

# BEST PRACTICES FOR INTEGRATING SALESFORCE WITH SAP



Geschrieben von...

Alexander Ilg

*Alex ist Produktmanager und Hauptberater mit mehr als 20 Jahren Erfahrung im Bereich Unternehmenssoftware, hauptsächlich Salesforce und SAP.*

Gepostet in [Architekten](#)

## Best Practices für die Integration von Salesforce mit SAP

Um die wichtigsten Geschäftsprozesse zu verwalten, verfolgen die meisten Unternehmen einen Best-of-Breed-Ansatz. Anstatt sich auf einen einzelnen, großen Anbieter wie SAP zu verlassen, wählen Unternehmen die besten technologischen Lösungen von mehreren Anbietern aus. Diese Nachfrage nach Lösungen verschiedener Anbieter macht Integration wichtiger denn je. Denn *selbst* die besten Plattformen können ihr volles Potenzial nur dann entfalten, wenn alle Lösungen nahtlos zusammenarbeiten.

Zwei der in großen Unternehmen am häufigsten verwendeten Lösungen sind Salesforce für Customer Relationship Management (CRM) und SAP für Enterprise Resource Planning (ERP). In diesem Artikel besprechen wir die Integration von Salesforce und SAP ECC 6.0 oder SAP S/4HANA, untersuchen mögliche Integrationsoptionen und veranschaulichen anschließend ein reales Beispiel einer erfolgreichen Integration.

## Vorteile der Integration von Salesforce mit ERP-Systemen wie SAP

Viele Prozesse und Lösungen in Salesforce, wie Sales Cloud oder Service Cloud, sind oft unvollständig, ohne zusätzliche Informationen aus anderen Systemen. Ohne Einblick in diese Daten sind Benutzer gezwungen, zwischen Systemen zu wechseln, um sich einen vollständigen Überblick zu verschaffen – ein wenig benutzerfreundlicher Ansatz, der die Produktivität der Mitarbeiter beeinträchtigen und zu kostspieligen Fehlern führen kann.

Im Gegensatz dazu führt die Integration von Salesforce mit SAP zu folgenden Vorteilen:

1. **Verbesserte Datenqualität:** Die Integration der beiden Systeme ermöglicht einen nahtlosen Datenaustausch zwischen ihnen, was dazu beitragen kann, die Datengenauigkeit zu verbessern und Datenduplizierungen zu vermeiden.
2. **Erhöhte Effizienz:** Die Integration trägt dazu bei, Geschäftsprozesse zu straffen, die Dateneingabe zu automatisieren und manuelle Fehler zu reduzieren, was zu einer höheren betrieblichen Effizienz führt.
3. **Verbesserte Sichtbarkeit:** Sobald Anwender einen besseren Überblick über ihre Kunden haben, einschließlich Daten zu Vertrieb, Marketing, Kundenservice und Abrechnung, können sie bessere Entscheidungen treffen und das gesamte Kundenerlebnis verbessern.

4. **Verbesserte Zusammenarbeit:** Die Integration ermöglicht eine stärkere Zusammenarbeit zwischen Ihren Vertriebs-, Marketing- und Kundendienstteams und bietet ihnen Echtzeitzugriff auf dieselben wichtigen Kundeninformationen.
5. **Reduzierte Kosten:** In einer integrierten Umgebung sparen Benutzer Zeit und Geld, indem sie die Dateneingabe automatisieren und manuelle Fehler reduzieren. Darüber hinaus ermöglicht ein effizienterer Verkaufsprozess, dass Vertriebsmitarbeiter mehr Zeit für den Verkauf und weniger Zeit für Verwaltungsaufgaben aufwenden können.
6. **Erhöhter Umsatz:** Durch die Verbesserung der Kundensichtbarkeit auf eine Weise, die das Kundenerlebnis verbessert, kann eine solide Integration letztendlich den Umsatz steigern und das Geschäftswachstum vorantreiben.

Innerhalb von Salesforce bietet Customer 360 ein gutes Beispiel für eine verbesserte Sichtbarkeit des Kundendaten. Innerhalb der Accountdetails kann ein Salesforce-Benutzer auf Kontakte, Opportunities, Aktivitäten und vieles mehr zugreifen. Kritische Informationen wie Kreditstatus, Zahlungshistorie, Rechnungen, Lieferungen und Attachments sind jedoch nur in SAP verfügbar. Hier kann eine gut konzipierte Salesforce-Integration eine entscheidende Rolle spielen.

