My First Year at Chef: Measuring All the Things

Nicole Forsgren, PhD
Director of Organizational Performance & Analytics, Chef
Who? Why? Huh?
Who? Why? Huh?

Basically,

**I rub science on things**

to make your Dev, ITOps and DevOps better with metrics.

@nicolefv
What Did I Do? And How Can You Do It Too?

aka The Talk Outline

• Where do I start?
• What numbers should I think about?
• What about benchmarking?
• What else should I think about?
What Did I Do? And How Can You Do It Too?

aka The Talk Outline

• Where do I start?
• What numbers should I think about?
• What about benchmarking?
• What else should I think about?
Where Do I Start?

Talk about it!

- Start at the **highest level**.
  - Is measurement happening?
  - Is metrics a thing?
Where Do I Start?

Talk about it!

- Start at the highest level.
- Is measurement happening?
- Is metrics a thing?
- But all we do is fight! Er, talk in circles!
- Metrics provide opportunity for communication and alignment

@nicolefv
Where Do I Start?

**Talk about it!**

- Start at the highest level.
  - Is measurement happening?
  - Is metrics a thing?
- But all we do is fight! Er, talk in circles!
  - Metrics provide opportunity for communication and alignment
- Uh… still not sure?
  - Ask around! **What is important to people?**
  - This can be done at **ANY** level of an organization
At Chef:

Lay some groundwork

- [Data Driven](https://oreilly.com), by @dpatil and @hmason
  - Free, and only 20-ish pages!
At Chef:

Lay some groundwork

- Data Driven, by @dpatil and @hmason
  - Free, and only 20-ish pages!

- Established Chef Scorecard 2015
  - Important milestones for every area of the business
  - KEY! It must fit on one page. No cheating on font size.
  - This can include things that are difficult or not currently measured
Chef Scorecard 2015 (scrubbed)

Ecosystem, Training, Community

• “When I’ve got a problem, do I know to go to Chef” – awareness of what’s in Chef right now
• Community health index – reach (including international), depth
• Internal: Internal chef engineering
• International infrastructure (legal)
• Open source footprint: % of market learning and using Chef, % of market with Chef on resume
• How effective is Learn Chef?
• # people who have been through Chef training
Chef Scorecard 2015 (scrubbed)

Financial
- Exit ARR: absolute, incremental by market (US, EMEA)
- Product bookings: new (including enterprise logos), expansion, retention / churn
- Gross Margin by product, service
- Services Revenue
- Exceed cash plan
- Control expenses

Marketing
- Conversions: Leads -> Opportunities, Leads -> Wins
- Clear Category Leader: DevOps, Automation, Compliance at Velocity
- Strong and increasing external buzz on Chef - enterprise vs startup
- Consistency on how we present our products and our customer
Chef Scorecard 2015 (scrubbed)

People
• Chef is an innovative and industry-leading work environment
• Chef attracts and retains top talent and uses our people as a competitive advantage
• Diversity
• HR material is localized where appropriate

Sales & Business Development
• Progress on solution road map: capability by platform
• Pipeline Growth: Size of opportunity pipeline (# adds by deal size) for next 12m m-o-m
• Time to close (velocity) – broken down by stage
• New rep ramp-up time
• Opportunities -> Wins

@nicolefv
Chef Scorecard 2015 (scrubbed)

Product & Engineering

- Increase commercial adoption of premium features
- Software developer life cycle - hiring, on-boarding, career progression, including diversity goals
- Ship every day
- Rebuild and reboot Hosted Chef
- Customer satisfaction and ease of use aka "ease of getting going"
- Deliver against 2015 road map
- Become industry reference for development of enterprise software at velocity
- Low touch conversion (download / OSS -> premium: time to value)
For each metric:

- Define
- Set target
- Measure – periodically
- Communicate

<table>
<thead>
<tr>
<th>Metric</th>
<th>Baseline</th>
<th>Current</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Your metrics aren’t set in stone.

Start with an MVP, then iterate and improve.

