MIGRANDO O ATOM 2.5.3 PARA 2.8.2

Para migrar para a versão 2.8.2 é necessário Ubuntu 20.04 LTS. A versão 2.5.3 está instalado no Ubuntu 18.04. Iremos aproveitar para atualizar e migrar para um novo servidor.

INSTALAR AS DEPENDÊNCIAS

MySQL

AtoM 2.8 requer MySQL 8.0 ou superior, pois usa expressões de tabela comuns. Além disso, tivemos resultados muito bons usando o Percona Server para MySQL 8.0, então não tenha medo e use-o se quiser!

```
sudo apt update
sudo apt install mysql-server
```

Por fim, vamos configurar nossos modos MySQL. O servidor MySQL pode operar em diferentes modos SQL, o que afeta a sintaxe SQL que o MySQL suporta e as verificações de validação de dados que ele executa.

Cole os seguintes valores em um novo arquivo em /etc/mysql/conf.d/mysqld.cnf e salve:

```
[mysqld]
sql_mode=ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
optimizer_switch='block_nested_loop=off'
```

Agora vamos reiniciar o MySQL:

```
sudo systemctl restart mysql
```

Elasticsearch

Um servidor de busca baseado no Apache Lucene e desenvolvido em Java que trouxe ao AtoM muitos recursos avançados, desempenho e escalabilidade. Esta é provavelmente a maior mudança introduzida no AtoM 2.x e estamos satisfeitos com os resultados.

O Ubuntu não fornece um pacote, mas você pode baixá-lo diretamente do site do Elasticsearch se não conseguir baixá-lo usando o método a seguir.

Certifique-se de que o Java esteja instalado. Neste exemplo, usaremos o OpenJDK, mas a JVM da Oracle também funcionaria.

```
sudo apt install openjdk-11-jre-headless apt-transport-https software-properties-common
```

Após instalar o Java com sucesso, prossiga para instalar o Elasticsearch. Baixe e instale a chave de assinatura pública usada no repositório deles:

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
```

Agora adicione o repositório deles:

```
echo "deb https://artifacts.elastic.co/packages/5.x/apt stable main" | sudo tee -a
/etc/apt/sources.list.d/elastic-5.x.list
```

Pronto para ser instalado. Execute:

```
sudo apt update
sudo apt install elasticsearch
```

Inicie o serviço e configure-o para iniciar quando o sistema for inicializado.

```
sudo systemctl enable elasticsearch
sudo systemctl start elasticsearch
```

PHP

O Ubuntu 20.04 agrupa o PHP 7.4, que é muito mais rápido do que versões mais antigas. O comando a seguir o instalará junto com o restante das extensões PHP necessárias pelo AtoM:

```
sudo apt install php-common php7.4-common php7.4-cli php7.4-curl php7.4-json php7.4-ldap
php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml php7.4-mbstring php7.4-xsl php7.4-zip
php-apcu php-apcu-bc
```

Se você estiver usando o Memcached como mecanismo de cache, você também precisará instalar o php-memcache:

```
sudo apt install php-memcache
```

Servidor de trabalho Gearman

O servidor de trabalho Gearman é exigido pelo AtoM a partir da versão 2.2.

```
sudo apt install gearman-job-server
```

Outros pacotes

Para gerar auxílios de busca em PDF, o AtoM requer que o Apache FOP esteja instalado. Felizmente, o Apache FOP agora pode ser instalado diretamente de pacotes do Ubuntu usando o comando abaixo.

O comando especificado abaixo usa o parâmetro --no-install-recommends: isso é intencional e garante que apenas dependências sejam instaladas e não pacotes 'recomendados'. Se --no-install-recommends não for especificado, o openjdk-8-jre será instalado como uma dependência para um dos pacotes recomendados. Como o openjdk-8-jre-headless já foi instalado na seção de instalação do Elasticsearch acima, queremos evitar a instalação do pacote openjdk-8-jre também.

