Модуль MIGRATE и другие средства импорта содержания в Drupal

Григорий НАУМОВЕЦ

Киев

Украина

План

- В чём задача и в чём проблема
- Средства для импорта содержания в Drupal из популярных CMS
- Универсальные средства: Migrate, Node import и другие
  - Хранение содержания в Drupal: обзор таблиц
  - Node import и типичные проблемы переноса данных
  - Migrate и вспомогательные модули
  - Новые возможности
- Перспективы

- Целевая аудитория:
  содержание должно быть понятно и начинающим ☺

Создание сайта на основе существующего

- Сколько статей/нод нужно перенести? Если мало – может, быстрее всего будет скопировать их вручную?
- Импорт содержания: задача-минимум – перенести основное содержание (заголовки и тела статей/нод)
- Задача-максимум – полный перенос сайта, включая
  - Классификационные категории (таксономия)
  - Файлы, их URLы и связь со статьями/нодами
  - URLы статей/нод
  - Языки статей/нод, связь между разноязычными версиями
  - Пользователи, их пароли, профили, права доступа
  - Авторство статей/нод
  - Даты создания/модификации статей/нод
  - Показатели оценки статей/нод пользователями или менеджерами сайта
  - И наверняка что-нибудь ещё, что вы заранее не смогли предугадать

“Переезд” на Drupal с других CMS: готовые решения

- Home » Installation guide » Migrating to Drupal http://drupal.org/handbook/migrating

- Предлагаются либо специальные модули для переезда с конкретной CMS на Drupal, либо наборы скриптов и инструкций по переносу баз данных и файлов
- Проверяйте, для каких версий исходной CMS и Друпала разработаны модули или скрипты: если с тех пор структуре баз данных, старые скрипты могут не сработать

Примеры модулей для “переезда” на Drupal с других CMS

- Joomla to Drupal
  http://drupal.org/project/joomla
- Wordpress Import
  http://drupal.org/project/wordpress_import
  WP2Drupal
  http://drupal.org/project/wp2drupal
- PHP-Nuke to Drupal
  http://drupal.org/project/phpnuke2drupal
- phpBB2Drupal
  http://drupal.org/project/phpbb2drupal
- vBulletin to Drupal
  http://drupal.org/project/vbtodrupal
- И т.д. и т.п.

Сравнение различных модулей для импорта и экспорта

- http://groups.drupal.org/node/21338
  Comparison of Content and User Import and Export Modules

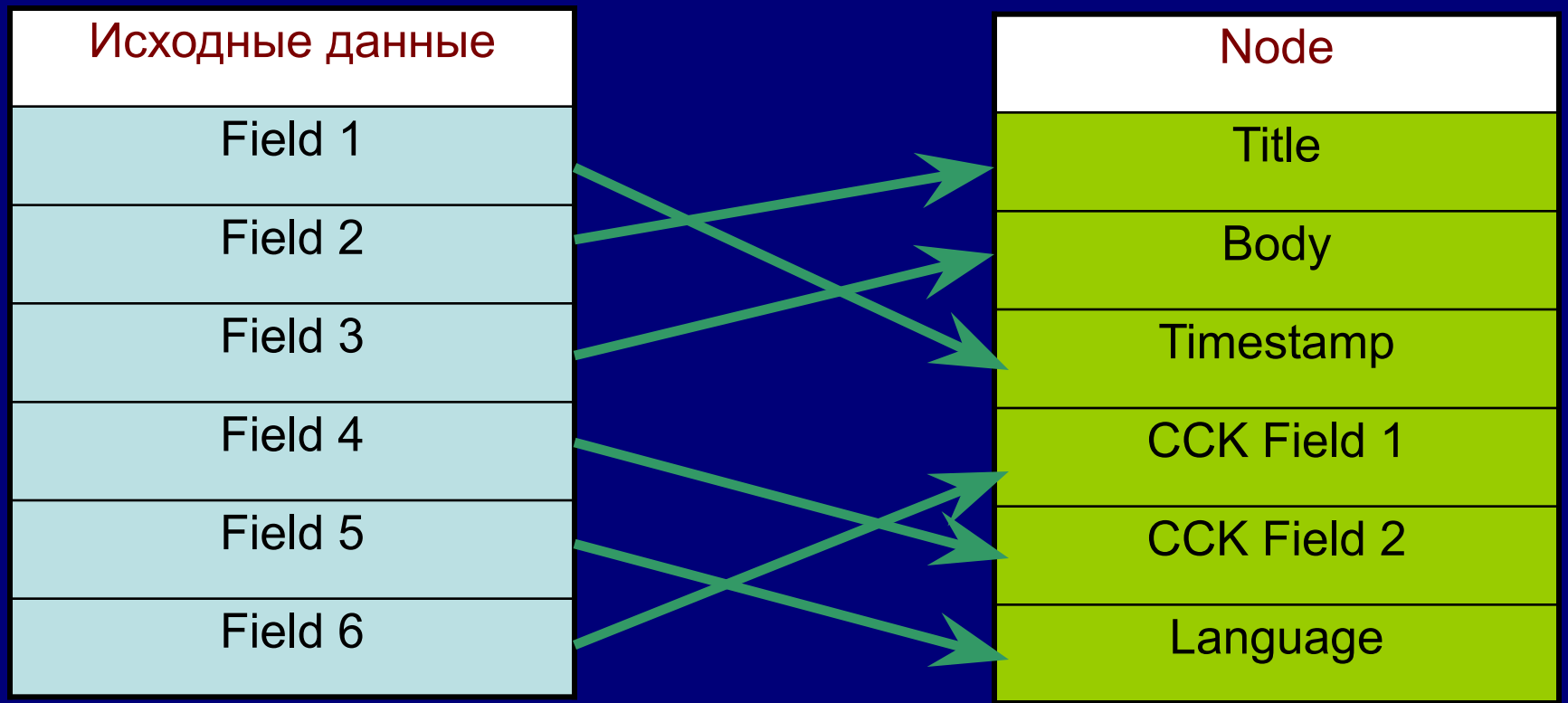
Модуль Node import

- http://drupal.org/project/node_import

- Импорт содержания из текстовых файлов в формате
  CSV (comma-separated values) или
  TSV (tab-separated values)
- CSV или TSV можно экспортировать из Microsoft Excel, или из phpmyadmin
- Текст в файле должен быть в кодировке UTF-8

