Nos clients en informatique industrielle :



- ✓ Airbus
- ✓ RATP
- ✓ Ansaldo
- ✓ Rockwell Collins
- ✓ Alstom
- ✓ Safran,
- ✓ Arcelor Mittal
- ✓ Sercel
- ✓ ASF
- ✓ Siemens
- ✓ Bouygues Telecom
- ✓ Smardtv
- ✓ CEA
- ✓ Solysic
- ✓ CLS
- ✓ Teamcast
- ✓ Cegelec
- ✓ Technicolor
- ✓ Continental Automotiv
- ✓ Thales
- ✓ CNES
- ✓ ...
- ✓ Dassault
- ✓ DGA
- ✓ DTI DSNA
- ✓ EADS
- ✓ Egisavia
- ✓ GDF
- ✓ Gemalto
- ✓ General Electric
- ✓ Honeywell
- ✓ I2S
- ✓ Lacroix-Sofrel
- ✓ Météo France
- ✓ Orange



2. Eurogiciel et Linux

Linux constitue aujourd’hui une véritable alternative aux OS propriétaires embarqués en offrant :

- un large support d’architectures matérielles,
- une pérennité du logiciel par l’accès aux codes source,
- un support unique de par la vaste communauté de développeurs.

Linux a su également évoluer pour apporter des réponses aux systèmes embarqués nécessitant un véritable traitement temps réel. Il est à présent une solution incontournable dans le domaine des OS embarqués.

Aujourd’hui, Eurogiciel accompagne ses clients sur leurs développements autour de Linux embarqué en proposant :

- des offres d’expertise,
- de l’accompagnement au travers de ses formations,
- de la réalisation de projets au forfait dans son centre de services Linux embarqué.

L’expertise d’Eurogiciel couvre notamment :

- Le portage d’un système existant vers une architecture Linux
- La mise en place et l’optimisation d’architecture Linux sur des systèmes embarqués
- L’expertise sur les piles réseau sous Linux
- Le choix et l’implémentation de solutions du temps réel sur Linux embarqué
- L’optimisation de la performance des systèmes Linux embarqué (stratégie d’ordonnancement, règles de codage..).
- L’optimisation du temps de démarrage des systèmes embarqués
- La résolution de problèmes complexes
- L’accompagnement de la gestion de situations de crise sur un système embarqué
- La formation et l’accompagnement dans la montée en compétence des équipes



3. La formation Linux chez Eurogiciel

Eurogiciel accompagne ses clients au travers de ses formations Linux dispensées en intra- ou inter- entreprise sur site ou dans nos centres de formation.

Eurogiciel propose d'adapter le contenu et la durée de vos formations intra-entreprise. Au travers d'un audit gratuit, nous définissons un contenu adapté à vos besoins afin de vous offrir une formation sur mesure optimisée. Nous proposons également le formateur le plus adapté à votre domaine d'activité.

Chacun de nos formateurs possède plus de 15 ans d'expérience en développement de systèmes embarqués sous Linux, open source et formation.

Références de nos formateurs:

- Diplômes : Supelec, doctorat Systèmes et réseaux et ordonnancement
- Sociétés : Ubicom (USA), 3com(USA), Thomson, Thales, France Telecom, Alcatel...
- Domaines : TV numérique, téléphonie mobile, équipements réseau, crypto/sécurité, stratégie d'ordonnancement ...
- Contribution open source: OpenMoko, Yaksee, Google Android...

Organisme agréé

Le groupe Eurogiciel est un organisme agréé ce qui offre la possibilité de formations dans le cadre du DIF ou d'un OPCA (Organisme Paritaire Collecteur Agréé) de type FAFIEC, OPCAIM, ANFA, etc.

Nous avons déjà formé plus de 40 professionnels sur ce périmètre depuis 2010. Quelques références : Technicolor, Faiveley, Teamcast, TES, SII, Sogeti, Sagem Sécurité, Thales...

Objectif de la formation:

- ✓ Ce stage vise à présenter la structure générale et la structure détaillée du noyau GNU Linux. De plus, il aborde les différents types de drivers existant sous ce système (drivers de blocs, de caractères, réseau, pci, usb ...) via des méthodologies de développement spécifiques. Enfin, il aborde les concepts avancés utilisés régulièrement dans le noyau (DMA, interruptions, I/O, synchronisation...).
- ✓ La formation est ponctuée de différents travaux pratiques pour rendre ces concepts plus appliqués.
- ✓ **Public visé : Développeurs Produits souhaitant acquérir une compétence sur le noyau GNU Linux et les méthodes liées à son utilisation.**

Introduction

- ✓ Historique
- ✓ Organisation du développement
- ✓ L'équipe fondatrice
- ✓ Portabilité
- ✓ Nouveautés de la version 2.6
- ✓ Versioning du noyau
- ✓ Caractéristiques principales du noyau
- ✓ Problématiques de la licence GPL

Génération d'un noyau Linux

- ✓ Outils de compilation (packages nécessaires)
- ✓ Récupération des sources officielles ([http](http://), [ftp](ftp://), [nfs](nfs://))
- ✓ Autre moyen par GIT
- ✓ Vérification de l'intégrité des sources
- ✓ Application et génération de patchs
- ✓ Configuration du noyau
- ✓ Compilation
- ✓ Description des installations
- ✓ Remonter un bug officiel

