



Platinum
Solution Partner
ENTERPRISE

JIRA UNTER LAST- VORGEHEN UND ERGEBNISSE



V1.0 08.10.2018

CATWORKX IM ÜBERBLICK

- Gegründet **1999**
- Ansässig in **Deutschland, Österreich und Schweiz**
- **Über 60 akkreditierte und zertifizierte Atlassian Experten (68 MA gesamt)**
- „In Love“ mit **Jira** seit Version 1.0
- **Ersteller** der ersten deutschen Jira-Übersetzung
- Seit **über 16 Jahren im Markt** mit Atlassian-Tools
- Seit 2014 firmenweit zu **100% auf Atlassian ausgerichtet**
- Atlassian **Platinum & Enterprise** Solution Partner
- Atlassian **Authorized** Training Partner
- Atlassian **Marketplace Vendor**



KERNDIENSTLEISTUNGEN FÜR IHREN ERFOLG - ÜBERBLICK

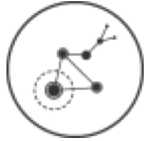
1. ATlassian CONSULTING

Integrierte, digitale Lösungen für ALM, PPM, BPM, ITSM und Ihre Geschäftsprozesse



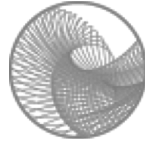
2. ATlassian OPERATIONS & SUPPORT SERVICES

Full-Service und Performance-Optimierung für den reibungslosen Betrieb Ihrer Atlassian-Infrastruktur



3. ATlassian DEVELOPMENT

Entwicklung kundenspezifischer Tools und Apps für mehr Effizienz von Atlassian-Produkten



4. ATlassian TRAININGS

Offizielle Atlassian-Trainingskurse in Hamburg, Köln, München, Winterthur und Wien oder bei Ihnen vor Ort (z.B. Bangalore)



5. ATlassian LIZENZEN

Konkurrenzlos schneller Lizenz-Service für Atlassian- und Partnerprodukte



PROJEKTZIEL / -RAHMEN

- Jira in definierten Use Cases unter Last zu setzen
- Jira auf Stabilität und Antwortverhalten zu prüfen und zu messen
 - Hierbei soll in definierten Use Cases automatisiert Last erzeugt und die Antwortzeit analysiert werden
- Benutzerreelle Simulationstests des gesamten HTTP Streams (Inkl. dynamischen Content wie JS, CSS ...)
 - Skalierung von Benutzermengen (500 -> 1000 -> 2000 gleichzeitige Benutzer)
 - Skalierung von Daten und Konfiguration via RampUp (1 Million Issues, 600 Projekte, 300 CFs ...)
 - Ergebnisdokumente im Endausbau bei 500/1000/2000 gleichzeitigen Benutzern generieren
- Perzentiles Reporting der Antwortzeiten

Zusätzlicher Nutzen:

- Load Simulation ist zyklisch (Changes etc.) wiederverwertbar
- Potentielle Messumgebung für weitere webbasierte Tools
- Als kontinuierliches Messinstrument für dauerhafte Messergebnisse einsetzbar