Модуль Node import

- Можно импортировать данные в поля нод стандартных типов (page, story, etc.) и нестандартных (CCK)
- В ходе импорта анализируется структура исходной CSV-таблицы и задаётся соответствие: какую колонку таблицы импортировать в какое поле ноды



Node import: возможные проблемы

Перенос классификационных категорий в таксономию: возможна разная структура записи

articlenumber	categories
16	8,40,41,58
17	17,93
18	17,93

Table: term_node

nid	vid	tid
16	16	8
16	16	40
16	16	41
16	16	58
17	17	17
17	17	93
18	18	17
18	18	93

В исходном материале категории для каждой статьи перечислены через запятую, в Друпале о каждой таксономической категории (tid) делается отдельная запись с указанием идентификаторов ноды (nid, vid)

Для простоты показан случай, когда номера нод и таксономических терминов равны номерам статей и классифкационных категорий в исходном материале – но реально это может быть не так

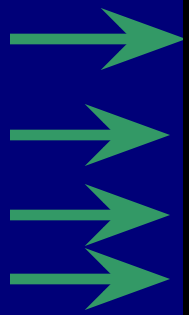
Возможное решение таких проблем

- Создаётся временное («техническое») поле, куда импортируются данные, которые нужно будет преобразовать (например, список категорий через запятую)
- Потом пишется sql-скрипт (или php-код), который выполняет нужное преобразование: в данном случае, просматривает содержимое временного поля и для каждой найденной категории создаёт соответствующую запись в таблице term_node
- Заодно, при необходимости, коды категорий исходного материала меняются на идентификаторы таксономических категорий Друпала
- Когда импорт содержания закончен и всё проверено, временные поля можно удалить

Node import: ещё один пример возможных проблем

Перенос многоязычного сайта: возможна разная схема записи данных о языке и стыковки разноязычных статей/нод

articlenumber	language
3756	1
3756	2
3757	1
3757	2



nid	language	tnid
1	en	1
2	ru	1
3	en	3
4	ru	3

Пример: в исходном материале язык описывается цифровым кодом (1,2,…), разноязычные версии стыкуются по номеру статьи (articlenumber)

В Друпале язык описывается буквенным кодом (en,ru,…), разноязычные версии стыкуются по tnid (номеру ноды-«оригинала»), при этом номера нод не равны номерам статей (nid≠articlenumber)

Возможное решение

- Исходный номер статьи (articlenumber) импортируется во временное поле
- Пишется sql-скрипт (или php-код), который должен:
  - Преобразовать коды языков для поля ‘language’ (‘1’->’en’, ‘2’->’ru’ и т.д.)
  - Сформировать значения поля ‘tnid’ таким образом, чтобы, допустим, для нод с language=’en’ tnid=nid а для нод с language=’ru’ tnid = номеру англоязычной ноды с таким же значением articlenumber.
- В конце временные поля удалить

А как вообще в Друпале хранится содержание?

- В каких таблицах базы данных хранится основное содержание ноды и прочие данные, которые с ней связаны?

Table: node

id ноды

id версии ноды

тип (*page*, *story*, etc.)

язык

заголовок

id пользователя

- Вопреки ожиданиям, в таблице “node” НЕТ ОСНОВНОГО СОДЕРЖАНИЯ НОДЫ (body)
- Оно хранится в таблице “node_revisions”

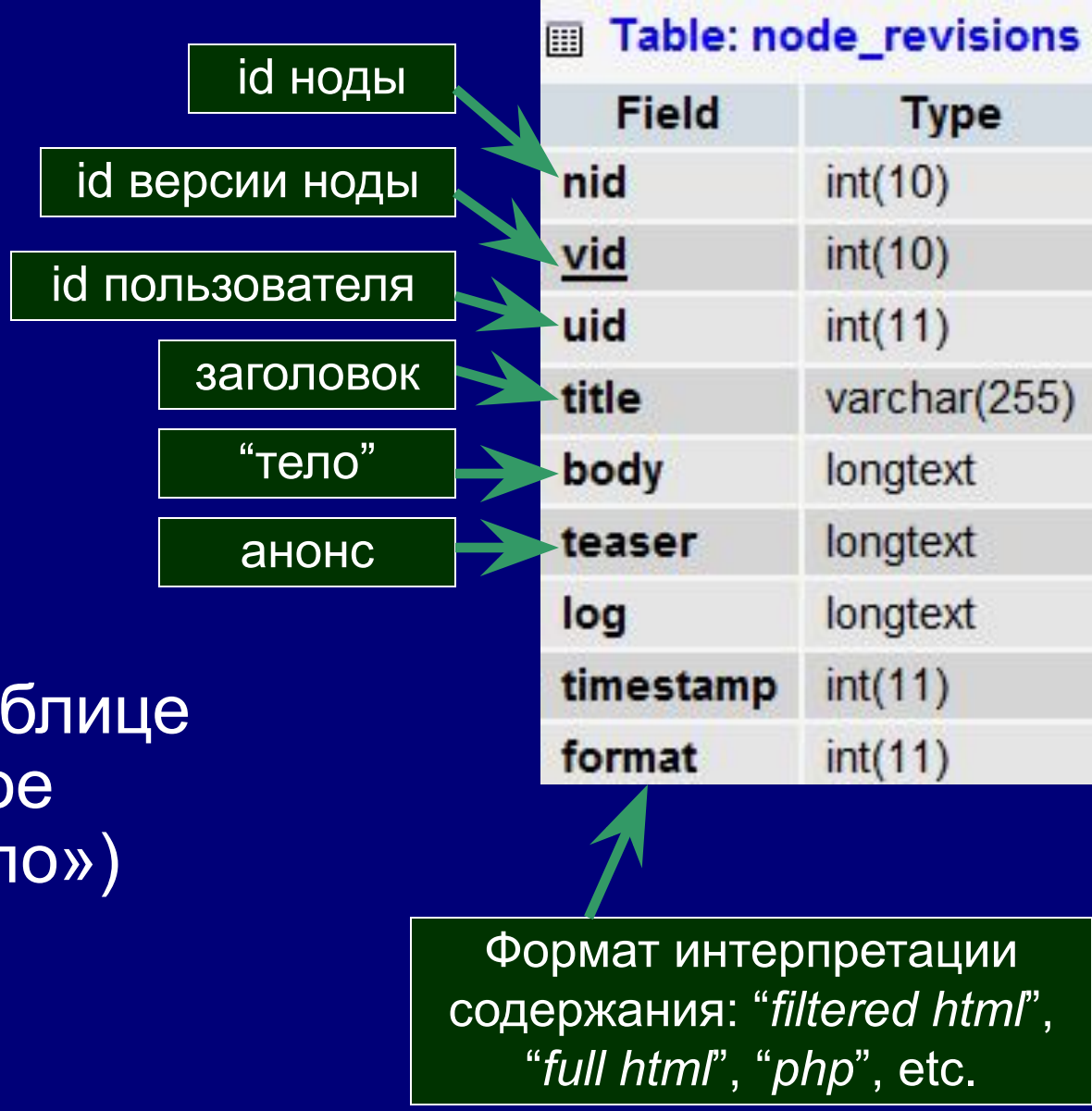
Связь разноязычных версий: id ноды-«оригинала»

