

BOINC - Um Sistema para Computação Voluntária

Jeferson Librelotto Prevedello
Faculdade de Informática - PUCRS
prevedello@inf.pucrs.br

Rafael Antonioli
Faculdade de Informática - PUCRS
rantonoli@inf.pucrs.br

Abstract

Computação voluntária é uma forma de computação distribuída na qual voluntários oferecem processamento e armazenamento para projetos de pesquisa científica. BOINC (Berkeley Open Infrastructure for Network Computing) é um software desenvolvido para auxiliar a computação voluntária. Consiste de um programa cliente que roda sobre os computadores dos voluntários, realizando o processamento de unidades de trabalho recebidas de um servidor central que está associado a um projeto. Sempre que estações voluntárias estiverem ociosas, ou não estiver utilizando toda sua capacidade de processamento, o BOINC entra em ação consumindo ciclos de CPU e posteriormente enviando o resultado processado para o servidor que delegou a tarefa. Este artigo descreve o funcionamento do BOINC, seus objetivos e características, assim como alguns projetos em atividade.

1. Introdução

O poder da computação mundial e o espaço em disco não estão mais concentrados somente sobre centros de supercomputadores e salas de máquinas, ao invés disso, está distribuída sobre centenas de milhões de computadores pessoais pertencentes a usuários comuns.

Esse paradigma é denominado computação voluntária e utiliza CPU ociosa de seus voluntários para realizar supercomputação científica e incentivar pesquisas científicas.

A seção 2 apresenta uma abordagem geral

sobre computação voluntária, sua origem, assim como sua evolução e consolidação até o surgimento da ferramenta BOINC [1]. A seção 3 apresenta os conceitos de computação voluntária e grade computacional, que apesar de semelhantes possuem diferenças bastante acentuadas. As seções 4 e 5 apresentam as principais características e objetivos do BOINC, assim como são apresentados alguns dos principais projetos em atividade. Por último, a seção 6 apresenta uma visão geral do sistema, onde são destacadas algumas de suas principais funcionalidades.

2. Computação voluntária

A computação voluntária emergiu em meados de 1990 com dois projetos, GIMPS [1] e Distributed.net. Em 1999 foi lançado o projeto SETI@home, o qual atraiu milhares de participantes ao redor do mundo e atualmente roda sobre 1 milhão de computadores, e fornece um poder computacional acima de 70 TeraFLOPS. Como base para a dimensão destes números, podemos comparar o poder de processamento de um dos supercomputadores mais potentes da atualidade, o NEC Earth Simulator, que fornece um poder computacional de aproximadamente 35 TeraFLOPs. O número de PCs conectados a internet cresce rapidamente e estima-se que até o ano de 2015 existam mais de 1 bilhão de computadores conectados a rede mundial. Juntos estes PCS podem fornecer vários PetaFLOPs de poder computacional e espaço para armazenamento, para efeito de comparação, se 100 milhões de usuários

fornecessem 10 Gigabytes de espaço em disco, este montante seria o suficiente para exceder a capacidade de qualquer sistema de armazenamento centralizado existente.

Apesar de toda a abundância de recursos, poucos projetos baseados no paradigma de computação voluntária obtiveram êxito. Um dos principais motivos que acarretaram no fracasso destes projetos esta diretamente relacionado à falta de middleware apropriado, o que implicava principalmente em software cliente e servidor e ferramentas de gerenciamento ineficientes. Alguns sistemas open-source foram desenvolvidos, como o Cosm, jxta, e XtremWeb, porém estes sistemas fornecem somente parte das funcionalidades necessárias para projetos com computação voluntária. Existem sistemas comerciais como Entropia e United Devices os quais são mais robustos, porém não são gratuitos.

O BOINC surgiu a partir do projeto GIMPS (*Great Internet Mersenne Prime Search*) desenvolvido em 1996 que utiliza um software instalado em microcomputadores onde através de uma computação em cluster divide o processo entre todos os computadores que tenham o software instalado. O objetivo do GIMPS é buscar o maior número primo utilizando para isso a fórmula matemática de Mersenne. Para efeito de conhecimento, um número primo de Mersenne é um número primo do tipo $2^n - 1$ onde n é um número natural, ou seja, M é um número primo de Mersenne se é efetivamente um número primo e se $M+1$ for uma potência de 2. Assim 7 é um número primo de Mersenne pois $(7+1=8 = 2^3)$ porém 13 não é, pois 14 não pode ser uma potência de 2, tampouco 15 porque nem primo é. Atualmente foram descobertos apenas 44 números primos de Mersenne, os quatro primeiros números são 3, 7, 31 e 127, o quadragésimo quarto número possui 9.808.358 dígitos e foi descoberto com o uso do GIMPS em 4 de setembro de 2006, utilizando um cluster com 700 computadores.

