Software is Eating the World

- *Marc Andreessen*

Markus Eisele
Developer Adoption Lead

Jürgen Hoffmann
Sr. Mgr. Sales Specialists

Red Hat

1

Die Cloud startet im Kopf.

Alles beginnt bei der Organisation. Besonders dann, wenn es darum geht Ihr Unternehmen auf die Herausforderungen der Zukunft einzustellen. Der Schritt in die Cloud ist nicht nur eine neue Kostenstelle für die IT. Es ist ein Paradigmenwechsel und Versprechen mit neuen Anwendungsarchitekturen und Prozessen die Produktionszeiten zu verkürzen.

Red Hat

2

Metriken sind die Straßenmarkierungen zum Erfolg.

Es wird viel gemessen in der Cloud. Aber die Menge macht es nicht. Statt Trendlinien in bunten Präsentationen gilt es den Werten Gehalt zu verleihen. Sind die Anzahl der Services oder Container so wichtig? Was bedeutet Erfolg wirklich?

Red Hat

3

Schränken Sie die Entwickler nicht ein.

Entwickeln ist eine kreative Kunst. Und wie bei jeder Kunst steckt in ihr bahnbrechendes Potential oder großes Chaos. Man kann das Eine aber nicht ohne das Andere haben. Wenn als einziges Werkzeug ein Hammer verfügbar ist, sieht auf einmal alles aus wie ein Nagel.

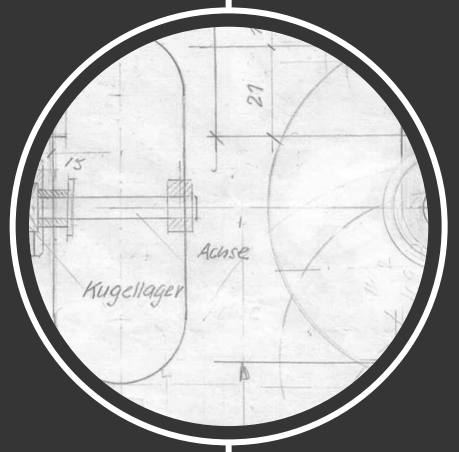
Red Hat

4

Daten sind die Treiber Ihrer Strategie.

Wirklich. Keine moderne Anwendung, egal in welcher Architektur, kommt ohne Daten aus. Sie waren, sind und werden auch zukünftig die Währung digitaler Unternehmen bleiben. Ihr digitales Gewicht entscheidet über den Ort ihrer Lagerung. Der Ort über die Strategie in die Cloud.

Red Hat



5

Wer selber baut wird nie fertig.

Würden Sie selber ein Haus bauen? Vom Fundament bis zum Dachstuhl und der Verkabelung aller Räume? Was von Außen vielfach standardisiert und viereckig aussieht, gibt es aber in vielen Varianten. Schon für einfache Ausbaustufen brauchen Sie viele Spezialisten.

Red Hat

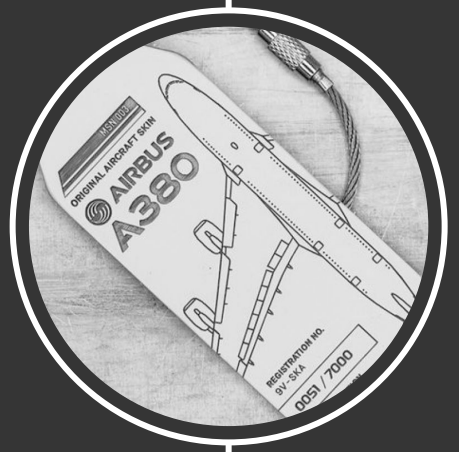
6

Kennen Sie ihre Anwendungen.

Ich hab Anwendungen immer als meine Kinder gesehen. Als Entwickler begleitet man sie von der Idee zum produktiven Leben. Und alle haben unterschiedliche Bedürfnisse. Manche müssen extrem gut skalieren, andere nur einmal im Jahr arbeiten.

Red Hat

7



Entwickeln Sie Migrationsszenarien.

Aus alt mach neu. Nur weil sich die Infrastruktur ändert, ergibt sich nicht automatisch ein Neubau von Anwendungen. Manchmal reicht es schon Bestehendes einfach zu verpflanzen. Vielleicht kann auch ein moderner Anbau helfen. Nur was wirklich ausgedient hat, muss weg. Oder halt neu. Vereinbaren Sie einheitliche Bewertungskriterien.

