HOW THE DEAD ROCK STAR STAYS ALIVE?

[1] CULTURE ANALYTICS MYSQL

JING YAN

2016/1/13

:: THEUNIQUEEYE@GMAIL.COM

On January 10, 2016, David Bowie, one of the most influential musicians of his era, whose persistent innovations and personal reinventions transformed him into a larger-than-life rock star and cultural icon, passed away.

When rock stars are dead, they are never gone. They keep being memorized by millions of their fans, music lovers, and the public. Their songs went on again and again. And as culture icons, their life stories are discovered, passed on, and exaggerated year by year.

:: QUESTIONS

I am interested in the topic: **how the dead rock star, like David Bowie, stays "alive".** If the users in the Seattle Public Library can be considered as a miniature of society, I'd like to examine the aliveness of David Bowie through analyzing the checkout information of his CDs and related books.

From 2005-2015, how does the times verify of CDs being checked out? Will the results have some patterns related to specific important dates such as birthday? What influence would happen after the musician released a new album? Is there anything interesting that the different checkout duration of books and CDs can reveal? Those **questions** make me curious, and based on them I set up my query.

:: APPROACH

In my view, **the "aliveness" of David Bowie can be examined through two dimension: the quantity and quality.** While the **checkout times** could be regarded as the quantity index, the **checkout duration** could be considered as the quality index.

I select the data of David Bowie's CDs' and related books' checkout times of every day and month and year separately, and each items'(with the same title) checkout duration.

all CDs' bibNumber of David Bowie's

```
SELECT DISTINCT
    bibNumber, title
FROM
    spl3._rawXmlDataCheckIns
WHERE
    callNumber LIKE '%CD 782.42166 B679%'
        AND deweyClass < 789
        AND deweyClass >= 780
```

all related books' bibNumber of David Bowie's

```
SELECT DISTINCT
    bibNumber, title
FROM
    spl3._rawXmlDataCheckIns
WHERE
    callnumber LIKE '%782.42166 B679%'
        AND SUBSTRING(itemType, 3, 4) =
'bk'
```

Something to explain

:: about call Number "CD 782.42166 B679" and "782.42166 B679"

At first, I tried to collect all the items related to David Bowie through "title" and "subject"(as keyword) with words "David Bowie". However, the result came out to be very incomplete. There are two problems with this method. First, most of his CDs do not has the word "David Bowie" in title. Second, the subject dataset covers only 65% of the information and is very subjective.

After examining the searching results from website of the Seattle Public Library, I found out that the call Number of each item is consisted of three parts.

$$CD \mid 782.42166 \mid B679$$

the type of the media books don't have this part ⎦

the deweyClass number

represent the author (to CD) or main subject (to Book) of this item, in this case B679 stands for David Bowie

The call Numbers of all David Bowie's related items contain the same part which is "782.42166 B679" and CD starts with the type "CD". Moreover, items with the "782.42166 B679" in call numbers are exclusively related to David Bowie. Thus, I use this special condition to select all David Bowie's CDs and related books.

:: about bibNumber

The call number is "unique to single item within the library catalog, and multiple copies of the same book will share the same call number". Meanwhile, "each bibliographic record for which the spl owns a copy is assigned a unique identifier." Thus according to the call number, I select the bibNumber for each of them and use this as condition to pick out items.

:: about deweyClass < 789 AND deweyClass >= 780

This is an additional condition to make sure for a second time that the items' subjects are within the music section which excludes potential unrelated stuffs.

:: about SUBSTRING(itemType, 3, 4)

This is a condition to select David Bowie's related books from all items through defining the type as book.

bibNumber	title
1952906	Aladdin sane
2148651	Best of Bowie
2023159	Bowie at the Beeb the best of the BBC radio sessions 68 72
2362135	Bowie at the Beeb the best of the BBC radio sessions 68 72
3029080	Bowie the biography
2023157	ChangesBowie
2299335	David live
1954184	Diamond dogs
1736211	Eart hl i ng
2627462	Earthling
2128387	Heathen
3061214	Heathen
1959048	Heroes
1950899	Hunky dory
2444123	Lets dance
2515298	Live Santa Monica 72
1976192	Lodger
2086964	Low
1953017	man who sold the world
2867039	next day
3057303	Nothing has changed
1639110	Outside
2363644	Outside
2567243	Pinups
2211550	Reality

| all CDs' bibNumber of David Bowie's

bibNumber	title
1836438	Bowie loving the alien
1965047	Bowiestyle
2277690	complete David Bowie
1823619	rise and fall of Ziggy Stardust and the Spiders from Mars David Bowie
2613601	Bowie a biography
2707908	David Bowie starman
2723037	Any day now David Bowie the London years 1947 1974
2815071	man who sold the world David Bowie and the 1970s
2917442	David Bowie is the subject
2941304	Bowie album by album
3029321	Bowie the biography

