



Content Management

Erick Kurniawan, S.Kom, M.Kom



Content Management

Yang akan dibahas:

- Storage problem
- Workflow problem
- Version control
- Look and feel design untuk individual pages
- Navigation design
- Information architecture



Content Management

- Salah satu metode CM untuk OLC adalah mengurangi jumlah dari type content
- Sebagai contoh jika kita ingin adanya fasilitas article (magnet content), komentar dari user pada article, new dari publisher, pertanyaan dari user, dan jawaban dari pertanyaan
- Biasanya kita membuat 6 table untuk menampungnya (`articles`, `comments_on_articles`, `news`, `comments_on_news`, `questions`, `answers`)
- Dari sudut pandang database tidak ada yang salah dengan hal diatas



Storage Problem

- Yang jadi masalah adalah cara untuk programming-nya
- Lebih banyak script yang dibutuhkan untuk:
- User-experience: melihat directory content dari tableA, melihat satu kategori tertentu, melihat item terbaru, menambahkan item baru, dll
- Admin-experience: approve, disapprove item, delete item, dll
- Banyak membuat statement SQL yang sama untuk tiap table-nya
- Banyak index untuk tiap table-nya



Storing Article

- Basic table untuk article, yang biasanya kita buat:

```
create table articles (  
  article_id integer primary key,  
  creation_user not null references users,  
  creation_date not null date,  
  language char(2) references language_codes,  
  mime_type varchar(100) not null,  
  one_line_summary varchar(200) not null,  
  body clob );
```

Storing Article

- Apakah semua article dapat diakses oleh user

```
create table articles (  
  article_id integer primary key,  
  creation_user not null references users,  
  creation_date not null date,  
  language char(2) references language_codes,  
  mime_type varchar(100) not null,  
  one_line_summary varchar(200) not null,  
  body clob  
  editorial_status varchar(30) check  
  (editorial_status in  
  ('submitted', 'rejected', 'approved', 'expired  
  ')) );
```

Storing Article

- Apakah anda akan ingat menuliskan `editorial_status = 'approved'` pada setiap script dalam site?

```
create table articles_raw (  
  article_id integer primary key,  
  ...  
  editorial_status varchar(30) check  
  (editorial_status in  
  ('submitted', 'rejected', 'approved', 'expired'  
  )) );
```

```
create view articles_approved as select *  
from articles_raw where editorial_status =  
'approved';
```



Storing Article

- Alternatif lain jika hanya menggunakan satu table saja dapat dibuat view sebagai berikut:

```
create view articles_approved as select *  
from articles_raw where editorial_status =  
'approved' ;
```

Storing Comments

- Comment article dapat direpresentasikan dalam table yang terpisah

```
create table comments_on_articles_raw (  
comment_id integer primary key,  
-- on what article is this a comment?  
refers_to not null references articles,  
creation_user not null references users,  
creation_date not null date,  
language char(2) references language_codes,  
mime_type varchar(100) not null,  
one_line_summary varchar(200) not null,  
body clob,  
editorial_status varchar(30) check  
(editorial_status in  
( 'submitted', 'rejected', 'approved', 'expired' ))  
);
```



Storing Comments

- Untuk memudahkan membaca comment dapat dibuat view

```
create view
comments_on_articles_approved as
select * from
comments_on_articles_raw where
editorial_status = 'approved';
```

Ide-nya

- Yang membedakan dengan article hanya satu column `refers_to`, bagaimana jika digabung saja?

```
create table content_raw (  
content_id integer primary key,  
-- if not NULL, this row represents a comment  
refers_to references content_raw,  
creation_user not null references users,  
creation_date not null date,  
language char(2) references language_codes,  
mime_type varchar(100) not null,  
one_line_summary varchar(200) not null,  
body clob,  
editorial_status varchar(30) check (editorial_status  
in ('submitted', 'rejected', 'approved', 'expired')) );
```

Ide-nya

- Untuk memudahkan dalam pembacaan data dapat dibuat view berikut:

```
create view articles_approved as select * from
content_raw where refers_to is null and
editorial_status = 'approved';
```

```
create view comments_on_articles_approved as
select * from content_raw where refers_to is
not null and editorial_status = 'approved';
```

```
-- let's build a single full-text index on both articles and comments
-- using Oracle Intermedia Text (formerly known as "Context")
```

```
create index content_ctx on content_raw (body)
indextype is ctxsys.context;
```



Storing News

- Apa beda news sehingga harus dipisahkan dengan table yang berbeda?
- News mempunyai expiration date
- Jika sudah basi disimpan dalam arsip
- Punya field `release_time` untuk menunjukkan waktu release-nya
- Design table-nya menjadi:

