Measuring and driving DevOps practices in the real world

Measurement and motivation
What gets measured gets done
Ben Mostafa
Engineering Manager &
Head of Developer Relations
MessageMedia

API First Business Messaging
A framework for your program

How does it help DevRel?

Save time and effort
Align your initiatives with your goal and ensure everything is driving you towards your vision.

Maintain focus
By measuring what you are doing, you’ll be able to maintain focus on what is most important in your program.

Get support and buy-in
Demonstrate a structured approach to assist with business buy in and support.
## Creating a DevOps dashboard

<table>
<thead>
<tr>
<th></th>
<th>Team 1</th>
<th>Team 2</th>
<th>Team 3</th>
<th>Team 4</th>
<th>Team 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployments</td>
<td>Green</td>
<td>Green</td>
<td>Green</td>
<td>Red</td>
<td>Yellow</td>
</tr>
<tr>
<td>Change Lead Time</td>
<td>Green</td>
<td>Green</td>
<td>Red</td>
<td>Red</td>
<td>Yellow</td>
</tr>
<tr>
<td>Codebase Health</td>
<td>Red</td>
<td>Yellow</td>
<td>Etc</td>
<td>etc</td>
<td>Etc</td>
</tr>
<tr>
<td>Services Health</td>
<td>Green</td>
<td>Yellow</td>
<td>etc</td>
<td>Etc</td>
<td>etc</td>
</tr>
</tbody>
</table>
In development!!
Deployments What to measure?

1. Deployments per engineer
2. Deployments duration
3. Time of day
4. Deployment success / failure
Change Lead Time

What to measure?

1. Time spent in each step
2. Wait time between each step
3. Pull request metrics – time to close, reviewers / approvals, comments and size
4. Overall change lead time
Change Lead Time

Visualization

Average Change Lead Time

5h

Average Wait Time

5h

Average PR Time to Close

5h
Wait Time vs % Resource Busy

Wait Time = (% Busy) / (% Idle)
Codebase Health What to measure?

1. Age of dependencies / libraries
2. Age of vulnerabilities / libraries with vulnerabilities
3. Overall health rating
Change Lead Time Visualisations

Average Dependency Age
3m

Overall Codebase Health
B

Known vulnerabilities
10

Library Age

<table>
<thead>
<tr>
<th>Library</th>
<th>Repository 1</th>
<th>Repository 2</th>
<th>Repository 3</th>
<th>Repository 4</th>
<th>Repository 5</th>
<th>Repository 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library 4</td>
<td>A</td>
<td></td>
<td></td>
<td>D</td>
<td>F</td>
<td>A</td>
</tr>
<tr>
<td>Library 3</td>
<td></td>
<td>B</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Library 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Library 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Service Health  What to measure?

1. Data stores and communication channels
2. Deployability
3. Alignment with architectural principals
4. Time since last deployment
## Service Health Score Card

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is HTTPS used for all HTTP communication?</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Is data encrypted at rest?</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Does this service handle more than one concern?</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Is this service stateless?</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Can this service be deployed without downtime / service interruption?</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>Can this service be scaled horizontally?</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Is the data decoupled from the applications?</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>How long since the service was last deployed to production?</td>
<td>-0.1</td>
<td>-2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>53</strong></td>
</tr>
</tbody>
</table>

### Overall Service Health

The overall service health score is **53**, which corresponds to a **C** rating.

<table>
<thead>
<tr>
<th>Score</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>75 - 85</td>
<td>A</td>
</tr>
<tr>
<td>60 - 75</td>
<td>B</td>
</tr>
<tr>
<td>50 - 60</td>
<td>C</td>
</tr>
<tr>
<td>35 - 50</td>
<td>D</td>
</tr>
</tbody>
</table>
What else to measure?

• Measuring the age and vulnerabilities in 3rd party software
• Measure platform utilisation
• Measure operational alerts per team or service
Recap

Measure Deployments
- Deployments per engineer
- Deployments duration
- Time of day
- Success / failure

Measure change lead time
- Time spent in each step
- Wait time
- Pull request metrics
- Overall change lead time

Codebase health
- Age of dependencies
- Age of vulnerabilities
- Overall health rating

Service health
- Data stores and communication channels
- Deployability
- Alignment with architectural principals
- Time since last deployment
Stay in touch!

linkedin.com/in/ben-mostafa

developers@messagemedia.com.au

developers.messagemedia.com/collaborate/slack