статус перевода

Table: node

Field	Type
nid	int(10)
vid	int(10)
type	varchar(32)
language	varchar(12)
title	varchar(255)
uid	int(11)
status	int(11)
created	int(11)
changed	int(11)
comment	int(11)
promote	int(11)
moderate	int(11)
sticky	int(11)
tnid	int(10)
translate	int(11)

Table: node_revisions



- Именно в этой таблице хранится основное содержание («тело») ноды

Дополнительные таблицы для нестандартных типов CCK и используемых в них полей

Таблица, где хранятся значения дополнительных полей CCK для нод определённого типа (“*intlproj*”)

привязка к ноде по id версии и ноды

значения дополнительных полей

Table: content_type_intlproj

Field	Type
vid	int(10)
nid	int(10)
field_implementorg_value	longtext
field_fundingorg_value	longtext
field_startdate_value	int(11)
field_enddate_value	int(11)
field_urlimplement_url	varchar(255)
field_urlimplement_title	varchar(255)
field_urlimplement_attributes	mediumtext

Таблица, где хранятся значения дополнительного поля CCK (“*areafocus*”), используемого в нодах нескольких типов

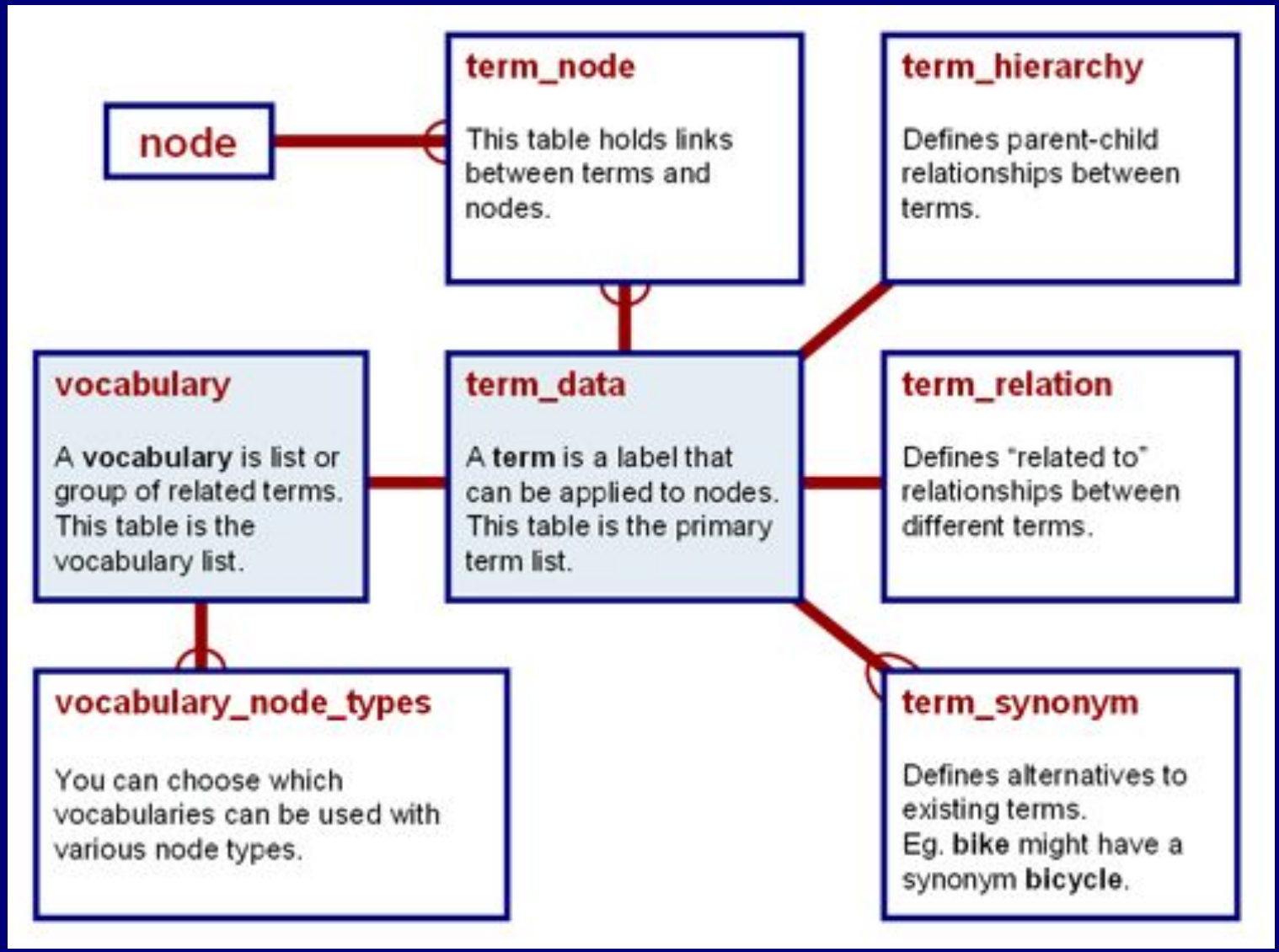
привязка к ноде по id версии и ноды

значение дополнительного поля

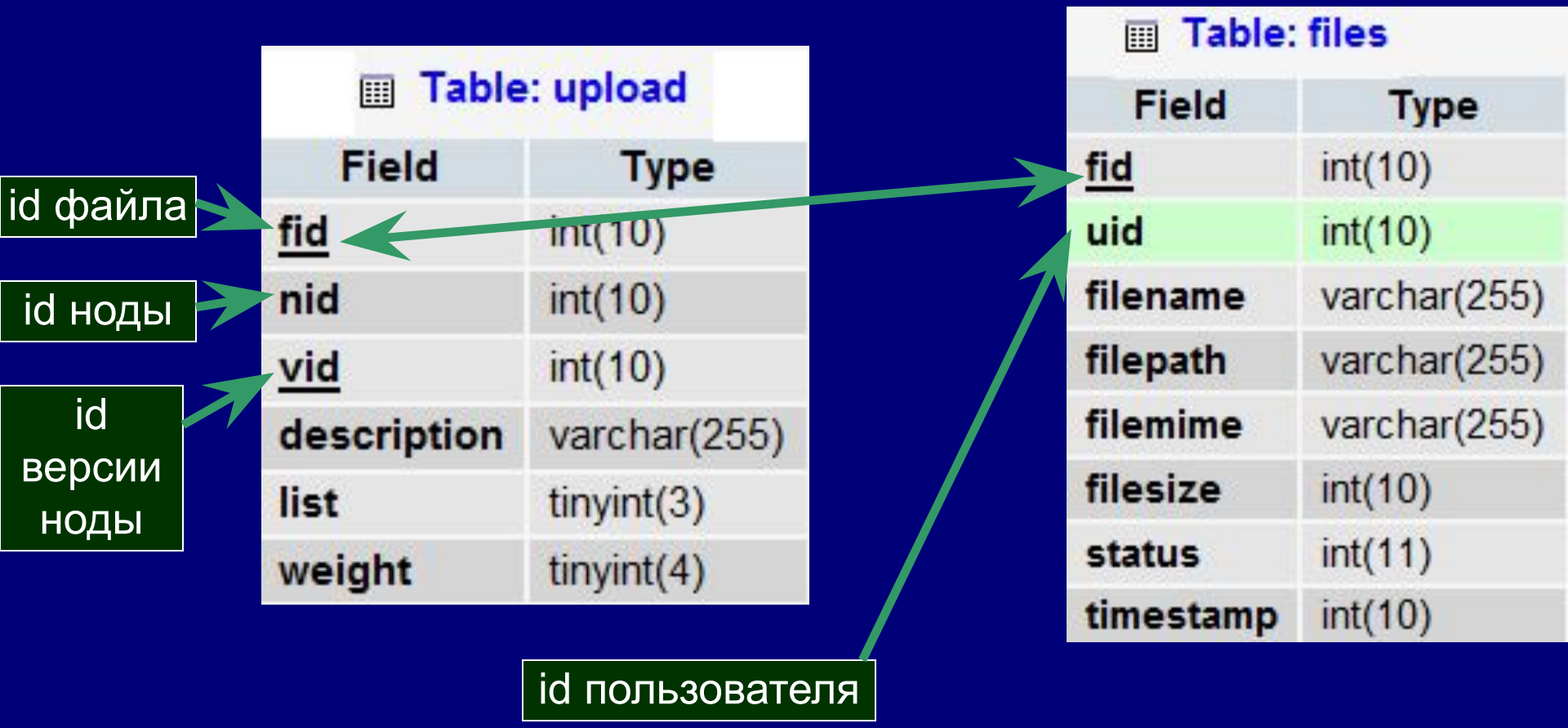
Table: content_field_areafocus

Field	Type
vid	int(10)
nid	int(10)
field_areafocus_value	longtext

Таблицы, в которых описана таксономия



Данные о файлах и их связи с нодами: таблицы upload и files



Данные об адресах (алиасах) нод: таблица url_alias

id адреса

Source – системный адрес ‘node/1’

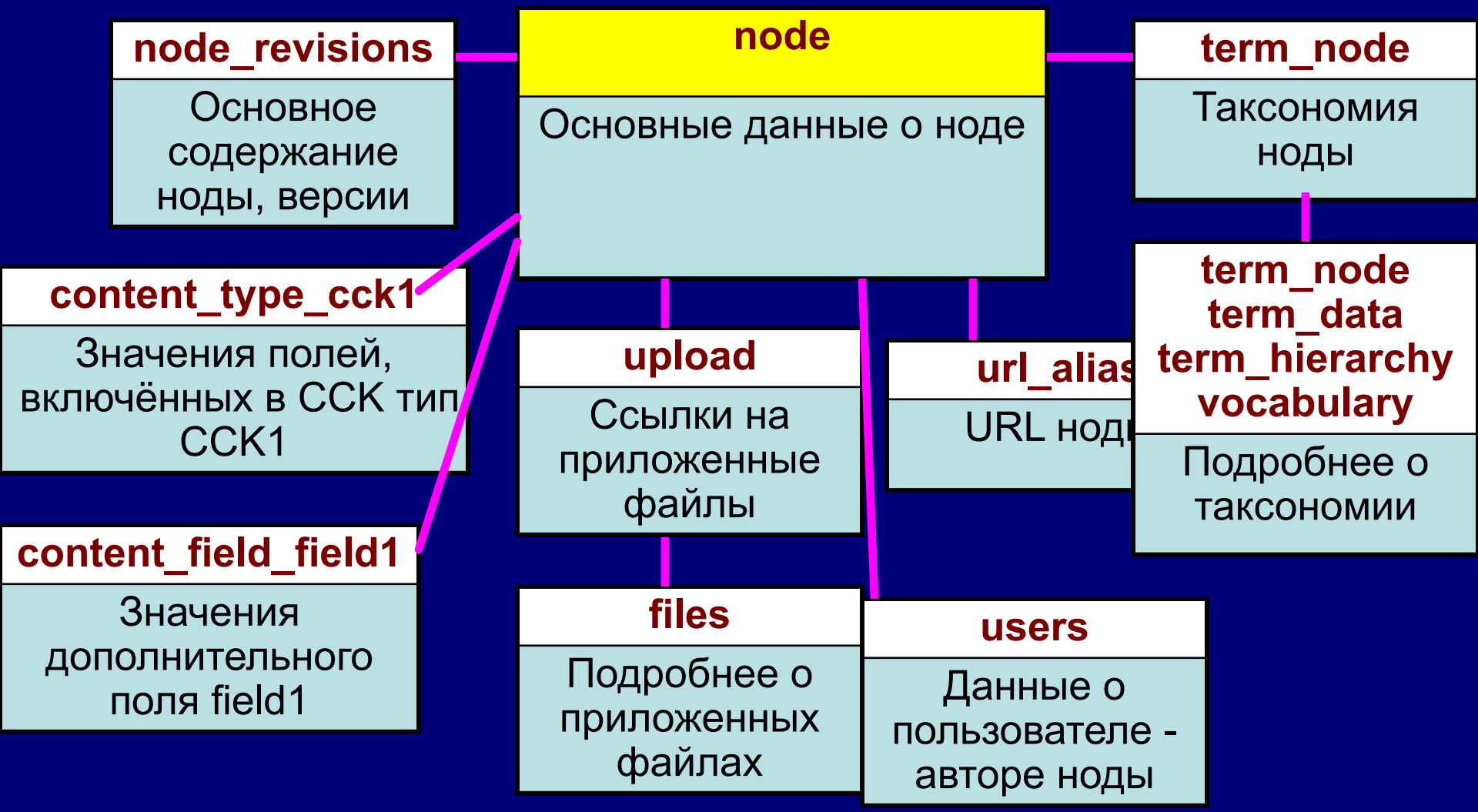
Destination – адрес-псевдоним ‘welcome’