# Vorbereitung des Integrationsprojekts

## Integrationsoptionen

Es gibt verschiedene Möglichkeiten, wie eine Integration umgesetzt werden kann. Sie können in drei grundlegende Kategorien eingeteilt werden.

### Kundenspezifische Entwicklung (Level 1-Integration)

Ein kundenspezifisches Entwicklungsprojekt könnte ohne die Hilfe einer Integrationsplattform abgeschlossen werden. Auf der Salesforce-Seite können hierzu die folgenden APIs verwendet werden:

- REST-API
- SOAP-API
- Streaming-API
- Web Service-Callouts
- Salesforce Connect

Wenn Sie eine direkte Integration von Grund auf erstellen, müssen Sie über Ihre Salesforce- und SAP-Kernlizenzen hinaus keine zusätzlichen Lizenzen oder Subscriptions bezahlen. Allerdings erfordert die kundenspezifische Entwicklung den höchsten Implementierungsaufwand und birgt das größte Projektrisiko. Außerdem kann Salesforce nicht direkt in SAP-Standards wie RFC (Remote Function Call) integriert werden, was Ihre Integrationsmöglichkeiten einschränkt.

Wenn Sie mehr über die benutzerdefinierte Entwicklungsoption erfahren möchten, vergessen Sie nicht, die [Ultimate Introduction to Salesforce Integration](#) zu lesen.

### IPaaS-Lösungen (Level-2-Integration)

Die Integration über ein IPaaS (Integration Platform as a Service), die nächste Integrationskategorie, bietet einige Vorteile gegenüber der kundenspezifischen Entwicklung. Insbesondere werden IPaaS-Lösungen oft mit Adaptern für SAP geliefert, die Sie nicht von Grund auf neu erstellen müssten. Die IPaaS kümmert sich um wichtige Aspekte wie Connection Handling, Authentifizierung und Sicherheit.

Die folgende Liste enthält einige der am häufigsten verwendeten iPaaS-Lösungen für die Integration von SAP in Salesforce:

- **MuleSoft** ist das iPaaS von Salesforce. Es hat seine Stärke auf der Salesforce-Seite und bietet Vorlagen für die SAP-Integration.
- **SAP CPI** (Cloud Platform Integration) ist das Angebot von SAP. Es unterstützt alle SAP-Protokolle und enthält Vorlagen für Salesforce-Integrationen.
- **Jitterbit, Dell Boomi, IBM App Connect** und viele andere Plattformen können ebenfalls für Ihre Integration verwendet werden, sind jedoch weniger verbreitet.

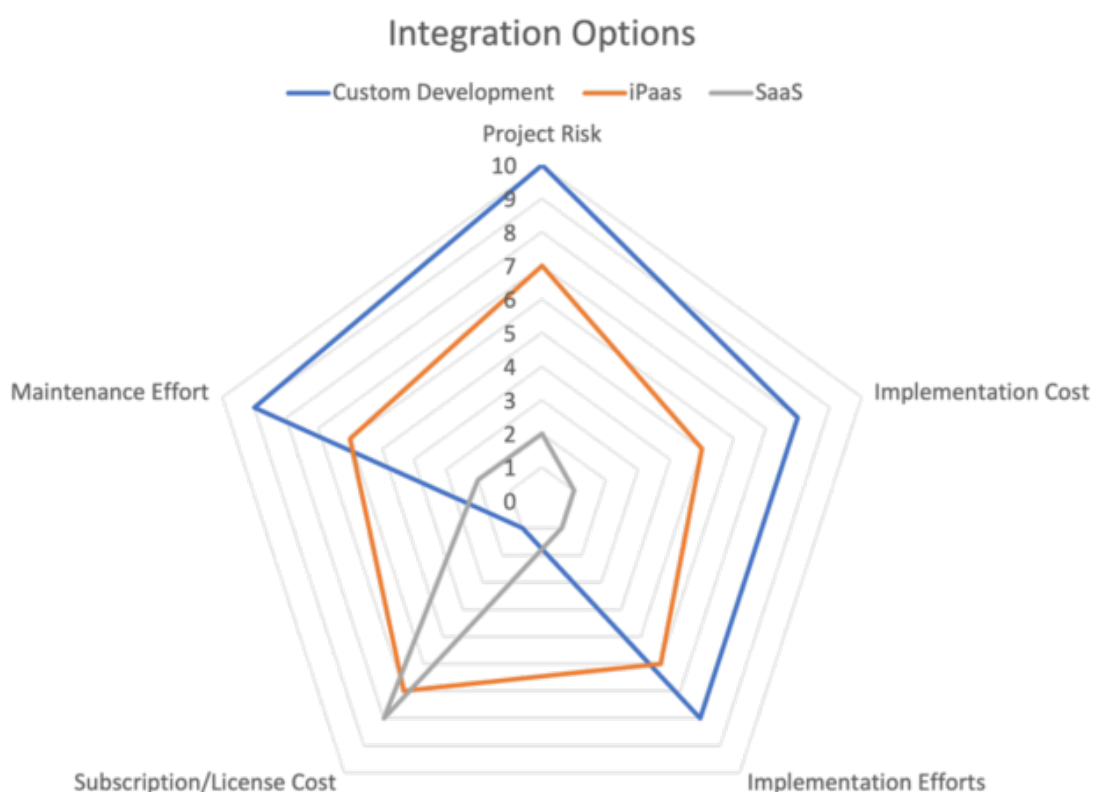
### SaaS mit gebrauchsfertigen Integrationen (Level-3-Integration)

