

2008

Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 407: Přístup k datům v časové řadě (TSDA)	ČSN EN 61970-407 33 4910
---	------------------------------------

idt IEC 61970-407:2007

Energy management system application program interface (EMS-API) -
Part 407: Time Series Data Access (TSDA)

Interface de programmation d'application pour système de gestion d'énergie (EMS-API) -
Partie 407: Accès aux données en série chronologique (TSDA)

Schnittstelle für Anwendungsprogramme für Energiemanagementsysteme (EMS-API) -
Teil 407: Zugriff auf Daten, die auf Zeitfolgen beruhen (TSDA)

Tato norma je českou verzí evropské normy EN 61970-407:2007. Překlad byl zajištěn Českým normalizačním institutem. Má stejný status jako oficiální verze.

This standard is the Czech version of the European Standard EN 61970-407:2007. It was translated by Czech Standards Institute. It has the same status as the official version.



© Český normalizační institut, 2008
Podle zákona č. 22/1997 Sb. smějí být české technické normy rozmnožovány
a rozšiřovány jen se souhlasem Českého normalizačního institutu.

81165

Informace o citovaných normativních dokumentech

IEC 61970-1 zavedena v ČSN EN 61970-1 (33 4910) Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 1: Směrnice a obecné požadavky (idt EN 61970-1:2006, idt IEC 61970-1:2005)

IEC/TS 61970-2 nezavedena

IEC 61970-301:2003 zavedena v ČSN EN 61970-301:2006 (33 4910) Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 301: Základ obecného informačního modelu (CIM) (idt EN 61970-301:2004, idt IEC 61970-301:2003)

IEC/TS 61970-401 nezavedena

IEC 61970-402 dosud nezavedena

OMG HDAIS:2003 nezavedeno

OMG DAF:2002 nezavedeno

OMG DAIS:2002 nezavedeno

OPC HDA:2003 nezavedeno

Obdobné mezinárodní normy

IEC 61970-407:2007 Energy management system application program interface (EMS-API) - Part 407: Time Series Data Access (TSDA)
(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 407: Přístup k datům v časové řadě (TSDA))

Porovnání s mezinárodní normou

Obsah normy je identický s IEC 61970-407:2007 a navíc obsahuje normativní přílohu ZA, kterou doplnil CENELEC.

Informativní údaje z IEC 61970-407:2007

Mezinárodní norma IEC 61970-407 byla připravena technickou komisí IEC TC 57: Řízení elektrizační soustavy a příslušná výměna informací.

Text této normy vychází z těchto dokumentů:

FDIS	Zpráva o hlasování
57/889/FDIS	57/908/RVD

Úplné informace o hlasování při schvalování této normy je možné nalézt ve zprávě o hlasování uvedené v tabulce.

Seznam všech Částí souboru IEC 61970 lze nalézt na internetové adrese IEC pod společným názvem *Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API)*.

Komise rozhodla, že obsah této publikace se nebude měnit až do konečného data vyznačeného na

internetové adrese IEC „<http://webstore.iec.ch>“ v termínu příslušejícímu dané publikaci. Po tomto termínu bude publikace

- znovu potvrzena;
- zrušena;
- nahrazena revidovaným vydáním, nebo
- změněna.

Dvoujazyčná verze této publikace může být vydána později.

Vypracování normy

Zpracovatel: ÚJV Řež a.s., divize Energoprojekt Praha, IČ 46356088, Ing. Jaroslav Mezera

Technická normalizační komise: TNK 97 Elektroenergetika

Pracovník Českého normalizačního institutu: Ing. Jiří Holub

Strana 3

EVROPSKÁ NORMA	EN 61970-407
EUROPEAN STANDARD	
NORME EUROPÉENNE	
EUROPÄISCHE NORM	Září 2007

ICS 33.200

Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) -

Část 407: Přístup k datům v časové řadě (TSDA)

(IEC 61970-407:2007)

Energy management system application program interface (EMS-API) -

Part 407: Time Series Data Access (TSDA)

(IEC 61970-407:2007)

Interface de programmation d'application
pour système de gestion d'énergie (EMS-API) -
Partie 407: Accès aux données en série
chronologique (TSDA)
(CEI 61970-407:2007)

Schnittstelle für Anwendungsprogramme
für Energiemanagementsysteme (EMS-API) -
Teil 407: Zugriff auf Daten, die auf Zeitfolgen
beruhen (TSDA)
(IEC 61970-407:2007)

Tato evropská norma byla schválena CENELEC 2007-09-01. Členové CENELEC jsou povinni splnit Vnitřní předpisy CEN/CENELEC, v nichž jsou stanoveny podmínky, za kterých se musí této evropské normě bez jakýchkoliv modifikací dát status národní normy.

Aktualizované seznamy a bibliografické citace týkající se těchto národních norem lze obdržet na vyžádání v Ústředním sekretariátu nebo u kteréhokoliv člena CENELEC.

Tato evropská norma existuje ve třech oficiálních verzích (anglické, francouzské, německé). Verze v každém jiném jazyce přeložená členem CENELEC do jeho vlastního jazyka, za kterou zodpovídá a kterou notifikuje Ústřednímu sekretariátu, má stejný status jako oficiální verze.

Členy CENELEC jsou národní elektrotechnické komitety Belgie, Bulharska, České republiky, Dánska, Estonska, Finska, Francie, Irska, Islandu, Itálie, Kypru, Litvy, Lotyšska, Lucemburska, Maďarska, Malt, Německa, Nizozemska, Norska, Polska, Portugalska, Rakouska, Rumunska, Řecka, Slovenska, Slovinska, Spojeného království, Španělska, Švédsko a Švýcarska.

CENELEC

Evropský výbor pro normalizaci v elektrotechnice

European Committee for Electrotechnical Standardization

Comité Européen de Normalisation Electrotechnique

Europäisches Komitee für Elektrotechnische Normung

Ústřední sekretariát: rue de Stassart 35, B-1050 Brusel

© 2007 CENELEC Veškerá práva pro využití v jakékoli formě a jakýmkoli prostředky jsou celosvětově vyhrazena členům CENELEC.

Ref. č. EN 61970-

407:2007 E

Strana 4

Předmluva

Text dokumentu 57/889/FDIS, budoucího 1. vydání IEC 61970-407, vypracovaný v technické komisi IEC TC 57, Řízení elektrizační soustavy a příslušná výměna informací, byl předložen k paralelnímu hlasování IEC-CENELEC a byl schválen CENELEC jako EN 61970-407 dne 2007-09-01.

Byla stanovena tato data:

- nejzazší datum zavedení EN na národní úrovni
vydáním identické národní normy nebo vydáním
oznámení o schválení EN k přímému používání
jako normy národní (dop) 2008-06-01
 - nejzazší datum zrušení národních norem,
které jsou s EN v rozporu (dow) 2010-09-01
- Přílohu ZA doplnil CENELEC.

Oznámení o schválení

Text mezinárodní normy IEC 61970-407:2007 byl schválen CENELEC jako evropská norma bez jakýchkoliv modifikací.

Strana 5

Obsah

Strana

Úvod

..... 6

1	Rozsah platnosti	6
2	Citované normativní dokumenty	7
3	Termíny, definice a konvence pro identifikaci	7
3.1	Termíny a definice	7
3.2	Konvence	7
4	Specifikace CIS	7
4.1	Podmínky vzniku (informativní)	7
4.2	Možnost použití Historika (informativní)	8
4.3	Datový model (normativní)	9
4.4	Zprávy (normativní)	12
4.5	Rozhraní (normativní)	13
4.5.1	Objekty a rozhraní	

.....	13
4.5.2 Rozhraní serveru a relace	
.....	15
4.5.3 Řídicí rozhraní	
.....	16
4.5.4 Prohledávací rozhraní	
.....	16
4.5.5 IO rozhraní	
.....	17
4.5.6 Rozhraní klienta	
.....	18
4.6 Mapování TSDA	
.....	19
Příloha ZA (normativní) Normativní odkazy na mezinárodní publikace a na jim příslušející evropské publikace.....	20
Obrázek 1 - Struktura řídicího systému	
.....	8
Obrázek 2 - TSDA server a klienti	
.....	9
Obrázek 3 - Vyžádání dat	
.....	9
Obrázek 4 - TSDA datový model	

..... 10

Obrázek 5 - TSDA objekty a
rozhraní

.....
13

Obrázek 6 - Typická interakce mezi TSDA
objekty..... 15

Strana 6

Úvod

Tato Část IEC 61970 je součástí souboru IEC 61970, který definuje rozhraní aplikačního programu (API - Application Program Interface) pro systém řízení elektrické energie (EMS - Energy Management System). Dokumenty souboru IEC 61970-4XX a IEC 61970-5XX obsahují specifikace rozhraní složek (CIS - Component Interface Specification). CIS v souboru IEC 61970-4XX jsou určeny jako modely nezávislé na platformě (PIM - Platform Independent Model) což znamená, že jsou nezávislé na vlastní technologii použité k jejich realizaci. Specifikace PIM se rovněž označují jako specifikace Úrovně 1. Oproti tomu jsou CIS v souboru IEC 61970-5XX určeny jako modely specifické pro platformu (PSM - Platform Specific Model). Specifikace PSM se rovněž označují jako specifikace Úrovně 2.

