Fundação Alexandre de Gusmão
Esplanada dos Ministérios, Bloco H,
Anexo II, Térreo
CEP: 70170-900 - Brasília - DF
Telefones: (61) 3411-6033/6034
Fax: (61) 3411-9125
www.funag.gov.br
E-mail: funag@itamaraty.gov.br

CONSEGI 2010

III Congresso
Internacional
Software Livre e
Governo Eletrônico

ISBN 85-7631-241-7

APOIO

Ministério
da Cultura

Ministério
da Educação

Ministério das
Relações Exteriores

PATROCÍNIO BRONZE

PATROCÍNIO PRATA

PATROCÍNIO OURO

PATROCÍNIO PLATINUM

REALIZAÇÃO

Ministério
da Fazenda

448

AMÃPYTUNA

CONSEGI 2010

CONSEGI 2010

AMÃPYTUNA*

*NUVEM FORTE EM TUPI-GUARANI.

III Congresso Internacional
Software Livre e Governo Eletrônico

www.consegi.gov.br

CONSEGI 2010

AMÃPYTUNÃ*

*NUVEM FORTE EM TUPI-GUARANI.

III Congresso Internacional
Software Livre e Governo Eletrônico

www.consegi.gov.br

AMÃPYTUNA

COMPUTAÇÃO EM NUVEM: SERVIÇOS LIVRES PARA A SOCIEDADE DO CONHECIMENTO

MINISTÉRIO DAS RELAÇÕES EXTERIORES

Ministro de Estado Embaixador Celso Amorim
Secretário-Geral Embaixador Antonio de Aguiar Patriota

FUNDAÇÃO ALEXANDRE DE GUSMÃO

Presidente Embaixador Jeronimo Moscardo

Ministério da Fazenda

Ministro de Estado Guido Mantega

Serviço Federal de Processamento de dados - SERPRO

Diretor Presidente Marcos Vinícius Ferreira Mazoni

A *Fundação Alexandre de Gusmão*, instituída em 1971, é uma fundação pública vinculada ao Ministério das Relações Exteriores e tem a finalidade de levar à sociedade civil informações sobre a realidade internacional e sobre aspectos da pauta diplomática brasileira. Sua missão é promover a sensibilização da opinião pública nacional para os temas de relações internacionais e para a política externa brasileira.

Ministério das Relações Exteriores
Esplanada dos Ministérios, Bloco H
Anexo II, Térreo
70170-900 - Brasília, DF
Telefones: (61) 3411-6033/6034
Fax: (61) 3411-9125
Site: www.funag.gov.br

CONSEGI 2010
III CONGRESSO INTERNACIONAL SOFTWARE LIVRE E GOVERNO ELETRÔNICO

Amãpytuna

Computação em Nuvem: serviços livres para a sociedade do conhecimento

FUNDAÇÃO ALEXANDRE DE GUSMÃO

Brasília, 2010

Ministério das Relações Exteriores
Esplanada dos Ministérios, Bloco H
Anexo II, Térreo
70170-900 Brasília – DF
Telefones: (61) 3411-6033/6034
Fax: (61) 3411-9125
Site: www.funag.gov.br
E-mail: funag@itamaraty.gov.br

Equipe Técnica:
Maria Marta Cezar Lopes
Cíntia Rejane Sousa Araújo Gonçalves
Erika Silva Nascimento
Fabio Fonseca Rodrigues
Júlia Lima Thomaz de Godoy
Juliana Corrêa de Freitas

Programação Visual e Diagramação:
Juliana Orem e Maria Loureiro

Impresso no Brasil 2010

C74 Congresso Internacional Software Livre e Governo Eletrônico (3. : 2010 : Brasília)
Amãpytuna: computação em nuvem: serviços livres para a sociedade do conhecimento. -- Brasília : FUNAG, 2010.
135 p. : il.

ISBN: 978-85-7631-241-3

1. Computação em nuvem (Cloud computing). 2. Sistema de informação. 3. Software livre. 4. Inovação. 5. TV digital

CDU: 004. 4

Depósito Legal na Fundação Biblioteca Nacional conforme Lei n° 10.994, de 14/12/2004.

Sumário

Prefácio

Lauro Luis Armondi Whately
Laboratório de Computação Paralela / COPPE / UFRJ

"When the network becomes as fast as the processor,
the computer hollows out and spreads across the network."
- Eric Schmidt (CTO, Sun Microsystems - 1993)

"There was a time when every household, town, farm or village
had its own water well.
Today, shared public utilities give us access to clean water by
simply turning on the tap;
cloud computing works in a similar fashion."
- Vivek Kundra (CIO, Governo EUA – 2010)

Os avanços recentes no poder de processamento, na capacidade de armazenamento em memória e disco e na largura de banda das redes de comunicação nos conduziram para uma era em que encontramos acesso a recursos computacionais virtualmente ilimitados e distribuídos globalmente. No modelo de computação[1] ainda corrente, usuários tem acesso a poderosos

[1] De forma mais precisa todo sistema computacional deve possuir seu modelo de computação, modelo de armazenamento e modelo de comunicação. Para simplificar a discussão, estou usando apenas o modelo de computação.

computadores pessoais, onde a maior parte do processamento é feito no hardware local ou em alguns casos em servidores em uma sala próxima. Na última década, assistimos a transferência do processamento dos dados em *desktops* para servidores remotos na Internet. Três avanços – virtualização, *utility computing* e aplicações Web (representados por AJAX, REST, SOA etc) ofereceram as ferramentas iniciais para esta mudança de paradigma, onde cada vez mais a computação em benefício do usuário é feita por servidores anônimos na Internet. Exemplos são os mais diversos, mas incluem serviços como edição de documentos, edição e transmissão de vídeo, simuladores financeiros, tradução de texto e mapas. Como resultado, nosso ambiente computacional passou por uma mudança radical. A mudança de uma visão de computação centrada no hardware para uma computação orientada por serviços. Nesse ambiente, a rede passa a ser vista como um repositório de computadores virtuais escalável e confiável. Esse repositório conjugado com os serviços oferecidos constituem o que está sendo chamado de "computação na nuvem". Os serviços nesse modelo estão disponíveis de modo ubíquo para os usuários e os provedores de serviços passam a não se preocupar com os incômodos da administração, manutenção e o custo de aquisição e propriedade de um sistema de computadores.

De maneiras diferentes, as três tecnologia citadas permitem que um sistema grande, totalmente integrado, seja construído a partir de componentes heterogêneos e muitas vezes incompatíveis. A virtualização elimina as diferenças entre as plataformas de computação de diferentes fabricantes, permitindo que aplicativos desenvolvidos para rodar em um sistema operacional possa ser executado em diferentes plataformas. As técnicas encontradas em *utility computing* permitem que os diversos componentes do sistema passem a atuar efetivamente como um único dispositivo, partilhem as suas capacidades e sejam alocados automaticamente para tarefas específicas de cada cliente. O cenário econômico atual reforça por várias razões o compartilhamento das máquinas entre muitos serviços, tais como o uso racional do espaço físico alocado, o alto custo da energia elétrica consumida pelo sistema e a grande quantidade de calor gerado pelos servidores. O compartilhamento dos componentes do sistema permite o uso mais efetivo dos recursos por multiplexação e oferecem economia de escala na administração e manutenção da plataforma. Aplicações Web oferecem protocolos otimizados e seguros para o diálogo entre os *web browsers* dos clientes e os servidores remotos. Elas oferecem aplicações com as mesmas

características das encontradas nos *desktops* combinada com a ubiquidade encontrada nos serviços na web.

Individualmente, todas estas tecnologias são interessantes, mas combinadas são capazes de criar um modelo verdadeiramente revolucionário. Juntamente com alta capacidade das redes de comunicação de fibra óptica, elas podem transformar um conjunto fragmentado de componentes de hardware e software em uma infra-estrutura única e flexível, que muitas empresas podem compartilhar. Uma vez que estas tecnologias evoluem, enquanto tecnologias novas e relacionadas emergem, a capacidade de fornecer este novo modelo de computação - e os incentivos econômicos para fazê-lo -, só vai continuar a crescer.

Mas as vantagens encontradas no uso da computação em nuvem estão acompanhadas de riscos e desafios inerentes ao novo modelo de computação. Como em qualquer mudança de paradigma, é de se esperar encontrar alguma confusão na transformação dos métodos e processos antigos. Alguns dos riscos não são novos e devemos ter em mente que eles já são encontrados na soluções conhecidas, como a segurança dos dados no modelo cliente-servidor em empresas conectadas à Internet, por exemplo. Alguns desafios encontrados como a segurança dos dados armazenados na nuvem, o risco de ficar preso a um provedor de plataforma, confiabilidade do serviço e integração com sistemas legados ainda não possuem soluções amplamente resolvidas. Esse livro apresenta a melhor forma de enfrentar os desafios encontrados na adoção da computação na nuvem: descrever com profundidade e conhecimento o modelo de computação e apresentar as soluções já conhecidas. As análises encontradas nos capítulos a seguir são valiosas, pois mesmo que o leitor decida não adotar a computação na nuvem, sua conclusão vai ser em benefício do teu negócio.

A computação em nuvem não é um modelo rígido e pode ser moldado às características e necessidades das empresas. O capítulo “Computação em Nuvens para Governos” analisa as adaptações do modelo que podem ser adotados por uma empresa. Por exemplo, os departamentos podem compartilhar os recursos locais de TI administrado centralmente em uma “nuvem” e devem conhecer as implicações e forças de influência que surgirão ao implementar o novo modelo. Dentro deste contexto, certamente ferramentas de apoio à decisão são de grande valia para análise de riscos e oportunidades. O capítulo “O Quadrante SERPRO de Tecnologia” apresenta uma ferramenta de análise do ciclo de tecnologia no âmbito da gestão do

conhecimento, utilizando como métricas a produtividade e maturidade da tecnologia.

Um caso bastante exemplar pode ser o de um pesquisador de uma empresa farmacêutica que precisa rapidamente analisar os dados que obteve de seu último experimento. Para obter os resultados a tempo e dentro de seu orçamento, são necessários 25 servidores. Na indústria farmacêutica, o custo de atraso de um produto é estimado a 150 dólares por segundo!

Existem, para este pesquisador, dois cenários:

1) Método tradicional: espere que o pedido de compra seja aprovado, espere até que os servidores sejam entregues, espere a configuração dos servidores etc. O total de espera pode chegar a 3 meses. O que significa um custo de mais de 1 bilhão de dólares.

2) Computação na nuvem: o pesquisador abre uma conta na *Amazon Web Services*, configura 25 servidores e dentro de 2 horas, ele está analisando os dados. Após completar a tarefa, recebe a conta da Amazon: 89 dólares!

Este não é um exemplo imaginário. Isto realmente aconteceu na empresa farmaceutica Eli Lilly[2]. Esse caso real demonstra um dos grandes benefícios do uso da computação em nuvem, mas ao mesmo tempo introduz um novo problema: como a Eli Lilly garante a segurança de seus dados na transmissão e no processamento remoto? Como garantir que não haverá nenhum traço de seus dados nas máquinas da Amazon ? Protocolos de segurança e auditoria precisam ser desenvolvidos para que este modelo de computação possa ser aceito e amplamente usado e oferecido pelas empresas. Sobre este tema, encontramos no livro uma ampla discussão sobre os aspectos de segurança dos dados na computação em nuvem. O capítulo "Segurança de Computação em Nuvem" indica que a transparência da forma como o provedor implementa, desenvolve e gerencia a segurança são fatores decisivos para este objetivo. A argumentação é ilustrada por exemplos de serviços implementados pelo próprio SERPRO.

Assim como o pesquisador do exemplo pode em duas horas implementar a aplicação de análise de dados na "nuvem" da Amazon, ele conseguirá executar esta aplicação em outras "nuvens" sem consumir outras duas ou mais horas ? Para aplicações mais complexas, o cliente ficará preso ao

[2] "Cloud Computing Security Framework May Ease Security Concerns", Information Security Magazine, março de 2009. http://searchsecurity.techtarget.com

provedor da "nuvem" ? Uma solução para este desafio dever ser promover referência de implementação aberta de forma a permitir que "nuvens" pertencentes a diferentes provedores possam conversar entre si. A referência deve permitir mover as aplicações de uma nuvem para outra sem reescrevê-las. Os capítulos "Modelo de Referência de Cloud" e "Computação na Nuvem – Uma Visão Geral" analisam os desafios relacionados com a interoperabilidade. O primeiro apresenta uma análise profunda dos padrões encontrados na implementação e integração dos componentes da computação em nuvem e o segundo capítulo citado discute os desafios para a engenharia de software e demonstra modelos de programação, como o Map-Reduce, amplamente adotado através de uma implementação livre, o Hadoop .

Uma tendência encontrada na computação orientada por serviço é o crescimento acelerado de serviços de *streaming* de conteúdo multimídia na Internet, como o oferecido, por exemplo, no *Youtube*, entre outros serviços. Serviços de *streaming* implicam na transmissão e na reprodução imediata de vídeo e áudio de notícias, eventos esportivos, entretenimento e sítios educacionais. Por exemplo, o serviço *Mogulus* chega a transmitir 120.000 canais de vídeo ao vivo na Internet, com toda a sua infra-estrutura encontrada na "nuvem".

Um grande desafio para os serviços de streaming de vídeo é a interatividade com o usuário. O serviços web vêm ocupando o espaço da televisão no dia a dia das pessoas. Certamente não devemos decretar a morte da televisão, devido essa tendência. Mas, pelo contrário, a previsão mais correta será o fortalecimento através da sinergia das mídias. O uso do canal digital, para a difusão de baixo custo de vídeos HD e 3D e a interatividade através do canal de retorno, encontrado no acesso à Internet feito na própria TV. As possibilidades de serviços, facilmente implementados na "nuvem", são tecnicamente apenas limitadas pela nossa imaginação. O capítulo "Ginga, NCL e NCLua - Inovando os Sistemas de TV Digital" apresenta a solução brasileira para este desafio. Apresentado como uma solução para IPTV e ISDB-Tb, para TV terrestre, o Ginga também poderia ser utilizado como uma interface interativa para os serviços de vídeo oferecidos na "nuvem".

Os serviços de vídeo diferem dos serviços web tradicionais em muitos aspectos, apresentando novas questões para desenvolvedores de sistemas e os provedores de serviços multimídia. Serviços de *streaming*, por razões do modelo de contabilidade do uso de recursos dos sistemas operacionais correntes, não possuem um controle de admissão interno que possa prevenir

a sobrecarga no servidor, ou alocar uma fração predefinida de recursos para um serviço em particular. O servidor de *streaming* não conhece qual recurso em especial deve ser monitorado para avaliar a disponibilidade (ou o uso) corrente da capacidade do sistema: a utilização do processador e memória, como também a banda da rede são altamente dependentes das características do serviço. Consequentemente, os serviços de computação em nuvem ainda oferecem controle de alocação ineficiente. A granularidade e o escalonamento dos recursos alocados, por não conhecerem exatamente a necessidade da aplicação, são pré-definidos estaticamente. Dentro deste contexto, a pesquisa e desenvolvimento no Laboratório de Computação Paralela[3], da COPPE/UFRJ, busca através das técnicas de virtualização e *utility computing* encontrar escalabilidade e eficiência na construção de servidores de streaming de vídeo.

O autor Nicholas Carr tem a opinião[4] que o maior entrave para a computação em nuvem não é tecnológico, mas de atitude. Como na passagem para o oferecimento público da energia elétrica (no lugar dos geradores particulares), os principais obstáculos são os pressupostos de gestão estabelecidas, as práticas tradicionais e os investimentos do passado. As grandes empresas vão desativar seus *datacenters* somente após a confiabilidade, a estabilidade e os benefícios deste novo modelo terem sido claramente estabelecidos. Para que isso ocorra, é necessário oferecer uma visão clara de como o modelo de computação em nuvem deve operar, bem como a imaginação e o desejo de fazer as coisas acontecerem. Os capítulos aqui apresentados fazem um apelo atraente, demonstrando que centralizar a gestão dos recursos anteriormente dispersos, não só reduz os custos de capital, mas também pode aumentar a segurança, melhorar a flexibilidade e reduzir o risco.

[3] Mais informações sobre a pesquisa no LCP podem ser encontradas em http://www.lcp.coppe.ufrj.br

[4] Nicholas G. Carr, "The End of Corporate Computing", MITSloan Management Review, spring 1995, vol.46, n.3

Apresentação

Por que falar de Computação em Nuvem?

Com o advento da Internet e das sociedades em rede, as relações humanas sofreram intensas transformações nos últimos anos. Esse contexto trouxe múltiplos desafios à Administração Pública. Alguns são aparentemente contraditórios. Por um lado, existem as demandas relacionadas ao crescente nível de exigência dos cidadãos e das empresas. O aumento da qualidade do serviço prestado, a modernização e a introdução de novos serviços, além da celeridade dos processos de atendimento tornaram-se palavras de ordem. Por outro lado, existem as demandas relacionadas ao controle da despesa pública. A redução de custos de funcionamento, o aumento da eficiência e a eliminação de processos burocráticos tornaram-se obrigação.

Por isso, uma importante pergunta impõe-se ao Serviço Público neste momento: como fazer melhor com menos? Com certeza, um dos caminhos a ser percorrido é a adoção de modelos de processos de trabalho suportados pela utilização da Tecnologia de Informação (TI) como indutora da inovação e da eficiência.

No atual momento do setor público, esses sistemas e processos de trabalho são mais informativos do que colaborativos. Eles são utilizados de forma desintegrada e para realizar procedimentos limitados, fazendo com que o esforço seja segmentado e executado por meio de trâmites protocolares. Essas características aumentam, em muito, o tempo de resposta ao cidadão.

Dessa maneira, a Administração Pública ainda está muito distante da situação social contemporânea: um ambiente de colaboração e interatividade apoiado em sistemas computacionais. Caso estivesse próxima, isso permitiria potencializar as inovações, melhorar o desempenho interno, eliminar processos redundantes, agir com pró-atividade, partilhar tarefas, além de reduzir substancialmente o tempo e os custos operacionais, facilitando o trabalho dos funcionários públicos e agregando valor e qualidade à prestação do serviço ao cidadão.

Na busca pela equiparação dessa condição, empresas do setor público como o Serviço Federal de Processamento de Dados – Serpro, a Empresa de Tecnologia e Informações da Previdência Social – Dataprev, o Banco do Brasil – BB e a Caixa Econômica Federal - CEF discutem a adoção da Computação em Nuvem (*Cloud Computing*) como forma de dinamizar a capacidade de processamento para serviços que possam estar cada vez mais presentes na vida dos cidadãos.

Além das estações de trabalho

Recursos computacionais em qualquer lugar e independente da plataforma utilizada. Variadas aplicações acessadas por meio da Internet com a mesma facilidade de tê-las instaladas em uma estação de trabalho. Esse é o conceito da Computação em Nuvem.

Com ela, aplicativos, arquivos e dados relacionados não precisam mais estar instalados ou armazenados em um computador local. Eles ficam disponíveis na "nuvem", isto é, distribuídos em vários servidores, tendo como meio físico de integração entre eles, uma rede de dados, como a Internet. O usuário final não precisa se preocupar em manter a informação, e pode utilizá-la a partir de qualquer ponto de acesso, computador, ou até mesmo, de acordo com a disponibilidade, a partir de *Smartphones* e *Thin Clients*.

Em sua essência, a Computação em Nuvem trata-se de uma forma de trabalho na qual o software é oferecido como serviço, não sendo necessário adquirir licenças de uso para instalação ou mesmo comprar computadores ou servidores para executá-lo. Nessa modalidade, normalmente paga-se um valor periódico - como uma assinatura - somente pelos recursos que utilizar e/ou pelo tempo de uso.

Essa revolução em marcha no mundo da TI é silenciosa, porém profunda. E daí, vem a importância do presente livro que analisa, sob a

perspectiva do governo, as transformações que ocorrerão nos próximos anos e sua consequente mudança na relação entre a Administração Pública e os cidadãos.

Boa leitura!

Marcos Vinícius Ferreira Mazoni
Diretor-Presidente do Serpro

Computação na Nuvem – Uma Visão Geral

Karin Breitman[1] *e Jose Viterbo*[2]

Abstract

Cloud computing is fast becoming an important platform for research in Software Engineering. Scientists today need vast computing resources to collect, share, manipulate, and explore massive data sets as well as to build and deploy new services for research. Cloud computing has the potential to advance research discoveries by making data and computing resources readily available at unprecedented economy of scale and nearly infinite scalability. In this chapter we provide an introduction to Cloud Computing, i.e, its fundamentals, definition, architecture, associated technologies, challenges and research opportunities.

Resumo

A computação na nuvem está rapidamente se tornando uma das mais importantes plataformas de pesquisa em Engenharia de Software. O cenário atual demanda um grande volume de recursos para o apoio as tarefas de elicitação, compartilhamento, manipulação e exploração

[1] Departamanto de Informática Pontifícia Universidade Católica do Rio e Janeiro (PUC-Rio).
[2] Departamento de Ciência e Tecnologia Universidade Federal Fluminense (UFF).

de grandes conjuntos de dados, bem como para o desenvolvimento e execução de serviços de apoio à Pesquisa. A computação na nuvem tem o potencial de contribuir no avanço da Ciência na medida em que pode disponibilizar um grande volume de recursos para utilização imediata, criando um cenário sem precedentes, onde a escalabilidade de serviços, processos e infra-estrutura é quase ilimitada. Neste artigo fazemos uma introdução à computação na nuvem: fundamentos, definições, arquitetura, tecnologias associadas, desafios e oportunidades de pesquisa na área.

1. Introdução

A idéia essencial da computação da nuvem é permitir a transição da computação tradicional para um novo modelo onde o consumo de recursos computacionais, por exemplo, armazenamento, processamento, banda entrada e saída de dados, será realizado através de serviços. Podemos traçar um paralelo entre o cenário atual com o final do século XIX, durante o período da Revolução Industrial, quando era comum que grandes fábricas fossem responsáveis pela produção de sua própria energia elétrica e ou mecânica. Hoje em dia as grandes fábricas consomem energia como um serviço, e pagam pela quantidade utilizada. A computação na nuvem tem uma proposta similar. Recursos computacionais passarão a ser de responsabilidade de algumas empresas especializadas, que ficarão responsáveis por sua gestão e comercialização através de serviços [Carr 2008].

Evidente que esta proposta representa uma grande quebra de paradigma, pois atualmente, tanto empresas quanto particulares, utilizam recursos computacionais de forma proprietária, ou seja, são os donos e responsáveis pela gestão, manutenção e atualização dos recursos computacionais que dispõem. O objetivo deste artigo é apresentar uma breve introdução a computação na nuvem, com o intuito de fornecer ao leitor subsídios para que entenda os desafios e oportunidades oferecidas por este novo modelo. Começamos por oferecer algumas definições para o termo, já que não existe uma única formulação para o conceito. A seguir introduzimos uma breve taxonomia, que define tipos, modalidades e características principais. Para facilitar o entendimento das novas oportunidades oferecidas pelo modelo de computação na nuvem, discutimos

os principais avanços tecnológicos que permitiram seu desenvolvimento. Na seção seguinte discutimos o modelo conceitual mais utilizado para descrever este novo paradigma. A seguir discutimos o impacto da computação na nuvem nos requisitos de novas aplicações de software. Finalizamos nossa contribuição apresentando os desafios e oportunidades de pesquisa na área.

1.1. Terminologia

Neste texto optamos, sempre que possível, por apresentar a tradução dos termos para o Português, formatado com letras minúsculas. Desta forma, o termo *Cloud Computing* foi traduzido para computação na nuvem. No entanto, o jargão da área inclui alguns anglicismos os quais encontramos dificuldade em traduzir. É o caso do termo datacenter. Nestes casos optamos por deixar o termo em inglês, porém suprimimos a utilização do itálico para facilitar a leitura. Casos mais simples, por exemplo, nomes próprios, de serviços, empresas, onde o termo está sendo reproduzido em inglês, são apresentados em itálico, seguido de tradução quando julgada necessária.

2. Definição

O termo computação na nuvem foi introduzido em 2006, quando o então CEO da Google, Eric Schmidt, utilizou o termo para descrever os serviços da própria empresa, e, pouco tempo depois, quando a Amazon utilizou o mesmo termo para lançar seu serviço EC2 (Elastic Compute Cloud). Foi, na verdade popularizado através de um artigo na edição de Outubro da Wired, através do artigo de George Gilder entitulado "*The Information Factories*" (as fábricas de informação) [Gilder 2006] .

Apesar de ainda não termos uma definição única objetiva para computação na nuvem, o termo é, geralmente, utilizado como rótulo tanto para determinadas aplicações que são acessadas pela internet, como para serviços de datacenters. No primeiro caso, as aplicações já conhecidas por serem utilizadas nos desktops, como editores de texto, planilhas ou, até mesmo, editores de imagens, são acessadas através da internet, e todo o processamento e armazenamento de dados que ocorriam no próprio computador do usuário, agora ocorrem online, ou "na nuvem".

Já nos serviços de datacenters, o termo computação na nuvem é utilizado quando o conjunto de recursos, como servidores, balanceadores de carga, armazenamento, etc., são comercializados por uso e, normalmente, cobrados por hora. Este novo modelo de negócios trouxe uma série de benefícios técnicos e financeiros para consumidores de seus serviços tais como: rápido provisionamento, escalabilidade, facilidade para lançamento de novos produtos/serviços por empresas de menor tamanho, entre outros. A seguir oferecemos algumas das definições mais encontradas na literatura para o termo.

2.1 As várias definições de Computação na Nuvem

O grupo Gartner define computação na nuvem como "*um estilo de computação onde capacidades de TI elásticas e escaláveis são providas como serviços para usuários através da Internet*." [Cearley 2009].

Markus Klems argumenta que características chaves de computação na nuvem são a escalabilidade imediata e a otimização da utilização de recursos. Estas são adquiridas através do monitoramento e automação dos recursos computacionais em utilização [Klems 2009].

Kepes define computação na nuvem da seguinte forma. "*De forma simplificada computação na nuvem é um paradigma de infra-estrutura que permite o estabelecimento do SaaS (software-como-serviço)... é um grande conjunto de serviços baseados na Web com o objetivo de fornecer funcionalidades, que até o momento demandavam enorme investimento de hardware e software, através de um novo modelo de pagamento por uso.*" [Kepes 2008].

Vaquero et al. realizaram um amplo estudo onde foram consideradas dezenas de diferentes definições para o conceito de computação na nuvem. Os autores chegaram a seguinte definição. "*Nuvens são grandes repositórios de recursos virtualizados (hardware, plataformas de desenvolvimento e/ou serviços), facilmente acessíveis. Estes recursos podem ser reconfigurados dinamicamente de modo a se ajustar a cargas variadas, otimizando a utilização destes mesmos recursos. Este repositório de recursos é tipicamente explorado utilizando-se um modelo do tipo pagamento-por-uso, onde os fornecedores de infra-estrutura oferecem garantias no formato de SLAs (service level*

agreements) customizadas." [Vaquero et al 2008]

Uma definição de computação na nuvem, ligada as características de hardware, é fornecida por Armbrust et al. [Armbrust 2009]. Segundo os autores, este novo paradigma oferece as seguintes novidades:

1. Ilusão de recursos computacionais infinitos, disponibilizados sob demanda, eliminando a necessidade do planejamento para a provisão de recursos a longo prazo.

2. Eliminação da necessidade de se fazer grandes investimentos iniciais em infra-estrutura, permitindo com que negócios sejam iniciados com um parque computacional pequeno e que aumentem sua infra-estrutura a medida em que suas necessidades demandarem.

3. Possibilidade da contratação de recursos computacionais a curto prazo, por exemplo, processadores por hora, armazenagem por um dia. Uma vez que estes não são mais necessários, capacidade de finalizar os contratos.

É interessante notar que os autores remarcam que não encontraram um denominador comum entre todas as definições estudadas, isto é, uma característica comum a todas as definições.

2.2 Distinções

Uma fonte comum de confusão é a comparação entre computação na nuvem e *grid computing*. A distinção não é clara, pois ambos paradigmas compartilham os mesmos objetivos de redução de custos, aumento de flexibilidade e confiabilidade através da utilização de hardware operado por terceiros. A maior distinção entre os dois diz respeito a alocação de recursos. Enquanto que em um grid se tenta fazer uma distribuição uniforme de recursos, em ambientes de computação na nuvem os recursos são alocados sob demanda. A utilização dos recursos também é diferenciada, pois a virtualização garante uma maior separação entre os recursos utilizados pelos vários usuários em ambientes de computação na nuvem.

Também existem importantes diferenças que distinguem o modelo de computação na nuvem do modelo tradicional de computação. A Tabela 1, a seguir, adaptada de [Cearley 2009], apresenta um resumo dessas diferenças.

Tabela 1 - Diferenças entre os modelos tradicionais e de computação na nuvem

	Computação Tradicional	Computação na nuvem
Modelo de aquisição	Hardware	aquisição de serviço
	Espaço físico	
	Infra estrutura de instalação e funcionamento	
Modelo de negócio	custo e depreciação de ativos	pagamento baseado na utilização
	Overhead administrativo (manutenção, suporte, segurança do equipamento, refrigeração)	
Modelo de acesso	Rede interna	Internet, através de vários tipos de dispositivos (não apenas computadores)
	Intranet	
Modelo técnico	único "morador"	Escalável
	sem compartilhamento	Elástico
	estático	Dinâmico
		Condomínio

3. Nuvens públicas, privadas e híbridas

O modelo de computação na nuvem se refere a aplicações disponibilizadas como serviços na internet, serviços de hardware, e sistemas em datacenters que fornecem este tipo de serviço. O conjunto do hardware e software é o que chamamos de nuvem [Cearley 2009]. Quando a nuvem é fornecida para o publico em geral e sob um contrato onde se paga pelo montante utilizado, chamamos a mesma de nuvem pública. Os serviços comercializados são geralmente chamados de computação utilitária (*Utility Computing*). Alguns exemplos são as plataformas da Amazon, Google AppEngine e Microsoft Azure. Uma lista mais completa é apresentada na seção 1.8 deste capitulo.

O termo nuvem privada é utilizado para designar um novo estilo de computação disponibilizado pelo provedor interno de TI, que se comporta

de forma semelhante a um ambiente de computação na nuvem externo. Neste modelo, capacidades de TI elásticas e escaláveis são oferecidas como serviços para usuários internos. A grande diferença entre a computação na nuvem publica e privada reside nos serviços de acesso e nos serviços de controle. São muitos os requisitos tecnológicos necessários para que o modelo privado funcione de forma adequada. Entre eles estão tecnologias de virtualização, automação, padrões e interfaces, que permitam o acesso compartilhado a servidores virtuais. Algumas destas tecnologias são discutidas na Seção 5 deste artigo.

Na literatura encontramos, além dos modelos de nuvem publica e privada, outros modelos que combinam os dois conceitos. São eles [Smith et al. 2009], computação híbrida e computação privada virtual. Alguns autores acreditam que em no futuro próximo, a maior parte das empresas vai utilizar a computação na nuvem de alguma forma. O modelo de computação híbrida prevê uma utilização mista, porém integrada, dos dois paradigmas, i.e., a combinação de serviços de computação na nuvem externos com recursos internos. Esta colaboração deve ser realizada de forma coordenada, de forma a garantir integração no nível dos dados, processos e camadas de segurança [Hassan et al 2009].

A computação privada virtual se configura pela divisão, seguida do isolamento, de uma porção de um ambiente de computação na nuvem publico. Este ambiente fica isolado, e é de utilização dedicada a um único grupo ou entidade. Além disto, um ambiente de computação privada virtual também pode ficar isolado da Internet, utilizando uma rede particular (rede privada virtual – *virtual private network* ou VPN), e/ou uma rede LAN para o acesso aos serviços. Estas soluções servem para melhorar a performance, o trafego de dados e, é claro, a segurança [Cearley 2009]. A Amazon já está oferecendo um serviço deste tipo comercialmente, o Amazon VPC.

4. Características essenciais

Apesar da falta de consenso sobre um conjunto essencial a aplicativos de computação na nuvem, existem algumas características que são centrais, citadas na maioria dos textos relevantes sobre o assunto [Cunha 2009]. São elas (a) virtualização de recursos, (b) independência de localização com acesso aos recursos via Internet, (c) elasticidade e (d) modelo de pagamento baseado no consumo.

A *virtualização de recursos*, conseguida através do uso de tecnologias já estabelecidas como *virtual machines*, virtualização de memória, Virtualização de armazenamento e virtualização de rede (VPN), desatrela os serviços de infra-estrutura dos recursos físicos (hardware, rede). Com essa abstração, a localização real dos recursos é tratada na camada mais baixa da computação na nuvem (IaaS), sendo transparente ("invisível") para as demais camadas. Desta forma, os recursos passam a ser disponibilizados e consumidos como uma utilitários. Em uma arquitetura baseada em serviços, características do consumidor são abstraídas das características do provedor através de interfaces de serviços bem-definidas. Estas interfaces ocultam os detalhes de implementação e possibilitam trocas automatizadas entre provedores e consumidores de serviços. Neste modelo, serviços ganham um nível a mais de abstração, ou seja, passam a ser desenhados para servir necessidades específicas de dos consumidores, ao invés de se ater a detalhes de como a tecnologia funciona.

A *independência de localização* na computação na nuvem, ocorre pois os serviços tornam-se acessíveis de qualquer lugar que se tenha acesso a infra-estrutura de rede. As nuvens aparentam ser o ponto único de acesso para todas as necessidades de computação dos usuários.

A *elasticidade* é, talvez, a característica mais inovadora de computação na nuvem. Há uma diferença sutil entre elasticidade e escalabilidade. Escalabilidade é a habilidade de satisfazer um requisito de aumento da capacidade de trabalho pela adição proporcional da quantidade de recursos. A escalabilidade é quantificada por uma medida de linearidade, mantendo uma relação constante entre recursos e capacidade de trabalho. Esta linearidade permite a organização planejar seus gastos com recursos de acordo com a expectativa de crescimento de sua capacidade de trabalho. Uma arquitetura escalável é construída tipicamente com base em uma infra-estrutura básica passível de repetição, cujo crescimento pode ser alcançado simplesmente com a adição repetidamente do mesmo conjunto de hardware básico [Ho 2009].

Tradicionalmente, a escalabilidade é projetada para garantir que o custo operacional possa crescer linearmente de acordo com a carga de trabalho. Usualmente, não há preocupação com a remoção de recursos nem a preocupação se os recursos são plenamente utilizados, por que os recursos adquiridos já são custo afundado.

Por outro lado, a elasticidade é a capacidade de provisionar e de-

provisionar rapidamente grandes quantidades de recursos em tempo de execução. A elasticidade pode ser medida quantitativamente através do seguintes fatores: velocidade no provisionamento / de-provisionamento dos recursos; quantidade máxima de recursos que podem ser provisionados; e granularidade na contabilidade do uso dos recursos.

É importante notar que mesmo que se consiga elasticidade, quando uma aplicação é colocada em um ambiente de computação na nuvem, a escalabilidade não é uma conseqüência direta, ou seja, não é um atributo natural da arquitetura do sistema. Para que o sistema seja de fato escalável é preciso tenha sido desenvolvido levanto este requisito não funcional em conta desde o inicio. Em casos extremos, o fato de colocar o aplicativo em um ambiente de computação na nuvem pode ter efeitos contrários do desejado, i.e., consumo de recursos maior que o esperado, sem benefícios adicionais.

A outra característica central da computação na nuvem é o *modelo de pagamento de acordo com a utilização dos recursos*. Atualmente, serviços utilitários, tais como água, eletricidade, gás e telefonia são considerados essenciais nas rotinas diárias. Estes serviços são utilizados tão freqüentemente que precisam estar disponíveis sempre que os consumidores necessitem deles. A maior parte destes serviços é oferecido através de modelos de contratos e convênios com provedores onde se paga apenas pela quantidade utilizada dos serviços.

Dentro do paradigma de computação na nuvem, os consumidores de serviços e recursos computacionais necessitariam pagar aos provedores apenas quando se utilizarem de tais serviços. A grande vantagem do modelo é permitir a contratação de novos recursos a medida em que estes se tornem necessários, fazendo com que grandes investimentos em infra-estrutura e manutenção se tornem desnecessários.

Do ponto de vista de negócio, há dois direcionadores principais para adoção da computação na nuvem, validos para adoção de novas tecnologias de um modo geral: redução de custos e aumento de capacidade .

No modelo de computação na nuvem, a redução de custos tem um caráter evolucionário, baseada principalmente no pagamento por uso (nos casos de utilização de um provedor de nuvem público) e na virtualização dos recursos no caso de uso da adoção de ambiente de computação na nuvem privado (ver Seção 3).

Já o aumento da capacidade proporcionado por este tipo de ambiente, principalmente por suas características de elasticidade e acesso aos recursos

via Internet, tem um caráter mais revolucionário. O uso de ambientes de computação na nuvem viabiliza o surgimento de novos serviços (ou aplicações) que se beneficiem dessas características (elasticidade no provisionamento de recursos), independência de localização e dos dispositivos utilizados para acesso, que não ficam limitados a computadores apenas.

A Figura 1, ilustra a contribuição das características dos ambientes de computação na nuvem com os dois principais direcionadores de negócios: redução de custos e aumento da capacidade da organização.

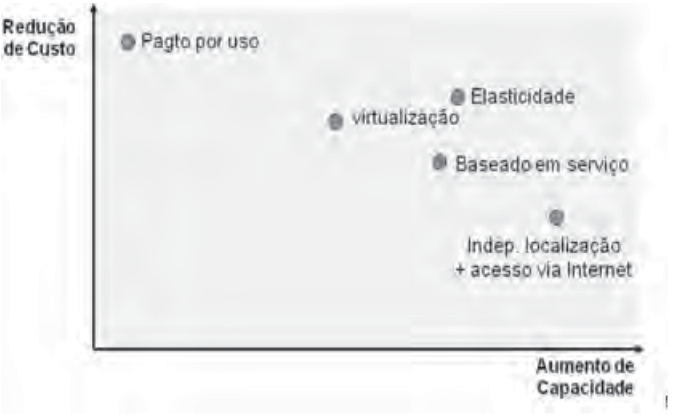


Figura 1 - Contribuição de características aos direcionadores de negócio

As características apresentadas tem o potencial de trazer grandes benefícios para os adeptos do modelo de computação na nuvem. Os benefícios da virtualização são, entre outros, o aumento da eficiência de TI; economia de servidores com a consolidação; redução de custos associados (espaço, energia, refrigeração); tempo de chegada ao mercado (time-to-market) é mais rápido; agiliza o *deployment* (processo de disponibilização) de servidores e aplicações; torna as configurações mais consistentes; e, finalmente, oferece níveis de serviço mais previsíveis. É claro que sempre haverá falhas e paradas não programadas, mas virtualização torna a recuperação mais rápida, fácil e com custo menor.

Os benefícios da elasticidade são o aumento da agilidade com usuários aptos a reprovisionar rapidamente recursos de infra-estrutura; a liberação

dos desenvolvedores de projetar para picos de carga, e, além disso, evitar a alocação superestimada de recursos, através do monitoramento do desempenho os recursos podem ser rapidamente ajustados.

Com o processamento em ambientes de computação na nuvem, uma gama de aplicações que fazem uso intensivo de recursos de infra-estrutura (ex: processamento, armazenamento) passam a contar com a possibilidade de acesso ubíquo, através de uma grande variedade de dispositivos móveis. Como grande parte do processamento é realizado na nuvem, estes dispositivos podem ser simples e desprovidos de grandes recursos de armazenamento e/ou processamento, provendo independência do dispositivo, da localização, e a possibilidade da criação de interfaces para o acesso universal aos recursos através de visualizadores (browsers).]

