

# IMPLEMENTÁCIA TECHNOLOGIE CORBA PRI AUTOMATIZÁCI PRENOSU ÚDAJOV MEDZI HETEROGÉNNYMI SYSTÉMAMI.

## THE CORBA TECHNOLOGY IMPLEMENTATION AT THE AUTOMATION OF DATA TRANSFERS BETWEEN HETEROGENEOUS SYSTEMS.

Pavol TANUŠKA – Michal ELIÁŠ

*Autori: Ing. Pavol Tanuška, PhD., Ing. Michal Eliáš*

*Pracovisko: Katedra aplikovanej informatiky a automatizácie, Materiálovotechnologická fakulta STU*

*Adresa: Hajdóczyho 1, 917 24 Trnava*

*Tel.: 00421 33 544 77 34*

*Email: [tanuska@mtf.stuba.sk](mailto:tanuska@mtf.stuba.sk), [elias@mtf.stuba.sk](mailto:elias@mtf.stuba.sk)*

### Abstract

*This contribution deals with problem of the data transfers between heterogeneous systems. The solution of this problem is the implementation of the Corba technology that eliminates many absences, which have another kind of data transfers technologies.*

*Tento článok pojednáva o problematike prenosu údajov medzi heterogénnymi systémami. Pri riešení tohto problému sme implementovali technológiu Corba, ktorá eliminovala mnohé nedostatky iných technológií prenosu údajov.*

### Key words

*Corba technology, Corba, automation, heterogeneous systems, teledozimetric systems, Oracle 8i, server, FTP, TDSClient module, information's systems, communication interface, communication*

*technológia Corba, Corba, automatizácia, systémy heterogénne, systémy teledozimetrické, Oracle 8i, server, FTP, modul TDSClient, systémy informačné, rozhranie komunikačné, komunikácia*

### Úvod

Pri vývoji informačného systému na archiváciu dát teledozimetrického systému (ATDS) pre Laboratórium radiačnej kontroly okolia SE a.s., EBO o.z. prevádzka v Trnave, sme riešili problém s komunikáciou medzi systémom TDS firmy Euromove s.r.o., ktorý zberá dáta zo snímačov v okolí cca 30 km od Atómovej elektrárne Jaslovské Bohunice, a práve nami navrhovanou aplikáciou, ktorá by tieto dáta archivovala prostredníctvom databázového servera Oracle 8i a sprístupňovala ich používateľom vo forme rôznych agregovaných prehľadov a grafov.

Samotnú problematiku možno charakterizovať nasledovnými atribútmi:

- dáta zo snímačov sa zbierajú nepretržite (reálny čas),
- každých 5 minút sa archivuje približne 800 hodnôt z meracích miest (čo činí viac ako 200 000 údajov za deň; 7,5 milióna za rok !),
- dáta je potrebné prebrať, uložiť a sprístupniť viacerým používateľom okamžite, ako boli namerané a následne prijaté v systéme ATDS,
- paralelne s ukladaním dát na server Oracle 8i (umiestnený v Jaslovských Bohuniciach), sa vykonáva z dôvodu bezpečnosti mesačná záloha na lokálnom databázovom serveri Sybase,
- obe aplikácie (TDS, ATDS) bežia na rôznych počítačoch, pod rôznymi operačnými systémami (Windows/Unix),
- sú implementované v rôznych vývojových prostrediach (C++/Delphi), čiže ide o komunikáciu v distribuovanom heterogénnom prostredí.

Z týchto dôvodov bolo potrebné rozdeliť vývoj tohto archivačného systému ATDS do dvoch modulov:

ATDS Viewer - modul na evidenciu meracích miest a vyhodnocovanie nameraných údajov

TDSClient - modul na preberanie dát z unixového systému TDS a ich ukládanie do databázy Oracle 8i.