```
sudo apt install --no-install-recommends fop libsaxon-java
```

Certifique-se de que o comando java padrão aponta para o binário java versão 11 (ignore erros):

```
sudo update-java-alternatives -s java-1.11.0-openjdk-amd64
```

Se você quiser que o AtoM seja capaz de processar objetos digitais em formatos como JPEG ou extrair o texto de seus documentos PDF, há certos pacotes que você precisa instalar. Eles não são obrigatórios, mas se forem encontrados no sistema, o AtoM os usará para produzir derivados de objetos digitais a partir de seus objetos mestres. para mais informações sobre cada um, consulte: Requisitos: outras dependências. O seguinte instalará todas as dependências recomendadas de uma vez:

```
sudo apt install imagemagick ghostscript poppler-utils ffmpeg
```

Baixar o AtoM

Agora que instalamos e configuramos todas as dependências, estamos prontos para baixar e instalar o AtoM em si. A maneira mais segura é instalar o AtoM a partir do tarball, que você pode encontrar na seção de download. No entanto, usuários experientes podem preferir verificar o código do nosso repositório público.

As instruções a seguir pressupõem que estamos instalando o AtoM em /usr/share/nginx e que você está usando o AtoM 2.8.0.

```
wget https://storage.accesstomemory.org/releases/atom-2.8.2.tar.gz
sudo mkdir /usr/share/nginx/atom
sudo tar xzf atom-2.2.0.tar.gz -C /usr/share/nginx/atom --strip 1
```

Crie o banco de dados

Supondo que você esteja executando o MySQL no localhost, crie o banco de dados executando o seguinte comando usando a senha que você criou anteriormente:

```
sudo mysql -h localhost -u root -p -e "CREATE DATABASE atom CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;"
```

Observe que o banco de dados foi chamado de atom. Sinta-se à vontade para mudar seu nome.

Caso seu servidor MySQL não seja o mesmo que seu servidor web, substitua “localhost” pelo endereço do seu servidor MySQL.

Além disso, é sempre uma boa ideia criar um usuário MySQL específico para o AtoM para manter as coisas mais seguras. É assim que você pode criar um usuário chamado atom com senha 12345 e as permissões necessárias para o banco de dados criado acima.

```
sudo mysql -h localhost -u root -p -e "CREATE USER 'atom'@'localhost' IDENTIFIED BY '<omitida>';"
sudo mysql -h localhost -u root -p -e "GRANT ALL PRIVILEGES ON atom.* TO 'atom'@'localhost';"
```

Note que os privilégios INDEX, CREATE e ALTER são necessários somente durante o processo de instalação ou quando você estiver atualizando o AtoM para uma versão mais nova. Eles podem ser removidos do usuário quando você terminar a instalação ou você pode alterar o usuário usado pelo AtoM em config.php.

Restaurando a base do banco de dados

Importante lembrar que é necessário o dump do banco para realizar a restauração.

```
sudo mysql -h localhost -u root -p atom < ~/atom.sql
```

MIGRANDO O INDICE NO ELASTICSEARCH

CONFIGURANDO O CAMINHO DO REPOSITÓRIO

O repositório dos snapshots no SciELO é um ponto de montagem NFS. Este ponto NFS deve ser montado no novo servidor. Mas antes instale o pacote nfs-utils:

```
sudo yum install nfs-utils -y
```

Crie o diretório do repositório:

```
sudo mkdir /var/lib/elasticsearch/backup-repo
```

Configure o /etc/fstab para persistir:

```
storage-scielo-251.scielo.org:/atom_es_bkp /var/lib/elasticsearch/backup-repo nfs
vers=4,rsize=8192,wsize=8192,timeo=14,intr 0 0
```

Agora basta montar:

```
sudo systemctl daemon-reload
sudo mount -a
```

ADICIONADO O ELASTICSEARCH NA CONFIGURAÇÃO

Edite o arquivo /etc/elasticsearch/elasticsearch.yml e adicione a linha abaixo de path.logs:

