

Exploratory Monitoring at Bing

AUTOMATED SYNTHETIC EXPLORATORY MONITORING OF DYNAMIC WEB SITES USING
SELENIUM



Outline

1. Modern Engineering Principles
2. Monitoring Approaches
3. Statistical Models
4. Selenium
5. Exploratory Principles
6. Exploratory Runs

Modern Engineering Principles

Current engineering landscape

- Hundreds of engineers
 - *2,000 engineers, across all continents*
- Ship 2x/day
 - Full build shipped to production, no live site issues!
- Agile
 - {design, dev, test} → ship (no P0 bugs) → repeat
- One source tree
 - Componentization, contracts, modularization
- 19.4% search market share (30% share if Yahoo! is included)

Modern Engineering Principles

Test-Driven Evolution: 10 Principles

1. Automate every test, but don't test everything
2. Run all tests for every single check-in
3. Tests are binary: either they all pass, or they all fail
4. No test selection. Run them all.
5. Embrace the Open-Source!
6. Testing in Production (deploy to production, test in production)
7. Deployment gated by tests: if any test fails, rollback
8. Defensive coding techniques (code + test case for every check-in, small check-ins, code behind flights, etc.)
9. Be truly data driven
10. Live Site remains the King!

Monitoring Approaches

1. Organic Monitoring

- a) Counters and rules
- b) “Number of 404s for endpoint X increases by X%, per Data Center → Alert”

2. Synthetic Simple Monitoring

- a) Keep-Alive Like
- b) Gomez, Keynote, xPing (*Microsoft Internal Tool*)

3. Synthetic Advanced Monitoring

- a) Exploratory Tests

Monitoring Approaches

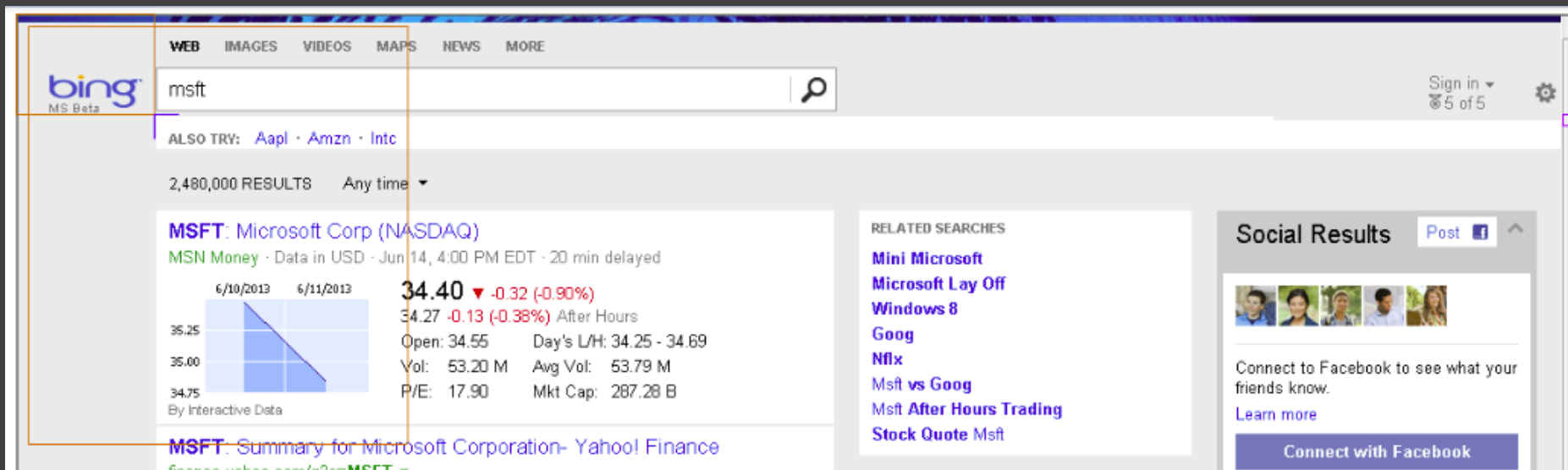
1. Synthetic Advanced Monitoring – Why?

