Ambiente Livre

Big Data & Data Science

SolrCloud

Indexação Clusterizada

Marcio Junior Vieira
CEO & Data Scientist, Ambiente Livre
Pesquisador da UNB.

Marcio Junior Vieira

Ambiente Livre
Big Data & Data Science

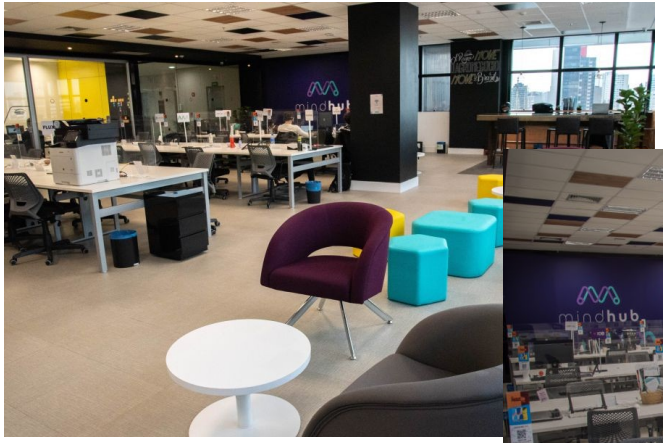
Mini-CV

- 22 anos de experiência em TI, vivência em desenvolvimento e análise de sistemas de gestão empresarial e ciência de dados.
- CEO da Ambiente Livre atuando como Cientista de Dados, Engenheiro de Dados e Arquiteto de Software.
- Professor dos MBAs em Big Data & Data Science, Inteligência Artificial e Business Intelligence e Analytics da Universidade Positivo.
- Professor do MBA Artificial Intelligence e Machine Learning da FIAP.
- Pesquisador do Laboratório de Tecnologias para Tomada de Decisão da Universidade de Brasília (Unb/Latitude).
- Trabalhando com Free Software e Open Source desde 2000 com serviços de consultoria e treinamento.
- Graduado em Tecnologia em Informática(2004) e pós-graduado em Software Livre(2005) ambos pela UFPR.
- Palestrante FLOSS em: FISL, TDC, Latinoware, Campus Party, Pentaho Day, Ticnova, PgDay e FTSL.
- Organizador Geral: Pentaho Day 2017, 2015, 2019 e apoio nas ed. 2013 e 2014.
- Data Scientist, instrutor e consultor de Big Data e Data Science com tecnologias abertas.
- Ajudou a capacitar equipes de Big Data na IBM, Accenture, Tivit, Serpro, Natura, MP, Netshoes, Embraer etc.
- Especialista em implantação e customização de Big Data com Hadoop, Spark, Pentaho, Cassandra e MongoDB.
- Contribuidor de projetos internacionais, tais como Pentaho, LimeSurvey, SuiteCRM e Camunda.
- Especialista em implantação e customização de ECM com Alfresco e BPM com Activiti, Flowable e Camunda.
- Certificado (Certified Pentaho Solutions) pela Hitachi Vantara (Pentaho).
- Membro da The Order Of de Bee (comunidade Alfresco para desenvolver o ecossistema Alfresco independente)

Sobre a Ambiente Livre

Open Software for Business

- Fundada em 2004 com foco em consultoria com FLOSS.
- Experts em 34 soluções para geração de negócios com Software Livre/Código Aberto.
- Atualmente estamos sediados no Hub de Inovação Mindhub em Curitiba (FAE).



Nosso Ecossistema de Serviços

Ambiente Livre
Big Data & Data Science

Big Data e Data Science

Análise de Dados da IoT
Análise Preditiva
Processamento Distribuído
Banco de Dados Colunares

Big Data & Data Lake
Big Data Analytics
Machine Learning

Consultoria | Treinamento | Projeto

CRM e CMS

Marketing e Vendas
Fidelização
SAC e Pós-vendas
Portais de Conteúdo

Customer Relationship Management
Content Management System
Pesquisa de Mercado & SLA

Consultoria | Treinamento | Projeto

ECM e BPM

Gestão de Documentos
Gerenciamento de Mídias
Processo de Negócio
BPMN e BPMS

Enterprise Content Management
Records Management
Business Process Management

Consultoria | Treinamento | Projeto

Business Intelligence

Painéis de Indicadores
Cubos de Análise
Relatórios Gerênciais
Tomada de Decisão

Business Intelligence & Analitycs
Dashboards e OLAP
Data Integration & Data Mining

