

Master HES of Sciences in Engineering

MRU hepia TIC

Les systèmes à grande échelle

Février 2010

I. Introduction

L'informatique du début du 21^{ème} siècle, qu'elle soit visible (système d'information d'une entreprise) ou enfoui dans un processus industriel (voiture, avion, etc.) est répartie «par nature». Les systèmes informatiques de nos jours sont par essence distribués, ils sont souvent composés de «sites» (processeurs, capteurs, ordinateurs, émetteurs, etc.) reliés en réseaux. Ces sites sont caractérisés par:

- une distribution géographique étendue,
- une hétérogénéité et mobilité des composants (PC, PDA, téléphones, capteurs, etc.),
- une volatilité et une disponibilité partielle.

La particularité essentielle de ces systèmes à grande échelle est la quantité considérable d'informations stockée, échangée et traitée. Les exemples suivants sont des cas typiques de systèmes à grande échelle.

- Dans le domaine de l'informatique industrielle et embarquée (réseaux ad hoc, contrôle de processus industriels : voitures, avions, etc.), les systèmes de contrôles et de collecte de données sont souvent composés de plusieurs modules autonomes qui s'échangent les données via divers réseaux.
- Dans le domaine des télécommunications et des transmissions de données multimédia, le recours à des architectures décentralisées et distribuées pour éviter les problèmes de congestion et de surcharge est de plus en plus courant.
- Dans le domaine du calcul scientifique et de haute performance (biotechnologie, chimie, physique, traitement d'images, etc.), les machines massivement parallèles et centralisées ont cédé la place à des configurations largement distribuées et hétérogènes.
- Dans le domaine du tertiaire (banques, assurances, etc.), les systèmes d'informations sont distribués et hétérogènes tant sur le plan des architectures matérielles que logicielles.

La dimension d'un système distribué à grande échelle est mesurée en fonction de plusieurs facteurs : taille du code, quantité de données stockées, accédées et manipulées, connexions et interdépendance entre les composants du système, hétérogénéité et/ou interopérabilité du ou des réseaux utilisés, intégrité des données, etc.

Cette vision globale des systèmes à grande échelle est le fil conducteur des modules d'approfondissement proposés par *hepia* dans le cadre du master HES-SO, en ce qui concerne les technologies de l'information et de communication (TIC).

L'objectif de ces modules est de maîtriser les concepts, méthodologies, techniques et outils permettant de spécifier, concevoir et mettre en place des systèmes distribués à grande échelle *fiables, sécurisés et performants*. Cette démarche correspond à un besoin identifié par les entreprises du tissu économique et industriel, non seulement en Suisse mais partout en Europe.

II. Modules d'approfondissement

Dans le cas du master HES-SO, six modules d'approfondissement TIC sont proposés par *hepia* pour l'année universitaire 2010/2011. Ils sont décrits dans les paragraphes qui suivent.

II.1 Virtualisation des services et cloud computing

Professeurs responsables : Gérald Litzistorf, Nabil Abdennadher

Nombre ECTS : 3

Présentation générale

Le principe de la virtualisation est d'implémenter plusieurs machines virtuelles sur la même infrastructure physique. Cette tendance s'explique d'abord avec des arguments économiques (matériel, énergie, salle machines, etc.), puis par des facilités de gestion telles que clone d'une machine virtuelle et/ou image instantanée (*snapshot*) utilisée en production ou en phase de test.

Remplacer un parc de serveurs par une architecture virtualisée est un passage obligé pour les entreprises comme le soulignent des analyses récentes de Gartner ou Forrester. L'analyse des performances de ces systèmes est essentielle pour maintenir les niveaux d'exigence initiaux (*Service Level Agreement*) ; elle devient même vitale

pour consolider un maximum de systèmes où il devient impératif d'optimiser l'utilisation des processeurs, des mémoires, des systèmes de stockage et des réseaux.

La disponibilité des services est un autre paramètre critique qui trouve des solutions efficaces dans des infrastructures redondantes et distribuées avec des technologies telles que VMotion, Distributed Resource Scheduling, Fault Tolerance ou Site Recovery.

Le développement et la maturité des technologies de virtualisation a permis l'émergence du concept de *cloud computing* qui a été adopté par les géants de l'informatique (IBM, Google, Amazon, Microsoft, etc.). Celui-ci consiste à utiliser les ordinateurs et les serveurs répartis dans le monde entier, pour exécuter une application (service) appartenant à un utilisateur quelconque. L'idée est de migrer les données et les traitements de nos ordinateurs «privés» vers de gigantesques centres de traitement. Ces architectures contribuent au développement de ce qui est communément appelé aujourd'hui : *greenIT*. La compétition s'annonce dure même si le modèle n'est pas encore mature.

