

Terceirização do desenvolvimento de *software* no Brasil e nos EUA



Carlos Alberto de Castilho Franco

Coordenador de desenvolvimento de *software* na Petrobras, e pós-graduado em Sistemas Integrados pela Universidade Federal do Rio de Janeiro (UFRJ - COPPE).



Rodrigo de Toledo

Professor adjunto na Universidade Federal do Rio de Janeiro (UFRJ - DCC,PPGI/IM) é graduado e mestre em computação pela PUC- Rio e PhD em computação pelo INRIA na França.

RESUMO

A terceirização dos processos de Tecnologia da Informação na Administração Pública Federal é uma realidade crescente desde o Decreto Lei 200 de 1967, no qual o legislador busca reduzir o tamanho da máquina do estado brasileiro, que naquele momento ocasionava problemas de burocratização, dificultando o crescimento do Estado. O objetivo deste artigo é analisar a evolução desse modelo de contratação para os órgãos públicos brasileiros, assim como sua aderência às práticas atuais de desenvolvimento de *software*. Nesse trabalho, fazemos uma pesquisa qualitativa do modelo de contratação e desenvolvimento de *software* do governo brasileiro comparando-o ao do governo norte-americano. A escolha dos Estados Unidos da América como referência para esse estudo se deve ao fato do governo desse país ser o maior consumidor individual de TI no mundo e possuir semelhanças políticas com o Brasil, como: regime presidencialista, capitalista e dimensão populacional.

Palavras-chave: Contratação. Lei. Lei 8.666. Desenvolvimento de *Software*. Governo Americano. Governo Brasileiro. Terceirização.

1. INTRODUÇÃO

A busca pela excelência administrativa, muitas vezes, esbarra em necessidades que extrapolam as com-



petências primárias das corporações. Alguns gestores ainda acreditam ser capazes de aumentar o lucro de suas empresas, realizando internamente todas as etapas dos serviços necessários ao funcionamento. Porém, nessa postura, corre-se o risco de perder o foco do negócio e acabar tendo prejuízos ou mesmo chegar à insolvência (Cury, 2000; Oliveira, 2004). A evolução da área de tecnologia da informação tem sido muito dinâmica, fazendo com que novas tecnologias surjam a cada instante, provocando uma verdadeira corrida, entre as empresas, na busca de manterem-se inseridas neste mercado (Boehm, 2006, p.12-29).

O custo de se acompanhar a evolução é muito elevado, e isso faz com que muitas organizações deleguem as atividades de tecnologia da informação para empresas terceirizadas a fim de se manterem atualizadas tecnologicamente. Sendo assim, a prática de contratar uma organização externa para desenvolver um projeto (*outsourcing*) em vez de desenvolver na própria matriz da organização contratante (*inhouse*), passa a ser uma ferramenta de diferencial competitivo nas organizações.

Conforme Barthélemy e DiRomualdo, as razões pelas quais empresas optam por terceirizar têm sido bem documentadas, incluindo: custo reduzido, melhor desempenho e acesso a mercados mais amplos de trabalho (Barthélemy et al. 2001, p. 60-69; DiRomualdo et al. 1998, p. 67-80). No entanto, a contratação de *software* envolve complexos elementos jurídicos, econômicos, gerenciais e tecnológicos (Whang 1992, p. 307-324,

Lacity et al. 1996, p. 13-25), que devem ser cuidadosamente avaliados antes de uma decisão eficaz acerca de se optar por esse caminho. Delegar a construção de um *software* para terceiros não se tem mostrado trivial, apesar de pesarem mitos como o de que a subcontratação é sempre a melhor escolha por representar cortes nos custos e pessoal (Hernandes, 2007).

Entre as empresas do governo brasileiro, o movimento de terceirização também é real (Hazan, 2010, p.1). A busca pela melhoria de processos utilizando ferramentas de tecnologia da informação tem sido cada vez mais uma realidade na administração pública, a qual procura na TI uma ferramenta para aumentar a capacidade de atendimento às necessidades do cidadão. No entanto, essa preocupação não é recente, conforme constatamos no Decreto-lei de número 200 de 1967, que dispõe sobre a organização da administração pública federal. Nesse decreto, o legislador estabelece diretrizes para a Reforma Administrativa e no capítulo terceiro, dispõe sobre a descentralização na administração pública:

Art. 10. A execução das atividades da Administração Federal deverá ser amplamente descentralizada. [...]