### Výber komunikačného rozhrania

Základným problémom, ktorý bolo treba vyriešiť ako prvý, bol správny výber komunikačného rozhrania, a jeho následná implementácia, medzi dve aplikácie. V súčasnosti to umožňujú tieto technológie:

**DCOM** – Distributed Component Object Model, umožňuje prístupovať k objektom COM umiestneným na iných počítačoch. Okrem vzdialeného vytvárania objektov poskytuje DCOM aj bezpečnostné funkcie, ktoré umožňujú serverom špecifikovať, ktorí klienti majú oprávnenie vytvárať inštancie akých serverov a aké operácie môžu uskutočňovať. Problémom je však jeho úzka spätosť so systémom Windows.

**Sockets** – sú určené na tvorbu ľahkých klientov. Využívajú zvláštnu aplikáciu na strane servera, ktorá prijíma požiadavky od klienta a odovzdáva ich samotnému serveru, čiže umožňuje klientom využívať služby servera.

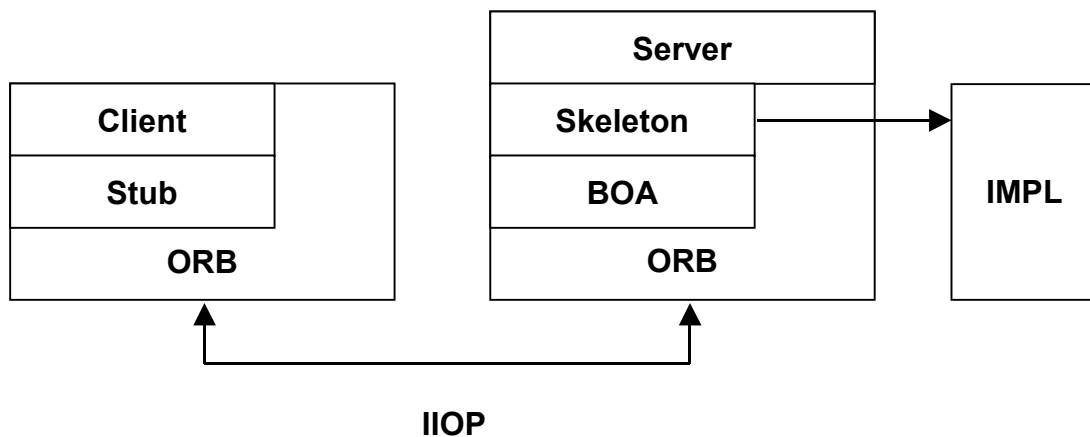
**Web Services** – sú založené na protokole SOAP (Simple Object Access Protocol). Je to ľahký protokol používaný na vzájomnú výmenu dát v distribuovanom prostredí. Je postavený na vzdialenom volaní procedúr pomocou protokolu HTTP. Dáta sa posielajú vo formáte XML. Webové služby sa samy popisujú (čo poskytujú, kde ich možno získať a ako ich volať) pomocou jazyka založeného na XML.

**CORBA** – Common Object Request Broker Architecture, je špecifikácia pre komunikáciu v distribuovanom heterogénnom prostredí. Používa vlastný protokol IIOP (Internet Inter-ORB Protocol) a jazyk IDL na definíciu rozhraní. Táto architektúra je určená pre prácu v objektovo orientovaných prostrediach. Program žiada o objekt pomocou modulu ORB, ktorý spravuje interakcie medzi jednotlivými objektami. CORBA je implementovaná v mnohých programovacích jazykoch (Java, C++, Delphi, Python, ...) na rôznych softvérových a hardvérových platformách.

Vzhľadom na mnohé nevýhody vyššie uvedených technológií sme sa rozhodli ako komunikačné rozhranie medzi našimi aplikáciami vybrať technológiu CORBA a následne ju implementovať do aplikácie ATDS.

### Technológia CORBA

Základom návrhu aplikácií používajúcich technológiu CORBA je návrh ich komunikačného rozhrania. Prostredníctvom rozhrania sú definované všetky objekty modelu CORBA. Serverové aplikácie poskytujú špecifické deklarácie typov, ale aj špecifické rozhrania a metódy, ktoré môže volať akákoľvek klientská aplikácia.



