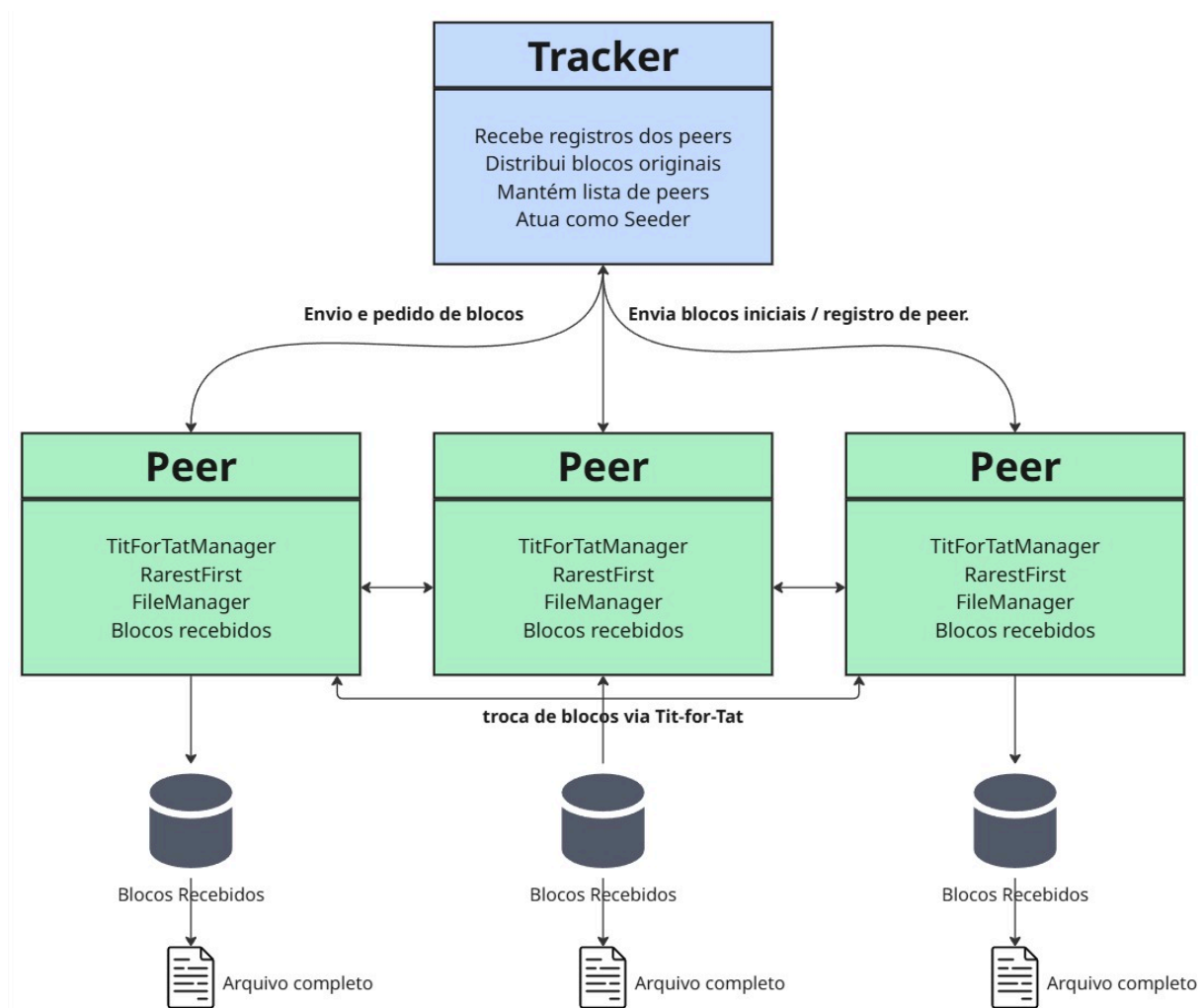


Relatório

Este relatório descreve a implementação, funcionamento e os resultados do sistema de compartilhamento de arquivos em uma rede peer-to-peer (P2P), com estratégias inspiradas no BitTorrent como rarest first e tit-for-tat. O objetivo do sistema é possibilitar a distribuição descentralizada de arquivos entre peers, minimizando a dependência de um servidor central.

