

# Das Digital Startup bei Boehringer Ingelheim

## Interviewpartner:

Clemens Utschig-Utschig, Global Head of IT Technology Strategy,  
Boehringer Ingelheim RCV GmbH & Co KG

---

## Warum habt ihr bei Boehringer BIX aufgebaut und was genau ist BIX überhaupt?

Wir haben schon früh erkannt, dass Digitalisierung für uns als ein führendes innovatives Pharmaunternehmen viele Möglichkeiten bietet, dass wir damit neue und noch bessere Lösungen für Patienten und Gesundheitssysteme schaffen können, etwa individualisierte Behandlungsoptionen. BIX ist das 2017 neu geschaffene Digital Lab von Boehringer Ingelheim, in dem wir neue Konzepte und digitale Lösungen für unsere Geschäftsbereiche Humanpharma und Tiergesundheit entwickeln wollen. Boehringer Ingelheim investierte 2017 über 10 Millionen Euro in das Startup, um unsere digitale Transformation durch Data Science, agile Softwareentwicklung und User Experience Design zu unterstützen und zu befeuern. Wir haben damit den idealen Rahmen geschaffen, um die Freiheiten und die Agilität eines Start-up mit den Stärken eines globalen Konzerns zu verbinden.

## Die Geschwindigkeit bei der Verfolgung neuer Ideen, der Wunsch nach Nachhaltigkeit und Anforderungen an die Compliance treffen aufeinander. Wie habt ihr diese Widersprüche gelöst?

Da haben wir als Start-up eines Pharmaunternehmens einen Vorsprung gegenüber anderen. Compliance war schon immer von höchster Bedeutung für uns und Richtschnur unseres Handelns. Entsprechend bringen wir das notwendige Mindset mit sowie einen reichhaltigen Erfahrungsschatz an Methoden und Technologien.

Blieben wir bei der Technologie: Wenn du softwaregetriebene Innovation betreibst – und Digitalisierung hat viel mit Technologie, somit inhärent also auch mit Softwareentwicklung zu tun –, brauchst du einen Entwicklungs-Stack, der es dem Entwickler erlaubt, sich auf die Fachlichkeit seiner digitalen Lösung zu konzentrieren. Hier gibt es vielfältige Ausprägungen: Application Development, Mobile Development oder Data Science Engineering. Entscheidend ist, sich nicht bei jeder neuen Initiative oder jedem MVP immer wieder mit der Frage beschäftigen zu müssen, wie ich meine Umgebung aufsetze. Oder wie das Testmanagement erfolgt. Oder: Wie baue ich das ganze Zeug zu einem lauffähigen Produkt zusammen? Und wo lege ich meine Daten und Dokumentation eigentlich ab, um auch die Regularien

zu beachten? Dokumentation und Geschwindigkeit sind auch keine Widersprüche. Im Gegenteil, eine gute Dokumentation kann mir bei der Entwicklung und dann im Betrieb viel Zeit ersparen. Somit treffen sich die beiden Ziele Geschwindigkeit auf der einen Seite sowie Nachhaltigkeit und Compliance auf der anderen Seite unmittelbar bei jedem Vorhaben im BI X.

Gerade Nachhaltigkeit ist ja auch entscheidend für Geschwindigkeit. Wenn zum Beispiel ein MVP als Pilot auf den Markt gebracht wird und dann der Pilot erfolgreich ist, übergeben wir das gesamte Wissen, die gesamte Code-Basis, die gesamte Dokumentation, also alles, was in Jira, Confluence, Git etc. abgelegt ist, im Rahmen eines Projekts an die IT und zur Nutzung an die Business-Organisation. Und damit ist klar, dass wir auch das gesamte Deployment Vehicle übergeben müssen, weil Continuous Integration und Continuous Delivery Teile der softwaregetriebenen Innovation sind. Geschwindigkeit und Compliance treffen sich also bei einer One-Click-Migration der gesamten Infrastruktur. Somit agieren wir nicht losgelöst im BI X, sondern entwickeln die Tools und Pipelines Hand in Hand – für BI X und für Boehringer Ingelheim.

### Woraus besteht dieser Entwicklungs-Stack, wie muss ich mir das vorstellen?

Der Stack besteht aus drei großen Teilen. Bei einem Teil dreht sich alles um Requirements Engineering, Dokumentation und agiles Software-Development-Management. In unserem Fall betrifft dies Atlassian Jira, Confluence und Bit Bucket. Der zweite große Teil beinhaltet das Container-based Development basierend auf Docker. Hier nutzen wir die Toolsets von Red Hat: Open Shift und Kubernetes. Und der dritte Teil ist das Continuous Integration Vehicle, in unserem Fall mit dem Tool Jenkins. Alle drei Teile sind voll integriert und erlauben uns extrem schnell, eine neue Initiative im BI X zu starten. Extrem schnell bedeutet, es dauert nur ca. drei Minuten, um etwa über ein Single-Click ein Self-Service Frontend und alle standardisierten Komponenten für eine Ionic Mobile App, ein AngularJS Frontend oder ein SpringBoot Backend hochzuziehen und arbeitsfähig zu sein. Unsere Entwickler oder auch unsere Data Scientists können daraufhin direkt mit der Entwicklung loslegen. Das kann sich sehen lassen, und wir sind sehr stolz darauf!