Consultoria | Treinamento | Projetos

pentaho
Alfresco
hadoop
Apache HOP
Apache Superset
RANCHER
docker
kubernetes
LIFERAY
Enterprise. Open Source. For Life.
SCYLLA
Cassandra
suiteCRM
LPI
Spark
Apache Zeppelin
Grafana
Flowable
Activiti
LimeSurvey
PostgreSQL
the world's most advanced open source database
Apache Flink
Airflow
Metabase
zeebe
by Camunda
camunda
APACHE nifi
APACHE HBASE
mongoDB.
moodle
MariaDB
Drupal

Índices de Pesquisa - Índice Invertido

Ambiente Livre
Big Data & Data Science

Conceito

- Estrutura de dados de índice que armazena um mapeamento de conteúdo, como palavras ou números, para suas localizações em um documento ou conjunto de documentos.

- É um hashmap como uma estrutura de dados que o direciona de uma palavra para um documento ou página da web.

- Existem dois tipos de índices invertidos: Um índice invertido em nível de registro contém uma lista de referências a documentos para cada palavra. Um índice invertido em nível de palavra contém adicionalmente as posições de cada palavra em um documento.

Índices de Pesquisa - Índice Invertido

Índice invertido - Exemplo

- Suponha que queremos pesquisar os textos **“hello everyone”**, **“This article is based on inverted index”**, “**which is a data structure similar to a hashmap**”. Se indexarmos por (texto, palavra dentro do texto), o índice com localização no texto é:

```
hello                (1, 1)
everyone             (1, 2)
this                 (2, 1)
article              (2, 2)
is                   (2, 3); (3, 2)
based                (2, 4)
on                   (2, 5)
inverted             (2, 6)
index                (2, 7)
which                (3, 1)
hashmap              (3, 3)
like                 (3, 4)
data                 (3, 5)
structure            (3, 6)
```

- A palavra “hello” está no documento 1 (“hello everyone”) começando na palavra 1, então tem uma entrada (1, 1) e a palavra “is” está no documento 2 e 3 nas posições '3ª' e '2ª' respectivamente (aqui a posição é baseada na palavra).
- O índice pode ter pesos, frequências ou outros indicadores.

Índices de Pesquisa - Índice Invertido

Words	Document
ant	doc1
demo	doc2
world	doc1, doc2

Passos para construir um índice invertido:

- Busque o documento.
- Removendo palavras de parada: são as palavras mais comuns e inúteis em documentos como “eu”, “o”, “nós”, “é”, “um”.
- Derivação da palavra raiz.
- Sempre que desejo pesquisar “gato”, desejo ver um documento que contém informações sobre ele. Mas a palavra presente no documento é chamada de “gatos” ou “maliciosos” em vez de “gato”. Para relacionar as duas palavras, cortarei alguma parte de cada palavra que leio para que possa obter a “palavra raiz”. Existem ferramentas padrão para fazer isso, como “Porter's Stemmer”.
- Registre IDs de documentos.
- Se a palavra já estiver presente, adicione a referência do documento para indexar, caso contrário, crie uma nova entrada. Adicione informações adicionais, como frequência da palavra, localização da palavra, etc.

Índices de Pesquisa - Índice Invertido

Ambiente Livre
Big Data & Data Science

Vantagens

- O índice invertido permite pesquisas rápidas de texto completo, a um custo de processamento aumentado quando um documento é adicionado ao banco de dados.
- É fácil de desenvolver.
- É a estrutura de dados mais popular usada em sistemas de recuperação de documentos, usados em grande escala, por exemplo, em motores de busca.

Desvantagens

- Grande sobrecarga de armazenamento e altos custos de manutenção na atualização, exclusão e inserção.

API Open Source

- Você coloca documentos nele (chamado de "indexação") via JSON, XML, CSV ou binário sobre HTTP.
- Você consulta via HTTP GET e recebe JSON, XML, CSV ou binário

- Adiciona novos documentos ao index.
- Deleta documentos do index.
- Constrói querys.
- Busca no index usando querys.