1. Diagrama da arquitetura



2. Protocolo de Comunicação

O sistema é composto por dois componentes principais: o **Tracker**, que atua como coordenador e seeder inicial, e os **Peers**, que solicitam e compartilham blocos do arquivo. Toda a comunicação ocorre via sockets TCP com mensagens no formato JSON.

Ao iniciar, um peer se conecta ao Tracker enviando sua identificação, IP, porta e os blocos que possui. O Tracker responde com a lista de todos os peers ativos na rede, informando quais blocos cada um possui. Os peers usam essas informações para decidir com quem trocar blocos.

Entre os peers, o protocolo de troca funciona da seguinte maneira: um peer envia uma solicitação indicando o número do bloco desejado. Se o peer remoto estiver desbloqueado e possuir o bloco, ele o envia como resposta. O peer que recebe salva o bloco e registra o remetente no log.

3. Estratégias de Rarest First e Tit-for-Tat

O algoritmo rarest first calcula a raridade dos blocos com base nos blocos que os peers possuem e ainda faltam ao solicitante. Blocos menos comuns são priorizados nas trocas.

Já o tit-for-tat simplificado define quais peers ficam desbloqueados (unchoked) para que ocorra a troca de blocos. A cada 10 segundos, o peer escolhe os 4 com maior score de blocos raros como desbloqueados fixos, e seleciona aleatoriamente mais 1 como optimistic unchoke. Se esse peer aleatório possuir blocos raros, ele pode substituir o último do top 4.

Esse mecanismo impede que os peers sempre troquem com os mesmos nós e garante a exploração de novas conexões. Os peers bloqueados não atendem a pedidos de blocos.

4. Comportamento da Rede

Nos testes realizados, observou-se que os peers que iniciam juntos tendem a completar o download quase simultaneamente. Isso acontece porque todos recebem blocos iniciais aleatórios do Tracker, compartilham uma visão semelhante da rede (atualizada a cada reconsulta), e o mecanismo de tit-for-tat garante que a maioria tenha oportunidades equivalentes de troca.

Contudo, foi identificado que após completar o arquivo, os peers não atuavam como seeders, ou seja, passavam a negar blocos aos novos peers por estarem fora da lista de desbloqueados. Esse comportamento foi ajustado, permitindo que peers que completam o arquivo continuem a servir blocos normalmente, respeitando o protocolo de desbloqueio.

5. Resultados dos Testes

Em um teste com **9 peers** e o **Tracker atuando como único seeder**, o sistema foi utilizado para distribuir um arquivo de **64 MB**, dividido em blocos de 256 KB. O tempo total para que todos os peers completassem o download foi de **21 minutos**.

A análise dos logs revelou que **a maioria dos blocos foi recebida do Tracker**, o que era esperado, visto que ele era o único peer com todos os blocos no início do processo. Apesar disso, houve colaboração entre peers intermediários, que também compartilharam blocos entre si conforme iam completando partes do arquivo. **Para poder testar também é necessário alterar os arquivos `start_peersh.cmd` e `start_tracker.cmd`**, colocando o seu ip e o caminho do arquivo completo dentro deles, além de não possuir nenhuma aplicação rodando na porta 9001 para o tracker poder usar.

6. Dificuldades Encontradas

Durante o desenvolvimento, enfrentamos diversos desafios:

- **Sincronização e reconsulta:** garantir que os peers tivessem uma visão consistente da rede e atualizassem suas listas de blocos conhecidos exigiu a implementação de reconsultas periódicas ao Tracker.
- **Transferência robusta de blocos via socket:** blocos de 256 KB exigiram um mecanismo confiável de leitura para garantir a entrega completa e precisa.
- **Implementação justa do tit-for-tat:** foi necessário ajustar a lógica para que a seleção dos peers desbloqueados considerasse corretamente os blocos raros e mantivesse a lista com 4 fixos e 1 otimista.
- **Logs detalhados:** o sistema foi aprimorado para registrar qual peer forneceu qual bloco, o que permitiu validações durante os testes.