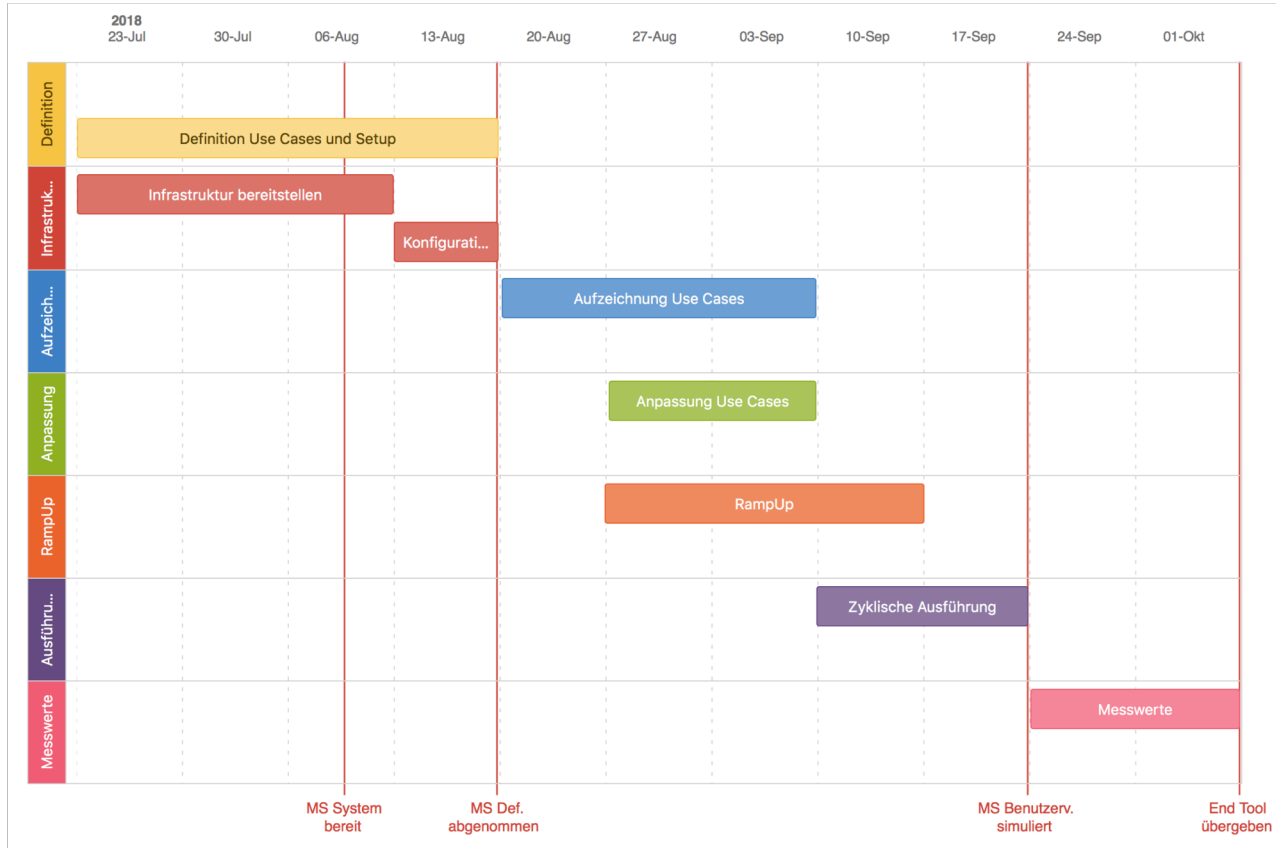
LEITSÄTZE (1)

- Unterschiedliche Atlassian Anwendungen sind nicht direkt miteinander vergleichbar, die Stabilität / Performance und das Verhalten der Anwendung hängt von der individuellen Konfiguration, installierten Apps (Plugins) und der Nutzungsweise ab. Somit schließt sich ein direkter Vergleich von unterschiedlichen Instanzen aus.
- Die einfache Fragestellung "wie schnell antwortet die Anwendung", ist nicht zielführend, stattdessen muss die Fragestellung erweitert werden: "Bei welcher Aktion ist die Antwortzeit langsam?", "Am Client?", "Am Server?", "Welcher Browser?", "Innerhalb des Unternehmensnetzwerkes?", "Mit oder ohne Apps?", "Mit oder ohne API Zugriffe?" - Grundsätzlich stellt sich die Frage, was ist schnell? Dies ist relativ und immer individuell zu bewerten.
- Um Performance Problemen / Instabilität vorzubeugen, müssen sowohl technischen als auch die organisatorischen Aspekte berücksichtigt werden, so ist zum Beispiel auf eine sinnvolle Art der Nutzung einer Anwendung zu achten. Häufig wird die Anwendung nicht so genutzt, wie sie genutzt werden soll, was in Folge mit entsprechend negativem Feedback hinsichtlich Stabilität, Performance oder Usability einhergeht.

