

# SAP S/4HANA - Architektur

» Hier geht's  
direkt  
zum Buch

# DIE LESEPROBE

# Kapitel 6

## Integration

*Dieses Kapitel gibt einen Überblick über die Technologien, die zur Integration von SAP S/4HANA mit anderen Anwendungen und Services verwendet werden können. Dies beinhaltet Schnittstellentechnologien, die Kommunikation zwischen SAP S/4HANA Cloud und On Premise, Integrations-Middleware, die Event-basierte Integration und die Datenintegration. Dabei werden wir auch das vereinfachte Kommunikationsmanagement in SAP S/4HANA Cloud besprechen.*

Softwareanwendungen und insbesondere ERP-Anwendungen sind Teil einer vernetzten Welt. Die Fähigkeit, ERP-Systeme in Geschäftsprozesse zu integrieren, die anwendungsübergreifend mit mehreren Softwareanwendungen arbeiten und sogar verschiedenen Organisationen gehören können, stellt eine der grundlegenden Herausforderungen in diesem Zusammenhang dar.

In diesem Kapitel geben wir Ihnen in Abschnitt 6.1 zunächst einen Überblick über die in SAP S/4HANA verfügbaren Schnittstellentechnologien. Anschließend stellen wir in Abschnitt 6.2 den SAP Business Accelerator Hub vor, das öffentliche Verzeichnis der freigegebenen Application Programming Interfaces (APIs) von SAP. Als Nächstes behandeln wir in Abschnitt 6.3 das SAP Application Interface Framework als zentrale Infrastruktur für das Schnittstellen-Monitoring und die Fehlerbehandlung. Abschnitt 6.4 deckt das Kommunikationsmanagement in der Cloud und Abschnitt 6.5 die Kommunikation zwischen Cloud- und On-Premise-Systemen über den Cloud Connector ab. Schließlich werden wir verschiedene Arten der Integration erläutern, nämlich Prozessintegration (siehe Abschnitt 6.6), Event-basierte Integration (siehe Abschnitt 6.7) und Datenintegration (siehe Abschnitt 6.8).

### 6.1 Schnittstellentechnologien zur Integration mit SAP S/4HANA

Dieser Abschnitt gibt einen kurzen Überblick über die Integrationstechnologien und erläutert die Strategie in Bezug auf die Schnittstellen in SAP S/4HANA.

### 6.1.1 Überblick über die Schnittstellentechnologien

In SAP S/4HANA sind alle Integrationstechnologien der ABAP-Plattform verfügbar, wie z. B. SOAP-Services, OData-Services, Business Events, IDocs, BAPIs und RFCs. Die folgende Liste gibt Ihnen einen Überblick über diese bekannten Integrationstechnologien und ihre Rolle in SAP S/4HANA:

- **OData-Services (Open Data Protocol)**

Mit *OData-APIs* können Sie Datenmodelle abfragen und ändern, die aus Entitäten bestehen, die über Navigationsbeziehungen miteinander verbunden sein können. OData-Services stellen ihre Metadaten im Format des *Entity Data Model* (EDM) zur Verfügung. OData-Entitäten und ihre Elemente können über Annotationen mit zusätzlichen Metadaten versehen werden. Wie Sie OData-Services mit dem ABAP RESTful Application Programming Model definieren können, haben wir bereits in Abschnitt 2.2.2 beschrieben.

Die OData-V2-Services in SAP S/4HANA sind synchrone Services, bei denen das Ergebnis einer Operation sofort im Body der HTTP-Antwort zurückgegeben wird. Wenn der Client beispielsweise eine POST-Anfrage sendet, um die Instanz einer Entität zu erstellen, erstellt der Service diese neue Instanz auf dem Server und gibt dann deren Repräsentation mit der Antwort zurück. OData-V4-Services unterstützen auch asynchrone Antworten. Wenn der Aufrufer angibt, dass er eine asynchrone Antwort bevorzugt, gibt SAP S/4HANA als Status zurück, dass die Anfrage angenommen wurde. Gleichzeitig gibt SAP S/4HANA eine URL zurück, unter der der Aufrufer den Stand der Verarbeitung abfragen und schließlich das Ergebnis abrufen kann. Neue OData-Services werden von SAP als V4-Services definiert.

- **SOAP-Services (Simple Object Access Protocol)**

SOAP-Services sind Services, die HTTP und das XML-basierte SOAP-Nachrichtenformat verwenden. Die von einem SOAP-Service bereitgestellten Operationen und das Schema für Eingabe- und Ausgabedaten werden mithilfe der Web Services Description Language (WSDL) spezifiziert. In der ABAP-Umgebung können SOAP-Services im Enterprise Services Repository oder im Backend Repository des ABAP-Anwendungsservers definiert werden. Für benutzerdefinierte Services gibt es auch die Möglichkeit, SOAP-Services aus bestehenden ABAP-Funktionsgruppen und BAPIs zu generieren.

In SAP S/4HANA werden SOAP-Services typischerweise für die asynchrone, verlässliche und transaktionale Kommunikation mit unidirektionalen Nachrichten ohne Antwort verwendet. Denken Sie beispielsweise an eine Beschaffungsanwendung, die eine Bestellung als asynchrone Nachricht an ein Lieferantensystem sendet. Die Beschaffungsanwendung sendet die Nachricht, ohne auf eine Antwort zu warten, und diese wird durch die Infrastruktur für verlässliche Nachrichtenkommunikation (*Reliable Messaging*) zugestellt, möglicherweise zu einem späteren Zeitpunkt. Wenn das Lieferantensystem eine Auftragsbestätigung senden möchte,

sendet es diese mit einer weiteren unidirektionalen Nachricht über einen asynchronen Service, der von der Beschaffungsanwendung bereitgestellt wird. Synchrone SOAP-Services kommen nur in Ausnahmefällen zum Einsatz.

- **Business Events (Geschäftsereignisse)**

Business Events können verwendet werden, um andere Anwendungen über die Erstellung, Löschung oder Änderung von Business-Objekten zu informieren. Bei Events handelt es sich um asynchrone Nachrichten, die von einer Quellenanwendung veröffentlicht werden und von mehreren konsumierenden Anwendungen empfangen werden können. Die Empfängeranwendungen müssen der Quellenanwendung nicht bekannt sein. Stattdessen veröffentlicht die Quellenanwendung Events, und die Empfänger können sich auf diese subscribieren. Bei der Event-basierten Integration mit anderen Systemen dient SAP Event Mesh als Infrastruktur zum Zwischenspeichern und Ausliefern der Events. Die Einzelheiten zur Event-basierenden Integration werden wir in Abschnitt 6.7 besprechen.

- **Remote Function Calls (RFC)**

Der klassische Mechanismus für die Remote-Kommunikation ist der *Remote Function Call* (kurz RFC), das SAP-Protokoll für den Aufruf eines remotefähigen ABAP-Funktionsbausteins. Alternativ zur Kommunikation über das native RFC-Protokoll können RFCs über WebSocket ausgeführt werden, insbesondere wenn nur HTTP-Konnektivität zur Verfügung steht. Es gibt verschiedene Varianten von RFCs. Der *synchrone RFC* wird sofort ausgeführt, wenn er aufgerufen wird, und das Ergebnis wird sofort zurückgegeben. Der *Background RFC* und kann für eine verlässliche asynchrone Kommunikation verwendet werden. Einzelheiten hierzu finden Sie in der SAP-Dokumentation für ABAP unter <https://help.sap.com>.

- **Business Application Programming Interfaces (BAPIs)**

Bei *Business Application Programming Interfaces* (BAPIs) handelt es sich um stabile, dokumentierte Schnittstellen, die als Methoden von Objekttypen definiert sind, die im Business Object Repository (BOR) aufgeführt sind. Technisch gesehen werden BAPIs als RFCs implementiert. BAPIs werden typischerweise als synchrone RFCs aufgerufen, sodass Aufrufer die Ergebnisse sofort erhalten.

- **Intermediate Documents (IDocs)**

Ein *Intermediate Document* (IDoc) ist ein SAP-Standardformat für den asynchronen Austausch von Nachrichten zwischen Systemen. IDocs definieren die ausgetauschten Daten unabhängig von der Methode der Datenübertragung. IDocs können über asynchrone RFCs, über HTTP im XML-Format, oder über HTTP als SOAP-Nachrichten übertragen werden. IDocs werden beim Electronic Data Interchange (EDI) zwischen Organisationen und für asynchrone Nachrichten in einem ALE-Szenario (Application Link Enabling) innerhalb einer Organisation verwendet. ALE verwendet IDocs für die asynchrone Kommunikation zur Änderung von Daten und für synchrone BAPI-Aufrufe zum Lesen von Daten.

### 6.1.2 Die API-Strategie von SAP S/4HANA

Während klassische Schnittstellen wie RFC, BAPI und IDoc in SAP S/4HANA weiterhin unterstützt werden, konzentriert sich SAP bei Neuentwicklungen auf SOAP- und OData-APIs. SOAP-Services werden für asynchrone Nachrichten (den früheren Anwendungsbereich von IDocs) verwendet, und OData-Services werden für synchrone Operationen auf Ressourcen (den früheren Anwendungsbereich von BAPIs) verwendet.

SAP hat diese Strategie für SAP S/4HANA Cloud bereits umgesetzt. Hier sind SOAP- und OData-Services bereits als die bevorzugten API-Technologien etabliert, und gemäß der Strategie von SAP werden keine neuen BAPIs, RFCs und IDocs für die Cloud implementiert. Für SAP S/4HANA Cloud beschränkt SAP den Einsatz klassischer Schnittstellentechnologien (IDoc, BAPI, RFC) auf ausgewählte Szenarien, damit SAP-Kunden die Möglichkeit haben, ihre bestehenden und bereits implementierten Integrationsszenarien weiter zu nutzen. In diesen Szenarien wird die RFC-Kommunikation zwischen Cloud- und On-Premise-Systemen in beide Richtungen durch den Cloud Connector ermöglicht (siehe Abschnitt 6.5).

## 6.2 SAP Business Accelerator Hub

Der SAP Business Accelerator Hub (vormals SAP API Business Hub) ist das SAP-Verzeichnis für öffentliche APIs, Business Events und verschiedene andere Arten von Inhalten wie etwa Integrations-Content, CDS-Views (Core Data Services) und Referenzgeschäftsprozesse. Der SAP Business Accelerator Hub ist verfügbar unter <https://api.sap.com>.

Im Hinblick auf die Integration mit SAP S/4HANA ist der folgende Inhalt besonders interessant:

- der Katalog der öffentlichen APIs von SAP S/4HANA (SOAP und OData)
- vordefinierter Integrations-Content für die Cloud-Integration-Funktion in der SAP Integration Suite
- der Katalog der Business Events in SAP S/4HANA Cloud, die über SAP Event Mesh konsumiert werden können
- die Business Add-Ins (BAIs), die SAP als Erweiterungspunkte für die Entwicklererweiterbarkeit von SAP S/4HANA Cloud freigegeben hat
- die freigegebenen Schnittstellen für Business-Objekte, die Sie bei der Entwicklererweiterbarkeit von SAP S/4HANA Cloud verwenden oder aufrufen können



### Öffentliche und nichtöffentliche APIs

APIs, die nicht im SAP Business Accelerator Hub aufgeführt sind, sind von SAP nicht als öffentliche APIs freigegeben. Ein Beispiel für nichtöffentliche APIs sind lokale Services, die nur von der Benutzeroberfläche einer SAP-S/4HANA-Anwendung aus aufgerufen werden. Solche Services sind nicht im SAP Business Accelerator Hub gelistet, es gibt keinen Kompatibilitätskontrakt für sie, und sie können von SAP jederzeit geändert oder entfernt werden.

