

Utilizando a API RXTX para manipulação da serial

Instalando e utilizando a API de comunicação serial RXTX para leitura e escrita em portas seriais para Windows e Linux

O Java tem uma grande vantagem frente a outras plataformas de desenvolvimento que é a possibilidade de um mesmo software ser executado em diversas plataformas sob uma mesma compilação, não é necessário reescrever ou nem mesmo recompilar o software para que este esteja disponível em outras plataformas (sejam elas Windows, Linux, Mac ou até mesmo PDA's de gama alta).

Mas toda essa facilidade tem certo custo, com Java a tarefa de chamar APIs nativas dos sistemas operacionais ou ainda fazer comunicação com o hardware diretamente se torna um pouco mais complexo. Mas para isso a Sun e demais empresas envolvidas no desenvolvimento Java disponibilizam diversas APIs para facilitar o trabalho, como é o caso da API RXTX para a comunicação tanto serial quanto paralela (existe também a possibilidade de comunicação via USB com esta mesma API).

A API RXTX é baseada na API Javacomm distribuída pela própria Sun, com a vantagem sobre esta de que ela é portátil para Linux, Windows e Mac, enquanto que a Javacomm em sua atual versão só é portátil para Linux engessando de certa forma o processo de desenvolvimento para múltiplas plataformas.

Obtendo a API RXTX

O primeiro passo é acessar o site <http://www.rxtx.org> e ir para a seção de download e localizar a API compilada *rxtx-2.1-7-bins-r2.zip (Final)*, se preferir também existe a possibilidade de baixar somente o código-fonte e fazer a compilação, mas isso é um procedimento mais complexo que por hora será deixado de lado. Atualmente a RXTX se encontra na versão 2.1.

A **figura 01** mostra a tela de download da API RXTX no site do seu desenvolvedor.

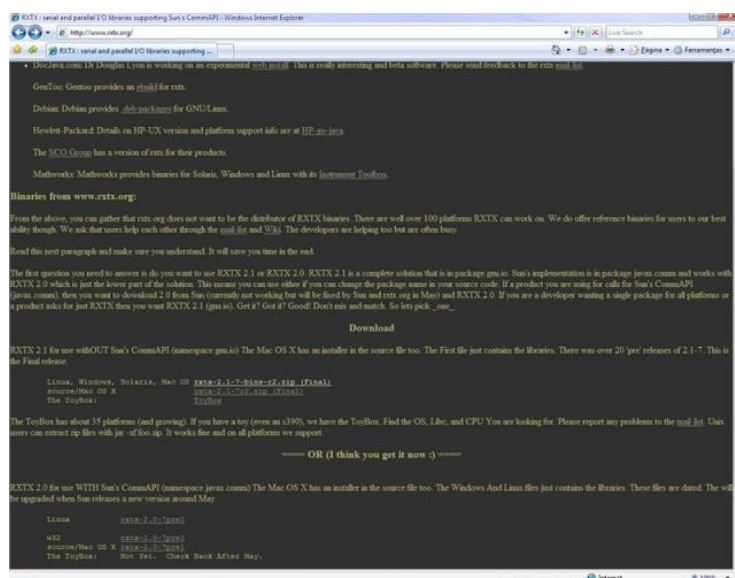


Figura 01: obtendo a API RXTX.

Instalando a API RXTX

Geralmente desenvolvedores que nunca trabalharam com esta API esquecem algum procedimento na sua instalação e devido a isso não obtém êxito no seu software na leitura serial ou paralela, dessa forma é muito importante o máximo de atenção, pois apesar de simples se alguns dos passos descritos não forem seguidos às chances de insucesso são grandes.

O processo de instalação será efetuado no Windows XP porém o processo é o mesmo para outros sistemas operacionais da família Windows ou até mesmo para as plataformas Linux e MAC OS, a única diferenciação será quanto aos arquivos de biblioteca de recursos, que no Windows são arquivos DLL, no Linux são arquivos SO e no MAC OS são arquivos JNILIB.

A primeira coisa a se fazer é extrair o arquivo em formato ZIP para dentro de alguma pasta, eu recomendo fortemente que a pasta escolhida para a extração seja a mesma pasta em que o JDK está instalado, pois isso irá facilitar o trabalho de manutenção e facilitará a sua instalação. A **figura 02** mostra a extração da API para a pasta de instalação do SDK.