| all related books' bibNumber of David Bowie's

David Bowie's CDs checkout times by year/month/day

```
SELECT
    DATE_FORMAT(checkout, '20%y-%m-%d') AS formatted_date,
    COUNT(checkout) AS checkoutTimes
FROM
    spl3._rawXmlDataCheckIns
WHERE
    bibNumber = '1952906'
        OR bibNumber = '2148651'
        OR bibNumber = '2023159'
        OR bibNumber = '2362135'
        OR bibNumber = '3029080'
        OR bibNumber = '2023157'
        OR bibNumber = '2299335'
        OR bibNumber = '1954184'
        OR bibNumber = '1736211'
        OR bibNumber = '2627462'
        OR bibNumber = '2128387'
        OR bibNumber = '3061214'
        OR bibNumber = '1959048'
        OR bibNumber = '1950899'
        OR bibNumber = '2444123'
        OR bibNumber = '2515298'
        OR bibNumber = '1976192'
        OR bibNumber = '2086964'
        OR bibNumber = '1953017'
        OR bibNumber = '2867039'
        OR bibNumber = '3057303'
        OR bibNumber = '1639110'
        OR bibNumber = '2363644'
        OR bibNumber = '2567243'
        OR bibNumber = '2211550'
        OR bibNumber = '3067034'
        OR bibNumber = '2631168'
        OR bibNumber = '1954182'
        OR bibNumber = '1880152'
        OR bibNumber = '2412531'
        OR bibNumber = '1976321'
        OR bibNumber = '2296937'
        OR bibNumber = '1972971'
        OR bibNumber = '2448150'
        OR bibNumber = '2612658'
        OR bibNumber = '1953009'
GROUP BY formatted_date
```

David Bowie's related books' checkout times by year/month/day

```
SELECT
    DATE_FORMAT(checkout, '20%y-%m-%d') AS formatted_date,
    COUNT(checkout) AS checkoutTimes
FROM
    spl3._rawXmlDataCheckIns
WHERE
    bibNumber = '1836438'
        OR bibNumber = '1965047'
        OR bibNumber = '2277690'
        OR bibNumber = '1823619'
        OR bibNumber = '2613601'
        OR bibNumber = '2707908'
        OR bibNumber = '2723037'
        OR bibNumber = '2815071'
        OR bibNumber = '2917442'
        OR bibNumber = '2941304'
        OR bibNumber = '3029321'
GROUP BY formatted_date
```

:: about manually type in the bibNumber

I tried to inner join the two tables and use the bibNumber result acquired from the query code before. However, the searching speed is super slow in that way. And I found manually typing in the bibNumer is way faster in this case.

:: about selection by year/month/day

I use the date format function to switch the checkout date into specific pattern.
DATE_FORMAT(checkout, '20%y-%m-%d') AS formatted_date
And the data is selected through "group by formatted_date". Thus the data result is classified by year-month-day.

formatted_date	checkoutTimes
2005-10-01	2
2005-10-04	1
2005-10-17	1
2005-10-27	1
2005-11-08	2
2005-11-15	2
2005-11-22	2
2005-11-29	1
2005-11-30	8
2005-12-04	3
2005-12-05	3
2005-12-06	1
2005-12-07	2
2005-12-09	3
2005-12-10	2
2005-12-12	10

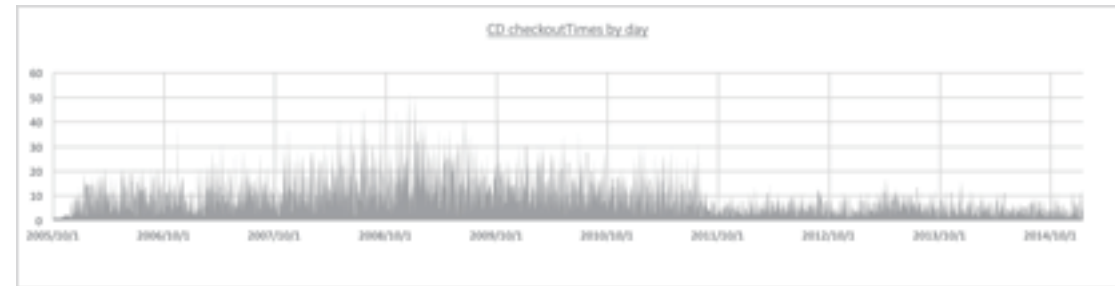| formatted_date	checkoutTimes
2005-10	5
2005-11	15
2005-12	124
2006-01	262
2006-02	254
2006-03	348
2006-04	216
2006-05	273
2006-06	296
2006-07	323
2006-08	286
2006-09	269
2006-10	303
2006-11	295
2006-12	187

| CDs' checkout times classified by day

| CDs' checkout times classified by month

ANALYSIS

As the result numbers are too vague and abstract to capture the pattern and meaning behind them, I transform them into simple charts with Excel model and try to dig out something from it.

