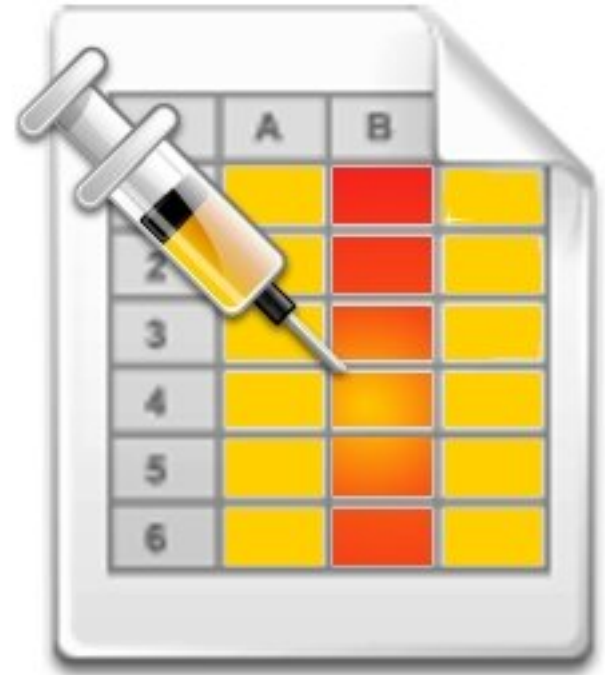


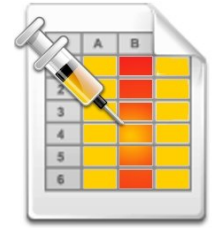
Self-Healing Databases



managing schema updates in the field
or, “peddling drugs to Sakila”

