

DISTRIBUIÇÃO LINUX EM UMA PLACA REMOVEDORA

Wesley Matos¹

Thales Will Silva Reis²

RESUMO

Este artigo proporciona uma introdução ao conceito de Computação na Nuvem. O mesmo aborda a definição deste novo formato de computação, características, arquitetura e infraestrutura necessária, aplicações, vantagens e desvantagens. O objetivo principal do projeto é proporcionar um sistema operacional ideal para quem vive conectado, com maior estabilidade, velocidade, simplicidade, segurança e sempre online, além de incluir uma interface de usuário simples e eficiente. Ele é de fácil utilização, com programas e arquivos armazenados em segurança nas nuvens, sem necessidade de hardware volumoso ou dispositivo de armazenamento extra, utilizando um computador de placa única de baixo custo. Desta forma toda esta engenhosidade ganhará movimento e força e se tornará acessível, pois será utilizado um sistema operacional para garantir a qualidade e eficiência da utilização desse novo modelo ainda pouco empregado. Sistema Operacional para um servidor pelo modelo Ubuntu Remasterizado – onde será instalado Remastersys em uma versão do Ubuntu, criado diversas contas de usuário e realizados algumas mudanças para tornar o original e adaptável ao projeto. Este modelo de utilização simplificado permite ainda que o usuário não tenha grandes dificuldades com instalação e manutenções, e ainda por efeito da flexibilidade, permite que os riscos atrelados à infraestrutura sejam minimizados. Outras flexibilidades incidem na facilidade de utilização dos serviços e compartilhamento de recursos, um dos pontos principais a ser abordados neste artigo. Abordam-se vantagens e desvantagens ao longo do artigo e demonstra que apesar de ainda pouco conhecida tem perspectiva de se tornar uma tecnologia comum usual em pouco tempo, pelo fato de que, cada vez mais, os serviços oferecidos pela Computação em Nuvem possibilitam que dispositivos com pouca capacidade de processamento e máquinas poderosas executem um mesmo aplicativo com a mesma eficiência.

PALAVRAS-CHAVE: Sistema Operacional. Computação nas nuvens. Placa removedora. Remasterização.

1 INTRODUÇÃO

¹ Graduado em Sistemas da Informação

² Orientador especialista em Análise de Sistema

A Computação em Nuvem tem se constituído nos tempos atuais e ao longo dos anos como uma importante plataforma de pesquisa que apresenta inúmeras vantagens e desafios estimulantes. Tarefas como alcance, compartilhamento, utilização e exploração de enorme quantidade de dados são absolutamente comuns no cenário atual, mas para isso a execução das mesmas demanda um grande volume de recursos, pois pode facilitar estas tarefas, à medida que pode disponibilizar de forma indefinida recursos de processamento, memória, armazenamento, dentre outros, para utilização imediata em um servidor com maior potencia.

A utilização destes recursos agrega uma série de vantagens para as organizações e usuários individuais, tendo em vista que preocupações com complexas como instalações e manutenções de infraestruturas deixam de existir e passam a ser de exclusiva responsabilidade dos provedores de serviços, além de possibilitar que os usuários se concentrem exclusivamente nas regras dos negócios que lhes são pertinentes.

Segundo Namjoshi, 2009, “A computação na nuvem ou *Cloud Computing*, simplificando, é uma nova forma de computação que permite ao usuário final acessar uma grande quantidade de aplicações e serviços de onde estiver e independente da plataforma, bastando para isso ter um terminal conectado à ‘nuvem’”.

O autor afirma ainda que a palavra nuvem insinua uma idéia de um ambiente não conhecido, ou seja, o mesmo podemos ver somente seu início e fim. Devido a isto esta plataforma tem sido muito bem empregada na nomenclatura deste novo modelo, onde toda a infraestrutura e recursos computacionais ficam “escondidos”, tendo o usuário o acesso apenas a uma interface padrão através da qual é disponibilizado todo o conjunto de variadas aplicações e serviços.