§ 7º Para melhor desincumbir-se das tarefas de planejamento, coordenação, supervisão e controle e com o objetivo de impedir o crescimento

desmesurado da máquina administrativa, a Administração procurará desobrigar-se da realização material de tarefas executivas, recorrendo, sempre que possível, à execução indireta, mediante contrato, desde que exista, na área, iniciativa privada suficientemente desenvolvida e capacitada a desempenhar os encargos de execução. (Decreto-lei 200, Brasil, 1967)

A terceirização dentro das empresas estatais é uma realidade. Mas, será que existe uma gestão que possibilite a utilização desta ferramenta em sua plenitude e que retorne valor gerando ganhos à sociedade? As leis que regulam as contratações permitem que o processo de construção de *software* possa ser pleno, possibilitando contratações transparentes e produtivas? Este artigo tem como objetivo fazer um estudo etnológico, o qual analisa conhecimentos, crenças, leis, costumes ou hábitos adquiridos pelo homem como membro de uma sociedade (Edward Tylor, 1871), analisando as leis que regulam o processo de contratação de *software* dentro da administração pública brasileira e americana. Nossa proposta apresenta pontos positivos e negativos através de práticas reconhecidas pelo mercado nacional e internacional. Este artigo também pretende propor mudanças que possam agregar valor para o processo de contratação na administração pública brasileira.

2. TERCEIRIZAÇÃO

Acompanhar a evolução da tecnologia da informação não tem sido uma tarefa trivial e, nesse contexto, as empresas públicas ou privadas necessitam dispor de ferramentas que possam auxiliá-las nesse desafio. Sendo assim, a contratação de mão de obra terceirizada passa a ser uma ferramenta muito utilizada por essas empresas (Barthélemy et al. 2001, p. 60-69; DiRomualdo et al. 1998, p. 67-80). Segundo Vazquez, podem existir vários níveis de terceirização: da codificação de *software* até o *outsourcing* de toda a equipe de sistemas (Vazquez et al., 2004).

Outros autores identificam características que justificam esta escolha como uma desobrigação do contratante em relação ao produto final (Hernandes, 2007). Porém, Pinheiro, em seu artigo, afirma que uma organização imatura no seu processo de aquisição tem o mesmo potencial de fracasso daquela em que o desenvolvimento ainda não possui um processo consistente, e que o maior problema detectado nas aquisições de *software* refere-se às práticas de gestão dentro das empresas (Pinheiro, 2006).

Guerra também defende a maturidade nos processos internos das organizações para que se possa gerenciar um contrato de desenvolvimento de *software* de forma a se alcançar todos os resultados esperados. Ele define maturidade como sendo a habilidade para gerenciar o desenvolvimento de *software* e os processos de manutenção, pois numa organização madura, o processo de *software* é cuidadosamente comunicado à equipe já existente e aos novos contratados, e as atividades de trabalho são realizadas de acordo com o planejamento, monitoração da qualidade dos produtos e a satisfação dos clientes (Guerra, et al., 2004, p. 232).

2.1 DIFICULDADES

Sabe-se que características próprias do seguimento de informática podem dificultar as empresas e órgãos do governo no momento da aquisição ou contratação de *software*. Hernandez destaca quatro problemas: (i) a mudança de requisitos, pois a demora em se contratar dentro da administração pública faz com que os requisitos encontrados no início do processo sofram mudanças; (ii) rotatividade de mão de obra, o mercado de tecnologia da informação tem como característica a troca de profissionais entre as empresas do ramo; (iii) perda do conhecimento por ocasião do término de contrato, os contratos entre empresas públicas, muitas vezes, possuem tempo limitado de duração, obrigando a utilização de técnicas de gerência de conhecimentos, na tentativa de garantir que o conhecimento do produto desenvolvido permaneça na empresa contratante; (iv) falta de conhecimento da integração dos processos de trabalho, pois muitas organizações licitam *software* sem o conhecimento amplo dos seus próprios processos de trabalho e das correlações existentes entre eles (Hernandes, 2007).

Dos problemas apresentados, destaca-se a mudança de requisitos durante o processo de contratação e desenvolvimento, pois, conforme Hernandez, o escopo de um *software* geralmente é definido ao início do projeto, mas dificilmente ficará imutável até o seu final (Hernandes, 2007). Segundo Watts Humphrey: “Em um sistema, os requisitos não serão completamente conhecidos até que os usuários o tenham usado” (Humphrey, 1995). Peter Wegner afirma ainda que “não é possível especificar completamente um sistema interativo” (Wegner, 1997). No entanto, há no mercado a ilusão de que é possível desenvolver um *software* sem que haja modificações e, por este motivo, são realizados contratos de escopo fixo, em que o cliente acredita

que contará com custo, prazo e escopo previsíveis. A Engenharia de Requisitos reconhece que os requisitos não permanecem estáticos até a conclusão do projeto de *software*. Diversos fatores fazem com que eles evoluam desde a sua concepção até a sua entrega (Kotonya, 1998). Destaca-se ainda Hadar Ziv, quando afirma que “a incerteza é inerente e inevitável em desenvolvimento de *software*, processos e produtos” (Ziv, 1997).

