

Table of Contents

Die Client/Server Properties werden verwendet um Informationen über den Client zum Server und vom Server zum Client zu übertragen. Die Übertragung von Properties ist bereits ins Kommunikationsprotokoll integriert und bedarf keiner speziellen Behandlung.

Bei der Übertragung werden nur neue, geänderte oder gelöschte Properties berücksichtigt. Weiters werden nur Strings, Numerische und Boolesche Werte übertragen jedoch keine serialisierbaren POJOs. Es wird garantiert, daß am Server und Client dieselben Properties zur Verfügung stehen und zum Zeitpunkt des Zugriffs synchronisiert sind.

Wenn eine Property am Client gesetzt wird, dann löst das nicht in allen Fällen eine sofortige Übertragung zum Server aus. Die Properties werden im Normalfall bei der nächsten Kommunikation vollautomatisch berücksichtigt. Es gibt jedoch auch Properties für die dies nicht zutrifft. Dabei handelt es sich um Properties mit denen spezielle Einstellungen wie z.B Timeouts gesteuert werden.

Die Properties werden am Client mit der Connection Klasse gesetzt:

```
MasterConnection macon = new MasterConnect(createConnection());  
macon.setProperty(IConnectionConstants.PREFIX_CLIENT + "language", "de_AT");
```

Die Properties werden als Key/Value Paar definiert. Der Key kann grundsätzlich beliebig gewählt werden. Es gibt jedoch vordefinierte Kategorien die durch IConnectionConstants PREFIX_CLIENT, PREFIX_SERVER, PREFIX_SESSION, PREFIX_SYSPROP und PREFIX_REQUEST festgelegt sind.

Alle Properties die mit PREFIX_SERVER + PREFIX_SESSION beginnen, lösen eine sofortige Übertragung vom Client zum Server aus wenn die Verbindung bereits hergestellt wurde. Und Properties mit dem PREFIX_CLIENT dürfen nur verändert werden, solange die Verbindung zwischen Client und Server noch nicht hergestellt wurde.

Beim Aufbau einer Verbindung werden bereits einige vordefinierte Properties gesetzt, wie z.B.:

```
PREFIX_CLIENT + PREFIX_SYSPROP + "user.name" with  
System.getProperty("user.name")  
PREFIX_CLIENT + PREFIX_SYSPROP + "os.name" with  
System.getProperty("os.name")  
PREFIX_CLIENT + PREFIX_SYSPROP + "os.version" with  
System.getProperty("os.version")  
...  
...  
PREFIX_CLIENT + "spec_version" mit jvx.rad.IPackageSetup.SPEC_VERSION
```

Am Server werden die Properties direkt in die Session gesetzt. Die Session ist über den SessionContext verfügbar:

```
SessionContext.getCurrentSession().setProperty("name", "value");
```

Auch Serverseitig werden beim Initialisieren der Session bereits Properties gesetzt:

```
PREFIX_SERVER + PREFIX_SYSPROP + "user.name" with  
System.getProperty("user.name")  
PREFIX_SERVER + PREFIX_SYSPROP + "os.name" with
```

```
System.getProperty("os.name")
PREFIX_SERVER + PREFIX_SYSPROP + "os.version" with
System.getProperty("os.version")
...
...
PREFIX_SERVER + "server_version" with
com.sibvisions.rad.IPackageSetup.SERVER_VERSION
PREFIX_SERVER + "spec_version" with jvx.rad.IPackageSetup.SPEC_VERSION
PREFIX_SERVER + "hostname" with the host name of the server
PREFIX_SERVER + "address" with the IP address of the server
```

From:

<http://doc.sibvisions.com/> - **Documentation**

Permanent link:

http://doc.sibvisions.com/de/jvx/communication/client_server_properties



Last update: **2024/11/18 10:34**