

Red Stack

Magazin

DOAG

SOUG
swiss oracle
user group

AOUG
AUSTRIAN ORACLE USER GROUP

inklusive BUSINESS NEWS



JAVASCRIPT

Aus der Praxis

Oracle Enterprise
Manager 13c – Patching
und Upgrade



Im Interview

Gerald Venzl,
Senior Director Product
Management, Oracle

Business News

Das Konzept der
Smart Cities



CloudLand
WWW.CLOUDLAND.ORG

Eventpartner: Heise Medien

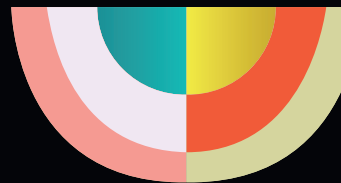
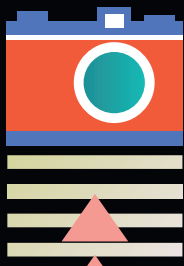
DAS CLOUD NATIVE FESTIVAL

DAS EVENT DER DEUTSCHSPRACHIGEN
CLOUD NATIVE COMMUNITY

18. – 21. JUNI



#CLOUDLAND2024





Kai Donato
DOAG Themen-
verantwortlicher
JavaScript

Liebe Mitglieder, liebe Leserinnen und Leser,

JavaScript ist eine der am rasantesten wachsenden Technologien des letzten Jahrzehnts. Sie begleitete uns zunächst als (mal mehr, mal weniger) sinnvolle Erweiterung für HTML-Webseiten und später als Grundlage der meisten Web-Anwendungen im Frontend sowie auch im Backend. Heutzutage ist es kaum möglich eine Webseite zu besuchen, ohne zumindest ein paar Zeilen JavaScript über die Leitung mitwandern zu lassen. Ein großes Ökosystem und die dazugehörigen Communities haben den Einflussbereich von JavaScript zunächst schlagartig und im weiteren Verlauf stetig größer werden lassen.

In dieser Ausgabe erfahren wir, neben dem spannenden Thema der Smart Cities in der integrierten Business News, auch vieles über JavaScript in APEX und über JavaScript in den unterschiedlichsten Facetten.

Ich wünsche Euch viel Spaß beim Lesen!

Euer Kai Donato



Ausgabe Nr. 1/2024
auf Abruf!

DOAG WEBSESSION

Die DOAG WebSessions* bieten Ihnen in regelmäßigen Abständen spannende Online-Vorträge und -Diskussionen zu einer Vielzahl von Themenbereichen aus den jeweiligen DOAG Communities.

Freuen Sie sich auf WebSessions rund um die Themen Datenbank, Data Analytics und NetSuite oder beteiligen Sie sich bei den DOAG DevTalks an interessanten Gesprächsrunden zu aktuellen Development-Themen!



www.doag.org/go/websessions



*Die Buchung der WebSessions erfolgt ganz einfach über unseren Shop. Mitglieder erhalten im Buchungsprozess automatisch **100 % Rabatt.**



08

Interview mit Gerald Venzl



30

AWR – Warehouse die Zentrale für AWR-Daten



66

MySQL, PostgreSQL und Oracle – ein einfacher Vergleich

Einleitung

- 3 Editorial
- 6 Timeline
- 8 „JavaScript hat sich definitiv als ‚Mainstream‘-Programmiersprache durchgesetzt und wird nicht mehr so schnell wieder verschwinden.“
Interview mit Gerald Venzl

JavaScript

- 12 Ersetzt Qwik alle bisherigen JavaScript Frameworks?
Robert Maier-Silddorff
- 18 Entwicklung offlinefähiger APEX-Anwendungen
Dennis Amthor
- 26 Node.js und seine Herausforderer: JavaScript Server Runtimes
Philipp Hartenfeller

Datenbank

- 30 AWR-Warehouse – die Zentrale für AWR-Daten
Ulf Schabram
- 54 Oracle Enterprise Manager 13c – Patching und Upgrade
Birgit Arndt
- 66 MySQL, PostgreSQL und Oracle – ein einfacher Vergleich
Raphael Salguero

Engineered Systems

- 74 Hardware-Ressourcen auf der Oracle Database Appliance verwalten
Sven Illert

APEX

- 80 APEX Dynamic Controller (ADC)
Jürgen Sieben

BUSINESS NEWS

Smart Cities

- 36 Das Konzept der Smart Cities
Dr. Christian Temath
- 42 Schlüsselrolle Künstliche Intelligenz – die Zukunft des ÖPNV in Smart Cities
Dr. Roxana Hess und Katja Rische
- 46 Ressourcen optimal nutzen, Effizienz steigern, Innovationen schaffen – das Konzept der Smart Cities
Dennis Hering
- 50 Privacy by Design als Basis für Datenschutz in Smart Cities
Yahya Mohamed Mao



36

Leitartikel | Das Konzept der Smart Cities



80

APEX Dynamic Controller (ADC)



46

Ressourcen optimal nutzen, Effizienz steigern, Innovationen schaffen – das Konzept der Smart Cities



50

Privacy by Design als Basis für Datenschutz in Smart Cities

Intern

- 89 Neue Mitglieder + Termine
- 90 Impressum + Inserenten

News

- 85 Oracle Datenbanken Monthly News
- 88 Best of DOAG Online

TIMELINE

17. Oktober 2023

Die Regio Freiburg trifft sich zum Regionaltreffen in Freiburg. Mit dabei sind Volker Deringer und Jörg Sobottka. In einem Vortrag werden die Stärken und Schwächen der beiden Entwicklungsumgebungen Forms und APEX in Bezug auf Installation, Entwicklung, Deployment und Oberflächen-gestaltung verglichen.

19. Oktober 2023

Im DevTalk mit Samuel Nitsche und Oliver Lemm erfahren die Teilnehmenden alles rund um das Thema "Back-end Testing bei DB-Entwicklungsprojekten – Teil 2".

25. Oktober 2023

Beim Regionaltreffen Osnabrück/Bielefeld/Münster heißen die beiden Vortragsthemen "Docker container in a nutshell (for DBAs)" und "Hübsch verpackt – Nutzung der von Oracle bereitgestellten Datenbank-Container unter Docker und Kubernetes".

26. Oktober 2023

Beim Regionaltreffen NRW in Wuppertal stehen gleich drei Vorträge auf der Agenda. "Privilege Access Management im Datenbank-Kontext mit BeyondTrust", "Service-Leveln für Datenbanken" und "Umsetzung von Service-Leveln mit Cohesity".

26. Oktober 2023

Das Regionaltreffen Dresden findet statt. Die zwei Referenten Uwe Czarnikow und Peter Ramm sprechen über "23c new Features" und "Aus der Praxis: "Richtige" Nutzung eines function based Index".

9. November 2023

Auf dem Regionaltreffen München/Südbayern berichtet Carsten Czarski über "REST Data Sources in APEX 23.1: Stand der Dinge und Ausblick auf 23.2". Daneben stehen zwei APEX-Erfahrungsberichte auf dem Programm.

10. November 2023

In der WebSession mit Andreas Jordan dreht sich alles rund um das Thema "Power Shell für den Oracle DBA".

16. November 2023

Beim Regionaltreffen Hamburg erwarten die Teilnehmenden die Vorträge "Features der 23.2-Version" von Florian Grasshoff und "Offline-Fähigkeit von APEX-Anwendungen" von Kai Donato.

20. und 21. November 2023

Die diesjährigen European NetSuite User Days finden am 20. und 21. November im NCC in Nürnberg statt. Die 187 Teilnehmerinnen und Teilnehmer erleben zwei Tage voller erstklassiger Beiträge rund um das cloudbasierte ERP-Sy-

stem NetSuite. Im Vergleich zum erfolgreichen Vorjahr ist das Programm noch einmal deutlich gewachsen: Die Anzahl der Sessions ist um 50 Prozent angestiegen und auch bei den interessierten NetSuite-Partnern vor Ort gibt es einen deutlichen Zuwachs zu vermelden. Der DOAG-Themenverantwortliche NetSuite Peter von Zimmermann stellt fest: "Es waren sehr intensive Tage in Nürnberg und man kann wirklich sagen, es ist uns gelungen, die NetSuite-Community zu versammeln. Insbesondere ist uns der Sprung von Deutschland nach Europa gelungen".

21. bis 24. November 2023

Die DOAG 2023 Konferenz + Ausstellung öffnet vom 21. bis 24. November im Nürnberg Convention Center (NCC) ihre Pforten für die rund 1800 Teilnehmenden. Gestartet wird mit einem Thementag, der 18 Themen in den Fokus rückt. Am Mittwoch und Donnerstag finden die beiden klassischen Konferenztage statt. Hochkarätige Vorträge zu den vier Hauptthemenbereichen "Datenbank & Infrastruktur", "Development & Middleware", "Data Analytics & KI" sowie "Strategie & Softskills" stehen auf der Agenda. Daneben erfreuen sich die Besucherinnen und Besucher an einem unterhaltsamen Rahmenprogramm, einem attraktiven Abendevent sowie spannenden und spaßigen Community-Aktivitäten wie zum Beispiel einer Unconference und Quizspielen. Zum Abschluss gibt es noch einen Schultag mit neun Workshops zur Vertiefung des Gehörten. DOAG-Vorstandsvorsitzender Björn Bröhl zeigt sich begeistert: "Wow – viele spannende Gespräche und ein reger persönlicher Austausch der Teilnehmenden. Das DOAG-Netzwerk hat sich im Rahmen der K+A wieder getroffen und mit neuen Themen wie KI erweitert".

22. und 23. November 2023

Premiere! Im Rahmen der DOAG Konferenz + Ausstellung findet im Nürnberg Convention Center (NCC) erstmals die Konferenz KI NAVIGATOR zur Praxis der KI in IT, Wirtschaft und Gesellschaft statt – ein gemeinsames Event von DOAG, Heise Medien und de'ge'pol. Im Fokus stehen die praktische Anwendung von KI und ihre Auswirkungen auf Technologie, Wirtschaft und Gesellschaft. Die 500 Besucherinnen und Besucher erwarten zwei Tage mit geballtem technischem, wirtschaftlichem und politischem Wissen mit vertieften Informationen zur Entscheidungsfindung und Orientierung für Unternehmen, Verbände, Entscheiderinnen/Entscheider und Verantwortliche. Die Teilnehmenden erlangen durch zwei spannende Keynotes von Sascha Lobo "Wie KI die Welt verändert und was das für IT-Profis bedeutet" und "Big Dataismus: Mit gesundem Menschenverstand durch die Digitalisierung" von Vince Ebert spannende Ein- und Ausblicke in das Thema der Stunde.

Der DOAG-Vorstand KI und Leiter der KI Community Oliver Szymanski zieht ein positives Fazit: "Insgesamt hat sich die KI NAVIGATOR als herausragende Plattform für den praxisorientierten Austausch über Künstliche Intelligenz gezeigt. Die gelungene Verknüpfung von verschiedenen Themenbereichen, die inspirierenden Keynotes und die interaktiven Elemente trugen gemeinsam dazu bei, die Teilnehmer für die Chancen und Herausforderungen von KI zu sensibilisieren und den Weg für deren erfolgreiche Integration in die Praxis zu ebnen".

JavaLand

on demand



JavaLand 2023 verpasst?

Jetzt On-demand-Ticket buchen und Vortragsaufzeichnungen anschauen!



Alle Angebote im On-demand-Ticket-Shop



ORACLE

„JavaScript hat sich definitiv als ‚Mainstream‘-Programmiersprache durchgesetzt und wird nicht mehr so schnell wieder verschwinden.“

Martin Meyer, Redaktionsleiter des Red Stack Magazins, unterhielt sich mit Gerald Venzl, Senior Director im Product Management bei Oracle in Kalifornien, über JavaScript, Developer Experience, SQL-Standards, Multilingual Engine (MLE), JSON-Funktionen und AI in der Anwendungsentwicklung.

Bitte stellen Sie sich kurz unseren Lesern vor. Wer sind Sie und mit was beschäftigen Sie sich?

Hallo, mein Name ist Gerald Venzl. Ich bin gebürtiger Österreicher und seit etwas mehr als 8 Jahren nun als einer der Produkt Manager für die Oracle-Datenbank im ehemaligen Hauptquartier in Kalifornien tätig. Dabei beschäftige ich mich mit vielen Themen rund um die Oracle-Datenbank und speziell auch mit der Benutzerfreundlichkeit unserer Software.

Auf der Oracle Cloud World wurde die „Developer Experience“ der Database 23c sehr oft erwähnt. Was hat es damit auf sich?

Während der Entwicklung von Oracle Datenbank 23c haben wir neben vielen anderen Bereichen auch den Bedürfnissen der Entwickler viel Aufmerksamkeit geschenkt. Von neuen revolutionären Technologien wie JSON Relational Duality und Domains bis hin zu lang gewünschten Verbesserungen, wie zum Beispiel den Table Value Constructor und Boolean Datentypen – 23c bringt das alles und mehr. Aber wir haben nicht bei neuer Funktionalität aufgehört. Mit 23c kommt auch eine neue kostenlose Oracle Database Free (oracle.com/oracledbfree), der Nachfolger der lang-geliebten Oracle Database Express Edition (XE). Und was mich besonders stolz macht, ist, dass wir zum ersten Mal in der Geschichte von Oracle die Oracle-Datenbank für die Entwickler zuerst veröffentlicht haben: mit Oracle Database 23c Free – Developer Release. Das haben wir wiederum gemacht, damit die Entwickler nicht mehr auf lange Oracle-Datenbank-Upgrade-Planungen warten müssen, bis sie ihre Hände an die neue Ver-

sion bekommen. Stattdessen können Entwickler und natürlich auch andere Benutzer sich schon seit einem halben Jahr mit den Neuerungen gerade im Applikationsentwicklungsbereich vertraut machen und sind dann „startklar“, wenn die Produktionssysteme auf 23c gehen, was wiederum den Kunden zugutekommt. Neben der Datenbank selbst gibt es auch noch andere Neuerungen rund um die Datenbank, die für Benutzer von Interesse sind, zum Beispiel ein neues Fehlermeldungs-Hilfsportal (docs.oracle.com/error-help/db/) oder die neue Oracle REST Data Services Version, die nun auch GraphQL unterstützt. Es tut sich viel rund um die Entwicklerfunktionalitäten und die Benutzerfreundlichkeit und es wird in den nächsten Jahren noch viel mehr kommen.

Unter Entwicklern wird heiß diskutiert, ob Geschäftslogik in die Datenbank gehört oder nicht. Wie sehen Sie das? Wo gehört Geschäftslogik Ihrer Meinung nach in die Datenbank, wo nicht mehr?

Ganz einfach: Die Logik gehört dorthin, wo sie am meisten Sinn macht. Ich würde keinem Kunden empfehlen, der viele Anwendungen gegen ein zentrales Datenmodell laufen hat, komplexe Datenvalidierungslogik in jeder Anwendung neu zu programmieren. Das ist nicht nur fehleranfällig, sondern auch reine Zeitverschwendung, wenn die Logik einmal zentral implementiert ist und nahe zu den Daten aufgerufen werden kann. Auf der anderen Seite würde es mich ärgern, wenn eine Webseite die Überprüfung einer validen E-Mail-Adresse durch den Webserver an die Datenbank schickt und ich darauf warten muss, obwohl das mein Browser direkt lokal auch machen kann. Dann gibt es na-

türlich noch die Debatte, was Geschäftslogik ist und was nicht, und ob Datenvalidierung da überhaupt mit dazu gehört. Ich werde bei solchen Debatten immer schnell ungeduldig. Es würde unserer Industrie viel Gutes tun, wenn wir aufhören uns immer wie manch religiöse Fanatiker zu verhalten. Viele Wege führen zum Ziel und unter der Haube wird so oder so nur Maschinencode auf einer CPU ausgeführt, und der ist es wiederum egal, von wo der Maschinencode kommt. Meiner Meinung nach ist es viel besser, sich auf die Geschäftsanforderungen zu konzentrieren und wie man Mehrwert für die Benutzer erzeugt, anstatt sich auf irgendwelche langen Debatten einzulassen, ob A oder B der einzig richtige Weg sei.

Oracle arbeitet an der Entwicklung des SQL-Standards mit. Gibt es besondere Schwerpunkte für Oracle?

Oracle ist seit der Gründung des SQL-Standards 1986 aktiv involviert. In der letzten Standards-Version, die gerade im Juni dieses Jahres rauskam (SQL:2023), war SQL/PGQ (Property Graph Queries) ein großes Thema. Für die neue Version müssen wir erst mal sehen, wohin es mit AI, Vektor-Datenbanken und anderen gerade aktuellen Trends geht. Der Standard braucht ja immer auch ein paar Jahre, bis es eine neue Version gibt und bis dahin kann sich vieles verändern.

Was ich manchmal schade finde, ist, dass es oft übersehen wird, wie gut wir es eigentlich mit SQL haben. Wenn man sich da so manche proprietären NoSQL-Datenbanken-API-Implementationen zu Gemüte führt, merkt man schnell, was für ein Luxus es eigentlich ist, wenn Datenbanken alle "SELECT * FROM table WHERE x=y;" verstehen und dabei auch das gleiche Resultat liefern.

Außerhalb des Standards ist es leider auch schwer zu sehen, wie sehr Oracle im Standard mit aktiv ist. Es macht mich besonders stolz, dass Oracle weiterhin viele neue Features proaktiv an den Tisch bringt und immer wieder eine aktive Führungsrolle einnimmt, wie es zum Beispiel mit SQL/JSON und SQL/PGQ der Fall war. Das spiegelt sich dann natürlich auch im Produkt wider. So war die im April veröffentlichte Oracle Database 23c Free – Developer Release schon SQL/PGQ-konform, obwohl der Standard erst drei Monate später veröffentlicht wurde.

Die Oracle Database enthält die Multilingual Engine (MLE). Welche Rolle sieht Oracle für die Programmierung mit JavaScript in der Datenbank?

JavaScript ist momentan die, oder zumindest eine der, beliebtesten Programmiersprachen am Markt, je nachdem welcher Umfrage man Glauben schenkt. Ich kenne nur sehr wenige Programmierer, die gerne ihre Programmiersprache(n) wechseln. Ich nehme mich da selbst nicht mit aus, ich schreibe (leider nur mehr gelegentlich) SQL, PL/SQL, Python und Java (und kein JavaScript, aber bitte sagen Sie das nicht weiter ;-)).

Ich denke nicht, dass mich irgendwer dazu bewegen kann, jetzt doch lieber Scala oder Kotlin zu verwenden. Warum also versuchen JavaScript-Entwickler zu konvertieren, anstatt es ihnen zu ermöglichen, JavaScript-Code in der Datenbank direkt auszuführen? Und damit bekam auch der eine oder andere APEX-Entwickler die langersehnte Ausführungsmöglichkeit von JavaScript.

Können Sie etwas zur Roadmap der Multilingual Engine sagen?

Es gab schon einen Grund, warum wir sie Multilingual Engine benannt haben, und der Grund ist nicht, weil damals GraalVM noch nicht GraalVM hieß. Multilingual Engine verwendet, wie gerade erwähnt, die GraalVM und diese kann mehrere Programmiersprachen ausführen. Unser Ziel ist es, mit der Zeit eine Engine zu liefern, die viele Programmiersprachen versteht, und es damit Entwicklern zu ermöglichen, Code in der Datenbank in ihrer Lieblingsprogrammiersprache zu schreiben. Letztlich ist es nicht nur der CPU, sondern auch der Datenbank egal, welche Programmiersprache die Instruktionen liefert. Deswegen auch der Name Multilingual Engine, „lingual“ im Sinne von Programmiersprachen. Manchmal höre ich die Frage, ob JavaScript/Multilingual Engine früher oder später PL/SQL ersetzen wird. Da kann ich Sie beruhigen, das ist definitiv nicht der Fall! Es geht mit PL/SQL weiter und wir haben ja auch in 23c einige coole neue Features eingebaut, wie etwa JSON Constructor und PL/SQL Aggregate Types.

Die Vision für die Multilingual Engine ist dabei, früher oder später eine Engine zu haben, die SQL, PL/SQL, JavaScript, Python, Java, etc. ausführen kann, anstatt für jede Programmiersprache eine neue Engine einzuführen, wie wir es für PL/SQL und Java gemacht haben.

Und wenn wir es dann einmal so weit haben, sind auch die Diskussionen bezüglich der SQL-PL/SQL Engine context switches eine Sache der Vergangenheit, da unter der Haube alles in einer einzigen Engine ausgeführt wird. Aber das wird noch eine Weile dauern, jetzt gucken wir uns erst mal Python an.

JSON-Funktionen sind seit einiger Zeit in der Datenbank verfügbar. Wie stark werden diese in konkreten Projekten aufgenommen?

Seit einiger Zeit ist vielleicht etwas leicht ausgedrückt. JSON-Unterstützung in der Oracle-Datenbank gibt es schon seit 9 Jahren – die kam mit 12.1.0.2 damals im Juli 2014. Das ist länger als ich als Produkt Manager tätig bin, nur um das mal so in den Raum zu stellen. JSON selbst ist superpopulär, wenn es um Datenaustausch zum Beispiel über REST und vieles mehr geht. Da ist es auch nicht schwer zu erkennen, dass JSON auch außerordentlich populär in Datenbanken geworden ist. Manche erinnern sich an die XML-Datenbanken-Zeiten, ich sehe es allerdings nicht ganz so drastisch. Dennoch würde ich es jedem Oracle-Benutzer empfehlen, sich zumindest ein bisschen mit der JSON-Funktionalität vertraut zu machen. Von Klein bis Groß, viele verwenden heutzutage JSON in der Oracle-Datenbank, oft auch im Zusammenhang mit relationalen Tabellen anstatt als Ersatz. Ich denke da zum Beispiel an die in Deutschland angesiedelte retraced GmbH, die super innovative Sachen bezüglich Versorgungsketten-Verfolgbarkeit macht und auch 2020 den Deutschen Nachhaltigkeitspreis gewonnen hat. Da basiert alles auf der Oracle Autonomous Database und viele der Daten werden über die JSON-Funktionalität verarbeitet. Wir sehen auch mehr und mehr Migrationen von MongoDB zur Oracle-Datenbank, weil MongoDB früher oder später mit dem Datenvolumen einfach nicht mithalten kann. Dabei kommt natürlich auch unsere neue MongoDB-kompatible API mit ins Spiel, die die Migrationen sehr erleichtert.

Durch die Verbesserung der Developer Experience und die breitere Aufstellung mit Features wie JSON-Support, Graph-Abfragen, Vektor-DB und MLE dürfte die Datenbank für mehr Entwickler interessant sein. Wie soll diese Gruppe an Oracle herangeführt werden?

Je nachdem wie einer am leichtesten lernt, da gibt es viele Möglichkeiten. Ich empfehle immer allen auf oracle.com/oracledbfree zu gehen und sich eine Oracle Database 23c Free herunterzuladen. Die gibt es auch als Docker/Podman-Container oder Virtual-Box VM. Wer nicht unbedingt die Software auf dem Rechner haben und dennoch „hands-on“ sein will, der kann auch die Oracle LiveLabs (oracle.com/livelabs) nutzen. Unsere ACEs und Produkt-Manager haben auch schon viel über die neuen Funktionalitäten gebloggt, Videos gemacht und natürlich wird da noch vieles mehr kommen. Ich selbst bin auch immer ein Fan des „New Features Guide“ in der Datenbank-Dokumentation, der relativ schnell aufzeigt, was sich alles getan hat und, was für den individuellen Benutzer von Vorteil sein kann.

Welche Potenziale sehen Sie in der Nutzung von AI in der Anwendungsentwicklung? Und: Welche Risiken sehen Sie?

Mehr mit weniger tun. AI hat ein großes Potenzial, die Anwendungsentwicklung extrem zu erleichtern. Ob Code- oder Dokumentations-Generierung, Fehlersuche, Sicherheitsrisiken erkennen oder automatische Behebung von Problemen, AI kann da bei vielem helfen. Bei Oracle arbeiten wir intern schon länger mit AI, vor allem in unserem Autonomous Database Cloud Service.

Da gibt es schon vieles, was AI erleichtern oder gar übernehmen kann. Die wirklich großen Innovationen werden aber weiterhin von kreativen Köpfen kommen, denke ich, zumindest in absehbarer Zeit. Bezüglich der Risiken kann ich sagen, das sich einige Leute getroffen habe, die Angst haben, bald ihre Arbeit an AI zu verlieren. Ich denke davon sind wir weit entfernt. Die größten Risiken, die ich momentan sehe, sind auf der einen Seite, dass man sich vielleicht ein bisschen zu sehr auf AI verlässt. Die künstliche Intelligenz lügt einen schon mal ab und zu überzeugend an, weil einfach sich widersprechende Datenpunkte (noch) nicht erfasst wurden. Auf der anderen Seite sehe ich gerade einen Trend, der dem von Blockchain nahekommmt. Alles und jeder muss jetzt gerade AI nutzen, ob dann doch die Mega-Vorteile für alle dabei rausspringen, wird sich erst erweisen müssen.

Welche Trends sehen und welche Entwicklungen erwarten Sie für die Zukunft hinsichtlich JavaScript und Development?

JavaScript hat sich definitiv als „Mainstream“-Programmiersprache durchgesetzt und wird nicht mehr so schnell verschwinden. Auch weil wir natürlich mittlerweile alle über das Internet verbunden sind und JavaScript ja quasi als Browser-Programmiersprache für das Internet geboren wurde. Aber so wie jede höhere Programmiersprache wurde sie für bestimmte Zwecke konstruiert und hat damit genauso Vor- und Nachteile. Diesbezüglich sehe ich eine breite Zukunft für JavaScript und JavaScript-Entwickler neben den Python-, Java-, PL/SQL- und vielen anderen Entwicklern.



GERALD VENZL

Gerald Venzl ist ein Senior Direktor für Produkt-Management für die Oracle-Datenbank. Vor seiner Zeit bei Oracle war Gerald als Entwickler, DBA, Performance Engineer, Software-Architekt, Consultant und Enterprise-Architekt tätig, was ihm einen umfassenden Überblick in viele Bereiche der IT-Branche gibt. Seine Leidenschaft liegt darin, wie man Systeme baut, die flexibel sind, aber auch die Anforderungen der Kunden erfüllen. Gerald ist auch ein Mitglied des ISO-SQL-Standard-Komitees und entwickelt weiterhin Software in seiner Freizeit.



Ersetzt Qwik alle bisherigen JavaScript Frameworks?

Robert Maier-Silldorff, iteratec

Metaframeworks wie Next.js, SvelteKit oder Astro sind zurzeit in aller Munde. Diese übernehmen gewisse Aufgaben rund um die Hydratation, wodurch die Startup-Time beschleunigt wird und User schneller mit Webseiten interagieren können. Qwik treibt diesen Ansatz weiter und führt Resumable ein. Doch was versteht man unter Resumable und welche Konzepte stehen noch hinter diesem Framework?



Abbildung 1: Hydration vs. Resumable [1] (Quelle: Builder.io)

```

export const ComplexCounter = component$(() => {
  const store = useStore({ count: 0 });

  return (
    <>
      <button onClick$={() => store.count++}>increment</button>
      <p>{store.count}</p>
    </>
  );
});

```

Abbildung 2: Beispiel einer Qwik-Komponente (Quelle: QwikDev)

Mit Single Page Applications (SPAs) erfolgt das Rendering clientseitig. Das heißt, dass die HTML-Seite einschließlich des verwendeten JavaScripts heruntergeladen werden muss, bevor mit einer Webseite interagiert werden kann. Fachleute nennen diesen Zeitpunkt den First Meaningful Paint (FMP). Um den FMP zu verbessern, kommt in anderen Frameworks Hydration zum Einsatz. Hydratation ist der Prozess, in dem die *Event Listeners* zum DOM hinzugefügt werden, der *Komponenten-Baum* aufgebaut und der *Application State* wiederhergestellt wird. Qwik vereinfacht diesen Prozess, indem alle notwendigen Informationen bereits im HTML vorhanden sind, wodurch cli-

entseitig keine Anpassungen notwendig sind. Dieses Konzept wird als *Resumable* bezeichnet (siehe *Abbildung 1*).

Historie der Webentwicklung

Um Qwik und die Konzepte hinter diesem Framework zu verstehen, müssen wir uns die Geschichte der Webentwicklung anschauen. Diese kann wie folgt gegliedert werden:

- **1. Generation:** Rendering am Server wie z. B. PHP und jQuery
- **2. Generation:** SPA Frameworks wie Angular oder React

- **2,5. Generation:** Hydration mittels Metaframeworks wie z. B. Next.js, SvelteKit oder Astro
- **3. Generation:** HTML-First-Konzept, JavaScript wird nur geladen, wenn es für die aktuelle User-Aktion benötigt wird, Qwik

Was ist Qwik?

Qwik ist ein JavaScript-Framework, welches nur das absolute Minimum an JavaScript lädt. Nur 1KByte an JavaScript wird benötigt, damit eine Webseite interaktiv ist beziehungsweise der FMP erreicht ist. JavaScript selbst wird nur geladen, wenn

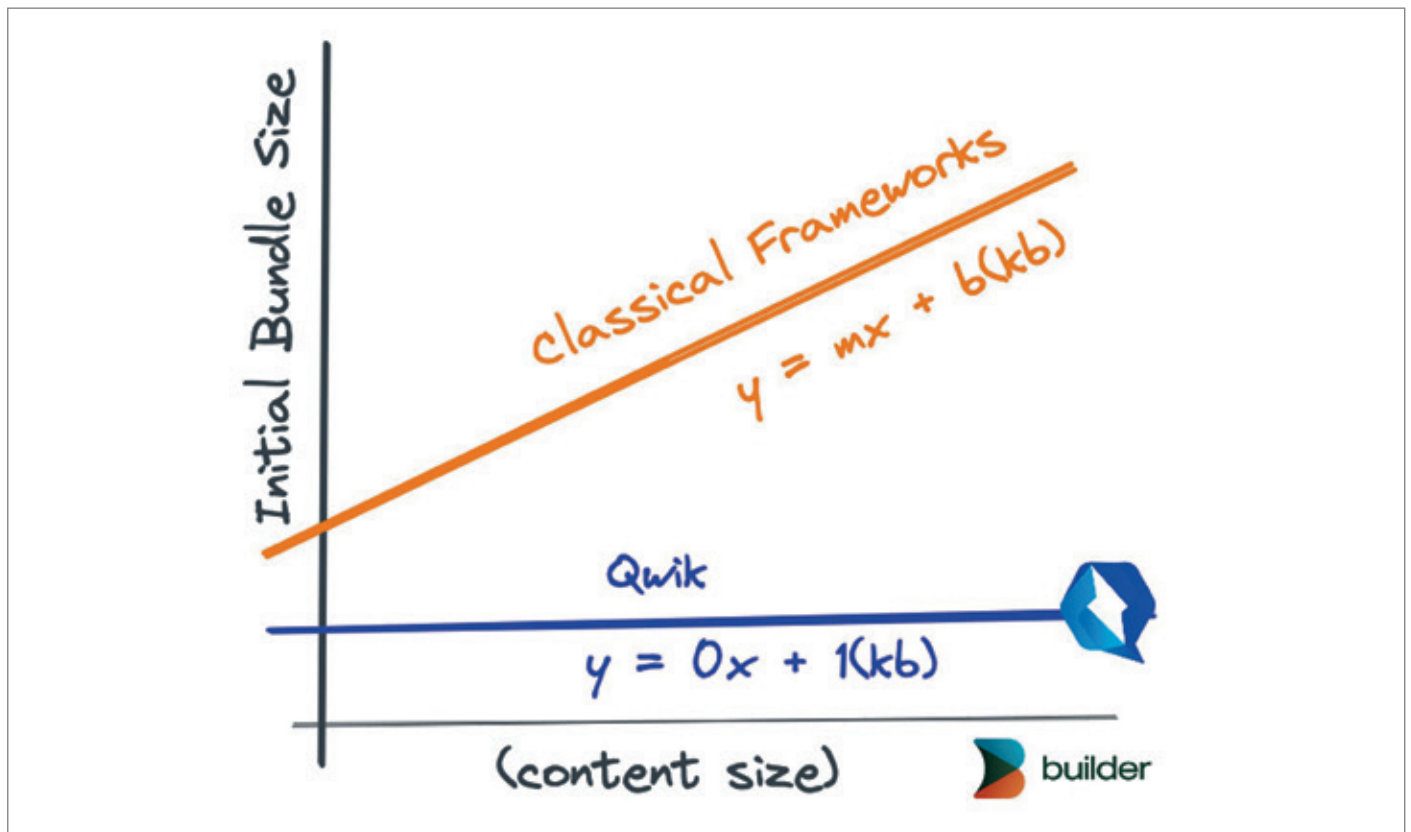


Abbildung 3: Bundle Size im Verhältnis zur Größe des Projekts [2] (Quelle: QwikDev)

es für eine User-Aktion benötigt wird. Dieses Konzept wird als Progressive bezeichnet. Im folgenden Beispiel (siehe Abbildung 2) wird erst beim Klick auf den Button *increment* das benötigte JavaScript geladen und der Counter in weiterer Folge erhöht werden. Durch das $\$$ -Zeichen weiß Qwik, dass JavaScript erst on-demand geladen werden soll.

Dadurch wächst die Bundle Size in weiterer Folge nicht linear zur Größe des Projekts, sondern bleibt, wie im unteren Diagramm ersichtlich, konstant (siehe Abbildung 3).

Reactivity in Qwik

Die Konzepte *Resumable* und *Progressive* sind das große Asset von Qwik. Wie schafft es Qwik jedoch, dass der Komponenten-Baum schnell und effektiv aufgebaut wird, beziehungsweise wie sieht die Reactivity in Qwik aus? Qwik verwendet Signals, welche eine fein granulare Reactivity ermöglichen, sodass nur jene Komponenten-Bereiche neu gerendert werden, welche sich geändert haben. Anders als bei anderen Frameworks wird dafür kein Compiler benutzt

oder werden Listeners registriert, sondern es kommen Proxies zum Einsatz. Diese Proxies erzeugen serialisierbare Subscriptions, welche für das erneute Rendering benutzt werden. Im angeführten Beispiel (siehe Abbildung 4) wird mit *useStore* solch ein Proxy erzeugt. Sobald der User den Button klickt, wird das entsprechende JavaScript heruntergeladen und der betroffene Teil der Komponente neu gerendert. Wichtig ist, dass nur jene Programmteile neu geladen werden, welche von der Aktion betroffen sind. Angenommen, es gäbe in der angeführten Komponente noch einen zweiten Counter, dann würde dieser nicht neu geladen werden.

Syntax in Qwik

Wie bereits aus den angeführten Beispielen ersichtlich, ist die Syntax in Qwik sehr stark an React angelehnt. Auch die Component Templates gleichen React beziehungsweise wird JSX verwendet. Es wird jedoch nur dieselbe Syntax genutzt. Die darunterliegenden Konzepte sind anders als in React. In Qwik kommt kein Virtual DOM zum Einsatz.

Qwik unterscheidet sich zu anderen JSX-Frameworks wie folgt [3]:

- Komponenten werden immer mit *component\$* erstellt
- *useSignal* kann für einen reactive State verwendet werden
- Event Handlers werden mit dem $\$$ -Suffix deklariert
- Bei *<input>* wird *onInput\$* statt *onChange* verwendet
- HTML-Attribute werden bevorzugt, *class* statt *className*

Doch wie erfolgt das Binding mittels Signals und wie kann eine Bedingung für das Rendering angegeben werden? Wie in Abbildung 5 ersichtlich, erfolgt das Binding über *bind:value* und mittels *isVisible.value* kann eine Bedingung angegeben werden.

Migration zu Qwik

Die beschriebenen Konzepte sind intuitiv und verbessern nicht nur die Start-Up Time, sondern auch die Runtime-Performance. Doch wie können bestehende Applikationen zu Qwik mi-


```

export const Counter = component$(() => {
  const store = useStore({ count: 0 });

  return <button onClick$={() => store.count++}>{store.count}</button>;
});

```

Abbildung 4: Counter Komponente in Qwik (Quelle: QwikDev)

```

export default component$(() => {
  const vorname = useSignal('');
  const isVisible = useSignal(false);

  return (
    <form>
      <input type="text" bind:value={vorname} />
      <input type="checkbox" bind:checked={isVisible} />

      {isVisible.value ?
        <div>Vorname: {firstName.value}</div> :
        <div>Nicht sichtbar</div>
      }

    </form>
  );
});

```

Abbildung 5: Qwik-Komponente mit Binding und Condition (Quelle: QwikDev)

griert werden? Für React-Applikationen wurde Qwik React eingeführt. Dieses ermöglicht es, React innerhalb von Qwik zu nutzen. Hierfür kann die Funktion `qwikify$` benutzt werden. *Abbildung 6* zeigt, wie dies funktioniert.

Für alle anderen SPA-Applikationen, wie etwa Angular, müsste die gesamte Applikation zu Qwik migriert werden.

Vergleich zu Angular

Da die Person, die hinter Qwik steckt, Miško Hevery, als Vater von Angular gilt, ist es interessant, Qwik mit Angular zu vergleichen. Generell wird jede Angular-Applikation, wie in *Abbildung 7* ersichtlich, als Komponenten-Baum repräsentiert. Zusätzlich funktioniert die Reactivity an-

ders als in Qwik. In Angular wird bei einer User-Aktion der Komponenten-Baum von unten nach oben durchforstet und es werden all jene Komponenten auf *dirty* gesetzt, welche von der Aktion betroffen sind. In *Abbildung 7* sind dies alle Parent-Komponenten der gelb eingezeichneten Komponente. Als nächstes führt die Change Detection (CD) einen Tick aus,

```

// Eine bestehende React Komponente
function ReactComponent() {
  return <div>Hallo</div>;
}

// Qwik Komponente, die die React Komponente beinhaltet.
export const QReactComponent = qwikify$(ReactComponent);

```

Abbildung 6: React-Komponente in Qwik verwenden (Quelle: QwikDev)

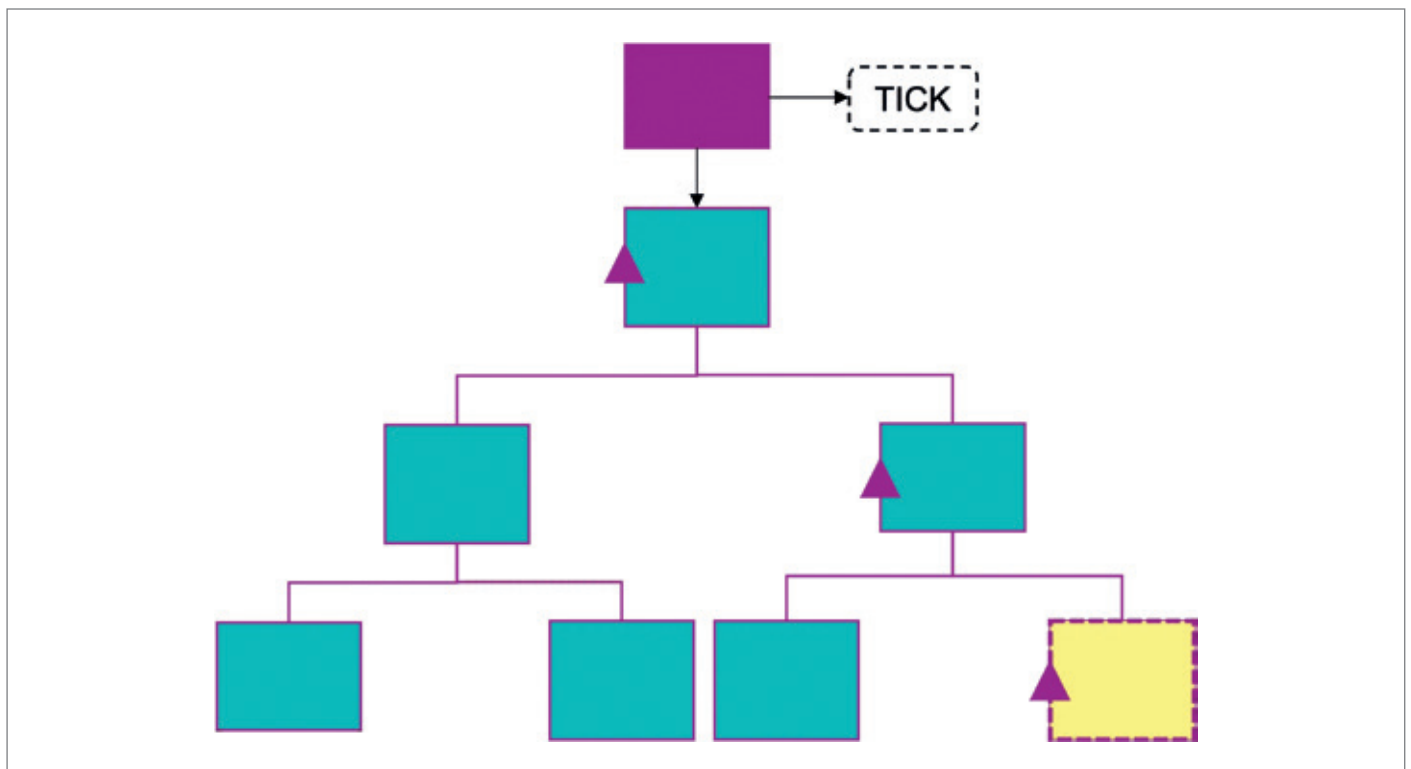


Abbildung 7: Change Detection in Angular (On-Push Change Detection Strategy)
(Quelle: <https://medium.com/bitsrc/rethinking-reactivity-in-angular-3bbe69a5a173>)

wodurch alle auf *dirty* gesetzten Komponenten neu gerendert werden.

Anzumerken ist, dass Angular mit der Version 16 Signals eingeführt hat und für zukünftige Versionen *Signal based* Components versprochen wurden, welche eine fein granulare Reactivity ermöglichen. Das heißt, dass im Falle von *Abbildung 7* nur der betroffene Template-Teil der gelb gekennzeichneten Komponente neu gerendert werden würde.

Zusätzlich setzt Angular auf Hydratation und baut diesen Ansatz immer weiter aus. Mit Angular 16 wurde non-destructive Hydratation eingeführt. Diese ermöglicht es, die vom Server gerenderte HTML-Seite weiter anzureichern, Event Listeners hinzuzufügen und den Application State aufzubauen, wodurch nicht die gesamte HTML-Seite vom Client neu geladen werden muss (*siehe Abbildung 8*). Aktuell wird an progressive und partial Hydratation ge-

forscht. Diese sollen es ermöglichen, viele kleine Bundles, im Gegensatz zu einem Großen, zu laden.

Fazit

Qwik verfolgt einen neuen innovativen Ansatz und wird zurecht als neue Generation von JavaScript-Frameworks bezeichnet. Es wird nicht nur das absolute

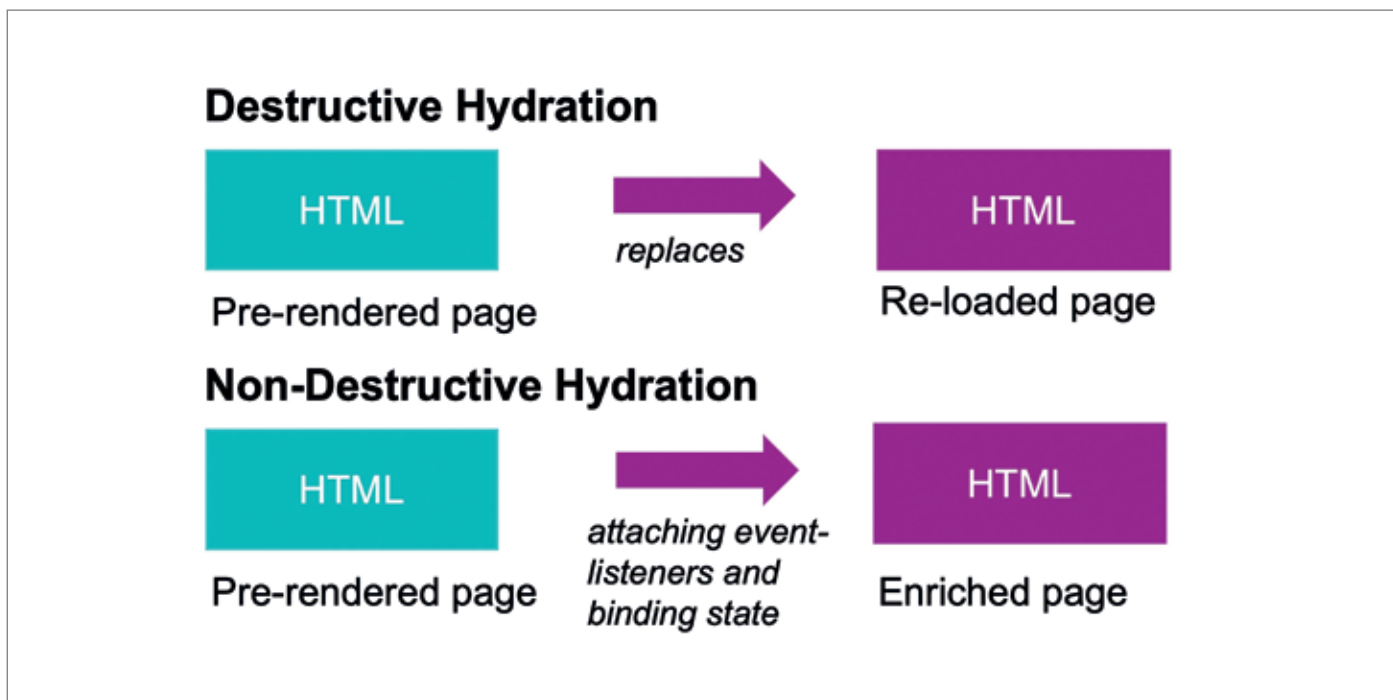


Abbildung 8: Destructive vs. non-destructive Hydration (Quelle: <https://medium.com/bitsrc/qwik-vs-angular-3e45e74853c4>)

Minimum an JavaScript geladen, sondern es ist auch nur 1KByte an JavaScript notwendig, um mit einer Webseite zu interagieren. Durch die einfache Einbindung von React-Komponenten können bereits einige Komponenten-Libraries in Qwik verwendet werden. Andere SPA-Framework-Komponenten werden jedoch nicht unterstützt. Qwik ist in keinem Beta-Status mehr, sondern kann seit Mai dieses Jahres produktiv verwendet werden. Ob Qwik alle bisherigen Frameworks ersetzt und eine neue Generation an JavaScript Frameworks einläutet, wird sich noch zeigen. Die Konzepte hinter Qwik würden dafürsprechen. Vor allem Applikationen, die zu einem großen Teil auf Mobile-Devices benutzt werden, würden sehr stark davon profitieren.

nur firmeninterne Vorträge und Workshops, sondern ist auch immer wieder als Speaker beim Vienna Angular Meetup vertreten. Zusätzlich ist er auf medium.com als Autor mit Themen rund um JavaScript und Angular unterwegs (<https://medium.com/@robert.maiersilldorff>). Best Practices und Clean Code sind ihm ein sehr großes Anliegen.