Os benefícios da virtualização são, entre outros, a grande redução no custo de investimento, que é convertido em custo de operação; a redução das barreiras de entrada, uma vez que a infra-estrutura é tipicamente provida por terceiros e não precisa ser adquirida para computação intensiva de tarefas eventuais; e o modelo de precificação é ajustado com base no uso de um ou mais recursos computacionais, vistos como utilitários.

5. Tecnologias Associadas

Para melhor entender o paradigma da computação na nuvem é importante ter uma melhor compreensão das tecnologias sob as quais o mesmo se baseia. Nesta seção faremos uma breve discussão acerca das mais importantes delas. O leitor interessado encontrará na bibliografia comentada vários ponteiros para outros trabalhos onde poderá se aprofundar nos tópicos discutidos a seguir.

5.1 Plataforma Hadoop

O projeto Apache Hadoop é um *framework* Java para armazenamento e processamento em clusters de dados em larga escala. Este projeto foi inspirado em [Dean e Ghemawat 2008] e, atualmente, é utilizado por diversas corporações como Yahoo!, Amazon e Facebook. O Hadoop se propõe a resolver problemas de escalabilidade das aplicações, possibilitando o processamento de grandes volumes de dados, da ordem de grandeza de *petabytes*, tempo aceitável. Foi criado a partir da necessidade de uma infra-

estrutura comum que seja confiável e eficiente para ser utilizada em *clusters*. Como exemplo de eficiência, em 2008, o projeto foi o vencedor do *Terabyte Sort Benchmark* (Sort Benchmark, 2010) ao ordenar 10 bilhões de registros de 100 bytes cada em 209 segundos, o recorde anterior era 297 segundos, em um *cluster* com 910 nós.

A arquitetura do Hadoop é dividida a partir dos subprojetos apresentados na Figura 2. Os subprojetos HDFS, sistema de arquivos distribuídos, e MapReduce, para sistema para agendamento e execução de tarefas distribuídas, são os módulos chave da arquitetura do Hadoop, e são detalhados a seguir.

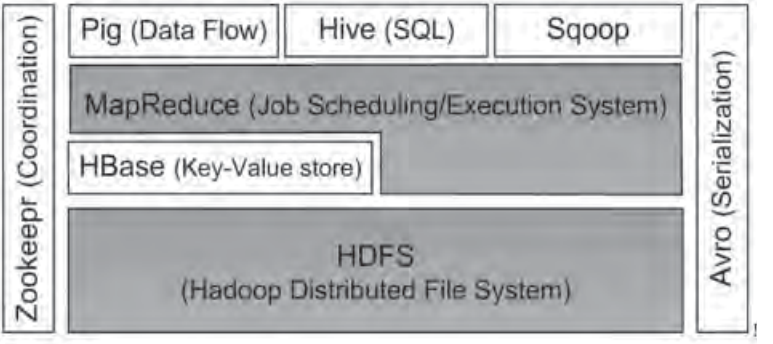


Figura 2 - Arquitetura do Projeto Apache Hadoop

5.2 HDFS

O HDFS – *Hadoop Distributed File System* – é o sistema de arquivos distribuídos do Hadoop, e tem como o objetivo o armazenamento de grandes quantidades de dados através de múltiplos nós. Baseado no Google File System (Ghemawat et. al., 2003), o HDFS realiza o tratamento a possíveis falhas de hardware ao replicar os dados em diferentes nós do *cluster*. Por padrão, os dados são divididos em blocos de 64 MB, e são replicados em três nós – dois nós no mesmo *rack* e outro em um *rack* diferente. Esta estratégia é chamada de *rack-awareness*, onde o framework tenta maximizar o tráfego de dados entre nós do mesmo *rack*.

A topologia do HDFS dividi-se em um *NameNode* e diversos *DataNodes*. O primeiro é responsável por gerenciar os metadados e a

localização dos diversos blocos de dados. Já os *DataNodes* tratam do armazenamento dos blocos. Como só existe um único *NameNode* em um *cluster* Hadoop, este se torna um ponto único de falha. Caso o *NameNode* falhe, o sistema de arquivos do Hadoop fica *offline.*

Desenvolvedores tem acesso ao HDFS através de interfaces Java que permitem não apenas utilizar o sistema de arquivos do Hadoop, mas também alternar para sistemas de arquivos distintos. Dentre os atualmente suportados, estão:

- HDFS: o próprio sistema de arquivos do Hadoop;
- Amazon S3: sistema de arquivos da infra-estrutura de computação na nuvem da Amazon. Em situações onde todos os nós são remotos, não há suporte a *rack-awareness*;
- CloudStore: implementação C++ do Google File System;
- FTP: armazena os dados em servidores FTP acessíveis remotamente (também não suporta *rack-awareness*).

5.3 MapReduce

O modelo de programação Map/Reduce foi apresentado inicialmente em (Dean e Ghemawat, 2008) para o processamento e geração de grandes quantidades de dados, e envolve dois passos: o mapeamento e a redução. No passo de mapeamento, o nó mestre divide os dados de entrada em subproblemas e os distribui para resolução pelos *mappers.* Basicamente, um *mapper (*elemento responsável por processar uma sub-tarefa, *mapeador)* processa o subproblema e envia a resposta ao nó mestre. Este, ao receber as respostas de todos os subproblemas distribuídos aos *mappers* passa à etapa de redução, onde estas respostas são combinadas e enviadas aos *reducers (*elementos responsáveis por combinar respostas enviadas pelos mapeadores, *redutores)* que processam estas combinações de respostas a fim de obter uma nova resposta e as enviam de volta ao nó mestre, que obterá então a resposta do problema original. A Figura 3 ilustra este fluxo de dados na resolução de um problema neste modelo de programação.

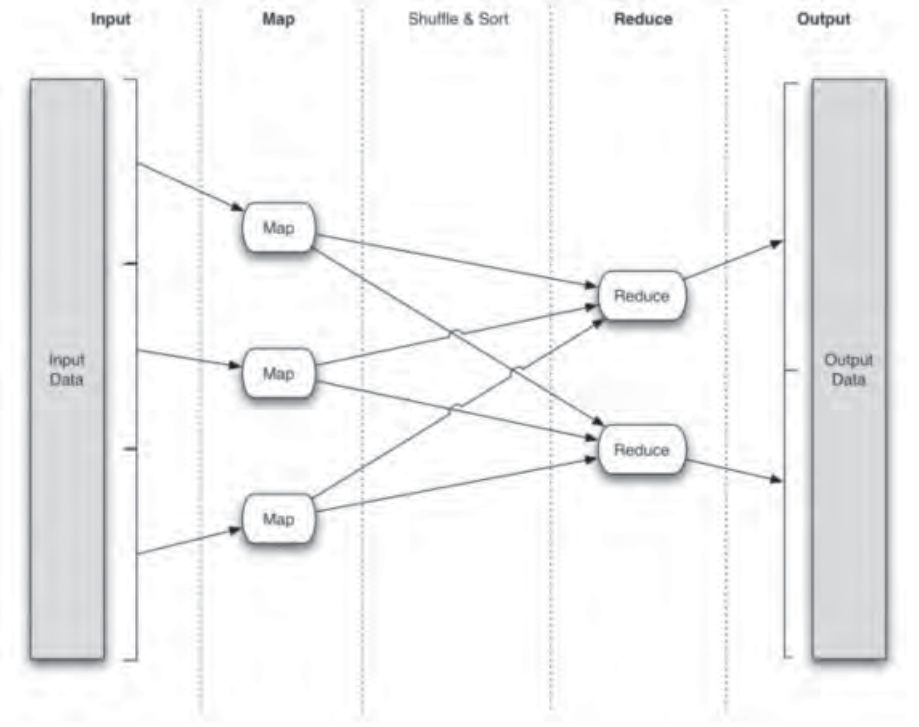


Figura 3 - Fluxo de dados na resolução de um problema no modelo de programação Map/Reduce

O Hadoop MapReduce implementa este modelo de programação e, atuando em conjunto com o HDFS, permite uma eficiente alocação *mappers* e dos *reducers* utilizados durante a resolução do problema. Desta forma, o processamento de um dado é direcionado para o nó onde este dado de entrada está replicado. Este processo implementa a seguinte máxima: "*moving computation is cheaper than moving data*", é mais barato mover o processo computacional do que mover dados.

De forma semelhante ao HDFS, o Hadoop MapReduce também apresenta um único servidor, encarregado de gerenciar os *mappers* e *reducers*, chamado de *JobTracker* (rastreador de tarefas). É o *JobTracker* que determina como dividir os dados de entrada, e para quais nós no *cluster* enviar os subproblemas para resolução. Ele recebe as respostas, combina-as e as envia para os *reducers*. É um ponto único de falha, como o

NameNode, e durante a execução de tarefa cria diversos *check points*, de acordo com a evolução da mesma. Caso o *JobTracker* falhe durante este processo, pode reiniciar a tarefa a partir do último *check point*. Os nós que realizam os subproblemas são chamados de *TaskTrackers*, e a alocação dos mesmos pelo *JobTracker* é feita de forma simples, não levando em conta o atual processamento dos mesmos, apenas considerando o número de processos que estão executando no nó. Como desvantagem, caso um *TaskTracker* demore a apresentar seu resultado, toda a realização da tarefa é prejudicada, visto que o início da fase seguinte – redução ou finalização – depende do sucesso de todos os *TaskTrackers* envolvidos. Na figura a seguir é exibida a topologia do Hadoop Mad/Reduce em paralelo com a do HDFS.

A execução de uma tarefa no Hadoop – chamada de *job* – segue o diagrama ilustrado pela Figura 4. Um cliente Hadoop contacta o *NameNode* para armazenamento e recuperação de dados ou o *JobTracker* para submissão de tarefas. O *NameNode* possui o mapeamento entre os blocos de arquivos e os nós escravos – *DataNodes.* O *JobTracker* submete subproblemas da tarefa original para os *TaskTrackers.* Como forma de otimizar o processo de resolução do problema, os subproblemas enviados aos *TaskTrackers* contém dados que estejam presentes no mesmo nó.

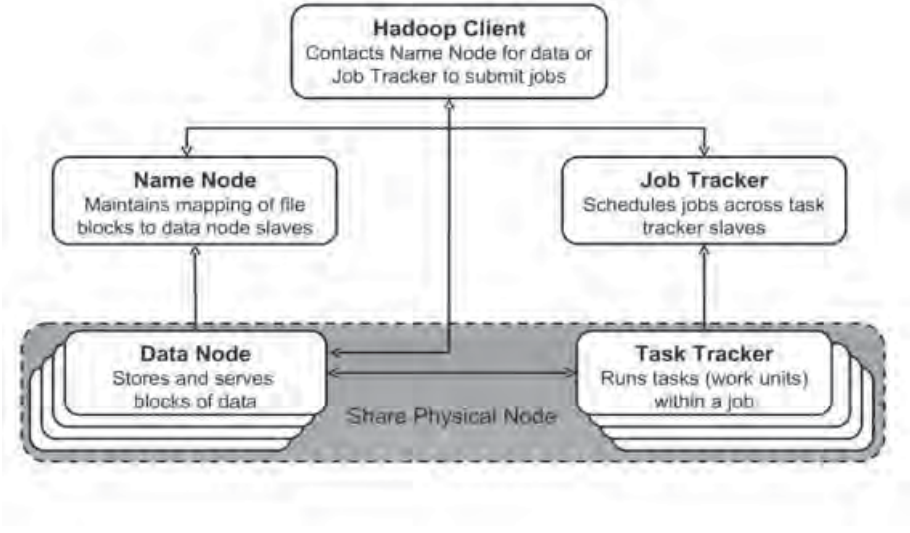


Figura 4 - Execução de uma tarefa no Hadoop

6. Arquitetura

O modelo conceitual de computação na nuvem encontrado com maior freqüência na literatura é a arquitetura em três camadas ilustrada na figura a seguir.

Figura 5 - Arquitetura em três camadas para a computação na nuvem

Neste modelo a camada inferior é composta por plataformas internet para desenvolver, testar, implantar e executar aplicações próprias. Comumente abreviada pelo acrônimo IaaS - *Infrastructure as a Service* (Infra-estrutura como Serviço), esta camada se caracteriza pela utilização de servidores (ou parte deles), para o desenvolvimento de software proprietário. Exemplos são o Force.com, Google App Engine, Bungee, LongJump, Coghead (SAP) e Etelos.

A camada intermediária é composta por hardware virtual, disponibilizado como serviços. São plataformas que oferecem algum tipo de serviço especifico, tais como um banco de dados, serviços de mensagem (MOM) e serviços de armazenamento de dados. É comumente abreviada pelo acrônimo PaaS - *Plataform as a Service* (Plataforma como Serviço). Exemplos são Rackspace, Mosso, Cloudera, Hadoop (ver seção 1.5.2 em diante), GData,

Flexiscale, Eucalyptus, Nimbus e os diversos tipos de plaformas de serviços comercializados pela comercializados pela Amazon: Web Services (AWS), EC2, S3, SimpleDB, SQS.

A camada superior é composta por aplicativos de software que são executados em um ambiente de computação na nuvem. São aplicações completas ou conjuntos de aplicações disponíveis pela Web, cujo uso é regulado por diversos modelos de negócio, permitem customização e, em geral, também podem ser utilizadas de forma offline. Comumente são abreviadas pelo acrônimo SaaS - *Software as a Service* (Software como Serviço). Exemplos são o Facebook, Google Docs , Microsoft SharePoint, SAP Bussiness ByDesign, entre muitos outros.

Youseff, Butrico e Silva propõem uma ontologia para computação na nuvem composta de cinco camadas: aplicativos, ambientes de software, infra estrutura de software, kernel (núcleo) de software, e hardware [Youseff et al. 2009]. Reproduzimos a ontologia proposta pelas autoras na Figura 6.

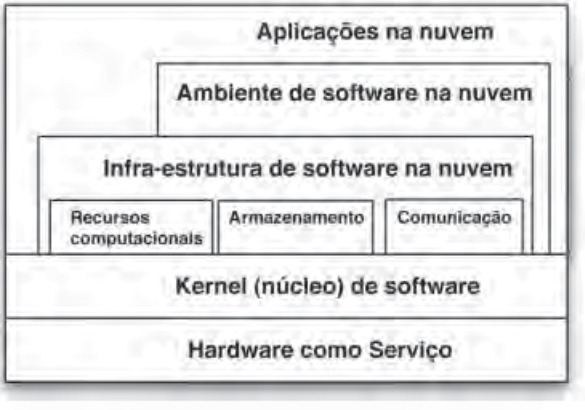


Figura 6 - Ontologia para computação na nuvem proposta por Youseff, Butrico e Silva [Youseff et al. 2009].

O interessante deste modelo é a elaboração das camadas inferiores. Os serviços que se encontram comumente agregados sob a abstração de IaaS (infra-estrutura como serviço) na maior parte dos modelos de arquitetura de nuvem, são elaborados nos serviços de infra-estrutura propriamente dita, dados, comunicação e nas camadas de kernel (núcleo) e de hardware. Esta separação permite com que se trate da virtualização de maquinas de forma

distinta da qualidade de serviço (QoS) da rede de comunicação, por exemplo. A separação em camadas destes três serviços: infra-estrutura, dados e comunicação, também permite um isolamento de questões relativas a gerência de software dos servidores físicos, i.e., a implementação propriamente dita, se como OS kernel, hypervisor, maquina virtual, ou utilizando-se um middleware para construção de grids. Imediatamente abaixo, está a camada de Haas – *hardware as a service* (harware como serviço) que trata dos equipamentos propriamente ditos. Os usuários desta camada são os próprios provedores de serviços de computação na nuvem.

7. Plataformas para a computação na nuvem

Nesta seção discutimos duas das principais plataformas mais utilizadas para a implementação de infra-estruturas e aplicativos de computação na nuvem.

7.1 Amazon Web Services

Sem dúvida alguma a Amazon fornece o conjunto mais completo de serviços de apoio a computação na nuvem. Das plataformas comerciais é a mais antiga, foi lançada em 2002. A seguir descrevemos alguns destes serviços.

Amazon EC2 – O Amazon Elastic Compute Cloud é um ambiente virtual de computação, que permite que se utilize interfaces baseadas em web services para instanciar servidores na nuvem. O EC2 provê três tipos de instâncias: padrão, com muita memória, e com grande capacidade de processamento. As instancias são criadas escolhendo-se dentre uma imensa variedade de imagens pré-configuradas, chamadas Amazon Machine Images (AMI). Estas imagens suportam uma grande variedade de sistemas operacionais, por exemplo. Windows, OpenSolaris, Debian, Fedora, e vários tipos de Unix (Ubuntu, opensSUSE, Gentoo, RedHat). Alem de oferecer varias imagens com aplicativo já instalados (IBM WebSphere, Oracle WebLogic Server, Ruby on Rails, JBoss), também permite com que os usuários preparem suas próprias imagens, que poderão ser replicadas em suas instancias no EC2. A grande vantagem do EC2 é o tempo reduzido para as operações de provisionamento e de deprovisionamento destas instancias, que gira em torno de 5 minutos.

Amazon S3 – Este é um serviço de armazenamento on line, que permite a manipulação (Leitura/Escrita/Deleção) de objetos cujo tamanho em bits varia de 1B a 5GB. O acesso aos objetos armazenados é realizado através de APIs no padrão REST ou SOAP. O serviço é um dos mais antigos oferecidos pela Amazon, e tem usuários famosos, entre eles os Twitter, que se utiliza do S3 para armazenar imagens e o SmugMug, serviço on line de impressão e armazenamento de fotografias. Estima-se que, no momento em que este texto foi preparado, este serviço seja responsável pelo armazenamento de quase cem bilhões de objetos.

Simple DB - É um um serviço que provê, através da nuvem, as principais funções de bancos de dados relacional, por exemplo, busca e indexacão. Este serviço permite a criação e armazenamento de múltiplos conjuntos de dados através de uma interface REST, o que permite a manipulação simples das informações. Este serviço não demanda um modelo conceitual para banco de dados, mas realiza a indexação automática dos dados.

Amazon Cloud Front - É um web service que tem como objetivo facilitar e racionalizar a distribuição de conteúdos (arquivos) na internet. Para tal conta com uma rede global de servidores, distribuídos em vários pontos estratégicos do planeta. Estes são chamados de *edge locations* e são responsáveis por redistribuir e guardar copias de arquivos. A cada requisição recebida, o serviço é responsável por avaliar e redirecionar automaticamente o pedido para o servidor na localização mais próxima. O Amazon CloudFront utiliza a Amazon S3 como seu servidor de origem, onde os objetos são inicialmente armazenados. A partir da primeira requisição em uma localidade distante, o objeto é transferido e mantido no servidor, para agilizar as requisições subseqüentes. O serviço também oferece informações detalhadas sobre o tráfego através dos logs de acesso. No momento da produção deste texto, a Amazon contava com servidores em diversas localidades do Estados Unidos, Europa e Ásia.

Amazon Elastic MapReduce – este serviço utiliza uma infra-estrutura Hadoop (veja seção 1.5 para mais detalhes) que está implementada e rodando sobre a própria infraestrutura provida pelos serviços Amazon Elastic Compute Cloud (Amazon EC2) e AmazonSimple Storage Service (Amazon S3). O Amazon Elastic MapReduce permite a implementação de aplicações para processamento de grandes volumes de informações usando diversas linguagens: Java, Perl, Ruby, Python,PHP ou C++.

Amazon Relational Database Services (RDB) – este serviço permite o projeto e implementação de uma banco de dados relacional na nuvem. Prove funcionalidades similares as do banco de dados MySQL, alem de adicionar código suplementar para fazer com que o código, aplicativos e ferramentas do usuário possam ser executadas utilizando-se o Amazon RDB. O serviço suporta cinco tipos diferentes de instancias, que vão desde a menor com a capacidade de uma CPU[3] até a maior com a capacidade de processamento de vinte e seis CPUs.

7.2 Microsoft Azure

O Windows Azure é a proposta da Microsoft para serviços de computação na nuvem . O serviço consiste em uma plataforma para execução de aplicações, conhecida como PaaS (Plataforma como serviço), onde o programador encontra um conjunto de recursos que facilitam o desenvolvimento de aplicativos de software. A plataforma é composta, essencialmente, por tres grandes componentes, que formam o núcleo do serviço. São eles as unidades de computação, o espaço para armazenamento e o Fabric. Na figura a seguir ilustramos os componentes básicos da arquitetura do Azure. As unidades, também chamados de nós, de computação podem ser máquinas virtuais ou físicas, e são caracterizadas pelo seguinte conjunto de atributos: velocidade do processador, quantidade de memória e tamanho do disco local. O espaço para armazenamento é composto de estruturas de dados criadas a partir de abstrações já conhecidas, como tabelas, arquivos e filas.

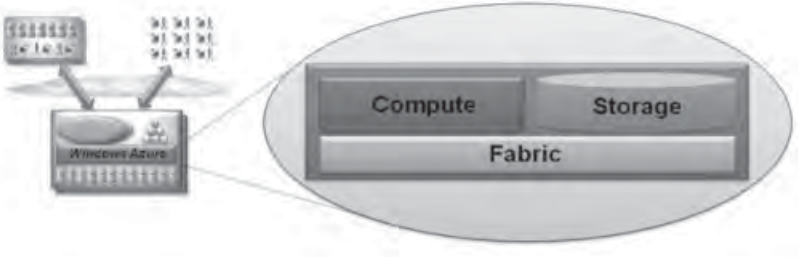


Figura 7 - Arquitetura do Microsoft Azure

[3] Neste caso está se considerando que uma CPU tem a capacidade de processamento de um processador Xeon ou Opteron 2007 de 1.0-1.2 GHz.

A plataforma, por sua vez, é composta por um componente chamado Fabric Controller, que é o software responsável por gerenciar e monitorar todos os recursos do datacenter, como: servidores, IPs, balanceadores de carga, switches, roteadores, entre outros. Este componente também é responsável pelo processo de gerência do ciclo e vida da aplicação, e pela manutenção de níveis satisfatórios de serviços, descritos através de SLAs. A plataforma Azure também intruduz o conceito de Fault Domain, que tem como objetivo reduzir o risco de indisponibilidade das aplicações. Este componente monitora e se utiliza da topologia do datacenter para realizar esta tarefa. Por exemplo, todas as máquinas físicas de um mesmo rack estão conectadas a um mesmo switch. No caso de uma falha deste switch, as aplicações ficariam indisponíveis. Então, o Fabric Controller pode alocar uma determinada aplicação em servidores diferentes. Resumidamente, o Fault Domain evita pontos únicos de falha, garantindo a disponibilidade do serviço.

Outro conceito importante é o Update Domain, que permite especificar quais partes da aplicação podem ficar offline simultaneamente. Assim, nem mesmo no evento de um upgrade, a aplicação fica inacessível. Os chamados papeis (roles) são funções em que as aplicações hospedadas são divididas. Existem três básicos de papéis: Web Role, FastCGI Web Role e Worker Role. O Web Role é um website tradicional, desenvolvido com a tecnologia ASP.NET e é utilizado como interface da aplicação, e fica disponível publicamente na internet. O FastCGI Web Role é semelhante ao anterior, diferenciando-se pela tecnologia utilizada, pois adiciona mecanismos (handlers) ao servidor web permitindo a execução de várias linguagens, como PHP, Phyton, Ruby, entre outras. Porque o Worker Role é utilizado para rodar processos em background, não possui uma interface pública. Cada papel pode ser executado em vários nós, garantindo a escalabilidade das aplicações. Todos os papeis tem acesso à área de armazenamento de dados, portanto, a intercomunicação entre os mesmos pode ser feita através das estruturas, que são explicadas a seguir. O espaço para armazenamento de dados (storage) possui três estruturas, como citado anteriormente, são elas: Blobs (arquivos), Tables (tabelas) e Queues (filas). Todos os dados criados no sistema de armazenamento são replicados automaticamente para fornecer alta disponibilidade e confiabilidade. A utilização destas estruturas se dá através de APIs fornecidas pelo serviço, que são chamadas passando-se como parâmetros uma identificação única e uma chave. Blob é uma representação genérica para arquivos. Estes arquivos devem ser organizados em Containers,

que são divisões lógicas semelhante a diretórios, onde se pode configurar permissões de acesso. Cada blob pode ter um tamanho máximo de 50GB e o upload é feito em blocos, permitindo o envio paralelo.

A Table é uma abstração para tabela, com algumas diferenças para as tradicionais tabelas dos bancos de dados relacionais. A principal diferença é a falta de estrutura fixa, ou seja, cada linha da tabela, ou Entity como foram nomeadas no Azure, pode conter um conjunto de propriedades (Properties) diferentes. Duas propriedades obrigatórias são utilizadas como identificador único dentro de uma tabela, são elas: PartitionKey e a RowKey.

A primeira (PartitionKey) possibilita escalabilidade, já que as tabelas podem ser divididas em partições e armazenadas em nós diferentes, mesmo pertencendo à mesma tabela. Já a RowKey funciona como um identificador dentro de uma partição. Por fim, a fila é um conjunto de mensagens ordenadas, com tamanho máximo de 8KB, e tem como objetivo principal a comunicação entre roles, como, por exemplo, a distribuição de trabalho assíncrono vista na implementação mostrada no próximo tópico.

8. Desafios de pesquisa em computação na nuvem

A novidade deste tópico, aliada aos pesados investimentos que a indústria vem fazendo na área, tem criado excelentes oportunidades para a pesquisa e desenvolvimento de software de qualidade. Com o objetivo de fornecer um pequeno panorama das oportunidades de pesquisa na área, discutimos, a seguir, quatro tópicos de grande interesse: proveniência, requisitos não-funcionais, bibliotecas de imagens e modelos de cobrança.

Proveniência ou rastreabilidade de dados é definida em função da capacidade de se definir a origem, lugar ou processo, dos dados. Metadados são definidos como dados sobre dados [Breitman 2007]. Tanto a proveniência de dados, quanto a gerência de metadados na nuvem ainda são assuntos de pesquisa abertos. Podemos classificar este tipo de informação nos seguintes tipos [Vouk 2008]:

- Proveniência de processos – dinâmica de controle dos fluxos e de sua progressão, informações relativas a execução de processos na nuvem, desempenho do código e rastreabilidade.
- Proveniência de dados – fonte dos dados, localização (física e virtual) de arquivos, informação de entrada e saída de processos e funções que alteram dado.

- Proveniência de fluxo de dados (workflows) – estrutura, forma, evolução do próprio fluxo [Belhajjame et al 2008].
- Proveniência de sistemas (ambientes) – Dados acerca dos atributos sistêmicos, sistemas operacionais, configurações, imagens utilizadas, compiladores, dispositivos de entrada e saída utilizados.

O leitor atento deve ter feito um paralelo com os modelos conceituais para a computação na nuvem apresentados na Seção 6. De fato, a questão da proveniência e da utilização de metadados é transversal a ontologia para a computação na nuvem discutida anteriormente, i.e., independente da camada de serviço, proveniência é uma questão importante [Li e Huan 2008]. Questões relevantes para a pesquisa na área englobam o desenvolvimento de novos métodos, técnicas e ferramentas para o levantamento e representação de informação de proveniência, o desenvolvimento, elaboração e adequação de padrões existem para a utilização junto a infra-estrutura, plataformas e aplicativos na nuvem; anotação automática de dados de proveniência, bem como metadados, Modelos de representação que facilitem a indexação e promovam a interoperabilidade de informações de proveniência e, finalmente, interfaces adequadas para a captura, manipulação e integração de destes dados por parte dos usuários [Balis et al. 2008].

A própria natureza dos aplicativos, plataformas e infra-estrutura de computação na nuvem faz com que os impactos de determinados requisitos não-funcionais passem a ser muito mais relevantes do que para sistemas do tipo "tradicional". Discutimos alguns deles no que se segue.

Segurança – Talvez a maior barreira para a adoção do paradigma de computação na nuvem seja a percepção de segurança por parte dos usuários. Notem que fazemos uma distinção entre a percepção e segurança propriamente dita. Para que os usuários finais se sintam a vontade em colocar dados e aplicativos "sensíveis" em um ambiente controlado por terceiros, é necessário assegurar-lhes que os mesmos estarão em um ambiente seguro, privado e sobretudo confiável. Estas questões trazem a tona várias oportunidades de pesquisa tanto na área da Engenharia de Requisitos quanto na área de Redes [Kaufman 2009, Jensen et al 2009, Kandukuri et al 2009].

Escalabilidade – Nas últimas décadas a prática da Engenharia de Software tem sido voltada ao tratamento da escalabilidade de sistemas, processos e serviços, i.e., a habilidade de manipular uma porção crescente de trabalho de forma uniforme, ou estar preparado para crescer [Bondi 2000]. O cenário

na computação na nuvem é diferente, pois, ao contrario da escalabilidade unidirecional, busca-se a elasticidade de sistemas, processos e serviços. Definimos como elasticidade a capacidade de adequação a variações de demanda, i.e., a capacidade de expansão ou retração voluntaria e controlada, como resposta a um estimulo. A elasticidade é um assunto muito pouco explorado na literatura de Engenharia de Software, e representa uma excelente área de pesquisa. [Zhang et al 2009, Catteddu e Hogben 2009].

Uma outra área passível de exploração é a construção de bibliotecas de imagens e componentes, que facilitem e agilizem o processo de contratação e terminação de instancias de maquinas virtualizadas. O gerenciamento de licenças de software, bem como novos modelos que explorem o lease, ou pagamento pela utilização de licenças de sofware proprietário são assuntos correlatos, e oferecem bons tópicos de pesquisa. [Dalheimer et al. 2009].

Finalmente gostaríamos de apontar para a necessidade de aplicativos de monitoração e medição do volume de recursos utilizados, por exemplo, armazenamento, processamento, tráfego, consultas e transações [Lim et al 2009]. Estes serão fundamentais na construções de novos modelos de *billing* (cobrança) de serviços externos contratados, no monitoramento do volume de recursos utilizado e na otimização dos processos de tomada de decisão. [Elmroth et al 2009, Li et al 2009, Greenberg et al 2009].

9. Conclusão

A computação na nuvem é um assunto muito novo, e muitas são as possíveis direções futuras. Em relação aos aplicativos de software, é muito possível que a grande parte destes seja executada de forma hibrida, i.e., parte nos clientes e parte na nuvem. Este fato gera a necessidade de mecanismos que façam com que a parte dos aplicativos localizada na nuvem seja de fato escalável, i.e., aumente e diminua para se adaptar a demanda. Este é um novo requisito não funcional, que pode mudar significativamente o modo em que desenvolvemos sistemas de software. Outros desafios são a segurança, privacidade e a prevenção do trancamento de dados na plataforma de terceiros.

Do ponto de vista da infra-estrutura é necessário desenvolver novos mecanismos de cobrança e monitoramento de aplicativos. O controle de licenças e novos modelos de licenciamento de software, baseado na utilização, tem de ser vistos.

Neste artigo fizemos uma breve introdução ao tema da computação na nuvem. Fizemos uma opção por fazer uma exposição horizontal, englobando o máximo de assuntos possível no espaço disponível. Desta forma, apresentamos um conjunto de definições, tecnologias associadas, características, requisitos, plataformas e discutimos algumas das questões de pesquisa da área, com o intuito de despertar o interesse do leitor para o tema, e ao mesmo tempo fornecer um conjunto de informações relevantes.

Agradecimentos

Este artigo não seria possível sem a intensa colaboração do Prof. Markus Endler, com quem temos dividido cursos, projetos, publicações e muitas discussões sobre computação na nuvem. Também gostaríamos de agradecer aos alunos ecompanheiros de descobertas: Marcello Azambuja, Marcelo Barbosa, Silvano Buback, Edward Condori, Herbet Cunha, Marcelo Malcher, Renato Mogrovejo, Rafael Pereira, Lincoln Silva e Leonardo Kaplan.

Referências Bibliográficas

Armbrust, M., Fox, M., Griffith, R., et al. - Above the Clouds: A Berkeley View of Cloud Computing - In: University of California at Berkeley Technical Report no. UCB/EECS-2009-28, pp. 6-7, February 10, 2009.

Balis, B., Bubak, M., Pelczar, M., and Wach, J. - Provenance Tracking and Querying in the ViroLab Virtual Laboratory. In Proceedings of the 2008 Eighth IEEE international Symposium on Cluster Computing and the Grid - CCGRID. IEEE Computer Society, Washington, DC, 675-680, 2008.

Belhajjame, K., Wolstencroft, K., Corcho, O., Oinn, T., Tanoh, F., William, A., and Goble, C. 2008. Metadata Management in the Taverna Workflow System. In Proceedings of the 2008 Eighth IEEE international Symposium on Cluster Computing and the Grid (May 19 - 22, 2008). CCGRID. IEEE Computer Society, Washington, DC, 651-656.

Bondi, A. - Characteristics of scalability and their impact on performance -0 Proceedings of the 2nd international workshop on Software and Performance, Ottawa, Ontário, Canadá, 2000, pp. 195 – 203.

Buyya,R.; Yeo, C. S.; Venugopal, S. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. 2008.

Carr, N. Big Switch: Rewiring the World, from Edison to Google - W.W. Norton & Company, 2008.

Catteddu, D. and Hogben, G. 2009. Cloud Computing: benefits, risks and recommendations for information security. Technical Report. European Network and Information Security Agency.

Cearley, D. et al –Hype Cycle for Application Development – Gartner Group report number G00147982 – Relatório técnico do grupo gartner. Acessível em: http://www.gartner.com/, 2009.

Chirigati, F.S. – Notas de aula - Disponível on line em: http://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2009_2/seabra/index.html.

Chung, L. et al. Non-Functional Requirements in Software Engineering. Kluwer Academic Publishers. 2000.

Dalheimer, M. and Pfreundt, F. 2009. GenLM: License Management for Grid and Cloud Computing Environments. In Proceedings of the 2009 9th IEEE/ACM international Symposium on Cluster Computing and the Grid (May 18 - 21, 2009). CCGRID. IEEE Computer Society, Washington, DC, 132-13.

Dean, J. and Ghemawat, S. 2008. MapReduce: simplified data processing on large clusters. Commun. ACM 51, 1 (Jan. 2008), 107-113.

Dikaiakos , M. D.; Pallis, G.; Katsaros, D.; Mehra, P.; Vakali, A. Cloud Computing – Distributed Internet Computing for IT and Scientific Research. IEEE Internet Computing, 13(5): 10-13, setembro/outubro 2009.

Elmroth, E., Marquez, F. G., Henriksson, D., and Ferrera, D. P. 2009. Accounting and Billing for Federated Cloud Infrastructures. In Proceedings of the 2009 Eighth international Conference on Grid and Cooperative Computing IEEE Computer Society, Washington, DC, 268-275.

Ghemawat, S., Gobioff, H., and Leung, S. 2003. The Google file system. In Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles (Bolton Landing, NY, USA, October 19 - 22, 2003). SOSP '03. ACM, New York, NY.

Gilder, G. The Information Factories, Wired Magazine, Outubro, 2006.

Hadoop - Project Apache Hadoop. http://hadoop.apache.org/. Acessado em fevereiro de 2010.

Hand, E. - Head in the clouds. Nature, (449):963, Oct 2007.

Hassan, M. M., Song, B., and Huh, E. 2009. A framework of sensor-cloud integration opportunities and challenges. In Proceedings of the 3rd international Conference on Ubiquitous information Management and Communication (Suwon, Korea, January 15 - 16, 2009). ICUIMC '09. ACM, New York, NY, 618-626.

Ho, R. - Between Elasticity and Scalability Computing. Pragmatic Programming Techniques. Disponível em: http://horicky.blogspot.com/2009/07/between-elasticity-and-scalability.html.

Jensen, M., Schwenk, J., Gruschka, N., and Iacono, L. L. 2009. On Technical Security Issues in Cloud Computing. In Proceedings of the 2009 IEEE international Conference on Cloud Computing - CLOUD. IEEE Computer Society, Washington, DC, 109-116.

Jinesh Varia, Cloud Architectures, artigo disponível através do site: http://jineshvaria.s3.amazonaws.com/public/cloudarchitectures-varia.pdf.

Johnston, S. – Cloud Computing Bill of Rights – acessível em: http://wiki.cloudcommunity.org/wiki.

Kandukuri, B. R., V., R. P., and Rakshit, A. 2009. Cloud Security Issues. In Proceedings of the 2009 IEEE international Conference on Services Computing Symposium on Compiler Construction. IEEE Computer Society, Washington, DC, 517-520.

Kaufman, L. M. 2009. Data Security in the World of Cloud Computing. IEEE Security and Privacy 7, 4 (Jul. 2009), 61-64.

Li, D. and Huan, L. 2008. The Ontology Relation Extraction for Semantic Web Annotation. In Proceedings of the 2008 Eighth IEEE international Symposium on Cluster Computing and the Grid (May 19 - 22, 2008). CCGRID. IEEE Computer Society, Washington, DC, 534-541.

Li, X., Li, Y., Liu, T., Qiu, J. and Wang, F. 2009. The Method and Tool of Cost Analysis for Cloud Computing. In IEEE International Conference on Cloud Computing, Bangalore, India, September 2009, 93-100.

Greenberg, A., Hamilton, J., Maltz, D. and Patel, P. 2009. The Cost of a Cloud: Research Problems in Data Center Networks. ACM SIGCOMM Computer Communication Review, 39, 1.

Lim, H. C., Babu, S., Chase, J. S., and Parekh, S. S. 2009. Automated control in cloud computing: challenges and opportunities. In Proceedings of the 1st Workshop on Automated Control For Datacenters and Clouds). ACDC '09. ACM, New York, NY, 13-18.

Miceli, C., Miceli, M., Jha, S., Kaiser, H., and Merzky, A. 2009. Programming Abstractions for Data Intensive Computing on Clouds and Grids. In Proceedings of the 2009 9th IEEE/ACM international Symposium on Cluster Computing and the Grid (May 18 - 21, 2009). CCGRID. IEEE Computer Society, Washington, DC, 478-483.

Natis, Y et al. Cloud, SaaS, Hosting and Other Off-Premises Computing Models. ID Number: G00159042. Julho, 2008.

Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., and Zagorodnov, D. 2009. The Eucalyptus Open-Source Cloud-Computing System. In Proceedings of the 2009 9th IEEE/ACM international Symposium on Cluster Computing and the Grid - CCGRID. IEEE Computer Society, Washington, DC, 124-13.

Richardson, L. – Restful Web Services – O'Reilly - 2007.

Ricky Ho - Between Elasticity and Scalability Computing. PragmaticProgramming Techniques. Disponível em http://horicky.blogspot.com/2009/07/between-elasticity-and-scalability.html.

Sato, K.; Sato, H.;Matsuoka, S. - "A Model-Based Algorithm for Optimizing I/O Intensive Applications in Clouds Using VM-Based Migration," ccgrid, pp.466-471, 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009.

Smith, D. et al. The What, Why and When of Cloud Computing. ID Number: G00168582. Gartner Junho, 2009.

SUN Microsystems - Take your Business to a Higher Level. Sun Cloud Computing – acessível em: http://www.propelmg.com/sunfed/fedminute/downloads/Sun_Cloud_Computing_Primer.pdf.

Vaquero, L. M., Rodero-Merino, L., Caceres, J., and Lindner, M. 2008. A break in the clouds: towards a cloud definition. SIGCOMM Comput. Commun. Rev. 39, 1 (Dec. 2008), 50-55.

Vogels, W., (2008) "A Head in the Clouds – The Power of Infrastructure as a Service", In: First workshop on Cloud Computing in Applications (CCA'08), October, 2008.

Vouk, M.A., Cloud Computing – Issues, Research and Implementations, 30th International Conference on Information Technology Interfaces, June, 2008, pp. 31-40.

Youseff, L., Butrico, M. and Da Silva, D. 2008. Toward a Unified Ontology of Cloud Computing. In Grid Computing Environments Workshop (GCE'08), Austin, Texas, USA, November 2008, 1-10.