Exécution

- ✓ Analyse de la phase de boot
- ✓ Exemple de bootloader
- ✓ Paramètres du noyau
- ✓ Extraction et génération d'une image initrd
- ✓ Chargement de modules au boot
- ✓ Déchargement dynamique de modules

Architecture du noyau Linux

- ✓ Vue simplifiée du noyau
- ✓ Vue modulaire du noyau
- ✓ Hypergraphes du noyau
- ✓ Rôles du noyau
- ✓ Découpage du noyau
- ✓ Le système de fichiers virtuel
- ✓ Le gestionnaire de processus
- ✓ Le gestionnaire de mémoire
- ✓ Le gestionnaire des entrées/sorties
- ✓ La pile réseau

Méthodologie de développement

- ✓ Méthodes, outils et chaîne de développement croisé
- ✓ Génération d'une toolchain avec crosstool-ng, eldk
- ✓ Mise en place d'un mini-système de développement croisé à base de Makefiles
- ✓ Utilisation de scratchbox

Génération d'un rootfs

- Définition d'un système minimal
- Choix de librairies : LibC, glibc, uClibC, NewLibC, dietlibc, bionic (Android)
- Chargement dynamique de pilotes de périphériques
- Gestion des entrées du devfs (/dev)
- Générateurs de rootfs : Buildroot, ptxdist, OpenEmbedded, debian/gentoo embedded

Configuration utilisateur & réseau

- Gestion de sessions utilisateurs
- Installation des services réseaux : Inetd, Rsh, telnet, Nfs
- Connexion réseau : remote shell rsh, telnet ...

Debug & optimisation

- ✓ Conseils pour les développements ayant un usage embarqué
- ✓ Mise au point par port série, par réseau, par sonde JTAG
- ✓ Utilisation de la console série
- ✓ Cross-debugging
- ✓ Profiling
- ✓ Couverture de code
- ✓ Mesure de la consommation de ressources
- ✓ Benchmark

Environnement graphique

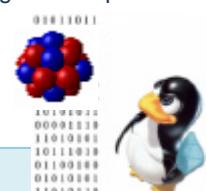
- ✓ nanoX (Tiny-X)
- ✓ XFree
- ✓ QtEmbedded/QTopia
- ✓ MicroWindows
- ✓ Framebuffer/directfb/fusion
- ✓ GTK/Embedded
- ✓ Serveur web embarqué

Introduction aux extensions Temps réel embarquées

- ✓ Temps réel souple: interface POSIX, extension API PThread , NPTL ...
- ✓ Temps réel dur: Patches low-latency, préemptifs, Temps Réel RTAI/RTLinux

Travaux Pratiques:

- ✓ Génération d'un toolchain
- ✓ Génération d'un rootfs
- ✓ Génération d'un firmware et flashage
- ✓ Mise en place d'un bootloader
- ✓ Debug & profiling sur une plateforme embarquée



Pré-requis : connaissance de la programmation Linux utilisateur et système

Formateur : plus de 12 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Ce stage permet aux développeurs en informatique, confrontés aux problèmes de portage d'un noyau GNU Linux et d'applications sous cible Linux embarqué, de pouvoir concevoir un système GNU Linux sur mesure, optimisé à l'aide des différentes solutions de configuration Linux et de mettre en œuvre une chaîne de développement croisé complète. La formation est ponctuée de différents travaux pratiques pour rendre ces concepts plus appliqués.
- ✓ Cette formation GNU Linux embarqué aborde les concepts du portage de ce système sur cible embarquée par un tour d'horizon des différentes distributions.
- ✓ **Public visé : Développeurs Produits souhaitant acquérir une compétence sur Linux en environnement industriel contraint.**

Introduction

- ✓ Définition et problématique des systèmes embarqués
- ✓ Préparation du noyau
- ✓ Configuration et compilation du noyau
- ✓ Intégration de drivers

Développement d'applications embarquées

- ✓ Espaces utilisateur et noyau
- ✓ Organisation d'un environnement de développement
- ✓ Outils disponibles
- ✓ Utilisation de bibliothèques statiques et dynamiques

Développement de drivers

- ✓ Méthodes
- ✓ Mise en œuvre à travers un exemple

Outils Linux pour l'embarqué

- ✓ Librairies adaptées à l'embarqué (glibc, uClibc, dietlibc...)
- ✓ Bootloader (Uboot, Redboot...)
- ✓ IHM adaptées à l'embarqué (miniGUI, Qt/embedded, Nano-X...)
- ✓ Introduction à Busybox



Choix du type de système de fichiers

- ✓ Ramdisk/initrd
- ✓ Systèmes de fichiers adaptés aux flashs (JFFS2, SQUAFS)
- ✓ Systèmes de fichiers adaptés à l'embarqué (ROMFS, CRAMFS)
- ✓ Systèmes de fichiers compatibles "desktop" (EXT2, FAT)
- ✓ Systèmes de fichiers temporaires / RAM disks / swap
- ✓ Systèmes de fichiers NFS, systèmes diskless, protocole Bootp