Für SOAP-APIs finden Sie eine Online-Referenzdokumentation und können die WSDL-Definition herunterladen. Für OData-APIs können Sie die API-Spezifikation online lesen oder in den Formaten OpenAPI und EDM herunterladen. Sie finden hier z. B. auch Informationen über unterstützte Authentifizierungsmethoden, über Scope Items, die für die Verwendung der API aktiviert werden müssen, sowie Links zur Dokumentation im SAP Help Portal.

Im SAP Business Accelerator Hub können Sie OData-APIs ausprobieren, wobei die Anfragen an eine von SAP bereitgestellte öffentliche Instanz von SAP S/4HANA Cloud geschickt werden. Sie können die APIs sogar mit Ihrer eigenen Instanz von SAP S/4HANA Cloud ausprobieren, nachdem Sie die erforderlichen Anmeldedaten im SAP Business Accelerator Hub konfiguriert haben. Freigegebene BAPIs und IDocs werden nicht im SAP Business Accelerator Hub veröffentlicht.

Im SAP Business Accelerator Hub finden Sie eine **Feedback**-Schaltfläche, über die Sie die Customer Influence Session für SAP S/4HANA zum Thema Integration unter <https://influence.sap.com/sap/ino/#/campaign/1175> aufrufen können. Dort können Sie neue Remote-APIs (oder Business Events) oder die Erweiterung bestehender APIs anfordern oder laufende Anforderungen anzeigen.

## 6.3 Schnittstellen-Monitoring und Fehlerbehandlung

In diesem Abschnitt erläutern wir die Kernkonzepte des *SAP Application Interface Framework*, einer Infrastruktur zum Monitoring von Integrationsschnittstellen, zur Fehlerbehandlung und zur Nachrichtenverarbeitung in ABAP-basierten Systemen. In der Public Edition von SAP S/4HANA Cloud sind alle asynchronen APIs und Business Events des ABAP-Backends zum Zweck der Überwachung und Fehlerbehandlung in das SAP Application Interface Framework integriert. Das bedeutet, dass SAP für diese Schnittstellen bereits die notwendige Konfiguration für das SAP Application Interface Framework bereitstellt. Bei der On-Premise-Version von SAP S/4HANA und in der Private Edition von SAP S/4HANA Cloud ist dies für die meisten Anwendungen nicht der Fall. In der On-Premise-Version können Sie Schnittstellen jedoch selbst so

konfigurieren, dass sie vom SAP Application Interface Framework überwacht und verarbeitet werden. Außerdem können Sie mit dem Framework Ihre eigenen Schnittstellen entwickeln.

Für die verschiedenen Schnittstellentechnologien des ABAP-Anwendungsservers gibt es unterschiedliche Tools zur Überwachung und Fehlerbehandlung. Diese Tools sind jeweils technologiespezifisch und sehr technisch und deshalb meist nur für Technologieexpert\*innen verständlich. Demgegenüber bietet das SAP Application Interface Framework eine einheitliche Infrastruktur für Monitoring, Alerting und Fehlerbehandlung auf der fachlichen Ebene. Die Benutzer\*innen können beispielsweise Fehler auf inhaltlicher Ebene korrigieren, ohne sich mit der zugrunde liegenden Technologie befassen zu müssen.

Abbildung 6.1 zeigt die konzeptionelle Architektur des SAP Application Interface Framework. Für eingehende und ausgehende asynchrone Aufrufe werden SOAP-Services, IDocs, RFCs und Business Events unterstützt. Für asynchrone ausgehende Aufrufe wird als zusätzliche Schnittstellentechnologie OData unterstützt. Für synchrone OData-Aufrufe ist nur eine Fehlerüberwachung vorgesehen. Eingehende Aufrufe werden von der entsprechenden Integrationsschnittstellenlaufzeit des ABAP-Anwendungsservers wie etwa der Webservicelaufzeit verarbeitet. Wenn das SAP Application Interface Framework für die Verwendung mit einer bestimmten Schnittstelle konfiguriert ist, registriert die Schnittstellenlaufzeit den Aufruf zunächst bei der Monitoring-Komponente des Frameworks. Das Framework ist nun über den Aufruf informiert und kann ihn in der Monitoring-Anwendung anzeigen. Anschließend ruft die Schnittstellenlaufzeit die eigentliche anwendungsspezifische Verarbeitung der Anfrage auf. Die Anwendung verarbeitet den Aufruf und schreibt den Fortschritt und eventuelle Fehler in das Anwendungsprotokoll. Wenn die Verarbeitung abgeschlossen ist oder fehlschlägt, wird die Kontrolle an die Schnittstellenlaufzeit zurückgegeben. Wenn Fehler auftreten, verzeichnet die Schnittstellenlaufzeit die Informationen über diese Fehler in ihrem Protokoll. Dann ruft sie erneut das SAP Application Interface Framework auf, um dieses über den Status des Aufrufs zu informieren. Das Framework aktualisiert den Status in seinen internen Monitoring-Daten und schreibt zusätzliche Fehlermeldungen in das Protokoll. Das Monitoring kann so konfiguriert werden, dass Warnmeldungen ausgelöst werden, um die Benutzer\*innen auf Fehler aufmerksam zu machen.

Auch die Verarbeitung von Business Events und der Verbindungsstatus für die Kommunikation mit Business Events können mit dem SAP Application Interface Framework überwacht werden. Dies gilt sowohl für ausgehende als auch für eingehende Events. Wie Sie in Abschnitt 6.7 erfahren werden, werden Events über SAP Event Mesh gesendet und empfangen. Bei einer vorübergehenden oder andauernden Unterbrechung der Verbindung zwischen SAP S/4HANA Cloud und SAP Event Mesh werden Monitoring-Informationen protokolliert, um Integrationsadministrator\*in-

nen rechtzeitig zu informieren. Wenn ein Business Event aus irgendeinem Grund nicht veröffentlicht oder konsumiert werden kann, werden die Monitoring-Informationen auch im SAP Application Interface Framework protokolliert. So können Anwender\*innen in solchen Situationen rechtzeitig und angemessen reagieren und z. B. erneut versuchen, ein eingehendes Event in SAP S/4HANA Cloud zu verarbeiten.

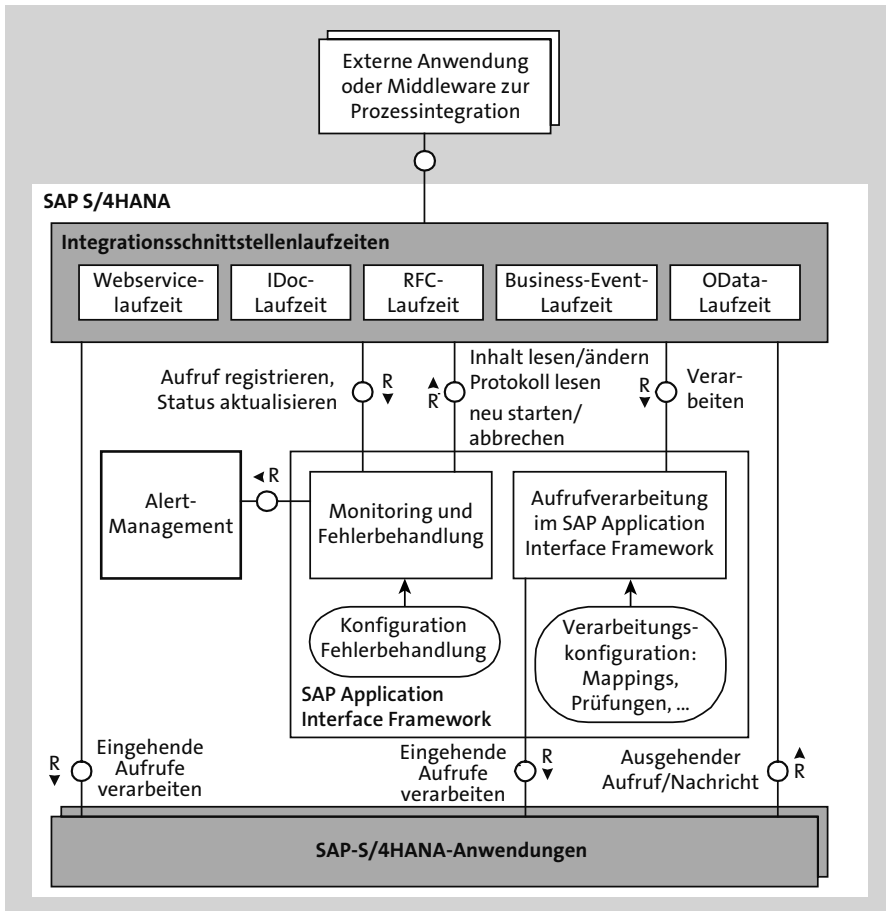


Abbildung 6.1 SAP Application Interface Framework

Mit der Monitoring-Funktionalität des SAP Application Interface Framework können die Benutzer\*innen die Aufrufe sowie die dazugehörigen Statusmeldungen und Detailangaben zu eventuell aufgetretenen Fehlern einsehen. Das SAP Application Interface Framework liest die Details aus den Protokollen aus und zeigt sie entsprechend der Konfiguration für die jeweilige Anwendung an. Das Framework kann auch Schlüsselfelder aus dem Nachrichteninhalt extrahieren und sie in Integrationsmonitoren anzeigen, wo sie von Ihnen als Suchkriterien verwendet werden können. Wenn ein Fehler auf Anwendungsebene aufgetreten ist, können die Benutzer\*innen versuchen,



diesen zu beheben, indem sie fehlende Daten bereitstellen oder fehlendes Customizing vornehmen. Als Teil der anwendungsspezifischen Konfiguration können Links definiert werden, mit denen man vom Monitoring direkt zur jeweiligen Geschäftsanwendung springen kann, um die entsprechenden Korrekturen vorzunehmen. Asynchrone Nachrichten werden von den Laufzeitumgebungen in Queues gespeichert. Das SAP Application Interface Framework ist in der Lage, die Nachrichteninhalte aus den verschiedenen Queues der Schnittstellenlaufzeiten auszulesen und für die Benutzer\*innen aufzubereiten. Die Benutzer\*innen haben die Möglichkeit, den Nachrichteninhalte in einem formularbasierten, einfach zu bedienenden Editor zu bearbeiten. Die bearbeitete Version wird dann vom SAP Application Interface Framework an die Schnittstellenlaufzeit zurückgegeben. Wenn das Problem schließlich behoben ist, kann das Framework die Schnittstellen-Laufzeit anweisen, die Nachrichten in der Warteschlange erneut zu verarbeiten.

Ausgehende Nachrichten und Anfragen werden nach einem ähnlichen Ablauf wie dem für die eingehende Kommunikation verarbeitet. Vor und nach dem Sendevorgang wird die Monitoring-Komponente des Frameworks entsprechend informiert, und falls der Sendevorgang fehlschlägt, können die Benutzer\*innen versuchen, das Problem zu beheben und eine erneute Verarbeitung zu initiieren.

