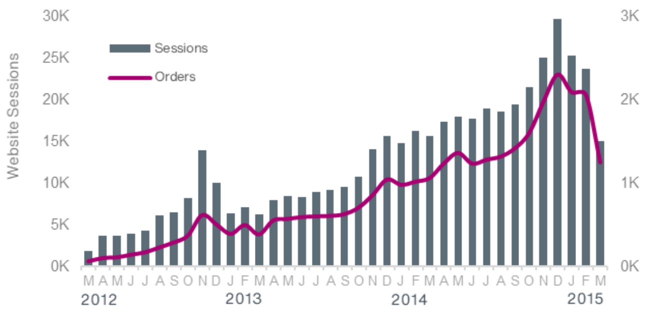
MAVEN

FUZZYFACTORY

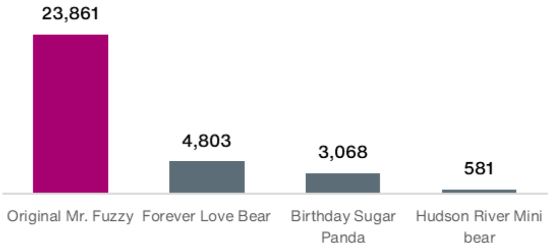
March 19, 2012 to March 19, 2015

Revenue	Sessions	Orders	Conversion Rate
$1.9M	473K	32K	7%

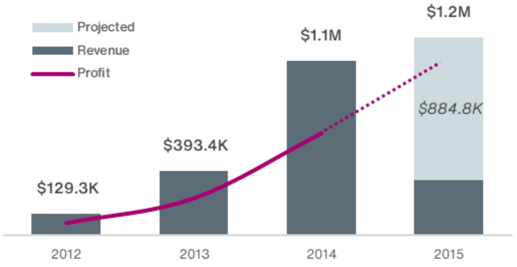
Website sessions and orders have increased each year



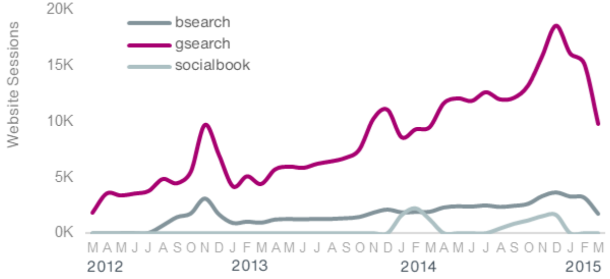
The Original Mr. Fuzzy has generated the most orders



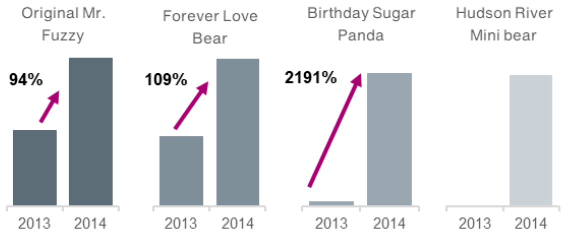
2015 Revenue and Profitability trend based on prior year performance



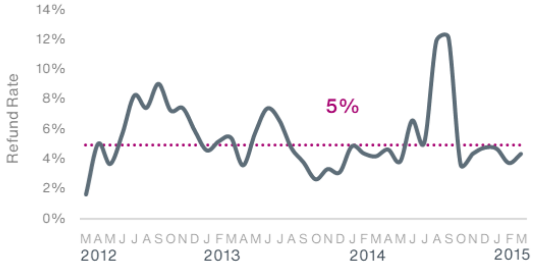
Gsearch is the main driver of paid session traffic