Busybox

- ✓ Présentation
- ✓ Configuration et compilation
- ✓ Lancement de programmes et services au démarrage du système

Création du système de fichiers racine

- ✓ Périphériques, programmes, librairies ...
- ✓ Installation des modules
- ✓ Recherche et installation des librairies dynamiques nécessaires
- ✓ Vérification de la cohérence du système de fichiers
- ✓ Utilisation d'outils de génération de systèmes de fichiers racine embarqués (OpenEmbedded, BuildRoot)

Buildroot

- ✓ Présentation
- ✓ Installation, configuration
- ✓ Hiérarchie des répertoires
- ✓ Introduction à Kconfig
- ✓ Configuration de la chaîne de compilation et librairie C

Notions de temps réel

- ✓ Définition
- ✓ Distinction en temps réel souple et dur
- ✓ Problématique d'ordonnancement
- ✓ Disciplines d'ordonnancement disponibles sous Linux

Travaux Pratiques:

- ✓ Chaque chapitre est étayé par des travaux pratiques permettant de mettre en œuvre les notions abordées.
- ✓ Les travaux pratiques sont constitués
 - d'un squelette de code permettant de réaliser l'exercice
 - d'une correction
- ✓ Un système de fichiers racine sera réalisé manuellement puis avec BuildRoot.

Cible embarquée: carte ARM9

Pré-requis : connaissance de la programmation Linux utilisateur et système

Formateur : plus de 12 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Ce stage aborde l'ensemble des problématiques temps réel en environnement GNU/Linux, des couches basses du noyau jusqu'aux applications utilisateur. Les API Linux et POSIX-RT sont étudiées.
- ✓ Les travaux pratiques intègrent l'utilisation des outils de l'environnement GNU/Linux pour la mise au point et l'analyse du temps réel.
- ✓ **Public visé : Développeurs souhaitant utiliser Linux pour des applications temps réel.**

✓ **Les problématiques du Temps Le temps-réel dans Linux Réel**

- ✓ Définitions et concepts
- ✓ Systèmes « classiques » contre systèmes temps réel : les différences, les limites
- ✓ Latence vs débit
- ✓ Illustrations

Présentation de GNU/Linux

- ✓ Bref historique
- ✓ Architecture générale du noyau
- ✓ Configurer et installer Linux
- ✓ Prise en charge croissante de la problématique temps réel

Fonctionnement du noyau

- ✓ Mode utilisateur, mode noyau
- ✓ Contextes d'exécution
- ✓ Le temps système dans Linux
- ✓ Mécanismes d'interruptions
- ✓ Processus et tâches : cycle de vie, priorités
- ✓ Les ordonnanceurs
- ✓ Verrous et préemption du noyau
- ✓ L'architecture des drivers
- ✓ Gestion de la mémoire
- ✓ Les appels système
- ✓ Systèmes multiprocesseurs
- ✓ Systèmes embarqués et optimisation de la consommation

Linux et la mesure du temps

- ✓ Horloges et temps système
- ✓ Les timers et la génération d'événements
- ✓ Les timers de haute résolution
- ✓ Résolution et précision
- ✓ Les registres de performances
- ✓ API noyau et API utilisateurs POSIX

- ✓ Historique de l'intégration progressive du TR dans le noyau Linux
- ✓ Limites actuelles de Vanilla-Linux
- ✓ Présentation du Patch-RT
- ✓ Patch-RT et interruptions
- ✓ Patch-RT et ordonnancement
- ✓ Exemples
- ✓ Limitations
- ✓ Le futur : ordonnanceur EDF

Concevoir temps réel en mode utilisateur

- ✓ La programmation temps -réel dans l'espace utilisateur
- ✓ APIs Linux
- ✓ API Posix RT
- ✓ Tâches, priorités , signaux, timers, mutex

Concevoir temps réel en mode noyau

- ✓ La programmation temps réel dans l'espace noyau
- ✓ Les tâches dans le noyau
- ✓ Drivers, interruptions et temps réel
- ✓ Softirqs, tasklets, workqueues...
- ✓ Gestion de la concurrence

Mise au point des applications temps réel

- ✓ Optimiser le noyau pour le temps réel
- ✓ Configurer, utiliser les mécanismes d'analyse statistique du noyau
- ✓ Détection, localisation et analyse des latences et des verrous
- ✓ Outils système utiles pour l'analyse des caractéristiques temps réel

Approches alternatives

- ✓ Présentation des architectures alternatives: virtualisation, hyperviseurs,micro-kernel
- ✓ Rtai, xenomai, rt-Linux, montavista...
- ✓ présentation des principes et des APIs.