Ide-nya

```
create table content_raw (  
content_id integer primary key,  
refers_to references content_raw,  
creation_user not null references users,  
creation_date not null date,  
release_time date, -- NULL means "immediate"  
expiration_time date, -- NULL means "never  
expires" language char(2) references  
language_codes,  
mime_type varchar(100) not null,  
one_line_summary varchar(200) not null,  
body clob,  
editorial_status varchar(30) check  
(editorial_status in  
( 'submitted', 'rejected', 'approved', 'expired' )) );
```

Ide-nya

- Bagaimana mencari news stories diantara content rows? Ditambah satu column lagi

```
create table content_raw (  
content_id integer primary key,  
content_type varchar(100) not null,  
refers_to references content_raw,  
creation_user not null references users,  
creation_date not null date,  
release_time date, -- NULL means "immediate"  
expiration_time date, -- NULL means "never  
expires" language char(2) references  
language_codes,  
mime_type varchar(100) not null,  
.....  
..... ) ;
```

Ide-nya

- Untuk memudahkan untuk menampilkan news dapat dibuat view berikut

```
create view news_current_and_approved as
select * from content_raw where
content_type = 'news' and (release_time is
null or sysdate >= release_time) and
(expiration_time is null or sysdate <=
expiration_time) and editorial_status =
'approved' ;
```



Diskusi

- Mengapa data article misalnya tidak disimpan dalam bentuk HTML files dalam direktori?
- Programmer menggunakan text editor, designer menggunakan web design tool dan FTP untuk upload, dll
- Bila salah script yang diupload bisa merusak sistem
- Sulit untuk pengaturan authentication
- Maintenance static html sulit



Workflow Problem

- Mudah untuk membuat dan me-maintain website jika:
 - Orang yang sama adalah publisher, author, dan programmer
 - Site terdiri atas sedikit pages
 - Tidak ada yang peduli terhadap konsistensi halaman
 - Tidak ada yang peduli terhadap versioning control
- Tetapi tidak demikian dengan kondisi yg ada
 - Pekerja dipecah menjadi publisher, information designer, graphic designer author, dan programmer
 - Site terdiri dari ribuan pages
 - halaman harus konsisten (dengan themes)
 - Versioning control is critical



Workflow Problem

- Publisher menentukan topik dari content utama yang ada
- Information Designer menentukan navigational link dari pages, bagaimana menyajikan content, dan graphic design apa yang dibutuhkan
- Graphic Designer mempunyai kontribusi untuk membuat drawing, logos, artwork services, mock-up template (static html) yang dibutuhkan information designer
- Programmer membangun production template dan program logic
- Editor melakukan approve content



Version Control (content)

- Orang bag administrasi berhak mengambil dan mengedit old-version dari content item
- Ada masalah jika beberapa orang mengubah data pada content secara bersamaan

```
create table content_raw
(
  -- the combination of these two is the key
  content_id integer,
  version_number integer,
  ...
  primary key (content_id, version_number)
);
```

Version Control (content)

- Menggunakan 2NF

```
create table content_raw (  
content_id integer primary key,  
content_type varchar(100) not null,  
refers_to references content_raw,  
creation_user not null references users,  
creation_date not null date,  
release_time date,  
expiration_time date,  
mime_type varchar(100) not null,  
zip_code varchar(5)  
);
```

Version Control (content)

- Menggunakan 2NF

```
create table content_versions (  
  version_id integer primary key,  
  content_id not null references content_raw,  
  version_date date not null,  
  language char(2) references language_codes,  
  one_line_summary varchar(200) not null,  
  body blob,  
  editorial_status varchar(30) check  
  (editorial_status in  
  ('submitted', 'rejected', 'approved', 'expired')),  
  -- audit the person who made the last change to  
  editorial_status editor_id references users,  
  editorial_status_date date );
```



Version Control (content)

- Untuk menampilkan semua versi dari content tertentu dapat dibuat query

```
select content_id, max(zip_code) from  
content_raw where content_id = 5657 group by  
content_id
```



Version Control (content)

- Untuk menampilkan versi terbaru
- Apakah sudah menjamin hanya menghasilkan 1 row saja?

```
select * from content_versions where  
content_id = 5657 and editorial_status =  
'approved' and version_date = (select  
max(version_date) from content_versions where  
content_id = 5657 and editorial_status =  
'approved' )
```

Version Control (content)

- content_id tidak memiliki unique constraint
- Karena bisa lebih dari satu versi
- Jika hanya ingin menampilkan versi yang paling baru saja

```
select * from content_versions where  
content_id = 5657 and editorial_status =  
'approved' and version_id =  
(select max(version_id) from content_versions  
where content_id = 5657 and editorial_status =  
'approved' )
```

Version Control (content)

- Ada database tertentu yang mempunyai trik query yang lebih mudah
- Pada oracle untuk mengambil versi yang paling baru dapat digunakan query

```
select * from content_versions where  
content_id = 5657 and editorial_status =  
'approved' and rownum = 1 order by  
version_date desc
```

```
select * from (select * from content_versions  
where content_id = 5657 and editorial_status =  
'approved' order by version_date desc) where  
rownum = 1;
```



