Shun-Yun Hu, Shao-Chen Chang, Jehn-Ruey Jiang

VORONOI STATE MANAGEMENT FOR PEER-TO-PEER MASSIVELY MULTIPLAYER ONLINE GAMES

Universidade da Beira Interior
Sistemas Distribuídos e Tolerância a Falhas 2010/2011
Pedro Pereira M3715
Ricardo Pesqueira E7111

Esquema da Apresentação

- Estado Actual dos Jogos Online
- Motivação
- Desafios
- MMOG Server Clusters
- Problemas dos MMOG Server Clusters
- Voronoi State Management
- VSM: Ideias Base
- Modelos de Rede para Jogos
- Modelos de Consistência para Jogos
- VSM: Controlo de consistência
- VSM: Balanceamento do Load
- VSM: Tolerância a Falhas
- Conclusão

Estado Actual dos Jogos Online

- Os Massive Multiplayer Online Games (MMOG) são o género de jogo com crescimento mais acentuado;

- World of Warcraft (9 milhões de subcrições, 500,000 jogadores online);
- Second Life (10 milhões de contas, milhões de transacções);

Estado Actual dos Jogos Online

- Máximo de utilizadores concorrentes num mundo:

- MMORPG: 2,000 ~ 3,000
- EVE Online: 30,000
- Second Life: 30,000 ~ 45,000

Motivação

- Limitações, como a escalabilidade que podem ser resolvidas com soluções peer-to-peer (P2P);

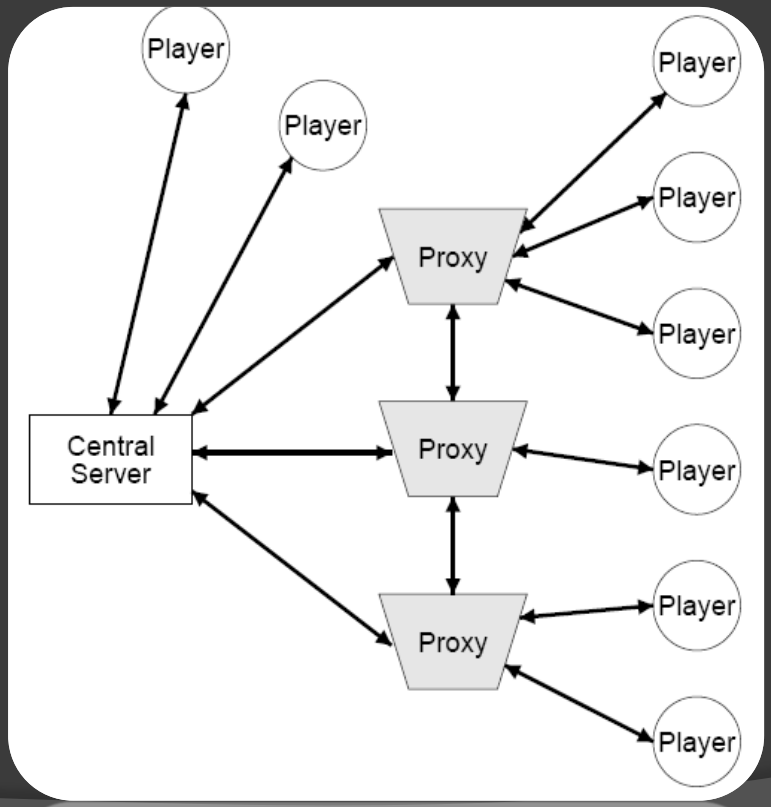
- **Objectivo**: Um mundo virtual contínuo e sem quebras com milhões de utilizadores concorrentes;