- a) Huge matrix: Browsers x Markets x Devices x Data Centers
- b) Another dimension: experiments (flights)
 - i. Hundreds of flights
 - ii. Collision of flights
- c) Some scenarios are more complex:
 - i. Home Page → Click Image → Search → Cached-Page → Back...

Monitoring Approaches

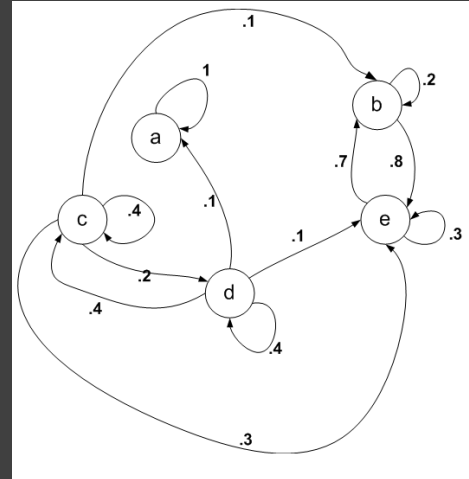
Synthetic Advanced Monitoring – Example

Two flights in collision course, specific browser, back button



Statistical Models

Markov Chains: A **Finite State Machine** where **transitions** from one state to another one are based on **probabilities** and are **only dependent on the current state**

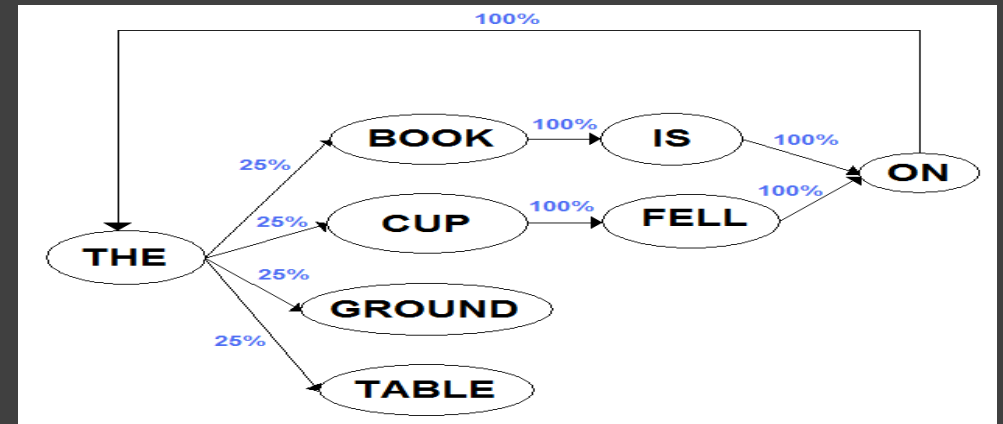


Statistical Models

Markov Chains

Example: automatic text generation

- Step 1: creating the Markov Chains from training data:
 - *"The book is on the table"*
 - *"The cup fell on the ground"*
- Step 2: generation of results by traversing the chains:
 - *"The book is on the ground"* (P = 25%)
 - *"The cup fell on the book is on the cup"* (P = 1.5625%)