Selon la vision du *cloud computing*, une entreprise ne serait plus propriétaire de son serveur informatique mais accèdera à de nombreux services en ligne sans avoir à gérer l'infrastructure sous-jacente, souvent complexe. Dans ce contexte, un langage de haut niveau généralement offert par le fournisseur d'infrastructure (IBM, Microsoft, Google, etc.) permet de «traduire» les besoins de l'utilisateur sous forme d'un ensemble de machines virtuelles qui seront déployées sur le cloud.

L'objectif de ce module est de :

- Maîtriser les outils et technologies associées à la virtualisation
- Découvrir certains langages cloud proposés par des intervenants comme Google, Microsoft, IBM, etc.

Contenu

- Principes de base de la virtualisation
- Virtualisation de la DMZ (web server – switch – firewall)
- Mécanisme de redondance des liens physiques (fail over & load balancing)
- Production d'une VM à partir d'un serveur physique (P2V = PhysicalToVirtual)
- Profils de sécurité des rôles d'administration
- Optimisation et monitoring des ressources
- Interfaces de programmation
- Fonctions évoluées : VMotion, Distributed Resource Scheduling, Fault, Site Recovery,
- Introduction au concept *cloud computing*
- Etude comparative des plateformes cloud disponibles sur le marché
- Etude de cas : déploiement d'une application type sur une plateforme de cloud computing

Organisation du module : théorie : 50%, travaux pratiques : 50%

Références :

- Katarina Stanoevska-Slabeva, Thomas Wozniak. Grid and Cloud Computing, a business perspective on technology and applications. Santi Ristol editors, Springer 2010.
- <http://www.tdeig.ch/vmware/liens.pdf>
- D'autres références seront données au début du cours

II.2 Systèmes parallèles et distribués à grande échelle

Professeurs responsables : Michel Lazeyras, Paul Albuquerque, Nabil Abdennadher

Nombre ECTS : 6

Présentation générale

Plusieurs familles d'applications informatiques sont consommatrices de temps processeurs, de mémoire centrale et d'espace disque dur. Elles concernent des domaines comme la biotechnologie, la chimie, la physique, le traitement d'images satellitaires et médicales, l'optimisation des ressources, la simulation, les applications bancaires et financières, etc. Les alternatives proposées pour pallier aux problèmes liés à la complexité des traitements, au stockage de grandes quantités de données et/ou de la recherche et indexation dans ces données sont souvent basées sur la notion de distribution et de parallélisation.

Ce module s'intéresse à l'étude des architectures matérielles et logicielles des systèmes distribués à grande échelle visant à supporter des applications intensives de traitement (processeur) et/ou de données. Trois grands chapitres seront abordés :

1. Les systèmes distribués à grande échelle (SDGE) : ces systèmes sont capables de déployer des applications de haute performance sur des infrastructures géographiquement distribuées. Les plateformes Grid, calcul volontaire et peer-to-peer constituent l'exemple parfait des architectures ciblées par ce chapitre. L'objectif est d'étudier la structure de telles plateformes ainsi que leurs modèles de programmation.
2. Les systèmes massivement parallèles : les supercalculateurs sont souvent des clusters de PC ordinaires interconnectés par un réseau performant. Depuis quelques années, une nouvelle tendance est en train de voir le jour : il s'agit d'utiliser la carte graphique (GPU) de son propre PC comme coprocesseur massivement parallèle. Toutefois, les modèles de programmation d'un cluster sont différents de ceux d'un GPU. L'objectif de ce chapitre est de présenter les architectures parallèles GPU et clusters, d'étudier la parallélisation d'algorithmes sur ces architectures et leurs outils de programmation.
3. Les systèmes de base de données répartis et distribués : Il s'agit d'étudier, l'accès à différentes bases de données gérées par des SGBD hétérogènes. Ces systèmes doivent supporter la cohérence et l'autonomie des sites locaux, la reprise en cas de panne sur un site local, la gestion des transactions réparties ainsi que la redondance des données. Les sites locaux sont souvent clients et serveurs en même temps.

Chacun de ces trois chapitres porte sur les concepts et fondements de l'architecture concernée, les technologies associées ainsi que des applications types issues en particulier des domaines des sciences du vivant, de la médecine et de l'imagerie.