Red Hat

8

Das Richtige bauen und betreiben.

Jetzt ist er offen. Aber es hat 14 Jahre gedauert. Rauch musste durch den Keller. Rolltreppen waren zu kurz. Monitore kaputt. Es herrschten Kabelsalat und Einsturzgefahr. Letztere droht in den Wolken hoffentlich nicht. Aber ohne erprobte Baupläne sollte man nicht starten.

Red Hat

9

Betrieb ist die hohe Kunst der Cloud.

Egal wie schön verpackt etwas daher kommt, der Teufel steckt zumeist im alltäglichen Umgang. Sind alle Teams in Ops oder ist alles nach starren Prozessen ausgerichtet und zum Stillstand verdammt? Nur weil man einen Rennwagen kauft, fährt man noch lange nicht die schnellste Strecke auf dem Ring.

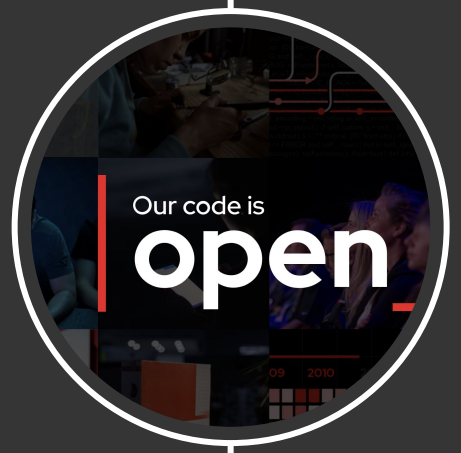
Red Hat

10

Vergessen Sie die geheime Zutat nicht!

Innovationen starten mit Zusammenarbeit.

Bringen Sie Ihre Teams zusammen. Denn nur durch Kollaboration kann etwas Großartiges entstehen. Geniale Ideen sind der Weg in die Zukunft. Mit Open Source ist sie nur einen kleinen Schritt entfernt.



Red Hat

Ready to innovate?

Finden Sie heraus, ob Ihr Unternehmen bereit ist, neue Technologien einzusetzen.

Machen Sie den Test.

Red Hat

Die Zukunft Cloud-nativer Entwicklung ist Quarkus

Sehen Sie sich das Video vom Red Hat Quarkus Day an!

Jetzt Video ansehen

Red Hat

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat

Container, Cloud und Microservices

1) Die neue Welt startet ganz oben (Organisatorische Änderungen, DevOps, Transformation)
2) Die richtigen Metriken Definieren (Warum in die cloud? Wie gut können wir erfolg messen? Anzahl Microservices?)
3) Schränken Sie ihre Entwickler nicht ein (Entwicklerproduktivität ist kritisch, Self Service)
4) Denken Sie über Ihre Daten nach (Datenhaltung on premise, public cloud, Governance, Migration, Kosten)
5) DIY ist kein Spiel (Plattformen müssen bestimmte Funktionen beinhalten. Outer Architecture, K8, Security, SRE Möglichkeiten)
6) Kategorisieren Sie Ihre Applikationen
7) Definieren Sie Migrationsscenarien (Rehost, Replatform, Refactor)
8) Das Richtige, richtig bauen und betreiben. (Leadership, Product, Development, Architecture, Operations)
9) „Wer viel misst, misst Mist“ (Die richtigen Metriken für SRE finden)
10) Die geheime Zutat nicht vergessen.

Red Hat

1) Die Cloud startet im Kopf

Buddy:

- Wir haben einen Vertrag mit einem Public Cloud Anbieter geschlossen
- Damit wir bessere Prozente bekommen, haben wir einen Commit Spent Vertrag gemacht, jetzt müssen wir so schnell wie möglich Last Auf die Cloud bringen
- Neue Anwendungen verwenden momentan exzessiv die Services in der Public Cloud…
- Aber damit wir die Mengen erreichen um in neue Discounts zu kommen, müssen wir auch die Bestandsapplikationen in die Cloud bringen.
- Ich habe mich noch nicht mit der Public Cloud beschäftigt, kannst du mir helfen, das besser zu verstehen und mir sagen, wo etwaige Fallstricke sind?

