

A Database to Preserve and Define Swing Dance Moves and Routines

A thesis submitted in partial fulfillment

of the requirements for

the degree of Master of Arts

in Interdisciplinary Computer Science

by

Amy Dewey

Approved by

Computer Science Advisor

Director of ICS program

Dr. Ellen Spertus
Professor, Department of Mathematics and
Computer Science
Mills College, Oakland, CA 94613

Director of Graduate Studies

Dr. Marianne Sheldon
Director of Graduate Studies
Mills College, Oakland, CA 94613

Abstract

Swing dance, a social dance born in the late 1920s, is experiencing resurgence even as the first generation of swing dancers passes away. To preserve the dance and build communication and collaboration among swing dancers, I built a database-driven web site of swing dance moves and routines.

Introduction

Swing dance originated in the late 1920s, just prior to the time of the stock market crash and the beginning of the Great Depression in the United States. During the Depression, swing dancers went to dance contests and dance-a-thons hoping to win money with fancy moves and great stamina [Pick]. As a result, some of them were secretive about their winning moves and were careful about what they taught to others [Rockwell]. Now, 70 years later, those dancers are aging and dying, while their dance form is experiencing resurgence.

The Great Depression is over. People don't rely on dance contests to earn their income anymore, so keeping fancy moves secret is no longer important. Instead, it is imperative to the continuation of the art form to pass along moves and routines to new students. However, swing is developing as it is taught to new generations of dancers. Each dancer adds personal flair, and that flair has built up to becoming some completely different kinds of swing dance. This evolution is only tangentially addressed in classes. If a student goes to a class that teaches a different style than what the student learned previously, it is ignored or "corrected." One would hope the history of swing dance would explain the various styles and how they came to be. The history can be found in books, on web sites, and in documentaries. However, these sources don't always include a way of distinguishing one swing dance from another, sometimes making it unclear which dance the history is about and how that dance relates to other dances.

In response, I built an online database of swing dance moves and routines. Users can enter moves and link them into routines. They can search for moves based on dancer, title or style. They can also add to the database by entering the same move done in different ways, provided they change the name slightly by adding a version number or detail about the move.

Related Work

A site similar to the one described here is the Digital Performance Archive (DPA). This database contains information about different performance pieces that use technology. The term "performance piece" refers to any form of art done by trained professionals that uses time as an element and is somehow performed. Each entry in the DPA is about a single piece. It contains basic details like who created the piece, what it is called, and links to web sites about the piece and the artist who created it. It has a written description of the piece and a biography of the artist. There is a place for a listing of whatever

material the DPA has in regards to the piece, such as reviews, a short description of what the material contains, and bibliographic information. Each entry in the DPA is researched and written by the people who maintain the site, though users can recommend pieces to be added. Because this site addresses only performance art, DPA researchers can get their information online or by contacting a single artist.

Swing dance, however, is a folk art. It changes as it is taught, re-taught and danced by innovative dancers. To research a swing dance step or routine, the researcher has to see and learn it from the people who perform it. Therefore, a database that collects and disseminates swing dance moves needs to collect movements from the people who are developing or doing them. For this reason, this site should allow users to add their own entries without having them approved.

Sonny Watson has created another online dance archive; his is devoted specifically to swing (Watson). While his database is over-flowing with historical information, he does not include any definitions of the different dances. He makes claims as to how dances are related to each other, but he does not give examples of the actual movement to support those claims. To contrast, my site contains less history (users can look for that on Watson's site) and more information on what the dances are and how they are different from one another.

Similar to Watson's site is the Lindy Café. It contains several articles about the history of Lindy Hop as well as historical writings. Again, there is no attempt at defining the movements themselves.

Project Description

The purpose of this project was to create an online database of swing moves and routines¹. This searchable database has some background on each move or routine and a written description of the steps. Users can add moves to the database using a simple interface and link them together to create routines.

At the beginning, I thought about the kinds of functions users should be able to perform. Should they be able to edit or delete moves? If so, should they be required to log in first, so as to lower the chances of someone ruining or deleting someone else's move? Should they be allowed to change or delete only their own moves? I decided to make this a write-once database. Moves and routines can be entered but not edited or deleted. Registered communities have to provide something users really want in order to motivate users to register. Once the database has enough content, it will be able to entice users to register. Until then, it needs to be open to everyone.

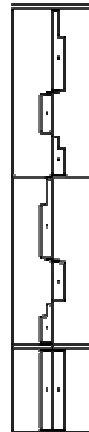
¹ A move is an individual step, something that cannot be separated into parts that stand alone. A routine is a set series of moves, possibly with historical significance.