2 SISTEMA OPERACIONAL DO SERVIDOR

A proposta básica da computação em nuvem é que a provisão de recursos computacionais seja de responsabilidade de empresas especializadas ou que seja abstraído o fornecimento dos mesmos em níveis que apenas especialistas venham se preocupar em gerenciá-los e mantê-los, e ainda os mesmo sejam disponibilizados como serviços (Carr, 2008, p. 52).

Para que toda essa engenhosidade ganhe movimento e se torne acessível, será utilizado um Sistema Operacional para garantir a qualidade e eficiência da utilização desse novo método ainda pouco utilizado.

Sistema Operacional para um servidor pelo seguinte método: Ubuntu Remasterizado – onde será instalado *Remastersys* em uma versão do Ubuntu, criado diversas contas de usuário e realizados algumas mudanças para torná-lo original e adaptável ao projeto.

2.1 DISTRO UTILIZADA

Ubuntu é um Sistema Operacional com base no Linux, criado pela comunidade e é normalmente utilizado em notebooks, desktops, servidores, dentre outros. Ele contém todos os aplicativos necessários para funcionar: um navegador web, programas de apresentação, programas de edição de texto, planilha eletrônica, comunicador instantâneo e muito mais.

Seja para uso doméstico, escolar ou empresarial, o Ubuntu contém todas as ferramentas que um Sistema Operacional necessita para operar, desde processador de texto e leitor de e-mails a servidores web e ferramentas de programação.

Uma pessoa com *Ubuntu* está aberta e disponível para outros, apoia os outros, não se sente ameaçada quando outros são capazes e bons, baseados em uma autoconfiança que vem do conhecimento que ele ou ela pertence a algo maior e é diminuída quando os outros são humilhados ou diminuídos, quando os outros são torturados ou oprimidos. (Arcebispo Desmond Tutu no livro "*No Future Without Forgiveness*", em português: "Sem Perdão Não Há Futuro").

Para Conrad, 2013, “[...] a origem e o significado da palavra Ubuntu [...] é uma antiga palavra africana que significa algo como ‘Humanidade para os outros’ ou ainda ‘Sou o que sou pelo que nós somos’, ou seja, a distribuição Ubuntu traz o sentido desta palavra para o mundo do software livre”.

2.2 REMASTERIZANDO O S.O.

A remasterização do Ubuntu pode ser feita de forma muito fácil. Quem está por trás dessa facilidade toda é o Remastersys, um utilitário de backup para o Klikit-Linux, e também Ubuntu.

O Remastersys pode fazer backup de um sistema Linux instalado no HD junto com dados do usuário, para restauração em caso de destruição do sistema ou cópia em outro hardware. Ele tem um utilitário gráfico e também de linha de comando.

Outra função dele é a geração de um liveCD instalável, com base em um sistema comum instalado no HD - ele não funciona em todos os sistemas, afinal foi feito tendo como base o Ubuntu.

Simioni, 2013, afirma que a ideia do programa é breve: consiste em pegar uma instalação (de preferência limpa) do Ubuntu ou seus derivados e configurá-la como quiser. Quando estiver "no ponto", com os programas desejados e alterações personalizadas, você chama o aplicativo e em poucos minutos tem sua imagem ISO. Ela pode ser gravada em CD ou DVD, e terá então um "novo Ubuntu", que roda direto do CD e pode ser instalado no HD, se desejado.

Figura 1 - Área de Trabalho do Ubuntu Remasterizado



Fonte: Arquivo Pessoal do Autor

3 COMPUTADOR DE PLACA ÚNICA

O aparelho utilizado para executar o Sistema Operacional do usuário será o Banana Pi. Um aparelho do tamanho de um cartão de crédito, de baixo custo, em média trinta dólares (US\$ 30,00), que se conecta a um monitor de computador ou TV, e usa um teclado e mouse padrão. É um computador de placa única fabricado na China e projetado pela equipe LeMarke, criado com base no Raspberry PI com muitos recursos avançados. Ela é útil para que as pessoas de todas as idades explorem a computação e aprenda a programar em linguagem desde o início. Este dispositivo permite fazer tudo o que se espera de um computador desktop: navegar na internet, fazer planilhas, processamento de texto, utilização de jogos e reprodução de vídeo em alta definição.

