



TOPOBASE™

c-plan®

TOPOBASE™

White Paper

TB Jobs

XANADU

Úvod

Pozadí

V dnešní době zažívají geografické informační systémy (GIS) bouřlivý rozvoj a začínají být široce rozšířeny. Vedle tradičního použití provádí současné systémy GIS i další úkoly, například zjišťování polohy apod.

Vedle rozšiřování do nových oblastí expandují systémy GIS i ve svém tradičním teritoriu. Zřejmou tendencí je snaha o sloučení heterogenních dat do jediného podnikového informačního systému.

Tento vývoj vyžaduje změny v samotné podstatě softwaru GIS. Požadavky v oblasti spolehlivosti a výkonu se ve skutečnosti neliší od standardů pro jakýkoliv jiný průmyslový víceuživatelský serverový software. Dostupnost 24 hodin denně 7 dní v týdnu, bezpečné oddělení dat, izolace uživatelů, robustní mechanismy zajišťující konzistenci informací - tyto aspekty vnáší do sféry softwaru GIS novou dimenzi. A jediným způsobem, jak odolat rostoucí zátěži a nově vznikajícím požadavkům, je vytvoření softwaru, který zvládne všechny stanovené úkoly pomocí specializovaných návrhových a implementačních řešení.

Výzva

Počítačová věda již objevila produkt, který naplňuje výše uvedené požadavky a poskytuje komplexní prostředí, které může obsahovat velké množství vzájemných vazeb a mnoho objektů. Konzistence těchto objektů není omezena pouze stavem jediného objektu, ale rozhodujícím způsobem závisí i na stavu ostatních.

Nízkoúrovňová omezení zde nabízejí logiku informací na abstraktní obchodní úrovni spolu s obchodními pravidly na vysoké úrovni, která jsou někdy tak složitá, že je lze jen těžko formalizovat.

Název takového produktu je Version Control System.

Správa verzí

Již velmi dlouho se při vývoji softwaru, ve kterém každý objekt zpracovává více uživatelů a ve kterém se logická granularita snižuje až na úroveň znaků, používá správa verzí. Pokud se změnila signatura metody a v celém projektu nebyla odpovídajícím způsobem změněna volání metody, program se nezkompile.

Správa verzí jiných než textových dat je ještě mnohem složitější. Samozřejmě existuje definovaná struktura a omezená sada dobře formalizovaných pravidel, která se používají na textové znaky, čáry a soubory. Zpracování různých textových formátů nepřináší nic nového, jedná se pouze o převodní filtry.

Data v systému GIS mají naproti tomu zcela odlišnou podstatu. Opět se zde objevují nízkoúrovňová omezení (například vazba mezi prostorovými objekty má smysl pouze v případě, že se oba nacházejí v jistém stavu) a obchodní pravidla na vysoké úrovni (jiná vazba je platná pouze v případě dalších změn).

Správa verzí prostorových dat v jejich obchodním kontextu proto rozhodně není jednoduchou úlohou.

Dlouhodobé transakce

Vzájemné vazby mezi daty jsou často tak složité, že jejich ruční vytváření koncovým uživatelem by trvalo dny i týdny. V těchto případech je nutná spolupráce celého týmu. Protože však nesmí dojít k porušení konzistence informací, musí být pracovní prostor oddělen od hlavních výrobních dat.

Objem dat v průmyslových aplikacích dosahuje gigabajtů a terabajtů. Nejjednodušší řešení, datový snímek, je zde proto nepoužitelný.

Mezi těmito požadavky a funkčností očekávanou od databázového transakčního podsystému existují zřejmé podobnosti. Existují zde ovšem dva rozdíly: 1) "transakce" na obchodní úrovni trvají mnohem déle a musí přestat více restartů databáze – mechanismus databázových transakcí se stává v tomto případě nepoužitelným; a 2) systém musí zaručovat konzistenci dat na vyšší obchodní úrovni.

