

Volunteer Computing

Anderson Andrei da Silva NUSP 8944025

Patrick Abrahão Menani NUSP 8941050

Vinícius Pessoa Duarte NUSP 8941043

Sumário

- . Introdução
- . Objetivo
- . História
- . Implementação
- . Benefícios e desvantagens
- . Exemplos:
 - . Middlewares
 - . Projetos
- . Conclusão
- . Referências

Introdução

Introdução

Motivação:

Recursos computacionais;

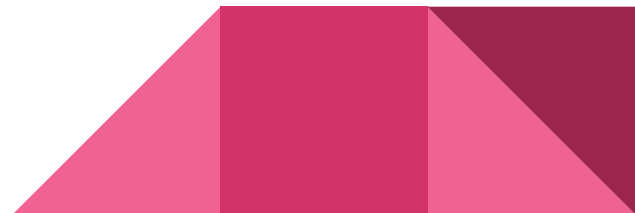
Infraestrutura.

Solução:

Voluntários:

Interconexão via Internet;

Recursos ociosos.



Objetivo

Agrupar recursos computacionais:

Armazenamento;

Processamento;

Middleware.



História

Volunteer Computing - Timeline



GIMPS

1996

Jan

1997

1998

1999

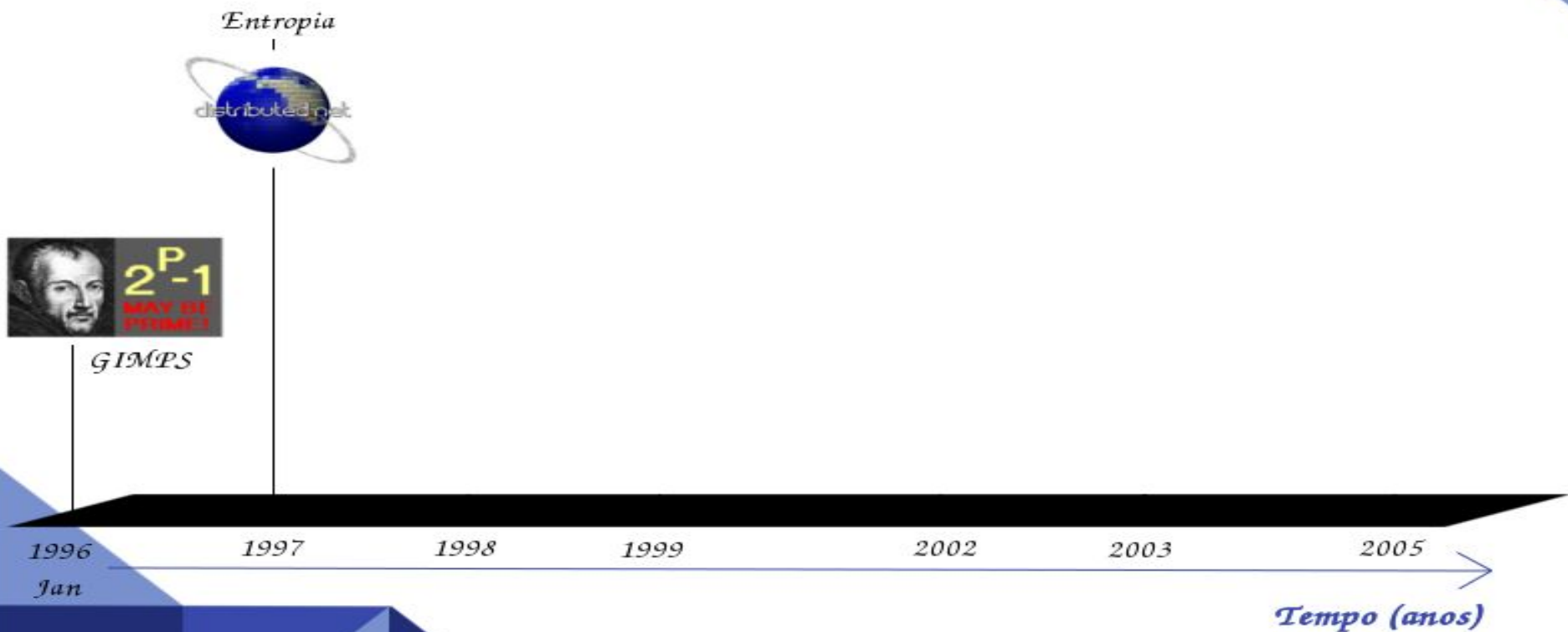
2002

2003

2005

Tempo (anos)

