

Auditing

Auditing may be achieved in an automated fashion by conforming to a few simple database conventions.

First, the table that you want to audit needs to have a "user_id" which correlates to the user making the change. So, when a user creates, updates, or deletes a row, their user_id is used to track the change. Then, a few SQL scripts may be executed to generate the triggers necessary to populate the audit table.

To create the audit triggers, the following scripts may be run to generate the corresponding SQL statements.

```
SELECT CONCAT('CREATE TRIGGER ', table_name, '_update_audit AFTER UPDATE ON ', table_name,
' FOR EACH ROW BEGIN DECLARE i_index BIGINT(20); SET i_index = (SELECT MAX(cluster)+1 FROM
lms_audit); ',
audit_column_gen(table_name, 'UPDATE'), ' END$$') FROM information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

```
SELECT CONCAT('CREATE TRIGGER ', table_name, '_insert_audit AFTER INSERT ON ', table_name,
' FOR EACH ROW BEGIN DECLARE i_index BIGINT(20); SET i_index = (SELECT MAX(cluster)+1 FROM
lms_audit); ',
audit_column_gen(table_name, 'INSERT'), ' END$$') FROM information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

```
SELECT CONCAT('CREATE TRIGGER ', table_name, '_delete_audit AFTER DELETE ON ', table_name,
' FOR EACH ROW BEGIN DECLARE i_index BIGINT(20); SET i_index = (SELECT MAX(cluster)+1 FROM
lms_audit); ',
audit_column_gen(table_name, 'DELETE'), ' END$$') FROM information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

The above scripts require the audit_column_gen function available in the schema/lms_functions.sql file.

To delete the audit triggers, the following scripts may be run to generate the corresponding SQL

statements.

```
SELECT CONCAT('DROP TRIGGER IF EXISTS ', table_name, '_update_audit$$') FROM
information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

```
SELECT CONCAT('DROP TRIGGER IF EXISTS ', table_name, '_insert_audit$$') FROM
information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

```
SELECT CONCAT('DROP TRIGGER IF EXISTS ', table_name, '_delete_audit$$') FROM
information_schema.columns
WHERE table_schema = DATABASE()
AND column_name = 'user_id'
AND table_name NOT LIKE 'lms_post'
AND table_name NOT LIKE 'lms_message'
AND table_name NOT LIKE 'lms_user'
AND table_name NOT LIKE 'lms_user_cookie'
AND table_name NOT LIKE '%audit%'
AND table_name NOT LIKE '%object%';
```

- [Log in](#) [1] to post comments

Source URL: <http://www.blackhillsystems.com/?q=node/64>

Links

[1] <http://www.blackhillsystems.com/?q=user/login&destination=node/64%23comment-form>