Řešení

Když jsou vývojáři vybaveni takovými technologiemi jako správa verzí a transakce, mohou naplnit všechny požadavky kladené na současný software GIS. Společnost C-plan nabízí vlastní implementaci takového produktu dlouhodobých transakcí a správy verzí s názvem Topobase Jobs, který navíc přesně odpovídá dalším zvláštnostem terénních aplikací.

Klíčové koncepty

Hlavním hráčem v produktu Topobase Jobs je samostatný proces nazývaný dlouhá transakce. Data se mohou změnit pouze v průběhu mutace, libovolná data jsou čtena pouze mimo všechny mutace. Taková data obvykle obsahují důvěryhodnou výrobní informaci, která je ve většině případů platná. Jiným příkladem jsou stará, málo se měnící data.

V produktu mohou být data ve stavu Live, Pending nebo Project. Data ve stavu Live jsou pouze pro čtení a jsou společná pro celou společnost (jsou viditelná stejným způsobem ve všech mutacích i mimo všechny mutace). Po zahájení procesu mutace dat je každá změna dat izolována v této mutaci a není viditelná v jiných dlouhých transakcích a mimo dlouhé transakce. Poté, co jsou změny připraveny ke kontrole a schválení, přechází dlouhá transakce (a všechna změněná data v mutaci) do stavu Pending.

Data mohou na schválení čekat velmi dlouho a v tomto stavu jsou již viditelná ve všech dalších projektech a mutacích. Tento stav je velmi podobný stavu dat Live, s jedinou výjimkou: i nadále je zde informace o vytvářející dlouhé transakci a celá tato dlouhá transakce může být zrušena nebo vrácena zpět do stavu Project pro pozdější zpracování.

Jamile je mutace schválena, je projekt připraven na převod do výrobních dat. Taková dlouhá transakce potom změní svůj stav na Live. V rámci nalezení možné ztráty dat se použije inteligentní sloučení. Spolehlivě vyhledá konflikty, ale zbytečně neobtěžuje uživatele, protože může určité typy konfliktů vyřešit automaticky. Uživatel může kontrolovat a opravit řešení nabídnutá produktem.

Funkce

Počet souběžných procesů/projektů mutace	Neomezený*
Počet stavů dat	4: Live, Pending, Projects, Deleted
Granularita Job-capable	Schéma databáze
Granularita Job-enable	Tabulka
Reverzibilní job-enabling	Ano
Exportovatelné schéma Job-capable	Ano

* Nemůže být větší než prostor identifikátorů typu NUMBER(10)