The Moves

The move entries consist entirely of written descriptions. Video or animation would make this database more useful and enticing to users but also make entering content more difficult for users. Having video would require users to have access to digital video cameras in order to enter moves. Additionally, the videos would have to be uploaded to a server. I would have to provide server space for that as well as a way for users to upload their videos, or I would have to require users to have their own server space. This would bar users without servers and digital video cameras from adding moves to the site. During the time I was putting this project together, YouTube and other video sharing services became popular. Because I was already too far into the project to make that sort of change, I did not consider it as a solution to the server space problem. I decided requiring server space or digital cameras would limit my audience too much. While creating an interface that allows users to animate their moves seems like an excellent idea, I decided it was outside the scope of this project. I could not see getting this project complete in a timely manner that included finding a way to animate the dance moves.

The information pertinent to each swing dance move is a breakdown of the move itself, the information required to link moves together, and the history of the move. The breakdown of the move is what users are coming to the site to find. The information used to link moves allows users to build routines with the moves, and it clarifies the move breakdown. The history backs up the move, it is a way for users to site their sources and hopefully create a lineage showing what kind of changes have happened in swing dance.

In the fall of 1999, I took a course in Labanotation (Kinkead), a very accurate way of writing down movement. It uses a vertical staff separated into columns; each column indicates a part of the body. Polygons are drawn in the columns of the staff. The shapes of the polygons indicate the horizontal direction from the center of the body that the specified body part is pointing and the shadings of the polygons indicate level. Moving up the staff indicates the passage of time. The staff on the right² shows a solo basic two-step. For counts 1-and, the dancer takes a step forward on the left foot and in place on the right foot. On count 2, the dancer takes a step forward on left foot. This is repeated beginning on the right foot for the second measure.



While Labanotation is incredibly detailed and accurate, it takes a long time to master. It is also only known by a very small number of people, making it useless to communicate movement to the general population. I extracted the key elements of the Labanotation model to come up with a detailed way of writing down specifically swing dance that would be immediately clear to my audience. Those elements are time, shape, and direction. For this web site, users indicate shape with radio buttons outlining the connection between partners. Time is broken down in the text fields users have for entering the move details. Direction is left out for lack of a clear way to depict that information.

² Staff was take from <http://www.dancenotation.org/Inbasics/frame0.html>

Time and Movement Breakdown

To describe time, I designed the site to make the users break down the steps into counts. I gave them text fields separated for each count, as shown in Image 1. When I think about swing or try to teach someone to swing dance, I don't think of it in single counts. Instead, the measure is broken down into pairs of counts. For this reason, I decided to allow users to pick the counts, so that they can have one text field for counts 1 and 2 or for just count 1. This numbering system strikes a good balance between making the form too rigid and still making users break the steps down. I included directions on how to label the counts so that there would be a global notation of counts in this database [Image 2].

Move Details		
Count	Lead's Steps	Follow's Steps
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

Image 1: For the move details, I used a table with 8 rows. Each row has a text field for the count(s) it represents and the lead's and follow's steps for that count.

How to specify the counts		
a digit [1-8]		
	specifies which beat in the measure	
-		
	connects two beats when the first isn't subdivided	
&		
	indicates a step happens on the half beat	
a		
	indicates a step happens on the quarter beat	
Examples		
	written	counted
no subdivisions	1 - 2	one two
half beat	1 & 2	one and two
half and quarter beat	1 & a 2	one and a two

Image 2: Directions on how to label the counts







I used the HTML definition list tags to define my list. The definition list organizes the data in the page source. The term and its definition are defined in the code as separate

entities. The <dl> tag surrounds the entire list, <dt> indicates the term and <dd> is the term's definition. Defining the entities separately like that in the code helps different pieces of software understand the page better. For example, this markup is helpful to software that reads the page for users with visual impairments.

Shape and Linking Moves

For shape, the pertinent parts are how the lead and follow are connected and which feet their weight is on. That limits what they can or cannot do next. Some common positions would be the cuddle, ballroom position and the lead's left hand holding the follow's right hand. I thought it would be most clear to look at the relationship of the lead and follow separated into how they are connected and how they relate to one another spatially. I made radio buttons to specify where each of the lead's hands connect to the follow and how the two dancers are positioned in space [Image 3]. This relationship changes during most moves. Rather than bogging users down with too much detail, I put the same set of radio buttons before and after the Move Details table, anchoring each move between a beginning and ending position. It is up to the users entering the data to clarify how the dancers get from the starting position to the ending position.

Beginning Position

Lead's	is holding Follow's					
	Is not holding	Left hand	Right hand	Waist	Hip	Shoulder
Left hand	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Right hand	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
						
	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Lead's first step is on the left foot right foot. Follow's first step is on the left foot right foot, unless this is a solo.

Image 3: The form for setting up the beginning position

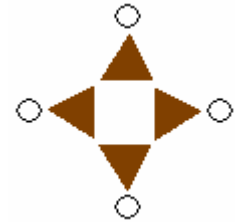
The lead and follow are defined by color: the lead is brown and the follow is green. Dance, being done physically, is easier to describe with images than with words. A goal of mine was to find ways to detail the moves using images that were clear without playing into stereotypes, e.g. blue for boys or leads and pink for girls or follows.