Quellen

- [1] <https://qwik.builder.io/docs/concepts/resumable/>
- [2] <https://twitter.com/QwikDev/status/1521202882124521472>
- [3] <https://qwik.builder.io/docs/components/rendering/#jsx>

Über den Autor

Robert Maier-Silldorff ist als Full-Stack Developer bei iteratec GmbH tätig, hält nicht



Robert Maier-Silldorff
Robert.Maier-Silldorff@iteratec.com



Entwicklung offlinefähiger APEX-Anwendungen

Dennis Amthor, MT GmbH

Eine APEX-Anwendung ohne Internet nutzen? Zum jetzigen Stand ist das praktisch unvorstellbar. Die Notwendigkeit einer Verbindung zur Datenbank ist bei APEX fest in der Architektur der Plattform verankert. Sämtliche Daten, die zur Nutzung der Anwendung benötigt werden, liegen in der Datenbank und werden auf Abruf über den Browser an diesen ausgeliefert. Doch mithilfe moderner JavaScript-Technologien ist auch eine Offlinenutzung möglich. In diesem Artikel werden die Hintergründe bei der Entwicklung von offlinefähigen APEX-Anwendungen betrachtet und aus Erfahrungen in diesem Bereich berichtet.



die angeforderten Daten aufgrund einer instabilen oder fehlenden Verbindung nicht ausgeliefert werden können. Neben der grundlegenden Netzabdeckung, die insbesondere im Bereich des Mobilfunks in ländlichen Gebieten noch ausbaufähig ist, kann es dabei bestimmte Situationen geben, in denen die Verbindung aufgrund der äußeren Arbeitsumstände unterbrochen wird. Folglich lassen sich webbasierte Anwendungen in solchen Bereichen nur schwer oder gar nicht einsetzen.

Anwendungsbeispiele

In manchen Fällen sorgen die Arbeitsbedingungen unvermeidbar für einen Abbruch der Verbindung. Beispiele hierfür sind Arbeitsfelder unter Tage oder auf hoher See. Die baulichen Gegebenheiten in Tunneln sowie die massiven Stahlwände eines Schiffes machen den Aufbau einer Internetverbindung unmöglich. Dennoch sind es gerade diese Anwendungsfälle, die den Einsatz von digitaler Datenerfassung interessant machen. So ließe sich beispielsweise die technische Abnahme von Bauteilen eines Schiffes ebenso komfortabel durch eine Webanwendung abbilden wie das Erfassen von geologischen Daten fernab von Sendemasten.

Status Quo

Das Standardverhalten einer APEX-Anwendung im Falle einer Verbindungsunterbrechung ist zum jetzigen Zeitpunkt eine Fehlermeldung an den oder die Benutzer:in. Diese tritt unmittelbar auf, wenn eine Aktion nicht ausgeführt werden kann und geschieht daher häufig unerwartet. Selbst wenn einzelne Daten wie zum Beispiel auf der Seite eingebundene Bilder noch im Cache des Browsers verfügbar sein sollten, gäbe es spätestens bei der Verarbeitung von Nutzdaten oder dem Laden der Seite mit dem Abruf neuer Daten einen Seitenladefehler. Die Folge: Eingaben müssen nach Wiedererlangen der Verbindung wiederholt werden, offline benötigte Daten müssen zudem im Vorhinein extern gespeichert worden sein. Sämtliche Vorteile einer webbasierten Anwendung wären somit nicht mehr gegeben.

Problemstellung

Im wachsenden digitalen Zeitalter werden nach wie vor immer mehr analoge Prozesse durch digitale Alternativen abgelöst. Dies trifft insbesondere auf den Bereich der mobilen Datenerfassung und Datenpflege zu. Wo früher noch Stift und Klemmbrett herhalten mussten, erfolgt die Eingabe heute häufig auf digitalem Wege. Die Verwendung von mobilen Webanwendungen kann dabei gegenüber clientseitigen nativen Apps von Vorteil sein, da eine system- und geräteunabhängige Entwicklung stattfinden kann und so der Wartungsaufwand reduziert wird.

Nun kann es in der Realität vorkommen, dass zu bestimmten Zeitpunkten

PWAs – Progressive Web-Apps

Mithilfe der integrierten PWA-Funktionalitäten in APEX lassen sich Webanwendungen leichter für den Gebrauch auf mobilen Endgeräten optimieren. Die Progressive Web-Apps sind damit ein Bindeglied zu den nativen Apps, indem sie scheinbar unabhängig vom Browser auf dem Gerät installiert und direkt vom Startbildschirm aus gestartet werden können. Das Herzstück einer PWA bildet dabei der Service Worker, der als eigenständiges Element im Hintergrund des Browsers agiert. Er dient als Proxy zwischen der Anwendung, dem Browser und dem Internet, indem er die Anfragen eines Nutzers oder einer Nutzerin entgegennimmt und die angefragten Daten entsprechend ausliefert (siehe Abbildung 1).

Der Service Worker stellt somit den Einstiegspunkt auf dem Weg zu einer offlinefähigen APEX-Anwendung dar. Dabei ist zu beachten, dass die Einbindung eines Service Workers allein noch keine automatische Offlinefähigkeit ermöglicht. Sämtliche Mechanismen zum Cachen der erforderlichen Seiten und der Nutzdaten müssen explizit entwickelt, beziehungsweise konfiguriert werden, damit der Service Worker im Offlinezustand die entsprechenden Daten bereitstellen kann. Im Folgenden soll dies mit einem genaueren Blick auf die zu verwendenden Technologien im APEX-Kontext näher betrachtet werden.

Service Worker

Service Worker basieren auf Standard-JavaScript-Technologien und sind daher in allen modernen Browsern verfügbar. Ein Service Worker arbeitet immer im Kontext eines bestimmten URL-Bereichs und muss daher zunächst in diesem registriert werden. In APEX kann dies über einen entsprechenden Schalter in den PWA-Einstellungen erfolgen. Einmal aktiviert, arbeitet der Service Worker im Hintergrund auf allen Seiten der Anwendung. Selbst wenn der Browser geschlossen wird, kann der Service Worker im Hintergrund Anfragen verarbeiten, um so beispielsweise Push Notifications oder andere asynchrone Aktionen durchzuführen. Die einzelnen Funktionen des

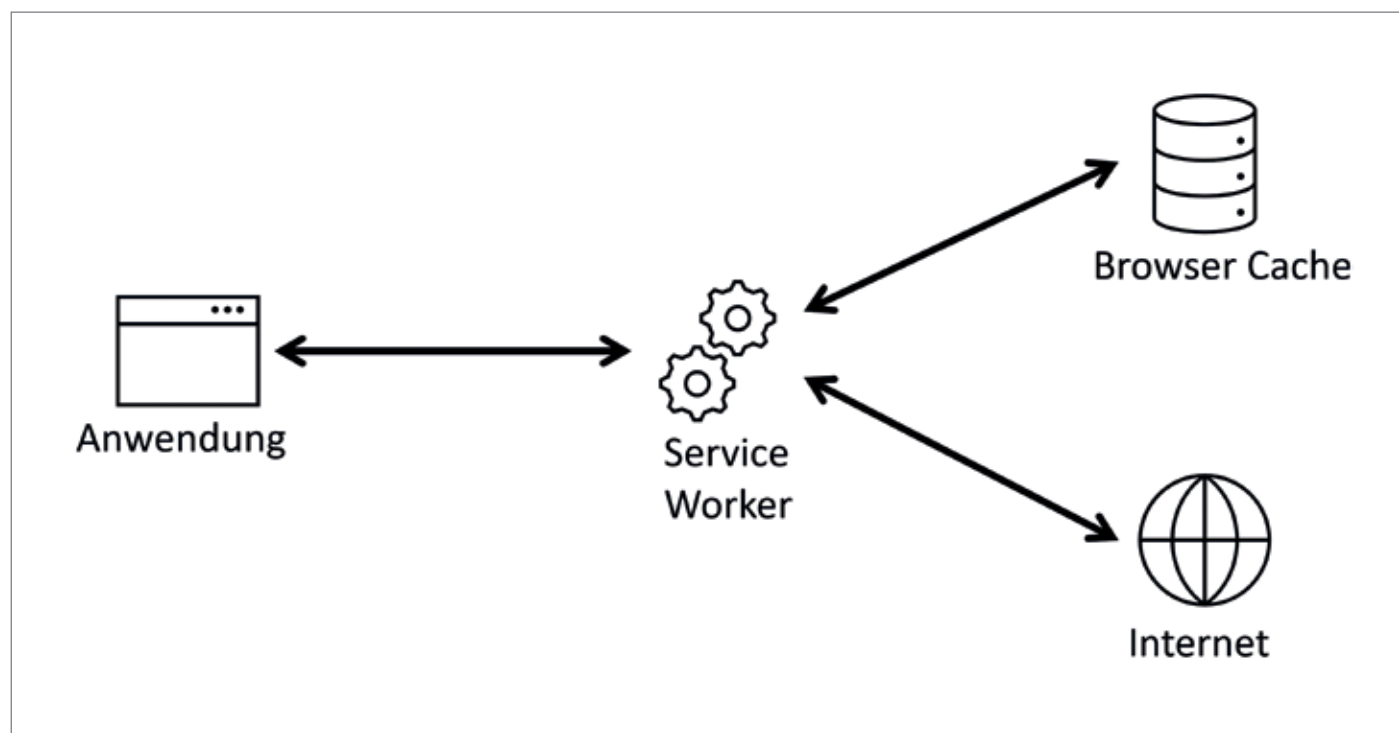


Abbildung 1: Der Service Worker vermittelt zwischen der Anwendung, dem Server und den lokalen Daten (Quelle: Dennis Amthor)

Service Workers sind dabei eventgesteuert. Neben den initialen Events bei der Registrierung sowie der Aktivierung auf einer bestimmten Seite, ist das wichtigste Event das Anfordern einer bestimmten Ressource (fetch). Im Standardfall wird hier die Anfrage einfach an den Webserver weitergeleitet und die entsprechende Ressource von dort ausgeliefert. Das können einfache Webseiten (also der angezeigte HTML-Inhalt) oder auch externe Ressourcen wie Bilder oder JavaScript-Dateien sein, die auf der geöffneten Seite eingebunden werden. Im Hinblick auf die Offlinefähigkeit ist dies der Punkt, an dem angeknüpft werden kann. Wenn keine Verbindung zum Webserver hergestellt wird und die angeforderte Ressource somit nicht erreicht werden kann, soll stattdessen im lokalen Browser Cache nach dieser Ressource gesucht werden. Damit dies funktioniert, muss der Service Worker an den entsprechenden Stellen erweitert werden, damit die Inhalte überhaupt erst im Cache abgelegt werden.

Die Standardimplementierung, die bei Aktivierung der PWA-Funktion bereits zur Verfügung steht, cacht bereits externe Inhalte, die beim Seitenaufruf abgerufen werden. Solange eine Verbindung zum Server besteht, werden diese automatisch vom Service Worker im Cache des Browsers abgelegt. Der Vorteil liegt

hier in einer gegebenenfalls schnelleren Bereitstellung der Ressource, da beim nächsten Seitenaufruf direkt aus dem Cache geladen werden kann, um die Serveranfragen zu reduzieren.

Die eigentlichen Seiteninhalte, also das gerenderte HTML der Webseiten, wird an dieser Stelle jedoch nicht automatisch gecacht. Um dies zu erzielen, muss der in APEX integrierte Service Worker erweitert werden. APEX bietet dabei die Möglichkeit, explizit einzelne Event-Methoden über sogenannte Hooks anzusprechen, über die der Standard-Code mit eigenem Code ergänzt werden kann. Alternativ kann auch ein komplett eigener Service Worker eingebunden werden, indem die entsprechende Datei in APEX hochgeladen und an dieser Stelle referenziert wird (siehe Abbildung 2).

Zum Cachen der Seiteninhalte und weiterer benötigter Daten existieren verschiedene Herangehensweisen, die im Folgenden kurz erläutert werden.

Cachen beim Seitenaufruf

Beim ersten Ansatz erfolgt das Cachen analog zu den Ressourcen, das heißt automatisch beim Seitenaufruf. Der Vorteil dabei liegt darin, dass der Cache zu jeder Zeit aktuell gehalten werden

kann. Damit immer die aktuelle Version der Seite im Cache verfügbar ist, sollte dabei ein „Server First“-Ansatz gewählt werden. Solange eine Verbindung zum Webserver besteht, wird die Seite von dort angefordert und parallel im Cache abgelegt. Sollte die Verbindung einmal abbrechen, können die entsprechenden Inhalte komplett aus dem Cache zur Verfügung gestellt werden. Bei diesem Ansatz stellt der Offlinemodus somit ein Fallback dar, um sicherzustellen, dass die Anwendung zu jeder Zeit verfügbar ist. Hierbei sollte beachtet werden, dass jede einzelne Seite nur dann gecacht wird, wenn sie einmalig vom Nutzer oder der Nutzerin aufgerufen wurde. Sollte die Verbindung beispielsweise direkt nach dem Login abbrechen, ist auch nur die Einstiegsseite offline verfügbar.

Um dieses Problem zu umgehen, kann der Caching-Prozess dahingehend angepasst werden, dass zu jedem Zeitpunkt immer alle erforderlichen Seiten in den Cache geladen werden. Da der Service Worker immer nur die aktuelle Seite kennt, empfiehlt es sich daher, in diesem Fall den Caching-Prozess auszulagern und dem Service Worker nur noch das Ausliefern der Ressourcen zu überlassen. Der Zugriff auf den lokalen Cache erfolgt dabei über entsprechende JavaScript-

APIs, die ebenfalls browserunabhängig verwendet werden können.

Welche Seiten gecacht werden sollen, kann dabei in einer Datenbanktabelle abgelegt werden. Beim Aufruf einer beliebigen Seite der Anwendung kann dann mittels eines AJAX-Prozesses in JavaScript der Inhalt dieser Tabelle ausgelesen werden und anschließend können alle angegebenen Seiten über einen fetch-Aufruf angesteuert werden. Die Ergebnisse dieser Aufrufe werden dann im Cache abgelegt. Eine Besonderheit bei diesem Ansatz zeigt sich beim Cachen der externen Ressourcen. Überlässt man das Cachen weiterhin dem Service Worker, so werden weiter ausschließlich die Ressourcen der Startseite gecacht, da die eingebundenen Ressourcen auf den Seiten, die manuell mittels fetch aufgerufen werden, nicht über den Service Worker abgerufen werden. Die weiteren gecachten Seiten müssen daher programmatisch auf eingebundene externe Dateien durchsucht und diese zusätzlich abgerufen und gecacht werden. Um die Schreiboperationen auf den Cache bei diesem Ansatz zu verringern, empfiehlt es sich, den Umfang der offline genutzten Seiten hier gering zu halten. Gerade bei größeren Anwendungen kann es sonst dazu kommen, dass bei jedem Seitenaufruf im Hintergrund für einige Zeit der Caching-Prozess läuft,

was gerade bei einer instabilen Verbindung zu Problemen führen kann.

Cachen auf Knopfdruck

Beim letzten Ansatz wird der Offlinemodus weniger als Fallback, sondern als separater Modus der Anwendung verstanden. Der grundlegende Caching-Prozess funktioniert hier analog zur vorherigen Lösung. Der Unterschied besteht darin, dass dieser nicht bei jedem Seitenaufruf angestoßen wird, sondern einmalig, wenn der oder die Nutzer:in einen entsprechenden Button in der Anwendung klickt. Nachdem alle notwendigen Seiten mit ihren Ressourcen in den Cache geladen wurden, wird zusätzlich im Cache eine Variable gesetzt, die dem Service Worker mitteilt, dass die Anwendung nun offline verwendet werden soll. Sämtliche Anfragen werden somit direkt an den Cache weitergeleitet, unabhängig davon, ob eine Verbindung besteht oder nicht. Der Vorteil hierbei besteht darin, dass deutlich klarer zwischen dem Online- und Offlinemodus der Anwendung differenziert werden kann. Gerade bei einer instabilen Verbindung wird so sichergestellt, dass dauerhaft in der Anwendung gearbeitet werden kann und keine unsicheren Zwischenstände entstehen. Sobald

der oder die Nutzer:in den Offlinemodus wieder verlässt, kann der Cache bereinigt werden und alle Anfragen laufen dann wieder direkt über den Webserver. Voraussetzung für die Verwendung dieser Lösung ist, dass entscheidbar ist, wann in den Offlinemodus gewechselt werden kann. Vor einer potentiellen Nutzung der Anwendung ohne Verbindung müssen die Nutzer:innen daher explizit das Caching anstoßen, da sonst die Verwendung der Anwendung bei Verbindungsabbruch nicht möglich ist.

Cache Storage für schnellen Zugriff

Der Cache Storage dient primär zum Zwischenspeichern von externen Ressourcen, damit sich diese bei Bedarf schneller laden lassen. Dieser wird daher für das Speichern der Seiteninhalte sowie der zusätzlich benötigten Dateien verwendet. Der Cache Storage ist in einzelne Container unterteilt, die sich frei benennen lassen. Das Matching der Einträge findet über URLs statt, die als Key für die abgelegten Inhalte dienen. Zusätzlich werden Typ, Größe sowie ein Zeitstempel, wann der Eintrag im Cache abgelegt wurde, mit angegeben. Je nach Anwendungsfall muss dabei gegebenen-

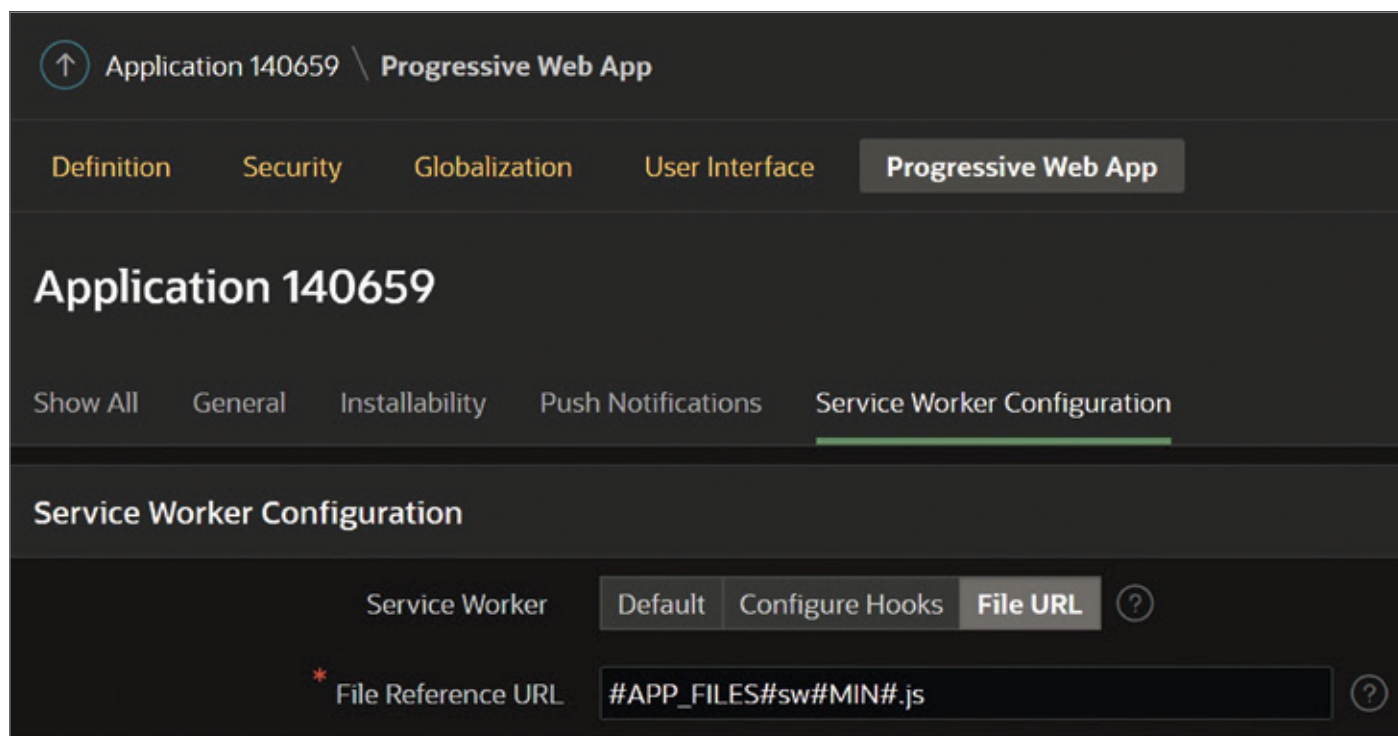


Abbildung 2: Ein eigener Service Worker kann in den Shared Components hinterlegt werden (Quelle: Dennis Amthor)

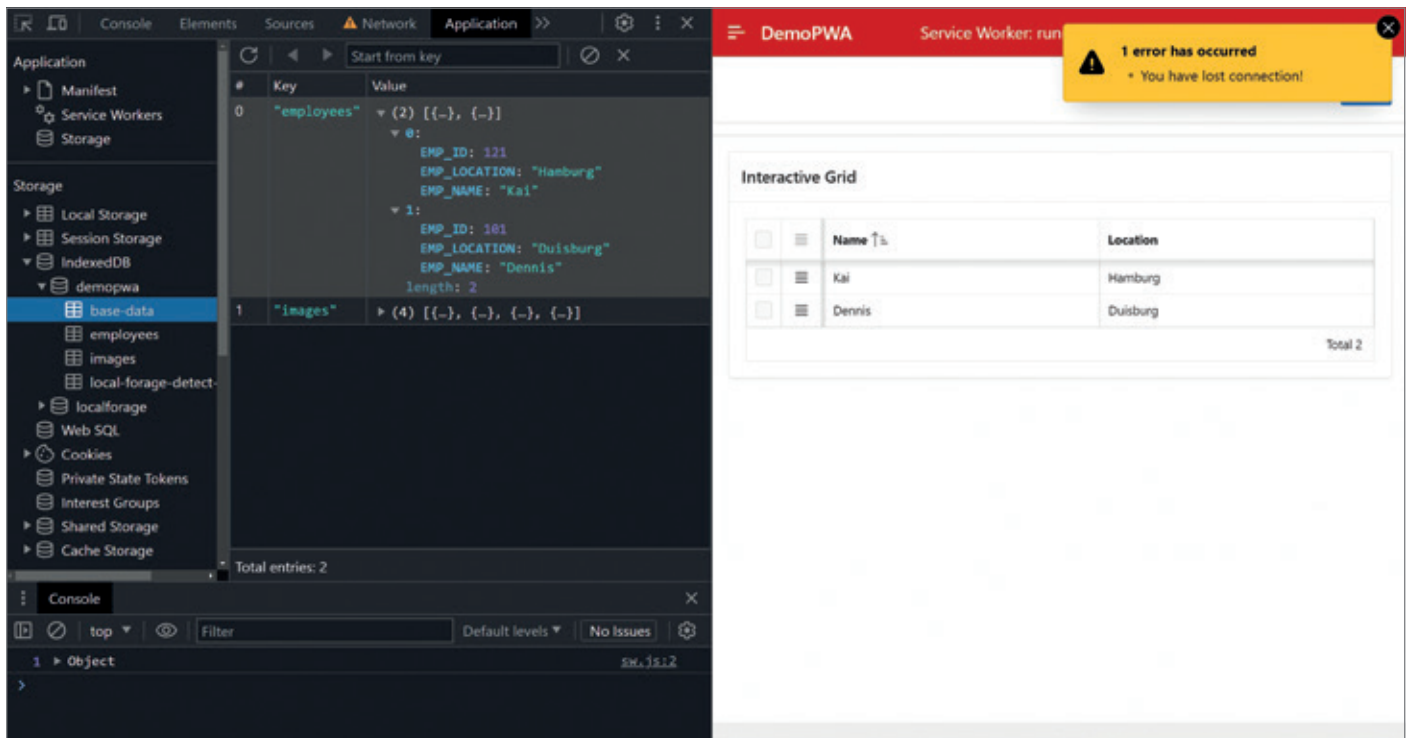


Abbildung 3: Bei Bedarf kann in der Anwendung direkt zwischen Online- und Offlinemodus gewechselt werden. Die erforderlichen Daten müssen dann vorher im Browser Cache abgelegt worden sein (Quelle: Dennis Amthor)

falls der Query String der übergebenen URL ignoriert werden, damit die korrekte Seite gefunden werden kann. Dies kann über entsprechende Funktionen in der Cache-API erfolgen.

IndexedDB für Nutzdaten

Ein weiterer wichtiger Bestandteil des Browser Caches im Hinblick auf die Offlinenutzung ist die IndexedDB. Sie bietet die Möglichkeit, große Mengen von strukturierten Daten abzulegen und diese indexbasiert abzurufen. Die Daten werden erneut in einzelnen Containern (z. B. je Anwendung) und darin wiederum in einzelnen Object Stores abgelegt. In der Praxis wird diese Speicherform für die Nutzdaten verwendet, die im Offlinemodus zur Verfügung stehen sollen. Dazu gehören sämtliche Tabellen, die in der Anwendung in Form von Reports, Charts oder Formulardaten abgerufen werden. Je nach gewähltem Caching-Ansatz werden die Daten zunächst gesammelt und anschließend mittels der entsprechenden JavaScript-API in die IndexedDB übertragen. Dabei empfiehlt es sich, je Tabelle einen einzelnen Object Store zu verwenden und darin je Zeile einen einzelnen Ein-

trag, um die Zugriffe auf die Datensätze möglichst performant zu halten. Die Daten können in Textform oder auch als JSON abgelegt werden. Bilder oder andere Dateien können nach einer vorherigen Konvertierung in Base64 Strings ebenfalls in der IndexedDB abgelegt werden.

Der verfügbare Speicherplatz im Browser Cache entspricht grundsätzlich der Größe an verfügbarem Speicherplatz auf der Hauptfestplatte des Gerätes, kann bei Bedarf jedoch auch durch Browser- oder Systemeinstellungen eingeschränkt werden. Der Cache ist zudem seiten- beziehungsweise umgebungsspezifisch, sodass nicht übergreifend auf die Daten von anderen Webseiten zugegriffen werden kann. Die im Cache gespeicherten Daten gehen nicht beim Schließen des Browsers verloren, können jedoch manuell durch den Nutzer oder die Nutzerin gelöscht werden.

Anforderungsanalyse

Bevor mit der Entwicklung einer offlinefähigen APEX-Anwendung gestartet werden kann, müssen verschiedene Faktoren berücksichtigt werden, die die Art der Imple-

mentierung sowie den Aufwand der Entwicklung bestimmen.

Verfügbarkeit

Der Aspekt der Verfügbarkeit der Anwendung bestimmt im Wesentlichen die Wahl der Caching-Lösung sowie die Entwicklung der einzelnen APEX-Seiten. Muss die gesamte APEX-Anwendung mit allen Funktionen offline zur Verfügung stehen? Dann empfiehlt es sich unter Umständen einen „Offline First“-Ansatz zu verfolgen, sofern die Anwendung nicht nur in Ausnahmefällen offline genutzt wird. So spart man sich zum Teil umständliche Sonderlösungen für einige APEX-Komponenten und Funktionen, damit diese offline verwendet werden können. Alternativ kann es auch sinnvoll sein, die Seiten für Online- und Offlinemodus separat voneinander zu entwickeln. So können auf beiden Seiten die jeweils besten Lösungen verwendet werden, ohne sich gegenseitig zu beeinflussen. Dies gilt insbesondere, wenn nur einzelne Bereiche der Anwendung offline genutzt werden sollen, oder es nur einen eingeschränkten Funktionsumfang gibt (siehe Abbildung 3).

Datenhaltung

Bei der Datenhaltung spielt der Umfang der Daten, die offline benötigt werden, eine große Rolle. Neben den Speicherbegrenzungen auf den lokalen Geräten sollte dabei auch der Zeitaspekt beim Cachen berücksichtigt werden. Da der Caching-Vorgang nur begrenzt parallelisiert werden kann, sollte hier immer beachtet werden, dass die Zeit zum Cachen der Daten in einem sinnvollen Verhältnis zur Wartezeit für die Nutzer:innen innerhalb der Anwendung steht, um den Arbeitsfluss nicht unnötig zu unterbrechen.

Ein weiterer wichtiger Punkt bei der Datenhaltung ist die Nutzungsform der Daten im Offlinemodus. Werden nur Daten angezeigt oder finden auch Änderungen an den Daten statt? Können offline neue Datensätze erzeugt werden? In diesen Fällen muss auch das Zurückschreiben der Änderungen bei Wiedererlangen der Verbindung berücksichtigt werden. Grundsätzlich ist das Lesen und Schreiben von Daten in der IndexedDB durch den keybasierten Zugriff problemlos möglich. Auch clientseitige Validierungen auf Eingabefeldern zu Plausibilitätsprüfungen sind mittels JavaScript umsetzbar. Im Hinblick auf die Datenintegrität muss jedoch immer zusätzlich eine serverseitige Validierung der Datensätze vor dem Zurückschreiben der Daten in die Datenbank erfolgen. Da die Daten im Browser Cache praktisch frei über entsprechende APIs zugänglich sind, kann allein clientseitig nicht die Korrektheit der Daten geprüft werden. Weiterhin muss der Umgang mit Konflikten beim Zurückschreiben beachtet werden. Sind konkurrierende Änderungen an einzelnen Datensätzen möglich? Was passiert mit offline geänderten Datensätzen, die bei der Synchronisierung nicht mehr in der Datenbank existieren? Neben einfachen Merging-Mechanismen, bei denen die Änderungen zeilenweise geprüft und entsprechend korrigiert werden können, empfiehlt es sich, immer die Rolle des Anwendungsadministrators mit einzubeziehen. Dieser hat im Problemfall eine umfängliche Einsicht in die aufgetretenen Konflikte und kann so in Einzelfällen das korrekte Vorgehen festlegen.

Umsetzung

Im Hinblick auf den Funktionsumfang bei offlinefähigen APEX-Anwendungen lässt sich bereits ein großer Teil von Funktionalitäten durch entsprechende Anpassungen verwenden. Die einzelnen Komponenten und Lösungen sollen im Folgenden kurz betrachtet werden.

Formulare

Der erste wichtige Punkt bei der Verwendung von Formularseiten ist, diese beim Cachen im leeren Zustand abzurufen. Die URL muss daher ohne das entsprechende Item, das zum Befüllen des Formulars herangezogen wird, aufgerufen werden. Sämtliche Items auf der Seite sind somit leer im HTML enthalten und können dann beim Laden der Seite im Offlinezustand mit den Daten aus dem Cache befüllt werden. Das Heranziehen der korrekten Daten zu einem einzelnen Datensatz wird in der Regel über die URL-Parameter realisiert, wobei im Offlinemodus grundsätzlich nicht der Session State verwendet werden kann, sodass die Parameter manuell mit JavaScript ausgelesen werden müssen. Anschließend können über die API die Daten aus der IndexedDB geholt und einzeln in das Formular geladen werden. Beim Speichern werden die Formulardaten umgekehrt wieder ausgelesen und wieder in den Cache zurückgeschrieben. Das müssen dabei nicht nur textuelle Daten sein, sondern ist auch für den Upload von Bildern und Dateien, nach entsprechender Umwandlung der hochgeladenen Daten, möglich.

Besonders beachtet werden muss bei dieser Vorgehensweise, dass sämtliche Bedingungen, die serverseitig zum Rendern der Seite verwendet werden, offline nicht funktionieren. Beispielsweise muss das Ein- und Ausblenden von einzelnen Elementen und Regionen daher clientseitig mittels JavaScript nachgebaut und beim Seitenaufbau basierend auf den geladenen Daten ausgeführt werden.

Reports

Classic und Interactive Reports können grundsätzlich auf den Offlineseiten so eingesetzt werden wie auf den Onlinesei-

ten. Beim Aufruf der Seite im Caching-Prozess werden die Abfragen für die Daten normal ausgeführt und sorgen so für ein Befüllen der Reports. Im gecachten HTML liegen die Reports somit bereits mit den geladenen Datenzeilen vor und können direkt angezeigt werden. Mehr Aufwand kommt hinzu, wenn zudem Daten offline erzeugt oder geändert werden. In diesem Fall müssen die angezeigten Daten in den Reports clientseitig manipuliert oder erweitert werden. Dies erfolgt durch gezielte DOM-Manipulation der entsprechenden Elemente basierend auf den lokalen Daten mithilfe von JavaScript beziehungsweise jQuery. Beim Interactive Report sollten dabei sämtliche Funktionen im Aktionsmenü deaktiviert werden, da diese prinzipiell ein Neurendern der Daten und somit eine Verbindung zur Datenbank erfordern.

Cards/Bilder

Card Regions funktionieren ähnlich wie Reports, indem das HTML clientseitig angepasst beziehungsweise in das DOM eingefügt wird. Auch die Anzeige von Bildern ist hierbei möglich. Dazu werden die Bilder, die als Base64 Strings im Cache vorliegen, geladen und als Attribut an entsprechender Stelle eingefügt.

Grid

Aufgrund der Verfügbarkeit des JavaScript-Modells mit der entsprechenden API im Hintergrund des Grids kann dieses auf deutlich komfortablere Weise offline verwendet werden. Empfehlenswert ist hier, wie beim Formular, das Grid leer auf der Seite zu rendern und nur die entsprechende Spaltenstruktur bereitzustellen. Im Offlinemodus können dann die anzuzeigenden Daten aus der IndexedDB geholt und an die Modell-API übergeben werden, die diese dann in das Grid einfügt. Beim Speichern können die geänderten Daten dann mit JavaScript wieder aus dem Modell ausgelesen und in die IndexedDB zurückgeschrieben werden. Auch hier gilt, dass alle zusätzlichen Funktionen im Aktionsmenü ausgeblendet werden sollten, um Fehlermeldungen bei mangelnder Verbindung zu vermeiden.

Externe Bibliotheken/ Plug-ins

Natürlich lassen sich auch externe Bibliotheken und Plug-ins in den Offlineseiten einbinden, sofern diese in sich bereits offlinefähig sind. So konnte in der Vergangenheit bereits ein Plug-in zur Erfassung von Koordinaten auf individuellen Karten eingebunden werden, welches vollständig auf der Basis von JavaScript funktioniert. Die erforderlichen Dateien für die Plug-ins sind ebenfalls auf der Seite eingebunden und werden so beim Caching berücksichtigt.

Berechtigungskonzepte

Wie bereits die serverseitigen Bedingungen können auch Berechtigungskonzepte nicht offline ausgeführt werden. Daher sollte darauf geachtet werden, dass beispielsweise Buttons entsprechend ausgeblendet werden, wenn die Benutzer:innen bestimmte Voraussetzungen nicht erfüllen, um so ungültige Aktionen zu verhindern. Die entsprechenden benötigten Informationen über den oder die aktuelle:n Benutzer:in müssen dabei ebenso wie die anderen Daten im Cache abgelegt sein. Aufgrund der beschriebenen Situation, dass die Daten im Cache frei zugänglich sind, ist dies jedoch in keinem Fall eine ausreichende Absicherung. Daher sollte immer beim Zurückschreiben von Änderungen in die Datenbank geprüft werden, ob ausschließlich valide Änderungen an den Datensätzen gemacht wurden. Hierzu können dann je nach Bedarf wieder die konfigurierten Autorisierungsmechanismen verwendet werden. Zudem sollte beachtet werden, dass offline keine Authentifizierung stattfinden kann. Der Aufruf der Seiten ist somit grundsätzlich jedem möglich, der Zugriff auf das Gerät hat. Auch hier können einzelne Prüfungen bereits clientseitig erfolgen, jedoch keine hinreichende Sicherheit garantieren.

Grenzen

Neben den beschriebenen bereits unterstützten Komponenten gibt es immer noch eine Vielzahl von Funktionen, die zum jetzigen Zeitpunkt noch nicht offlinefähig realisiert werden konnten. Dies be-

trifft primär sämtliche dynamische Funktionen, die auf die angezeigten Daten angewendet werden können. Dazu gehören insbesondere das Filtern, Suchen und Sortieren von Datensätzen beim Interactive Report und Interactive Grid. Während die Suchfunktion für das Popup LOV bereits realisiert werden konnte, müssten all diese Funktionen ebenfalls mit JavaScript nachgebaut beziehungsweise die Anfragen an den Server entsprechend durch den Service Worker abgedeckt werden. Ebenso wurde bislang noch keine Umsetzung für Diagramme realisiert, bei denen ebenfalls die Frage nach der Bereitstellung der Daten sowie das korrekte Rendern nicht ausreichend geklärt ist.

Ein weiterer Punkt, der nach wie vor ein kritischer Aspekt bei der Entwicklung ist, ist die Menge an individuellem Code für die Bereitstellung der Offlinefunktionalitäten. Neben dem Caching, das bereits durch wenig Mehraufwand umgesetzt werden kann, erfordert häufig jede einzelne Anwendungsseite eine eigene Betrachtung, je nach gewünschten Funktionen. Speziell die Verarbeitung von offline geänderten Daten ist dabei sehr individuell und anwendungsfallspezifisch zu betrachten und lässt sich nur sehr schwer verallgemeinern. Somit kann hier der Low-Code-Ansatz von APEX abseits vom Bereitstellen der groben Seitenstruktur durch die Regionen und Komponenten für den Offline-Modus nicht verfolgt werden.

Ausblick

Das Thema Offlinefähigkeit kann in Zukunft in APEX noch eine große Rolle spielen und die bisherigen Ansätze zeigen, dass eine datenbankbasierte Webanwendung und Offlinenutzung sich nicht ausschließen müssen. Das Ziel sollte dabei sein, nach und nach mehr Funktionen zu ermöglichen, um die Produktivität der Anwendung hochzuhalten. Neben den oben beschriebenen noch nicht unterstützten Funktionen steht dabei vor allem die Generalisierung im Vordergrund, um gerade bei Standardfällen simplere und dennoch individuell anpassbare Lösungen bieten zu können und so die Implementierung und den Entwicklungsaufwand reduzieren zu können. Gerade im Hinblick auf die bereits in APEX verfügbaren Optionen für PWA-Funktionalitäten

könnte dies dazu beitragen, näher am Low-Code-Standard bleiben zu können und die vorhandenen Funktionen in den Lösungen effektiv zu erweitern.

Über den Autor

Dennis Amthor arbeitet als Berater im Bereich APEX mit dem Schwerpunkt auf JavaScript bei der MT GmbH in Ratingen. Zu seinen Hauptaufgaben zählen die Entwicklung von individuellen APEX-Anwendungen, zuletzt insbesondere in Bezug auf Offlinefähigkeit. 2023 hielt er dazu einen Vortrag auf der APEX Connect in Berlin. Zudem ist er an der Weiterentwicklung von spezifischen APEX-Lösungen zum Thema Prozessmodellierung mit Flows for APEX sowie Testautomatisierung mit LCT beteiligt.



Dennis Amthor

dennis.amthor@mt-itsolutions.com



Kai Donato

kai.donato@mt-itsolutions.com

APEX *connect* by DOAG

22. - 24.04.2024

**VAN DER VALK AIRPORTHOTEL
DÜSSELDORF**



apex.doag.org



Node.js und seine Herausforderer: JavaScript Server Runtimes

Philipp Hartenfeller, MT GmbH

JavaScript im Backend ist seit Node.js seit Jahren ein profiliertes Werkzeug. Doch diese Laufzeit bekommt durch andere Anwarter mit neuen Konzepten Konkurrenz. Besonders Bun, welches jüngst in Version 1.0 erschienen ist, sorgt mit riesigen Performanz-Versprechen für große Aufregung.

JavaScript hat einen erstaunlichen Werdegang. 1995 in wenigen Tagen von Brendan Eich für Netscape entworfen, ist sie heutzutage laut der jährlichen Stack-Overflow-Studie die populärste Programmiersprache [1]. Ein Faktor für die weite Verbreitung dürfte die konstant steigende Relevanz von Webseiten sein, für deren Programmierung JavaScript fast alternativlos ist. WebAssembly könnte das in Zukunft ändern, steckt aktuell allerdings noch in den Kinderschuhen.

Damit JavaScript auf allen möglichen Endgeräten mit verschiedenen Prozessorarchitekturen funktionieren kann, wurde die Arbeit der Umwandlung des Quellcodes in Prozessoranweisung einfach auf die Browser ausgelagert. Dies bedeutet, dass Webseiten lediglich den Quelltext ausliefern und es den Browserherstellern überlassen, eine Runtime mitzuliefern, die den Quellcode interpretiert beziehungsweise „Just-in-time kompiliert“, um die Anweisungen letztendlich auszuführen. Diese Strategie war sehr erfolgreich und machte das Web allgegenwärtig. In den letzten Jahrzehnten sind eine Reihe an Endgeräten wie Smartphones, Fernseher, Kühlschränke oder Spielkonsolen erschienen, welche selbstverständlich über einen Browser verfügen und mit JavaScript umgehen können.

Ein weiterer großer Schritt, um JavaScript allgegenwärtig zu machen, war es, eine dieser Browser-Runtimes zu extrahieren, mit Betriebssystem-Schnittstellen zu versorgen und somit die Ausführung

der Sprache ohne Browser zu ermöglichen. Vorteilhaft hierbei ist, dass Code aus dem Frontend auch im Backend wiederverwendet werden kann. Außerdem können Entwickler sich ohne große Hürden an beiden Enden zurechtfinden. „Full Stack Developer“ ist seitdem eine stets weit verbreitete Jobbezeichnung.

Node.js

Eine solche serverseitige JavaScript-Laufzeit setzte Ryan Dahl im Jahr 2009 mit Node.js um. Er verwendete hierfür die aus dem Chrome-Browser stammende Engine „V8“. Wegen des Kerns entschied er sich für eine effiziente Architektur dank Ereignissteuerung, welche beim Arbeitsspeicher und Thread sparsam ist.

Der Erfolg ließ nicht lange auf sich warten, bereits im Jahr 2018 knackte das Framework die Download-Marke von einer Milliarde [2]. Diese beachtliche Verbreitung erzielte Node.js dank verschiedenster Einsatzzwecke wie Web-Server, API-Schnittstellen, Automatisierungsskripten oder Entwicklerunterstützung wie Codeanalyse oder das Bauen von Webseiten. Ein eher überraschender Einsatzzweck sind ebenfalls Desktopanwendungen. Weit verbreitete Programme wie Visual Studio Code, Microsoft Teams, Slack, und viele mehr finden sich auf fast allen Maschinen im IT-Umfeld [3].

Ein großer Faktor für diesen Erfolg ist der eigene Paketmanager „NPM“. Da

JavaScript keine eigene umfangreiche Standardbibliothek mitbrachte, war die Not groß, triviale Codeschnipsel extern einzubeziehen.

Dies machte den Paketmanager schnellstens beliebt, im Jahr 2019 knackte er bereits die Marke von einer Million verfügbarer Pakete [4]. Bei der Anzahl gibt es wahrscheinlich für fast jedes Problem eine mehr oder weniger gut gewartete Abhängigkeit. Daraus resultiert auch der Witz, dass JavaScript-Entwickler kaum programmieren, sondern Pakete nach Bedarf zusammenbasteln und versuchen, diese wöchentlich upzudaten, ohne dass Kompatibilitätsprobleme erneut alles zum Stillstand bringen.

Im positiven Sinne hat dieser Abhängigkeitswahn allerdings auch Bibliotheken wie React, Angular, TypeScript und viele mehr hervorgebracht, die einen beachtlichen Fußabdruck in der Branche hinterlassen haben.

Deno

Was kann bei dem Erfolg von Node.js schon schiefgehen? Zum Beispiel, wenn der Erfinder der Runtime auf der größten JavaScript-Konferenz einen Vortrag mit dem Titel „10 Things I Regret About Node.js“ hält. Dies tat Ryan Dahl auf der JSConf 2018 und um Worten Taten folgen zu lassen, kündigte er im selben Zug an, mit einer neuen Runtime diese Dinge zu verbessern [5]. Für den Namen arrangier-




	 Node.js	 Deno	 Bun
Schöpfer	Ryan Dahl		Jarred Sumner
Version 1.0	2015 (als v4.0, v1.0 existiert nicht)	2020	2023
JS- Engine	V8 (Chrome)		JavaScriptCore (Safari)
Lizenz	MIT		

Abbildung 1: Faktenvergleich der JavaScript Server Runtimes (Quelle: Philipp Hartenfeller)

te er die Buchstaben von Node neu zu „Deno“ an und als Logo wurde naheliegender ein Dino gewählt.

Dies schlug sofort riesige Wellen in der Branche, was nicht verwunderlich ist, wenn der Erfinder selbst große Kritik übt. Da JavaScript-Entwickler bekanntlich immer die neuesten Dinge ausprobieren wollen (ich zähle mich dazu), kam bald die Theorie auf, dass Node.js bald obsolet sei.