Bisher haben wir die Funktionen des SAP Application Interface Framework zur Überwachung und Fehlerbehandlung besprochen. Das Framework verfügt aber über noch weitere Funktionen. Es kann sich in die Verarbeitung von Aufrufen und Nachrichten einklinken, um beispielsweise Mappings von Werten und Strukturen vorzunehmen oder um Prüfungen auszuführen. Hierzu kann im SAP Application Interface Framework für jede Schnittstelle eine Verarbeitungskonfiguration spezifiziert werden. Wenn eine solche Konfiguration vorliegt, ruft die Schnittstellenlaufzeit zur Verarbeitung eingehender Aufrufe nicht die Anwendung auf, sondern die Aufrufverarbeitung im SAP Application Interface Framework. Das Framework kann nun Prüfungen und Mappings auf der Grundlage der schnittstellenspezifischen Konfiguration vornehmen. Wenn beispielsweise eine aufrufende Anwendung in einem bestimmten Feld einer Nachricht stets den falschen Codewert sendet, könnte man dies an dieser Stelle mit einem Werte-Mapping korrigieren. Nachdem die konfigurierten Mappings und Prüfungen abgeschlossen sind, ruft das SAP Application Interface Framework schließlich die Anwendung mit den potenziell durch das Mapping geänderten Daten zur weiteren Verarbeitung auf. Bei ausgehenden Aufrufen ruft die Schnittstellenlaufzeit vor dem Senden die Verarbeitung durch das SAP Application Interface Framework auf, um Mappings und Prüfungen ausführen zu können. Das Framework liefert die durch das Mapping bearbeitete Nachricht zurück, die dann von der Schnittstellenlaufzeit gesendet wird. Unabhängig hiervon wird das Monitoring für Registrierungen und Statusaktualisierungen aufgerufen, wenn es entsprechend konfiguriert ist.

Das SAP Application Interface Framework steht nicht nur für die Verarbeitung von Schnittstellenaufrufen zur Verfügung, sondern kann von Anwendungen auch in anderen Kontexten zur Durchführung von Mappings, Prüfungen usw. aufgerufen und verwendet werden.

## 6.4 Kommunikationsmanagement in SAP S/4HANA Cloud

SAP S/4HANA Cloud verfolgt einen neuen Ansatz für die Konfiguration der Kommunikation mit externen Systemen und Diensten, der als *Kommunikationsmanagement* bezeichnet wird. Das Ziel dabei besteht darin, die Kommunikation mit anderen Systemen über die verschiedenen Schnittstellentechnologien hinweg zu vereinfachen und zu vereinheitlichen. In diesem Abschnitt werden die Konzepte des Kommunikationsmanagements aus zwei Gründen ausführlicher behandelt: Erstens, weil es sich um ein neues und von der Architektur her interessantes Konzept handelt, und zweitens, weil es ein gutes Beispiel für die Vereinfachungen und die höhere Benutzerfreundlichkeit in SAP S/4HANA Cloud ist.

Das Kommunikationsmanagement in SAP S/4HANA Cloud basiert auf den folgenden Prinzipien:

- Für die verschiedenen Arten von technischen Schnittstellen (wie OData, SOAP und IDoc) wird dasselbe einheitliche Konzept für das Kommunikationsmanagement verwendet.
- Die Benutzer\*innen konfigurieren die Konnektivität zu anderen Systemen auf einer hohen Abstraktionsebene. Dabei bleiben die Laufzeitumgebungen der zugrunde liegenden Schnittstellentechnologien und deren Konfiguration den Benutzer\*innen verborgen. Erforderliche technische Artefakte – wie Destinations oder logische Ports für SOAP-Services – werden im Hintergrund automatisch angelegt. Dies trägt nicht nur zu einer Verbesserung der Benutzerfreundlichkeit bei, sondern ist auch eine unbedingte Voraussetzung für ein SaaS-Angebot, bei dem die Benutzer\*innen keinen Zugang zu den zugrunde liegenden technischen Tools und Infrastrukturen haben.
- Die Kommunikation mit anderen Systemen wird jeweils für eine Menge von Eingangs- und Ausgangsschnittstellen verwaltet, die für ein sinnvolles Szenario erforderlich sind. Ein bestimmter Geschäftsprozess kann es beispielsweise erfordern, dass mit einer anderen Anwendung Informationen über verschiedene Schnittstellen in beide Richtungen ausgetauscht werden. Mit dem Kommunikationsmanagement werden die hierfür erforderlichen Konfigurationen an einer Stelle verwaltet und nicht individuell für jede einzelne Schnittstelle.

Abbildung 6.2 zeigt die Elemente des Kommunikationsmanagements und ihre Beziehungen auf konzeptioneller Ebene. Die Elemente im oberen Teil des Diagramms wer-

den vorab definiert, in der Regel von SAP. Der untere Bereich zeigt die Elemente, die Benutzer\*innen anlegen, wenn sie die Kommunikation für konkrete Integrationen mit konkreten Systemen konfigurieren.

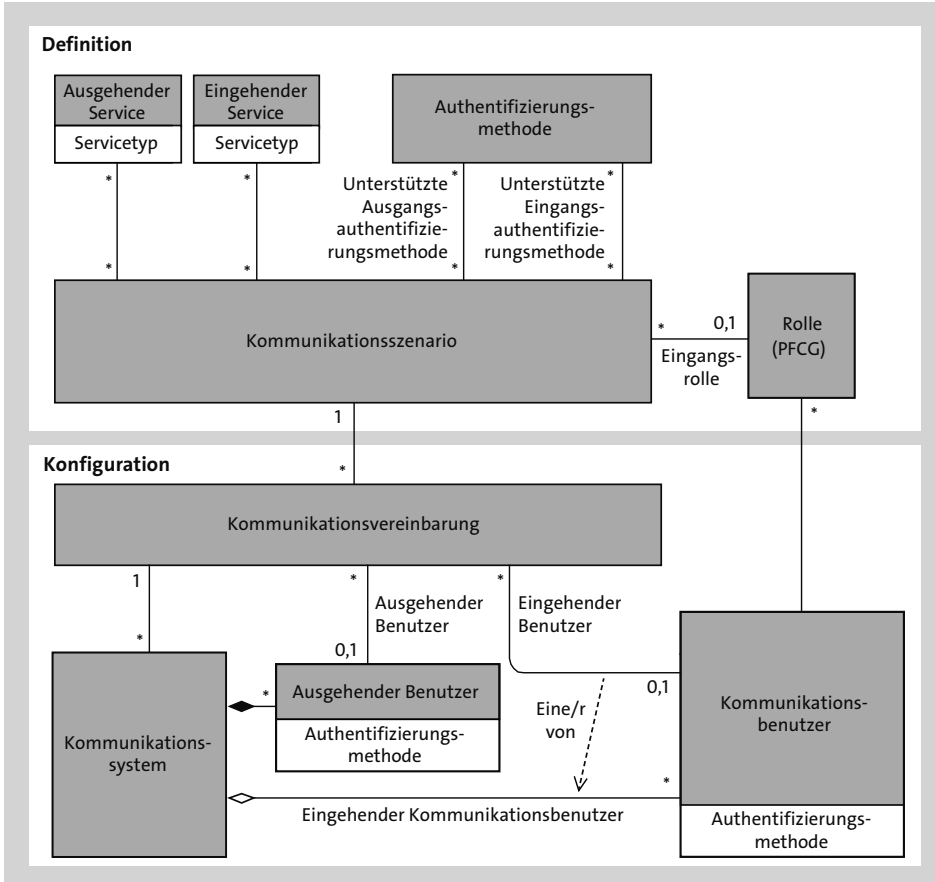


Abbildung 6.2 Kommunikationsmanagement in SAP S/4HANA Cloud

In den folgenden Abschnitten werden wir auf die wesentlichen Elemente des Kommunikationsmanagements näher eingehen.

### 6.4.1 Kommunikationsszenario

Das *Kommunikationsszenario* fasst die ein- und ausgehenden Services zusammen, die erforderlich sind, um ein betriebswirtschaftliches Ziel zu erreichen. Betrachten wir als Beispiel hierfür das Kommunikationsszenario *SAP\_COM\_0508* für die Integration von SAP S/4HANA mit *SAP Business Network for Procurement* (früher bekannt als Ariba Network) über *SAP Integration Suite, verwaltetes Gateway für Spend Management* und *SAP Business Network*. Dieses Kommunikationsszenario fasst die Services

zusammen, die benötigt werden, damit die Beschaffungsfunktion in SAP S/4HANA Informationen mit einem Lieferanten in SAP Business Network for Procurement austauschen kann. Dazu gehören beispielsweise ausgehende Services für den Versand von Bestellungen und Zahlungsavisen an Lieferanten und eingehende Services für den Empfang von Auftragsbestätigungen und Rechnungen von Lieferanten.

Jeder ein- und ausgehende Service eines Kommunikationsszenarios hat einen Servicetyp, der seine Schnittstellentechnologie angibt. Servicetypen für eingehende Services sind z. B. OData V4, OData V2, SOAP, IDoc, Event Consumer oder RFC. Beispiele für ausgehende Servicetypen sind SOAP, IDoc, HTTP-Service, RFC oder SMTP (Simple Mail Transfer Protocol, E-Mail). In einem Kommunikationsszenario können Services unterschiedlicher Servicetypen miteinander kombiniert werden.

Im Kommunikationsszenario ist auch festgelegt, welche Authentifizierungsprotokolle für die eingehende und die ausgehende Kommunikation unterstützt werden. Zur Authentifizierung eingehender Aufrufe stehen Basic Authentication mit Benutzername und Kennwort, X.509-Zertifikate und OAuth zur Auswahl. Für die Authentifizierung ausgehender Aufrufe gibt es die Möglichkeiten Basic Authentication mit Benutzername und Kennwort, X.509-Zertifikat, OAuth oder Kommunikation ohne Authentifizierung.

Kommunikationsszenarien sind wie Vorlagen, die von den Administrator\*innen instanziiert werden können, um die Kommunikation mit einem bestimmten Remote-System zu konfigurieren. Dazu müssen die Benutzer\*innen die technischen Details für die Verbindung mit dem Remote-System und die Authentifizierungsdaten angeben, die für die eingehende und die ausgehende Kommunikation verwendet werden sollen. Diese Informationen werden mithilfe von drei Konzepten spezifiziert: Kommunikationsbenutzer, Kommunikationssystem und Kommunikationsvereinbarung.

### 6.4.2 Kommunikationsbenutzer

Bei einem Aufruf eines eingehenden Service muss zunächst eine Authentifizierung erfolgen. Sobald der Aufrufer authentifiziert ist, kann SAP S/4HANA Cloud prüfen, ob er berechtigt ist, und wird wissen, welche Berechtigungsregeln bei der Verarbeitung des Aufrufs angewendet werden müssen. Im Kommunikationsmanagement werden die Identitäten für die Authentifizierung von eingehenden Services definiert, indem durch *Kommunikationsbenutzer* angelegt werden; bei denen es sich um eine besondere Art von technischen Benutzern handelt. Bei diesen Benutzern handelt es sich nicht um Menschen, sondern um technische Entitäten, die dem Informationsaustausch zwischen den Systemen dienen. Kommunikationsbenutzer können über ihren Benutzernamen und ihr Passwort oder mit einem Authentifizierungszertifikat authentifiziert werden.

### 6.4.3 Kommunikationssystem

Bevor ein Kommunikationsszenario für ein bestimmtes Remote-System instanziiert werden kann, müssen Sie dieses System als Kommunikationspartner bekannt machen und die für die Kommunikation erforderlichen technischen Informationen bereitstellen. Zu diesem Zweck müssen Sie ein Objekt erstellen, das als *Kommunikationssystem* bezeichnet wird. Ein Kommunikationssystem steht für ein Remote-System, das eingehende Services aufrufen oder das Ziel von ausgehenden Serviceaufrufen sein kann. Bei der Erstellung eines Kommunikationssystems müssen Verbindungsattribute angegeben werden, z. B. der Hostname, die OAuth2-Endpunkte oder erforderlichenfalls die Angaben zum Cloud Connector. Für die eingehende Kommunikation müssen dem Kommunikationssystem ein oder mehrere Kommunikationsbenutzer zugewiesen werden. Es ist möglich, dieselben Kommunikationsbenutzer verschiedenen Kommunikationssystemen zuzuordnen. Für die ausgehende Kommunikation legen Sie ausgehende Benutzer an, die zum Kommunikationssystem gehören. Bei diesen handelt es sich nicht um Benutzer in SAP S/4HANA, sondern solche für den Aufruf des Remote-Systems. Allen ausgehenden Benutzern sind spezifische Authentifizierungsmethoden zugeordnet. Je nach Authentifizierungsmethode werden mit den ausgehenden Benutzern unterschiedliche Details gespeichert, z. B. der Benutzername und das Kennwort, die OAuth-Client-Anmeldeinformationen oder Client-Zertifikate.