Volunteer Computing - Timeline



Volunteer Computing - Timeline

Bayanihan + "Volunteer Computer"

Entropia



GIMPS

1996

Jan

1997

1998

1999

2002

2003

2005

Tempo (anos)

Volunteer Computing - Timeline

Bayanihan + "Volunteer Computer"

Entropia



Folding@home
distributed computing

SETI HOME



GIMPS

1996

Jan

1997

1998

1999

2002

2003

2005

Tempo (anos)

Volunteer Computing - Timeline

Bayanihan + "Volunteer Computer"

Entropia



Folding@home
distributed computing

SETI HOME

BOINC



GIMPS

1996

1997

1998

1999

2002

2003

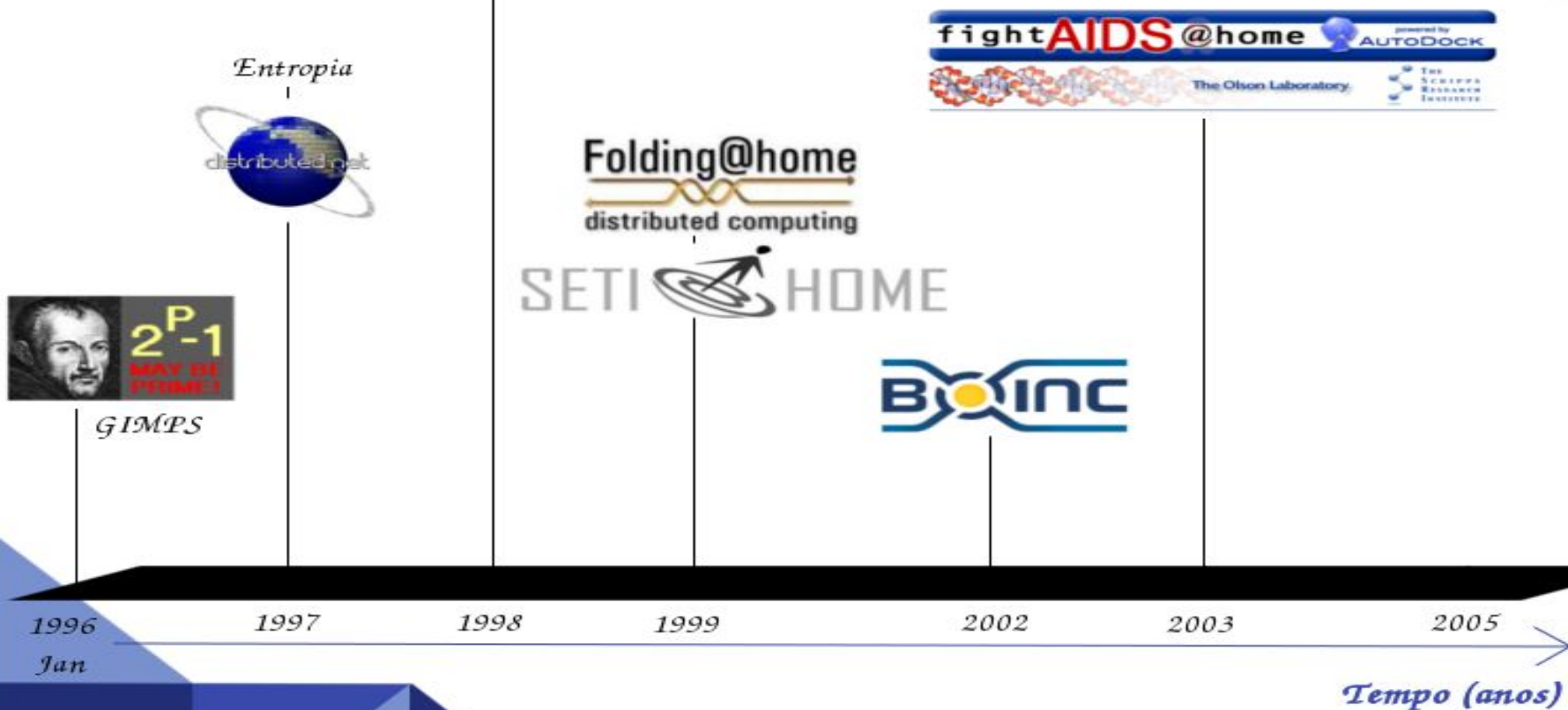
2005

Jan

Tempo (anos)