Letztendlich waren viele Kritikpunkte an Node von Ryan Dahl tiefergehender technischer Natur. Die beachtlichsten Änderungen, die Deno mit sich brachte, waren ein nativer TypeScript-Support und die Vereinfachungen der Syntax. Für beides gibt es ebenso in Node Lösungen, für die lediglich NPM-Module hinzugezogen werden müssen. Ein Eigenmerkmal von Deno ist allerdings eine verbesserte Sicherheitsarchitektur. Nutzer müssen nun beim Aufruf von Skripten etwa explizit den Zugriff auf Netzwerke oder das Dateisystem erlauben.

Eine weitere Änderung, welche zu Ernüchterung führte, war, dass Deno das direkte Referenzieren von Internet-Skript-URLs erlaubte und somit keinen separaten Paketmanager benötigt. Der Haken lag hierbei daran, dass Deno auch keinen Paketmanager unterstützte. Folglich waren alle jahrelang zu schätzen gelernten NPM-Module nicht mehr verwendbar,

was für viele Projekte natürlich ein riesiges Problem war.

Diesen Fehler korrigierten die Entwickler erst in Version 1.28, welche Ende 2022 veröffentlicht wurde. Seitdem kann jedes NPM-Paket via Skript-URL mit dem Präfix „npm:“ verwendet werden [6]. Inzwischen ist Deno eine echte Alternative, wenn die Unterschiede zu Node.js erwünscht sind. Dies scheint allerdings eine Nische zu sein, mit 5 Millionen Downloads im Jahr 2022 steht die Runtime noch deutlich im Schatten von Node.js [7].

Bun

Ich dachte, dass nach Deno erst einmal Ruhe beim Thema JavaScript Server Runtimes einkehren würde. Dies änderte sich schlagartig, als Jarred Sumner mit seinem Projekt namens „Bun“ (engl. etwa „kleines Brötchen“) großes Aufsehen erlangte. Anstatt wie Deno einzelne Eigenschaften von Node zu überdenken, hat Jarred Sumner mit Bun deutlich größere Visionen.

Denn statt wie Node und Deno auf die JavaScript Engine V8 aus Chrome zu setzen, hat Sumner „JavaScriptCore“ aus Safari gewählt. Zusätzlich verwendete er die Programmiersprache Zig für alles Weitere rundherum. Diese Entscheidungen haben einen herausragen-

den Vorteil: Performanz. Und in diesem Punkt setzt Bun große Ausführungszeichen. Durchweg zeigen sämtliche Benchmarks, dass die neue Runtime um ein Vielfaches schneller ist als die der beiden Alternativen.

Weitergehend soll Bun eine „All-in-one-Lösung“ werden. Neben der Laufzeit liefert Bun direkt einen Bundler, Test-Runner und Paketmanager mit. Außerdem werden eine optimierte SQLite-Schnittstelle, TypeScript-Unterstützung und mächtige Standard-APIs mitgeliefert. Das Ziel ist, dass weniger auf Abhängigkeiten Dritter zurückgegriffen werden muss. Da dies Risiken und Kompatibilitätsprobleme minimiert, ist dieser Aspekt sehr begrüßenswert.

Ein zentraler Fokus des Projektes, welcher sich ebenfalls sehr positiv auf den Erfolg des Projektes auswirkt, ist die fantastische Node.js-Kompatibilität. Die Node.js-internen Schnittstellen funktionieren weitestgehend nahtlos in Bun, obwohl die Laufzeit oft auch eigene optimierte Schnittstellen beinhaltet. Auch NPM-Pakete können problemlos verwendet werden [8]. In der Regel kann Node.js ohne Mehraufwand einfach durch Bun ausgetauscht werden.

Diese Punkte sorgten dafür, dass Bun spätestens seit Version 1.0 (September 2023) große Wellen schlägt. Die Resonanz ist weitestgehend positiv und eini-

ge Projekte wechseln bereits von Node.js zu Bun.

Fazit

Ich selbst habe mir angeeignet, statt „node“ oder „npm“ in der Kommandozeile immer öfter „bun“ in kleinen und privaten Projekten zu tippen. In Node.js sekundenlang andauernde Testausführungen oder TypeScript-Transpilierungen brauchen auf meiner Hardware mit Bun nur einige Zehntelsekunden. Übereifrig sofort alles nach Bun zu migrieren ist wahrscheinlich nicht empfehlenswert. Allerdings in Fällen, in denen die Entwicklererfahrung verbessert werden kann oder bei geschwindigkeits-kritischen Anwendungen, lohnt sich ein Blick auf die neue Laufzeit.

Ich bin auf die langfristigen Auswirkungen dieser Runtime auf die JavaScript-Welt gespannt, sie könnten gravierend sein. Eventuell kann Node selbst größere Performanz-Sprünge erzielen. Andererseits könnten viele Projekte auf Bun wechseln. Beispielsweise würde mich eine Visual-Studio-Code-Version mit der Geschwindigkeit von Bun sehr erfreuen. Davon würde dann auch die bald erscheinende neue SQL-Developer-Version, die auf diesen Editor aufbaut, profitieren. Die wichtigsten Eigenschaften werden in *Abbildung 1* abschließend veranschaulicht.

Quellen

- [1] Stack Overflow (2023): Stack Overflow Developer Survey – Most Popular Technologies. URL: <https://survey.stackoverflow.co/2023/#technology-most-popular-technologies>.
- [2] Node.js (2018): More Than a Billion Downloads of Node.js. URL: <https://nodejs.medium.com/more-than-a-billion-downloads-of-node-js-952a8a98eb42>.
- [3] Wikipedia (2023): List of software using Electron URL: https://en.wikipedia.org/wiki/List_of_software_using_Electron.
- [4] NPM (2020): So long, and thanks for all the packages! URL: <https://blog.npmjs.org/post/615388323067854848/so-long-and-thanks-for-all-the-packages>.
- [5] Ryan Dahl (2018): 10 Things I Regret About Node.js - Ryan Dahl - JSConf EU. URL: <https://www.youtube.com/watch?v=M3BM9TB-8yA>.
- [6] Deno (2022): Deno 1.28: Featuring 1.3 Million New Modules. URL: <https://deno.com/blog/v1.28>.
- [7] Deno (2023): Deno in 2022: npm, Edge Functions, Series A, Fresh, and more. URL: <https://deno.com/blog/deno-in-2022>.
- [8] Bun (2023): Bun is a fast JavaScript runtime/package manager/test runner/all-in-one-toolkit/bundler. URL: <https://bun.sh/>.

Über den Autor

Philipp Hartenfeller ist seit 2016 als Berater für die MT GmbH tätig. An einem normalen Tag setzt er Kundenanforderungen in der Oracle-Datenbank und APEX um. Nebenbei interessiert er sich sehr für Web-Entwicklung. Dies umfasst den gesamten Stack, wie JavaScript-intensive Frontends, UI/UX, Webservices, alle Arten von Backends und (natürlich) Datenlösungen.



Philipp Hartenfeller
philipp.hartenfeller@mt-itsolutions.com



AWR-Warehouse – die Zentrale für AWR-Daten

Ulf Schabram, OEDIV/Datenbank Administrator

Wenn ein DBA seinen AWR-Report zentral sammeln möchte, muss er sich nicht selbst etwas bauen. In diesem Artikel wird auf die Implementierung eines AWR-Warehouse via Oracle-Cloud-Control eingegangen. Wie wird AWR-Warehouse implementiert? Wie funktioniert der Transport der Daten?

AWR-Warehouse kurz um

Vor einiger Zeit berichtete ein sehr guter Kunde von AWR-Warehouse und hat uns neugierig gemacht. Bis dahin war uns der Begriff noch nicht untergekommen. Wir stellten fest, dass AWR-Warehouse eine gute Möglichkeit ist, um seine AWR-Snapshots zentral zu speichern. Mit wenigen Klicks können AWR-Reports via Oracle Cloud Control erstellt werden. Dabei lässt sich der Gesamtzeitraum nutzen, der gespeichert wurde. Das Speichern der Snapshots kann durch Einstellungen begrenzt werden, der Standard ist 24 Mo-

nate. Der Betrieb des AWR-Warehouse ist sehr komfortabel, alle Export- und Import-Jobs werden automatisch bei der Einbindung erstellt. Eine übersichtliche Konsole in der Webseite des Cloud-Control verrät, wie weit wir die Auswertung machen könnten.

Wie funktioniert es?

AWR-Warehouse basiert auf dem Standard AWR-Reports. Die Datenbank sammelt wie gewohnt die AWR-Reports im SYSAUX Tablespace. Es werden sieben Tage

der Reports aufbewahrt. Da kommt dann das AWR-Repository ins Spiel. Der Oracle Cloud Agent kann diese Daten durch einen Export in das Filesystem des Servers auslagern. Im nächsten Schritt werden die Daten durch Zippen der Dateien für den weiteren Transport verkleinert. Die exportierten AWR-Snapshots werden dann auf eine zentrale Repository-Datenbank kopiert. Für diesen Schritt wird das SCP-Protokoll benutzt, das eine Verbindungsmöglichkeit des Repository-Server und der Zieldatenbank voraussetzt. Wenn die Snapshot-Daten auf dem Repository-Server angekommen sind, werden sie per

Import in die Repository-Datenbank geladen. Somit ist der Vorgang abgeschlossen. Dieser Vorgang wird in der Standard-Einstellung einmal täglich vorgenommen. Die Aufbewahrung kann so lang erfolgen, wie man möchte. In *Abbildung 1* ist schematisch aufgelistet, wie der Weg des Reports erfolgt.

Voraussetzungen müssen sein

In unserer Umgebung haben wir ausschließlich „On Primes“-Datenbanken auf Linux- Servern im Einsatz. Für das AWR-Warehouse benötigen wir eine Oracle-Cloud- Control-Installation in der Version 13.5. Jedes angeschlossene Datenbank-System benötigt einen Cloud Control Agent. Weiterhin benötigt der Cloud-Control-Server einen SSH-Zugriff auf das Betriebssystem und einen Zugriff auf der CDB- oder nonCDB-Oracle-Datenbank. Leider benötigt der WAR-User DBA-Rechte, da sonst der Load-Prozess nicht richtig durchgeführt wird. Für das Speichern der Reports beziehungsweise der Snapshots haben wir eine PDB-Datenbank in einem Enter-

prise System im Einsatz. Diese PDB ist zusätzlich via Dataguard abgesichert. Es darf aber auch eine nonCDB-Datenbank parallel auf dem Cloud-Control-Server genutzt werden.

Implementierung via WAR-Repository-Datenbank

Zur Implementierung benötigen wir eine WAR-Repository-Datenbank. Diese muss auch im Cloud Control als Target eingebunden sein. In unserem Fall haben wir eine PDB-Datenbank auf einem Enterprise-Dataguard-Cluster genutzt. Leider speichert der WAR-Job die Daten im SYS-AUX Tablespace und ließ sich in unserer Version auch nicht anpassen. Der Knoten ist richtig lizenziert, somit steht der Nutzung nichts im Wege. Die Ziele (Targets), die später einen AWR-Report liefern sollen, müssen mit dem „Diagnostik- & Tuning-Pack“ lizenziert sein. Für das Einbinden der Repository-Datenbank legen wir einen DBA-Benutzer an (*siehe Listing 1*). Für das Einbinden der Datenbank nutzen wir die GUI. Hier gibt es leider noch keine emcli-Kommandos (*siehe Abbildung 2*).

Ziele hinzufügen

Nun wird es Zeit, dass wir die Ziele anschließen, die uns die AWR-Reports liefern sollen. Dafür müssen wir einige Schritte ausführen. Wir benötigen SSH Credentials für den Linux Server (*siehe Listing 2*). Dann haben wir auf der Ziel-Datenbank einen „AWR-Report“-Benutzer mit entsprechenden Rechten angelegt (*siehe Listing 2*). Diese Credentials werden ebenfalls im Cloud Control hinterlegt (*siehe Listing 2*). Jetzt sind wir bereit, das Ziel dem AWR-Repository bekannt zu geben (*siehe Listing 2*). Für das Einbinden der Datenbank nutzen wir die emcli-Konsole des Cloud-Control-Server.

ETL-Job bringt die Daten

Der ETL-Job, wie der Export-/Import-Job auch genannt wird, läuft im Wesentlichen in vier Schritten ab. Als erstes werden alle angefallenen AWR-Snapshots per AWR-Export in einem Dump auf dem Filesystem des Datenbankservers gespeichert. Die Dump-Datei wird danach vom Oracle Cloud Agent auf den Zentralen AWR- Re-

```
CREATE USER AWRWAREHOUSE IDENTIFIED BY "geheim" DEFAULT TABLESPACE SYSAUX TEMPORARY TABLESPACE TEMP;
ALTER USER AWRWAREHOUSE QUOTA UNLIMITED ON SYSAUX;
GRANT CONNECT TO AWRWAREHOUSE;
GRANT RESOURCE TO AWRWAREHOUSE;
grant sysdba to AWRWAREHOUSE;
GRANT INHERIT PRIVILEGES ON USER SYS TO AWRWAREHOUSE;
GRANT EXECUTE ON DBMS_SQL TO AWRWAREHOUSE;
```

Listing 1: DBA-User anlegen

```
emcli login -username=sysman
emcli set_credential -target_type=oracle_database -target_name="ZIELDB1" -credential_set=DBCredsSYSDBA
-user=sysman \
-columns="sysDBAUserName:AWRWAREHOUSE;sysDBAPassword:geheim;sysDBARole:sysdba"

emcli set_credential -target_type=host -target_name=Zeil-Server -credential_set=OHCreds -user=sysman
-column="OHUsername:oracle;OHPassword:<newPass>" -oracle_homes="ZIELDB1"

emcli awrwh_add_src_db -target_name=ZIELDB1 -target_type=oracle_database -db_host_cred=SSH_CRED_ZIELDB1
-db_cred=DB_CRED_ZIELDB1

emcli logout
```

Listing 2: Erstellen einer Tabelle für die Historisierung mit Gültigkeitszeitraum

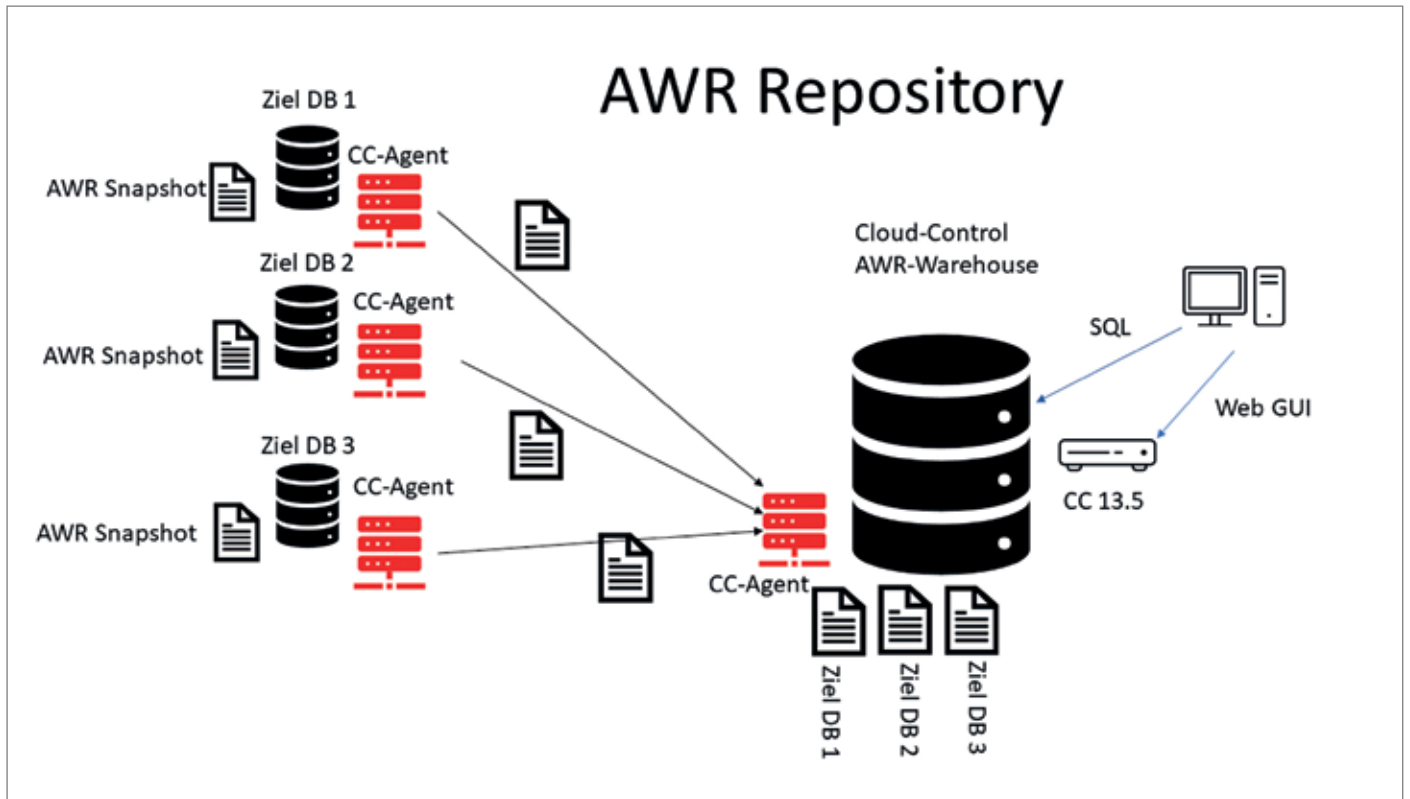


Abbildung 1: AWR-Schaubild (Quelle: Ulf Schabram OEDIV)

The screenshot shows the Oracle Enterprise Manager Cloud Control 13c interface. At the top, the navigation bar includes "Enterprise", "Targets", and "SYSMAN". A blue banner at the top left displays "Information" and "AWR Warehouse is not configured." The main content area is titled "Automatic Workload Repository Warehouse" and includes a sub-section "Collecting and Distributing Snapshots with AWR Warehouse". This section contains a diagram showing "Source Database 1", "Source Database 2", and "Source Database 3" feeding into an "AWR Warehouse Repository". The repository is configured to store "Database 1 Snapshots", "Database 2 Snapshots", and "Database N Snapshots". A note states "Administrators Use Historical AWR Snapshot Data in Several Database Diagnostics Features" and lists "Performance Home", "ASH Analytics", "AWR Report", "Compare Period ADDM", and "Compare Periods Report". A note below the diagram says "No Impact on source databases". On the right side of the screenshot, the "AWR Warehouse Repository Configuration" panel is visible. It shows the status as "Not Configured" and provides buttons for "Configure", "Edit", and "Diagnose". Below this, the "AWR Warehouse Database" configuration fields are listed: Name, Version, Configured By, Host, Port, and SID / Service Name. The "AWR Storage" section includes fields for Tablespace, Space Used (MB), Space Available (MB), and Retention Period. At the bottom of the storage section, there is a "Space Used (%)" gauge showing 0% usage.

Abbildung 2: GUI zur Eingabe der Repository-DB (Screenshot Anwendung CC 13.5) (Quelle: Ulf Schabram)

AWR Data Source **Historical - AWR Warehouse (Mar 23, 2021 12:00:18 AM - Aug 9, 2023 4:00:47 AM)** ▼

Abbildung 3: Auswahlmenü (Screenshot aus der Anwendung CC 13.5) (Quelle: Ulf Schabram)

```
col SUM_CPU_TIME format 999999999999999999
col SUM_DISK_READS format 999999999999999999

select * from ( select m.target_name,sum(CPU_TIME_DELTA) as SUM_CPU_TIME,
sum(DISK_READS_DELTA) as SUM_DISK_READS,
count(*) as COUNT
from
DBA_HIST_SQLSTAT a, dba_hist_snapshot s,dba_hist_database_instance di,
dbsnmp.caw_dbid_mapping m
where di.host_name='&host'
and di.dbid in m.new_dbid
and m.new_dbid = a.dbid
and a.snap_id = s.snap_id
and s.begin_interval_time > sysdate-30
group by m.target_name
order by sum(CPU_TIME_DELTA) desc)
```

Listing 3: SQL für eine Auswertung der CPU und der IO-Daten

pository-Server kopiert. Dort wird die Datei für den Import in der WAR-Repository-Datenbank vorbereitet. Nachdem die Datei angepasst wurde, werden die Snapshots in die AWR Repository Database geladen. Damit sind die Daten auf der GUI verfügbar.

Auswertungen via GUI

Das Zusammenstellen eines AWR-Reports aus dem Repository unterscheidet sich nicht von dem herkömmlichen Report. Im Auswahlmenü „AWR DataSource“ lässt sich das AWR Repository auswählen (siehe Abbildung 3).

Damit hat der Benutzer die Wahl über den ganzen Zeitraum der gespeicherten Daten im AWR Repository. So lassen sich große Zeiträume miteinander vergleichen.

Auswertung via SQL

Ein guter Weg, um die AWR-Daten zu nutzen, ist eine Auswertung aus der AWR-Repository-Datenbank zu erzeugen. Da alle Daten verfügbar sind, lassen sich sogar mehrere Datenbanken miteinander vergleichen. In Verbindung mit APEX lassen sich dann sehr schöne grafische Darstellungen erzeugen (siehe Listing 3).

Fazit

Durch die Lizenzierung für das „Diagnostik- & Tuning-Pack“ entstehen natürlich Kosten. Wenn dieser Schritt aber erstmal gegangen wurde, ist es durchaus vorteilhaft, die AWR-Daten zentral zu sammeln. Somit stehen die AWR-Daten selbst nach dem Abbau einer Datenbank noch zu Verfügung. Es können im Cloud Control noch Daten zu Testzwecken ausgewertet werden. Ein weiterer Vorteil ist das Vergleichen von mehreren Datenbanken/Servern miteinander. Mit dieser Methode hat der Datenadministrator ein gutes Werkzeug, um Voraussagen zu treffen.

Quellen

- [1] <https://www.oracle.com/technical-resources/articles/enterprise-manager/potvin-awr-em12c.html>
- [2] <https://docs.oracle.com/en/database/oracle/oracle-database/19/tdpnt/using-automatic-workload-repository-warehouse.html#GUID-532BD061-FE4A-4AF6-AA86-37F436EEA005>

Über den Autor

Ich arbeite seit 2017 als Datenbank-Administrator für die OEDIV. Speziell befasse ich mich mit dem Betrieb von Oracle-

Datenbanken. Um die Arbeit so effektiv wie möglich zu machen, setze ich Tools wie Cloud Control, Ansible und viele weitere ein und informiere mich immer wieder über Weiterentwicklungen. Technologien wie Multitaint, APEX, PDB unplug/plugin sind Themen, die ich gerne vorantreibe. Seit circa 2019 arbeite ich mit OpenSource-Datenbanken wie PostgreSQL und MariaDB. Natürlich alles im Datenbank-Team der OEDIV.



Ulf Schabram
ulf.schabram@oediv.de

Javaland

www.javaland.eu

AM NÜRBURGRING

09. -

⚡ 11.04. 2024



JETZT TICKETS

Präsentiert von:



Heise Medien

DOAG

NEUE LOCATION!

10TH
ANNIVERSARY

SICHERN!

Veranstalter: *JavaLand*

B
U
S
I
N
E
S
S
E
S

NEWS

01/2024

SMART CITIES



Das Konzept der Smart Cities

Dr. Christian Temath, Kompetenzplattform Künstliche Intelligenz Nordrhein-Westfalen (KI.NRW)

In einer Smart City wird intelligente Informations- und Kommunikationstechnologie verwendet, um die Teilhabe und Lebensqualität der Bürgerinnen und Bürger zu erhöhen und eine ökonomisch, ökologisch und sozial nachhaltige Kommune oder Region zu schaffen – so lautet eine gängige Definition von Smart Cities [1]. Vernetzte Lebenswelten, intelligente Energielösungen im urbanen Raum sowie Themen rund um saubere Städte, aber auch autonomes Fahren, Parkraumbewirtschaftung und Stauregulierung durch Künstliche Intelligenz (KI) sind einige Schlagworte, die das Konzept der Smart Cities greifbarer machen. In diesem Artikel lernen Sie bereits existierende (Forschungs-)Projekte und Lösungen überwiegend aus Nordrhein-Westfalen kennen, die nicht zuletzt mithilfe von KI zur nachhaltigen Entwicklung von Städten und Kommunen beitragen können. Viele dieser Beispiele finden sich auch bereits auf der KI.NRW-Landkarte [2], einer interaktiven Übersicht über das vielfältige nordrhein-westfälische KI-Ökosystem, die fast täglich wächst.

Vernetzte Lebenswelten am Beispiel neuer Mobilitätskonzepte

Täglich pendeln Millionen Menschen zu ihrer Arbeit, erledigen Besorgungen mit dem Auto oder transportieren Waren. Wir befinden uns bereits mittendrin in der Digitalisierung und Flexibilisierung des Personen- und Güterverkehrs und bewegen uns sukzessive in Richtung autonomes Fahren. Damit der Wandel ein Erfolg wird, sind neue innerurbane Mobilitätskonzepte gefragt, die die digitalen Dienste in das städtische Leben integrieren. Hier setzen Projekte wie „APEROL“ [3], [4] an, das sich der autonomen, personenbezogenen Organisation des Verkehrs und digitaler Logistik widmet. Zentral ist hierbei, dass auf Basis von Cloud- und Kommunikationstechnologien die individuellen Mobilitätsbedürfnisse von Bürgerinnen und Bürgern berücksichtigt werden. Das heißt, wenn Bürger*innen jederzeit in ein autonom fahrendes Fahrzeug, wie etwa in einen Shuttlebus steigen können, ist das eigene Auto nicht mehr nötig und sie können die tägliche Fahrtzeit zur Arbeit frei nutzen. Wenn diese Menschen ihren Pkw also stehen lassen, kann dieser beispielsweise per Carsharing von anderen genutzt werden, um Waren und Güter in ländliche Regionen zu transportieren. Das bedeutet, dass neben der Flexibilisierung des Stadtverkehrs mittelfristig auch die Mobilität für den stadtnahen ländlichen Raum verbessert wird. Das Konzept wurde in Aachen umgesetzt und die vorliegenden Ergebnisse haben Modellcharakter für andere Kommunen.

Vernetzte Lebenswelten dank ÖPNV

Dem öffentlichen Personennahverkehr, kurz ÖPNV, kommt ebenfalls eine große Bedeutung zu, wenn es um die Verbindung von Lebenswelten geht: Private Fahrten von der Stadt ins grüne Umland oder vom ländlichen Raum in die Stadt zählen genauso dazu, wie der Weg von Zuhause zur Schule, Universität oder Arbeit. Die „ÖPNV Digitalisierungsoffensive NRW“ [5] setzt an diesem Punkt an und ermöglicht mit modernen Mobilitätslösungen und barrierefreien Techniken, die nahtlos ineinandergreifen, den Menschen in NRW einen unkomplizierten Zugang zum ÖPNV und zu multimodaler Mobilität. Die Offensive bildet den Organisationsrahmen für vier Digitalisierungsprojekte des öffentlichen Nahverkehrs, die wiederum aus etwa 20 Unterprojekten bestehen. Diese werden von Verkehrsunternehmen und -verbänden, Zweckverbänden und weiteren Partnern im NRW-Nahverkehr pilotiert sowie landesweit ausgerollt.

1. In den „Querschnitts- und VDV-Projekten“ werden Teilprojekte zum Ausbau der digitalen Infrastruktur als Basis für betreiberneutrale Services initiiert und koordiniert. Hierzu zählt beispielsweise ein bereits durchgeführter Wettbewerb zur Förderung des KI-Einsatzes im nordrhein-westfälischen ÖPNV.
2. Das Projekt „EFM-Technologien“ wiederum bezieht sich auf die Zusammenführung von Nutzungsmedien für den digitalen Vertrieb und die Optimierung von Prozessen. So wurde unter anderem in Bonn ein Pilotprojekt für ein Account- bzw. ID-basiertes Ticketing umgesetzt.

3. Im Projekt „eTarif“ wurde eine landesweite eTarif-Lösung für NRW erarbeitet und es wurden zudem Modelle zur Gewinnung von Neukund*innen erprobt.
4. Die Teilprojekte zum Thema „Information und Datenqualität“ richten sich auf den Abbau von Zugangsbarrieren zum öffentlichen Nahverkehr und den qualitativen Ausbau der Fahrgastinformation. Dazu gehört beispielsweise ein Sprachassistent zur elektronischen Fahrplanauskunft, den alle Verkehrsunternehmen einsetzen können.

Vernetzte Lebenswelten in Kommunen

Es gibt auch noch ein weiteres spannendes Projekt im Kontext „Vernetzte Welt“, an dem das Fraunhofer IOSB-INA mitwirkt: Darin wird die Frage gestellt, wie das Internet der Dinge (IoT) das Leben für die Menschen in der Stadt erleichtern, attraktiver und sicherer machen kann. Die daraus hervorgegangene Innovations- und Testplattform „Lemgo Digital“ [6] mit angeschlossenem Reallabor ermöglicht es den Partnern aus Stadtverwaltung, Wirtschaft und Gesellschaft, Lösungen rund um Digitalisierungsvorhaben in Klein- und Mittelstädten unter realen Bedingungen zu erproben und die Smart-City-Anwendungen mehrwertbringend in der Stadt einzusetzen. Dabei sind bereits zahlreiche erprobte IoT-Lösungen in den Bereichen Mobilität, attraktive Innenstadt und Umwelt entstanden, die von anderen Kommunen für die digitale Transformation adaptiert werden können.

Intelligente Energielösungen in Gewerbeeinheiten

Ob im Kontext von gewerblich genutzten Immobilien oder in den eigenen vier Wänden: Energie einzusparen, wann immer es geht, ist für die nachhaltige Entwicklung von Städten unabdingbar. Auch hier kann Künstliche Intelligenz helfen. Denken wir an den Energieverbrauch und CO₂-Ausstoß von Gewerbeeinheiten. Hier lassen sich mithilfe von KI-ausgewerteten Sensordaten die Lüftungs- und Heizungseinstellungen optimieren und auf diese Weise 10 bis 20 Prozent Energie einsparen. Dafür kommt eine selbstlernende Technologie zum Einsatz, die klimatechnische Anlagen bedarfsgerecht aus der Cloud regelt. Als Basis werden die Betriebs- und Verbrauchsdaten des Gebäudes genutzt und zusätzlich mit vielfältigen weiteren Datenströmen und Prognosen beispielsweise zu Wetter, Kund*innenfrequenz, aber auch Belegung und Öffnungszeiten gefüttert. Durch KI-basierte Algorithmen werden die technischen Anlagen laufend optimiert [7].

Intelligente Energielösungen in Stadtquartieren

Wenn wir von Bestandsgebäuden weitergehen zur Planung und zum Bau von ganzen Stadtquartieren: Wie sieht es in diesem Bereich mit smarten Energielösungen aus? Wie kann dort auf den Einsatz fossiler Energieträger in großen Teilen verzichtet werden, um die Klimaschutzziele der Bundesregierung zu erreichen? Diese und weitere Fragen rund um die klimaneutrale Energieversorgung beantwortet das Projekt „SmartQuart“ [8] und zeigt, dass dies schon heute technisch und wirtschaftlich möglich ist. Dabei werden möglichst viele Beteiligte von Beginn an in die Entwicklung der innovativen Systeme mit einbezogen. Das Projekt ist Teil des Programms „Reallabore der Energiewende“ und wird gefördert vom Bundesministerium für Wirtschaft und Klimaschutz. In insgesamt drei ländlichen und urbanen Quartieren, darunter zwei in NRW, erarbeitet ein Konsortium aus mehreren Projektpartnern bis 2024 neue Produkte und Lösungen für die Planung, Errichtung und den Betrieb energieoptimierter Quartiere für ganz Deutschland – mit Beteiligung von Bürger*innen und Entscheider*innen vor Ort. So ist es bereits gelungen, lokal verfügbare erneuerbare Energien, insbesondere aus Wind und Sonne, mithilfe der Digitalisierung intelligent

für die notwendige Energieversorgung zu nutzen sowie innerhalb und zwischen den Quartieren smart zu vernetzen. Das bedeutet, der Verbrauch und die Erzeugung werden bereits auf lokaler Ebene optimiert und etwaige Überschüsse bilanziell den anderen Quartieren zur Verfügung gestellt. Wie sich zeigt, endet intelligentes Energiemanagement nicht vor der eigenen Haustür.

Ein weiteres Element, das immer mehr an Bedeutung im städtischen Raum gewinnt, ist die Infrastruktur von E-Ladesäulen für Autos. Damit diese zuverlässig funktionieren, kommt Künstliche Intelligenz zum Einsatz. Hierbei übernimmt die KI voll automatisiert die Fernüberwachung [9]. Dafür werden die KI-Systeme mit Echtzeitdaten der Ladepunkte versorgt, die unter anderem Informationen wie technische Parameter, die Auslastung, aber auch Abweichungen, zum Beispiel den Rückgang der Ladeaktivität oder wiederholte Abbrüche des Ladevorgangs, beinhalten. Kommt es zu Abweichungen, werden Service-Mitarbeitende informiert. Seit diesem Jahr übernimmt eine KI auch die vorausschauende Wartung, um den Ausfall von Ladesäulen zu verhindern [10].

KI im Straßenverkehr: Autonomes Fahren

Die technologischen Entwicklungssprünge in der KI und im Maschinellen Lernen treiben die Entwicklung digitaler Lösungen für innovative Mobilitätskonzepte an und ermöglichen vielfältige Einsatzmöglichkeiten wie beispielsweise im Bereich der intelligenten Verkehrsflusssteuerung, aber auch im hochautomatisierten bis autonomen Fahren. In dem von der nordrhein-westfälischen Landesregierung geförderten KI.NRW-Flagship-Projekt „bergisch.smart_mobility“ [11], das kürzlich erfolgreich zu Ende ging, wurde eindrücklich gezeigt, wie Mobilität im bergischen Städtedreieck zukünftig aussehen kann und welche Voraussetzungen dafür geschaffen werden müssen: Handlungsfelder wie „Autonomes Fahren im Bergischen Städtedreieck“, „Intelligente Verkehrssteuerung durch die Erprobung von Sensortechnik“, „Fahrzeug-Sharing-Angebote auf Abruf“ und „Smart Fahren im Quartier“ wurden in Zusammenarbeit von Industrie, Verwaltung und Wissenschaft besonders forciert. Daraus sind insgesamt 70 neue Produktentwicklungen entstanden, die zum Patent angemeldet wurden, und Dienstleistungen wie der On-

Demand-Service der „Hol mich!“-App. Dieses Leuchtturmprojekt hat Modellcharakter und dessen Ergebnisse lassen sich auf andere Kommunen übertragen. Ein anderes Beispiel aus der Praxis findet sich in Monheim am Rhein: Dort drehen seit Anfang 2020 hochautomatisierte Shuttles zwischen Busbahnhof und Altstadt ihre Runden [12].

Auch zahlreiche andere Projekte haben sich dem Zukunftsthema „Autonome Fahrzeuge“ gewidmet: So werden im Projekt „New Mobility Lab“ [13] selbstfahrende Busse eingesetzt, die nach Bedarf von Nutzer*innen und Nutzern angefordert werden können, während im nächsten Beispiel ein Start-up ein intelligentes, elektrisches und automatisiertes Logistikfahrzeugsystem entwickelt hat, das besonders für den CO₂-schonenden Einsatz in der Industrie interessant sein kann [14].

KI im Straßenverkehr: Intelligente Regulierung und Stauvermeidung

Insbesondere beim autonomen Fahren stellt sich immer wieder die Frage, wie sicher und zuverlässig die Technologie dahinter ist. So müssen beispielsweise Verkehrszeichen korrekt durch die Bilderkennung erfasst werden – auch solche mit dynamischen Informationen zur aktuellen Verkehrslage. Der Ansatz „Auto-Construct“ [15] des Fraunhofer-Instituts für Intelligente Analyse- und Informationssysteme IAIS ermöglicht ein solches System ausgestattet mit Kameras. Darüber werden die Informationen verarbeitet, inhaltlich verstanden und zur Weiterverarbeitung verfügbar gemacht. Mit Deep Learning erkennt die Software klassische Muster schnell und effizient.

Eine dynamische Verkehrsanalyse verfolgt das Projekt „KonSensData“ [16] und zwar mit dem Ziel, den Verkehrsfluss in Städten zu optimieren. Dafür wurde ein flexibles und wetterfestes Sensorsystem entwickelt, das Verkehrsdaten erfasst, diese KI-gestützt analysiert und datenschutzkonform übermittelt. Das System erlaubt es, Vorhersagen von Verkehrsflüssen durch Pkws, Lkws, Radfahrende sowie Fußgänger*innen zu treffen und eine Echtzeit-Kommunikation mit autonomen Fahrzeugen oder Fahrassistenzsystemen zu ermöglichen. Die Ergebnisse lassen sich aber beispielsweise auch zur Optimierung von Ampelschaltungen nutzen.

Im „Stauland“ Nordrhein-Westfalen beschäftigen sich zahlreiche Forschungsvorhaben mit dem Thema Stauvermeidung

beziehungsweise -regulierung und damit einhergehend der Reduzierung von Umweltbelastungen mithilfe von KI: So haben die beiden Projekte „KI4LSA“ [17] und „EnDyVA“ [18] durch die intelligente Schaltung von Lichtsignalanlagen (LSA) Lösungen entwickelt. Üblicherweise werden diese überwiegend mit Festzeitsteuerung in Kombination mit Bedarfsanforderungen geschaltet. Ziel der erfolgreich durchgeführten Projekte war es, die Schaltzeiten von LSA zu optimieren und beispielsweise im Verbund dynamisch und in Echtzeit an aktuelle Verkehrssituationen anzupassen. Speziell für die Verbesserung des Verkehrsflusses in logistikintensiven Bereichen in der Stadt, wie in Gewerbegebieten oder Binnenhäfen, setzt sich das Projekt „logisticsCS.NRW“ ein [19].

KI im Straßenverkehr: Parkraumbewachung

Neben dem hohen Verkehrsaufkommen in Städten stellt die hohe Nachfrage an freien Parkplätzen die Verkehrs- und Stadtplaner*innen, aber auch die Autofahrer*innen vor Herausforderungen. In Smart Cities unterstützen Sensorsysteme bei der Verkehrsplanung. Sie erkennen lokal verschiedene Veränderungen und übermitteln diese. So werden etwa parkende und wartende Autos erfasst, um die Parkplatznutzung beziehungsweise -nachfrage oder die Stausituation zu ermitteln. Mithilfe der gewonnenen Daten lassen sich mittels Künstlicher Intelligenz Prognosen erstellen, die Aufschluss über die Verkehrsauslastung in den Städten oder auf den Straßen geben [20]. Besonders Stellplätze für Fernbusse und Lkws sind begehrt. Konkret bei der Suche nach freien Buchten unterstützt ein KI-gestütztes Tool, das im Rahmen des Projekts „Smart Optimized Lorry Parking – SOLP“ [21] entwickelt wurde. Über ein Benutzer-Interface erfolgt die Übermittlung der prognostizierten Auslastung von Straßen und Parkplätzen direkt an die Fahrerinnen und Fahrer.

KI für alle, die unterwegs sind

Legen wir abschließend den Fokus auf eine andere wichtige Gruppe von Verkehrsteilnehmenden in der Stadt: Die Fußgängerinnen und Fußgänger, die die Straße an einer Ampel überqueren wollen. Untersuchungen zeigen, dass die Grünphasen nicht die notwendigen Wartezeiten aller Personengruppen, wie

beispielsweise Eltern mit Kinderwagen oder Rollstuhlfahrenden, berücksichtigen. In dem Projekt „KI4PED“ [22] wurde daher ein Algorithmus zur intelligenten Bilderkennung entwickelt, der auf Basis von LiDAR-Daten (datenschutzkonform) wartende Personen an Fußgängerampeln erkennt. Dieses System reagiert dynamisch auf die Personen, verkürzt Wartezeiten und erhöht gleichzeitig die Verkehrssicherheit.

Saubere Luft – Saubere Städte

Am Ende ist es für eine „smarte“ City aber entscheidend, dass die Bürger*innen gerne in ihr leben. Hierauf zielen insbesondere Konzepte ab, die sich mit Aspekten wie Umweltschutz, Müllreduktion und sauberer Luft befassen. Auf zwei Beispiele möchte ich eingehen: Zum einen bietet ein Start-up aus NRW mit seiner KI-basierten Lösung „CORTEXIA“ [23] Kommunen die Möglichkeit, die Stadtsauberkeit digital zu messen, um die Lebensqualität der Bürger*innen zu verbessern und gleichzeitig die Umwelt zu schützen. Mithilfe von Bilderkennungs-Algorithmen werden anhand von zuvor aufgenommenen Kameradaten kleine Abfälle wie Kaugummi, Zigaretten, Plastikmüll oder Flaschen erkannt. Durch die bereitgestellten Daten können die Kommunen ihre Reinigungsressourcen intelligenter verteilen.

Zum anderen setzt das Projekt „SAUBER“ [24] beim Thema verkehrsbedingte Luftverschmutzung an. Ziel ist es, eine Informationsplattform für die nachhaltige Stadt- und Regionalentwicklung zu schaffen, um die Luftqualität speziell im urbanen Raum zu verbessern. Das Projekt wurde bereits in Nordrhein-Westfalen erprobt, wo sich zeigte, dass der Einsatz von KI es ermöglicht, flächendeckende und detaillierte Prognosen und Simulationen der zukünftigen Luftqualität durchzuführen.

Intelligente Stadtverwaltung

In den Stadt- und Rathäusern hierzulande liegt ein großes Potenzial für den Einsatz von Künstlicher Intelligenz, insbesondere repetitive, administrative Prozesse können künftig in großen Teilen durch eine KI unterstützt werden. Aber auch die Online-Interaktionen der öffentlichen Verwaltung mit ihren Bürger*innen, die viel Zeit in Anspruch nimmt, ist so ein Bereich: Hier setzt das Projekt „GovBot“ [25] an, in dem ein selbstlernender Chatbot entwickelt wur-

de, der den Bürger*innenservice verbessert, da er intuitiv bedienbar ist und rund um die Uhr Fragen zu Verwaltungsservices beantwortet. Je nach Konfiguration liefert der Chatbot allgemeine Informationen, vereinbart Termine, nimmt Wohnsitzummeldungen entgegen oder reserviert Wunschkennezeichen.

Um das Potenzial von KI in Einklang mit deutschem Recht zu bringen, legt das Projekt „KILaw“ [26] seinen Fokus auf die Chancen und Herausforderungen, die mit KI in der Verwaltung einhergehen, und entwickelt Regulierungs- und Sicherungsmechanismen. Darin werden die normativen Rahmenbedingungen für ein digitalisiertes, aber demokratisch kontrolliertes staatliches Verwaltungshandeln und flankierende Sicherungsmechanismen entworfen, wie etwa Transparenz- und Begründungspflichten.

Fazit

Es zeigt sich nicht zuletzt im Bereich der Smart Cities deutlich: KI ist da, um zu helfen. Und gute Konzepte sind bereits in großer Zahl vorhanden. Bei all den zum Einsatz kommenden Systemen ist es jedoch wichtig, dass unsere Persönlichkeitsrechte gewahrt werden. Wir bei KI.NRW stellen die Menschen in den Mittelpunkt einer vertrauenswürdigen KI. Genau darum kümmern wir uns beispielsweise im Rahmen des KI.NRW-Flagship-Projekts „ZERTIFIZIERTE KI“ [27]. Dort werden standardisierte Prüfverfahren für sichere KI-Systeme erarbeitet. Die genannten Herausforderungen der Technologie mitberücksichtigend, sollten wir aber insbesondere die Chancen von Künstlicher Intelligenz nutzen, die sie für Smart Cities bereithält.