язык, к которому привязан адрес

Table: url_alias

Field	Type
pid	int(10)
src	varchar(128)
dst	varchar(128)
language	varchar(12)

Связь таблиц, где хранится информация о ноде



Проблема импорта из множества связанных таблиц

- В Друпале содержание разбросано по множеству связанных таблиц
- Что, если в исходной CMS содержание тоже разбросано по разным таблицам?
- Прежде чем создавать CSV-файл и подсовывать его под node_import, придётся соединить нужные данные из разных таблиц в одной
- Т.е. нужна предварительная обработка исходного материала

Импорт содержания

CMS – источник данных

Выделить содержание определённого типа

↓

Манипуляция данными (например, объединение таблиц, перекодировка, замена значений)

↓

Экспорт таблицы данных

↓

Манипуляция данными

Drupal

Создать/определить тип CCK и соответствующие поля для импорта содержания

↓

Проанализировать импортируемую таблицу, задать соответствие полей

↓

Импорт таблицы данных

↓

Манипуляция данными

↓

Доработка напильником и др. инструментами

↓

PROFIT! ☺

Модули для преобразования данных

- Import / Export API
  http://drupal.org/project/importexportapi
- Transformations
  http://drupal.org/project/transformations
  - This module transforms data.
    It doesn't care which data, and it doesn't care how.
    This module is complex, and strongly object-oriented.
    If you're afraid of classes and objects in PHP, run away now.
  - **This module can do lots in principle, but little out of the box.**

Модуль Migrate: зависимость от других модулей

- Table Wizard
- Schema
- Views
- Migrate Extras.
  (для импорта полей CCK и взаимодействия с некоторыми другими модулями)
- Advanced help - рекомендуется

Модули Table Wizard (TW) и Schema

- Модуль **Schema**: API для операций с таблицами в базе данных
- **Table Wizard** позволяет:
  - Просматривать, фильтровать и т.п. таблицы в базе данных средствами Views
  - Анализировать данные – показывать диапазон значений и т.п.
  - Устанавливать связи между таблицами и совмещать их данные
  - Импортировать данные из CSV-таблиц
- TW предоставляет API для работы других модулей с таблицами базы данных через Views

Последовательность действий при импорте через Migrate

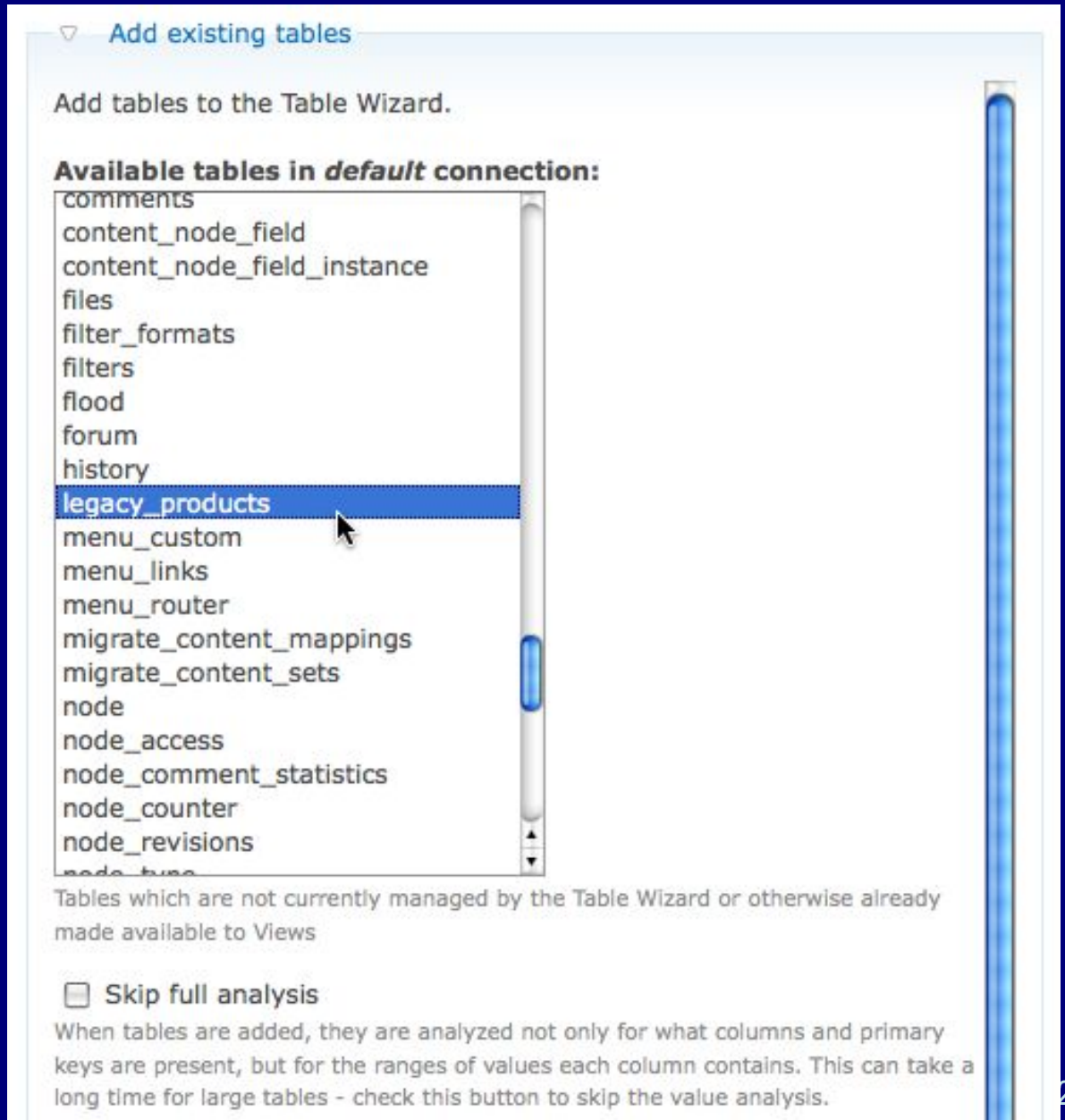
- http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard
- http://civicactions.com/blog/2009/jul/25/migrating_your_website_drupal