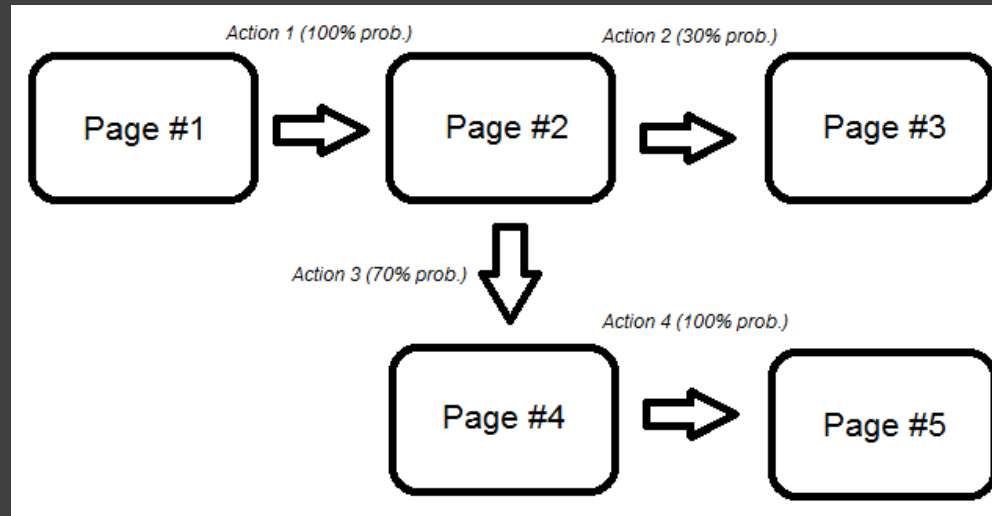


Statistical Models

Markov Chains

We can use Markov Chains to model the user's behavior of a web system:

- **States:** landing pages
- **Actions:** clicks, queries, etc.



Statistical Models

Distribution

Another dimension of usage is the overall distribution of users in different dimensions:

- Browsers Distribution
- Devices Distribution
- Markets Distribution
- Experiments (flights) Distribution

<i>Browsers Distribution</i>	
Browser	Percentage of users
IE7	6%
IE8	8%
IE11	15%
Firefox	9%
Others	62%

<i>Devices Distribution</i>	
Device	Percentage of users
Windows Phone	34%
iPhone	17%
Kindle Fire	17%
Android	9%
Others	23%

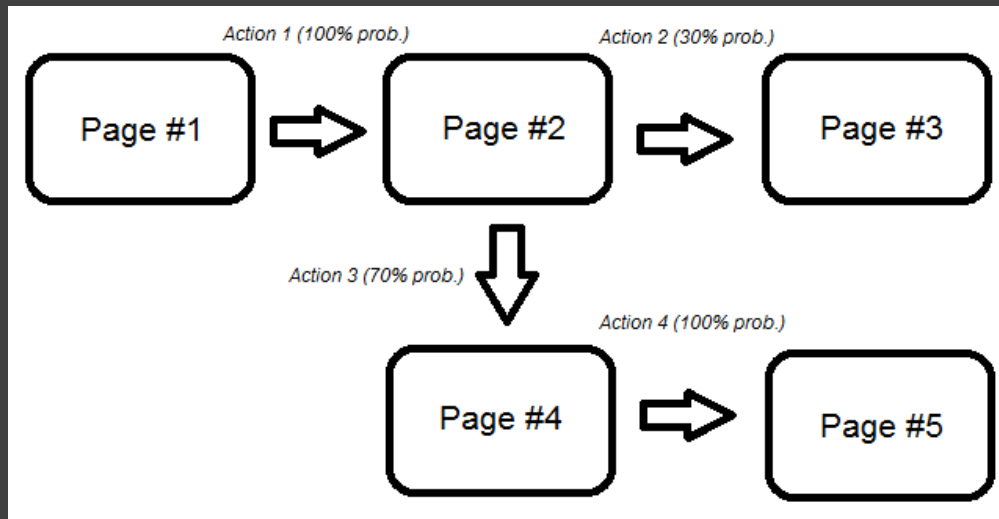
<i>Markets Distribution</i>	
Market	Percentage of users
United States	52%
China	17%
Brazil	4.5%
Canada	7%
Others	19.5%

<i>Experiments Distribution</i>	
Experiment	Percentage of users
Experiment #1: light-blue background color	2%
Experiment #2: larger font size for titles	3%
Experiment #3: larger images	20%
Experiment #4: new relevance ranker	1%

Statistical Models

Markov Chains and Distribution

We'll use these two tools to build the exploratory runs... stay tuned...