*Obr. 1 Architektúra modelu CORBA*

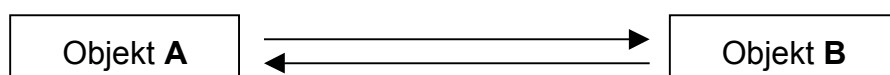
kde:

- Client - objekt využívajúci služby serverového objektu
- Stub - obraz serverového objektu na strane klienta
- ORB - rozhranie spravujúce interakcie medzi klientom a serverom
- IIOP - komunikačný protokol, nadstavba protokolu TCP/IP
- Server - serverový objekt, jeho metódy a dáta, ktoré sú poskytované klientovi
- Skeleton - definícia tried a dát serverového objektu
- IMPL - implementácia tried serverového objektu
- BOA - (Basic Object Adaptor) sprostredkováva komunikáciu medzi implementáciou serverového objektu a ORB.

Na definovanie rozhrania sa používa jazyk IDL (Interface Definition Language). Je nezávislý na programovacom jazyku, podobá sa však na jazyk C++ alebo Java. Dodávatelia rozhrania ORB poskytujú vlastné prekladače/preprocesory (napr. idl2cpp, idl2pas, ...), ktoré na základe inštrukcií v súbore IDL generujú súbory so zdrojovým kódom v cieľovom programovacom jazyku.

Príklad použitia na strane klienta:

objekt A volá funkciu definovanú v rozhraní merac1: `hodnota = A.akt_stav();`



*Obr. 2 Základný princíp komunikácie*

kde:

objekt A je inštanciou klientskej časti (client stub) rozhrania merac1

objekt B je inštanciou serverovej časti (server skeleton) rozhrania merac1

CORBA zabezpečí:

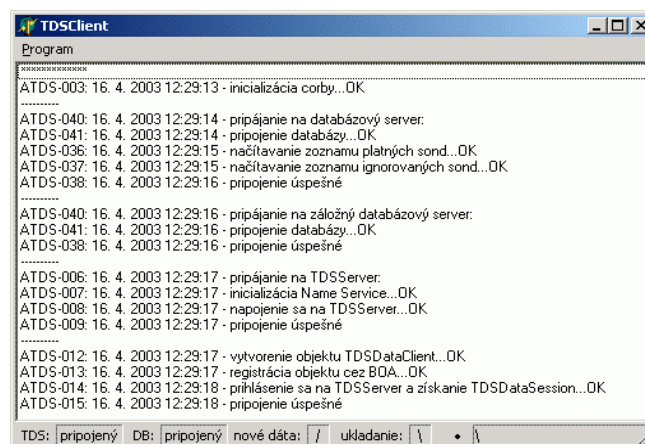
- prenesenie informácie o tom, ktorý objekt je volaný (**B**), ktorá z jeho metód je volaná (**akt\_stav**), aké sú hodnoty parametrov (v tomto prípade žiadne)
- spustenie metódy **akt\_stav** objektu **B**
- prenesenie výsledku
- odovzdanie výsledku objektu **A**, ktorý výsledok uloží do premennej **hodnota**.

### Návrh modulu TDSClient

Ako prvý sme museli implementovať modul TDSClient, pomocou ktorého sme spracovávali dáta zo systému TDS firmy Euromove s.r.o. Práve v tomto module sme implementovali technológiu Corba, ktorá nám zabezpečila požadovanú výkonnosť a spoľahlivosť pri spracovávaní veľkých objemov dát.

Modul má jednoduché používateľské rozhranie, ktoré informuje používateľa o všetkých dôležitých udalostiach a umožňuje mu v obmedzenej miere zasahovať do procesu preberania a spracovania dát. Všetky udalosti sú zapisované aj do logovacieho súboru, ktorý slúži na dodatočnú identifikáciu problémov.

