

Programação de Aplicações Distribuídas

Ricardo Gomes Leal Costa
Orientadora: Noemi Rodriguez



❖ As **aplicações geograficamente distribuídas** operam em conjuntos de máquinas fisicamente separadas e interligadas em rede. Este tipo de aplicação é comum em bancos, aeroportos, sistemas de e-commerce e até em jogos multiplayer online.

❖ Um dos grandes **desafios** do seu desenvolvimento é a necessidade de manter um estado consistente entre todos os participantes frente às condições variáveis da rede e do grupo de máquinas, sem interromper o andamento da aplicação.

❖ Nossa **solução** foi criar a biblioteca **DALua** para dar suporte ao desenvolvimento de aplicações distribuídas, utilizando a linguagem de programação Lua.

Implementação

- ❖ Construída com a linguagem interpretada Lua.
- ❖ Utiliza o sistema ALua (Asynchronous Lua):
 - ❖ Modelo assíncrono e orientado a eventos (callbacks).
 - ❖ Cada mensagem é um trecho de código em Lua executado automaticamente no processo receptor.
- ❖ Uma aplicação ALua é composta de vários processos que podem ser distribuídos em diferentes máquinas.

Recursos de Comunicação

- ❖ **Envio de mensagens** como chamadas de função remotas, com suporte a *multicast*.
- ❖ **Ordenação de mensagens** do tipo causal ou total com uso de relógios lógicos, se necessário.
- ❖ **Exclusão mútua distribuída** para que apenas um processo da aplicação tenha acesso a algum recurso compartilhado num dado instante.

Recursos de Tolerância a Falhas

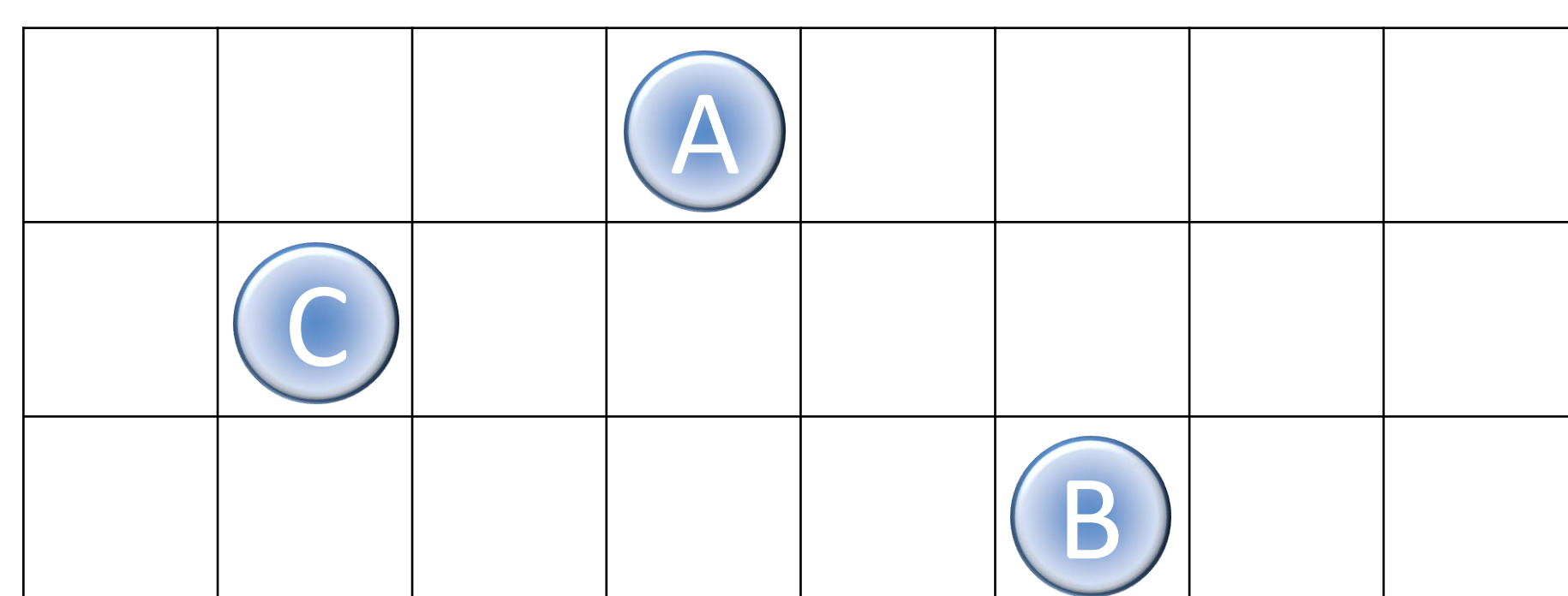
- ❖ **Sistema de eventos** para todas as ações do DALua, que podem ser monitorados e tratados a critério do usuário.
 - ❖ Eventos de erro são tratados automaticamente por padrão, informando a localização e o motivo do erro.
- ❖ **Sistema de temporizadores** que executa comandos em Lua periodicamente, podendo ser utilizado para simular várias situações de falha ou de atraso no envio e recebimento das mensagens.