```
path.repo: /var/lib/elasticsearch/backup-repo
```

Reinicie o elasticsearch

```
sudo systemctl restart elasticsearch
```

Devemos garantir que o usuário elasticsearch seja o dono do diretório /var/lib/elasticsearch/backup-repo:

```
sudo chown elasticsearch. -R /var/lib/elasticsearch/backup-repo
```

CRIANDO O REPOSITÓRIO

```
curl -XPUT -H 'Content-Type: application/json'
'http://localhost:9200/_snapshot/es_backup_atom' -d '{
    "type": "fs",
    "settings": {
        "compress" : true,
        "location": "/var/lib/elasticsearch/backup-repo/es_backup_atom"
    }
 }'
```

Consulte o repositório:

```
curl -X GET "localhost:9200/_cat/snapshots/es_backup_atom?v&s=id&pretty"
```

```
root@atom:/usr/share/nginx/atom# curl -X GET "localhost:9200/_cat/snapshots/es_backup_atom?v&s=id&pretty"
id               status start_epoch start_time end_epoch  end_time duration indices successful_shards failed_shards total_shards
20240911-025924 SUCCESS 1726023564  02:59:24   1726023564 02:59:24     72ms       1                 4             0            4
20240912-025922 SUCCESS 1726109962  02:59:22   1726109963 02:59:23    118ms       1                 4             0            4
20240913-025919 SUCCESS 1726196359  02:59:19   1726196359 02:59:19     86ms       1                 4             0            4
20240914-025916 SUCCESS 1726282756  02:59:16   1726282756 02:59:16    109ms       1                 4             0            4
20240915-025916 SUCCESS 1726369156  02:59:16   1726369156 02:59:16    112ms       1                 4             0            4
20240916-025918 SUCCESS 1726455558  02:59:18   1726455558 02:59:18    125ms       1                 4             0            4
20240917-025916 SUCCESS 1726541956  02:59:16   1726541956 02:59:16     93ms       1                 4             0            4
20240918-025945 SUCCESS 1726628385  02:59:45   1726628385 02:59:45    149ms       1                 4             0            4
20240919-025915 SUCCESS 1726714755  02:59:15   1726714756 02:59:16    282ms       1                 4             0            4
20240920-025919 SUCCESS 1726801159  02:59:19   1726801159 02:59:19    113ms       1                 4             0            4
root@atom:/usr/share/nginx/atom#
```

RESTAURANDO O ÚLTIMO SNAPSHOT

```
curl -X POST "localhost:9200/_snapshot/es_backup_atom/20240920-025919/_restore?pretty"
```

LISTANDO OS INDICES RESTAURADOS

```
curl -X GET "http://localhost:9200/_cat/indices?v"
```

```
root@atom:/usr/share/nginx/atom# curl -X GET "http://localhost:9200/_cat/indices?v"
health status index uuid                   pri rep docs.count docs.deleted store.size pri.store.size
yellow open   atom  C_7dYc0QSteOElrp41MoRQ   4   1       1410            6      3.4mb          3.4mb
root@atom:/usr/share/nginx/atom#
```

Baixar o AtoM

Agora que instalamos e configuramos todas as dependências, estamos prontos para baixar e instalar o AtoM em si. A maneira mais segura é instalar o AtoM a partir do tarball, que você pode encontrar na seção de download. No entanto, usuários experientes podem preferir verificar o código do nosso repositório público.

As instruções a seguir pressupõem que estamos instalando o AtoM em /usr/share/nginx e que você está usando o AtoM 2.8.0.