When I first wrote this form, I had the idea that users would leave irrelevant parts blank. For example, they wouldn't put anything for the lead's right hand if the lead's right hand wasn't connected to the follow in some way. As I was testing the form, I quickly discovered that leaving a set of radio buttons blank moved all the subsequent information up an element. For example, if the lead's right hand wasn't connected to the follow, the script handling the move form read the lead's right hand as being on the foot the lead starts on. For this reason, I put in these default values: the lead's hand is not holding the follow, and the move has only the lead in the spatial relationship. When the move is a

solo, the user should fill out the details for the lead, leaving everything else set to its default value.

Thoughts on Direction

The third part of movement that Labanotation records is direction -- whether people or their limbs are moving forward, backward, diagonally or sideways. I could see putting arrows for four of those directions at the end of each row in the move description for forwards, backwards, right and left. The problem was that the meaning wasn't clear. If the dancer changes direction by turning halfway around, do the arrows assume the dancer's new direction, or do they stick to the original orientation of the dancer? And how would a turn be notated?



Another solution might be to record the floor pattern. This would ignore the direction the dancers face. It would be a map of where they move in space. It could be implemented with a square of check boxes. Where in that square would the dancer start? How would the information be stored in the database to present to the users looking up the move? How do the lead and follow patterns fit together? Would the intended meaning be clear? Because there seemed to be no clear resolution to these questions, I decided to leave it up to the user to include directional information in the description.



Background

The last bit of information stored by the database is the background. The background of each entry includes the lineage and the general category it fits in. The lineage is the dancer associated with the move or routine, the dancer who taught the user the move or routine, and the user. My hope is to create a lineage of changes, a way for users to see that everyone who learned from one dancer or that dancer's students dances one way while another dancer's students all dance a different way. It would give validity to variations that some dancers think are "wrong" and create a dialog about what swing dance is.

When users view a move, they are presented with the move title, the starting position, the move details, the ending position and other information including the move category and lineage [Image 4]. The positions are described in words. The move details are presented in the same layout as they are in the form.

8-Count Basic with Follow Double Swivel

Create a Move

Create a Routine

Search for a Move or Routine

What is this?

Send in your Feedback!

Starting Position

Dancers face each other, lead's left hand holding the follow's right.

Move Details

Count	Lead	Follow
1-2	Rock-step back on left	Swivel step right-left
3&4	Triple-step on left, right arm on follow.s waist, walking clockwise to the other side of her	Triple-step on right
5-6	Step left, step right, going clockwise to the original position, let go of follow.s waist on 5	5&6, triple step on left
7&8	Triple-step on right	7-8, Swivel step right-left

Ending Position

Dancers face each other, lead's left hand holding the follow's right.

Other information

This is 8-count Lindy Hop
Entered by Amy Dewey
Taught by Craig Faulkner

Image 4: The final move presentation

The Routines

A routine is a list of moves and some background information. The background information is the same as for the moves – the title, user, teacher, dancer associated with the routine, and any notes that the user wants to include. There is no category for the routines. The different categories of swing dance are mostly compatible and are often combined in routines.

When I first starting working on this form, I though the user would painstakingly enter the names of each move to create the list of moves. I would make it easier by having a “look up” button next to each text field allowing users to look up the moves they want to add. At the end of the form would be a check box the user could check if more moves needed to be added to the routine [Image 5].

Routine

Measure	Move Name	
1	<input type="text"/>	Look up move
2	<input type="text"/>	Look up move
3	<input type="text"/>	Look up move
4	<input type="text"/>	Look up move
5	<input type="text"/>	Look up move
6	<input type="text"/>	Look up move

Add more moves to the routine

Image 5: Preliminary form for entering the moves into a routine

If the user checked the box to “Add more moves” and then clicked “Save this Routine,” the next page would have another table just like the routine table here, only the first measure would be seven and there would be more rows. The checkbox would still be available at the bottom to add more moves.

When it came down to writing the script itself, I wasn’t sure how to make my original vision of it work. I wanted the “Look up move” links to open in a pop-up window with a list of possible moves. The user would click on a move, the pop-up would close, and the chosen move name would be automatically populated in the correct text field. Following the path of least resistance, I found a way I liked better. I gave users a single text field for the move name and a look-up button next to it [Image 6].

Routine

Measure	Move Name	
1	<input type="text"/>	<input type="button" value="Look up move"/>

Look up the move to add it to the routine. Once all your moves are added, click 'Save this Routine'

Image 6: Final form for enter the moves into a routine

When users click on “Look up move,” they are taken to a page of search results. If the user is adding the first move to the routine and doesn’t enter anything into the Move Name field, a list of all the moves in the database will be displayed [Image 7].



Image 7: Moves that can be added to the routine

After selecting a move, users are taken back to the routine form. Now, the first measure is written in; there is a new field for a second measure; and the “Save this Routine” button is enabled [Image 8].