2014 saw extreme order growth for all primary products



The average refund rate is 5%



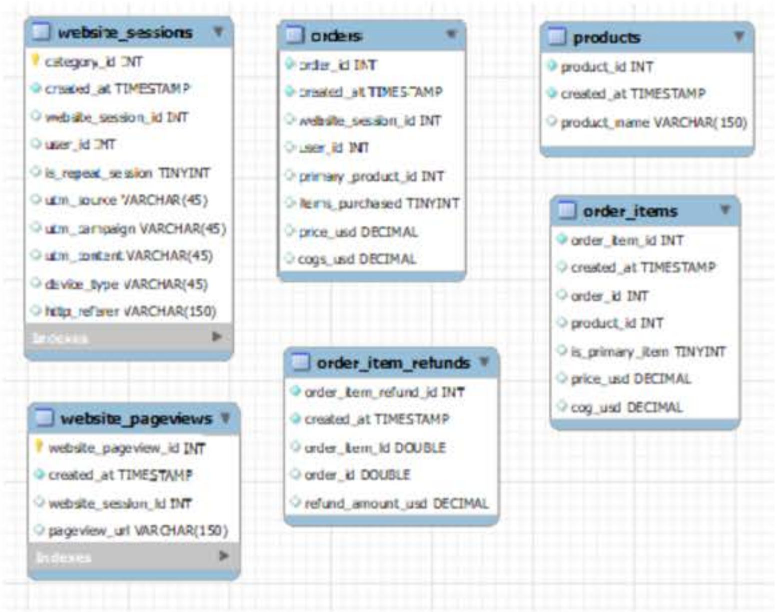
```sql
-- Maven Fuzzy Factory exploratory analysis

use mavenfuzzyfactory;

-- 1. Which tables will help you analyze revenue information? orders
SELECT
    YEAR(orders.created_at) AS Year,
    SUM(price_usd) AS Revenue,
    SUM(cogs_usd) AS Cost,
    SUM(refund_amount_usd) AS Refund,
    SUM(price_usd) - SUM(cogs_usd)  -  SUM(refund_amount_usd) as Profit,
    count(DISTINCT orders.order_id) as Orders
FROM
    orders
    LEFT JOIN order_item_refunds
    on orders.order_id = order_item_refunds.order_id
GROUP BY YEAR(orders.created_at);

-- 2. How many sales has the company made? 32,313
SELECT count(DISTINCT order_id) as TotalSales FROM orders;

-- What are the highest volume primary_product_id values? 1 & 2
SELECT
    primary_product_id,
    product_name,
    COUNT(DISTINCT order_id) AS OrderCount
FROM
    orders
    LEFT JOIN products
    ON orders.primary_product_id = products.product_id
GROUP BY primary_product_id, product_name
ORDER BY ordercount DESC;

-- 3. Order Items table: what is the date range covered?  March 19, 2012 to March 19, 2015
SELECT
    MIN(created_at) as FirstDate,
    MAX(created_at) as LastDate
FROM
    order_items;

-- How many unique products do you see in the table?  4
SELECT DISTINCT
     (product_id)
FROM
    order_items;

-- Are prices fixed or do they vary over time?  Prices are fixed
SELECT DISTINCT
     (price_usd), product_id
FROM
    order_items;

-- Find max and min prices
SELECT
    MAX(price_usd) AS MaxPrice, MIN(price_usd) AS MinPrice
FROM
    order_items;

-- 4. Website_sessions: what does each row represent?  One website visit
-- What are the most popular values for utm_source?  gsearch and organic traffic
SELECT
    utm_source,
    COUNT(DISTINCT website_session_id) AS SessionCount
FROM
    website_sessions
```

```sql
GROUP BY utm_source
ORDER BY COUNT(DISTINCT website_session_id) DESC;

-- What are the most popular values for device_type?  Desktop and Mobile
SELECT DISTINCT
    (device_type), COUNT(website_session_id) as SessionCount
FROM
    website_sessions
GROUP BY device_type
ORDER BY COUNT(website_session_id) DESC;

-- 5: Website_pageviews: what does each row represent?  An instance of a page being loaded/a visit to a page
-- What are the most popular pageview_url values?  /products, /the-original-mr-fuzzy
SELECT
    pageview_url, COUNT(website_pageview_id) as PageViews
FROM
    website_pageviews
GROUP BY pageview_url
ORDER BY COUNT(website_pageview_id) DESC;

-- What is the maximum pageviews listed on one session?  7
SELECT
    website_session_id,
    COUNT(DISTINCT website_pageview_id) AS PagesViewed
FROM
    website_pageviews
GROUP BY website_session_id
ORDER BY COUNT(DISTINCT website_pageview_id) DESC;

-- 9. Pull a monthly trend of a count of website sessions and a monthly trend of total orders sold for the lifetime of the business
SELECT
    left(monthname(website_sessions.created_at),3) AS Month,
    count(DISTINCT website_sessions.website_session_id) AS Sessions,
    count(DISTINCT orders.order_id) as Orders,
    round(count(DISTINCT orders.order_id)/count(DISTINCT website_sessions.website_session_id),2) as ConversionRate
FROM
    website_sessions
        LEFT JOIN orders
    ON website_sessions.website_session_id = orders.website_session_id
GROUP BY year(website_sessions.created_at), month(website_sessions.created_at);

-- 10. Pull a count of orders sliced by primary_product_id for the year 2014
-- Compare that to a count of orders from 2013 and calculate the YoY increase
CREATE TEMPORARY TABLE Temp_Table
SELECT
    primary_product_id,
    product_name,
    year(orders.created_at) as Year,
    count(orders.created_at) as Orders,
    count(case when year(orders.created_at) = 2014 then orders.order_id else null end) as 'Orders_2014',
    count(case when year(orders.created_at) = 2013 then orders.order_id else null end) as 'Orders_2013'

 FROM
    orders
        LEFT JOIN products
    ON orders.primary_product_id = products.product_id
WHERE
    YEAR(orders.created_at) in(2014,2013)
GROUP BY primary_product_id, product_name
order by primary_product_id;