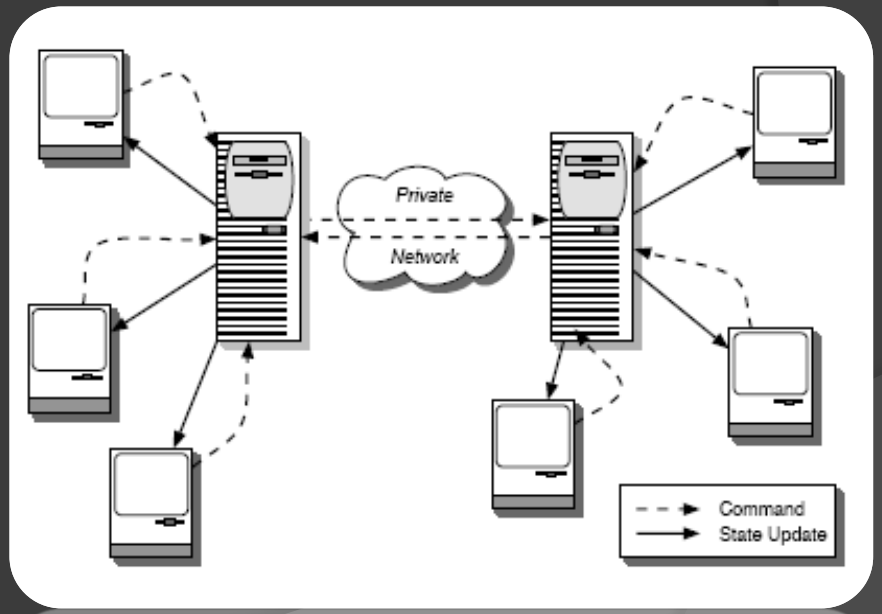
Desafios

- Heterogeneidade;
- Churn;
- Hacking;

MMOG Server Clusters

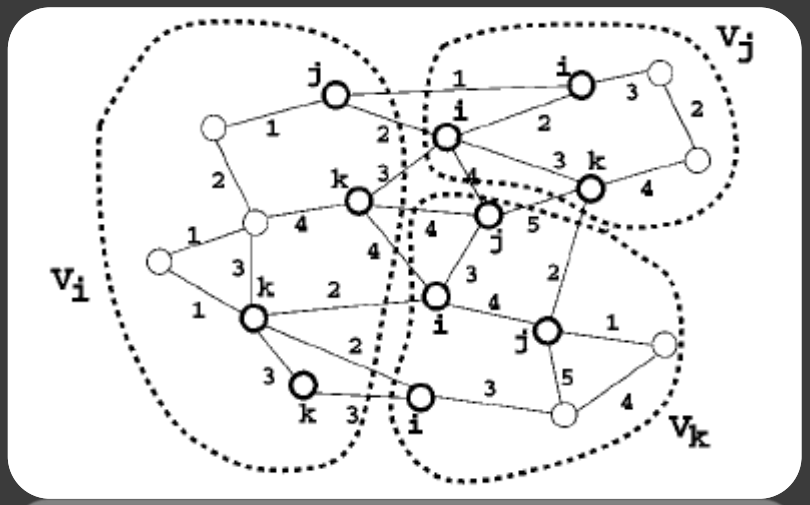
- Baseados em replicação (proxy server e mirror servers)