Quellen

- [1] „Smart City – Bundesamt für Sicherheit in der Informationstechnik (BSI)“, <https://www.bsi.bund.de>, letzter Zugriff am 30.10.2023 um 12:54 Uhr.
- [2] „KI.NRW-Landkarte – KI.NRW“, <https://www.ki.nrw/ki-landkarte/>, letzter Zugriff am 30.10.2023 um 12:55 Uhr.
- [3] „KI.NRW-Landkarte – Autonome, personenbezogene Organisation des Straßenverkehrs und digitale Logistik (APEROL)“, <https://www.ki.nrw/ki-landkarte/#/autonome-personenbezogene-organisation-des-strassenverkehrs-und-digitale-logistik-aperol>, letzter Zugriff am 30.10.2023 um 12:58 Uhr.
- [4] „APEROL“ Autonome, personenbezogene Organisation des Straßenver-



- kehrs und digitale Logistik – aperol“, https://www.autonomousshuttle.de/wp-content/uploads/2021/06/APEROL_Schlussbericht_Langversion_final.pdf, letzter Zugriff am 30.10.2023 um 13:35 Uhr.
- [5] „ÖPNV Digitalisierungsoffensive Nordrhein-Westfalen – Kompetenzzentrum Digitalisierung / Verkehrsverbund Rhein-Ruhr“, <https://digitalemobilitaet.nrw/index.html>, letzter Zugriff am 30.10.2023 um 13:37 Uhr.
- [6] „KI.NRW-Landkarte – Lemgo Digital“, <https://www.ki.nrw/ki-landkarte/#/lemgo-digital>, letzter Zugriff am 30.10.2023 um 13:40 Uhr.
- [7] „KI.NRW-Landkarte – Recogizer Group GmbH“, <https://www.ki.nrw/ki-landkarte/#/recogizer-group-gmbh>, letzter Zugriff am 30.10.2023 um 13:41 Uhr.
- [8] „SmartQuart – E.ON“, <https://smartquart.energy>, letzter Zugriff am 30.10.2023 um 13:42 Uhr.
- [9] „KI.NRW-Landkarte – Fernüberwachung von Ladestationen mit künstlicher Intelligenz“, <https://www.ki.nrw/ki-landkarte/#/fernueberwachung-von-ladestationen-mit-kuenstlicher-intelligenz>, letzter Zugriff am 30.10.2023 um 13:43 Uhr.
- [10] „E.ON: KI-Tool Evercharge optimiert Verfügbarkeit von Ladenetzen – ecomento“, <https://www.electrive.net/2023/06/05/e-on-stellt-ki-tool-evercharge-vor/>, letzter Zugriff am 30.10.2023 um 13:45 Uhr.
- [11] „KI.NRW-Flagships – bergisch.smart_mobility“, <https://www.ki.nrw/flagships/bergisch-smart-mobility/>, letzter Zugriff am 30.10.2023 um 13:47 Uhr.
- [12] „Bahnen der Stadt Monheim“, <https://www.bahnen-monheim.de/autonomer-bus/ziele-und-motivation>, letzter Zugriff am 30.10.2023 um 13:54 Uhr.
- [13] „KI.NRW-Landkarte – New Mobility Lab (a-Bus)“, <https://www.ki.nrw/ki-landkarte/#/new-mobility-lab-a-bus>, letzter Zugriff am 30.10.2023 um 13:54 Uhr.
- [14] „KI.NRW-Landkarte – DroidDrive GmbH“, <https://www.ki.nrw/ki-landkarte/#/droiddrive-gmbh>, letzter Zugriff am 30.10.2023 um 13:57 Uhr.
- [15] „KI.NRW-Landkarte – Erkennung von Straßenverkehrszeichen (AutoConstruct)“, <https://www.ki.nrw/ki-landkarte/#/erkennung-von-strassenverkehrszeichen-autoconstruct>, letzter Zugriff am 30.10.2023 um 13:58 Uhr.
- [16] „KI.NRW-Landkarte – Dynamische Verkehrsanalyse und sicheres autonomes Fahren (KonSens-Data)“, <https://www.ki.nrw/ki-landkarte/#/dynamische-verkehrsanalyse-und-sicheres-autonomes-fahren-konsensdata>, letzter Zugriff am 30.10.2023 um 13:59 Uhr.
- [17] „KI.NRW-Landkarte – Künstliche Intelligenz für Lichtsignalanlagen (KI4LSA)“, <https://www.ki.nrw/ki-landkarte/#/kuenstliche-intelligenz-fuer-lichtsignalanlagen-ki4lsa>, letzter Zugriff am 30.10.2023 um 14:00 Uhr.
- [18] „KI.NRW-Landkarte – Entwicklung eines bedarfsgerechten dynamischen Schaltverhaltens von Lichtsignalanlagen (Ampeln) zur Optimierung des Verkehrsflusses in der Stadt Aachen (EnDyVA)“, <https://www.ki.nrw/ki-landkarte/#/entwicklung-eines-bedarfsgerechten-dynamischen-schaltverhaltens-von-lichtsignalanlagen-ampeln-zur-optimierung-des-verkehrsflusses-in-der-stadt-aachen-endyva>, letzter Zugriff am 30.10.2023 um 14:01 Uhr.
- [19] „KI.NRW-Landkarte – Crowd Solving – Intelligente Infrastrukturnutzung am Beispiel Neuss-Düsseldorfer Häfen (logistiCS.NRW)“, <https://www.ki.nrw/ki-landkarte/#/crowd-solving-intelligente-infrastrukturnutzung-am-beispiel-neuss-duesseldorfer-haefen-logisticsnrw>, letzter Zugriff am 30.10.2023 um 14:02 Uhr.
- [20] Anbietende Start-ups in NRW sind beispielsweise PARKLAB Technologie („KI.NRW-Landkarte – PARKLAB Technologie GmbH“, <https://www.ki.nrw/ki-landkarte/#/parklab-technologie-gmbh>, letzter Zugriff am 30.10.2023 um 14:06 Uhr) und SONAH („KI.NRW-Landkarte – SONAH GmbH“, <https://www.ki.nrw/ki-landkarte/#/sonah-gmbh>, letzter Zugriff am 30.10.2023 um 14:07 Uhr).
- [21] „KI.NRW-Landkarte – Smart Optimized Lorry Parking / KI-gestützte Parkplatzsuche für Lkws und Fernbusse (SOLP)“, <https://www.ki.nrw/ki-landkarte/#/smart-optimized-lorry-parking-ki-gestuetzte-parkplatzsuche-fuer-lkws-und-fernbusse-solp>, letzter Zugriff am 30.10.2023 um 14:08 Uhr.
- [22] „KI.NRW-Landkarte – KI-basierte Optimierung von Fußgängerüberquerungszeiten (KI4PED)“, <https://www.ki.nrw/ki-landkarte/#/ki-basierte-optimierung-von-fussgaengerueberquerungszeiten-ki4ped>, letzter Zugriff am 30.10.2023 um 14:09 Uhr.
- [23] „KI.NRW-Landkarte – REMONDIS Digital Services GmbH – CORTEXIA“, <https://www.ki.nrw/ki-landkarte/#/remondis-digital-services-gmbh-cortexia>, letzter Zugriff am 30.10.2023 um 14:10 Uhr.
- [24] „KI.NRW-Landkarte – Satellite Data for Better Air Quality (SAUBER)“, <https://www.ki.nrw/ki-landkarte/#/satellite-data-for-better-air-quality-sauber>, letzter Zugriff am 30.10.2023 um 14:11 Uhr.
- [25] „Tools zur Kommunikation mit Bürger:innen, oder untereinander: GovBot – publicplan“, <https://www.publicplan.de/loesungen/kommunikation-chatbots>, letzter Zugriff am 30.10.2023 um 14:28 Uhr.
- [26] „KI.NRW-Landkarte – Künstliche Intelligenz in Staat und Verwaltung (KILaw)“, <https://www.ki.nrw/ki-landkarte/#/kuenstliche-intelligenz-in-staat-und-verwaltung-kilaw>, letzter Zugriff am 30.10.2023 um 14:28 Uhr.
- [27] „ZERTIFIZIERTE KI“, <https://www.zertifizierte-ki.de/>, letzter Zugriff am 30.10.2023 um 14:31 Uhr.

Dr. Christian Temath
info-kinrw@iais.fraunhofer.de

Dr. Christian Temath ist seit September 2020 Geschäftsführer der Kompetenzplattform KI.NRW und arbeitet am Fraunhofer IAI in Sankt Augustin mit seinem Team daran, die Marke „KI made in NRW“ zu etablieren und die technologische Souveränität des Landes NRW zu stärken. Als promovierter Wirtschaftsinformatiker verfügt er über langjährige Erfahrung in der Managementberatung im Bereich Technologie sowie in der praktischen Anwendung von KI-Technologien bei einem internationalen e-Commerce-Unternehmen.



Schlüsselrolle Künstliche Intelligenz – die Zukunft des ÖPNV in Smart Cities

Dr. Roxana Hess und Katja Rische, INIT GmbH

Der ÖPNV ist von entscheidender Bedeutung, um die Mobilität der Menschen sicherzustellen und dabei gleichzeitig die Verkehrsbelastung, Luftverschmutzung und den Energieverbrauch zu reduzieren. Doch traditionelle öffentliche Verkehrssysteme stoßen gerade in urbanen Zentren immer wieder an ihre Grenzen. Hier setzt die Idee der Smart Cities an, in denen vernetzte Technologien, Digitalisierung, Künstliche Intelligenz (KI) und Datenanalyse eine elementare Rolle spielen und auch den ÖPNV effektiver, effizienter und nachhaltiger gestalten können.

In diesem Artikel werden die Aktivitäten der INIT, einem weltweit führenden Anbieter für IT-Lösungen im öffentlichen Nahverkehr mit Sitz in Karlsruhe, vorgestellt und gezeigt, wie der Einsatz von KI in ihren Produkten den ÖPNV in Smart Cities revolutionieren kann.

Das Karlsruher Unternehmen arbeitet bereits erfolgreich mit KI und hat einige seiner Lösungen mit Echtzeitdaten aus Bussen und Bahnen trainiert. Die so trainierten Algorithmen können heute schon im Livebetrieb Abfahrtsprognosen präzisieren und zukünftig sogar Disponenten in der Leitstelle mit Lösungen unterstützen, indem intelligente Assistenzsysteme optimale Routen vorschlagen, die beispielsweise kurzfristig Staus umgehen oder auf der Basis vergangener Maßnahmen dem Leitstellenpersonal die auf die jeweilige Situation angepasste Maßnahme vorschlagen. KI ebnet außerdem den Weg für einen ÖPNV, der sich flexibel an die Bedürfnisse der Fahrgäste anpasst und sie in den Vordergrund stellt.

Training mit Echtzeitdaten

Die Grundlage von Künstlicher Intelligenz bilden Daten. Und die sind bereits jetzt schon reichlich vorhanden. Ob historische oder Echtzeit-Verkehrsdaten, Daten aus dem Ticketing, der Fahrplanauskunft oder dem Betriebsleitsystem der Verkehrsbetriebe: Die gesammelten Daten sind mittlerweile so detailliert und umfangreich, dass sie für das Training von Algorithmen weiterverwendet werden können. Die Anwendung von KI auf diese Daten kann Veränderungen und Trends präziser und in kürzerer Zeit aufzeigen. Das ermöglicht verbesserte Abfahrtsprognosen, eine optimale Routenplanung sowie schnelle Reaktionen auf aktuelle Ereignisse wie Unfälle oder Staus.

Verlässlichere Abfahrtsprognosen dank Maschinellem Lernen

In einem Pilotprojekt von INIT und dem Verkehrsunternehmen Golden Gate Bridge, Highway & Transportation District in San Francisco wurde beispielsweise mithilfe einer KI-basierten Softwarelösung der österreichischen INIT Tochter inola die Genauigkeit der Abfahrtsprognosen von Bussen erheblich verbessert. ML-Core, das Softwareprodukt von inola, stellt Prognosen auf Basis historischer Daten und Echtzeitinformationen zur Verfügung. Es verwendet innovative Algorithmen zur Mustererkennung und kann große Datenmengen (Big

Data) verarbeiten. Das Auswertungs- und Statistiksystem von INIT sammelt die Betriebsdaten und bereitet diese mit vielen weiteren Informationen wie Fahrzeiten auf. Auf dieser Datengrundlage aufbauend, stehen dem ML-Core unterschiedliche Methoden zur automatischen Mustererkennung und Zeitprognose zur Verfügung. Die jeweils beste Methode wird von der Software automatisch erkannt und eingesetzt. Verschlechtert sich die Prognosequalität, wird automatisch ein Neutraining des ML-Modells angestoßen. Somit werden Veränderungen im Verkehrsfluss oder Linienänderungen sofort erkannt und abgebildet. Mit herkömmlichen Prognoseverfahren wäre dies nicht möglich.

Auf der Basis des trainierten Modells berechnet ML-Core eine Fahrtzeitprognose für alle Fahrabschnitte. Aus diesen Einzelwerten werden dann die neu prognostizierten Abfahrtszeiten für die Haltestellen zusammengesetzt und an die diversen angeschlossenen Systeme weitergegeben. Dabei werden auch aktuelle Verkehrseinschränkungen oder die Fahrtdauer der unmittelbar vorausfahrenden Fahrzeuge berücksichtigt. Fahrgäste können direkt die Live-Prognosen einsehen, so wird die Fahrt noch besser planbar. Die Konsequenz: Die Zufriedenheit der Fahrgäste steigt, denn dadurch gibt es keine Sprünge in der Ankunftsprognose mehr.

Mit KI zu zuverlässigeren Auslastungsinformationen und mehr Fahrgastkomfort

Die Kenntnis darüber, wie voll ein Bus oder eine Bahn werden wird, erlaubt Fahrgästen, ihre Reisepläne entsprechend anzupassen und gegebenenfalls auf eine weniger ausgelastete Verbindung auszuweichen. Dafür sind zuverlässige Informationen über die Auslastung der Fahrzeuge in Echtzeit erforderlich, wie sie INITs System zur Auslastungsinformation und Fahrgastlenkung liefert. Die technologische Herausforderung liegt dabei in der Bereitstellung einer soliden Datenbasis zur Bestimmung des erwarteten Besetzungsgrades eines Fahrzeugs.

Die aktuelle Auslastung wird mithilfe von Fahrgastzählsensoren anhand der übertragenen Ein- und Aussteigerzahlen nach jeder Abfahrt von einer Haltestelle ermittelt, mit Fahrplandaten verknüpft und auf Plausibilität geprüft. Damit liegt eine tatsächlich gemessene Anzahl der Fahrgäste in einem Fahrzeug vor. Unter

Heranziehung der Fahrzeugkapazitäten wird die aktuelle Auslastung berechnet und abgespeichert.

Für die Ermittlung einer in der Zukunft liegenden Auslastung kommt ein patentiertes Verfahren und ein KI-Algorithmus zum Einsatz. Damit wird im Hintergrundsystem der Echtzeit-Besetzungsgrad mit dem aus historischen Daten gewonnenen typischen Ein- und Ausstiegsverhalten an der Folgestelle korreliert. Auf diese Weise findet auch die Zahl der voraussichtlichen Aussteiger Berücksichtigung.

Im Forschungsprojekt Mobile Data Fusion arbeitete INIT an der Nutzung weiterer Datenquellen für die Besetzungsprognose, für den Fall, dass die Fahrzeuge nicht mit Zählsensoren ausgestattet sind. Infrage kommen dafür beispielsweise WLAN- und Bluetooth-Signale von Smartphones oder Anfragen an die Verbindungsauskunft und die Mobilithek des Bundesministeriums für Digitales und Verkehr (BMDV), ein offenes Datenportal, das Mobilitäts-, Geo- und Wetterdaten zur Verfügung stellt.

Besonders entscheidend ist die Bereitstellung von Auslastungsinformationen über verschiedene Fahrgastinformationskanäle, wie beispielsweise Verkehrsunternehmens-Apps oder -Websites. Auf diese Weise haben Fahrgäste die Möglichkeit, ihre Reisepläne entsprechend anzupassen und gegebenenfalls auf weniger stark frequentierte Verbindungen auszuweichen. Dies trägt wiederum zu einer gleichmäßigeren Auslastung bei.

Forschungsfeld KI

Der Einsatz von künstlicher Intelligenz ist ebenfalls Teil zahlreicher Forschungsprojekte, an denen INIT beteiligt ist und die bei der Weiterentwicklung der unternehmenseigenen Produkte und Lösungen unterstützen. Ein Leuchtturmprojekt in diesem Zusammenhang ist das Verbundvorhaben KARL [1]. Es beschäftigt sich mit den Auswirkungen von künstlicher Intelligenz auf das Arbeitsumfeld und die Betriebsorganisation mit dem Ziel, menschenzentrierte, transparente und lernförderliche KI-unterstützte Arbeits- und Lernsysteme zu konzipieren und praktisch zu erproben.

Unter der Leitung der Hochschule Karlsruhe untersucht das überwiegend aus Karlsruher Partnern bestehende Konsortium diese Fragestellung anhand von acht Anwendungsfällen aus den Bereichen Mobilität, wissensintensive Dienstleistungen



© INIT | Kerstin Groh

(Literaturrecherche, intelligente Lernsysteme) und IKT-Systeme, produzierendes Gewerbe sowie Bildung. INIT leitet einen der Mobilitäts-Use-Cases und arbeitet mit der Hochschule Karlsruhe, dem Forschungszentrum Informatik (FZI) und dem Unternehmen lavrio.solutions daran, mithilfe von KI die Arbeit von Disponenten in einer ÖPNV-Leitstelle zu unterstützen.

Digitaler Vorschlagsassistent für die Leitstelle

Die Arbeit in der Leitstelle ist meist von Stress und Zeitdruck geprägt. Denn sobald eine Störung auftritt, muss in kürzester Zeit eine Reihe von Entscheidungen mit teilweise weitreichenden Konsequenzen getroffen werden. Diverse Prozesse von der Unfallmeldung an die Polizei, über die Fahrerkommunikation, die Herstellung eines möglichst funktionalen Ersatzbetriebs bis hin zur Fahrgastinformation gilt es anzustoßen. Viele dieser Maßnahmen werden auf der Basis langjähriger Erfahrung ausgeführt. Im Rahmen des Forschungsprojektes KARL entsteht ein Vorschlagsassistent mit konkreten Handlungsempfehlungen, der das Personal in der Leitstelle entlasten und zu weniger Einschränkungen im Nahverkehr beitragen soll.

Dazu wird der Vorschlagsassistent mit historischen Daten aus dem ITCS (Intermodal Transport Control System), dem zentralen Betriebsleitsystem, trainiert.

Diese umfassen die Daten aus den Fahrzeugen, die Fahrzeugpositionen, Fahrplandaten und die jeweils ausgelösten dispositiven Maßnahmen. Der mit KI-Verfahren trainierte Assistent schlägt dann Maßnahmen vor, die genau auf die jeweilige Situation zugeschnitten sind. Die Verknüpfung mit dem ITCS sorgt dafür, dass dispositive Maßnahmen automatisch ausgelöst werden können, wenn die Disponenten den KI-Vorschlag annehmen. Es handelt sich folglich um ein KI-gestütztes Vorschlagswesen, das die Disponenten entlastet, ihnen aber gleichzeitig die volle Kontrolle über die Vorgänge lässt. Ein besonders wertvoller Einsatzbereich des Systems kann die Einarbeitung von neuem Leitstellenpersonal oder die Unterstützung von Disponenten mit wenig Erfahrung (zum Beispiel Ersatz-Disponenten) sein. So hilft der Vorschlagsassistent dabei, den Betriebsablauf zu optimieren.

Herausforderungen im Projektablauf

Trotz umfangreicher Datenlage stellt eine der größten Herausforderungen immer noch der Aufbau einer konsistenten und umfassenden Datenbasis dar, anhand derer die KI sinnvoll trainiert werden kann. Denn die Informationen sind oft in unterschiedlichen proprietären Systemen vorhanden, die eine Integration erschweren. Des Weiteren erreichen die Informationen die Leitstelle über verschiedene

Kanäle, wie Funk oder Telefon oder auch von externen Stellen, zum Beispiel Polizei oder Feuerwehr. Hierüber erhalten die Disponenten Informationen zu Unfällen und beispielsweise gesperrten Streckenabschnitten. Selbstverständlich werden diese Informationen dokumentiert, jedoch aufgrund extremen Zeitdrucks gegebenenfalls wenig strukturiert und unter Umständen vermischt mit personenbezogenen Daten, was eine automatisierte Auswertung erschwert oder zum Teil unmöglich macht. Nicht digitalisierte Kanäle wie Funk oder Telefon sind überdies für die KI nicht nutzbar. Diesen Herausforderungen versucht man momentan mithilfe umfassender Datenaufbereitung und Integration weiterer zusätzlicher Datenquellen zu begegnen. Dabei wird derzeit der Mehrwert der einzelnen Datenquellen evaluiert und erforscht, welche Auswirkungen diese auf das KI-Modell beziehungsweise auf die Ergebnisse des Vorschlagsassistenten haben.

Weitere Schritte

Im Verlauf des Forschungsprojektes werden in Rücksprache mit dem Leitstellenpersonal der Albtal-Verkehrs-GmbH in Karlsruhe die Anforderungen an den Vorschlagsassistenten weiter konkretisiert und gegebenenfalls weitere Informationsquellen einbezogen. Ziel ist es, Disponenten bestmöglich zu unterstützen,

ohne sie bei weiteren wichtigen Tätigkeiten abzulenken wie zum Beispiel der Kommunikation mit einem in einen Unfall verwickelten Fahrer. Im Vordergrund steht dabei die Praxistauglichkeit im stressigen Alltag in der Leitstelle.

Nachhaltig, nahtlos und intermodal mobil mit KI-Methoden

In einem weiteren Forschungsprojekt – DaKiMo – wird unter Leitung des Fraunhofer IOSB gezeigt, wie Mobilitätsdaten von Endnutzern, Kommunen und Verkehrsbetrieben mithilfe von künstlicher Intelligenz aufbereitet und mit weiteren Daten, etwa zur Verkehrslage oder Witterung, angereichert werden können, um intelligente Informationsdienste für Bürger und Wirtschaft zu schaffen. Das Ziel ist es, eine ressourceneffiziente, intermodale und nachhaltige Mobilität zu fördern und so einen Beitrag zur Verkehrswende zu leisten. Soll dies gelingen, muss die Nutzung einer intermodalen Verbindungskette (wie zum Beispiel die Kombination von [Leih-]Fahrrad, ÖPNV und e-Scooter) mindestens genauso einfach und angenehm sein wie der Griff zum eigenen Autoschlüssel.

Im Rahmen des Projekts bringt INIT ihre Expertise zu den Datenstrukturen im Öffentlichen Nahverkehr ein und liefert Daten aus dem täglichen ÖPNV-Betrieb. Diese werden gemäß der erarbeiteten Anforderungen über maschinelle Lernverfahren aufbereitet, was zu einer Verbesserung der Prognosen, zum Beispiel für Besetzgrade und Fahrtzeiten, führt und den ÖPNV planbarer macht. Umgekehrt werden über DAKIMO weitere Datenquellen eingebunden, die neue Services ermöglichen und von denen beispielsweise der Fahrgast durch Bereitstellung in den ÖPNV-Fahrzeugen über Anzeiger oder App profitiert. So sollen die Services auch Fahrgäste unterstützen, die mobilitätseingeschränkt oder mit Gepäck oder Kinderwagen unterwegs sind, wenn zum Beispiel defekte Infrastruktur ein Weiterkommen an Haltestellen verhindert.

Beide Forschungsprojekte, sowohl KARL als auch DaKiMo, werden durch das Bundesministerium für Bildung und Forschung (BMBF) im Programm „Zukunft der Wertschöpfung – Forschung zu Produktion, Dienstleistung und Arbeit“ gefördert und vom Projektträger Karlsruhe (PTKA) betreut.

Kundenfreundlicher und effizienter Bedarfsverkehr vom ersten bis zum letzten Kilometer

INITs integrierte Buchungs-, Dispositions- und Optimierungslösung für Bedarfsverkehre ist ein weiteres Beispiel für den Einsatz von künstlicher Intelligenz zur Verbesserung des ÖPNV. Das Thema First/Last Mile spielt eine große Rolle im Bestreben, im Zuge der Mobilitätswende mehr Fahrgäste für den ÖPNV zu begeistern. Denn für die Fahrgäste wird eine reibungslose Beförderung vom Ausgangsort der Reise, wie etwa der Wohnung, zum Startpunkt des jeweiligen Verkehrsmittels (First Mile) sowie vom Zielpunkt des jeweiligen Verkehrsmittels (etwa Hauptbahnhof) bis zum eigentlichen Reiseziel (Last Mile) immer entscheidender für die Nutzung des ÖPNV. Das hat zur Folge, dass für Verkehrsbetriebe abgestimmte Bedarfsverkehre, etwa im ländlichen Bereich oder auch in Schwachlastzeiten des Stadtverkehrs, immer wichtiger werden.

Auch bei INITs Bedarfsverkehrslösung spielt künstliche Intelligenz eine wichtige Rolle, denn: Ein KI-basierter Optimierungsalgorithmus verknüpft die Fahrtwünsche der Fahrgäste effizient und kundenfreundlich. Die hohe Performance erlaubt kurze Buchungsfristen auf Basis der Echtzeit-Fahrzeugpositionen sowie On-Demand-Verkehr mit virtuellen Haltestellen, Adressen oder Geokoordinaten. Durch die Vorgabe von Fixpunkten (zur strategischen Positionierung des Fahrzeugs, etwa Betriebshof oder Bahnhof) und -zeiten können mit der Bedarfsverkehrslösung die Anforderungen von Bedarfs- und Linienverkehr erfüllt werden.



Katja Rische
krische@initse.com

Katja Rische ist die treibende Kraft hinter den Online- und Social-Media-Aktivitäten der INIT GmbH. Sie entwickelt Content-Strategien, um relevante Informationen, Produktupdates und Erfolgsgeschichten über verschiedene Plattformen, darunter die Unternehmenswebseite, LinkedIn und X (ehemals Twitter), mit Interessierten zu teilen und damit die Präsenz des Unternehmens im digitalen Raum zu stärken



Dr. Roxana Hess
rhess@initse.com

Dr. Roxana Hess ist Teamleiterin des Forschungsteams bei der INIT und koordiniert die Forschungstätigkeiten rund um die aktuellen Themen in der Mobilitätsbranche. In verschiedenen Forschungsprojekten, unter anderem zu autonomem Fahren, intermodaler Mobilität und Barrierefreiheit, erprobt das Team neue Technologien mit dem Ziel, innovative IT-Systeme zur Optimierung des ÖPNV zu entwickeln.

Fazit

Öffentlicher Nahverkehr ist im Wandel. Verkehrsunternehmen sind als Mobilitätsdienstleister ihrer Region gefordert, mithilfe moderner Informationstechnologie diesen Wandel fahrgastfreundlich voranzutreiben. Maschinelles Lernen und künstliche Intelligenz sind wichtige Hilfsmittel für mehr Fahrgastzufriedenheit und Effizienz. Die Vision einer nahtlos vernetzten und hochgradig optimierten Mobilitätslandschaft rückt näher, und der ÖPNV steht im Zentrum dieser digitalen Transformation in Richtung smarterer und lebenswerter Städte.

Quellen:

- [1] Kompetenzzentrum KARL - Künstliche Intelligenz für Arbeit und Lernen in der Region Karlsruhe:
<https://kompetenzzentrum-karl.de/>
(Letzter Zugriff am 26.10.2023 um 11:55 Uhr)



Ressourcen optimal nutzen, Effizienz steigern, Innovationen schaffen – das Konzept der Smart Cities

Dennis Hering, adesso

Das Konzept der „Smart Cities“ hat in den letzten Jahren beträchtlich an Bedeutung gewonnen und steht im Mittelpunkt städtischer Entwicklungsstrategien weltweit. Angesichts des raschen Urbanisierungstrends und der dringenden Notwendigkeit, den ökologischen Fußabdruck der Städte zu minimieren, bietet dieses Konzept einen integrativen Ansatz, um die Effizienz städtischer Abläufe zu steigern und die Lebensqualität der Bürgerinnen und Bürger zu erhöhen. Mit einem Zusammenspiel aus moderner Informationstechnologie, Datenanalyse und vernetzter Infrastruktur zielen Smart Cities darauf ab, Ressourcen optimal zu nutzen, nachhaltige Lösungen zu fördern und einen dynamischen Raum für Innovationen zu schaffen.

Was bedeutet „Smart City“?

Derzeit gibt es noch keine allgemein anerkannte Definition, was eine Smart City ist. Die Steigerung der Effizienz, Effektivität und Resilienz des Gesamtsystems ist das zentrale Anliegen einer Smart City. Denn Städte spielen eine wichtige Rolle bei der Gestaltung des demografischen Wandels und bei der Bewältigung der Herausforderungen durch den Klimaschutz. Mit Hilfe von Informations- und Kommunika-

tionstechnologien sowie neuen Technologien stellen sich die Bürgerinnen und Bürger, die Wirtschaft und die Verwaltung in einer Smart City diesen Herausforderungen. Ziel ist die Ausschöpfung des größtmöglichen Potenzials zur Verbesserung von Umwelt und Gesellschaft.

Im Kontext eines Konzeptes lässt sich allerdings schon heute die Bedeutung von Smart Cities in verschiedene Aspekte gliedern, auf die im Folgenden eingegangen wird:

Digitale Vernetzung: Das Rückgrat der Smart City

Eine zentrale Rolle im Kontext von Smart Cities spielt die integrierte Informationstechnologie (IT). Damit ist das Zusammenspiel von Technologien gemeint, die Daten sammeln, analysieren und nutzen, um städtische Funktionen und Dienstleistungen zu optimieren. Intelligente Verkehrssysteme in vielen Städten sind eine beispielhafte Anwendung. Hier überwachen Sensoren und

Kameras den Verkehrsfluss. Die gesammelten Daten werden ausgewertet, um Verkehrsregelungen in Echtzeit anzupassen und Informationen über Staus oder freie Parkplätze zu liefern.

Solche Systeme haben viele Vorteile. Sie können die Effizienz städtischer Prozesse steigern. Dadurch werden Ressourcen besser genutzt und Kosten gesenkt. Darüber hinaus ermöglicht die kontinuierliche Erfassung von Daten den Städten eine schnellere Reaktion auf unerwartete Ereignisse und die Unterstützung vorausschauender Planungsstrategien. Im Ergebnis könnte dies zu einer allgemeinen Verbesserung der Lebensqualität für die Menschen in den Städten führen.

Integrierte IT-Systeme zu implementieren, ist jedoch auch mit Herausforderungen verbunden und stößt auf Bedenken. Da die Sammlung großer Datenmengen potenzielle Risiken birgt, bleibt der Datenschutz ein kritischer Punkt. Hinzu kommt, dass es mit hohen Kosten verbunden ist, solche Technologien einzuführen, und dass ihr komplexes Management Fachwissen erfordert. Ein zu starkes Vertrauen in die Technologie kann Städte auch anfällig für IT-Ausfälle oder Cyber-Angriffe machen.

Grüne Initiativen: Nachhaltige Städte der Zukunft

Nachhaltigkeit ist in den letzten Jahrzehnten zu einem zentralen Thema in der globalen Diskussion geworden, da sowohl Regierungen als auch private Organisationen nach Lösungen suchen, um ein Gleichgewicht zwischen wirtschaftlichem Wachstum, sozialem Wohlergehen und dem Schutz der Umwelt für heutige und zukünftige Generationen zu gewährleisten.

Ein konkretes Beispiel für die Anwendung von Nachhaltigkeitsprinzipien findet sich in der Architektur und im Bauwesen. Ein Beispiel dafür sind „grüne Gebäude“. Dies sind Gebäude, die bei ihrer Planung und Errichtung möglichst geringe negative Auswirkungen auf die Umwelt haben und gleichzeitig für ihre Bewohnerinnen und Bewohner gesund sind. Dies kann durch den Einsatz erneuerbarer Energien, durch Wärmedämmung, durch die Nutzung von Regenwasser oder durch die Verwendung umweltfreundlicher Baumaterialien erreicht werden.

Der Nutzen der Nachhaltigkeit ist vielfältig. Zum einen werden natürliche Ressourcen geschont und der ökologische Fußabdruck verkleinert. Dies kann langfristig das Überleben vieler Arten, einschließlich des Menschen, sichern. Außerdem kann ein nachhaltiger Ansatz zu Kosteneinsparungen führen, insbesondere wenn die langfristigen Ausgaben für den Ersatz von Ressourcen oder die Behebung von Schäden berücksichtigt werden. Die Berücksichtigung sozialer Nachhaltigkeitsaspekte kann zu gerechteren Gesellschaften führen und soziale Spannungen abbauen.

Auf der anderen Seite stehen die politischen Entscheidungsträger oft vor der Herausforderung, kurzfristige wirtschaftliche Interessen mit den langfristigen Zielen der Nachhaltigkeit in Einklang zu bringen. In vielen Fällen können nachhaltige Lösungen zunächst teurer sein als konventionelle Alternativen. Dies kann sie in wirtschaftlich schwierigen Zeiten weniger attraktiv machen. Hinzu kommt, dass der Übergang zu nachhaltigeren Praktiken häufig einen kulturellen Wandel erfordert und auf Widerstand stoßen kann, vor allem dann, wenn Arbeitsplätze oder traditionelle Lebensweisen in Gefahr zu sein scheinen.

Trotz dieser Herausforderungen ist klar, dass der Weg zur Nachhaltigkeit beschritten werden muss, wenn die Welt eine lebenswerte Zukunft für künftige Generationen sichern will. Es erfordert



Zukunftssicherheit: Ihr Partner für Digitale Intelligenz

Nachhaltige Technologien für Unternehmen von morgen

Sie haben die Daten, wir die Vernetzung zum Erfolg – individuell für Ihre Bedürfnisse. Mit innovativen Cloud-Lösungen und Oracle-Applikationen, die eine praxisnahe Umsetzung garantieren. Sicher. Einfach. Überall.

Jetzt informieren: www.promatis.de

ORACLE | Partner

PROMATIS

aber ein sorgfältiges Abwägen der Interessen und nicht selten mutige Entscheidungen von Managern auf allen Ebenen.

Infrastruktur 2.0: Intelligente Netzwerke für effizientere Städte

In der modernen urbanen Landschaft spielt die vernetzte Infrastruktur eine immer zentralere Rolle. Dieser Begriff beschreibt die Integration und Verbindung verschiedener physischer und technologischer Systeme mit dem Ziel der Optimierung des Informationsaustausches und der Zusammenarbeit. Ein prominentes Beispiel hierfür ist das „Smart Grid“ oder intelligente Stromnetz. In diesem Zusammenhang wird die herkömmliche Energieinfrastruktur mit Sensoren, Kommunikationstechnologien und automatisierten Steuerungssystemen ausgestattet, die es ermöglichen, den Verbrauch und die Bereitstellung von Energie in Echtzeit aufeinander abzustimmen.

Es gibt eine Reihe von Vorteilen, die mit der Implementierung einer solchen vernetzten Infrastruktur verbunden sind. Durch die optimale Zuweisung von Ressourcen und die Reduzierung von Verschwendung kann sie zur Effizienzsteigerung beitragen. Durch die erhöhte Reaktionsfähigkeit solcher Systeme ist ein schnelles Reagieren auf Probleme möglich und durch den ständigen Fluss von Daten können Entscheidungen in Kenntnis der Sachlage und mit Blick auf die Zukunft getroffen werden. Dies wiederum kann sich positiv auf die Flexibilität der Systeme und die Verbesserung ihrer Fähigkeit zur Anpassung an neue Umstände oder Technologien auswirken.

Die Herausforderungen und möglichen Nachteile müssen jedoch ebenfalls berücksichtigt werden. Durch die Integration vieler Systeme wird die Komplexität erhöht, was wiederum die Verwaltung und Wartung erschwert. Insbesondere im Hinblick auf Cyber-Angriffe birgt eine intensive Vernetzung zudem potenzielle Sicherheitsrisiken. Nicht zuletzt können die anfänglichen Investitionskosten für den Aufbau vernetzter Infrastrukturen hoch sein. Eine übermäßige Abhängigkeit von solchen Systemen könnte sich im Falle von Ausfällen oder Störungen als problematisch erweisen.

Vernetzte Infrastrukturen haben das Potenzial, urbane Räume sowohl hinsichtlich ihrer Effizienz als auch der Lebensqualität ihrer Bewohnerinnen und Bewohner zu verändern. Eine sorgfältige Planung und Abwägung der Vor- und Nachteile ist jedoch für ihre Umsetzung erforderlich.

Mobilität neu definiert: Smarte Lösungen für den Stadtverkehr

Der weltweit fortschreitende Prozess der Urbanisierung stellt Städte und Metropolregionen vor eine Vielzahl von Herausforderungen, insbesondere im Bereich der Mobilität. Die Verbesserung des öffentlichen Personennahverkehrs ist ein zentraler Ansatzpunkt zur Bewältigung dieser Herausforderungen. Bus-Rapid-Transit-Systeme (BRT) sind hierfür ein beispielhaftes Modell. Diese Systeme zeichnen sich dadurch aus, dass sie speziell für Busse reservierte Fahrspuren aufweisen. Dadurch sind die Busse vom allgemeinen Straßenverkehr getrennt und können schneller vorankommen.

Es gibt eine Vielzahl von Vorteilen, die mit der Einführung solcher verbesserten ÖPNV-Systeme verbunden sind. Dazu gehören vor allem die Steigerung der Effizienz durch die Verkürzung der Fahrzeiten und die Erhöhung der Zuverlässigkeit des Angebots. Auch die Umwelt profitiert: Durch die verstärkte Nutzung des ÖPNV werden weniger individuelle Fahrten mit dem Auto zurückgelegt, was zu einer Verringerung des CO₂-Ausstoßes und der Luftverschmutzung führt. Durch den Wegfall der Notwendigkeit, ein eigenes Fahrzeug zu besitzen, können zudem Kosteneinsparungen für den Einzelnen realisiert werden. Ein leistungsfähiges öffentliches Verkehrssystem hat zudem das Potenzial, Staus zu vermindern.

Solche Systeme einzuführen und zu optimieren, ist jedoch nicht ohne Hindernisse. Die anfänglichen Investitionen in die Infrastruktur können zum Teil beträchtlich sein. Hinzu kommen die laufenden Kosten, um diese zu unterhalten und zu warten. Insbesondere in dicht besiedelten Gebieten kann der Raum für zusätzliche oder erweiterte Verkehrslinien knapp sein. Und nicht zuletzt können Veränderungen im Verkehrssystem auch auf den Widerstand verschiedener Interessengruppen stoßen.

Insgesamt bildet ein effizient organisierter öffentlicher Verkehr eine zentrale Säule für die Mobilität der Zukunft im urbanen Raum. Die Vorteile liegen auf der Hand. Es bedarf jedoch einer sorgfältigen Planung und Abwägung, um die bestmöglichen Lösungen für spezifische städtische Kontexte zu finden.

E-Government: Digitale Dienste für die moderne Bürgerin und den modernen Bürger

Im heutigen digitalen Zeitalter verändert der digitale Zugang zu öffentlichen

Dienstleistungen die traditionellen Wege der Interaktion zwischen Bürgerinnen und Bürger und öffentlichen Einrichtungen. E-Government-Portale, wie sie in vielen Kommunen weltweit zu finden sind, helfen den Bürgern dabei, behördliche Angelegenheiten zu erledigen. So können zum Beispiel die Anmeldung einer Geburt, die Beantragung eines Führerscheins oder die Zahlung von Steuern online über diese Plattformen abgewickelt werden.

Diese digitale Transformation ist mit erheblichen Vorteilen verbunden. Sie schafft Bequemlichkeit. Denn Dienstleistungen können unabhängig von Ort und Zeit in Anspruch genommen werden. Darüber hinaus können die Bearbeitungszeiten verkürzt und die Prozesse effizienter gestaltet werden. Die Transparenz für die Bürgerinnen und Bürger steigt: Informationen sind leichter zugänglich und Anträge können besser nachvollzogen werden. Durch den effizienteren Einsatz von Ressourcen verspricht die Digitalisierung zudem langfristig Kosteneinsparungen für die Verwaltung.

Dieser Wandel geht jedoch nicht ohne Herausforderungen einher. Es besteht die Gefahr einer „digitalen Spaltung“, das heißt, ein Teil der Bevölkerung hat keinen Zugang zu digitalen Diensten und ist dadurch im Nachteil. Datenschutz und Datensicherheit sind zentrale Anliegen in der digitalen Welt und technische Ausfälle können ein Hindernis für den Zugang zu dringend benötigten Diensten sein. Hinzu kommt, dass die Einführung neuer Systeme und Prozesse eine Anpassung und Schulung sowohl der Verwaltungsmitarbeiter als auch der Bürger erfordert.

Obwohl der digitale Zugang zu öffentlichen Dienstleistungen erhebliche Chancen zur Modernisierung und Effizienzverbesserung bietet, ist eine umsichtige Umsetzung, die mögliche Herausforderungen berücksichtigt, unabdingbar.

Stimme der Bürgerinnen und Bürger: Partizipative Plattformen in Smart Cities

Die Bürgerbeteiligung, die häufig als Ausdruck der partizipativen Demokratie betrachtet wird, hat sich als ein Instrument etabliert, das es den Gemeinschaften ermöglicht, ihre Stimme in die Entscheidungsprozesse der Politik und Verwaltung einfließen zu lassen. Das Konzept des „Bürgerhaushalts“, bei dem die Kommunen über die Verteilung eines Teils des städtischen

Haushalts entscheiden, ist ein bekannter Anwendungsfall. Dieses Modell wird weltweit in verschiedenen Städten umgesetzt. Es ermöglicht den Bürgerinnen und Bürgern, direkten Einfluss auf die Prioritäten der kommunalen Finanzen zu nehmen.

Die positiven Effekte einer solchen Beteiligung können vielfältig sein: Entscheidungen, die unter Einbeziehung einer breiten Basis getroffen werden, haben oftmals eine höhere Legitimität und Akzeptanz. Unterschiedliche Perspektiven und kollektives Wissen können zu fundierteren Lösungen führen. Auch das Gemeinschaftsgefühl kann durch diese Form der Beteiligung gestärkt werden: Die Bürgerinnen und Bürger werden aktiver in das Gemeinwesen eingebunden.

Es gibt aber auch Herausforderungen, wie sie bei vielen Ansätzen auftreten. Der Prozess der Bürgerbeteiligung kann zeitaufwändig sein. Er kann Ressourcen binden und den politischen Entscheidungsprozess verlangsamen. Ein weiterer kritischer Punkt ist, dass die Beteiligung möglicherweise verzerrt ist, da nur bestimmte Gruppen teilnehmen und andere nicht gehört werden. Hinzu kommt, dass komplexe Themen, die ein tiefgehendes Fachwissen erfordern, weniger geeignet sind, um breit diskutiert zu werden.

Diese Dynamiken machen deutlich, dass Bürgerbeteiligung sowohl Vor- als auch Nachteile hat. Sie kann das demokratische Gefüge stärken und zu Entscheidungen führen, die besser akzeptiert werden. Dabei muss die Bürgerbeteiligung aber sorgfältig geplant und durchgeführt werden, da die Themen einfach zugänglich aufbereitet werden müssen und auch sichergestellt werden muss, dass es sich nicht nur um auswirkungslose Umfragen handelt.

Moderne Sicherheitstechnologien: Schutz und Überwachung in Smart Cities

Sicherheit ist ein zentrales Thema in der modernen Gesellschaft. Dabei bezieht sich die Sicherheit nicht nur auf den physischen Schutz, sondern erstreckt sich auch auf Bereiche wie den Schutz der Privatsphäre und die Cyber-Sicherheit. Die zunehmende Installation von Überwachungskameras auf öffentlichen Plätzen und in Verkehrsmitteln ist ein Beispiel dafür.

Solche Maßnahmen dienen in erster Linie der Gewährleistung der Sicherheit der Bürgerinnen und Bürger. Sie können vor

Verbrechen abschrecken, eine schnelle Reaktion der Sicherheitsdienste ermöglichen und als Beweismittel bei Strafverfolgungsmaßnahmen dienen.

Es gibt jedoch auch eine Kehrseite. Permanente Überwachung kann als Eingriff in die Privatsphäre empfunden werden. Technologische Systeme sind anfällig für Fehler, was in einigen Fällen dazu führen kann, dass ein falscher Alarm ausgelöst wird. Außerdem besteht die Gefahr, dass eine übermäßige Technologieabhängigkeit von anderen präventiven Sicherheitsmaßnahmen ablenkt.

Daraus lässt sich schließen, dass technologische Sicherheitsmaßnahmen zwar erhebliche Vorteile bieten, aber sorgfältig gegen mögliche Nachteile abgewogen werden müssen. Entscheidend ist ein ausgewogener Ansatz, der sowohl den Schutz als auch die Privatsphäre der Bürgerinnen und Bürger berücksichtigt.

Innovation als Antrieb: Wirtschaftliches Wachstum in intelligenten Städten

Die wirtschaftliche Entwicklung, die sich auf den Fortschritt und die Erweiterung eines Wirtschaftssystems bezieht, ist ein zentrales Anliegen vieler Nationen. Ein herausragendes Beispiel ist Südkorea. Das Land hat in den letzten Jahrzehnten des 20. Jahrhunderts einen bemerkenswerten Wandel von einer Agrarwirtschaft zu einer führenden Industrienation vollzogen. Dem Land ist es gelungen, seinen Lebensstandard, gemessen an Indikatoren wie dem Bruttoinlandsprodukt, durch gezielte Politik, Investitionen in Bildung und die Stärkung von Exportindustrien, deutlich zu erhöhen.

Diese Entwicklung bringt zahlreiche Vorteile mit sich. Die Bevölkerung kann durch eine verbesserte Infrastruktur in Bereichen wie Gesundheit, Bildung und Verkehr von einem höheren Lebensstandard profitieren. Zudem ermöglicht ein höherer wirtschaftlicher Entwicklungsstand oft eine stärkere globale Vernetzung, die den Zugang zu internationalen Märkten und neuesten Technologien erleichtert.

Wirtschaftliche Entwicklung bringt jedoch nicht nur Vorteile. Wenn nicht alle Gesellschaftsschichten gleichermaßen profitieren, kann dies zu wachsenden sozialen Ungleichheiten führen. Zudem kann ein rasches industrielles Wachstum erhebliche Auswirkungen auf die Um-

welt haben. Diese reichen von Umweltverschmutzung bis hin zur Übernutzung natürlicher Ressourcen. Nicht zuletzt besteht die Gefahr, dass im Zuge der Modernisierung traditionelle Kulturen und Lebensweisen verloren gehen.

Der Blick auf Korea und andere wirtschaftlich aufstrebende Länder zeigt, dass Fortschritt und Wachstum in einem ausgewogenen Verhältnis stehen sollten, um die positiven Aspekte der wirtschaftlichen Entwicklung zu maximieren und die potenziellen negativen Auswirkungen auf ein Mindestmaß zu beschränken.

Fazit

Smart Cities zielen darauf ab, moderne Technologien zu nutzen, um die Lebensqualität der Bürgerinnen und Bürger zu verbessern und städtische Prozesse effizienter zu gestalten. Smart Cities integrieren Informations- und Kommunikationstechnologien zur Optimierung der Ressourcennutzung, zur Steuerung des Verkehrs, zur Senkung des Energieverbrauchs und zur Verbesserung des Zugangs zu öffentlichen Dienstleistungen. Neben den Vorteilen bringen Smart Cities aber auch Herausforderungen mit sich. Dies gilt insbesondere für den Datenschutz und die Sicherheit. Zusammenfassend lässt sich sagen, dass Smart Cities das Potenzial haben, den städtischen Raum radikal zu verändern, wobei es wichtig ist, das richtige Gleichgewicht zwischen technologischem Fortschritt und ethischen Erwägungen zu finden.