3.1 SISTEMAS OPERACIONAIS DISPONÍVEIS

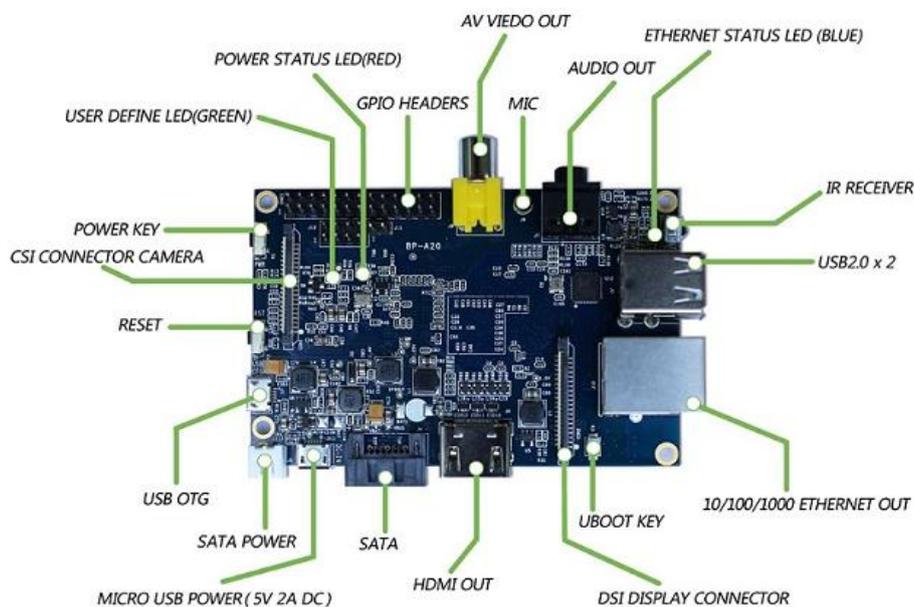
Essa placa é compatível com o Sistema Operacional baseado em Linux. O Sistema Operacional é tecnicamente apenas o kernel, com a coleção total de drivers, serviços e aplicações que compõem o sistema operacional.

Conforme Nick, 2014, a placa “Banana PI é um novo hardware de código aberto que foi produzido para executar o sistema operacional open source Elastos.org entre outros”. O autor expõe ainda outras utilidades, afirmando que “ele é compatível com o sistema operacional baseado em Linux e que existem muitas distribuições desenvolvidos para Banana Pi Hardware. Algumas dessas distribuições incluem Lubuntu, Android, Debian, Bananian, Berryboot, OpenSuse, arranhões, Fedora, Gentoo, Aberto MediaVault, OpenWRT. E também suporta o sistema BSD”.

A placa é uma excelente ferramenta de aprendizagem educacional que pode ser usado para muitos projetos, incluindo: construção de projetos de multimídia, robôs, aplicações Arduino ou de programação de computador com muitas linguagens de programação populares disponíveis incluindo Scratch (uma linguagem de programação de arrastar e soltar para as pessoas começando a aprender de código).

3.2 ESPECIFICAÇÕES

É tentador pensar no Banana Pi como uma placa de desenvolvimento para microcontrolador como o Arduino, ou como um substituto de laptop. Na verdade, é mais parecido com os componentes internos, porém expostos, de um celular (ou PDA), com muitos conectores acessíveis a criadores (Makers) para as várias portas e funções. Essa placa foi feita para ser um computador bastante barato, pequeno e flexível para o dia a dia com um sistema A20 Allwinner em um chip (SoC), que inclui um ARM Cortex-A7 Dual-core (ARMv7-A) 1GHz, Mali -400 GPU MP2 e 1 GB de SDRAM DDR3. É utilizado um cartão microSD para reinicializar o SO e também tem uma interface SATA 2.0 para conectar um disco rígido para obter