```
wget https://storage.accesstomemory.org/releases/atom-2.8.2.tar.gz
sudo mkdir /usr/share/nginx/atom-new
sudo tar xzf atom-2.2.0.tar.gz -C /usr/share/nginx/atom-new --strip 1
```

Copie seus dados antigos

Agora, vamos copiar o conteúdo dos diretórios antigos de uploads e downloads, bem como o banco de dados. Vamos primeiro mover a instalação 2.5.3 renomeando:

```
sudo mv /usr/share/nginx/atom /usr/share/nginx/atom-old
```

Vamos mover a nova versão:

```
sudo mv /usr/share/nginx/atom-new /usr/share/nginx/atom
```

Vamos copiar a pasta uploads e downloads para a nova instalação:

```
rsync -av /usr/share/nginx/atom-old/uploads/ /usr/share/nginx/atom/uploads/
rsync -av /usr/share/nginx/atom-old/downloads/ /usr/share/nginx/atom/downloads/
```

Executando a tarefa de atualização

Esta é talvez a etapa mais crítica no processo de atualização. Se você encontrar algum erro, consulte nosso Fórum de Usuários ou, se não encontrar uma solução, sinta-se à vontade para postar uma pergunta lá você mesmo. Também tentaremos adicionar ao nosso FAQ conforme recebermos feedback, para ajudar os usuários a solucionar quaisquer problemas de atualização encontrados.

```
cd /usr/share/nginx/atom
php -d memory_limit=-1 symfony tools:upgrade-sql
```

Regenerar a referência do objeto digital e as imagens em miniatura

Primeiro, certifique-se de que você não alterou o diretório (/usr/share/nginx/atom).

Agora, execute a tarefa regen-derivatives:

```
php symfony digitalobject:regen-derivatives
```

Reconstruir índice de pesquisa e limpar cache

Para que todas essas alterações entrem em vigor, você precisará reindexar os arquivos que importou para seu banco de dados e limpar o cache do aplicativo.

Primeiro, reconstrua o índice de pesquisa:

```
php -d memory_limit=-1 symfony search:populate
```

Resultado:

```
Index populated with 1214 documents in 15.7 seconds.
The following errors have been encountered:
Couldn't find information object (id: 2705)
Please, contact an administrator.
```

Em seguida, limpe o cache para remover quaisquer dados desatualizados do aplicativo:

```
php symfony cc
```

Permissões do sistema de arquivos

Por padrão, o Nginx é executado como o usuário www-data. Existem alguns diretórios no AtoM que devem ser graváveis pelo servidor web. A maneira mais fácil de garantir isso é atualizar o proprietário do diretório AtoM e seu conteúdo executando:

```
sudo chown -R www-data:www-data /usr/share/nginx/atom
```

Se você estiver implantando o AtoM em um ambiente compartilhado, recomendamos que preste atenção às permissões atribuídas a outros. A seguir, um exemplo de como limpar todos os bits de modo para outros:

```
sudo chmod o= /usr/share/nginx/atom
```

Se você estiver planejando fazer uploads DIP do AM, verifique a seção do diretório de depósito do SWORD para definir as permissões desse diretório.

Implantação de workers

Gearman é usado no AtoM para dar suporte a tarefas assíncronas, algumas das quais são funcionalidades principais, como atualizar o status de publicação de uma hierarquia descritiva, mover descrições para um novo registro pai e muito mais. Um worker é apenas uma tarefa CLI que você pode executar em um terminal ou supervisionar com ferramentas específicas como upstart, supervisord ou systemd. O worker aguardará por jobs atribuídos pelo servidor de jobs.

Usaremos systemd para gerenciar o worker do AtoM; crie o seguinte arquivo de serviço /usr/lib/systemd/system/atom-worker.service:

```
[Unit]
Description=AtoM worker
After=network.target
# High interval and low restart limit to increase the possibility
# of hitting the rate limits in long running recurrent jobs.
StartLimitIntervalSec=24h
StartLimitBurst=3

[Install]
WantedBy=multi-user.target

[Service]
Type=simple
User=www-data
Group=www-data
WorkingDirectory=/usr/share/nginx/atom
```

```
ExecStart=/usr/bin/php7.4 -d memory_limit=-1 -d error_reporting="E_ALL" symfony jobs:worker
KillSignal=SIGTERM
Restart=on-failure
RestartSec=30
```

Agora recarregue o systemd, habilite e inicie o AtoM worker:

```
sudo systemctl daemon-reload
sudo systemctl enable atom-worker
sudo systemctl start atom-worker
```

PHP-FPM

Nossa maneira favorita de implementar o AtoM é usando PHP-FPM, um gerenciador de processos que escala melhor do que outras soluções como FastCGI.