SaaS-Lösungen (Software as a Service) bieten mehr als nur eine Plattform zum Erstellen von Integrationen – sie stellen auch vordefinierte Anwendungsinhalte bereit, die sofort verwendet werden können. Dies bietet den entscheidenden Vorteil, dass Zeit und Kosten für die Durchführung eines langen, riskanten Integrationsprojekts gespart werden. Stattdessen können Sie die Integrationen installieren und in der Regel innerhalb weniger Wochen in Betrieb nehmen. Durch eine Lizenzgebühr oder ein Abonnement erhalten Sie nicht nur Zugriff auf die Lösung, sondern auch auf Support und Produktaktualisierungen.

Zwei solche SaaS-Lösungen sind [Enosix](#) und [Vigience Overcast](#). Sie bieten vordefinierte Integrationen in SAP, einschließlich der Benutzeroberflächen auf der Salesforce-Seite. Overcast verfügt über über 150 Integrationen in SAP und andere ERP-Systeme wie Microsoft Dynamics und Oracle NetSuite. Zusätzlich zu den sofort einsatzbereiten Funktionen ermöglicht Ihnen Overcast die Erstellung eigener Integrationen.

Bei der Auswahl Ihrer Integrationsart ist noch ein weiterer Faktor zu berücksichtigen – die laufende Wartung Ihrer Lösung, die (nach dem Go-Live) auf Stufe 1 (kundenspezifische Entwicklung) am höchsten und auf Stufe 3 (SaaS wie Enosix oder Overcast) am niedrigsten ist. Integrationen müssen ständig getestet und an neue Salesforce- und SAP-Releases angepasst werden.

Das folgende Spinnendiagramm zeigt den Vergleich der drei Optionen (der niedrigere Wert ist besser):



# Spezifikationen

Wie bei jedem erfolgreichen Softwareprojekt sollten gute Spezifikationen der Ausgangspunkt sein. Für ein Integrationsprojekt sollten einige wichtige Dinge im Vorfeld dokumentiert werden, damit Ihr Projekt ein Erfolg wird.

## Was muss integriert werden?

SAP bietet eine schier endlose Anzahl an Geschäftsprozessen, Objekten und Daten. Es ist wichtig, die richtigen Informationen auszuwählen. Um herauszufinden, was Ihre Benutzer benötigen, können Sie sie entweder interviewen oder, *noch* besser, ihre Prozesse und Aktivitäten beobachten. Benutzer vergessen oft, Dinge zu erwähnen, die sie für offensichtlich halten, aber wenn Sie mit ihnen zusammensitzen und beobachten, wie sie tatsächlich arbeiten, wird dies nicht passieren.

Dokumentieren Sie unbedingt, wie Ihre Benutzer mit Salesforce und SAP interagieren – und welche Informationen sie in welches System eingeben. Wo werden heute Informationen zwischen Systemen kopiert und eingefügt? Wo würden sie von Echtzeit-Einblicken profitieren?