What I find from the "CD checkoutTimes by month" chart is that **the peaks and valleys** of each broken line is almost **evenly distributed** from year 2005 to 2011, and the **highest peak** lies in 2008-12. There is a **sudden drop** after 2011-06 and the situation remains in the doldrums. Furthermore, the amount of Checkout times of books and CDs are hugely **different in scale**.
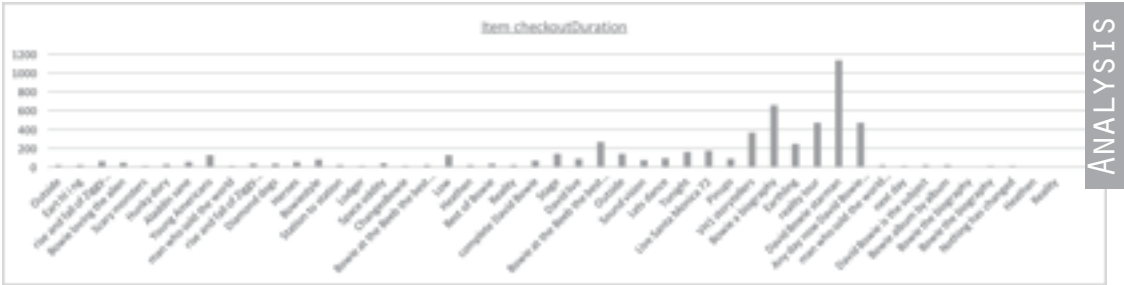

CD checkoutTimes by day

Book checkoutTimes by day

CD checkoutTimes by month

Book checkoutTimes by month

David Bowie's CD + related books' checkout duration

```
SELECT
    bibNumber,
    title,
    AVG(TIMESTAMPDIFF(DAY, checkout, checkin)) AS averageDuration
FROM
    spl3._rawXmlDataCheckIns
WHERE
TIMESTAMPDIFF(DAY, checkout, checkin)<365 ANd
    bibNumber = '1836438'
        OR bibNumber = '1965047'
        OR bibNumber = '2277690'
        OR bibNumber = '1823619'
        OR bibNumber = '2613601'
        OR bibNumber = '2707908'
        OR bibNumber = '2723037'
        OR bibNumber = '2815071'
        OR bibNumber = '2917442'
        OR bibNumber = '2941304'
        OR bibNumber = '3029321'
        OR bibNumber = '1952906'
        OR bibNumber = '2148651'
        OR bibNumber = '2023159'
        OR bibNumber = '2362135'
        OR bibNumber = '3029080'
        OR bibNumber = '2023157'
        OR bibNumber = '2299335'
        OR bibNumber = '1954184'
        OR bibNumber = '1736211'
        OR bibNumber = '2627462'
        OR bibNumber = '2128387'
        OR bibNumber = '3061214'
        OR bibNumber = '1959048'
        OR bibNumber = '1950899'
        OR bibNumber = '2444123'
        OR bibNumber = '2515298'
        OR bibNumber = '1976192'
        OR bibNumber = '2086964'
        OR bibNumber = '1953017'
        OR bibNumber = '2867039'
        OR bibNumber = '3057303'
        OR bibNumber = '1639110'
        OR bibNumber = '2363644'
        OR bibNumber = '2567243'
        OR bibNumber = '2211550'
        OR bibNumber = '3067034'
        OR bibNumber = '2631168'
        OR bibNumber = '1954182'
        OR bibNumber = '1880152'
        OR bibNumber = '2412531'
        OR bibNumber = '1976321'
        OR bibNumber = '2296937'
        OR bibNumber = '1972971'
        OR bibNumber = '2448150'
        OR bibNumber = '2612658'
        OR bibNumber = '1953009'
GROUP BY bibNumber
```

bibNumber	title	averageDuration
1639110	Outside	16.0889
1736211	Eart hl i ng	17.6354
1823619	rise and fall of Ziggy Stardust and the Spiders from Mars David Bowie	56.6471
1836438	Bowie loving the alien	47.3448
1880152	Scary monsters	13.8357
1950899	Hunky dory	28.8568
1952906	Aladdin sane	52.7686
1953009	Young Americans	124.0293
1953017	man who sold the world	13.8979
1954182	rise and fall of Ziggy Stardust and the spiders from Mars	34.6812
1954184	Diamond dogs	35.1224
1959048	Heroes	49.6816
1965047	Bowiestyle	78.0189
1972971	Station to station	16.7232
1976192	Lodger	12.3770

| David Bowie's CD + related books' checkout duration

Item checkoutDuration

:: Analysis + Development

As the data located in a very large scale (every day from 10 years), it's hard to examine the detail of them through a static chart. **An interactive chart that allows you to zoom in** will be much better to examine the changing pattern with complexity.

The **timeline** of all the charts above are **unevenly distributed** due to the limit of excel chart. Moreover, as the **data of CD and Book** are largely different in scale, they can be **hardly compared** through this way. These problems will be solved later in processing.

As two dimensions of David Bowie's "aliveness", the data of **times and duration should be combined together** into one chart. And more detailed analysis should be carried out about the **correlation between the pattern and some particular reasons and dates**.

:: Other Idea about Visualization

Since the main idea of this data query is aliveness, the later visualization could be related to **heart beat graph**.