



IUG 24.

TOGETHER WE DRIVE INNOVATION

March 25 - 27, 2024 | Detroit, Michigan

Hackathon: Detect Changes to Sierra Code Tables

Bob Gaydos: Stark Library, OH

Rebecca King: Thousand Oaks Library, CA

Devan Parkison: Skokie Public Library, IL

The Problem

The Sierra Administrator adds, occasionally renames and on rare occasions, deletes fundamental system codes.

Common examples include Location Codes, Material Types, Item Types, Statistic Groups, Patron Types and Loan Rules.

Unfortunately, the busy admin can forget to notify all colleagues, vendors and integration partners of these updates to the Sierra database.

The admin also needs to remember to update related tables, such as Admin Corner's Locations Served and Bibliographic Locations tables.

The Solution

1. A scheduled Python job takes nightly snapshots of targeted Sierra codes tables and stores these until the next run of the job.
2. Upon that next run, query each codes table and compare it to its snapshot.
3. If any additions, deletions or changes are detected, send email to a list of contacts who need to or wish to be alerted to these changes.
4. Take a new set snapshots as a baseline to use for the next run of this job.



Contact Maintenance Form Using PHP Coding

```
1 <?php
2
3 include "admin/db_inc.php";//database connection
4
5 session_start();
6
7     $_SESSION['reg']=array();
8     $FirstName=$_POST['FirstName'];
9     $LastName=$_POST['LastName'];
10    $Age=$_POST['Email'];
11    $Phone=$_POST['ContactType'];
12    $School=$_POST['Organization'];
13    if(empty($_POST))
14    {
15        header("location:contact.php");
16    }
17    else
18    {
19        $link=mysql_connect("Server IP","username","password");
20        mysql_select_db("dbname",$link);
21        $p="select * from Contact where FirstName='$FirstName'";
22        $ros=mysql_query($p,$link);
23
24        if(empty($FirstName))
25        {
26            $_SESSION['Contact']['FirstName']="Please enter your First Name.";
27        }
28        if(empty($LastName))
29        {
30            $_SESSION['Contact']['LastName']="Please enter your Last Name.";
31        }
32        if(empty($Email))
33        {
34
35            $_SESSION['Contact']['Email']="Please enter your Email.";
36        }
37        if(empty($ContactType))
38        {
39            $_SESSION['Contact']['ContactType']="Please enter your Contact Type.";
40        }
41        if(empty($Organization))
42        {
43            $_SESSION['Contact']['Organization']="Please enter your Organization.";
44        }
45    }
46    if(empty($_SESSION['Contact']))
47    {
48        $sql="INSERT INTO Contact (FirstName,LastName,Email,ContactType,Organization)
49            VALUES('$FirstName','$LastName','$Email','$ContactType','$Organization)";
50        mysql_query($sql,$link);
51        header("location:confirmation.php");
52    }
53    else
54    {
55        $_SESSION['data']=array();
56
57        foreach($_POST as $PID=>$val)
58        {
59            $_SESSION['data'][$PID]=$val;
60        }
61        header("location:contact.php");
62    }
63
64 ?>
```

Contact Maintenance Form on Web View

Contact

Name *

First Name

Last Name

Email *

example@example.com

Type *

- Internal
 External

Code Table *

- Location Code
 Item Type
 Patron Type
 Material Type
 Statistical Group
 Loan Rule

Submit

Contact Data from PHP Insert to the Table

The screenshot shows the phpMyAdmin interface for a MySQL database. The browser address bar indicates the server is 'sql5.freemsql.com', the database is 'sql5693958', and the table is 'Contact'. The 'Structure' tab is active, showing the table's schema. A green message at the top states 'Table Contact has been altered successfully.' Below this, the SQL command used for the alteration is displayed: `ALTER TABLE `Contact` CHANGE `Internal` `ContactType` CHAR(50) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL;`. The table structure is shown in a table with columns: #, Name, Type, Collation, Attributes, Null, Default, Comments, Extra, and Action. The columns are: 1. First name (char(50), utf8_general_ci, No, None), 2. Last Name (char(50), utf8_general_ci, No, None), 3. Email (varchar(50), utf8_general_ci, No, None), 4. ContactType (char(50), utf8_general_ci, No, None), and 5. Organization (char(63), utf8_general_ci, No, None). Each column has a 'Change' icon and a 'Drop' icon. Below the table structure, there are options to 'Check all', 'With selected', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', and 'Index'. At the bottom, there is a section for 'Indexes' with a message: 'No index defined!'.

Server: sql5.freemsql.com » Database: sql5693958 » Table: Contact

Table structure Relation view

✓ Table Contact has been altered successfully.

```
ALTER TABLE `Contact` CHANGE `Internal` `ContactType` CHAR(50) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL;
```

[Edit inline] [Edit] [Create PHP code]

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	First name	char(50)	utf8_general_ci	No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values
<input type="checkbox"/>	2	Last Name	char(50)	utf8_general_ci	No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values
<input type="checkbox"/>	3	Email	varchar(50)	utf8_general_ci	No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values
<input type="checkbox"/>	4	ContactType	char(50)	utf8_general_ci	No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values
<input type="checkbox"/>	5	Organization	char(63)	utf8_general_ci	No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values

Check all With selected: Browse Change Drop Primary Unique Index

Print Propose table structure Move columns Improve table structure

Add column(s)

Indexes

No index defined!