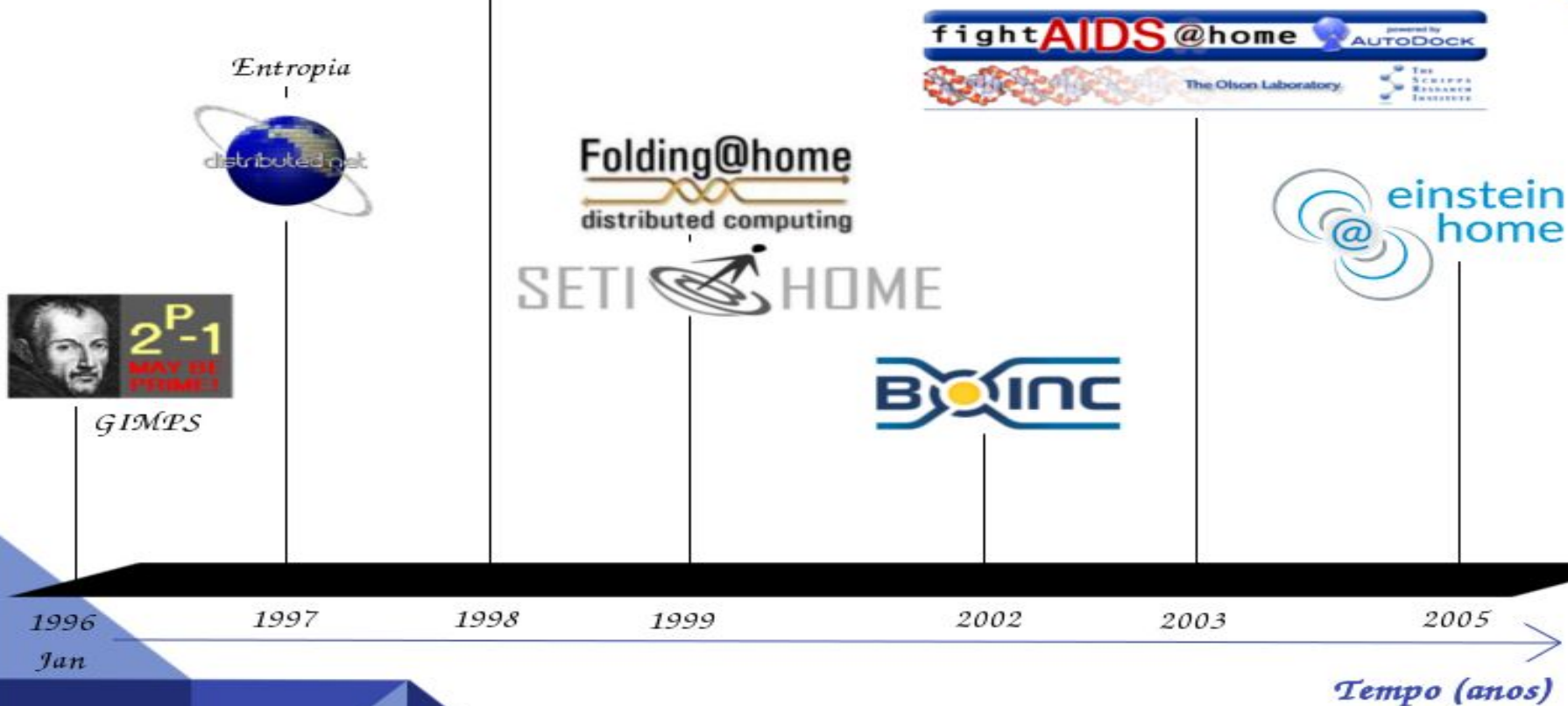
Volunteer Computing - Timeline

Bayanihan + "Volunteer Computer"