Casos de Teste

- ❖ **Verificação de funcionamento:** testes automatizados com temporizadores, para cada recurso implementado.
- ❖ **Aplicação exemplo:** protótipo de jogo multiplayer distribuído onde cada jogador representa um processo distinto. Os jogadores entram e saem dinamicamente.
 - ❖ Utiliza todos os recursos da biblioteca DALua, permitindo a identificação e solução de problemas e deficiências da biblioteca mais facilmente.

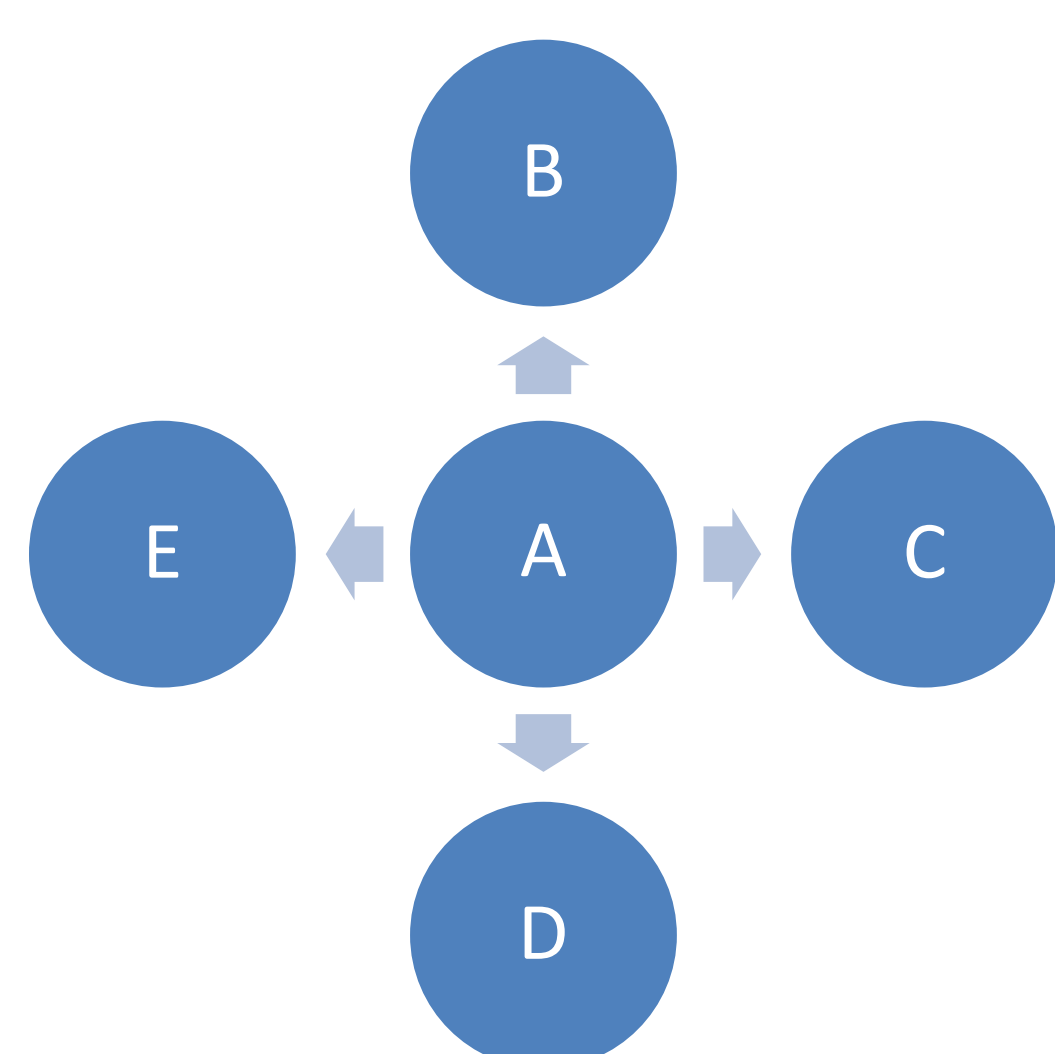
Protótipo de Jogo MMORPG com DALua

- ❖ Jogo *Massive Multiplayer Online Role Playing Game* (MMORPG) estilo *Player Versus Player* (PVP).
- ❖ Cada jogador (processo) está posicionado num mapa 2D, que é compartilhado entre todos os processos.
- ❖ O objetivo do jogo é atacar e matar outros jogadores.



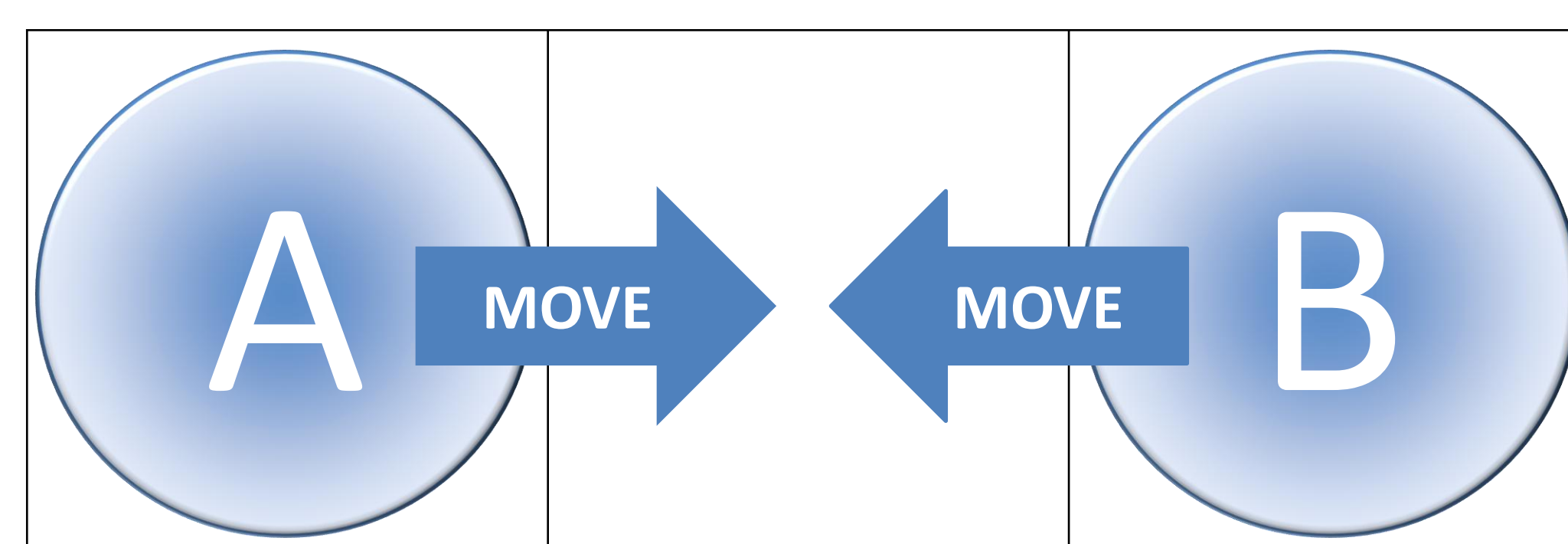
Visão do mapa com os jogadores A, B e C.