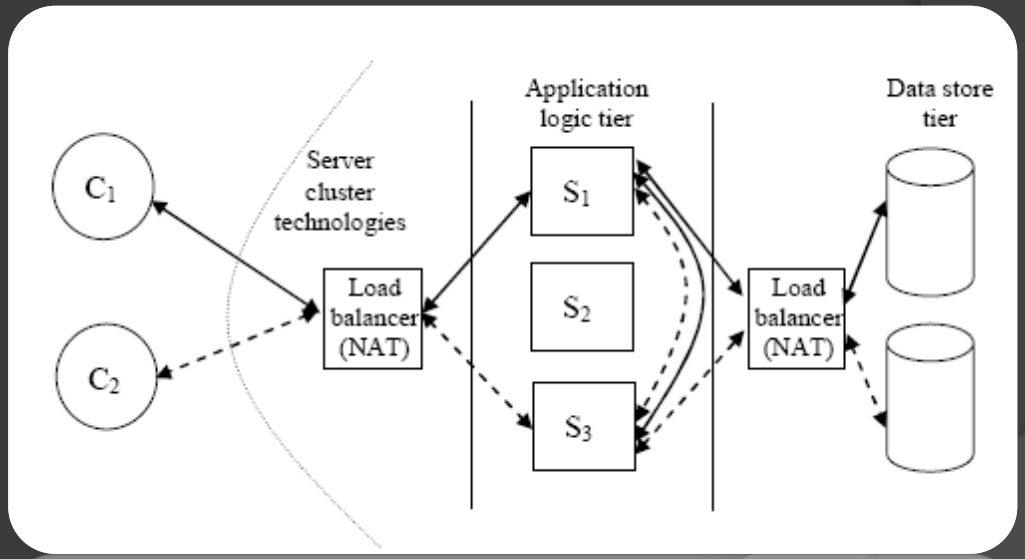




MMOG Server Clusters

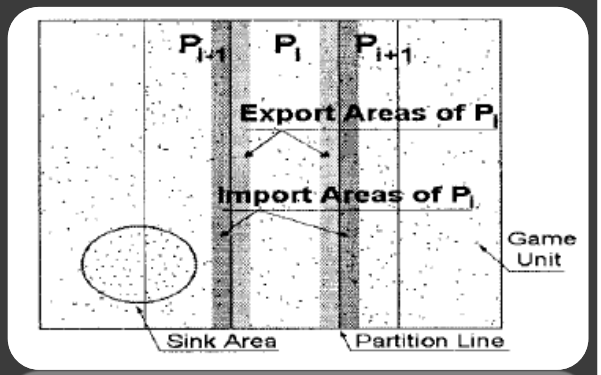
- Baseados em objectos

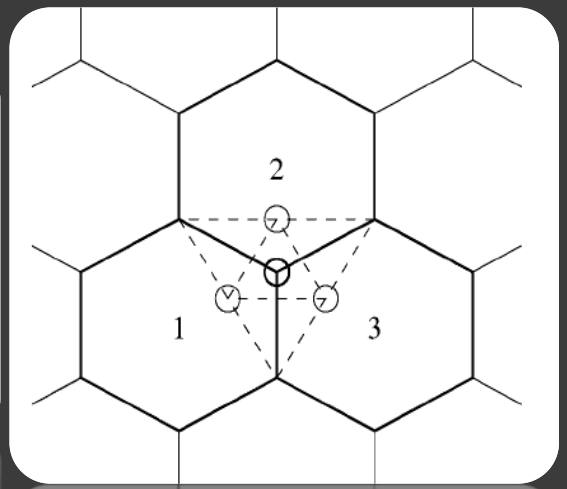


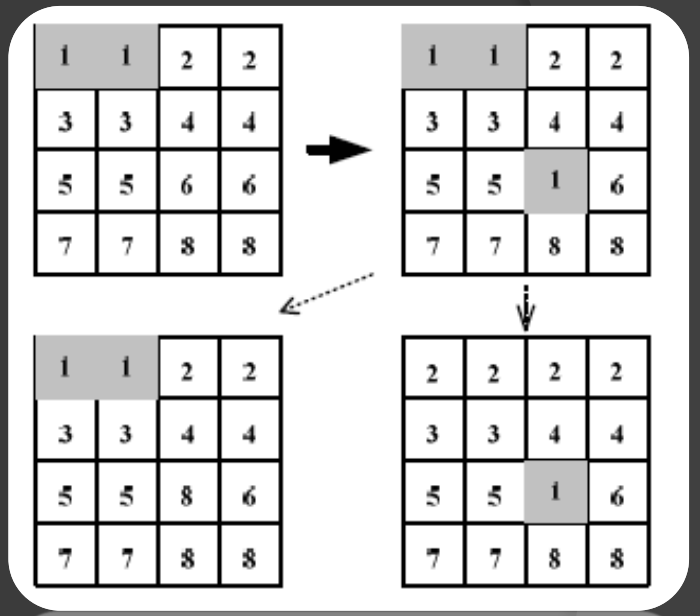


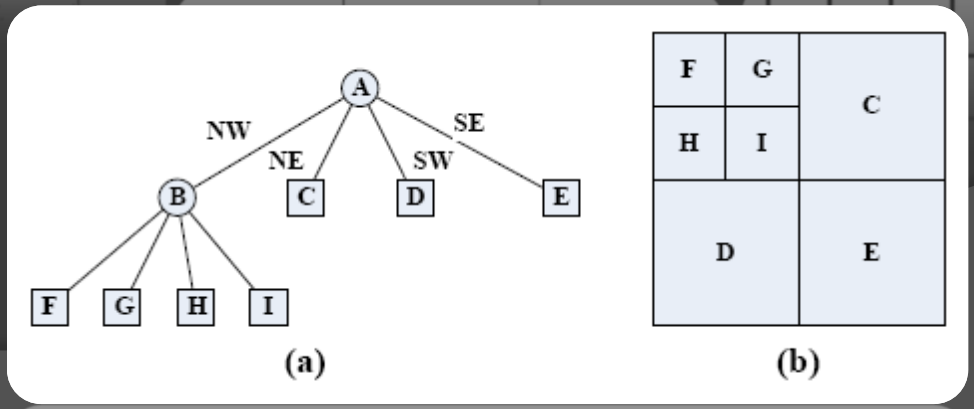
MMOG Server Clusters

- Baseados em zonas









Problemas dos MMOG Server Clusters

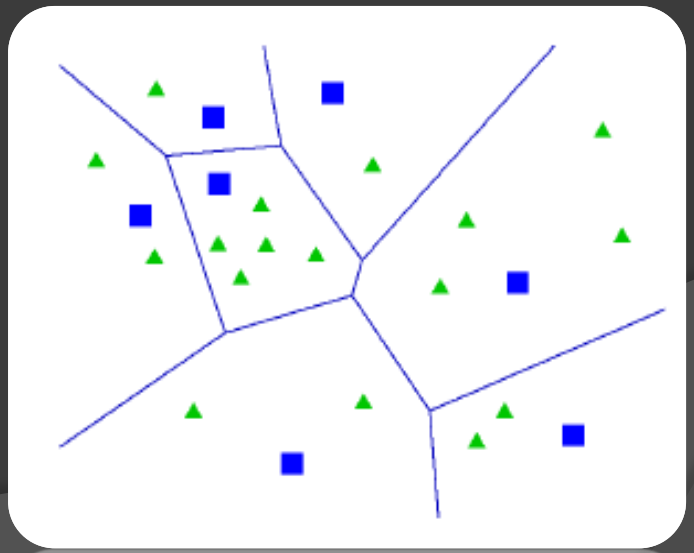
- Particionamento (estático, dinâmico)
- Balanceamento do “Load” (global vs. local)
- Problema principal:
  - “Computation load” vs. “Inter-server communication”
- Limitações Principais:
  - Escalabilidade (limitação pelo total dos recursos existentes)
  - Balanceamento do “Load” (densidade levada de jogadores – “hotspots”)

Voronoi State Management

- Assumpção: os estados são guardados em objectos (x;y)
- Ideia inicial:
  - Permitir que os estados de jogo sejam geridos por todos os clientes;
  - Dois papéis para cada cliente: peer e arbitrator;
  - Particionamento usando o algoritmo de Voronoi;