3. LEI BRASILEIRA

Mesmo tendo necessidade de se capacitar para uma gestão mais efetiva, as empresas do estado são regidas por normas. As contratações em empresas públicas brasileiras são regulamentadas por diversas leis, decretos ou instruções normativas. Dentre elas, destaca-se a lei 8.666 de 1993, conhecida também como a “lei das licitações”, que estabelece:

Art. 1º - Esta Lei estabelece normas gerais sobre licitações e contratos administrativos pertinentes a obras, serviços, inclusive de publicidade, compras, alienações e locações no âmbito dos poderes da União, dos Estados, do Distrito Federal e dos Municípios. (Brasil, 1993).

Em uma primeira análise dessa lei pode-se observar que o legislador buscou garantir que os recursos públicos fossem bem aplicados por ocasião de contratações, compras, alienações ou locações. A lei se refere especificamente à área de informática nos seguintes trechos:



Art. 24. É dispensável a licitação [...]

XVI - [...] para prestação de serviços de informática a pessoa jurídica de direito público interno, por órgãos ou entidades que integrem a Administração Pública, criados para esse fim específico;

Art. 45. O julgamento das propostas será objetivo, devendo a Comissão de licitação ou o responsável pelo convite realizá-lo em conformidade com os tipos de licitação, os critérios previamente estabelecidos no ato convocatório e de acordo com os fatores exclusivamente nele referidos, de maneira a possibilitar sua aferição pelos licitantes e pelos órgãos de controle. [...]

§ 4º Para contratação de bens e serviços de informática, a administração observará o disposto no art. 3º da Lei nº 8.248, de 23 de outubro de 1991, levando em conta os fatores especificados em seu parágrafo 2º e adotando obrigatoriamente o tipo de licitação “técnica e preço”, permitido o emprego de outro tipo de licitação nos casos indicados em decreto do Poder Executivo. (Lei 8.666, Brasil, 1993)

A seguir, analisar-se-á o contexto na época da criação da lei e o momento atual.

3.1 CONTEXTO DA LEI 8.666/1993

A informática na década de oitenta teve seu foco em processos sequenciais. Dessa forma, os sistemas de computação seguiam uma abordagem de desenvolvimento em cascata que tinha como característica principal a necessidade do encerramento da etapa anterior para se iniciar a próxima. Esse modelo dominou a forma de desenvolvimento de *software* até o início dos anos 90. No entanto, autores como Frederich Brooks alertavam para problemas gerados ao se adotar uma visão sequencial de tarefas (Brooks, 1987, p. 10-19). Tom Gilb desencorajava o uso do modelo cascata para a construção de grandes *softwares*, acreditando ser mais produtivo o desenvolvimento incremental, pois segundo ele, apresenta menores riscos e maiores possibilidades de sucesso (Gilb, 1999).

No entanto, a lei de licitações determina que a contratação de desenvolvimento de *software*, deverá ter característica semelhante à de um projeto de engenharia, em que a etapa de planejamento é feita sabendo-se todas as necessidades para o desenvolvimento do pro-

jeto. Identifica-se esta característica na lei 8.666/1993, quando o legislador determina na Seção III - Das Obras e Serviços, que

Art. 7º - As licitações para a execução de obras e para a prestação de serviços obedecerão ao disposto neste artigo e, em particular, à seguinte sequência:

- I - projeto básico;
- II - projeto executivo;
- III - execução das obras e serviços.

§ 1º - A execução de cada etapa será obrigatoriamente precedida da conclusão e aprovação, pela autoridade competente, dos trabalhos relativos às etapas anteriores, à exceção do projeto executivo, o qual poderá ser desenvolvido concomitantemente com a execução das obras e serviços, desde que também autorizado pela Administração. (lei 8.666, Brasil, 1993).

A necessidade de se obter todas as características de um *software* ainda na fase de planejamento é evidenciada ao se definir o conceito de Projeto Básico, encontrado na própria lei:

Projeto Básico – conjunto de elementos necessários e suficientes, com nível de precisão adequado, para caracterizar a obra ou serviço, ou complexo de obras ou serviços objeto da licitação, elaborado com base nas indicações dos estudos técnicos preliminares, que assegurem a viabilidade técnica e o adequado tratamento do impacto ambiental do empreendimento, e que possibilite a avaliação do custo da obra e a definição dos métodos e do prazo de execução, devendo conter os seguintes elementos:

[...] f) orçamento detalhado do custo global da obra, fundamentado em quantitativos de serviços e fornecimentos propriamente avaliados; (lei 8.666, Brasil, 1993).