Markus:

- IaaS
- PaaS
- Plattform

2) Metriken sind die Straßenmarkierungen zum Erfolg

Buddy:

- Also, um Mehrwerte aus der Public Cloud zu ziehen, muss ich mir die Public Cloud als mehr vorstellen, als nur ein fremd gemangedes Rechenzentrum…
- Aber zumindest muss ich mir über den Betrieb der Infrastruktur und das Monitoring keine Gedanken mehr machen, denn der Public Cloud Anbieter übernimmt ja alle SLA's
- Das wird uns eine Menge Zeit und Kosten für den Betrieb ersparen, oder übersehe ich da etwas?

Markus:

- Marketing vs. Realität ;-)
- Anbieter SLAs bekannt?
- Definiert den Erfolg! Findet die richtigen Metriken!

3) Schränken Sie die Entwickler nicht ein

Buddy:

- Ok, also anstatt wie in der Vergangenheit unsere Infrastruktur zu überwachen, verlagern sich Metriken… Wir überwachen nicht mehr einzelne Systeme, sondern Anwendungen und deren Prozesse…
- Aber Markus, hilf mir… Wie ist das mit dem Application Lifecycle Management? Den haben wir in der Vergangenheit selbst im Betrieb gemanaged.
- In der Vergangenheit, habt ihr uns fertige Anwendungen für den Betrieb übergeben, und wir haben diese auf Basis unserer regulatorischen Anforderungen betrieben…
- Prinzipiell hat das doch sehr gut geklappt, bis auf den kurzen Ausfall der Produktion letztes Jahr, das waren aber nur 2 Stunden.
- Du hast von 12 factor Apps und Cloud Native gesprochen, ganz ehrlich, mir ist es prinzipiell egal, welche Architekturen ihr in euren Anwendungen verwendet, so lange wir sie sicherheitstechnisch überprüfen und unter Einhaltungen der regulatorischen Anforderungen betreiben können…
- Können wir so weitermachen?

Markus:

- DevOps
- Rückbezug auf Metriken
- Marketplace
- Governance vs. Guardrailed
- Schneller durch Flexibilität und Selfservice

4) Daten sind die Treiber Ihrer Strategie

Buddy:

- Oha, das habe ich mir einfacher vorgestellt, das hört sich ja nach einer vollkommen neuen Art und Weise an, wie wir künftig Software entwickeln und betreiben. Ich kann mich nur schwer mit dem Gedanken anfreunden, euch einfach Services in einem Portal buchen zu lassen, ohne diese vorher geprüft zu haben, aber wenn es dabei hilft den Konsum in der Public Cloud schnell nach oben zu bringen, ist das ja ein Thema, dass wir auch im Nachgang klären können.
- Wie ist das denn mit den Daten? Im letzten Jahr hatten wir ein massives Sicherheitsproblem, und uns sind Kundenstammdaten abhanden gekommen. Zwar nur in einem Testsystem und die Daten waren anonymisiert, aber ich muss sicherstellen, dass die Daten auch in der Public Cloud sicher sind…. Wie siehst du den Umgang mit Daten?

Markus:

- Data Sovereignty
- Data Mobility

5) Wer selber baut wird nie fertig

Buddy:

- Daten nah an der Anwendung, und agil? Einfach so in die Public Cloud transferieren? Das macht mir ein bisschen Kopfschmerzen, wenn ich ehrlich bin. Aber das können wir bestimmt auf die Schultern des Cloud Anbieters packen, und den haftbar machen
- Wenn ich dich richtig verstanden habe, dann brauchen wir eine Art Plattform auf der Plattform des Public Cloud Anbieters, richtig? Also eine Plattform, die unsere Metriken misst, die uns eine Art Self-Service für Entwickler bereit stellt, in dem man die Cloud Services und eigene Services einfach zu Projekten hinzu buchen kann und die unsere Anforderungen an Daten erfüllt…
- Gibt es für so etwas bereits fertige Lösungen, oder müssen wir das selbst bauen? Wenn ich ehrlich bin, haben wir eigentlich kein Budget mehr verfügbar um eine fertige Lösung zu kaufen… Aber zum Glück haben wir ja richtig gute Entwickler, die uns das bauen können, richtig?