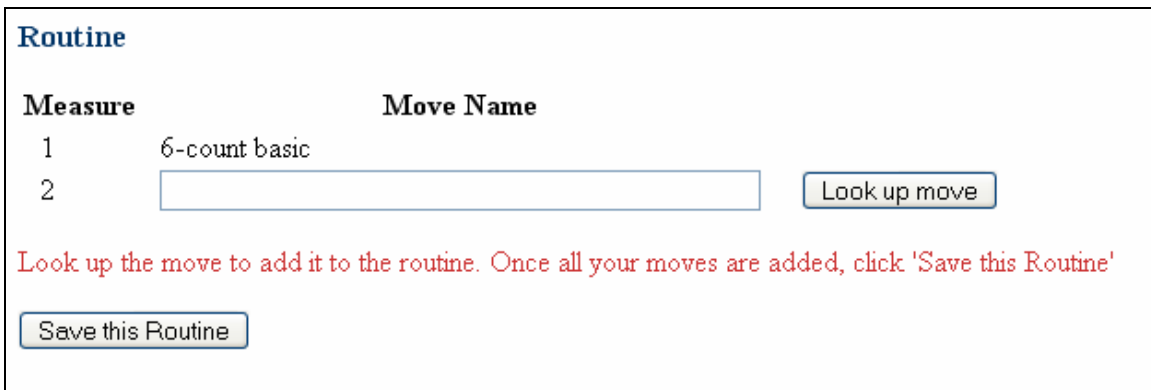


Image 8: The user can add a second move or save the routine as is.

The next time the user clicks on “Look up move,” the results are limited to moves that can come after the move already in the routine [Image 9]. The criteria for such moves are as follows:

- The solo value is the same as that of the first move.
- The lead’s left hand at the end of the first move is in the same place it is in at the beginning of the second move.

- The lead's right hand at the end of the first move is in the same place it is in at the beginning of the second move.
- The lead and follow are in the same place in relation to each other at the end of the first move as the beginning of the second move.
- The last steps of the lead and the follow in the first move are on opposite feet as their first steps in the second move.

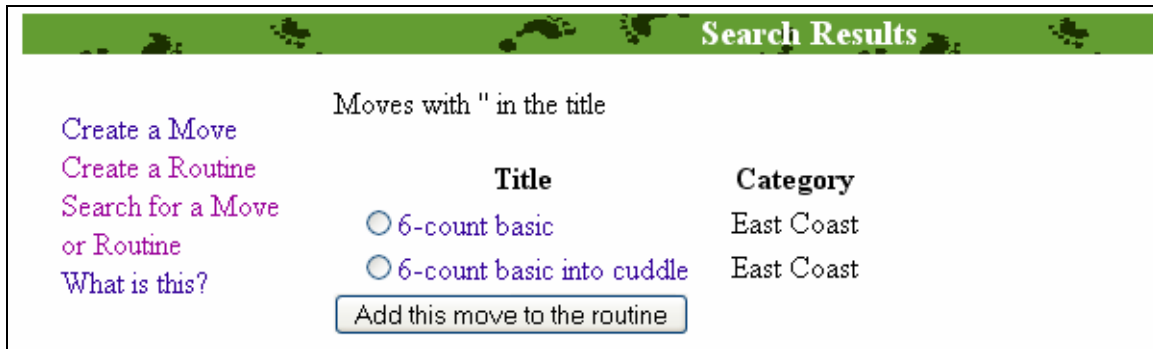


Image 9: For the second move, the search is limited to the moves that can follow the first move.

If the move is a solo, the three middle criteria are not looked at and only the lead's first and last step and the solo value are considered. The following is the part of the Perl script that builds the search query:

```
#build the search query

$selectq = "select moves.move_title, categories.cat_name,
           moves.move_num";
$selectq = "$selectq from moves, categories";
$selectq = "$selectq where categories.cat_num = moves.category";
$selectq = "$selectq and moves.move_title like '%$move%'";

#match ending position.
#if solo is null, this is the first move and there isn't a move to
  match it to
if($solo =~ /\d/){
  $selectq = "$selectq and moves.solo = $solo and
moves.start_weight_l != $end_lead";
  if($solo == 0){ not a solo
    $selectq= "$selectq and moves.start_weight_f != $end_follow
and";
    $selectq= "$selectq moves.start_relation = '$end_relation'
and moves.start_left_hand";
    $selectq= "$selectq = '$end_left'";
    if(!($end_right=~/NULL/i)){
      $selectq= "$selectq and moves.start_right_hand =
'$end_right'";
    }
  }if
}if
$selectq = "$selectq;"; end it with a semi colon
```

In order to make this work, the ending position and solo value of the most recent move added are stored in hidden form fields and passed to the script. The entire routine that has been entered so far is also stored in hidden form fields so that it can be passed along as the user moves between the two different forms. This includes the background information.

For the user to be able to either add more moves or submit the routine, the page with the move list has two forms. The first ends with the “Look up move” button. The same information that is in that form is repeated in hidden form fields in the second form, which ends with the “Save this Routine” button.