Zhang, X., Schiffman, J., Gibbs, S., Kunjithapatham, A., and Jeong, S. 2009. Securing elastic applications on mobile devices for cloud computing. In Proceedings of the 2009 ACM Workshop on Cloud Computing Security - CCSW '09. ACM, New York, NY, 127-134.

Fundamentos de Computação Nuvem para Governos

Adriano Martins
Serviço Federal de Processamento de Dados (SERPRO)
Brasília – DF – Brasil
adriano.martins@serpro.gov.br

Abstract. *This paper describes the history of the emergence of the term cloud computing, its concepts from the bare metal to utility computing models and also include its services models and clouds models practiced by much of the current IT market worldwide. It also warns those involved in its implementation on the implications for IT as well as the forces of influence for a new concept that comes to the formation of this new model. Finally, discusses essentials standards for those who need to implement it.*

Resumo. *Este artigo descreve o histórico do surgimento do termo computação em nuvem, seus conceitos desde o bare metal até a computação utilitária passando também pelos seus modelos de serviços e de nuvens praticado pela grande parte do atual mercado de TI mundial. Além disso, alerta os envolvidos em sua implementação sobre as implicações para TI bem como as forças de influência de um novo conceito que surge para formação desse novo modelo. Por fim, trata de padrões de tecnologia essenciais para quem necessita começar sua implementação.*

1. O que é a nuvem?

O termo nuvem tem sido usado historicamente como uma metáfora para a internet. Seu uso foi originalmente derivado de sua descrição em

diagramas de rede como um delineamento de uma nuvem, usados para representar transportes de dados através backbones de rede até um outro ponto final do outro lado da nuvem. Esse conceito é datado do início do ano de 1961 quando o Professor John McCarthy sugeriu que a computação de compartilhamento de tempo poderia levar a um futuro onde o poder computacional e até aplicações específicas seriam vendidas através de um modelo de negócios utilitário.

Essa idéia era muito popular no final de década de 60. Entretanto, no meio da década de 70 ela foi abandonada quando se tornou claro que as tecnologias da informação da época não estavam aptas a sustentar um modelo desses de computação futurística.

Entretanto, com a virada do milênio, esse conceito foi revitalizado e foi neste momento de revitalização que a computação em nuvem começou a emergir nos círculos de tecnologia.

2. O surgimento da computação em nuvem

Computação utilitária pode ser definida como o provisionamento de recursos computacionais e de armazenamento como um serviço que pode ser medido, similar àqueles providos pelas empresas de públicas que prestam esses tipos de serviços. E é claro que isso não é uma nova idéia nem mesmo algo com um nível alto de inovação ou outro mesmo quebra de paradigma. Esse conceito novo de computação tem crescido e ganhado popularidade, tanto que empresas têm estendido o modelo de computação em nuvem provendo servidores virtuais em que departamentos de TI e usuários podem requerer acesso sob demanda.

Algumas empresas que estão entrando agora no ramo da computação utilitária usam principalmente para necessidades não críticas; mas isso está mudando rapidamente, já que questões de segurança e confiabilidade estão sendo resolvidas.

Algumas pessoas pensam que computação em nuvem é o próximo grande boom do mundo de TI. Outros acreditam que é somente outra variação de computação utilitária que foi repaginada na década passada como uma nova tendência.

Entretanto, não é somente a buzzword “computação em nuvem” que está causando confusão entre as massas. Atualmente, com poucos players de mercado praticando esta forma de tecnologia e mesmo analistas de

diferentes companhias definindo o termo diferentemente, o significado do termo tem se tornado nebuloso.

3. Evolução das Máquinas

É importante entender a evolução da computação para se ter a contextualização do ambiente que se fala hoje de computação em nuvem. Olhando para a evolução do hardware, desde a primeira geração até a quarta e atual, mostra como nós chegamos até aqui.

A primeira geração é de 1943, quando os computadores Mark I e Colossus foram desenvolvidos, eles foram construídos usando circuitos hard-wired e tubos a vácuo ambos usados durante a guerra.

A segunda geração é de 1946, ocasião em que foi construído o famoso ENIAC. Esse foi o primeiro computador reprogramável e capaz de resolver um grande leque de problemas computacionais. Foi construído com tubos termiônicos e teve aplicabilidade durante a guerra. Mas o que marcou a segunda geração foram os computadores com transistores, o que dominou o final dos anos 50 e início dos 60. Apesar de usarem transistores e circuitos impressos, eles eram caros e pesados.

A terceira geração foi consagrada pelos circuitos integrados e microchips e foi nesta era que começou a miniaturização dos computadores e eles puderam ser portados para pequenos negócios.

A quarta geração é a que estamos vivendo neste momento, que utilizamos um microprocessador que põe a capacidade de processamento computacional em um único chip de circuito integrado.

O hardware, entretanto, é somente parte desse processo revolucionário. Dele fazem parte, também, o software, redes e regras/protocolos de comunicação.

A padronização de um único protocolo para a Internet aumentou significativamente o crescimento de usuários on-line. Isso motivou muitos tecnólogos a fazerem melhorias nas atuais formas de comunicação e criação de outras em alguns casos. Hoje falamos de IPv6 a fim de mitigarmos preocupações de endereçamento na rede e incrementar o uso da comunicação na internet. Com o tempo, foi-se abstraindo os conceitos e criou-se uma interface comum de acesso à Internet que se usou de facilidade de hardware e software: o uso de web browser.

Com isso, começava a popularização do uso da rede e a disseminação em massa de conhecimentos para a humanidade, mesmo que de forma não intencional era o começo da "migração" do modelo tradicional para um modelo de nuvem. O uso de tecnologias como virtualização de servidores, processamento de vetores, multiprocessamento simétrico e processamento paralelo massivo impulsionaram ainda mais a mudança.

4. Definição de Cloud Computing

Cloud computing é um modelo que habilita de forma simplificada o acesso on-demand a uma rede, a qual possui um pool de recursos computacionais configuráveis (por exemplo, redes, servidores, storages, aplicações e serviços) que podem ser rapidamente provisionados, configurados e liberados com um esforço de gerenciamento mínimo e automatizado. Esse modelo de cloud provê alta disponibilidade e é composto de cinco características essenciais, três modelos de serviços e quatro modelos de implantação.

5. Características Essenciais

On-demand self-service: um consumidor pode unilateralmente provisionar recursos computacionais, como servidor dns ou storage, de acordo com sua necessidade, sem a obrigatoriedade de interação humana com o provedor de serviço.

Acesso a rede: acesso a rede permitida por diferentes mecanismos e heterogeneidade de plataformas clientes: Mobiles, laptops e Pdas.

Pool de Recursos: os recursos computacionais de um provedor são agrupados, a fim de servirem múltiplos consumidores num modelo multiuso, com recursos físicos e virtuais diferentes, provisionados e reprovisionados de acordo com a demanda do cliente. Há um senso de localização independente; o cliente não sabe exatamente onde estão localizados os recursos aprovisionados e nem tem o controle e conhecimento desse local. Os recursos normalmente são: processador, memória, banda de rede e máquinas virtuais.

Rápida elasticidade: capacidade de rapidamente e elasticamente provisionar recursos, e em alguns casos automaticamente, para

rapidamente aumentar os seus recursos e logo após o término voltar ao estado inicial. Para o usuário final, esta capacidade de crescer e provisionar mais recursos parece ser ilimitada e pode ser conseguida em qualquer quantidade e a qualquer tempo.

Serviço mensurado: automaticamente, sistemas em cloud controlam e otimizam recursos levando em conta a capacidade de medir em algum nível de abstração apropriado pelo cada tipo de serviço (p.ex: storage, processamento, banda de rede e usuários ativos). Recursos usados podem ser monitorados, controlados e reportados com transparência, tanto para o provedor quanto o consumidos dos serviços usados.

6. Modelos de serviços

A Computação em Nuvem está ligada a três áreas da TI: infraestrutura, plataforma e software. O que muitas vezes pode ser referenciado como formas, segmentos, estilos, tipos, níveis ou camadas de computação em nuvem.

Ao invés de se falar em diferentes funcionalidades providas, é melhor pensar em diferentes camadas, porque infraestrutura, plataforma e software logicamente construídas e subseqüentemente interligados dão um caráter mais arquitetural e de integração entre os níveis.

Como a entrega dos recursos de TI ou capacidades como um serviço é uma característica importante de Cloud Computing, as três camadas de arquitetura de Cloud Computing são:

1. Infraestrutura como serviço (IaaS);
2. Plataforma como Serviço (PaaS);
3. Software como Serviço (SaaS).

Figura 1: As três camadas de computação em nuvem: SaaS, PaaS, and IaaS

7. Infraestructure as a Service – IaaS (Infraestrutura como serviço)

Característica provida para o cliente que provisiona, processamento, storage, rede e outros recursos computacionais fudamentais onde o cliente está apto a implantar e rodar qualquer software, o que pode incluir sistemas operacionais e aplicações. O cliente não gerencia ou controla os recursos por trás dessa infraestrutura; contudo, tem controle sobre o sistema operacional, storage, aplicações e possibilidade de controle limitada a alguns tipos de componentes de rede como, por exemplo, firewall.

IaaS oferece recursos computacionais como processamento ou armazenamento, os quais podem ser obtidos como se fossem um serviço. Exemplos são a Amazon Web Services com seu Elastic Compute Cloud (EC2) para processamento e Simple Storage Service (S3) para armazenamento e Joyent o qual provê uma infraestrutura sob demanda

escalável para rodar web sites aplicações web de interface ricas. Provedores de PaaS and SaaS podem recorrer a ofertas de IaaS baseadas em interfaces padronizadas.

Ao invés de vender infraestrutura de hardware, provedores de IaaS oferecem infraestrutura virtualizada como um serviço.

Foster *et al* (2008) denomina o nível de hardware puro, como computação, armazenamento e recursos de rede como camada de fábrica. Virtualizando, recursos de hardware que são abstraídos e encapsulados e conseqüentemente ser expostos à próxima camada e aos usuários finais através da padronização de interfaces como recursos unificados na forma de IaaS.

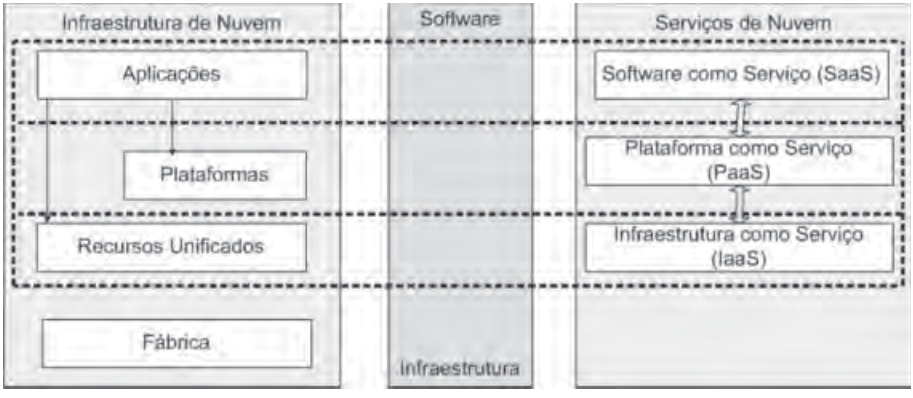


Figura 2: Arquitetura de nuvem relacionada com os Serviços de Nuvem

Já antes do advento da computação em nuvem, a infraestrutura já tinha sido colocada à disposição como um serviço por um bom tempo. Ela era referenciada como computação utilitária, o que é usada por muitos para denotar a camada de infraestrutura de computação em nuvem.

Entretanto, comparada aos recentes modelos de computação utilitária, IaaS denota sua evolução em direção ao suporte integrado dos três layers (IaaS, PaaS e SaaS) na nuvem.

Para as recentes ofertas de mercado de computação utilitária, ficou claro que para seus provedores terem sucesso, eles precisarão fornecer uma interface fácil de acessar, entender, programar e usar, como, por exemplo, uma API que habilita a fácil integração com a infraestrutura de clientes potenciais e

desenvolvedores de aplicações SaaS. Os centros de dados de provedores de computação utilitária serão utilizados suficientemente somente se tiverem abaixo de si massas críticas de dados de clientes e provedores de SaaS.

Como uma conseqüência de requisitos para o fácil e abstrato acesso à camada física da nuvem, virtualização da camada física e plataformas programáveis para desenvolvedores emergem como principais características das nuvens.

8. Platform as a Service – PaaS (Plataforma como serviço)

Característica provida pela nuvem que possibilita ao usuário portar dentro da nuvem aplicações produzidas pelo cliente ou de terceiros, usando linguagens de programação e ferramentas suportadas pela nuvem. O cliente não gerencia ou mesmo controla os ativos que compõem essa infraestrutura; entretanto, tem controle sobre a aplicação implantada dentro da nuvem e configurações de ambiente dentro da mesma.

Plataformas são camadas de abstração entre aplicações de software (SaaS) e a infraestrutura virtualizada. As ofertas de PaaS são alvo dos desenvolvedores de software que podem escrever suas aplicações de acordo com as especificações de uma plataforma em particular sem a necessidade de se preocuparem com a camada subjacente de infraestrutura de hardware. (IaaS).

Os desenvolvedores fazem o upload de seus códigos para a plataforma, o que deve aumentar o alerta pela monitoração e gerenciamento automático quando o uso da aplicação cresce. As funcionalidades providas pelo PaaS podem cobrir todas as fases de desenvolvimento de software ou talvez especializada em uma dada área como o gerenciamento de conteúdo

A camada PaaS da nuvem tem como base a padronização de interface da camada IaaS que virtualiza o acesso a recursos disponíveis, provê interfaces padronizadas e plataforma de desenvolvimento para a camada SaaS.

9. Software as a Service – SaaS (Software como serviço)

Serviço disponibilizado aos clientes que permite o uso de aplicações no provedor que rodam dentro da infraestrutura de nuvem. As aplicações estão acessíveis a qualquer cliente através de vários tipos de dispositivos, como uma interface web. O consumidor não gerencia ou controla o que há por

baixo da infraestrutura como rede, servidores, sistemas operacionais, storage ou até mesmo algumas aplicações específicas.

SaaS é software de um provedor que é possuído, entregável e gerenciável por este de forma remota e negociado de forma pay-per-use. SaaS é a camada mais visível em cloud computing para usuários finais, porque são as aplicações de software que são acessadas e usadas

Da perspectiva dos usuários, obter um software como serviço é mais motivador pelas vantagens de custo ao modelo de pagamento baseado em utilitário.

Os usuários mais comuns do SaaS normalmente não têm conhecimento nem controle sobre a camada abaixo, seja ela a imediatamente abaixo à plataforma como serviço ou os hardwares da infraestrutura como serviço. Entretanto, essas camadas abaixo são de grande relevância para o provedor de SaaS, porque elas são a base da infraestrutura, podendo ser vendidas e terceirizadas.

Como exemplo típico, cita-se que uma aplicação pode ser desenvolvida em uma plataforma qualquer e rodar em infraestrutura de terceiros. Ter-se uma plataforma e infraestrutura como serviço é um atrativo para os provedores de SaaS, pois pode aliviá-los de pesadas licenças de software e custos de investimentos em infraestrutura, além da flexibilidade. Isso também possibilita a corporação a focar em suas competências principais, que estão intimamente relacionadas ao negócio da empresa.

De acordo com os analistas de mercado, o crescimento pela inserção dentro do modelo de SaaS pelas empresas e a alta pressão de reduzir custos de TI são os maiores drivers pela alta demanda e crescimento do SaaS, também pelo crescimento por Cloud Computing nos próximos anos.

10. Modelos de Nuvens

Nuvem Privada: a infraestrutura de cloud é operada por uma organização e pode ser gerida pela própria organização ou por empresa terceira.

Nuvem comunitária: a infraestrutura de cloud é compartilhada por algumas organizações e abrange uma comunidade específica que tem os mesmo valores. (missão, requisitos de segurança, políticas e considerações de conformidade). Pode ser administrada pelas organizações ou por empresa terceira.

Nuvem Pública: a infraestrutura de cloud está disponível para o público geral ou um grupo de indústrias, ou é de propriedade de uma organização que vende os serviços da nuvem.

Nuvem Híbrida: a infraestrutura de cloud é composição de uma ou mais nuvens (privada, comunitária ou pública) que se mantêm como entidades únicas; entretanto, são ligadas pela padronização ou propriedade tecnológica, que permite portabilidade de aplicações e de dados.

11. Implicações para TI

Com este novo modelo, muitas implicações para uma empresa de TI podem surgir. Tais implicações derivam desta nova maneira de comunicação e estão direcionando mudanças na interatividade dos negócios. Hoje, os negócios precisam de respostas na velocidade da internet com novos serviços, funcionalidades diferentes e a quase obrigatoriedade de estar à frente de seu tempo e principalmente dos concorrentes.

Apesar disso, muitas corporações ainda não estão aptas a responder nesta velocidade e possuem o modelo tradicional de aquisições para compras de ativos de infraestrutura, o que lhes traz implicações negativas e compromete a agilidade de provisionamento. Muitas vezes, o equipamento está disponível; entretanto, há um processo burocrático de preparação e disponibilidade para uso; os recursos precisam estar prontos para uso com o aval técnico positivo. O que acontece é que se tem muitos processos que envolvem pessoal do storage, rede, segurança e algumas outras facilidades.

Normalmente essas dificuldades estão relacionadas a:

Planejamento de Capacidade: Para a maioria das organizações, não existe um planejamento de capacidade consistente, com planos de provisionamento de dados, serviços onde deve-se por tal equipamento ou mesmo como será o crescimento de tal aplicação. Isso é um impeditivo grande para implantação de um modelo baseado em cloud computing.

Equilíbrio das forças que o mercado demanda versus a utilização de ativos: A TI deve estar em sinergia em controlar seus gastos e ser responsável pelo negócio. A assertiva de que mais um servidor resolve o problema não colabora para o processo de construção desse novo modelo.

A empresa sempre quer algo rápido e consistente. O pessoal da área de negócio sempre vem com demandas para a área de TI que não têm orçamento aprovado, esperando que a TI consiga dar um jeito de produzir de qualquer

maneira. As questões relacionadas com segurança da informação são um elo que deve ser bem fortalecido, principalmente por se tratar de uma nova área de negócio e tecnologia que tem muitos processos e lacunas não preenchidas, sendo passíveis de construções muitas vezes não muitos sólidas e prejudiciais à empresa, exigindo cautela nos detalhes de serviços que serão prestados e nas transações executadas.

Outras implicações é que empresas dispostas a entrar nesse nicho e modificar a forma de orientação dentro do seu data center devem estar atentas à infraestrutura de TI, que deverá maximizar o gerenciamento e eficiência. E se gerenciar uma quantidade em massa de um data center não é uma das competências principais da empresa, ela deve delegar isso a uma empresa que:

Tem superioridade Econômica: Os grandes provedores de aplicações e serviços de TI, que compram tantos servidores, storages e outros muitos equipamentos de data center e que têm um enorme poder de negociação quando se fala de preço de hardware, licença de software e contratos de suporte.

Melhores Práticas: As maiores corporações têm investido não apenas em melhores processos, mas também investiram na construção de ferramentas de gerenciamento e administração que permitem a elas espalhar aplicações através de milhares de servidores de forma rápida.

Expertise em gerenciamento de capacidade dinâmica: Para grandes empresas, a produtividade de seus ativos é fundamental, assim como o custo de seus serviços é diretamente proporcional às despesas correntes do centro de dados. Quanto maior a produtividade que se pode tirar de cada metro quadrado de espaço, maior é a rentabilidade de um serviço. Por isso, é necessário uma monitoração de perto do consumo de recursos por cada aplicação dentro da infraestrutura disponibilizada.

12. Forças de influência

Quando se olha para um data center contemporâneo e moderno, não há como negar que eles são muito diferentes dos data centers dos de 10 ou 5 anos atrás. Certamente muitos o hardware existentes são de diferentes fabricantes, muitos têm heterogeneidade de servidores, como plataformas baixas e mainframes e aplicações. Nesta diversidade, provavelmente algum nível de organização deve-se ter notado ao longo do tempo. Entretanto, essa

organização, para muitas empresas, é sinônimo de caos, pois somente alguns profissionais têm o *modus operandi* em mente, e não é suficientemente sustentável para um data center dentro do paradigma de cloud computing.

Algumas forças sustentam esse novo conceito e são elas que vão diferenciar a tradicional maneira de hosting do novo modelo preconizado dentro de cloud:

Comoditização: É lamentável que para muitos o termo *commodity* tenha uma conotação diferente; ela é a evolução de produtos feitos a mão, um sinal de avanço de produção e existência de mercados com liquidez – para encurtar, progresso econômico. Comoditização depende de uma vasta rede integrada de infraestrutura de compradores, distribuidores, fornecedores e montadores. Quando se vê uma commodity, pode-se ver por trás uma rede complexa para sustentar a produção, o que inclui produzi-la, distribuí-la, apoiá-la e entregá-la. O processo de comoditização em um mercado maduro gradualmente leva o foco da competição entre organizações da funcionalidade para a qualidade, serviços complementares e, por último, o preço.

Fazendo um paralelo com o data center dentro do modelo de cloud computing, pode-se entender que ele será a rede e as demandas estão atreladas à sua atomicidade. Para o mundo externo, ele é uma única manufatura que possibilitará com transparência a execução de uma tarefa para produzir algum produto ou serviço através da padronização especializada por função.

Virtualização: Em computação, virtualização é um termo genérico utilizado para se referir à abstração dos recursos do computador. Uma definição seria: uma técnica para mascarar as características físicas dos recursos do computador de forma que outros sistemas, aplicações ou usuários finais possam interagir com tais recursos. Atrelado a esse conceito pode-se quebrar em mais duas forças: Miniaturização e Massificação. A primeira permite que em um servidor possa ter inúmeros sistemas operacionais virtualizados e massificados, ou seja, em várias outros equipamentos, referente ao segundo termo.

Portanto, o grau de abstração de uma solução de cloud computing depende também de quanto seu ambiente está virtualizado.

Independência de Aplicações e Sistemas Operacionais: A arquitetura que o ambiente de cloud provê hoje tem que estar habilitada para aceitar qualquer tipo de aplicação que o cliente deseja hospedar, já que ele não precisará acessar diretamente o hardware ou outros elementos internos da estrutura.

Liberdade de instalação de software ou hardware: Provisionamento tem que ser automático e sem burocracia. Não existe o processo de área e transações que é aplicado no modelo tradicional vigente nas empresas.

Integração: Integração é fundamental nessa nova abordagem, pois terá a necessidade de integrar nos vários níveis: software, hardware e middleware de forma global e esquecer a maneira transacional de integrar partições, sistemas em clustering, grids, barramentos e outros. O data center será um objeto atômico, não divisível para o mundo externo.

13. Modelos Tecnológicos

Alguns modelos tecnológicos devem ser padronizados para começo de construção de uma estrutura em nuvem. Os principais serão destacados nas subseções seguintes.

13.1. Modelo de Arquitetura

Neste modelo, deve ser definido de que forma será a integração entre as aplicações, serviços e outras nuvens externas. Deve-se mapear e fazer o projeto da infraestrutura comoditizada de hardware e software com definições macros de software e container dos softwares virtualizadores. Nesse momento, também terá que materializar a questão da disponibilidade e performance da solução através da topologia de rede e ligação dos servidores, questões de segurança como políticas de dados dentro e fora do ambiente produtivo. Enfim, este modelo será um arcabouço da solução com várias definições alto nível desde políticas escritas até garantia de segurança dos dados.

13.2. Modelo de Grid Computing

Computação em grade (do inglês Grid Computing) é um modelo computacional capaz de alcançar uma alta taxa de processamento, dividindo as tarefas entre diversas máquinas, podendo ser em rede local ou rede de longa distância, que formam uma máquina virtual. Esses processos podem ser executados no momento em que as máquinas não estão sendo utilizadas pelo usuário, assim evitando o desperdício de processamento da máquina utilizada.

Nos anos 90, uma nova infraestrutura de computação distribuída foi proposta, visando auxiliar atividades de pesquisa e desenvolvimento científico. Vários modelos desta infraestrutura foram especificados; dentre eles, a Tecnologia em Grade, em analogia às redes elétricas ("power grids") se propõe a apresentar-se ao usuário como um computador virtual, mascarando toda a infraestrutura distribuída, assim como a rede elétrica para uma pessoa que utiliza uma tomada, sem saber como a energia chega a ela. Seu objetivo era casar tecnologias heterogêneas (e muitas vezes geograficamente dispersas), formando um sistema robusto, dinâmico e escalável, onde se pudesse compartilhar processamento, espaço de armazenamento, dados, aplicações, dispositivos, entre outros.

Pesquisadores da área acreditam que a tecnologia de grades computacionais seja a evolução dos sistemas computacionais atuais, não sendo apenas um fenômeno tecnológico, mas também social, pois, num futuro próximo, reuniria recursos e pessoas de várias localidades, com várias atividades diferentes, numa mesma infraestrutura, possibilitando sua interação de uma forma antes impossível.

13.3. Modelo de Cluster

Um *cluster* é formado por um conjunto de computadores, que utiliza um tipo especial de sistema operacional classificado como sistema distribuído. Muitas vezes é construído a partir de computadores convencionais (*personal computers*), os quais são ligados em rede e comunicam-se através do sistema, trabalhando como se fossem uma única máquina de grande porte. Há diversos tipos de *cluster*. Um tipo famoso é o cluster da classe Beowulf, constituído por diversos nós escravos gerenciados por um só computador.

Quando se fala de cluster de um HD (Hard Disk), refere-se ao cruzamento de uma trilha com um setor formatado. Um HDD (hard disk drive) possui vários clusters que serão usados para armazenar dados de um determinado arquivo. Com essa divisão em trilhas e setores, é possível criar um endereçamento que visa facilitar o acesso a dados não contíguos, assim como o endereçamento de uma planilha de cálculos.

Existem vários tipos de *cluster*; no entanto, há alguns que são mais conhecidos, os quais são descritos a seguir:

Cluster de Alto Desempenho: Também conhecido como cluster de alta performance, ele funciona permitindo que ocorra uma grande carga de

processamento com um volume alto de gigaflops em computadores comuns e utilizando sistema operacional gratuito, o que diminui seu custo.

Cluster de Alta Disponibilidade: São clusters cujos sistemas conseguem permanecer ativos por um longo período de tempo e em plena condição de uso; sendo assim, pode-se dizer que eles nunca param seu funcionamento. Além disso, conseguem detectar erros se protegendo de possíveis falhas.

Cluster para Balanceamento de Carga: Esse tipo de cluster tem como função controlar a distribuição equilibrada do processamento. Requer um monitoramento constante na sua comunicação e em seus mecanismos de redundância, pois, se ocorrer alguma falha, haverá uma interrupção no seu funcionamento.

13.4. Modelo de Armazenamento

A abstração usada para armazenar dados em sistemas computacionais é o arquivo. Para que esses arquivos sejam acessados, modificados e que sejam criados outros arquivos, é necessária uma estrutura que permita tais operações. Essa estrutura recebe o nome de sistema de arquivos.

A motivação básica dos sistemas distribuídos é o compartilhamento de recursos e, no âmbito dos sistemas de arquivos, o recurso a ser compartilhado são os dados sob a forma de arquivos.

Um Sistema de arquivos distribuído, ou SAD, é um sistema de arquivos no qual os arquivos nele armazenados estão espalhados em hardwares diferentes, interconectados através de uma rede. Eles têm vários aspectos semelhantes aos dos sistemas de arquivos centralizados, além de operações de manipulação de arquivos, preocupações com redundância, consistência, dentre outros atributos desejados de um sistema de arquivos.

O SAD deve prover transparência nos seguintes contextos:

- De acesso: aplicações que acessam os arquivos do SAD não devem estar cientes da localização física deles.
- De localização: todas as aplicações devem ter sempre a mesma visão do espaço de arquivos.
- De mobilidade: com a movimentação dos arquivos, nem programas do cliente e nem tabelas de administração precisam ser modificadas, de modo a refletir essa movimentação.
- De desempenho: programas clientes devem executar satisfatoriamente, independente de variação de carga do serviço de arquivos.

• De escalabilidade: o serviço pode ser expandido por crescimento horizontal, e não vertical, de modo a se adequar à carga demandada e à capacidade da rede disponível.

14. Desafios para a nuvem

Os maiores desafios para as empresas serão o armazenamento de dados seguro, acesso rápido à internet e padronização. Armazenar grandes volumes de dados está diretamente relacionado à privacidade, identidade, preferências de aplicações centralizadas em locais específicos, levanta muitas preocupações sobre a proteção dos dados. Essas preocupações são pertinentes ao framework legal que deve ser implementado para um ambiente de computação em nuvem.

Outra preocupação é a banda larga. Computação em nuvem é impraticável sem uma conexão de alta velocidade, caso tenha-se problema de alta velocidade, a computação em nuvem torna-se inviável para as massas acessarem os serviços com a qualidade desejada.

Além disso, padrões técnicos são necessários para que se tenha todo o arcabouço de computação em nuvem funcionando. Entretanto, eles não estão totalmente definidos, publicamente revistos ou ratificados por um órgão de supervisão. Não existe essa padronização nem da academia nem do mercado. Até mesmo consórcios formados por grandes corporações necessitam transpor esse tipos de obstáculos e terem uma solução factível para que possam evoluir e gerar novos produtos que contribuam de alguma forma para a nuvem, sem essa definição precisa, ainda espera-se entregas em ritmo não tão acelerado.

Ao lado desses desafios discutidos anteriormente, a confiabilidade da computação em nuvem tem sido um ponto controverso nos encontros de tecnologia pelo mundo afora. Dada a disponibilidade pública do ambiente de nuvem, problemas que ocorrem na nuvem tendem a receber muita exposição pública; por isso, gerência e monitoração desse ambiente de TI são essenciais.

Em outubro de 2008, o Google publicou um artigo online que discute as lições aprendidas no armazenamento de milhões de clientes corporativos no modelo de computação em nuvem.

A métrica de disponibilidade do Google foi a média de uptime por usuário baseado nas taxas de erro do lado do servidor. Eles acreditavam que essa métrica de confiabilidade permitia uma verdadeira comparação com outras soluções. As medidas eram feitas para todas as requisições ao servidor para

cada usuário, a cada momento do dia, onde até mesmo um pequeno milissegundo de delay era registrado. O Google analisou os dados coletados do ano anterior e descobriu que sua aplicação Gmail está disponível mais de 99,9% do tempo.

E alguém pode perguntar como 99,9% de métrica de confiabilidade pode se comparar ao modelo convencional usado para email das corporações. De acordo com uma pesquisa da empresa Radicati Group, companhias com soluções de email convencionais têm normalmente de 30 a 60 minutos de parada não programada e mais de 36 a 90 minutos de parada programada por mês, comparados aos 10 a 15 minutos do gmail. Baseado nessas análises o Gmail é duas vezes mais confiável que a solução GroupWise da Novell e quatro vezes mais confiável que a solução da Microsoft, o Exchange, e ambas soluções requerem uma infraestrutura montada específica para suas soluções de email, centralizada e com regras próprias de ambientes.

Baseado nesses dados, Google estava suficientemente confiante para anunciar publicamente em outubro de 2008 que 99,9% de nível de serviço oferecido aos clientes empresariais Premier também se estenderiam ao Google Calendar, Google Docs, Google Sites e Google Talk. Como milhões de negócios usam as Apps Google, ele fez uma série de compromissos para aperfeiçoar a comunicação com seus clientes durante qualquer interrupção e fazer com que todas as questões estejam visíveis e transparentes através de grupos abertos.

Uma vez que o próprio Google roda suas aplicações dentro das plataformas Apps Google, o compromisso que eles fizeram tem sustentabilidade, pois as usam em suas operações do dia a dia. O Google lidera a indústria na evolução do modelo de computação em nuvem para se tornar uma parte do que está sendo chamado de Web 3.0, a próxima geração de Internet.

15. Conclusão

Neste artigo, viu-se a importância de saber a origem do termo nuvem para então poder entrar no ambiente de computação em nuvem. Foram examinadas também questões históricas de evolução do hardware, software, redes e a forma de comunicação e percebe-se o quanto ajudaram no crescimento da internet nos últimos anos, além de sua popularização com o acesso a browsers.

Após isso, foi tratado de forma mais pragmática o conceito profundo de computação em nuvem, explicitando-se seu modelo de negócio, modelo ser serviços, forças de influência e implicações para a área de TI.

Viu-se que a padronização, virtualização, processamento paralelo, distribuído e um modelo de negócio bem definido são essenciais para sair-se do campo teórico e buscar a aplicação efetiva em modelos que possam ser propagados para toda a sociedade da computação e administração de TI.

Além disso, verificou-se que existem mecanismos de cunho legais e técnicos que precisam ser condensados para que o mercado tenha um modelo único e denso dentro de cada empresa e possa ser dissipado não só entre governo, mas academias, meio privado e organismos internacionais.

Para finalizar, evidenciou-se que o Google com um arcabouço de gerência e monitoração bem definidos; conseguiu usufruir por completo o que a computação em nuvem pode proporcionar e mostrou que, se bem aplicado, esse conceito pode trazer mudanças significativas na maximização de recursos computacionais, além de superar a barreira daquela computação tradicional.

Referências

ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R.; KONWINSKI, A.; LEE, G.; PATTERSON, D.; RABKIN, A.; STOICA, I.; ZAHARIA, M. Above the Clouds: A Berkeley View of Cloud Computing. EECS Department, University of California, Berkeley, fevereiro 2009.

AMAZON. Elastic Compute Cloud. Disponível em: <http://aws.amazon.com/ec2/>. Acesso em: 18 fev. 2010.

FOSTER, I. What is the Grid? A Three Point Checklist. Argonne National Laboratory & University of Chicago, julho 2002.

MOHAMED, A. A history of Cloud Computing. ComputerWeekly.com, março 2009.

SUN MICROSYSTEMS, INC. Introduction to Cloud Computing Architecture. White Paper, 1ª edição, junho 2009a.

TANENBAUM, A. S.;STEEN, M. Distributed Systems Principles and Paradigms. Nova Jersey: Prentice Hall. 2002.

WANG, L et al. Cloud Computing: a Perspective Study. RIT Digital Media Library, 2008.

Modelo de Referência de Cloud

CETEC – membros
Luis Claudio Pereira Tujal - CETEC
SERPRO – Serviço Federal Processamento Dados

Resumo

Este trabalho tem o objetivo de sistematizar as fundações do modelo de computação em nuvem e a partir delas oferecer um direcionamento para o SERPRO na sua jornada nos rumos da modernização e da inovação. Computação em nuvem desponta no cenário tecnológico mundial amparado por tecnologias inovadoras, numa tentativa de viabilizar um velho anseio: ambientes computacionais seguros, eficientes, compartilhados e gerenciados.

Para a consecução de maturidade e da maximização de vantagens relativos ao paradigma da computação em nuvem vamos desenhar cenários compostos por soluções orientadas a serviços, focadas em ausência de estados, baixo acoplamento, modularidade e de interoperabilidade semântica. Estabelece-se, portanto, correlação de cloud computing e SOA (Sevice Oriented Architecture): SOA numa perspectiva holística e estratégica, como um padrão de arquitetura, e a computação em nuvem como uma opção ou mesmo instância de arquitetura, num viés tático, uma forma de resolver um problema.

*Outra tendência que aqui também se relaciona à computação em nuvem é a chamada Tecnologia da Informação Verde, a TI verde (**Green***

***IT**). Construir uma infraestrutura com base nos fundamentos da sustentabilidade e da racionalização no uso de recursos são dois conceitos que permeiam estes dois paradigmas de TI.*

1. Introdução

Para iniciar, permita-se o leitor um pequeno exercício de imaginação. Não importa o quão "nebuloso" seja o futuro, para a Web ele está inexoravelmente atrelado ao paradigma de serviços. A Internet de Serviços, uma plêiade de serviços de TI (tecnologia da informação), de telecomunicações, de media sendo ofertados, utilizados, recombinados, vendidos, redefinidos por uma rede mundialmente distribuída de provedores, *brokers*, agregadores, produtores e consumidores de serviços. Não é por acaso que as tecnologias de *web services* se tornam exponencialmente sólidas e populares. Também não é um capricho do destino que as arquiteturas orientadas a serviço estejam cada vez mais atrativas e consistentes. Este cenário conduz a mudanças culturais, uma verdadeira revolução na acepção da palavra, trazendo uma releitura e uma redefinição do mundo e da própria tecnologia da informação.

No tempo deste documento, ainda é um fato a supremacia do software proprietário como plataforma para se construir esse cenário de serviços. Mas o software de código aberto e gratuito vem rapidamente ganhando expressivo espaço no cenário da Internet de serviços. Há hoje soluções *open source* maduras que, com algum empenho daqueles que as adotam, em nada ficam a dever àquelas proprietárias.

Há ainda muitos serviços atrelados a itens do mundo físico, como a venda de equipamentos como iPods, MP10's, a venda de filmes em mídias Blu-Ray ou mesmo os ultrapassados discos DVD, a venda de Kindles e até mesmos os milenários livros em papel. Mas, há serviços como o EC2 (*Elastic Compute Cloud*) da Amazon, concebidos para se vender um pequeno centro de dados virtualizado, ao invés de um feixe de pulverizadas e complicadas vendas múltiplas de CPUs, monitores, teclados, discos rígidos, *switches*, sem falar em aluguel de salas, em contas de luz, etc. Ou seja, ao invés de sofrer toda a logística para se montar um ambiente para se oferecer um serviço na *web*, é possível comprar e se pagar pelo poder computacional, como se fosse água ou luz – este é o IaaS, (Infrastructure as a Service), ou a infraestrutura como serviço.

Cloud Computing, a computação em nuvem, vem se consolidando no cenário mundial como a grande opção capaz de suportar um modelo web global e aberto, orientado a serviços, proporcionando a granularidade necessária para o atendimento das demandas nas cadeias de valores de uma crescente massa de negócios realizados em escala planetária. Visceralemete relacionada à computação em *grid*, *cloud* se constrói sobre alguns dos conceitos básicos, dos produtos e da infraestrutura desta modalidade da ciência da computação. Trabalhá-los em separado seria como querer começar um projeto de um marco zero, e o grande exemplo dos *foundries* na indústria dos tigres asiáticos está aí para mostrar o quão equivocada pode ser uma postura atrelada a uma visão auto-suficiente. Igualmente importante é a base representada pela computação em *cluster*, um dos blocos básicos da infraestrutura de nuvem.

Um ponto de vital importância que deve ser salientado é que a partir de pesquisas junto a lideranças em governos e na indústria de TI pôde ser mapeada a maior necessidade ainda por ser solucionada: **a falta de consenso**, que deriva da multiplicidade de padrões e de terminologias. Exatamente por não haver entendimento entre os atores da nuvem, a segurança (*security*) e a privacidade (*privacy*) não podem ser adequadamente garantidas. Segundo pesquisa da iniciativa americana conhecida por GTRA (Government Technology Research Alliance), corroborada por um estudo da IDC (International Data Corporation) para computação em nuvem, emtorno de 75% (setenta e cinco por cento) dos CIOs têm nestes requisitos (*security* e *privacy*) a sua preocupação primordial. Neste direcionamento o NIST (National Institute of Standarts and Technology) emitiu no segundo semestre do ano de 2009 modelos de uso para agências de governo desejosas de se aventurar nos auspícios da *cloud computing*.

Estabelecendo-se a clivagem governo eletrônico brasileiro, válida para suas esferas federal, estadual e municipal, o presente trabalho propõe uma construção baseada em organizações virtuais, erigidas com base em *private clouds*, ou nuvens privadas, que poderão interagir entre si e com outras nuvens dentro de um paradigma de federação, através da consubstanciação de regras em acordos de nível de serviço (ANSs, do inglês SLAs, ou *service level agreements*). Apresenta-se aqui um modelo para uma arquitetura de nuvem aberta (*open*) e pautada em serviços (*service oriented*), que no caso do governo são vistos sob a perspectiva da tríade serviços públicos, serviços de utilidade publica e serviços privados.

