civicon Denver 2015

Forena Reports: Alternative Reporting

Eliot Mason
Mercury Motos
eliot@mercurymotos.com

What this session Is and Isn't

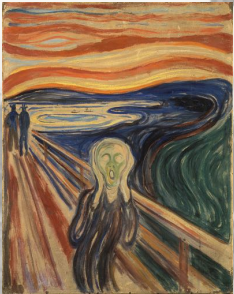| IS	ISN'T
❖ A look at an alternate method of generating reports	❖ A recommendation
❖ For Drupal sites	❖ A reporting method cage-match
❖ For people who can, or want to, write SQL queries	❖ A How-To for Forena

---

Background

❖ CiviReport does lots of great things for us... But there are times when we need more

❖ BI solutions (e.g. Jasper, Pentaho) are powerful alternatives - but with significant costs & complexities

❖ Drupal Views can do wonders with content, was not made for summary stats

---

A Note on CiviReport

❖ Thanks to Lola Slade of Freeform for her CiviReport 101 session

❖ Customizing it requires multiple skill sets and configurations

  ❖ Php class? Array? SQL aliases? PHP Debug

Customizing CiviReport Makes You Feel?

Forena

- ❖ Drupal module (v6 & v7)
- ❖ Fairly rapid development
- ❖ drupal.org/project/forena
- ❖ In-module help, drupal forum, author videos
- ❖ Runs on your Drupal server, uses PDO
  - ❖ Can access multiple databases simultaneously

Adoption Case

- ❖ Client had a need for a range of reports on custom data with a lot of summary statistics
- ❖ Multiple groups with the organization, each one with a different set of data, granting agencies and general data demands

- ❖ Elected to try a different approach …

Forena Features

- ❖ Custom parameter/drop-downs
- ❖ CrossTab/Pivot tables (dynamic!)
- ❖ Multiple Reports (queries) on a single page
- ❖ Easy color-coding of results with CSS
- ❖ Report Builder interface (currently "experimental")
- ❖ Create Drupal Blocks
- ❖ Insert Forena Block into a Views page
- ❖ URL Parameters
- ❖ Graphing/Charting

Caveats

- There is a learning curve here... How does CiviCRM store data?
- Your SQL statements aren't future proof - and not necessarily platform proof
- Outside of core
  - Bad Community Member?

Two Pieces of Forena

- Forena is built around the idea that data and presentation are distinct steps
1. Build your data using SQL (data sources & data blocks)
2. Present that data in a multitude of ways (report template)

Forena Pt 1 - Data Blocks

- Very Straightforward SQL queries, stored in text files
- Files can be nested with an "—Include=" statement
- Accept Parameters and basic conditional statements

- => Allows modular, re-useable logic

```
--ACCESS=access content
select
sum(if(chicas_household_count=1,1,0)) as n_single_child,
sum(if(chicas_household_count=2,1,0)) as n_double_child,
sum(if(chicas_household_count=3,1,0)) as n_triple_child,
sum(if(chicas_household_count=4,1,0)) as n_quad_child,
sum(if(chicas_household_count>4,1,0)) as n_bigfamily_child

from (
select cc.household_name, cc.id as household_id, count(cc.id)
as chicas_household_count, '1' as family_indexor from
civicrm_contact as cc
join civicrm_relationship as cr on cr.contact_id_b=cc.id
 join (
--INCLUDE=membership_list
) as members on members.contact_id=cr.contact_id_a
where cc.contact_type='Household'
group by cc.id
) as households;
```