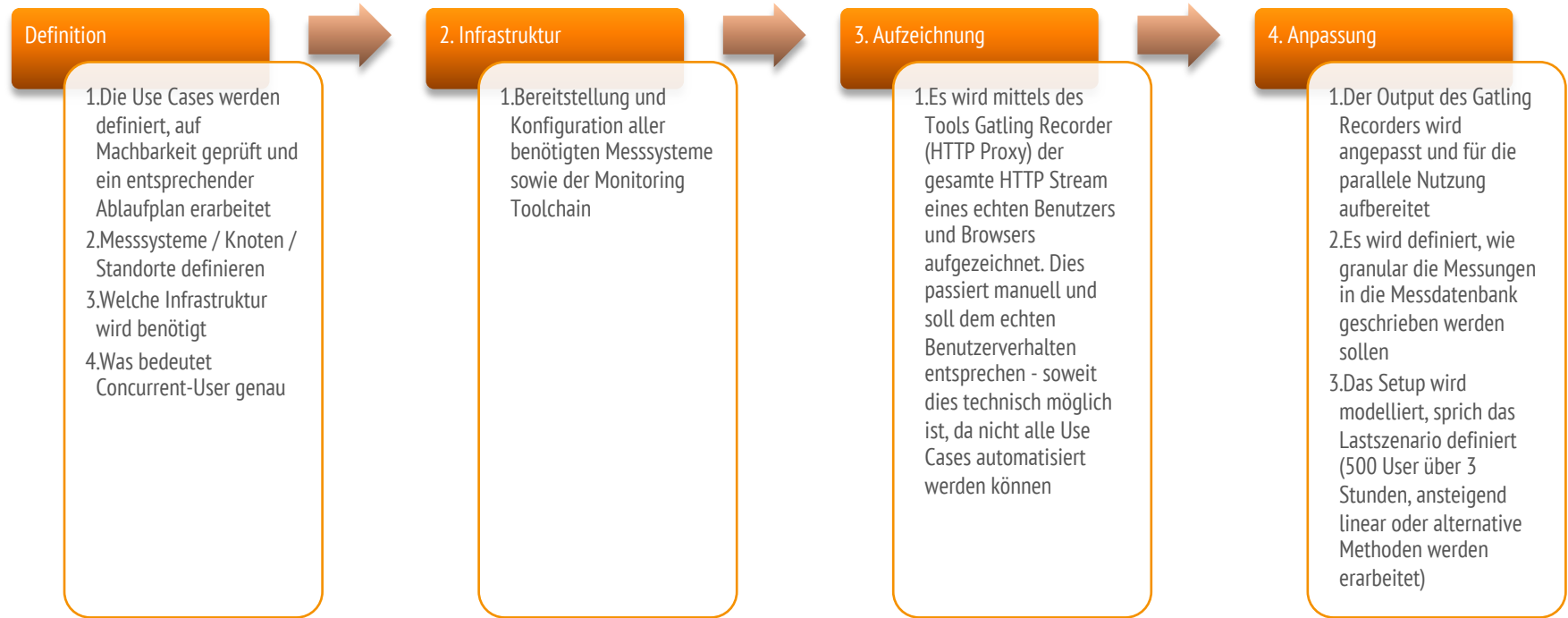
LEITSÄTZE (2)