Travaux Pratiques:

- ✓ Préparation, configuration et installation d'un noyau Linux temps-réel avec le patch-RT
- ✓ Ecriture de tâches périodiques en espace utilisateur, puis en espace noyau
- ✓ Manipulation d'outils systèmes variés
- ✓ Manipulation des attributs temps réel de la tâche (ordonnancement, priorités)
- ✓ Ecriture d'un gestionnaire d'interruption temps réel dans l'espace noyau.
- ✓ Mesure et analyse des jitters et latence : utilisation des outils statistiques du noyau
- ✓ Comparaison du comportement temps réel entre un noyau classique et un noyau optimisé pour le temps réel



Pré-requis : connaissance de la programmation Linux utilisateur et système

Formateur : plus de 12 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Ce stage aborde les principaux concepts systèmes utilisés dans le développement d'une application Linux (appels système et bibliothèques dont principalement la LibC). Il traite notamment du multi-tâches sous Linux (processus/thread), de la gestion de la mémoire, de la communication et de la synchronisation inter-processus (signaux, files de message, mémoire partagée, sémaphores et mutex, tubes de communication), de la gestion des entrées/sorties et de la programmation réseau.
- ✓ L'ensemble du cours est étayé de réalisations pratiques permettant aux stagiaires d'assimiler de manière concrète les thématiques abordées.
- ✓ **Public visé : Développeurs et chefs de projets souhaitant obtenir une vision d'ensemble sur le développement d'une application Linux.**

Introduction

- ✓ Historique Unix
- ✓ Présentation de Linux
- ✓ Licences (GPL, LGPL, BSD)
- ✓ Distributions, gestion de paquets

Signaux

- ✓ Présentation des signaux
- ✓ Liste des signaux sous Linux
- ✓ Émission d'un signal
- ✓ Réception des signaux
- ✓ Blocage des signaux

Gestion de la mémoire

- ✓ Fonctions d'allocation
- ✓ Fonction de libération
- ✓ Règles de sécurité
- ✓ Débogage des allocations mémoire
- ✓ Verrouillage de pages en mémoire
- ✓ Protection de l'accès à la mémoire

Fichiers

- ✓ Structure du système de fichiers
- ✓ Fichiers et devices
- ✓ Descripteurs de fichiers, E/S standards
- ✓ Fonctions de bas niveau
- ✓ Fonctions de la librairie standard d'E/S
- ✓ Verrous et autres fonctions

Tubes

- ✓ Présentation des tubes
- ✓ Tubes entre processus
- ✓ La fonction pipe
- ✓ Utilisation lors d'un fork()
- ✓ Les tubes nommés

Sockets

- ✓ Présentation et concepts
- ✓ Domaines, types et protocoles
- ✓ Création d'un socket
- ✓ Affectation d'adresse
- ✓ Mode connecté et mode non connecté
- ✓ Attente de connexions
- ✓ Demander une connexion
- ✓ Fermeture d'un socket
- ✓ Recevoir et envoyer des données
- ✓ Options des sockets

Processus

- ✓ Présentation et concepts
- ✓ Attributs et identification des processus
- ✓ Création de processus
- ✓ Attente d'un processus
- ✓ Terminaison de processus

Mémoire partagée

- ✓ Présentation et concepts
- ✓ Création d'un segment de mémoire partagée
- ✓ Connexion et déconnexion au segment

Travaux Pratiques:

- ✓ Chaque chapitre est étayé par des travaux pratiques permettant de mettre en œuvre les notions abordées
- ✓ Les travaux pratiques sont constitués :
 - d'un squelette de code permettant de réaliser l'exercice
 - d'une correction

Threads

- ✓ Présentation et concepts
- ✓ Threads vs. processus
- ✓ Création de threads
- ✓ Terminaison de threads
- ✓ Synchronisation, sections critiques, mutex

Sémaphores

- ✓ Présentation et concepts
- ✓ Création d'un sémaphore
- ✓ Opérations sur un sémaphore
- ✓ Contrôle et destruction des sémaphores

Pré-requis : Connaissance minimale en environnement desktop Linux et programmation

Formateur : plus de 12 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Complément de la formation « Linux Système », ce stage aborde les principaux concepts utilisés dans le développement d'un système Linux embarqué. Il traite notamment de la génération du système de fichiers racine.
- ✓ L'ensemble du cours est étayé de réalisations pratiques permettant aux stagiaires d'assimiler de manière concrète les thématiques abordées.
- ✓ **Public visé : développeurs et chefs de projet souhaitant obtenir une vision d'ensemble sur le développement d'une application Linux.**

Introduction

- ✓ Problématique des systèmes embarqués
- ✓ Les outils Linux pour l'embarqué
- ✓ Les bibliothèques adaptées à l'embarqué (glibc, uClibc, dietlibc...)
- ✓ Les bootloaders (Uboot, Redboot...)
- ✓ Les IHM adaptées à l'embarqué (miniGUI, Qt/embedded, Nano-X...)
- ✓ Introduction à Busybox

Création du système de fichiers racine

- ✓ Pérophériques, programmes, bibliothèques ...
- ✓ Installation des modules
- ✓ Recherche et installation des bibliothèques dynamiques nécessaires
- ✓ Vérification de la cohérence du système de fichiers
- ✓ Utilisation d'outils de génération de systèmes de fichiers racine embarqués (OpenEmbedded, BuildRoot)

Travaux Pratiques:

- ✓ Chaque chapitre est étayé par des travaux pratiques permettant de mettre en œuvre les notions abordées. Les travaux pratiques sont constitués
 - d'un squelette de code permettant de réaliser l'exercice
 - d'une correction
- ✓ Un système de fichiers racine sera réalisé manuellement puis avec BuildRoot