7. Reflexão individual

Lucas: Eu fui responsável por implementar o algoritmo Rarest First, a estrutura do bloco (Block) e por elaborar o diagrama da arquitetura do sistema. Essas atividades me permitiram aprofundar meus conhecimentos em organização de dados para comunicação entre peers. Trabalhar na lógica de seleção de blocos raros ajudou o meu entendimento sobre estratégias de balanceamento e eficiência nas redes P2P. A criação do diagrama também me ajudou a visualizar melhor as interações entre os componentes, contribuindo para o entendimento global da solução implementada.

Paulo: Fiquei responsável por implementar boa parte da lógica do peer, e posso dizer que foi uma experiência bem intensa. A parte de trocar blocos com outros peers, lidar com Tit-for-Tat, e integrar tudo com o tracker exigiu bastante teste e paciência. No começo parecia meio confuso, mas aos poucos as peças foram se encaixando. Um dos maiores desafios foi garantir que os peers realmente trocassem blocos entre si e que o sistema se comportasse como um P2P de verdade, sem depender só do tracker. Ver o sistema funcionando e os blocos sendo distribuídos foi bem satisfatório, principalmente depois de ajustar a lógica para os peers virarem seeders automaticamente. Aprendi bastante sobre redes, concorrência com threads e até estratégias de distribuição como o rarest first. No geral, achei um projeto trabalhoso, mas que valeu muito a pena.

Brenno: Durante o desenvolvimento deste projeto, fui responsável por implementar a funcionalidade consistia na elaboração da leitura dos arquivos conforme o programa disponibilizado no repositório. Essa tarefa envolveu a interpretação e adaptação do código existente para garantir que os arquivos fossem processados corretamente, seguindo a lógica estabelecida pelo sistema P2P. Confesso que enfrentei desafios significativos durante esse processo, principalmente devido à minha pouca familiaridade com a linguagem Python, que foi essencial para a implementação. No entanto, esses obstáculos se

transformaram em oportunidades de aprendizado. Com a ajuda dos meus colegas, consegui superar as dificuldades iniciais e entender melhor a estrutura do código, especialmente os aspectos relacionados à manipulação de arquivos e à comunicação entre os peers. Além disso, baseei-me nas respostas e soluções apresentadas pelos outros membros do grupo para consolidar meu entendimento e garantir que minha parte estivesse alinhada com o restante do projeto. Essa experiência foi extremamente enriquecedora, pois me permitiu não apenas aprimorar minhas habilidades técnicas em Python, mas também compreender a importância do trabalho colaborativo em um ambiente de desenvolvimento. A troca de conhecimentos e a integração entre as diferentes partes do sistema reforçaram minha percepção sobre como cada componente, mesmo que aparentemente simples, é crucial para o funcionamento harmonioso do todo. No final, pude contribuir de maneira significativa para o projeto e, ao mesmo tempo, expandir meu conhecimento sobre sistemas P2P e organização de dados.

Eduardo: Achei o projeto bem desafiador inicialmente, principalmente por ser algo diferente do que costumo fazer, com bastante uso de sockets e comunicação entre máquinas. Fiquei responsável pelo Tracker e pela classe PeerInfo, que no começo pareciam simples, mas foram essenciais pra tudo funcionar depois. Curti bastante entender como funcionava o envio de blocos, como os peers se registravam e como o sistema começava a distribuir os dados. Também ajudei em várias partes do Peer, mesmo sem ter feito ele todo, principalmente na parte de integração com o Tracker. Foi interessante ver o sistema ganhar vida aos poucos e perceber como cada componente precisava estar bem encaixado. No geral, foi um trabalho que exigiu bastante paciência e testes, mas que ajudou muito a entender melhor como funciona um sistema P2P de verdade.

8. Conclusão

O sistema atendeu aos requisitos fundamentais de um protocolo P2P, com descentralização parcial e colaboração ativa entre peers. O uso combinado de rarest first e tit-for-tat favoreceu a distribuição eficiente de blocos e impediu a dependência excessiva de apenas um nó.

Apesar de o Tracker ter sido o principal fornecedor no teste, o comportamento esperado de troca entre peers também foi observado. Com ajustes adicionais, como múltiplos seeders ou entrada dinâmica de peers, o sistema pode escalar para redes mais complexas.

O projeto foi desenvolvido de forma modular e extensível, com componentes separados, logs completos e uma arquitetura simples de executar e manter.