Version Control (program)

- Setiap program pasti ada perubahan / pembaharuan
- Tidak mungkin tetap / sudah sempurna
- Pada Internet program harus selalu dalam keadaan online
- Bagaimana solusinya untuk pengembangan versi development Aplikasi Berbasis Internet?
 - Tiga HTTP Server (dapat berada pada satu physical computer)
 - Dua atau tiga RDBMS user / tablespace (dapat dibuat dalam satu instance RDBMS)
 - One version control repository (menggunakan CVS Concurrent Versions System)



Version Control (program)

- Tiga HTTP Server (misal site: foobar.com)
- Dibagi menjadi:
- Production server : web server yang diakses oleh pengunjung di Internet (/web/foobar/)
- Development server : untuk pengembangan web-nya, dimana programmer dapat 'mencoba-coba' tanpa beresiko (setelah dites dan ok, baru dipindah ke production server) url: /web/foobar-dev/
- Untuk website yang lebih kompleks dapat ditambahkan Staging server (untuk tester dan development secara simultan) url: /web/foobar-staging/



Version Control (program)

- Cara kerja ketiga server tersebut adalah:
- Developer bekerja secara kontinu pada /web/foobar-dev/
- Jika publisher puas dengan development site, maka dibuat versi baru dan dipasang pada /web/foobar-staging/
- Tester melakukan tes dan perbaikan-perbaikan pada /web/foobar-staging/ server
- Setelah dirasa tidak ada bugs dan performanya bagus maka dipindah ke /web/foobar/ (production)
- Bugs pada staging yang belum mendapat perbaikan dikembalikan kembali ke development



Look and Feel

- Space
- Time
- Words
- Color



Space

- Space pada monitor / tampilan
- Apakah penting untuk menampilkan informasi tanpa scrolling? (konsep newspaper)
- Apakah anda memanfaatkan dengan baik screen space yang anda miliki?
- Biasanya yang memakan banyak tempat adalah penggunaan banyak icon
- Biasanya user tidak mengerti arti dari icon jadi kita perlu untuk memberi keterangan
- Umumnya kebijakan yang diambil “*let the information be the interface*”
- Misal dalam menampilkan kategori tidak perlu icon



Time

- Mana yg anda pilih fast or slow service time?
- *Fast is better than slow*, response time harus selalu sama dari session ke session
- Menurut penelitian IBM (1970)
 - 0.1 second untuk direct manipulation
 - 1 second for screen-click-screen system
 - Kurang dari 10 second untuk membuat user tetap memperhatikan
- Masalah bandwidth antar user dan server (diluar jangkauan kita)
- Page dengan banyak table buruk dan membuat lambat (membutuhkan waktu lama untuk render)



Time

- Penggunaan property width dan height pada tag image disarankan
- Sehingga image dapat dirender terlebih dahulu dengan blank space
- Kita dapat membandingkan service time website yang dibuat dengan web yang berhasil seperti e-bay, yahoo, atau amazon
- Ingat bahwa pada buku dan majalah kita dapat berpindah halaman dengan kecepatan yang sama

Word

- Instruction dan error message
- Instruction: Enter your name, Enter your password

Bad	Date: <input type="text"/>
Better	Date (MM/DD/YYYY): <input type="text"/>
Best	Date: <input type="text" value="January"/> <input type="text" value="01"/> <input type="text" value="2005"/>

- Kita tidak bisa mengeliminasi semua kemungkinan error
- Ada dua pilihan lazy dan energetic
- Lazy akan memberikan response syntax error
- Energetic menyediakan ataucorrect atau saranperbaikan error-nya



Color

- Teks lebih mudah dibaca jika berwarna hitam dan background-nya putih
- Hyperlink biasanya biru, visited hyperlink ungu
- Pemilihan warna dalam web harus selektif



Navigation

- Strategi paling bagus untuk memilih navigasi adalah meniru familiar Internet Application yang paling populer
- Elemen yang paling populer:
 - Navigation directory dari halaman
 - News dan event
 - Single text input box untuk site-wide search
 - Quick form targeting untuk services yang paling sering di request
- Harus dapat menjawab pertanyaan where am i?, where have I been?, where can I go?



Information Architecture

- Misalnya ada 1000 content item dalam sebuah site, cara untuk organizing, labeling, dan presenting 1000 item tersebut mengacu pada information architecture dari site tersebut
- Meski satu halaman juga membutuhkan information architecture, misalnya:
 - sort from newest to oldest (good for experienced users)
 - sort from highest quality to lowest quality (might be good for first-time users)
- Dikategorikan berdasarkan apa (Potraits, Architecture, wedding, family, animal)



Information Architecture

- Dikategorikan berdasarkan tipe kamera “Digital Point of Shoot, Digital SLR, 35mm”