### 6.4.4 Kommunikationsvereinbarung

Bevor SAP S/4HANA Cloud mit dem Remote-System kommunizieren kann, müssen Sie eine *Kommunikationsvereinbarung* erstellen, die ein spezifisches Kommunikationsszenario für ein spezifisches Kommunikationssystem instanziiert. In der Kommunikationsvereinbarung müssen ein eingehender und ein ausgehender Benutzer spezifiziert werden (sofern eingehende und ausgehende Services beteiligt sind). Diese Benutzer müssen dem Kommunikationssystem als eingehender bzw. ausgehender Benutzer zugeordnet sein, und ihre Authentifizierungsmethode muss für die eingehende bzw. ausgehende Kommunikation von dem betreffenden Szenario unterstützt werden.

Mit einer Kommunikationsvereinbarung legt das System automatisch die notwendigen technischen Einstellungen und Artefakte für die Kommunikation an, die intern von den verschiedenen Laufzeitumgebungen für die Kommunikation benötigt werden. Wie bereits in Abbildung 6.2 gezeigt, kann ein Kommunikationsszenario eine Rolle für die eingehende Kommunikation vorsehen. Beim Anlegen einer Kommunikationsvereinbarung wird diese Rolle den eingehenden Kommunikationsbenutzern zugewiesen. Dadurch wird sichergestellt, dass die erforderlichen Berechtigungen vorhanden sind, wenn eingehende Services mit den Anmeldedaten dieser Kommunikationsbenutzer aufgerufen werden.

### 6.4.5 Aufrufen eingehender Services mit Weitergabe von Benutzeridentitäten

Normalerweise erfolgt die Authentifizierung eines eingehenden Serviceaufrufs mit den Authentifizierungsdaten des eingehenden Kommunikationsbenutzers, der in der Kommunikationsvereinbarung angegeben ist. Der eingehende Aufruf wird dann mit der Identität und den Berechtigungen dieses technischen Kommunikationsbenutzers ausgeführt. Der eingehende Service wird in diesem Fall sozusagen im Namen des vertrauenswürdigen Remote-Systems ausgeführt und nicht im Namen eines bestimmten Anwendungsbenutzers oder einer bestimmten Anwendungsbenutzerin. Dieser Ansatz eignet sich beispielsweise für typische Szenarien der Prozessintegration und Datenverteilung.

Manchmal muss der eingehende Aufruf jedoch im Namen der realen Person ausgeführt werden, die an dem beteiligten Remote-System angemeldet ist. In so einem Fall wird die Identität der angemeldeten Person vom Remote-System an SAP S/4HANA Cloud weitergegeben, und diese wird dort dem entsprechenden Anwendungsbenutzer oder der entsprechenden Anwendungsbenutzerin zugeordnet. Der eingehende Aufruf wird dann mit den Berechtigungen dieses Anwendungsbenutzers bzw. dieser Anwendungsbenutzerin ausgeführt. In SAP S/4HANA Cloud erfolgt diese *Weitergabe von Benutzeridentitäten* mithilfe des *OAuth 2.0 SAML Bearer Assertion Flow* (SAML: Security Assertion Markup Language). Das funktioniert vereinfacht so: Das Remote-System verfügt über eine SAML-Assertion, mit der es die Identität der Person SAP S/4HANA Cloud gegenüber nachweisen kann. SAP S/4HANA Cloud vertraut dieser SAML-Assertion, weil sie von einem Identity Provider ausgestellt und signiert wurde, der zuvor beim Anlegen des Kommunikationssystems als vertrauenswürdig konfiguriert wurde. Bevor das Remote-System den eingehenden Service in SAP S/4HANA aufruft, verwendet es diese SAML-Assertion, um vom OAuth-Autorisierungsserver von SAP S/4HANA Cloud ein OAuth-Zugriffstoken für den eingehenden Service anzufordern. SAP S/4HANA Cloud validiert die Tokenanforderung und ordnet die darin angegebene Benutzeridentität einem Anwendungsbenutzer oder einer Anwendungsbenutzerin im System zu. Dann erzeugt SAP S/4HANA Cloud ein entsprechendes OAuth-Zugriffstoken und schickt es an das Remote-System zurück. Dieses Token bestätigt die Benutzeridentität in SAP S/4HANA Cloud und die Erlaubnis zum Aufrufen des eingehenden Service. Das Remote-System verwendet dieses Token zur Authentifizierung des Serviceaufrufs, der dann in SAP S/4HANA mit der Identität und den Berechtigungen des damit identifizierten Anwendungsbenutzers bzw. der identifizierten Anwendungsbenutzerin ausgeführt werden. In diesem Szenario werden die Anmeldedaten des eingehenden Kommunikationsbenutzers nicht für den eigentlichen Serviceaufruf verwendet. Stattdessen enthalten sie den Benutzernamen und das Kennwort, die zur Anforderung des Tokens beim OAuth-Autorisierungsserver benötigt werden.

## 6.5 Cloud Connector

Beim *Cloud Connector* handelt es sich um eine Komponente, die auf einem Computer im On-Premise-Netzwerk installiert wird, um die Kommunikation zwischen Cloud- und On-Premise-Anwendungen zu ermöglichen. Sie können den Cloud Connector beispielsweise für die Kommunikation zwischen SAP S/4HANA Cloud und On-Premise-Systemen, zwischen kundeneigenen Anwendungen auf der SAP Business Technology Platform (SAP BTP) und SAP S/4HANA oder zwischen SAP S/4HANA und Cloud-Services wie SAP Event Mesh einsetzen. Der Cloud Connector unterstützt HTTP- und Nicht-HTTP-Protokolle, wie etwa RFC, oder den Zugriff auf eine SAP-HANA-Datenbank über Open Database Connectivity (ODBC)/Java Database Connectivity (JDBC).

In den folgenden Abschnitten werden wir einen Überblick über die wichtigsten Konzepte im Zusammenhang mit dem Cloud Connector geben und die RFC-basierte Kommunikation erläutern.

### 6.5.1 Grundsätze der Kommunikation über den Cloud Connector

Der Cloud Connector fungiert als Reverse Proxy zwischen dem On-Premise-Netzwerk auf der einen Seite und SAP S/4HANA Cloud oder der SAP BTP auf der anderen Seite. Die Kommunikation erfolgt über eine sichere und verschlüsselte Tunnelverbindung, die vom Cloud Connector als ausgehende Verbindung aufgebaut wird (siehe Abbildung 6.3). Über diesen Tunnel können Sie eingehende Aufrufe aus der Cloud tätigen, ohne irgendwelche Ports in Ihrer On-Premise-Firewall öffnen zu müssen. Darüber hinaus ermöglicht der Cloud Connector eine Nicht-HTTP-Kommunikation in beide Richtungen.

Administrator\*innen müssen explizit festlegen, welche Ressourcen aus den On-Premise-Systemen über den Cloud Connector nach außen verfügbar sind. Dabei können sie auch einschränken, welche Cloud-Anwendungen den Cloud Connector verwenden dürfen, indem sie eine Liste vertrauenswürdiger Cloud-Anwendungen einrichten.

Bei an On-Premise-Systeme gerichteten Aufrufen kann die Authentifizierung eines technischen Service durch Basic Authentication mit Benutzername und Passwort erfolgen oder mithilfe der Weitergabe von Benutzeridentitäten. Bei der Weitergabe von Benutzeridentitäten verwendet die aufrufende Cloud-Anwendung ein Token eines SAML2-Identity-Providers, durch das die Benutzeridentität bestätigt wird. Das On-Premise-Zielsystem ordnet diese Identität einem lokalen Benutzer oder einer lokalen Benutzerin zu und führt den Aufruf mit dessen oder deren Berechtigungen aus. Für die Weitergabe von Benutzeridentitäten muss man zuvor zwei Vertrauensbeziehungen konfigurieren: Erstens muss der Cloud Connector dem Identity Provider ver-

trauen, der das Token ausgestellt hat, und zweitens muss das On-Premise-Zielsystem dem Cloud Connector vertrauen.

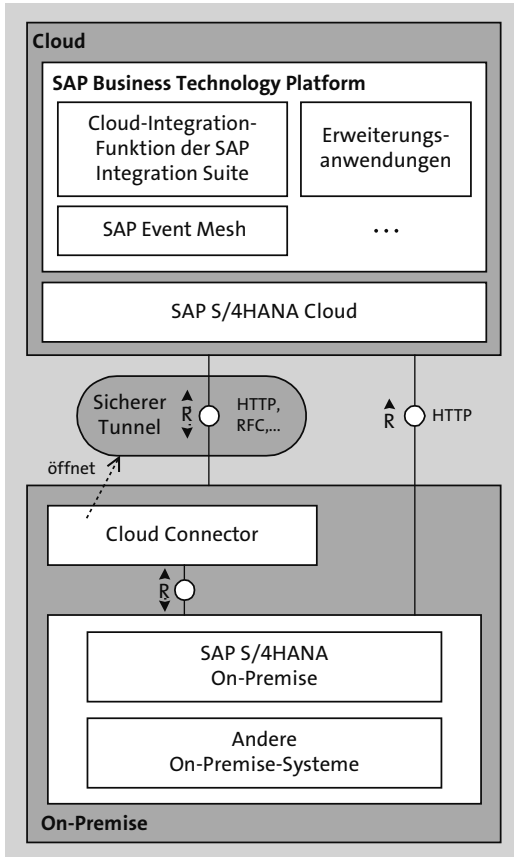


Abbildung 6.3 Cloud Connector

### 6.5.2 RFC-Kommunikation mit SAP S/4HANA Cloud

Der Cloud Connector ermöglicht native RFC-Aufrufe in beide Richtungen – von Cloud-Systemen zu On-Premise-Systemen und von On-Premise-Systemen zu Cloud-Systemen. So ist es beispielsweise möglich, dass Ihr On-Premise-System eine RFC-Schnittstelle in SAP S/4HANA Cloud aufruft. Dazu legen Sie einen Servicekanal in Ihrem Cloud Connector an, der RFCs zu Ihrem SAP-S/4HANA-Cloud-Tenant erlaubt.

Beachten Sie, dass Sie in einem SAP-S/4HANA-Cloud-Tenant nicht beliebige RFC-basierte Schnittstellen aufrufen können. Für die Kommunikation mit SAP S/4HANA Cloud muss ein vordefiniertes Kommunikationsszenario vorhanden sein (siehe Abschnitt 6.4). SAP hat nur eine begrenzte Anzahl von Kommunikationsszenarien mit traditionellen RFC-basierten Schnittstellen freigegeben (siehe auch Abschnitt 6.1.2).



## 6.6 Prozessintegrationstechnologie

SAPs Strategie zur Prozessintegration zwischen SAP-Anwendungen basiert hauptsächlich darauf, dass die Anwendungen über abgestimmte APIs direkt miteinander kommunizieren. Es gibt jedoch Fälle, in denen es sinnvoll ist, stattdessen eine Integrations-Middleware einzusetzen. Eine Integrations-Middleware ist z. B. hilfreich, wenn Sie Produkte von Drittanbietern integrieren, eigene Integrations-Workflows entwickeln, Nachrichten an verschiedene Empfängersysteme weiterleiten oder komplexe Mappings zentral verwalten möchten.