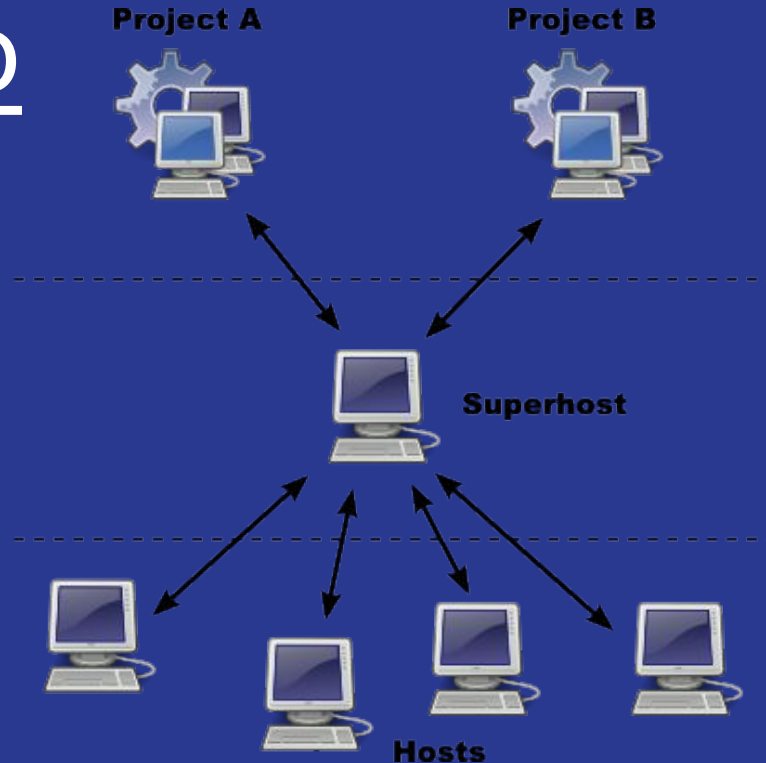
Volunteer Computing - Timeline

Bayanihan + "Volunteer Computer"



Implementação

- . Como funciona
- . Servidores
- . Clientes
- . Tratamento de erros
- . Um exemplo simples



Como funciona

Um problema divisível em tarefas menores é definido;

Um conjunto de servidores e clientes trabalham para distribuir tarefas e resolvê-las;

As soluções das tarefas individuais são recebidas por alguém que entende como montar a resposta final;

A resposta final pode ser extraída.



Servidores

Particionam o problema em tarefas que podem ser distribuídas;

Ferramentas como o BOINC podem estimar para uma tarefa:

- Número de cálculos de ponto flutuante;

- Espaço de armazenamento necessário;

- Conexão de internet para transmitir os dados.

Distribuem as tarefas para os clientes.



Servidores - Por quê estimar esses custos?

As estimativas ajudam a garantir que um cliente receba tarefas factíveis de serem realizadas com os recursos disponíveis;

Evita gasto de tempo e processamento com tarefas inviáveis:

- Cliente não consegue armazenar os dados gerados;

- Tempo para realizar a tarefa pode ser grande para o poder computacional.



Cientes

Os clientes são as máquinas que doam seus recursos para um projeto;

Têm um software instalado para receber as tarefas de um servidor;

O cliente realiza a tarefa recebida em algum momento;

Entrega para o servidor os dados gerados na computação da tarefa;

Recebe uma nova tarefa .



Clientes - Quando a tarefa é executada?

Em geral, o software que gerencia a execução das tarefas espera a máquina estar em estado ocioso - Proteção de tela ativada, por exemplo;

A prioridade da tarefa é baixa;

A ideia é evitar perda de performance quando o usuário está muito ativo;

O usuário pode não apoiar o projeto, caso contrário.



Tratamento de erros

Diversos problemas podem ocorrer nesse processo:

- Clientes podem não retornar os dados computados;

- Clientes com malfuncionamento podem retornar respostas erradas;

- Clientes tentam fraudar as respostas:

 - Receber crédito por completar mais tarefas;

 - Prejudicar o projeto por quaisquer motivos.



Tratamento de erros

Para garantir a integridade dos dados, os sistemas podem utilizar o paradigma de computação replicada;

Os servidores distribuem uma mesma tarefa para mais de um cliente;

Quando os clientes retornam os dados sobre uma tarefa, estes são comparados entre os clientes que realizaram a mesma tarefa;

Se os dados são compatíveis dentro de uma margem de erro, o resultado é aceito;

Caso contrário, a tarefa deve ser executada novamente.



Um exemplo simples - Almoço para os alunos

Imagine que o professor Alfredo decidiu organizar, com os professores DCC, um almoço para os alunos de escola pública que participam de um projeto de computação no IME.

O professor Alfredo define que 3 coisas são necessárias:

Comida;

Sobremesa;

Entretenimento.



Um exemplo simples - Almoço para os alunos

Agora, é preciso pedir que os professores ajudem com as tarefas;

Os professores Gubi, Leliane, Carlinhos e Daniel se voluntariam para ajudar;

O Alfredo distribui, de acordo com as habilidades dos professores:

Leliane e Daniel - Ficam responsáveis pela comida;

Carlinhos - Será o responsável pela sobremesa;

Gubi - Cuidará do entretenimento (Mágica).



Um exemplo simples - Almoço para os alunos

Na medida em que um professor completa sua tarefa, leva a comida ou apresentação para o saguão do bloco B, onde o almoço será montado;

O Alfredo então verifica se cada professor levou o combinado, senão, pede para que o professor vá fazer a tarefa correta.