mais
espaço de

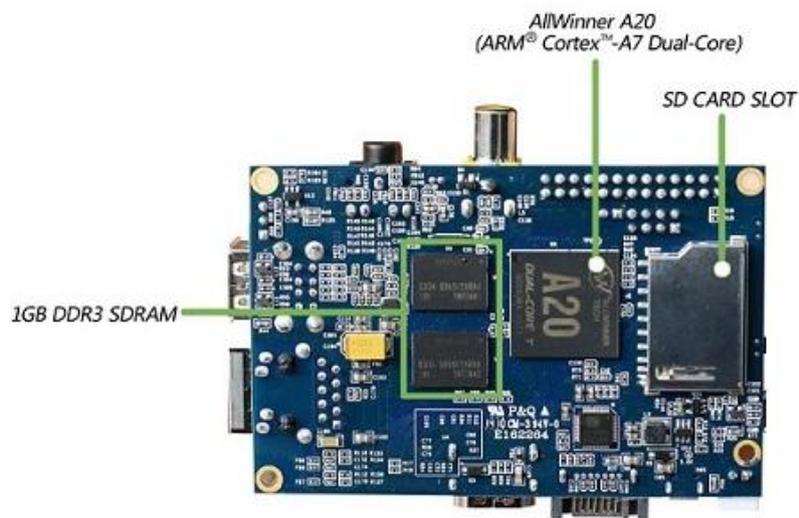
armazenamento. Em comparação com o Raspberry Pi, o Banana Pi adiciona um WiFi on-board 802.11 b/g/n AP6181 e integrado de vídeo composto e saída de áudio a uma tomada TRRs 3,5 milímetros para obter mais espaço para fazer cabeçalhos de extensão de 40 pinos.

Figura 2 Banana Pi (Frente e Verso)

Fonte: <http://www.lemaker.org/article-43-1.html>

4 SISTEMA OPERACIONAL DO USUÁRIO

função
 acesso
 nuvens
 trabalho
 usuário
 arquivos,
 como



A
 desse OS é o
 remoto pelas
 na área de
 do servidor, e
 possibilitando o
 de acessar
 aplicativos,
 dentre outros,
 antes citados.

O Banana Pi já utiliza diversas distribuições Linux disponíveis, para todos os gostos e preferências. O uso de uma distribuição Linux facilita bastante o trabalho, principalmente para quem está começando seus estudos e ainda não tem domínio nessa área.

Entretanto unido com a facilidade de uso, vem a falta de flexibilidade no ponto em que trata de um menor tempo de boot ou um conjunto específico de aplicações, torna-se limitado no quesito controle, que fica mais difícil de conseguir quando se usa uma distribuição Linux já pronta. Porém se for preciso ter controle sobre os pacotes usados na distribuição e suas respectivas licenças, o que é essencial para produtos industriais, o ideal é construir um sistema Linux customizado para as necessidades do projeto.

Isso se dará por uma adaptação feita em componentes do Linux, para que o Sistema Operacional seja utilizado em uma placa removedora, mas com atributos similares, com os seguintes componentes adaptados a uma arquitetura AMD: Toolchain, Bootloader, Kernel e Rootfs.

4.1 TOOLCHAIN

Segundo Retrieved, 2013, o Toolchain é um conjunto de ferramentas de desenvolvimento de software distintas que estão ligados em conjunto por etapas específicas, tais como GCC, binutils e glibc (uma parte do GNU Toolchain).

Afirma ainda que, opcionalmente, este conjunto pode conter outras ferramentas, como um depurador ou um compilador para uma linguagem de programação específica, como, C++. Muitas vezes, esta conjuntura usada para desenvolvimento de sistemas embarcados é um conjunto de ferramentas cruz, ou mais comumente conhecido como um compilador cruzado.

Nomeia-se compilação cruzada todos os programas (como o GCC) que são executados em um sistema de acolhimento de uma arquitetura específica (tal como x86), mas produzem código binário (executáveis) para ser executado em uma arquitetura diferente (por exemplo, ARM). É a maneira típica de construção de software embarcado.

Consoante a isto, podemos considerar que um simples agrupamento de software consiste em um compilador e vinculador para transformar o código-fonte em um programa com probabilidade de execução, bibliotecas para fornecer interfaces para o sistema operacional, e um depurador. Um produto complexo, como um jogo de vídeo precisa de

ferramentas para a preparação de efeitos sonoros, música, texturas, modelos em três dimensões e animações, e outras ferramentas para combinar esses recursos para o produto acabado.