CIS v IEC 61970-4XX stanovují funkční požadavky na rozhraní, která mají realizovat určitou složku (nebo aplikaci) pro výměnu informací s ostatními složkami (nebo aplikacemi) a/nebo pro přístup k veřejně dostupným datům standardním způsobem. Rozhraní složek popisují konkrétní typy událostí a obsahy zpráv, které mohou aplikace k tomuto účelu použít.

IEC 61970-404 definuje rozhraní pro efektivní přenos dat v časové řadě v rozloženém prostředí. Malé objemy dat jsou přenášeny s krátkým zpožděním, přičemž i velké objemy dat jsou přeneseny v krátkém čase, i když s poněkud větším zpožděním. Rovněž je zajištěno snímání dat v časové řadě. Toto je běžný požadavek u SCADA systému, který pracuje jako poskytovatel dat v reálném čase pro ostatní dílčí systémy. Tyto charakteristiky TSDA mohou být užitečné i pro jiné systémy než je SCADA. TSDA je velmi vhodný, požadují-li se krátké časy zpoždění a rovněž hromadný přenos dat.

Tyto specifikace rozhraní složek se odvolávají na objekty entit pro oblast elektrizační soustavy definované v souboru IEC 61970-3XX, včetně IEC 61970-301.

1 Rozsah platnosti

Specifikace přístupu k datům v časové řadě (TSDA - Time Series Data Access) v IEC 61970-407 definuje univerzální rozhraní pro efektivní výměnu dat. Specifikace uvažuje čekací doby (latence) způsobené místní sítí (LAN) zajišťující efektivní přenos dat i po dalších místních sítích.

IEC 61970-407 vychází ze specifikace Přístupu k historickým datům z průmyslových systémů (HDAIS) od Skupiny pro správu objektů (OMG). Základem pro OMG HDAIS jsou specifikace OMG Prostředků pro přístup k datům (DAF) a OPC přístupu k historickým datům (HDA). OMG HDAIS je model specifický pro

platformu (PSM) s CORBA jako platformou a OPC HDA je PSM s Microsoft COM jako platformou. Specifikace v IEC 61970-407 uvádí funkce těchto PSM způsobem nezávislým na technologii (tj. jako model nezávislý na platformě (PIM)). Proto objasňuje funkce na úrovni, kterou lze použít k vytvoření dalších PSM nebo která slouží jako úvod do existujících PSM, tj. HDAIS a OPC HDA. Realizátoři, kteří potřebují úvod do OMG HDAIS a OPC HDA, si mají přečíst tyto dokumenty.

TSDA rozhraní je určeno pro spolupráci s ostatními rozhraními vycházejícími z IEC 61970. Z tohoto důvodu je možno použít informace vybrané z ostatních rozhraní pro přístup k téže informaci pomocí tohoto rozhraní, například:

- identifikátory objektů,
- názvy nebo identifikátory atributů,
- názvy nebo identifikátory tříd.

Článek 4.6 poskytuje univerzální mapování pro třídy a atributy CIM.

Způsob organizování dat v serveru realizujícím TSDA rozhraní lze vidět pomocí prohledávacích rozhraní pro data a metadata. Je možno rovněž použít přímo rozhraní pro přístup k datům bez použití prohledávacích rozhraní, pokud klient *předem* zná identifikátory objektu, třídy a atributu. Identifikátory objektů lze získat pomocí dat z jiných rozhraní, například CIMXML souboru nebo rozhraní z IEC 61970-404. Informace o tom, jaké třídy a atributy jsou k dispozici budou uvedeny v dokumentech IEC 61970-45X, například SCADA data, historické výsledky estimátoru stavů, apod.

IEC 61970-1 poskytuje referenční model EMS-API, z něhož vychází tato norma. V tomto referenčním modelu je zavedena terminologie použitá v této Části IEC 61970 a objasněna funkce CIS.

IEC 61970-401 poskytuje přehled a strukturu norem CIS (IEC 61970-4XX).

Mapování IEC 61970-407 na realizaci konkrétních technologií nebo PSM je dále uvedeno v samostatném souboru dokumentů, tj. připravované IEC 61970-5XX. U skutečných realizací se použijí připravovaná IEC 61970-5XX, OMG HDAIS, OMG DAF nebo OPC HDA.

2 Citované normativní dokumenty

Pro používání tohoto dokumentu jsou nezbytné dále uvedené referenční dokumenty. U datovaných odkazů platí pouze citovaná vydání. U nedatovaných odkazů platí poslední vydání referenčního dokumentu (včetně změn).

IEC 61970-1 Energy management system application program interface (EMS-API) - Part 1: Guidelines and general requirements
(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 1: Směrnice a obecné požadavky)

IEC/TS 61970-2 Energy management system application program interface (EMS-API) - Part 2: Glossary
(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 2: Výklad zvláštních výrazů)

IEC 61970-301:2005 Energy management system application program interface (EMS-API) - Part 301: Common Information Model (CIM) base

(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 301: Základ obecného informačního modelu (CIM))

IEC 61970-401 Energy management system application program interface (EMS-API) - Part 401: Component Interface Specification (CIS) Framework

(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 401: Struktura specifikace rozhraní složek (CIS))

IEC 61970-402 Energy management system application program interface (EMS-API) - Part 402: Component Interface Specification (CIS) - Common Services

(Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 402: Specifikace rozhraní složek (CIS) - Obecné služby)

Historical Data Access from Industrial Systems (HDAIS), OMG Adopted Specification, Version 1.0, dtc/2003-02-01, November 2003 (Referred herein as „OMG HDAIS“)

(Přístup k historickým datům z průmyslových systémů (HDAIS), specifikace převzatá OMG, Verze 1.0, dtc/2003-02-01, listopad 2003 (Zde označovaná jako „OMG HDAIS“))

Utility Management System (UMS) Data Access Facility (DAF), OMG Adopted Specification, Version 2.0, formal/02-11-08, November 2002 (Referred to herein as „OMG DAF“)

(Prostředky pro přístup k datům (DAF) v řídicím systému společnosti (UMS), specifikace převzatá OMG, Verze 2.0, oficiální/02-11-08, listopad 2002 (Zde označovaná jako „OMG DAF“))

Data Acquisition from Industrial Systems (DAIS), OMG Adopted Specification, Version 1.0, formal/2002-11-07, November 2002 (Referred herein as „OMG DAIS“)

(Sběr dat z průmyslových systémů (DAIS), specifikace převzatá OMG, Verze 1.0, oficiální/2002-11-07, listopad 2002 (Zde označovaná jako „OMG DAIS“))

OPC Historical Data Access Custom Interface Specification, Version 1.20, OPC Foundation, December 2003 (Referred to herein as „OPC HDA“)

(Zákaznická specifikace rozhraní pro OPC přístup k historickým datům, Verze 1.20, OPC nadace, prosinec 2003 (Zde označovaná jako „OPC HDA“))

3 Termíny, definice a konvence pro identifikaci

3.1 Termíny a definice

Pro účely tohoto dokumentu platí termíny a definice uvedené v IEC 61970-2.

