***Concorrência em sistemas distribuídos e concorrência em sistema centralizado ou shared-memory concurrency***

Concorrência shared-memory – Os recursos estão no mesmo processo e são partilhados entre threads diferentes. Os mecanismos de controlo de concorrência podem ser locais e são disponibilizados pelo sistema operativo (Mutex, monitores, ...)

Concorrência (distribuida) centralizada – Os recursos estão em processos/máquinas diferentes e é necessário implementar algoritmos distribuidos para controlo de concorrência.

***Confiabilidade relacionada com sistema envolvendo múltiplos computadores (nós) distribuídos***

Um sistesma distribuido, é em principio, mais disponivel e mais tolerante a falhas que um sistema não distribuido. Contudo colocam-se problemas como a consistência eventual, coordenação de eventos, ordenação de eventos e todas as questões relacionadas a sistemas distribuidos que são pontos adicionais de falha.

***A falha, no paralelo entre centralizado (de RAM) e distribuído***

Um sistema distribuido é mais tolerante a falhas porque existe redundância, mas é mais dificil de recuperar porque o processo está a ser executado em diversos serviços.

Uma falha num sistema centralizado é mais fácil de recuperar.

***Latências introduzidas na comunicação (interação) entre nós***

A latência é inevitável entre sistemas distribuidos uma vez que implica a comunicação entre diversos serviços eventualmente hospedados em localizações diferentes. É importante verificar que existe disponibilizadade de acordo com a localização principal do publico alvo. E que os serviços que servem determinada população estão relativamente perto uns dos outros.

**Chamada local e RPC, aspetos relevantes em cada um dos modelos**

Uma chamada local deve ser em termos progrmáticos idêntica a uma chamada remota, de forma a garantir transparência. Os detalhes de comunicação com o serviço e envio de pacotes deve ser abstraidos da interface disponibilizada ao programador. Contudo o programador não se deve esquecer quando está a fazer chamadas remotas que têm uma latência maior que chamadas locais.

**A insolubilidade do problema dos dois generais e a sua abordagem no modelo RPC**

O problema dos dois generais é que nunca pode haver certeza total porque uma mensagem pode sempre falhar. Entrando os dois serviços numa confirmação infinita do ACK do ACK.