Die SAP Integration Suite umfasst eine Vielzahl von Integrationswerkzeugen wie z. B. Cloud Integration, SAP Event Mesh (siehe Abschnitt 6.7), Trading Partner Management, API Management, Integration Assessment, Integration Advisor und Open Connectors.

Als Middleware für die Prozessintegration in der Cloud empfiehlt SAP die Cloud-Integration-Funktion der SAP Integration Suite. *Cloud Integration* ist ein Service, der auf der SAP BTP läuft und für die Integration von Anwendungen eingesetzt werden kann, sowohl für Cloud-to-Cloud-Szenarien als auch für Cloud-to-On-Premise-Szenarien (gegebenenfalls unter Verwendung des Cloud Connectors). Die Funktionsweise von Cloud Integration wird in Abbildung 6.4 veranschaulicht.

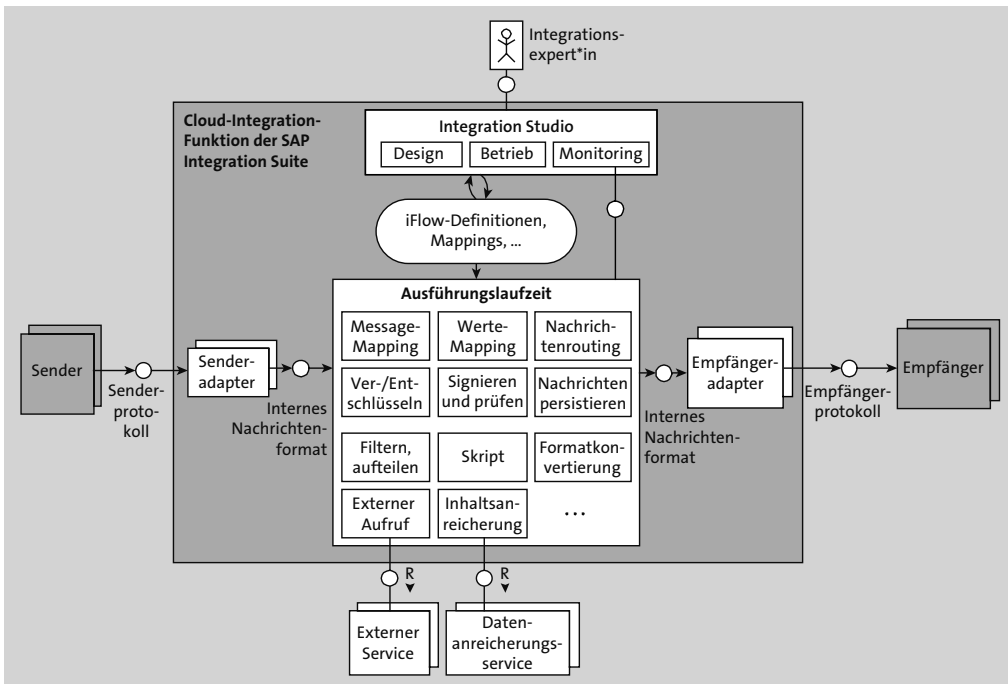


Abbildung 6.4 Cloud Integration

Die Designtime- und Konfigurationsanwendungen von Cloud Integration haben eine Multi-Tenant-Architektur mit isolierten Tenants auf gemeinsam genutzten Rechenressourcen. Für die Laufzeit sieht dies anders aus. Hier werden den Kunden Single-Tenant-Worker-Nodes zur Verfügung gestellt, um eine maximale Isolierung der einzelnen Tenants zu gewährleisten. Mit dieser Architektur hat die Laufzeitauslastung eines einzelnen Tenants keinerlei Auswirkungen auf die Performance und die Verfügbarkeit der übrigen Tenants.

Die von den Sendern kommenden Nachrichten werden von Cloud Integration verarbeitet und an die Empfänger weitergeleitet. Sowohl Sender als auch Empfänger sind mit Cloud Integration jeweils über Adapter verbunden, die verschiedene Kommunikationsprotokolle implementieren. Die Senderadapter konvertieren eingehende Daten aus den externen Protokollen der Sender in ein internes Nachrichtenformat, das Cloud Integration verarbeiten kann. Nach erfolgter Verarbeitung werden die resultierenden ausgehenden Nachrichten durch einen Empfängeradapter vom internen Format in das Protokoll des jeweiligen Empfängers konvertiert. Eingehende Daten können von den Sendern (etwa über HTTP) nach dem Push-Verfahren gesendet oder in regelmäßigen Abständen von Cloud Integration abgerufen werden.

Cloud Integration verfügt über Adapter für verschiedene Protokolle wie SOAP, OData, HTTP/REST, Advanced Message Queuing Protocol (AMQP), Kafka, IDoc via SOAP, RFC (nur Client), E-Mail, Java Messenger Service (JMS), Applicability Statement (AS2), AS4, commerce eXtensible Markup Language (cXML) für SAP-Ariba-Lösungen und viele mehr. Mit den Open Connectors der SAP Integration Suite können Sie eine Vielzahl von APIs von Drittanbietern integrieren. Mit Open Connectors können Sie auf einheitliche Weise auf APIs zahlreicher Cloud-Anbieter zugreifen – mit standardisierter Authentifizierung, Fehlerbehandlung, Paginierung und standardisierten Datenstrukturen.

Die Verarbeitung und Weiterleitung von Nachrichten wird durch Integration Flows (iFlows) beschrieben. iFlows definieren, wie eingehende Nachrichten verarbeitet und an die Empfänger verteilt werden. iFlows werden mithilfe des Standards *Business Process Model and Notation* (BPMN) visuell modelliert. Die dafür benötigten Designtools sind in das webbasierte Integration Studio integriert.

iFlows stellen die Schrittfolge zur Verarbeitung und Weiterleitung von Nachrichten grafisch dar. Dabei sind beispielsweise folgende Schritte möglich:

- Mapping von Nachrichtenstrukturen
- Werte-Mappings
- Filterung von Nachrichtenteilen
- Änderung von Nachrichteninhalten

- Kodierung und Dekodierung (ZIP, Base64), Verschlüsselung und Entschlüsselung von Nachrichten
- Konvertierung von Nachrichtenformaten (z. B. zwischen JavaScript Object Notation [JSON], eXtensible Markup Language [XML], Comma-Separated Values [CSV])
- digitale Signaturen und Signaturprüfungen
- Aufrufen externer Services (asynchron oder synchron)
- Datenanreicherung, Zusammenführung von Inhalten mit Daten aus externen Services
- Aufteilung komplexer Nachrichten
- Persistieren von Nachrichten
- Weiterleitung von Nachrichten an einen oder mehrere Empfänger
- benutzerdefinierte Verarbeitungsschritte, die durch ein Skript in Groovy oder JavaScript implementiert werden

iFlows werden von der Cloud-Integration-Laufzeitumgebung ausgeführt, die auf dem Open-Source-Integrations-Framework *Apache Camel* beruht.

SAP stellt vordefinierten Integrations-Content zur Verfügung, der im SAP Business Accelerator Hub veröffentlicht wird. Dieser Content steht im Integration Studio zur Verfügung, damit Sie ihn in Ihrem Tenant verwenden können. Für Szenarien, in denen SAP-Produkte direkt über APIs integriert werden, bietet SAP Integrations-Content an, den Sie als Ausgangspunkt für die Einrichtung Ihrer benutzerdefinierten Integration mit Cloud Integration verwenden können. In diesen Fällen leitet der von SAP bereitgestellte Content die Nachrichten einfach nur durch, aber Sie können ihn als Grundlage für Ihre eigenen Erweiterungen und Ergänzungen verwenden.

Als Integrations-Middleware für On-Premise-Szenarien setzen viele SAP Kunden *SAP Process Integration* und *SAP Process Orchestration* ein und verwenden diese auch zur Integration mit SAP S/4HANA. Die darin enthaltene *Advanced Adapter Engine Extended* ist mit Version 7.5 auch in der Lage, mit Cloud Integration erstellten Integrations-Content on-premise auszuführen. SAP hat jedoch die SAP Integration Suite schon seit einiger Zeit als das Integrationsprodukt der Zukunft positioniert und die schrittweise Migration von Integrationsszenarien von SAP Process Integration/Orchestration auf die SAP Integration Suite empfohlen. Ein wichtiger Wegbereiter für diese Migration ist die neue Option *Edge Integration Cell*, die kürzlich zur SAP Integration Suite hinzugefügt wurde. Edge Integration Cell ist eine optionale Erweiterung von Cloud Integration, die hybride Integrationsszenarien unterstützt, die sowohl On-Premise-Systeme als auch Cloud-Anwendungen umfassen. Edge Integration Cell kann on-premise installiert werden oder bei Ihrem Cloud-Infrastruktur-Provider. Es führt Ihre mit Cloud Integration erstellten iFlows lokal aus und kann somit Integra-

tionen zwischen On-Premise-Systemen realisieren, ohne dass die Daten Ihre Umgebung oder Ihr Rechenzentrum verlassen.

## 6.7 Event-basierte Integration

Manchmal möchten Sie andere Systeme oder Services mittels Events über Business Events benachrichtigen, die in SAP S/4HANA stattgefunden haben, etwa über die Anlage von Geschäftspartnern, die Veröffentlichung einer Ausschreibung oder eine Änderung an einem Kundenauftrag. Zu den wichtigen Anwendungsfällen gehören Side-by-Side-Erweiterungen, die über Änderungen in SAP S/4HANA informiert werden müssen (siehe auch Abschnitt 5.3.5). Hier dienen Events als Ergänzung zu den üblichen APIs. Durch die Veröffentlichung von Events können andere Systeme benachrichtigt werden, ohne dass eine direkte Kommunikationsverbindung zwischen den beteiligten Systemen aufgebaut werden muss. Ohne Events und die zugehörige Infrastruktur müssten die Informationen entweder durch API-Aufrufe an die Zielsysteme geschickt oder von den Zielsystemen durch regelmäßige API-Aufrufe vom Quellsystem kontinuierlich abgefragt werden. Dies würde voraussetzen, dass die Systeme die technischen Details zur Kommunikation mit dem anderen System kennen und sich bei diesem authentisieren können. Darüber hinaus würde das Übermitteln großer Mengen von Nachrichten an viele Empfänger eine zusätzliche Belastung für das Quellsystem darstellen. Synchroner API-Aufrufe würden zudem voraussetzen, dass die empfangende Seite zu dem Zeitpunkt online ist, zu dem die Benachrichtigung gesendet werden soll.

Daher erfolgt die Verteilung von Events in der Regel unter Zuhilfenahme eines speziellen Service, der als *Event Broker* bezeichnet wird. Dieser Broker ermöglicht durch Zwischenspeicherung die zeitliche Entkopplung durch asynchrone Kommunikation und arbeitet nach dem Publish/Subscribe-Muster. Das Quellsystem schickt die zu veröffentlichenden Events an den Event Broker, der sie an einen oder mehrere Empfänger verteilt. Auf diese Weise muss erstens das Quellsystem nichts über die Identität und die technischen Details der Event-Empfänger wissen, und zweitens werden die Speicherung und die Weiterleitung großer Mengen von Events an den Event Broker ausgelagert. Event Broker nutzen standardisierte Messaging-Protokolle und Message-Queues und ermöglichen damit die Entkopplung von Sender und Empfänger.

Der Service, mit dem Sie Ihre Anwendungen Event-basiert mit SAP S/4HANA integrieren können, heißt *SAP Event Mesh*. Bevor wir darauf eingehen, wie Events in SAP S/4HANA definiert, veröffentlicht und konsumiert werden, geben wir Ihnen daher zunächst einen kurzen Überblick über die Konzepte von SAP Event Mesh.