## Datenreplikationen oder Echtzeit?

Im Allgemeinen gibt es zwei Arten von Integrationen:

1. **Datenreplikation:** Daten werden von SAP nach Salesforce kopiert, entweder in ein Standard- oder ein benutzerdefiniertes Objekt. Typische Anwendungsfälle hierfür sind:
  - SAP-Geschäftspartner zum Salesforce-Accounts
  - SAP-Material zum Salesforce-Product
  - SAP-Listenpreise zu Salesforce-PriceBooks
2. **Echtzeitzugriff :** Daten werden nicht in Salesforce gespeichert, sondern bei Bedarf von SAP angefordert. Oft ist dies die einzig zuverlässige Möglichkeit der Integration, da (1) die Daten zeitkritisch sind und sich häufig ändern, (2) das Datenvolumen zu groß ist, um alles zu kopieren, oder (3) die Geschäftslogik in SAP sehr komplex ist und nicht in Salesforce neu erstellt werden kann. Zu den Anwendungsfällen für Echtzeitintegrationen gehören Preisberechnungen in SAP und Verfügbarkeitsprüfungen für Materialien.

Manchmal kann es auch sinnvoll sein, ein Hybridszenario zu erstellen, bei dem Teile der Daten repliziert und andere in Echtzeit geladen werden. Beispielsweise müssen wir möglicherweise die Ausrüstung aus dem SAP-Kundendienst in die Anlage in Salesforce replizieren, die Merkmale und Messwerte jedoch in Echtzeit laden.

Dokumentieren Sie unbedingt, welche Arten von Integrationsszenarien Sie verwenden und warum. Versuchen Sie als Best Practice, so viel Echtzeitzugriff wie möglich aufzubauen, da dies auf lange Sicht die Dinge erleichtert.

## Zuordnung der Datenmodelle zwischen Salesforce und SAP

Bei der Dokumentation Ihrer Integration ist die Feldzuordnung ein wichtiger Aspekt.

- Für die Datenreplikation müssen Sie wissen, welches Feld aus SAP in welchem Feld in Salesforce gespeichert werden soll. Sie müssen Fragen stellen wie:
  - Sind die Feldtypen und -längen identisch oder muss ich die Daten konvertieren?
  - In welche Richtung möchte ich replizieren?
  - Ist meine Replikation nur in eine Richtung (SAP zu Salesforce oder Salesforce zu SAP) oder bidirektional?

- Für den Echtzeitzugriff müssen Sie definieren, welche Informationen aus SAP gelesen werden müssen und wie diese dem Benutzer angezeigt werden sollen.

Denken Sie daran, dass dies dokumentiert werden *muss* – wir verwenden zu diesem Zweck oft eine Excel-Tabelle. So könnte das Mapping aussehen:

Salesforce Object		Log Prefix						
Account		KNA1-						
Table	Field	Example	Direction	Layout	Object	Field	Salesforce	Fixed Value
			-		Account	RecordTypeId		0123X000000eNy4QAE
KNA1	Customer Number		→		Account	Kna1Kunnr__c		
KNA1	Name1		→		Account	Name		
KNA1	Name2		→		Account	Name		
KNA1	City		→		Account	BillingCity		
KNA1	Post Code		→		Account	BillingPostalCode		
KNA1	Country		→		Account	BillingCountryCode		
KNA1	Sort		-					
KNA1	Street		→		Account	BillingStreet		
KNA1	Phone		→		Account	Phone		
KNA1	Fax		→		Account	Fax		
KNA1	Industry Code		→		Account	Industry__c		
KNA1	Mail		→		Account	EmailforInvoicing__c		
KNA1	KNA1-LOEVM		→		Account	MarkedForDeletionSAP__c		

*Beispiel einer Feldzuordnung in MS Excel*

## Filter

Nicht alle Daten aus SAP müssen in Salesforce repliziert oder angezeigt werden. Tatsächlich ist es oft genau das Gegenteil: Je weniger repliziert und angezeigt wird, desto wahrscheinlicher ist es, dass Sie eine gute Performance und eine benutzerfreundliche Benutzeroberfläche erreichen.

Stellen Sie sicher, dass Sie dokumentieren, wie die Daten gefiltert werden sollen. Benötigen Sie beispielsweise nur Accounts in Salesforce, die in SAP aktiv sind, oder auch die inaktiven? Welches SAP-Feld enthält diese Schlüsselinformation?