3.2 Konvence

Konvence použitá v tomto dokumentu pro jednoznačnou identifikaci UML atributu je zřetězit název třídy a název atributu s tečkou mezi nimi, například atribut „id“ ve třídě „Uzel“ pak bude pojmenován „Uzel.id“. U atributů v dílčích strukturách se mohou použít vícenásobné názvy atributů, například „Položka.id.uzel_id“, kde „uzel_id“ je určitá část ve struktuře „Položka.id“.

4 Specifikace CIS

4.1 Podmínky vzniku (informativní)

Z historických důvodů se řídicí systémy pro různé průmyslové procesy vyvíjely různými směry. Řídicí systémy pro elektrizační soustavy byly vyvíjeny na bázi UNIX a řídicí systémy pro většinu ostatních průmyslových procesů byly vyvíjeny na bázi Windows. U řídicích systémů na bázi Windows se stal rozhodující normou OPC. U systémů založených na bázi UNIX byl vyvinut DAIS/HDAIS API definovaný v Jazyku pro definování rozhraní (IDL - Interface

Strana 8

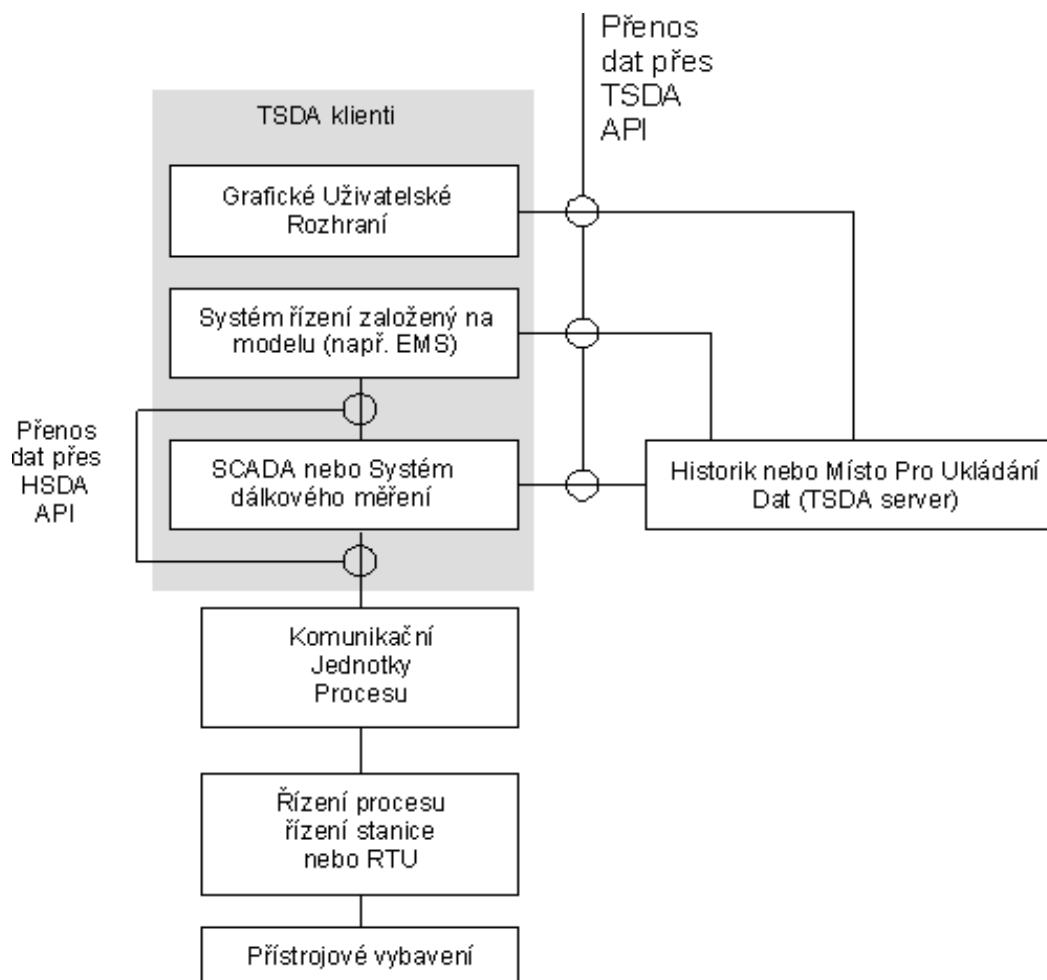
Definition Language) Architektury zprostředkovatele požadavků na obecný objekt (CORBA - Common Object Request Broker Architecture). DAIS/HDAIS vychází z OPC těžice z úspěšnosti OPC a umožňuje snadnou emulaci na OPC. S tímto záměrem zahájila Skupina pro správu objektů (OMG) v roce 1997 vývoj rozhraní založeného na CORBA se stejnými funkcemi jako OPC. TSDA má funkce z OMG HDAIS a OPC HDA uvedeny způsobem nezávislým na technologii, protože TSDA je zamýšlen jako model nezávislý na platformě (PIM).

4.2 Možnost použití Historika (informativní)

Důležitou komponentou v řídicím systému společnosti je Historik nebo Místo Pro Ukládání Dat Společnosti (UDW - Utility Data Warehouse), působící jako archiv uplynulých nebo budoucích dat v časové řadě. UDW vystupuje vůči složkám klienta jako server. Takovýto systém má obvykle následující složky:

- Přístrojové vybavení procesu poskytující data z čidel a možnosti ovládání.
- Vzdálené terminály (RTU) nebo řídicí systémy stanice snímající data z čidel a řídicí ovládače.
- Komunikační jednotky procesu připojené na RTU nebo řídicí systémy stanice. Dálková komunikace je obvykle řešena pomocí specializovaného RTU nebo provozních sběrníkových protokolů (např. IEC 60870-5). Soubor IEC 61850 je novou normou pro komunikaci se stanicemi a uvnitř stanic. Rozhraní definovaná v této specifikaci lze použít jako normalizované API pro zahrnutí takovýchto řešení komunikace na straně klienta.
- Dílčí systém SCADA poskytující zpracovaná data z čidel a možnosti ovládání obsluhy, aplikací nebo jiným systémům.
- Systém řízení založený na modelu, například systém řízení elektrické energie (EMS), používající dílčí systém SCADA pro rozšířené zpracování a řízení.
- Grafická uživatelská rozhraní (GUI).
- UDW, které uchovává data v časové řadě.

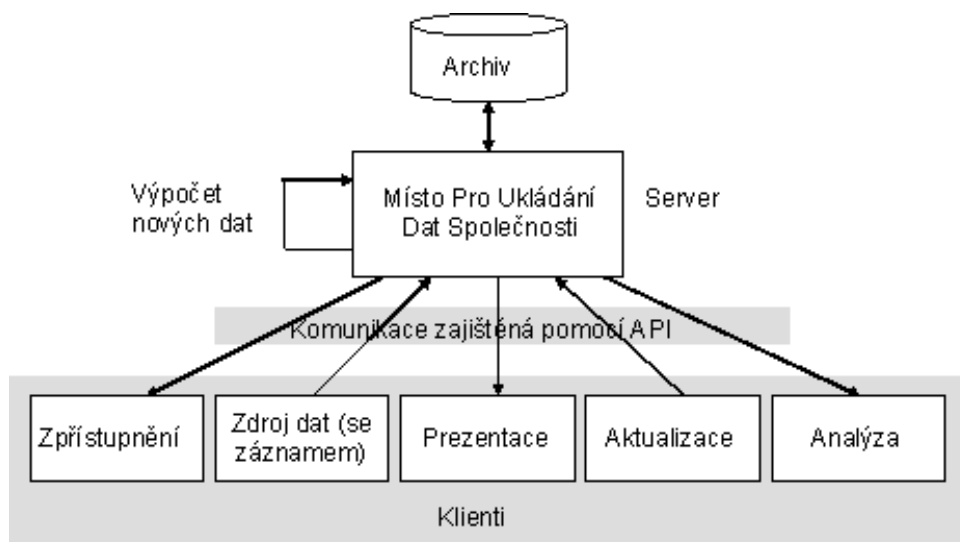
Architektura je zobrazena na obrázku 1 dále.