```

```
SELECT
    primary_product_id,
    product_name,
    orders_2014,
    orders_2013,
 round((orders_2014 - orders_2013)/orders_2013*100)  AS 'YoYGrowth%'
FROM
   Temp_Table;

-- 11. Use the website_sessions table to pull a monthly trended count of website_sessions for each of the 3 major paid traffic sources
SELECT
    YEAR(created_at) as Year,
    left(MONTHNAME(created_at),3) as Month,
    utm_source,
    COUNT(DISTINCT website_session_id) AS Sessions
FROM
    website_sessions
WHERE
    utm_source IN ('gsearch' , 'bsearch', 'socialbook')
GROUP BY utm_source, YEAR(created_at), MONTHNAME(created_at)
ORDER BY utm_source, YEAR(created_at), MONTH(created_at), COUNT(DISTINCT website_session_id) DESC;

-- 12. Pull a trend of the order_refund rate by month for the life of the business
SELECT
    YEAR(orders.created_at) AS Year,
    left(MONTHNAME(orders.created_at),3) AS Month,
    COUNT(DISTINCT orders.order_id) AS Orders,
    COUNT(order_item_refunds.order_id) AS Refunds,
    round(COUNT(order_item_refunds.order_id)/COUNT(DISTINCT orders.order_id) * 100,1) as 'RefundRate%'
FROM
    orders
        LEFT JOIN
    order_item_refunds ON orders.order_id = order_item_refunds.order_id
GROUP BY YEAR(orders.created_at), MONTHNAME(orders.created_at)
ORDER BY YEAR(orders.created_at), MONTH(orders.created_at);

-- Project revenue for 2015 based on prior year performance
SELECT
    YEAR(orders.created_at) AS Year,
    month(orders.created_at) as Month,
    SUM(price_usd) AS Revenue

FROM
    orders
GROUP BY YEAR(orders.created_at), Month(orders.created_at);
```

MAVEN
FUZZYFACTORY
EER DIAGRAM

Before



website_sessions
- category_id INT
- created_at TIMESTAMP
- website_session_id INT
- user_id INT
- is_repeat_session TINYINT
- utm_source VARCHAR(45)
- utm_campaign VARCHAR(45)
- utm_content VARCHAR(45)
- device_type VARCHAR(45)
- http_referer VARCHAR(150)

orders
- order_id INT
- created_at TIMESTAMP
- website_session_id INT
- user_id INT
- primary_product_id INT
- items_purchased TINYINT
- price_usd DECIMAL
- cogs_usd DECIMAL