MySQL Miniconf, linux.conf.au 2007
Jonathan Oxe <jon@ivt.com.au>

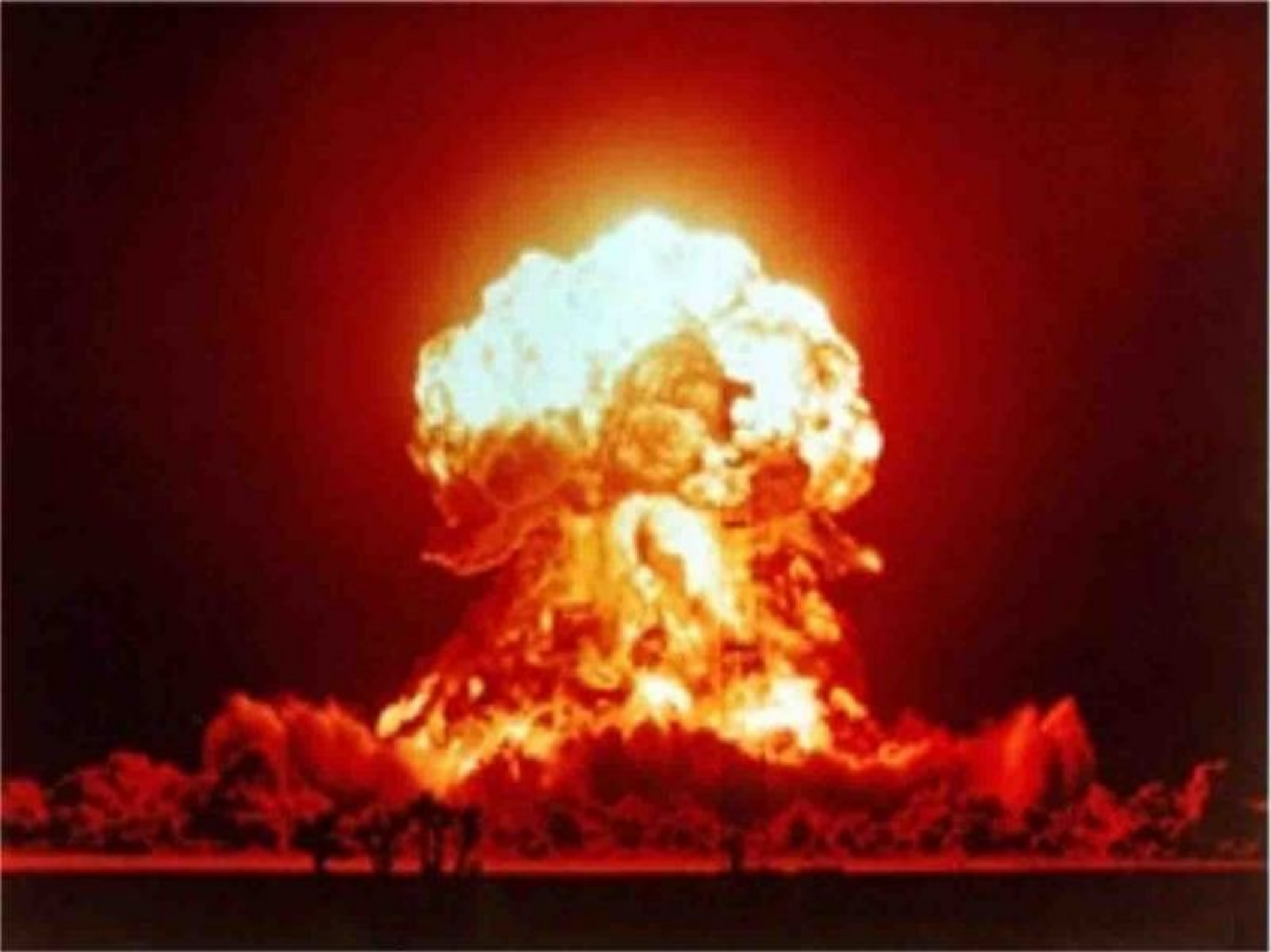
The Problem



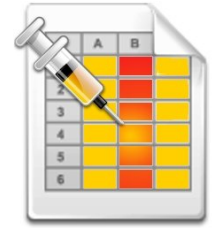
Applications are not static.

New versions mean schema changes.

App / schema mismatches are bad.



The Problem

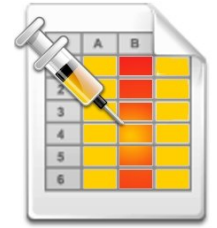


Applications are not static.

New versions mean schema changes.

App / schema mismatches are bad.

The Problem



Applications are not static.

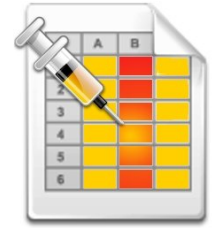
New versions mean schema changes.

App / schema mismatches are bad.

Schema changes mean pain.



The Problem



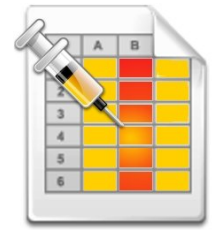
Applications are not static.

New versions mean schema changes.

App / schema mismatches are bad.

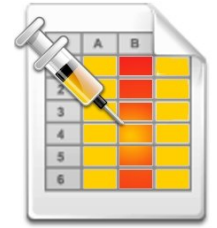
Schema changes mean pain.

Obvious Solution



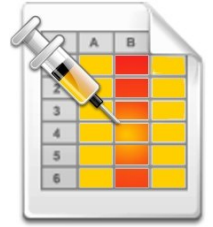
Update scripts

Update Scripts

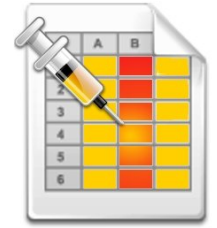


~~Run manually~~

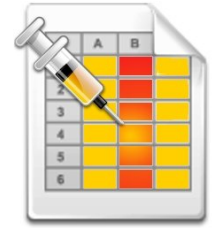
Update Scripts



~~Statically defined~~

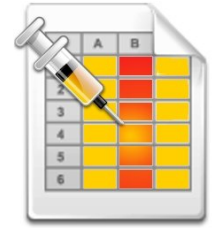


A better way?



Self- Healing Databases

Reasons For Change



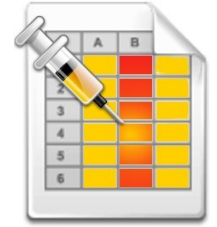
New tables required.

New columns required.

Alterations to columns.

Alterations to contents of tables.

Failure Modes



New tables required.

“Unknown table”

New columns required.