3. Contraste com grade computacional

Computação voluntária [5] e grades computacionais [2] compartilham o objetivo de utilizar recursos computacionais existentes. Recursos computacionais baseados em grades envolvem supercomputadores, clusters, laboratórios de pesquisa, e empresas do setor privado. Tais recursos são gerenciados por profissionais de TI, possuem recursos dedicados maior parte do tempo, e estão conectados em tempo integral, através de links com alta capacidade de vazão. Também existe um relacionamento simétrico entre as organizações, onde cada um pode fornecer ou usar recursos. Em contraste, computação voluntária envolve um relacionamento assimétrico entre projetos e participantes.

Os projetos são compostos basicamente por pequenos grupos de pesquisas acadêmicas compostos por uma quantidade limitada de poder de processamento. A maioria dos participantes são indivíduos que operam sobre sistemas operacionais tais como, Windows, Macintosh e Linux conectados a internet através de conexão discada, cable modem ou ADSL, e que em sua maioria operam atrás de NAT's e *firewalls*. Os computadores são desligados freqüentemente ou desconectados da rede, os participantes não são usuários experientes e somente colaboram com os projetos se existe a possibilidade de receber incentivos, tal como a obtenção de pontos.

Existem diferentes exigências no *middleware* para computação voluntária em relação à grade computacional. Uma arquitetura de Grade deve suportar muitos sistemas comerciais e normalmente orientados a pesquisa acadêmica, além de fornecer um mecanismo geral para descobrir os recursos e acessá-los. Por outro lado, computação voluntária requer pouca entrada e saída e muito poder de CPU, os demais recursos dos computadores voluntários não são utilizados.

4. Objetivos do projeto BOINC

O principal objetivo do BOINC está relacionado ao avanço do paradigma de recursos de computação voluntária [3], afim de incentivar um grande número de projetos, induzindo que seus participantes associem-se simultaneamente em projetos diversos. Os objetivos principais incluem:

- **Reduzir as barreiras de entrada para computação voluntária:** BOINC permite que um pesquisador científico com habilidades limitadas com computadores possa criar e operar um grande projeto, com aproximadamente uma semana de trabalho inicial e uma hora semanal dedicada a manutenção. Um servidor para um projeto BOINC pode consistir de uma simples máquina configurada com software *open-source*, tais como Linux, Apache, PHP, MySQL e Python.
- **Compartilhar recursos através de projetos autônomos:** Projetos BOINC são autônomos, ou seja, os projetos não são autorizados ou registrados de forma centralizada. Cada projeto opera em seu próprio servidor e é administrado pelo criador do projeto. Usuários podem participar de vários projetos e podem definir graus de recursos para cada projeto, tal como utilização de CPU e espaço em disco. Se um voluntário participar de mais de um projeto, então o desempenho global destes projetos tendem a ser melhorado, enquanto um projeto estiver inacessível para manutenção, outros projetos que estavam temporariamente inertes herdam seu poder computacional.
- **Suportar aplicações diversas:** BOINC suporta uma grande variedade de aplicações, fornecendo um mecanismo flexível e escalável para distribuição de dados. Aplicações escritas em linguagens de programação como (C, C++, FORTRAN) podem rodar como uma aplicação BOINC com pequenas modificações, ou até mesmo nenhuma.

- **Participantes recompensados:** Projetos baseados em computação voluntária utilizam sistema de pontuação para atrair e reter participantes. O principal incentivo para muitos participantes está relacionado a pontuação, que se refere a medida quantitativa que expressa o quanto o usuário contribuiu. BOINC fornece um sistema de bilhetagem que viabiliza aos seus usuários obterem informações referentes a diversos tipos de recursos, tais como CPU, rede e disco.

5. Projetos usando BOINC

Qualquer usuário que disponha de uma infraestrutura básica, pode modelar seu problema para utilizar a computação voluntária oferecida pelo BOINC. Atualmente existem grandes projetos [5] em produção utilizando o BOINC, além de outros projetos que estão surgindo, abrangendo diversas áreas do conhecimento humano, tais como medicina, meteorologia e astronomia. A seguir serão brevemente descritos alguns destes projetos:

- **SETI@home:** o objetivo é analisar informações coletadas através de rádio telescópios, afim de tentar descobrir vida em outras galáxias. Os participantes do projeto processam dados provenientes de ondas de rádio captadas no universo pela NASA, radiotelescópios varrem distâncias até 200 anos luz da Terra
- **Climateprediction.net:** o objetivo é desenvolver um modelo avançado de prognóstico do clima. A aplicação foi implementada na linguagem de programação Fortran e seu código ultrapassa um milhão de linhas, produzindo um arquivo de 2GB de informações para cada simulação abrangendo o período de cinquenta anos. Simulações são realizadas utilizando-se parâmetros alternados, tais quais, atividades vulcânicas, solares e interferências humanas.
- **Worldcommunitygrid:** projeto desenvolvido pela IBM cujo objetivo é auxiliar

em pesquisas na área da saúde, em especial Câncer e AIDS. A primeira pesquisa utilizando esta rede será a análise de algumas das 35 milhões de moléculas de uma droga para o tratamento da varíola. A organização espera que cerca de seis projetos possam utilizar esta rede por ano, não se limitando apenas à área de saúde, mas também para auxiliar a prevenção de catástrofes naturais e apoiar projetos para o fornecimento mundial de água e alimentos.