Em seu item f, a lei apresenta a crença do legislador de que um projeto a ser licitado, possa ter o conhecimento total das necessidades do cliente ainda no momento do planejamento. Essa crença demonstra a defasagem com a realidade da área de tecnologia da informação por parte do legislador brasileiro, conforme Frederich Brooks em “*No Silver Bullet: Essence and*

Accidents of Software Engineering”, a ideia de especificar totalmente um *software* antes do início da implementação é impossível (Brooks, 1987, p. 10-19).

3.2 ADEQUAÇÃO DA LEI

Dados do *Standish Group* de 1995, utilizando como base 8.380 projetos de *software*, mostrou que apenas 16% desses projetos foram entregues dentro dos prazos e custos acordados, e com todas as funcionalidades especificadas, que 32% foram cancelados antes mesmo de terem sido completados e que 52% foram entregues com prazos e custos maiores ou com falta de funcionalidades especificadas no início do projeto. Segundo a mesma pesquisa, nos casos em que os projetos respeitaram os limites de prazo e custo, observaram-se características de baixa qualidade no desenvolvimento, acarretando um número elevado de manutenção corretiva. Como resultado desta análise, a pesquisa identificou que o motivo principal das falhas encontradas foi a utilização do Modelo Clássico de desenvolvimento. A pesquisa, em sua conclusão final, recomenda que os *softwares* sejam feitos de maneira incremental (Standish Group, 1995).

Na busca de regulamentar a legislação brasileira com relação à contratação de bens de serviço de informática e automação, a Presidência da República, por meio da Casa Civil, publicou o Decreto número 7.174 de 12 de maio de 2010, que regulamenta os processos de contratação de tecnologia da informação por entidades do governo brasileiro; busca regulamentar as disposições da Lei nº 8.248, de 23 de outubro de 1991, revogar o Decreto nº 1.070, de 2 de março de 1994 e alterar o Decreto nº 3.555, de 8 de agosto de 2000. Em uma primeira análise, observa-se a existência de um detalhamento maior aos temas que tratam da aquisição de bens (*Hardware*), em detrimento do tema sobre aquisição de serviços (*Software*), além de uma abordagem de desenvolvimento em cascata, assim como a necessidade de se levantar todas as características do projeto, ainda na fase de planejamento, permanecem como obrigações neste decreto, conforme podemos constatar no segundo artigo:

Art. 2º A aquisição de bens e serviços de tecnologia da informação e automação deverá ser precedida da elaboração de planejamento da contratação, incluindo projeto básico ou termo de referência contendo as especificações do objeto a ser contratado [...] (Decreto 7.174, Brasil, 2010).

Percebe-se o distanciamento do legislador brasileiro da realidade do desenvolvimento de *software*, pois desde 2001 com o advento do Manifesto Ágil, tem havido uma verdadeira revolução na maneira de se construir solução de *software*, que possui como pilares da sua metodologia: indivíduos e interações em vez de processos e ferramentas; *software* executável no lugar de documentação; colaboração do cliente ao contrário de negociação de contratos; respostas rápidas a mudanças em vez de seguir planos.

4. LEI AMERICANA

A garantia de transparência dentro de um processo licitatório em organismos públicos é a principal justificativa do legislador na hora de solicitar um planejamento completo do serviço a ser executado antes do início. Mas, será que não existe a possibilidade de se ter garantias de lisura e transparência dentro de órgãos e empresas do governo, ao mesmo tempo em que se possa aplicar boas práticas de construção de *software*? O governo americano acenou para esta preocupação e publicou a lei *Clinger-Cohen* de 10 de fevereiro de 1996, proposta pelo deputado William Clinger e o senador William Cohen, a qual revogou a Lei de Brooks de 1965, que havia emendado a lei *Federal Property and Administrative Services* de 1949.