## Die SAP-Seite – Verfügbare APIs

Für ein erfolgreiches Projekt benötigen Sie SAP-Know-how in Ihrem Projektteam, um den besten Integrationsweg zu ermitteln. SAP bietet verschiedene Arten von APIs an:

- **RFCs/BAPIs** : BAPIs (Business Application Programming Interfaces) werden über RFCs (Remote Function Calls) aufgerufen. RFC ist ein proprietäres Protokoll, das von SAP implementiert wird. Sowohl SAP ECC als auch SAP S/4HANA bieten Zehntausende BAPIs für alle relevanten Geschäftsprozesse zum Lesen und Schreiben von Daten. Um eine Verbindung zu einem BAPI herzustellen, muss einer der SAP-Konnektoren verwendet werden – SAP JCo (Java Connector) oder NCo (.Net Connector). Es besteht auch die Möglichkeit, Daten aus beliebigen **SAP- Datenbanktabellen** per RFC auszulesen.
- **SOAP-WebServices**: SAP bietet mehrere Webservices an, die auf dem SOAP-Standard basieren, aber sie decken nur einen kleinen Teil der gesamten SAP-Funktionalität ab.
- **OData**: Basierend auf dem OData-Protokoll hat SAP eine neue Schicht von Web Services erstellt, die leichtgewichtiger sind als BAPIs und SOAP. Die OData-Unterstützung ist bisher noch nicht so umfassend wie die der BAPIs, aber dies ist die Zukunft der API von SAP.

Als Best Practice ist es immer empfehlenswert, über eine Standard-SAP-API zu integrieren, ohne Code im SAP-System zu schreiben. Die Standard-APIs haben den Vorteil, dass sie die SAP-Konfiguration und -Customizing berücksichtigen. Das Schreiben eigener APIs in SAP würde auch zu einem Projekt auf SAP-Seite führen – eine potenziell komplexe und zeitaufwändige Aufgabe. APIs müssen in den Release- und Update-Zyklus von SAP passen, was Ihr Projekt weiter verzögern könnte.

*Auch* ohne eine Implementierung auf SAP-Seite ist in jedem Integrationsprojekt tiefes SAP-Know-how erforderlich, einschließlich Kenntnisse über Geschäftsprozesse und verfügbare APIs.

# Umsetzung einer Integration am Beispiel von Vigilance Overcast

Lassen Sie uns anhand des folgenden Beispiels veranschaulichen, wie eine Integration aussehen kann – die Anzeige der Rechnungen eines Accounts auf der Related Lists der Detailseite des Accounts. Da diese Rechnungen in SAP gespeichert sind, müssen wir eine Integration implementieren.

Innerhalb von SAP haben wir Millionen bestehender Rechnungen und täglich werden unzählige neue erstellt. Sie alle zu replizieren würde viel Speicherplatz auf der Salesforce-Seite und eine hohe Belastung des SAP-Systems erfordern. Daher möchten wir Rechnungen in Echtzeit sehen und diese nur bei Bedarf anzeigen.

Vigience Overcast bietet die ideale Lösung für diese Integration. Tatsächlich bietet Overcast bereits eine sofort einsatzbereite Komponente zur Anzeige von Rechnungen an, die wir problemlos aus der Komponentenbibliothek installieren könnten. Für dieses Tutorial schauen wir uns aber an, wie wir dieselbe Komponente von Grund auf erstellen würden.

**SAP Invoice History**  
Display invoice history and drill-into order details & related document flows for an Account

**SAP Invoices (6+)**

**Invoice (F2) 90040408**  
Billing Cat.: Delivery-related billing document  
Net Value: €260,250.00  
Billing Date: Oct 11, 2018

**Credit Memo 90040406**  
Billing Cat.: Order-related billing document  
Net Value: €60.00  
Billing Date: Mar 3, 2018

**Invoice Cancellation 90040404**  
Billing Cat.: Order-related billing document  
Net Value: €200.00

[Customize](#) [Details](#)