Dennis Hering
dennig.hering@adesso.de

Dennis Hering ist ein erfahrener Softwareentwickler. Er verfügt über fundierte Kenntnisse in JavaScript/TypeScript, PHP und C#. Bei adesso SE hat er sein Know-how als Softwarearchitekt weiter ausgebaut. Weitere Schwerpunkte sind Angular, Vue, SvelteKit, Laravel, ASP.NET Core und Cloud Services wie Microsoft Azure, AWS Amplify und GCP Firebase. Als zertifizierter Microsoft-Trainer, Referent und Prüfer bei der IHK Ostwestfalen in Bielefeld sowie Redner auf Konferenzen und Meetups hat er wertvolle Einblicke in die Branche gegeben.



Privacy by Design als Basis für Datenschutz in Smart Cities

Yahya Mohamed Mao, Swiss GRC

In den Städten der Zukunft, in denen Datenströme unaufhörlich fließen und Technologie unsere städtischen Lebensräume prägt, rückt der Datenschutz mehr denn je in den Mittelpunkt. Dies ist entscheidend, um die Privatsphäre der Bürger zu schützen und gleichzeitig die Vorteile der Smart-City-Technologien zu realisieren.

In einer Ära, in der die Digitalisierung und technologische Fortschritte unsere Welt rasant verändern, nehmen Smart Cities eine herausragende Position ein. Die Vision, unsere urbanen Lebensräume durch intelligente Technologien zu optimieren, verspricht eine erhöhte Lebensqualität und eine gesteigerte Effizienz in unse-

ren Städten. Die Vorstellung von nahtlos fließendem Verkehr, optimierter Energieversorgung und maßgeschneiderten öffentlichen Diensten fasziniert uns. Doch in dieser Zukunftsvision lauert eine unsichtbare Gefahr – der potenzielle Verlust unserer Privatsphäre. Genau an diesem Punkt tritt das Konzept „Privacy by De-

sign“ auf die Bühne. Der Ansatz stellt sich den drängenden Fragen des Datenschutzes in Smart Cities auf innovative Weise. Privacy by Design ist mehr als nur eine reaktive Maßnahme zur Einhaltung von Datenschutzrichtlinien. Es ist eine tiefdurchdachte Philosophie, die Datenschutz von Grund auf in die Struktur von

Smart-City-Systemen einbettet. Anstatt Datenschutz als nachträgliche Ergänzung zu behandeln, wird er integraler Bestandteil des gesamten Gestaltungsprozesses. Dieser Ansatz zielt darauf ab, die Balance zwischen technologischem Fortschritt und der Bewahrung der Privatsphäre der Bürger zu finden.

Smart Cities: Grundlagen und Realitäten einer visionären Zukunft

Bis zum Jahr 2050 wird voraussichtlich mehr als 70 % der Weltbevölkerung in urbanen Gebieten leben [1]. Angesichts jahrzehntelangen (Welt-)Bevölkerungswachstums und unkontrollierter städtischer Ausdehnung auf der ganzen Welt haben Smart Cities an Bedeutung gewonnen. Sie sind nicht länger nur eine utopische Vision für die Zukunft, sondern eine unaufhaltsame Realität, die sich global verbreitet. Smart Cities sind urban gestaltete Lebensräume, in denen fortschrittliche Technologien und innovative Ansätze eingesetzt werden, um das Leben der Bürger zu verbessern und die Effizienz städtischer Abläufe zu steigern. Dies umfasst eine breite Palette von Anwendungen, von intelligenter Verkehrssteuerung bis zur Förderung nachhaltiger Energieversorgung und der Verbesserung der städtischen Sicherheit. Eine zentrale Säule von Smart Cities ist die umfassende Vernetzung von Geräten und Infrastrukturen [2], die es ermöglicht, Echtzeitdaten zu sammeln und zu analysieren. Diese Daten bilden die Grundlage für datengetriebene Entscheidungsfindung und unterstützen Städte dabei, Ressourcen effizienter zu nutzen und Herausforderungen schneller anzugehen. Zugleich betonen Smart Cities die aktive Beteiligung der Bürger, indem sie Plattformen und Technologien bereitstellen, die es Einwohnern ermöglichen, an der Gestaltung ihrer Stadt teilzuhaben und Feedback zu geben. Die Vision von Smart Cities geht jedoch über Technologie hinaus und erstreckt sich auf das Streben nach einer nachhaltigen und lebenswerten Zukunft für alle Bewohner. In diesem Kontext spielt Datenschutz eine entscheidende Rolle, nicht nur als rechtliche Anforderung, sondern als grundlegendes Element, um das Vertrauen der Bürger in Smart-City-Initiativen zu gewinnen und die Wahrung der Privatsphäre sicherzustellen.

Die Rolle des Datenschutzes in Smart Cities

In der Welt der Smart Cities ist es unerlässlich zu erkennen, dass intelligente Städte in erster Linie menschliche Städte sein sollten. Technologie allein definiert nicht, was eine Stadt „smart“ macht [3]. Stattdessen sollte die Technologie den Bürgern einen greifbaren Nutzen bringen und ihr Leben bereichern. Dieser Grundsatz ist von zentraler Bedeutung, um sicherzustellen, dass Smart Cities nicht nur innovativ, sondern auch lebenswert sind. Smart Cities sollten sich nicht nur auf die Erfassung von Daten und die Automatisierung von Prozessen konzentrieren, sondern auch auf die Schaffung einer offenen und partizipativen Umgebung. Bürgerbeteiligung und -feedback sollten aktiv gefördert werden, um sicherzustellen, dass Technologie den Bedürfnissen der Menschen gerecht wird. Die Integration von Technologie in eine Stadt sollte daher stets von einem menschenzentrierten Ansatz begleitet werden. Daher ist Datenschutz in Smart Cities nicht nur eine Frage der Konformität, sondern auch eine ethische und praktische Notwendigkeit. Es geht darum, die Balance zwischen dem Potenzial datengetriebener Innovationen und dem Schutz individueller Privatsphäre zu finden. Letztendlich geht es darum, das Vertrauen der Bürger in die Smart-City-Initiativen zu stärken und eine Stadt zu schaffen, in der Innovation und Privatsphäre Hand in Hand gehen. Datenschutz in diesem Kontext hat mehrere zentrale Dimensionen [4]:

- Schutz der Privatsphäre der Bürger: Smart Cities sammeln eine Vielzahl persönlicher Daten, sei es durch Überwachungskameras im öffentlichen Raum, Sensoren zur Verkehrssteuerung oder Datenerfassung in den Bereichen Gesundheitswesen und Bildung. Datenschutzvorschriften und -richtlinien müssen sicherstellen, dass diese Daten angemessen geschützt und anonymisiert werden, um die Privatsphäre der Bürger zu wahren.
- Transparenz und Kontrolle: Bürger sollten die Kontrolle über ihre eigenen Daten haben und darüber informiert sein, wie ihre Daten erfasst, gespeichert und genutzt werden. Transparenz in Bezug auf die Datenverarbeitung und klare Mechanismen für die Einwilligung und Ablehnung sind entscheidend, um das Vertrauen der Öffentlichkeit zu erhalten.

- Sicherheit vor Datenmissbrauch: Angesichts der zunehmenden Bedrohungen im Bereich der Cyberkriminalität müssen Smart Cities sicherstellen, dass die erfassten Daten vor unbefugtem Zugriff geschützt sind. Datenschutz sollte auch Mechanismen zur schnellen Identifizierung von Datenschutzverletzungen und zur Behebung von Sicherheitslücken einschließen.
- Ethik und Fairness: Datenschutz in Smart Cities sollte auf ethischen Grundsätzen beruhen. Dies schließt die Vermeidung von Diskriminierung und Ungerechtigkeit bei der Nutzung von Daten ein. Die Entwicklung von Richtlinien zur datenbasierten Entscheidungsfindung sollte faire Ergebnisse für alle Bürger gewährleisten.
- Datennutzung für das Gemeinwohl: Während Datenschutz den Schutz der Rechte und der Privatsphäre der Bürger sicherstellt, sollte er gleichzeitig die Nutzung von Daten für das Gemeinwohl ermöglichen. Daten können dazu beitragen, Städte effizienter und lebenswerter zu gestalten, beispielsweise bei der Bewältigung von Verkehrsproblemen, der Verbesserung der Luftqualität oder der Anpassung an den Klimawandel.

Die Stadt der Zukunft hat ein Privatsphäre-Dilemma

Das Konzept von Smart Cities stützt sich stark auf die umfassende Sammlung von personenbezogenen Daten, was erhebliche Bedenken hinsichtlich ihrer Verwendung aufwirft. Trotz der zahlreichen Vorteile, die Smart-City-Projekte bieten, bedrohen sie weltweit das Recht auf Privatsphäre der Bürger. Die Sorge um die Privatsphäre in Smart Cities ist berechtigt, da das gesamte Konzept auf einer umfassenden Datensammlung [5] basiert. Sensoren, Kameras und vernetzte Geräte sind überall in der Stadt präsent und erfassen kontinuierlich Informationen zu verschiedenen Aspekten des städtischen Lebens, angefangen bei Verkehrsdaten bis hin zu persönlichen Bewegungsprofilen der Bürger. Diese Daten sind entscheidend für die effiziente Steuerung der städtischen Abläufe und die Bereitstellung optimierter Dienstleistungen, bergen jedoch auch das Potenzial für Missbrauch. Eine der größten Datenschutzbedenken in Smart Cities ist die damit verbundene umfassende Datensammlung selbst. Menschen befürchten zu Recht den Verlust

ihrer Privatsphäre, da sie in einer hochvernetzten Umgebung praktisch rund um die Uhr überwacht werden.

Die Möglichkeit, umfassend überwacht zu werden, könnte das individuelle Recht auf Anonymität und Privatsphäre [6] gefährden. Ein weiteres zentrales Anliegen ist die Identifizierung und Überwachung von Bürgern. Die gesammelten Daten können genutzt werden, um Einzelpersonen zu identifizieren und ihre Aktivitäten zu verfolgen. Dies wirft ernsthafte Fragen hinsichtlich der Privatsphäre und der individuellen Freiheit auf, da Menschen möglicherweise zögern, ihre Meinungen und Handlungen frei auszudrücken, wenn sie befürchten, überwacht zu werden. Gleichzeitig stellen Datensicherheit und Datenschutzverletzungen eine weitere große Herausforderung dar. Die tatsächlichen Auswirkungen hängen oft von der Art und Weise der Überwachung, den damit verbundenen Rechtsvorschriften und den Schutzmaßnahmen ab, die ergriffen werden, um die Rechte und Freiheiten der Bürger zu wahren. Es ist wichtig, eine ausgewogene Balance zwischen Sicherheit und individuellen Rechten zu finden.

Die zunehmende Vernetzung und Speicherung großer Datenmengen erhöht das Risiko von Datenverletzungen und Cyberangriffen, die persönliche Informationen und sensible Daten gefährden können. Je mehr Daten gesammelt und gespeichert werden, desto attraktiver werden sie für Angreifer, da diese Daten eine Fülle von Informationen über Personen, Organisationen und Systeme enthalten können. Angreifer sehen in diesen Daten die Möglichkeit, Identitäten zu stehlen, finanzielle Betrugereien durchzuführen, Erpressung auszuüben oder geistiges Eigentum zu stehlen, was erhebliche Schäden und Missbrauchsmöglichkeiten birgt. Um diesen Datenschutzbedenken zu begegnen, sind klare Datenschutzgesetze und -richtlinien von entscheidender Bedeutung. Diese sollten sicherstellen, dass die Vorteile der Technologie nicht auf Kosten der Privatsphäre und Sicherheit der Bürger gehen. Transparenz, Kontrolle und Datenschutzmaßnahmen sind erforderlich, um die Privatsphäre der Bürger zu schützen, während gleichzeitig datengetriebene Innovationen gefördert werden.

Die Zukunft von Smart Cities wird stark davon abhängen, wie wir diese Datenschutzbedenken angemessen berücksichtigen und eine ausgewogene Balance zwi-

schen Fortschritt und Privatsphäre finden. Datenschutz sollte nicht nur eine rechtliche Anforderung sein, sondern auch eine ethische und praktische Notwendigkeit. Die ethische Dimension des Datenschutzes betont die moralische Verpflichtung, die Privatsphäre und die Rechte der Menschen zu respektieren und zu schützen. Es geht darum, die Prinzipien der Privatsphäre, Transparenz und Fairness in der Datenverarbeitung zu wahren, selbst wenn es keine rechtlichen Verpflichtungen gibt. Dies trägt dazu bei, das Vertrauen der Bürger in die Smart-City-Initiativen zu stärken und eine Stadt zu schaffen, in der Innovation und Privatsphäre Hand in Hand gehen.

Smart und sicher dank Privacy by Design?

Die Tatsache, dass Technologie zur treibenden Kraft städtischer Innovationen geworden ist, wirft die Frage nach dem Schutz der Privatsphäre in einer vernetzten Welt immer drängender auf. Die zunehmende Integration von Technologie in das städtische Leben bringt zahlreiche Vorteile mit sich, aber sie wirft auch erhebliche Fragen zur Datensicherheit und zum Datenschutz auf. In diesem Kontext gewinnt das Konzept der Privacy by Design zunehmend an Bedeutung und könnte den Schlüssel zu einer smarten und dennoch sicheren Zukunft für Städte weltweit darstellen. Privacy by Design ist ein integrativer Ansatz, der Datenschutz und Datensicherheit von Anfang an in die Planung und Umsetzung von Smart-City-Initiativen einbezieht.

Anhand von Kernprinzipien wie Transparenz, Kontrolle, Sicherheit und Ethik zielt das Konzept darauf ab, Datenschutz nicht als nachträgliche Überlegung zu behandeln, sondern von Beginn an in den Entwicklungsprozess [7] einzubinden. Eine zentrale Säule von Privacy by Design in Smart Cities ist die Schaffung von Transparenz und die Gewährleistung von Bürgerkontrolle über ihre eigenen Daten. Dies bedeutet, dass Bürger umfassend über die Datensammlung und -nutzung informiert werden und die Möglichkeit haben sollten, ihre Einwilligung zu geben oder abzulehnen. Datenschutzrichtlinien sollten klar und verständlich sein und Bürgern sollte das Recht zugesprochen werden, ihre eigenen Daten einzusehen und zu überwachen. Ein weiterer zentraler Aspekt ist die Berücksichtigung ethischer Grundsätze bei der Datennutzung. Dies schließt die

Vermeidung von Diskriminierung und ungerechter Datenverwendung ein, um sicherzustellen, dass Daten zum Wohl aller Bürger eingesetzt werden. Für Unternehmen, die diese Grundsätze erfolgreich umsetzen und fördern, eröffnet sich eine strategische Chance, denn dieser Ansatz steht im Einklang mit den gesetzlichen Vorschriften und trägt gleichzeitig dazu bei, die Marktposition durch gesteigerte Glaubwürdigkeit zu stärken [7].

Privacy by Design erfordert außerdem die Implementierung robuster Sicherheitsmaßnahmen [8], um Daten vor unbefugtem Zugriff und Missbrauch zu schützen. Dies umfasst die Anonymisierung und Verschlüsselung von Daten, die kontinuierliche Überwachung von Sicherheitslücken und die schnelle Reaktion auf Datenschutzverletzungen. Privacy by Design ist somit von entscheidender Bedeutung, um Smart Cities nicht nur innovativ, sondern auch sicher und datenschutzfreundlich zu gestalten. Die konsequente Implementierung dieses Konzepts ist unerlässlich, um das Vertrauen der Bürger in Smart-City-Initiativen zu stärken und eine Zukunft zu gestalten, in der Technologie und Privatsphäre harmonisch koexistieren können.

Ausblick: Die Zukunft von Datenschutz in Smart Cities

Die Integration von Privacy by Design in Smart Cities ist zweifellos ein entscheidender Schritt zur Bewältigung der Datenschutzherausforderungen in einer zunehmend vernetzten Welt. Doch wie wird sich die Zukunft des Datenschutzes in Smart Cities gestalten? Diese Frage erfordert einen Blick auf die sich entwickelnden Trends und auf die sich verändernde Landschaft des Datenschutzes.

In den kommenden Jahren wird die Bedeutung von Datenschutz und Privatsphäre in Smart Cities weiter zunehmen. Bürgerinnen und Bürger werden sich zunehmend ihrer Datenschutzrechte und -bedenken bewusst und fordern verstärkt Transparenz und Kontrolle über ihre persönlichen Daten. Dies wird die Entwicklung strengerer Datenschutzgesetze und -richtlinien vorantreiben, die sicherstellen sollen, dass Smart Cities die Privatsphäre der Menschen achten und schützen.

Technologie wird in Smart Cities eine noch prominentere Rolle spielen, da sie innovative Lösungen zur Verbesserung des städtischen Lebens bieten kann. Künstliche

Intelligenz, maschinelles Lernen und das Internet der Dinge werden dazu beitragen, die Effizienz und Lebensqualität in Städten zu steigern. Doch mit diesen technologischen Fortschritten werden auch neue Datenschutzherausforderungen einhergehen. Datenschutzrichtlinien und -praktiken müssen daher kontinuierlich weiterentwickelt werden, um mit den sich verändernden Technologien Schritt zu halten.

Privacy by Design ist zukunftsweisend, da sie nicht nur auf den gesetzlichen Rahmen achtet, sondern auch einen hohen ethischen Anspruch verfolgt. In einer Zeit, in der die ethische Nutzung von Daten und der Schutz vor Diskriminierung durch algorithmische Entscheidungsfindung immer mehr an Bedeutung gewinnen, ist es von entscheidender Bedeutung, Smart Cities auf einer Grundlage von Ethik und Fairness zu gestalten. Die Vision von Smart Cities ruht auf vier tragenden Säulen:

- **Regierung:** Die Aufgabe der Regierung besteht darin, den rechtlichen Rahmen für Datenschutz und Antidiskriminierung zu schaffen. Sie setzt die Normen fest, die die Nutzung von Daten in Smart Cities regulieren.
- **Unternehmen:** Unternehmen spielen eine Schlüsselrolle, indem sie sicherstellen müssen, dass sie nicht nur den rechtlichen Anforderungen gerecht werden, sondern auch ethischen Normen folgen. Sie tragen die Verantwortung dafür, technische Sicherheitsvorkehrungen zu treffen und den Schutz der Daten ihrer Nutzerinnen und Nutzer zu gewährleisten.
- **Zivilgesellschaft:** Das Einverständnis und das Vertrauen der Zivilgesellschaft sind von größter Bedeutung, da sie ihre Daten für die notwendigen Innovationen zur Verfügung stellen müssen. Die Partizipation und das Verständnis der Zivilgesellschaft in Bezug auf Datenschutz- und Ethikfragen sind entscheidend für den Erfolg von Smart City-Initiativen.
- **Technologie:** Die technologische Infrastruktur bildet die vierte Säule und ermöglicht die Umsetzung innovativer Lösungen in Smart Cities.

Die Zusammenarbeit und Koordination dieser vier Säulen ist von entscheidender Bedeutung, um eine ausgewogene Balance zwischen Innovation und Datenschutz in Smart Cities zu finden. Eine Vision, die auf

ethischen Grundsätzen und rechtlichen Vorschriften basiert, ist der Schlüssel zum Erfolg in der Gestaltung unserer zukünftigen Städte.

Insgesamt wird Privacy by Design in Smart Cities eine zentrale Rolle dabei spielen, eine Zukunft zu gestalten, in der Technologie und Privatsphäre harmonisch koexistieren. Sie wird dazu beitragen, das Vertrauen der Bürger in Smart-City-Initiativen zu stärken und sicherzustellen, dass Innovation und Datenschutz Hand in Hand gehen. Die Smart Cities von morgen werden nicht nur technologisch fortschrittlich sein, sondern auch einen hohen Standard an Privatsphäre und Datensicherheit bieten. Diese Entwicklung ist unaufhaltsam und erfordert eine fortlaufende Anpassung unserer Datenschutzpraktiken und -gesetze, um die Chancen der Technologie zu nutzen, ohne die Privatsphäre zu gefährden.

Quellen

- [1] Sprus, C. M. (2020) "Wie sieht die Zukunft der Smart Cities aus?" EY, https://www.ey.com/de_ch/digital/what-does-the-future-of-smart-cities-look-like (Zugriffsdatum: 19.09.2023)
- [2] Gil-Garcia, J. Ramon, Chen, Tzu hao, & Gasco-Hernandez, Mila. (2023). "Smart City Results and Sustainability: Current Progress and Emergent Opportunities for Future Research." MDPI - Sustainability, <https://www.mdpi.com/2071-1050/15/10/8082> (Zugriffsdatum: 21.09.2023)
- [3] Neubert, H. (o.J.) "Smart Cities Should Be Human Cities." Gensler Dialogue, <https://www.gensler.com/dialogue/33/smart-cities-should-be-human-cities> (Zugriffsdatum: 22.09.2023)
- [4] Bismillah, J. (o.J.) "Big Data Privacy and Ethics." Educative, <https://www.educative.io/answers/big-data-privacy-and-ethics> (Zugriffsdatum: 24.09.2023)
- [5] Smith, B. (2020) "Why Smart City Data Threatens Citizens' Right to Privacy." Urbanet, <https://www.urbanet.info/why-smart-city-data-treatens-citizens-right-to-privacy/> (Zugriffsdatum: 25.09.2023)
- [6] Jalali-Sohi, M., Dr. (2023) "Smart Cities und Privatsphäre: Datenschutzrisiken in einer hypervernetzten Welt." Industry of Things, <https://www.industry-of-things.de/smart-cities-und-privatsphaere-datenschutzrisiken-in-einer-hypervernetzten-welt-a-012e47019d084220c635c4ded0a55550/> (Zugriffsdatum: 27.09.2023)
- [7] Kuqi, B. & Mao, Y.M. (2023) "Das neue Datenschutzgesetz als Chance Computeworld, [<schutzgesetz-chance-2884345.html> \(Zugriffsdatum: 28.09.2023\)](https://www.computerworld.ch/business/best-practice/neue-daten-

</div>
<div data-bbox=)

- [8] Frackiewicz, M. (2023) "Privacy by Design for Smart Cities: Balancing Innovation and Privacy." TS2 Space, <https://ts2.space/en/privacy-by-design-for-smart-cities-balancing-innovation-and-privacy/> (Zugriffsdatum: 29.09.2023)



Yahya Mohamed Mao
yahya.mao@swissgrc.com

Yahya Mohamed Mao ist Head Business Development & Marketing bei Swiss GRC, einem führenden Schweizer Technologieunternehmen in den Bereichen Governance, Risk & Compliance (GRC). Yahya ist Sprecher auf internationalen Konferenzen, darunter AIBC Summit Malta, DMEA Connecting Digital Health, The Wall Street Journal und Affiliate Grand Slam Dubai. Darüber hinaus ist er ein aktiver Beitragender in Branchenpublikationen und schreibt regelmäßig über Themen wie FinTech, GRC, Künstliche Intelligenz, Digital Marketing und Smart Cities.



Oracle Enterprise Manager 13c – Patching und Upgrade

Birgit Arndt, SVA System Vertrieb Alexander

Die effiziente Administration des Oracle Enterprise Manager Cloud Control ist für ein reibungsloses Funktionieren und einen angemessenen Schutz vor potenziellen Sicherheitslücken unerlässlich. Regelmäßige Patches sind daher von großer Bedeutung. Darüber hinaus ist das Upgrade von Version 13.4 auf Version 13.5 ein wichtiger Schritt, um von den neuesten Funktionen und Verbesserungen zu profitieren. In diesem Artikel werden die wesentlichen Schritte für ein erfolgreiches Patchen sowie der Prozess des Upgrades erläutert.

Sowohl das Patchen als auch das Upgrade erfordern zunächst eine gründliche Planung und Vorbereitung, um einen reibungslosen Ablauf zu gewährleisten.

Patchen des Oracle Enterprise Managers (OEM)

Beim OEM gibt es verschiedene Komponenten, die regelmäßig gepatcht werden sollten:

- Repository-Datenbank
- OEM
- Agenten
- Weblogic-Server
- Betriebssystem

Hier wird nur auf das Patchen des OEM, der Agenten und des Weblogic-Servers eingegangen. Um die benötigten Patches für die Oracle-Produkte zu finden, empfiehlt sich der Einstieg über die hilfreiche My Oracle Support (MOS) Note: „Patch Set Updates for Oracle Products (Doc ID 854428.1)“. Eine direkte Quelle für die aktuellen Patches des OEM und der Agenten ist die MOS-Note „*Overview of the Enterprise Manager Proactive Patch Program (Doc ID 822485.1)*“.

Bevor mit dem Patchen begonnen wird, ist es wichtig, die entsprechenden READMEs zu lesen und die erforderlichen Vorbereitungen zu treffen. Dazu gehört beispielsweise das Aktualisieren der Patch-Tools „OPatch“ und „OMSPatcher“. Gerade bei „OPatch“ für den OEM gibt es einige Neuerungen, die beachtet werden sollten, da andernfalls „OPatch“ nicht mehr ordnungsgemäß funktionieren kann und möglicherweise eine Wiederherstellung des Oracle Homes erforderlich ist. Zusätzlich wird empfohlen, Backups der Repository-Datenbank, der OEM-Konfiguration und des Oracle Home des OEM zu erstellen.

Sobald alle Voraussetzungen erfüllt sind, kann das Patchen des OEM in wenigen Schritten durchgeführt werden. Eine vereinfachte Anleitung dazu ist in *Listing 1* dargestellt.

Sollten beim Patchen Probleme auftreten, ist es erforderlich, die entsprechenden Logdateien zu überprüfen und die aufgetretenen Fehler zu beheben. Anschließend kann das Skript „run_script_singleoms_resume.sh“ gestartet werden,

um den Patchvorgang fortzusetzen. Das Skript befindet sich im Verzeichnis „OEM_ORACLE_HOME/omspatcher_storage/oms_session/scripts_<Datum_Uhrzeit>“.

Nach erfolgreichem Patchen des OEM wird empfohlen, die Agenten zeitnah zu patchen. Dies kann bequem über die Webconsole des OEM durchgeführt werden. Dazu muss der Patch des Agenten zusammen mit den Metadaten als XML-Datei über die Webconsole in die Software Library hochgeladen werden. Die XML-Datei wird gemeinsam mit der ZIP-Datei des Patches heruntergeladen (*siehe Abbildung 1*).

Die *Abbildungen 2, 3 und 4* veranschaulichen den Vorgang des Uploads des Agenten-Patches mit der Metadaten-XML-Datei in die Software Library.

Die Agenten werden mithilfe eines Patch-Plans gepatcht, der ebenfalls über die Webconsole erstellt wird. Der hochgeladene Patch für den Agenten kann über den Menüpfad „Enterprise“ – „Provisioning & Patching“ – „Saved Patches“ gefunden werden. Durch Anklicken der Patchnummer gelangt man zur Übersichtsseite des Patches. Von dort aus gestaltet sich die Erstellung des Patch-Plans äußerst unkompliziert, indem man den Anweisungen des Patch-Wizards folgt. Aus Erfahrung empfiehlt es sich, den Patch-Plan zunächst nur für den sogenannten „central agent“ zu erstellen, also dem Agenten, der zusammen mit dem OEM installiert ist, und den Patch zunächst dort anzuwenden. Dies dient als Test, um mögliche Probleme beim Patchen zu identifizieren. Denn wenn beim „central agent“ Fehler auftreten, ist es wahrscheinlich, dass sie auch bei anderen Agenten vorkommen. Bei der Erstellung des Patch-Plans ist es ebenfalls wichtig, den Haken bei „OPatch Upgrade“ in Schritt 3 „Deployment Options“ zu entfernen. Standardmäßig ist dieser Haken gesetzt und führt in der Re-

gel zu einem Fehler während der Patch-Analyse. Zudem wurde für Agenten der Version 13.5 „OPatch“ durch das Tool „Agentpatcher“ ersetzt, wodurch der Haken vorerst keine Funktion mehr hat (*siehe Abbildung 5*).

Nachdem der „central agent“ erfolgreich gepatcht wurde, kann der Patch-Plan als Patch-Template gespeichert werden, das als Vorlage für alle anderen Agenten dienen kann. Es ist ratsam, in einem Patch-Plan nicht mehr als acht bis zehn Agenten hinzuzufügen. Aus bisher ungeklärten Gründen kann es zu Abbrüchen des Patch-Vorgangs kommen, wenn mehr als zehn Agenten in einem Patch-Plan zusammengefasst sind.

Aber wenn der OEM und die Agenten gepatcht wurden, ist der Patch-Vorgang noch nicht abgeschlossen. Der Weblogic-Server, dessen Patches oft wichtige Sicherheitslücken schließen, muss ebenfalls aktualisiert werden. Jedoch gestaltet sich die Suche nach dem richtigen Patch Set Update bei MOS oft als zeitaufwendig. Ich persönlich merke mir nicht, welche Weblogic-Server-Version mit dem OEM installiert ist, und suche über das Online beziehungsweise Offline Patching nach dem passenden Patch des Weblogic-Servers. Die *Abbildungen 6 bis 8* zeigen, wie es über die Webconsole des OEM in einfachen Schritten möglich ist, den richtigen PSU zu finden – in diesem Beispiel im Offline-Modus.

Nachdem die Katalog-ZIP-Datei erfolgreich hochgeladen wurde, wird ein Job namens „Refresh From My Oracle Support“ erstellt, der die Patch-Empfehlungen für die entsprechenden Ziele zur Verfügung stellt. Die benötigten Patches sind nun über den Menüpfad „Enterprise“ – „Provisioning & Patching“ – „Patches & Updates“ – „All Recommendations“ zu finden. In dieser Ansicht kann das Patch Set Update (PSU) für den Weblogic-Ser-

```
cd <Patchverzeichnis>
omspatcher apply -analyze
emctl stop oms
omspatcher apply
opatch lspatches
emctl start oms
emcli sync
```

Listing 1: Vereinfachte Anleitung für das Patchen des OEM

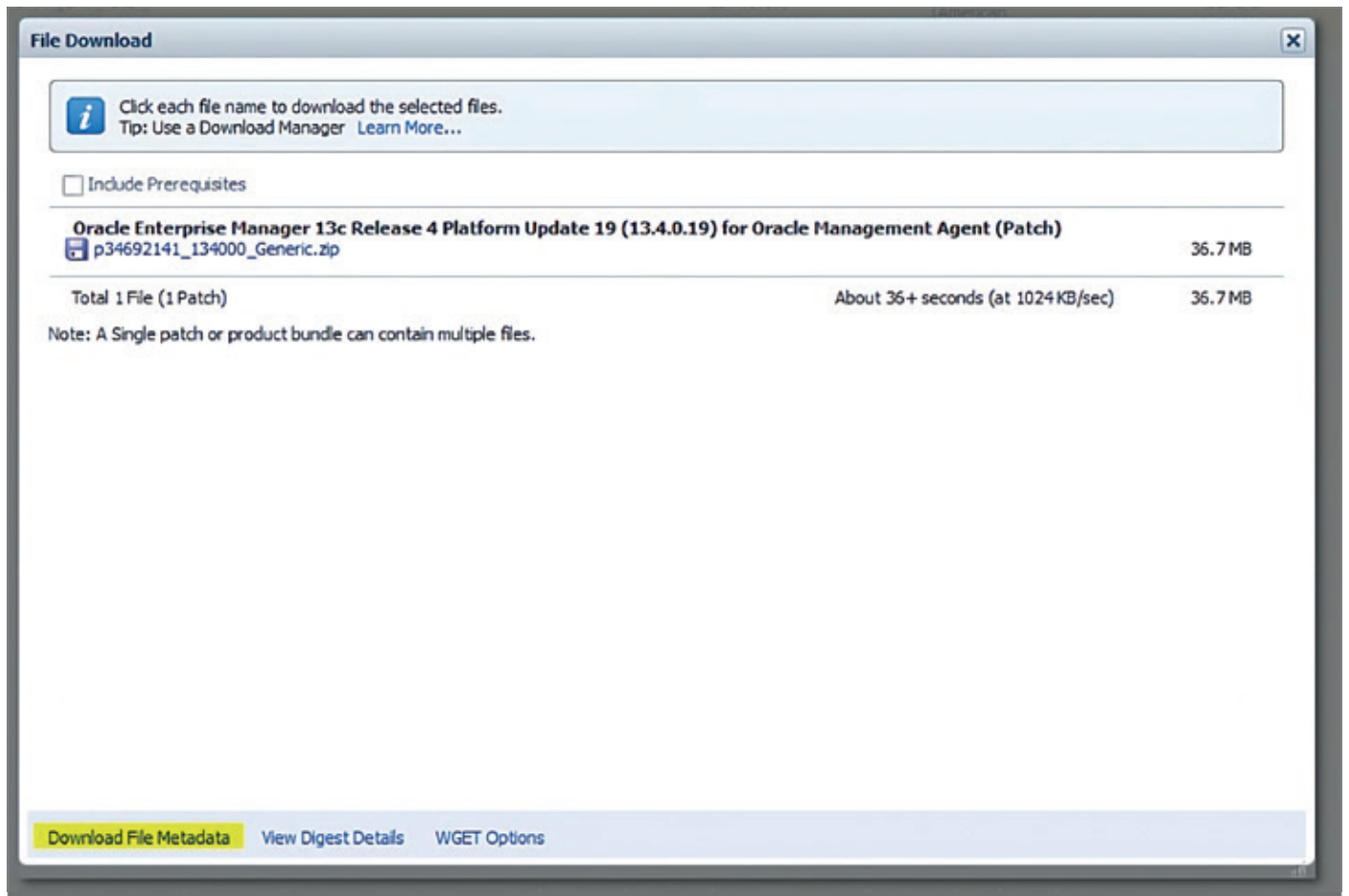


Abbildung 1: Metadaten XML-Datei herunterladen (© My Oracle Support)

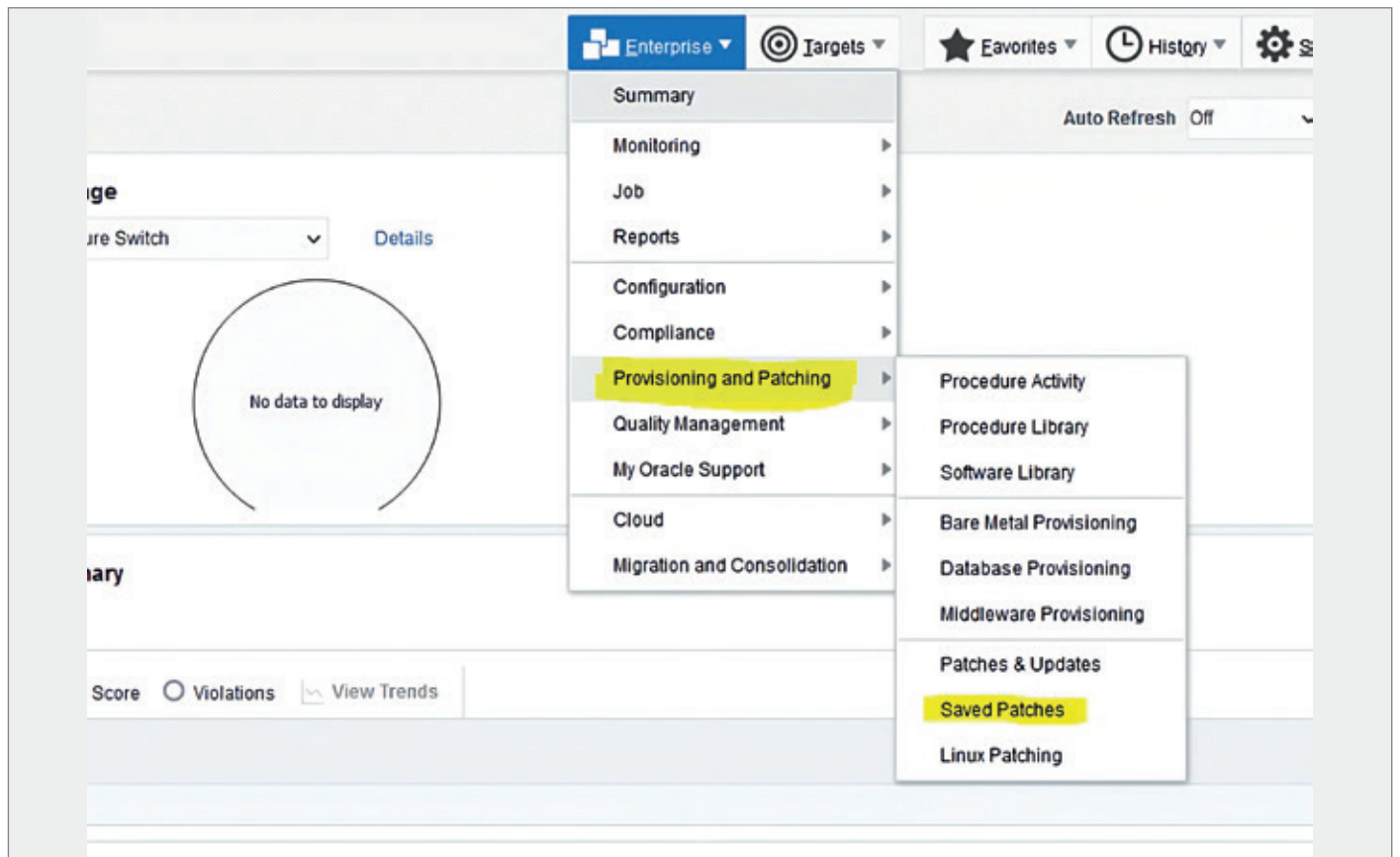


Abbildung 2: Schritt 1 Agent Patch hochladen (© Oracle, Screenshot OEM)

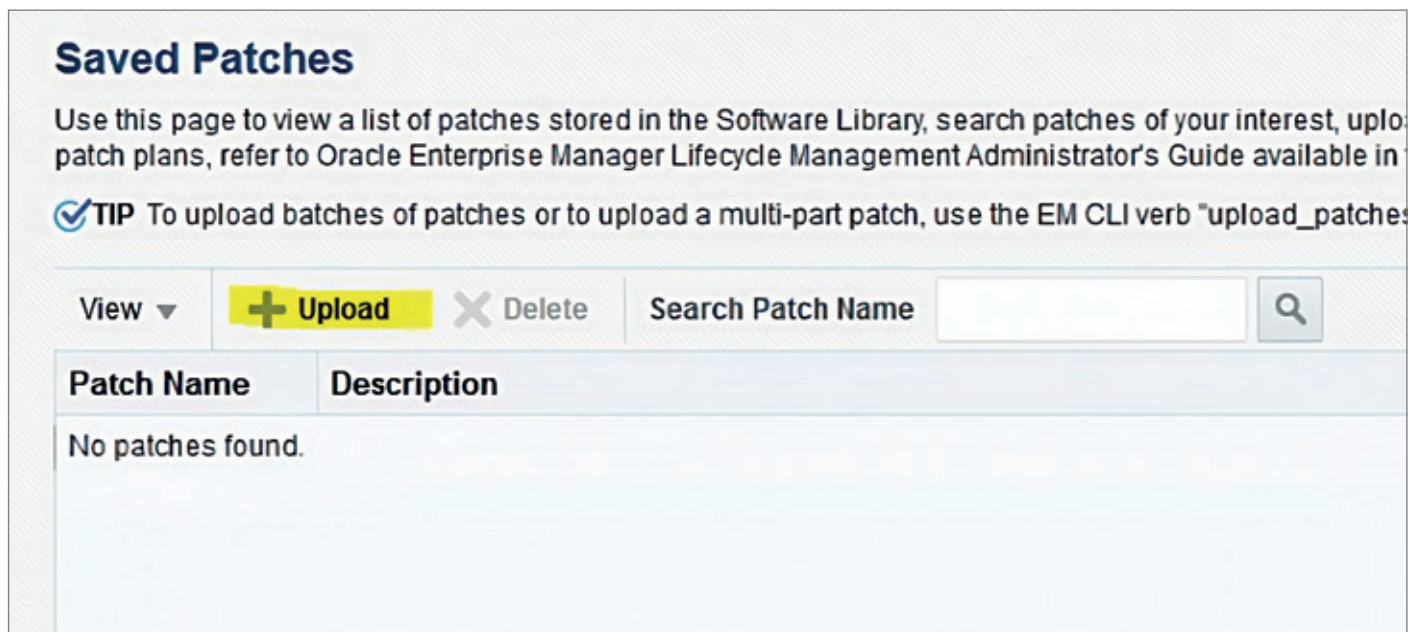


Abbildung 3: Schritt 2 Agent Patch hochladen (© Oracle, Screenshot OEM)

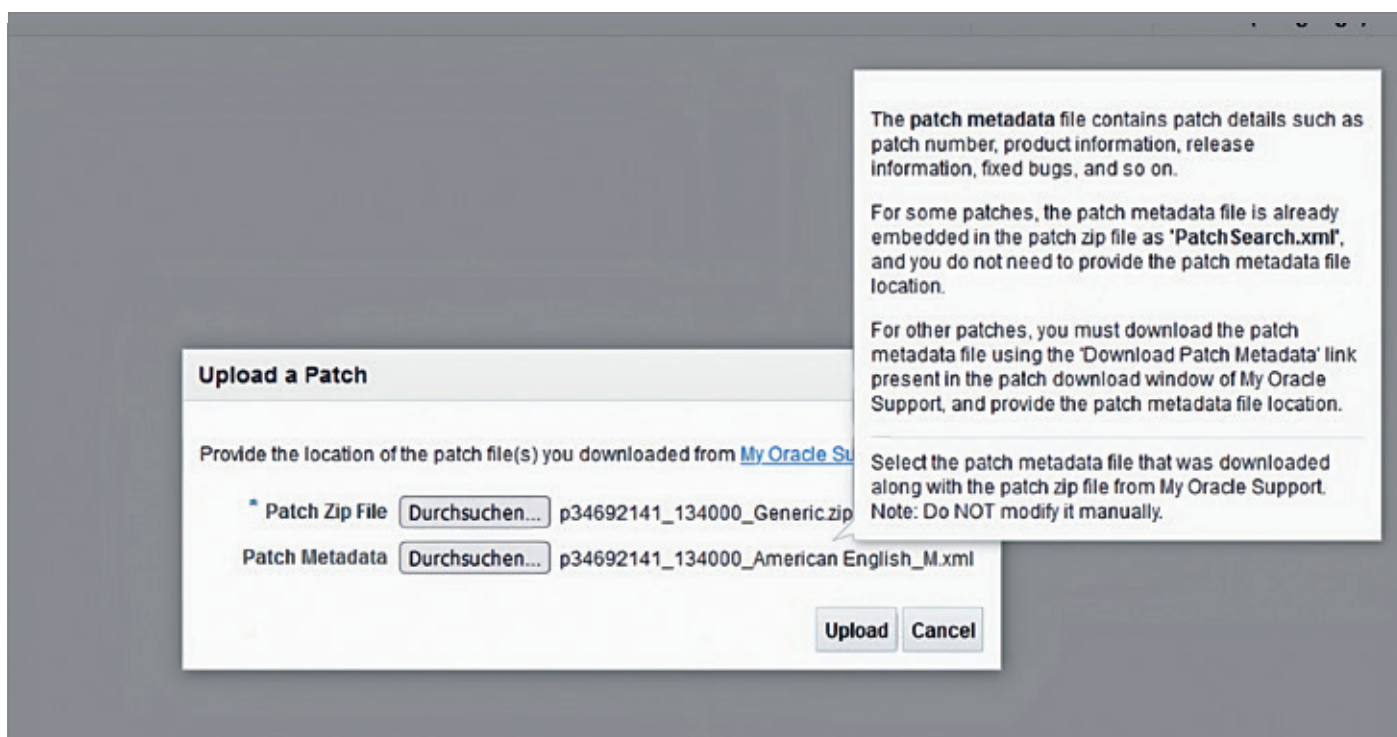


Abbildung 4: Schritt 3 Agent Patch hochladen (© Oracle, Screenshot OEM)

ver identifiziert werden (siehe Abbildung 9). Mithilfe der gefundenen Patch-Nummer kann der erforderliche Weblogic Patch über die Patch-Suche in MOS gefunden werden.

Bei genauerem Lesen des dazugehörigen READMEs stellt man jedoch fest, dass das Einspielen des PSUs alles andere als einfach ist. Das README weist darauf hin, dass es sehr empfehlenswert ist, auch das entsprechende

JDK zu aktualisieren. Dieser Schritt ist ein aufwendiger, aber einmaliger Vorgang. Eine weitere Empfehlung besteht darin, das neue JDK nicht im Oracle Home des OEM zu entpacken, sondern es in einem separaten Verzeichnis zu installieren (siehe auch MOS-Note „How to Install and Maintain the Java SE Installed or Used with FMW 11g/12c/14.1.1 Products (Doc ID 1492980.1)“ und siehe Abbildung 10).