### 6.7.1 SAP Event Mesh

Bei SAP Event Mesh handelt es sich um einen vollumfänglich von SAP verwalteten, skalierbaren Cloud-Service auf der SAP BTP, mit dem Anwendungen Events über die Standardprotokolle AMQP 1.0 und Message Queuing Telemetry Transport (MQTT), über eine proprietäre REST-API und über Webhooks senden und empfangen können. Wenn eine Anwendung ein Event in SAP Event Mesh veröffentlichen möchte, betitelt sie das Event mit einem *Topic*, das typischerweise beschreibt, worum es bei dem Event geht. Technisch gesehen handelt es sich bei einem Topic um eine Zeichenkette mit durch Schrägstriche voneinander getrennten Segmenten, sodass man sie entsprechend einer Hierarchie von Themen und Unterthemen aufbauen kann. In SAP S/4HANA beginnen die Topics der Events mit einem Namensraumpräfix, gefolgt vom Event-Typ. Hier ein Beispiel für ein Topic:

```
sap/s4/beh/businesspartner/v1/BusinessPartner/Created/v1
```

SAP Event Mesh unterstützt die Möglichkeit, dass eine Anwendung direkt die sie interessierenden Topics konsumiert, ohne zwischengeschaltete Queues. In diesem Fall erhält die Anwendung alle Events, die mit diesen Topics veröffentlicht werden, aber nur während sie mit SAP Event Mesh verbunden ist. Damit Empfänger auch dann Events empfangen können, wenn sie zum Zeitpunkt der Veröffentlichung nicht verbunden waren, unterstützt SAP Event Mesh das Speichern von Events in *Queues*. Dabei ist eine Queue genau einer Anwendung zugeordnet, die die darin gespeicherten Events konsumiert. Eine Queue kann auf bestimmte Topics subskribiert werden, sodass die unter diesen Topics veröffentlichten Events in der Queue gespeichert werden. Dabei können Topic-Muster mit Platzhaltern wie einem Sternchen »\*« verwendet werden, um die Queue auf alle passenden Topics zu subskribieren. Indem Sie mehrere Queues auf dieselben Topics subskribieren, können Sie ankommende Events an mehrere Queues und damit an mehrere empfangende Anwendungen verteilen.

Bei SAP Event Mesh handelt es sich um einen Multi-Tenant-Service, bei dem die Kunden jeweils ihren eigenen Tenant erhalten. Sie können Ihren SAP-Event-Mesh-Tenant als separaten virtuellen Event Bus betrachten, mit dem sich Ihre Anwendungen verbinden und so miteinander kommunizieren können. Dazu muss jede Anwendung am Event Bus als *Event-Bus-Client* registriert werden. Jeder Client hat seine eigene Konfiguration, in der Sie Regeln festlegen können, durch die eingeschränkt wird, welche Topics der Client senden und welche Queues er konsumieren kann. Jeder Client hat seine eigenen Zugangsdaten, mit denen er sich bei SAP Event Mesh mit dem OAuth Client Credentials Flow authentisiert.

Zu SAP Event Mesh gehört eine Webanwendung, mit der Administrator\*innen die Client-Konfigurationen einsehen sowie die Queues und Topic-Subskriptionen verwalten können. In dieser Anwendung finden Sie auch für die einzelnen verbundenen

Systeme jeweils ein Verzeichnis der Events, die dieses System veröffentlichen kann. Hierzu muss das angeschlossene System diese Funktion unterstützen und einen Event-Discovery-Service anbieten, mit dem SAP Event Mesh diese Information abfragen kann.

### 6.7.2 Event-basierte Integration mit SAP S/4HANA

Abbildung 6.5 zeigt die neue Architektur zum Veröffentlichen und Konsumieren von Business Events in SAP S/4HANA mithilfe des ABAP RESTful Application Programming Model (siehe Abschnitt 2.2). Anwendungen in SAP S/4HANA lösen Events aus, wenn Business-Objekte geändert werden. Dazu nutzen die Anwendungen das ABAP RESTful Application Programming Model, das die Events direkt in derselben Transaktion an die Komponente *Enterprise Event Enablement* weiterleitet. Ebenso ist es möglich, Event Consumer zu definieren. Darüber hinaus bietet das ABAP RESTful Application Programming Model auch Erweiterungsmöglichkeiten für Events.

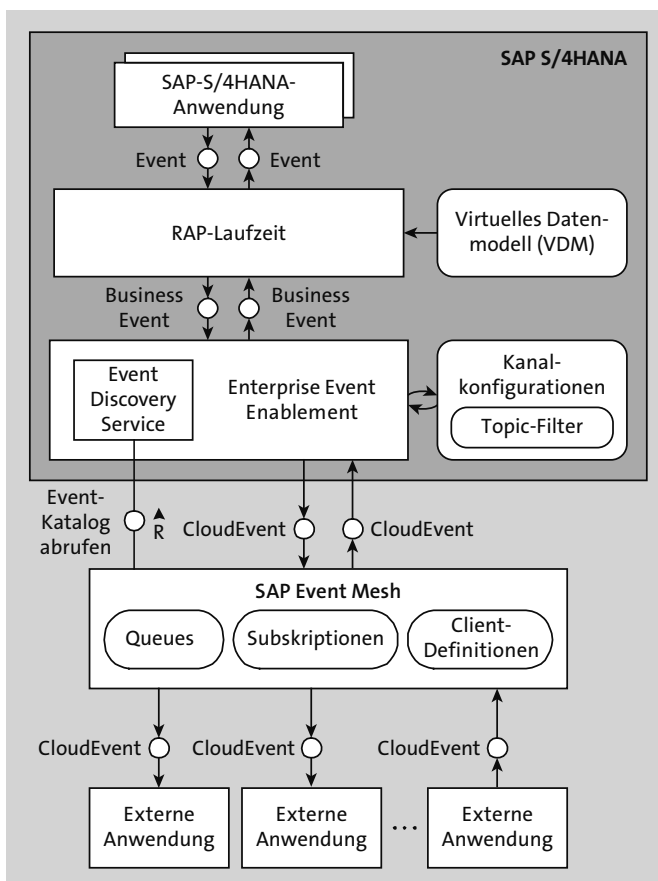


Abbildung 6.5 Architektur für Event-basierte Integration mit SAP S/4HANA

Ausgehende Events werden von Enterprise Event Enablement auf SAP Event Mesh veröffentlicht. Das Format der von SAP S/4HANA veröffentlichten Events entspricht der *CloudEvents-Spezifikation* (<https://cloudevents.io>). Beachten Sie, dass die Events in der Regel nicht die vollständigen Daten der betroffenen Business-Objektinstanzen enthalten. Typischerweise enthält das Event die Kennungen (IDs) der betroffenen Objekte und begrenzte, ausgewählte Daten für die Filterung der Events. Wenn die empfangende Anwendung weitere Informationen benötigt, muss sie die entsprechenden APIs von SAP S/4HANA aufrufen, um nähere Informationen zu den erstellten oder geänderten Instanzen zu erhalten. Um dies zu erleichtern, nutzen CloudEvents und die OData-API-Services das gleiche Feldformat.

### 6.7.3 Business Events in SAP S/4HANA

Lassen Sie uns nun im Detail erläutern, was wir im technischen Sinne unter einem Business Event in SAP S/4HANA verstehen. Ein Business Event in SAP S/4HANA ist ein Event, das extern unter Verwendung von SAP Event Mesh veröffentlicht werden kann. Das Business Event dient zur Benachrichtigung über ein im Zusammenhang mit einem Business-Objekt eingetretenes Ereignis, z. B. das Anlegen oder die Änderung eines Produkts oder das Anlegen, Ändern, Freigeben oder Stornieren eines Auftrags. Für die verschiedenen Business-Objekttypen können unterschiedliche Event-Typen definiert werden.

Technisch gesehen sind Business Events als Events auf *SAP-Objekttypen* definiert. Das Konzept der SAP-Objekttypen wurde eingeführt, um Business-Objekttypen (wie etwa Kundenaufträge oder Produkte) unabhängig von der Art ihrer Implementierung in einen zentralen Katalog mit sinnvollen und einheitlichen Namen zu erfassen. Daneben werden SAP-Objekttypen auch für andere Zwecke verwendet. In Abschnitt 5.1.2 haben wir erklärt, dass sie die Struktur von erweiterbaren Anwendungen im Kontext des Erweiterungskonzepts beschreiben.

Ein Verzeichnis der öffentlichen Events, die in SAP S/4HANA für die verschiedenen Business-Objekte verfügbar sind, finden Sie im SAP Business Accelerator Hub unter dem Menüpunkt **Events**. Wie die öffentlichen APIs werden alle offiziell von SAP freigegebenen Events im SAP Business Accelerator Hub aufgelistet und dokumentiert.

Mit dem ABAP RESTful Application Programming Model können Sie Ihre eigenen Event-Erzeuger erstellen und Ihre eigenen Events veröffentlichen. Darüber hinaus können Sie Event-Consumption-Modelle zur Verarbeitung eingehender externer Events erstellen. In SAP S/4HANA Cloud wird dies durch die Funktionen zur In-Stack-Entwicklererweiterbarkeit ermöglicht (siehe Abschnitt 5.2).

Nachfolgend finden Sie einige Links zu Blogs und Tutorials über Enterprise Event Enablement mit Anleitungen dazu, wie Sie ihre eigenen Event Consumer und Ihre eigenen Business Events entwickeln können:

- An Introduction to Enterprise Event Enablement for SAP BTP ABAP Environment (SAP Blogs): <http://s-prs.co/v567509>
- Consume an Event Using Event Consumption Model (SAP Blogs): <http://s-prs.co/v567510>
- Event Consumption in ABAP Development Tools for Eclipse (Tutorials for SAP Developers): <http://s-prs.co/v567511>
- How to Create RAP Business Events in SAP BTP ABAP Environment (SAP Blogs): <http://s-prs.co/v567512>

Diese Blogs und Tutorials basieren zwar auf der ABAP-Umgebung der SAP BTP, sind aber auch auf die In-Stack-Entwicklererweiterbarkeit in der ABAP-Umgebung von SAP S/4HANA Cloud anwendbar.

#### 6.7.4 Event-Kanäle und Topic-Filter

Mit Enterprise Event Enabling können Sie mehrere *Kanäle* für Business Events anlegen. Dank der Möglichkeit, mehrere Event-Kanäle zu definieren, kann dasselbe SAP S/4HANA-System mit unterschiedlichen SAP-Event-Mesh-Tenants verbunden werden, oder auch mehrfach mit demselben, mit jeweils unterschiedlichen Konfigurationen. Die Event-Kanäle funktionieren in beide Richtungen und können sowohl zum Senden als auch zum Empfangen von Events verwendet werden.

Zu den Attributen eines Kanals gehören die HTTP-Destination für den Endpunkt in SAP Event Mesh, die OAuth-Anmeldeinformationen des für die Verbindung verwendeten Messaging-Clients und einige Kommunikationsparameter wie die Servicequalität (ob die Nachrichtenübermittlung garantiert ist oder nicht), die Anzahl der Versuche für einen erneuten Verbindungsaufbau und die Wartezeit bis zu einem erneuten Verbindungsaufbau. Ein weiteres wichtiges Attribut eines Kanals ist sein *Topic-Namensraum* (Topic Space), ein kanalspezifisches Präfix, das den Topics aller Events vorangestellt wird, die in diesem Kanal veröffentlicht werden.