Obrázek 1 - Struktura řídicího systému

UDW a možní klienti, kteří je používají, jsou uvedeni na obrázku 2:

- Zdroje dat zaznamenávající data, například SCADA zaznamenávající data v reálném čase, EMS zaznamenávající výsledky výpočtů nebo GUI zaznamenávající ručně vložená data.
- Komunikační Jednotka Procezu poskytující data v časové řadě ze vzdálených zdrojů.
- Klienti, například GUI, vyhledávající data pro zpřístupnění, prezentaci a aktualizaci/opravy.
- Klienti, například analytické programy, vyhledávající data pro analýzy a nebo jako vstup pro výpočty.
- UDW provádí obvykle rovněž výpočty z již uložených dat v časové řadě, které poskytují nové výsledky.



Obrázek 2 - TSDA server a klienti

TSDA zajišťuje jak operaci vyžádání tak čtení/zápis. Pojem vyžádání je znázorněn na obrázku 3.

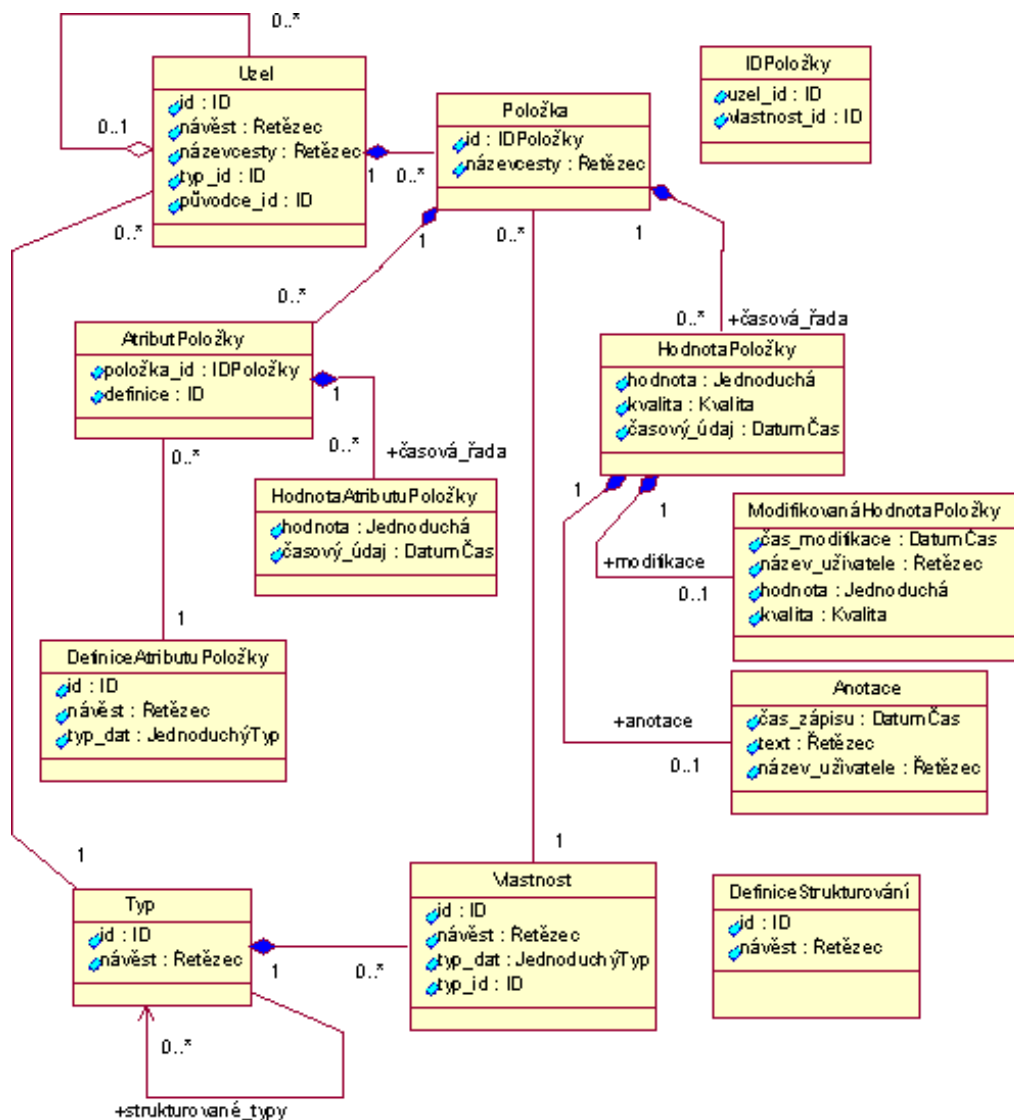


Obrázek 3 - Vyžádání dat

Vyžádání zahrnuje server, který zveřejňuje data, a klienty, kteří vyžadují přijetí těchto dat. Server *předem* nezná své klienty; server se s nimi seznamuje když klient vydává vyžádání. Jakmile je přihlášená žádost, volá server zpětně klienta, když jsou daná data k dispozici nebo když jsou aktualizována.

4.3 Datový model (normativní)

TSDA datový model popisuje, jak jsou data získávaná prostřednictvím TSDA rozhraní organizována v serveru. Realizace serveru může organizovat data různě, ale klient používající TSDA uvidí datový model tak, jak je uveden na obrázku 4.



Obrázek 4 - TSDA datový model

V modelu jsou definovány následující objekty:

- Typ, který popisuje objekty se stejnými charakteristikami časové řady, například časový interval mezi hodnotami, výpočty hodnot, apod. Proto se TSDA Typ liší od HSDA Typu. HSDA Typy obvykle popisují typy definované v IEC 61970-301, zatímco TSDA Typy neodpovídají popisu v IEC 61970-301. TSDA Typ se však obvykle odvolává na HSDA Typ.
- Vlastnost, který popisuje Položky zaznamenané jako časová řada. TSDA Vlastnost se obvykle odvolává na HSDA Vlastnost. Kterýkoliv Typ může mít libovolný počet Vlastností.
- DefiniceAtributuPoložky, který popisuje AtributyPoložky specifické pro data v časové řadě. DefiniceAtributu Položky jsou podobné Vlastnostem v tom, že popisují data v časové řadě, ale rozlišují, jaká data jsou specifická pro danou časovou řadu. Pro Typ obvykle existuje několik DefiniceAtributuPoložky.
- AtributPoložky, který popisuje, jak s těmito Položkami pracuje TSDA server, tj. jsou to metadata pro časovou řadu Položky. Proto AtributyPoložky nelze získat pomocí HSDA. Protože se Atribut může měnit v čase, tvoří vlastní časovou řadu, tj. HodnotyAtributuPoložky.
- Uzel, který je objekt, jenž má jednu či více Položek zaznamenaných jako časovou řadu. TSDA Uzel

má obvykle odpovídající HSDA Uzel.

- Položka, který je objekt mající jednu časovou řadu, tj. HodnotyPoložky. Položka má řadu AtributůPoložky popisujících časovou řadu.
- HodnotaPoložky, který je hodnotou s časovým označením a kódem kvality. Každá Položka má časovou řadu obsahující HodnotyPoložky.

Strana 11

- ModifikovanáHodnotaPoložky, který je modifikace provedená u HodnotyPoložky. Protože není přípustné měnit HodnotuPoložky, jsou místo toho změny zaznamenány pomocí ModifikovanéHodnotyPoložky.
- Anotace je text anotace, který lze doplnit k HodnotěPoložky.
- DefiniceStrukturování, který popisuje výpočty, jež lze použít u dat v časové řadě, například střední hodnota, maximální hodnota, apod.