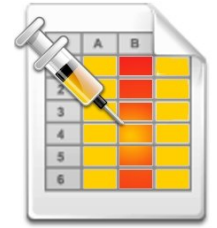
“ Unknown column”

Alterations to columns.

?

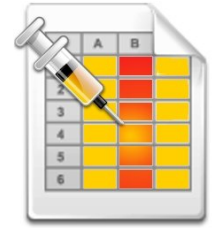
Alterations to contents of tables.

?



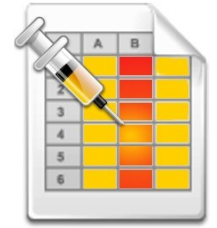
Reactive, not proactive

Smart Error Trapping



1. Run queries blindly.
2. Detect failure conditions.
3. Fix them.
4. Profit!

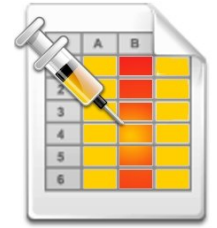
But...



...if you don't have a db
abstraction layer you're

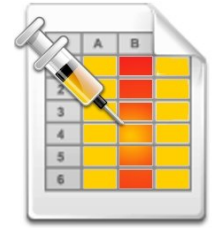
stuffed!

Build, Borrow or Steal



One central query executor

MySQL Errors



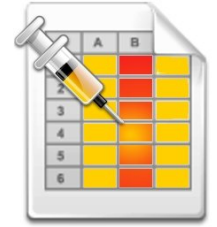
MySQL has built-in error reporting: use it!

In PHP:

```
$errno = mysql_errno($link);  
$error = mysql_error($link);
```

Specify the link or you'll get the value from the last opened connection, not the last error from your connection.

MySQL Errors



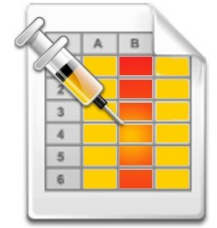
Check for specific errors, such as:

1146: Table doesn't exist

1054: Unknown column

dev.mysql.com/doc/refman/5.0/en/error-handling.html

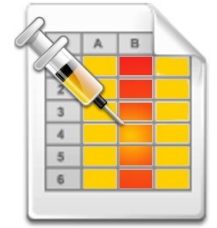
Missing Table



- Store reference schemas in app
- Trap “1146” errors
- Examine error to determine table name
- Load reference schema
- Create table
- Rerun original query
- Return result

The user never even notices a glitch :-)

Missing Table

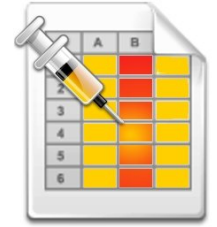


Embed reference schemas into your app.

```
[modulename]/sql/articles.sql:
```

```
CREATE TABLE `articles` (  
  `Serial` INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
  `Title` VARCHAR( 255 ) NOT NULL ,  
  `Article` TEXT NOT NULL  
) ENGINE = MYISAM ;
```

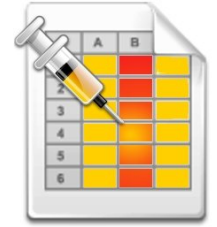
Missing Column



- Record schema changes in “alter” file
- Trap “unknown column” errors
- Load and execute alter file
- Rerun original query
- Return result

No harm, no foul.

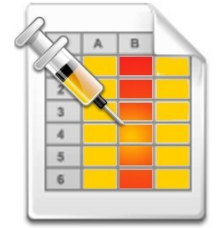
Missing Column



Make execution of “alter” file idempotent.

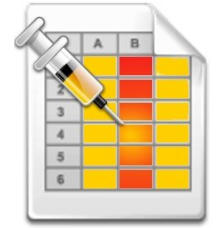
```
[modulename]/sql/alter.php:
```

```
if (!$dbase->field_exists("news", "Modified"))
{
    $s = "ALTER TABLE news ADD `Modified` TIMESTAMP
        NOT NULL";
    $dbase->query($s);
}
```



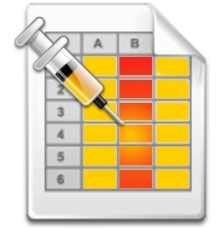
That's not all, folks!

