

1)

Jaké hodnoty se budou používat pro

GlobalUniqueIdentifier,

URN:catalog:[InstitutionCode]:[CollectionCode]:[CatalogNumber]

InstitutionCode,

*S největší pravděpodobností zkratky herbářů (pro jihočeské muzeum CB) a tam, kde nejsou zkratky herbářů, pak zažité zkratky (např. Agentura ochrany přírody a krajiny – AOPK, Správa ochrany přírody SOP). U osob by se to generovalo ze jména a příjmení jako do teď (1. 3 písmena ze příjmení + 1. 3 písmena ze jména, takže osoba: Vladimír Hans – InstitutionCode: HanVla)*

CollectionCode

*Plantae, popř. bych to nechal na uživateli, jestliže tato data vznikla v rámci projektů či grantů a byly by potřeba to nějak odlišit.*

CatalogNumber?

*Přírůstkové číslo.*

Podrobný popis polžek viz:

[http://darwincore.calacademy.org/Documentation/DarwinCore2Draft\\_v1-4\\_HTML](http://darwincore.calacademy.org/Documentation/DarwinCore2Draft_v1-4_HTML)

2)

Nehodily by se i jiné položky z DarwinCore2?

**InformationWithheld** – nástřel kategorií:

*veřejně publikovatelné,*

*přístup pouze přispěvatelé*

*neveřejné (pravděpodobně by se tyto položky nedostávaly do centrální databáze – co tam není, nemůže uniknout ani být zneužito :-))*

**Country** – v případě, že by tam byly i nálezy z ciziny (např. Rakousko)

3)

Jaký je význam položky "pas" v tabulce Lokality?

*žádný*

4)

Jaká verze Checklistu se má použít?

viz str. 15 UseCase Analýzy <http://urtax.ms.mff.cuni.cz/plantlore/data/uc.pdf>

*V případě klasického zápisu taxonů bych se přikláněl k první variantě, kde se zobrazují všechny kytky. Alternativní přístup, kde si nejdříve musíš vybrat taxony, které budeš používat, je vhodný u škrtačích seznamů<sup>1</sup>, které bych chtěl rozhodně implementovat (viz stávající verze db).*

5)

Jaké formáty importu a exportu se mají podporovat? (kromě stávající db. samozřejmě).

*Export do nějakého rozumného formátu, nejlépe textového (csv, XML). Exportované sloupce by si určil uživatel. U CSV je nutné řešit kódové stránky a konce řádků (Unix x*

---

1 **Škrtačí seznam** se vytváří při floristickém průkumu nějakého území. Š.s. je seznam všech taxonů, které v řešeném území můžou růst. Dané území si rozčleníš podle středoevropské mapovací sítě a pro každý čtverec určité velikosti máš jeden š.s. Pak ten čtverec procházíš a zaškrtaáš nalezené taxony. Přístup š.s. je efektivnější, než si to všechno psát do zápisníku.

Windows, MacOS?).

*Import z csv souboru, který by měl předem danou strukturu. Moc se mi to nelíbí, ale chápu, že pro některé přispěvatele asi nemá smysl installit PlantLore. A ještě k tomu jsou botanici počítačoplaší :-).*

*A samozřejmě i výměnný formát, kterým by se daly přenášet data mezi jednotlivými instalacemi PlantLore.*

*Abych nezapomněl, export pro DMAP.*

Pro prenos nalezovych dat vypada jako jediny standard HISPID3, viz <http://plantnet.rbgsyd.nsw.gov.au/HISCOM/HISPID/HISPID3/hispidright.html>

HISPID vypadá celkem použitelně, jsem pro jeho implementaci pro přenos dat.

6)

Kontaktovali nás lidé z Janitoru, jestli nechceme místo vyvoje Plantlore pracovat s nimi. *Toto rozhodnutí nechám na vás.*

Jaky je vlastne nas vztah k Janitoru?