Für jeden Kanal müssen Sie angeben, welche Events über diesen Kanal veröffentlicht oder konsumiert werden sollen. Hierzu definieren Sie einen oder mehrere *Topic-Filter* für den Kanal. Wenn Enterprise Event Enabling ein ausgehendes Event von der RAP-Laufzeitumgebung erhält, veröffentlicht es das Event über alle Kanäle, die über einen passenden Topic-Filter verfügen.

Für eingehende Events können Sie Event-Kanäle in SAP S/4HANA auf Queues in SAP Event Mesh subscribieren. Das bedeutet, dass SAP S/4HANA Events aus diesen Queues von SAP Event Mesh empfangen wird. Um die eingehenden Events innerhalb von SAP S/4HANA weiter zu verteilen, müssen bei den Event-Kanälen sogenannte *eingehende Topic Bindings* angelegt werden. Dabei handelt es sich um Topic-Filter, die



festlegen, welche eingehenden Events der Kanal akzeptiert. Wenn Enterprise Event Enablement ein Event von einer Queue in SAP Event Mesh empfängt, leitet es das Event an jeden Kanal weiter, der auf die Queue subskribiert ist und ein eingehendes Topic-Binding hat, das dem Topic des Events entspricht. Das eingehende Event wird dann an die Anwendungen weitergeleitet, die dafür als Event Consumer registriert sind.

Für die On-Premise-Edition von SAP S/4HANA müssen die Kanäle und die HTTP-Destination von einem Administrator oder einer Administratorin angelegt werden. In SAP S/4HANA Cloud erfolgt dieser Vorgang auf der Grundlage von Kommunikationsszenarien für die Integration mit Business Events. Sie müssen ein Kommunikationssystem anlegen, das die Details für die Verbindung zu SAP Event Mesh enthält. Darüber hinaus müssen Sie einen Kommunikationsbenutzer für den eingehenden Event-Discovery-Service anlegen, den SAP Event Mesh aufruft, um den Event-Katalog abzurufen. Beim Anlegen der Kommunikationsvereinbarung müssen Sie den Namen des Kanals angeben. Optional können Sie noch zusätzliche Parameter angeben. Die HTTP-Destination und der Kanal werden automatisch angelegt, wenn Sie die Kommunikationsvereinbarung aktivieren. Nachdem Sie die Kommunikationsvereinbarung angelegt haben, können Sie mithilfe einer SAP-Fiori-App die weitere Konfiguration für den Kanal vornehmen.

### 6.8 Datenintegration

Ziel der Prozessintegration ist die Implementierung verteilter Geschäftsprozesse. Der Nachrichtenaustausch dient typischerweise dazu, Prozessschritte über verschiedene Systeme oder Tenants hinweg zu verbinden und dann den nächsten Prozessschritt auf der Empfangsseite auszulösen. Das Anlegen eines Kundenauftrags etwa kann im empfangenden System mehrere verkaufsbezogene Prozessschritte wie eine Lieferung oder die Ausstellung einer Rechnung auslösen. Bei der *Datenintegration* hingegen steht nicht die Ausführung verteilter Prozesse im Vordergrund. Hier werden die Daten für verschiedene Zwecke über einen generischen Mechanismus übertragen, etwa für Analytics, Machine Learning oder einfach zur Synchronisierung zweier Datenspeicher in einem Replikationsszenario. Wenn Kundenaufträge in ein Data Warehouse oder einen Data Lake repliziert werden, lösen sie keine Geschäftsprozessschritte wie etwa Verkaufs- oder Lieferaktivitäten aus.

Die wesentlichen Infrastrukturen für die Datenintegration – nämlich die CDS-gestützte Datenextraktion, das Datenreplikations-Framework (DRF) und SAP Master Data Integration – werden wir in den folgenden Abschnitten erörtern.

### 6.8.1 CDS-basierte Datenextraktion

*CDS-basierte Datenextraktion* ist die strategische Technologie für die Datenintegration für Zwecke wie Analytics und Machine Learning. Dieser Abschnitt gibt einen Überblick über die CDS-basierte Datenextraktion in SAP S/4HANA, die in On-Premise- und Cloud-Umgebungen verwendet werden kann. Sie kommt beispielsweise zum Einsatz, wenn es um das Laden von Daten aus SAP S/4HANA in SAP Business Warehouse (SAP BW) und in SAP Data Intelligence geht. Die CDS-basierte Datenextraktion ist ein schönes Beispiel dafür, wie CDS-Views und das virtuelle Datenmodell (VDM) über die gesamte Architektur von SAP S/4HANA hinweg an verschiedenen Stellen und für verschiedene Zwecke verwendet werden. CDS-Extraktions-Views definieren ein generisches Modell für die Datenextraktion, das über verschiedene Kanäle für die verschiedenen Datenintegrationsszenarien genutzt werden kann.

#### Überblick über die CDS-basierte Extraktion

Die CDS-basierte Datenextraktion wird auf der Grundlage von CDS-Views implementiert, die mit speziellen CDS-Annotationen (`@Analytics.dataExtraction.enabled`) als Extraktions-Views gekennzeichnet sind. Bei diesen Extraktions-Views handelt es sich in der Regel um Basic oder Composite Views im VDM (siehe Abschnitt 2.1.3). Während der Extraktion werden die aus den CDS-Extraktions-Views generierten Datenbank-Views verwendet, um die extrahierten Daten auszulesen.

Die Extraktion kann nach zwei verschiedenen Verfahren erfolgen: Bei der *vollständigen Extraktion* werden alle verfügbaren Daten aus den Extraktions-Views extrahiert. Bei der *Delta-Extraktion* werden nach einer initialen vollständigen Extraktion später nur noch die jeweiligen Änderungen extrahiert. Dies setzt voraus, dass die CDS-Extraktions-Views als *deltafähig* gekennzeichnet sind (`@Analytics.dataExtraction.delta.*`).

Abbildung 6.6 zeigt die Architektur der CDS-basierten Datenextraktion für die Datenintegration mit SAP BW und mit SAP Data Intelligence. Es gibt mehrere APIs, über die externe Systeme Daten auf der Grundlage von CDS-Views extrahieren können:

- Das Framework *Operational Data Provider* (ODP) ist Teil der Infrastruktur für Analytics und für die Extraktion von operativen Daten aus SAP-Geschäftssystemen wie SAP S/4HANA. Neben vielen anderen Funktionen unterstützt es auch die Datenextraktion mit den entsprechenden APIs und Extraktions-Queues. In einem On-Premise-Szenario können Systeme wie SAP BW einen Remote Function Call (RFC) aufrufen, der durch das ODP-Framework in SAP S/4HANA bereitgestellt wird. Dieselbe API kann in SAP S/4HANA Cloud über einen SOAP-Service verwendet werden, der im Wesentlichen den RFC kapselt.
- Die CDS-basierte Datenextraktion ist auch über das API *Cloud Data Integration* (CDI) möglich. CDI ist eine auf offenen Standards basierendes Datenextraktions-

API und unterstützt die Datenextraktion über OData-Services. Wie Abbildung 6.6 zeigt, basiert diese Option ebenfalls auf dem ODP-Framework. Der *CDI-Consumer-Operator* in SAP Data Intelligence kann verwendet werden, um Daten aus SAP S/4HANA über das CDI-OData-API zu extrahieren.

- Der *ABAP-CDS-Reader-Operator* ist eine alternative Option zum Extrahieren von Daten nach SAP Data Intelligence. Er verwendet RFC-Aufrufe über WebSocket, um Daten aus der *ABAP Pipeline Engine* innerhalb des SAP-S/4HANA-Systems zu extrahieren. Die ABAP Pipeline Engine ist in der Lage, ABAP-Operatoren auszuführen, die in Daten-Pipelines in SAP Data Intelligence verwendet werden, so auch den ABAP-CDS-Reader-Operator.

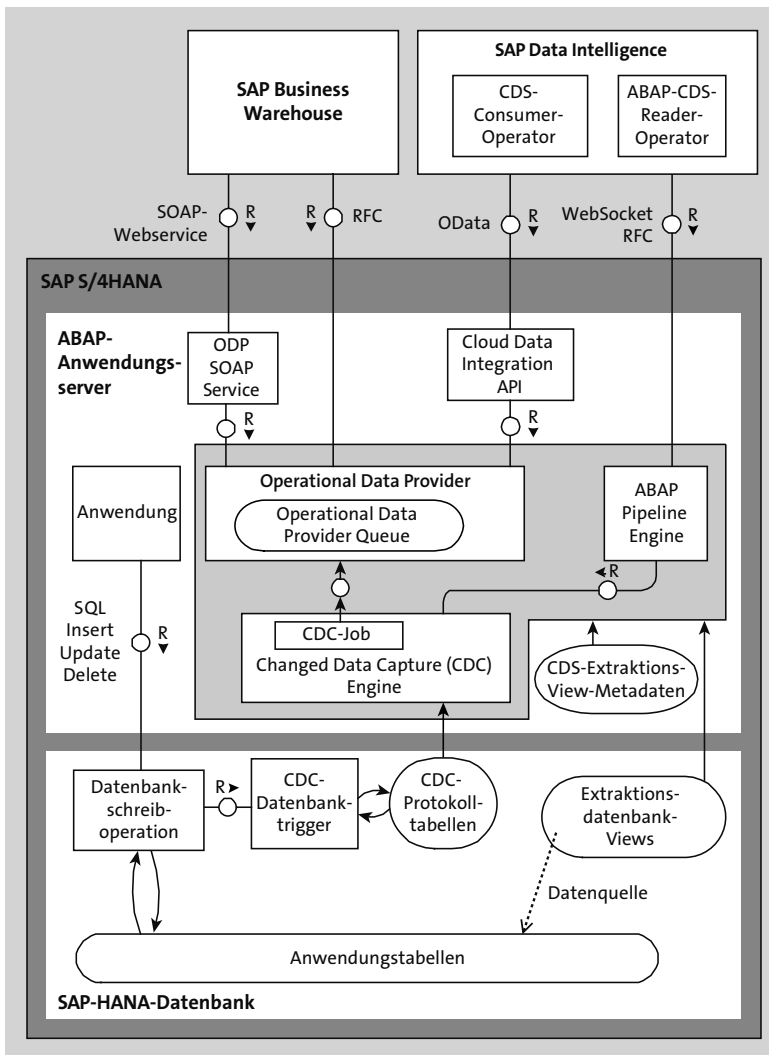


Abbildung 6.6 CDS-basierte Datenextraktion

Für die vollständige Extraktion lesen das ODP-Framework und die ABAP Pipeline Engine alle aktuell verfügbaren Datensätze aus den Extraktions-Views aus.

### Delta-Extraktion mit der Changed Data Capture Engine

Für die Delta-Extraktion greift die in Abbildung 6.6 dargestellte Architektur auf die Trigger-basierte *Datenänderungserfassung* (Changed Data Capture, CDC) zurück, ein Ansatz, der schon für SAP Landscape Transformation benutzt wird. In SAP S/4HANA übernimmt die *Changed Data Capture Engine* diese Aufgabe. Bei der Trigger-basierten Delta-Extraktion werden für die Datenbanktabellen, die den Extraktions-Views zugrunde liegen, Datenbank-Trigger angelegt. Das führt dazu, dass bei jeder Erstellung, Löschung oder Änderung von Datensätzen in diesen Tabellen entsprechende Informationen über die Änderungen in entsprechende Protokolltabellen geschrieben werden. Wie bereits erwähnt, müssen die CDS-Extraktions-Views mit entsprechenden Annotationen versehen werden, um anzuzeigen, dass sie diese Option unterstützen. Bei komplexen, auf mehreren Tabellen basierenden Views müssen die Entwickler\*innen als Teil der Definition des CDS-Extraktions-Views auch Informationen über das Mapping zwischen den Views und den Tabellen bereitstellen. In einfachen Fällen kann das System das Mapping automatisch machen. Die Delta-Extraktion mit der Changed Data Capture Engine ist jedoch mit einigen Einschränkungen verbunden. Unter anderem dürfen die Views nicht zu komplex sein. So dürfen delta-fähige Extraktions-Views beispielsweise keine aggregierten Daten enthalten und nicht zu viele Joins.