Apache Solr - Funcionalidades

Ambiente Livre
Big Data & Data Science

Recursos avançados de pesquisa de texto completo

Alimentado por Lucene ™ , o Solr permite recursos de correspondência poderosos, incluindo frases, curingas, junções, agrupamentos e muito mais em qualquer tipo de dados

Otimizado para tráfego de alto volume

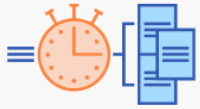
Solr é comprovado em escalas extremamente grandes em todo o mundo

Interfaces abertas baseadas em padrões - XML, JSON e HTTP

O Solr usa as ferramentas que você usa para facilitar a criação de aplicativos

Flexível e adaptável com fácil configuração

O Solr's foi projetado para se adaptar às suas necessidades enquanto simplifica a configuração



Indexação quase em tempo real

Quer ver suas atualizações agora? O Solr aproveita os recursos de indexação quase em tempo real do Lucene para garantir que você veja seu conteúdo quando quiser.

Interfaces de administração abrangentes

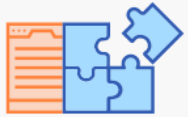
O Solr vem com uma interface de usuário administrativa responsiva integrada para facilitar o controle de suas instâncias do Solr

Fácil Monitoramento

Precisa de mais informações sobre suas instâncias? Solr publica cargas de dados métricos via JMX

Altamente escalável e tolerante a falhas

Construído no Apache Zookeeper testado em batalha, o Solr facilita o aumento e a redução. Solr trabalha em replicação, distribuição, rebalanceamento e tolerância a falhas fora da caixa.



Arquitetura de plug-in extensível

O Solr publica muitos pontos de extensão bem definidos que facilitam o plug-in de plug-ins de índice e de tempo de consulta. Obviamente, como é um código aberto licenciado pela Apache, você pode alterar qualquer código que desejar!

Apache Solr – Funcionalidades Estendidas

Esquema quando quiser, sem esquema quando não quiser

Use o modo sem esquema orientado a dados do Solr ao começar e, em seguida, bloqueie-o quando chegar a hora da produção.

Extensões poderosas

O Solr vem com plug-ins opcionais para indexação de conteúdo rico (por exemplo, PDFs, Word), detecção de idioma, agrupamento de resultados de pesquisa e muito mais

Pesquisa facetada e filtragem

Fatie e divida seus dados como achar melhor usando uma grande variedade de algoritmos de facetação

Pesquisa geoespacial

A ativação da pesquisa baseada em localização é simples com o suporte integrado do Solr para pesquisa espacial

Análise de texto configurável avançada

O Solr é fornecido com suporte para a maioria dos idiomas falados no mundo (inglês, chinês, japonês, alemão, francês e muitos outros) e muitas outras ferramentas de análise projetadas para tornar a indexação e consulta de seu conteúdo o mais flexível possível

Cache altamente configurável e extensível pelo usuário

Controles refinados nos caches integrados do Solr facilitam a otimização do desempenho

Análise Rica de Documentos

O Solr é fornecido com o Apache Tika integrado, facilitando a indexação de conteúdo rico, como Adobe PDF, Microsoft Word e muito mais.

Vários índices de pesquisa

O Solr oferece suporte a arquiteturas multilocatários, facilitando o isolamento de usuários e conteúdo.

Otimizações de desempenho

O Solr foi ajustado para lidar com os maiores sites do mundo

Segurança integrada

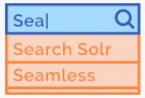
Solr seguro com SSL, autenticação e autorização baseada em função. Plugável, é claro!

Opções avançadas de armazenamento

Com base nos recursos avançados de armazenamento do Lucene (codecs, diretórios e mais), o Solr facilita o ajuste de suas necessidades de armazenamento de dados para atender ao seu aplicativo

Registro Monitorável

Acesse facilmente os arquivos de log do Solr na interface de administração



Sugestões de consulta, ortografia e muito mais

O Solr vem com recursos avançados para preenchimento automático (pesquisa de digitação antecipada), verificação ortográfica e muito mais

Seus dados, do seu jeito!

JSON, CSV, XML e mais são suportados imediatamente. Não perca tempo precioso convertendo todos os seus dados em uma representação comum, basta enviá-los para o Solr!

