

Uso do XNAT

Instalação, tecnologias e alternativas

André Meneghelli Vale

Semestre 2014-1

O objetivo desta apresentação é introduzir a ferramenta XNAT, analisando seus pontos positivos e negativos , estratégias complementares ou até mesmo estratégias substitutas.

É importante observar que a licença de uso principalmente isenta a responsabilidade sobre os dados salvos aos desenvolvedores e garante que aplicativos desenvolvidos a partir do XNAT tenham referência aos desenvolvedores originais apenas com uma permissão por escrito.

Tópicos:

- 1 Instalação
- 2 Tecnologias envolvidas
- 3 Pontos positivos
- 4 Pontos negativos
- 5 Alternativas suplementares

Existem dois modos de instalação.

- 1 VMs
- 2 Manualmente

A principal vantagem na utilização das VMs é a facilidade de instalação. Mais rápida e sem necessidade de instalação de todas as tecnologias extras envolvidas. No entanto, torna-se necessário configurar corretamente cada nova instância. Atualmente temos disponível quatro tipos de sistema operacional para as VMs:

- ❶ 32-bit Ubuntu Desktop 2.6 GB
- ❷ 32-bit Ubuntu Server 2.7 GB
- ❸ 64-bit Ubuntu Desktop 4.5 GB
- ❹ 64-bit Ubuntu Server 3.8 GB

A instalação manual gera mais trabalho para se ter uma máquina funcionando rapidamente, mas evita o processo de pós configuração.

O XNAT é uma aplicação java que roda sobre o TOMCAT e utiliza um banco de dados em POSTGRESQL. As etapas de instalação e configuração não são muito complicadas e não devem tomar muito mais tempo que o processo de configuração de uma VM.

A melhor estratégia para lidar com a instalação do XNAT deve ser uma mescla das duas opções disponíveis, criando uma VM própria com instalação manual, tornando-a mais segura que um exemplo de aplicação disponível na WEB e minimizando ao máximo as necessidades de pós-configuração.

Além das tecnologias envolvidas com a instalação do XNAT temos alguns APIs e aplicativos disponíveis em um Marketplace:

- 1 CSV Importer - permite que tabelas CSV sejam importadas para o XNAT;
- 2 XNAT Calendar - gerencia os dados dos experimentos com um calendário;
- 3 Face Masking - ferramenta utilizada para anonimizar imagens de rostos de dispositivos MR;
- 4 XNAT REST API - API de comunicação genérica que utiliza RESTful Web Services do próprio XNAT;
- 5 PyXNAT - API de comunicação para scripts python que utiliza RESTful Web Services do próprio XNAT.

- 1 Ferramenta consolidada para armazenamento de arquivos do tipo DICOM;
- 2 Grande capacidade de extensão do aplicativo original por plugins, APIs e até mesmo outros aplicativos;
- 3 Já possui recursos para a anonimização dos exames;
- 4 Capacidade de trabalhar com múltiplos dispositivos;
- 5 Pipelines podem facilitar algumas tarefas administrativas;
- 6 Boa documentação.

Pontos negativos

- 1 Necessita de uma porta e configuração manual para cada dispositivo;
- 2 Embora existam algumas formas de trabalhar com outros tipos de arquivos, este não é o padrão e as aplicações são escaças;
- 3 O arquivamento dos arquivos DICOM serão apenas temporários;
- 4 Provável necessidade de extensão da aplicação para converter alguns tipos de dados.

- 1 Extensão da capacidade de trabalho do XNAT para poder armazenar outros tipos arquivos, NIFT por exemplo;
- 2 Desenvolvimento de um Wrapper para evitar a utilização de multiplas portas;
- 3 Desenvolvimento de API similar a do XNAT caso seja interessante trabalhar com outros tipos de protocolo;

Existem vários meios de ter uma alternativa substituta ao XNAT para tratar do armazenamento da informação. O protocolo DICOM atual é capaz de informar qual o dispositivo de origem do arquivo. O que pode diminuir a necessidade de administração do serviço todas as vezes que um novo dispositivo seja instalado.

Por este não ser o padrão original do DICOM então não deve ser utilizado por uma solução genérica.

Uma solução alternativa é o desenvolvimento de uma arquitetura padrão no sistema de arquivos. Armazenando dados e metadados de maneira consciente e substituindo qualquer tipo complexo de conexão, seja por API ou por sshfs por uma simples conexão FTP.

Exemplo:

- `\hospital1\departamento1\dispositivo1\exame1`
- `\hospital1\departamento1\dispositivo1\exame1.json`
- `\hospital1\departamento1\dispositivo1\exame1.jpeg`
- `\hospital1\departamento1\dispositivo1\exame1.nift`