Browsers Distribution	
Browser	Percentage of users
IE7	6%
IE8	8%
IE11	15%
Firefox	9%
Others	62%

Devices Distribution	
Device	Percentage of users
Windows Phone	34%
iPhone	17%
Kindle Fire	17%
Android	9%
Others	23%

Markets Distribution	
Market	Percentage of users
United States	52%
China	17%
Brazil	4.5%
Canada	7%
Others	19.5%

Experiments Distribution	
Experiment	Percentage of users
Experiment #1: light-blue background color	2%
Experiment #2: larger font size for titles	3%
Experiment #3: larger images	20%
Experiment #4: new relevance ranker	1%

Selenium

- <http://docs.seleniumhq.org/>
- Open Source
- Becoming the industry standard for browser automation
- Supports for multiple browsers and devices
- Scalability thru Selenium Grid
- At Bing: wrapper around the Grid for Reliability

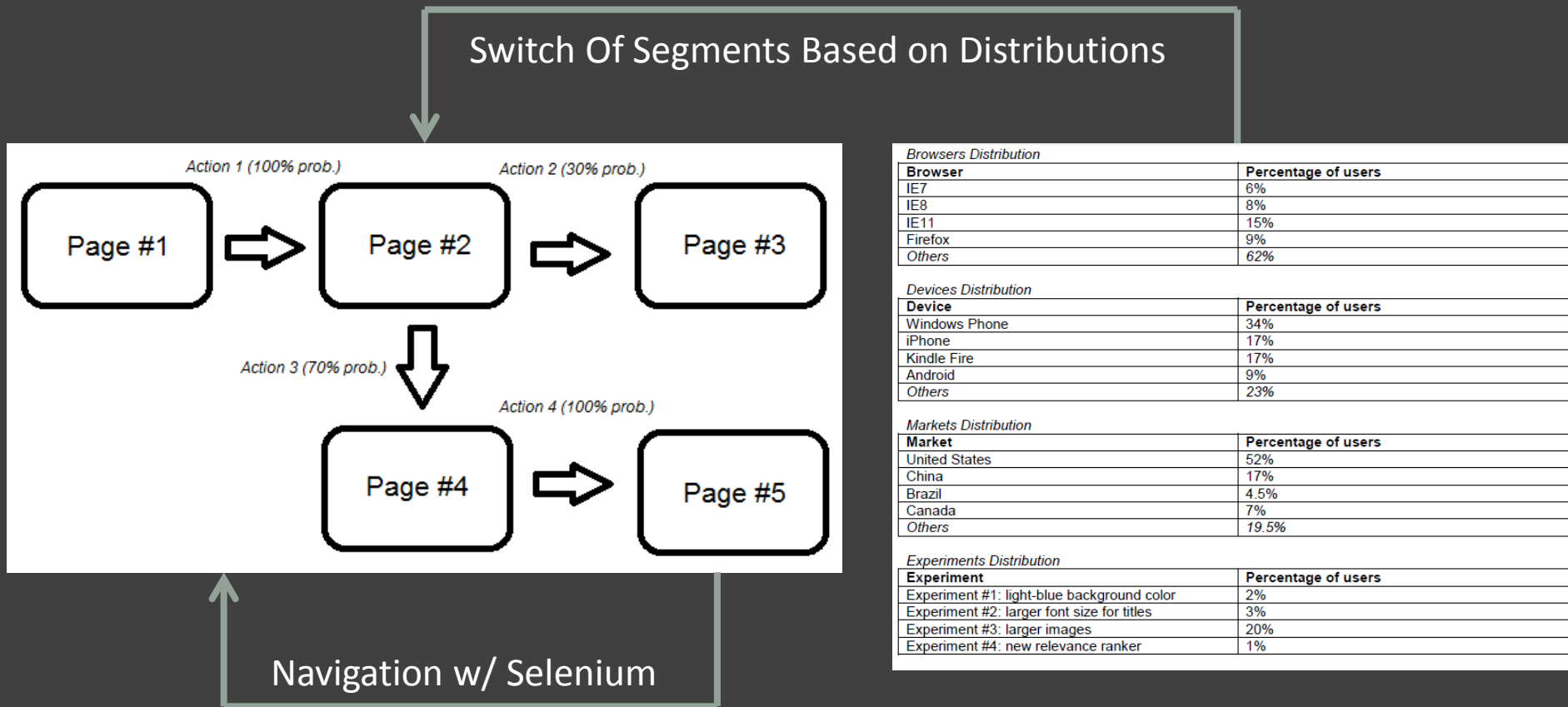


Exploratory Principles

➤ Combining:

- Markov Chains from the Activity Logs (no PII)
- Dimensions Distributions
 - ✓ Browsers
 - ✓ Devices (actual device or user-agent)
 - ✓ Markets
 - ✓ Flights
- Selenium for navigation/checks

Exploratory Principles



Exploratory Principles – Test Models

TRADITIONAL MODEL

```
SampleTestCase() {  
    Pre-ValidationSetup();  
    Validation();  
    Post-ValidationSetup(); //Teardown  
}
```

- *Read/Write*
- *Deterministic*

SUBSCRIPTION-BASED MODEL

```
SubscriptionBasedSampleTestCase() {  
    If(IsRelevantState(this.CurrentState))  
        Validation();  
}
```

- *Read-Only*
- *Opportunistic*

Exploratory Principles – Validation Models

Custom Validation Model

- Specific to a certain state (condition)
- When the condition triggers, the validation takes place
- For Example: Deep-Links
 - *If(Deep-Links) → Validate*



Exploratory Principles – Validation Models

Invariant Validation Model