Proces pripájania je plne automatizovaný, modul sa pripojí na databázové servre; skontaktuje sa so systémom TDS a rozhraniami TDSDataClient a TDSDataSession nadviaže komunikáciu.



```
TDSClient
Program
-----
ATDS-003: 16. 4. 2003 12:29:13 - inicializácia corby...OK
-----
ATDS-040: 16. 4. 2003 12:29:14 - pripájanie na databázový server:
ATDS-041: 16. 4. 2003 12:29:14 - pripojenie databázy...OK
ATDS-036: 16. 4. 2003 12:29:15 - načítavanie zoznamu platných sond...OK
ATDS-037: 16. 4. 2003 12:29:15 - načítavanie zoznamu ignorovaných sond...OK
ATDS-038: 16. 4. 2003 12:29:16 - pripojenie úspešné
-----
ATDS-040: 16. 4. 2003 12:29:16 - pripájanie na záložný databázový server:
ATDS-041: 16. 4. 2003 12:29:16 - pripojenie databázy...OK
ATDS-038: 16. 4. 2003 12:29:16 - pripojenie úspešné
-----
ATDS-006: 16. 4. 2003 12:29:17 - pripájanie na TDS Server:
ATDS-007: 16. 4. 2003 12:29:17 - inicializácia Name Service...OK
ATDS-008: 16. 4. 2003 12:29:17 - napojenie sa na TDS Server...OK
ATDS-009: 16. 4. 2003 12:29:17 - pripojenie úspešné
-----
ATDS-012: 16. 4. 2003 12:29:17 - vytvorenie objektu TDSDataClient...OK
ATDS-013: 16. 4. 2003 12:29:17 - registrácia objektu cez BOA...OK
ATDS-014: 16. 4. 2003 12:29:18 - prihlásenie sa na TDS server a získanie TDSDataSession...OK
ATDS-015: 16. 4. 2003 12:29:18 - pripojenie úspešné
-----
TDS: | pripojený | DB: | pripojený | nové dáta: | / | ukladanie: | \ |
```

*Obr. 3 Aplikácia TDSClient*

Popis komunikácie medzi TDS a TDSClient:

- pripojenie:  
Po spustení naša aplikácia (TDSClient) nadviaže spojenie s aplikáciou TDS a odovzdá jej informácie o našom serverovom objekte (TDSDataClient) a následne obdrží informácie o serverovom objekte z TDS (TDSDataSession).
- prenos nameraných dát:  
Ak TDS získa nové dáta zo sond, tak prostredníctvom metódy `dataInQueue` z TDSDataClient oznámi, že sú dáta k dispozícii. Aplikácia následne zavolaním metódy `getData` z TDSDataSession preberie namerané údaje a spracuje ich (uloží do databázy).

- overovanie spojenia:  
Ak v čase, ktorý prekročí definovaný interval, neprebehne vzájomná komunikácia, tak si TDSDataSession overuje, či beží naša aplikácia volaním funkcie ping z TDSDataClient, na čo TDSDataClient musí odpovedať volaním metódy alive z TDSDataSession.
- strata spojenia:  
Spojenie sa môže prerušiť pri poruche siete, príp. pri reštarte operačného systému. V prípade straty spojenia TDS čaká na nové pripojenie našej aplikácie TDSClient. Dáta sa zhromažďujú vo fronte voliteľnej dĺžky, kým sa spojenie neobnoví. Dĺžka fronty bola stanovená na 1000 záznamov, čo zodpovedá približne 5-7 minútam. V prípade dlhšieho výpadku spojenia je potrebné stratené dáta vyžiadať explicitne z plávajúceho archívu (dĺžky jeden mesiac) v systéme TDS.