Choix du type de système de fichiers

- ✓ Ramdisk/initrd
- ✓ Systèmes de fichiers adaptés aux flashes (JFFS2, SQUAFS)
- ✓ Systèmes de fichiers adaptés à l'embarqué (ROMFS, CRAMFS)
- ✓ Systèmes de fichiers compatibles "desktop" (EXT2, FAT)
- ✓ Systèmes de fichiers temporaires / RAM disks / swap
- ✓ Système de fichiers NFS, système diskless, protocol bootp
- ✓ Configuration du kernel

Buildroot

- ✓ Présentation
- ✓ Installation, configuration
- ✓ Hiérarchie des répertoires
- ✓ Introduction à kconfig
- ✓ Configuration de la chaîne de compilation et librairie C
- ✓ Sélection de paquets
- ✓ Configuration du kernel
- ✓ Configuration du système de fichiers et bootloader
- ✓ Utilisation de BuildRoot
- ✓ Répertoires générés
- ✓ Ajout de nouveaux paquets (basés ou non sur autotools)
- ✓ Comparaison BuildRoot vs. Open Embedded

Busybox

- ✓ Présentation
- ✓ Configuration et compilation
- ✓ Lancement de programmes et services au démarrage de Busybox



Pré-requis : un minimum de connaissances en environnement desktop Linux et programmation

Formateur : plus de 12 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Cette formation aborde en profondeur le marché Linux, les enjeux techniques, les risques organisationnels, les risques liés à la formation des équipes et les risques juridiques. Elle permet d'obtenir une vision technique globale sur tous les aspects Linux embarqué et temps réel.
- ✓ **Public visé :** Cette formation est dédiée aux chefs de projet, directeurs de projet, responsables de laboratoire R&D, intégrateurs, responsables validation souhaitant acquérir une vision globale sur tous les aspects Linux embarqué et temps réel.

1- Présentation d'un système embarqué

- ✓ Définitions
- ✓ Contraintes techniques
- ✓ Domaines d'application

2- Historique de Linux

3- Marché de l'embarqué

- ✓ Parcours de l'offre
- ✓ Positionnement de Linux
- ✓ Intérêt de Linux

4- Support d'exécution et architecture

- ✓ Espaces utilisateur et noyau
- ✓ Système de fichiers
- ✓ Notion de processus et de thread
- ✓ Description du processus de boot

5- Configuration et génération du noyau

6- Mise en oeuvre de Linux pour l'embarqué

- ✓ Chaîne de développement
- ✓ Choix, utilisation/production d'une distribution

7- Développement d'applications

- ✓ API disponibles
- ✓ Outils disponibles
- ✓ Production de code
- ✓ Débogage et profilage de code
- ✓ Gestion de versions de code

8- Aspects temps réel

- ✓ Définition du temps réel
- ✓ Notions de tâches et d'échéance
- ✓ Distinction entre temps réel dur et mou
- ✓ Introduction à l'ordonnancement temps réel

9- Aspects juridiques

- ✓ Licences "open source"
- ✓ Licences libres

10- Synthèse

Pré-requis : pas de pré-requis

Formateur : plus de 15 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Cette formation vous permettra d'acquérir toutes les notions pour maîtriser les autotools.

- ✓ **Public visé :** **Cette formation est dédiée aux développeurs et aux intégrateurs.**

- ✓ Autotools introduction
- ✓ Mastering the buildprocess
- ✓ Quick GNU/Make reminder
- ✓ The autotools process
- ✓ Using the autoscan tool
- ✓ The autogen.sh script
- ✓ Using libtool
- ✓ The configure.{ac/in} template
- ✓ The Makefile.am template
- ✓ Using the library's template
- ✓ Using the application's template
- ✓ The config.h support
- ✓ The silent/verbose support
- ✓ Managing dependencies with pkg-config
- ✓ Generating a documentation

- ✓ Launching unitary test
- ✓ Managing the debug mode
- ✓ Managing the optimization support
- ✓ Managing the profiling support
- ✓ Cross-compilation
- ✓ M4 macros

Travaux Pratiques:

- Cette formation est étayée par des travaux pratiques permettant de mettre en œuvre les notions abordées.



Pré-requis : de préférence, un minimum de connaissances dans l'utilisation d'un système Unix et des GNU/make

Formateur : plus de 15 ans en expertise Linux et open source

Objectif de la formation:

- ✓ Cette formation permet à des équipes ayant travaillé en C de bien comprendre l'apport du C++ et son utilisation.
- ✓ **Public visé :** Cette formation est dédiée aux développeurs ayant développé en C et souhaitant basculer vers le développement C++.

