Beyond the Tally Sheet

Communicating Data

Jay Forrest, Georgia Tech Library
RaeAnne Forrest, Georgia Tech Library

Jay Forrest is the Collections Assessment & Statistics Coordinator at the Georgia Tech Library
RaeAnne Forrest is an Information Associate I at the Georgia Tech Library
Evolution

{ From paper tick sheets to browsers
Prior to the online system, we used paper tally sheets and recorded observations at each service point during over 8 weeks (3 in Spring/Fall and 2 in Summer). We recorded directional, reference, and research questions according to ARL definitions and tracked transactions by contact method (phone, in-person, virtual).

In 2002, in preparation for the opening of the Library West Commons, we expanded our data collection efforts and recorded data weekly. We also captured IT specific questions as our new Information Services Desk would handle both Reference and IT questions.
In 2003, Scott Riggle, Georgia Tech Office of Information Technology, and Kathy Tomajko, Georgia Tech Library, created our first tally sheet using PHP and MySQL. This tally sheet included 4 types of questions (directional, traditional, information technology, and other services. In the beginning we created a single library-wide tally sheet.
Beginning with 2004, Jay started to customize the tally sheet to reflect the needs of each department, with each unit head being able to determine the questions that they wanted to track.

Some units like Security and Information Delivery (Interlibrary Loan) didn’t have all types of tallies, because these transactions did not occur at their service point.

The sidebar tally sheet grew out of a common set of links that were kept for the service desk computers. We began with a frameset and later used a custom made firefox plug-in. The sidebar allows for an incredible amount of flexibility, and lets us unite other elements.
In Summer 2011, Georgia Tech implemented a single service point: the Library Service Desk.

This was to be a one-stop shop to serve patron needs: reference assistance, access services, and IT support.

Problem: alterations in workflow and pacing affected staff ability to accurately tally.

Solution: redesign and restructure the tally sheet to fit the new needs.
The pre-2011 Tally Sheets. The one on the left was used by the Information Services (Reference) Desk, and the one on the right by the Circulation Desk.
The Information Services Desk sidebar includes:

Widgets for the in-house developed calendar and schedule widgets (a virtual in/out board, a virtual calendar, and a daily and weekly desk scheduling tool).

A tally sheet, with drop-downs. (For subject librarians this tally sheet also includes the ability to set a transaction date, and select a contact method)

Common links using roll-over menus are divided by category, and quick search widgets.
The Circulation sidebar uses two panes and has a toggle to go back and forth between the panes.

The tally sheet is a one-click model designed by RaeAnne that includes an imbedded time spent – based on the experience at the desk and our history of past transaction tallies.

The department sidebar features common links and divides the links into Access Services and IT resources.

The pages originally were hosted on two separate campus servers to facilitate coding. The tally sidebar was the original default coding, until hiccups occurred with the host server. The department sidebar was then made the default.
For the new design, the best elements from each sidebar were used.

The Calendar and Schedule Widgets

Common buttons for flipping between the tally sheet and departmental links (the page refreshes to the tally sheet after 30 seconds from the links pages)

A feedback box for tallies.

Contact method buttons.

A special checkbox for staff that handle access services email questions.

One click tallies divided by type and differentiated by color. Yellow (Orange), Green (Reference), Purple (Access Services), Blue (IT), Yellow (Advising): large buttons for more frequent transactions, and further down the page smaller buttons for less frequent transactions.
Department links sidebars.

Reference includes toggled links by category.
How the sidebar appears on the page. We have developed a sidebar plugin for Mozilla-based browsers that loads the sidebar with each new window and places the sidebar on the left side of the page. There is also a toggle icon on the menu bar to turn the sidebar on and off.
Circulation talliesheet/sidebar deployed in 2008 – note increase in tallies

Partway through 2010, changes in the back-end databases meant the links page was made the “front” primary view and the talliesheet had to be accessed by clicking the link – note the drop in tallies

2011 – Still had to click to view tally sheet; reference and circulation were combined at one desk and tally sheets were different between departments; lack of emphasis on tallies

2012 – two separate sidebars, but both loaded on every computer and staff could switch between the two easily by clicking a button in the browser. Increased emphasis on tallies but staff complained about ability to tally quickly.
Redesigned tally sheet deployed in April 2013
Tallies increased by 194% between Aug 2012 (1,147) and Aug 2013 (3,376)
Tallies increased by 173% between Sept 2012 (1,258) and Sept 2013 (3,445)
Visualization

Sharing data with the world
Visualization began with exporting the Tally Sheet Data from MySQL into EXCEL, creating Graphs in EXCEL, and exporting the graphs as .png files. The graphs were then added to a public website for Tally and Attendance statistics.
PHPlot allows for dynamic graphs that are server-side generated. Here we show the time and time spent for library wide tallied transactions across three types (other services, information technology, and traditional reference). You can also see the individual questions that were tallied.

The logarithmic scale facilitates comparison (each graph uses the same axis) and allows the mapping of two functions (tallies and time spent) on the same graph.

http://www.phplot.com/
In 2013, Jay added tally transactions to our library dashboard.  
http://www.library.gatech.edu/dashboard.

The dashboard provides monthly updates to several library metrics.

Graphs are generated on the client-side, using the Google Chart API:  
https://developers.google.com/chart/

We generally upload the data monthly to a MySQL database (exceptions include locally generated data (attendedance, tally transactions, instruction)). To compensate for the lag, especially with several complex data sets, we have an internal php page that runs the queries and updates that chart data. We then copy the html and save into the public facing page. It means the data is not real-time, but as most of the data relies on a monthly data upload, this was not a large concern.
Application

{ Our tally sheet, your library
We are now going to take you through the steps to create your own library tally sidebar.

For more detailed instructions, see the companion website: http://www.library.gatech.edu/libdata/demoTally/main.html
Basic XHTML header.

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
xml:lang="en">
<head>
<title>Library Tally Sidebar: Demo</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<style type="text/css"></style>
</head>
</html>
```
The demo tally is available for use via the Gnu Public License v.3: http://www.gnu.org/copyleft/gpl.html
Connecting to the MySQL database using MySQLi
Here we create a feedback box on the tally sheet. It is empty until a tally is submitted.

The tally is validated and if it passes validation it is added to the database. The page is then refreshed to clear the feedback box.

Each tally requires its own validation step, but all use the same databases insert.
Here is the example of one tally sheet button, and the closing of the webpage.
Next let's make a dashboard using Google Chart.

Google Chart has some nice features over PHPPlot, including the ability to hover over data points to get values, and over the legend to see the results.
Starting with the xhtml header, we add the div id to the css to get the layout effect that we want. In this our case, last 13 months on the left, and past 10 years on the right. We chose to use last complete year rather than having a growing column for the first year.
We load the api from Google’s server, and we add our own local data. We can add the data directly into html if we want to. At GT we use PHP/MySQL to load the data.
We then ask the browser to draw the graph and output it to a specific DOM element, in this case the div tag `chartTransactionMonth`.
We add the div tags and close the page.
Some links to the demoTally, the guide, and the GT Library Dashboard.

- Demo Tally: http://www.library.gatech.edu/libdata/demotally
- Demo Tally How-To:
  http://www.library.gatech.edu/libdata/demotally/main.html
- GT Library Dashboard:
  http://www.library.gatech.edu/dashboard
- Jay Forrest: jay.forrest@library.gatech.edu
- RaeAnne Forrest: raeanne.forrest@library.gatech.edu