Markus:

- Was braucht so eine Applikationsplattform alles?
- Selber bauen “können” vs. Zeit/Kosten
- ARO/ROSA kann auch im Rahmen des Commit Spend Vertrages benutzt/bezahlt werden

6) Kennen Sie ihre Anwendungen

Buddy:

- Also eine solche Plattform selbst zu bauen, würde viel zu lange dauern… Wir würden so sehr lange brauchen um unsere vereinbarten Konsummengen zu erreichen.
- Es gibt fertige Lösungen, die wir sogar mit unserem Commit Spent Vertrag bezahlen können und die uns somit schneller in die Lage bringen neue Prozesse für die Software Delivery aufzusetzen und unsere Konsumziele schneller zu erreichen. Das finde ich super!
- Wir haben letztes Jahr 3 Firmen gekauft. Glücklicherweise haben wir die Anwendungen der Acquisitionen genau katalogisiert und betreiben diese momentan in VM's. Diese VM's können wir aber problemlos in die Public Cloud schieben, oder nicht? Quasi As-Is…

Markus:

- 6 R's
  - Retain
  - Retire
  - Re-Host
  - Re-Platform
  - Re-Purchase
  - Re-Architect

7) Entwickeln Sie Migrationsszenarien

Buddy:

- Ok, nur das ich das richtig verstehe, man kann zwar den Status Quo so wie er ist, in die Cloud packen, aber davon hat man prinzipiell nichts, sondern nur die Kosten für den Betrieb verdoppelt… richtig?
- Das hat uns der Hyperscaler viel viel einfacher verkauft. Eine sogenannte Turn-Key Solution. Es wäre sehr sehr einfach, den Bestand in die Cloud zu migrieren…
- Das hört sich an, als bräuchten wir einen Plan. Wie lange dauert es einen Plan zu entwickeln?

Markus:

- FreeForm

8) Das Richtige bauen und betreiben

Buddy:

- Puh … das haut mich etwas um. Ich dachte das würde alles etwas schneller gehen.
- Lass mich das alles noch einmal alles zusammen fassen.
    - Wir brauchen etwas, wie einen Developer Self-Service.
    - Daten müssen nah bei den Applikationen lagern und eigentlich frei zwischen der Public Cloud und dem Rechenzentrum hin und her fliessen.
    - Für die Migration des Bestands brauchen wir einen Plan
    - Wir müssen eine Plattform aufbauen, um deine Anforderungen umzusetzen
    - Wir müssen Metriken überwachen, die eine neue Form der Software Delivery oder Industrialisierung der Software Delivery überwachen
    - Aber zumindest müssen wir uns in der Public Cloud keine Gedanken um die Systemsicherheit und Netzwerksicherheit machen, oder?

Markus:

-

9) Betrieb ist die hohe Kunst der Cloud

Buddy:

- Also selbst mit Sicherheit ist es nicht einfach… Eigentlich wollten wir ja durch das Einführen der Public Cloud schneller werden. Jetzt macht es mir den Eindruck, dass es einfach nur ein Verlagern von Komplexität vom Rechenzentrum in die Public Cloud ist…
- Auf uns lastet immer noch sehr große Verantwortung.
  - Der Hyperscaler betreibt zwar die Infrastruktur
  - Wir müssen eine Betriebsplattform auf dem IaaS Layer entwickeln
  - Die Überwachung der Anwendungsverfügbarkeit lastet auf uns
  - Wir sind für die Sicherheit der Netzwerkkonfiguration verantwortlich
  - Wir müssen regulatorische Anforderungen umsetzen
- Zum Glück ist mein Team recht aufgeschlossen und in der Lage neue Anforderungen schnell umzusetzen. Oder siehst du das anders?

Markus:

- Reminder DevOps
- Interdisciplinäre teams
- Emphasize Cultural Change

10) Vergessen Sie die geheime Zutat nicht!

Buddy:

- Abschließend bleibt also zu sagen, dass es nicht reicht die Public Cloud einzuführen.
- Wir müssen die Art und Weise wie wir Software entwickeln verändern
- Wir müssen uns Gedanken machen, welche Metriken für uns wichtig sind, und wie wir diese messen
- Wir müssen Entwickler ermächtigen Software selbst zu bestellen und automatisiert bereitgestellt zu bekommen
- Daten müssen frei vom RZ in die Cloud hin und her transportiert werden können
- Wir müssen einen Migrationsfahrplan für die Bestandsanwendungen entwickeln
- Security müssen wir uns ebenfalls genau ansehen, genauso wie regulatorische Anforderungen
- Und wir müssen unsere Kultur des Betriebes und der Softwareentwicklung anpassen
- Gibt es da etwas am Markt, dass man vielleicht adaptieren könnte?

Markus:

- Open SOurce
- Communities