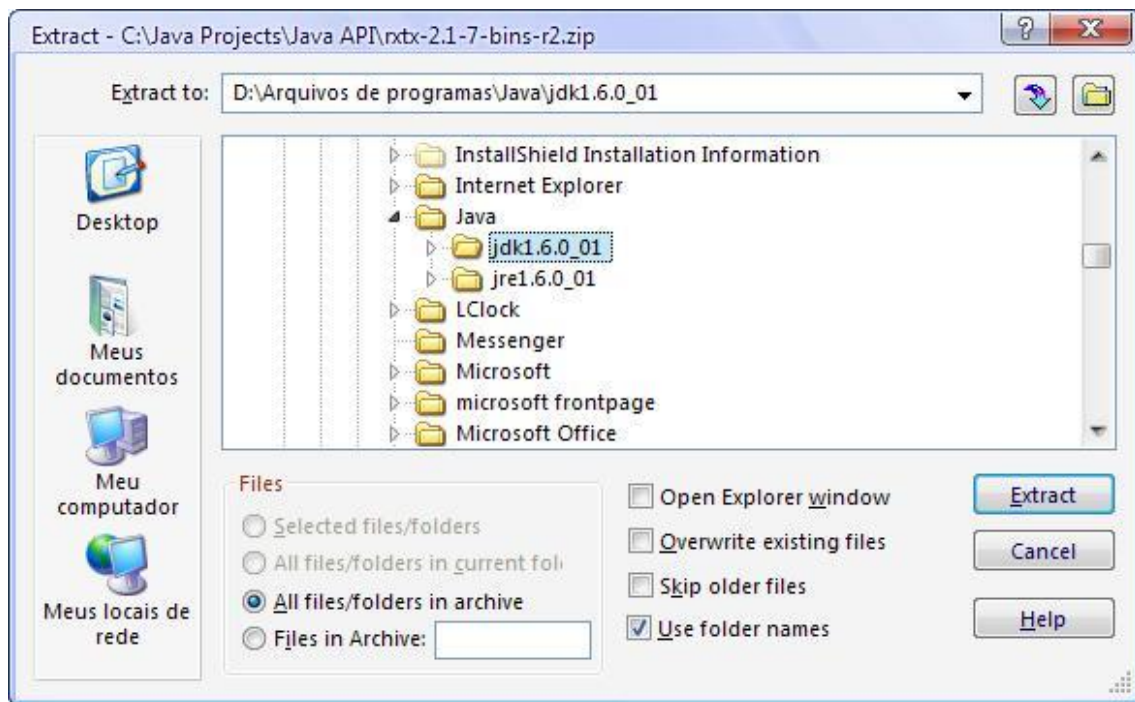


Figura 02: Extração da API RXTX para a mesma pasta do SDK.

No meu caso a pasta de instalação será *D:\Arquivos de programas\Java\jdk1.6.0_01\rxtx-2.1-7-bins-r2* mas isso irá depender de onde está instalado o seu SDK.

O próximo passo é copiar os arquivos *rxtxParallel.dll* e *rxtxSerial.dll* que se encontram dentro da pasta *\Windows\i386-mingw32* que está dentro da pasta de instalação da RXTX para a pasta de instalação do Windows e da pasta *\System32* que fica dentro da pasta de instalação do Windows.

The screenshot shows a Windows XP desktop environment. The primary focus is a 'Windows Explorer' window titled 'D:\WINDOWS\system32'. The window's address bar shows the path 'D:\WINDOWS\system32'. The left sidebar displays a tree view of the file system, with 'system32' selected. The main pane shows a grid of files and folders, including 'system32' and various DLLs and executables. The taskbar at the bottom features the Start button, a clock, and a system tray with a volume icon and a network status icon.

Também é necessário copiar os arquivos *rxtxParallel.dll* e *rxtxSerial.dll* para dentro da pasta de instalação do seu SDK e do seu JRE na pasta */bin*. Vale notar que se você for um desenvolvedor e instalou o kit de desenvolvimento Java (SDK) irá ter as duas pastas de instalação do JDK e do JRE e precisar copiar os arquivos para as duas bases de instalação, porém se você for apenas um cliente terá somente o JRE instalado, nesse caso basta copiar os arquivos para a base do JRE, nesse caso é necessário copiar os arquivos *rxtxParallel.dll* e *rxtxSerial.dll* tanto para a pasta */bin* como para a pasta */bin/client*.

Para MAC OS a pasta padrão para copiar os arquivos RXTXcomm.jar e librtxSerial.jnilib é a /Library/Java/Extensions.

Pronto agora oficialmente a API RXTX está instalada no seu computador e pronta para uso, porém em alguns computadores não temos as variáveis do Java devidamente instaladas, isso também poderá comprometer a utilização da API, então os passos

descritos abaixo serão válidos somente se o seu computador não tem as variáveis de ambiente JAVA_HOME, PATH e CLASSPATH configuradas.

Configurando as variáveis de ambiente

Configurar as variáveis de ambiente não tem segredo nenhum, basta ir em painel de controle, desempenho e manutenção e por último sistema, irá abrir um tela, nessa tela acesse a aba Avançado e clicar no botão variáveis de ambiente. A **figura 04** mostra a tela para a configuração das variáveis de ambiente.