Introduction et concepts

- ✓ Problèmes avec programmation non OO
- ✓ Types abstraits
- ✓ Objets, classes
- ✓ Encapsulation, relations, compositions, agrégations
- ✓ Abstraction, héritage et polymorphisme
- ✓ Data hiding versus access protection

Le langage C++

- ✓ Comparaison avec C (C89 et C99)
- ✓ Fichiers, headers, préprocesseurs, compilation
- ✓ Librairies standards C

Programmation C++

- ✓ Opérateurs
- ✓ << et >>
- ✓ new et delete
- ✓ Précédence
- ✓ Confusion array – initialisation
- ✓ Opérateur ->

Constantes, variables, références

- ✓ const, #define
- ✓ références

Scope et namespaces

- ✓ Scope boucle for
- ✓ Types static et extern
- ✓ namespace et using namespace

Structures, union et classes

- ✓ Public versus private
- ✓ Structures versus unions versus classes
- ✓ Classes et membres

Constructeurs et destructeurs

- ✓ Initialisation d'objets
- ✓ Destruction d'objets
- ✓ Liste d'initialisation
- ✓ Self initialisation

Paramètres par défaut

Fonctions inline

Surcharge de fonctions

Classes abstraites

- ✓ Fonctions et classes friend
- ✓ Classes virtuelles et pures
- ✓ Destructeur virtuel
- ✓ Constructeur de copie
- ✓ Copie shallow versus deep
- ✓ Assignations virtuelles
- ✓ Table virtuelle

Surcharge des opérateurs

Design de classes

Strings

- ✓ std::string et std::wstring
- ✓ Construction et destruction
- ✓ Opération sur strings (longueur, capacité, accès, assignations, ajouts, insertions ...)

I/O, opérations sur fichiers

- ✓ Input et output streams
- ✓ istream, ostream, ios
- ✓ stringstream
- ✓ I/O sur fichiers

STL

- ✓ Containers (vector, list, map, deque...)
- ✓ Itérateurs
- ✓ Allocateurs
- ✓ Functors
- ✓ Algorithmes

Programmation C++ avancée

- ✓ Conversions, casts
- ✓ Le pointeur this
- ✓ Membres statiques
- ✓ Templates
- ✓ Exceptions
- ✓ Auto et smart pointers
- ✓ RTTI
- ✓ RAII
- ✓ Singletons
- ✓ Boost
- ✓ C++0x

Pré-requis : Bonne connaissance en développement C sous Linux

Formateur : plus de 15 ans en expertise Linux et open source



4. Formations à venir courant 2011

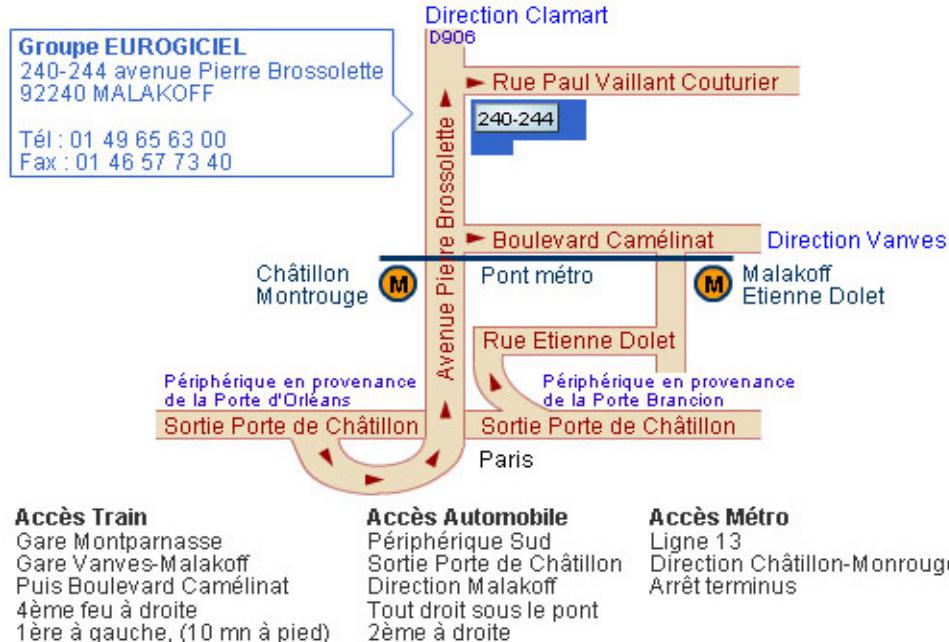
- **Programmation réseau sous Linux** (4 jours)
- **Développement d'application sous Android** (4 jours)
- **Intégration système sous Android** (4 jours)



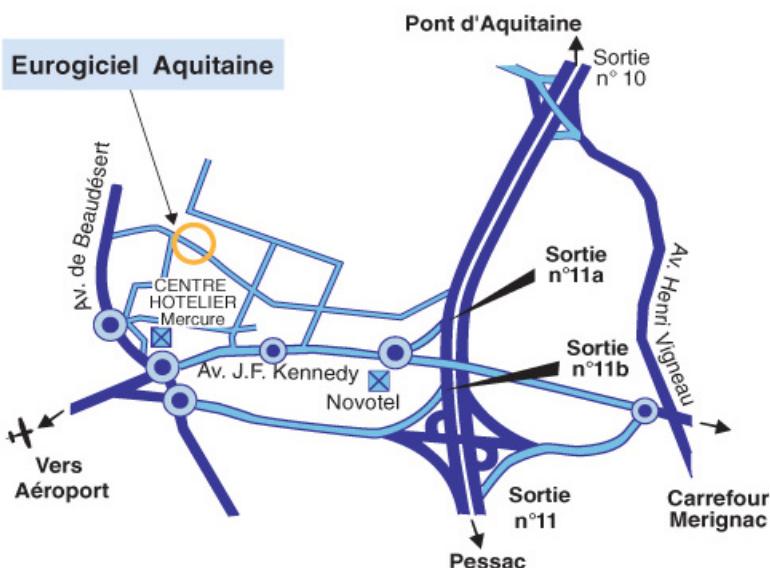
5. Plan d'accès à nos centres de formation

