Etherclan

Rede distribuída descentralizada para localização de outros usuários

Henrique Gemignani Passos Lima

12 de novembro de 2013

Orientador: Prof. Dr. Daniel Macedo Batista

1. Introdução: Motivações e Objetivos
2. Conceitos
3. Funcionamento da rede
4. Resultados

Motivações

Interesse em:

- Jogos eletrônicos.
- Redes de computadores.

Motivações

Interesse em:

- Jogos eletrônicos.
- Redes de computadores.

Uma forma comum de juntar os dois é um jogo multijogador através da Internet!

Motivações - Internet

Mas quando se fala de um projeto open-source, quais dificuldades surgem?

Motivações - Internet

Mas quando se fala de um projeto open-source, quais dificuldades surgem?

1. Grande parte da lógica nos servidores

Motivações - Internet

Mas quando se fala de um projeto open-source, quais dificuldades surgem?

1. Grande parte da lógica nos servidores
   - Necessidade de manter servidores potentes, caro.

Motivações - Internet

Mas quando se fala de um projeto open-source, quais dificuldades surgem?

1. Grande parte da lógica nos servidores
   - Necessidade de manter servidores potentes, caro.
2. Servidores dedicados por parte dos jogadores

Motivações - Internet

Mas quando se fala de um projeto open-source, quais dificuldades surgem?

1. Grande parte da lógica nos servidores
   - Necessidade de manter servidores potentes, caro.
2. Servidores dedicados por parte dos jogadores
   - Como encontrar esses servidores de maneira automatizada?

Proposta de TCC

Etherclan

Proposta de TCC

Etherclan

- Rede peer-to-peer, isto é, uma rede descentralizada.

Proposta de TCC

Etherclan

- Rede peer-to-peer, isto é, uma rede descentralizada.
- Função de encontrar outros membros da rede.

Proposta de TCC

Etherclan

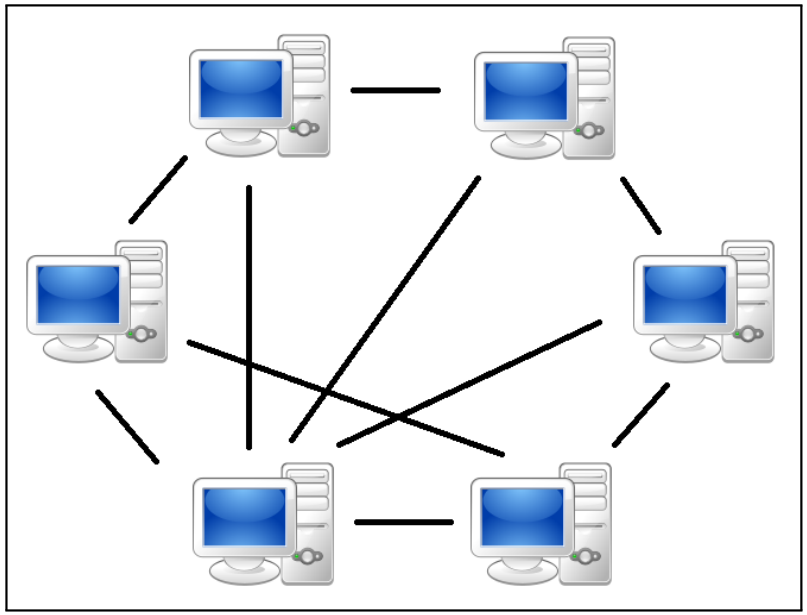
- Rede peer-to-peer, isto é, uma rede descentralizada.
- Função de encontrar outros membros da rede.
- Pode ser utilizada por mais de um jogo simultaneamente.