Následující UML atributy a odkazy se váží k identifikaci:

- Typ.id, Vlastnost.id, Uzel.id, DefiniceAtributuPoložky.id a DefiniceStrukturování.id jsou jednoznačné identifikační kódy ID typu v systému. ID typ je řetězec nebo číslo s rozsahem postačujícím pro možnosti mezinárodně jednoznačných identifikátorů. Nepožaduje se však, aby id bylo mezinárodně jednoznačné. Tento id je určen pro použití počítači (stroji).
- Položka.id u typu IDPoložky je jednoznačná identifikace Položky v systému, skládající se z IDPoložky.uzel_id, které označuje Uzel, a IDPoložky.vlastnost_id, které označuje Vlastnost. Položka.id je určeno pro použití počítači (stroji).
- Typ.návěst, Vlastnost.návěst, Uzel.návěst, DefiniceAtributuPoložky.návěst a DefiniceStrukturování.návěst jsou názvy čitelné pro člověka. Typ.návěst nebo Vlastnost.název jednoznačně určuje Typ nebo Vlastnost. Uzel.návěst je jednoznačné pouze pro Uzly, které byly odvozeny z téhož původního Uzu. Návěst je určena pro používání lidmi.
- Uzel.názevcesty je jednoznačný název v systému. Obsahuje všechny návěsti od uzlu, který je jako základ. Je to stejný způsob jako u cesty souboru. Přesná struktura Uzel.názevcesty je specifická pro platformu a realizaci. Doporučuje se však, aby vždy když je to možné musela daná realizace použít stávající normy, například XCesta (XPath) z W3C. Názevcesty je určen pro používání lidmi.
- Položka.názevcesty je jednoznačný název v systému. Obsahuje všechny názvy od uzlu, který je jako základ, a konce s Vlastnost.návěst pro Vlastnost popisující danou Položku. Přesná struktura Položka.názevcesty je specifická pro platformu a realizaci. Doporučuje se však, aby vždy když je to možné musela daná realizace použít stávající normy, například XCesta (XPath) z W3C. Názevcesty je určen pro používání lidmi.

Následující UML odkazy se váží k asociacím:

- Typ.strukturované_typy vyčísluje ID Typu Uzlů, které mohou být odvozeny od Uzlu tohoto Typu. Toto se používá k zamezení tomu, aby se Typ Uzlů mohl vyskytnout jako odvozený od jiných Uzlů v hierarchii Uzlů, například jestliže Uzel Typ Stanice může obsahovat Uzly Typ Pole nebo Měření, vyčísluje Typ.strukturované_typy Typy Stanice a Měření.
- Vlastnost.typ_id označuje Typ a jemu příslušející Vlastnost. Realizuje odkaz Vlastnost.typ.
- Uzel.typ_id je Typ.id u Typu, který má Uzel.
- Uzel.původce_id označuje původní Uzel, pokud existuje. Vytváří hierarchickou strukturu Uzlů, která je základem pro tvorbu názvůcesty z návěstí na cestě do základního Uzlu.
- IDPoložky.uzel_id označuje Uzel k němuž náleží daná Položka.
- IDPoložky.vlastnost_id označuje Vlastnost charakterizující danou Položku.
- Položka.časová_řada je sekvence HodnotPoložky představující skutečnou časovou řadu.
- HodnotaPoložky.modifikace obsahuje případnou modifikaci HodnotyPoložky.
- HodnotaPoložky.anotace obsahuje případnou anotaci k HodnotěPoložky.
- AtributPoložky.položka_id a AtributPoložky.definice označuje Položku a DefiniciAtributuPoložky.
- AtributPoložky.časová_řada je sekvence HodnotAtributuPoložky představujících skutečnou časovou řadu.

Následující UML atributy obsahující data:

- Vlastnost.typ_dat popisuje typ dat Hodnota.položky, například libovolný řetězec, číslo, boolovskou hodnotu, apod.
- DefiniceAtributuPoložky.typ_dat popisuje typ dat AtributPoložky.hodnota, například libovolný řetězec, číslo, boolovskou hodnotu, apod.
- HodnotaPoložky.hodnota je hodnota typu dat Vlastnost.typ_dat. Typ dat Jednoduché je libovolný typ jenž nelze rozdělit na dílčí typy, například jakýkoliv typ čísla, řetězce či boolovská hodnota.

- HodnotaPoložky.kvalita je kód Kvality HodnotaPoložky.hodnoty. Kód Kvality udává, je-li HodnotaPoložky.hodnota platná a není-li platná předává příčinu proč není HodnotaPoložky.hodnota platná. Standardní kód Kvality je „Správná“, tj. HodnotaPoložky.hodnota, která nemá přiřazen konkrétní kód Kvality bude vždy z pohledu HSDA rozhraní brána jako „Správná“. Existuje značný počet specifikací definujících kódy Kvality, například RTU protokoly, ICCP, ELCOM, OPC DA, OMG DAIS DA, atd. Tato specifikace nedefinuje universální systém kódů Kvality, ale vychází z kódů Kvality definovaných v OMG DAIS DA.

- HodnotaPoložky.časový_údaj je čas, kdy byla HodnotaPoložky.hodnota naposledy aktualizována. Pokud této HodnotaPoložky.hodnotě není přiřazen žádný čas, je ponechán HodnotaPoložky.časový_údaj neurčen. U konfiguračních parametrů, pokud existují, se může jako HodnotaPoložky.časový_údaj použít čas, kdy byla HodnotaPoložky.hodnota vložena nebo aktualizována.
- HodnotaAtributuPoložky.hodnota je hodnota typu dat DefiniceAtributuPoložky.typ_dat. Typ dat Jednoduché je libovolný typ jenž nelze rozdělit na dílčí typy, například jakýkoliv typ čísla, řetězce či boolovská hodnota.
- HodnotaAtributuPoložky.časový_údaj je totéž jako HodnotaPoložky.časový_údaj.

4.4 Zprávy (normativní)

Užitečný obsah datových zpráv s časovou řadou mezi serverem a klienty (v obou směrech) se skládá z:

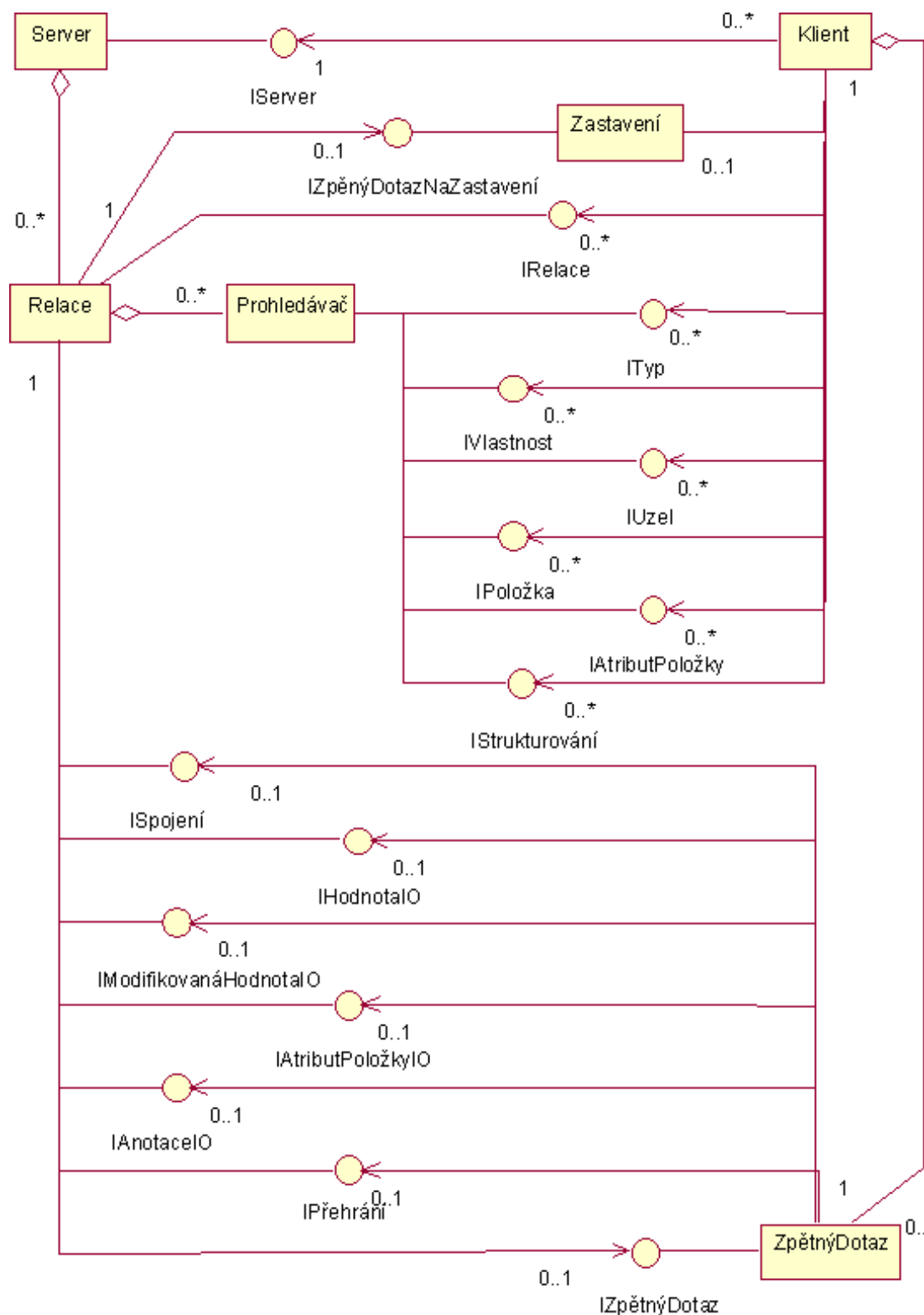
- Identifikace Položky, která jednoznačně určuje Položku v serveru (Položka_id na obrázku 4).
- Hodnot s časovým údajem, kde každá hodnota se skládá z
 - Hodnoty dat (HodnotaPoložky.hodnota na obrázku 4).
 - Kvality (HodnotaPoložky.kvalita na obrázku 4) hodnoty dat, tj. je-li daná hodnota spolehlivá. Je-li hodnota špatná, indikuje kvalita rovněž příčinu proč je tato hodnota špatná. Standardní kód kvality je „Správná“, tj. hodnota je platná.
 - Časového údaje (Položka.časový_údaj na obrázku 4), který indikuje u minulých hodnot kdy byla hodnota položky zaznamenána nebo u budoucích hodnot plánovaný čas.