Paris

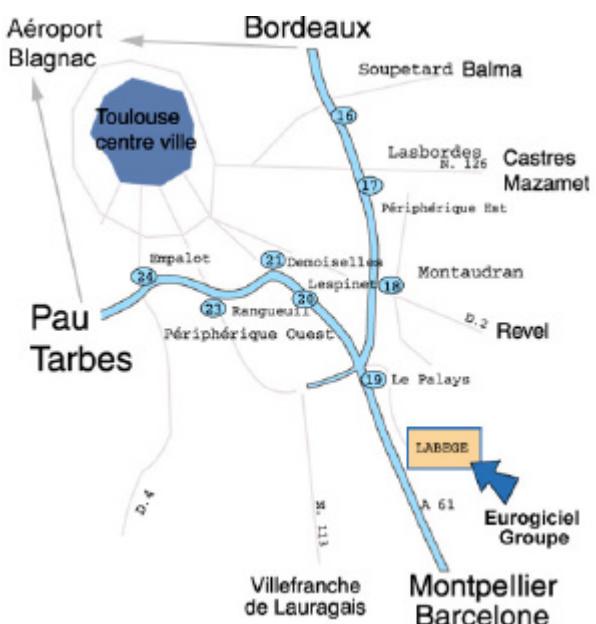
240-244 av. Pierre-Brossolette
92240 Malakoff



Bordeaux
6, rue Henry-le-Châtelier, Domaine du Millénium
33700 Mérignac



Toulouse Rue Ampère 31675 LABEGE



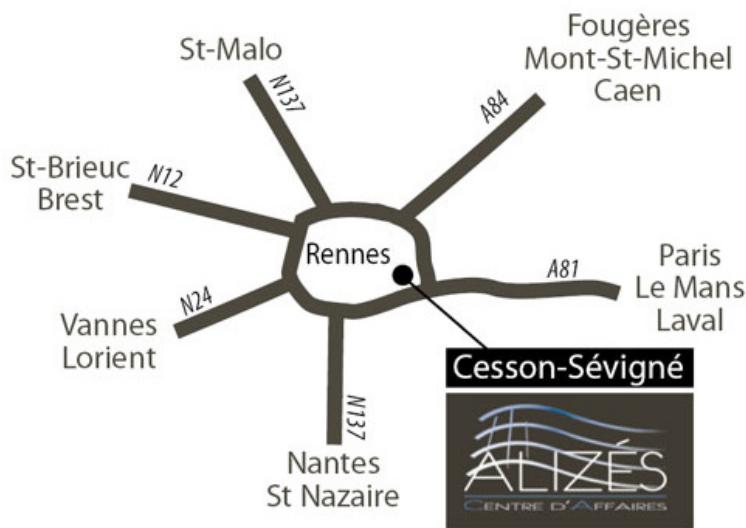
Nantes

104, Rue du Leinster, 2, Rue du Tyrol
Bât C / 1er étage
44 240 La Chapelle sur Erdre





Rennes
Centre d'Affaires Alizés
La Rigourdière
35 510 Cesson-Sévigné



Sophia-Antipolis
Frégate
2400, route des Crêtes
06560 SOPHIA-ANTIPOLIS





Marseille

Immeuble Optimum, Bât. C
165 Av. du Marin Blanc
ZI Les Paluds
13400 AUBAGNE





6. Conditions générales

EUROGICIEL accompagne ses clients au travers de ses formations Linux dispensées en intra- ou inter-entreprise sur le site du client ou dans nos centres de formation.

Informations Générales sur la formation intra-entreprise

La solution intra-entreprise sur site peut s'avérer très économique lorsque le nombre de participants est important.

Dans le cas d'une formation sur site, les stagiaires utilisent alors le matériel informatique de leur entreprise et évoluent dans un environnement qui leur est familier. L'environnement logiciel requis en fonction de la formation sera communiqué au client.

EUROGICIEL peut également fournir le matériel informatique (surcoût lors de la formation).

Assistance et conseil :

Le service de formation d'EUROGICIEL se tient à votre disposition afin d'adapter le contenu et la durée vos formations intra-entreprise. Au travers d'un audit gratuit nous définissons un contenu adapté à vos besoins afin de vous offrir une formation sur mesure optimisée. Nous proposons également le formateur le plus en rapport avec votre domaine d'activité.

Préparation et déroulement du stage intra-entreprise

Pour les formations sur le site du client ou choisi par le client, un minimum de conditions doivent être respectées :

- ✓ La salle de formation doit être dédiée au cours. Elle doit être équipée en conséquence en fonction du nombre de stagiaires. Il ne doit pas y avoir plus de 2 stagiaires par poste de travail. Les cours sont réalisés avec 10 stagiaires au plus.
- ✓ Equipements pédagogiques et de projection doivent être en état de fonctionnement.
- ✓ En aucun cas, les stagiaires ne seront dérangés pendant le cours.