Python Code Example (Database Querying)

```
#connecting to database
try:
    conn = psycopg2.connect("dbname='%s' user='%s' host='%s' port='1032'
        password='%s' sslmode='require'" % (DB_NAME, DB_USER, DB_HOST, DB_PASSWORD,))
except psycopg2.Error as e:
    print("Unable to connect to database: " + str(e))

cursor = conn.cursor()

#query table properties
cursor.execute("SELECT * from sierra_view.ptype_property_name")
oldPtypeTable = cursor.fetchall()
cursor.close()
# (mock up "changed table")
currentPtypeTable = oldPtypeTable.copy()
currentPtypeTable.append((800, 'Hackathon Test', 1))
currentPtypeTable.append((801, 'Hackathon Test 2', 1))
```

Python Code Example (Detecting Changes)

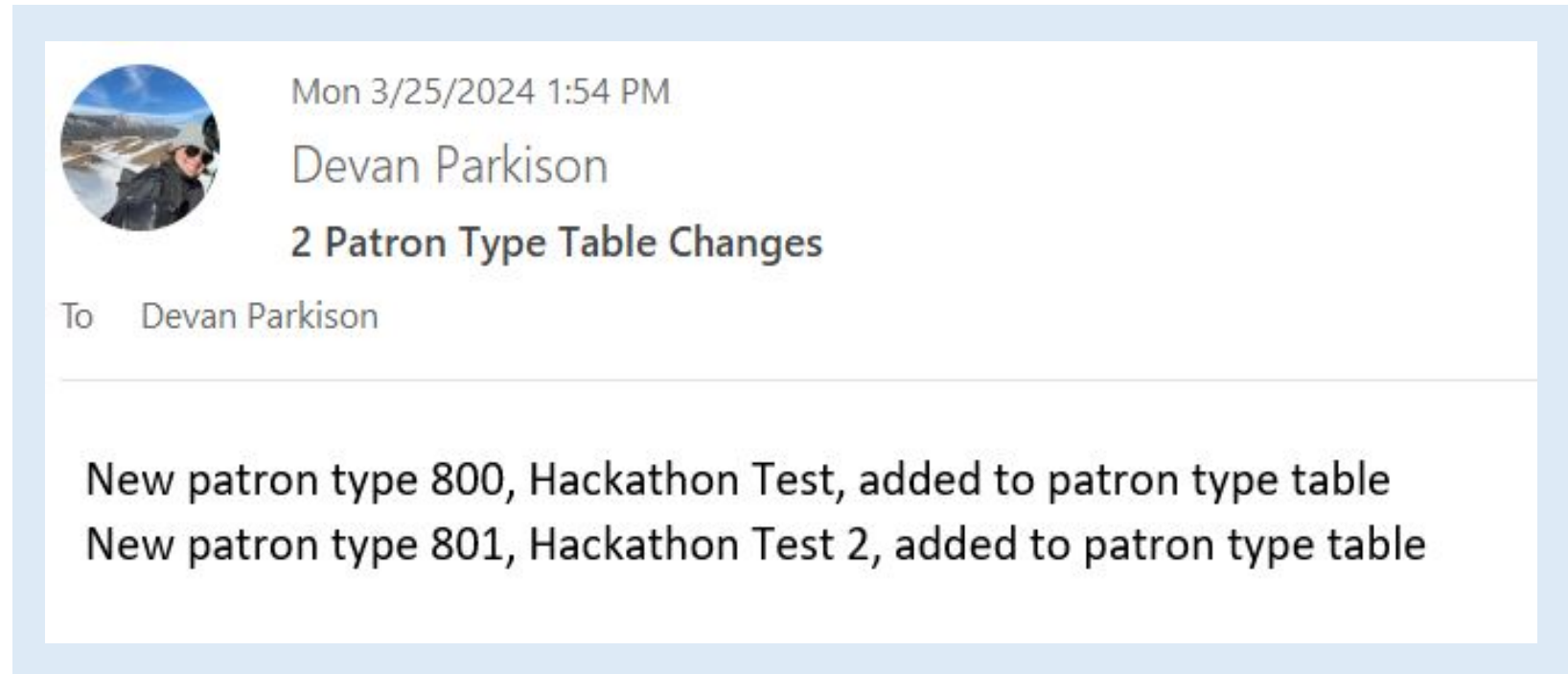
```
# List to hold changes in data and email lines
changed_data = []
body = []

for ptype in currentPtypeTable:
    found = False
    for oldtype in oldPtypeTable:
        if ptype[0] == oldtype[0]: # Check if IDs match
            if ptype != oldtype: # Check if the rows are different
                changed_data.append((oldtype, ptype))
            found = True
            break
    if not found: # If ptype not found in oldPtypeTable
        changed_data.append((None, ptype))
```


Output to Send Email

Based on the contact database from the contact form, an email is sent to interested parties outlining changes made to sierra tables.

Email Sample





IUG 24.

TOGETHER WE DRIVE INNOVATION

March 25 - 27, 2024 | Detroit, Michigan

THANK YOU

Questions?

Bob Gaydos bgaydos@starklibrary.org

Rebecca King rking@tolibrary.org

Devan Parkison DParkison@skokielibrary.info