Toss what doesn’t work.
What Did I Do? And How Can You Do It Too?

aka The Talk Outline

• Where do I start?
• **What numbers should I think about?**
  • What about benchmarking?
  • What else should I think about?
How to think about your world

What kinds of numbers should we be thinking about?

1. **External**: outward-facing, customer-focused
2. **Internal**: inward-facing, process improvement
3. **Cultural**: because people matter
How to think about your world

What kinds of numbers should we be thinking about?

1. **External**: outward-facing, customer-focused
2. **Internal**: inward-facing, process improvement
3. **Cultural**: because people matter

**EXTRA CREDIT**
Also think about:
- Objective v. Subjective
- Leading v. Lagging
Where should I start?

Start with your IMPORTANT THINGS to my organization / team list.
Where should I start?

Start with your IMPORTANT THINGS to my organization / team list.

Could include things like:
- Maintain excellent customer satisfaction
- Increase speed of software delivery
- Hit revenue targets
- Increase software quality or usefulness
Where should I start?

Start with your IMPORTANT THINGS to my organization / team list.

Could include things like:
• Maintain excellent customer satisfaction
• Increase speed of software delivery
• Hit revenue targets

How does this apply to me?
Where should I start?

Start with your IMPORTANT THINGS to my organization / team list.

Could include things like:
• Maintain excellent customer satisfaction
• Increase speed of software delivery
• Hit revenue targets

How does this apply to me?

• Gives insight into metrics important to other areas
• Identifies potential external metrics
• Focuses efforts on value-add metrics

@nicolefv
An Example at Chef: Engineering initiative
Where should I start?

Identify a goal

• Is there existing data?
• Who are the key people?
• What things are important to support this goal?
Chef Scorecard 2015 (scrubbed)

Product & Engineering

• Increase commercial adoption of premium features
• Software developer life cycle - hiring, on-boarding, career progression, including diversity goals

• **Ship every day**
  • Rebuild and reboot Hosted Chef
  • Customer satisfaction and ease of use aka "ease of getting going"
  • Deliver against 2015 road map
  • Become industry reference for development of enterprise software at velocity
  • Low touch conversion (download / OSS -> premium:time to value)
Where should I start?

Identify a goal: SHIP EVERY DAY

• Is there existing data?
• Who are the key people?
• What things are important to support this goal?
At Chef: Engineering initiative

Identify a goal: SHIP EVERY DAY

- **Test Maturity**
  - Unit tests
  - Component tests
  - Integration tests
  - Upgrade tests
  - Compatibility tests
  - Ancillary tests
At Chef: Engineering initiative

Identify a goal: SHIP EVERY DAY

- **Test Maturity**
  - Unit tests
  - Component tests
  - Integration tests
  - Upgrade tests
  - Compatibility tests
  - Ancillary tests

**The data**
- After investigation, no consistent sources of objective data available
- Initial steps: Subjective data collected
- Interviews conducted monthly with team leads of each product
- Percentage complete for each area calculated and communicated
At Chef: Engineering initiative

Identify a goal: SHIP EVERY DAY

• **Test Maturity**
  • Unit tests
  • Component tests
  • Integration tests
  • Upgrade tests
  • Compatibility tests
  • Ancillary tests

**Integration Tests:** Software tested as part of the distributed application ecosystem prior to release.
• Are acceptance tests reviewed and defined as part of feature definition?
• Is acceptance manually reviewed prior to shipping by stakeholders?
• Are acceptance tests automated?
• Are acceptance test results reviewed as part of the release process?
• Are tests regularly reviewed for accuracy?
• Are non-valuable tests deleted (or reviewed for deletion)?
• Is there an explicit limit for how long your entire component test suite will take?
• Are tests reviewed for duration?
At Chef: Engineering initiative

Charts for Test Maturity within a Product
At Chef: Engineering initiative

Charts for ALL Test Maturity for each Product or Feature
What Did I Do? And How Can You Do It Too?