2. Taxonomia de *cloud*

A exemplo da iniciativa de *Carl Von Linée* para a biologia, é entendimento do autor que estabelecer conceitos sem ambiguidades é de importância epistemológica fundamental. Numa tentativa de firmar *cloud computing* como um ramo da ciência da computação, é de importância essencial que se estabeleçam termos que permitam a sua distinção de outros ramos. Daí a relevância da atividade de taxonomia como a porta de entrada para este novo mundo que ora se descortina.

A definição de computação em nuvem está longe de ser consensual entre os diversos atores que participam de seu ciclo de vida. Uma pequena pesquisa na *engine* de busca web preferida do leitor o levará uma plêiade de conotações e cosmovisões sobre cloud computing. Cada segmento da indústria do hardware e do software tenta estabelecer *cloud* conforme seus interesses e suas expectativas para o mercado, para a pesquisa, para a inovação. A tentativa neste trabalho é de disciplinar cientificamente os conhecimentos sobre esta modalidade da computação, criando um arcabouço empírico e uma modelagem conceitual consistente que permita estruturar diretrizes da computação em nuvem maduras o suficiente para serem adotadas pelo governo eletrônico brasileiro.

Uma primeira visão para *cloud* pode ser facilmente obtida pela Wikipédia, Em http://pt.wikipedia.org/ wiki/Computação_em_nuvem encontra-se o seguinte: "*cloud computing* é um conjunto de serviços acessíveis pela internet que visam fornecer os mesmos serviços de um sistema operacional. Esta tecnologia consiste em compartilhar ferramentas computacionais pela interligação dos sistemas, semelhantes as nuvens no céu, ao invés de ter essas ferramentas localmente (mesmo nos servidores internos). O uso desse modelo (ambiente) é mais viável do que o uso de unidades físicas".

O laboratório de tecnologia da informação do NIST (National Institute of Standards and Technology) fornece uma definição clara e concisa que adotaremos neste trabalho.

Pelo NIST tem-se: "Computação em Nuvem é um modelo do tipo 'pague pelo uso' para possibilitar acesso de rede disponível, conveniente e sobre demanda a um pool compartilhado de recursos computacionais configuráveis (e.g., servidores, armazenamento, redes, aplicações, serviços) que podem ser rapidamente aprovisionados e liberados com o mínimo esforço gerencial ou de interação de provedor de serviços."

A definição original pode ser vista abaixo:

> *"Cloud computing is a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."*

Este modelo de computação em nuvem compreende cinco características chave:

a) **auto-serviço sob demanda**, ou *on-demand self-service*. Sem a necessidade de interação humana com cada provedor de serviço o consumidor de serviços em nuvem (*cloud consumer*) tem a possibilidade de alocar, conforme necessidade, as capacidades computacionais (como por exemplo tempo de processador ou *storage* de rede).

b) **acesso de rede ubíquo**. As capacidades estão em rede e são acessíveis por meio de mecanismos padrão que promovem o uso por meio de plataformas cliente heterogêneas, sejam elas *gordas* ou *magras*.

c) ***pooling* de recursos independente de localização.** Os recursos computacionais do provedor são organizados em *pools* de recursos para servir a todos os consumidores através de um modelo *multi-tenant* (uma única instância do software é executado em um servidor, que atende múltiplas organizações de clientes (*tenants* ou inquilinos). Neste modelo diferentes recursos físicos e virtuais são dinamicamente atribuídos de acordo com a demanda, sendo que o consumidor tem pouco ou nenhum controle ou conhecimento acerca da localização dos recursos providos.

d) **elasticidade rápida.** As capacidades podem ser rápida e elasticamente alocadas para escalar em forma de aumento, e rapidamente liberadas para escalar como redução. Para o consumidor há a percepção de recursos infinitos, que podem ser comprados em qualquer quantidade a qualquer tempo.

e) **pago pelo uso.** As capacidades são tarifadas com o emprego de um modelo mensurável, com um *billing* ou cobrança baseado em serviço consumido ou em propaganda veiculada, que promove a otimização do uso de recursos.

Para ilustrar a definição apresentada, observe-se a ilustração 1 (um) abaixo. Pode se entender que a definição acima enseja a organização de

três níveis tróficos bem caracterizados no ecossistema de computação em nuvem.

a) O **Cloud Provider**, ou provedor de infraestrutura de nuvem, que é o ator ou conjunto de atores responsáveis pelo tecido, o fabric, ou ambiente físico. Este conjunto de empresas, abstraídas como provedor, opera no nível do hardware e equipamentos de rede e transmissão e a sua oferta pode ser tratada como computação utilitária entregue ao provedor de serviços.

b) O **Service Provider**, que vai consumir infraestrutra para oferecer serviços. Estes serviços podem ser essenciais, como software, infraestrutura, armazenamento e podem ser mais sofisticados, como governança, segurança e testes. A figura 1 (um) ilustra o caso de SaaS, *Software as a Service*, software ou aplicações como um serviço, que é talvez o mais palatável serviço de nuvem para uma primeira abordagem.

c) O **Cloud Service Consumer**, consumidor ou usuário final, que através de uma aplicação web, ofertada por um *browser*, consome *cloud computing*.

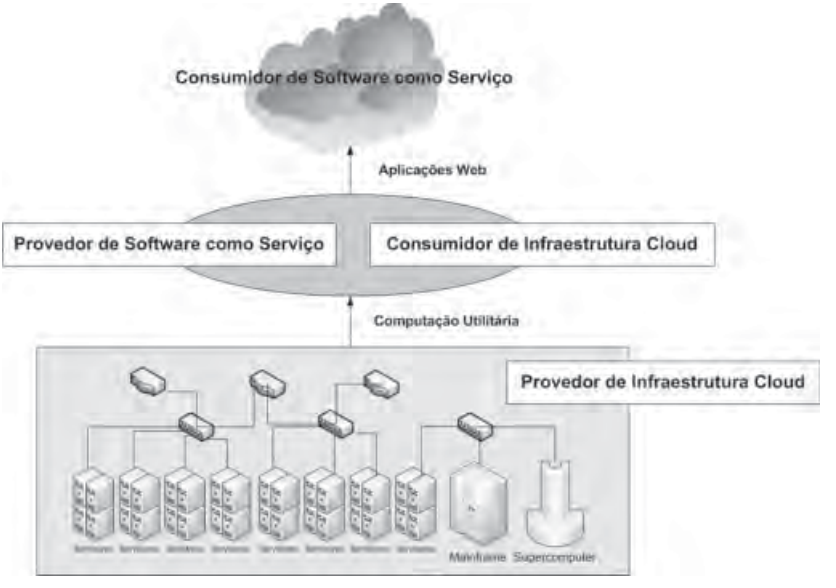


Figura 1. Topologia básica de nuvem

A seguir fornece-se uma versão resumida dos termos empregados para se construir o jargão desta modalidade de ciência da computação.

Além de conceitos como Armazenamento, Desempenho, Framework e Resiliência, para citar apenas alguns de uso comum nas sendas da tecnologia da informação, apresenta-se a lista abaixo, que tem grande vinculação com cloud computing conforme acima definida.

cloud app - uma aplicação de software que nunca é instalada em uma máquina local - é sempre acessada pela Internet.

cloud arcs – nome curto para arquiteturas nuvem. Desenhos para aplicações de software que podem ser acessadas e utilizadas através da Internet. (Cloud-chitecture é muito difícil de pronunciar).

cloud bridge - executar uma aplicação de tal forma que seus componentes sejam integrados em ambientes de cloud múltiplos (que poderia ser qualquer combinação de nuvens internas/privadas e externas/públicas).

cloudcenter - uma grande empresa, como a Amazon, que aluga a sua infraestrutura.

cloud client - dispositivo de computação para computação em nuvem. Versão atualizada do thin client.

cloud enabler - fornecedor que provê a tecnologia ou o serviço os quais permitem que um cliente ou outro fornecedor tire proveito da computação em nuvem.

cloud envy - usado para descrever um vendedor que se insere no contexto da computação em nuvem pela remarcação pura e simples dos serviços existentes, isto é, sem agregar valor ou código.

cloud OS – também conhecido por plataforma como serviço (platform-as-a-service - PaaS). Como exemplo tem-se o Google Chrome.

cloud portability - a capacidade de mover os aplicativos e dados associados através de vários ambientes de computação em nuvem.

cloud provider – torna o armazenamento ou o software disponível para outras entidades através de uma rede privada ou na rede pública (como a Internet.)

cloud service architecture (CSA) - uma arquitetura em que as aplicações e componentes de aplicação agem como serviços na Internet

cloudburst - o que acontece quando a nuvem tem uma falha ou violação de segurança e os dados se tornam indisponíveis.

cloud as a service (CaaS) - um serviço de computação em nuvem que foi aberto em uma plataforma na qual outras pessoas possam trabalhar, evoluir e construir.

cloudstorm – conectar múltiplos ambientes de computação em nuvem. Algumas vezes conhecido como rede de computação em nuvem, ou *cloud network*.

cloudware - software que permite a construção, implantação, execução ou gerenciamento de aplicações em um ambiente de computação em nuvem.

cloudwashing – colocar a palavra "cloud" em produtos ou serviços que a empresa possui.

compute unit - ferramenta de medição utilizada para comparar e contrastar diferentes servidores virtuais (instâncias). Um EC2 Compute Unit fornece a capacidade da CPU equivalente a um 1,0-1,2 GHz 2007 Opteron ou 2007 Xeon. Este também é o equivalente a um antigo processador de 2006 1,7 GHz Xeon.

external cloud - um ambiente de computação em nuvem que é externo aos limites da organização.

funnel cloud – debate acerca de computação em nuvem que prossegue de forma reorrente e diletante, sem jamais se tornar ação (jamais toca o chão, i.e.).

hybrid cloud - um ambiente computacional que combina ambientes de computação em nuvem tanto privados quanto públicos.

internal cloud – também chamado de uma nuvem privada. Um ambiente do tipo computação em nuvem que existe dentro dos limites de uma organização.

private cloud – uma nuvem interna atrás do firewall da organização. O departamento de TI da empresa fornece softwares e hardware como serviço aos seus clientes - as pessoas que trabalham para a empresa. Fornecedores são adeptos deste conceito.

public cloud – um ambiente de cloud computing que está aberto para uso do público em geral.

roaming workloads- o produto backend de cloudcenters.

vertical cloud - um ambiente de computação em nuvem otimizado para uso em uma determinada indústria vertical.

virtual private cloud (VPC) - semelhante a VPN, mas aplicada a computação em nuvem. Pode ser usado como bridge (ponte) entre ambientes e nuvem privados e públicos.

3. Anatomia de *cloud*

O paradigma da nuvem se baseia na orientação a serviços, focando:
(a) o baixo acoplamento,
(b) a ausência de estados (statelessness),
(c) a modularidade e
(d) a interoperabilidade semântica.

Há várias abordagens de computação em nuvem, uma verdadeira diversidade semântica, que resulta na prática a veiculação de diversos modelos de implantação. Pensando-se numa linha comum à diversidade "doutrinária", definem-se abaixo os deployment models (modelos de implantação) mais representativos que o mercado tem adotado.

1. Nuvem Privada ou Interna. A infra-estrutura de nuvem é de propriedade ou alugada por uma única organização e é operada exclusivamente para essa organização.

2. Nuvem Comunitária. A infraestrutura de nuvem é compartilhada por diversas organizações e suporta uma comunidade determinada com preocupações em comum (por exemplo, considerações sobre compatibilidade, requisitos de segurança, política e finalidade).

3. Nuvem Pública. A infraestrutura de nuvens é propriedade de uma organização que vende serviços da nuvem para o público em geral ou diretamente para um grupo de grandes indústrias.

4. Nuvem Híbrida. A infraestrutura de nuvens é uma composição de duas ou mais nuvens (privada, comunidade, ou pública) que mantêm sua individualidade, mas estão interligadas por uma tecnologia padronizada ou proprietária que possibilita a portabilidade de informações e de aplicações (por exemplo, cloud burst – vide taxonomia, no item 2 [dois] acima).

Esquematicamente, podem ser observados esses conceitos na figura 2 (dois) a seguir. Neste ponto, cumpre ressaltar que para a operação de uma nuvem e para seu relacionamento com outras nuvens devem ser observados os acordos de nível de serviços, ou ANS, que representam a qualidade e quantidade que pode ser exigida de cada serviço e as faixas de tolerância para se buscar alternativas.

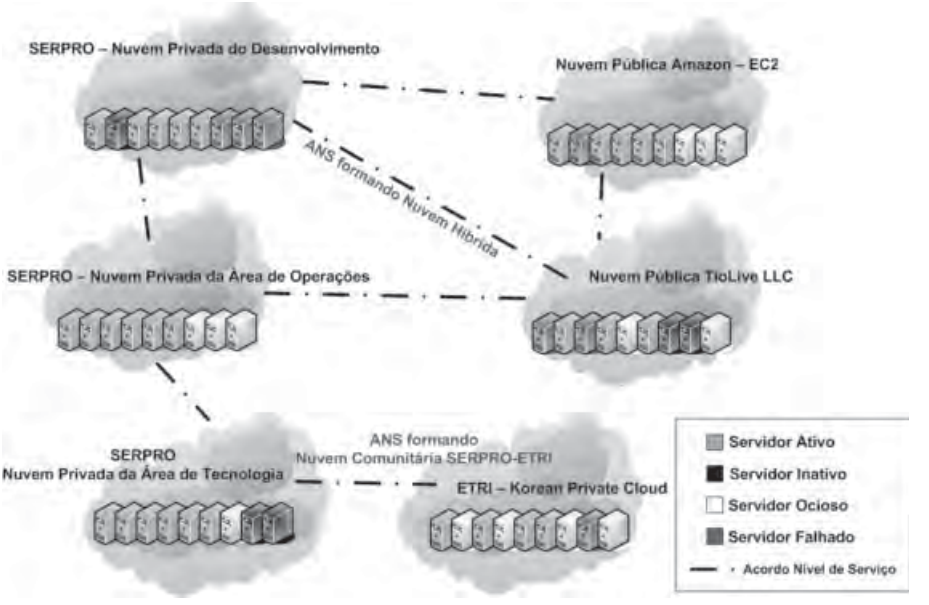


Figura 2. Exemplo de modelos de implantação

Na figura 2 (dois) acima pode se observar que os acordos de nível de serviços determinam em que situações, por exemplo, a Nuvem privada do desenvolvimento do SERPRO poderá utilizar o poder computacional ocioso da nuvem privada da área de operações, da nuvem pública da TioLive, da nuvem híbrida formada com a TioLive, e assim por diante. Observe-se que nuvens comunitárias e híbridas se respaldam nos ANS e são materializadas pelas tecnologias empregadas de forma adicional a um modelo público e/ou privado.

À medida que a computação em nuvem se consolida no panorama da tecnologia do século XXI, grande são os debates e as opiniões sobre a maneira de descrevê-la na qualidade de modelo computacional. Modelos de Capacidade e Maturidade vêm sendo sugeridos e publicados pelos diversos fabricantes de equipamentos e pelos provedores, mas, até o momento, estes modelos se encaixam adequadamente em suas próprias plataformas e em seus nichos de mercado.

Para se descrever melhor nossa computação objeto, a nuvem, define-se uma pilha de categorias ou padrões de tecnologia, que representam padrões

de feixes de serviços básicos, como doravante serão referidos. São estes padrões de serviços básicos e os seus inter-relacionamentos que vão compor o modelo de serviços. Uma vez mais, observe-se que não há um consenso sobre a quantidade destes padrões, esses serviços básicos oferecidos pela nuvem, nem tampouco há acordo sobre suas interações e seus desdobramentos. Há grande tendência em se estabelecer ao menos 3 (três) categorias básicas, as de (1) Infraestrutra, (2) Plataforma e (3) Software como serviços. Considerando este ponto de partida, o presente trabalho define 6 (seis) grandes categorias ou padrões de tecnologia de computação em nuvem, os três vistos anteriormente e mais três categorias que os autores entendem se destacarem por estarem presentes de forma transversal à infraestrutura, plataforma e software. São elas a (4) Integração, (5) Segurança e (6) Governança e Gerenciamento como serviços. A estas seis categorias essenciais podem ser destacadas 6 (seis) especializações ou categorias desdobradas, as quais, no futuro, poderão ser elevadas à categorias, à medida que a complexidade e a divisão de tarefas se intensificam, justificando a sua existência autônoma relativa às demais componentes.

Em seguida realiza-se a categorização e fornece-se uma breve descrição.

1. Infrastructure-as-a-service

Infraestrutura como serviço corresponde à entrega de um centro de dados na forma de serviço remotamente hospedado. Os recursos computacionais de um datacenter, como servidores físicos, switches, roteadores e o software básico destes podem ser oferecidos de modo habilitado para o acesso remoto. Deste modo, IaaS pode ser consubstanciado por um serviço do tipo "mainstream cloud" com interfaces e métricas definidas, ou por um serviço que provê acesso à máquina física e seu software básico.

1.1. Storage-as-a-service

Armazenamento como um serviço, (também conhecido como espaço em disco em demanda), é a capacidade de alocar o armazenamento que existe fisicamente em um sítio remoto e que se oferece logicamente como um recurso de armazenamento local para qualquer aplicação que exija armazenamento. Este é um componente amplamente aceito da computação

em nuvem e é um padrão que se relaciona com a maior partedos demais serviços básicos da nuvem.

1.2. Database-as-a-service

O padrão banco de dados como um serviço fornece a capacidade de alocar serviços de um banco de dados hospedado remotamente, entregando-os logicamente, em termos de desempenho e funcionalidades, como se o banco fosse local.

2. Platform-as-a-service

Plataforma como um serviço corresponde à entrega aos consumidores de uma plataforma completa e remotamente hospedada, incluindo o desenvolvimento de aplicações, de interfaces e de bancos de dados, além de armazenamento e ambiente de testes. Adotando o modelo de compartilhamento de tempo (time-share), permite que se pratiquem baixos preços de subscrição para se oferecer a possibilidade de se criar aplicações corporativas de uso local ou sob demanda.

2.1. Simulation-as-a-service

Simulação como serviço oferece uma perspectiva que abrange desde testes unitários e integrados de aplicações até a composição de cenários possíveis em simulações e projeções para se oferecer prognósticos e predições com determinada confiabilidade e certeza.A oferta de testes como serviço permite que sejam utilizados software e serviços remotamente baseados para se promover testes em sistemas locais ou entregues pela nuvem. Uma característica interessante consiste na possibilidade de recorrência, quando o próprio software de nuvem testa a si mesmo e aos seus testes.

3. Software-as-a-service

Software ou aplicação como um serviço (AaaS ou SaaS) consiste em qualquer aplicação fornecida para um usuário final através da plataforma web, tipicamente por meio de um browser. Estas aplicações podem ser corporativas

como as do Salesforce, Rackspace e Amazon ou simplesmente para automação de escritórios como as do Google.

3.1. Information-as-a-service

Informação como um serviço consiste na capacidade de se consumir qualquer espécie de informação armazenada remotamente, de forma distribuída e desconhecida, usando uma interface bem definida, como uma API.

3.2. Process-as-a-service

Processo como um serviço é um recurso remoto capaz de associar e combinar outros recursos como outros serviços e dados, quer hospedados dentro do mesmo recurso de nuvem, quer hospedados remotamente, de modo a se criar processos de negócios. Um processo de negócio pode ser visto como uma meta-aplicação que abrange sistemas, serviços chave e informações que são combinados de forma seqüencial a fim de se formar processos, que podem ser mais facilmente alterados do que aplicações conferindo a agilidade necessária a motores de processo que precisam ser entregues sob demanda.

3.3. Communication-as-a-Service

O CaaS, ou comunicação como um serviço consiste numa solução de outsourcing para a entrega sob demanda de serviços de comunicações corporativos. Constitui-se na gestão do hardware e software envolvido no fornecimento de serviços como Voz sobre IP, VoIP, Mensagens instantâneas e conferências por vídeo.

4. Integration-as-a-service

Integração como um serviço corresponde a entrega das funcionalidades e capacidades de tecnologias de integração corporativa de aplicações como um feixe de serviços de integração veiculados por meio da nuvem. Estes serviços compreendem as interfaces entre sistemas e aplicações, a intermediação de semânticas, vários níveis mais abstratos de controle de fluxo e o projeto de integração.

5. Security-as-a-service

Segurança como um serviço corresponde às capacidades para se entregar serviços de segurança essências de modo remoto, através da Internet. Embora os serviços de segurança típicos fornecidos são rudimentares, os serviços mais sofisticados, tais como gerenciamento de identidade estão se tornando disponíveis.

6. Management-as-a-service e Governance-as-a-service

A Gestão como serviço se constitui em qualquer serviço sob demanda que oferece a capacidade de gerir um ou mais serviços fundamentais na nuvem, como a topologia, a utilização de recursos, virtualização e gestão de *uptime*. Com a evolução da tecnologia, sistemas de governança estão se tornando disponíveis, oferecendo serviços relativos a capacidade de impor políticas definidas para dados e serviços, evoluindo da simples gestão para governança como serviço.

De forma esquemática, apresentamos estes conceitos ilustrados na figura 3 seguinte.

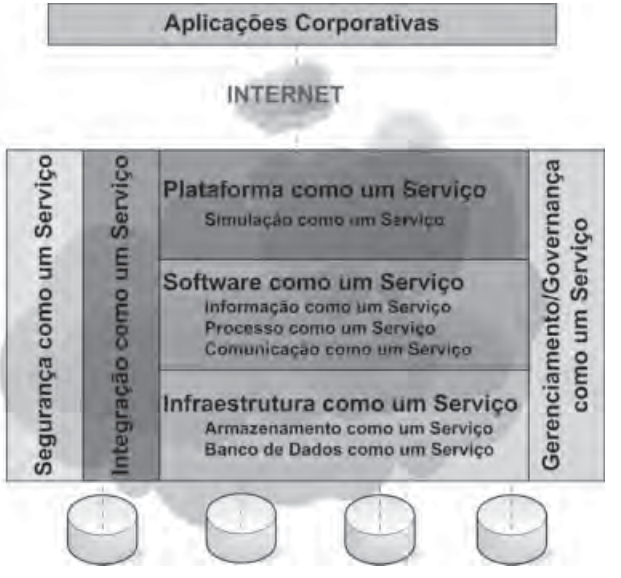


Figura3. Os padrões de serviços básicos em *cloud*

Outra questão relevante para esta caracterização da anatomia da computação em nuvem diz respeito ao grau de abertura adotado para o modelo tecnológico. Sob uma perspectiva geral podem ser especificados 4 (quatro) níveis de abertura ou *openess*, representados na figura 4.

1. Closed or Proprietary Cloud – Nuvem fechada ou proprietária. Nesta categoria há uma utilização majoritária de tecnologia proprietária na construção e operação da nuvem. O acesso é normalmente restrito e geralmente este grau de abertura vem associado a um modelo de implantação privado.

2. Open Cloud – Nuvem aberta ou de padrões abertos. Utiliza formatos abertos, tais como o OVF, o *open virtualization format*, ou formato aberto de virtualização. Também se vale de APIs abertas para a construção de interfaces. O licenciamento, contudo, é proprietário para a maior parte das soluções empregadas.

3. Open Source Cloud – uma extensão do modelo visto no item anterior. Neste caso, é massivo o emprego de soluções *open source*, ou de código aberto.

4. Free Cloud – Neste caso emprega-se cumulativamente liberdade e gratuidade. São adotados formatos (open formats) e APIs (open APIs) abertos, código de fonte aberto (open source code), com licenciamento livre, e dados abertos (open data). Usualmente este formato é representado por um modelo de implantação como nuvem pública, acrescido da não cobrança de tarifas de uso.

Figura 4. Grau de abertura da nuvem computacional

Cumpre observar que, da mesma forma que num *grid*, na nuvem é possível se atender às necessidades de compartilhamento coordenado de recursos e solução de problemas com o conceito de **organizações virtuais** dinâmicas e distribuídas entre várias instituições.

Uma organização virtual (OV) para *cloud* se constitui num conjunto de pessoas, físicas e/ou jurídicas, indivíduos e/ou instituições, definido por acordos de nível de serviços (ANS) e regras de governança específicas. Congrega pessoas com questões e requisitos tecnológicos comuns, compreendidas nas seguintes clivagens:

(a) relacionamentos muito flexíveis, variando de cliente-servidor a ponto a ponto;

(b) controle granular e distribuído por muitos *stakeholders*, com provedores de recursos e clientes definindo precisamente as condições de ocorrência, quem pode e o que é compartilhado;

(c) compartilhamento não apenas em troca de arquivos, mas também acesso a computadores, softwares, informações, sensores, redes e outros recursos;

(d) diversos modos de uso, variando de usuário único a multiusuário, de sensível a desempenho a sensível a custo, sujeito a demandas de QoS, agendamento, co-alocação e contabilidade.

A figura 5 em seguida ilustra um caso onde organizações virtuais são constituídas.

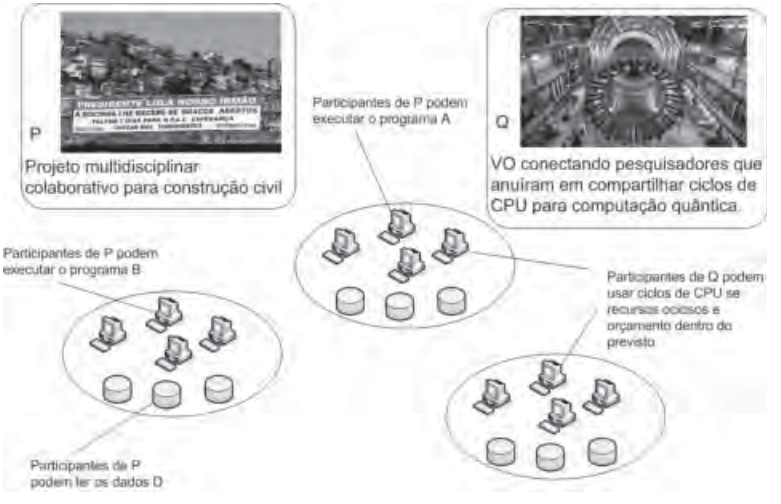


Figura 5. Exemplo de organizações virtuais.

Com relação aos protocolos de nuvem faz-se presente a mesma sistemática observada para grades computacionais. Os protocolos, seja em business grids, seja em cloud computing, exercem a mediação entre os comportamentos expressos pelas aplicações e o tecido ou trama formada pelas tecnologias. Sob uma perspectiva holística, distinguem-se protocolos de **conectividade**, responsáveis pela comunicação e segurança, protocolos de **recursos**, com tarefas relacionadas a armazenamento em dispositivos, a gravações e recuperações de dados em SGBDs e a busca de informações, e protocolos **coletivos**, com a incumbência de realizar os testes, a gestão, o encadeamento de processos e a governança de múltiplos recursos. Este assunto será melhor explorado ao tratar-se a arquitetura, no tópico 5 deste trabalho.

Para que os sistemas e aplicações de *cloud* sejam estruturados em acordo com os princípios de uma arquitetura orientada a serviços faz-se a escolha de w*eb services* para compor a infraestrutura e o framework. Outrossim, é assumido que as interfaces de serviços estarão definidas em WSDL (*Web Service Description Language*) em suas versões 1.1 e 2.0.

O XML é o padrão para descrição e representação, mas sua adoção é tornada flexível em determinados contextos onde se necessita desempenho. O protocolo SOAP é o formato básico de troca de mensagens para serviços em padrão de arquiteturas abertas de serviços, (*open services architecture*). As definições de serviço estão em consonância com o processo de interoperabilidade em *web services*, o WS-I, (*web service inspection*).

Em questões de segurança adotam-se os padrões OASIS, incluindo o *Security Assertion Markup Language* (SAML) v2.0, o *eXtensible Access Control Markup Language* TC v2.0 (XACML). Está em evolução o trabalho no rumo ao *Cross-Enterprise Security and Privacy Authorization* (XSPA*) Profile of* XACML v2.0.

4. Fisiologia de *cloud*

Na visão anatômica abordaram-se definições e propriedades essenciais de uma nuvem. Em fisiologia será detalhado como uma *cloud* funciona e como suas tecnologias podem ser construídas e aplicadas, complementando a visão orientada a protocolos. Explicita-se como as tecnologias de *cloud* e de *web services* devem ser alinhadas para se trabalhar na linha de uma arquitetura orientada a serviços (SOA)

4.1. Web Services

O termo web service descreve um importante paradigma de computação distribuída heterogênea que tem seu foco em padrões baseados em Internet, como o XML. *Web services*, ou WSs, definem uma técnica para descrever componentes de software a serem acessados, métodos para acessá-los e métodos de descoberta ou pesquisa que tornam possível a resolução de provedores de serviço pertinentes. WSs são imunes a linguagens de programação, a modelos de desenvolvimento e a sistemas de software.

Web services são definidos pelo W3C e por outros organismos de padronização e formam a base de maciço investimento da indústria da computação. Em *cloud computing* interessam basicamente três padrões: SOAP, WSDL e WS-I, conforme mencionado no capítulo anterior.

• O SOAP, ou *Simple Object Access Protocol*, provê um meio para troca de mensagens entre um provedor de serviço e um requisitante por serviço. Este protocolo consiste num simples componente de empacotamento para dados XML, que define convenções de chamada de procedimento remoto (RPC, *remote procedure call*) e de troca de mensagens. Enfatiza-se que SOAP não esgota as possibilidades de comunicação: quando desempenho é crítico é necessário se adotarem soluções de troca de mensagens que executam sobre protocolos especializados de rede.

• O WSDL, *Web Service Description Language*, consiste num documento escrito em XML que além de descrever o serviço, especifica como acessá-lo e quais as operações ou métodos disponíveis. WSDL descreve web serviços como um conjunto de endereços de rede, ou portas, operando em mensagens contendo cargas de dados que podem ser orientadas a documentos ou mesmo RPC. É um protocolo extensível, permite a descrição dos endereços de rede e da representação concreta de suas mensagens para uma diversidade de formatos de mensagem e de protocolos de rede.

• O WS-I compreende uma linguagem simples e as relativas convenções para a localização das descrições de serviços publicadas por um provedor de serviço. Descrição de serviço é normalmente uma URL para um documento WSDL, ocasionalmente pode ser uma referência a uma entrada em registro UDDI (*Universal Description, Discovery and Integration*). Um documento WSIL contém uma coleção de descrições de serviços e *links* para outras fontes de descrições. Um link é uma URL para outro documento WS-I e, às

vezes, uma referência para uma entrada UDDI. Observe-se que documentos WSIL podem ser organizados em outras formas de indexação.

O framework de WSs traz duas grandes vantagens para o propósito dos *clouds*. Primeiro, para suportar a pesquisa dinâmica e a composição de serviços em ambientes heterogêneos são necessários mecanismos para registrar e descobrir definições de interfaces e descrições para pontos finais de implementações, bem como mecanismos para geração dinâmica de *proxies* baseados em *bindings* para interfaces específicas. Segundo, a franca adoção de mecanismos WS significa que um framework baseado em WSs pode se beneficiar muito com o grande número de ferramentas e de serviços existentes.

4.2. *Open Cloud Services*

Neste tópico serão tecidas considerações acerca da utilidade compreendida numa arquitetura de *cloud* aberta orientada a serviços, quais são as características essênciais dos serviços e a da importância de se virtualizar estes.

Na visão orientada a serviços pode se bissectar o problema da interoperabilidade num subproblema de definição das interfaces de serviço e em outro de se buscar um acordo dentro de um conjunto de protocolos, ou identificar os protocolos que podem ser usado para invocar uma interface específica. Deste modo, a visão orientada a serviços enfatiza a necessidade para mecanismos de padronização de interfaces, de transparência local e remota, de adaptações para serviços locais de sistemas operacionais e de semântica uniforme de serviços.

Uma visão orientada a serviços também simplifica a virtualização, entendida também como o encapsulamento de diversas implementações atrás de uma interface comum. Virtualização permite o acesso consistente de recursos entre plataformas múltiplas e heterogêneas, com transparência local e remota. Também possibilita o mapeamento de diversas instâncias de recursos lógicos para o mesmo recurso físico e o gerenciamento deste dentro de uma organização virtual baseada em composições de recursos que vêm desde o nível mais baixo de recursos.

Na figura 6 (seis) a seguir apresentam-se vários estágios de vida de um *data mining*, uma mineração de dados, decomposto inicialmente em atividades como a invocação remota de serviço básica, o gerenciamento de ciclo de vida e as funções de notificação.

Estes serviços atômicos acima enunciados vão compor um quadro mais amplo de computação em nuvem. Este caso de mineração de dados se traduz na fisiologia da nuvem computacional num exemplo de entrega combinada de serviços. Aqui se destaca o fornecimento de

(a) Storage-as-a-Service, ou armazenamento como serviço, com o uso de fábrica de alocação de serviço;

(b) Database-as-a-Service, ou banco de dados como serviço, com o uso da estrutura de database distribuída e remota como se fosse local, e

(c) Software-as-a-Service, softwares ou aplicações entregues pela web, através de um browser.

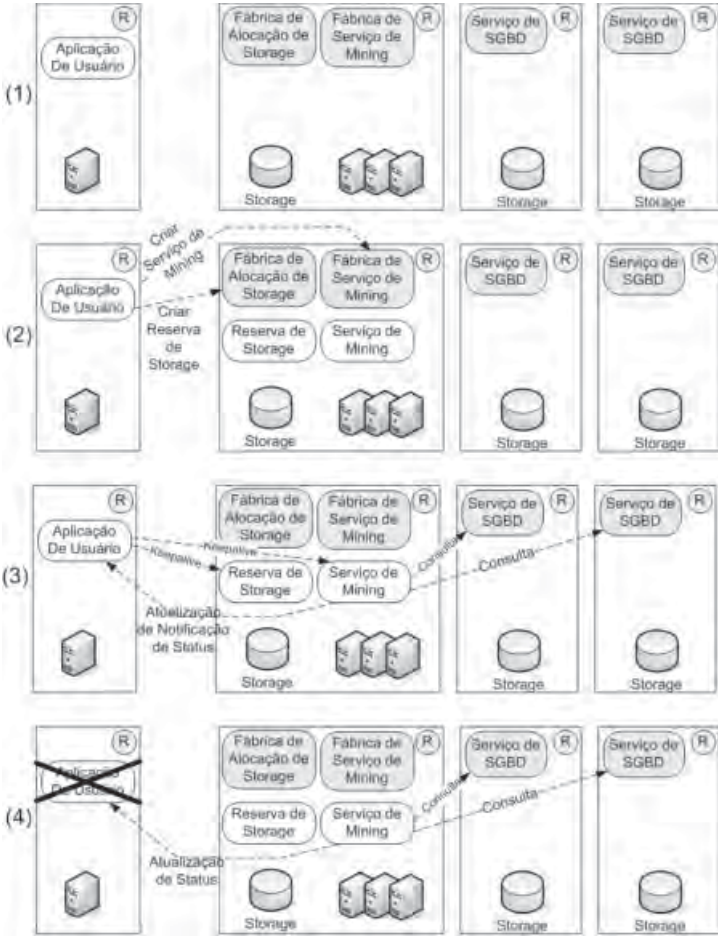


Figura 6. Exemplo para serviços básicos de cloud

Explicação da situação expressa na figura 6 (seis).

(1). A aplicação compreende inicialmente 04 (quatro) ambientes do tipo host simples: (da esquerda para a direita, numa mesma linha) ambiente que executa as aplicações do usuário; ambiente que encapsula os recursos de computação e de armazenamento (e que suporta dois serviços do tipo "fábrica", um para alocar reservas no *storage* e outro para criar serviços de mineração), um terceiro e quarto ambientes que encapsulam serviços de banco de dados, ou de SGBD (Sistema Gerenciador de bancos de Dados). Os "R"s representam serviços locais de registro e um serviço adicional de registro em uma organização virtual provê informação acerca da localização de todos os serviços representados.

(2). A aplicação de usuário emite requisições do tipo "criar serviço *cloud*" para as duas fábricas no segundo ambiente host, requisitando a criação de : (a) um "serviço *data mining*" que irá executar a operação de *data mining* de sua atribuição e (b) a alocação de espaço de armazenamento em *storage* requerido para aquele processamento. Cada requisição envolve autenticação mútua do usuário e da fábrica relevante (usando um mecanismo de autenticação especificado na descrição do serviço de fábrica), seguido por uma autorização da requisição. Como cada pedido é bem sucedido, isto resulta na criação de uma instância de serviço de *cloud* com algum tempo de vida inicial. A nova instância de serviço de *data mining* é também provida pelas credenciais de Proxy delegadas que permitem ela executar futuras operações remotas em atendimento ao usuário.

(3). O recém-criado serviço de *data mining* utiliza suas credenciais de Proxy para iniciar a requisição de dados a partir dos dois serviços de bancos de dados (SGBDs), armazenando seus resultados intermediários no *storage* local. O serviço de *data mining* também utiliza mecanismos de notificação para fornecer à aplicação de usuário com atualizações periódicas de seu *status*. Neste meio tempo, a aplicação de usuário gera requisições periódicas do tipo *keepalive*, (i.e., mantenha ativo) para as duas instâncias de serviço de *cloud* criadas.

(4) A aplicação de usuário falha por algum motivo. O processamento computacional do *data mining* continua por enquanto, mas, como nenhum

outro recurso tem interesse em seus resultados, nenhuma mensagem adicional de *keepalive* é gerada.

Devido então à falha da aplicação ilustrada, as mensagens de *keepalive* param de ser emitidas e portanto as duas instâncias de *cloud* eventualmente entram em time out e são finalizadas, liberando os recursos computacionais e de armazenamento em *strorage* que estavam consumindo.

5. Arquitetura de *cloud*

É entendimento de amplo aceite que uma arquitetura de nuvem é responsável por:

(a) identificar os componentes fundamentais dos sistemas;
(b) especificar o propósito e a função destes componentes e
(c) indicar as várias interações destes componentes entre si.

Esta arquitetura é ponto de partida para que se trabalhar o ciclo de vida da tecnologia de nuvens computacionais. Juntas, tecnologia e arquitetura podem representar o que comumente se denomina *middleware* de nuvem, visto sob a ótica dos serviços necessários para suportar um conjunto de aplicações num ambiente de rede distribuído [35].

A natureza da arquitetura de *cloud* pode ser definida como orientada a protocolos, os quais governam a interação entre componentes, e não suas implementações. Do mesmo modo que o advento da web revolucionou a forma de se compartilhar informações através de uma sintaxe e protocolos universais (HTTP e HTML), nuvens também necessitam de padrões de sintaxe e protocolos para poderem funcionar em consonância com os preceitos de orientação a serviços, promovendo compartilhamentos entre organizações virtuais, além de se observar os rumos que estão sendo consolidados sob a égide da *Green IT*, ou TI verde.

Num ambiente em rede, a adoção de protocolos comuns possibilita a interoperabilidade, a qual representa o principal aspecto a ser abordado. Interoperar torna-se essencial para a constituição das OVs (organizações virtuais) em *cloud computing*, dada a natureza fluida e dinâmica das relações entre partes quaisquer, comportando participantes novos dinamicamente, suportando diversas plataformas, linguagens e ambientes de desenvolvimento.

Complementando o aspecto dos protocolos, destacam-se também as *Application Programming Interfaces* (APIs) e os *Software Development Kits* (SDKs). Outra orientação importante para a arquitetura de *cloud* consiste nos serviços. Deste modo, estabelece-se aqui que um serviço se encontra definido em função do protocolo que ele usa para se comunicar e do comportamento que ele implementa.