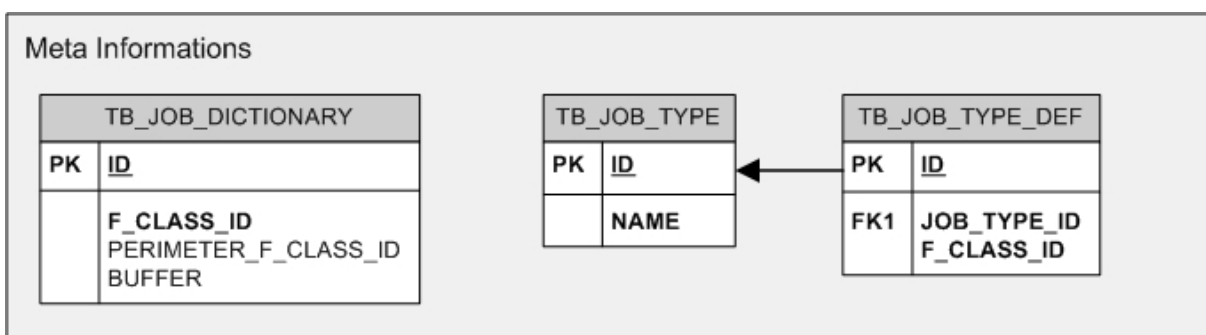
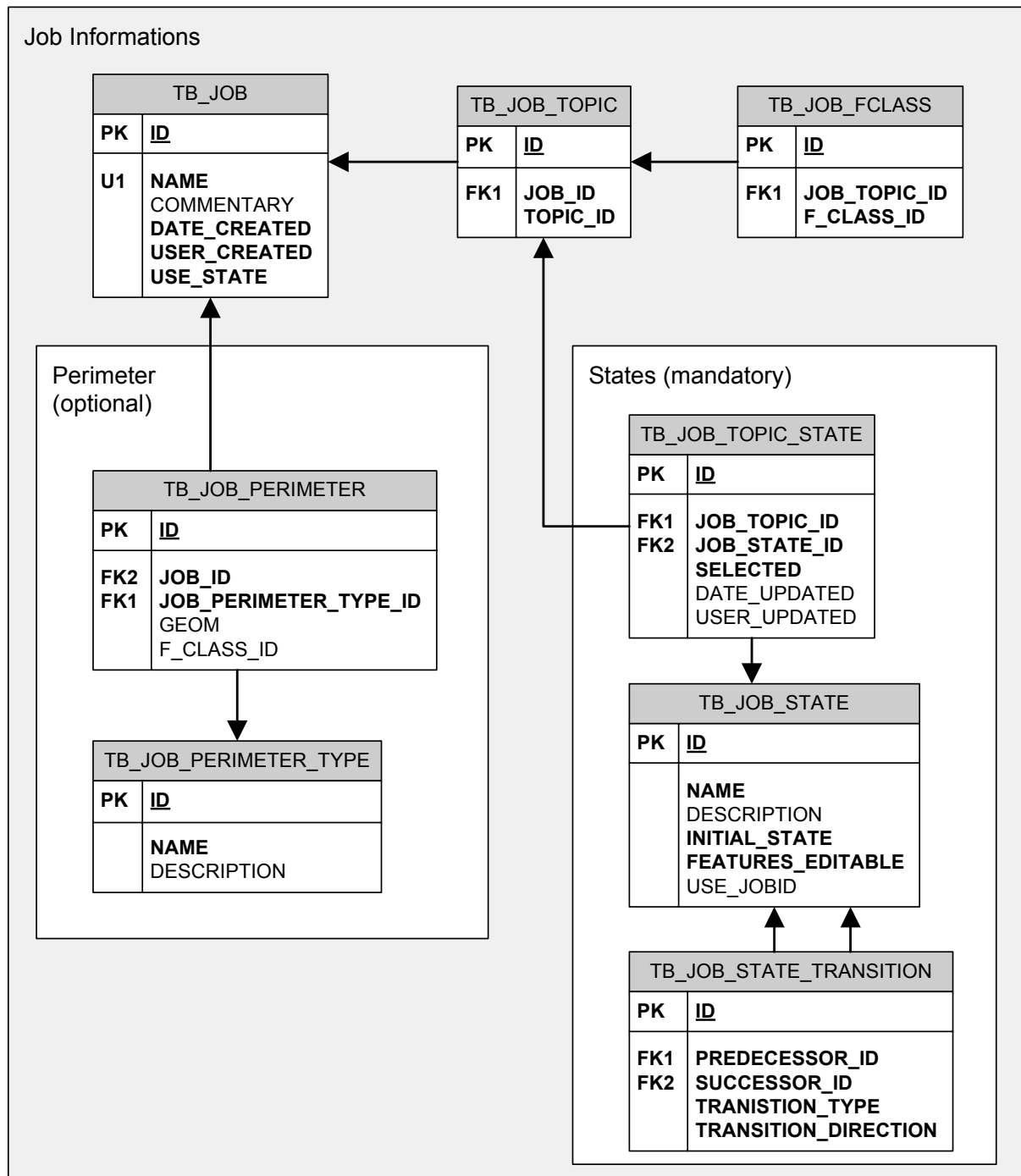
Technický přehled

Strana serveru

Tato kapitola nabízí přehled hlavní infrastruktury, která umožňuje fungování Job na straně serveru.

XANADU

Model metadat



Konfigurace stavů dlouhých transakcí

TB_JOB_STATE

Initial State (0 / 1) : Hodnota označující, zda se jedná o počáteční stav všech dlouhých transakcí. Důležité: může existovat pouze jeden počáteční stav!

Features Editable (0 / 1) : Hodnota označující, zda může být prvek upraven (vložen, aktualizován, odstraněn).