- Ein Benutzerverhalten kann nur erahnt werden, oder über die Zeit in einem Logfile gesammelt werden. Die unter Annahmen erstellten Use Cases sind mit hoher Wahrscheinlichkeit unterschiedlich als das echte Benutzerverhalten. Solche Annahmen sind deshalb als statisch anzunehmen und reflektieren nicht das dynamische Benutzerverhalten. Ein echtes Benutzerverhalten ist immer der Dynamik von Nutzungsmöglichkeiten unterworfen (eingesetzte Apps, Dashboards, Filter, etc).
- Die Einflussfaktoren hinsichtlich Stabilität und Performance auf eine Anwendung können sehr weitläufig und komplex sein. Mittels einer Load Simulation ist es möglich, einige Einflussfaktoren zu prüfen und die Auswirkungen zu bewerten. Konkret bedeutet dies, dass in einer Umgebung in welcher kontrollierte Änderungen stattfinden, die jeweiligen Änderungen auf die Auswirkungen geprüft werden können. Ein Beispiel: Ein definiertes Szenario (Useranzahl, Use Cases etc.) bleiben identisch, die Anwendung wird aber in der Konfiguration / Komplexität angepasst - beispielsweise die Anzahl an Projekten, Issues, Custom Fields, Versionen oder auch installierte Apps/Add-On's - sollte sich nach der Anpassung die Performance / Stabilität negativ verändern, so ist entweder eine Obergrenze erreicht oder ein Problem identifiziert welches entsprechend analysiert und wenn möglich korrigiert werden muss. Die Einflussfaktoren sollten gemeinsam mit dem Kunden evaluiert, bewertet und dann gemeinsam geprüft werden. Zu berücksichtigen ist hier, dass es durchaus sein kann, dass nicht alle Faktoren ohne echte Benutzer auf dem System zu testen sind (Gewisse Faktoren werden nicht zu simulieren / automatisieren sein).