- ❖ Os jogadores podem trocar mensagens de texto, que são transmitidas por broadcast para todos os processos.

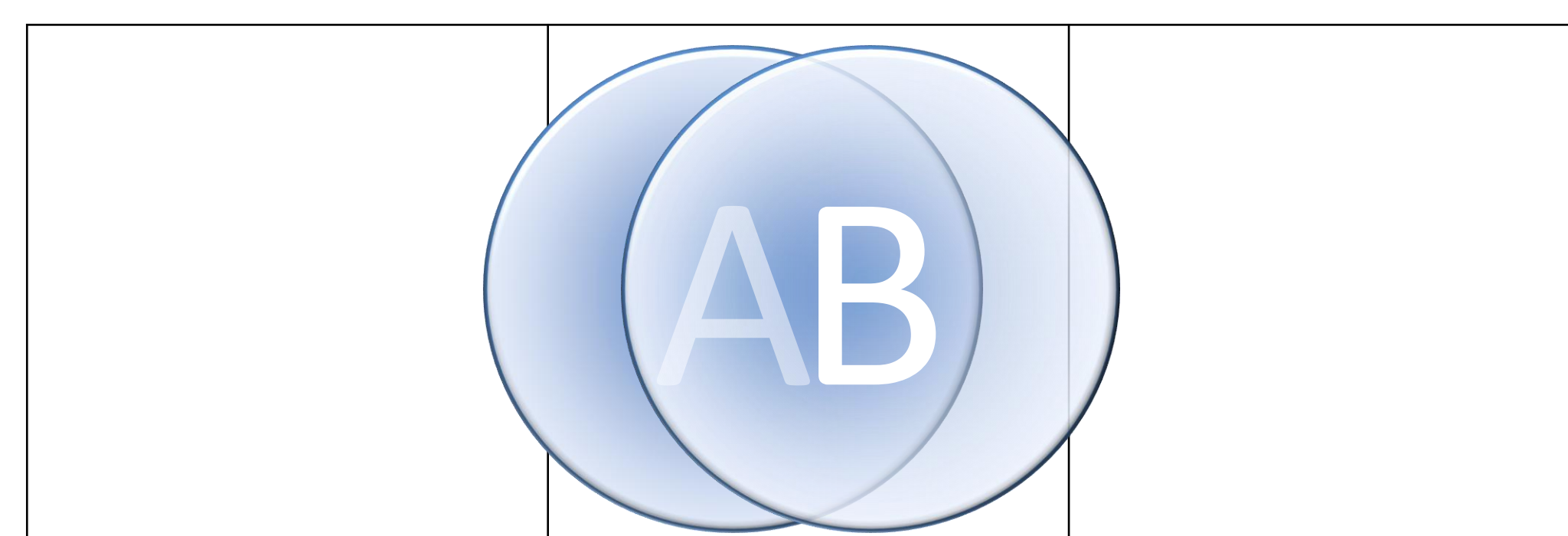


O processo A envia uma mensagem de texto para os demais processos através de broadcast.

- ❖ Comandos que modificam o mapa (movimento ou ataque) requerem o uso de exclusão mútua distribuída, que utiliza o algoritmo de Ricart-Agrawala, para evitar condições de corrida como no exemplo abaixo.



Movimento simultâneo dos jogadores A e B.



Os dois jogadores colidem na mesma posição.

- ❖ Analogamente, a entrada e saída de jogadores requer acesso exclusivo para evitar nomes duplicados no mapa ou inconsistências dentre os participantes do jogo.
- ❖ Com o uso dos temporizadores, foi possível simular os comandos dos jogadores para criar testes automatizados.