Последовательность действий при импорте через Migrate

- Втянуть таблицу с данными в MySQL
- У таблицы д.б. единственный основной ключ (*primary key);* при необходимости - отредактировать
- Использовать Table Wizard (admin/content/tw) и пометить нужные таблицы
- Если данные хранятся в нескольких таблицах – построить связи (admin/content/tw/relationships)
- Проанализировать данные и построить таблицы соответствий (*content sets*) между исходными полями и полями, куда нужно импортировать данные (admin/content/migrate/destinations)
- Выбрать таблицу соответствий и запустить процесс миграции (admin/content/migrate/process)

Шаги импорта данных через Migrate

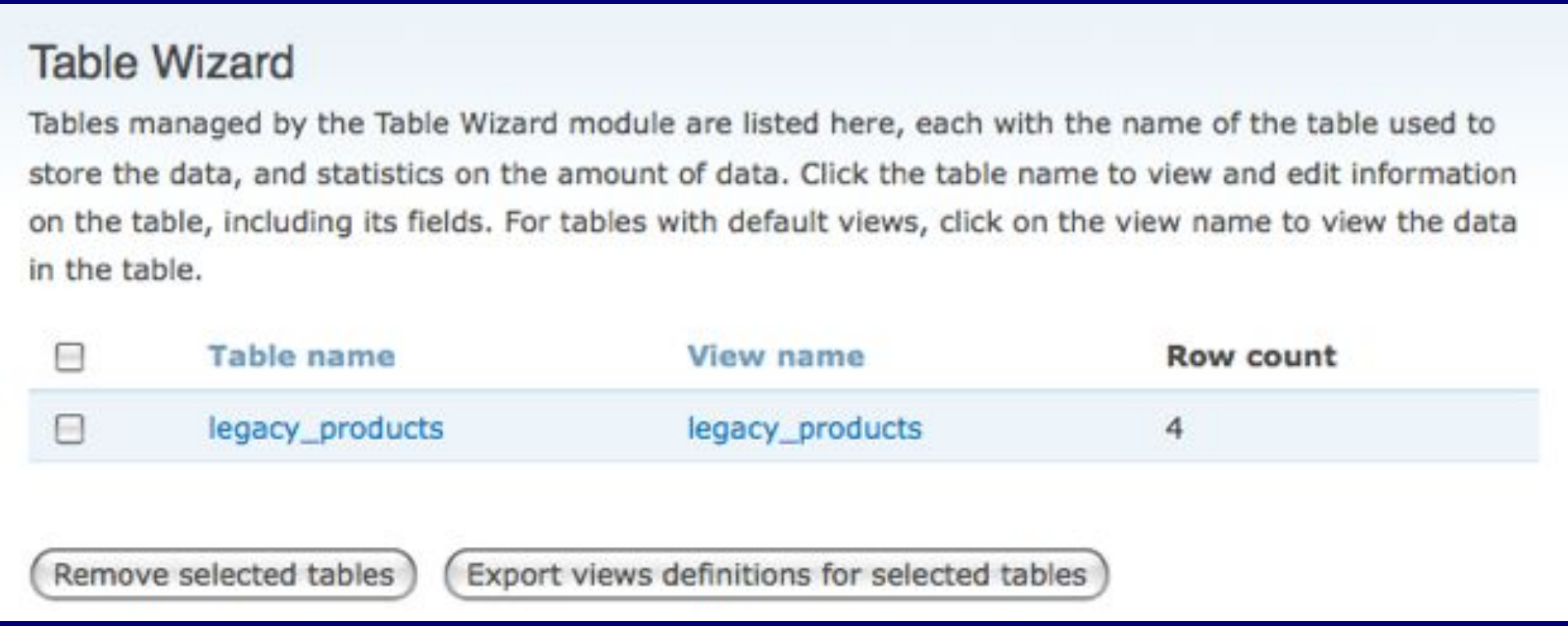
- Выбор таблицы в Table Wizard: таблица ‘legacy_products’



Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

- Таблица ‘legacy_products’ выбрана
  показана информация о её содержании:
  количество записей - 4



Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

- Анализ таблицы ‘legacy_products’
  поля можно комментировать, помечать как игнорируемые

Field name	Ignore	Empty	PK	Available key	Type	Text length	Range	Comments
sku			Yes	☑	varchar	8-8	BAN001X2 TRO593W2	A unique product ID.
name	☐			☐	varchar	6-12	Banana bread Trophy	Name of the product. Maps to node title.
description	☐			☐	text	20-46	A trophy to make you feel good about you The kitties just love it!	Description of the product. Maps to node body.
price	☐			☐	float		9.99-45.78	The price. Duh. :P
internal_flag	☑			☐	int		1-9	For internal use only. Don't need to import.

Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

- Просмотр содержания таблицы ‘legacy_products’ (без игнорируемых полей)

Contents of legacy_products

This is a view of a raw database table. It may be sorted in various ways by clicking the column headers.