Projeto Apache Lucene™

- Desenvolve software de pesquisa de código aberto. O projeto lança uma biblioteca de pesquisa central, denominada **Lucene Core™**, bem como **PyLucene**, uma integração de Python com Lucene.
- **Lucene Core** é uma biblioteca Java que oferece recursos avançados de indexação e pesquisa, bem como verificação ortográfica, realce de ocorrências e recursos avançados de análise/tokenização.
- O Solr era um subprojeto do Lucene, e se tornou um Projeto de Nível Superior (TLP).
- Tecnologia adequada para praticamente qualquer aplicativo que exija pesquisa estruturada, pesquisa de texto completo, facetamento, pesquisa de vizinho mais próximo em vetores de alta dimensionalidade, correção ortográfica ou sugestões de consulta.

Apache Lucene - Funcionalidades

Ambiente Livre
Big Data & Data Science

Indexação escalável e de alto desempenho

- Mais de 800 GB/hora em hardware moderno.
- Pequenos requisitos de RAM - apenas 1 MB de heap.
- Indexação incremental tão rápida quanto a indexação em lote.
- Tamanho do índice aproximadamente 20-30% do tamanho do texto indexado.

Apache Lucene - Funcionalidades

Algoritmos de pesquisa poderosos, precisos e eficientes

- Pesquisa classificada -- melhores resultados retornados primeiro
- Muitos tipos de consulta: consultas de frase, consultas curinga, consultas de proximidade, consultas de intervalo e muito mais.
- Pesquisa por campo (por exemplo, título, autor, conteúdo)
- Pesquisa de vizinho mais próximo para vetores de alta dimensionalidade
- Classificação por qualquer campo.
- Pesquisa de índice múltiplo com resultados mesclados.
- Permite atualização e busca simultâneas.
- Facetamento flexível, realce, junções e agrupamento de resultados
- Sugestões rápidas, eficientes em termos de memória e tolerantes a erros de digitação
- Modelos de classificação conectáveis, incluindo o Vector Space Model e o Okapi BM25
- Mecanismo de armazenamento configurável (codecs)