```
sudo apt install php7.4-fpm
```

Vamos adicionar um novo pool PHP para AtoM adicionando o seguinte conteúdo em um novo arquivo chamado /etc/php/7.4/fpm/pool.d/atom.conf:

```
[atom]

; The user running the application
user = www-data
group = www-data

; Use UNIX sockets if Nginx and PHP-FPM are running in the same machine
listen = /run/php7.4-fpm.atom.sock
listen.owner = www-data
listen.group = www-data
listen.mode = 0600

; The following directives should be tweaked based in your hardware resources
pm = dynamic
pm.max_children = 30
pm.start_servers = 10
pm.min_spare_servers = 10
pm.max_spare_servers = 10
pm.max_requests = 200

chdir = /

; Some defaults for your PHP production environment
; A full list here: http://www.php.net/manual/en/ini.list.php
php_admin_value[expose_php] = off
php_admin_value[allow_url_fopen] = on
php_admin_value[memory_limit] = 512M
php_admin_value[max_execution_time] = 120
php_admin_value[post_max_size] = 72M
```

```
php_admin_value[upload_max_filesize] = 64M
php_admin_value[max_file_uploads] = 10
php_admin_value[cgi.fix_pathinfo] = 0
php_admin_value[display_errors] = off
php_admin_value[display_startup_errors] = off
php_admin_value[html_errors] = off
php_admin_value[session.use_only_cookies] = 0

; APC
php_admin_value[apc.enabled] = 1
php_admin_value[apc.shm_size] = 64M
php_admin_value[apc.num_files_hint] = 5000
php_admin_value[apc.stat] = 0

; Zend OPcache
php_admin_value[opcache.enable] = 1
php_admin_value[opcache.memory_consumption] = 192
php_admin_value[opcache.interned_strings_buffer] = 16
php_admin_value[opcache.max_accelerated_files] = 4000
php_admin_value[opcache.validate_timestamps] = 0
php_admin_value[opcache.fast_shutdown] = 1

; This is a good place to define some environment variables, e.g. use
; ATOM_DEBUG_IP to define a list of IP addresses with full access to the
; debug frontend or ATOM_READ_ONLY if you want AtoM to prevent
; authenticated users
env[ATOM_DEBUG_IP] = "10.10.10.10,127.0.0.1"
env[ATOM_READ_ONLY] = "off"
```

The process manager has to be enabled and started:

```
sudo systemctl enable php7.4-fpm
sudo systemctl start php7.4-fpm
```

If the service fails to start, make sure that the configuration file has been has been pasted properly. You can also check the syntax by running:

```
sudo php-fpm7.4 --test
```

If you are not planning to use the default PHP pool (`www`), feel free to remove it:

```
sudo rm /etc/php/7.4/fpm/pool.d/www.conf
sudo systemctl restart php7.4-fpm
```

Nginx

In Ubuntu, the installation of Nginx is simple:

```
sudo apt install nginx
```

Warning

These instructions assume that the Nginx package is creating the directory `/usr/share/nginx` and that is the location where we are going to place the AtoM sources. However, we have been told this location may be different in certain environments (e.g. `/var/www`) or you may opt for a different location. If that is the case, please make sure that you update the configuration snippets that we share later in this document according to your location.

Nginx deploys a default server (aka VirtualHost, for Apache users) called **default** and you can find it in `/etc/nginx/sites-available/default`. In order to install AtoM you could edit the existing server block or add a new one. We are going to you show you how to do the latter:

```
sudo touch /etc/nginx/sites-available/atom
sudo ln -sf /etc/nginx/sites-available/atom /etc/nginx/sites-enabled/atom
sudo rm /etc/nginx/sites-enabled/default
```

We have now created the configuration file and linked it from sites-enabled/, which is the directory that Nginx will look for. This means that you could disable a site by removing its symlink from sites-enabled/ while keeping the original one under sites-available/, in case that you want to re-use it in the future. You can do this with the Nginx default server.

The following is a recommended server block for AtoM. Put the following contents in `/etc/nginx/sites-available/atom`.

Warning

This example listens for connections on port 80 using basic http without encryption.

While this is ok for testing AtoM locally on a private network, any public implementation of AtoM should be secured using TLS/SSL certificates such that your content is served over HTTPS.

The Mozilla SSL Configuration Generator is useful for assisting with adding the appropriate blocks to your Nginx configuration file.

