

**Curso Técnico em Informática Integrado ao Ensino Médio**

**Unidades curriculares:** Ferramentas de controle de versão

**Professor:** John

**Alunos:** Alisson Figueira, Enzo Zanon, Gabriele A. Kraminski, Guilherme Lando, João Vitor e Vicente Rauber

## **Introdução**

A TechCorp Soluções, empresa atuante no setor de tecnologia, enfrenta atualmente um desafio estratégico em seus processos de desenvolvimento: a falta de padronização no uso de sistemas de controle de versão. Enquanto algumas equipes utilizam modelos centralizados (CVCS), outras trabalham com sistemas distribuídos (DVCS). Essa heterogeneidade tem gerado dificuldades de integração, mobilidade entre projetos e perda de eficiência nos fluxos de trabalho.

Diante desse cenário, a Diretoria de Engenharia estabeleceu como prioridade a análise comparativa entre os dois modelos de versionamento, a fim de identificar a solução mais adequada para as necessidades da organização. Este documento, portanto, tem como objetivo apresentar uma avaliação detalhada dos sistemas centralizados e distribuídos, destacando suas principais características, vantagens e limitações.

Além da análise técnica, será elaborada uma proposta de rota de migração que ofereça à TechCorp um caminho estruturado e seguro para a padronização de suas práticas. Dessa forma, busca-se não apenas apoiar a tomada de decisão da diretoria, mas também promover ganhos de produtividade, colaboração e escalabilidade no ambiente de desenvolvimento da empresa.

## **Objetivo**

O objetivo deste estudo é realizar uma análise comparativa entre os sistemas de controle de versão centralizados (CVCS) e distribuídos (DVCS), destacando suas principais características, vantagens e desvantagens. A comparação busca fornecer subsídios técnicos e estratégicos que permitam à TechCorp tomar uma decisão fundamentada quanto à padronização de seus processos de versionamento.

Além disso, pretende-se propor uma rota de migração viável, que oriente a transição da empresa para o sistema mais adequado às suas demandas de eficiência, escalabilidade e colaboração entre equipes. Dessa forma, o documento não se limita a apresentar conceitos teóricos, mas também indica um caminho prático para a melhoria contínua do desenvolvimento de software na organização.

**Sistemas Centralizados (CVCS)**

### **O que são Sistemas Centralizados**

* Em um **sistema centralizado**, **toda a informação, processamento e controle** ficam concentrados em um **único servidor ou computador principal**.
* Os usuários (clientes/terminais) acessam esse servidor para obter dados ou executar tarefas.
* Exemplo clássico: **mainframes antigos** em bancos, ou até um servidor central que guarda todos os arquivos de uma empresa.

### **Características principais**

* **Controle único:** só existe uma fonte central que coordena tudo.
* **Gestão mais simples:** como está tudo em um ponto, backup, segurança e manutenção são centralizados.
* **Dependência alta:** se o servidor central cair, todo o sistema para.
* **Escalabilidade limitada:** aumentar a capacidade pode ser caro, já que depende de uma única máquina mais potente.

**Sistemas Distribuídos**

## **O que são Sistemas Distribuídos**

* Em um sistema distribuído, o **processamento, armazenamento e controle são divididos** entre vários computadores independentes conectados por uma rede.
* Esses computadores (ou nós) **trabalham juntos como se fossem um único sistema**, oferecendo serviços ou executando tarefas de forma colaborativa.
* Exemplo clássico: **serviços de nuvem** como Google Drive, Netflix, ou até redes como o BitTorrent.

## **Características principais**

* **Distribuição de tarefas:** cada máquina pode executar uma parte do processamento ou armazenar uma parte dos dados.
* **Escalabilidade horizontal:** é possível aumentar a capacidade do sistema apenas adicionando mais máquinas à rede.
* **Tolerância a falhas:** se um dos nós falhar, os outros podem continuar funcionando normalmente.
* **Concorrência:** múltiplos usuários podem acessar e interagir com o sistema ao mesmo tempo, sem depender de um único ponto.

## Recomendaria sistemas distribuídos quando:

* Você precisa de alta disponibilidade e o sistema não pode parar mesmo que uma parte falhe.
* Há grande volume de dados ou usuários e você precisa escalar de forma eficiente.
* A aplicação exige desempenho distribuído, como em sistemas de busca, plataformas de streaming, redes sociais, etc.
* Quer distribuir o sistema geograficamente, por exemplo, para atender usuários em diferentes regiões com baixa latência.

Recomendação de CVCS

* Equipes pequenas/médias
* Quando segurança e controle são prioritários
* Infraestrutura de rede estável
* Facilidade de gestão é importante