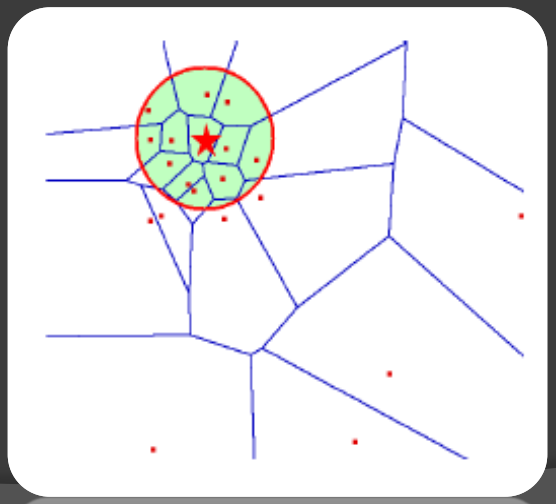
Voronoi State Management

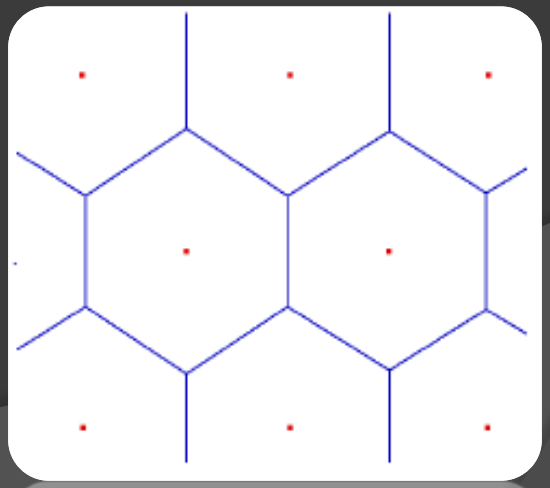
- Existem três problemas:
  - Conexões de ordem $O(n^2)$ nos “hotspots”;
  - Algumas células são maiores que as restantes;
  - Transferências de controlo constantes (ownership);



VSM: Ideias Base

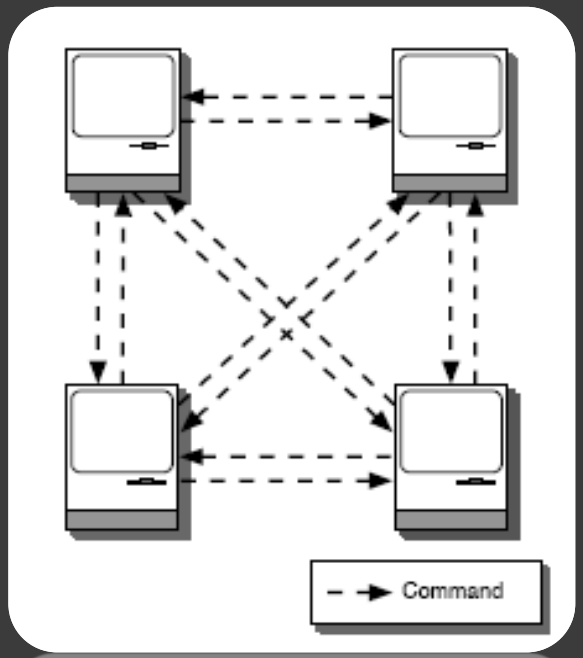
- “Overload” de conexão: clustering de “aggregators”
- Células grandes: virtual peers
- Transferências constantes: transferências explícitas;