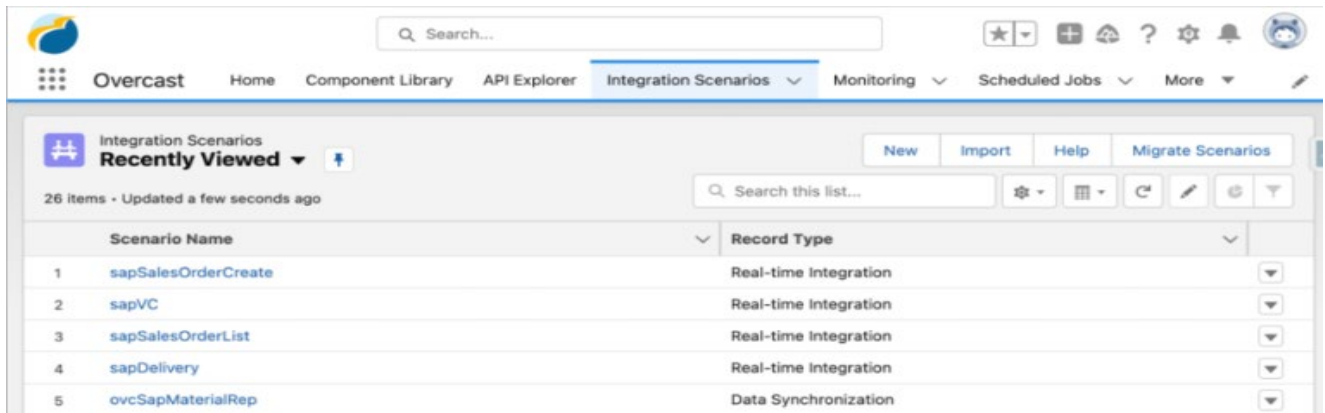
*Overcast-Komponente, die Rechnungen eines Accounts in Echtzeit anzeigt*

Overcast wird über AppExchange als Managed Package in Ihrer Salesforce-Org installiert. Die Installation und Konfiguration wird in diesem Artikel nicht behandelt, aber Sie können mehr darüber erfahren, indem Sie [die Overcast-Hilfe](#) lesen.

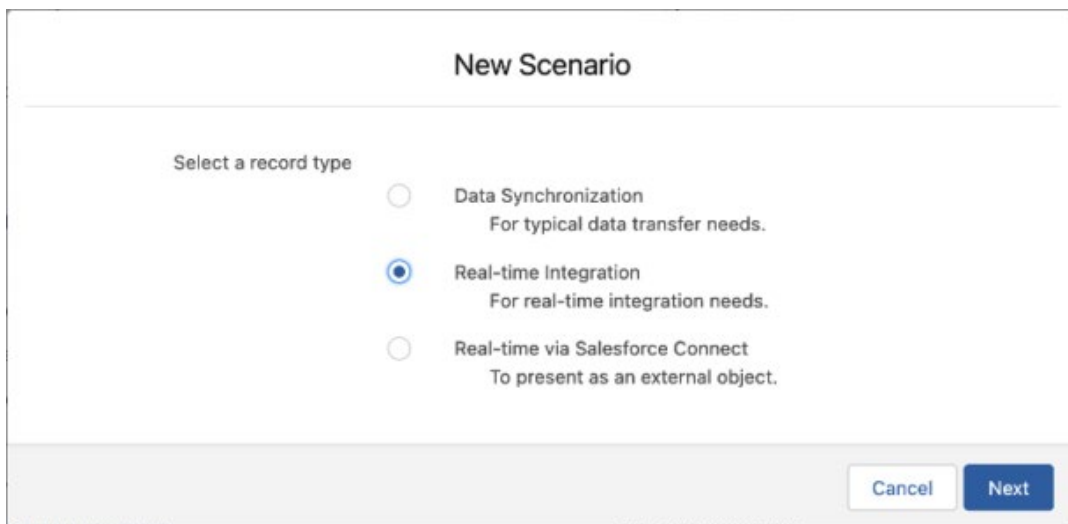
# Erstellen eines Integrationsszenarios

Lassen Sie uns zunächst ein Integrationsszenario erstellen, das auf eine bestimmte API unseres Backend-Systems verweist. Gehen Sie dazu folgendermaßen vor:

1. Öffnen Sie den App-Launcher und wählen Sie „Overcast“.
2. Gehen Sie zur Registerkarte „Integrationsszenarien“ und wählen Sie „Neu“.



3. Wählen Sie im Popup den Typ des Szenarios aus – „Datensynchronisation“ oder „Echtzeit“. In diesem Fall werden wir uns für eine „Echtzeitintegration“ entscheiden.



4. Als nächstes müssen Sie die Verbindung auswählen, über die auf SAP zugegriffen wird, und dem Szenario einen eindeutigen Namen geben.

5. Im letzten Schritt können wir definieren, auf welches BAPI bzw. welche Tabelle in SAP wir zugreifen möchten – in unserem Fall ist es die Tabelle WB2\_V\_VBRK\_VBRP2. Wie Sie sehen, sind BAPI- und Tabellennamen in SAP oft kryptisch, sodass häufig ein SAP-Experte zur Unterstützung des Teams erforderlich ist. Feldnamen können eine ähnliche Herausforderung darstellen, aber Overcast ermöglicht es Ihnen, genau anzugeben, welche Felder aus SAP von der Integration verwendet werden sollen. Sie können sie im selben Dialog auswählen.