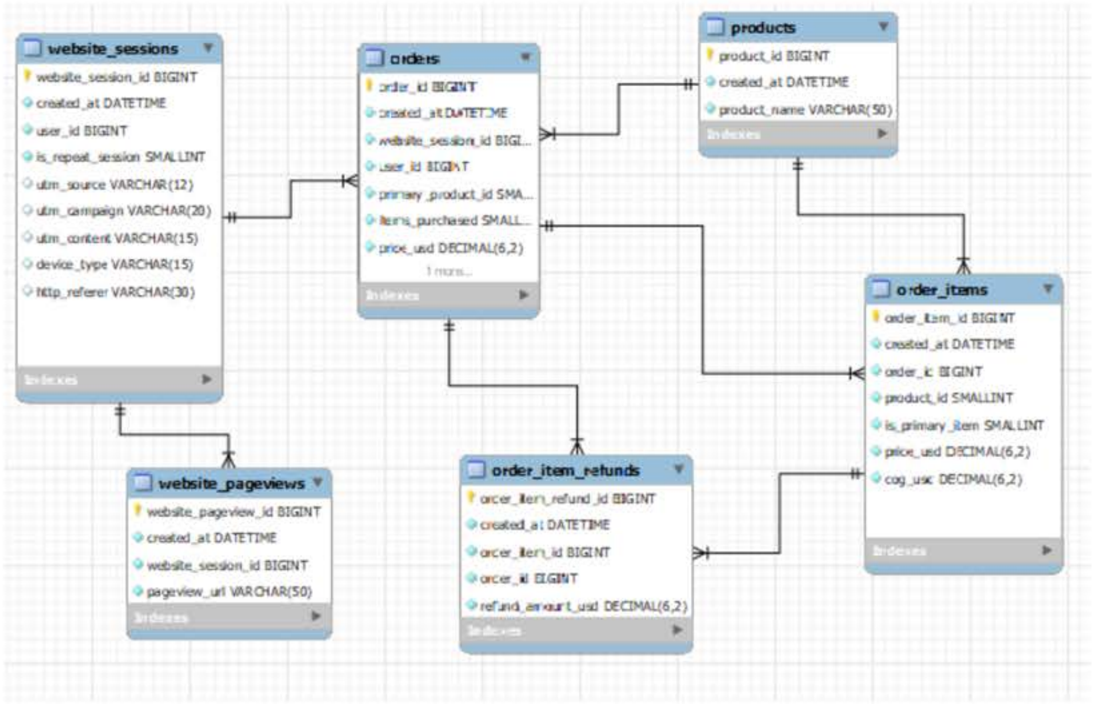
products
- product_id INT
- created_at TIMESTAMP
- product_name VARCHAR(150)

order_items
- order_item_id INT
- created_at TIMESTAMP
- order_id INT
- product_id INT
- is_primary_item TINYINT
- price_usd DECIMAL
- cog_usd DECIMAL

order_item_refunds
- order_item_refund_id INT
- created_at TIMESTAMP
- order_item_id DOUBLE
- order_id DOUBLE
- refund_amount_usd DECIMAL

website_pageviews
- website_pageview_id INT
- created_at TIMESTAMP
- website_session_id INT
- pageview_url VARCHAR(150)

After



website_sessions
- website_session_id BIGINT
- created_at DATETIME
- user_id BIGINT
- is_repeat_session SMALLINT
- utm_source VARCHAR(12)
- utm_campaign VARCHAR(20)
- utm_content VARCHAR(15)
- device_type VARCHAR(15)
- http_referer VARCHAR(30)

orders
- order_id BIGINT
- created_at DATETIME
- website_session_id BIGI...
- user_id BIGINT
- primary_product_id SMA...
- items_purchased SMALL...
- price_usd DECIMAL(6,2)
- 1 more...

products
- product_id BIGINT
- created_at DATETIME
- product_name VARCHAR(50)

order_items
- order_item_id BIGINT
- created_at DATETIME
- order_id BIGINT
- product_id SMALLINT
- is_primary_item SMALLINT
- price_usd DECIMAL(6,2)
- cog_usd DECIMAL(6,2)

website_pageviews
- website_pageview_id BIGINT
- created_at DATETIME
- website_session_id BIGINT
- pageview_url VARCHAR(50)

order_item_refunds
- order_item_refund_id BIGINT
- created_at DATETIME
- order_item_id BIGINT
- order_id BIGINT
- refund_amount_usd DECIMAL(6,2)