5.1. Descrição

Procedendo à descrição arquitetural, observa-se que ela não enumera exaustivamente protocolos, serviços, APIs e SDKs. A descrição permite que se identifiquem os requisitos para uma classe mais genérica de componentes, organizados em camadas. Para especificarem-se as camadas, seguem-se os princípios do modelo "relógio de sol", [33], que traduz muitos comportamentos de alto nível (topo do relógio de sol) mapeados a muitas tecnologias (base do relógio) por um pequeno número de protocolos (meio ou garaganta do relógio). A figura abaixo expressa as camadas de arquitetura de *cloud* e seu mapeamento com o modelo de camadas TCP/IP.

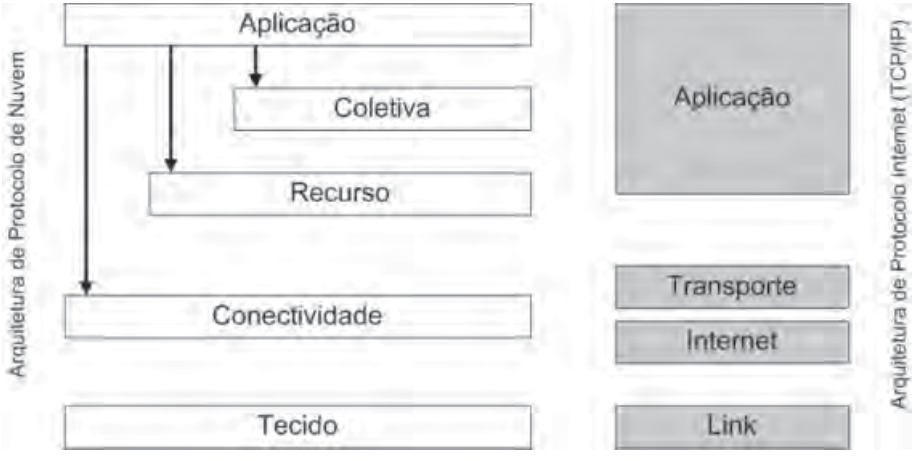


Figura 7. Arquitetura de *cloud* e seu relacionamento com a arquitetura TCP/IP

Para as camadas da arquitetura de *cloud*, observa-se o seguinte:

Tecido – camada *fabric*, o meio, que provê recursos para a mediação do acesso compartilhado pelos protocolos do *cloud*. Os componentes da camada tecido implementam as operações locais específicas de recurso que resultam de operações em níveis mais altos da pilha de camadas. No mínimo, os recursos devem implementar mecanismos de pesquisa que permitem a descoberta de sua estrutura, estado e capacidades (tais como recursos computacionais, recursos de armazenamento, recursos de rede, repositórios de código e catálogos, e.g.) e mecanismos de gerenciamento de recursos que provêem controle da qualidade de serviço.

Conectividade – que define os protocolos de comunicação e autenticação fundamentais para transações de rede específicas de *cloud*. Requisitos de comunicação são basicamente transporte, roteamento e resolução de nomes e são supridos pelos protocolos da pilha TCP/IP. Soluções de autenticação para OVs devem ter características como *single sign on*, delegação, integração com varias soluções de segurança local e relacionamentos de confiança baseados em usuários.

Recurso – esta camada é construída sobre os protocolos de comunicação e autenticação para definir protocolos, APIs e SDKs para negociação segura, monitoração, controle, contabilidade e pagamento de operações de compartilhamento sobre recursos individualizados. Duas classes de protocolos de recurso se destacam: informação e gerenciamento.

Coletiva – enquanto a camada de recurso tem enfoque em interações relacionadas a um único recurso, a camada coletiva contém protocolos, serviços, APIs e SDKs de natureza global e que apreendem interações entre coleções de recursos. Uma vez que os componentes desta camada são construídos sobre a estreita garganta do modelo do relógio de sol, i.e., os poucos protocolos das camadas de conectividade e de recurso, eles podem implementar grande quantidade de comportamentos sem adicionar novos requisitos aos recursos compartilhados. Como exemplos, pode se citar: serviços de diretório, co-alocação, agendamento, colaborativos, *brokering*, contábeis, monitoração, diagnóstico, pagamentos e pesquisa; sistemas de programação para *cloud* e gerenciamento de carga de trabalho, e, frameworks de colaboração.

Aplicação – camada que compreende as aplicações de usuário que executam num ambiente de organizações virtuais. Estas aplicações em termos

de serviços de quaisquer camadas, assim como também podem chamar serviços definidos em quaisquer camadas.

5.2. Requisitos

Torna-se mister ressaltar que os sistemas em nuvem têm por objetivo integrar, virtualizar e gerenciar recursos dentro de organizações virtuais distribuídas, heterogêneas e dinâmicas. Uma chave para o sucesso é a padronização, pois ela permite que se criem componentes interoperáveis, portáveis e reutilizáveis, possibilitando que eles sejam pesquisados, acessados, alocados, monitorados, contabilizados, cobrados e principalmente que sejam gerenciados como um sistema virtual único (mesmo quando oriundos de fornecedores distintos e operados por diversos outros *players*).

Este modelo de referência se desenvolve a partir de uma arquitetura de padrões abertos e orientada a serviços, que materializa o anseio por padronização ao definir um conjunto básico de serviços, relacionado a questões fundamentais de *cloud*, composto de comportamentos e por capacidades.

Abaixo relaciona-se o conjunto de requisitos funcionais e não-funcionais, que se originam de casos de uso explicitados em uma arquitetura de padrões abertos e orientada a serviços.

(1) Interoperabilidade e suporte a ambientes heterogêneos e dinâmicos

Ambientes em nuvem tendem a ser heterogêneos e distribuídos, englobando variedade de ambientes *host*, sistemas operacionais, dispositivos e serviços, de diversos fabricantes. Também tendem a ter longa duração e a serem dinâmicos, muitas vezes evoluindo de maneira não prevista inicialmente. Ter de suportar a diversidade se traduz em requisitos que incluem virtualização de recursos, capacidades comuns de gerenciamento, pesquisa, consulta e descoberta de recursos e esquemas e protocolos padrão.

(2) Compartilhamento de recursos entre organizações

Suportar uso e compartilhamento entre domínios administrativos sejam eles entre instituições ou dentro da mesma corporação. Torna-se necessário que sejam estabelecidos mecanismos capazes de prover contextos que associem usuários, requisições, recursos, políticas e acordos

interorganizacionais, como, por exemplo, definição de um *namespace* global, autonomia de sites, serviços de metadados e dados de uso de recursos *(resource usage data*, i.e.).

(3) Otimização

Consiste na eficiência e eficácia na alocação de recursos a fim de suprir as necessidades de consumidores e de produtores. O uso de recursos por políticas de alocação flexível, como reserva antecipada de recursos com período definido e o *pooling* de recursos de armazenamento. É necessário ter-se também uma otimização de demanda capaz de gerenciar vários *workloads*, incluindo-se demandas de *workloads* agregados. Também é essencial ter-se mecanismos de registro de uso de recursos, como *metering*, *monitoring* e *logging*. A questão da otimização é passo certo nos caminhos da sustentabilidade e da preservação de recursos originais, que compõem o paradigma da computação verde (*Green IT*), modalidade da ciência da computação alinhada à persecução dos preceitos enunciados pela economia verde (*Green Economy*).

(4) Assegurar qualidade de serviço (QoS Assurance) e qualidade de experiência (QoE)

Dimensões chave de QoS e QoE são segurança, disponibilidade e desempenho. São assim apontados requisitos como acordos de nível de serviços (ANS - ou SLA, parte de *Service Level Agreements*), acordos de nível de consecução (ANC – SLA, parte de *Service Level Attainment*) e migração.

(5) Execução de tarefas (*Jobs*)

Governança para a execução de trabalhos (ou *jobs*) definidos pelo usuário, ao longo dos seus ciclos de vida. Tem-se requisitos como suporte a vários tipos de trabalhos (*jobs*), gerenciamento de tarefas (*jobs*), agendamento e provisionamento de recursos.

(6) Serviços de dados

Acessos eficientes a grandes quantidades de dados e a compartilhamentos são cada vez mais demandados pelas empresas. Delineiam-se requisitos tais como o acesso eficiente e rápido, a consistência, a persistência, a integração e a gestão da alocação, todos relativos a diversos tipos de dados e informações.

(7) Segurança

Como já foi abordado anteriormente, destaca-se a necessidade de mecanismos de autenticação, de autorização, de interoperação por múltiplas infraestruturas de segurança e, por fim, de soluções para perímetros de segurança.

(8) Redução de custo administrativo

A complexidade inerente à administração de sistemas heterogêneos, de larga escala de distribuição, aumenta os custos de gerenciamento e o risco de falha humana. De primo, faz-se necessário um gerenciamento consistente, baseado em políticas, engendrado para automatizar o controle da nuvem computacional. Em segundo, um gerenciamento de conteúdo das aplicações, contendo dispositivos que facilitem o *deployment*, a configuração e a manutenção de sistemas complexos. Por fim, mecanismos de determinação de problemas, para que os gestores reconheçam e tratem situações de problema que surjam. Neste contexto surgem mecanismos, técnicas, algoritmos e componentes oriundos da computação autonômica, constituindo-se numa verdadeira revitalização da inteligência artificial.

(9) Escalabilidade

Um dos valores agregados adicionados por *clouds* é a redução maciça do tempo de *turn around* de trabalhos (*jobs*), permitindo o uso transparente e alternado ora de pequena quantidade de recursos, ora de grande quantidade de recursos, possibilitando a continuidade do serviço dentro das especificações formalizadas no acordo de nível de serviço e até a composição de novos serviços. Contudo, a grande escala do sistema apresenta novas necessidades, como uma arquitetura de gerenciamento que escale para milhares de recursos potenciais e de natureza diversa. Também se faz mister mecanismos de alto *throughput* para ajustar e otimizar a execução de trabalhos em paralelo.

(10) Disponibilidade e Resiliência

A alta disponibilidade esta usualmente associada a equipamentos tolerantes a falhas de grande custo ou sistemas complexos de *cluster*. Por outro lado, a demanda crescente por serviços públicos e de utilidade pública, como no caso dos serviços de um governo eletrônico, levam os sistemas a ter de operar em altos níveis de disponibilidade. Considerando que as tecnologias de *cloud*

possibilitam um acesso transparente a enormes *pools* de recursos, entre organizações e dentro delas, podemos enxergá-lo como uma importante "peça" do quebra-cabeça tecnológico, peça esta fundamental para se construir ambientes de execução estáveis e de alta confiabilidade. Entretanto, certos percalços precisam ser superados, como o fato da heterogeneidade do *cloud* permitir a existência de componentes com MTTR (*mean-time-to-repair*) indefinido ou mesmo não confiáveis. Esta questão cria certas dificuldades, mas vem se tornando cada vez mais contornável, com a evolução do *middleware*, principalmente com os mecanismos de recuperação de desastres e os de gerenciamento de falhas.

(11) Extensibilidade e facilidade de uso

São necessários mecanismos provedores de abstrações úteis e no nível desejado. Porém, não é possível se prever de antemão os usos que consumidores e produtores farão do *cloud*. Deste modo, construíram-se componentes extensíveis e de fácil substituição, permitindo evolução arquitetural e a possibilidade dos usuários elaborarem suas próprias soluções. Até mesmo os core system componentes podem ser substituídos, customização e extensibilidade devem ser providas de modo a não se comprometer o caractere de interoperabilidade

5.3. Capacidades

Uma arquitetura de nuvem com padrões abertos e orientada a serviços facilita o uso livre e o gerenciamento de recursos distribuídos e heterogêneos. Os termos aqui empregados estão em seu sentido lato: a "arquitetura", conforme a Wikipédia,

> "*consiste dos componentes de software, suas propriedades externas, e seus relacionamentos com outros softwares. O termo também se refere à documentação da arquitetura de software do sistema*".

"Distribuído" varia em um espectro que vai de recursos contíguos geograficamente e conectados por algum tecido ou meio até recursos em multidomínios globais, conectados em modo fraco ou intermitente. "Recursos" dizem respeito a quaisquer artefatos, entidades ou conhecimentos necessários para se completar operações em sistemas.

A computação utilitária, advinda da infraestrutura que está consonante com a arquitetura ora proposta, se materializa a partir do conjunto de capacidades que doravante se esboça. A seguir, na figura 8 (oito), é possível se observar a representação lógica, abstrata, das capacidades e a aglutinação não rígida (sem a obrigatoriedade predefinida de se basear nas camadas imediatas para prestar e fornecer serviços, i.e.) destas em 3 (três) camadas:

(a) **camada de base**, de recursos base, que são os recursos físicos e lógicos de um provedor de nuvem (*cloud provider*), recursos estes de relevância externa a uma arquitetura de nuvem com padrões abertos e orientada a serviços;

(b) **camada do meio**, representando nível mais alto de virtualização e abstração, composta pelos **serviços básicos** em um provedor de nuvem e que são relevantes para uma arquitetura de *cloud* aberta e orientada a serviços;

(c) **camada do topo** que representa as aplicações e outras entidades de um provedor de serviços que usam a arquitetura e suas capacidades para executarem suas funções e processos.

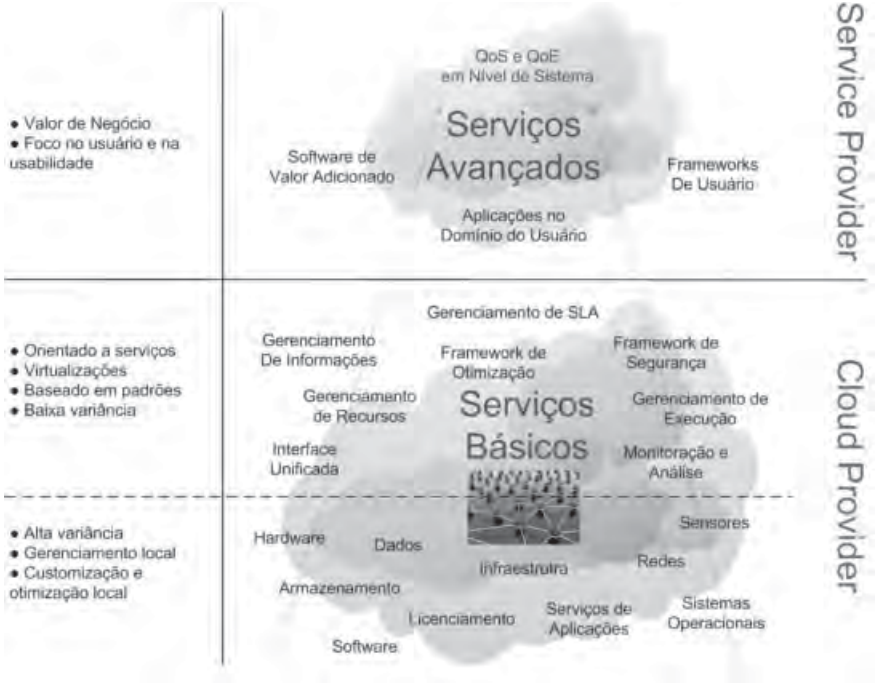


Figura 8. Visão Conceitual da Infraestrutura de *cloud*

Parte-se a seguir para uma caracterização das capacidades. Uma explanação pormenorizada pode ser depreendida da análise dos trabalhos recomendados na bibliografia, em particular os itens [59] e [60].

5.3.1. Framework de Serviços

Esquematicamente, estão elencados na figura seguinte (Figura 9, i.e., nove) os serviços da arquitetura de *cloud*. Não é uma lista exaustiva, nem números clausos, não esgota possibilidades. A figura apresentada em seguida, de número 10 (dez), mostra uma perspectiva diferente da figura 9 (nove), focando em numerosas relações e interações possíveis entre os serviços.

Observe-se que "serviços" são componentes com fraco acoplamento que, por eles próprios ou como parte de um grupo interativo, cumprem as capacidades de uma arquitetura de *cloud*, constituída sob padrões abertos e orientada a serviços, através de implementação, composição ou interação. Os serviços de arquitetura de *cloud* assumem a preexistência de um ambiente físico, que pode incluir hardware de computador ou redes e até equipamentos fiscos, como telescópios e microscópios de força quântica.

Observem-se as figuras seguintes, baseadas em [59] e [60].

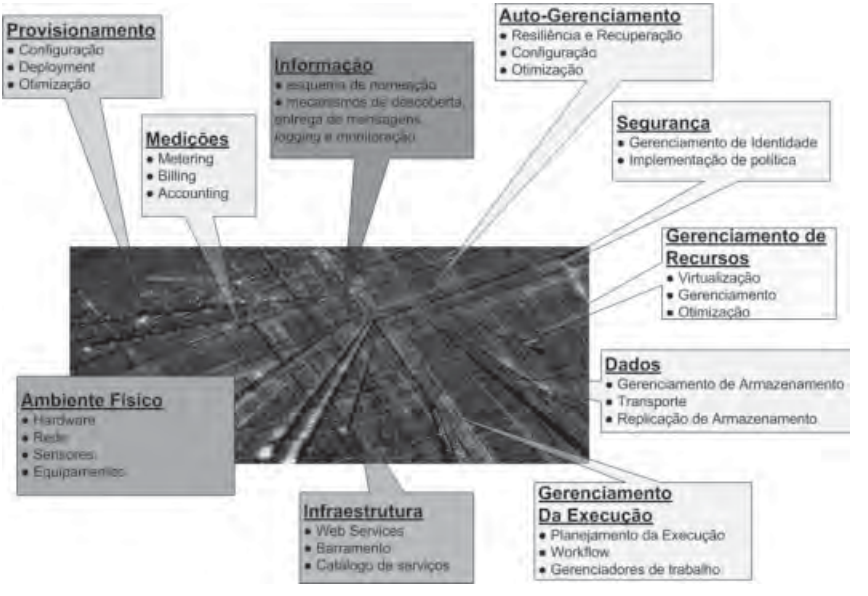


Figura 9. Framework de Arquitetura de *Cloud*

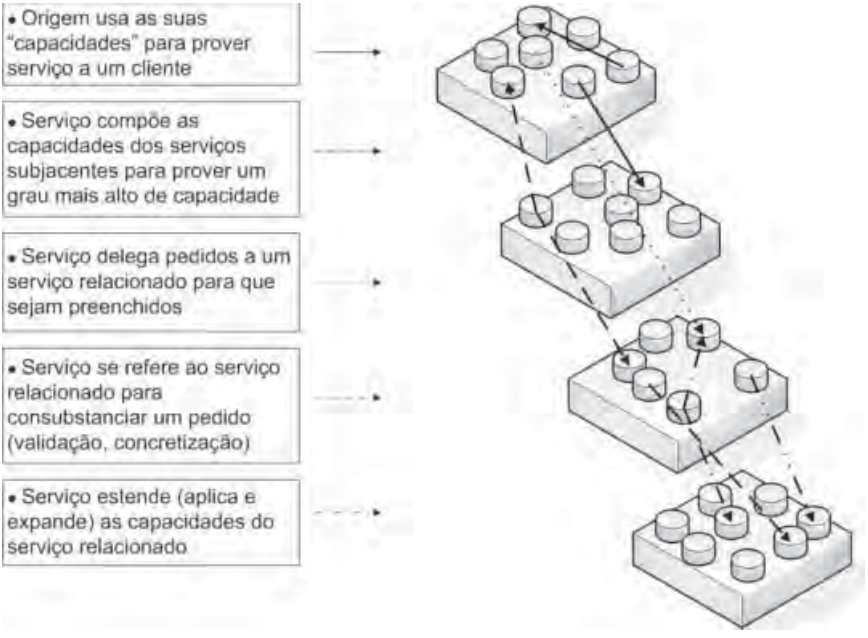


Figura 10. Relacionamentos entre Serviços

5.3.1.1. Serviços de Infraestrutura

Conforme já dissertado, a arquitetura de *cloud* por uma perspectiva subsidia, por outra, se apóia, no conjunto de especificações referente à *web service architecture* [WS-Architecture]. Esta escolha, como foi evidenciado, coloca este Modelo de Referência no caminho da orientação à serviços, valendo-se das boas práticas, utilizando padrões de amplo aceite e larga adoção no mercado.

5.3.1.2. Serviços de Gerenciamento de Execução

Têm a incumbência de lidar com problemas de instanciação e gerenciamento das unidades de trabalho (aplicações de *cloud* e aplicações de legado, ou não-*cloud*), até que estas se completem ou se resolvam (*time out*, *respawn*, e.g.). Abaixo se reproduz, baseado em [60], um metamodelo de SGEs, para uma melhor compreensão.

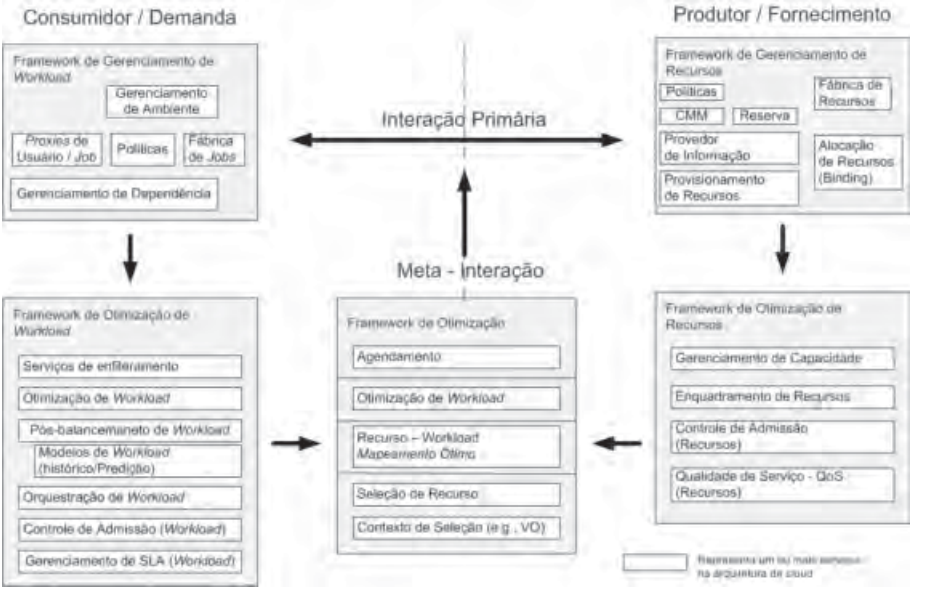


Figura 11. Metamodelo Notacional para SGEs

5.3.1.3. Serviços de Dados

Utilizados para a movimentação de dados para onde estes estão sendo necessários, para se gerenciar múltiplas cópias, para se executar operações do tipo *queries* e *updates* e para se transformar dados em novos formatos. Também estes serviços provêem as capacidades necessárias para gerenciar os metadados que descrevem os serviços de dados e outros dados. Observe-se abaixo um metamodelo de serviços de dados (Sds), baseado em [60].

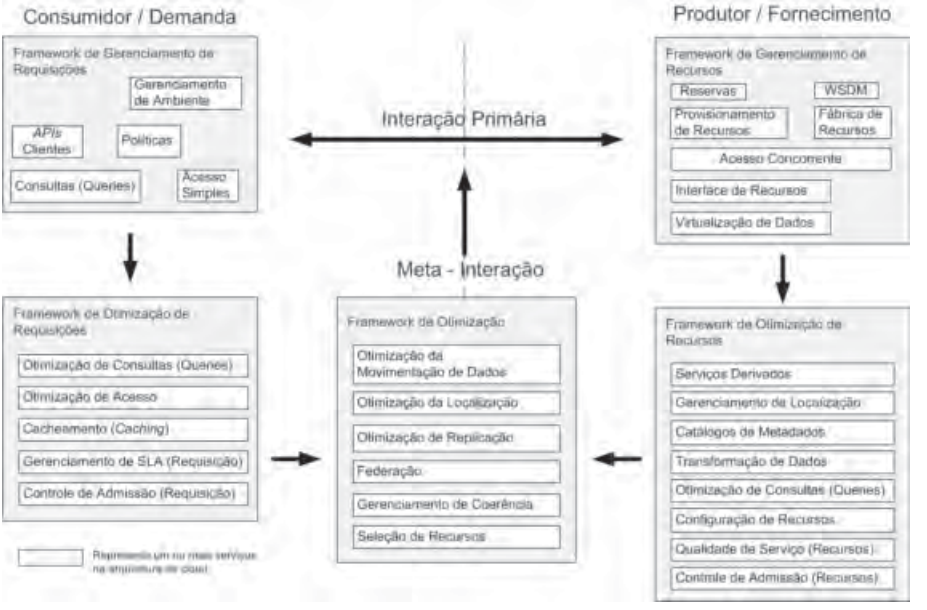


Figura 12. Metamodelo Notacional para SDs

5.3.1.4. Serviços de Gerenciamento de Recursos

Os serviços de gerenciamento de recursos, os SGRs, realizam essencialmente 3 (três) tipos de atividades:

(a) auto-gerenciamento dos recursos (e.g., *reboot* de um *host*, configuração de VLANs em um *switch*);

(b) gerenciamento de recursos no *cloud* (reserva de recursos, monitoração e controle);

(c) gerenciamento de infraestrutura de arquitetura de *cloud*, que por sua vez também é composta de recursos.

Observe-se a figura abaixo, baseada em [60].

Tipo de Gerenciamento	Nível de Interface	Interface
Gerenciamento dos Recursos	Nível de Recurso	CIM/WBEM, SNMP
	Nível de Infraestrutura	WSRF, WSDM
Gerenciamento dos Recursos na cloud	Nível das funções de arquitetura de *cloud*	Interfaces Funcionais
Gerenciamento da infraestrutura de arquitetura de cloud		Interfaces de capacidade de gerenciamento especifico

Figura 13. Níveis de gerenciamento e relacionamento com as interfaces

5.3.1.5. Serviços de Auto-Gerenciamento

Esta gama de serviços foi engendrada para se diminuir os custos e as complexidades inerentes a se possuir e se operar a infrestrutura de TI necessária para o *cloud*. Neste ambiente de auto-gerenciamento os componentes de hardware e software envolvidos têm três capacidades básicas: se recuperam, se configuram e se otimizam automaticamente, com baixa ou nenhuma intervenção humana.

Como atributos básicos, o auto-gerenciamento contem políticas, níveis de acordo de serviços (os SLAs, ou *service level agreement*)e o modelo de gerenciador de nível de serviço.

5.3.1.6. Serviços de Segurança

Estes serviços surgem com o viés de enfatizar a aplicação de políticas de segurança no contexto de uma organização virtual (OV). Vide figura abaixo, baseada em [59] e [60].

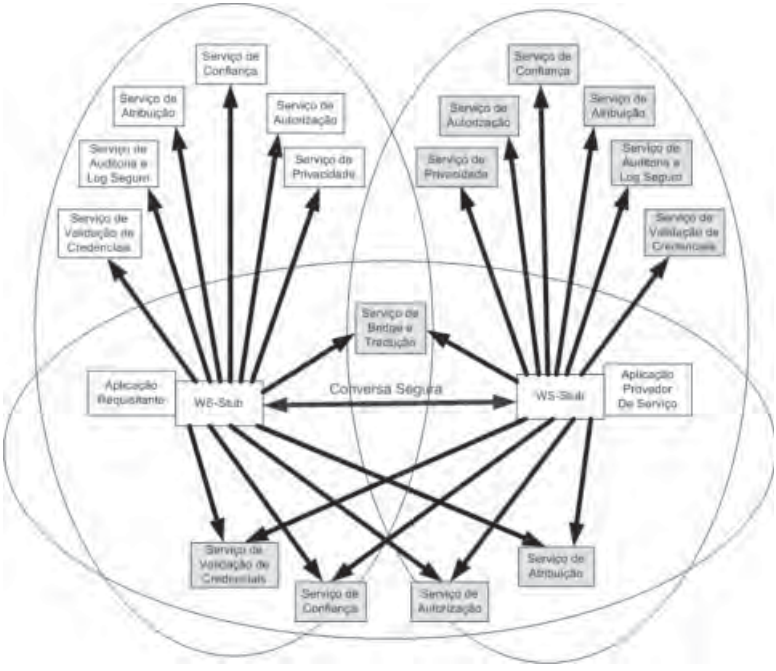


Figura 14. Serviços de segurança num ambiente de VO

De uma forma geral, enfatizar políticas de segurança é assegurar que os objetivos de negócio em nível mais abstrato ou alto sejam alcançados. Na figura acima se fornece um panorama de serviços de segurança em OVs.

5.3.1.7. Serviços de Informação

O termo "informação" se refere a dados ou eventos dinâmicos usados para a monitoração de *status*. Assim, estes serviços dizem respeito às capacidades de se manipular e acessar d modo eficiente a informação acerca de aplicações, recursos e serviços em ambiente de *cloud*.

Cinco grandes capacidades são então citadas:

(1) esquema de nomeação, mecanismos de

(2) descoberta,

(3) entrega de mensagens

(4) *logging*,

(5) monitoração e de

5.3.1.8. Serviços de Medição

Estes serviços destacam-se como uma especialização dos serviços de gerenciamento de recursos. São 3 (três) os serviços diretamente relacionados à medição: (a) *metering*, ou medição no sentido *strictu*, ou estrito, que responde pelos indicadores, as métricas e as soluções de armazenamento dos valores ; (b) *billing*, ou tarifação, que corresponde ao processo de gerar um *invoice* (contrato de compra-e-venda) para recuperar o preço de venda pela parte do cliente; e (c) *accounting* ou contabilização, que corresponde à totalizações que levam em conta as medições e tarifações expressas em acordos de nível de serviços.

A seguir enumeramos os requisitos para os serviços de medição, conforme referencia [61]. Isto nos permitem individualizar estes serviços.

1. LOCATION UNAWARENESS, ou
desconhecimento sobre a localização.
2. SERVICE ELASTICITY, elasticidade do serviço.
3. SERVICE BILLING, ou a tarifação como um serviço.
4. COMPLEX PRICING, precificação complexa.
5. ADAPTABLE DESIGN, um projeto adaptável.
6. FLEXIBLE DATA FORMAT, formato de dados flexível de modo a comportar extensões futuras
7. SERVICE ACCOUNTING, contabilização como serviço.
8. COMPENSATIONS, compensações, que junto com uso são contabilizadas com base nos acordos de nível de serviços.

5.3.1.9. Serviços de Provisionamento

Estes serviços também são aqui individuados a partir dos serviços de gerenciamento de recursos. Nesta categoria encontramos:

(a) os serviços de provisionamento em sentido *strictu* (estrito), que são essencialmente

(a.1) descoberta de serviços (*service discovery*) e

(a.2) balanceamento de carga (*load-balancing*) e

(b) os serviços de planejamento de capacidade (*capacity planning*). Este planejamento consiste em:

(b.1) determinação de objetivos;

(b.2) coleta de métricas e fixação de limites

(b.3) explicitação de tendências e prognósticos baseados nas métricas e limites e

(b.4) publicação e gerenciamento da capacidade.

5.4 *Topoi* de Arquitetura

Inicialmente observa-se que o termo **topoi** se refere ao plural de **topos**, a palavra em grego para "lugar", do latim **locus**. O termo vem aqui empregado em sua acepção adotada pela filosofia, onde especifica os chamados "lugares comuns", compilações com base em material tradicional e consolidado, em particular as descrições de ambientes padronizados.

Assim, com base na literatura, notadamente em [23] e [68], apresenta-se a seguir um modelo topográfico de referência para *cloud comp*uting – computação em nuvem.

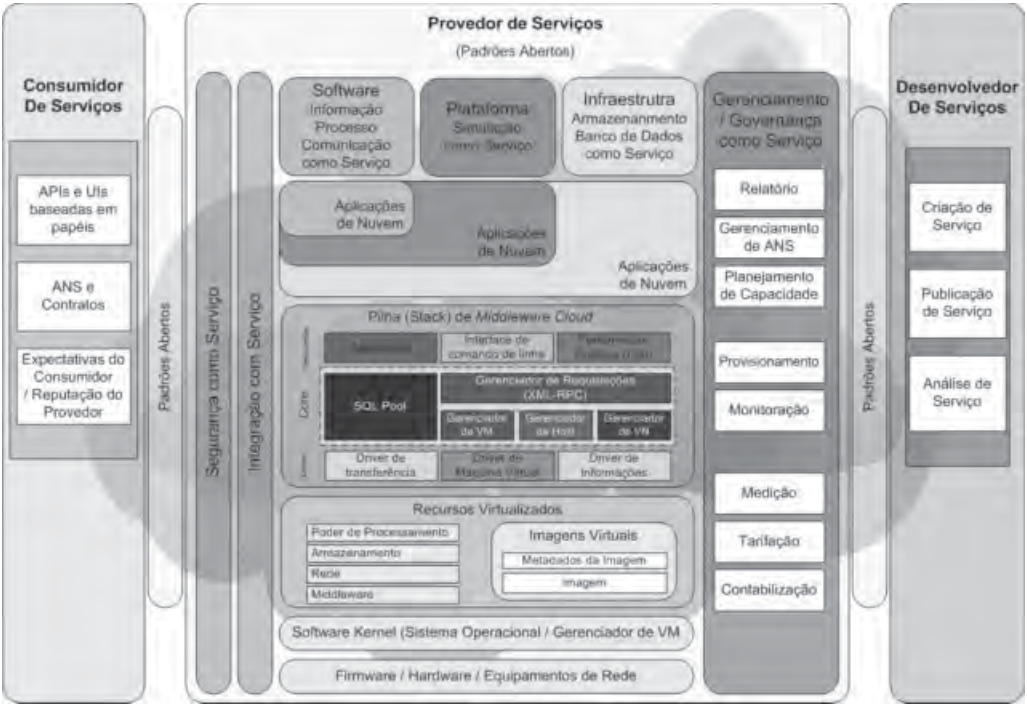


Figura 15. Modelo Topográfico de Referência para Computação em Nuvem

Abaixo, detalha-se o modelo.

5.4.1. Consumidor de Serviços.

Representa o usuário final ou a corporação que de fato vai consumir o serviço, de uma das 6 (seis) categorias apresentadas no capítulo 3 (infraestrutura, plataforma, software, integração, segurança ou governança como serviços). Três questões se destacam, a saber: APIs e UIs baseadas em papéis, ANS e contratos e as expectativas do consumidor aliadas a reputação do provedor.

APIs e UIs baseadas em papéis. De acordo com tipo do serviço e o seu papel, o consumidor tem de trabalhar com diferentes interfaces de usuário e de programação. Algumas destas interfaces, no entanto, são transparentes, tem aspecto semelhante ao de uma aplicação usual, o consumidor não necessita quaisquer conhecimentos de computação em nuvem para usar o serviço. Outras interfaces provêem funcionalidades administrativas tais como gerenciamento da nuvem ou a inicialização e a parada de máquinas virtuais. Consumidores que estão escrevendo código de aplicações usam interfaces de programação que variam conforme a própria aplicação.

ANS e Contratos. Representam os acordos negociados por meio da intervenção humana entre as partes dos atores consumidores e provedores. Representam o ingresso da computação em nuvem na esfera jurídica, considerando que estas manifestações de vontade produzem seus efeitos regularmente, observando licenciamentos e os direitos positivados e costumes dos países envolvidos. Dois tipos de contrato vêm se destacando com relação às negociações reputadas à computação em nuvem: (1) os contratos solúveis por meio de **arbitragem**, que apresentam maior flexibilidade e agilidade, considerando-se a vigente multiplicidade de esferas jurídicas independentes e a morosidade no direito internacional privado para que se consubstanciem as regras de conexão entre direitos pátrios; e (2) os contratos **joint venture**, que flexibilizam as possibilidades de soluções e as obrigações entre as partes, prevendo compensações e levando em conta regras comuns aos diversos direitos envolvidos nas negociações.

Expectativas do Consumidor / Reputação do Provedor. Representam duas questões mais importantes envolvidas nas negociações de acordo de níveis de serviços (ANS) e contratos. De acordo com diversos estudos, as expectativas essenciais das partes consumidoras são as relacionadas à segurança, confidencialidade e privacidade de dados e aplicações. Como acontece em qualquer negócio tipicamente humano, a marca do fornecedor, no caso o provedor, tem peso definitivo no processo de escolha empreendido pelo consumidor. A marca se traduz de forma mais ampla como a reputação, que envolve questões jurídicas como a regularidade legal da sociedade empresarial e a quantidade de decisões transitadas em julgado que esta sociedade tenha sido ré e condenada. Assim, a reputação é critério importante para contratos e ANS do governo brasileiro, pois os efeitos jurídicos dela vão, por exemplo, afetar o resultado de processos de licitação.

5.4.2. Provedor de Serviços

É o responsável pelo fornecimento do serviço ao consumidor. Este serviço varia conforme as 6 (seis) categorias apresentadas no capítulo 3 (três), citadas no item 5.4.1. Em relação às camadas, tendo por base a figura 15 anterior, tem-se o detalhamento abaixo, discorrendo-se a partir do nível ou camada mais elementar para a camada mais sofisticada.

***Firmware, Hardware* e Equipamentos de Rede**. Compõem a camada mais baixa do diagrama, sobre a qual toda a infraestrutura é construída. Aqui se incluem os equipamentos fiscos, os servidores, roteadores, switches, as estantes, o cabeamento, enfim, os componentes físicos alimentados pelas forças eletromagnéticas. Inclui-se também a microprogramação, os códigos e instruções básicas que são fornecidos como uma primeira camada de abstração ao meio físico, abaixo mesmo do sistema operacional.

Software Kernel (Sistema Operacional e Gerenciador de VM). Sobre a camada básica apresentada acima está o *kernel* (núcleo ou supervisor) do sistema operacional (SO), composto por sua pilha de protocolos e pelas primitivas (operações) de sistema, que especificam um serviço de sistema operacional. No mesmo nível do *kernel* de SO pode ser posicionado o Gerenciador de VM (*virtual machines*, ou máquinas virtuais) constituindo-se em uma camada de software que executa diretamente sobre a camada de

Firmware, *Hardware* e Equipamentos de Rede, a camada básica do item anterior. No que tange algumas primitivas de sistema, o Gerenciador de VM, também conhecido como *hypervisor*, substitui o *kernel* do sistema operacional. Com o *hypervisor* é possível se executar sobre o hardware múltiplos SOs *guest*, ou hóspede, em modo concorrente.De forma típica, esta camada de software kernel irá conter o *hypervisor* e um *kernel* SO com acesso direto à camada anterior e com responsabilidades próprias executando em situação coordenada ao *hypervisor*. Ambos são responsáveis por hospedar a infraestrutura suportada pela nuvem.

Recursos Virtualizados. Estes recursos e imagens compreendem as primitivas de serviço de *cloud computing*, tais como poder de processamento, armazenamento, *networking* e *middleware*. Nesta camada estão as imagens virtuais, correspondendo a sistemas operacionais *guest* para, e.g., servidores, *switches* e roteadores. Estes SO estão na camada acima da representada pelo ***Software Kernel***, a camada do item anterior. Nos recursos virtualizados também encontram-se os chamados *metadados* da virtualização, que são as informações responsáveis pelo gerenciamento das máquinas virtuais.

Pilha (*Stack*) de *Middleware Cloud*. Acima das camadas de sistema operacional (*Software Kernel* e Recursos Virtualizados), sobre o kernel de SO, *hypervisor* as imagens virtuais e metadados pode se encontrar a pilha de *software* de nuvem. Esta camada habitualmente contém componentes de software para as modalidades de computação paralela e distribuída, muitos desses constituindo soluções de *clusters* e de *grids*. Alguns elementos se destacam:

(a) Drivers, que são módulos plugáveis ou adaptadores responsáveis por interagir com *middlewares* específicos para mecanismos e serviços de informação relacionados a transferências, máquinas virtuais e a informações.

(b) Core ou Núcleo de gerenciamento, contendo gerenciadores de requisições de clientes, de máquinas virtuais, de transferências, de redes virtuais de recursos físicos hosts e de bancos de dados.