Für die Aktualisierung des JDK müssen einige Dateien im Oracle Home und im Domain Home des OEM bearbeitet werden:

- `$ORACLE_HOME/oui/.globalEnv.properties`
- `$ORACLE_HOME/install/envVars.properties`
- `$ORACLE_HOME/oui/bin/compareInventory.sh`

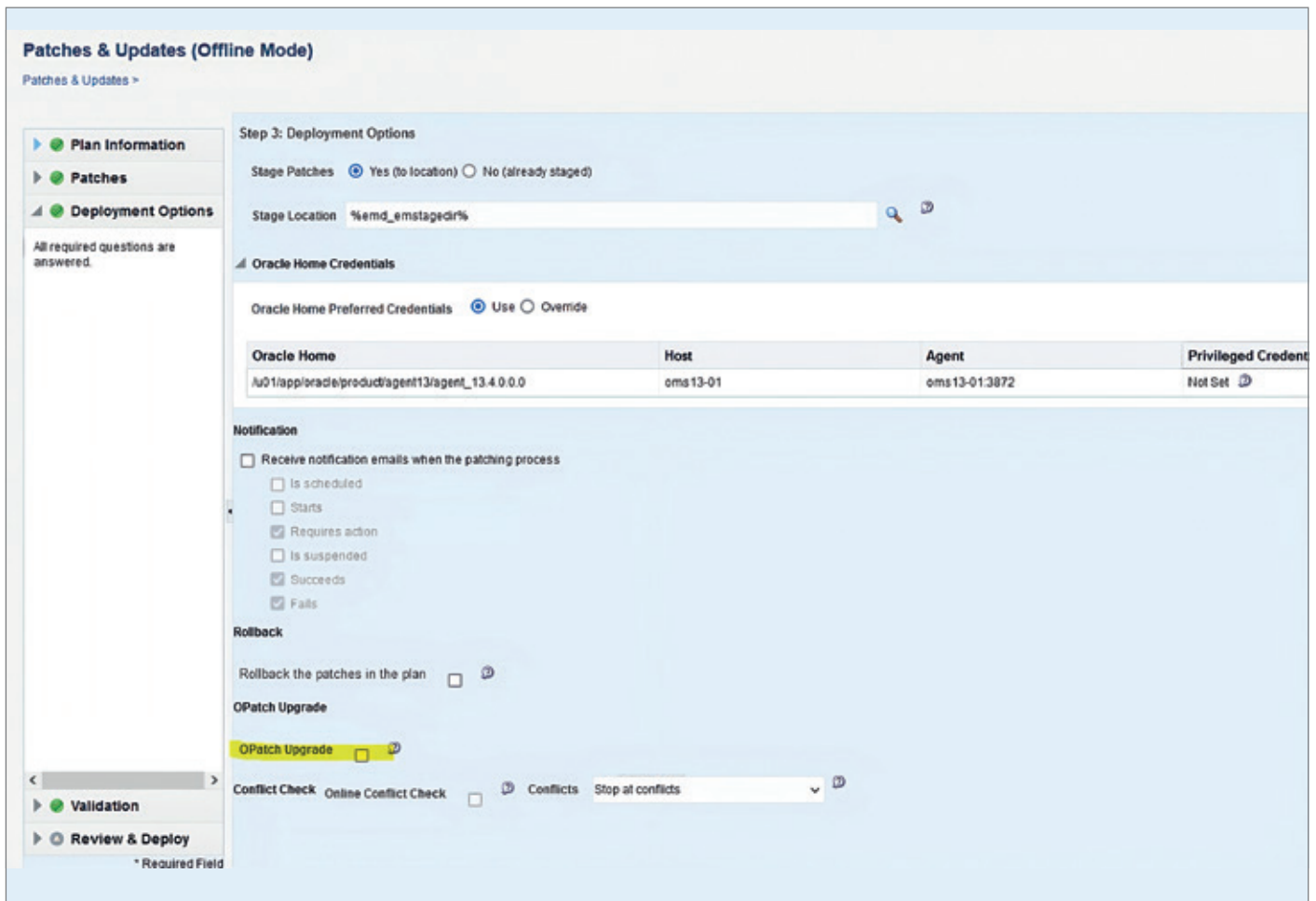


Abbildung 5: OPatch Upgrade herausnehmen (© Oracle, Screenshot OEM)

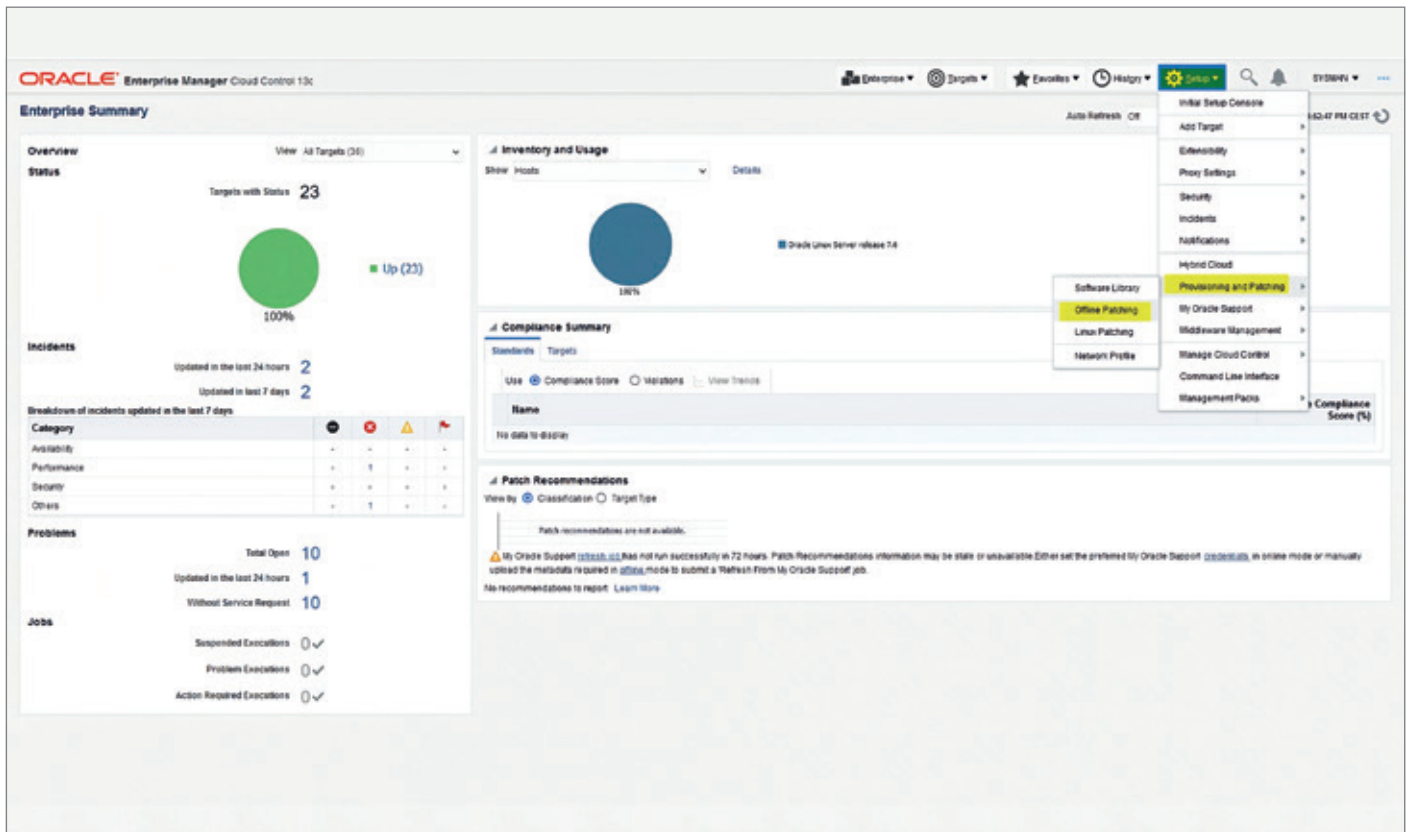


Abbildung 6: 1. Schritt zum richtigen Weblogic Patch (© Oracle, Screenshot OEM)



Abbildung 7: URL der ZIP-Datei zum Herunterladen (© Oracle, Screenshot OEM)

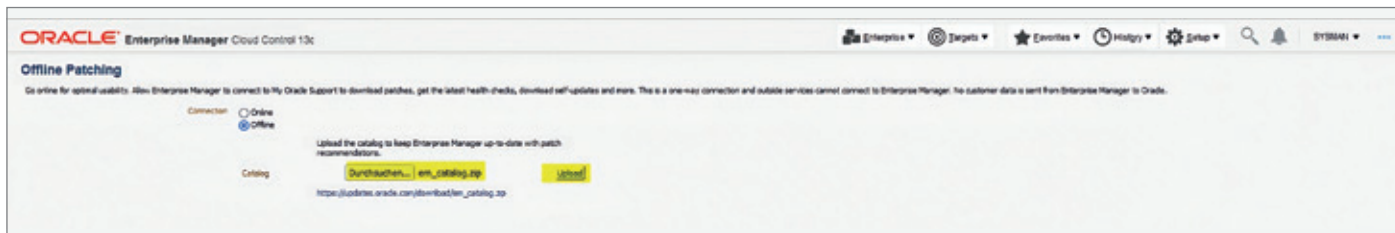


Abbildung 8: Upload der heruntergeladenen ZIP-Datei (© Oracle, Screenshot OEM)

Patches & Updates (Offline Mode)

Patch Name	Description	Release	Platform	Classification	Product Family	Detected	...
35122395	Coherence 12.2.1.3 Cumulative Patch 22 (12.2.1.3.22)	12.2.1.3.0	Generic Platform	Security	Oracle Coherence	In 1+ hour	...
32097173	WLS OVERLAY PATCH FOR 12.2.1.3.0 OCT 2020 PSU	12.2.1.3.201001	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
32441456	GUIA UPGRADE PATCH FOR WLS 12.2.1.3.0	12.2.1.3.201217	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33290784	JDBC 12.2.0.1 FOR CPUJAN2022 (WLS 12.2.1.3)	12.2.1.3.0	Generic Platform	Security	JDBC	In 1+ hour	...
33598515	ADR FOR WEELOGIC SERVER 12.2.1.3.0 CPU JUL 2022	12.2.1.3.0	Linux x86-64	Security	Oracle WebLogi...	In 1+ hour	...
33691226	Fix for Bug 33691226	12.2.1.3.210929	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33735326	Fix for Bug 33735326	12.2.1.3.211222	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33791665	Fix for Bug 33791665	12.2.1.3.211222	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
34339878	WLS PSU DOESNT PATCH ORACLE OWSAPCOM-BEA-CORE-APACHE-LOG4J.JAR IN NON WLS HOMES	12.2.1.3.220329	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
34339878	WLS PSU DOESNT PATCH ORACLE OWSAPCOM-BEA-CORE-APACHE-LOG4J.JAR IN NON WLS HOMES	12.2.1.3.220620	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
35247514	WLS PATCH SET UPDATE 12.2.1.3.230402	12.2.1.3.0	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
32097173	WLS OVERLAY PATCH FOR 12.2.1.3.0 OCT 2020 PSU	12.2.1.3.201001	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
32441456	GUIA UPGRADE PATCH FOR WLS 12.2.1.3.0	12.2.1.3.201217	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33290784	JDBC 12.2.0.1 FOR CPUJAN2022 (WLS 12.2.1.3)	12.2.1.3.0	Generic Platform	Security	JDBC	In 1+ hour	...
33598515	ADR FOR WEELOGIC SERVER 12.2.1.3.0 CPU JUL 2022	12.2.1.3.0	Linux x86-64	Security	Oracle WebLogi...	In 1+ hour	...
33691226	Fix for Bug 33691226	12.2.1.3.210929	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33735326	Fix for Bug 33735326	12.2.1.3.211222	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33791665	Fix for Bug 33791665	12.2.1.3.211222	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
34339878	WLS PSU DOESNT PATCH ORACLE OWSAPCOM-BEA-CORE-APACHE-LOG4J.JAR IN NON WLS HOMES	12.2.1.3.220329	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
34339878	WLS PSU DOESNT PATCH ORACLE OWSAPCOM-BEA-CORE-APACHE-LOG4J.JAR IN NON WLS HOMES	12.2.1.3.220620	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
35247514	WLS PATCH SET UPDATE 12.2.1.3.230402	12.2.1.3.0	Generic Platform	Security	Oracle WebLogi...	In 1+ hour	...
33014823	OHS (NATIVE) BUNDLE PATCH 12.2.1.3.210617	12.2.1.3.0	Linux x86-64	Other Recomm...	Oracle HTTP Se...	In 1+ hour	...
34523467	OHS (NATIVE) BUNDLE PATCH 12.2.1.3.220823	12.2.1.3.0	Linux x86-64	Security	Oracle HTTP Se...	In 1+ hour	...

Rows Selected 1 Columns Hidden 2

★ Patch 35247514: WLS PATCH SET UPDATE 12.2.1.3.230402

Abbildung 9: Patch-Ergebnisse auflisten und Weblogic PSU identifizieren (© Oracle, Screenshot OEM)

Important: ORACLE_HOME/oui/.globalEnv.properties is centrally used as a pointer to the JDK directory.

- This is expected to be a valid JDK directory **outside** of the FMW/WLS Oracle home for use with WebLogic Server and your applications.
- Ensure this is **not** using ORACLE_HOME/oracle_common/jdk, as this is a jre-only location for internal tools.
- This is to avoid OPatch getting into self-patching mode and inventory corruption issues.

Abbildung 10: Empfehlungen für JDK-Update (© Oracle, Auszug MOS-Note 1492980.1)

MANDATORY STEP IF UPGRADING FROM 13.4 - DO NOT SKIP

If you are upgrading to 13.5, from 13.4 RU10 or later, it is necessary to upgrade to latest available Release Update for OEM 13.5. This is to ensure that there is no loss of functionality post-upgrade and also to be on the latest version in 13.5 using a single maintenance window.

To achieve this you should use the "Upgrade software only with plug-ins and Configure Later" method, which allows you to upgrade to the latest 13.5 RU.

For detailed instructions, refer to this document: 13.5: How to Apply Release Update on the OMS During the Install/Upgrade [Note 2810169.1](#)

NOTE: Upgrading from 13.4 RU10 or later to 13.5 base release is NOT SUPPORTED, and there will be a loss of functionality and it is irreversible.

Here is the table stating the upgrade path to be taken:

Current OEM Release	Upgrade Path to OEM 13.5
13.3, 13.4 RU9 or Earlier	Upgrade to 13.5 Base release or later
13.4 RU10, RU11 or RU12	Upgrade to 13.5 RU01 or later
13.4 RU13	Upgrade to 13.5 RU02 or later

As a best practice, irrespective of the Release Update that is applied on 13.4, consider upgrading to the latest available Release Update for 13.5.

Abbildung 11: OEM-Migrationspfad (© Oracle, MOS-Note 2761728.1)

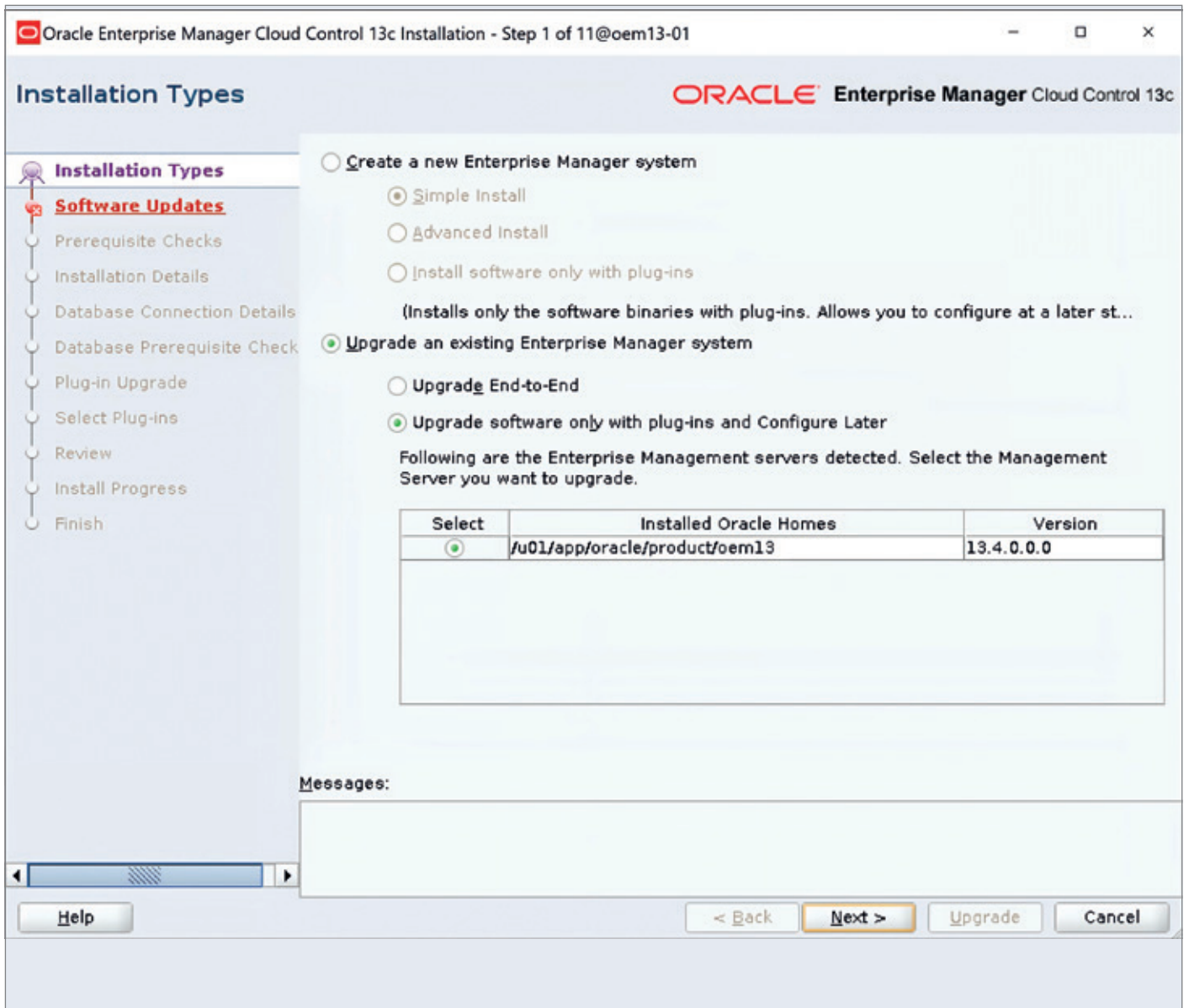


Abbildung 12: Upgrade software only with plug-ins and configure later (© Oracle, Screenshot OEM)



Abbildung 13: Prüfung, zusätzliche Plug-ins (© Oracle, Screenshot OEM)

- `$ORACLE_HOME/oui/bin/launch.sh`
Die Datei kann auch unter dem Namen "install.sh" zu finden sein.
 - `$ORACLE_HOME/oui/bin/setProperty.sh`
 - `$ORACLE_HOME/oui/bin/viewInventory.sh`
 - `$ORACLE_HOME/oracle_common/common/bin/commBaseEnv.sh` (JAVA_HOME wird aus der Datei `#{MW_HOME}/oui/bin/getProperty.sh` ausgelesen)
 - `$ORACLE_HOME/oracle_common/common/bin/commEnv.sh`
 - (keine Anpassungen erforderlich)
 - `$ORACLE_HOME/oracle_common/common/bin/commExtEnv.sh`
 - (keine Anpassungen erforderlich)
 - `$ORACLE_HOME/oracle_common/common/bin/setHomeDirs.sh` (JAVA_HOME wird aus der Datei `#{MW_HOME}/oui/bin/getProperty.sh` ausgelesen)
 - `$_DOMAIN_HOME/bin/setDomainEnv.sh`
 - `$_DOMAIN_HOME/bin/setNMJavaHome.sh`
 - `$_DOMAIN_HOME/nodemanager/nodemanager.properties`
- Nachdem das JDK aktualisiert wurde, kann der Weblogic-Server gepatcht werden. Der Ablauf dieses Vorgangs wird in *Listing 2* dargestellt.
- ### Fazit zum Patchen des OEM
- Zusammenfassend lässt sich zum Patchen des OEM folgendes festhalten:
- Es ist sehr wichtig, die READMEs zu den Patches sorgfältig zu lesen und zu befolgen.
 - Das Patchen des OEM und der dazugehörigen Agenten ist in der Regel einfach und schnell. In den meisten Fällen muss lediglich der OMSPatcher ausgetauscht werden.
 - Das Patchen des Weblogic-Servers erfordert etwas mehr Aufwand. Wenn das JDK außerhalb des Oracle Homes des OEM abgelegt wird, muss dieser Schritt nur einmal durchgeführt werden. Oftmals ist auch ein Austausch des OPatch-Tools erforderlich.
 - Das Patchen aller Komponenten erfordert eine Downtime, was ungünstig ist, wenn zu diesem Zeitpunkt OEM-Jobs ausgeführt werden, zum Beispiel für Backup-Jobs. Während der Downtime ist die Überwachung durch den OEM eingeschränkt.
 - Nach der Downtime und dem Abschluss des Blackouts werden häufig alle während dieser Zeit aufgetretenen Incidents gleichzeitig gemeldet.

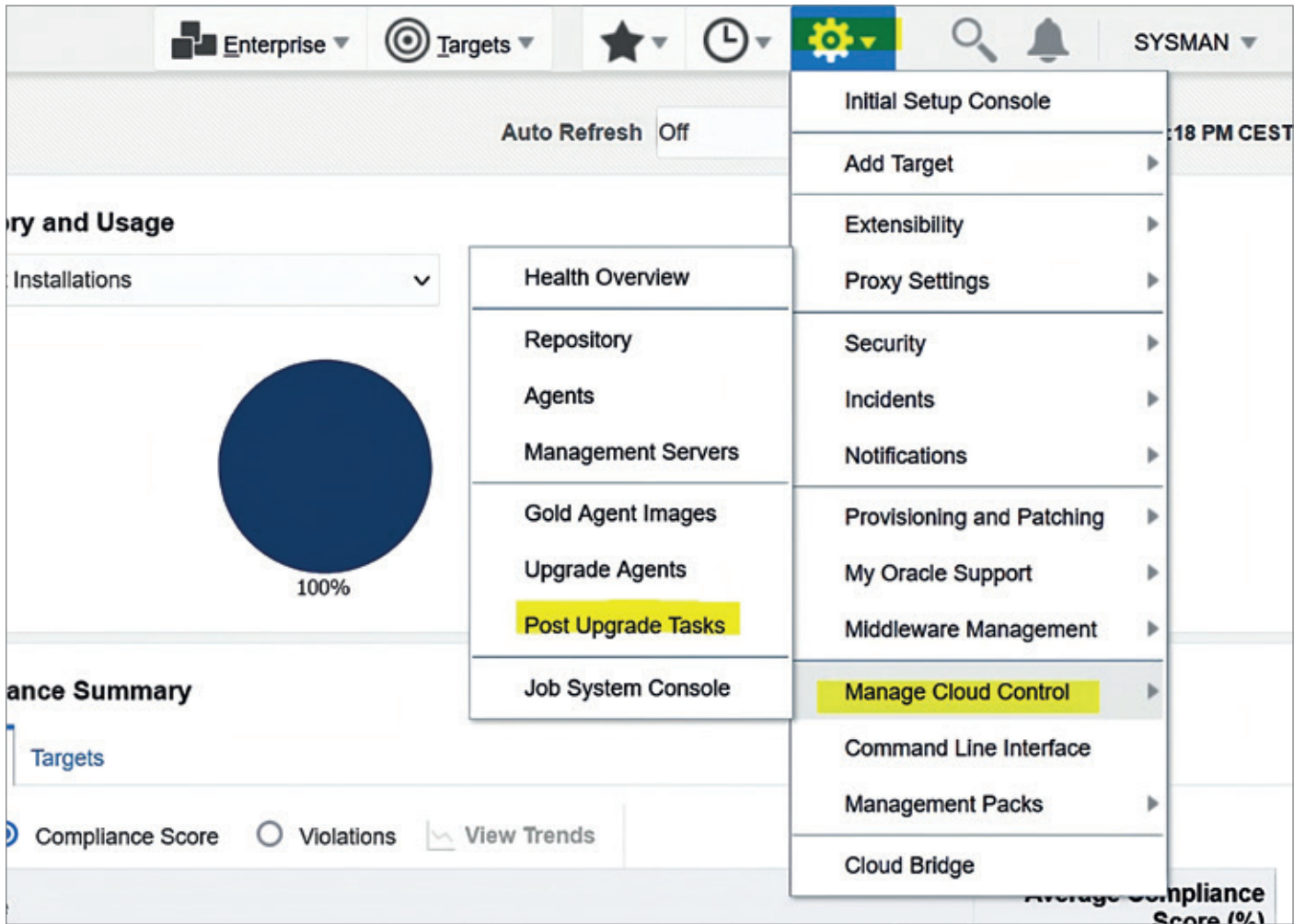


Abbildung 14: OEM Post Upgrade Tasks (© Oracle, Screenshot OEM)

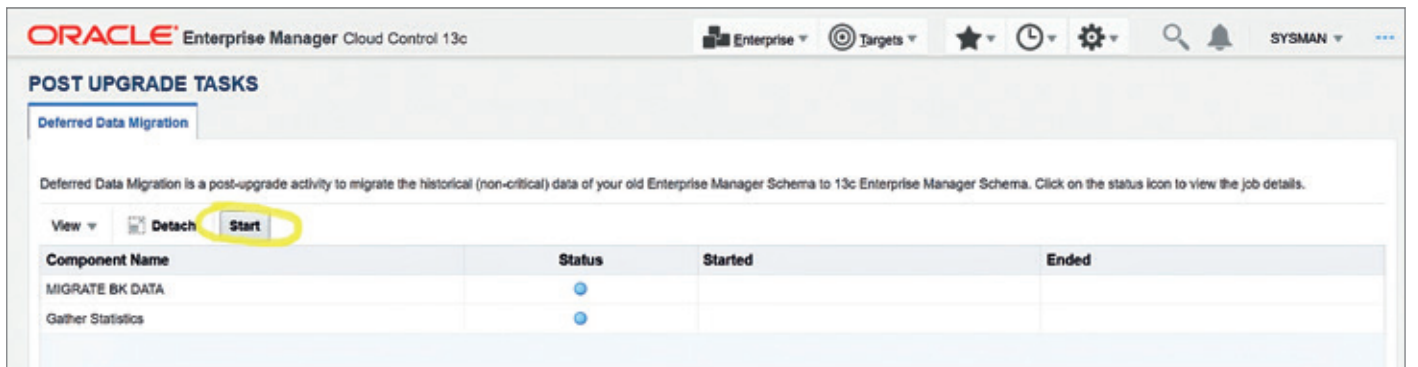


Abbildung 15: OEM Post Upgrade Tasks, Jobs starten (© Oracle, Screenshot OEM)

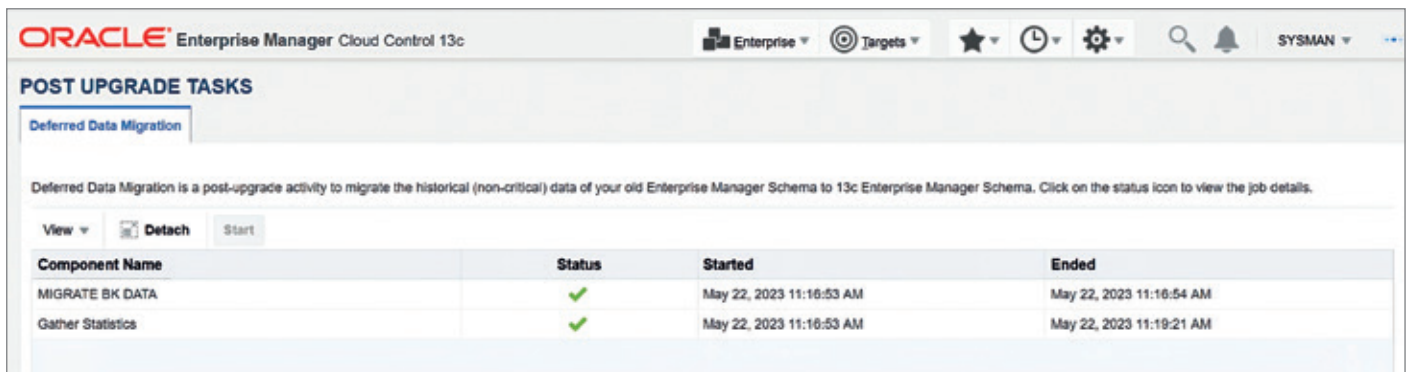


Abbildung 16: Erfolgreiche OEM Post Upgrade Tasks (© Oracle, Screenshot OEM)

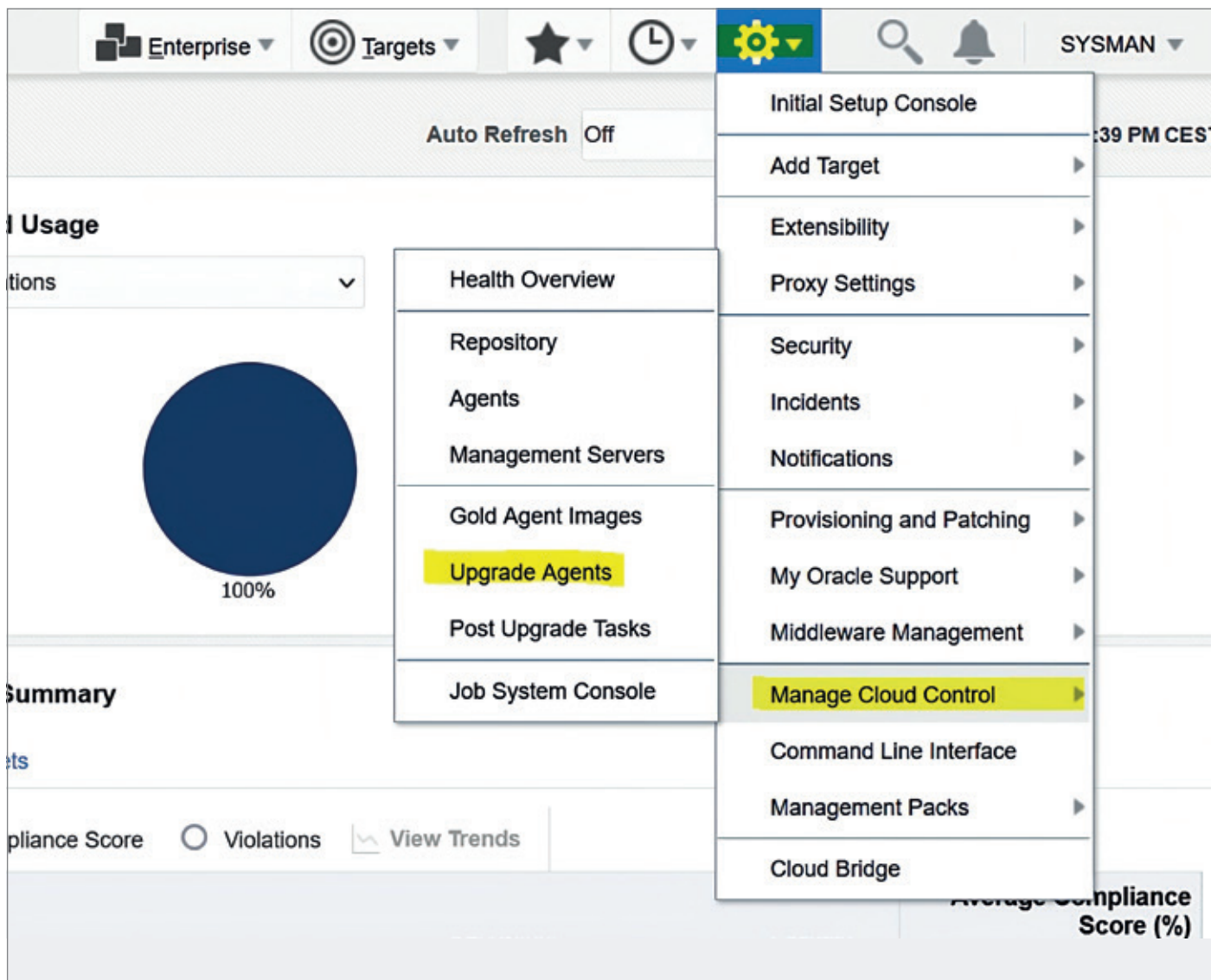


Abbildung 17: Start des Upgrades der Agenten (© Oracle, Screenshot OEM)

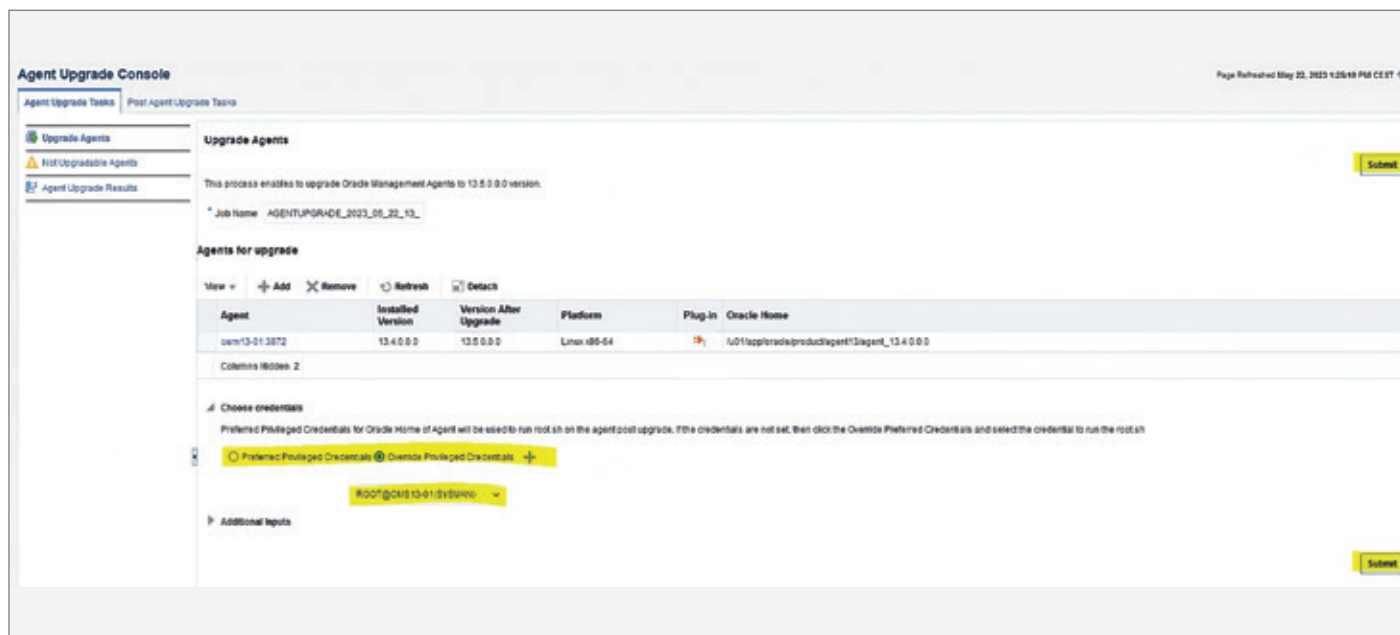


Abbildung 18: Agent Upgrade root Credentials (© Oracle, Screenshot OEM)

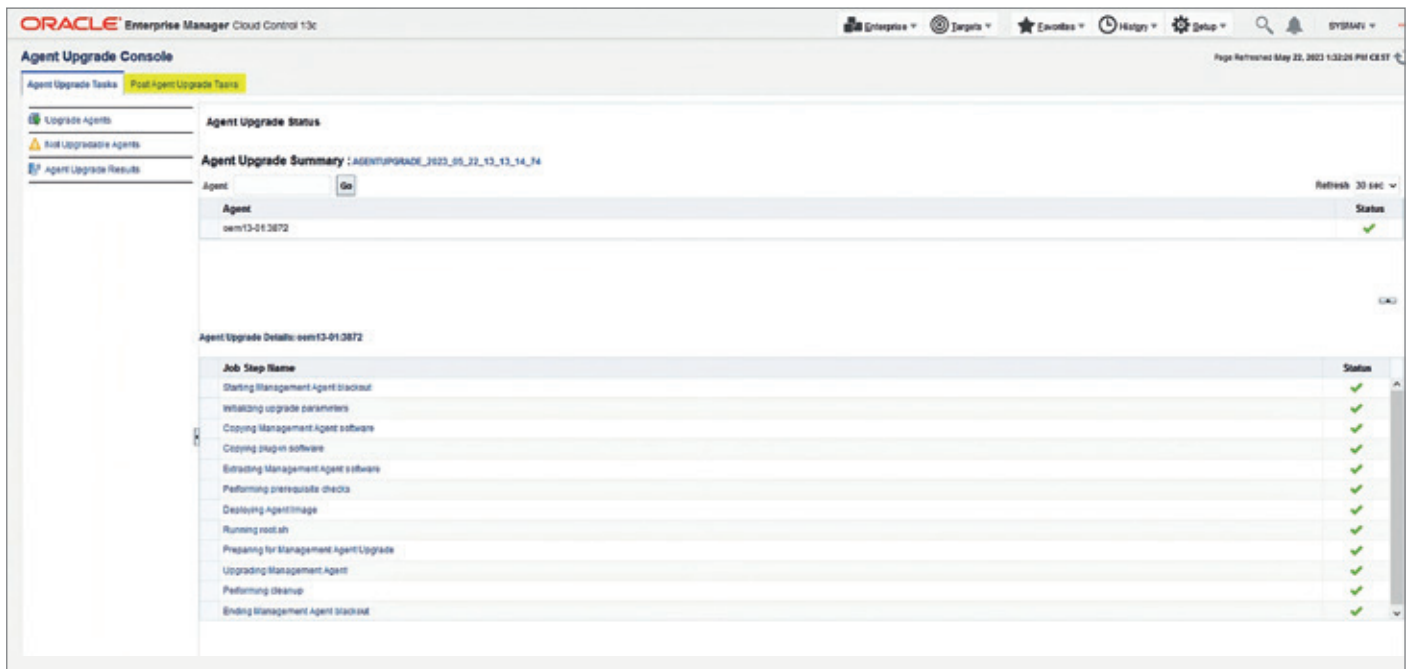


Abbildung 19: Agent Post Upgrade Tasks (© Oracle, Screenshot OEM)

```
cd <Patchverzeichnis>
emctl stop oms -all
opatch apply
opatch lspatches
emctl start oms
```

Listing 2: Patchen des Weblogic-Servers

```
cd <Patchverzeichnis>
omspatcher apply -analyze -bitonly
omspatcher apply -bitonly
omspatcher lspatches
```

Listing 3: Patchen des Weblogic-Servers mit „bitonly-Option“

Beispiel-Upgrade des OEM von 13.4 auf 13.5

Wie anfangs bereits erwähnt, ist für das Upgrade des OEM eine sorgfältige Planung und Vorbereitung erforderlich, um einen reibungslosen Ablauf zu gewährleisten. Die MOS-Note „13.5: Checklist for Upgrading Enterprise Manager Cloud Control from Version 13.3/13.4 to 13.5 (Doc ID 2761728.1)“ enthält eine nahezu vollständige Schritt-für-Schritt-Anleitung für das Upgrade. Aufgrund des vorgegebenen Migrationspfads von Oracle (siehe Abbildung 11) gestaltet sich der Upgrade-Prozess etwas zeitaufwendiger.

In der genannten MOS-Note werden insgesamt 19 Punkte aufgeführt, die als

Vorbereitung für das Upgrade beachtet werden müssen. Dabei sind auch Links zu einer weiteren Dokumentation und anderen MOS-Notes enthalten. Es empfiehlt sich daher, eine eigene Schritt-für-Schritt-Anleitung unter Berücksichtigung aller verfügbaren Quellen zu erstellen und diese systematisch abzuarbeiten.

Der Installationsvorgang selbst kann variieren, abhängig von Faktoren wie zum Beispiel dem Vorhandensein zusätzlicher Plug-ins. Daher ist es wichtig, den Installer entsprechend der individuellen Gegebenheiten zu starten.

- Die heruntergeladenen ZIP-Dateien sollten in dasselbe Verzeichnis entpackt werden.

- Starten der Installation mit dem Befehl:
`./em13500_<platform>.bin`
- Falls im Verzeichnis /tmp nicht ausreichend Speicherplatz vorhanden ist (mindestens 14 GB), kann ein anderer temporärer Speicherort angegeben werden:
`./em13500_<platform>.bin -J-Djava.io.tmpdir=<Pfad zum Verzeichnis>`
- Wenn zusätzliche Plug-ins erforderlich sind, muss der Installer mit der Option `PLUGIN_LOCATION` gestartet werden. Der Parameter benötigt den vollständigen Pfad zu den OPAR- bzw. ZIP-Dateien an:
`./em13500_<platform>.bin PLUGIN_LOCATION=<Kompletter Pfad zu den opar/zip-Dateien>`

Es wird empfohlen, die Schritte bei der Installation genauer zu betrachten, wie in den Abbildungen 12 und 13 dargestellt.

Nachdem die Installation erfolgreich abgeschlossen wurde, sollte der OMSpacher ausgetauscht und das neueste Release Update (RU) für die Version 13.5 eingespielt werden. Listing 3 zeigt den Ablauf des Patch-Vorgangs.

Während des Patch-Vorgangs können viele Warnungen und Fehler auftreten, die aber aufgrund der „bitonly-Option“ ignoriert werden können. Nach erfolgreicher Installation des aktuellen RUs, kann dann das eigentliche Upgrade mit dem Skript „NEUES_OEM_ORACLE_HOME/

sysman/install/ConfigureGC.sh“ durchgeführt werden. Während des Upgrades werden alle notwendigen Informationen abgefragt und nach dem erfolgreichen Upgrade steht der neue OEM wie gewohnt zur Verfügung.

Post Upgrade Tasks des OEM

Nach Abschluss des OEM-Upgrades gibt es noch einige Aufgaben, die durchgeführt werden sollten.

- **OEM Post Upgrade Tasks**
Es handelt sich hierbei um zwei Jobs, die gestartet werden müssen. Weitere Informationen dazu finden Sie in den *Abbildungen 14-16*.
- **Das Upgrade der Agenten** (*siehe Abbildung 17 und 18*)
Dieses sollte ebenfalls in der empfohlenen Reihenfolge erfolgen, zuerst auf dem „central agent“ und anschließend schrittweise auf den restlichen Agenten. Der Wizard unterstützt bei diesem Upgrade-Prozess. Es ist wichtig, die Root-Zugangsdaten anzugeben (*siehe Abbildung 18*). Falls dies nicht geschieht, muss das „root.sh-Skript“ manuell auf jedem Host ausgeführt werden.
- **Agenten Post Upgrade Tasks**
Diese Aufgaben umfassen unter anderem das Entfernen des alten Oracle Home des Agenten. Der entsprechende Wizard ist unter dem Menüpunkt „Setup“ – „Manage Cloud Control“ – „Upgrade Agents“ zu finden (*siehe Abbildung 19*).

Nach Abschluss dieser Arbeiten ist der OEM zusammen mit allen Agenten aktualisiert.

Fazit zum Upgrade des OEM

Zusammenfassend kann festgestellt werden, dass der Upgrade-Pfad, der von Oracle vorgegeben wird, zeitaufwendig ist und eine Vielzahl von Vorabprüfungen erfordert. Es gibt verschiedene MOS-Notes und keine einheitliche Checkliste, zusätzlich muss die Oracle-Dokumentation berücksichtigt werden. Wenn alle Voraussetzungen erfüllt sind, verläuft das Upgrade in der Regel (fast) fehlerfrei.

Aufgrund des hohen Planungs- und Vorbereitungsaufwands könnte eine Neuinstallation des OEM in der Version 13.5 eine schnellere Alternative sein. Es ist jedoch zu beachten, dass dabei historische Metrikdaten verloren gehen. Darüber hinaus können nicht alle Arten von Jobs aus dem alten OEM exportiert und in den neuen OEM importiert werden, wie beispielsweise der Multi-Task-Job. Eine Liste der zu exportierenden Jobs kann mit dem Befehl „emcli get_job_types“ erhalten werden. Bei einer Neuinstallation des OEM müssen auch angelegte Administratoren, Rollen, benannte Anmeldeinformationen (Named Credentials) und Gruppen wieder manuell im neuen OEM erstellt werden. Zusätzlich müssen alle bereits installierten Agenten neu konfiguriert werden. Da dies einen Neustart der Agenten erfordert, kann es nicht über den OEM als Job gesteuert werden. Je nach Anzahl der Agenten kann dies einen erheblichen Aufwand bedeuten.

Persönliche Zusammenfassung

Abschließend möchte ich nochmals betonen, dass das Patchen und ein Upgrade des Oracle Enterprise Managers zwar einen gewissen Aufwand erfordert, aber von großer Bedeutung für die Stabilität und Sicherheit des Systems ist. Es lohnt sich, die Zeit in die Planung und Vorbereitung zu investieren, um von den neuesten Funktionen und Verbesserungen zu profitieren. Die wichtigste Empfehlung ist, die Dokumentation und die MOS-Notes sorgfältig zu lesen. Mit regelmäßigen Patches und Upgrades bleibt der OEM auf dem aktuellen Stand und gewährleistet eine effiziente Verwaltung der Oracle-Infrastruktur.

Quellen

- [1] MOS-Note Overview of the Enterprise Manager Proactive Patch Program (Doc ID 822485.1)
- [2] MOS-Note Obtaining Java SE (JDK/JRE) for Oracle Fusion Middleware Products (Doc ID 1506916.1)
- [3] MOS-Note How to Install and Maintain the Java SE Installed or Used with FMW 11g/12c/14.1.1 Products (Doc ID 1492980.1)
- [4] MOS-Note 13.5: How To Upgrade Enterprise Manager 13.5 Cloud Control OMS-

Patcher Utility to the Latest Version (Doc ID 2809842.1)

- [5] MOS-Note EM 13c: How to Upgrade OPatch to the Latest Version on the Enterprise Manager 13.4 or 13.5 Cloud Control OMS (Doc ID 2728285.1)
- [6] MOS-Note 13.4: How To Apply Release Updates on OMS During Installation/Upgrade (Doc ID 2670039.1)
- [7] MOS-Note 13.5: Checklist for Upgrading Enterprise Manager Cloud Control from Version 13.3/13.4 to 13.5 (Doc ID 2761728.1)
- [8] MOS-Note EM 13c: Checklist for Upgrading Enterprise Manager Cloud Control from Version 12.1.0.5/13.1/13.2 to 13.3 (Doc ID 2418352.1)

Über die Autorin

Birgit Arndt ist seit 2018 bei SVA System Vertrieb Alexander GmbH beschäftigt und seit April dieses Jahres dort Leiterin des Competence Centers für Datenbanken. Sie verfügt über eine 20-jährige Erfahrung als Oracle-Datenbankadministratorin mit umfassendem Fachwissen im Bereich Oracle Enterprise Manager. In dieser Zeit hat sie bei verschiedenen Kunden zahlreich OEM installiert, konfiguriert und verschiedene Upgrade-Szenarien durchgeführt.