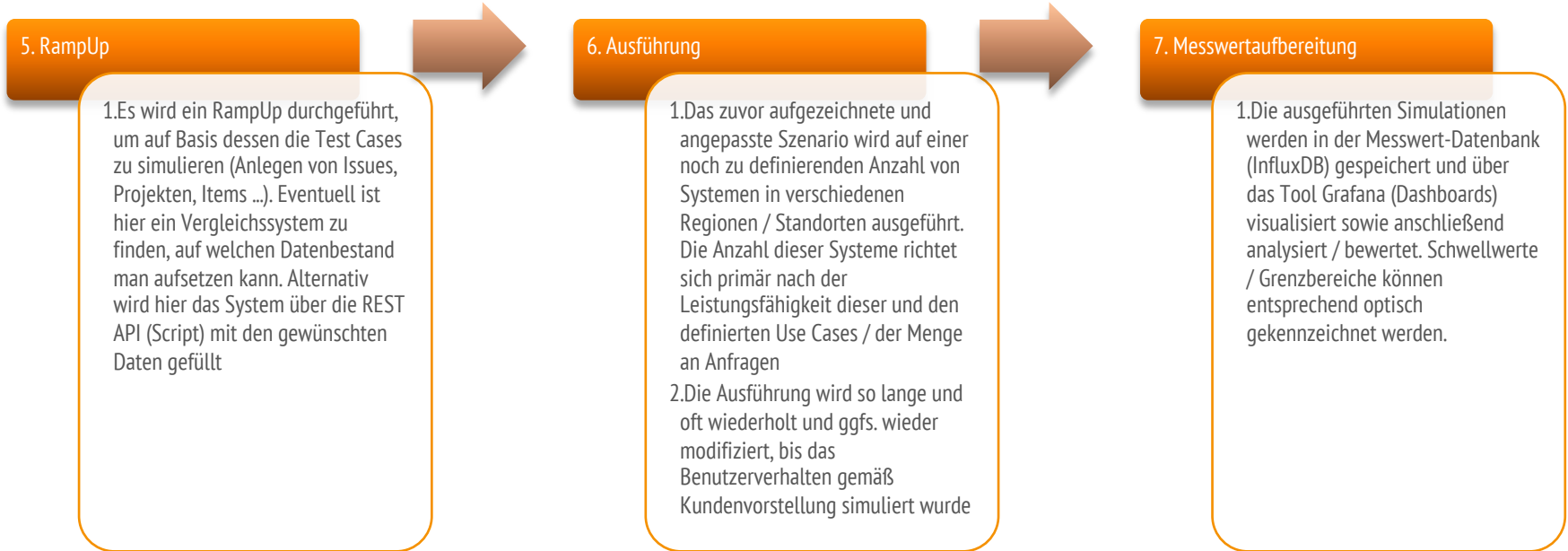
ZEITRAHMEN



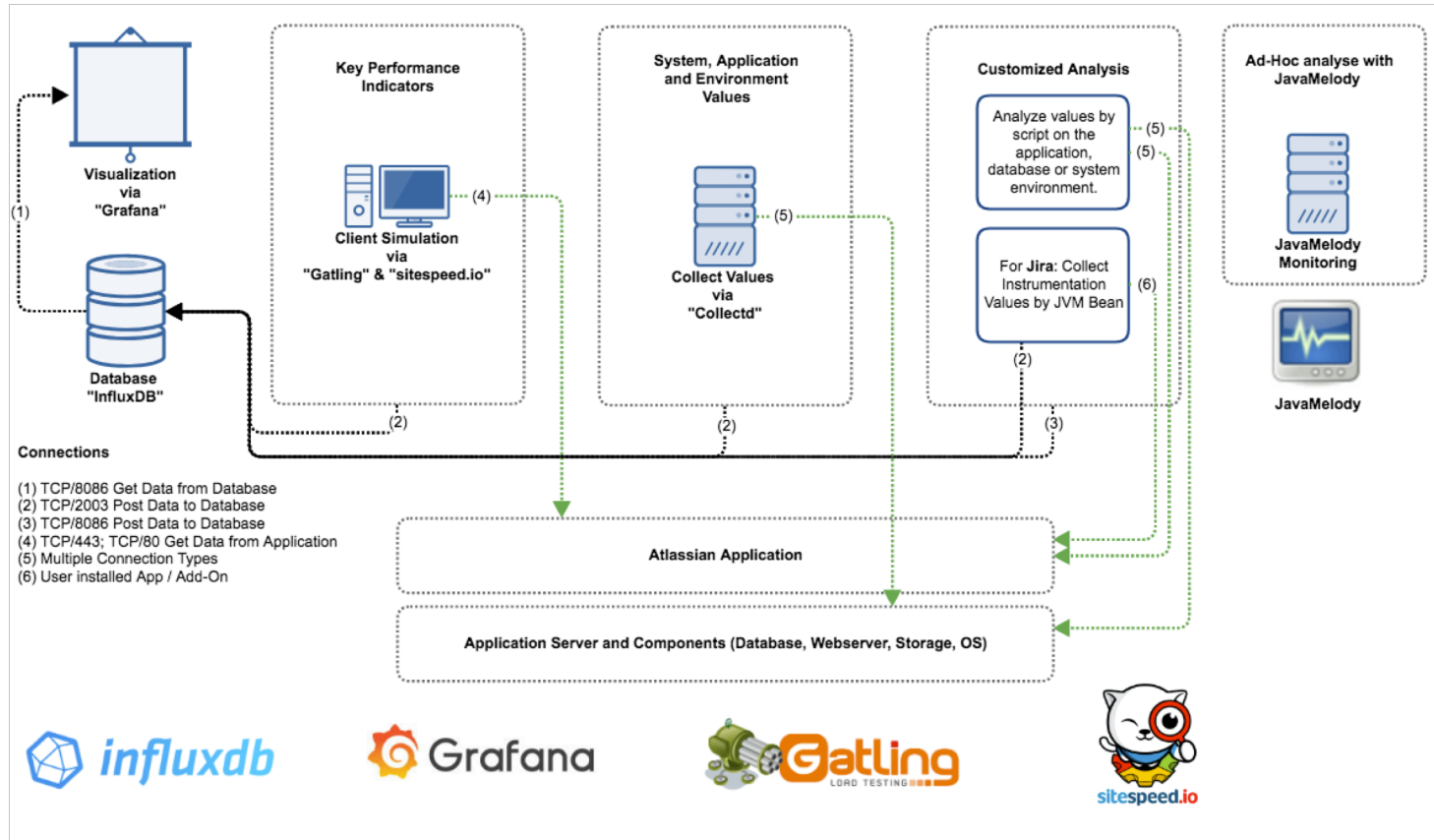
ABLAUF (1)