(c) Tools ou Ferramentas, como o:

(c.1) *Scheduler* ou Agendador, que trabalha requisições remotas de procedimento consolidando em código e algoritmos a definição de políticas de alocação e gerenciamento de recursos e políticas *load*

aware, i.e., de presença de carga, que subsidiam mecanismos de balanceamento, resiliência, escalabilidade e reconfiguração;
(c.2) Interface de comando de linha e
(c.3) Interfaces gráficas.
(c.4) MapReduce, framework de software que suporta computação distribuída em grandes conjuntos de dados alocados em clusters de computadores,
(c.5) Bibliotecas de MPI, Message Passing Interface, para suportar distribuição e paralelismo em sistemas operacionais e oferecer APIs que subsidiem SDKs, Toolkits e Frameworks e
(c.6). *Distributed File System*, ou sistema de arquivos distribuído.

Categorias de Tecnologia de Cloud. Esta é a camada de mais alto nível, apresentando software que permeia as demais camadas. Conforme o diagrama topográfico da figura 15, quanto a responsabilidades e abrangência, observa-se que:

1. Para **Segurança como um Serviço**, o provedor se responsabiliza pelas aplicações de segurança, como a gestão de identidade, e pela privacidade e confidencialidade de dados e informações. Observe-se que a segurança é transversal a todas as camadas ou níveis do diagrama, o que significa sua presença em todos os *topoi* (lugares) do provedor de serviço. Pode-se afirmar que a segurança permeia o modelo topográfico em questão.

2. Para a **Integração como um Serviço**, provedor e consumidor assumem suas responsabilidades conforme contrato ou acordo de níveis de serviço. O provedor usualmente trabalha a mediação entre as corporações e as aplicações, garantindo a segurança, e o consumidor tem a incumbência de realizar a integração interna as suas aplicações. A integração, a exemplo da segurança, também é transversal ao modelo da figura 15, perpassando todas as camadas.

3. Em relação a ***Software* como um Serviço**, o modelo clássico prevê que o provedor tem as responsabilidades de instalar, gerenciar e manter o *software*. O provedor não necessariamente detém a propriedade da infraestrutura física sobre a qual os programas executam. Tipicamente o consumidor não possui acesso à esta infraestrutura, nem a configurações de software básico, ele apenas se interessa pelo acesso à aplicações.

4. Para a **Plataforma como um Serviço** o provedor gere a infraestrutura de nuvem para a plataforma, que usualmente consiste em um framework para uma dada espécie em particular de aplicações. A aplicação do consumidor não acessa diretamente quaisquer elementos da infraestrutura que repousa sobre a plataforma, esta é a regra clássica.

5. Na **Infraestrutura como um Serviço**, o provedor tem a responsabilidade de manter o armazenamento, o banco de dados, a fila de mensagens e outros *middlewares* presentes no ambiente que hospeda as máquinas virtuais. O consumidor usa o serviço concebendo-o como uma unidade de armazenamento (um *disk drive*, i.e.), um banco de dados, uma fila de mensagens ou mesmo uma máquina. Na forma típica, isto se concretiza sem que este consumidor possa acessar a infraestrutura que hospeda estes serviços acima apontados.

6. Em relação às ofertas de **Gerenciamento e Governança como um Serviço**, observa-se que elas são fundamentais para as operações do provedor e do consumidor de serviços. Os modelos de negócio mais comumente verificados em prática prevêem como atividades características do provedor relacionadas ao gerenciamento de nível básico, as seguintes: (a) medição, determinando quem usa os serviços e qual a extensão deste uso, (b) aprovisionamento, para se determinar a alocação de recursos aos consumidores e (c) monitoração, a fim de se rastrear os estados dos sistemas e da nuvem e os seus recursos. Para atividades características de provedor relacionadas a um nível avançado tem-se: (a) tarifação e contabilização, de modo a se recuperar custos com base nas atividades de medição, (b) planejamento de capacidade, para garantir que as demandas de consumo serão preenchidas, (c) gerenciamento de acordo de nível de serviço, de modo a se garantir a aderência aos termos de serviço contratados entre provedor e consumidor e (d) Relatórios, reporte de informações para os diversos interessados.

6. Conclusão

Este trabalho se construiu sobre a perspectiva de se explicitar os principais elementos da computação em nuvem e estabelecer os seus inter-relacionamentos. Em fevereiro de 2010, *Cloud Computing* rapidamente está

se movendo para fora da fase do ciclo de vida tecnológico conhecida como *hype*, momento de euforia e ilusões. Bem posicionada no cenário das tecnologias inovadoras, está em ascendente adoção pelo mundo corporativo, nos mercados em expansão.

À medida que vem sendo trabalhada teórica e experimentalmente por iniciativas públicas, privadas, individuais e coletivas, empreendidas em todos os cantos do planeta, a nuvem ganha consistência. Para algumas corporações norte-americanas do segmento de TI, *cloud computing* já é uma realidade, tem tamanha densidade que é capaz de ameaçar a sobrevivência de médio prazo para os segmentos de mercado nos EUA que ainda não se posicionaram em relação aos novos modelos de negócio que com as nuvens se precipitam.

Os *early adopters*, os pioneiros, como a Amazon, a Google e o Governo Americano, todos se apresentam com invejável saúde financeira. Há prognósticos da possibilidade de uma estarem situados numa "bolha de crescimento", sem sustentabilidade, mas a verdade é que a situação destes atores está se consolidando com o tempo.

Deixando de lado concepções maniqueístas, podem ser destacadas reais vantagens de se iniciar na adoção de boas práticas preconizadas pelos estudiosos de *cloud*. Uma que vem de imediato se refere à diminuição do custo total de propriedade (TCO – *Total Cost of Ownership*) e o conseqüente aumento do retorno sobre os investimentos (ROI, *Return On Investiments*). Para que se alcance isto a proposta que vem com o emprego de soluções de computação em nuvem é a de se aumentar o uso efetivo dos equipamentos que compõem o ecossistema de um centro de dados. Estatísticas conhecidas afirmam que a média de ocupação deste meio físico no modelo tradicional é de 15% (quinze por cento); no modelo de nuvem apontam-se taxas de uso que gravitam em torno de 65% (sessenta e cinco por cento).

As implicações do cenário desenhado no parágrafo acima são na construção de um ciclo virtuoso: com o melhor emprego dos recursos, diminui-se o gasto com equipamentos, aumenta-se a eficiência e a eficácia operacionais, permitindo que se alavanque o atendimento às demandas reprimidas e que se compartilhem os recursos de forma planejada, otimizando o retorno sobre os gastos. Este é o caminho da sustentabilidade e do consumo ótimo e racional de recursos, o que vem sendo desenhado como um cenário de "economia verde" ou ecologicamente correta. Observe-se que toda a perspectiva que foi explanada acima não é uma virtude que só se obtém pela adoção de técnicas e tecnologias de computação em nuvem, outros caminhos

existem, principalmente sob a égide dos paradigmas de SOA, isto é, de arquiteturas orientadas a serviços.

Outro grande benefício que *cloud* pode trazer se constitui na economia de tempo. Devido às exigências das transações web modernas vão se pulverizando as necessidades, aumenta-se a granularidade a atomicidade das requisições e eleva-se o número de usuários simultâneos competindo pelos recursos. Para se ter flexibilidade, para que as soluções de serviços possam escalar de forma segura e para que os inevitáveis riscos e desastres sejam mitigados é necessário se adaptar e evoluir o mais rápido possível. O paradigma da computação em nuvem vem imbuído de características de tempo real, adequadas às crescentes pressões de *time-to-market*.

Uma grande vantagem para se trabalhar na nuvem vem da exigência e da adoção em larga escala de padrões abertos por parte dos atores que estão experimentando as tecnologias de *cloud*.

Por fim, o uso intensivo de software *open source* para compor soluções é outra constatação que se pode depreender da análise deste mercado. Além disso, o trabalho colaborativo congregando entidades de diversos segmentos, como pesquisa, governo ou mercado é uma constatação que a cada dia vem assumindo presença maior nas experiências de computação em nuvem.

Para concluir este trabalho, salienta-se que *cloud computing* não é a única opção de direcionamento, mas representa uma interessante possibilidade.

Referências Bibliográficas

[1] ALICE. http://alice.dante.net/

[2] ALICE2. http://alice2.redclara.net/

[3] Amazon EC2. http://aws.amazon.com/ec2/

[4] Apache Software Foundation. http://www.apache.org

[5] Arquitetura e-ping. http://www.governoeletronico.gov.br/acoes-e-projetos/e-ping-padroes-de-interoperabilidade

[6] CEA Commissariat à l'Énergie Atomique. http://www.cea.fr/

[7] CERN. http://public.web.cern.ch/public/

[8] EELA. http://www.eu-eela.org/first-phase.php

[9] EELA2 http://www.eu-eela.eu/

[10] Enomaly. http://www.enomaly.com/

[11] Eucalyptus. http://www.eucalyptus.com/

[12] Gèant. http://www.geant.net/pages/home.aspx

[13] Gèant2. http://www.geant2.net/

[14] Globus Alliance Grid. http://www.globus.org/

[15] GOOGLE. http://groups.google.com/group/cloud-computing

[16] HP – Labs. Cloud Center. http://www.hpl.hp.com/research/cloud.html

[17] IBM. http://www.ibm.com/ibm/cloud/

[18] KVM Virtualização. http://www.linux-kvm.org/

[19] MITRE. Research non-profitable organization in USA. http://www.mitre.org/

[20] NASA Nebula. http://nebula.nasa.gov/

[21] Niftyname. http://www.niftyname.org/

[22] Mygrid. http://www.mygrid.org.uk/

[23] OpenNebula. http://www.opennebula.org/

[24] ORACLE. http://www.oracle.com/technology/tech/cloud/index.html

[25] OurGrid. http://www.ourgrid.org/

[26] Public Tech. UK. http://www.publictechnology.net/

[27] RackSpace. http://www.rackspacecloud.com/

[28] TeraGrid. https://www.teragrid.org/

[29] XEN Virtualização. http://xen.org/

[30] *Approaching the Cloud: Better Business Using Grid Solutions – Twenty-five Successful Case Studies From BeinGrid*. The BeinGrid Consortium, 2009.

[31] *Cloud Computing Use cases White Paper, Version 2.0*. Cloud Computing Use Case Discussion Group, 2009.

[32] *Interoperable Clouds - A White Paper from the Open Cloud Standards Incubator.* Distributed Management Task Force, Inc. (DMTF)., 2009.

[33] *Realizing the Information Future: The Internet and Beyond*. National Academy Press, 1994. http://www.nap.edu/readingroom/books/rtif/.

[34] *Security Guidance for Critical Areas of Focus in Cloud Computing V2.1*. Cloud Security Alliance, 2009.

[35] Aiken, R., Carey, M., Carpenter, B., Foster, I., Lynch, C., Mambretti, J., Moore, R., Strasnner, J. and Teitelbaum, B. Network Policy and Services: A Report of a Workshop on Middleware, IETF, RFC 2768, 2000. http://www.ietf.org/rfc/rfc2768.txt.

[36] Armbrust M., Fox A, Griffith R., Joseph A., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., Stoica I., Zaharia M., *Above the Clouds: A Berkeley View of Cloud computing*, Technical Report No. UCB/EECS-2009-28, University of California at Berkley, USA, Feb. 10, 2009.

[37] Bedrouni A., Mittu R., Boukhtouta A., Berger ., *Distributed Intelligent Systems - A Coordination Perspective*, Springer Science+Business Media, LLC, 2009.

[38] Bertino E., Martino L. D., Paci F., Squicciarini A. C., S*ecurity for Web Services and Service-Oriented Architectures*, Springer, 2010.

[39] Buyya R., *High Performance Cluster Computing, Architecture and Systems*, Prentice Hall, 1999.

[40] Buyya R., Pandey S., and Vecchiola C., *Cloudbus Toolkit for Market-Oriented Cloud Computing*, The University of Melbourne, Australia, 2009.

[41] Buyya R., Pandey S., and Vecchiola C., *High-Performance Cloud Computing: A View of Scientific Applications*, The University of Melbourne, Australia, 2009.

[42] Buyya R., Yeo C. S., Venugopal S., Broberg J., and Brandic I., *Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility*, Future Generation Computer Systems, vol. 25, no. 6, Elsevier Science, 2009.

[43] Baker B., Smith L., *Parallel Programming*, MCGraw-Hill, Inc., 1996.

[44] Brown P. C., *Implementing SOA: Total Architecture in Practice*, Addison Wesley Professional, 2008.

[44] Casanave C., *MDA & SOA in the Enterprise: Case Study*, U.S. General Services Administration, GSA, 2005

[45] Catlett C. E., *TeraGrid: A Foundation for US Cyberinfrastructure*, in Network and Parallel Computing, LCNS vol. 3779, Springer, 2005.

[46] Chiu, W..: *From Cloud Computing to the New Enterprise Data Center*, IBM High Performance On Demand Solutions, IBM, 2008.

[47] Culler D. E., Singh J. P., *Parallel Computer Architecture: A hardware and Software Approach*, Morgan Kaufmann, 1999.

[48] Date C. J., *An Introduction to Database Systems*. Addison-Wesley, Reading, MA, 6 a. ed, 1995.

[49] Dimitrakos, Theo; Martrat, Josep; Wesner, Stefan. *Service Oriented Infrastructures and Cloud Service Platforms for the Enterprise: A Selection of Common Capabilities Validated in Real-life Business Trials by the BEinGRID Consortium.* Springer, 2009.

[50] Galli D. L., Distributed Operating Systems, Prentice Hall, 2000.

[51] Evangelinos C., Hill C. N., *Cloud Computing for Parallel Scientific HPC Applications: Feasibility of Running Coupled Atmosphere-Ocean Climate Models on Amazon's EC2*, Cloud Computing and Its Applications 2008 Chicago, 2008.

[52] Ferreira F., Santos N., Antunes M., Clusters de alta disponibilidade – uma abordagem Open Source, Instituto Politécnico de Leiria, 2006.

[53] Foster, I. The Grid: A New Infrastructure for 21st Century Science. *Physics Today*, 55 (2). 42-47. 2002.

[54] Foster, I. and Kesselman, C. Globus: A Toolkit-Based Grid Architecture. In Foster, I. and Kesselman, C. eds. *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 1999, 259-278.

[55] Foster, I. and Kesselman, C. (eds.). *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1999.

[56] Foster, I., Kesselman, C., Lee, C., Lindell, R., Nahrstedt, K. and Roy, A., A Distributed Resource Management Architecture that Supports Advance Reservations and Co-Allocation. In *Proc. International Workshop on Quality of Service*, (1999), 27-36.

[57] Foster, I., Kesselman, C., Nick, J.M. and Tuecke, S. Grid Services for Distributed Systems Integration. *IEEE Computer*, 35 (6). 2002.

[58] Foster, I., Kesselman, C., Tsudik, G. and Tuecke, S. A Security Architecture for Computational Grids. *In ACM Conference on Computers and Security*, 1998, 83-91.

[59] Foster, I., Kesselman, C. and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15 (3). 200-222. 2001.

[60] Foster, I., Kesselman, C., Nick, J. and Tuecke, S. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. Globus Project, 2002.

[61] Fox, G., Balsoy, O., Pallickara, S., Uyar, A., Gannon, D. and Slominski, A. Community Grids. Community Grid Computing Laboratory, Indiana University, 2002.

[62] Frey, J., Tannenbaum, T., Foster, I., Livny, M. and Tuecke, S., Condor-G: A Computation Management Agent for Multi-Institutional Grids. In *10th International Symposium on High Performance Distributed Computing*, (2001), IEEE Press, 55-66.

[63] Josuttis N. M., *SOA in Practice – The Art of Distributed System Design*, O'Reilly Media, Inc., New York, 2007.

[64] Gropp W., Lusk E., and Skjellum A., *Using MPI: Portable Parallel Programming with the Message-passing Interface*, ser. Scientific And Engineering Computation.MIT Press, 1994.

[65] Keahey K., Freeman T., *Science Clouds: Early Experiences in Cloud computing for Scientific Applications*, Cloud Computing and Its Applications, Chicago, 2008.

[66] Klems M., Nimis J., Tai S., *Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing* FZI Forschungszentrum Informatik Karlsruhe, Germany, 2008.

[67] Koomey, J.: *A Simple Model for Determining True Total Cost of Ownership for Data Centers*, Uptime Institute, 2007.

[68] Linthicum, David S., *Cloud computing and SOA convergence in your enterprise : a step-by-step guide*, Pearson Education, Inc., 2010.

[69] Liu H., Salerno J. J., Young M. J., Social Computing and Behaviorial Modeling, Springer Science+Business Media, LLC., 2009.

[70] Mather T., Kumaraswamy S., and Latif S., *Cloud Security and Privacy*, O'Reilly Media, Inc., 2009.

[71] Nurmi D., Wolski R., Grzegorczyk C., Obertelli G., Soman S., Youseff L., and Zagorodnov D., *The Eucalyptus Open-source Cloud Computing System*, Proc. 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2009), Shanghai, China, 2009.

[72] Ommeren E. van, Duivestein S. deVadoss J., Reijnen C., Gunvaldson E., *Collaboration in the Cloud - How Cross-Boundary Collaboration is Transforming Business*, Microsoft and Sogeti, 2009.

[73] Reese, G. (ed), *Cloud Application Architectures*, O'Reilly Media, Inc, 2009.

[74] Rittinghouse, J.W., Ransome, J.F., *Cloud Computing Implementation, Management, and Security*., CRC Press, 2010.

[75] Schlossnagle, T.: *Scalable Internet Architectures*, Sams Publishing, 2006.

[76] Sakawa M., Nishizaki I., *Cooperative and Non-cooperative Multi-Level Programming*, Springer LLC, 2009.

[77] Stanoevska-Slabeva, Katarina; Wozniak, Thomas; Ristol, Santi. *Grid and Cloud Computing: A Business Perspective on Technology and Applications*, Springer, 2009.

[78] Tanenbaum A. S., *Modern Operating Systems*, Prentice Hall, 1992.

[79] Tanenbaum A. S., *Redes de Computadores*, Rio de Janeiro, Editora Campus, 2003.

[80] Tanenbaum, A. S., Steen M. van, *Distributed Systems: Principles and Paradigms*, Prentice Hall, 2002.

[81] Tanenbaum A. S., Woodhull A. S., Sistemas Operacionais: Projeto e Implementação. Bookman, 1999.

[82] Thain D., Tannenbaum T., and Livny M., *Distributed computing in practice: The condor experience, Concurrency and Computation*, Practice and Experience, vol. 17, pp. 323–356, February, 2005.

[83] Vecchiola C., Chu X., Buyya R., *Aneka: a software platform for .NET-based Cloud computing*, in High Performance & Large Scale Computing, Eds., IOS Press, 2009.

[84] Westland J., *The Project Management Life Cycle*, Great Britain, Cambridge University Press, 2006.

[85] Williams D. E., *Virtualization with Xen™: Including XenEnterprise™, XenServer™, and XenExpress™*, Syngress Publishing, Inc., Elsevier, Inc., 2007.

[86] Williams R., *Computer Systems Architecture: a Networking Approach*, Pearson Education Limited, 2006.

Ginga, NCL e NCLua
Inovando os Sistemas de TV Digital

Luiz Fernando Gomes Soares
Departamento de Informática
Universidade Católica do Rio de Janeiro
Rua Marquês de São Vicente 225
Fone: (21) 3527-1530 (FAX)
CEP 22453-900 – Rio de Janeiro - RJ - BRASIL
lfgs@telemidia.puc-rio.br

Resumo

Este artigo apresenta as inovações brasileiras: a linguagem NCL, a biblioteca NCLua e a arquitetura do *middleware* Ginga, que compõem a Recomendação ITU-T para serviços IPTV e o sistema ISDB-T_B para TV terrestre, e que fazem desse *middleware* um dos mais expressivos e eficientes.

Palavras-chave: Ginga, NCL, Lua, *middleware*, ambiente declarativo, ambiente imperativo, TV digital.

1 - Introdução

Para oferecer suporte ao desenvolvimento de aplicações para TV digital (TVD) e permitir que elas possam ser projetadas e executadas independente da plataforma de hardware e software (sistema operacional) do dispositivo exibidor (o aparelho receptor), uma camada de software é posicionada entre o código das aplicações e a infra-estrutura de execução, como ilustra a Figura 1. Essa camada é denominada *middleware*, e Ginga é o nome do *middleware* para o sistema nipo-brasileiro de TV digital terrestre ISDB-TB [1 e 2] e para serviços IPTV conformes com a Recomendação ITU-T [3].

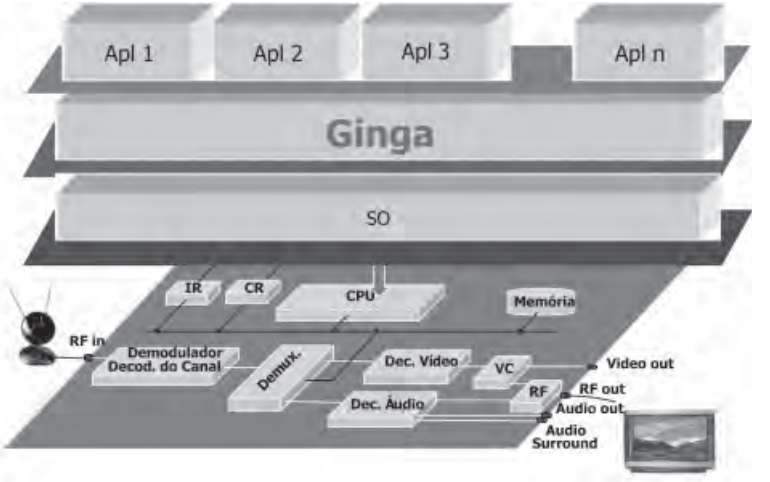


Figura 1: Middleware Ginga

O universo das aplicações de TVD pode ser particionado em um conjunto de aplicações declarativas e um conjunto de aplicações imperativas. A entidade inicial de uma aplicação, isto é, aquela que dispara a aplicação, define a que conjunto a aplicação pertence, dependendo se essa entidade é codificada segundo uma linguagem declarativa ou imperativa. Note que aplicações declarativas podem conter entidades imperativas e vice-versa, o que as caracteriza é apenas a entidade inicial.

Linguagens declarativas enfatizam a descrição declarativa de uma tarefa, ao invés de como essa tarefa deve ser realizada, ou seja, sua decomposição passo a passo, em uma definição algorítmica do fluxo de sua execução em uma máquina, como fazem as descrições imperativas. Por ser de mais alto nível de abstração, tarefas descritas de forma declarativa são mais fáceis de serem concebidas e entendidas, sem exigir um programador especialista, como é usualmente necessário nas tarefas descritas de forma imperativa. Contudo, uma linguagem declarativa usualmente visa um determinado domínio de aplicações e define um modelo específico para esse domínio. Quando uma tarefa casa com o modelo da linguagem declarativa, o paradigma declarativo é, em geral, a melhor escolha.

Linguagens imperativas são mais expressivas e são de propósito geral, porém, a um custo elevado. Como mencionado, elas usualmente exigem um programador especialista, geralmente colocam em risco a portabilidade de

uma aplicação, e o controle da aplicação é muito mais sujeito a erros cometidos pelo programador. No entanto, nos casos onde o foco de realização de uma tarefa não casa com o foco da linguagem declarativa, o paradigma imperativo é, em geral, a melhor escolha.

Por tudo o mencionado acima, os *middlewares* para TV digital usualmente provêem suporte para o desenvolvimento tanto seguindo o paradigma declarativo quanto o imperativo. O ambiente declarativo de um *middleware* dá o suporte necessário às aplicações declarativas, aquelas cuja entidade inicial da aplicação é expressa através de uma linguagem declarativa, enquanto o ambiente imperativo dá o suporte necessário às aplicações imperativas, aquelas cuja entidade inicial da aplicação é expressa através de uma linguagem imperativa. No caso do *middleware* do padrão brasileiro, os dois ambientes são exigidos nos receptores fixos e móveis, enquanto apenas o ambiente declarativo é exigido nos receptores portáteis. Na Recomendação ITU-T para serviços IPTV, também apenas o ambiente declarativo é exigido. Em todos os casos mencionados, o ambiente declarativo deve seguir a especificação Ginga-NCL, a única inovação com tecnologia inteiramente brasileira para esses sistemas de DTV.

O ambiente declarativo Ginga-NCL provê suporte para o desenvolvimento de aplicações declarativas desenvolvidas na linguagem NCL (Nested Context Language), que podem conter entidades imperativas especificadas na linguagem Lua. Principalmente por sua grande eficiência e facilidade de uso, Lua é a linguagem de script de NCL.

Este artigo tem por finalidade apresentar resumidamente o ambiente declarativo NCL e suas linguagens NCL e Lua. A Seção 2 apresenta um breve histórico das pesquisas que culminaram com o desenvolvimento dessas tecnologias. A Seção 3 introduz a arquitetura de referência do middleware Ginga. A Seção 4 se dedica ao ambiente declarativo Ginga-NCL e, finalmente, a Seção 5 é reservada às considerações finais.

2 - Um Breve Histórico

As pesquisas sobre sistemas hipermídia[1] iniciaram-se em 1990 no laboratório TeleMídia do Departamento de Informática da PUC-Rio, em um trabalho conjunto com o Centro Científico da IBM – Rio.

[1] Middlewares para sistemas de DTV é um caso particular de sistemas hipermídia.

Em 1991, o modelo NCM (Nested Context Model), que serviu posteriormente de base para o desenvolvimento da linguagem NCL, foi proposto [4]. NCM resolvia um problema em aberto na área de sistemas hipermídia e teve suas propostas incorporadas ao padrão MHEG [5] da ISO (International Organization for Standardization), em 1992. O padrão MHEG viria a se tornar nos anos seguintes a linguagem do primeiro middleware desenvolvido para TV Digital e que é até hoje adotado pelo sistema de TV digital britânico.

Em 1998, o laboratório TeleMídia foi considerado pelo IEEE um dos cinco mais importantes grupos no desenvolvimento de sistemas hipermídia. A partir desse ano, o laboratório começou uma importante interação com o grupo do CWI de Amsterdam, onde nasceu a linguagem SMIL para sincronização de objetos em aplicações multimídia. Também em 1998, foi apresentado o primeiro exibidor para aplicações seguindo o modelo NCM, denominado Formatador HyperProp. Esse sistema viria a ser o precursor do ambiente Ginga-NCL.

Em 2000, com o advento da metalinguagem XML, as estruturas de dados do modelo conceitual NCM foram especificadas como uma aplicação XML, dando origem a primeira versão da linguagem NCL (Nested Context Language).

Em 26 de novembro de 2003, o sistema brasileiro de TV digital, SBTVD, foi instituído pelo Decreto presidencial 4.901. Como foco principal, o decreto definiu para o SBTVD o requisito: "promover a inclusão social, a diversidade cultural do País e a língua pátria por meio do acesso à tecnologia digital, visando à democratização da informação". Cabe aqui ressaltar a clareza do governo brasileiro em reconhecer, desde então, que a democratização da informação não se faz apenas pelo direito de acesso a informação, mas também pelo direito de sua geração. Naquele momento nascia a necessidade de um sistema que oferecesse um modo eficiente, porém fácil e compreensível, para a produção de informação. Nascia, como conseqüência, a necessidade do desenvolvimento de uma nova solução de *middleware*. Assim, em julho de 2005, durante o Congresso da Sociedade Brasileira de Computação, o Ministério das Comunicações convoca a comunidade de pesquisadores de computação do país a participar do esforço da concepção do software para o SBTVD, incluindo seu middleware.

Ainda em 2005 foram lançados os 22 editais do FUNTTEL (Fundo para o Desenvolvimento das Tecnologias de Telecomunicações) para o

desenvolvimento do SBTVD. Entre esses editais, dois se referiam aos ambientes declarativo e imperativo do middleware para o SBTVD. Dos laboratórios contratados, TeleMídia, na PUC-Rio, e Lavid, na UFPB, surgiram, respectivamente, os sistemas MAESTRO e FlexTV. Em 2006 esses sistemas se unem em uma única arquitetura denominada Ginga[2] e passam a se chamar, respectivamente, Ginga-NCL e Ginga-J.

Em 2006 um grande passo foi dado: a implementação de referência do Ginga-NCL é colocada disponível em código aberto e gratuito. Nascia a comunidade Ginga Brasil, hoje com mais de dez mil membros colaboradores, graças ao grande apoio do portal de software público do Ministério do Planejamento. A comunidade de software livre passa então de grande aliada a também partícipe no desenvolvimento do Ginga-NCL.

Em julho de 2007, foi apresentado no Congresso da SBC a primeira implementação Ginga, juntamente com várias aplicações. A partir de então, o Ginga-NCL começa a receber várias manifestações internacionais em seu apoio, incluindo para sua adoção em outros países fora do ISDB-T_B.

No final de 2007, o Ginga-NCL foi aprovado como padrão ABNT para o sistema brasileiro de TV digital, para todos os tipos de terminais receptores.

Em 2008, o Japão propõe o Ginga-NCL como padrão mundial para serviços IPTV. Começa então um grande esforço do laboratório TeleMídia e da Anatel dentro dos grupos de padronização do ITU-T.

Em abril de 2009, Ginga-NCL e a linguagem NCL (incluindo sua linguagem de script Lua) se tornaram a Recomendação ITU-T H.761 para serviços IPTV. Pela primeira vez na história das TICs o Brasil tem um padrão mundial reconhecido.

Em setembro de 2009, a Recomendação ITU-R BT 1699-1 para harmonização de formatos declarativos para aplicações interativas para TV digital terrestre também adota o Ginga-NCL, NCL e Lua em suas especificações.

No Final de 2010, o Ginga-NCL foi adotado como o *middleware* do sistema de TV digital argentino. Na mesma época foi criada a comunidade de software livre Ginga argentina, uma das mais ativas nas contribuições para o Ginga-NCL.

[2] Por que o nome Ginga? Ginga é uma qualidade de movimento e atitude que os brasileiros possuem e que é evidente em tudo o que fazem. A forma como caminham, falam, dançam e se relacionam com tudo em suas vidas. Ginga é flexibilidade, é adaptação, qualidades inerentes ao *middleware* brasileiro.

Em maio de 2010, a implementação do Ginga-NCL do laboratório TeleMídia da PUC-Rio se torna a implementação de referência da ITU-T.

3 - Arquitetura de Referência

A arquitetura do middleware Ginga pode ser dividida em três módulos principais: Ginga Common-Core (ou Ginga-CC), Ambiente de Apresentação Ginga (ou Ginga-NCL) e ambiente de execução Ginga (ou Ginga-Imp), como mostra a Figura 2.

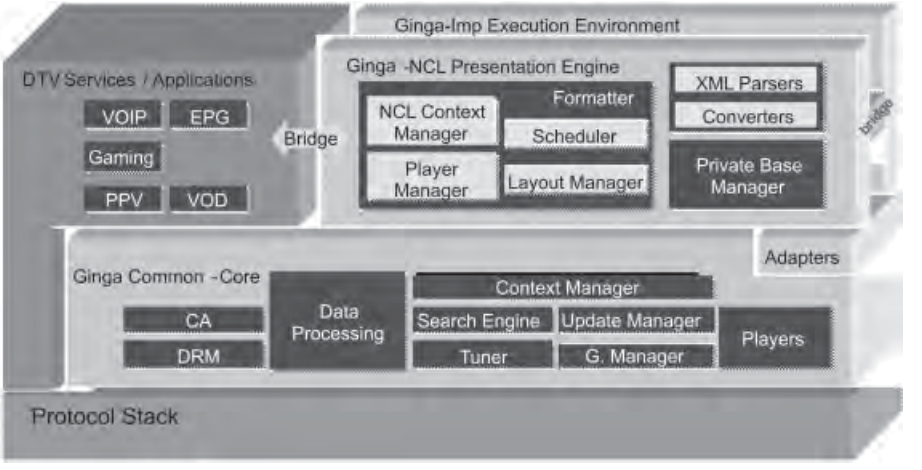


Figura 2: Arquitetura de referência do *middleware* Ginga

Ginga-Imp é o subsistema lógico do *middleware* Ginga responsável pelo processamento de aplicações imperativas. No caso do SBTVD, a especificação desse subsistema cabe à Norma ABNT NBR 15606-4 [6] e estabelece a linguagem Java como a linguagem imperativa.

Ginga-NCL é o subsistema lógico do *middleware* Ginga responsável pelo processamento de aplicações declarativas NCL. NCL e sua linguagem de script Lua compõem a base para o desenvolvimento de aplicações declarativas no SBTVD terrestre (Normas NBR 15606-2 [1] e ABNT NBR 15606-5 [2]), e para serviços IPTV (Recomendação ITU-T H.761 [3]).

Na arquitetura Ginga, apenas o ambiente Ginga-NCL é obrigatório. Ginga-Imp é opcional.

Ginga-CC é o subsistema lógico que provê todas as funcionalidades comuns ao suporte dos ambientes Ginga-NCL e Ginga-Imp. A arquitetura do sistema garante que apenas o módulo Ginga-CC deva ser adaptado à plataforma onde o Ginga será embarcado. Ginga-CC provê, assim, um nível de abstração da plataforma de hardware e sistema operacional, acessível através de APIs (*Application Program Interfaces*) bem definidas.

Um conjunto de exibidores comuns faz parte dos componentes do Ginga-CC. Eles são exibidores de áudio, vídeo, texto e imagem, etc. O acesso a tais exibidores se dá através de adaptadores, responsáveis por notificar eventos de apresentação e seleção (interação do usuário).

Entre os exibidores se encontra o exibidor (agente do usuário) HTML. Se encontra também o exibidor Lua, capaz de interpretar objetos com código Lua baseados na biblioteca NCLua definida para a linguagem de script da NCL.

Uma ponte desenvolvida entre os ambientes Ginga-NCL e Ginga-Imp provê suporte a aplicações híbridas com entidades especificadas em NCL, Lua e a linguagem suportada pelo ambiente imperativo.

4 - O Ambiente Ginga-NCL

Ginga-NCL é uma proposta totalmente brasileira para middlewares de sistemas de TV digital. O ambiente tem por base a linguagem NCL (uma aplicação XML) e sua linguagem de script Lua, ambas desenvolvidas nos laboratórios da Pontifícia Universidade Católica do Rio de Janeiro.

Os ambientes declarativos dos sistemas americano (ACAP-X), europeu (DVB-HTML) e japonês (BML-ARIB) têm por base a linguagem XHTML. XTHML carrega o legado de tecnologias anteriormente desenvolvidas para navegação textual. Ao contrário, aplicações para TVD são usualmente centradas no vídeo. Além disso, o modelo da linguagem XHTML tem o foco no suporte à interação do usuário telespectador. Outros tipos de relacionamentos, como relacionamentos de sincronização espaço-temporal e relacionamentos para definição de alternativas (adaptação de conteúdo e de apresentação), são usualmente definidos através de uma linguagem imperativa; no caso de todos os três sistemas citados a linguagem ECMAScript.

O modelo da linguagem NCL visa um domínio de aplicações mais amplo do que o oferecido pela linguagem XHTML. NCL visa não apenas o suporte declarativo à interação do usuário, mas ao sincronismo espacial e temporal em sua forma mais geral, tratando a interação do usuário como um caso particular. NCL visa também o suporte declarativo a adaptações de conteúdo e de formas de apresentação de conteúdo; o suporte declarativo a múltiplos dispositivos de exibição, interligados através de redes residenciais (HAN – Home Area Networks) ou mesmo em área mais abrangente; e a edição/produção da aplicação em tempo de exibição, ou seja, ao vivo. Como esses são os focos da maioria das aplicações para TV digital, NCL se torna a opção preferencial no desenvolvimento da maioria de tais aplicações. Para os poucos casos particulares, como por exemplo, quando a geração dinâmica de conteúdo é necessária, NCL provê o suporte de sua linguagem de script Lua.

Diferente das linguagens baseadas em XHTML, NCL define uma separação bem demarcada entre o conteúdo e a estrutura de uma aplicação. A linguagem não define nenhuma mídia per si. Ao contrário, ela define apenas a cola que relaciona os objetos de mídia (imagens, textos, vídeos, áudios, objetos com código imperativo, objetos com código declarativo, etc.) que compõem as apresentações DTV. Entre os vários objetos de mídia suportados por Ginga-NCL, o objeto de mídia baseado em XHTML é obrigatório. A NCL não substitui, mas embute documentos (ou objetos) baseados em XHTML. Como conseqüência, é possível ter navegadores definidos por outros sistemas (LIME, BML, DVB-HTML e ACAP-X) embutidos em um exibidor de documento NCL.

Outro objeto de mídia que deve ser necessariamente suportado pelo Ginga-NCL é aquele cujo conteúdo é formado por códigos imperativo Lua, baseados na biblioteca NCLua. Lua é uma das linguagens de script mais eficientes; muito mais rápida do que ECMAScript, presente na maioria dos padrões de middleware declarativo, e com um *footprint* de memória bem menor, como indica a Figura 3, obtida do site de avaliação de linguagens (http://shootout.alioth.debian.org/): Lua é, em média, 7 vezes mais rápida e com um uso de memória 40 vezes menor. Lua é hoje a linguagem mais importante na área de entretenimento.

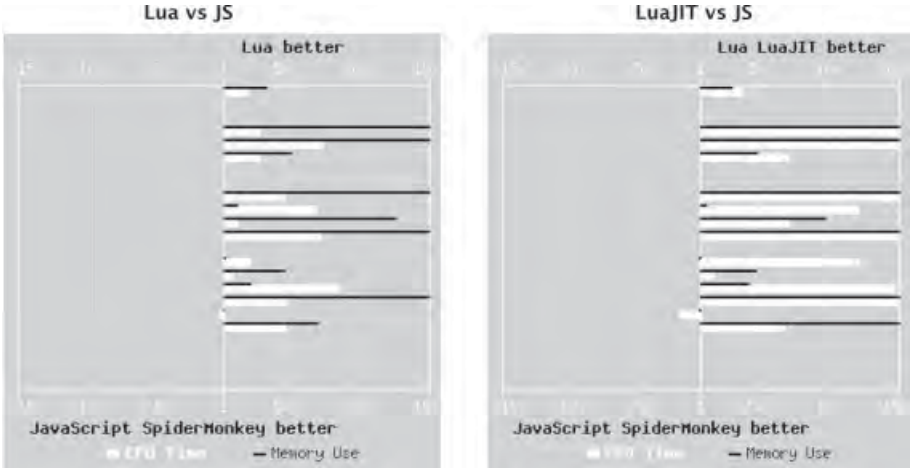


Figura 3: Comparação entre Lua e o JavaScript SpiderMonkey do navegador Mozila Firefox

Voltando a atenção para o ambiente Ginga-NCL da Figura 2, o componente Formatador NCL tem como responsabilidade orquestrar toda a execução de uma aplicação NCL, garantindo que os relacionamentos espaço-temporais definidos pelo autor da aplicação sejam respeitados. O Formatador NCL trata de aplicações recebidas pelo Ginga-CC e depositadas em uma estrutura de dados chamada "base privada".

No Ginga-NCL, uma aplicação de TVD pode ser gerada ou modificada ao vivo (em tempo real), através de comandos de edição. O conjunto de comandos de edição pode ser dividido em três grupos. O primeiro grupo de comandos é responsável pela ativação e desativação de uma base privada, ou seja, a habilitação de aplicações de um determinado canal de TV. Em uma base privada, aplicações NCL podem ser ativadas, pausadas, retomadas e desativadas, através de comandos bem definidos pertencentes ao segundo grupo de comandos. O terceiro grupo define comandos para modificações de uma aplicação ao vivo.