4.2 BOOTLOADER

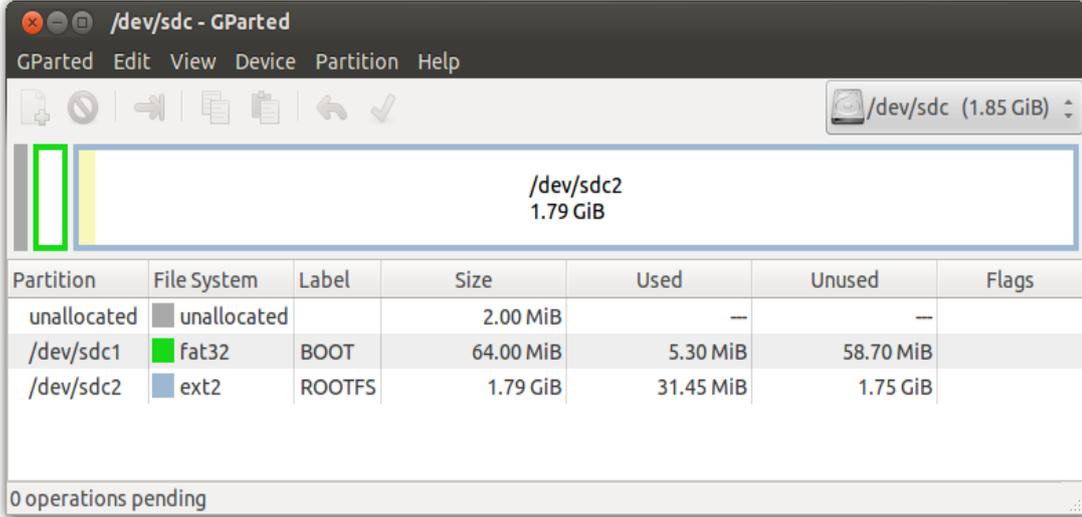
O processo de boot do Banana Pi é um pouco diferente quando comparada à outras plataformas ARM disponíveis a venda no mercado. Quando você energiza a placa, quem assume o controle é a GPU do SoC (o núcleo ARM11 está desabilitado neste andamento).

A GPU procura na primeira parte do cartão SD um código de boot, e com mais alguns processos adicionais, irá carregar e executar o kernel Linux.

Entretanto é preciso preparar um cartão SD com 2 partições:

- Partição 1 (FAT32): partição de 64M para armazenar o código de boot, arquivos de configuração da GPU e a imagem do kernel Linux.
- Partição 2 (EXT2): partição para armazenar o sistema de arquivos.

Figura 3 - Particionamento do Cartão SD



Partition	File System	Label	Size	Used	Unused	Flags
unallocated	unallocated		2.00 MiB	--	--	
/dev/sdc1	fat32	BOOT	64.00 MiB	5.30 MiB	58.70 MiB	
/dev/sdc2	ext2	ROOTFS	1.79 GiB	31.45 MiB	1.75 GiB	

0 operations pending

Fonte: <http://sergioprado.org/wp-content/uploads/2012/11/gparted.png>

4.3 KERNEL

O Kernel é um dos elementos do Sistema Operacional, mas fica tão oculto que a maioria dos usuários domésticos sequer ouviu falar nele. Embora esse fator não diminua em nada sua importância (muito pelo contrário do que pode parecer) ele é tão essencial para o funcionamento de um computador que é melhor mantê-lo a salvo de pessoas inexperientes.

Um simples computador divide-se, basicamente, em duas partes essenciais, software e hardware. Aqui entra a importância e a essencialidade do Kernel: ele é o grande responsável por fazer a interação entre essas duas vertentes, ou seja, é o Kernel que gerencia os recursos do sistema e permite que os programas façam uso deles.

Inicialmente, ele começa a funcionar no instante em que o computador é ligado; nesse momento ele começa a detectar todos os hardwares indispensáveis ao funcionamento da máquina (monitor, placa de vídeo, mouse, teclado, etc.). O Sistema Operacional é carregado em seguida é solicitado ao usuário que faça o login. Uma vez logado, o Kernel passa a administrar as principais funções dentro do sistema operacional, isso inclui o gerenciamento da memória, dos processos, dos arquivos e de todos os outros dispositivos periféricos conectados a máquina.