Overcast Home Component Library API Explorer Integration Scenarios Monitoring Scheduled Jobs Connection Strings Settings

Scenario Edit **SFBenIntegration** Help for this Page ?

**Information**

Record Type: Real-time Integration  
 Scenario Name: SFBenIntegration  
 Description: Integration with SAP Invoices  
 Connection String: SAP\_S4H

**SAP Object Info**

Please select a SAP BAPI, RFC or Table to integrate with.  
 In case data needs to be updated in the SAP system, please check Commit Work.  
 Re-activate the Scenario when changing this section.

SAP BAPI / RFC  SAP Table  Search

Commit Work

Fields: 9 fields selected Update selection

NAME	DESCRIPTION	TYPE NAME	LENGTH
AUBEL_J	Sales Document	CHAR	10
AUPOS_J	Sales Document Item	NUM	6
AUREF_J	Sales document was created from reference	CHAR	1
ERNAM	Created by	CHAR	12
KNUMA	Agreement	CHAR	10
KUNRG	Payer	CHAR	10
MATNR_J	Material Number	CHAR	40
STCEG	VAT Reg. No.	CHAR	20
VBELN	Sales document	CHAR	10

**Advanced Settings**

Use in Flow   
 Log Level: Global Default  
 Don't Re-Generate Integration Objects   
 Timeout (ms)

Save & Activate Cancel

6. Wählen Sie nun die Schaltfläche „Speichern & Aktivieren“. Dadurch wird das Integrationsszenario gespeichert und im Hintergrund alle erforderlichen APEX-Klassen und -Strukturen erstellt.

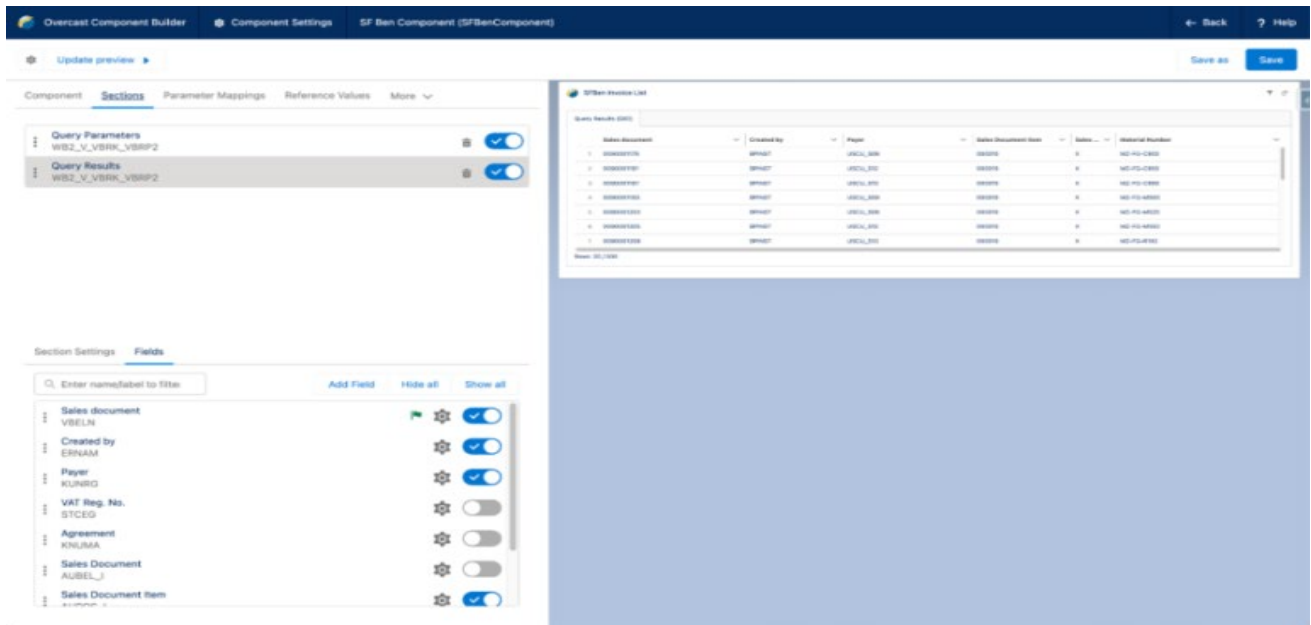
Wir haben nun unsere Salesforce-Org mit SAP integriert, um eine Rechnungsliste zu erhalten, ohne eine Zeile Code schreiben oder Salesforce verlassen zu müssen.