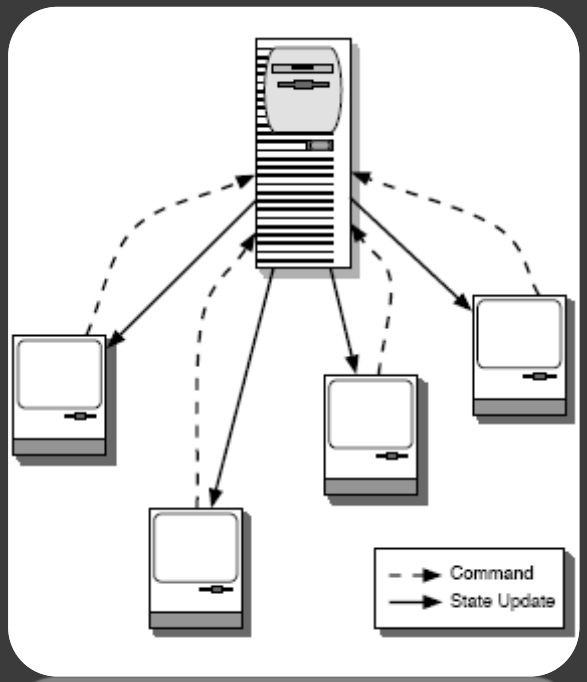
Modelos de Rede para Jogos

- Point-to-Point



Ex: RTS

- Cliente-servidor



Ex: FPS, MMOG

Modelos de Consistência para Jogos

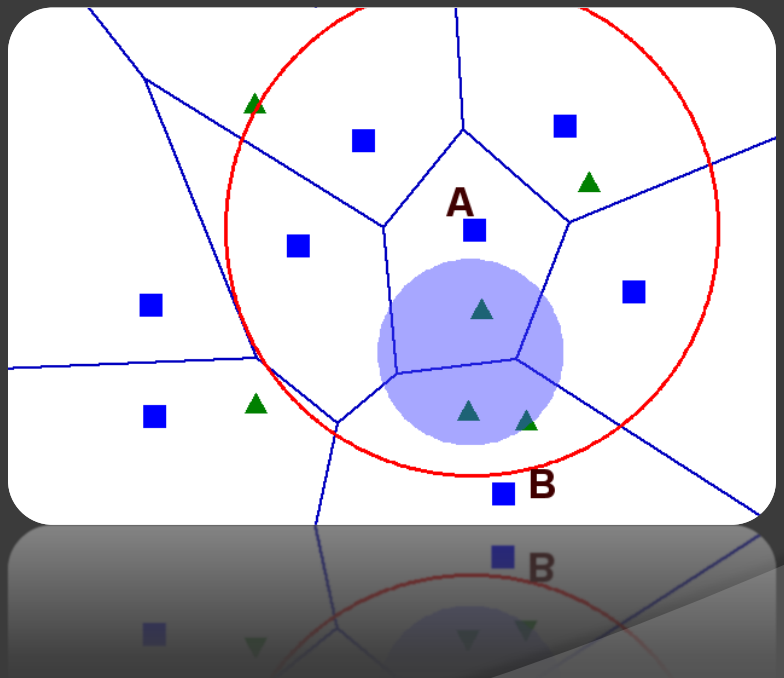
- Event-based (normalmente usado em point-to-point)
  - Eventos enviados para todos os nodos;
  - Nodos avançam em tempo lógico em conjunto;
  - Mesmos estados e a mesma execução de eventos;
- Update-based (normalmente usado em cliente-servidor)
  - Eventos enviados apenas para o nodo servidor;
  - Servidor faz avançar o tempo lógico;
  - Estados do servidor e sincronização dos clientes feita através de updates;

VSM: Controlo de Consistência

- Update-based;
- Eventos são enviados para o arbitrator em gestão;
- O arbitrator em gestão decide se dá seguimento;
- Cada arbitrator toma as suas decisões;
- Envia os updates;
- Fecho dos atributos durante as transacções:
  - O arbitrator A notifica o B (faz-se o fecho/lock);
  - A modifica os estados;
  - B modifica os estados, liberta o fecho;
  - A recebe uma confirmação, transacção efectuada;

VSM: Controlo de Consistência

- O arbitrator que está encarregue da gestão recebe e processa os eventos;
- Se necessário saltam-se eventos;
- Os updates são enviados para os arbitrators afectados por estes;