aka The Talk Outline

• Where do I start?
• What numbers should I think about?
• **What about benchmarking?**
• What else should I think about?
A word about Benchmarking:

A Benchmark is ESSENTIAL

MUST HAVES:
• Truth! Even a bad baseline is good
• At least ONE reference group
  • Within team
  • Other: Other team? Whole company? Industry?
• Communicated / visible

@nicolefv
At Chef: Engineering initiative

Charts for ALL Test Maturity for each Product or Feature

- ✔ Truth!
- ✔ Communicated / visible
- At least one reference group

Even a bad baseline is good

@nicolefv
Your metrics aren't set in stone.
Start with an MVP, then iterate and improve.
Toss what doesn’t work.
At Chef:

**Identify a goal: SHIP EVERY DAY**

- **Test Maturity**
  - Unit tests
  - Component tests
  - Integration tests
  - Upgrade tests
  - Compatibility tests
  - Ancillary tests

- 1. External
- 2. Internal
- 3. Cultural
At Chef:

Identify a goal: SHIP EVERY DAY

- **Test Maturity**
  - Unit tests
  - Component tests
  - Integration tests
  - Upgrade tests
  - Compatibility tests
  - Ancillary tests

1. External
2. Internal
3. **Cultural**
## Westrum cultural measure

<table>
<thead>
<tr>
<th>Pathological</th>
<th>Bureaucratic</th>
<th>Generative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power-oriented</td>
<td>Rule-oriented</td>
<td>Performance-oriented</td>
</tr>
<tr>
<td>Low cooperation</td>
<td>Modest cooperation</td>
<td>High cooperation</td>
</tr>
<tr>
<td>Messengers shot</td>
<td>Messengers neglected</td>
<td>Messengers trained</td>
</tr>
<tr>
<td>Responsibilities shirked</td>
<td>Narrow responsibilities</td>
<td>Risks are shared</td>
</tr>
<tr>
<td>Bridging discouraged</td>
<td>Bridging tolerated</td>
<td>Bridging encouraged</td>
</tr>
<tr>
<td>Failure leads to scapegoating</td>
<td>Failure leads to justice</td>
<td>Failure leads to inquiry</td>
</tr>
<tr>
<td>Novelty crushed</td>
<td>Novelty leads to problems</td>
<td>Novelty implemented</td>
</tr>
</tbody>
</table>
## Westrum cultural measure

<table>
<thead>
<tr>
<th>Pathological</th>
<th>Bureaucratic</th>
<th>Generative</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Power-oriented</em></td>
<td><em>Rule-oriented</em></td>
<td><em>Performance-oriented</em></td>
</tr>
<tr>
<td>Low cooperation</td>
<td>Modest cooperation</td>
<td>High cooperation</td>
</tr>
<tr>
<td>Messengers shot</td>
<td>Messengers neglected</td>
<td>Messengers trained</td>
</tr>
<tr>
<td>Responsibilities shirked</td>
<td>Narrow responsibilities</td>
<td>Risks are shared</td>
</tr>
<tr>
<td>Bridging discouraged</td>
<td>Bridging tolerated</td>
<td>Bridging encouraged</td>
</tr>
<tr>
<td>Failure leads to scapegoating</td>
<td>Failure leads to justice</td>
<td>Failure leads to inquiry</td>
</tr>
<tr>
<td>Novelty crushed</td>
<td>Novelty leads to problems</td>
<td>Novelty implemented</td>
</tr>
</tbody>
</table>

@nicolefv 15% 52% 33%
At Chef:

Some teams have started measuring Westrum Culture scores quarterly. It is also included in our yearly employee survey.