### Wie funktioniert das im Detail? Wie ist der weitere Arbeitsablauf?

Also, unsere Welt beginnt mit dem Pitch einer Initiativenidee im Digital Steering Committee. Wenn diese angenommen wird, setzen wir ein sogenanntes MVP Development auf. Durchgeführt wird dies von einem dedizierten Scrum Team mit 7–8 Personen, wobei die Business-Seite einen Product Owner stellt. Das Team

beziehungsweise der Lead Architect nutzt unsere Provisioning App und diese generiert einen Jira Space, einen Confluence Space, ein Project Repository und weitere notwendige Tools und deren Infrastruktur. Im Rahmen von Design Sprints verwenden sie die gleiche Provisioning App immer wieder, um basierend auf Standardkomponenten und Quick-Start-Templates die benötigte Entwicklungs- und Dokumentationsumgebung zu erzeugen.

Typische Standardtechnologien sind aktuell nodeJS, SpringBoot, Python und Scala oder für das Frontend AngularJS oder eben Ionic für eine Mobile App. Das heißt, wir versuchen den Entwicklerteams mehr Geschwindigkeit und Agilität zu geben, indem wir ihnen einen „Sandkasten“ zur Verfügung stellen, in dem sie sich ausprobieren können. Und alles andere, was später benötigt wird, ist längst schon da: die Continuous Integration Pipeline, die wir aufbauen, die Projekte in Open Shift, die wir hochziehen, Web Hooks, die bei jedem Check-in den Merge-Cycle und Workflow triggern. All das wird generiert von der Provisioning App.

### Warum ist Harmonisierung bzw. Standardisierung so wichtig? Wie reagieren die Entwickler darauf?

Wir fokussieren uns auf wenige Standardtechnologien, zumal wir Nachhaltigkeit und Compliance sicherstellen müssen. Das ist vielleicht auch ein bisschen anders als in einem Digital Hub, wo ein Technology-Trial-and-Error-Vorgehen üblich ist. Aber wir müssen nun mal auch sicherstellen, dass die erfolgreichen Piloten am Ende des Tages schnell produktiv werden können und ihren Nutzen beweisen. Betriebswirtschaftlich kommt einer schnellen Time-to-Market bei neuen digitalen Lösungen ein besonderer Stellenwert zu. Hierzu müssen wir das verfügbare Wissen im IT-Operating optimal ausnutzen und dürfen uns nicht in einem Wildwuchs verlieren, der Nachhaltigkeit, Geschwindigkeit und Compliance gefährdet.

Ja, es gibt immer wieder diese Diskussionen mit den Teams, ob wir nicht den Development Stack erweitern können, und in vielen Fällen einigen wir uns auf Standardtechnologien. Aber es hat auch Fälle gegeben, wo wir gesagt haben: Gut, wir erweitern den Baukasten, bauen einen neuen Quick Starter, um einfach den Vorteil der neuen Technologie zu heben. Die Verwendung von Scala ist ein gutes Beispiel hierfür.

## Wie kann man den Vorteil der Plattform kurz zusammenfassen?

Wenn richtig aufgesetzt und richtig durchdacht, gewinnst du mit Platforming sehr viel: Standardisierung, Skill-Sharing und Skill-Skalierung, Geschwindigkeit und Entwickler, die sich weit mehr mit der Lösung und weniger mit Infrastruktur, Continuous Integration und anderen Prozessen auseinandersetzen müssen. Das schafft uns also Freiräume, sodass wir uns auf Innovationen konzentrieren können. Hier haben wir die Best Practices aus dem Product Engineering mitgenommen, und ich konnte meinen Teil durch meine langjährige Erfahrung in der Produktentwicklung in den USA einbringen.

## Wir verfolgen gemeinsam die Open-Source-Stellung dieser Entwicklungsplattform. Warum genau geht ihr jetzt den Open-Source-Weg?

Ich halte das, was wir gemeinsam entwickelt haben, für etwas, was in Wirklichkeit jeder braucht. Jeder, der heute modernes Software Engineering machen will, braucht eine solche Plattform, damit er schnell, aber auch standardisiert vorgehen kann. Ich glaube nicht, dass dies eine Competitive Differentiation darstellt, sondern es handelt sich einfach nur um einen Schritt in Richtung Industrialisierung über Innovation und durch Digitalisierung. In Zeiten des DevOps-Hypes wird dieser Plattform-Ansatz die DevOps-Welle noch stärker nach vorne bringen. All das, was wir da getan haben, also One-Click Provisioning, Quick-Starter, Migration von einem Environment zum nächsten, das macht uns unser Leben einfacher. Warum sollten wir dies also nicht rausgeben? Warum nicht mehr Leute durch den Open-Source-Ansatz an solchen Erfolgslösungen beteiligen? Warum nicht DevOps damit weitertreiben? Und natürlich gibt es auch den Aspekt, dem Markt durch die Open-Source-Initiative zu zeigen, dass Boehringer Ingelheim eine Heimat für „Rockstar“-Entwickler ist, weil wir Software Development treiben können – und zwar hochklassig, modern und zukunftsweisend.