Zprávy od klientů na server mohou rovněž obsahovat doplňující informace, pokud hodnota s časovým údajem musí nahradit již existující hodnotu, tj. musí být vytvořena ModifikovanáHodnotaPoložky.

4.5 Rozhraní (normativní)

4.5.1 Objekty a rozhraní

TSDA rozhraní, objekty a jejich vazby jsou znázorněny na obrázku 5.



Obrázek 5 - TSDA objekty a rozhraní

Obrázek 5 uvádí objekty a rozhraní u realizací TSDA serveru a klienta. Schéma používá notaci UML 1, kdy rozhraní jsou zobrazena jako malé kroužky spojené čarou s třídou realizující dané rozhraní. Názvy rozhraní mají rovněž počáteční velké písmeno „I“.

TSDA rozhraní se dělí do skupin:

- Prohledávání, která se používají k zjišťování dat. Prohledávací rozhraní jsou umístěna v objektu Prohledávač.
- Přístup k datům, která se používají ke skutečnému získání dat. Rozhraní pro přístup k datům jsou umístěna v objektu Relace.
- Řízení, která se používají k řízení spojení mezi Serverem a Klientem.
- Zpětný dotaz, která používá Server k předání dat na Klienty.

Konfigurace datových objektů v serveru obsahujícím data není předmětem této Části IEC 61970.

Server je objekt, který realizuje rozhraní IServer a který může mít libovolný počet Klientů jež jej používají. Objekt Serveru má řadu objektů Relace. Nutno poznamenat, že objekty Server a Relace mohou být zkombinovány do jednoho objektu, jako u OPC HDA.

Objekt Relace má řadu rozhraní:

- IRelace, které se používá pro řízení relace.
- ISpojení, které se používá k řízení spojení mezi Klientem a Serverem.
- IHodnotaIO, které se používá pro přístup k datům v časové řadě dané Položky. U těchto rozhraní nahrazují modifikace, pokud existují, původní zaznamenané hodnoty.
- IModifikovanáHodnotaIO, které se používá ke čtení starých HodnotPoložky, jak existovaly před modifikací.
- IAtributPoložkyIO, které se používá pro přístup k datům v časové řadě daného AtributuPoložky.
- IAnotaceIO, které se používá pro přístup k Anotacím dané HodnotyPoložky.
- IPřehrání, které se používá k přehrání (reprodukcí) zaznamenaných dat ve fiktivním čase.

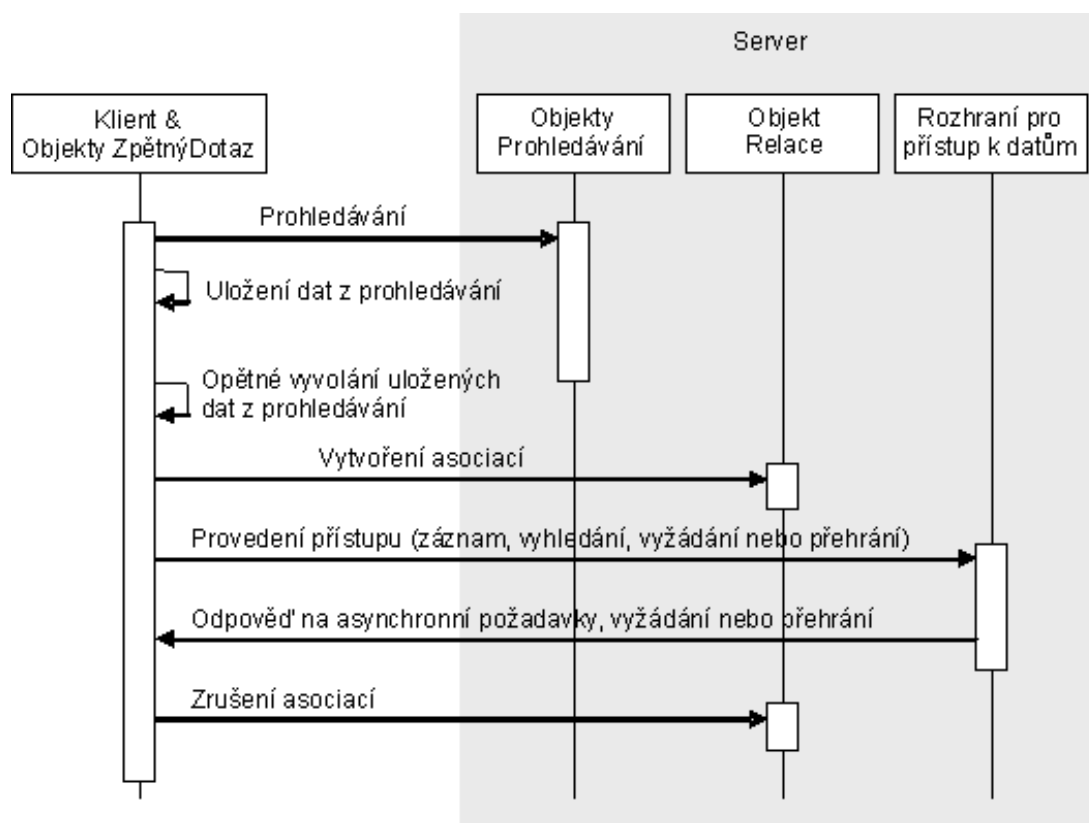
Objekt Relace může rovněž mít objekty Prohledávače mající následující rozhraní:

- ITyp je prohledávač, který se používá pro nalezení metadat o datových objektech Uzlu realizovaných TSDA serverem.
- IVlastnost je prohledávač, který se používá pro nalezení metadat o datech Položky realizovaných TSDA serverem.
- Prohledávač IUzel, který se používá pro nalezení datových objektů Uzlu jež byly konkretizovány v TSDA serveru.
- Prohledávač IPoložka, který se používá pro nalezení Položek, které existují u objektu Uzlu.
- Prohledávač IAtributPoložky, který se používá pro nalezení DefinicAtributuPoložky popisujících data v časové řadě HodnotAtributuPoložky.
- Prohledávač IStrukturování, který se používá k získání DefinicStrukturování, které existují.

Objekt ZpětnýDotaz realizuje rozhraní IZpětnýDotaz a je realizován Klientem. Každý objekt Relace může mít přidružen objekt ZpětnýDotaz. Klient může vytvořit libovolný počet Relací a sdružených objektů ZpětnýDotaz.

Typický sled interakcí mezi DA Serverem a Klientem je znázorněn na obrázku 6.

Strana 15



Obrázek 6 - Typická interakce mezi TSDA objekty

Klient obvykle začne prohledáváním serveru, aby zjistil, jaká data v časové řadě jsou k dispozici. K tomuto se použijí prohledávací rozhraní (ITyp, IVlastnost, IUzel, IPoložka, IAttributPoložky a IStrukturování). Klient vybere dílčí soubor Položek nalezených při prohledávání a uloží je pro pozdější použití. Klient, který provádí prohledávání může být editorem pro sestavení zobrazení, sestavení dialogu nebo generování databáze.