ABLAUF (2)



AUFBAU DES MESSSYSTEMS

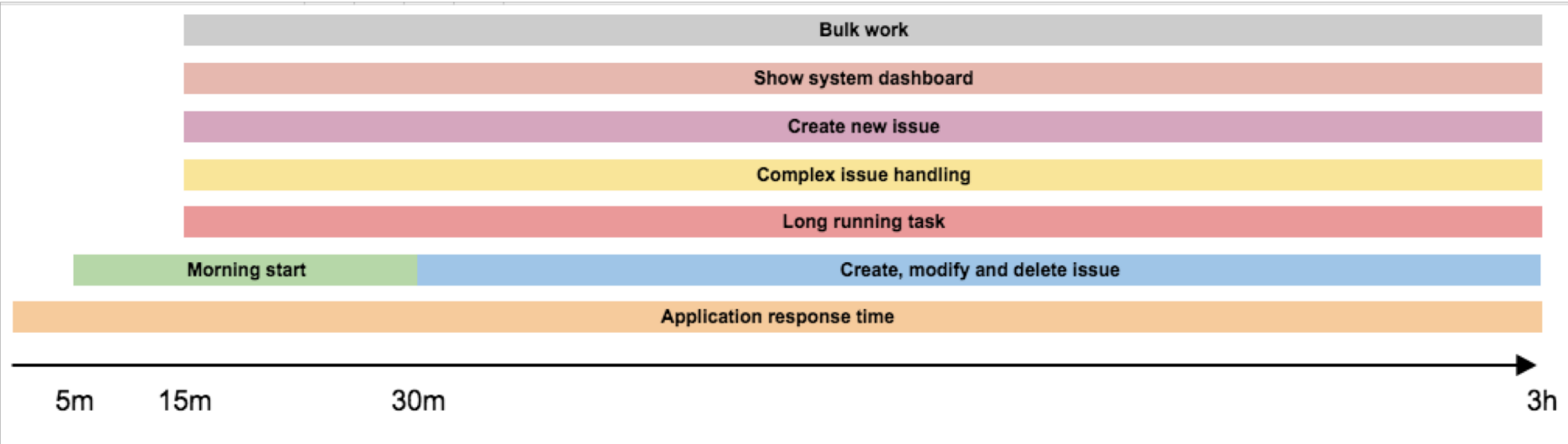


USE CASES

Überschrift	Feasibility	Ranking	Originator	Status
Add attachment	✓	1		TESTED
Add comment	✓	1	catworkx	TESTED
Assign issue	✓	1	catworkx	TESTED
Bulk edit	✓	2		TESTED
Create issue	✓	1	catworkx	TESTED
Create Kanban or Scrum Board	✓	3		TESTED
Edit Issue	✓	1		TESTED
Export filter to XML and CSV	✓	2	catworkx	TESTED
Export Issue to XML and Printable	✓	2		TESTED
Issue linking	✓	1		TESTED
Login	✓	1	catworkx	TESTED
Remove attachment	✓	1		TESTED
Remove issue	✓	1	catworkx	TESTED
Transition issue	✓	1	catworkx	TESTED
View filter	✓	1	catworkx	TESTED
View Gantt Chart	✓	1		TESTED
View issue	✓	1	catworkx	TESTED
View Kanban or Scrum Board	✓	1	catworkx	TESTED
View Structure	✓	1		TESTED
View System Dashboard	✓	1	catworkx	TESTED

ZUSAMMENFASSUNG ZU ACHT SZENARIEN

Die einzelnen, zuvor definierten, Use Cases werden in unterschiedlichen Szenarien zusammengefasst und simuliert. So soll sowohl der „normale“, der „power“ User simuliert und diejenigen dazwischen simuliert werden.