TB_JOB_STATE_TRANSITION

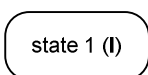



Predecessor : Jedná se o cizí klíč na předchozí stav ve stavovém diagramu.

Successor : Jedná se o cizí klíč na následující stav ve stavovém diagramu.

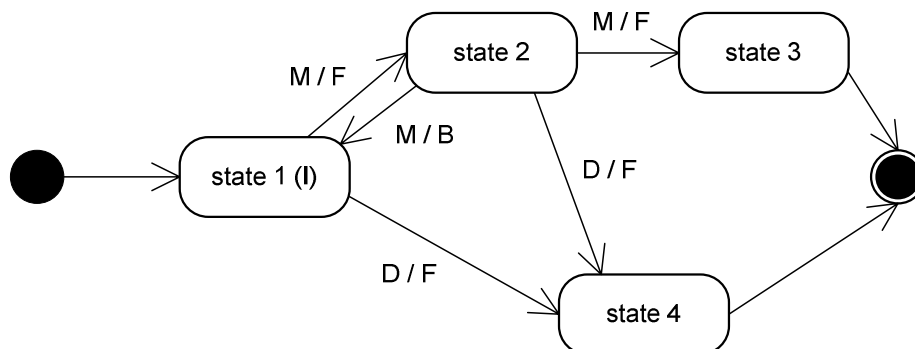
Transition Type (M / D / N) : Hodnota označuje akci softwaru při provedení přechodu stavu. M = sloučit prvky, D = odstranit upravené prvky (zpět), N = nedělat nic.

Transition Direction (F / B) : Hodnota označuje, zda se jedná o dopředný (F) nebo zpětný (B) přechod. Dopředný přechod může být procesem sloučení, zpětný přechod může být vrácením procesu sloučení. Důležité: přechod typu D, odstranit, nemůže být nikdy zpětný!

Stavy a přechody stavů mohou být snadno nakresleny pomocí stavového diagramu UML. Na dolním obrázku je příklad takového stavového diagramu s odpovídající legendou.

Značky v diagramu	
	Tato značka představuje stav dlouhé transakce. Toto je počáteční stav.
	Tato značka představuje přechod stavu. Směr přechodu může být dopředný nebo zpětný. Zde uvedený typ je M pro sloučení.
	Tato značka představuje počáteční bod. Následníkem počátečního bodu může být pouze jeden počáteční stav. To znamená, že zde může existovat pouze jeden počáteční stav.
	Tato značka představuje koncový bod.

Vzorový diagram se čtyřmi stavy:



DBMS_RLS

Hlavním účelem balíku DBMS_RLS je přidání klauzule WHERE do všech příkazů SELECT, INSERT, UPDATE nebo DELETE. Tato klauzule se nazývá predikát. Její definice je uložena v zásadách.

Na tomto balíku je založen VPD. Je například možné definovat predikát dynamicky pomocí parametru USER. Tento způsob se používá v případě, že se mají uživatelé zobrazit pouze některé řádky tabulky (-> virtuální PRIVATE databáze).

Koncept TB3-Job má stejný cíl: zobrazit pouze některé řádky podle vybrané dlouhé transakce. Pro TB3-Job není predikát založen na informacích o uživateli, ale na informacích o dlouhé transakci.

Predikát není nic jiného, než funkce pl/sql vracející klauzuli WHERE, která se přidá ke všem příkazům SELECT, INSERT, UPDATE nebo DELETE. Tato funkce se aktivuje přidáním zásady (DBMS_RLS.ADD_POLICY) k odpovídající tabulce nebo pohledu. Zásada je definována cílovou tabulkou nebo pohledem, vlastníkem schématu cílové tabulky nebo pohledem a funkcí (predikátovou funkcí).

Kontext

Kontext může být zobrazen jako globální proměnná pro sezení Oracle.