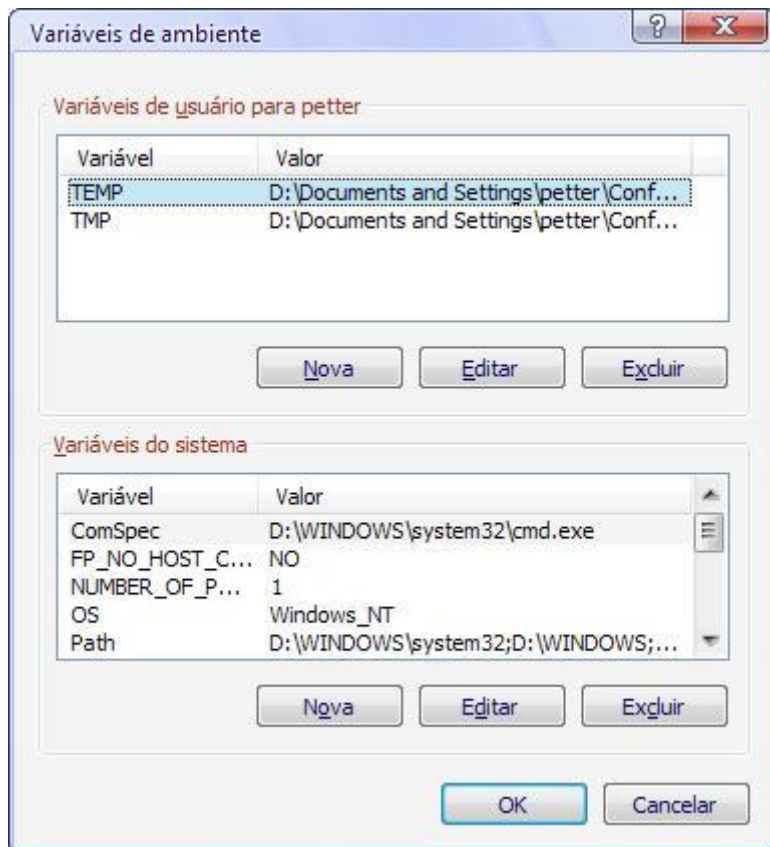


Figura 04: Configurando variáveis de ambiente

São ao total 3 variáveis a ser configura, a primeira delas é a JAVA_HOME, que é a base onde o Java está instalado. Basta para isso clicar em Nova em Variáveis de Ambiente, irá se abrir uma nova tela, digite JAVA_HOME no nome da variável e depois o caminho da instalação do seu Java, no meu caso ficou como na **figura 05**.

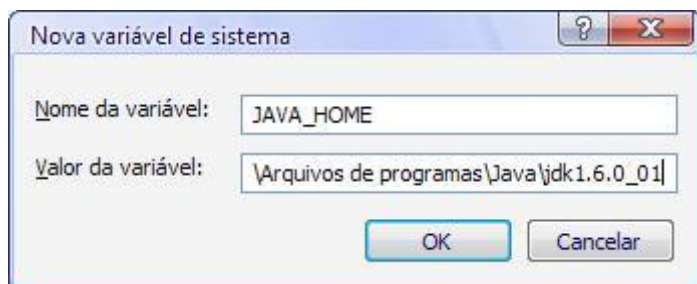


Figura 05: criando a variável JAVA_HOME

Agora é só clicar em OK que a sua variável já está criada no sistema. Devemos em seguida criar a variável CLASSPATH, o processo é o mesmo do descrito acima, basta alterar os seus valores conforme a **figura 06**.

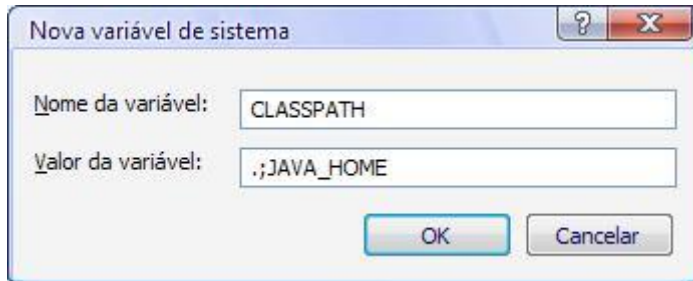


Figura 06: criando a variável CLASSPATH

A última variável a ser configurada é a PATH, geralmente essa variável já existe no seu sistema então você terá que localizá-la e editá-la ao invés de criar uma nova. Fora isso o processo é o mesmo do descrito acima, basta alterar o valor da variável a acrescentar a pasta dos arquivos binários do seu SDK como mostra a **figura 07**.

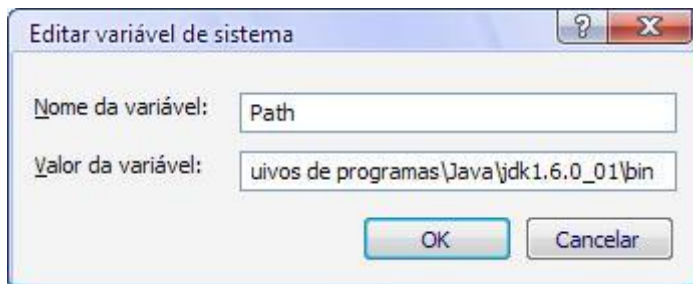


Figura 07: configurando a variável PATH

Pronto, agora com certeza sua API está totalmente funcional, se por acaso não funcionar provavelmente algum passo foi efetuado de maneira incorreta, basta refazer o processo de instalação.

Para executar o envio de SMS via modem é necessário criar a pasta lock (Linux, MAC OS), com o seguinte comando:

```
mkdir /var/lock  
chmod go+rwx /var/lock
```