

6. Data API

- 6.1.1. ObjectInfo a OpenObject
- 6.1.2. Prístup k STRUCTDEF dátam
- 6.1.3. Prístup k archívnym dátam
- 6.1.4. Volanie D2000 RPC procedúr
- 6.1.5. Send Byte Array z klientskej aplikácie
- 6.1.6. JAPI event listeners
 - 6.1.6.1. Send Byte Array z prostredia D2000 Java
 - 6.1.6.2. Volanie RPC procedúr z prostredia D2000 Java

6.2. D2000 JAPI Utility

6.3. Management API

- 6.3.1. License Managent API
- 6.3.2. Audit Management API

Dátová as D2000 JAPI slúži na prenos dáta a event-ov medzi klientskou aplikáciou a D2000 aplikáciou v oboch smeroch komunikácie. Podrobný popis API je dostupný vo forme javadoc.

6.1.1. ObjectInfo a OpenObject

Klientská aplikácia môže získať informácie o objektoch D2000 a ich hodnoty pomocou metód D2Session:

- `getObjectInfo(...)`
- `openObject(...)`

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.ObjectInfoTest`
- `sk.ipesoft.d2000.d2japi.tests.OpenObjectTest`

6.1.2. Prístup k STRUCTDEF dátam

Klientská aplikácia môže získať informácie o definícii štruktúr D2000 pomocou metódy D2Session:

- `getStructDefInfo(...)`

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.StructDefInfoTest`

6.1.3. Prístup k archívnym dátam

Klientská aplikácia môže získať archívne dáta D2000 pomocou metódy D2Session:

- `getArchiveValues(...)`

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.ArchiveTests`

6.1.4. Volanie D2000 RPC procedúr

Klientská aplikácia môže volať RPC procedúry existujúcich Event objektov D2000 pomocou metód D2Session:

- `callRPC(...)`
- `callRPCNoreply(...)`

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.RPCTest`

6.1.5. Send Byte Array z klientskej aplikácie

Jedná sa o komunikáciu medzi klientskou aplikáciou a RPC procedúrou Event objektu, ktorá musí byť implementovaná v prostredí D2000 Java. Klientská aplikácia vie poslať objekt typu `byte[]` RPC procedúre a získať návratovú hodnotu rovnakého typu `byte[]` ako odpoveď. Objekty typu `byte[]` slúžia napríklad na netypový prenos IN/OUT parametrov medzi klientskou aplikáciou a prostredím D2000 Java v prípade, že transformácia na objekty typu Unival nie je možná. Pri použití tohto typu komunikácie je dôležité, aby obe strany rozumeli IN/OUT `byte[]` dátam na aplikanej úrovni.

Metódy D2Session:

- `sendByteArray(...)`
- `sendByteArrayNoreply(...)`

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.SBATest`

6.1.6. JAPI event listeners

Klientská aplikácia využívajúca JAPI si dokáže zaregistrovať listener objekty a reagovať tak na SBA alebo RPC volania event procesov D2000. Jedná sa o obsluhu volaní `CALL ... ASYNC ON`, `CALL ... SYNC ON`, `sendByteArraySync`, `sendByteArrayAsync`, `callESLSync` a `callESLAsync`. Registrácia listener objektov klientskej aplikácie prebieha pomocou metód `D2Session`:

- `setRPCListener(...)`
- `setSBAListener(...)`

6.1.6.1. Send Byte Array z prostredia D2000 Java

Táto as JAPI slúži na asynchrónne prijímanie objektov typu `byte[]` z prostredia D2000 Java (z bežiaceho D2000 Event procesu) klientskou aplikáciou. Klientská aplikácia si zaregistruje listener na odoberanie `sendByteArraySync` a `sendByteArrayAsync` volaní. D2000 aplikácia vie poslať z prostredia procedúry Event objektu implementovanej v D2000 Java request, ktorý bude obsluhovaný klientskou aplikáciou.

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.SBAEventsTest`

6.1.6.2. Volanie RPC procedúr z prostredia D2000 Java

Táto as JAPI slúži na asynchrónne prijímanie eventov z prostredia D2000 Java alebo ESL, klientskou aplikáciou. Klientská aplikácia si zaregistruje listener na odoberanie RPC volaní `CALL ... ASYNC ON`, `CALL ... SYNC ON`, `callESLSync` a `callESLAsync`. D2000 aplikácia vie poslať z prostredia RPC procedúry Event objektu (Java alebo ESL) request, ktorý bude obsluhovaný klientskou aplikáciou.

Referenná implementácia:

- `sk.ipesoft.d2000.d2japi.tests.RPCEventsTest`

6.2. D2000 JAPI Utility

Pre jednoduchšie použitie D2000 JAPI vznikli tieto utility triedy:

`sk.ipesoft.d2000.d2japi.D2SessionUtils`

6.3. Management API

Táto as D2000 JAPI slúži na management D2000 runtime prostredia a týka sa predovšetkým infraštruktúry D2000 Servera. Táto as JAPI nie je primárne určená pre aplikčných programátorov. Hlavná funkcionálnosť management API je:

- Sledovanie stavu procesov D2000 servera
`openProcessList(...)`
- Komunikácia s procesmi D2000 servera pomocou TellCommands
`getSupportedTellCommands(...)`
`tellCommand(...)`
- Nastavovanie debug informácií procesov D2000 servera
`getDebugInfoCategories(...)`
`setDebugInfo(...)`
- Real Time Info komunikácia s procesmi D2000
`getRTIProviders(...)`
`execRTICommand(...)`
- Monitoring Console Output D2000 procesov
`openCono(...)`

6.3.1. License Management API

Manažment licencií a runtime license info je dostupné pomocou metód `D2Session`

- `getLicenseStatus(...)`
- `consoleInfo(...)`
- `changeConsoleInfo(...)`

6.3.2. Audit Management API

Táto as slúži na získanie dát o všetkých aktivitách a zmenách konfigurácie objektov typu *Užívateľ* a *Skupina objektov*. Dáta sú k dispozícii iba v prípade, že je aktívne úplné monitorovanie – vi konfiguráciu príruku systému D2000, kapitola *Užívateľ a prístupové práva v systéme/Monitorovanie zmien v konfigurácii objektu Užívateľ* a *Monitorovanie zmien v konfigurácii objektu Skupina Objektov*.

K načítaniu záznamov z databázy slúžia operácie

- **getAuditUserOperations(...)**
- **getAuditResourceGroupOperations(...)**

K načítaniu zoznamu možností pre filtrovanie podmienky týchto dvoch volaní slúži metóda

- **getAuditData(...)**