Birgit Arndt
Birgit.Arndt@sva.de



MySQL, PostgreSQL und Oracle – ein einfacher Vergleich

Raphael Salguero Aragón, EnterpriseDB

Dieser Artikel stellt einen grundlegenden Vergleich zwischen MySQL, PostgreSQL und Oracle dar. Der Fokus soll darauf liegen, einen ersten Überblick über Gemeinsamkeiten und Unterschiede der drei Technologien zu gewinnen.

MySQL, Oracle und PostgreSQL zählen zu den beliebtesten relationalen Datenbankmanagementsystemen (**RDBMS**). Während Oracle und MySQL sich mittlerweile seit mehreren Jahrzehnten in di-

versen Rankings an der Spitze befinden, läuft speziell PostgreSQL seit einigen Jahren vielen anderen Anbietern den Rang ab und verzeichnet einen steilen Anstieg (*siehe Abbildung 1*).

Es gibt bereits zahlreiche Vergleiche zwischen MySQL, Oracle und PostgreSQL, doch meist ist die Betrachtung nicht vollständig objektiv. Ich möchte mich in diesem Artikel daher weniger auf Zahlen,

Fakten und Bewertungen beziehen, sondern aus reiner Anwender-, beziehungsweise DBA-Sicht einige Aspekte aufzeigen und gegenüberstellen.

Zunächst aber ein paar grundlegende Details: MySQL, Oracle und PostgreSQL lassen sich über die bekannten Wege auf allen gängigen Betriebssystemen (Linux, Solaris und Windows) installieren und betreiben. Sowohl XML-Unterstützung als auch der Support für die klassischen Konnektoren (zum Beispiel JDBC, ODBC, .NET) ist in jedem Fall gegeben. Alle Kandidaten unterstützen außerdem verschiedenste Indizierungen, Partitions- und Replikationsmöglichkeiten, die Verwendung von Triggern und bieten zusätzliche Security- und Enterprise-Funktionalitäten. Wo liegen also die Unterschiede?

Der Platzhirsch

Während viele der Unterschiede zwischen MySQL, Oracle und PostgreSQL im Detail liegen, gibt es einen sehr offensichtlichen Unterschied: Der Platzbedarf.

Während das Datenbank-Home, also das Softwareverzeichnis bei einer Oracle 19c EE (Enterprise Edition) Installation 10 GB in Anspruch nimmt, fällt die Software bei PostgreSQL (Version 113.2, 29 MB) und MySQL (Version 8.0, 182 MB) deutlich schlanker aus.

Das gleiche Verhältnis zieht sich auch bei der initialen Größe einer neuerstellten Datenbank durch (vergleiche Tabelle 1).

Sicherlich hinkt der Vergleich etwas, da Oracle zum Beispiel ein vollständiges Java mitliefert, trotzdem ist die Tendenz deutlich.

Ein Blick auf die Architekturen

Auch wenn sich die Architekturen zwischen Oracle, PostgreSQL und MySQL natürlich in einigen Details unterscheiden, gibt es doch konzeptionelle Gemeinsamkeiten.

Zu allen drei Systemen gehören immer Systemprozesse, die unterschiedliche Verwaltungsaufgaben übernehmen und für einen sauberen Betrieb der Technologien verantwortlich sind.

Während PostgreSQL hier in einer einfachen Konfiguration mit 5-6 Serverprozessen auskommt (siehe Listing 1) und MySQL sogar mit einem einzigen zentralen Daemon-Prozess laufen kann (siehe Listing 2), summieren sich die Oracle-Prozesse schnell auf über 60 laufende Prozesse (siehe Listing 3).

Das DBA „Ein-Mal-Eins“

Werfen wir nun einen kurzen Blick in das „Ein-Mal-Eins“ eines DBAs im Umgang mit den Datenbanken und beginnen dabei mit Oracle:

Bevor wir eine Datenbank starten können, müssen wir für unseren OS-Benutzer entsprechende Umgebungsvariablen setzen. Dafür kann das Tool **oraenv** verwendet werden, welches auf Basis einer **oratab**-Datei die Variablen setzt (siehe Listing 4).

Auf Basis der **ORACLE_SID** und dem **ORACLE_HOME** lassen sich die Binaries und die Ziel-Instanz definieren.

Anschließend lässt sich mit Hilfe von **sqlplus** eine Verbindung zur „heruntergefahrenen Instanz“ aufbauen, welche im

nächsten Schritt gestartet werden kann (siehe Listing 5).

Auf Basis einer Parameterdatei, welche per Default im Verzeichnis \$ORACLE_HOME/dbs gesucht wird, wird die Instanz konfiguriert und gestartet (siehe Listing 6).

In diesem konkreten Fall befindet sich lediglich eine Parameterdatei namens **init.ora** im dbs-Verzeichnis. Typischerweise würde diese Datei nicht verwendet werden, wenn stattdessen ein **PFILE** oder ein **SPFILE** existiert, welche die **SID** der Instanz (identisch zu \$ORACLE_SID) im Namen beinhalten (vergleiche Tabelle 2 für die konkrete Reihenfolge).

Während eine Oracle-Instanz im normalen Umfeld (gilt zum Beispiel nicht für eine **Oracle Database Appliance (ODA)**) empfiehlt sich für PostgreSQL und MySQL zum Starten und Stoppen der Instanz die Verwendung eines Services (siehe Listing 7 und Listing 8).

Durch die Verwendung eines Services, lassen sich ohne großen Aufwand mehrere Instanzen auf einem System verwalten.

Im Vergleich zur Oracle-Variante mit **oraenv**, muss für PostgreSQL und MySQL allerdings im Nachhinein immer sichergestellt werden, dass wir uns beim Arbeiten auch mit dem richtigen Cluster verbinden.

Hierfür können ebenfalls Umgebungsvariablen (zum Beispiel PGPORT, PGDATA, PGHOST für PostgreSQL) gesetzt werden, die einen unkomplizierten Wechsel zwischen verschiedenen Umgebungen ermöglichen.

Dies ist selbstverständlich nur ein kleiner Ausschnitt aller Möglichkeiten, soll aber dennoch einen kleinen Überblick liefern.

RDBMS	Software	Default-Datenbank
Oracle 19.7	10 GB	6,1 GB
MySQL 8.0	182 MB	173 MB
PostgreSQL 13.2	29 MB	43 MB

Tabelle 1: Speicherbedarf RDBMS [2]

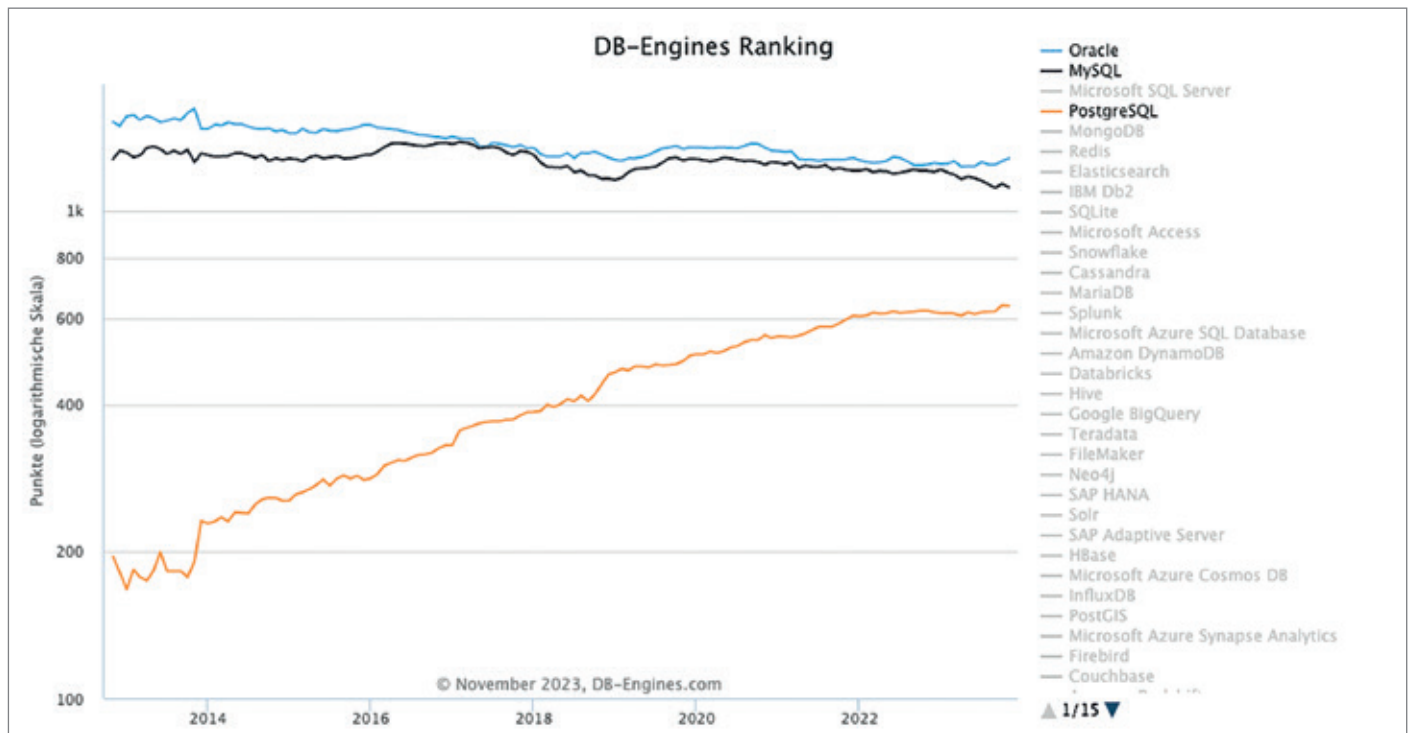


Abbildung 1: DB-Engines Ranking [1]

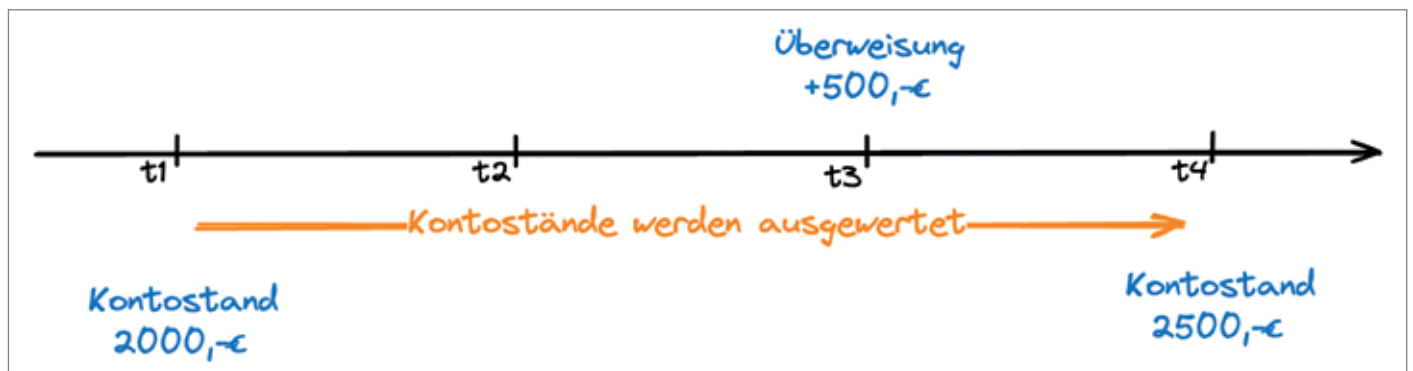


Abbildung 2: parallele Transaktionen, Quelle: ©Raphael Salguero Aragón

```
ip-172-31-2-106:/var/lib/postgresql []$ ps -ef | grep "15/app"
postgres 844574      1  0 15:08 ?        00:00:00 /usr/lib/postgresql/15/bin/postgres -D /var/lib/post-
gresql/15/app -c config_file=/etc/postgresql/15/app/postgresql.conf
postgres 844575 844574  0 15:08 ?        00:00:00 postgres: 15/app: checkpointer
postgres 844576 844574  0 15:08 ?        00:00:00 postgres: 15/app: background writer
postgres 844578 844574  0 15:08 ?        00:00:00 postgres: 15/app: walwriter
postgres 844579 844574  0 15:08 ?        00:00:00 postgres: 15/app: autovacuum launcher
postgres 844580 844574  0 15:08 ?        00:00:00 postgres: 15/app: logical replication launcher
```

Listing 1: Serverprozesse PostgreSQL Cluster „15/app“

```
root@ip-172-31-25-199:~# ps -ef | grep mysql
mysql      19393      1  0 Nov04 ?        02:11:12 /usr/sbin/mysqld
```

Listing 2: Serverprozess MySQL

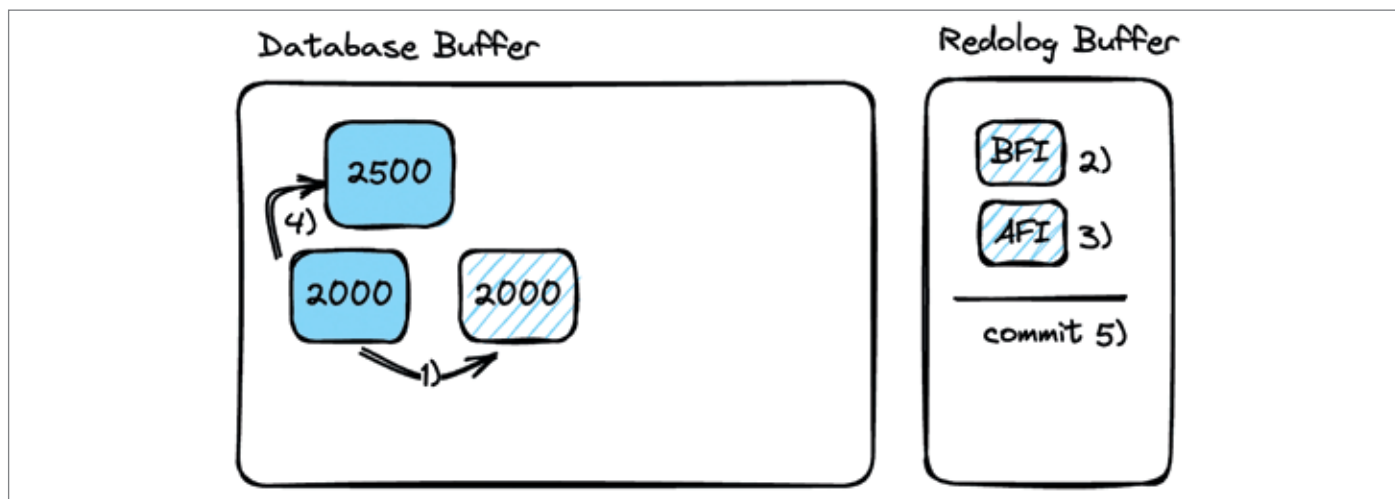


Abbildung 3: MVCC Oracle (vereinfacht), Quelle: ©Raphael Salguero Aragón

```
[oracle@localhost ~]$ ps -ef | grep oracle | wc -l
63
```

Listing 3: Summe der Oracle-Serverprozesse

```
[oracle@localhost ~]$ cat /etc/oratab
XE:/opt/oracle/product/21c/dbhomeXE:N
[oracle@localhost ~]$ . oraenv
ORACLE_SID = [oracle] ? XE
The Oracle base has been set to /opt/oracle
[oracle@localhost ~]$ env | grep ORACLE
ORACLE_SID=XE
ORACLE_BASE=/opt/oracle
ORACLE_HOME=/opt/oracle/product/21c/dbhomeXE
```

Listing 4: Umgebungsvariablen Oracle

```
[oracle@localhost ~]$ sqlplus / as sysdba
SQL*Plus: Release 21.0.0.0.0 - Production on Sun Oct 1 20:30:40 2023
Version 21.3.0.0.0
Copyright (c) 1982, 2021, Oracle. All rights reserved.
Connected to an idle instance.
SQL> startup
ORACLE instance started.
Total System Global Area 1140848688 bytes
Fixed Size          9685040 bytes
Variable Size       687865856 bytes
Database Buffers   436207616 bytes
Redo Buffers        7090176 bytes
Datenbank mounted.
Datenbank geöffnet.
```

Listing 5: Start einer Oracle-Instanz

Ohne zusätzliche Schritte ist nun zwar keine der drei Datenbanken in der Lage, Verbindungen von externen Applikati-

onen entgegenzunehmen – es fehlen unter anderem noch Benutzer, Berechtigungen, Anwendungsdaten und Schnitt-

stellen (zum Beispiel der Oracle Listener) – doch können wir nun mit unseren Administrationsbenutzern die Standard-Cli-

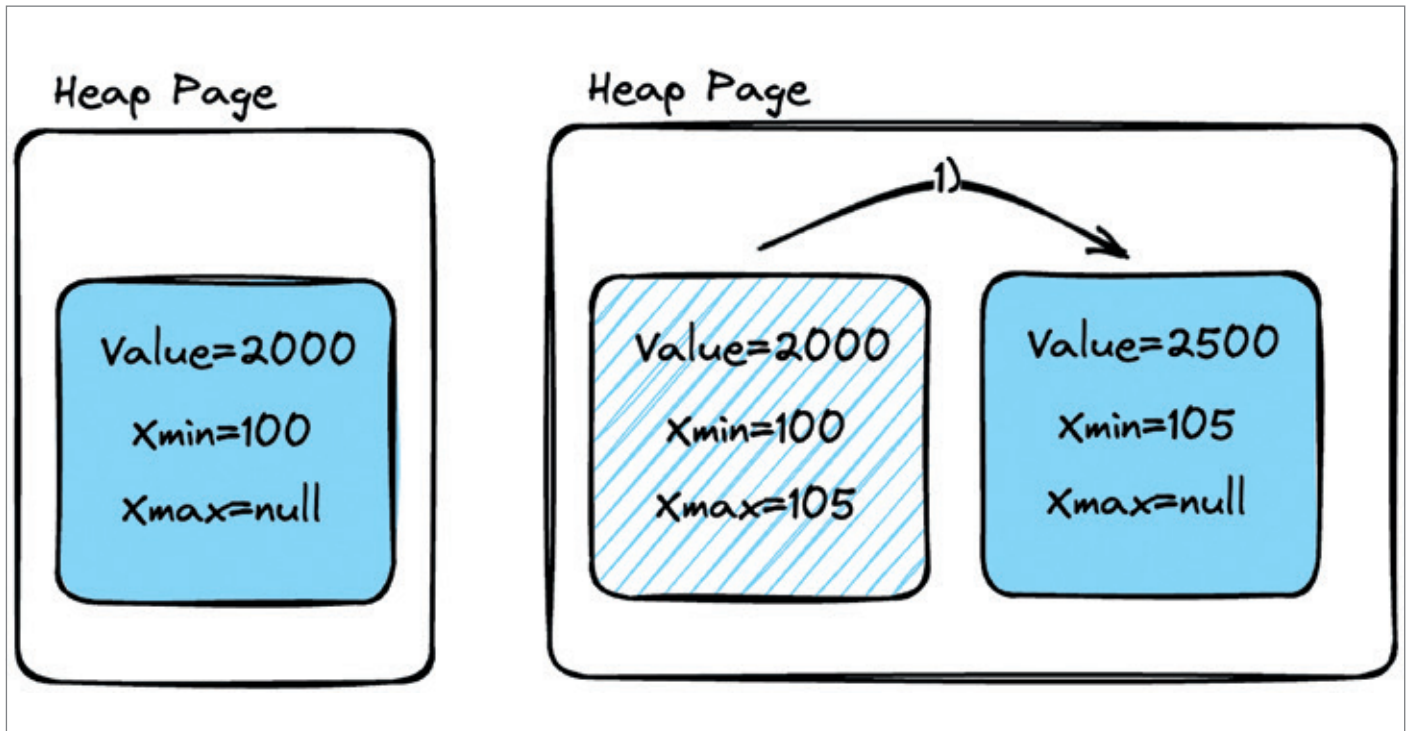


Abbildung 4: MVCC PostgreSQL (vereinfacht), Quelle: ©Raphael Salguero Aragón

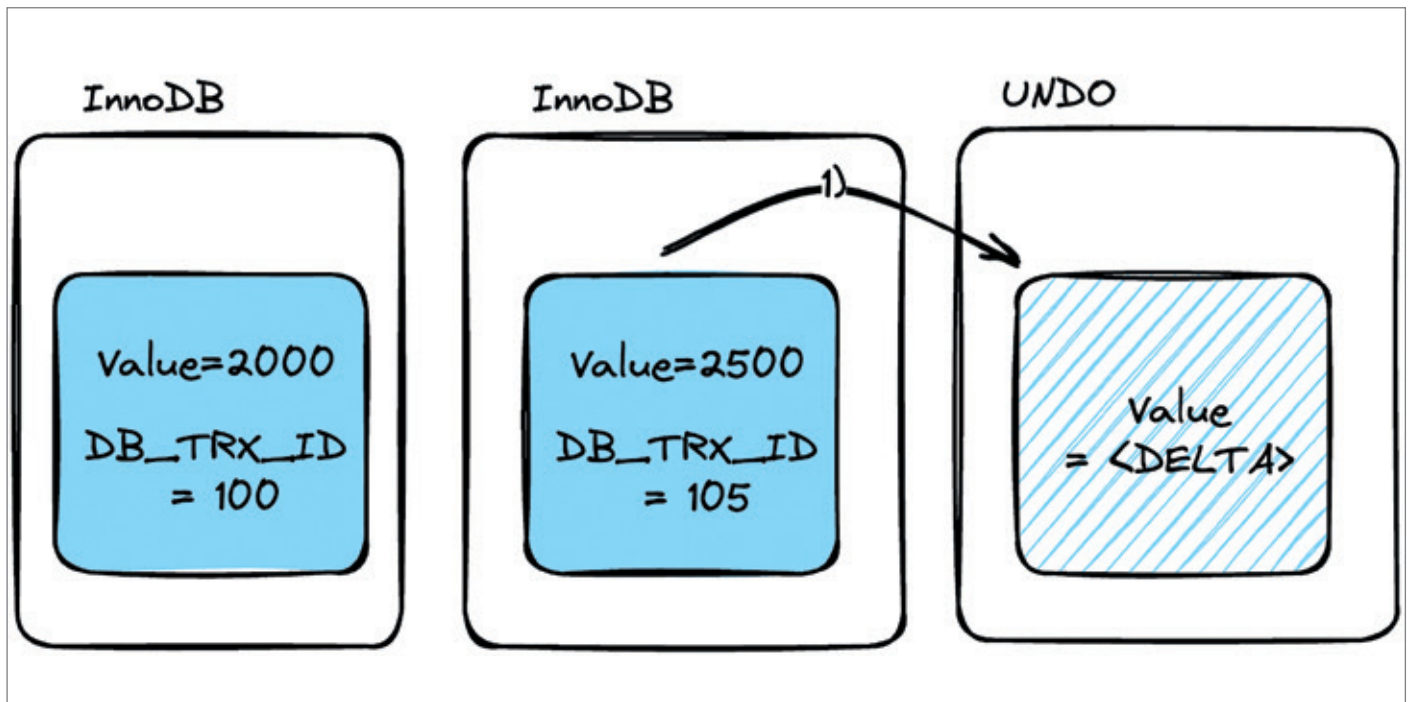


Abbildung 5: MVCC MySQL InnoDB (vereinfacht), Quelle: ©Raphael Salguero Aragón

ents bedienen und die Datenbanken konfigurieren.

Die Konfigurationen

Wir wissen bereits, dass beim Start einer Oracle-Instanz eine Parameter-Datei gesucht wird.

Bei Oracle wird hier zwischen einem **PFILE** und einem **SPFILE** unterschieden. Während das **PFILE** lesbar und editierbar ist, versteckt sich hinter dem **SPFILE** eine binäre Datei. Der Unterschied liegt darin, dass eine Datenbankinstanz, welche mit einem **SPFILE** gestartet wird, Parameteranpassungen erlaubt, die einen Neustart überle-

ben und somit persistent sind. Im Vergleich hierzu werden Anpassungen an einem **PFILE** nur durch einen Neustart der Instanz wirksam.

MySQL und PostgreSQL verwenden ein ähnliches Prinzip.

Es gibt jeweils eine lesbare Konfigurationsdatei (**postgresql.conf** und **my.conf**) sowie eine dynamische Kon-

Priorität / Reihenfolge	Datei
1.	spfileSID.ora
2.	spfile.ora
3.	initSID.ora
4.	init.ora

Tabelle 2: Reihenfolge PFILE/SPFILE beim startup

figurationsdatei, welche mit dem **SPFILE** von Oracle vergleichbar sind und Konfigurationsänderungen konsistent machen (**postgresql.auto.conf** und **mysqld-auto.cnf**).

Durch die Einführung der dynamischen Konfigurationsdateien wurde mehr Flexibilität beim Konfigurieren der Datenbanken geschaffen. DBAs können nun für jede Änderung entscheiden, ob sie nur für die gerade laufende Instanz gelten (zum Beispiel SCOPE = MEMORY bei Oracle), ob sie sowohl zur Laufzeit als auch persistent, oder erst bei einem Neustart Gültigkeit gewinnen sollen.

Wie war das noch gleich mit dem ACID-Prinzip?

Essenziell für den Betrieb von relationalen Datenbankmanagementsystemen ist seit vielen Jahren die Einhaltung des **ACID**-Prinzips. Dies bedeutet konkret, dass eine Transaktion innerhalb der Datenbank atomar ausgeführt wird (**Atomicity**), ein konsistenter Datenbestand sichergestellt werden kann (**Consistency**), mehrere Transaktionen parallel laufen können (**Isolation**) und diese Transaktionen auch dauerhaft Bestand haben (**Durability**). Eine Gehaltszahlung soll schließlich nicht mit der Zeit auf dem Konto „verblassen“, ohne dass ich das Geld eigenständig für Datenbanklizenzen ausgegeben habe.

Um die unterschiedlichen Herangehensweisen deutlich zu machen, werden wir die internen Abläufe etwas verein-

facht darstellen und anhand des folgenden Szenarios erklären:

Ihr Bankberater startet zum Zeitpunkt t1 eine Auswertung der Kontostände all seiner Kunden. Sie bekommen in der Zwischenzeit einen Betrag von 500,-€ überwiesen (Zeitpunkt t3). Die Auswertung Ihres Bankberaters endet einige Minuten später zum Zeitpunkt t4. Wird die Überweisung bei der Auswertung Ihres Bankberaters berücksichtigt? (siehe *Abbildung 2*) Die Antwort vorweggenommen: nein.

Das **ACID**-Prinzip muss an dieser Stelle sicherstellen, dass die gestartete Transaktion des Bankberaters zum Zeitpunkt t1 isoliert stattfindet und die verwendeten Daten zu Beginn und zum Ende der Transaktion konsistent sind.

Ohne einen zusätzlichen Mechanismus, der diese Konsistenz und Isolation sicherstellt, würde nur die Option einer Sperre der Kontostände übrigbleiben. Da eine solche Sperre allerdings Auswirkungen auf weitere Transaktionen hätte und diese verzögern würde, haben unsere Kandidaten ein **MVCC**-Modell (**Multiversion Concurrency Control**) implementiert.

MVCC Oracle

Innerhalb einer Oracle-Datenbank stellt eine Kombination aus dem **Database Buffer** und dem **Redolog Buffer** sicher, dass Transaktionen parallel laufen und stets mit einem konsistenten Datenbestand arbeiten können.

Die Auswertung des Bankberaters sorgt initial dafür, dass Ihr persönlicher Kontostand in den **Database Buffer** gelesen wird. Bevor nun eine Überweisung stattfindet und Ihr Kontostand angehoben wird, erstellt Oracle eine identische Kopie des Datensatzes und schreibt sogenannte **Before Images** und **After Images** in den **Redolog Buffer**. Diese Informationen stellen sicher, dass die durchgeführte Transaktion im Fehlerfall wieder zurückgesetzt werden kann.

Erst nach Erstellung dieser Images führt Oracle die Anpassung des Kontostandes durch (siehe *Abbildung 3, Schritt 4*). Ist die Anpassung abgeschlossen, sorgt ein anschließendes „commit“ dafür, dass der neue Datensatz festgeschrieben wird.

Die Auswertung Ihres Beraters arbeitet weiterhin mit dem initialen Datensatz und wird durch die Überweisung daher nicht beeinträchtigt.

Erst eine Auswertung, die nach dem „commit“ ausgeführt werden würde, arbeitet auf Basis Ihres neuen Kontostandes.

PostgreSQL setzt das **ACID**-Prinzip im Vergleich zu Oracle mit der Erstellung zusätzlicher Tupel um. Hierzu pflegt PostgreSQL für jeden Datensatz einen zusätzlichen **Xmin**- und **Xmax**-Wert.

In unserem konkreten Beispiel wird der Kontostand als Datensatz in einer **Heap Page** gespeichert. Der **Xmin**-Wert definiert den Zeitpunkt der letzten ändernden Transaktion für diesen Datensatz (erstmaliges Insert oder Update). Die Überweisung führt nun dazu, dass der Datensatz kopiert und der **Xmax**-Wert des alten Datensatzes gesetzt wird. Das neu erstellte Tupel spiegelt den neuen Kontostand inklusive Überweisung wider und erhält ebenfalls einen erstmaligen **Xmin**-Wert. Der **Xmax**-Wert beträgt NULL (siehe *Abbildung 4*).

Die Kombination aus **Xmin** und **Xmax** spiegelt nun die Verfügbarkeit der Datensätze wider. Eine Session, wie die Auswertung Ihres Bankberaters, kann nur auf das Tupel zugreifen, dessen Transaktions-ID sich in dem Bereich zwischen **Xmin** und **Xmax** befindet. In unserem konkreten Fall greift der Bankberater also weiterhin auf den alten Kontostrand zu, während alle zukünftigen Sessions auf das neue Tupel zugreifen werden.


```
[oracle@localhost dbs]$ ls -al $ORACLE_HOME/dbs
insgesamt 8
drwxr-xr-x. 2 oracle oinstall  22 27. Sep 09:46 .
drwxrwxr-x. 61 oracle oinstall 4096  1. Okt 20:32 ..
-rw-r-----. 1 oracle oinstall 3079 14. Mai 2015  init.ora
```

Listing 6: Inhalt \$ORACLE_HOME/dbs

```
ip-172-31-2-106:/var/lib/postgresql []$ sudo systemctl start postgresql@15-app
ip-172-31-2-106:/var/lib/postgresql []$ sudo systemctl status postgresql@15-app
• postgresql@15-app.service - PostgreSQL Cluster 15-app
   Loaded: loaded (/lib/systemd/system/postgresql@.service; enabled-runtime; vendor preset: enabled)
   Active: active (running) since Sun 2023-11-19 15:08:13 UTC; 25min ago
   Process: 844569 ExecStart=/usr/bin/pg_ctlcluster --skip-systemctl-redirect 15-app start
code=exited, status=0/SUCCESS
   Main PID: 844574 (postgres)
     Tasks: 6 (limit: 9384)
    Memory: 20.2M
       CPU: 762ms
    CGroup: /system.slice/system-postgresql.slice/postgresql@15-app.service
           └─844574 /usr/lib/postgresql/15/bin/postgres -D /var/lib/postgresql/15/app -c
config_file=/etc/postgresql/15/app/postgresql.conf
...
...

Nov 19 15:08:11 ip-172-31-2-106 systemd[1]: Starting PostgreSQL Cluster 15-app...
Nov 19 15:08:13 ip-172-31-2-106 systemd[1]: Started PostgreSQL Cluster 15-app.
```

Listing 7: PostgreSQL-Clusterstart

```
root@ip-172-31-25-199:~# systemctl start mysql
root@ip-172-31-25-199:~# systemctl status mysql
• mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-11-04 21:36:48 UTC; 2 weeks 0 days ago
   Main PID: 19393 (mysqld)
   Status: "Server is operational"
     Tasks: 42 (limit: 4598)
    Memory: 403.5M
       CPU: 2h 11min 20.826s
    CGroup: /system.slice/mysql.service
           └─19393 /usr/sbin/mysqld

Nov 04 21:36:47 ip-172-31-25-199 systemd[1]: Starting MySQL Community Server...
Nov 04 21:36:48 ip-172-31-25-199 systemd[1]: Started MySQL Community Server.
```

Listing 8: MySQL-Instanzstart

Achtung: Wie die *Abbildung 4* schon skizziert, führt dieses Verfahren zu einem „Aufpusten“ der **Heap Pages** und somit zu einer Vergrößerung der Tabelle.

Innerhalb von PostgreSQL sorgen erst **Vacuum**-Prozesse wieder dafür, dass der Speicherplatz von nicht mehr benötigten Tupels freigegeben wird.

MVCC MySQL

Der Ansatz von MySQL mit der **InnoDB Engine** ist in etwa vergleichbar mit dem Ansatz von PostgreSQL.

Der ursprüngliche Datensatz wird mit einer Transaktions-ID **DB_TRX_ID** abgespeichert und bekommt dadurch seinen individuellen Gültigkeitsbereich.

Die Überweisung führt an dieser Stelle ebenfalls zu einem weiteren Datensatz im **UNDO** des MySQL-Clusters. Im Gegensatz zu PostgreSQL bleibt in der ursprünglichen Tabelle allerdings nur der aktuelle Datensatz vorhanden. Ein Pointer verweist hier auf den vorherigen Datensatz im **UNDO**. Ein weiterer Unterschied besteht darin, dass im **UNDO** nicht der gesamte Datensatz, sondern lediglich ein Änderungsvektor gespeichert wird. Dadurch wird deutlich weniger Speicherplatz allokiert und die eigentliche Tabelle schlank gehalten.

In unserem Beispiel wird die Auswertung des Bankberaters auf Grund einer älteren Transaktions-ID über den neu gesetzten Pointer auf den alten Konto-stand verwiesen.

Fazit

Oracle, PostgreSQL und MySQL sind jeweils souveräne und verlässliche RDBMS, die sich in einigen Punkten ziemlich ähnlich sind und doch – wie ich bereits anklingen lassen konnte – einige Unterschiede aufweisen (*siehe ACID-Prinzip*). Auch wenn es viele Artikel und Vorträge versuchen, es lässt sich doch schwer „allgemein“ beantworten, welches System für welchen Use-Case das richtige ist, ohne wirklich alle Aspekte in der Tiefe zu beleuchten.

Ich für meinen Teil kann nur bekräftigen, dass alle drei Systeme ihren Reiz haben und es Spaß macht, mit ihnen zu

arbeiten. Die Konkurrenz untereinander stellt sicher, dass wir auch künftig sicherlich viele Funktionalitäten auf jeder Seite wiederfinden können.

In der Zukunft werde ich den Vergleich detaillieren und einzelne Teilaspekte in der Tiefe betrachten, um „irgendwann“ ein Gesamtbild teilen und diskutieren zu können.

Bis dahin wünsche ich allen Lesenden weiterhin viel Spaß mit der Arbeit an den bisher favorisierten Systemen.

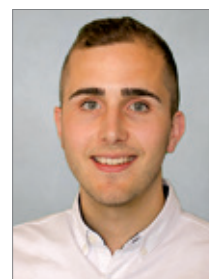
Quellen

- [1] <https://db-engines.com/de/ranking>
- [2] <https://www.informatik-aktuell.de/betrieb/datenbanken/beyond-postgresql-einblicke-in-die-erweiterbarkeit-von-postgresql.html>

Über den Autor

Raphael Salguero Aragón hat sein duales Studium 2017 absolviert und arbeitet mittlerweile als „Senior Migrations Consultant“ bei der Firma EnterpriseDB und ist dort überwiegend in Migrationsprojekten involviert.

Neben PostgreSQL gehören auch seit Jahren Oracle und MySQL zu seinen Kerngebieten. Seit 2018 teilt Raphael seine Erfahrungen aus der Praxis regelmäßig in Form von Präsentationen auf diversen Veranstaltungen.



Raphael Salguero Aragón
Raphael.salguero@enterprisedb.com



Hardware-Ressourcen auf der Oracle Database Appliance verwalten

Sven Illert, Robotron Datenbank-Software

Die kleine Schwester der Exadata, die Oracle Database Appliance (ODA), wird von vielen als Konsolidierungsmaschine genutzt und ersetzt das Öffnen eines größeren Zoo von virtuellen oder physischen Maschinen. Dabei gilt es, die Ressourcen der ODA nach verschiedenen Kriterien sinnvoll an die verschiedenen Datenbanken und virtuellen Maschinen zu verteilen. Werfen wir einen Blick darauf, wie dies mit den zur Verfügung stehenden Bordmitteln erreicht werden kann.

Spricht man bei der ODA im Speziellen und bei Datenbanksystemen im Allgemeinen von Hardware-Ressourcen mit direktem Einfluss auf die Leistungsfähigkeit einer Datenbank, so geht es in der Regel um Hauptspeicher (RAM), Prozessor (CPU) und persistenten Speicher (Storage). Da letzteres in der Database Appliance prinzipiell sehr gut durch ASM und/oder ACFS Autoresizing gemanagt wird, soll dies hier nicht weiter beschrieben werden. Das Gleiche trifft auf andere Ressourcenarten wie die Netzwerkanbindung zu, was man in einem eigenen ausführlichen Artikel darstellen könnte.

Einführung

Die Administration der ODA wird üblicherweise über das sogenannte Browser User Interface (BUI) oder das Command Line Interface (CLI) durchgeführt. Zur Veranschaulichung werden in diesem Fall die CLI in Form des Befehls `odacli` und weitere Linux-Befehle verwendet. Konzentriert wird sich hier außerdem auf den Bereich der Zuteilung von CPU- und RAM-Ressourcen für Oracle-Datenbanken und virtuelle Maschinen, die mittels der Appliance-Software verwaltet werden.

Grundlegend gibt es bei der ODA die Möglichkeit mittels Capacity On Demand nur die CPU-Ressourcen physisch zur Verfügung zu stellen, die auch lizenziert wurden. Dies gilt vor allem dann, wenn Enterprise-Edition-Datenbanken ohne Virtualisierung auf der ODA als Bare-Metal-Datenbanken ausgeführt werden sollen. In diesem Fall kann man grundsätzlich die Anzahl der freigeschalteten CPU-Cores in Zwischenschritten, wie in *Listing 1* dargestellt, anpassen. Bei einer HA-ODA mit zwei Cluster-Knoten gilt die An-

gabe der Cores für jeweils beide Knoten (2 Cores entsprechen 4 Cores im Cluster).

Die Änderung der CPU-Cores erfordert einen Neustart der Hosts und kann nur erfolgen, wenn keine Open Problems im ILOM existieren, weshalb es ratsam ist, dies vorher im ILOM zu überprüfen.

Datenbank-Shapes

Neben der CLI-Parameterstruktur und der Arbeitsweise mittels JSON-Nachrich-

```
[odaadmin@odax5ha0 ~]$ odacli modify-cpucore --cores 2
```

Listing 1: Anpassung der CPU-Cores auf 2 pro Cluster-Knoten

```
[oracle@odax5ha0 ~]$ odacli describe-database --name DOAG01 |
grep Shape
                Shape: odb1
[oracle@odax5ha0 ~]$ odacli modify-database --dbshape odb2 --
name DOAG01
```

Listing 2: Darstellung und Änderung der Shape-Größe der Datenbank DOAG01

```
ALTER SYSTEM SET resource_manager_plan='DEFAULT_CDB_PLAN';
```

Listing 3: Einstellung des Resource-Manager-Plans auf einen Standardplan für Container-Datenbanken

ten fällt auch die Verwaltung der Datenbank-Ressourcen mittels sogenannter Shapes als Nähe zur Oracle Cloud Infrastructure auf. Diese Shapes geben vor, wie das Sizing einer Datenbank in Bezug auf CPU-Count, Speicherverteilung und Redo-Größe initial vorgenommen wird. Auch eine nachträgliche Änderung auf eine andere Shape-Größe lässt sich damit erwirken (siehe Listing 2). Allerdings sollte man dabei darauf achten, dass man keinen Shape größer der Anzahl der lizenzierten CPU-Cores angibt, dies wird sonst mit einer Fehlermeldung quittiert.

Die im Namen des Shapes enthaltene Ziffer gibt die Anzahl der CPU-Cores an, welche die Datenbank nutzen soll. Doch damit dies auch Wirkung zeigt, muss noch ein weiterer Schritt unternommen werden. Denn in der Standardkonfiguration bewirkt das Setzen des Initialisierungsparameters `cpu_count` – wie das bei der Änderung des Shapes stattfindet – noch gar keine Limitierung in der Nutzung der Ressourcen. Dafür benötigt es nämlich noch die Konfiguration eines Resource-Manager-Plans, der auch effektiv die verursachte Auslastung der Datenbankprozesse auf die Anzahl der logischen CPUs entsprechend des `cpu_count`-Parameters begrenzt. Zu beachten ist, dass der Parameter nicht die Anzahl der Cores beziffert, sondern die Anzahl der genutzten logischen CPUs. Bei Intel x86/x64 entspricht dies den Threads, sofern Hyperthreading aktiviert ist. Gesetzt werden kann ein solcher Plan mit

dem folgenden SQL-Statement (siehe Listing 3).

Die Nutzung dieser Ressourcenbeschränkung, die auch als Instance-Caging bekannt ist, führt dazu, dass die Datenbank sich selbst auf eine Auslastung äquivalent dieser logischen CPU-Anzahl drosselt. Dies geschieht auch, wenn mehr CPU-Ressourcen zur Verfügung stünden.

CPU-Pools

Die Eingrenzung der Datenbanken nach dem vorgenannten Schema bietet sich dann an, wenn man keine Überprovisionierung betreiben möchte und ganz klare Lastvorgaben und -abgrenzung hat. Will man jedoch verschiedene Datenbank-Klassen miteinander um eine bestimmte Anzahl von Cores konkurrieren lassen, die kleiner als die Gesamtzahl der insgesamt zur Verfügung stehenden Cores ist, so reicht dafür die Konfiguration mittels Shapes nicht aus. Die Zusage einer festen Anzahl von Cores für kritische Produktionsdatenbanken und die Abschottung dieser CPUs zur Nutzung durch Hilfs- oder Testdatenbanken kann mittels CPU-Pools erreicht werden. Damit können Bereiche von CPU-Cores für einzelne Datenbanken oder Datenbankklassen reserviert und exklusiv zur Verfügung gestellt werden. Es ist dann möglich, dass mehr Last innerhalb dieses Pools erzeugt wird, als Cores zur Verfügung stehen. Doch diese

Auslastung wird andere Cores und CPU-Pools nicht beeinträchtigen.

Es gibt verschiedene Arten von Pools zu unterscheiden. Es gibt jene für Bare-Metal-Datenbanken, die nicht virtualisiert laufen und es gibt solche, die für DB-Systeme und virtuelle Maschinen genutzt werden können. Der Unterschied zwischen ersterem und den beiden letztgenannten ist nicht nur in der Art der Verwendung zu finden, sondern auch in der technischen Implementierung. Während bei den virtuellen Varianten die CPU-Cores im Hypervisor gepinnt und diskret verteilt werden, handelt es sich bei den Bare-Metal-Pools um sogenannte Control Groups, die es seit 2008 im Linux-Kernel gibt. Wichtig zu wissen ist, dass Bare-Metal-CPU-Pools keine Möglichkeit sind, eine lizenzkonforme Core-Reduzierung zu bewirken. Dafür ist es entweder notwendig, ein DB-System zu verwenden oder, die physikalisch verfügbaren Cores mittels `odacli modify-cpucore` zu beschränken.

Um Datenbanken in einem Bare-Metal-CPU-Pool zu organisieren, muss dieser zuerst angelegt werden. Danach kann eine Datenbank-Konfiguration wie in Listing 4 so angepasst werden, dass diese den CPU-Pool nutzt. Eine solche Änderung zieht jedoch einen Neustart der Datenbank nach sich, sodass man für sie eine Downtime einplanen sollte.

Doch was passiert eigentlich in der Datenbank und dem Betriebssystem, wenn man eine Datenbank einem Bare-Metal-CPU-Pool zuweist? In der Datenbank wird dabei das SQL-Statement aus Listing 5 ausgeführt, um den Parameter `processor_group_name` zu setzen. Aber woher wissen Datenbank und Betriebssystem, wie groß dieser CPU-Pool sein soll? Mit dem Befehl aus Listing 4 wird eine entsprechende Kontrollstruktur im `/sys-Pseudodateisystem` angelegt, um die Control Group zu konfigurieren und die entsprechende Anzahl an Cores zuzuweisen. Diese Struktur wird in Listing 6 untersucht.