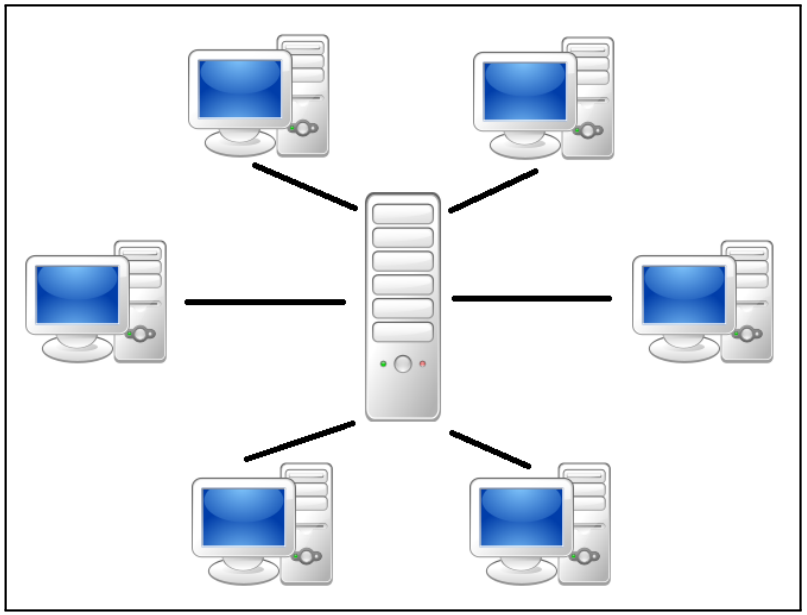
Proposta de TCC

Etherclan

- Rede peer-to-peer, isto é, uma rede descentralizada.
- Função de encontrar outros membros da rede.
- Pode ser utilizada por mais de um jogo simultaneamente.
- Capaz de buscar por membros de um jogo específico.

Peer-to-peer X Cliente-Servidor





Numa rede peer-to-peer cada nó da rede atua como consumidores e fornecedores de recursos, em contraste com redes cliente-servidor onde nós clientes apenas requisitam serviços de servidores centrais.

Funcionamento da rede

Entrando na rede

Para entrar na rede, é necessário dois passos.

Entrando na rede

Para entrar na rede, é necessário dois passos.

1. Criar um nó.

Entrando na rede

Para entrar na rede, é necessário dois passos.

1. Criar um nó.
2. Informar algum membro da rede sobre o novo nó.

Bootstrapping

Mas como eu consigo esse primeiro nó?

Bootstrapping

Mas como eu consigo esse primeiro nó?

- Lista fixa de IP e porta.

Bootstrapping

Mas como eu consigo esse primeiro nó?

- Lista fixa de IP e porta.
- Busca cega.

Bootstrapping

Mas como eu consigo esse primeiro nó?

- Lista fixa de IP e porta.
- Busca cega.
- Aproveitar alguma outra rede P2P?

Protocolo de Rede

Para que dois nós consigam se comunicar, é necessário que ambos sigam um protocolo de rede.

Protocolo de Rede

Para que dois nós consigam se comunicar, é necessário que ambos sigam um protocolo de rede.

Qual o básico necessário que esse protocolo tem que permitir?

Protocolo de Rede

Para que dois nós consigam se comunicar, é necessário que ambos sigam um protocolo de rede.

Qual o básico necessário que esse protocolo tem que permitir?

- Informar um novo nó na rede.

Protocolo de Rede

Para que dois nós consigam se comunicar, é necessário que ambos sigam um protocolo de rede.

Qual o básico necessário que esse protocolo tem que permitir?

- Informar um novo nó na rede.
- Requisitar uma lista de nós da rede.

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

- Um UUID (Universally Unique IDentifier), segundo RFC 4122.

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

- Um UUID (Universally Unique IDentifier), segundo RFC 4122.
- Endereço IP e porta.

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

- Um UUID (Universally Unique IDentifier), segundo RFC 4122.
- Endereço IP e porta.
- Lista de serviços fornecidos.