Finalmente, Ginga-NCL oferece suporte a múltiplos dispositivos de entrada e saída. Tal facilidade declarativa, juntamente com os comandos de edição ao vivo, únicos do sistema Ginga, provê suporte para o grande domínio de aplicações interativas de TVD que se descortina: as aplicações para as chamadas TV Social (*Community* ou *Social* TV), onde uma comunidade de

usuários cria ao vivo, sobre o conteúdo e aplicações recebidas, novas aplicações (geração de novos conteúdos e informações personalizados), que são trocadas entre seus membros.

5 - Considerações Finais

A implementação de referência do Ginga-NCL foi desenvolvida em código aberto e pode ser obtida em www.gingancl.org.br, sob de licença GPLv2. A máquina Lua também se encontra disponível pelo mesmo site, através de licença MIT. Ferramentas de autoria para o desenvolvimento de aplicações NCL, também em código aberto, estão disponível ainda no mesmo site, bem como tutoriais, livros, artigos e exemplos de várias aplicações NCL e Lua. Em http:/club.ncl.org.br, um repositório de aplicações interativas é encontrado, onde autores podem divulgar suas idéias, talentos, e, ainda, suas técnicas de desenvolvimento usando a linguagem NCL com scripts Lua.

Várias listas de discussão e contribuições no desenvolvimento de aplicações podem ser encontradas na comunidade Ginga, em www.softwarepublico.gov.br. Documentos, artigos e tutoriais a respeito do *middleware* Ginga, podem ser obtidos em www.ginga.org.br.

Referências

[1] Soares L.F.G., Rodrigues R.F. Nested Context Model 3.0 Part 1 – NCM Core. *Technical Report. Informatics Department of PUC-Rio*, MCC 18/05. Rio de Janeiro, May, 2005. ISSN 0103-9741.

[2] Televisão digital terrestre - Codificação de dados e especificações de transmissão para radiodifusão digital - Parte 2: Ginga-NCL para receptores fixos e móveis - Linguagem de aplicação XML para codificação de aplicações.

[3] Televisão digital terrestre - Codificação de dados e especificações de transmissão para radiodifusão digital - Parte 5: Ginga-NCL para receptores portáteis - Linguagem de aplicação XML para codificação de aplicações.

[4] ITU-T Recommendation H.761. *Nested Context Language (NCL) and Ginga-NCL for IPTV Services*. Geneva, April, 2009.

[5] ISO/IEC 13522-1 Information technology - Coding of multimedia and hypermedia information - Part 1: MHEG object representation — Base notation (ASN.1). 1997.

[6] Televisão digital terrestre - Codificação de dados e especificações de transmissão para radiodifusão digital - Parte 4: Ginga-J para receptores fixos e móveis. 2010.

Segurança de Computação em Nuvem

Coordenação Estratégica de Tecnologia (CETEC)[1]
Serviço Federal de Processamento de Dados (SERPRO)

Resumo

Além do desafio atual de desenvolver sistemas seguros, a computação em nuvem apresenta um nível adicional de riscos que estão diretamente associados com a terceirização dos serviços essenciais (serviços em nuvem). Esta condição exige atenção especial para os aspectos de integridade, disponibilidade, confidencialidade e privacidade dos dados, bem como suporte e conformidade. Entretanto, a computação em nuvem traz benefícios relacionados ao fato de que não se necessita desenvolver controles internos de segurança e nem capacitar pessoas neste segmento. Neste novo modelo de prestação de serviço é fundamental que seja estabelecido entre o cliente e o provedor uma relação de confiança. A transparência da forma como o provedor implementa, desenvolve e gerencia a segurança é o fator decisivo para se atingir este objetivo.

1. Introdução

A computação em nuvem se insere na proposta de um princípio organizador de grande poder computacional como serviço, porém carece de

[1] Gleyner Martins Novais, Indiana Belianka Dias Kosloski, Jose Maria Leocadio, Leandro Resende Gomes, Marcos Allemand Lopes, Maria do Carmo Soares de Mendonca, Pedro Andre de Faria Freire, Silvio Fernando Vieira Correia Filho,Tarcizio Vieira Neto.

controles efetivos de segurança para conviver nesse cenário de virtualização de rede e compartilhamento de recursos de forma autonômica. Além do desafio atual de desenvolver sistemas seguros, a computação em nuvem apresenta um nível adicional de riscos que estão diretamente associados com a terceirização dos serviços essenciais. Esta condição exige atenção especial para os aspectos de integridade, disponibilidade, confidencialidade e privacidade dos dados, bem como suporte e conformidade. Por outro lado, a computação em nuvem traz benefícios aos clientes, uma vez que não terão que desenvolver controles internos de segurança e nem capacitar pessoas neste segmento. Sempre poderão contar com especialistas e controles dedicados à proteção dos dados, tratamento de registros de logs e gestão de identidades.

Considerando que neste modelo de entrega de serviço até mesmo aplicações básicas tais como de correções (patch) e estabelecimento de filtros, podem ficar sob a responsabilidade do provedor do serviço de nuvem, é fundamental que seja estabelecido entre o cliente e o provedor uma relação de confiança. A transparência da forma como o provedor implementa, desenvolve e gerencia a segurança são fatores decisivos para atingir este objetivo. É importante destacar que em último caso o cliente ainda é o responsável pela conformidade e proteção dos seus dados críticos, mesmo que estes tenham sido transferidos para a nuvem.

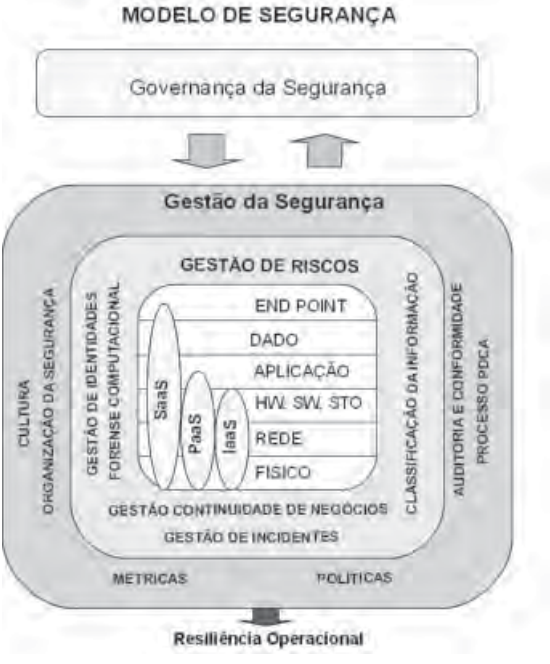
A computação em nuvem pode ser disponibilizada em diversos modelos de serviço. Cada um deles exige diferentes níveis de responsabilidades para a gestão da segurança. Modelo de serviços Software as a Service (SaaS) – grande parte da responsabilidade pela gestão da segurança está à cargo do provedor do serviço de nuvem; Modelo de serviços Plataform as a Service (PaaS) – permite aos clientes assumirem mais responsabilidades, como por exemplo a gestão da configuração e a segurança de banco de dados; e Modelo de serviços Infrastructure as a Service (IaaS) – transfere ainda mais o controle e a responsabilidade pela segurança do provedor para o cliente, como por exemplo a segurança de aplicações.

2. Modelo de segurança para computação em nuvem

O modelo proposto foi concebido visando contemplar as características típicas de um ambiente de computação em nuvem e os requisitos de segurança associados. O modelo exige uma abordagem inicial de governança da seguranca, orientando e dando apoio para as questões de gestão, dentre elas

a gestão de risco, gestão de identidades e gestão de continuidade de negócios. Como forma de implementar as determinações da Politica de Segurança são definidos segmentos e camadas, representados por processos e mecanismos de segurança.

Finalmente, o conceito de resiliência é também integrado ao modelo como forma de usufruir das próprias características do ambiente de nuvem e ao mesmo tempo garantir a continuidade dos serviços. Assim o Modelo pretende assegurar, além da proteção, serviços resilientes quando necessários.



2.1. Governança da Segurança

Governança pode ser entendida como sendo a camada estratégica do negócio a ser estruturado, administrado e continuado. Este direcionamento estratégico deve se respaldar nos conceitos clássicos da governança inseridos em novos paradigmas, onde:

- Transparência, demonstrada pela existência de efetivos e robustos controles de segurança para garantir ao cliente que suas informações serão protegidas de acessos não autorizados, mudanças ou destruições;

• Privacidade, deve ser obtida por meio da garantia de controles sobre prevenção, detecção, reação e identificação e controle de brechas/ vulnerabilidades;

• Conformidade, adoção de leis, regulamentos e padrões relativos a computação em nuvem e devida adequações ao tipo de serviço prestado, responsabilidades contratuais e leis de outros países que possam interferir;

• Gestão de riscos e responsabilidades;

• Armazenagem de informações, quando e onde a informação pode ser armazenada em nuvem, localização física e obrigações legais;

• Certificação.

Assim, a governança como camada estratégica para atendimento ao segmento do negócio Computação em Nuvem, envolve a aplicação de políticas com foco no uso dos serviços e mecanismos de controles a partir da definição de um framework com padrões de gestão da qualidade, conformidade, auditoria e ciclos de melhoria.

A definição do plano de governança pressupõe a decisão inicial sobre a forma de fornecimento de serviço se IaaS, PaaS ou SaaS e se for nuvem privada, nuvem pública ou nuvem híbrida, a estruturação de um processo de governança pode ser a partir do framework baseado em PDCA (Plan, Do, Check, Act), observando que o alinhamento estratégico, a gestão de riscos e qualidade de performance são fatores preponderantes na definição do plano, implementações de controles, monitoração, revisão e ciclo de melhorias.

2.2. Gestão de segurança

A Segurança da Informação pode ser alcançada por meio de um conjunto de controles ou medidas que abrange políticas, práticas, normas, procedimentos, estruturas organizacionais e funções de software e hardware. A gestão da segurança permite que este conjunto de controles possa ser definido, implementado, operado, mantido, monitorado, avaliado e melhorado.

A NBR ISO/IEC 27001 prevê um sistema de gestão da segurança baseado no modelo PDCA (plan-do-check-act), como segue:

Planejar (plan) – estabelecer a política, objetivos, processos, procedimentos, considerando os riscos ao negócio (computação em nuvem);

Fazer (do) – implementar e operar a política, controles, processos e procedimentos definidos;

Checar (check) – monitorar a avaliar o desempenho dos processos estabelecidos;

Agir (act) – manter e melhorar o sistema de gestão como um todo.

Um programa de segurança pode contemplar o sistema de gestão da segurança e detalhar seu funcionamento. Dentro deste contexto, alguns componentes são fundamentais para a gestão da segurança:

Política de segurança: documento direcionador sobre a segurança da informação. Do desdobramento da política surgem as normas e procedimentos;

Gestão de riscos: processo de avaliação e tratamento dos riscos de segurança visando a implantação de controles (administrativos, técnicos ou físicos);

Sistemáticas: metodologias específicas que visam apoiar os objetivos do programa de segurança. Podem incluir: gestão de continuidade de negócios, classificação dos ativos de informação, gestão de incidentes de segurança e forense computacional;

Organização da segurança: definição e relacionamento da coordenação de segurança e demais entidades e grupos envolvidos com a segurança; definição dos papéis e responsabilidades;

Métricas: medições dos processos e controles cujos resultados visam a análise crítica do seu desempenho e a tomada de decisão;

Auditoria e conformidade: garantia de aderência às regulamentações, contratos, políticas, normas e procedimentos operacionais;

Cultura: abrange a conscientização, treinamento e educação permanente em segurança da informação.

2.2.1. Gestão de identidade e acesso

Gestão de identidade é a coleção de componentes de tecnologia, processos e práticas padronizadas, estruturadas para Autenticação, Autorização e Auditoria (AAA) dos usuários acessando os serviços providos por uma organização.

Autenticação: processo de verificação da identidade de um usuário ou sistema (ex. LDAP verificando as credenciais apresentadas pelo usuário). Para tecnologia de nuvem a autenticação usualmente implica em uma forma mais robusta de identificação e em alguns casos de uso, tais

como interação serviço-a-serviço, envolve verificar se os serviços de rede estão requisitando acesso para informações servidas por outro serviço.

Autorização: processo que determina os privilégios conferidos ao usuário ou sistema uma vez que a identidade é estabelecida, sendo usualmente subsequente ao passo de autenticação. Autorização é o processo que reforça as políticas de acesso da organização.

Auditoria: processo que exige a revisão e exame dos registros de autenticação e autorização para determinar a adequação dos controles de sistema e verificar conformidade com políticas e procedimentos estabelecidos.

Com o objetivo de suportar o negócio da organização, a gestão de identidade deve suportar os processos:

Gestão de usuários: atividades para governança efetiva e gerenciamento do ciclo de vida da identidade;

Gestão de autenticação: atividades para governança efetiva e gerenciamento de processos para determinar que uma entidade é quem reivindica ser;

Gestão de autorização: atividades para governança efetiva e gerenciamento de processos para determinar os direitos de acesso que determinam qual recurso uma entidade (ex. usuários, sistemas ou componentes de sistemas) tem permissão para acessar de acordo com as políticas da organização;

Gestão de acesso: reforço de políticas para o controle de acesso em resposta a solicitação de uma entidade demandando acesso a um recurso de TI dentro da organização;

Gestão e provisionamento de dados: propagação da identidade e dos dados para autorização aos recursos de TI por meio de processos automatizados ou manuais;

Monitoração e auditoria: monitorar, auditar e reportar conformidade referente ao acesso a recursos e sua utilização pelos usuários da organização com base em políticas previamente definidas.

No ambiente de computação em nuvem a Gestão de Identidades requer a implementação de funcionalidades visando permitir a execução de atividades essenciais, como:

- o provisionamento de acesso aos dados e recursos de TI;
- a gestão de credenciais e atributos, incluindo o tratamento de senhas e outras formas de identificação de usuários (biometria, certificado digital, etc.);

• a gestão de direitos de acesso através do estabelecimento e manutenção de políticas de acesso; e

• a gestão de conformidade, contemplando a capacidade de monitorar e rastrear o uso dos recursos da organização e finalmente a gestão de federação de identidade, visando o gerenciamento de relacionamentos de confiança estabelecidos além dos limites da rede ou dos limites de domínio administrativos da organização.

A tecnologia de federação é tipicamente empregada no ambiente de computação em nuvem, devendo ser construída sobre uma arquitetura de gestão de identidade centralizada, cujos padrões de indústria são reconhecidamente: SAML (Security Assertion Markup Language), WS- (WS Federation) ou Liberty Alliance, sendo que das três principais famílias de protocolos associadas com federação, SAML parece ser reconhecida como um padrão de fato para federações controladas por organizações distintas.

2.2.2. Segurança de pessoas

As principais questões relacionadas à segurança de pessoas na computação em nuvem são similares àquelas existentes em outros ambientes/ tecnologias de TIC. Essas questões dizem respeito às ações realizadas antes, durante e no encerramento da contratação da força de trabalho envolvidas nas operações da computação em nuvem. Como não poderia deixar de ser, existe o compromisso entre os riscos e o custo da implantação dos controles associados.

De acordo com a NBR ISO/IEC 27002, as ações antes da contratação referem-se à definição de papéis e responsabilidades não somente com relação aos trabalhos na nuvem mas também com destaque nas questões da política de segurança. Isso inclui os controles na contratação e na alocação de pessoal ao trabalho, checagem pré admissional, formalizando os termos e condições de trabalho, obrigações contratuais, regulamentações, políticas e procedimentos a serem seguidos e executados.

Enquanto perdurar o contrato de trabalho devem estar conscientizados com relação à segurança, obrigações e responsabilidades, capacitados para a correta execução das atividades e responsabilizados nos casos de violação da política de segurança. Importante destacar a necessidade de treinamento

contínuo de forma a manter atualizados os conhecimentos para o desempenho de suas funções.

No encerramento da contratação ou na mudança de trabalho são destacadas as questões relativas a devolução de ativos e cancelamento ou adequação dos diretos de acesso.

2.2.3. Gestão de tratamento e resposta a incidentes

Tratamento e resposta a incidentes é o processo responsável por identificar, analisar e mitigar incidentes computacionais de forma rápida e precisa, comunicando as informações relevantes para a alta direção, clientes e parceiros. A atividade de tratamento de incidentes é realizada por uma equipe especializada com definições de atividades e período de operação que atenda as necessidades e exigências do serviço.

O processo de tratamento e resposta a incidentes deve ser realizado para os ambientes computacionais onde há a exposição de sistemas e serviços a eventos maliciosos e tentativas de ataque, com abrangência tanto voltada a computação tradicional como também os serviços de computação em nuvem. A computação em nuvem não traz mudança do paradigma cumprido pelo processo de tratamento de incidentes, mas apresenta novas características de serviços que requerem a ampliação do conhecimento com base em novas vulnerabilidades e ameaças aos serviços em nuvens. Como exemplo, a tecnologia de virtualização poderá apresentar vulnerabilidades que se estendem as máquinas virtuais, instâncias de sistemas operacionais e aplicações que podem transcender a um ambiente virtual e afetar a integridade, disponibilidade e confidencialidade de vários serviços e usuários que compartilham fisicamente os mesmos recursos. O tratamento e resposta a incidentes pode ser mapeado em macro atividades envolvendo:

Preparação: treinamento e capacitação da equipe de tratamento de incidentes de segurança computacional, de forma a estar pronta para o processo de resposta quando da ocorrência de incidentes.

Identificação: procurar pela causa de um incidente (caso intencional ou não) desenvolvendo atividades de , rastreamento do problema através de múltiplos níveis do ambiente de computação em nuvem. A equipe de tratamento de incidentes colabora com membros de outras equipes internas para diagnosticar a origem de um dado incidente de segurança. Essa atividade também inclui as atividades de detecção de vulnerabilidades e assinaturas

utilizadas e a triagem de eventos que se configuram em ataques com violação de dados, equipamentos, sistemas e aplicações que compõe o serviço.

Contenção: uma vez que a causa do incidente foi encontrada e houve o reconhecimento do ataque, a equipe de tratamento de incidentes trabalha com as equipes necessárias implementando ações para conter o incidente. Como o processo de contenção ocorre, depende dos impactos na infraestrutura, nos serviços e negócios causados aos clientes, a partir do incidente.

Mitigação: a equipe de tratamento de incidente coordena em conjunto com equipes de entrega de produtos e serviços as ações para reduzir o risco de recorrência do incidente.

Recuperação: em continuidade ao trabalho com outros grupos, quando necessário, a equipe de tratamento de incidentes apoia o processo de recuperação do serviço.

Lições aprendidas: depois da resolução do incidente, as equipes envolvidas se reúnem buscando o registro das estratégias aplicadas e lições aprendidas durante o processo de resposta.

O processo de tratamento de incidentes deve assegurar também que os eventos de segurança e vulnerabilidades associadas com sistemas são comunicadas de forma a permitir que as ações de correção sejam tomadas. Relato de evento formal e procedimentos de escalação devem ser adotados e reportados ao cliente proprietário do sistema ou serviço impactado. As responsabilidades e papéis dentro do processo de gestão de incidentes devem estar claramente definidos para ambas as partes: provedor e clientes.

2.2.4. Gestão de riscos

A gestão de riscos se constitui num importante processo na definição do tipo de proteção a ser dada à informação, de forma a que os controles de segurança sejam definidos, aplicados e gerenciados, considerando as necessidades de negócio.

Gestão de riscos pode ser considerada como um processo sistematizado de avaliação e tratamento dos riscos, por meio da adoção de controles a serem implementados, de forma a proteger a informação de danos que possam ser causados por falhas de segurança. Neste processo, consideram-se ameaças, vulnerabilidades e impactos que possam afetar os recursos, a

probabilidade dessas ocorrências, a viabilidade da adoção dos controles necessários e a aceitação e comunicação dos riscos. Neste contexto, alguns termos são considerados:

Ameaça: o que tem potencial para causar perda ou dano.

Vulnerabilidade: fraqueza que pode ser explorada.

Impacto: resultado negativo da exploração de uma vulnerabilidade.

Risco: combinação da probabilidade de um evento e sua consequência.

Avaliação dos riscos: processo de identificação e valoração dos riscos.

Tratamento dos riscos: processo de seleção e implementação de controles para modificar um risco.

Controle: forma de gerenciar o risco, incluindo políticas e procedimentos, práticas ou estruturas organizacionais.

Muitos dos riscos frequentemente associados com a computação em nuvem não são novos e já são conhecidos nas organizações hoje em dia. Como ilustração, são relacionados a seguir alguns dos principais riscos da computação em nuvem, de acordo com o ISACA, Enisa e Gartner Group:

- Perda de governança;
- Falta de viabilidade a longo prazo;
- Falha no manuseio das informações (quebra do ANS – confidencialidade e disponibilidade);
- Falha na deleção de dados;
- Perda da compartimentação de dados entre os clientes;
- Falha nos mecanismos de isolamento de ativos;
- Falha na segregação de dados;
- Não conformidade com requerimentos ou legislação;
- Falha de backup, resposta a incidentes e recovery;
- Acesso privilegiado aos dados; e
- Ataque interno por empregado malicioso.

É importante destacar que o processo de gestão de riscos deve ser permanente, de maneira a deixar claro para a organização os riscos potenciais a que ela está sujeita enquanto provedor. A computação em nuvem acrescenta riscos adicionais ao ambiente, que devem ser tratados não apenas do ponto de vista operacional, mas também do ponto de vista legal e de negócio, visando atingir os níveis apropriados de segurança e privacidade.

2.2.5. Gestão de continuidade de negócios

De acordo com a Norma NBR 15.999, gestão de continuidade de negócios é a capacidade estratégica e tática da organização de se planejar e responder a incidentes e interrupções de negócios, para conseguir continuar suas operações em um nível aceitável previamente definido.

A gestão da continuidade de negócios é um processo abrangente de gestão que identifica ameaças potenciais para uma organização e os possíveis impactos nas operações de negócio, caso estas ameaças se concretizem.

De acordo com a Norma o ciclo de vida da gestão de continuidade de negócios está composta dos seguintes elementos:

- Gestão do programa de GCN (abrangência, política, responsabilidades, SGCN);
- Levantamento de informações da organização (BIA, análise de riscos, interdependências, priorização;
- Definição da estratégia de continuidade;
- Desenvolvimento dos planos (plano de gerenciamento de incidentes, plano de continuidade de negócios, plano de recuperação de negócios);
- Execução de testes (programa de testes); e
- Inclusão da GCN na cultura da organização (conscientização e treinamento).

O modelo PDCA deve ser aplicado ao SGCN (sistema de gestão de continuidade de negócios) com o objetivo de garantir que a continuidade de negócios esteja gerenciada e seja aprimorada de forma eficaz.

2.2.6. Gestão de auditoria e conformidade

Um modelo sustentável de computação em nuvem deve prever uma política de auditoria e de conformidade, observando que essas políticas devem se basear na política de riscos.

A auditoria pode ser interna ou externa, deve verificar se os controles definidos para riscos críticos estão aplicados conforme os requisitos especificados no plano de negócio enquanto que a conformidade é uma atividade de rotina e tem o objetivo de verificar o quanto os controles estão aplicados corretamente, em conformidade com os dispositivos legais e os indicadores de melhoria.

Considerações básicas de auditoria:

- deve seguir normas internacionais, como SAS 70, *Type* II e ISO 27001;

Considerações básicas de conformidade:

- deve verificar se os objetivos de negócio, os contratos de clientes e as politicas, padrões e procedimentos internos estão sendo atendidos, de forma a aplicar as correções em tempo hábil;
- identificar e corrigir os fatores que interferem no resultados dos objetivos;
- identificar os requisitos/requerimentos críticos e checar se a prática das politicas, procedimentos, processos e sistemas estão atendendo;
- monitorar e checar o cumprimento dos requisitos; e
- construir um modelo sustentável de forma a estabelecer controles fortes e que possam ser aplicados a todos os clientes.

2.2.7. Forense computacional

Trata-se de processo de investigação em ambiente digital com o objetivo de identificar, por meio de análises desses ambientes como ocorreu o incidente. O processo de análise forense se sustenta em requerimentos legais, politicas sobre a captura e retenção de registros de *logs* de *firewall*, base de dados, dispositivos móveis e outros componentes, mídias, para garantir a análise dos fatos.

2.3. Segurança em camadas

A abordagem de defesa em camadas é um elemento fundamental visando prover uma infraestrutura de nuvem confiável e segura, permitindo delimitar controles em múltiplos níveis, empregar mecanismos de proteção e estratégias de mitigação de risco de forma organizada e orientada às principais necessidades de segurança de cada segmento na prestação dos serviços.

Os serviços que utilizam os conceitos de computação em nuvem, SaaS, IaaS ou PaaS, são virtualizados de forma que os clientes podem ter ativos que não são facilmente associados com a presença física devido aos diferentes níveis de compartilhamento. O dado também pode ser armazenado virtualmente e distribuído através de múltiplas localizações. Basicamente isto

significa que identificar controles de segurança de forma organizada por camadas e determinar sua utilização de forma compartilhada, atendendo aos requisitos de segurança específicos de cada nível, deverá agilizar o processo de implementação das soluções de serviço no modelo de computação em nuvem.

2.3.1. Segurança física

O controle de acesso físico compreende uma política, procedimentos e tecnologia que estabeleçam as condições e controles para acesso autorizado de pessoas a ambientes físicos da organização, desde os ambientes de escritórios até os de infraestrutura.

Os controles de acesso pressupõem o uso de ferramentas de monitoração a fim de identificar o acesso autorizado e bloquear o não autorizado, além de acompanhar o deslocamento do usuário nas instalações. A ferramenta para suportar o controle deve ser compatível com a criticidade do ambiente.

2.3.2. Segurança de rede

Os controles tradicionais de segurança na camada de rede, obtidos com implementação de firewalls, gateways de aplicação, dispositivos de prevenção e detecção de intrusão, devem continuar. O ponto focal da gestão de risco desloca-se para mais próximo do nível de objetos em uso no ambiente de nuvem, reforçando as necessidades de segurança de dados, como exemplo dos recipientes de armazenamento estático ou dinâmico e dos objetos de máquinas virtuais e ambientes run-time. Esses objetos, sobre a visão de segurança de rede, deverão estar compartimentados ou segmentados em redes virtuais privadas (VPNS ou VLANs) conforme as características do modelo de implementação da nuvem aos quais estão atrelados.

No modelo de nuvem privada ou interna não há novos ataques, vulnerabilidades e mudanças, pois embora a arquitetura de TI possa mudar com a implementação de nuvem privada, sua topologia de rede corrente provavelmente não irá modificar significantemente. Nesse modelo é importante manter a garantia de isolamento de tráfego do cliente proprietário da nuvem, por meio da adoção de VPNs e aplicar os mecanismos de segurança para filtragem de tráfego em listas de controles de acesso configurados em roteadores e firewalls.

Por outro lado, a adoção de serviços utilizando o modelo de nuvens públicas deverá modificar os requisitos de segurança e a topologia de rede, endereçando questões como a forma de integração com outras topologias (outras nuvens públicas, nuvem privadas, comunitárias ou híbridas). No caso de implementação de nuvens públicas, os seguintes fatores de risco são significantes:

- confidencialidade e integridade dos dados em trânsito "para" ou "de" uma nuvem pública de provedor. Alguns recursos e dados anteriormente confinados em uma rede privada estão agora expostos para a Internet e o compartilhamento de redes públicas pertencendo a provedores de nuvens terceirizados;
- disponibilidade dos recursos a partir da Internet em uma nuvem pública que tenha sido designada para um cliente pelo provedor de nuvem. O provedor de nuvem deve reconhecer a abrangência de problemas envolvendo ataques de disponibilidade. No modelo de serviços IaaS, o risco de uma ataque tipo DDOS não é somente externo (serviços expostos a internet), mas há também o risco de que ataques tipo DDOS através da porção da IaaS afetem outros serviços privados, se o provedor de rede utilizou a mesma infraestrutura para seus clientes internos. Nesse caso, a estrutura de rede interna (não roteável) foi construída com base em recursos compartilhados e poderá ser afetada ou comprometida por problemas de disponibilidade de outros clientes; e
- modelo de zonas e segmentos de rede substituídos por controles lógicos de domínios. Considerando-se as modalidades de serviços IaaS e PaaS disponibilizados em nuvens públicas, não existe um modelo de isolamento estabelecido em Zonas desmilitarizadas (ZDMs) ou segmentos de rede. O modelo tradicional foi substituído em nuvens públicas como "grupos de segurança" ou "domínios públicos" ou "centros de dados virtuais" que tem separação lógica entre os segmentos e são muito menos precisos, suportando um menor nível de proteção em comparação aos modelos tradicionais.

Os fatores de riscos devem ser tratados ou mitigados através da aplicação de controles preventivos ou detectivos:

[2] http://rationalsecurity.typepad.com/blog/2009/01/a-couple-of-followups-on-my-edos-economic-denial-of-sustainability-concept.html

Controles Preventivos	Ferramentas de controle de acesso a rede e aplicações (*firewall* de redes ou de *hosts* incluídos em máquinas *VM* ou *hosts* virtuais) e criptografia de dados sensíveis em trânsito (*SSL, IPSEC*)
Controles Detectivos	Mecanismos de prevenção de intrusão e ferramentas de gestão, agregação e correlação de eventos e *logs* (*SIEM*), ferramentas de detecção e prevenção de intrusão em rede (*Network IDS/IPS*)

Fonte: Cloud Security and Privacy - Tim Mather, Subra Kumaraswamy, and Shahed Latif

Outro aspecto importante é a monitoração de tráfego, responsável pela coleta, armazenamento e análise em tempo real dos dados provenientes dos ativos de segurança de rede, tais como firewalls, gateways de aplicação, dispositivos de prevenção de intrusão. As equipes de gerenciamento de acesso aplicam Listas de Controle de Acesso (*ACLs*) alinhadas aos segmentos de redes virtuais (V*LANs*), aplicações e serviços conforme necessário, sendo portanto essencial a visão e reconhecimento de eventos e alertas causados no ambiente de rede. Ressalta-se também a necessidade de integração entre os processos de: monitoração, correlação e gestão de eventos (realizado pelo COS – Centro de Operação em Segurança) e o tratamento de incidentes, visando a tomada de ações de proteção e defesa dos ambientes computacionais da nuvem.

2.3.3. Segurança de dados

Com a forte tendência mundial de se ampliar o acesso aos repositórios de dados para a sociedade de uma forma geral, a segurança de dados em computação em nuvem envolve diversos riscos associados com a confidencialidade e integridade dos dados para clientes, usuários e prestadores de serviço de nuvem.

O cliente da nuvem deve se responsabilizar pela proteção dos dados confidenciais e críticos para o seu negócio. A dificuldade do cliente da nuvem, no seu papel de controlador de dados, está efetivamente em verificar o processamento de dados que o prestador de nuvem realiza e ter a certeza de que os dados são tratados de forma lícita e segura.

Por ser uma arquitetura distribuída, a computação em nuvem implica em um aumento significativo dos dados em trânsito em relação às infraestruturas tradicionais. Os dados devem ser transferidos para sincronizar imagens de várias máquinas distribuídas, imagens distribuídas em várias máquinas físicas, entre infraestrutura de nuvens e clientes web remotos. As técnicas de ataques por sniffing,

spoofing, man-in-the-middle, canal lateral e replay são potenciais fontes de ameaças.

A possibilidade de ocorrer violações de segurança dos dados que não são notificados ao controlador pelo provedor da nuvem podem ser minimizados com a configuração do SGBD para criptografar as tabelas, utilizar sistema de arquivos com proteção criptográfica a nível de sistema operacional, utilizar mecanismos de verificação de hash ou assinatura digital, implementar mecanismos que previnam contra a perda acidental ou intencional dos dados, arquivos de logs para fins de auditorias/forenses e uso de criptografia e protocolos seguros, como SSL/TLS e IPSEC para garantir a comunicação segura dos dados.

Um dos graves problemas da aplicação cliente na nuvem é perder o controle dos dados processados pelo prestador de nuvem. Esse problema é maior no caso de múltiplas transferências de dados, por exemplo, entre os prestadores de nuvens federadas. Como o provedor de nuvem pode receber dados que não foram legalmente coletados pelos seus clientes (controladores), isso pode gerar problemas jurídicos entre o provedor da nuvem e o usuário da nuvem. Para se ter o controle deste problema, é fundamental que exista um contrato de prestação de serviço e responsabilidade legal dos dados armazenados, além de processos bem definidos de gestão do ciclo de vida dos dados (guarda, descarte e destruição de dados) integridade dos backups, tratamento de incidentes e plano de continuidade, rastreabilidade (arquivos de logs) para suportar os processos forenses e preservação de provas (cadeia de custódia).

Outro fator importante é a compreensão dos riscos associados à dependência de fornecedores referente a portabilidade de dados e serviços. Recomenda-se possuir procedimentos documentados e APIs para exportar dados da nuvem, interoperar em formatos padrões, permitir que o cliente possa efetuar a sua própria extração de dados. Além disso, é exigência que o cliente tenha a liberdade exportar seus dados para outro provedor de nuvem e garantia de poder mudar de provedor de forma transparente e sem impactos para seu negócio.

2.3.4. Segurança de aplicação

Projetar e implementar aplicações orientadas para a implantação em uma plataforma de computação em nuvem demanda que o programa de segurança existente seja reavaliado quanto as práticas e padrões atuais, principalmente quanto a adoção de um processo de segurança no desenvolvimento de aplicações.

Atualmente, é uma prática comum utilizar uma combinação de controles de segurança de perímetro e controle de acesso à rede e servidores para proteger aplicativos Web implantados em um ambiente controlado, incluindo redes internas (intranets) corporativas e nuvens privadas. Neste novo cenário, uma vez que o navegador é tido como o principal aplicativo cliente para que o usuário final acesse aplicativos na nuvem, é fundamental que as ações de segurança contemplem a inclusão de controles de segurança para o navegador no âmbito da segurança de aplicativos.

2.3.4.1. Ciclo de desenvolvimento seguro

É essencial que todos os fornecedores de software tratem as ameaças à segurança do software. A segurança é um requisito fundamental para todos os fornecedores de software, e ela é influenciada pelas pressões do mercado, pela necessidade de proteger as infraestruturas críticas e pela necessidade de criar e preservar a confiabilidade no ambiente de TIC. Um desafio importante para todos os fornecedores de soluções é a criação de softwares mais seguros, que precisem de menos atualizações através de correções (patches) e possibilitem um gerenciamento de segurança com poucas ocorrências de problemas.

Para que os softwares sejam construídos efetivamente de forma segura é necessário que as atividades de verificação e validação de segurança sejam adequadamente distribuídas em todo ciclo de desenvolvimento, conforme a figura abaixo:

Ciclo de desenvolvimento seguro

Fonte: Cloud Security and Privacy Tim Mather et al, 2009

2.3.4.2. Aspectos relevantes quanto a disponibilidade da aplicação

- DoS e EDoS (economic denial of sustainability)

Existem ataques no nível da aplicação de negação de serviços (DoS - Denial of service e DDoS - Distributed Denial of service) que podem potencialmente interromper os serviços contidos na nuvem por um tempo indesejado e indeterminado. Ataques de negação de serviço podem manifestar-se como um grande volume de requisições de atualização de páginas web, requisições XML para *web services* (sobre HTTP ou HTTPS) ou requisições especificas de outros protocolos suportados pela nuvem.

Ataques *DoS* em aplicações de computação em nuvem que realizam procedimentos de contabilização por consumo de recursos irão resultar em um aumento dramático na conta do serviço contido na nuvem. Neste contexto será observado maior utilização da largura de banda de rede, CPU, armazenamento e consumo. Este tipo de ataque também está sendo caracterizado como a negação de sustentabilidade econômica (*EDoS*[3]).

- SaaS

Provedores de serviços SaaS são responsáveis pela segurança das aplicações e componentes que oferecem aos seus clientes. Os clientes são normalmente responsáveis pelas atribuições de segurança operacional, incluindo o gerenciamento de acesso e de usuários. É uma prática comum para potenciais clientes, geralmente sob acordo de confidencialidade, solicitar informações relacionadas com as práticas de segurança do fornecedor. Estas informações deverão abranger design, arquitetura, desenvolvimento, testes de segurança (caixa branca e preta) nos aplicativos e modelo de gerenciamento de versões.

- PaaS

Por definição, uma nuvem de serviços PaaS (pública ou privada), oferece um ambiente integrado que proporciona a realização do design, codificação, testes e implantação. A segurança de aplicações para a PaaS abrange duas camadas de software:

[3] http://rationalsecurity.typepad.com/blog/2009/01/a-couple-of-followups-on-my-edos-economic-denial-of-sustainability-concept.html.

• Segurança da própria plataforma PaaS (por exemplo: mecanismo de *runtime*); e
• Segurança das aplicações dos clientes implantadas na plataforma PaaS.

Os Desenvolvedores geralmente devem esperar que os provedores de serviço de nuvem ofereçam um conjunto de funcionalidades de segurança, incluindo autenticação de usuário, single *sign-on* (SSO) usando federação, autorização e suporte a SSL ou TLS, disponibilizados através de API.

- IaaS

Provedores de nuvens de serviços IaaS (por exemplo, a Amazon EC2, GoGrid e Joyent) tratam as instâncias virtuais como uma caixa preta, portanto, não possuem controle das operações e gerenciamento de aplicações do cliente. Assim, o cliente é o principal responsável pelas ações de segurança nas aplicações.

Em resumo, a arquitetura para aplicações hospedadas em uma nuvem IaaS se assemelha as aplicações web corporativas com uma arquitetura distribuída em n-camadas. Em uma empresa, aplicações distribuídas são executadas tendo vários controles visando a segurança dos servidores e a rede de conexão distribuídos.

2.3.5. Segurança de servidores e armazenamento de dados (storage)

Os riscos de segurança dos servidores devem ser considerados com vista no modelo de entrega de serviço da computação em nuvem (SaaS, PaaS e IaaS), bem como os modelos de implantação (público, privado e híbrido).

Os modelos de serviços SaaS e PaaS possuem uma camada de abstração do servidor que esconde do cliente o sistema operacional em uso. A diferença chave entre esses dois modelos está no nível de acesso à esta camada de abstração:

• no SaaS a camada de abstração não é visível aos usuários, ela está disponível somente para os funcionários do provedor de serviços em nuvem. Neste modelo, o provedor de serviços em nuvem desenvolve e implanta a aplicação; e

• no PaaS é dado ao cliente (usuário) um acesso indireto à camada de abstração, por meio de uma API que, por sua vez, acessa esta camada. Neste modelo, o próprio cliente pode desenvolver e implantar a aplicação.

Em ambos os modelos, a responsabilidade pela segurança do servidor é do provedor de serviços de computação em nuvem.

Ao contrário dos modelos SaaS e PaaS, a segurança dos servidores fornecidos por uma IaaS são de responsabilidade do cliente. Partindo do princípio que todos os serviços IaaS implementam virtualização, duas categorias de segurança devem ser consideradas:

• Segurança no software de virtualização;

• Segurança nos servidores virtuais ou sistemas operacionais hospedeiros.

Apesar de não haver novas ameaças específicas para os servidores da computação em nuvem, alguns vulnerabilidades de segurança estão presentes neste ambiente, tais como:

• ameaças contra a integridade e a disponibilidade do *hypervisor*;

• escalonamento de privilégios; e

• uso de comandos maliciosos para sair do ambiente virtual *(virtual machine escape)*.

2.3.6. Segurança de estação de trabalho (*endpoint*)

Quando se trata de gerar economia para as organizações, as soluções de virtualização de *endpoints* garantem um espaço de trabalho mais produtivo (*desktops*, aplicativos e informações) e disponível o tempo todo. Porém o desafio é gerenciar a segurança destes *desktops* dinâmicos com o provisionamento de aplicativos e eliminação de conflitos.

A Distribuição sob demanda de aplicativos para *endpoints* físicos e virtuais a fim de garantir proatividade na conformidade de licenças e recuperação automatizada de licenças para otimização de custos, exige controles efetivos para regular o uso de forma segura.