The only problem with having two forms in the page is that users cannot make changes to the routine and immediately save it. They have to add a move in order for their changes to be entered into the “save” form. No one has commented on this requirement during user testing.

Create a Routine

[Create a Move](#)
[Create a Routine](#)
[Search for a Move or Routine](#)
[What is this?](#)

A routine is a series of moves. If you have a move in your routine that isn't in the database, please add it before creating your routine. All routines must have unique titles.

Routine Title

Lineage

User: Your name

Teacher: This routine was taught by this Swing dancer (Optional)

Dancer: This Swing dancer has credit for first doing this routine (Optional)

Notes

Anything else?

Routine

Measure	Move Name
1	<input type="text"/> <input type="button" value="Look up move"/>

Look up the move to add it to the routine. Once all your moves are added, click 'Save this Routine'

Image 10: The entire Create a Routine form

When users view a routine, they are presented with a list of the titles of the moves in the routine [Image 11]. Each move is hyperlinked. When the users click on a move title, the list of moves opens up and the details of the requested move are presented [Images 12 and 13].

Routine Test

Moves

Create a Move
 Create a Routine
 Search for a Move
 or Routine
 What is this?

1. 8-Count Basic with Follow Double Swivel
2. 8-Count Basic
3. jump two high
4. 8-Count Basic

Other information

Entered by Amy Dewey

Image 11: A sample routine as it first appears

Routine Test

Moves

Create a Move
 Create a Routine
 Search for a Move
 or Routine
 What is this?

1. 8-Count Basic with Follow Double Swivel

1-2	Rock-step back on left	Swivel step right-left
	Triple-step on left, right arm on	
3&4	follow.s waist, walking clockwise to the other side of her	Triple-step on right
	Step left, step right, going	
5-6	clockwise to the original position, let go of follow.s waist on 5	5&6, triple step on left
7&8	Triple-step on right	7-8, Swivel step right-left
2. 8-Count Basic
3. jump two high
4. 8-Count Basic

Other information

Entered by Amy Dewey

Image 12: The same routine after the user has clicked the title of the first measure

Routine Test

Create a Move

Create a Routine

Search for a Move
or Routine

What is this?

Moves

1. 8-Count Basic with Follow Double Swivel

1-2 Rock-step back on left Swivel step right-left

Triple-step on left, right arm on

follow.s waist, walking clockwise Triple-step on right

to the other side of her

Step left, step right, going

5-6 clockwise to the original position, 5&6, triple step on left

let go of follow.s waist on 5

7&8 Triple-step on right 7-8, Swivel step right-left
2. 8-Count Basic
3. jump two high

1-2 crouch low and jump crouch low and jump

3-4 land and sit land and sit
4. 8-Count Basic

Other information

Entered by Amy Dewey

Image 13: The same routine after the user has clicked the title of the third measure and the first measure

Users can view the details of any number of moves in the routine and can close up the details as well. The moves are presented in an HTML ordered list, and the details of the moves are presented in an ordered list that is nested within the outer list's list items. There is JavaScript code that changes the styles to open and close the move details.

Because the JavaScript code changes the styles of the inner lists, the list contents are still in the page source -- whether or not the list is open. Users with screen readers and other devices that read the page source can still access the contents of the routine.

Searching

The last form in this web site allows users to search the moves and routines that exist in the database. It searches moves and routines separately based on the value in a pair of radio buttons. Users can leave the entire form blank and get a listing of all the moves in the database. They can also enter a dancer's name, up to six different categories, or a title. Additionally, they can choose if they want to search for a routine instead of a move. If they include more than one piece of information, the search criteria are put together with "and" statements, making the results narrower. There are "if" statements in the code to make sure pieces left blank do not limit the search. Here is the part of the Perl script that creates the routine search query:

```

$selectq = "select routines.rout_title, routines.rout_num";
$selectq = "$selectq from routines";

if($title =~ /.+/{ if title matches one character or more
    $titleq = "routines.rout_title like '%$title%'";
}title

if($dancer =~ /.+/{
    $dancerq = "dancers.dancer_name like '%$dancer%'";
    $dancerq = "$dancerq and (dancers.dancer_num = routines.teacher
        or ";
    $dancerq = "$dancerq dancers.dancer_num = routines.choreographer
        or ";
    $dancerq = "$dancerq dancers.dancer_num = routines.user)";
    if($title =~ /.+/{
        $selectq = "$selectq, dancers where $titleq and $dancerq";
    }else{
        $selectq = "$selectq, dancers where $dancerq";
    }
}elsif($title =~ /.+/{
    $selectq = "$selectq where $titleq";
}

```

Originally, I was going to include categories for the routines, but then I remembered doing routines that mixed moves from all different categories. Considering that the different categories are mostly compatible and routines are likely to link moves from any number of categories, I decided against assigning routines categories. Users can still enter a category when searching for routines, but it won't affect the search results, and there is text on the page stating this [Image 14].