Em fevereiro de 1996, o Congresso americano aprovou a Lei *Clinger-Cohen* com o objetivo de reformar e melhorar a maneira como as agências federais adquirem e gerenciam os recursos de TI, sendo o ponto central para a implementação destas reformas, a necessidade de estabelecer uma liderança de TI dentro de cada agência do governo. Essa lei descentralizou a autoridade e responsabilidade para a aquisição de recursos de tecnologia da informação, além de unir as leis de Reforma da Gestão da Tecnologia da Informação e a lei de Reforma da Aquisição Federal. A Lei *Clinger-Cohen*, de 1996 criou a função de CIO (*Chief Information Office*), procurando atentar mais para os resultados que poderia alcançar por meio de investimentos em TI, numa época em que a ideia de fazer investimentos em TI em larga escala era relativamente nova. A lei *Clinger-Cohen* enfatizou o rigor e a estrutura na forma de como as agências devem selecionar e gerenciar projetos de TI. Por ela, o CIO passa a ser o responsável por estabelecer visibilidade e gestão necessárias para a realização das disposições específicas da lei, possuindo como atribuições básicas: ajudar a controlar os riscos de desenvolvimento de sistemas; gerir melhor os gastos com tecnologia e conseguir al-

cançar melhorias reais e mensuráveis no desempenho da agência.

Por meio da lei *Clinger-Cohen*, o governo americano apresenta as diretrizes para o desenvolvimento de *software*, como: planejar, controlar investimento para maximizar valor, avaliar/gerir riscos nas aquisições de tecnologia da informação e a contratar grandes sistemas de tecnologia da informação de maneira modular, conforme descrito na própria lei:

[...] Utilizar contratação modular na medida do possível, para uma aquisição de um sistema maior de tecnologia da informação [...]. (*Clinger-Cohen Act*, USA, 1996).

A Seção 4.1, a seguir, explica o contexto temporal da lei americana enquanto a Seção 4.2 retoma a questão da modularização.

4.1 CONTEXTO DA LEI CLINGER-COHEN

Antes de se analisar o conteúdo da lei *Clinger-Cohen* de 1996, deve-se entender o contexto em que se encontrava o legislador no momento de sua criação: Marc Andreessen em seu artigo, “Por que o *Software* está consumindo o Mundo?” (Andreessen, 2011), afirma que em 2013, o *software* começaria a consumir os contratos dos órgãos federais americanos, gerando cortes de postos de trabalho devido a uma redução significativa no orçamento federal. Segundo Rockwell Collins, a pressão orçamentária e a capacidade de recursos afetam diretamente a contratação no governo americano, e resume o momento atual de pressões orçamentárias como: “incerteza sem precedentes para todas as empresas que apoiam o departamento de defesa”. A pressão do orçamento obriga as empresas a buscarem redução de custos, tanto com demissões ou por meio da utilização de soluções de *software* inteligentes, fazendo com que o consumo descrito por Marc Andreessen seja considerado em todas as contratações do governo, principalmente as de tecnologia da informação.

Apesar de o artigo de Collins afirmar a inexistência, em anos anteriores, da pressão do orçamento nas empresas que trabalhavam para o Departamento de Defesa americano, a disciplina orçamentária e o gasto em tecnologia da informação não é um fato novo para as demais empresas em todo o mundo, conforme afirma Augustine, que a realidade dos gerentes de projeto tem sido, durante muito tempo, lidar com redução do orçamento para seus projetos, bem como a diminuição das

equipes (Augustine, 2005, p. 70–74), por sua vez Earl, defende que a terceirização em tecnologia da informação originou-se da necessidade de cortar custos e reduzir pessoal devido às crescentes despesas com o setor de informática (Earl, 1998), enquanto Wang também elenca entre os quatro motivos que levam uma organização a terceirizar a área de informática à diminuição do custo da produção (Wang, 1995, p. 24-50).

4.2 ADEQUAÇÃO DA LEI E A MODULARIZAÇÃO

Para auxiliar as empresas do governo e autarquias ao cumprimento da lei Clinger-Cohen, a Casa Branca em 14 de junho de 2012, editou o Guia de Contratação para Suportar o Desenvolvimento Modular, que busca orientar os CIOs na aplicação dessa lei, buscando mais benefícios para o governo e alertando aos riscos existentes na hora de se contratar *software*. Conforme pode-se constatar, há uma preocupação do governo americano em garantir que recursos aplicados na solução possam efetivamente retornar valor ao contribuinte:

... o desenvolvimento responsável necessita de um detalhamento completo de requisitos antes que o trabalho possa começar. Embora uma hipótese aparentemente razoável de evidência, prática e experiência no setor privado tem mostrado que grandes e complexas implementações de TI se depa-ram frequentemente com custos e atrasos em seus cronogramas, pois um meticuloso processo de coleta de requisitos muito frequentemente leva anos para ser concluído. [...] O Governo aumenta seus riscos de investimento nestas situações por que: (1) as soluções de TI podem não ser mais necessárias ou prioritárias depois de levantados os requisitos; (2) recursos elevados são alocados para soluções ultrapassadas sem qualquer retorno sobre os investimentos, ou (3) as empresas do governo podem ter cortes de recursos antes da entrega final do *software* (CGSMD, 2012).