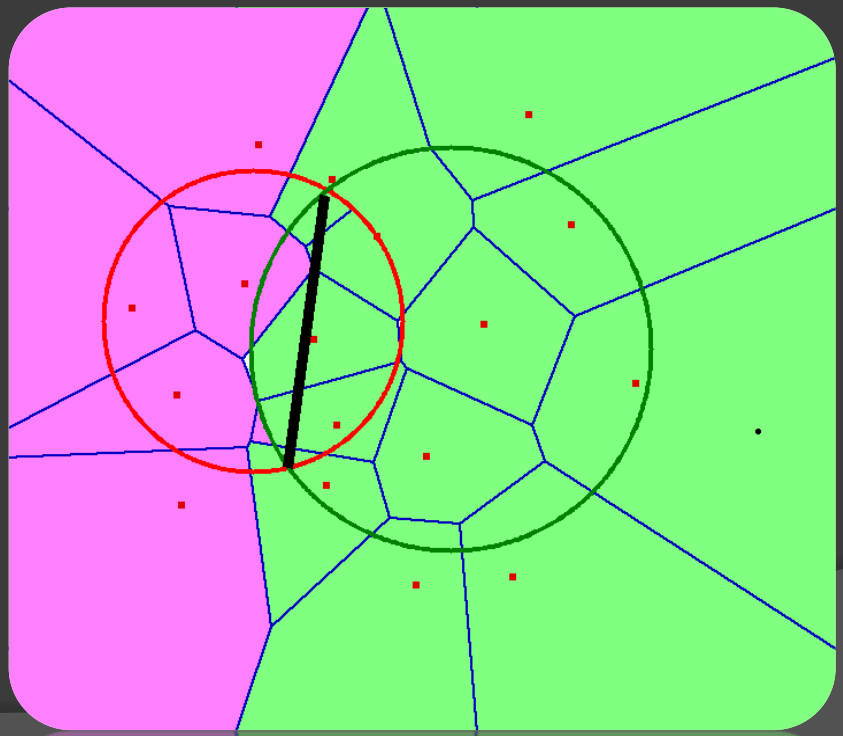
VSM: balanceamento do Load

- Tradicional: nodos com capacidade elevada primeiro, efectuar o ajustamento;
- VSM: nodos com capacidade baixa primeiro, efectuar o cluster de seguida;

- É assumido que se sabe quais os nodos a fazer “load”;
- Definição de “overload” e “underload”;
  - Overload: pede ao gateway o agregator, submete o controlo;
  - Underload: desintegra, liberta o controlo;

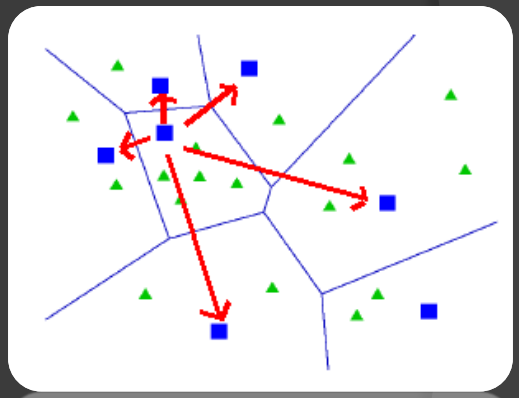
VSM: balanceamento do Load

- A esfera de controlo é ajustável;
- Caso exista mais do que um agregator, escolhe-se o mais próximo;



VSM: Tolerância a Falhas

- Arbitrator normal:
  - Escolhe um arbitrator de backup, efectua o backup dos estados;
  - O backup transfere o controlo para os arbitrators mais próximos;



- Aggregators:
  - Escolhe um aggregator de backup;
  - Assume o papel do original caso exista uma falha;
  - Escolhe um novo backup;

Conclusão

- VSM utiliza:
  - Particionamento Voronoi;
  - Controlo de consistência;
  - Clustering e superpeers (heterogeneidade);
  - Nodos de backup (churn);
- Propostas futuras:
  - Os aggregators movem-se com os nodos;
  - Separação de gestão estática de dinâmica de objectos;