Datablock Quirks

- Use Single Quotes in your SQL statements - Double Quotes get stuck
  - e.g. where contact_type='Individual'
- The Forena parser expects Forena syntax at the beginning of the line (e.g. "—Include", "—IF")
- Since there is no abstraction layer, your SQL statements might be specific to your SQL server
- Tied directly to Drupal Permissions
- The debugger blames the wrong file

```
--ACCESS=access content
select
sum(if(chicas_household_count=1,1,0)) as n_single_child,
sum(if(chicas_household_count=2,1,0)) as n_double_child,
sum(if(chicas_household_count=3,1,0)) as n_triple_child,
sum(if(chicas_household_count=4,1,0)) as n_quad_child,
sum(if(chicas_household_count>4,1,0)) as n_bigfamily_child

from (
select cc.household_name, cc.id as household_id,
count(cc.id) as chicas_household_count, '1' as
family_indexor from civicrm_contact as cc
join civicrm_relationship as cr on cr.contact_id_b=cc.id
 join (
--INCLUDE=membership_list
) as members on members.contact_id=cr.contact_id_a
where cc.contact_type='Household'
group by cc.id
) as households;
```

Forena Pt II - Presentation

* Datablock output is run through a separate presentation layer, defined in an XHTML document ("Forena Template")

* Has some XPATH calculation abilities (e.g. can make its own sums)

* Datablocks and Presentation are de-coupled

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
<!ENTITY nbsp " ">
]>
<html xmlns:frx="urn:FrxReports">
<head>
<title>Chicas Attendance</title>
<frx:category>Chicas</frx:category>
<frx:options/>
<frx:parameters>
    <frx:parm id="p_school_name"
        data_source="local_civicrm_drupal/plist_school_name"
        type="select" label="School Name" require="1">
        </frx:parm>
    <frx:parm id="p_school_year"
        data_source="local_civicrm_drupal/plist_school_year"
        type="select" label="School Year" require="1">2014-2015
        </frx:parm>
    <frx:parm id="p_calendar_quarter"
        data_source="local_civicrm_drupal/plist_calendar_quarter"
        type="select" label="Calendar Quarter">
        </frx:parm>
    <frx:parm id="p_activity_name"
        data_source="local_civicrm_drupal/plist_chicas_attendance"
        type="select" label="Activity">
        </frx:parm>
</frx:parameters>
<frx:docgen>
</frx:docgen>
<frx:fields>
<frx:field id="sort_name" link="civicrm/contact/view?
reset=1&amp;cid={contact_id}"></frx:field>
</frx:fields>
<frx:menu/>
<frx:cache/>
<style/>
</head>
<body>
<div id="activity_block" class="FrxTable"
frx:block="local_civicrm_drupal/chicas_attendance">
    <table>
        <thead>
        <tr>
            <th>Name</th>
            <th>Attended</th>
            <th>Excused</th>
            <th>Not Excused</th>
            <th>Late</th>
            </tr></thead>
        <tbody>
        <tr id="activity" frx:foreach="*">
            <td>{sort_name}</td>
            <td>{Attended}</td>
            <td>{Excused}</td>
            <td>{Not Excused}</td>
            <td>{Late}</td>
        </tr></tbody></table></div></body>
</html>
```

So … How's it working?

* Deployed with one client … who is at the base of the adoption curve

* Learning curve... But its mostly one of terminology

* VERY fast execution

* Easy to modify and expand

    * Add a field? Add a parameter? no problem!

Forena Pt III (?)

* Forena has an experimental report builder

* Provide Data Blocks … let users do what they will…

* But it's in development

    * involves lots of clicking

    * I don't want the current client to have it yet

    * and if I want to click stuff, there's Views

Deploy & Share

❖ Easy to deploy across sites

  ❖ Install Module
  ❖ Configure data blocks
  ❖ Copy Directories & Files

    ❖ Use git?

  ❖ No SQL abstraction layer

    ❖ potential issues across flavors of SQL

Let's Look at it in ACTION

Conclusion

❖ Forena is a viable tool to address your custom report needs

❖ The Report Builder could work for savvy users

❖ There is potential for community sharing of data blocks

GOTTA FLY

Seriously. I'm going to the airport **now**.
I'd love to handle any questions you have via email … eliot@mercurymotos.com