Contenu

Chapitre 1 : Systèmes distribués à grande échelle (SDGE)

- Architecture des SDGE : composants d'un SDGE, gestion des ressources, gestion des données
- Sécurité des SDGE : authentification AAI, Certificats X509, certification des résultats.
- Modèles de programmation : comment décomposer/paralléliser une application sur des SDGE ?
- Etudes de cas : gLite, Advanced Resource Architecture (ARC), Condor, XtremWeb-CH, Berkely Open Infrastructure for Network Computing (BOINC)
- Etude de deux applications industrielles
 - o Optimisation de l'affectation de personnel (entreprise opérant dans le domaine du développement de logiciels pour compagnies aériennes),
 - o Construction de cartes neuronales fonctionnelles (Hôpitaux universitaires de Genève)

Chapitre 2 : Systèmes massivement parallèles

- Concepts et fondements du calcul parallèle : gain et performances
- Modèles de programmation : échange de messages, mémoire partagée, parallélisme de données
- Architectures : cluster de PC, Graphics Processing Unit (GPU)
- Implémentation d'algorithmes avec les bibliothèques CUDA sur GPU et MPI sur cluster de PC (algorithmes de tris, sur les graphes, d'algèbre linéaire)
- Application en bio-informatique : alignement de séquences de protéines

Chapitre 3 : Bases de données distribuées

- Les transactions ACID et leur mise en œuvre dans les SGBD
- Le contrôle de la concurrence des transactions
- Etude de protocoles d'ordonnement des transactions dans le contexte des systèmes distribués
- Classification des types d'applications parallèles/distribués gérées par les SGBD

Organisation du module : théorie : 50%, travaux pratiques : 50%

Références :

- Vipin Kumar, Ananth Grama, Anshul Gupta, and George Karypis. Introduction to Parallel Computing: Design and Analysis of Algorithms. 2nd edition, Addison Wesley, 2003.
- Barry Wilkinson and Michael Allen. Parallel Programming: Techniques and Applications using Networked Workstations and Parallel Computers. Prentice Hall Inc, New Jersey, 1999.
- Ian Foster. Designing and Building Parallel Programs. Addison-Wesley, 1995.
- CUDA Programming Guide. NVIDIA http://developer.nvidia.com/object/cuda_2_3_downloads.html
- David Kirk, Wen-mei Hwu. Programming Massively Parallel Processors: A Hands-on Approach. NVIDIA, 2010.
- Ian J. Taylor, Andrews B. Harrison. From P2P and Grids to Services on the Web. Evolving distributed communities. Springer 2009.
- Thierry Priol, Marco Vanneschi. From Frids to Service and Pervasive Computing, Springer 2008.
- Ian J. Taylor and all. Workflows for e-Science. Springer 2008

II.3 Réseau de confiance et identité numérique

Professeur responsable : Tewfiq El Maliki

Nombre ECTS : 3

Présentation générale

L'identité numérique et les réseaux de confiance seront l'un des sujets essentiels dans les années à venir dans le développement de la sécurité aussi bien dans le domaine des systèmes d'information que sur le réseau internet, et notamment les sites transactionnels en ligne qui imposent l'authentification et la certification. Ces réseaux constituent une nouvelle voie prometteuse pour assurer la sécurité des systèmes à grande échelle.

L'extension de l'Internet aux réseaux domestiques et mobiles, la multiplication des formes de connexion, l'indépendance entre accès, terminal et localisation, fera de la personne le principal point de convergence des réseaux et services.

La construction des identités, leur gestion, leur degré d'autonomie vis à vis de leur générateur, ou encore leur force probante dans le cas de transactions, vont être exposés avec un accent mis sur la mobilité et la confiance et sur l'étude des cas pratiques.

L'objectif est de ce module est de:

- Comprendre l'évolution future des identités numériques
- Comprendre la théorie de la confiance, de répudiation et de vie privée
- Etudier et implémenter un système basé sur la confiance
- Choisir et mettre en œuvre un système de gestion d'identités numériques surtout dans le cas de mobilité.