The image shows a web search form with a green header bar containing the word "Search". Below the header, there are several links on the left: "Create a Move", "Create a Routine", "Search for a Move or Routine", and "What is this?". The main content area is titled "Search by Title, Dancer or Category". It contains the following elements:

- A paragraph of instructions: "All fields are optional. You'll get a listing of every move in the database if you click 'Search' without entering any information. Select the radio button at the bottom of the page to get all the routines."
- A "Title" section with a text input field.
- A "Category" section with five checkboxes: "East Coast", "Lindy Hop", "Charleston", "West Coast", and "Balboa", followed by an "Other" checkbox and a text input field.
- A "Dancer (Database user, famous dancer, teacher...)" section with a text input field.
- A section titled "I am looking for:" with two radio buttons: "moves" (which is selected) and "routines".
- A paragraph of instructions: "Searching for routines disables the category search. Routines can have moves from any number of different categories and so they are not categorized."
- Two buttons at the bottom: "Clear Form" and "Search".

Image 14: The search form

The Back End

I used a mySQL database to store all the data. I started with the following tables:

- Moves
- Routines
- Dancers
- Categories
- Counts

The moves table initially had space in it for eight counts. It would store eight count ID numbers to connect to the counts table, where the count number, lead details and follow details would be stored. Later, I took the counts out of the moves table by adding two columns, one for the move ID and one for the sequence number, to the counts table. The dancers would be defined in the moves table as dancer, teacher and user, though the three slots would hold ID numbers from the dancers table. The moves table would hold the ID numbers matching the appropriate category in the category table. It also would hold the beginning and ending positions.

I made the column data types what immediately made sense to me. The solo value and which foot the dancers' weight is on were Boolean values. The total number of counts was a tinyint. The category was a tinyint as well, since there's a separate table to hold the category names, which are finite. I used varchars for the positions—lead's right hand and left hand and the spatial relationship between the dancers. Because these are defined with radio buttons, I was able to define exactly what those values would be every time, making searches exact. In retrospect, I should have made the values tinyints. This would have made comparisons and searches more efficient.

Similar to the eight count columns in the move table, the routine table originally had space in it to hold the ID numbers of all the moves in the routine. This would require a limit on how many moves could be in a routine. It also had a disadvantage: each routine took up a constant amount of memory even when the routines weren't using all of it. I created a new table called rout_moves to solve this problem. Rout_moves has three columns: one holds the routine ID number, one holds the move ID number, and the last holds the move's placement in the sequence of moves making the routine. This way, there is no limit to the number of moves in each routine, and each routine only takes up the amount of memory it needs.

The dancers table has two columns: ID number and name. The categories table has two columns: ID number and category name. The counts table has the ID, the count, the lead details and the follow details.

In trying to better handle moves with the same name, I added a versions table to keep track of different moves entered with the same name. This table would allow users to enter whatever name they deem most appropriate for their moves, regardless of what else is in the database. I decided this wasn't the best solution, because a user creating a routine with a move that had several variations in the database would have to go through all of

them to determine which move is the right one. I thought it would be much more intuitive if moves all had unique names when it came time to linking the moves into a routine. For this reason, I removed the versions table.

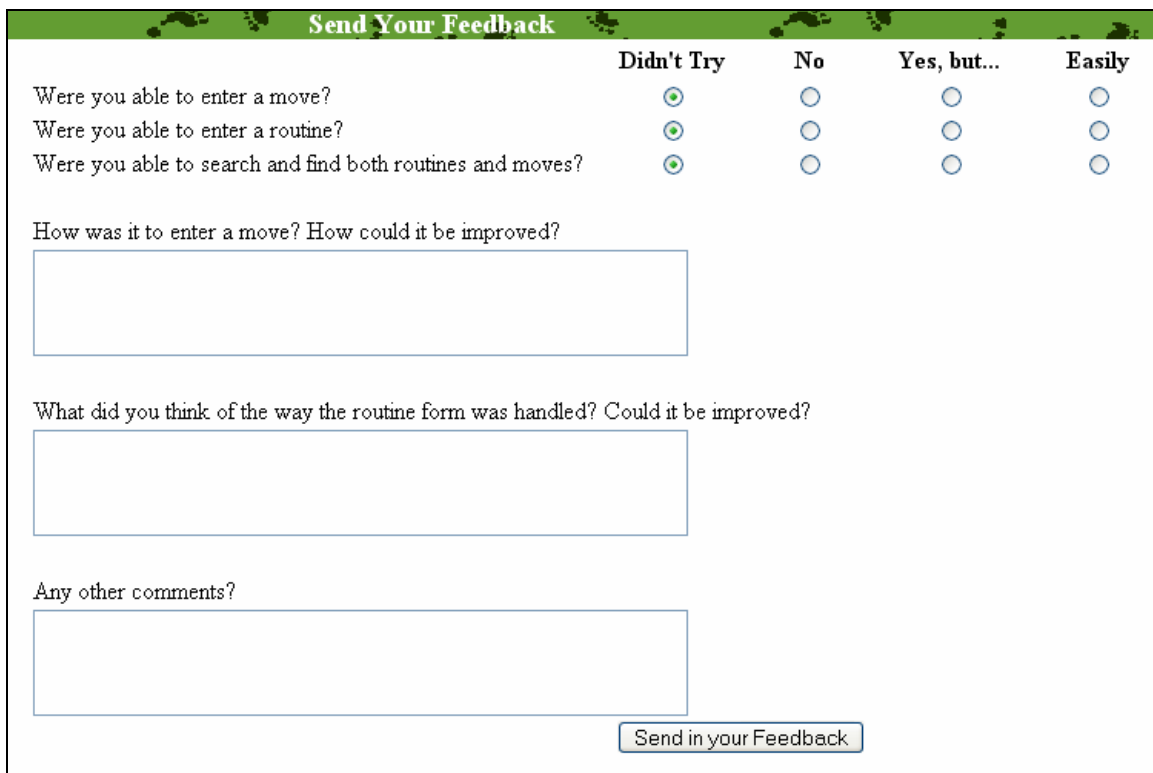
The final list of tables in the mySQL database follows:

- Moves
- Routines
- Dancers
- Categories
- Counts
- Rout_moves

The appendix has the complete SQL table schema: the table names, the column names, the data types and a sample row from each table.

Results and Discussion

As a finishing touch, I wrote a fourth form and another script just for feedback [Image 15]. After completing the project, I removed the form from the site. The form first asks users if they were able to enter a move, enter a routine and search for either one. It then asks for comments about how these functions worked and comments on anything else about the site. The results were saved on the server.



The image shows a feedback form with a green header bar containing the text "Send Your Feedback". Below the header, there are three rows of radio button options. The first row has four options: "Didn't Try", "No", "Yes, but...", and "Easily". The second row has four options: "Didn't Try", "No", "Yes, but...", and "Easily". The third row has four options: "Didn't Try", "No", "Yes, but...", and "Easily". Below the radio buttons, there are three text input fields. The first field is labeled "How was it to enter a move? How could it be improved?". The second field is labeled "What did you think of the way the routine form was handled? Could it be improved?". The third field is labeled "Any other comments?". At the bottom right of the form, there is a button labeled "Send in your Feedback".

	Didn't Try	No	Yes, but...	Easily
Were you able to enter a move?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Were you able to enter a routine?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Were you able to search and find both routines and moves?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

How was it to enter a move? How could it be improved?

What did you think of the way the routine form was handled? Could it be improved?

Any other comments?

Send in your Feedback

Image 15: The feedback form

I advertised the site via a swing dance email list serve and three different online swing communities. I told individual people, mostly swing dancers, about it and gave out the URL. A month and a half later, one user versed in swing besides myself had entered two moves. Some users entered feedback regarding usability. Other users looked at the moves in the database and gave feedback on the accuracy of the moves. One gave feedback on the user interface.

I made simple design changes in response to some criticism. I changed the placement of labels and form fields in response to a comment from a user using a small screen. I wrote a "What is this" page to answer some of the criticism of the moves that are in the database and choices that I made while creating the database. I added a paragraph of directions to the search page when a user asked for functionality that already exists.

A lot of the feedback I received in my user study was about the moves in the database being wrong. For example, one user said that the follow shouldn't do a rock-step on counts 1-2, that the follow isn't following correctly in that case. I can point to old dance footage, interviews with famous swing dancers, and essays about swing dance to back up the steps that I put in the database and that other users put in. I can point to the same exact resources to back up the ways my critics think swing should be danced. I suspect that users will either decide that what is in the database is wrong and, therefore, not use it or decide that it is right and possibly use it.

After receiving feedback that users wanted me to correct the moves they'd entered, I split the move form up. Now, when a user enters a move, the move is displayed with a message asking for confirmation that the move is correct. Users can click the final submit button on the page to enter the move, or they can click the browser's "back" button to edit the move.

Future Work

The biggest piece of future work would be to include some kind of visual representation of the moves. From the feedback I received, that's what users want, and it makes sense to me. I would love to find a very intuitive way for users to move shapes around in some kind of GUI -- similar to creating Flash animation -- that would accurately and easily display the moves and be stored in the database. Users could click a play button to see their moves performed. The routines would work the same way I have them now -- they'd be a string of moves, and users could look at individual moves or the entire routine performed by animated stick figures.

As an intermediate step, including links to digital videos would be great. There would ideally be three videos for partnered moves: the lead, the follow, and the two dancers together. The videos could be stored on YouTube. These videos would not replace the written description.

Any readers interested in building on this code can contact me at amye@alumnae.mills.edu.

Conclusion

Swing is a dance dating back to the late 1920s. It has developed into many different dances, all under the umbrella name of swing. Starting in the early 1990s, there has been a resurgence of swing. An increasing number of people are interested in it, are dancing it, and are writing books about its history. As a folk art, swing has grown and changed as it has been taught, re-taught and danced in new places with new influences. It has developed into many different dances, sometimes with the exact same names. Dancers from different strains of swing do not understand these differences and try to “correct” each other.