Později, například při vyvolání obrazu, budou uložené Položky opět vyvolány a použity Klientem pro osazení objektu Relace. Pokud byl vytvořen objekt Relace, lze jej použít ke čtení, zápisu nebo vyžádání dat. Požadavky na vyžádání nebo asynchronní čtení či zápis vedou ke zpětnému dotazu. U vyžádání budou zpětné dotazy probíhat dokud je Klient neukončí.

4.5.2 Rozhraní serveru a relace

Rozhraní IServer má následující atributy a metody:

- čtení pouze atributu popisujícího status Serveru, například fyzický stav serveru, čas spuštění, aktuální čas, informace prodejce, apod.

- vytvoření_relace_přístupu_k_historickým_datům(), která se používá k vytvoření objektu Relace.
- vytvoření_relace_přístupu_k_historickým_datům_pro_přehled(), která se používá k vytvoření objektu Relace. Tuto metodu lze použít, zajišťuje-li server více hierarchií. Každá hierarchie pak odpovídá určitému přehledu.
- nalezení_přehledů(), která vrací přehledy jež jsou zajišťovány serverem.
- čtení pouze atributu, který udává jaké funkce jsou zajištěny tímto rozhraním.
- vytvoření_relace_přístupu_k_historickým_datům()
- čtení pouze atributu, který udává maximální počet hodnot v časové řadě (HodnotPoložky), které může server vrátit.
- čtení pouze atributu Status serveru, který udává status Serveru, například informace prodejce, čas spuštění, aktuální čas apod.
- čtení pouze atributu, který udává funkce zajišťované rozhraním.

Strana 16

Objekt Relace realizuje rozhraní ITSDARelace, které má následující atributy a metody:

- čtení pouze atributu stavu, který popisuje status Relace, například název, čas kdy byla zahájena, aktuální čas a počet skupin.
- atribut obsahující nepovinný objekt Zastavení.
- vytvoření_prohledávače(), která se používá k vytvoření objektu Prohledávání.

4.5.3 Řídící rozhraní

Objekt Relace rovněž realizuje rozhraní ISpojení, které má následující atributy a metody:

- atribut obsahující objekt ZpětnýDotaz vytvořený Klientem.
- vytvoření(), což je metoda použitá k osazení Relace Položkami.
- odstranění(), což je metoda použitá k odstranění Položek z Relace.
- validace(), což je metoda použitá k ověření platnosti, existují-li Položky.
- zrušení(), což je metoda pro zrušení odchozích asynchronních požadavků.

4.5.4 Prohledávací rozhraní

Objekt Prohledávání realizuje následující atributy a rozhraní:

- Atribut základní_čas_prohledávání, který používá klient k nastavení základního času pro prohledávání. Prohledávač využije tento čas k lokalizaci objektů, které byly definovány v tomto čase. To je užitečné pro vyhledávání zrušených objektů, které existovaly v uplynulé době. Znamená to rovněž, že server musí též udržovat historii u zrušených objektů. Není-li k dispozici základní čas, použije se aktuální čas.
- Prohledávač ITyp.

- Prohledávač IVlastnost.
- Prohledávač IUzel.
- Prohledávač IPoložka.
- Prohledávač IAtributPoložky
- Prohledávač IStrukturování.

Rozhraní IUzel a IPoložka mají následující metody:

- nalezení(), která vrácí více informací o jednom Uzlu nebo Položce určených jejich id.
- nalezení_každé(), která vrácí více informací o několika Uzlech nebo Položkách určených jejich id.
- nalezení_pomocí_původce(), která vrácí všechny odvozené prvky od původce s určeným id.
- nalezení_pomocí_typu(), která rekurzivně vrácí všechny prvky s daným Typ.id odvozené od původce s určeným id.
- získání_názvůcest(), která převádí určitý počet id na odpovídající názvycesty.
- získání_id(), která převádí určitý počet názvůcest na id.

Rozhraní ITyp má následující metody:

- nalezení(), která vrácí více informací o Typu určeném jeho id.
- nalezení_pomocí_schéma(), která vrácí Typ.id pro všechny Typy náležející schématu určenému daným id.

Rozhraní IVlastnost má následující metody:

- nalezení(), která vrácí více informací o určitém počtu Vlastností určených jejich id.
- nalezení_pomocí_uzlu(), která vrácí všechny Vlastnosti pro Uzel s určeným id.
- nalezení_pomocí_typu(), která vrácí všechny Vlastnosti pro Typ s určeným id.

Rozhraní IAtributPoložky a IStrukturování mají následující metody a data:

- nalezení(), která vrácí více informací o určitém objektu určeném jeho id.
- nalezení_všech(), která vrácí informace o všech objektech.
- Definování všech identifikací objektů (ID Atributu Položky a Strukturování), které existují.

4.5.5 IO rozhraní

IO rozhraní zajišťují následující funkce:

- Synchronní čtení a zápis.
- Asynchronní čtení a zápis.
- Vyžádání.
- Přehrání.

IO rozhraní jsou:

- IHodnotaIO.
- IModifikovanáHodnotaIO

- IAttributPoložkyIO.
- IAnotaceIO.
- IPřehrání.

Synchronní rozhraní IHodnotaIO má následující metody:

- synchronní_čtení_prvotních_dat(), která se používá k synchronnímu čtení dat zaznamenaných nebo aktualizovaných ve stanoveném časovém intervalu.
- synchronní_čtení_zpracovaných_dat(), která se používá k synchronnímu čtení dat s použitím výpočtu Strukturování ve stanoveném časovém intervalu.
- synchronní_čtení_podle_času(), která se používá k synchronnímu čtení dat v několika stanovených časech.
- synchronní_vložení(), která se používá k synchronnímu vložení nových hodnot dat v časové řadě (tj. HodnotPoložky).
- synchronní_nahrazení(), která se používá k synchronnímu nahrazení existujících hodnot dat v časové řadě (tj. HodnotPoložky).
- synchronní_vložení_nahrazení(), která se používá k synchronnímu nahrazení existujících hodnot dat v časové řadě (tj. HodnotPoložky) nebo, pokud určitá hodnota neexistuje, k jejímu vložení jakožto nové.
- synchronní_zrušení_prvotních_dat(), která se používá k synchronnímu zrušení zaznamenaných dat ve stanoveném časovém intervalu.
- synchronní_zrušení_podle_času(), která se používá k synchronnímu zrušení dat zaznamenaných ve stanovených časech.

Asynchronní rozhraní IHodnotaIO má následující metody:

- asynchronní_čtení_prvotních_dat(), která se používá k asynchronnímu čtení dat zaznamenaných nebo aktualizovaných ve stanoveném časovém intervalu. Rozhraní zpětného dotazu použité u této metody je při_kompletním_čtení().
- asynchronní_čtení_zpracovaných_dat(), která se používá k asynchronnímu čtení dat s použitím výpočtu Strukturování ve stanoveném časovém intervalu.
- asynchronní_čtení_podle_času(), která se používá k asynchronnímu čtení dat v několika stanovených časech. Rozhraní zpětného dotazu použité u této metody je při_kompletním_čtení().
- asynchronní_vložení(), která se používá k asynchronnímu vložení nových hodnot dat v časové řadě (tj. HodnotPoložky). Rozhraní zpětného dotazu použité u této metody je při_kompletní_aktualizaci().
- asynchronní_nahrazení(), která se používá k asynchronnímu nahrazení existujících hodnot dat v časové řadě (tj. HodnotPoložky). Rozhraní zpětného dotazu použité u této metody je při_kompletní_aktualizaci().
- asynchronní_vložení_nahrazení(), která se používá k asynchronnímu nahrazení existujících hodnot dat v časové řadě (tj. HodnotPoložky) nebo, pokud určitá hodnota neexistuje, k jejímu vložení jakožto nové. Rozhraní zpětného dotazu použité u této metody je při_kompletní_aktualizaci().
- asynchronní_zrušení_prvotních_dat(), která se používá k asynchronnímu zrušení zaznamenaných dat ve stanoveném časovém intervalu. Rozhraní zpětného dotazu použité u této metody je při_změně_dat().
- asynchronní_zrušení_podle_času(), která se používá k asynchronnímu zrušení dat zaznamenaných ve stanovených časech. Rozhraní zpětného dotazu použité u této metody je

Rozhraní vyžádání IHodnotaIO má následující metody:

- vyžádání_prvotních_dat(), která se používá ke čtení dostupných zaznamenaných prvotních dat a budoucích dat, která ještě nebyla zaznamenaná. Dostupná data jsou dodána okamžitě, zatímco nová data jsou dodána jakmile jsou zaznamenaná. Rozhraní zpětného dotazu použité u této metody je při_změně_dat().
- vyžádání_zpracovaných_dat(), která se používá ke čtení zaznamenaných dat s použitím výpočtu strukturování a budoucích dat, která ještě nebyla zaznamenaná. Dostupná data jsou dodána okamžitě, zatímco nová data jsou dodána jakmile jsou zaznamenaná. Rozhraní zpětného dotazu použité u této metody je při_změně_dat().