- Applicable to all states and conditions
- Should always be true, or always be false (*invariant*)
- The more Invariants, the higher the ROI
- Examples:
 - *Links*: no links should lead to 404 pages
 - *Server Error*: no state/action should lead to server errors
 - *Security*: no state/action should expose any security flaw
 - *Overlapping*: no state/action should contain overlapped elements

Exploratory Runs

Putting it all together!

1. Log Mining → Markov Chain
2. Percentage Distribution per Context (Browsers, Devices, ...)
3. Build Custom & Invariant Validation Models
4. Stochastically run thru the Markov Chain using Selenium
 - a) *+Query Set*
5. Sporadically (time-based) switch contexts based on #2
6. At each state and at each action, apply the Validation Models

Exploratory Runs

Key Insights

- ✓ Exploratory is more suitable for post-production (monitoring)
- ✓ The longer it runs, the higher the coverage
- ✓ But critical paths will always be covered!
- ✓ Validation Models have different priorities in case of an alert
- ✓ Need to account for normal entropy of the system
 - ✓ Statistics and Retries

Exploratory Runs

Statistics

- ✓ 10 high-end dedicated servers (running 24/7)
- ✓ 2,000,000 state transitions/day
- ✓ 120 validation methods
 - ✓ 100 custom
 - ✓ 20 invariants
- ✓ Supported browsers: IE7, IE8, IE9, IE10, IE11, Chrome (latest), Firefox (latest)
- ✓ Devices: Windows Phone, iPhone, Android, Surface, iPad, Kindle Fire
- ✓ 100's flights, 100's markets
- ✓ 8-10 alerts per day (most of them known issues)

Exploratory Runs

Example of alert

- Invariant – HTTP 500 (Internal Server Error)
- Due to incompatibility of two flights

The image shows a screenshot of a Bing error page. At the top left is the Bing logo in its characteristic lowercase font.