Contenu

Chapitre 1 : identité numérique

- Identité, vie privée
- Fédération d'identités
- Sécurité

Chapitre 2 : Confiance

- Réputation et confiance
- Machine de confiance
- Mise en œuvre d'un système de confiance

Chapitre 3 : gestion des identités

- Plateformes : Higgins, WFI, etc.
- Mobile identity management

Organisation du module : Théorie : 50%, Travaux pratiques : 50%

Référence :

- Digital identity (Phillip J. Windley)
- Computer and information security Morgan Kaufmann

II.4 Transmissions numériques audio et vidéo optimisées

Professeur responsable : Andrés Revuelta

Nombre ECTS : 3

Présentation générale

Le volume des transmissions de données numériques de nature audiovisuelles est en constante augmentation. Les domaines concernés représentent un vaste éventail d'applications aussi bien interactives (téléphonie, visiophonie) que simplement diffusées (TV et VoD).

Par exemple, la TV sur IP (IPTV) se développe à une allure vertigineuse (plus de 80 millions d'abonnés prévus en 2013). Les acteurs en présence, fournisseurs de programmes, distributeurs, opérateurs et en bout de chaîne, abonnés/télespectateurs, composent un système typique de réseau à grande échelle.

Les services audiovisuels sont de plus en plus gourmands en ressources (TV-HD, TV-3D, etc.). Par ailleurs, le grand public exige de ces services une «bonne» qualité. Le défi technologique consiste à assurer cette qualité sur les réseaux IP, sans les surcharger. Les problèmes qui en découlent sont importants :

- taux d'erreurs d'origines diverses importants ;
- encombrement du réseau ;
- sensibilité des transmissions aux contraintes de temps.

L'objectif de ce module est de fournir les outils nécessaires à la résolution des problèmes de conception, planification, mise en œuvre et maintenance des réseaux de diffusion multimédias.

Contenu

- Bases des techniques numériques audio et vidéo : codage, compression, transmission, TV numérique
- Protocoles du streaming : RTSP, RTP-RTCP, SIP
- Codage vidéo : MPEG, AVC/H.264
- Transmissions audio/vidéo d'utilisateur à utilisateur : architecture de systèmes VoIP, UC, CTI
- Transmissions audio/vidéo diffusées : architecture réseau DVB, VoD, TVoIP

Organisation du module : théorie : 50%, travaux pratiques : 50%

II.5 Modélisation formelle des systèmes réactifs et temps réel

Professeur responsable : Luigi Zaffalon

Nombre ECTS : 3

Présentation générale

Les systèmes réactifs exhibent une spécificité majeure constituée par des interactions multiples, complexes et souvent étriquées, entre :

- le système considéré et son environnement et,
- les parties du système lui même.

La fiabilité, la sûreté et le déterminisme, constituent donc des propriétés intrinsèques fondamentales des systèmes réactifs et temps réel.

Les applications concernées sont propres aux domaines des systèmes embarqués, du contrôle/commande de procédés industriels, aux transports (*automotive*, rail, avionique, etc.), au monitoring médical et environnemental, aux circuits intégrés, aux protocoles de systèmes répartis et au traitement intensif. En effet, dans le cas du non respect d'une quelconque contrainte des spécifications, les conséquences peuvent s'avérer, si ce n'est catastrophique, pour le moins coûteuses.

La réalisation basée sur des concepts et des formalismes mathématiquement fondés permet notamment, de valider préalablement le modèle (programme) en développement en rapport avec les spécifications, que ce soit du point de vue de l'exactitude du contrôle, des réactions et du respect des échéances temporelles. Le comportement du modèle doit être prédictible. La vérification formelle notamment, consiste à implémenter un algorithme permettant d'obtenir, pour une propriété souhaitée du système en développement, une preuve de sa satisfiabilité et, si ce n'est pas le cas, un contre-exemple illustrant l'erreur du système. Par exemple, s'il est possible de démontrer que les portes de l'ascenseur ne s'ouvriront pas lorsque celui-ci est en mouvement, alors dans le cas contraire un scénario sera fourni, illustrant la séquence d'événements et d'actions qui le prouveront.

Ce module porte essentiellement sur les thèmes suivants:

- Approches de modélisation asynchrones vs synchrones.
- Sensibilisation aux concepts et fondements de la vérification formelle.
- Validation de systèmes réactifs.
- Intégration avec des politiques d'ordonnancement temps réel.

Il permet de maîtriser les concepts spécifiques du domaine de la programmation de systèmes réactifs (matériels et logiciels) et ce à l'aide d'environnements de conception (*design*) appropriés. A cette fin, des travaux pratiques ainsi que des projets de laboratoire permettront d'amener l'apprenant aux méthodes, concepts et pratiques permettant de les réaliser et de les valider. Ils seront notamment en mesure de comprendre les différents paradigmes et d'effectuer les choix qui s'imposent.