Em uma análise mais profunda do Guia, identificamos ainda características reconhecidas na área de tecnologia da informação como ágeis:

garantir a satisfação do consumidor entregando rapidamente e continuamente *softwares* funcionais; *softwares* funcionais são entregues frequentemente (semanas, ao invés de meses); *softwares* funcionais são a principal medida de progresso do projeto; até mesmo mudanças tardias de escopo no projeto são bem-vindas; cooperação constante entre pessoas que entendem do ‘negócio’ e desenvolvedores; projetos surgem através de indivíduos motivados, e que deve existir uma relação de confiança; design do *software* deve prezar pela excelência técnica; simplicidade; rápida adaptação às mudanças; indivíduos e interações mais do que processos e ferramentas; *software* funcional mais do que documentação extensa; colaboração com clientes mais do que negociação de contratos; responder a mudanças mais do que seguir um plano (Agile Manifesto, 2001).

A postura apresentada pelo governo americano é aderente ao mercado, conforme Charette que em seu artigo, compara métodos ágeis com as metodologias tradicionais pesadas e demonstra que projetos que utilizavam metodologia ágil obtiveram melhores resultados de prazos, custos e qualidade (Charette, 2001).

5. COMPARAÇÃO ENTRE AS LEIS

Numa primeira análise entre a lei americana *Clinger-Cohen* de 1996 e a lei de licitações brasileira 8.666 de 1993, encontra-se em ambas a necessidade de uma disciplina orçamentária e de cumprimento do plano estratégico do governo federal de ambos os países, o que não foge as diretrizes orçamentárias existentes em todos os governos. Entretanto, a maior diferença identificada está na preocupação do legislador americano



ao determinar que grandes projetos de tecnologia da informação devam ser entregues de maneira modular, acreditando que esta prática permite um menor risco ao projeto, possibilitando um retorno maior e mais imediato ao investimento feito na solução. Conforme o Guia de Contratação para Suportar o Desenvolvimento Modular:

pequenas e rápidas entregas feitas pela área de tecnologia da informação permite ao contribuinte receber um produto utilizável antes do término do projeto, permitindo que, caso haja alguma mudança em sua necessidade, à mesma possa ser efetivada sem se perder tempo ou recurso no desenvolvendo da nova solução, além disso, essa abordagem permite que: “a área de TI esteja sempre inserida nas novas soluções do mercado” (CGSMD, 2012).

Ao voltar para a lei 8.666/1993 não se encontram referências que proibam o desenvolvimento modular por parte das empresas brasileiras. No entanto, a necessidade do levantamento total da solução antes do início da construção, e a obrigatoriedade de se iniciar uma etapa somente quando terminada a anterior, faz com que a contratação por módulos da solução seja muito dificultada, além da falta de regulamentação que incentive esta prática dentro do governo brasileiro.

Será que se pode concluir então que a legislação americana está mais aderente às modernas práticas de desenvolvimento de *software* conhecidas no mundo, possibilitando às empresas do governo americano um maior aproveitamento dos recursos investidos no desenvolvimento de solução de *software*? A Lei *Clinger-Cohen* conseguiu atender a necessidade de uma boa gestão de recursos do governo, ou seja: o que se está comprando é a coisa certa e pelo motivo certo? Como se sabe?

Em 2006, Tom Davis, Presidente da Comissão de Reforma do Governo Americano, afirmou que:

Desde a passagem da lei *Clinger-Cohen* o governo começou a tomar uma abordagem holística para a tecnologia da informação, utilizando para resolver problemas de negócios e alcançar melhorias no seu desempenho. Temos um longo caminho a percorrer, mas o gigante que é o governo federal está bem no seu caminho para o século 21.

Segundo Andrués

O *Clinger-Cohen Act* e as leis e regulamentos que se sucederam são todos parte de um impulso elementar para impor limites, exigir resultados organizacionais e estabelecer uma base de prestação de contas.

A imposição de estrutura, no entanto, é apenas parte do que será necessário para fazer *Clinger-Cohen* o agente de mudança que foi projetado para ser. O que é mais necessário são boas notícias, e os contadores de histórias devem ser os *CIOs* federais. Devem cultivar um ambiente em que os resultados podem ser comprovados, exaltado e exposto. Eles devem criar condições para alcançar melhorias definitivas por intermédio da tecnologia da informação.