Dessa forma podemos afirmar que o Kernel é um grande organizador: é ele o responsável por garantir que todos os programas terão acesso aos recursos de que necessitam (memória RAM, por exemplo) simultaneamente, fazendo com que haja um compartilhamento concorrente – mas sem oferecer riscos à integridade da máquina.

4.4 ROOTFS

Trata-se do sistema principal de arquivos. Possui as bibliotecas do sistema para uso dos serviços exportados pelo kernel, além das bibliotecas e aplicações do usuário. Para adapta-lo será usado o Buildroot, que é um sistema de build completo, capaz de gerar o tool-chain, bootloader, kernel e rootfs, porém é usado apenas para gerar o rootfs.

4.4.1 Buildroot

É um conjunto de makefiles e manchas que simplifica e automatiza o processo de construção de um completo e inicializável ambiente Linux para um sistema embarcado,

enquanto estiver usando cross-compilação para permitir a construção de várias plataformas de destino de um sistema de desenvolvimento baseado em Linux. Segundo Sirotkin, 2011, [...] ele pode automaticamente construir o cross-compilação, necessária, um conjunto de ferramentas, criar um sistema de arquivos raiz, compilar um kernel do Linux imagem, e gerar um carregador de boot para o sistema embarcado alvo, ou ele pode executar qualquer combinação independente dessas etapas. Por exemplo, um conjunto de ferramentas cross-compilação já instalado pode ser usado, enquanto Buildroot só cria o sistema de arquivos raiz.

5 VANTAGENS

Uma das principais vantagens da computação em nuvem, além do baixíssimo custo da placa removedora, é a facilidade de acesso aos dados e aplicações de onde estiver, desde que haja conexão com internet de qualidade, fazendo com que o usuário encontre simplicidade na mobilidade e flexibilidade na utilização de todos os recursos.

Este modelo de utilização simplificado permite ainda que o usuário não tenha grandes problemas com instalação e manutenções que estarão sob cuidados do gerenciador.

Outro fator preponderante é a facilidade e simplicidade de pagamento pelo uso: este sistema possibilita ao usuário pagar somente o que necessita, evitando desperdício de recursos, e também graças a esta escalabilidade é possível ampliar a disponibilidade de recursos conforme o usuário verifica necessidade do mesmo.

Esta maleabilidade permite que os riscos atrelados à infraestrutura sejam minimizados, pois o usuário ou a empresa não precisa comprar muitos recursos físicos e não assume responsabilidade sobre a infraestrutura que contrata.

Outras flexibilidades incidem na agilidade de utilização dos serviços e compartilhamento de recursos, além da credibilidade do trabalho, uma vez que as empresas que oferecem os serviços são medidas por sua reputação, principalmente pela capacidade em manter os dados seguros através de cópias de segurança, criptografia e controle de acesso rigoroso onde cada usuário possua seu login e senhas pessoais.

6 DESVANTAGENS

Uma das grandes desvantagens são os pontos-chaves para a evolução e aceitação da mesma, dentre eles estão: segurança, escalabilidade, interoperabilidade, confiabilidade e disponibilidade.

Na segurança o maior desafio e mais visível a ser enfrentado, é a informação que antes era armazenada localmente e agora irá localizar-se na nuvem em lugar físico que não se tem ao certo onde é, e nem que tipos de dados estão sendo armazenados junto a ela. A privacidade e integridade das informações são então itens de extrema importância, pois especialmente em nuvens públicas existe uma grande exposição a ataques, mesmo com toda precaução do gerenciador em relação a segurança. Dentre as capacidades requeridas para evitar a violação das informações está: a criptografia dos dados, o controle de acesso rigoroso e sistema eficaz de gerenciamento de cópias de segurança.

A escalabilidade é uma particularidade essencial na computação em nuvem, pois as aplicações para uma nuvem precisam ser escaláveis (ou “elásticas”). Desta forma os recursos utilizados podem ser modificados conforme a demanda. Para que isso aconteça, as aplicações e seus dados devem ser flexíveis o suficiente. Este trabalho pode não ser simples e normalmente depende da implementação.