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

- Um UUID (Universally Unique IDentifier), segundo RFC 4122.
- Endereço IP e porta.
- Lista de serviços fornecidos.

Serviços?

Sobre os Nós

Antes de se aprofundar no protocolo, quais informações temos sobre cada nó?

- Um UUID (Universally Unique IDentifier), segundo RFC 4122.
- Endereço IP e porta.
- Lista de serviços fornecidos.

Serviços?
Funcionam com expansões do protocolo da rede.
Útil para saber se o nó é do mesmo jogo que o seu.

Desenvolvimento do Protocolo

Estratégia inicial foi listar todas as necessidades do projeto de início, similar a como se modela um banco de dados.

Desenvolvimento do Protocolo

Estratégia inicial foi listar todas as necessidades do projeto de início, similar a como se modela um banco de dados.

Pessoal: Pelo menos 1 mês foi gasto travado nesse passo, sem progresso. :(

Desenvolvimento do Protocolo

Estratégia inicial foi listar todas as necessidades do projeto de início, similar a como se modela um banco de dados.

Pessoal: Pelo menos 1 mês foi gasto travado nesse passo, sem progresso. :(

Projeto começou a andar bastante quando a estratégia foi mudada para implementar o que precisa para o momento, mudando sempre que necessário, seguindo ideologia de programação ágil.

Resultados

Daemon

```
Server Start with timeout 1
Create Inbound Connection
Received:'KEEP_ALIVE ON'
Received:
'ANNOUNCE_SELF 84FBBD0D-DDF9-4083-AA91-728909F24E78 14084 chat'
Received:'REQUEST_NODE_LIST'
Sending:
'NODE_INFO 84FBBD0D-DDF9-4083-AA91-728909F24E78 127.0.0.1 14084 chat
Received:'KEEP_ALIVE OFF'
Finish
Remove Connection
```

Proposta: fornecer um nó para servir como bootstrap para outros.

Chat



Chat



Proposta: um chat que envia e recebe mensagens para nós da rede, utilizando um serviço de nome *chat*. Não possui nomes de usuário, nem confirmação de quem recebeu as mensagens enviadas.

Chat



Proposta: um chat que envia e recebe mensagens para nós da rede, utilizando um serviço de nome *chat*. Não possui nomes de usuário, nem confirmação de quem recebeu as mensagens enviadas.

Desenvolvido como uma prova de conceito, demonstrou os seguintes pontos:

Chat



Proposta: um chat que envia e recebe mensagens para nós da rede, utilizando um serviço de nome *chat*. Não possui nomes de usuário, nem confirmação de quem recebeu as mensagens enviadas.

Desenvolvido como uma prova de conceito, demonstrou os seguintes pontos:

- Funciona!

Chat



Proposta: um chat que envia e recebe mensagens para nós da rede, utilizando um serviço de nome *chat*. Não possui nomes de usuário, nem confirmação de quem recebeu as mensagens enviadas.

Desenvolvido como uma prova de conceito, demonstrou os seguintes pontos:

- Funciona!
- Precisa de melhorias no controle de nós conhecidos

Futuro

- Definir estratégias melhores para:
  - Considerar um nó como morto
  - Parar busca por novos nós
  - Começar outra busca por novos nós

Futuro

- Definir estratégias melhores para:
  - Considerar um nó como morto
  - Parar busca por novos nós
  - Começar outra busca por novos nós

- Fazer uma biblioteca em C++ para facilitar integração com outros projetos.
  - Um projeto em C++ é facilmente exportado para diversas outras linguagens. (Ouroboros)

Futuro

- Definir estratégias melhores para:
  - Considerar um nó como morto
  - Parar busca por novos nós
  - Começar outra busca por novos nós
- Fazer uma biblioteca em C++ para facilitar integração com outros projetos.
  - Um projeto em C++ é facilmente exportado para diversas outras linguagens. (Ouroboros)
- Utilizar em outros projetos!

kthxbye

Fim

Perguntas?