If you identify a particular field that does not need to be used in views of this table, go to the analysis page and check the *Ignore* box for that field. It will then no longer appear here.

sku	name	description	price
BAN001X2	Banana bread	Delicious banana bread. Mmmm. Bananas...	9.99
PLU124D0	Plunger	For plunging things.	10.49
KIT823Q1	Kitty litter	The kitties just love it!	25
TRO593W2	Trophy	A trophy to make you feel good about yourself.	45.78

Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

- Разметка импорта полей из таблицы в ноду

Source field	Default value	Destination field
<none>		Node: Authored by (username)
<none>	1	Node: Authored by (uid)
<none>		Node: Authored on
<none>		Node: Last updated on
<none>		Node: Published
<none>		Node: In moderation queue
<none>		Node: Promoted to front page
<none>		Node: Sticky at top of lists
<none>		Node: Create new revision
name (1)		Node: Title
description (2)		Node: Body
description (2)		Node: Teaser
price (1)		CCK: Price value
sku (1)		CCK: SKU Number value

Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

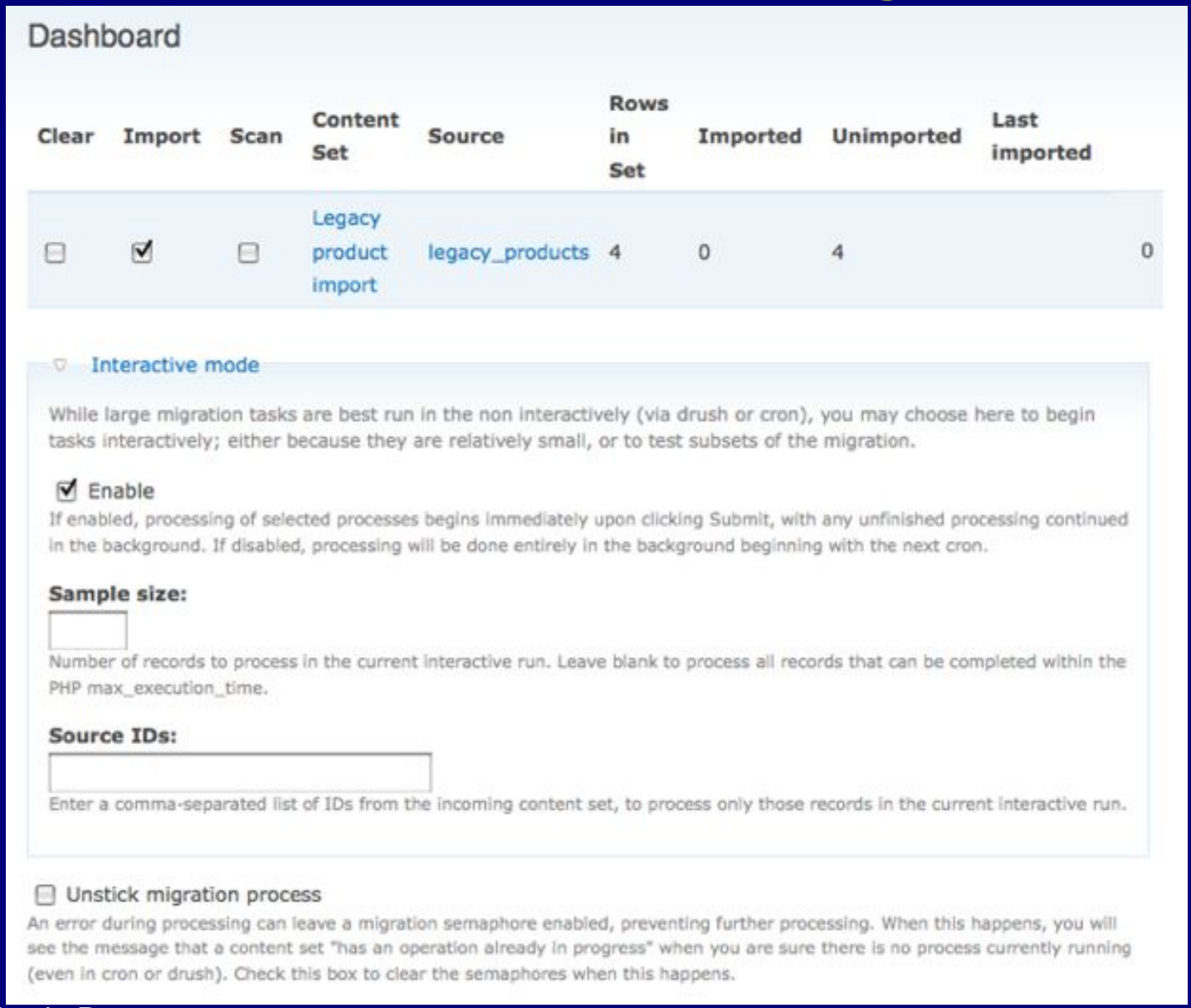
Шаги импорта данных через Migrate

- При этом задаются такие назначения для импорта:

Source field	Default value	Destination field
<none>	1	Node: Authored by (uid)
name		Node: Title
description		Node: Body
description		Node: Teaser
price		CCK: Price value
sku		CCK: SKU Number value