Empresas que Solr – No Mundo

Empresas que Solr – No Brasil

Ambiente Livre
Big Data & Data Science

MP DFT

TRIBUNAL DE JUSTIÇA
S P
3 DE FEVEREIRO DE 1874

sei!
SEI - Sistema Eletrônico de Informações

MPPR
Ministério Público do Paraná

ESTADO DO RIO GRANDE DO SUL
PODER JUDICIÁRIO
TRIBUNAL DE JUSTIÇA

MTi 50 ANOS
Empresa Mato-grossense de Tecnologia da Informação

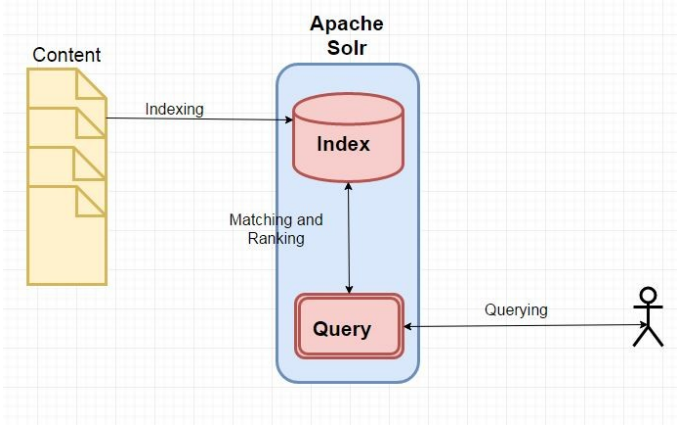
MP PE
Ministério Público de Pernambuco

TJDFT
TRIBUNAL DE JUSTIÇA DO DISTRITO FEDERAL E DOS TERRITÓRIOS

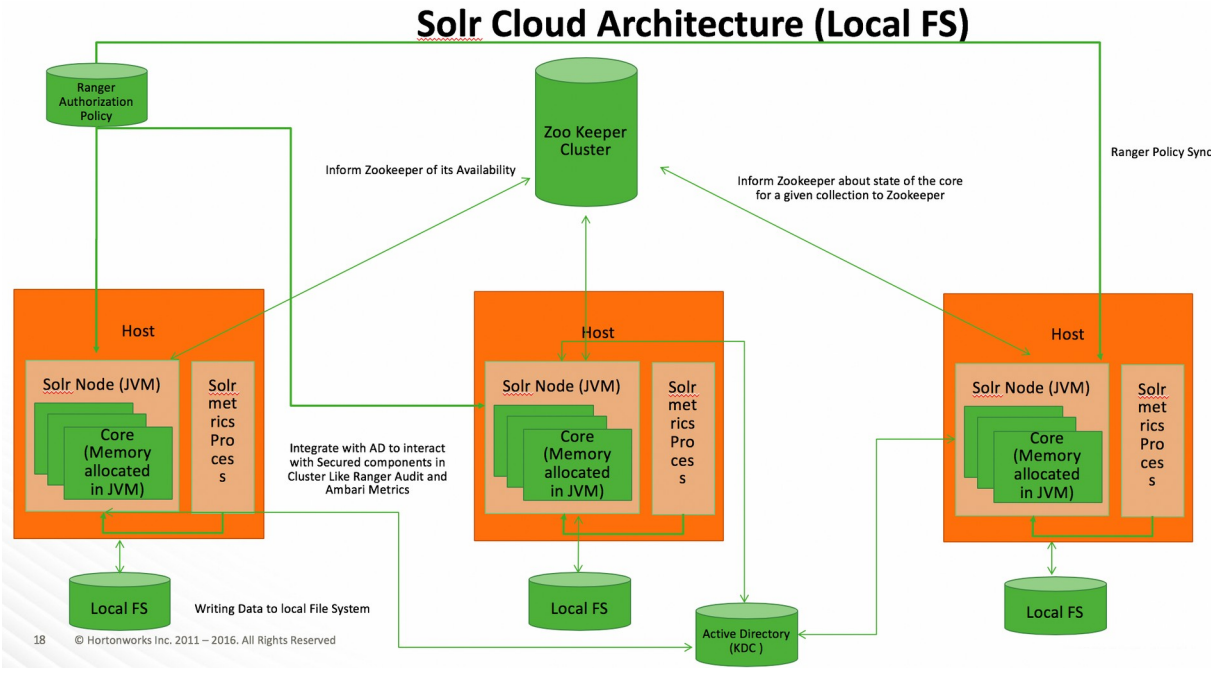
Tipos de Cluster Solr

Tipos

- Quando não estamos em cluster chamamos de Solr **Standalone mode**.
- Quando configuramos um cluster chamamos de **SolrCloud Mode**.



Standalone mode.



Tipos de Cluster Modes

- Um cluster Solr é um grupo de servidores (**nodes**) em que cada um executa o Solr.

- Existem dois modos gerais de operação de um **cluster** de nós Solr. Um modo fornece coordenação central dos nós Solr (**SolrCloud Mode**), enquanto o outro permite operar um cluster sem essa coordenação central (**User-Managed Mode**).

- Ambos os modos compartilham conceitos gerais, mas diferem em última análise na forma como esses conceitos são refletidos na funcionalidade e nos recursos.

Shards

Conceito

- Em ambos os modos de cluster, um único índice lógico pode ser dividido entre nós como shards. Cada shard contém um subconjunto do índice geral.

- O número de shards determina o limite teórico para o número de documentos que podem ser indexados no Solr. Também determina a quantidade de paralelização possível para uma solicitação de pesquisa individual.

Replicas

Conceito

- Para fornecer algum failover para cada shard, cada shard pode ser copiado como uma réplica. Uma réplica tem a mesma configuração que o shard e quaisquer outras réplicas para o mesmo índice.

- É possível ter réplicas sem ter criado shards. Neste caso, cada réplica seria uma cópia completa de todo o índice, em vez de ser apenas uma cópia de parte de todo o índice.

- O número de réplicas determina o nível de tolerância a falhas que todo o cluster possui no caso de falha de um node. Também determina o limite teórico do número de solicitações de pesquisa simultâneas que podem ser processadas sob carga pesada.

Leaders

Conceito

- Depois que as réplicas forem criadas, um líder(Leaders) deverá ser identificado.