RAMPUP (DATA & CONFIGURATION)

Überschrift	Customfields	Projects	Issues p.P.	Components p.P.	Releases p.P.	Issues
Start environment	150	300	850	30	15	255.000
Start +20%	180	360	1020	36	18	367.200
Start +40%	210	420	1156	42	21	485.520
Start +60%	240	480	1530	48	24	734.400
Start +80%	270	540	1530	54	27	826.200
Target environment	300	600	1700	60	30	1.020.000

MESSUNG

Overview of performance data

Szenario	75th percentil	min	mean	max
Application Response Time: View Issue (Without Resources) @ fr04063vmx	240.00 ms	155.00 ms	228.45 ms	497.00 ms
Bulk Work: Bulk Change (Without Resources) @ fr04063vmx	8.00 ms	6.00 ms	8.92 ms	40.00 ms
Bulk Work: Export Filter (Without Resources) @ fr04063vmx	1.80 s	1.28 s	1.67 s	2.28 s
Bulk Work: Login and Dashboard (Without Resources) @ fr04063vmx	87.00 ms	39.00 ms	76.27 ms	247.00 ms
Bulk Work: View Filter (Without Resources) @ fr04063vmx	166.00 ms	108.00 ms	157.39 ms	397.00 ms
Create, Modify and Delete Issue: Create Issue (Without Resources) @ fr04063vmx	76.00 ms	43.00 ms	79.30 ms	1.04 s
Create, Modify and Delete Issue: Delete Issue (Without Resources) @ fr04063vmx	81.00 ms	45.00 ms	72.32 ms	204.00 ms
Create, Modify and Delete Issue: Modify Issue (Without Resources) @ fr04063vmx	219.00 ms	140.00 ms	204.25 ms	652.00 ms

VISUALISIERUNG



FINDINGS (EXAMPLE)

3	RESOLVED	Message "ConnectionPoolHealthSqlInterceptor: Dangerous use of multiple connections"	Codeblock language = bash title = atlassian-jira.log <pre>... 2018-09-03 00:35:35,385 StreamsCompletionService::thread-373 WARN loadsim 35x741886x150 lac0bty 10.179.172.121,100.96.5.86 /plugins/servlet/streams [G.a.lira.ofbiz.ConnectionPoolHealthSqlInterceptor] Dangerous use of multiple connections: taken => count=2; marks=[1-0]; pool=60/60 ...</pre> Connection Pool http://jira-wed-test.zona2.agileo.com/de 2018-09-03T12:09:55.678+02:00
---	----------	---	---

SIMULATION RESULTS

Step	Status	Concurrent Users	Description	Issues ★ (starting position)
1	SUCCESSFULLY COMPLETED	500	Start environment	~ 2.000
2	SUCCESSFULLY COMPLETED	500	Start RampUp environment	255.000
3	SUCCESSFULLY COMPLETED	500	Start +20%	367.200
4	SUCCESSFULLY COMPLETED	500	Start +40%	485.520
5	SKIPPED BECAUSE OF RAMPUP TIME	500	Start +60%	734.400
6	SKIPPED BECAUSE OF RAMPUP TIME	500	Start +80%	826.200
7	SUCCESSFULLY COMPLETED	500	Target environment	1.020.000
8	SUCCESSFULLY COMPLETED	1000	Target environment	1.030.000
9	SUCCESSFULLY COMPLETED	2000	Target environment	1.060.000



★ starting position Steps 1 through 7 show defined starting values to which the system had to be "ramped up". We used a script driven mechanism to fill the Jira instance with issues in order to reach the target number specified in the "Issues" column. As for step 8 and 9, the number of issues in the system has increased through the simulation runs, during which issues were added to the system. The exact number can be seen on the detailed results report pages.