Wer mit Linux Control Groups auch außerhalb der ODA arbeitet, weiß, dass die Änderungen unter `/sys` nicht persistent sind und man die Konfiguration in `/etc/cg-config.conf` hinterlegen muss. Das ist auf der ODA nicht notwendig, hier kümmert sich die Appliance-Software darum.

```
[odaadmin@odax5ha0 ~]$ odacli create-cpupool -c 2 -n doag01 -bm
[oracle@odax5ha0 ~]$ odacli modify-database --dbname DOAG01 --
cpupool doag01
```

Listing 4: Erstellung eines CPU-Pools und dessen Zuweisung zu einer Datenbank

```
ALTER SYSTEM SET processor_group_name='doag01' SCOPE=SPFILE SID='*';
```

Listing 5: Setzen des Parameters `processor_group_name` bei der Zuweisung eines CPU-Pools

```
[root@odax5ha0 ~]# find /sys/fs/cgroup/cpuset/* -type d
/sys/fs/cgroup/cpuset/doag01
[root@odax5ha0 ~]# cd /sys/fs/cgroup/cpuset/doag01
[root@odax5ha0 cpuset]# cat cpuset.cpus
0-1
```

Listing 6: Die Struktur einer Linux Control Group für Bare-Metal-CPU-Pools

```
[odaadmin@odax5ha0 ~]$ odacli create-dbsystem -p $PWD/dsdoag01.json
[odaadmin@odax5ha0 ~]$ grep odb dsdoag01.json
    "shape": "odb2",
[odaadmin@odax5ha0 ~]$ odacli list-cpupools
```

Name	Type	Configured on	Cores	Associated resources
Created	Updated			
-----	-----	-----	-----	-----
doag01	BM	odax5ha0, odax5ha1	4	DOAG01
2023-05-16 11:48:54 CEST	2023-05-16 22:40:30 CEST			
cfcd4af884	DB_SYSTEM	odax5ha0, odax5ha1	2	dsdoag01
2023-05-17 22:30:05 CEST	2023-05-17 23:47:0			

```
[odaadmin@odax5ha0 ~]$ odacli create-cpupool --dbsystem -c 8 -n doag02
[odaadmin@odax5ha0 ~]$ odacli modify-dbsystem -n dsdoag01 -cp doag02
[odaadmin@odax5ha0 ~]$ odacli list-cpupools
```

Name	Type	Configured on	Cores	Associated resources
Created	Updated			
-----	-----	-----	-----	-----
doag01	BM	odax5ha0, odax5ha1	4	DOAG01
2023-05-16 11:48:54 CEST	2023-05-16 22:40:30 CEST			
doag02	DB_SYSTEM_SHARED	odax5ha0, odax5ha1	8	dsdoag01
2023-05-22 20:54:08 CEST	2023-05-22 20:54:08 CEST			

Listing 7: Die Verwaltung von virtuellen CPU-Pools und die Zuweisung zu einem virtualisierten DB-System

```
[odaadmin@odax5ha0 ~]$ odacli create-cpupool -c 2 -n doag01 -dbs
[odaadmin@odax5ha0 ~]$ odacli describe-job -i e13aa1f3-7bb8-44ad-bc4a-6a09a826bfed
```

Job details

```
-----
ID: e13aa1f3-7bb8-44ad-bc4a-6a09a826bfed
Description: CPU Pool doag01 creation
Status: Failure
Created: May 15, 2023 11:29:56 PM CEST
Message: DCS-10045:Validation error encountered: Not enough physical CPUs available for
CPU Pool 'doag01' on node 'odax5ha0', requested 4 CPUs.
Available CPUs on node 'odax5ha0': 0 (Socket 0: 0, Socket 1: 0).
Available CPUs on node 'odax5ha1': 0 (Socket 0: 0, Socket 1: 0).
```

Task Name	Status	Node Name	Start Time	End
Validate CPU Pool doesn't exist	Success	odax5ha0	May 15, 2023 11:29:56 PM CEST	May 15, 2023 11:29:56 PM CEST
Validate NUMA state	Success	odax5ha0	May 15, 2023 11:29:56 PM CEST	May 15, 2023 11:29:56 PM CEST
Create metadata	InternalError	odax5ha0	May 15, 2023 11:29:56 PM CEST	May 15, 2023 11:29:56 PM CEST

Listing 8: Bei der Erstellung eines virtuellen CPU-Pools muss darauf geachtet werden, dass die reservierten CPU-Cores nach Erstellung des Pools noch frei sind

```
[odaadmin@odax5ha0 ~]$ odacli update-osconfigurations -nhp 81920
```

Listing 9: Anpassung der Hugepages an eine neue Größe

Virtuelle CPU-Pools

Bei den virtuellen CPU-Pools muss man grundsätzlich zwei verschiedene Arten unterscheiden. Da wären jene für die DB-Systeme und jene für virtuelle Maschinen. Erstere werden, sofern nicht anders gewünscht, automatisch angelegt, so bald eine solche Ressource erstellt wird. Dieser CPU-Pool ist dann exklusiv für diese Datenbank vorgesehen. Wird jedoch eine virtuelle Maschine erzeugt, so wird diese nicht automatisch einem CPU-Pool zugewiesen, sondern nutzt einfach alle Bare-Metal-CPU's für ihre logischen CPU's, die nicht reserviert sind. In beiden Fällen ist es jedoch möglich, selbst einen entsprechenden Pool anzulegen und gezielt einzelnen oder mehreren virtuellen Ressourcen zuzuweisen. Zudem sind beide Varianten für ein lizenzkonformes CPU-Pinning nutzbar.

In *Listing 7* wird zuerst ein DB-System mit implizit erstelltem CPU-Pool angelegt und dann ein eigens Erzeugter zugewiesen. Es ist auch darauf zu achten (*siehe Listing 8*), dass jeweils der erste Core einer CPU für die Nutzung von CPU-Pools ausgeschlossen und für Bare-Metal-Nutzung reserviert ist. Dies könnte man mittels des Schalters `--use-reserved-cores` umgehen, was jedoch nicht empfohlen wird. Unter anderem benötigt die Grid Infrastructure gewisse zugesicherte CPU-Ressourcen, damit keine Node-Eviction durch Timeouts bei hoher Last auftritt, was diese Reservierung notwendig macht.

Die Verwaltung von CPU-Ressourcen für virtuelle Maschine gestaltet sich analog, nur dass hier für den Typ des CPU-Pools entsprechend der Schalter `--vm` bei der Erstellung angegeben werden muss. Zudem ist zu beachten, dass bei `modify-vm` die Änderungen mit `--change` persistiert werden können und nur gleichzeitig aktiviert werden, wenn auch `--live` angegeben wird.

Hauptspeicher

Mit den vorangegangenen Methoden kann man die Prozessorleistung gezielt bestimmten Datenbanken und virtuellen Systemen zuweisen und ein ausgefeiltes Last- und Lizenzmanagement betreiben. Beim Hauptspeicher sind die Möglichkeiten dazu etwas begrenzt. Da jede DB ih-

ren Cache auch bei weniger temporärer Auslastung behalten sollte, gibt es hier auch nicht so viele Tuning-Möglichkeiten. Die Größe des Speicherbereichs, der einer Datenbank zugewiesen wird, ergibt sich aus der Wahl des Datenbank Shapes und kann danach auch noch manuell über die Initialisierungsparameter angepasst werden.

Bei der ODA werden grundsätzlich nur Hugepages verwendet und hier liegt auch der einzige Fallstrick begraben, der bei einer Standardinstallation auftreten kann. Sollte man nämlich manuell oder per Shape-Anpassung den Speicherbereich einer Datenbank vergrößern wollen, kann es vorkommen, dass nicht ausreichend Hugepages im System reserviert wurden. Das äußert sich dann zum Beispiel über die Fehlermeldung *SGA: Realm creation failed im Alert-Log*. Standardmäßig konfiguriert Oracle die ODA nämlich mit ca. 50% des Hauptspeichers für die Verwendung mit den großen Speicherseiten. Doch laut Performance Tuning Guide [2] sollte man als Richtwert für eine OLTP-Datenbank, die hier als Lastszenario angenommen wird, ca. 65% des Speichers für die SGA vorsehen. Daraus ergibt sich folgendes Beispiel zur Berechnung der optimalen Hugepages-Kapazität für eine ODA mit 256 GB Hauptspeicher:

256 GB - 12 GB (Betriebssystem) = 244 GB für die DB
 244 GB * 65% ≈ 160 GB = 163840 MB = 81920 2MB Hugepages

Will man nun die Hugepages auf diesen Wert ändern, kann man `odacli update-osconfigurations` verwenden, welches normalerweise genutzt wird, wenn man der ODA ein Hauptspeicher-Upgrade gegönnt hat. Doch die Speicherseiten kann man damit wie in *Listing 9* anpassen.

Fazit

Die Oracle Database Appliance bietet mit ihren Bordmitteln einige Techniken, um die Verteilung der Rechenleistung und Speicherkapazität in geordnete Bahnen zu lenken. Alle vorgenannten Techniken lassen sich auch wunderbar miteinander kombinieren, um unterschiedlichste Bedürfnisse zu bedienen. Besonders auch im Ausblick auf neuere Prozessorgene-

rationen ergeben sich damit auf dieser Plattform vielfältige Möglichkeiten zur Konsolidierung verschiedenster Workload-Arten. Währenddessen hat sich auch gezeigt, dass Oracle die Möglichkeiten der Appliance-Software immer weiter um neue Funktionen anreichert und das Ressourcenmanagement hier sicher noch nicht am Ende ist.

Quellen

- [1] Oracle Database Appliance Documentation - <https://docs.oracle.com/en/engineered-systems/oracle-database-appliance/index.html>
- [2] Database Performance Tuning Guide - <https://docs.oracle.com/en/database/oracle/oracle-database/19/tgdba/tuning-system-global-area.html>

Über den Autor

Sven Illert kommt seit 2008 mit verschiedensten Oracle-Technologien beruflich in Berührung und trug auch schon zu einigen DOAG-Veranstaltungen als Vortragender bei. Aktuell ist er als Leitender Systemberater der Robotron Datenbank-Software GmbH in vielfältigen Projekten, unter anderem bei den Themen hochverfügbare Datenbank-Infrastrukturen, Engineered Systems und Oracle Cloud Infrastructure, tätig.



Sven Illert
 sven.illert@robotron.de



IJUG

Verbund

www.ijug.eu

FÜR 29,00 €
BESTELLEN

Java aktuell

JAHRESABO

Mehr Informationen zum Magazin und Abo unter:
www.ijug.eu/de/java-aktuell





APEX Dynamic Controller (ADC)

Jürgen Sieben, ConDeS

APEX-Anwendungsseiten haben eine klare Trennung der Zuständigkeiten zwischen der Kontrolle der Seite und der Darstellung dieser. Diese Trennung ergibt sich zwangsläufig aus der verwendeten Programmiersprache PL/SQL, die im Browser nicht funktioniert. Daher sind alle Prozesse zur Kontrolle des Seitenaufbaus und zur Verarbeitung von Benutzereingaben der Datenbank vorbehalten, während die Anwendungsseite selbst sich auf die Visualisierung der Daten beschränkt. Die Logik, die aus Berechnungen, serverseitigen Bedingungen, Autorisierungsschemata, aber auch Validierungen, Verarbeitungsprozessen und Verzweigungen besteht, kontrolliert daher die Seite vollständig aus der Datenbank heraus. Dies gilt zumindest, solange die Anwendungsseite nicht dynamisch auf die Bedienung durch den Anwender reagieren muss.

Das Problem

Soll die Anwendungsseite dynamisch auf Benutzereingaben reagieren können,

steht deklarativ nur die dynamische Aktion zur Verfügung, die strukturell anders arbeitet als die Logik in der Datenbank: Im Kern stellen dynamische Aktionen

Codegeneratoren dar, die einfaches, dynamisches Verhalten in die Seite „injizieren“. Sie stellen eigentlich keinen Controller in dem Sinn dar, dass auf eine Eingabe

des Benutzers dynamisch reagiert wird, sondern fügen der Seite einfache dynamische Reaktionen für vorhersehbare Benutzerinteraktionen hinzu. Um konditional auf die Eingaben des Benutzers zu reagieren, steht lediglich eine einfache Ja-Nein-Entscheidung zur Verfügung. Dies ist auch völlig in Ordnung, solange die Anforderungen an das dynamische Verhalten entsprechend einfach sind.

Komplexe Formularseiten stellen aber häufig Anforderungen an das dynamische Verhalten, die durch Dynamische Aktionen allein nicht mehr beherrscht werden können. Dies kann bereits durch überraschend einfache Anforderungen geschehen:

- Ein Eingabefeld soll dynamisch gegen Datenbanklogik validiert werden.
- Basierend auf Daten der Datenbank soll ein Feld dynamisch ein- oder ausgeblendet werden.
- Ein Eingabefeld soll konditional verpflichtend oder optional gestellt werden.
- Eine Auswahlliste soll aktualisiert und direkt ein Standardwert festgelegt werden.
- ...

Um diese Probleme zu lösen, ist die übliche Herangehensweise, das dynamische Verhalten in JavaScript zu implementieren, was naheliegend erscheint, weil eine andere Programmiersprache im Browser nicht zur Verfügung steht. Mit dieser Entscheidung entfällt allerdings auch die Möglichkeit, mit den Mitteln von Low Code eine Anwendungsseite zu kontrollieren. Erschwerend kommt hinzu, dass die meisten Anwendungsfälle die Strichaufzählung erfordern, sodass der JavaScript-Code mit der Datenbank kommuniziert, um eine Entscheidung treffen zu können. Dies führt zu vielen Roundtrips zur Datenbank. Zudem können dynamische Aktionen mit dem Ergebnis der Berechnung innerhalb der Datenbank nicht direkt umgehen, sondern müssen dies in ein weiteres, verstecktes Eingabefeld auf der Seite speichern und mit einer zweiten dynamischen Reaktion auf eine Änderung dieses Felds reagieren. Einige Beispiele für solche Anwendungsfälle:

- Zeige das Feld „Bonus“ nur, falls der aktuell gewählte Beruf bonusberech-

tigt ist. In diesem Fall ist das Feld ein Pflichtfeld.

- Prüfe, ob sich das Gehalt bezüglich des eingestellten Berufs sich im erlaubten Gehaltskorridor befindet.
- Prüfe, ob der angemeldete Benutzer berechtigt ist, Geld an die angegebene Bank zu überweisen.

Diese Prüfungen können clientseitig nur dann entschieden werden, wenn die Entscheidungsdaten, zusätzlich zu den Eingaben des Benutzers, dem Browser zur Verfügung gestellt werden. Dies erhöht den Aufwand noch einmal erheblich.

Alternativlösung

Als Alternative für diese Vorgehen schlage ich vor, einen „echten“, dynamischen Controller einzuführen und diesen innerhalb der Datenbank auszuführen. Dynamische Controller platzieren auf der Anwendungsseite sogenannte „Observer“, also Eventhandler, die dem Controller lediglich mitteilen, dass etwas von Interesse auf der Anwendungsseite geschehen ist. Sie übergeben dem Controller relevante Informationen der aktuellen Seite. Diese Informationen möchte ich, in Anlehnung an den SessionState, „PageState“ nennen. In ihm ist nicht nur der aktuelle Wert relevanter Eingabeelemente enthalten, sondern auch Informationen über das auslösende Seitenelement und weitere Informationen.

Dieser PageState wird bei jedem Aufruf an die Datenbank kommuniziert und dort mit einem Regelwerk aus Anwendungsfällen verglichen, welches das nun erforderliche Verhalten berechnet und als Antwort an die Anwendungsseite zurückschickt. Diese Antwort hat die Form eines JavaScript-Codeblocks, der in die Seite eingefügt und direkt ausgeführt wird.

Im Gegensatz zu dynamischen Aktionen bietet dieser Ansatz den Vorteil, Entscheidungen direkt innerhalb der Datenbank, basierend auf den dort vorhandenen Daten und den Methoden in PL/SQL, treffen zu können. Ein wesentlicher, weiterer Vorteil: Sollte die Datenbank entscheiden, den PageState zu ändern und damit einen weiteren Anwendungsfall anzustoßen, wird dieser rekursiv innerhalb der Datenbank berechnet und

in die Antwort integriert. Auf diese Weise haben auch komplexe Szenarien lediglich einen Roundtrip zur Datenbank zur Folge, was in Summe diesen Ansatz leichtgewichtiger macht als eine entsprechende Programmierung in JavaScript mit mehreren Roundtrips.

Die Anwendungsfälle, die ich in der vorhergehenden Strichaufzählung als problematisch gekennzeichnet hatte, sind dies nun nicht mehr, denn die dynamische Validierung ist innerhalb der Datenbank kein Problem, ebenso wenig die Entscheidung, ob ein Seitenelement nun ein Pflichtfeld sein soll oder nicht. Datenbasierte Validierungen sind ohnehin kein Problem innerhalb der Datenbank und bieten zudem den Vorteil, eine einmal geschaffene Validierungslogik innerhalb der Datenbank auch für das dynamische Verhalten wiederverwenden zu können. Voraussetzung für diesen Ansatz ist eine JavaScript-Komponente, die als Kommunikationsweg für den dynamischen Controller verwendet werden kann. Diese Komponente möchte ich in diesem Artikel näher vorstellen.

Realisierungsvorschlag

Realisiert wird der dynamische Controller als Dynamic Action Plugin mit dem Namen ADC, was für „APEX Dynamic Controller“ steht. Das Plugin kann auf Seite 0 beim Seitenladen installiert werden und benötigt keinerlei Parametrierung auf der Anwendungsseite. Wird das Plugin verwendet, ändert sich bereits das Verhalten der Anwendungsseite: Pflichtfelder werden dynamisch validiert, ebenso reagieren Datums- und Zahlenfelder unmittelbar auf Falscheingaben.

Natürlich ist eine entsprechende Logik auf der Datenbankseite erforderlich. Diese Logik wird einerseits in Packages realisiert, aber auch von einer APEX-Anwendung flankiert, mittels derer das dynamische Verhalten für die Zielanwendung kontrolliert wird. Im Kern bestehen diese Anwendungsfälle aus einfachen Metadaten, die zu einer Entscheidungstabelle zusammengefasst werden, und anhand dieser Daten wird dann entschieden, welche Aktion als Reaktion auf einen spezifischen PageState erforderlich ist.

Für einfachere Fälle können diese Anwendungsfälle vollständig deklarativ ein-

Nr. ↑	Wenn der Anwender ...	dann ...	technische Bedingung	initial	aktiv
10	die Seite öffnet	<ul style="list-style-type: none"> mache Selektor ".sadc-mandatory" zum Pflichtfeld führe Anwendungsfälle des Elements "P7_EMP_JOB_ID" aus 	initializing = c_true	▶	✓
20	einen Beruf mit Bonusberechtigung wählt	<ul style="list-style-type: none"> mache Feld "P7_EMP_COMMISSION_PCT" zum Pflichtfeld 	sadc_ui.is_comm_eligible(P7_EMP_JOB_ID) = C_TRUE	✗	✓
30	einen Beruf ohne Bonusberechtigung wählt	<ul style="list-style-type: none"> mache Feld "P7_EMP_COMMISSION_PCT" optional setze die Sichtbarkeit des Seitenelements "P7_EMP_COMMISSION_PCT" auf Status "SHOW_DISABLE" 	sadc_ui.is_comm_eligible(P7_EMP_JOB_ID) = C_FALSE	✗	✓

1 Zeilen ausgewählt 1 - 3

Abbildung 1: Anwendungsfälle einer Seite (Quelle: Jürgen Sieben)

Eigenschaften >

+ Aktion erstellen

🗑️ Aktion löschen

Aktion

*** Sequenz** ?

*** Aktion** ?

*** Seitenelement** ?

Zeichenkette oder Meldungsname ?

jQuery-Selektor ?

Abbildung 2: Parametrierung des Aktionstypen (Quelle: Jürgen Sieben)

gerichtet werden. Doch ist es auch möglich, die Antwort dynamisch in PL/SQL zu berechnen und somit beliebig komplexe Anwendungsfälle mit geringem Codeaufwand zu realisieren. Für den Entwickler

bietet sich so die Möglichkeit, den Abstraktionsgrad des Anwendungsfalls vollständig frei zu wählen: Einfache Dinge werden deklarativ erledigt, komplexere auf abstrakterer Ebene programmatisch gelöst.

Bedienung

Die Idee von ADC besteht darin, dass eine APEX-Anwendungsseite durch das Plugin zu einer „dynamischen Seite“ erweitert

Eigenschaften >

+ Aktion erstellen

🗑️ Aktion löschen

Aktion

*** Sequenz** ?

*** Aktion** ?

*** Seitenelement** ?

*** PL/SQL-Prozedur** ?

Kommentar

Fehlerhandler ?

Aktiv ?

ist Validierung ?

Rekursiv ?

Abbildung 3: Aufruf einer Prozedur zur Validierung von Eingaben (Quelle: Jürgen Sieben)

wird. Auf ihr werden „Anwendungsfälle“ definiert, die, wenn sie auftreten, „Aktionen“ auslösen. Als Anwendungsfall werden bei ADC nicht notwendigerweise die „klassischen“ Anwendungsfälle einer Anwendung verstanden, sondern Aktivitäten des Benutzers, auf die dynamisch reagiert werden soll. Dies kann zum Beispiel die Betätigung einer Schaltfläche sein, die

Eingabe in ein Formularfeld oder das Auswählen einer Zeile in einem Bericht.

Damit die Definition der Anwendungsfälle deklarativ gelingt, steht im ADC eine Administrationsanwendung zur Verfügung, die im Kern aus dem ADC-Designer besteht. Diese Anwendungsseite ist dem Page-Designer nachempfunden und ermöglicht die zentrale Erfassung von An-

wendungsfällen für alle dynamischen Anwendungsseiten aller APEX-Anwendungen eines Workspaces.

Der Anwendungsfall wird fachlich formuliert, als Fortführung des Satzes „Wenn der Anwender ...“. Dafür gibt es selbstverständlich keinen technischen Grund, doch hat sich gezeigt, dass ein zentraler Vorteil von ADC in der guten Dokumen-

Seitenkommando

* Aktionstyp

* Aktionsname

* Darstellungstext

Beschreibung

Bestätigungsmeldung

Icon-Klasse

Icon

Tooltip

Tastaturkürzel

Zugeordnete Seitenelemente

Abbildung 4: Einrichtung eines Seitenkommandos (Quelle: Jürgen Sieben)

tation des dynamischen Verhaltens besteht, und diese profitiert von fachlich formulierten Anwendungsfällen.

Wird dieser Anwendungsfall erkannt, können, wie bei Dynamischen Aktionen, eine oder mehrere Aktionen aus einer Liste vordefinierter Aktionstypen ausgewählt und parametrisiert werden. Im Gegensatz zu Dynamischen Aktionen ist es nicht möglich, eine Fallunterscheidung in TRUE- oder FALSE-Aktionen zu berücksichtigen, doch wird dieser Nachteil durch die Möglichkeit überkompensiert, komplexe Szenarien vollständig in PL/SQL programmieren zu können.

Damit ADC erkennen kann, ob ein Anwendungsfall aufgetreten ist oder nicht, muss eine „technische Bedingung“ formuliert werden. Die Syntax zur Formulierung dieser technischen Bedingung entspricht einer where-Klausel in SQL. Dabei stehen alle Seitenelemente der aktuellen

Anwendungsseite als Spalten mit ihren aktuellen Eingabewerten typischerweise zur Verfügung (Ein Datumsfeld also als „echtes“ Datum etc.). Diese Eigenschaft ist zentral für die Leistungsfähigkeit des Plugins, denn da die Entscheidung über einen Anwendungsfall durch SQL getroffen wird, stehen alle Möglichkeiten dieser Sprache, zusätzlich aber auch alle Möglichkeiten von PL/SQL, zur Entscheidungsfindung zur Verfügung.

Beispiel

Als Beispiel soll die Anwendung auf die Auswahl eines Berufes reagieren: Ist der ausgewählte Beruf bonusberechtigt, soll das Feld für die Höhe des Bonus eingeblendet und zum Pflichtfeld gemacht werden, anderenfalls soll das Feld optional sein und ausgeblendet werden.

In ADC lautet der erste fachliche Anwendungsfall: „Wenn der Anwender einen Beruf mit Bonusberechtigung wählt“, als Aktion wird vereinbart, das Eingabefeld P7_EMP_COMMISSION_PCT zum Pflichtfeld zu machen, wie in *Abbildung 1* gezeigt. Da ein Pflichtfeld, das nicht sichtbar oder deaktiviert ist, keinen Sinn ergibt, enthält der Aktionstyp immer auch das Anzeigen und Aktivieren des Feldes. Als technische Bedingung kann nun eine PL/SQL-Funktion der Anwendungslogik aufgerufen werden, die entscheidet, ob der Beruf bonusberechtigt ist:

```
sadc_ui.is_comm_eligible(P7_EMP_JOB_ID) = C_TRUE
```

Beachten Sie, dass der aktuelle Wert des Feldes P7_EMP_JOB_ID als Spaltenwert zur Verfügung gestellt wird und daher direkt als Parameter an die selbsterstellte

Methode `sadc_ui.is_comm_eligible` übergeben werden kann. Die Funktion liefert einen Wahrheitswert zurück, der in SQL verwendet werden kann (freuen wir uns auf Version 23c ...). Die Spalte `C_TRUE` enthält ebenfalls diesen Wert, so dass der Vergleich direkt auf die oben beschriebene Weise ausgeführt werden kann.

Diesem Anwendungsfall wird nun der Aktionstyp „*mache Feld verpflichtend*“ zugeordnet und dort parametrisiert, dass das Feld `P7_EMP_COMMISSION_PCT` hiervon betroffen sein soll, wie in *Abbildung 2* gezeigt. Analog wird ein zweiter Anwendungsfall mit der umgekehrten Bedingung angelegt und dort festgelegt, dass das Eingabefeld ausgeblendet werden soll. Dadurch wird es direkt in ein optionales Feld umgewandelt.

Überraschend mag sein, dass das Plugin keinerlei Administration erfordert. Woher aber weiß das Plugin, welche Observer auf der Anwendungsseite einzurichten sind? Die Observer ergeben sich aus der Summe der technischen Bedingungen, denn nur, wenn ein Seitenelement in einer technischen Bedingung angesprochen wird, ergibt ein Observer auf dieses Element Sinn, denn eine Änderung dieses Elements hat eine Reaktion des Regelwerks zur Folge. Umgekehrt: Taucht ein Seitenelement in den technischen Bedingungen nicht auf, ergibt ein Observer keinen Sinn, weil eine Änderungsmeldung für dieses Seitenelement

keine Antwort des Controllers ermitteln kann, es liegen ja keine Anwendungsfälle hierfür vor. Das Regelwerk selbst kann also ermitteln, welche Observer es benötigen wird. Diese Liste wird dem Plugin beim Rendern der Seite mitgeteilt. Da sich aber die Aufgabe der clientseitigen Komponente darin erschöpft, die Observer einzurichten, Events zu melden und die Antwort auszuführen, ergibt sich keine Notwendigkeit für eine manuelle Administration.

Auffällig an der Anwendung ist die gute Dokumentationsqualität der dynamischen Aktivitäten: In *Abbildung 1* erkennen Sie auf einen Blick, was passiert, wenn die Seite geöffnet wird und welche Reaktion auf die Auswahl eines Berufs erfolgt.

Der Schwerpunkt des ADC ist die dynamische Validierung von Benutzereingaben sowie die Steuerung des Anzeigestatus der Seite. Anhand eines zweiten Beispiels wird deutlich, wie einfach eine dynamische Validierung eingerichtet werden kann. Das Problem dieser Validierungen besteht darin, dass im Regelfall eine einfache, clientseitige Prüfung auf ein Muster nicht ausreicht, um eine Benutzereingabe zu validieren. Anforderungen wie „*Das Datum muss mindestens 5 Bankarbeitstage in der Zukunft liegen*“ ist ohne Daten nicht umsetzbar. Hinzu kommt, dass auf Grund des schlechten Vertrauensverhältnisses zwischen Client und

Server diese Validierung auf der Serverseite noch einmal implementiert werden muss. Viel einfacher ist es, wenn auch für die clientseitige Validierung die serverseitig implementierte Validierungsmethode genutzt werden kann.

Da ADC stets in der Datenbank arbeitet und direkten Zugriff auf die aktuellen Werte der Anwendungsseite hat, ist diese Anforderung kein Problem. Als Anwendungsfall wird nun ein allgemeiner Fall angenommen: „*Wenn der Anwender eine Zahl eingibt*“, als technische Bedingung wird lediglich `P6_NUMBER` is not null geschrieben. Dann kann als Validierung ein PL/SQL-Block ausgeführt werden, wie in *Abbildung 3* zu sehen ist.

Der Code der Prozedur `sadc_ui.validate_p6_number` ist in *Listing 1* zu sehen.

Der entscheidende Aufruf lautet `adc.register_error`. Mit dieser Methode wird ein Fehler im ADC registriert und damit auf der Anwendungsseite angezeigt. Selbst dieses triviale Beispiel zeigt die Mächtigkeit dieses Ansatzes, denn nun kann die gesamte Programmierung innerhalb der Datenbank wiederverwendet werden. Es ist lediglich eine Wrapper-Methode erforderlich, die das Validierungsergebnis entgegennimmt, für ADC aufbereitet und dort registriert.

Im Gegensatz zum „normalen“ APEX erfolgt die Darstellung der Fehlermeldungen feingranular: Nur die behobe-

Oracle Datenbanken Monthly News

Auf dem deutschsprachigen Oracle-Blog ist die November-Ausgabe der News-Serie erschienen.

DOAG Online

Es ist wieder so weit: die neue Ausgabe ist online! Das sechsköpfige Redaktionsteam von Oracle Deutschland hat wieder Neuigkeiten rund um die Oracle-Datenbank für On-Premises und Cloud-Installation zusammengestellt.

Alles wird wieder in einem Video präsentiert.

In der aktuellen Ausgabe wird wieder ein zusätzliches Quick Link Posting (in Englisch) zur Verfügung gestellt, um

einen schnellen Zugriff auf die zugehörigen Links zu gewährleisten.

<https://www.doag.org/de/home/news/oracle-datenbanken-monthly-news-28/>



nen Fehler werden ausgeblendet, alle noch bestehenden Fehlermeldungen bleiben sichtbar. Zudem verfügt ADC auch über die Möglichkeit, Warnungen auszugeben.

Seitenprozesse

Zusätzlich zu den Aktionstypen lassen sich Seitenprozesse definieren. Diese Prozesse stellen, technisch gesehen, eine deklarative Oberfläche für APEX-Actions dar, die das APEX-Team selbst sehr intensiv verwendet, zum Beispiel im interaktiven Grid, für das es aber keine deklarative Oberfläche gibt. Sinnvoll wäre dies allemal, denn mit APEX-Actions lassen sich Aufgaben zentral vereinbaren und an mehrere Oberflächenelemente binden (zum Beispiel an Schaltflächen, aber auch an Listeneinträge oder Einblendmenüs). ADC bietet nicht nur die Möglichkeit, deklarativ mit Seitenprozessen zu arbeiten, sondern auch ein Hilfspackage, das es ermöglicht, bereits bestehende APEX-Actions dynamisch an die Seitensituation anzupassen. Damit ist es möglich, einer Schaltfläche, basierend auf Aktivitäten des Anwenders, unterschiedliche Funktionalität zuzuweisen.

Im ADC ist das Standardverhalten einer APEX-Action, ADC über das Auslösen des Seitenkommandos zu informieren. In der Datenbank kann nun dynamisch entschieden werden, ob zum Beispiel eine Weiterleitung auf eine andere Anwendungsseite ausgeführt oder eine Warnmeldung ausgegeben werden soll.

Als Beispiel für Seitenkommandos soll implementiert werden, dass immer dann, wenn ein Benutzer eine Zeile eines Berichts wählt, entschieden werden soll, ob eine Schaltfläche zum Editieren dieses Benutzers aktiv oder inaktiv geschaltet werden soll. Alternativ wäre dies mit Bordmitteln über eine entsprechende Spalte mit einem Link zu lösen, doch geht es hier darum, ein kompaktes Beispiel zu finden.

Zunächst wird ein Seitenkommando erstellt, mit einem Namen und optional auch mit weiteren Angaben wie Label, Icon, Tastaturkürzel versehen. Außerdem wird es deklarativ einer Schaltfläche zugewiesen, wie in *Abbildung 4* zu sehen ist (Schaltflächen und Regionen

```
procedure validate_p6_number
as
  l_number number;
begin
  -- Initialization
  l_number := utl_apex.get_number('P6_NUMBER');

  if l_number between 100 and 1000 and mod(l_number, 3) = 0 then
    null;
  else
    adc.register_error(
      p_cpi_id => 'P6_NUMBER',
      p_error_msg => 'Die Zahl muss zwischen 100 und 1000 liegen
        und durch 3 teilbar sein.');
```

Listing 1: Validierungsprozedur

```
procedure adact_control_action
as
  l_emp_id sadc_ui_adact.emp_id%type;
  l_is_manager adc_util.flag_type;
  l_label adc_util.ora_name_type;
begin
  pit.enter_mandatory;

  -- Initialization
  l_emp_id := adc_api.get_event_data;
  adc_apex_action.action_init('edit-employee');

  select emp_is_manager, substr(emp_first_name, 1, 1) || '. ' ||
emp_last_name || ' bearbeiten'
  into l_is_manager, l_label
  from hr_jobs
  join hr_employees
    on job_id = emp_job_id
  where emp_id = l_emp_id;

  if l_is_manager = adc_util.C_TRUE then
    adc_apex_action.set_label('Nicht bearbeitbar');
    adc_apex_action.set_disabled(true);
  else
    adc_apex_action.set_label(l_label);
    adc_apex_action.set_disabled(false);
    adc_apex_action.set_href(
      utl_apex.get_page_url(
        p_page => 'edemp',
        p_param_items => 'P9_EMP_ID',
        p_value_list => l_emp_id,
        p_triggering_element => 'B8_EDIT_EMP',
        p_clear_cache => 9));
  end if;
  adc.add_javascript(adc_apex_action.get_action_script);

  pit.leave_mandatory;
exception
  when others then
    pit.handle_exception(msg.PIT_PASS_MESSAGE, msg_args(adc_api.
get_event_data));
end adact_control_action;
```

Listing 2: Methode zur Manipulation eines Seitenkommandos

sind für ADC verfügbar, sobald ihnen eine statische ID zugeordnet wurde).

Wird das Seitenkommando ausgeführt, wird dies der Datenbank über die Pseudospalte COMMAND mitgeteilt, die nun den Namen des aufgerufenen Seitenkommandos enthält. Die technische Bedingung kann dies also durch `command = 'edit-employee'` detektieren und darauf reagieren. Wieder wird eine Prozedur aufgerufen, die das Verhalten der Schaltfläche berechnen kann.

Die eigentliche Manipulation des Seitenkommandos erfolgt über das Hilfspackage `ADC_APEX_ACTION`, das über entsprechende Methoden die erforderlichen JavaScript-Snippets für die Antwort berechnet, wie in *Listing 2* gezeigt wird.

Ich denke, als Einstieg in die Funktionsweise sollte dies ausreichen. Natürlich wird die Anwendung durch Möglichkeiten flankiert, die erstellten Anwendungsfälle als Skriptdatei zu exportieren und so auszuliefern. Zudem stehen Möglichkeiten zur Verwaltung der Stammdaten von ADC selbst zur Verfügung, die es erlauben, eigene Aktionstypen zu erstellen. Eine Besprechung führt hier aber zu weit.

Mit den vorgestellten Möglichkeiten lassen sich bereits komplexe Formulareseiten dynamisch kontrollieren. Für extremere Fälle, wie zum Beispiel die ADC-Designer-Seite selbst, können die Aktionstypen nicht nur deklarativ über die Oberfläche einem Anwendungsfall zugeordnet werden, sondern auch programmgesteuert direkt aus PL/SQL. Die überraschende Konsequenz hieraus ist, dass nun, im Gegensatz zu Dynami-

schen Aktionen, der Abstraktionsgrad eines Anwendungsfalls frei gewählt werden kann: Ähnliche Reaktionen, die durch unterschiedliche Ereignisse ausgelöst werden, können in einer Prozedur behandelt werden, die auf Grund von Entscheidungslogik die Auswahl und Parametrierung der Aktionstypen dynamisch vornimmt. Daher fällt in diesen Szenarien auch die Unfähigkeit, TRUE- und FALSE-Aktionen einzurichten, nicht ins Gewicht, sondern wird durch einen ungleich mächtigeren Mechanismus ersetzt: Nämlich der freien Programmierbarkeit durch PL/SQL. Als Ergebnis wird der ADC-Designer durch gerade einmal sieben Anwendungsfälle kontrolliert und hat doch ein reiches, dynamisches Verhalten.

Auch die Codemenge, die für die Berechnung der dynamischen Effekte erforderlich ist, hält sich mit etwa 800 Zeilen PL/SQL-Code in engen Grenzen für das erreichte Ergebnis.

Fazit und Verfügbarkeit

ADC hat sich in den Projekten, in denen es eingesetzt wird, als beinahe unersetzliches Werkzeug bewährt. Auch Einsteigern in APEX fällt es hiermit leicht, komplexes dynamisches Verhalten in APEX zu realisieren. Zudem werden die Grenzen dessen, was ohne umfangreiche JavaScript-Programmierung möglich ist, deutlich ausgeweitet. Sicher ist nicht jeder Wunsch durch dieses Werkzeug realisierbar und es ist definitiv auch kein vollständiger Ersatz für Dynamische Aktionen, doch gelingt die Kon-

trolle dynamischer Anwendungsseiten leicht, in hoher Dokumentationsqualität und guter Performanz.

Das Plugin ist als Open-Source-Projekt über GitHub erhältlich (www.github.com/j-sieben/ADC) und unterliegt der MIT-Lizenz. Allerdings konnte ich dem allgemeinen Trend zur Ein-Klick-Installation nicht recht folgen, zu groß wären die Einschränkungen oder der Aufwand, eine entsprechende Version parallel zu pflegen. ADC basiert auf folgenden Open-Source-Projekten:

- PIT (www.github.com/j-sieben/PIT)
Neben der vollständigen Instrumentierung des Codes inkl. Ausgabe in APEX ist PIT für die Validierungen und die Internationalisierung des Datenmodells verantwortlich.
- UTL_TEXT (www.github.com/j-sieben/UTL_TEXT)
Aus diesem Utility-Package verwendet ADC vor allem den CodeGenerator zur Erzeugung der internen Entscheidungstabellen und Exportskripte.
- UTL_APEX (www.github.com/j-sieben/UTL_APEX)
Dieses, für mich unverzichtbare, Tool kapselt einige Teile der APEX-API. Es erlaubt einen typischeren Zugriff auf den Session-Status und wird für viele weitere Zwecke verwendet.



Jürgen Sieben
j.sieben@condes.de

BEST OF DOAG ONLINE

Eine Auswahl der besten DOAG News Oktober/November 2023



DOAG Datenbank Kolumne: Die Datenbank, der Client und NLS_LANG

Rund um das Thema "Passende Spracheinstellungen auf dem Datenbank-Client" gibt es viele Mythen, so dass es sich lohnt, die zentrale Variable in diesem Kontext, NLS_LANG, mal genauer anzuschauen.



Nächste Runde im DOAG.tv

In den vergangenen vier Jahren entstanden fast 100 Clips im DOAG.tv. Auf dem Konferenz-Triumvirat vom 21. bis 24. November in Nürnberg werden die nächsten Hochkaräter interviewt.



"Ich fühle mich manchmal wie eine unsichere Hochstaplerin und Betrügerin"

Dr. Melissa Sassi widmet sich in ihrer Keynote auf der DOAG Konferenz + Ausstellung dem Imposter-Syndrom, einem weitverbreiteten psychologischen Phänomen. Vorab stand uns die US-Amerikanerin zum Interview zur Verfügung.



DOAG Datenbank Kolumne: Alert.log-Monitoring selbstgemacht

Als Administratorin einer überschaubaren Anzahl von Datenbanken (ohne Multitenant) lasse ich mich gern per Mail informieren, wenn irgendetwas Wissenswertes im Alert-Log auftaucht.



DOAG 2023 Konferenz + Ausstellung: Zurück zu alter Stärke

Die Anwenderkonferenz begeisterte rund 1.800 Teilnehmerinnen und Teilnehmer in Nürnberg.



KI NAVIGATOR 2023: Erfolgreiche Premiere

Rund 500 Teilnehmerinnen und Teilnehmer besuchten die erste Ausgabe der Konferenz zur Praxis der KI in IT, Wirtschaft und Gesellschaft.



Wir begrüßen unsere neuen Mitglieder

Natürliche Mitglieder:

- Kai Blüme
- Dr. Jens Oehlschlägel
- Ulf Krum
- Frank Böhmer
- Michael Ternes
- Volker Enkrodt
- Manuel Gandt
- Lida Gholizadeh
- Abdelrhman Mohamed
- Alexander Groene

Korporative Mitglieder:

- Fraunhofer-Institut für Lasertechnik ILT, Repräsentantin: Alexandra Reinartz
- Kieback & Peter GmbH & Co. KG, Repräsentant: Christian Wolff



Termine

Januar

01

11.01.2024

**DOAG DevTalk: SQLcl-
Erfahrungsaustausch
DevTalk mit Robert Marz und
Oliver Lemm**

Online

April

04

09. - 11.04.2024

JavaLand 2024
Zwei ereignisreiche Konferenztage mit
anschließendem Schulungstag rund
um das Java-Ökosystem

Nürburgring, Nürburg

22. - 24.04.2024

APEX connect 2024
Konferenz mit zahlreichen Vorträgen
und Workshops zu den Themen APEX,
JavaScript und PL/SQL

Düsseldorf

25. - 27.04.2024

**DOAG Leitungskräfteforum und
Delegiertenversammlung**

Berlin

Mai

05

15. - 16.05.2024

DOAG 2024 Datenbank mit Exaday
Konferenz rund um die Oracle
Database und Engineered Systems

Düsseldorf

Impressum

Red Stack Magazin inkl. Business News wird gemeinsam herausgegeben von den Oracle-Anwendergruppen DOAG Deutsche ORACLE-Anwendergruppe e.V. (Deutschland, Tempelhofer Weg 64, 12347 Berlin, www.doag.org), AOUG Austrian Oracle User Group (Österreich, Lassallestraße 7a, 1020 Wien, www.aoug.at) und SOUG Swiss Oracle User Group (Schweiz, Dornacherstraße 192, 4053 Basel, www.soug.ch).

Red Stack Magazin inkl. Business News ist das User-Magazin rund um die Produkte der Oracle Corp., USA, im Raum Deutschland, Österreich und Schweiz. Es ist unabhängig von Oracle und vertritt weder direkt noch indirekt deren wirtschaftliche Interessen. Vielmehr vertritt es die Interessen der Anwender an den Themen rund um die Oracle-Produkte, fördert den Wissensaustausch zwischen den Lesern und informiert über neue Produkte und Technologien.

Red Stack Magazin inkl. Business News wird verlegt von der DOAG Dienstleistungen GmbH, Tempelhofer Weg 64, 12347 Berlin, Deutschland, gesetzlich vertreten durch den Geschäftsführer Fried Saacke, deren Unternehmensgegenstand Vereinsmanagement, Veranstaltungsorganisation und Publishing ist.

Die DOAG Deutsche ORACLE-Anwendergruppe e.V. hält 100 Prozent der Stammeinlage der DOAG Dienstleistungen GmbH. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. wird gesetzlich durch den Vorstand vertreten; Vorsitzender: Björn Bröhl. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. informiert kompetent über alle Oracle-Themen, setzt sich für die Interessen der Mitglieder ein und führt einen konstruktiv-kritischen Dialog mit Oracle.

Redaktion:

Sitz: DOAG Dienstleistungen GmbH
(Anschrift s.o.)
ViSdP: Fried Saacke
Redaktionsleitung Red Stack Magazin:
Martin Meyer
Redaktionsleitung Business News:
Marcos López
Kontakt: redaktion@doag.org

Weitere Redakteure (in alphabetischer Reihenfolge): Dennis Amthor, Birgit Arndt, Kai Donato, Philipp Hartenfeller, Dr. Roxana Hess, Dennis Hering, Sven Illert, Katja Rische, Robert Maier-Silldorff, Yahya Mohamed Mao, Raphael Salguero, Ulf Schabram, Jürgen Sieben, Dr. Christian Temath, Gerald Venzl.

Titel, Gestaltung und Satz:

Diana Tkach
DOAG Dienstleistungen GmbH
(Anschrift s.o.)

Fotonachweis:

Titel: © macrovector | www.freepik.com
S. 12: © Martin Meyer
S. 18: © nockewell1 | www.pixabay.com
S. 26: © Bru-nO | www.pixabay.com
S. 30: © Liggraphy | www.pixabay.com
Titel S. 36: © freepik | www.freepik.com
S. 42 © grünheller | www.istockphoto.com
S. 46: © macrovector | www.freepik.com
S. 50: @ videoflow | www.pixabay.com
S. 54: © NoName_13 | www.pixabay.com
S. 66: © videoflow | www.pixabay.com
S. 74: © 12138562 | www.pixabay.com
S. 80: © pexels | www.pixabay.com
S. 129: © freepik | www.freepik.com

Anzeigen:

sponsoring@doag.org

Mediadaten und Preise:

www.doag.org/go/mediadaten

Druck:

WIRmachenDRUCK GmbH,
www.wir-machen-druck.de

Alle Rechte vorbehalten. Jegliche Vervielfältigung oder Weiterverbreitung in jedem Medium als Ganzes oder in Teilen bedarf der schriftlichen Zustimmung des Verlags.

Die Informationen und Angaben in dieser Publikation wurden nach bestem Wissen und Gewissen recherchiert. Die Nutzung dieser Informationen und Angaben geschieht allein auf eigene Verantwortung. Eine Haftung für die Richtigkeit der Informationen und Angaben, insbesondere für die Anwendbarkeit im Einzelfall, wird nicht übernommen. Meinungen stellen die Ansichten der jeweiligen Autoren dar und geben nicht notwendigerweise die Ansicht der Herausgeber wieder.

Inserentenverzeichnis

DOAG e.V.
www.doag.org

U 2, U 3, U 4

DOAG e.V.
www.doag.org

S. 3, S. 7, S. 25, S. 34, S. 79,

Promatis Gruppe
www.promatis.de

S. 47

DOAG

DOAG
Datenbank
mit Exaday

2023

ON DEMAND

DATENBANK 2023 VERPASST?

**Jetzt On-demand-Ticket buchen und
Vortragsaufzeichnungen anschauen!**



**ALLE ANGEBOTE
IM TICKETSHOP**



DOAG

DOAG²⁰²⁴ Datenbank
mit Exaday

15. und 16. Mai in Düsseldorf

datenbank.doag.org