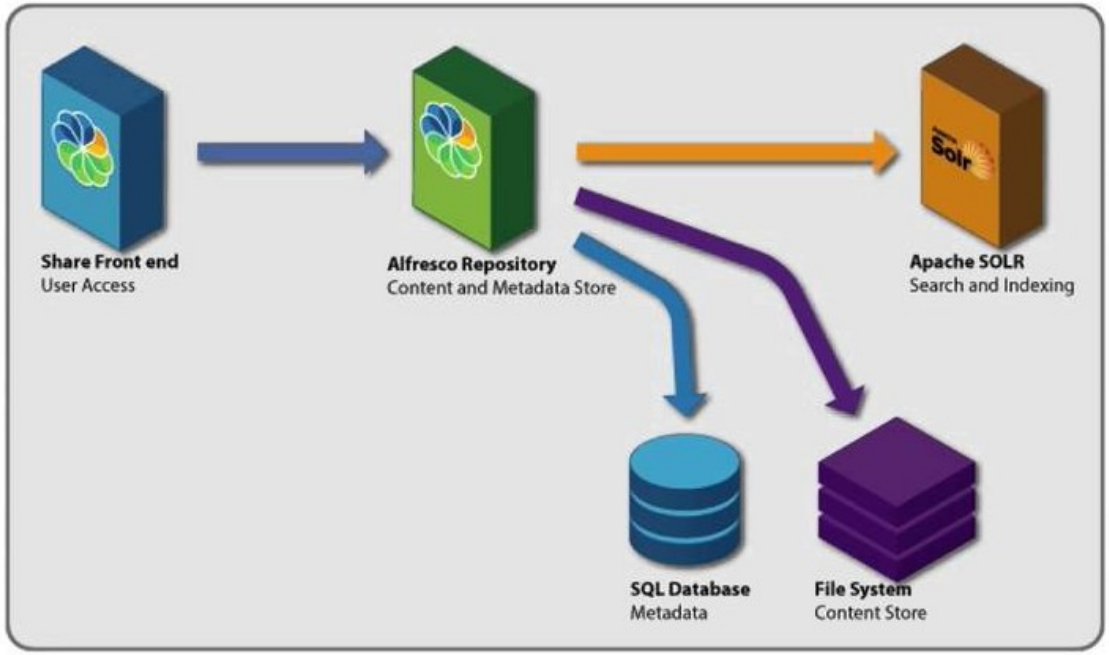
- A responsabilidade do líder é ser uma fonte de verdade para cada réplica. Quando são feitas atualizações no índice, elas são processadas primeiro pelo líder e depois por cada réplica.

- As réplicas que não são **leaders** são **followers**.

Alfresco Search Services 2.0

Alfresco Search

- Um modelo de Configuração do Solr focado para o Alfresco Content Service.
- Solr versão 6.x (SolrClod somente Solr 6.6).



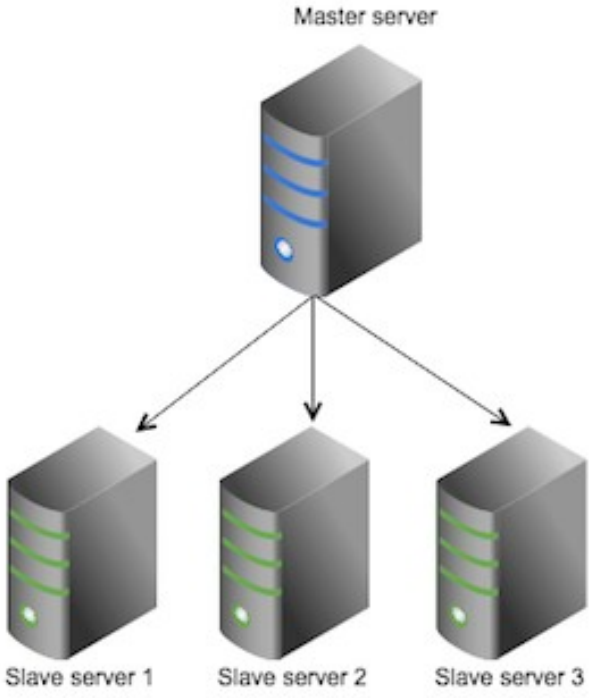
Solr replication

Replicação Solr

- Usa o modelo master-worker para distribuir cópias completas de um índice mestre para um ou mais servidores worker.

- O mestre recebe todas as atualizações e todas as alterações são feitas. As alterações feitas no mestre são distribuídas a todos os servidores workers que atendem todas as solicitações de consulta dos clientes.

- Todos os rastreadores devem ser habilitados nos nós mestres, enquanto apenas o rastreador de modelo e o rastreador de metadados devem ser habilitados nos escravos.