A interoperabilidade consiste na disponibilidade dos usuários de executar seus programas e seus dados e nuvens diferentes, admitindo assim que eles não fiquem limitados somente a uma nuvem. Essa é uma característica vastamente desejável na atmosfera de computação em nuvem. Embora muitas aplicações tenham tentado levar em consideração esse fator, existe a necessidade de implementação de padrões e interfaces para que essa portabilidade seja possível.

A confiabilidade está relacionada à frequência com que o sistema falha e qual o efeito de suas falhas (perda ou não dos dados). Os aplicativos desenvolvidos para computação em nuvem devem ser confiáveis, ou seja, elas devem possuir uma arquitetura que permita que os dados permaneçam invioláveis mesmo que haja falhas ou erros em um ou mais servidores ou aparelhos virtuais sobre os quais essas aplicações estão sendo executadas. Essa característica está ligada à política e gerenciamento das cópias de backup.

A disponibilidade é uma grande preocupação, pois mesmo sistemas grandiosos como da Google, Gmail, por exemplo, ficou indisponível. Mesmo que o sistema fique sempre on-line o usuário precisa do funcionamento da internet com qualidade de sinal, que também é

um serviço que não possui disponibilidade ao nível de uma rede local. Uma alternativa plausível é ter mais de um prestador, assim, mais de uma nuvem, o que permitiria aos usuários executar seu programa em outra nuvem enquanto outra está fora do ar. No entanto, esta alternativa não é tão simples, pois requer que haja interoperabilidade entre as nuvens.

CONCLUSÃO

Esse projeto propõe uma mudança na forma como computadores pequenos trabalham. Problemas como atualizações de software, backups, instalações, manutenção e vírus vêm atormentando o mundo há tempos, motivos estes que serviram de inspiração e incentivo na construção deste artigo e possivelmente um pontapé inicial para mudanças futuras na forma de utilização de tecnologia inovadora e acessível a todas as classes, onde as desvantagens ainda podem ser estudadas e melhoradas, pois ainda é um projeto em contínua melhoria.

Com um excelente custo benefício, a expansão dessa estrutura que é uma tecnológica invisível aos olhos será uma crescente ainda sem horizonte. Apesar de ainda pouco conhecida tem perspectiva de se tornar uma tecnologia comum usual em pouco tempo, está a caminho de sofrer uma grande mudança pelo fato de que cada vez mais os serviços oferecidos pela Computação em Nuvem possibilitem que dispositivos com pouca capacidade de processamento e máquinas poderosas executem um mesmo aplicativo com a mesma eficiência.

ABSTRACT

This article provides an introduction to the concept of Cloud Computing. The same addresses the definition of this new computing format, features, architecture and necessary infrastructure, applications, advantages and disadvantages. The main objective of the project is to provide an ideal operating system for those who live connected for stability, speed, simplicity, safety and always online, and includes a simple and efficient user interface. It is easy to use, with programs and files stored safely in the cloud, without the need for bulky hardware or extra storage device using a single board computer of low cost. Thus all this ingenuity gain motion and strength and become accessible as it will use an operating system to ensure the quality and use efficiency of this new model still little used. OS to a server for Ubuntu Remastered model - which will be installed Remastersys on a version of Ubuntu, set up multiple user accounts and made some changes to make the original and adaptable to the

project. This simplified use model also allows the user to have no major difficulties with installation and maintenance, and also the effect of flexibility, allows the risks linked to infrastructure are minimized. Other flexibilities focus on ease of use of services and resource sharing, one of the main points to be addressed in this article. Addresses advantages and disadvantages over the article and shows that though still little known has prospect of becoming a usual common technology before long, by the fact that, increasingly, the services offered by Cloud Computing enable devices with little processing power and powerful machines running the same application with the same efficiency.

KEYWORDS: Operational system. Cloud computing. Remover board. Remastering.

REFERÊNCIAS

BORGES, Hélder Pereira; SOUSA José Neuman; SHEUSE Bruno; e MURY Antonio Roberto, **COMPUTAÇÃO EM NUVEM**. Disponível em: livroaberto.ibict.br/bitstream/1/861/1/COMPUTAÇÃOEMNUVEM.pdf. Acesso em 08 ago. 2015.