Contenu

- Introduction aux problématiques de conception de systèmes fiables et sûrs.
- Evolution de la modélisation des systèmes réactifs temps réel.
- Modélisation synchrone, hypothèse d'instantanéité (traitement en temps nul: *zéro delay*), formalismes impératifs et à flots de données.
- Sensibilisation aux concepts et fondements des techniques de *model-checking*.
- Méthodologie de validation et de vérification formelle de systèmes réactifs logiciels (*software*) et/ou matériels (FPGA).
- Intégration avec des politiques d'ordonnancement temps réel et déploiement *multirate* sur plateformes *multicore*.

Organisation du module : théorie : 35%, travaux pratiques : 65%

Références :

- P. Schnoebelen et coll. *Vérification de logiciels. Techniques et outils du model-checking*. Vuibert, 1999.
- L. Zaffalon. *Programmation synchrone de systèmes réactifs avec Esterel et les SyncCharts*. PPUR, 2005, 488 pages.
- L. Zaffalon. *Méthodologie de conception de systèmes critiques*. Polycopié, 2010.

II.6 Logiques programmables pour systèmes complexes et performants

Professeurs responsables : René Beuchat, Fabien Vannel, Andrés Upegui

Nombre ECTS : 6

Certaines applications informatiques et de télécommunication nécessitent des performances élevées pour lesquelles un système informatique classique s'avère insuffisant ou trop coûteux. Le développement d'un système spécifique utilisant des logiques programmables peut apporter une solution pour des cas tels que:

- l'accélération du temps d'exécution d'un algorithme,
- le développement de réseaux de télécommunication à très haut débits
- le développement d'un système informatique embarqué avec unités de traitements spécialisées et spécifiques.

Ce type d'architecture peut se retrouver à divers niveaux : routeurs de données à haut débits, unités de traitements spécialisées (accélérateur de calcul), systèmes embarqués reliés par réseaux, etc.

Ce module s'intéresse à l'étude et à la mise en œuvre de composants programmables du type FPGA et à leur utilisation pour résoudre des problématiques complexes. Il concerne principalement le développement du matériel informatique (hardware), de l'électronique numérique (logique programmable à concevoir dans des langages de haut niveau, VHDL par exemple) et de la programmation de processeurs embarqués sur logiques programmables. Ces composants de dernières générations disposent de ressources spécifiques permettant de traiter des transferts de données à très haut débits et des algorithmes très complexes nécessitant un haut degré de parallélisme.

Dans le cas d'accélérateur pour des transferts de données rapides (de l'ordre des Giga bits/s) avec des protocoles dynamiques, les FPGA sont aujourd'hui un élément adaptatif de choix en comparaison aux composants à architectures spécialisées et fixes (ASIC).

Après avoir rappelé les notions de base des logiques programmables, ce cours abordera l'étude de tous les blocs spécifiques pouvant être utilisés dans les FPGA.

Les architectures comprennent fréquemment un ou plusieurs processeurs (systèmes multiprocesseurs) sur la même FPGA en utilisant des bus internes normalisés. Il est ainsi possible d'étendre facilement les fonctionnalités du circuit complet développé.

Toutes ces technologies sont étudiées en cours et mises en œuvre lors de laboratoires. Durant les laboratoires un système composé de plusieurs processeurs, mémoires, blocs de calculs DSP, interfaces séries à très hautes vitesses et interfaces programmables spécifiques seront étudiés. Ce cours s'appuiera sur des composants de dernières générations pouvant répondre à des besoins fortement exigeants pour lesquels un système à base d'ordinateur ne serait pas approprié.

Contenu

- Conception de systèmes embarqués sur FPGA
 - o architecture de systèmes sur FPGA avec processeurs *hardcore* et *softcore*,
 - o conception et réalisation d'unités accélératrices spécialisées, avec accès direct en mémoire et/ou à flot continu de données (*streaming*), cours et laboratoire associés sur FPGA (Altera).
- Conception de systèmes à haute performance en exploitant des ressources spécifiques des FPGAs.
 - o connections série à haute vitesse (Paires différentielles LVDS, Rocket IO, PCI-express, Gigabit Ethernet, etc.),
 - o blocs de calculs DSP,
 - o contrôle d'un système à différentes fréquences d'horloge,
- systèmes auto-reconfigurables (Xilinx - ICAP).

Organisation du module : théorie : 50%, travaux pratiques : 50%

Références : support de cours, liste de livres de référence distribuée en début de cours, documentations techniques ALTERA et XILINX.