ERKENNTNISSE

- Diese Art der Simulation ermöglicht ein abschätzen des Anwendungsverhalten bei späterer Last-/Ausbaustufe
- Probleme können vorab identifiziert, reproduziert und i.d.R. behoben werden
- Ein dediziertes RampUp einzelner Attribute (Issues, Projekte, CFs, Versions etc.) ermöglicht einen Nachweis über deren Einflussfaktor / Kritikalität.
- Die maximale Ausbaustufe kann grob ermittelt werden
- Die Infrastruktur kann optimal angepasst / ausgewählt werden
- Bei Changes oder für kontinuierliche Messungen ist die Umgebung wiederverwertbar
- Die Use Cases und Simulationen müssen detailliert geprüft und skizziert werden, um ein möglichst Benutzernahes Verhalten zu simulieren (Nicht alle Use Cases sind umsetzbar)

TEAM CWX / AUFWAND

Mitarbeiter	Rolle
Carsten Bälkner	Projektleiter (Koordinator, Termin- und Ressourcenplanung)
Benjamin Dombrowsky	Fachlicher Leiter (Infrastruktur, Setup, Testbegleitung)
Sönke Martens	Use Case Design, Setup, Durchführung
Andre Rossy	Technische Umsetzung
Alexander Bundt	RampUp

Lfd.-Nr. Phase	Name	Beschreibung	Aufwand
1	Definition Use Cases und Setup	*Ausarbeitung Use Cases *Bewertung Use Cases *Machbarkeitsprüfung *Zeitlichen Ablaufplan / Szenario erarbeiten	10 PT
2	Infrastruktur	*Konfiguration der Messsysteme in verschiedenen Standorten - Gatling Hosts *Bereitstellung Monitoring Toolchain auf zentralem Messsystem - InfluxDB, Grafana	5 PT
3	Aufzeichnung Use Cases	*Aufzeichnung HTTP Stream der Use Cases *Transformierung in Scala Code	8 PT
4	Anpassung Use Cases	*Modifikation des Scala Code *Definition Messpunkte *Definition Setup Szenario (Load Simulation)	7 PT
5	RampUp	*Anwendung mit Daten gemäß Spezifikation befüllen *Datenintegrität prüfen *Abnahme zu testende Anwendung durch den Kunden	10 PT
6	Ausführung	*(Multiple) Ausführung und zyklische Anpassung der Load Simulation	5 PT
7	Messwerte	*Aufbereitung der Messergebnisse *Analyse und Bewertung *Übergabe und Einweisung in die Toolchain	4 PT

catworkx Gruppe

Deutschland | Schellerdamm 16 | 21079 Hamburg | Tel. +49 40 890646-0 | Fax +49 40 890646-66 | info-de@catworkx.com | www.catworkx.com
Österreich | Gußhausstr. 23/1/18 | 1040 Wien | Tel. +43 1 2369317-22 | Fax +43 1 2369317-9 | info-at@catworkx.com | www.catworkx.com
Schweiz | Bahnhofplatz 17 | 8400 Winterthur | Tel. +41 52 56022-20 | Fax +41 52 56021-00 | info-ch@catworkx.com | www.catworkx.com

VERWENDETE APPS

Atlassian JIRA - Plugins - Transition Trigger Plugin - 4.0.8

FishEye Plugin - 7.1.5

Issue Templates for JIRA (Gold Edition) - 7.0.2

JIRA Agile - 7.3.0-DAILY20170523023608

JIRA Bamboo Plugin - 7.4.5

JIRA DVCS Connector Plugin - 4.1.3

JIRA Suite Utilities - 2.1.0

JIRA for Software Plugin - 2.2.0

JavaMelody Monitoring Plugin - 1.65.0

Kantega Single Sign-on with Kerberos and SAML - 2.7.2

Structure - 4.6.0

Structure.Gantt - 1.0.1

catworkx Project Metadata - 7.0.1.8

catworkx Portfolio Management for Jira - 7.0.2.14

eazyBI Reports and Charts for Jira - 4.4.1