BOWATER, Donna. Mini Raspberry Pi computer goes on sale for £22, **The Daily Telegraph**. Disponível em: <http://www.telegraph.co.uk/technology/news/9112841/Mini-Raspberry-Pi-computer-goes-on-sale-for-22.html>. Acesso em: 29 de Fevereiro de 2015.

BROWN, Eric. **Raspberry Pi Clones Match the Connectors, But Boost the Firepower**. Disponível em: <http://www.linux.com/news/embedded-mobile/mobile-linux/771048-raspberry-pi-clones-match-the-connectors-but-boost-the-firepower>. Acesso em: 24 April 2014.

CARR, Noah. **Big Switch: Rewiring the World, from Edison to Google**. Norton & Company, 2008. (traduzido). Disponível em: <http://ubuntu-br.org/>. Acesso em 03 ago. 2015.

CELLAN-JONES, Rory. **A £15 computer to inspire young programmers BBC News**. Disponível em: http://www.bbc.co.uk/blogs/thereporters/rorycellanjones/2011/05/a_15_computer_to_inspire_young.html. Acesso em: 05 maio 2015.

CONRAD , Adam. **Ubuntu 8.04 (Hardy Heron)**. Disponível em: <https://lists.ubuntu.com/archives/ubuntu-announce/2013-March/000168.html>. Acesso em: 21 de jul. 2015.

HEATH, Nick , **Raspberry aa Pi: Cinco alternativas para hackers e modders**. Disponível em: <http://www.techrepublic.com/pictures/raspberry-pi-five-alternative-boards-for-hackers-and-modders/5/>. Acesso em: 20 de jul. 2015.

HEATH, Nick. **Raspberry Pi: Five alternatives for hackers and modders**. Disponível em: <http://www.techrepublic.com/pictures/raspberry-pi-five-alternative-boards-for-hackers-and-modders/5/>. Acesso em: 21 maio 2015.

INTEL, 2010, **Cloud Computing**, Disponível em: http://www.nextgenerationcenter.com/detalle-curso/Cloud_Computing.aspx?PageID=1. Acesso em: 04 ago. 2015.

LAWLER, Richard, **Pi credit-card sized Linux PCs are on sale now, \$25 Model A gets a RAM bump, Engadget**. Disponível em: <http://www.engadget.com/2012/02/29/raspberry-pi-credit-card-sized-linux-pcs-are-on-sale-now-25-mo/Raspberry>. Acesso em: 29 de jul. 2015

NAMJOSHI, J.; GUPTA, A. **Service Oriented Architecture for Cloud Based Travel Reservation Software as a Service. Cloud Computing**, 2009. CLOUD '09. IEEE [International Conference on] , vol. 03, no. 02, p.147-150, 21-25 Sept. 2009.)

PEDROSA, Paulo; NOGUEIRA Tiago, **Computação em Nuvem**. Disponível em: www.ic.unicamp.br/~ducatte/T2/Artigos/G04-095352-120531-t2.pdf. Acesso em: 06 jul. 2015

RODRIGUES, Fabrício. **Um estudo sobre os benefícios e os riscos de segurança na utilização de Cloud Computing**. Artigo publicado na UNISUAM. Disponível em: <https://pt.wikipedia.org/wiki/UNISUAM>. Acesso em: 02 ago 2015.

SHAPIRO, Jonathan S.; SMITH , Jonathan M.; FARBER, David J. **EROS: um sistema de recurso rápido**. Disponível: <http://portal.acm.org/citation.cfm?doid=319151.319163>. Acesso em: 20 de jul. 2015.

SIROTKIN, Alexander. **O Manual do Usuário Buildroot**. Disponível: <http://www.buildroot.org/downloads/manual/manual.pdf>. Acesso em: 20 de jul. 2015.

VANDRESEN Rogério Schueroff , MAGALAES Willian Barbosa, Conceito e **Aplicações da Computação em Nuvem**. Disponível em: <http://ftp.unipar.br/~seinpar/2013/artigos/RogérioSchueroffVandresen.pdf>. Acesso em: 21 de jul. 2015.