Synchronní rozhraní IModifikovanáHodnotaIO má následující metodu:

- synchronní_čtení_modifikovaných_dat(), která se používá k synchronnímu čtení dat zaznamenaných ve stanoveném časovém intervalu. Existující aktualizace (tj. ModifikovanéHodnotyPoložky) jsou ignorovány.

Asynchronní rozhraní IModifikovanáHodnotaIO má následující metodu:

- asynchronní_čtení_modifikovaných_dat(), která se používá k asynchronnímu čtení dat zaznamenaných ve stanoveném časovém intervalu. Existující aktualizace (tj. ModifikovanéHodnotyPoložky) jsou ignorovány. Rozhraní zpětného dotazu použité u této metody je při_kompletním_čtení_modifikovaných_dat().

Synchronní rozhraní IAtributPoložkyIO má následující metodu:

- synchronní_čtení_atributu(), která se používá k synchronnímu čtení HodnotAtributuPoložky ve stanoveném časovém intervalu.

Asynchronní rozhraní IAtributPoložkyIO má následující metodu:

- asynchronní_čtení_atributu(), která se používá k asynchronnímu čtení HodnotAtributuPoložky ve stanoveném časovém intervalu.

Synchronní rozhraní IAnotaceIO má následující metody:

- synchronní_čtení(), která se používá k synchronnímu čtení anotací ve stanoveném časovém intervalu.
- synchronní_vložení(), která se používá k synchronnímu vložení anotací do stanovených HodnotPoložky.

Asynchronní rozhraní IAnotaceIO má následující metody:

- asynchronní_čtení(), která se používá k asynchronnímu čtení anotací ve stanoveném časovém intervalu. Rozhraní zpětného dotazu použité u této metody je při_kompletním_čtení_anotace().

- `asynchronní_vložení()`, která se používá k asynchronnímu vložení anotací do stanovených HodnotPoložky. Rozhraní zpětného dotazu použité u této metody je `při_kompletním_vložení_anotace()`.

Asynchronní rozhraní IPřehrání má následující metody:

- `přehrání_prvotních_hodnot_aktualizací()`, která se používá k asynchronnímu přehrání dat zaznamenaných od stanoveného časového intervalu. Rozhraní zpětného dotazu použité u této metody je `při_přehrání()`.
- `přehrání_zpracovaných_dat_aktualizací()`, která se používá k asynchronnímu přehrání dat s použitím výpočtu Strukturování od stanoveného časového intervalu. Rozhraní zpětného dotazu použité u této metody je `při_přehrání()`.

4.5.6 Rozhraní klienta

Pro zajištění asynchronních požadavků a vyžádání musí klient realizovat objekt ZpětnýDotaz, který má následující metody:

- `při_změně_dat()`, která se používá pro příjem zpětných dotazů na vyžádání Klientem.
- `při_kompletním_čtení()`, která se používá pro příjem asynchronních odpovědí na čtení.
- `při_kompletní_aktualizaci()`, která se používá pro příjem asynchronních odpovědí na vložení, aktualizaci nebo zrušení.
- `při_kompletním_čtení_modifikovaných_dat()`, která se používá pro příjem asynchronních odpovědí na čtení u nemodifikovaných hodnot.
- `při_kompletním_čtení_anotace()`, která se používá pro příjem asynchronních odpovědí na čtení u anotací.
- `při_kompletním_vložení_anotace()`, která se používá pro příjem asynchronních odpovědí pro vkládané anotace.
- `při_přehrání()`, která se používá pro příjem dat pro přehrání.

4.6 Mapování TSDA

TSDA rozhraní mohou přenášet data pocházející ze zdroje dat vyhovujícího IEC 61970-301. U serveru vyhovujícího IEC 61970-301, který zajišťuje TSDA, musí být rozhraní použita takto:

- Prohledávací rozhraní IUzel předkládá objekty, obsahující informace v časové řadě. Třídy Měření v IEC 61970-301 běžně definují takovéto objekty, například Analogové, AnalogováHodnota, apod. Objekty mohou být hierarchicky uspořádány ve vymezené struktuře. Pokud tomu tak je, jde o FyzickýPřehled uvedený v IEC 61970-402. Skutečná hierarchie je transparentní vůči prohledávacímu rozhraní IUzel a mohou být předloženy jiné hierarchické struktury.
- Prohledávací rozhraní IPoložka předkládá hodnoty vlastností v časové řadě, které existují v

TSDA severu. Takovéto vlastnosti mohou být definované třídami v IEC 61970-301, například Analogový.normálníHodnota=„500“, AnalogováHodnota.hodnota=„450“, apod.

- Prohledávací rozhraní ITyp předkládá typy dat v časové řadě, které TSDA server zajišťuje, například plánované hodnoty výroby, minutové hodnoty, hodinové hodnoty, apod. IEC 61970-301 aktuálně neobsahuje všechny definice těchto druhů dat.
- Prohledávací rozhraní IVlastnost předkládá atributy tříd (metadata) definované pro daný Typ, tj. vlastnosti jež existují v TSDA serveru pro konkrétní Typ. Takovéto vlastnosti mohou být definovány pomocí tříd v IEC 61970-301, například Analogový.normálníHodnota, AnalogováHodnota.hodnota apod.

Strana 20

Příloha ZA (normativní)

Normativní odkazy na mezinárodní publikace a na jim příslušející evropské publikace

Pro používání tohoto dokumentu jsou nezbytné dále uvedené referenční dokumenty. U datovaných odkazů platí pouze citovaná vydání. U nedatovaných odkazů platí poslední vydání referenčního dokumentu (včetně změn).

POZNÁMKA Pokud byla mezinárodní publikace upravena společnou modifikací, vyznačenou pomocí (mod), používá se příslušná EN/HD.

<u>Publikace</u> <u>Rok</u>	<u>Rok</u>	<u>Název</u>	<u>EN/HD</u>
	IEC 61970-1	⁻¹⁾ Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 1: Směrnice a obecné požadavky	EN 61970-1 2006 ²⁾
	IEC/TS 61970-2	⁻¹⁾ Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 2: Výklad zvláštních výrazů	CLC/TS 61970-2 2005 ²⁾
	IEC 61970-301	2003 Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 301: Základ obecného informačního modelu (CIM))	EN 61970-301 2004
	IEC/TS 61970-401	⁻¹⁾ Rozhraní aplikačního programu pro systémy řízení elektrické energie (EMS-API) - Část 401: Struktura specifikace rozhraní	- -

složek (CIS)

IEC 61970-402	- ³⁾	Rozhraní aplikačního programu pro systémy - řízení elektrické energie (EMS-API) - Část 402: Specifikace rozhraní složek (CIS) - Obecné služby
OMG HDAIS	2003	Přístup k historickým datům z průmyslových - systémů (HDAIS)
OMG DAF	2002	Prostředky pro přístup k datům (DAF) v řídicím - systému společnosti (UMS)
OMG DAIS	2002	Sběr dat z průmyslových systémů (DAIS) -
OPC HDA	2003	Zákaznická specifikace rozhraní - pro OPC přístup k historickým datům

1) Nedatovaný odkaz.

2) Platná edice v době vydání.

3) Ve stádiu návrhu.

-- Vynechaný text --