Systém ukládá různé kontextové informace:

Id dlouhé transakce

Stav prvku pro aktuální dlouhou transakci

Informace o tom, zda je dlouhá transakce použita v predikátu

Atributy v datové tabulce

Informace dlouhé transakce jsou uloženy v TB_JOB__VERSION

Hlavní tabulka [tabulka] obsahuje pouze atribut JOB_VERSION.

Do tabulky TB_JOB__VERSION je přidáno 6 atributů:

Pole	Popis
JOB_VERSION	Povinné: verze objektu
JOB_OLD_VERSION	Volitelné: předchozí verze objektu
JOB_ID	Povinné: NUMBER
JOB_OPERATION_ID	Povinné: NUMBER 1: INSERT, 2: UPDATE, 3: DELETE
OS_USER_NAME	Uživatel operačního systému
OPERATION_DATE	Datum operace
STATE	Stav prvku.

Při inicializaci tabulky dlouhé transakce je JOB_VERSION nové číslo (vytvořené pomocí sekvence TB_JOB_VERSION_S) pro každý objekt, JOB_OLD_VERSION je -1 pro každý objekt, JOB_ID je 0 pro každý objekt, JOB_OPERATION_ID je 1 pro každý objekt.

Čísla verzí (JOB_VERSION) jsou spravována pomocí sekvence. JOB_VERSION je primární klíč.

Predikát

Predikát je možné definovat pro každý koncept dlouhé transakce. Je třeba změnit pouze funkci pl/sql.

Triggery v datové tabulce

Uživatel schématu obsahuje pro každou zapnutou třídu prvků soubor [FeatureClass]_PK. Tento balík obsahuje vnořenou tabulku třídy prvků a proceduru pro uložení řádků do této tabulky. Druhá procedura načte hodnoty z této tabulky, vkládá je do třídy prvků a odstraňuje je z vnořené tabulky.

Jedna proměnná definuje, zda se s příkazem INSERT třídy prvků pracuje jako v funkci dlouhé transakce nebo ne (BOOLEAN [enableInsertTrigger]).

Před provedením akce triggerů se volá funkce "[Ensure](#)", která provede některé nezbytné kontroly.

Insert – Trigger:

Název triggeru se skládá z názvu třídy prvků a přípony "_JBI".

Jedná se o BEFORE INSERT každé události řádku a provede aktualizaci [FeatureClass].job_version novou hodnotou sekvence pro verzi.

Potom se vloží atribut 'job_version' s novou hodnotou ze sekvence a do TB_JOB_VERSION se vloží nový řádek:

JOB_VERSION	JOB_OLD_VERSION	JOB_ID	JOB_OPERATION_ID
Hodnota sekvence	-1	Skutečné JOB_ID	1

Update – Trigger:

Aktualizace je tvořena dvěma triggerem. Jedním triggerem řádku BEFORE UPDATE (_JBU) a jedním triggerem příkazu AFTER UPDATE (_JAU).

Trigger řádku BEFORE UPDATE zachová starý řádek a nový řádek s hodnotami z aktualizace.

Musí proběhnout před událostí, aby mohl zachovat původní řádek nastavením všech hodnot :new na hodnoty :old.

Nový řádek získá 'job_version' ze sekvence a je zachován ve vnořené tabulce balíku.

Potom se nový řádek vloží do TB_JOB_VERSION:

JOB_VERSION	JOB_OLD_VERSION	JOB_ID	JOB_OPERATION_ID
Hodnota sekvence	:OLD.job_version	Skutečné JOB_ID	2

Trigger AFTER UPDATE nastaví proměnnou 'enableInsertTrigger' na FALSE a vloží uchované řádky z vnořené tabulky do třídy prvků.

Potom se proměnná 'enableInsertTrigger' nastaví na TRUE.

Delete – Trigger:

Odstranění je tvořeno dvěma triggerem. Jedním triggerem řádku BEFORE DELETE (_JBD) a jedním triggerem příkazu AFTER DELETE (_JAD).

Trigger řádku BEFORE DELETE zachová starý řádek a nový řádek s hodnotou 'job_version' ze sekvence ve vnořené tabulce pouze v tom případě, že prvek nebyl vytvořen v této dlouhé transakci.