6. CONCLUSÃO

A lei deve ser honesta, justa e estatuída para a utilidade comum dos cidadãos e não para benefício particular, adequada à natureza e aos costumes, conveniente no tempo, necessária e proveitosa e clara, sem obscuridade que provoque dúvida.” (Santo Isidoro de Sevilha)

A lei é um resultado da realidade social. Ela emana da sociedade, por seus instrumentos e instituições destinados a formular o Direito, refletindo o que a sociedade tem como objetivos, bem como suas crenças e valorações, o complexo de seus conceitos éticos e finalísticos. (Herkenhoff, 1993).

Desta forma, não se pode analisar as leis sem se aprofundar em seus círculos de convivência como: família, escola, clubes, igrejas, trabalho e outros. Ao se considerar essa afirmativa, entende-se que leis, usos, tradições e costumes, são flexíveis, dependendo do tempo, da crença ou do local, diferindo entre estados, países e continentes. Num determinado período de tempo ou espaço, algo pode ser considerado absurdo, enquanto que em outro ser certo ou aceito. Não se analisa uma lei sem antes entender a época ou local em que foi promulgada, ou as verdades nas quais ela se baseou. Associa-se a lei ao contexto de sua criação, levando em consideração os componentes sociais como família, sociedade, cultura e crenças.

Ao se propor a analisar as diferenças entre as leis de contratação de *software* entre o Brasil e os Esta-

dos Unidos, depara-se com essa realidade, em que se identifica a evolução do processo de criação de *software* nas empresas públicas desses dois países. Observa-se a evolução das leis para atenderem as diversas formas de se construir uma solução de Tecnologia da Informação. Passa-se por regulamentações que se preocupavam mais com *hardware* para outras, mais atuais, focadas em *software*. Atualmente, observa-se uma preocupação maior do legislador, principalmente o americano, em garantir que o *software* entregue atenda às necessidades requeridas e que também retorne maior ganho para a sociedade.

Para isso, identifica-se no trabalho, que os pilares de sucesso para um projeto de criação de *software*: prazo, custo e qualidade, possuem uma melhora quando aplicada a metodologia ágil, faltando apenas o quarto pilar, escopo que, como definido dentro dessa metodologia, deve ser flexível buscando as necessidades do cliente. Apesar da metodologia ágil ainda se encontrar no início de 2001, ela já apresentava resultados consistentes, como descritos por Charette, em que se comparam os métodos ágeis com as metodologias tradicionais e mostra-se que os projetos que utilizaram esta metodologia obtiveram melhores resultados nas disciplinas de cumprimento de prazos, custos e padrões de qualidade. Nesse mesmo estudo mostra que a quantidade de projetos e de equipes utilizando metodologias ágeis cresce no meio de TI (Charette, 2001). Metodologias ágeis possuem como principal característica a aceitação à mudança nos requisitos durante a construção do *software*. Ela acredita que desta forma é possível realmente entregar ao cliente o produto que ele necessita, não rejeitando os processos e ferramentas, a documentação, a negociação de contratos ou o planejamento. Simplesmente, os métodos ágeis consideram sua importância como secundária quando comparado com os indivíduos e interações, com o *software* executável, com a colaboração do cliente e com respostas rápidas a mudanças e alterações (Manifesto Ágil, 2001).

Dessa forma, ao contrário do determinado pela lei brasileira, o escopo de um projeto de *software* deve ser flexível, para que se possa garantir que os demais pilares possam ser cumpridos dentro da expectativa do consumidor, garantindo custo, prazo e qualidade definidos. Mas para isso existe a necessidade de uma revisão na lei 8.666 de 1993 em que se deve flexibilizar o conceito do Projeto Básico, ou ainda, a exemplo da legislação americana, sugere-se ao Ministério da

Ciência e Tecnologia a publicação de um guia para contratação de solução de *software* pelas empresas públicas e autarquias no governo brasileiro, que considere a criação de *software* de forma modular e aderente à metodologia de desenvolvimento ágil, pois se acredita que atualmente a abordagem ágil possa ser a grande ferramenta que possibilite ao governo brasileiro alcançar os objetivos.

REFERÊNCIAS

AGILE MANIFESTO 2001. *Manifesto for agile software development*. Disponível em: <http://www.agilemanifesto.org>. Acessado em: 12 de março de 2013

ANDERSEN, M., *Why software is eating the world*. *The Wall Street Journal*. <http://online.wsj.com/article/SB10001424053111903480904576512250915629460.html> Acessado em: 21 de março de 2013

AUGUSTINE, S., "Adotando métodos de entrega Ágil", *Revista PMI* n. 11, p. 70-74. 2006

BOEHM B., "A View of 20th- and 21st-Century Software Engineering.", University of Southern California. P. 12-29. 2006.