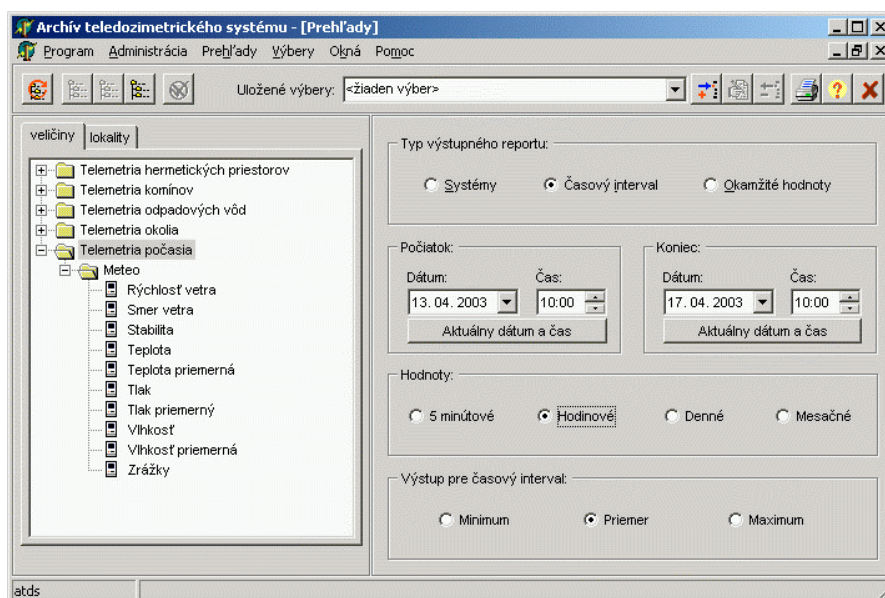
### Návrh modulu ATDS Viewer

Modul ATDS Viewer bol navrhnutý na sledovanie stavu snímačov umiestnených v blízkom okolí AE Bohunice. Snímače sú prehľadne usporiadané v dvoch stromových štruktúrach zoskupených podľa meranej veličiny alebo podľa lokality merania. Používateľ má k dispozícii tri druhy výstupov:

- sledovanie jednej veličiny vo viacerých lokalitách merania v danom čase s výstupom do grafu (Excel),
- sledovanie priebehu viacerých veličín vo viacerých lokalitách v časovom intervale s výstupom do grafu (Excel),
- sledovanie okamžitých (aktuálnych) hodnôt vybraných sond.

Často sledované meracie sondy a nastavenia (typ výstupu, dátum a čas,...) si môže používateľ uložiť do výberov.

Administrátor navyše môže pridávať používateľov, zasahovať do evidencie sond (meracích miest) a tiež upravovať stromovú štruktúru. Môže riadiť proces dopĺňania chýbajúcich dát zo systému TDS na základe časového intervalu a identifikačných čísiel sond, pre ktoré tieto dáta chýbajú. A v neposlednom rade spúšťa proces mesačného zálohovania.



**Obr. 4** Aplikácia ATDS

## Zhodnotenie

Obe vytvorené aplikácie odbremenili systém TDS, ktorý zhromažďuje údaje zo sond a poskytuje ich vlastnému prezeráču, od ďalších inštalácií prezeráča v sieti. Prostredníctvom aplikácie ATDS majú používatelia prístup k nameraným údajom aj zo staršieho obdobia ako bol jeden mesiac. Majú možnosť sledovať vývoj rôznych veličín v reálnom čase, a tiež ich vzájomné závislosti.

Administrátor môže riadiť proces zálohovania dát v mesačnom, príp. inom intervale, a tiež pristupovať k dátam zálohovaným v predchádzajúcich obdobiach.

### Literatúra:

- [1] W3C: Simple Object Access Protocol, <http://www.w3.org/TR/SOAP/>
- [2] Mederly, P.: Úvod do architektúry CORBA, <http://www.uniba.sk/www/sw-eng/01-02-zs/>
- [3] Object Management Group: Catalog of OMG Specifications, [http://www.omg.org/technology/documents/corba\\_spec\\_catalog.htm](http://www.omg.org/technology/documents/corba_spec_catalog.htm)
- [4] Microsoft: DCOM, <http://www.microsoft.com/com/tech/DCOM.asp>