Problem:



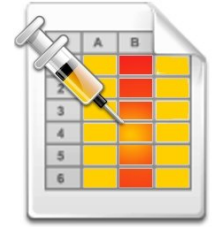
Multiple module
instances require
data partitioning

Solution:



Three-tier dynamic table naming scheme

Dynamic Table Names



1: Module instance

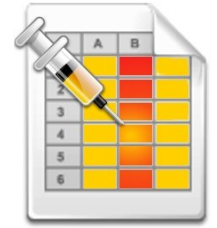
2: Module name

3: Specific table

hotstuff_news_articles

hotstuff_news_comments

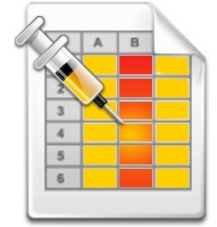
Benefits



Storage of schema with module: error handler can deduce path from table.

Upgrade of tables when you don't know their name.

Schema Templates

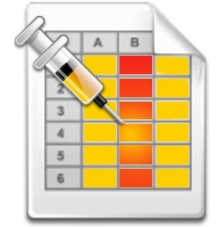


Placeholders in reference schemas

[modulename]/sql/articles.sql:

```
CREATE TABLE `articles` (  
  `Serial` INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
  `Title` VARCHAR( 255 ) NOT NULL ,  
  `Article` TEXT NOT NULL  
) ENGINE = MYISAM ;
```

Schema Templates

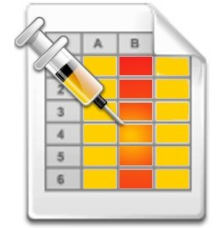


Placeholders in reference schemas

[modulename]/sql/articles.sql:

```
CREATE TABLE <articles> (  
  `Serial` INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
  `Title` VARCHAR( 255 ) NOT NULL ,  
  `Article` TEXT NOT NULL  
) ENGINE = MYISAM ;
```

“Alter” Templates

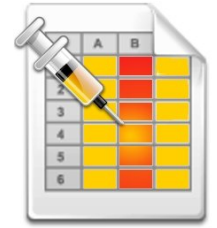


Make table names in “alter” file dynamic

[modulename]/sql/alter.php:

```
if (!$dbase->field_exists("articles", "Modified"))
{
    $s = "ALTER TABLE `articles` ADD `Modified` TIMESTAMP
        NOT NULL";
    $dbase->query($s);
}
```

“Alter” Templates

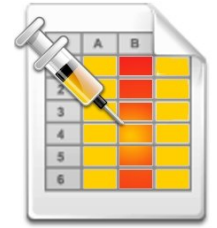


Make table names in “alter” file dynamic

```
[modulename]/sql/alter.php:
```

```
$articles = $instance.'_news_articles';  
if (!$dbase->field_exists($articles, “Modified”))  
{  
    $s = “ALTER TABLE $articles ADD `Modified` TIMESTAMP  
        NOT NULL”;  
    $dbase->query($s);  
}
```

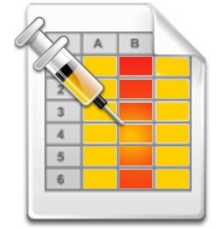
Benefits



Stop caring about:

- App / schema mismatches
- Knowing what tables are called
- Telling users to run upgrade scripts

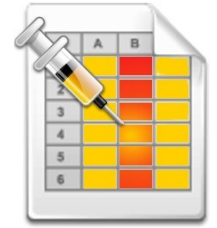
Is This A Fairy Tale?



Technique in production use in the SiteBuilder web application framework and modules for more than 6 years:

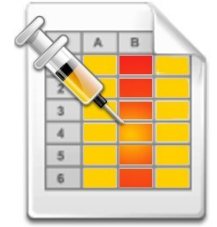
- 1 million lines of PHP
- 96 modules
- 6,829 SQL statements
- 793 embedded table schemas

Is This A Fairy Tale?



Deployments include the Siemens intranet which has over 2,500 dynamically managed tables.

Self-Healing Databases



Thankyou :-)

These slides: jon.oxer.com.au/talks

Shameless plug: www.sitebuilder.com.au

Contact: Jonathan Oxeer <jon@ivt.com.au>