- **Rosetta@home:** o objetivo é auxiliar pesquisadores a encontrarem cura para diversas doenças humanas. Procura estruturas de proteínas, aminoácidos e suas interações para construir proteínas complexas e possibilitar cura ou tratamento como Câncer, HIV, Alzheimer e Malária.
- **Predictor@home:** a curto prazo o Predictor@home pretende testar e avaliar novos métodos e algoritmos para analisar estruturas de proteínas. A médio prazo irá proceder-se à calibração dos métodos do Predictor@home com algumas estruturas conhecidas. A longo prazo pretende-se que o Predictor@home seja um recurso que ajude a analisar e compreender a estrutura das proteínas.
- **LHC@home:** o objetivo é simular o funcionamento do LHC (*Large Hadron Collider*), maior acelerador de partículas no mundo, que está em construção no CERN. O projeto simula como as partículas viajarão pelos 27 Km do túnel. Com a ajuda das informações calculadas, os superímãs que controlam o feixe de energia podem ser calibrados com maior precisão.
- **Einstein@home:** O objetivo é procurar por pulsares (também conhecidas como estrelas de nêutrons) e ondas gravitacionais emitidas por Pulsares, Buracos Negros, Estrelas de Nêutrons, Estrelas de Quarks e outros objetos bastante densos, que teoricamente, podem emitir fortes Ondas Gravitacionais.

6. Visão geral

O BOINC contempla uma série de funcionalidades [4] que o tornam cada vez mais confiável, escalável e de fácil interação. A seguir são listadas algumas destas funcionalidades:

- **Computação redundante:** são implementados mecanismos que identificam e rejeitam resultados errôneos. Uma unidade de trabalho é enviada para no mínimo 3 participantes, o resultado da computação é verificado pelo servidor após atingir a maioria de resultados corretos.
- **Falha e backoff:** situações inusitadas como várias centenas de usuários tentarem conectar-se ao mesmo servidor em um mesmo instante podem acarretar uma sobrecarga ao sistema. Tal situação é tratada definindo um *delay* baseado no número de tentativas de conexão.
- **Preferências dos participantes:** o sistema possibilita uma forte interação com seus participantes. Através de uma intuitiva interface web é possível controlar como e quando seus recursos serão utilizados.
- **Sistema de crédito:** softwares clientes rodam benchmarks periodicamente sobre CPU e baseado nos resultados obtidos são adicionados créditos aos usuários. Se uma unidade de trabalho for enviada para três participantes, o primeiro que reportar o resultado correto da computação, receberá um crédito maior que os seguintes.

Também cabe salientar que diversos mecanismos de segurança foram desenvolvidos e adicionados ao BOINC, entre eles merece destaque o mecanismo redundante para impedir falsificação de resultados, a assinatura de código para impedir a distribuição de aplicações forjadas e limite do tamanho máximo do arquivo de saída para impedir ataques do tipo de negação de serviço.

7. Conclusão

Neste artigo foi descrito o paradigma da computação voluntária versus grade computacional. Foram apresentados os principais objetivos que impulsionam e mantém o BOINC em operação, assim como também foram descritos alguns dos projetos em atividade.

Acreditamos que a computação voluntária é uma solução extremamente eficiente, que está viabilizando a criação e execução de um grande número de projetos, que até então eram inviáveis visto a limitação de poder de processamento.

Atualmente video-games e telefones celulares de última geração já estão equipados com processadores, estes quando ociosos poderiam estar cooperando com o progresso da ciência.

References

- [1] Maurer, J. 2005. A conversation with David Anderson. *Queue* 3, 6 (Jul. 2005), 18-25.
- [2] Cicotti, P., Taufer, M., and Chien, A. A. 2005. DGMonitor: A Performance Monitoring Tool for Sandbox-Based Desktop Grid Platforms. *J. Supercomput.* 34, 2 (Nov. 2005), 113-133.
- [3] Anderson, D. P., Korpela, E., and Walton, R. 2005. High-Performance Task Distribution for Volunteer Computing. In *Proceedings of the First international Conference on E-Science and Grid Computing (December 05 - 08, 2005)*. E-SCIENCE. IEEE Computer Society, Washington, DC, 196-203.
- [4] Anderson, D. P. and Fedak, G. 2006. The Computational and Storage Potential of Volunteer Computing. In *Proceedings of the Sixth IEEE international Symposium on Cluster Computing and the Grid (Ccgrid'06) - Volume 00 (May 16 - 19, 2006)*. CCGRID. IEEE Computer Society, Washington, DC, 73-80.
- [5] Anderson, D. P. 2004. BOINC: A System for Public-Resource Computing and Storage. In *Proceedings of the Fifth IEEE/ACM international Workshop on Grid Computing (Grid'04) - Volume 00 (November 08 - 08, 2004)*. GRID. IEEE Computer Society, Washington, DC, 4-10.