Solr Config - Alfresco Content Services - Master

Ambiente Livre
Big Data & Data Science

/solrhome/templates/re-rank/conf/solrconfig.xml

```
 <requestHandler name="/replication"
class="org.alfresco.solr.handler.AlfrescoReplicationHandler" >
        <lst name="master">
          <str name="replicateAfter">commit</str>
          <str name="replicateAfter">startup</str>
          <str name="confFiles">schema.xml,stopwords.txt</str>
 </requestHandler>
```

- **solrcore.properties**
  enable.master=false
  enable.slave=true

Solr Config - Alfresco Content Services - Worker

Ambiente Livre
Big Data & Data Science

/solrhome/templates/re-rank/conf/solrconfig.xml

```
 <requestHandler name="/replication"
class="org.alfresco.solr.handler.AlfrescoReplicationHandler" >
        <lst name="slave">
          <str name="masterUrl">http://your-master-hostname:8983/solr</str>
          <str name="pollInterval">00:00:60</str>
        </lst>
 </requestHandler>
```

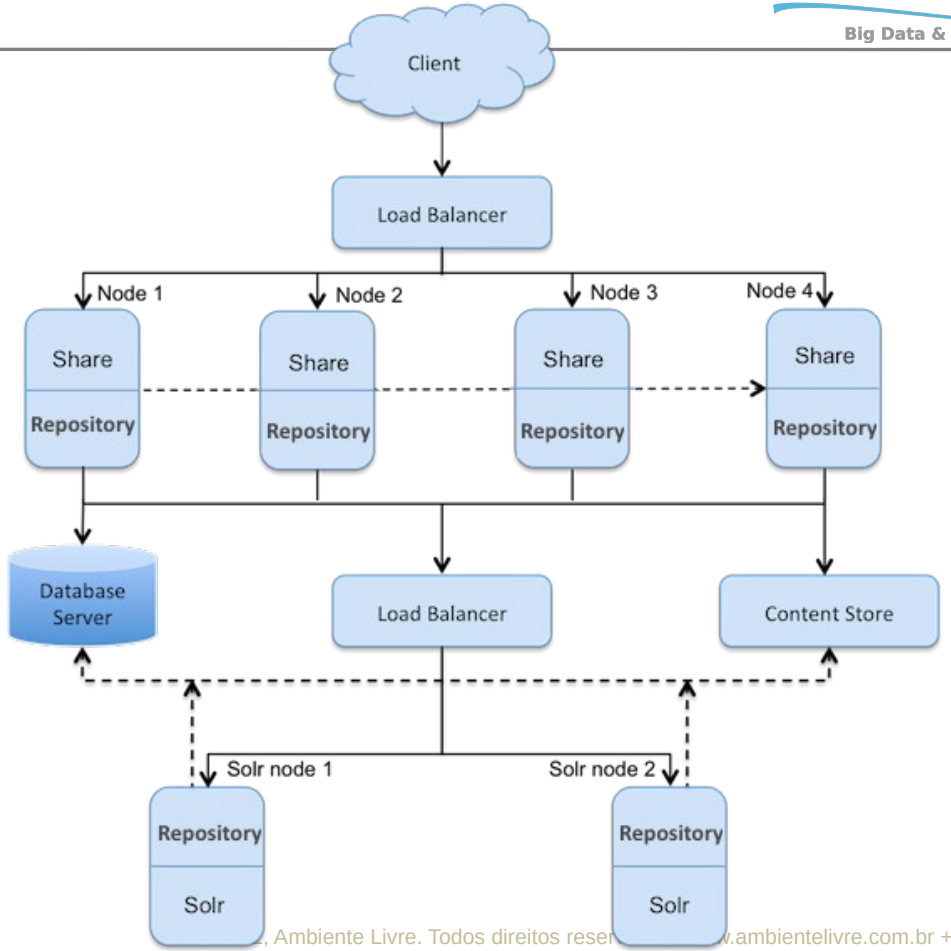
- **solrcore.properties**
  enable.master=false
  enable.slave=true

Alfresco em Cluster com Solr - Master-Master

Ambiente Livre
Big Data & Data Science

Replicação Solr

- Master-Master.
- Simplicidade de configuração.
- Apache Active MQ (Opcional)



Obrigado

Marcio Junior Vieira
marcio@ambientelivre.com.br
@marviojvieira @ambientelivre
https://www.linkedin.com/in/mvieira1/
Blog: http://blogs.ambientelivre.com.br/marcio/

Youtube: https://www.youtube.com/c/AmbienteLivreOpenSoftware

Ambiente Livre
Big Data & Data Science