Estes ativos podem se beneficiar por exemplo de um serviço de segurança completamente gerido externamente, permitindo-lhes monitorizar e e configurar as suas próprias proteções *anti-malware* a partir de qualquer local e a qualquer momento, ou atribuindo essa responsabilidade a um provedor de serviços geridos.

2.4. Resiliência Operacional

Resiliência é um termo que tem origem nas ciências exatas e diz respeito à capacidade de um elemento retornar ao estado inicial após sofrer influência externa. De uma forma mais detalhada, é a capacidade de:

- mudar (adaptação, expansão, conformidade e conforto) quando uma força é aplicada;
- funcionar adequadamente ou minimamente enquanto a força é efetiva; e
- retornar a um estado normal (esperado e pré-definido) quando a força se torna ineficaz ou é retirada.

Considerando o modelo de negócio e o modelo de entrega de serviço de computação em nuvem adotados, o conceito de resiliência pode ser aplicado nos seguintes níveis:

Negócio: habilidade de uma organização, recurso ou estruturas para sustentar o impacto de uma interrupção de negócios, recuperar e retomar suas operações para prestar níveis mínimos de serviço. Os seguintes aspectos precisam ser considerados:

- criação de conscientização nas questões de resiliência;
- seleção de componentes organizacionais essenciais;
- auto avaliação das vulnerabilidades;
- identificação e priorização das vulnerabilidades principais; e
- ações visando aumentar a capacidade adaptativa.

Sistemas: sistemas desenvolvidos com a capacidade de fornecer os serviços para os quais foi projetado e mantê-los em níveis aceitáveis de performance, mesmo na presença de condições negativas.

Um sistema resiliente deve (tipicamente) possuir as seguintes capacidades:

- responder de forma rápida e eficiente aos distúrbios normais e ameaças;
- monitorar de forma contínua os distúrbios e ameças, além de revisar os padrões de monitoração; e
- antecipar mudanças futuras no ambiente que possam afetar a forma de funcionamento do sistema e se preparar para tais mudanças.

Operações: propriedade associada com as atividades que a organização executa visando manter serviços, processos de negócios e ativos viáveis e produtivos mesmo sob condições de risco.

Apresenta um conjunto de ferramentas, técnicas e metodologias para auxiliar as organizações na condução de uma ação integrada entre os processos de gestão de operações de TI, gestão da continuidade e gestão da segurança. Como uma base comum para estes processos está a gestão de riscos, que permite identificar as ameaças, vulnerabilidades e elencar os controles adequados para minimizar os riscos.

Infraestrutura: é geralmente confundido na prática com recuperação de desastres ou planejamento de continuidade de negócios. De forma ideal, as atividades de recuperação e continuidade devem ser ativadas apenas nas situações de falha na resiliência.

Engenharia de resiliência: processo no qual uma organização projeta, desenvolve, implementa e gerencia a proteção e a sustentabilidade de seus serviços críticos, relacionados com os processos de negócio e associados aos ativos (pessoas, informação, tecnologia e instalações). Possui os seguintes desafios:

- construir sistemas capazes de se recuperar de distúrbios; e
- ter a segurança como uma propriedade emergente do sistema ao invés de tê-la baseada exclusivamente na dependência de componentes confiáveis.

3. Segurança como serviço

A adoção de segurança como serviço permite a entrega de controles de segurança estruturados e automatizados na modalidade de serviços que são de propriedade, entregues ou gerenciados remotamente por um ou mais provedores. O provedor entrega um serviço de segurança baseado em um conjunto compartilhado de tecnologias, melhores práticas e métricas que são consumidos no modelo de "um para muitos" pelos clientes a qualquer instante com pagamento pelo uso ou como uma assinatura.

Os serviços definidos nessa seção foram selecionados com base na capacidade de provimento de soluções desenvolvidas a partir do

conhecimento já existente no escopo de segurança pelas várias áreas de uma organização e que possam evoluir do ambiente de TI atual para o ambiente de tecnologia de computação em nuvem. O benefício primordial aos clientes dos serviços de segurança é o uso de múltiplos mecanismos, técnicas e processos, cujo compartilhamento da estrutura traz como consequência direta a redução de custos e complexidade de implementação para o cliente. Ressalta-se ainda o fato de que esses serviços requerem recursos técnicos (pessoas) com alto nível de especialização e gestão nos diversos temas que compõe a segurança da informação, disciplina esta muitas vezes distante dos objetivos finais de negócio do cliente. Dessa forma, o cliente pode então fazer uso de avançados métodos, técnicas, ferramentas e processos de segurança visando a proteção de seus serviços sem a necessidade de obter especializações fora de seu contexto de atuação.

3.1. Centro de operações em segurança

A utilização de um modelo central de monitoração de ambientes de infraestrutura pode ser um componente essencial para controle de aplicações web no segmento de computação em nuvem e ainda pode ser um instrumento fundamental para as atividades do programa de governança. Adicionalmente, deve ser observado que a estrutura de controle deve ser aderente com a política de governança adotada pela organização, os aspectos legais jurisdicional e contratuais e os padrões de segurança definidos para prevenção de ataques nos serviços web, controle da rede, sistemas, aplicações e serviços.

Benefícios do serviço para os clientes:

- dispor do controle eficiente e eficaz para o tratamento de alertas e eventos de segurança, com a correlação das diversas infraestruturas de TIC no provimento de serviços em nuvem;
- possibilitar o acompanhamento em tempo real da saúde da segurança a partir de relatórios e acessos a um ambiente de gerenciamento de segurança; e
- disponibiliza informações para identificação de registros de logs e outros eventos no ambiente de TIC do provedor com a finalidade de auditoria e investigação forense.

3.2. Gerenciamento de vulnerabilidade

Esse serviço compreende o fornecimento de informações resultantes da análise de vulnerabilidade sobre os ativos de sistemas do cliente. O provedor de serviço de computação em nuvem utiliza técnicas e ferramentas para descoberta de falhas e vulnerabilidades em componentes de software (sistemas, aplicações e serviços) que podem ser exploradas no ambiente de produção e formula as ações de correções que precisam ser aplicadas para eliminar riscos potenciais de ataque sobre o componente em referência.

Benefícios do serviço para os clientes:

- o reconhecimento e a apresentação de soluções possibilitam eliminar ataques oriundos da exploração de vulnerabilidades sobre os componentes de software; e
- reforço da gestão de segurança das aplicações disponibilizadas como serviço pelos clientes, com a redução do risco potencial de exploração por ataques em seus ativos.

3.3. Gerenciamento do Tratamento de incidentes

O serviço de tratamento de incidentes compreende a adoção de ferramentas, processos e técnicas que permitam identificar, analisar e mitigar incidentes computacionais de forma rápida e precisa, comunicando as informações relevantes para a alta direção, clientes e parceiros.

Os clientes terão a partir desse serviço, relatórios de eventos de segurança e vulnerabilidades associadas com sistemas identificados no ambiente de produção, permitindo que as ações de correção sejam tomadas. Quando da ocorrência de incidentes, o relato formal dos procedimentos de tratamento e escalação adotados são reportados ao cliente proprietário do sistema ou serviço impactado.

Benefícios do serviço para os clientes:

- uso de ferramental adequado, técnicas e processos pelo provedor do serviço de computação em nuvem para mitigar e conter incidentes de segurança permitindo aumento da segurança e confiança do serviço disponibilizado na nuvem do provedor;

• estabelecimento de controles voltados ao tratamento de incidentes é requisito de diversos processos de certificação em gestão da segurança da informação que pode ser necessária ao sistema ou serviço do cliente; e

• aumento da da confiança e credibilidade com foco nos serviços disponibilizados ou utilizados pelo cliente.

3.4. Gerenciamento de vulnerabilidade

Esse serviço compreende o fornecimento de informações resultantes da análise de vulnerabilidade sobre os ativos de sistemas do cliente. O provedor de serviço de computação em nuvem utiliza técnicas e ferramentas para descoberta de falhas e vulnerabilidades em componentes de software (sistemas, aplicações e serviços) que podem ser exploradas no ambiente de produção e formula as ações de correções que precisam ser aplicadas para eliminar riscos potenciais de ataque sobre o componente em referência.

Benefícios do serviço para os clientes:

• o reconhecimento e a apresentação de soluções possibilitam eliminar ataques oriundos da exploração de vulnerabilidades sobre os componentes de software; e

• reforço da gestão de segurança das aplicações disponibilizadas como serviço pelos clientes, com a redução do risco potencial de exploração por ataques em seus ativos.

3.5. Filtro de correio Eletrônico (anti-malware, Anti-spam, anti-phishing)

O serviço de filtro para correio eletrônico compreende a limpeza de mensagens de correio contendo spam e phishing e a identificação de malware incluído no fluxo de e-mail entrante na organização, permitindo a entrega de e-mails limpos e seguros para organização.

Spam: e-mails não solicitados, que geralmente são enviados para um grande número de pessoas (fonte: http://cartilha.cert.br/)

Phishing/scam: tipo de spam amplamente utilizado como veículo para disseminar esquemas fraudulentos, que tentam induzir o usuário a acessar páginas clonadas de instituições financeiras ou a instalar programas maliciosos

projetados para furtar dados pessoais e financeiros (fonte: http://cartilha.cert.br/)

Malware: programas especificamente desenvolvidos para executar ações danosas em um computador. Dentre eles serão discutidos vírus, cavalos de tróia, spywares, backdoors, keyloggers, worms, bots e rootkits (fonte: http://cartilha.cert.br/)

O serviço poderá também incluir backups e arquivamento de caixas de correio, envolvendo o armazenamento e indexação das mensagens e arquivos anexados em um repositório centralizado. O repositório deverá permitir aos clientes indexar e pesquisar por vários parâmetros incluindo faixa de dados, recipiente, assunto, conteúdo e emissor da mensagem. Essas informações são particularmente úteis para propósito de processos legais, que podem ser excessivamente caros e trabalhosos sem essas características.

Benefícios do serviço para os clientes:

- melhoria de performance uma vez que mensagens de correio não terão latência aplicada a mecanismos internos nas estações de trabalhos e servidores de correio dos clientes;
- melhoria da gestão anti-malware realizado no provedor, evitando possíveis infecções na rede ou em recursos do cliente;
- redução do consumo de banda e da necessidade de armazenamento de mensagens pela rede ou servidores do cliente, devido a limpeza de mensagens de spam e phishing; e
- melhoria da efetividade dos canais (recipientes) de comunicação do cliente eliminando a necessidade de realizar esforços de limpeza anti-malware, anti-phishing e anti-spam.

3.6. Filtro de correio Eletrônico (anti-malware, Anti-spam, anti-phishing)

O serviço de filtro para correio eletrônico compreende a limpeza de mensagens de correio contendo spam e phishing e a identificação de malware incluído no fluxo de e-mail entrante na organização, permitindo a entrega de e-mails limpos e seguros para organização.

Spam: e-mails não solicitados, que geralmente são enviados para um grande número de pessoas (fonte: http://cartilha.cert.br/)

Phishing/scam: tipo de spam amplamente utilizado como veículo para disseminar esquemas fraudulentos, que tentam induzir o usuário a acessar páginas clonadas de instituições financeiras ou a instalar programas maliciosos projetados para furtar dados pessoais e financeiros (fonte: http://cartilha.cert.br/)

Malware: programas especificamente desenvolvidos para executar ações danosas em um computador. Dentre eles serão discutidos vírus, cavalos de tróia, spywares, backdoors, keyloggers, worms, bots e rootkits (fonte: http://cartilha.cert.br/)

O serviço poderá também incluir backups e arquivamento de caixas de correio, envolvendo o armazenamento e indexação das mensagens e arquivos anexados em um repositório centralizado. O repositório deverá permitir aos clientes indexar e pesquisar por vários parâmetros incluindo faixa de dados, recipiente, assunto, conteúdo e emissor da mensagem. Essas informações são particularmente úteis para propósito de processos legais, que podem ser excessivamente caros e trabalhosos sem essas características.

Benefícios do serviço para os clientes:

- melhoria de performance uma vez que mensagens de correio não terão latência aplicada a mecanismos internos nas estações de trabalhos e servidores de correio dos clientes;
- melhoria da gestão anti-malware realizado no provedor, evitando possíveis infecções na rede ou em recursos do cliente;
- redução do consumo de banda e da necessidade de armazenamento de mensagens pela rede ou servidores do cliente, devido a limpeza de mensagens de spam e phishing; e
- melhoria da efetividade dos canais (recipientes) de comunicação do cliente eliminando a necessidade de realizar esforços de limpeza anti-malware, anti-phishing e anti-spam.

4. Conclusão

A segurança da nuvem requer uma abordagem estruturada e que seja ágil e elástica. O modelo de segurança apresentado considera estes aspectos e destaca a abrangência e respectivos processos a serem tratados.

Proteger e assegurar para os clientes que os serviços e dados são tratados de acordo com os níveis acordados de disponibilidade, integridade e

confidencialidade é apenas o desafio inicial. Estar em conformidade com padrões nacionais e internacionais é uma forma adequada de conduzir as ações de segurança.

Referências

Universidade de Canterbury – Nova Zelândia – Projeto RESORGS; http://www.resorgs.org.nz/index.shtml.

Universidade Carnegie Mellon – CERT/SEI – CERT resiliency management model – v1.0; http://www.cert.org/resiliency/.

CERT.BR - Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil – Cartilha de segurança http://cartilha.cert.br/

Cloud Security Guidance – IBM recommendations for the implementation of Cloud security Redpaper, IBM – 2009.

Cloud Security and Privacy – Tim Mather, Subra Kumaraswamy, and Shahed Latif, O'Reilly - 2009.

Security Guidance for Critical Areas of Focus in Cloud Computing V2.1 Prepared by the CSA - Cloud Security Alliance . 2009.

Quadrante SERPRO de Tecnologia (QST) - Uma Ferramenta de Apoio à Gestão do Ciclo de Tecnologia em tempos de Novos Paradigmas da Computação nas Nuvens

Almir Fernandes
Coordenador Estratégico de Tecnologia
SERPRO – Rio de Janeiro

Palavras - chave: Indicadores, Sistema de Informação, Inovação, Ciclo de Tecnologia da Informação, Nível de Serviço, Infraestrutura, Avaliação e Produtividade.

Abstract

This article presents the fundamentals of SERPRO Quadrant Technology (QST). A support tool designed to assess the risks and opportunities of technology systems and services produced in SERPRO. The QST is also intended to be a trigger to signal on applications for research projects and innovation in partnerships with academic centers. Contextual aspects are described in this article, including main objectives, motivations, input / output related to this artifact as a key component of SERPRO Technology Observatory.

Keywords: Indicators, Information System, Innovation, information Technology Cycle, infrastructure, Service level, Evaluation and Productivity.

I - Introdução

Esse trabalho apresenta os fundamentos do Quadrante SERPRO de Tecnologia (QST) - Uma ferramenta para apoiar análise de riscos e

oportunidades do ciclo de tecnologia no âmbito de serviços e sistemas estruturadores de governo. O QST visa também ser um sinalizador das demandas e prioridades em pesquisa e inovação junto às entidades parcerias e centros acadêmicos. São descritos aspectos contextuais, as principais motivações, insumos, objetivos que embasam esse artefato-chave como um componente fundamental do Observatório SERPRO de Tecnologia [1].

> *"The more you practice what you know, the more shall you know what to practice…."*
> *— W. Jenkin*

Em tempo de novos paradigmas da computação nas nuvens, não há dúvida que organizações que buscam a sustentabilidade num ambiente de mudanças intensas tomam decisões com base e fatos, dados e indicadores que lhes credenciam como parte da solução para o sucesso de seus clientes. Uma organização pode ser percebida como um sistema de filtro, onde por um lado entram intenções, expectativas e interesses das partes e do outro saem realizações ([2]Fernandes). Quer seja no âmbito de governo ou na iniciativa privada a informação útil é uma entidade que reduz a incerteza sobre um evento ou estado ([3]Lucas), quanto maior a quantidade e qualidade de informação, menor a incerteza e vice versa; o problema é que a informação se deprecia muito rapidamente, daí a necessidade de repositórios inteligentes e meios ágeis para permitir que se possa tirar o melhor proveito de sua utilidade[4]. Hoje no século XXI ninguém questiona a importância da tecnologia da informação (TI) como variável crítica nos principais segmentos da indústria de maneira geral com impactos definitivos em qualidade e produtividade. No entanto, por sua natureza dinâmica, decisões sobre o ciclo de vida da tecnologia nas organizações ainda tem sido motivo de muitos estudos e pesquisas rumo a cenários desejáveis, em resposta, o peso relativo dos investimentos de TI passou a ser um dos itens mais significativos nos orçamentos das mesmas, desafiando-as a conjugarem qualidade e agilidade nas suas decisões. Hoje com a dinamicidade e flexibilidade exigida na nuvem, num ambiente onde os recursos são intensamente compartilhados, os desafios

[1] Portal de Direcionamento Tecnológico e Gestão de Conhecimento de Tecnologia do SERPRO.
[2] Administração Inteligente – Futura – Fernandes, A. a2 Ed – 2003.
[3] Information Systems Concepts for Management – McGraw Hill. Lucas, Henry C – 1990.
[4] Administração Inteligente – Fernandes, A – Futura, 2ª Edição – 2003.

torna-se ainda maiores, as organizações que dominarem esse ciclo de decisões efetivamente terão vantagem competitiva para transformar intenções em efetivas realizações. No mundo real numa perspectiva puramente econômica todo desperdício é caracterizado pelo desequilíbrio da relação entre insumos e saídas traduzidos em produtos e serviços gerados, o que implica no curto prazo, em abrir mão de algo tangível. Isso é o que ([5]Samuelson) define como anomalias na fronteira das possibilidades da produção; o desafio, portanto está em realizar no presente e desenvolver a capacidade de alinhar-se ou mesmo antecipar as tendências futuras. Estudos e pesquisas comprovam que existe uma correlação positiva entre comprometimento com a realização e as escolhas em tecnologia da informação, sendo essa decisão um dos fatores críticos de sucesso das organizações ([6]Laudon); paradoxalmente quando essas decisões envolvem por diversas razões escolhas erráticas invariavelmente conduzem ao insucesso, entre elas estão: falta de visão estratégica para definir o seu próprio rumo, carência de dados e informações para garantir sustentabilidade de processos e projetos prioritários e "religião por tecnologia" entre outros males doutrinários que infestam uma gestão adequada dos ciclos de tecnologia numa organização. Essa realidade ainda é mais complexa em organizações de TIC, onde a obsolescência e apego a tradição em oposição à sistematização e inovação cobra preço alto, quer seja na qualidade dos serviços com perdas para a sociedade como um todo, quer seja traduzidas em desperdício e deseconomias de escala como conseqüência do mau direcionamento e uso de tecnologias. Em síntese uma sociedade que desperdiça não está sendo capaz de aprender com a memória do seu passado, ignora o presente e se torna refém do acaso. O passado é memória, o presente é ação e o futuro construção

II - Motivações e Objetivos do Quadrante SERPRO de Tecnologia - QST

Investimentos em tecnologia de uma forma geral têm um peso significativo no orçamento das empresas e impacto na qualidade dos processos e serviços. Decisões de investimentos há muito deixaram de ser objeto de abordagens

[5] Economics – (9th Edition) – Samuelson, Paul Antony- 1973.
[6] Management Information systems. Managing the Digital firm(8a.ed) Pearson education – 2004.

empíricas ou experiências ao acaso para se apoiar em bases consistentes. Com efeito, experiências mais bem sucedidas nesse campo nos governos no mundo têm demonstrado o papel fundamental das instituições de pesquisa e desenvolvimento em projetos com enfoque na gestão desse conhecimento, pesquisa e inovação. Vários são os exemplos de experiências em todo o mundo que têm cada vez mais reafirmado o importante papel e a alta relevância da TI como um dos atores principais na indução de mudanças no segmento de TI nos sistemas estruturadores de governo.

A biologia identifica as espécies mais evoluídas pela capacidade de antever situações futuras. O processo decisório quase sempre é quem determina a diferença entre o sucesso e o fracasso ([7] Chris & Tara). Normalmente um processo de avaliação de tecnologia envolve diversas fontes especializadas de origem internas e externas, convergindo para decisões de aquisição. Tais decisões normalmente envolvem pareceres técnicos de especialistas, grupos de trabalho, e referências a lições aprendidas de projetos anteriores entre outros meios para garantir a sustentabilidade nesse processo. Dependendo do nível exigido de controle todo processo pode ser ainda objeto de designações formais, decisões de Diretoria entre outras práticas de regulação instituídas, até que se chega a um veredicto final onde a escolha é finalmente feita. Uma vez definida a tecnologia outro aspecto crítico, raramente mencionado em estudos e pesquisas, é o fator "resistência velada" que se instaura quando uma vez é definida a orientação e direcionamento. Surgem os "mártires silenciosos", que muitas das vezes estiveram ancorados na zona de conforto, e conspiram em silêncio incomodados com as mudanças de rumo. Reagem anônimos, na maioria das vezes, com um nível mínimo de racionalidade que não contribua para a efetiva melhoria da decisão; as considerações são semânticas inapropriadas que escondem na verdade um vazio de posicionamento cuja intenção é a manutenção do status-quo. Infelizmente esse cenário cria ruídos e desgastes com consumo de energia indesejável que só contribui para confirmar a hipótese de que "não é o presente contra o futuro, mas o ortodoxo contra o heterodoxo" ([8] Hamel).

[7] Chris Tara, Brady – Rules of the Game - Pearson Education Limited -2000.
[8] Leading Revolution – Hamel, G – Harvard Business School Press - 2000.

III - Dimensões do QST

O Quadrante Serpro de Tecnologia (QST) é um artefato simples para apoiar as atividades de gestão do ciclo de tecnologia projetado em duas dimensões: a primeira em X enfoca "produtividade" da tecnologia no uso interno, traduzida em indicadores definidos por segmento; a segunda em Y enfoca o grau de maturidade da tecnologia no ambiente externo e o seu potencial para atender as demandas de soluções de governo. A produtividade do segmento de sistemas, por exemplo, é calculada a partir do esforço medido em pontos de função realizados por homem/dia. A maturidade da tecnologia é obtida a partir de consolidação de check-Lists, envolvendo a rede de especialistas com notório conhecimento acerca da evolução das principais tendências da tecnologia no mercado, incluindo entidades de rede governo, SERPRO, academia entre outras parcerias importantes. A figura a seguir resume as dimensões citadas e respectivos quadrantes representados.

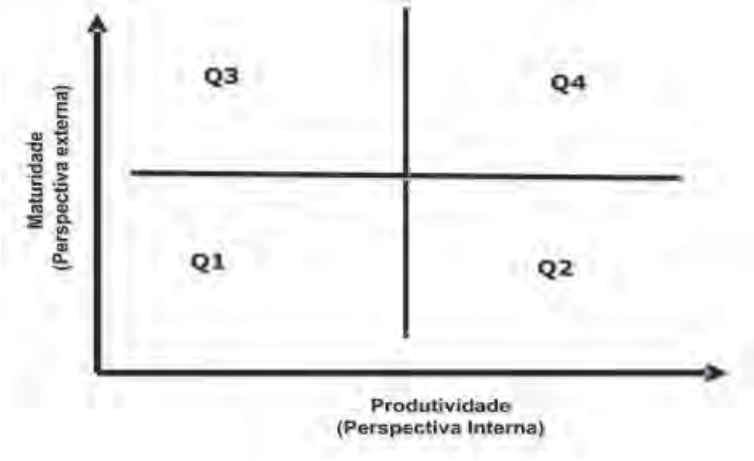


Figura 3.1 - Dimensões do Quadrante SERPRO de Tecnologia - QST

O contexto das decisões apoiadas pelo QST envolve os segmentos de Modelagem, Construção de Códigos e infraestrutura enfocando ferramentas de apoio ao desenvolvimento, Banco de Dados, Frameworks, Sistema Operacional, Servidores de aplicação, Data warehouse/BI,

Plataforma alta, Navegadores WEB e Componentes de software. As decisões focalizam Modelagem de Software (Engenharia de Requisitos, Analise de sistemas), Metodologias de Desenvolvimento de software e Processos de Desenvolvimento de software, Verificação, Validação e Testes, Frameworks, Programação e Manutenção de sistemas, Planejamento e Gerência de Projetos, Qualidade de software; Métricas de software, Padrões de Projeto e Medição. A seguir é apresentada uma amostra dos principais enfoques da avaliação QST, relacionando Riscos e Oportunidade no âmbito das perspectivas Interna e externa:

Tabela 3.1 – Riscos e Oportunidades de Tecnologia - Perspectiva Interna

Enfoque	Pior	Melhor
Conhecimento	Concentrado (Bolsões)	(Distribuído/disseminado)
Métodos de Medição	Inexistentes ou Confusos	Padronizados
Coleta de Dados	Manual/Caótica	Automatizada
Sistema de Informação	Precário	Integrado
Qualidade da Informação	Poluição	Precisão
Infra-estutura	Gambiarra	Necessária
Fluxo de Comunicação	Favores	Workflow
Organização de Dados	Anedótica	Elegante
Documentação Ref. Rápida	Papelório	Acesso Online
Gestão da Informação	Pró-forma	Tomada de Decisão
Auditoria de informação	Inexistente	Melhoria Continua
Treinamento	Inexistente	Um the Joe
Método de Gerenciamento	Pressão reativa	Gestão Proativa
Reclamação de Cliente	Coleção	Qualidade Assegurada
Resultados	Falados	Medidos
Capacidade de Reação	Zero	Imediata
Tecnologia	Religião	Utilidade
Produtividade	Abaixo da Média	Acima da Média

Tabela 3.2 - Riscos e Oportunidades de Tecnologia - Perspectiva Externa

Enfoque	Pior	Melhor
Padrões de Arquitetura	Fechados	Abertos
Plataforma Tecnológica	Dependência de Componentes	Independência de Componentes
Portabilidade	Baixa	Alta
Escalabilidade	Baixa	Alta
Base de Usuários	Irrelevante	Relevante
Suporte	Precário	Eficiente
Manutenabilidade	Baixa	Adequada
Treinamento	Loteria	Necessário
Documentação	Baixa	Alta
Comunidade de Usuários	Não possui	Ativa
Economicidade de Recursos	Baixa	Alta
Usabilibilidade	Baixa	Alta
Política comercial/Contrato	Predatória	Negociável
Padrões de Arquitetura	Fechados	Abertos

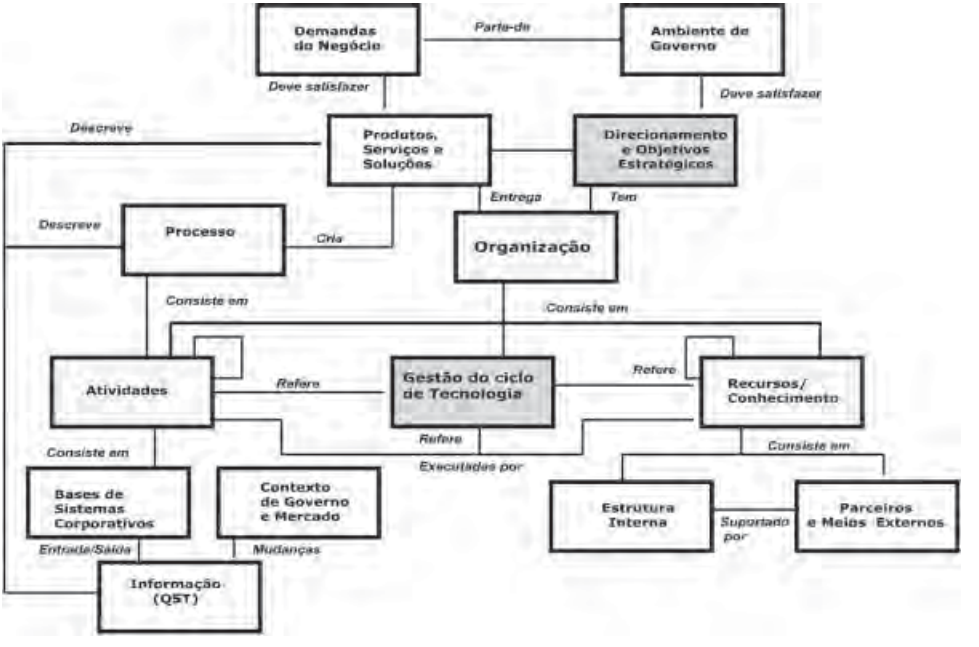
Atualmente o processo de avaliação de tecnologia no SERPRO está baseado em diretrizes institucionais e pareceres que envolvem as opiniões de técnicos com notório saber lotados nas várias áreas da empresa. O processo de construção dos pareceres, dependendo da complexidade e extensão no âmbito técnico é conduzido por grupos de trabalho por meio de designações formais de Diretoria; tais grupos fazem avaliações e constroem pareceres, que em última análise são base de sustentação dos investimentos em questão. A dependência de modelos mentais heterogêneos, incertezas e ranço genético a tecnologias como "religião", são exceções, mas quando recorrentes criam dificuldades que tem origem na relutância de seus atores em abrir mão de conhecimento já adquirido. As melhores práticas organizacionais para abordagem do problema demonstram que a formação de silos de resistência às inovações no mundo da computação são na realidade os maiores entraves

à inovação, inviabilizando a criação de uma base de conhecimento sustentável para apoiar decisões futuras.

IV - Arquitetura de Suporte ao QST e principais objetivos

O QST como ferramenta que apóia as decisões durante o ciclo de vida de tecnologia envolve, conforme já mencionado, decisões relacionadas aos seguintes temas: Engenharia de Requisitos, Análise e construção de Sistemas, Metodologias e Processos de Desenvolvimento de software, Frameworks, Ferramentas de Programação, Testes Verificação, Validação e Manutenção de sistemas, Qualidade de software; Métricas de software, Padrões de Projeto e Medição e Planejamento e Gerência de Projetos. As melhores práticas têm evidenciado a necessidade da construção de um processo formal, estruturado e consistente baseado no conhecimento explícito e transparente, envolvendo todas as partes interessadas direta e indiretamente nas decisões de tecnologia. A síntese da arquitetura para suporte ao QST é descrita na Figura 4.1 a seguir:

4.1 - Arquitetura de Suporte ao QST



O SERPRO é um importante agente indutor no mercado de soluções de Tecnologias de Informação de governo. O modelo praticado no QST gerencia o conhecimento sobre o ciclo de vida tecnologia de tecnologia e faz aproximação efetiva com parceiros externos, envolvendo academia e outros centros de pesquisa. Nesse contexto inserem-se os projetos desenvolvidos com Centros de pesquisa e Inovação. O QST, portanto, surge como o principal instrumento do Observatório de tecnologia ao dar visibilidade do painel de tecnologias em utilização no cenário atual e projetado, com vistas ao alcance dos seguintes objetivos:

- Contribuir para o aumento dos níveis de produtividade do desenvolvimento de sistemas com base em soluções inovadoras;
- Reduzir a carga de serviços de suporte e manutenção pela melhoria da qualidade e inovação nas soluções em serviços de TIC;
- Aumentar a portabilidade, interoperabilidade e integração dos sistemas;
- Prevenir problemas potenciais relacionados à segurança de dados e informações;
- Melhorar os níveis de escalabilidade do ambiente operacional;
- Maximizar o retorno dos investimentos em infraestrutura;
- Qualificar as análises de construção ou aquisição;
- Aumentar as oportunidades de utilização de soluções baseadas em arquiteturas abertas;
- Reduzir os riscos em investimentos nos ativos de TIC;
- Facilitar o planejamento de tecnologia em bases sustentáveis;
- Sustentar as decisões de investimento sem comprometer a coerência do direcionamento estratégico;
- Contribuir para redução da complexidade da infraestrutura de TIC.

V - Principais Insumos do Modelo de Avaliação QST

O principal insumo de avaliação do modelo QST é a informação derivada de fontes internas e externas que permitem a consolidação e tratamento dos dados no Quadrante. Essa informação tem origem nas áreas de Negócio, Desenvolvimento e Infraestrutura entre outros, como por exemplo, a Universidade SERPRO. Na perspectiva externa dos Clientes, incluem-se parceiros de solução, envolvendo rede governo, academia entre outros atores

que atuam no desenvolvimento de projetos de pesquisa e estudos aplicados. Todos de modo integrado contribuem para a formação de uma base sólida de informações traduzidas em Indicadores. Os principais insumos têm em vista:

- Produtividade do desenvolvimento de sistemas;
- Níveis de Serviços;
- Freqüência de incidentes, problemas e mudanças relativos a serviços de serviços de suporte e manutenção.
- Uso de Framework corporativo (Demoiselle) e participação em Comunidade;
- Reutilização de componentes;
- Incidentes e problemas relacionados à segurança de dados e informações;
- Indicadores de evolução e escalabilidade do ambiente operacional;
- Retorno dos investimentos em projetos indicadores de utilização de soluções inovadoras;
- Uso de arquiteturas abertas.

VI - Visão dos Processos do QST

Uma organização tem maiores chances de alcançar os objetivos do seu negócio se, e somente se, unir pessoas, tecnologia e outros recursos incluindo sistemas de informação em processos que se complementem ([9] Weske). O QST como ferramenta que apóia as decisões durante o ciclo de vida de tecnologia envolve decisões relacionadas aos seguintes temas : Engenharia de Requisitos, Análise e construção de Sistemas, Metodologias e Processos de Desenvolvimento de software, Frameworks, Ferramentas de Programação, Testes Verificação, Validação e Manutenção de sistemas, Qualidade de software; Métricas de software, Padrões de Projeto e Medição e Planejamento e Gerência de Projetos.

Os processos que suportam o Quadrante SERPRO de Tecnologia possuem referência no modelo de arquitetura corporativa proposto por Zachman/TOGAF[10]; esse modelo posiciona Tecnologia como um dos ativos

[9] Business Process Management – Concepts, Languages, architectures – Weske, M – Springer – Verlag Berlin Heidelberg – 2007.
[10] Zachman Framework – The Open Group Architecture Framework.

mais estratégicos para as organizações e considera: Visão de arquitetura do Negócio, Arquitetura de processos, Sistemas de informações, Tecnologia e Gestão de Mudanças e de Projetos, de forma integrada até a consolidação das soluções. As atividades e conexões que sustentam QST envolvem definições relativas à: Direcionamento Estratégico de Tecnologia, Catálogos de Configurações de componentes de hardware e Software, Check-List de Maturidade que traduzem a voz da rede de especialistas e parceiros de solução, em última análise entidades que formam a rede de sustentação do Quadrante SERPRO de Tecnologia para efetuar as avaliações por segmento de aplicação, conforme síntese apresentada na figura 6.1:

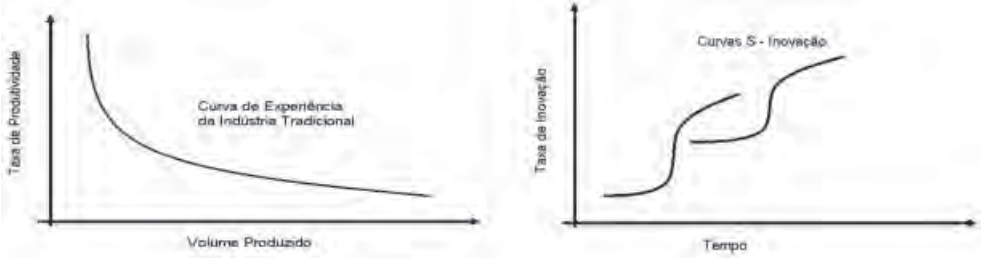
Figuras 6.1 – Componentes do QST

VII - Conclusão

Os desafios que o paradigma da computação em nuvem impõe às organizações no segmento de TIC exigem um posicionamento com respostas mais ágeis. Infraestrutura estável e ao mesmo tempo suficientemente flexível, capaz de suportar a heterogeneidade em multi-ambientes operacionais altamente compartilhados onde as palavras de ordem são qualidade e produtividade é premissa. Notadamente num ambiente cada vez mais orientado a serviço, como é o caso da computação nas nuvens, sob pena de

inviabilizarem seus respectivos modelos de negócio Ao alterar o ambiente físico, as inovações tecnológicas preparam os alicerces para eventuais mudanças na cultura organizacional ([11]Chris Tara, Brady), nessa perspectiva, cada vez mais as organizações de TIC devem se utilizar de métodos, processos e ferramentas alinhados e integrados para melhor produtividade nas suas escolhas. Com efeito, na indústria tradicional a curva de experiência é um fenômeno que torna possível identificar e prever melhorias de produtividade, à medida que os trabalhadores se tornavam mais hábeis nas respectivas tarefas, conforme curva de aprendizado exibida a seguir ([12] Hagel, Brown, Davison). Embora válida para indústria tradicional essa curva perde consistência num contexto altamente mutável como o segmento de Tecnologia da Informação; estudos recentes demonstram que uma curva S melhor expressa, com seus pontos de ruptura, a importância de buscar a inovação. Para tanto são mais que necessários processos, métodos e ferramentas como o QST que promovam a inovação e contribuam efetivamente para garantir a sustentabilidade de produtos e serviços.

Curvas de Experiência Vs Inovação



As ofertas e opções de tecnologia são as mais diversificadas ante a um novo paradigma da computação utilitária; esse novo momento torna a avaliação da base tecnológica existente e as decisões quanto o futuro da mesma um exercício de constante atualização e observação. Trata-se de uma questão estratégica de sobrevivência, sob pena de ser parte da história com verbos no passado. A questão é optar por seguir a tendência imposta, ou construir o

[11] Chris Tara, Brady – Rules of the Game – Pearson Education Limited – 2000.
[12] Harvard Business Review – John Hagel III, John Seely Brown – and Lang Davison Guest Edition – July 18, 2010 (3:34 PM).

seu caminho e fazer a sua história. A proposta do QST tem no seu DNA tem em vista oferecer avaliações elegantes que correlacionem notório saber com produtividade e maturidade no contexto de Gestão do Conhecimento em TIC. O seu principal produto é um painel de bordo para posicionamento do SERPRO, como sendo um dos principais indutores no segmento de soluções de TIC para governo.

Bibliografia

(2,4) Administração Inteligente – Futura – Fernandes, A. a2 Ed - 2003.

(3) Information Systems Concepts for Management, Loudon, McGraw Hill. Lucas, Henry C- 1990.

(5) Economics – (9th Edition) – Samuelson, Paul Antony- 1973.

(6) Management Information systems. Managing the Digital firm (8a.ed) Pearson education- 2004.

(7) Chris Tara, Brady - Rules of the Game - Pearson Education Limited - 2000.

(8) Leading Revolution - Hamel, G — Harvard Business School Press - 2000.

(9)Business Process Management – Concepts, Languages, architectures – Weske, M Springer - Verlag Berlin Heidelberg - 2007.

(10) Zachman Framework - The Open Group Architecture Framework.

(11) Chris Tara, Brady - Rules of the Game - Pearson Education Limited - 2000.

(12) Harvard Business Review - John Hagel III, John Seely Brown - and Lang Davison Guest Edition - July 18, 2010 (3:34 PM).

CONSEGI 2010

III Congresso Internacional Software Livre e Governo Eletrônico

APOIO

Jardim Botânico de Brasília

BRASIL ABC Agência Brasileira de Cooperação

COBRA TECNOLOGIA

Ministério da Cultura

Ministério da Educação

Ministério das Relações Exteriores

FUNDAÇÃO ALEXANDRE DE GUSMÃO

PATROCÍNIO BRONZE

PREVIDÊNCIA SOCIAL DATAPREV

CAIXA

PATROCÍNIO PRATA

BANCO DO BRASIL

PATROCÍNIO OURO

SEBRAE

PATROCÍNIO PLATINUM

BNDES

SERPRO Serviço Federal de Processamento de Dados

REALIZAÇÃO

ESAF Ministério da Fazenda

ISBN 85-7631-241-7

9 788576 312413