Les frais et le temps de déplacement de l'animateur seront facturés en sus.



Conditions générales de la formation



Inscription :

Dans l'intérêt du stagiaire il est impératif de s'assurer d'un niveau minimum de connaissances pour suivre le cours choisi.

Les inscriptions doivent être adressées par courrier, télécopie ou courriel à EUROGICIEL à l'aide d'un bulletin d'inscription signé et portant le cachet de l'entreprise ou de l'organisme.

Les inscriptions doivent être reçues par EUROGICIEL au plus tard 15 jours avant le début du cours. Passé ce délai, l'inscription ne peut être garantie.

Les inscriptions signifient acceptation de nos conditions générales de formation. Toute clause différente émanant de nos clients ne nous serait pas opposable.

Convocation :

Une semaine avant le début de la formation, une convocation pour chaque participant est adressée par courrier au responsable formation de l'entreprise qui sera chargé de les communiquer aux stagiaires.

La convocation comporte toutes les informations utiles quant à l'organisation matérielle ainsi que le programme détaillé de la formation.

Annulation :

En cas d'annulation par l'entreprise de l'inscription de tout ou partie des personnes inscrites dans un délai inférieur à 10 jours avant la date de début de formation, l'entreprise est tenue de payer la somme correspondant aux dépenses effectivement réalisées ou engagées par le dispensateur de la formation, notamment les frais administratifs entraînés par l'inscription du (ou des) salarié(s) au stage.

Toute annulation doit nous être notifiée par courrier ou par télécopie 10 jours ouvrables avant le début du cours. Pour toute annulation parvenant moins de dix jours ouvrables avant la date du début de la formation, les frais de participation seront dus en totalité

Les horaires :

L'accueil des participants s'effectue à partir de 9h15

Les cours débutent à 9h30 et se terminent à 17h00.



Les repas :

Une pause repas d'une heure et demie est prévue de 12h30 à 14h.

Les prix :

Les prix sont indiqués Hors Taxes et sont à majorer de la TVA au taux applicable en vigueur à la date de facturation. Les prix sont forfaitaires et incluent la remise d'un support de cours. Ils ne comprennent pas les déjeuners. Les frais de trajet et de séjour restent à la charge du client.

Matériel pédagogique :

EUROGICIEL conserve l'intégralité des droits d'auteur relatifs au contenu des formations et des supports de formation. Toute reproduction, modification ou divulgation à des tiers de tout ou partie de ces formations et/ou documents, sous quelque forme que ce soit, est interdite sans l'accord écrit préalable d'EUROGICIEL. EUROGICIEL se réserve le droit de modifier ou d'adapter le programme des formations en fonction de l'évolution des informations techniques.

Conditions de Règlement :

La facture est établie à l'issue de la formation.

Le règlement s'effectue à réception de la facture.

Toute somme non payée à l'échéance figurant sur la facture entraîne l'application des pénalités d'un montant égal à une fois et demi le taux d'intérêt légal. Ces pénalités seront exigibles sur simple demande de la société EUROGICIEL.

Toute formation commencée est due intégralement.

Convention de formation :

A la réception du bulletin d'inscription, EUROGICIEL vous communiquera une convention de formation qu'il faudra lui renvoyer signée avec le cachet de la société.

Une attestation de présence sera jointe à la facture.



Le dispensateur de la formation

Les formations sont dispensées par « SAS ETOP international » filiale du groupe EUROGICIEL.

SAS ETOP INTERNATIONAL

Groupe EUROGICIEL

Rue AMPERE

BP 47506

31675 LABEGE CEDEX

N° SIRET : 405 345 471 000 16

Enregistré sous le numéro d'organisme de formation : 73 31 05015 31 auprès du Préfet de la région Midi-Pyrénées



Pour vous inscrire ou pour tout renseignement:

Pour vous inscrire :

Renvoyer le bulletin d'inscription ci-contre dûment rempli, signé, avec le cachet de la société.

Par fax :

+33 (0)2 51 13 94 38

Par courrier :

EUROGICIEL
Service Formation

Parc Tertiaire du Vieux Moulin
Bat C
2, rue du Tyrol - 104, rue de Leinster
44240 LA CHAPELLE SUR ERDRE

Par courriel :

formations@eurogiciel.fr

Pour tout renseignement technique sur la formation :

Patrick.leprince@eurogiciel.fr



7. BULLETIN D'INSCRIPTION

ENTREPRISE

Société :

Contact formation : Email :

Adresse :

Code postal : Ville :

Téléphone : Fax :

ADRESSE DE FACTURATION

Société :

Contact:.....

Email :

Adresse si différente :

Code postal : Ville :

Téléphone : Fax :

STAGES COMMANDES

Nom Prénom Stagiaire	Identifiant du cours	Dates	Prix (en €)

J'accepte les conditions générales de vente des formations EUROGICIEL décrites dans le Catalogue des Formations

Bon pour accord, le à,.....

Cachet de l'entreprise :

Signature :

Nom du signataire