*Vztah je takový, že Janitor je koncipován pro uživatele, kteří mají GPS popř. digitální mapy (nutnost zadávat souřadnice nálezu), PlantLore by měla být určena pro uživatele, kteří nemůžou (mají pouze papírové turistické mapy), nebo nechtějí (zdržuje to a souřadnice jsou ochotní zjišťovat pouze u významných nálezů).*

Mame podporovat nejaký jejich format souboru?

*Ve verzi 2 zatím nevím o exportech, je to zatím ve vývoji. Z verze 1.x bych podporoval exportní dbf soubor. Ale je otázka, zda podporovat mrtvou verzi, od které se bude ustupovat a měla by ji nahradit verze 2.*

7)

Existuje protokol BioCASE a pro něj je vytvořeny tzv. provider software. Tento software umožňuje dotazování (primárně biologických) databází přes web.

Chtěli bychom ho použít místo psaní vlastního webového klienta. Psaní vlastního webového klienta by byla poměrně zbytečná práce, když už existuje BioCASE. Krom toho, když se použije jejich provider software (přizpůsobíme tomu trochu náš db. model), tak bude automaticky triviální záležitost zapojit se do tohoto evropského projektu.

Ten software je určený pro prohlížení a hledání v datech, není možné pomocí něj data přidávat nebo upravovat (s tím jsme ale nepočítali ani u našeho webového klienta).

*Jsem pro, nač objevovat znovu Ameriku :-).*

### Db model

Problém bez rozumného řešení je checklist.

Navrhovaný db model počítá s relačním přístupem, tzn. ukládají se kódy taxonů. Nejsou tam zahrnuta synonyma. Z db hlediska je to správný pohled. Na to navazuje otázka správy checklistu. Ideální by byl model Turbovegu – viz níže. Tento přístup je sice náročnější, ale umožňuje snadnou výměnu dat a dotazování se na data. Stejně to má i Survey. Výhodné by bylo používat pro všechny db (Survey, PlantLore, Turboveg) jeden checklist, ale to může být nerealizovatelné. V případě nálezů z jiných zemí bych použil tamní checklisty (a

ukládal to do zvláštní db, aby se to nemíchalo. Do centrální db by asi stejně šly pouze nálezy z ČR).

Současný stav je operativní – není-li taxon v checklistu, tak se nic neděje, ale tento přístup je mírně až silně bordeloidní – neodpovídá db standardům (to by zas tak nevadilo), ale u taxonů zapisovaných přímo do db hrozí překlepy a pak se na tyto údaje špatně ptá a další otázka souvisí s efektivitou – kdy by se měl taxon, který není v checklistu do něj zapsat? Vím dopředu, zda ho použiji jednou, nebo opakovaně a pak kolikrát?

Strukturu checklistu bych nechal na širší diskuzi, ale přikláním se k první variantě.

Další nápad – použít checklist z Turbovegu – vůbec bych se tomu nebránil a dokonce bych tuto myšlenku podporoval.

### Různé

Nebylo by špatné se podívat na národní fytoecologickou databázi (sw. Turboveg, Juice). Zatím jsem ji neviděl, ale zajímavé postřehy od lidí, kteří s tím pracují – checklist je spravován jedním člověkem (J. Danihelka). V případě, že uživatel chybí taxon, tak požadavek pošle Danihelkovi, ten ho přidá do checklistu a pověsí na web. Uživatel si ho stáhne a aktualizuje ho. Zadávání kyttek je přes zkratky – 1. 3 písmena z rodového a 1. 3 písmena z druhového názvu. Při zápisu kytky se omezuje obsah listboxu pouze na ty položky, které vyhovují zapisované zkratce. Jestliže jedna zkratka je u více taxonů, tak se z těchto vybere myší. Tento způsob urychluje práci.

Jak bude řešená historie? Verzováním řádků? (při editaci by se vytvořila kopie řádku s aktuálními daty). Nemělo by být CCREATEDWHEN, CCREATEDWHO, CUPDATEWHEN, CUPDATEWHO i u lokalit?

