

Using Variables in Settings Files

The `quickfix.SessionSettings` class supports variable interpolation using a configured set of substitution values. By default, the substitution values are JVM system properties.

The variable syntax is `"${variable}"`. For example, if system property `"user.name"` is `"USER"`...

```
FileStorePath=logs/${user.name}
```

becomes...

```
FileStorePath=logs/USER
```

Remember that system properties can be specified to the JVM using the `-D` argument on the command line. You can also use the method `SessionSettings.setVariableValues()` to configure your own set of substitution values.

```
Properties settingsVariableValues= new Properties();
settingsVariableValues.put("myvariable", "somevalue");
... other custom settings

settings.setVariableValues(settingsVariableValues);
```

The substitution values could be loaded from a property file.

```
Properties myCustomSettingsVariables = new Properties();
myCustomSettingsVariables.load(inputStream);
settings.setVariableValueS(myCustomSettingsVariables);
```

You can implement simple property inheritance by constructing a chain of `java.util.Properties` objects.

```
Properties settingsVariableValues= new Properties(System.getProperties());
settingsVariableValues.put("myvariable", "somevalue");
... other custom settings, possibly overriding the default system
properties
```

These techniques give you a simple way to create templates of settings files that are customized based on run time values.