BARTHELEMY, Jérôme. *The hidden cost of IT outsourcing*. *Sloan Management Review*, Cambridge, p. 60-69, 2001.

BRASIL. Decreto Lei 200, de 25 de Fevereiro de 1967. *Diário Oficial da União*, Brasília, DF, Seção 1 - Suplemento - 27 de fevereiro de 1967, Página 4

BRASIL. Lei nº 8.666, de 21 de junho de 1993. *Diário Oficial da União*, Brasília, DF, Seção 1 - 22 de junho de 1993, Página 8.269.

BRASIL. Decreto Lei 7.174, de 12 de Maio de 2010. *Diário Oficial da União*, Brasília, DF, Seção 1 - 13 de maio de 2010, Página 1.

BROOKS, F. *No Silver Bullet: Essence and Accidents of Software Engineering*. Proc. IFIP, IEEE CS Press, pp. 1069-1076; reprinted in *IEEE Computer*, p. 10-19, Apr. 1987.

CHARETTE, R. "Fair Fight? Agile Versus Heavy Methodologies", Cutter Consortium E-project Management Advisory Service, 2001.

CLINGER-COHEN ACT, Title 40, *Public Law* 104-106, Division E. 10 de fevereiro de 1996

CGSMD, *Contracting Guidance to Support Modular Development*, 2012, White House, USA, <http://www.whitehouse.gov/sites/default/files/omb/procurement/guidance/modular-approaches-for-information-technology.pdf>. Acessado em 11 de janeiro de 2013

CURY, Antonio. "Organização e métodos: uma visão holística." 7ª ed. São Paulo: Atlas, 2000.

DiROMUALDO, Anthony; Gurbaxani, Vijav. "Strategic intent for IT outsourcing". *Sloan Management Review*, Cambridge, p. 67-80, 1998.

EARL, Michael J. "Deve-se terceirizar a informática?" *HSM Management*, São Paulo, v. 1, n. 6, p. 126-132, 1998.

GILB, T. *Principles of Software Engineering Management*. Addison-Wesley, 1999.

GUERRA, Ana Cervigni; ALVES, Ângela Maria. *Aquisição de Produtos e Serviços de Software*. Rio de Janeiro: Elsevier, p. 232. 2004.

HAZAN C., "Como evitar armadilhas em contratos de fábricas de software" *Revista do TCU* 117, p. 1 Jan/Abr 2010.

HERKENHOFF, João Baptista. *Direito e utopia*, p. 17-20. 1993.

HERNANDES, Carlos Alberto Mamede. *Mitos sobre terceirização de serviços de Tecnologia da Informação*. Brasília: s.n., 2007.

HUMPHREY, Watts, *A Discipline of Software Engineering*. Reading, MA: Addison Wesley. ISBN: 0-201-54610-8, 1995.

IMPROVE IT. Contrato de escopo negociável. Disponível em: <http://www.improveit.com.br/xp/praticas/contrato>. Acesso em: 17 de abril de 2013.

KOTONYA, G.; SOMMERVILLE, I. *Requirements engineering: processes and techniques*. New York : J. Wiley, 1998.

LACITY, M. C. and R. Hirschheim, "The Value of Selective IT Sourcing," *Sloan Management Rev.*, p. 13-25, 1996.

OLIVEIRA, Djalma R. *Sistemas, organização e métodos: uma abordagem gerencial*. 14 ed. São Paulo: Atlas, 2004.

PINHEIRO, Lucia Nigro Pereira. *Um processo de apoio à gerência de aquisição de produtos e serviços de software*. Rio de Janeiro: s.n., 2006.

STANDISH GROUP, Chaos report, 586 *Olde Kings Highway*. Dennis, MA 02638, USA, 1995.

TYLOR E., "Primitive Culture" 1871 - fonte Wikipedia, acessado em 16 de abril de 2013.

VAZQUEZ, Carlos Eduardo; SIMÕES, Guilherme Siqueira; ALBERT, Renato Machado. *Gestão de contratos de desenvolvimento de software com a Análise de Pontos de Função*. Belo Horizonte, MG : SUCESU, 2004.

WANG, E. and T. Barron, "Controlling Information System Departments in the Presence of Cost Information Asymmetry" *Information Systems Res.*, 6, 1, p. 24-50, 1995.

WEGNER, P. "Why interaction is more powerful than algorithms." *Commun. ACM* (May), 80-91. 1997.

WHANG, S., "Contracting or Software Development", p. 307-324, 1992.

ZIV H. and D. Richardson, "The Uncertainty Principle in Software Engineering," in submitted to *Proceedings of the 19th International Conference on Software Engineering (ICSE'97)*, 1997.