```
upstream atom {
   server unix:/run/php7.4-fpm.atom.sock;
}

server {

   listen 80;
   root /usr/share/nginx/atom;

   # http://wiki.nginx.org/HttpCoreModule#server_name
```

```
    # _ means catch any, but it's better if you replace this with your server
    # name, e.g. archives.foobar.com
    server_name _;

    client_max_body_size 72M;

    location ~* ^/(css|dist|js|images|plugins|vendor)/.*\.(css|png|jpg|js|svg|ico|gif|pdf|woff|tt

    }

    location ~* ^/(downloads)/.*\.(pdf|xml|html|csv|zip|rtf)$ {

    }

    location ~ ^/(ead.dtd|favicon.ico|robots.txt|sitemap.*)$ {

    }

    location / {
       try_files $uri /index.php?$args;
       if (-f $request_filename) {
          return 403;
       }
    }

    location ~* /uploads/r/(.*)/conf/ {

    }

    location ~* ^/uploads/r/(.*)$ {
       include /etc/nginx/fastcgi_params;
       set $index /index.php;
       fastcgi_param SCRIPT_FILENAME $document_root$index;
       fastcgi_param SCRIPT_NAME $index;
       fastcgi_pass atom;
    }

    location ~ ^/private/(.*)$ {
       internal;
       alias /usr/share/nginx/atom/$1;
    }

    location ~ ^/(index|qubit_dev)\.php(/|$) {
       include /etc/nginx/fastcgi_params;
       fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
       fastcgi_split_path_info ^(.+\.php)(/.*)$;
       fastcgi_pass atom;
    }

}
```

Agora você precisa habilitar e recarregar o Nginx:

```
sudo systemctl enable nginx
sudo systemctl reload nginx
```

CRIANDO O USUÁRIO PARA INTEGRAR O ATOM AO ARCHIVEMATICA

```
useradd -c "Archivematica User" archivematica
```

Altere o bash editando o /etc/passwd

```
archivematica:x:1001:1001:Archivematica User:/home/archivematica:/usr/bin/rssh
```

Ajuste a permissão do diretório /DIPs que foi copiado do servidor de origem:

```
chown -R archivematica. /DIPs/
```

CRIANDO A RELAÇÃO DE CONFIANÇA

No diretório home (/home/archivematica) dentro da pasta .ssh crie o arquivo authorized_keys

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDLmna1yXOa5xFRuTgx8HeaGxqy3b3H5EksPIeVm+pfSElbrs8gnnrz3vFzYLve2c
sLP0QDnq6mq9zS8O1h5OAKNN1oKJ8C1o0kvcB7XDHwSqAU8SJ48c3mTWl7OSqRUznyvdXFL37qnpeF7SA0BUVYRb1CIrpi
ZLg6btrV6tLgcdF3LnqvJUuWjLXLOhIj6dU7TW9h1HpLB2ZmpkkWc0LIY1kUrJA/w9w2HnE7Qc0UjEldlhOMN/Bpc2wLi9
Cs3ZA6iRwnbV7f3M86R/0ZspRfNA6lg0ZId44AGS5OxB5QhU8nLsfXfj2PgCeUtWlwEQVPTE055crz8JlxXI7P2Rnl
archivematica@archivematica.scielo.org
```

Ajuste a permissão:

```
chmod 400 /home/archivematica/.ssh/authorized_keys
```

Revision #3
Created 23 September 2024 17:08:03 by Rondineli G. Saad
Updated 23 September 2024 18:02:14 by Rondineli G. Saad