Шаги импорта данных через Migrate

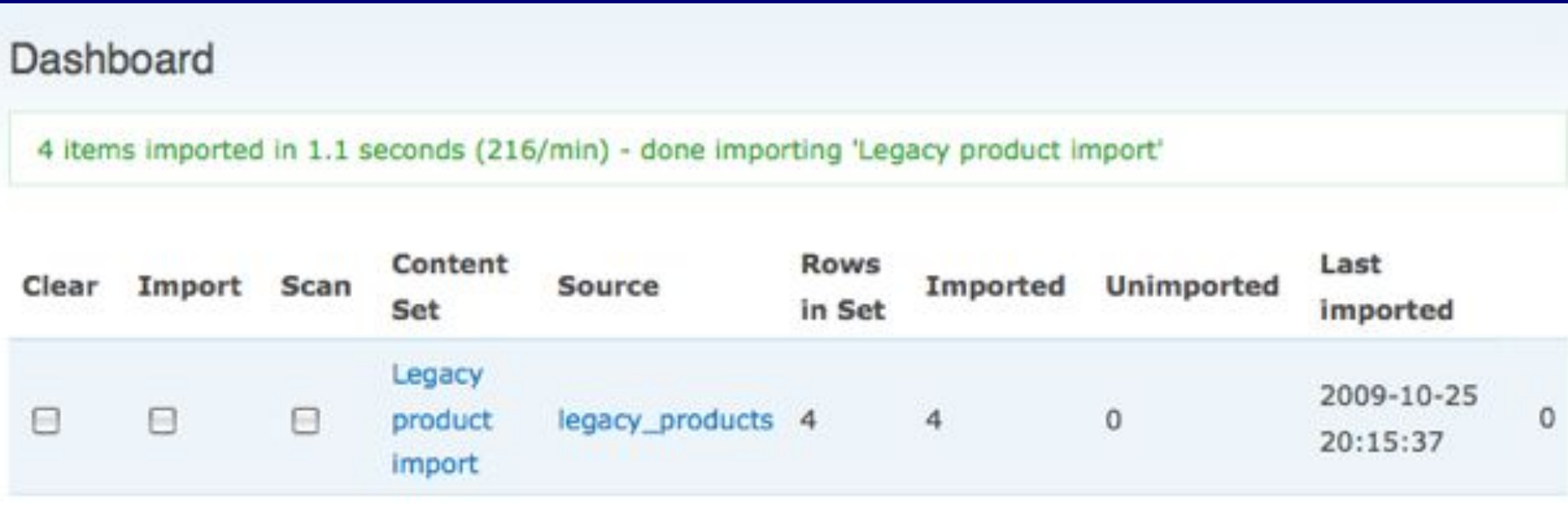
- Задание команды на импорт



Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

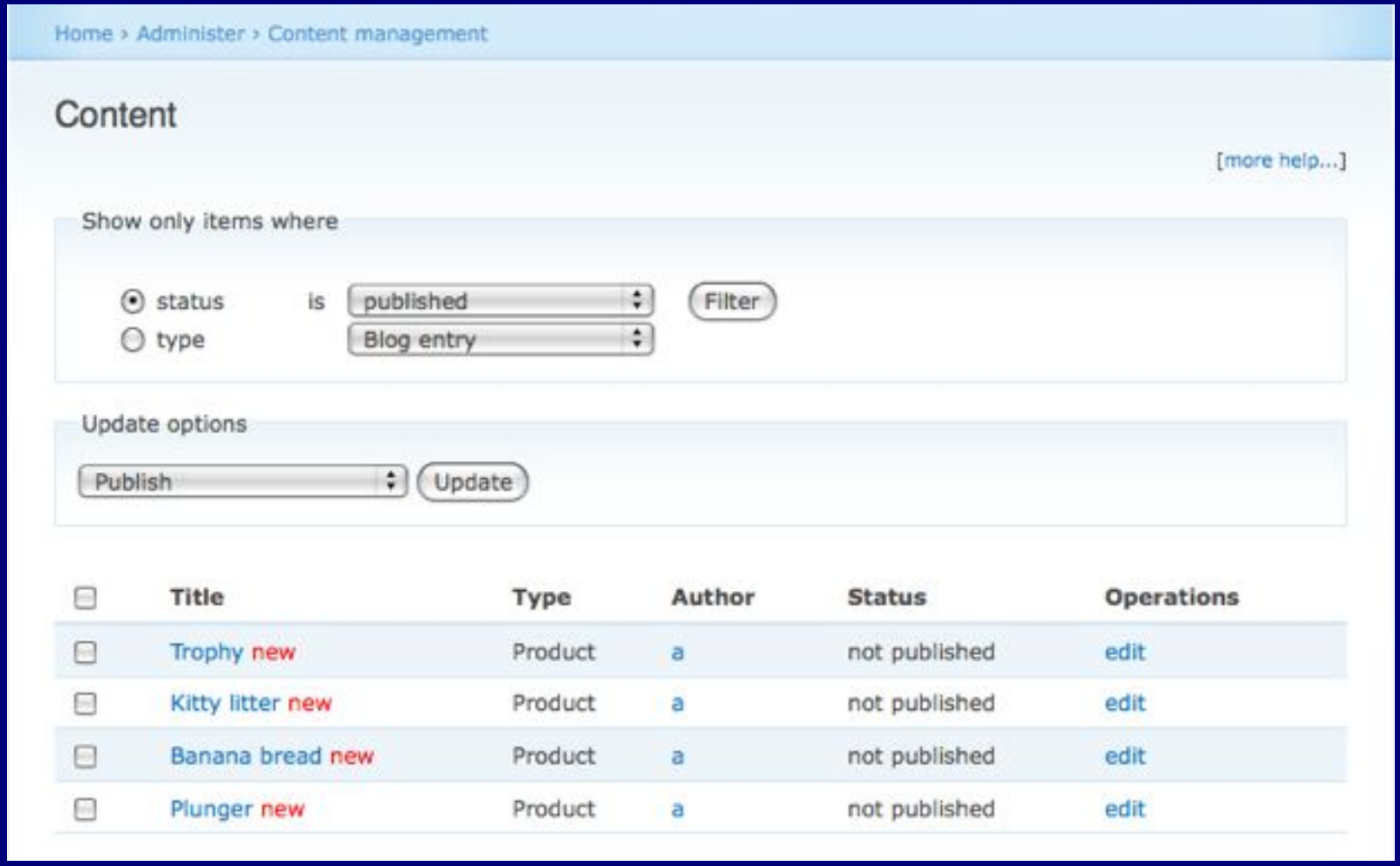
- Сообщение о результатах импорта: импортировано 4 записи



Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Шаги импорта данных через Migrate

- В списке нод появилось 4 новых ноды
  они пока не опубликованы – чтобы можно было проверить, всё ли с ними в порядке



Pictures from manual by Angie Byron
http://www.lullabot.com/articles/drupal-data-imports-migrate-and-table-wizard

Новое и полезное в Migrate (+TW):

- Работа с таблицами в базе данных
- Возможность объединения данных из нескольких таблиц
- Более удобный и наглядный интерфейс для анализа, фильтрации, просмотра, установки соответствий
- Возможность автоматического разбора значений, перечисленных через запятую
- Hooks для разработчиков дополнительных модулей:
  см. Migrate: Hooks
  Migrate API - new in migrate-1.0-beta4 and beyond.
  http://drupal.org/node/415190

Развитие Migrate

- Для Drupal 6 сделан Migrate 1.0
- Migrate 2.0 разрабатывается для D7 – после чего планируется backport на D6.
- Migrate 2.0 не должно зависеть от модулей Views и Table Wizard

Спасибо за внимание!

- Контактная информация:
  - http://camp10.drupal.ua/users/GN
  - gnaumovets@gmail.com