Rate how strongly you agree (7) or disagree (1) to the following statements:

1. On my team, information is actively sought.
2. On my team, failures are learning opportunities, and messengers of them are not punished.
3. On my team, responsibilities are shared.
4. On my team, cross-functional collaboration is encouraged and rewarded.
5. On my team, failure causes enquiry.
6. On my team, new ideas are welcomed.
Summary of Engineering Initiative inside of Chef

Collect 3 types of metrics:
- External
- Internal
- Cultural

Initial metrics efforts:
- Ship Every Day (among others)
- Test Maturity (and components)
- Westrum culture

Benchmark metrics

Benchmarked ✔

Consider:
- Subjective v. Objective
- Leading v. Lagging

Next steps as we iterate:
- Tooling for Test Maturity (Subj -> Obj)
- Most metrics are lagging (normal). Westrum is leading & lagging.
  Consider adding WIP limits (leading)
Another example at Chef:
External Customers
Where should I start?

Identify a goal

• Is there existing data?
• Who are the key people?
• What things are important to support this goal?
Chef Scorecard 2015 (scrubbed)

Product & Engineering

- **Increase commercial adoption of premium features**
- Software developer life cycle - hiring, on-boarding, career progression, including diversity goals
- Ship every day
- Rebuild and reboot Hosted Chef
- Customer satisfaction and ease of use aka "ease of getting going"
- Deliver against 2015 road map
- Become industry reference for development of enterprise software at velocity
- Low touch conversion (download / OSS -> premium: time to value)
Where should I start?

Identify a goal: INCREASE COMMERCIAL ADOPTION OF PREMIUM FEATURES

- Is there existing data?
- Who are the key people?
- What things are important to support this goal?
At Chef: External customer initiative

Goal: INCREASE COMMERCIAL ADOPTION OF PREMIUM FEATURES

Success grid becomes

Dojo = DevOps

Journey Assessment

The data

- After investigation, SalesForce data was not always consistent
- Premium Features were being reported as a count – and nothing else
- Technical Account Managers and Customer Success Engineers knew that a more holistic view was more meaningful
- New data is collected as part of a structured interview with customers. They see their profile emerge.

@nicolefv
At Chef: External customer initiative

Goal: INCREASE COMMERCIAL ADOPTION OF PREMIUM FEATURES

Dojo = DevOps Journey Assessment

Key areas for new, holistic approach
- Organizational success factors
- Local development
- Code collaboration
- Continuous integration
- Chef code deployment
- Application deployment
- Virtualization as a Service
- Full stack automation
- Continuous deployment
At Chef: External customer initiative
Charts for DevOps Journey Assessment
At Chef: External customer initiative
Charts for DevOps Journey Assessment

- ✔ Truth!
- ✔ Communicated / visible to customers

Even a bad baseline is good
At least one reference group

@nicolefv
Summary of External Customer Initiative at Chef

Collect 3 types of metrics:
- External
- Internal
- Cultural

Initial metrics efforts:
- Organizational success factors
- Local dev, CI, CD, Chef deployment, Application deployment, VaaS, Full stack automation
- Code collaboration, Org success factors

Benchmark metrics

Benchmarked ✔
- Next steps: create customer benchmark

Consider:
- Subjective v. Objective
- Leading v. Lagging

Next steps as we iterate:
- Data will remain subjective for now
- Most metrics are lagging (normal).
- Investigate cultural metrics and use of other leading metrics

@nicolefv
What Did I Do? And How Can You Do It Too?

aka The Talk Outline

• Where do I start?
• What numbers should I think about?
• What about benchmarking?
• **What else should I think about?**
What else should I think about?

Advanced Topics

• Measurement targets should use distributions of probable outcomes. Because targets should include discussions of likelihood.

• Discuss variance of metrics, especially ones thought to influence things. This speaks a lot to the consistency of your process and work.

• Don’t ignore the rate of change in your data. Change is good, yes, but how is it changing? This yields valuable insights.
What else should I think about?

Advanced Topics

• Be careful about the use of normalizations in your reporting and metrics. Normalizations are useful for comparison across groups. Improper use can lead to improper smoothing (e.g., a defect is still a defect and someone will still feel pain).

• Distributions in operations and development are rarely normal, so the use of means and standard distributions are inappropriate. Consider the use of means (or transform the underlying distribution first.)
Thank you

nicole@chef.io
@nicolefv
nicoleforsgren.com