# Der Overcast Component Builder

Mit dem Overcast Component Builder kann ein Administrator die Benutzeroberfläche der Integration in Form einer Lightning-Komponente erstellen. Der Component Builder bietet zunächst verschiedene Muster, wie Widgets, Listenansichten, Detailseiten und vieles mehr. Es handelt sich um ein Low-Code-Tool, das vollständig auf Metadaten basiert und eine Vorschau der Benutzeroberfläche bietet, während wir sie erstellen.

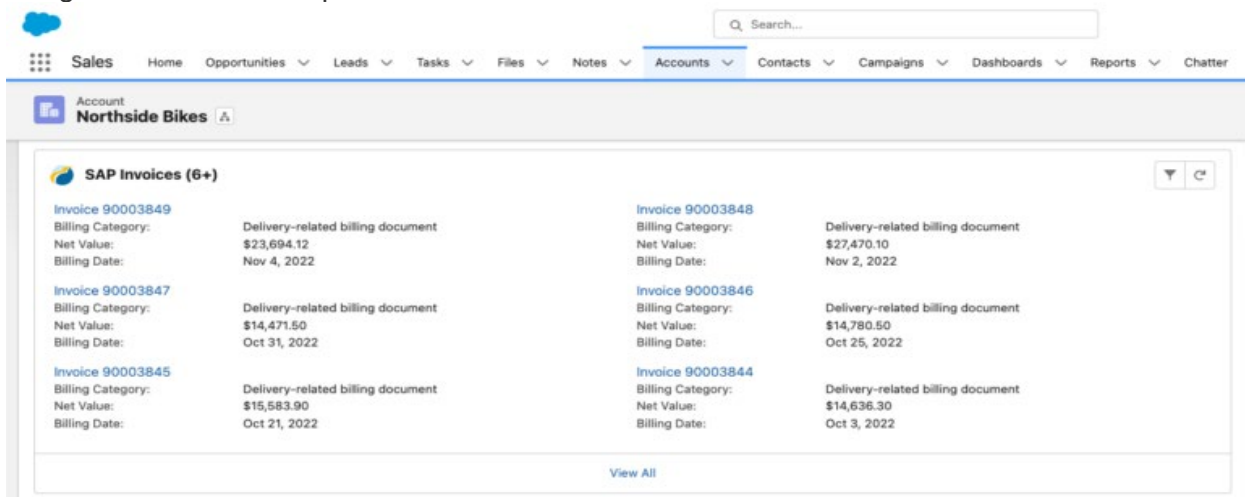
Der folgende Screenshot zeigt den Component Builder mit der Liste der Rechnungen:



Im Component Builder können wir die Eingabeparameter der Integration mit Attributen eines zugehörigen Objekts verknüpfen. In unserem Beispiel verknüpfen wir das SAP-Feld KUNRG (Payer) mit dem benutzerdefinierten Account-Feld „Geschäftspartnernummer“ im Salesforce-Account Objekt, wodurch sichergestellt wird, dass nur Rechnungen dieses Accounts angezeigt werden.

Genau wie normale Lightning-Komponenten können Overcast-Komponenten mit dem Salesforce AppBuilder in Seitenlayouts eingebettet werden.

Die gerade erstellte Komponente sieht so aus:



# Zusammenfassung

Die Integration von Salesforce in andere Aufzeichnungssysteme wie SAP wird immer wichtiger – *insbesondere* für große Unternehmen, die Salesforce-Benutzern das volle Potenzial von Salesforce und SAP bieten möchten. Dank durchdachter Integrationen müssen Benutzer nicht mehr zwischen Systemen wechseln oder Daten zweimal eingeben.

Damit Integrationen erfolgreich sind, ist es wichtig, zunächst das richtige Tool auszuwählen und im Vorfeld detaillierte Spezifikationen zu verfassen. SAP-Kenntnisse in Ihrem Team können von entscheidender Bedeutung sein, *insbesondere* für die Auswahl der richtigen APIs und der Analyse aller potenziellen Auswirkungen der Integration auf Ihr SAP-System.

Welche Erfahrungen haben Sie mit der Integration von Salesforce mit SAP gemacht? Lass es uns unten in den Kommentaren wissen!