Por fim, o Alfredo organiza o almoço e chama as crianças!



Benefícios e Desvantagens

Benefícios

Conscientizar a população sobre projetos de pesquisa;

Aumentar o engajamento do público;

Permitir métodos de pesquisa mais baratos;

Gerar resultados mais rapidamente;

Utiliza os recursos computacionais de forma mais eficiente.



Desvantagens

Tecnologia ainda em desenvolvimento;

Aumento do consumo de energia do doador;

Decréscimo na performance da máquina do doador (voluntário).



Exemplos

Middleware

Middleware

Berkeley Open Infrastructure for Network Computing (BOINC) é o sistema de middleware mais utilizado. Ele oferece um sistema de software para os clientes em diversas plataformas, como Windows, Mac OS X, Linux, Android, e outras variantes de Unix;

XtremWeb é usado principalmente como uma ferramenta de pesquisa, ele é desenvolvido por um grupo baseado na University of Paris-South;



Middlewares

Xgrid é desenvolvido pela Apple. Ele fornece um sistema de cliente e servidor que são exclusivos do sistema Mac OS X;

Grid MP é uma plataforma de middleware desenvolvida pela United Devices, foi utilizada em alguns projetos de volunteer computing como o grid.org, World Community Grid, Cell Computing e Hikari Grid.



Exemplos

Projetos


Folding@HOME

Criado em Stanford com objetivo de entender como funcionam interações entre proteínas e suas relações com doenças;

O Alzheimer, por exemplo, é causado pela agregação de algumas pequenas proteínas chamadas Abeta peptídeos (agregados);

Tóxicos para os neurônios, causando a morte de várias células neuronais.

Criar drogas para combater a doença, além de ser possível ter ideia de como esses agregados se formam, talvez sendo possível impedir a formação deles.



einstein@HOME

Procura encontrar estrelas de nêutrons a partir da análise de dados gerados por detectores de ondas gravitacionais, rádio-telescópios e satélites de raios-gama;

Os voluntários do projetos já encontraram cerca de 50 estrelas de nêutrons;

Tem como meta fazer a primeira detecção direta de ondas gravitacionais emitidas por estrelas de nêutrons em rotação (pulsares).

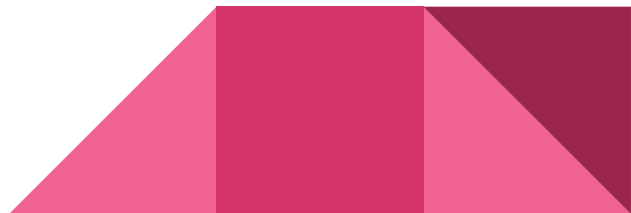


FightAIDS@HOME

Gerenciado pelo laboratório Olson no instituto de pesquisa Scripps, localizado nos Estados Unidos;

Modela a evolução do vírus HIV à drogas e avalia possíveis substâncias candidatas a drogas contra o HIV.

Recentemente se juntou a World Community grid, a qual agora tem mais de um milhão de voluntário registrados.



Conclusão

Conclusão

O uso de computação voluntária pode ser de extrema utilidade quando tratamos de processo computacionais extremamente custosos que podem ser distribuídos entre diversos dispositivos;

O campo científico tem muito a ganhar com a computação voluntária, o custo de se obter os resultados, que seriam necessários clusters extremamente caros, se torna praticamente nenhum para os pesquisadores, possibilitando uma maior evolução de diversas pesquisas.



Referências

- <http://www.volunteer-computing.org/>
- <http://setiathome.berkeley.edu/>
- <https://pt.wikipedia.org/wiki/BOINC/>
- <http://folding.stanford.edu/diseases/>
- <https://einsteinathome.org/pt-br/about>
- <http://www.wseas.us/e-library/conferences/2007corfu/papers/540-118.pdf>
- https://en.wikipedia.org/wiki/Volunteer_computing
- <http://www.volunteer-computing.org/EN/why-use-volunteer-computing.html>
- <http://www.volunteer-computing.org/EN/volunteer-powered-projects.html>
- https://pdfs.semanticscholar.org/4fbd/68ba641f16d1566d3f1dc7704f2f63b938a3.pdf?_ga=2.980344.89964398.1497913612-119129416.1497913612
- <http://www.sciencedirect.com/science/article/pii/S0167739X99000187>
- <http://groups.csail.mit.edu/cag/bayanihan/papers/eurotools98/>
- <http://groups.csail.mit.edu/cag/bayanihan/papers/fgcs/html/>