Potom se nový řádek vloží do TB_JOB_VERSION:

JOB_VERSION	JOB_OLD_VERSION	JOB_ID	JOB_OPERATION_ID
Hodnota sekvence	:OLD.job_version	Skutečné JOB_ID	3

Pokud byl prvek vytvořen touto dlouhou transakcí, budou řádky z TB_JOB_VERSION a TB_UFID odstraněny. Nebude zachován žádný řádek. Může začít událost odstranění.

Trigger AFTER UPDATE nastaví proměnnou 'enableInsertTrigger' na FALSE a vloží uchované řádky z vnořené tabulky do třídy prvků.

Potom se proměnná 'enableInsertTrigger' nastaví na TRUE.

Procedura – Ensure:

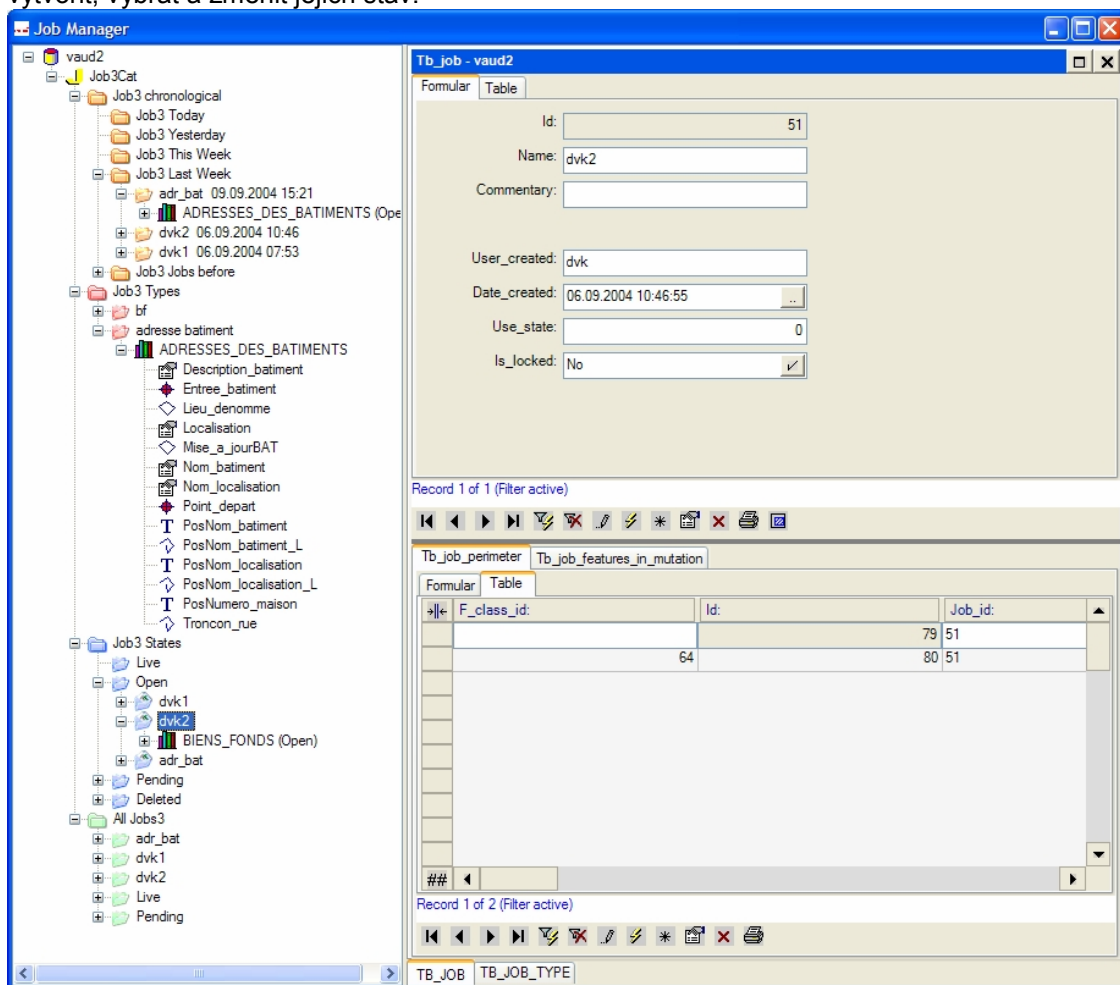
Tato procedura kontroluje následující proměnné:

- Pokud je aktuální JOB_ID = 1, zobrazí se chybové hlášení sdělující, že je vybrána kořenová dlouhá transakce
- Pokud je skutečné JOB_ID < 0, zobrazí se chybové hlášení sdělující, že není vybrána žádná dlouhá transakce a není možné provést žádnou úpravu
- Pokud je dlouhá transakce uzamčena, zobrazí se chybové hlášení sdělující, že není možné provést žádnou úpravu, když je dlouhá transakce uzamčena
- Pokud není možné upravovat předmět třídy prvků (PENDING, LIVE, DELETED), zobrazí se chybové hlášení sdělující, že není možné upravovat předmět třídy prvků.

Strana klienta

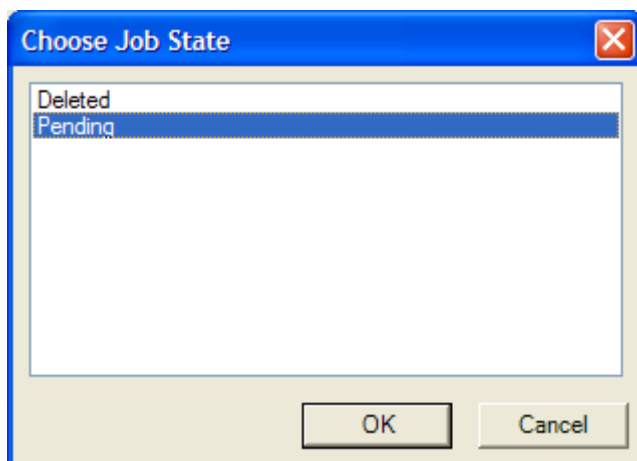
Správa dlouhých transakcí pomocí grafického uživatelského rozhraní

Spravuje dlouhé transakce v nástroji pro správu dlouhých transakcí podobném Průzkumníku. Typy dlouhých transakcí je možné vytvořit, upravit nebo odstranit. Dlouhé transakce lze vytvořit, vybrat a změnit jejich stav.



Správa sloučení a řešení konfliktů pomocí grafického uživatelského rozhraní

Změnu stavu dlouhých transakcí je možné provést vybráním odpovídajícího stavu.



Pokud se vyskytnou konflikty, program je nalezne a zobrazí.

Závěr

Popisované řešení splňuje všechny složité požadavky aktuálních aplikací Geoinformačních systémů. Především je třeba ho vzít v úvahu v takových případech, kdy jde o spolupráci a separaci změn spolu se spolehlivou a inteligentní ochranou před ztrátou dat.

Další informace

Další informace o funkcích systému Topobase od společnosti C-plan jsou dostupné na internetové adrese:

<http://www.c-plan.com/>

a u prodejců produktu.

latest edition: 23.09.2003 vytvořil CMO

c-plan, with a headcount of more than 60, is one of the leading GIS solution providers in Europe. The TOPOBASE™ solution developed by c-plan is based on the ORACLE Spatial 9i database and on Autodesk graphic technology. With these standards, there are application modules available for all work in planning, processing, managing and analyzing geodata. Compliance with OpenGIS specifications means easy, cost-efficient and, above all, future-proof work processes. Three locations – Gümliigen at Berne (CH), Steinheim/Stuttgart (D) and Münster (D) – and qualified distribution partners in Switzerland, Germany, Austria, Sweden, Belgium, Italy and Luxemburg, Iceland and USA ideally serve the GIS market.

For further information:

c-plan® ag

Worbstrasse 223

CH - 3073 Gümliigen

Phone: +41 31 958 20 20

Fax: +41 31 958 20 22

<http://www.c-plan.com>

XANADU a.s., České Budějovice, info@xanadu.cz