Bing services aren't available right now

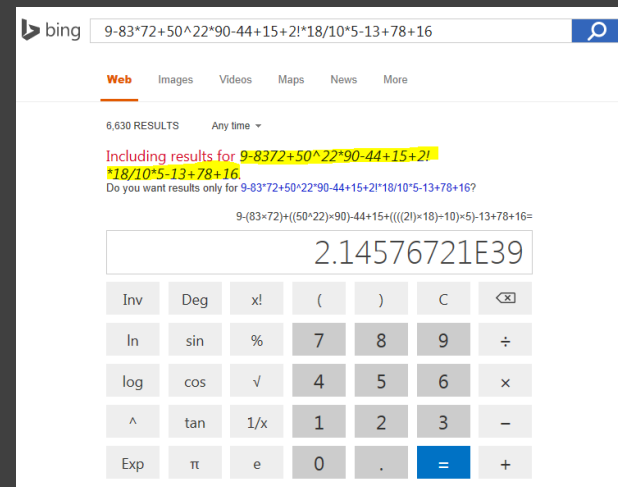
We're working to restore all services as soon as possible.
We know you want to get back to searching. Please check back soon.

Ref A: AF57A3A700B84A81877ED70AAA719EA6 Ref B: 772F8EA31FFD1AA12EAEF4C650B0B993 Ref C: Sat May 31 18:38:41 2014 PST

Exploratory Runs

Example of alert

- Custom – Math Answer
- *If there is a math answer, no alteration should occur!*



bing 9-83*72+50^22*90-44+15+2!*18/10^5-13+78+16

Web Images Videos Maps News More

6,630 RESULTS Any time ▾

Including results for 9-83*72+50^22*90-44+15+2!*18/10^5-13+78+16

Do you want results only for 9-83*72+50^22*90-44+15+2!*18/10^5-13+78+16?

9-(83*72)+((50*22)>90)-44+15+(((2!*18)-10)*5)-13+78+16=

2.14576721E39

Inv	Deg	x!	()	C	<X>
In	sin	%	7	8	9	÷
log	cos	√	4	5	6	×
^	tan	1/x	1	2	3	-
Exp	π	e	0	.	=	+

Exploratory Runs

Example of alert

- Custom – Movies Reviews
- *If there is a movie answer, it should have reviews!*

The Judge (2014)



The Judge is a 2014 American drama film directed by David Dobkin. The film stars Robert Downey, Jr., Robert Duvall, Vera Farmiga, Vincent D'Onofrio, Dax Shepard, Jeremy Strong, Sarah Lancaster, and Billy Bob Thornton. The film was released in the United States on October 10, 2014.

en.wikipedia.org

en.wikipedia.org

Summary: R · 2hr 21min · Drama

Release date: Oct 10, 2014

Director: [David Dobkin](#)

Production company: [Village Roadshow Pictures](#)

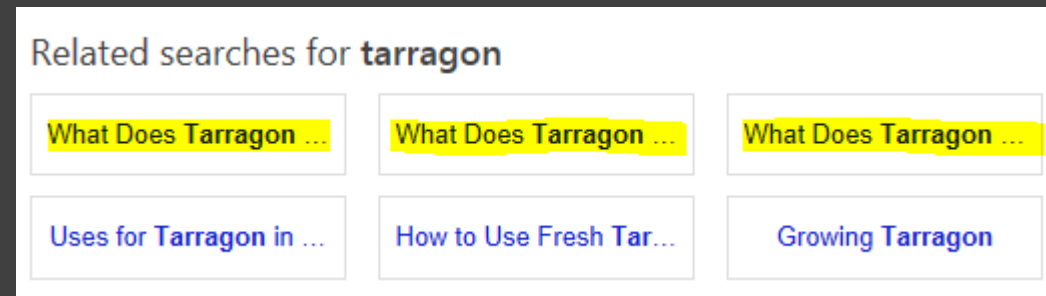
Screenwriters: [Nick Schenk](#) · [David Seidler](#) · [Bill Dubuque](#)

Music by: [Thomas Newman](#)

Exploratory Runs

Example of alert

- Invariant – No duplicates on any module
- *Due to a truncation bug, duplicates in Related Searches on the Bing Kindle Fire Experience*



Questions?