## **ANALYSIS OF THE SYSTEM**

Administrace databáze: Pravděpodobně by tam měl být účet určený pro administraci uživatelů a databází. Pod tímto účtem by se nemělo rutinně pracovat (něco jako root na Unixech). Pak bych měl jednu technickou připomínku – heslo účtu sysdba – masterkey je pouze na Woknech, u Linuxu se generuje vždy nové heslo při každé instalaci (i když jsem to instaloval na stejný komp ze stejného RPM). Z bezpečnostních důvodů se nedoporučuje pracovat pod účtem sysdba, takže asi by bylo nejlepší při instalaci serveru vytvořit účet, pod kterým by se PlantLore přihlašoval.

### ***Edit the Account***

ad2) U anonymního účtu (pravděpodobně by to byl účet **www**) by nebylo heslo a mohl by pouze prohlížet data. Jeho nastavení by bylo fixní, aby nikdo nemohl jeho povýšením způsobit bezpečnostní incident (podobně jako účet Guest ve Woknousech).

### ***Manage the DB***

Nutno řešit i zálohování db a jejich obnovu ze zálohy. Sice to lze dělat i jiným sw, ale asi by bylo dobré, aby to uměl PlantLore. Tohle je jediný korektní způsob, jak fyzicky odstranit smazané záznamy z db. Při obnově db by se měl i nastavit dialekt SQL pro db na hodnotu

3. Standardně je to 1. Umí to JayBird?

## ***Export***

### **Format specification**

Asi bych zrušil binární formáty (XLS, DBF) a podporoval nějaké rozumné textové – CSV, XML. Binární formáty dělají problémy a blbě se na ně shání JDBC ovladače, zvláště volně šiřitelné (O GPL ovladačích ani nemluvě. Pro DBF sice existuje TinySQL, ale dělal mi problémy při větším množství zpracovávaných dat. Pro XLS vím pouze o komerčních ovladačích). Je to sice v rozporu se zadáním, ale zbytečně bych nepřidělával problémy.

## ***Adding and Editing the Data***

### **Add**

Tady bych to nekomplikoval dvěma formuláři, ale spíš bych udělal jeden s ergonomickým rozmístěním prvků a pak bych někde měl nastavení, která pole by měly být přístupné a která ne. Je to flexibilnější.

### **Add Plant**

Tady je organizační záležitost správy checklistu. Nutno probrat. Jaký checklist použít. Stávající, nebo ho přebrat z Turbovegu/Surveye? Jestliže bychom měli vlastní checklist (bylo by to asi trochu podivné, aby každý db systém měl svůj vlastní checklist v této malé zemičce :-)), spravoval by ho někdo? Nebo by byla anarchie a každý by si tam přidal, co by chtěl?

## ***Accessing the System via an Internet Browser***

Tady je to psané obecně, ale předpokládám nasazení s jednou databází na serveru katedry botaniky.

## ***Questions and Unresolved Issues***

Složitější podmínky (AND/OR) bych neřešil, protože by se to muselo dělat přes SQL a to je neprůchodné. Alternativa by byl jednoduchý SQL editor pro podmínku, kde by byl seznam sloupců, operátory a pak list se seznamem hodnot, které se nachází v jednotlivých sloupcích – vyberu sloupec a pak např. kontextovým menu dám „vyplň hodnoty“ a ono mi je do toho listu vypíše. Musel by se samozřejmě ošetřit SQL Injection.

Sloupce	Operátor	
ID	BETWEEN	11
Inventarní	LIKE	12
Vlastník	IN	15
updater	IS NULL	20
ID_foto_Praha	IS NOT NULL	21
Den	>=	22
Mesic	<=	45
Rok	=	

Vyplň hodnoty

Tímto by se dala rozumně sestavit podmínka SQL dotazu i se závorkami, ale předpokládám, že to bude používat minimální počet lidí (možná jen já :-)) a tím pádem by to nestálo za tu námahu.

Další možnost by byla, podívat se, jak to dělá MS Access u filtrů u tabulek (na kartě je vztah mezi poli AND a mezi kartami OR).

### ***Answers and Resolved Issues “to-be-added”***

Ideální by bylo, kdyby pro více DB mohl být jeden seznam kytek, ale to rozumné SQL servery nepodporují, včetně Firebirdu.