In order to facilitate understanding and create a large swing community that includes all the different kinds of swing, I have built the framework for an online database of swing dance moves and routines. This database allows users to enter moves and routines, associate them with styles of swing, and create a lineage of moves danced by the same people. Users can search for moves and routines entered by other users based on name, category or dancer.

Bibliography and Other References

Digital Performance Archive. 10 November 2004. Original location:
<http://dpa.ntu.ac.uk/dpa_site/>. Retrieved 15 March 2007, from
<http://web.archive.org/web/20020802205715/dpa.ntu.ac.uk/dpa_site/>.

Faulkner, Craig. *Swing Lessons*. Ashland, OR: 1997-98.

Forta, Ben. Teach Yourself SQL in Ten Minutes. Indianapolis, IN: Sams Publishing, 2004.

Greenspun, Philip. Philip and Alex's Guide to Web Publishing. San Francisco, CA: Morgan Kaufmann, 1999.

Gupta, Chetna. "A Web based solution for cataloguing, organizing and retrieving image data using Java" (master's thesis, Mills College), 2002.

Kinkead, Mary Ann. Elementary Labanotation and Intermediate Labanotation. Oakland, CA: n.p., 1995.

The Lindy Café. 9 November 2004. Original Location:
http://www.lindycafe.com/_dance/dean.html. Retrieved 15 March 2007, from:
<http://web.archive.org/web/20040411203923/www.lindycafe.com/main.shtml>.

"3.6.9. Using `AUTO_INCREMENT`". MySQL 5.0 Reference Manual. 28 March 2006
<<http://dev.mysql.com/doc/refman/5.0/en/index.html>>.

"MySQL Basics -- A Helpful MySQL Tutorial." The Analysis and Solutions Company. 28 March 2006 <<http://www.analysisandsolutions.com/code/mybasic.htm>>.

Nelson, Karl. "Expanding Nested Lists." 29 August 2006
<<http://www.karlnelson.net/nestedlists/>>.

"Perl and SQL Databases." PerlGuy's Pages. 22 June 2006
<<http://www.jalix.org/ressources/programmation/perl/~vrac/perl-sql.html>>.

Rockwell, Steve. *Swing Lessons: Dean Collins' Shim Sham*, Spring 2003.

Sebesta, Robert W. Programming the World Wide Web. University of Colorado: Colorado Springs, 2003.

Sorensen, Jasper, Vibeke Winding, Tana Ross. *The Call of the Jitterbug*. Green Room Productions. New York, NY: Filmmakers Library, 1988.

Tarbouriech, Georges. "MySQL and Perl, the marriage of convenience." Linux Focus.org. 2002. 2 April 2006
<<http://www.linuxfocus.org/English/January2002/article226.shtml>>.

W3 Schools. 21 March 2006 <<http://www.w3schools.com/default.asp>>.

Watson, Sonny. Dance History Archives. Sonny Watson's Street Swing.com. 14 July 2004 <<http://www.streetswing.com/histmain/d5index.htm>>.

Appendix: Final Database Tables

The following tables are the schema for the SQL database that backs this website. Bolded rows are referencing information from the table listed in the “Reference Table” column.

Moves			
<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>	<u>Reference Table</u>
	int		
move_num	unsigned	1	
move_title	varchar	8-Count Basic with Follow Double Swivel	
user	int unsigned	1	dancers
teacher	int unsigned	6	dancers
dancer	int unsigned	NULL	dancers
counts	tinyint	8	
category	tinyint	3	categories
start_weight_l	bool	1	
start_weight_f	bool	0	
start_left_hand	varchar	right	
start_right_hand	varchar	Not	
start_relation	varchar	facing	
solo	bool	0	
end_right_hand	varchar	Not	
end_left_hand	varchar	Right	
end_relation	varchar	facing	
end_weight_l	bool	0	
end_weight_f	bool	1	

Routines			
<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>	<u>Reference Table</u>
rou_num	int unsigned	3	
rou_title	varchar	Routine Test	
user	int unsigned	1	dancers
teacher	int unsigned	NULL	dancers
choreographer	int unsigned	NULL	dancers
notes	blob		

Dancers		
<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>
dancer_num	int unsigned	1
dancer_name	varchar	Amy Dewey

Categories

<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>
cat_num	tinyint	1
cat_name	Varchar	East Coast

Counts

<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>	<u>Reference Table</u>
move_num	int unsigned	1	moves
count_num	tinyint	1	
beat	varchar	1-2	
lead	varchar	Rock-step back on left	
follow	varchar	Swivel step right-left	

Rout_moves

<u>Column Title</u>	<u>Data Type</u>	<u>Sample Row</u>	<u>Reference Table</u>
rout_num	int unsigned	5	routines
move_num	int unsigned	21	moves
sequence	tinyint	3	