Die Datenänderungserfassung für die CDS-basierte Delta-Extraktion funktioniert wie folgt: In regelmäßigen Abständen wird ein CDC-Job ausgeführt, der alle geänderten Daten aus den Anwendungstabellen extrahiert. Für die Extraktion entnimmt die Changed Data Capture Engine den Protokolltabellen die Information, was geändert wurde, und liest dann die geänderten Daten mithilfe der aus den CDS-Extraktions-Views generierten Datenbank-Views. Der CDC-Job schreibt die extrahierten Daten in die ODP-Queue des ODP-Frameworks, aus der sie wie zuvor beschrieben über APIs abgerufen werden können.

Wie bereits erläutert, basiert die Cloud-Data-Integration-API auf dem ODP-Framework und verwendet dementsprechend denselben Mechanismus für die Delta-Extraktion. Die Delta-Extraktion wird auch von der ABAP-Pipeline unterstützt. Für die Delta-Extraktion ruft sie die Change Data Capture Engine auf, die dann die Daten mithilfe der Protokolltabellen aus den aus den CDS-Extraktions-Views generierten Datenbank-Views liest.

Vor der Einführung der Trigger-basierten Datenänderungserfassung war die Delta-Extraktion nur auf der Grundlage von Zeitstempeln möglich. Dementsprechend war im Datenmodell ein Zeitstempelfeld erforderlich, um die nach einem bestimmten

Zeitpunkt erfolgten Änderungen ermitteln zu können. Bei der Trigger-basierten Erfassung von Datenänderungen ist dies nicht erforderlich. Darüber hinaus erfordert die zeitstempelbasierte Delta-Extraktion während der Laufzeit mehr Aufwand, insbesondere bei der Erkennung von Löschungen. Die CDS-basierte Delta-Extraktion mit Zeitstempeln wird durch das OPD-Framework implementiert. Diese Option wird immer noch unterstützt; für neuere Entwicklungen empfiehlt SAP jedoch die Trigger-basierte Datenänderungserfassung.

### 6.8.2 Datenreplikations-Framework

Das Datenreplikations-Framework (DRF) reagiert auf lokale Änderungen an Business-Objekten und entscheidet dann, welche Business-Objektinstanzen in bestimmte Zielsysteme repliziert werden sollen. Das Framework ist in ABAP implementiert und bildet einen Teil des Softwarestacks von SAP S/4HANA. Seine Architektur ist in Abbildung 6.7 dargestellt. Geschäftsanwendungen in SAP S/4HANA können lokale Änderungs-Events für ihre Business-Objekte im DRF registrieren und spezifizieren, mit welchen ausgehenden Schnittstellen die Daten der Business-Objekte jeweils verteilt werden sollen. Innerhalb einer produktiven SAP-S/4HANA-Landschaft können Integrationsexpert\*innen ein Replikationsmodell definieren, das die Filterbedingungen und die Zielsysteme für die Replikation für ein bestimmtes Business-Objekt festlegt.

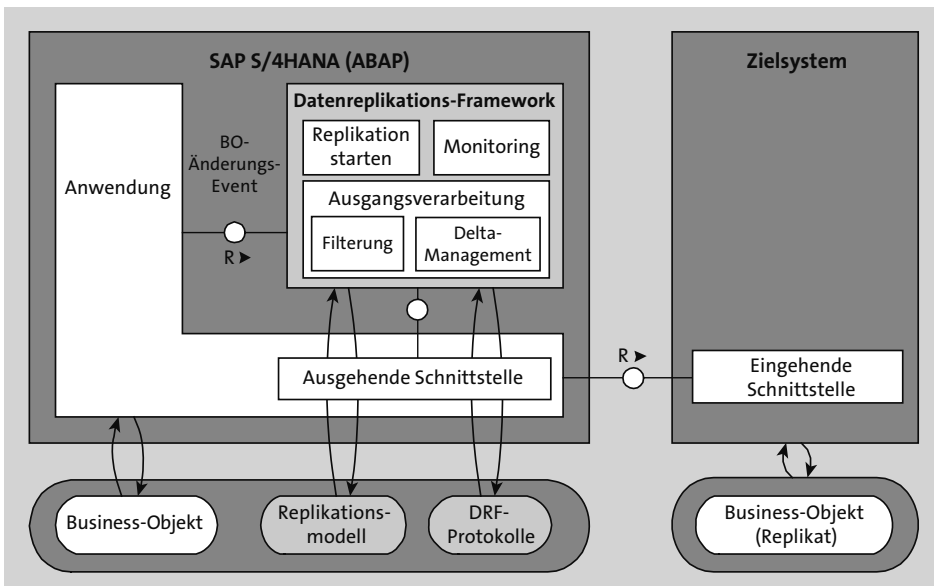


Abbildung 6.7 Datenreplikations-Framework

Zur Laufzeit informiert die Anwendung das DRF durch ein Event darüber, dass ein Business-Objekt angelegt oder geändert wurde, z. B. ein Kundenauftrag. Das DRF

prüft dann die Filterbedingungen für die Kundenaufträge. Wenn es zu dem Ergebnis kommt, dass dieser Kundenauftrag repliziert werden soll, sendet es seine Daten über die zuvor spezifizierte ausgehende Schnittstelle (etwa einen SOAP-Service) an das definierte Zielsystem oder die definierten Zielsysteme.

Dabei werden immer vollständige Instanzen der Business-Objekte versendet. Die Replikation von Daten erfolgt nach dem Push-Prinzip, ausgelöst durch ein Änderungs-Event im Quellsystem, was dann die Übertragung in die Zielanwendungen zur Folge hat, beispielsweise durch Aufruf eines SOAP-Service. Das DRF wird durch das Key Mapping Framework und das Value Mapping Framework ergänzt. Diese werden benötigt, wenn in Quell- und Zielsystemen unterschiedliche Schlüssel (IDs) für dieselbe Instanzen verwendet werden oder wenn Codelisten nicht systemübergreifend harmonisiert sind.

SAP S/4HANA verwendet das DRF auch als Trigger für die Stammdatenverteilung mit SAP Master Data Integration, auf die wir im nächsten Abschnitt eingehen werden.

### 6.8.3 SAP Master Data Integration

In der Cloud werden Geschäftsprozesse auf einer verteilten, modularen Architektur ausgeführt, zu der die benötigten Funktionalitäten von verschiedenen, verteilten Geschäftsanwendungen und -services beigetragen werden. Voraussetzung dafür sind der Austausch und die Synchronisation von gemeinsamen Stammdatenobjekten zwischen diesen Anwendungen und Services. *SAP Master Data Integration* ist ein Multi-Tenant-fähiger Cloud-Service zur Stammdatenintegration, der eine konsistente Sicht auf die Stammdaten in einer hybriden Landschaft ermöglicht. Technisch gesehen ist SAP Master Data Integration eine generische Engine zur Verteilung von Stammdaten. Die Stammdaten werden zunächst in einer Anwendung oder einem Service lokal erzeugt, gespeichert und verwaltet. Diese bzw. dieser nutzt SAP Master Data Integration, um die Stammdatenobjekte und deren Aktualisierungen zu verteilen. SAP Master Data Integration führt dabei keine anwendungsspezifische Logik aus, sondern nur einige grundlegende Validierungen der eingehenden Daten, z. B. eine Validierung des Datenschemas. Die Datenmodelle der Stammdaten in SAP Master Data Integration folgen dem *SAP One Domain Model*, dem von SAP definierten Framework für abgestimmte Datenmodelle, die anwendungsübergreifend verwendet werden.

Beim Anlegen oder Ändern von Stammdaten ruft die Geschäftsanwendung die Change-Request-API von SAP Master Data Integration auf. SAP Master Data Integration validiert die eingehenden Daten und schreibt für die akzeptierten Änderungen entsprechende Change-Events in ein Protokoll. Anwendungen, die sich für Änderungen an bestimmten Stammdatenobjekten interessieren, können diese über die

Change-Event-API von SAP Master Data Integration abfragen. Die Anwendungen können dabei mit entsprechenden Filtern spezifizieren, welche Stammdaten sie erhalten. Die Filterung kann auf der Instanzebene (zur Eingrenzung welche Stammdatensätze von Interesse sind) und sogar auf der Ebene der einzubeziehenden Felder erfolgen.

SAP S/4HANA nutzt SAP Master Data Integration in mehreren End-to-End-Prozessen, um verschiedene Arten von Stammdaten zwischen verschiedenen SAP-Anwendungen zu synchronisieren. Im End-to-End-Prozess Source-to-Pay werden mittels SAP Master Data Integration beispielsweise Lieferanten- und Kostenstellenstammdaten aus SAP S/4HANA Cloud nach SAP Ariba repliziert. Ein weiteres Beispiel ist die Integration zwischen SAP S/4HANA und SAP SuccessFactors Employee Central. Hier dient SAP Master Data Integration zum Übermitteln von Personalstammdaten, Stellenklassifikationen und Organisationseinheiten aus SAP SuccessFactors Employee Central auf der einen Seite sowie zum Übermitteln von Kostenstellenstammdaten aus SAP S/4HANA auf der anderen Seite. Über SAP Master Data Integration können viele weitere Arten von Stammdaten von oder nach SAP S/4HANA synchronisiert werden, z. B. Bankdaten, Stammdaten für das Public Sector Management, Projektcontrolling-Objekte und Geschäftspartner. Es ist geplant, die Liste der von SAP Master Data Integration unterstützten Stammdatenobjekte stetig zu erweitern. Einen Überblick, welche Integrations szenarien geplant sind, finden sie über den Road Map Explorer unter <https://roadmaps.sap.com/board>, wenn Sie nach dem Produkt »SAP S/4HANA Cloud« und dann weiter unter **Focus Topics** nach dem Schwerpunktthema »Integration« filtern.

### 6.9 Zusammenfassung

In diesem Kapitel haben wir die technischen Optionen zur Integration von SAP S/4HANA mit anderen Systemen oder Services untersucht. Wir haben einen Überblick über die verfügbaren Schnittstellentechnologien gegeben und den SAP Business Accelerator Hub als zentralen öffentlichen Katalog für APIs, Business Events und andere integrationsrelevante Inhalte vorgestellt. Als Nächstes haben wir die Konzepte für das Monitoring und die Fehlerüberwachung mithilfe des SAP Application Interface Framework besprochen. Wir haben das Kommunikationsmanagement in SAP S/4HANA Cloud vorgestellt, das das Konfigurieren der Kommunikation mit anderen Systemen oder Anwendungen vereinheitlicht und vereinfacht. Als Nächstes haben wir dargestellt, wie der Cloud Connector die Kommunikation zwischen Cloud- und On-Premise-Anwendungen ermöglicht.

Die SAP Integration Suite enthält mit Cloud Integration eine Middleware zur Prozessintegration, die insbesondere für die Integration von Drittanbietersystemen oder für hybride Integrationszenarien zwischen Cloud- und On-Premise-Komponenten verwendet werden kann. Daneben unterstützt SAP S/4HANA auch die Event-basierte Integration. Sie haben gesehen, wie SAP S/4HANA Business Events über SAP Event Mesh veröffentlichen und empfangen kann. Schließlich haben wir verschiedene Technologien zur Datenintegration vorgestellt.

Im nächsten Kapitel gehen wir auf die Anforderungen hinsichtlich Datenschutz und Privatsphäre und deren Umsetzung mithilfe von SAP S/4HANA ein.