

Inhaltsverzeichnis

MANAGEMENT SUMMARY (ABSTRACT)	3
1 EINLEITUNG	4
1.1 Abgrenzung.....	4
1.2 Ausblick.....	4
2 QDX-INTEGRATIONSPROZESS	6
2.1 Routing.....	6
2.1.1 Initiator Kunde.....	9
2.1.2 Initiator Lieferant	10
2.2 Vorgangsreferenz	11
2.3 Kommunikationstypen.....	13
2.3.1 Aktive Kommunikation.....	13
2.3.1.1 Initiator Kunde.....	13
2.3.1.2 Initiator Lieferant	15
2.3.2 Passive Kommunikation.....	16
2.3.2.1 Initiator Kunde.....	16
2.3.2.2 Initiator Lieferant	18
3 KOMMUNIKATIONSVERFAHREN	20
3.1 OFTP (Aktive Kommunikation)	20
3.1.1 Übertragung von QDX-Nachrichten per OFTP	21
3.1.1.1 Übertragung von QDX-Nachrichten ohne Attachments	22
3.1.1.2 Übertragung von QDX-Nachrichten mit Attachments	22
3.1.2 Übertragung von Beanstandungen per OFTP	23
3.1.3 Übertragung von 8D-Reports per OFTP	24
3.2 Webservices (Passive Kommunikation).....	25
3.2.1 Authentisierung	25
3.2.2 Abholen von Beanstandungen.....	25
3.2.3 Senden eines 8D-Reports.....	27
4 FORMAT DES DATENPAKETS	29
4.1 Header	30
4.2 Body.....	31
4.3 QDX-Envelope	31
4.3.1 QDXEnvelope (Aktive Kommunikation)	32
4.3.2 QDXEnvelopeRequest (Passive Kommunikation)	32
4.3.3 QDXEnvelopeResponse (Passive Kommunikation)	33
4.4 Neue QDX-Dokumente	33
4.4.1 QDX-Dokumente mit technischen Informationen.....	33
4.4.1.1 QDXAcknowledgeComplaint.....	34
4.4.1.2 QDXResetAcknowledgeStatusComplaint	34
4.4.1.3 QDXAcknowledgeReport8D.....	35
4.4.1.4 QDXComplaintList.....	36
4.4.2 QDX-Dokumente zur Anfrage von Informationen	36
4.4.2.1 QDXComplaintListRequest	36
4.4.2.2 QDXComplaintRequest.....	37

4.4.2.3	QDXAcknowledgeReport8DRequest	37
4.5	Attachments	38
4.6	Zusammenfassung	39
5	ANHANG	40
5.1	WSDL	40
5.2	Definition neuer XSD	43
5.2.1	QDXAcknowledgeComplaint	43
5.2.2	QDXAcknowledgeReport8D	44
5.2.3	QDXAcknowledgeReport8DRequest	45
5.2.4	QDXEnvelope	46
5.2.5	QDXEnvelopeRequest	46
5.2.6	QDXEnvelopeResponse	47
5.2.7	QDXComplaintList	48
5.2.8	QDXComplaintListRequest	49
5.2.9	QDXComplaintRequest	50
5.2.10	QDXResetAcknowledgeStatusComplaint	51
5.3	Statuscodes für QDXEnvelopeResponse	52
5.3.1	Statuscodes für den Erfolgsfall	52
5.3.2	Statuscodes für den Fehlerfall	52
5.4	Optionale individuelle Partnervereinbarungen	53
5.5	Glossar	53
5.6	Download	54

Management Summary (Abstract)

Nachdem die Inhalte der mittels QDX zu übertragenden XML-Dateien im eigentlichen QDX hinreichend beschrieben waren, wurde festgestellt, dass die direkte Kommunikation (der direkte Datenaustausch) zwischen den IT-Systemen zweier Geschäftspartner nur dann funktionieren kann, wenn darüber hinaus auch festgelegt wird, auf welchem Weg diese Dokumente ausgetauscht werden.

Im Sinne einer einheitlichen und zentralen Vorgabe wurde in einer Arbeitsgruppe, aus Mitarbeitern der am Ende dieses Vorworts aufgeführten Unternehmen ein Leitfaden / eine Empfehlung erarbeitet.

Der vorliegende Leitfaden (technische Spezifikation) bildet somit die technische Grundlage für die mit QDX beabsichtigte Standardisierung des Austauschs von Qualitätsdaten. Hiermit wird es möglich QDX-Schnittstellen zu realisieren, über die ohne weitere technische Anpassung mit allen Geschäftspartnern kommuniziert werden kann, sofern diese ebenfalls eine Schnittstelle entsprechend dieses Leitfadens implementiert haben.

Aus der Vielzahl möglicher Kommunikationswege haben sich neben dem klassischen Weg über OFTP und EDI die modernere Kommunikation über WebServices als einzig verbleibende Alternativen herauskristallisiert.

Im Verlauf der Arbeit an dem vorliegenden Leitfaden hat sich herausgestellt, dass weitere QDX-Dokumente notwendig sind, welche Informationen zur eigentlichen Datenübertragung beinhalten. Der QDX-Standard muss in Folge davon entsprechend erweitert werden. Die zusätzlichen QDX-Dokumente werden im Rahmen dieses Leitfadens inhaltlich beschrieben. Um die aktuellen QDX-Spezifikationsdokumente einsehen zu können, müssen Sie sich kurz registrieren. Dazu senden Sie bitte eine E-mail mit Ihren Kontakt- und Firmendaten an liebscher@vda-qmc.de. Nachdem Sie sich registriert haben, bekommen Sie innerhalb von 2 Werktagen eine E-Mail, die einen Link enthält, über den Sie sich die Spezifikationsdokumente downloaden können.

Registrierung und Download sind kostenfrei und mit keinerlei weiterführenden Verpflichtungen verbunden.

Die eigentlichen QDX-Nachrichten werden in Umschläge "gepackt" (vergleichbar mit einem Briefumschlag), um die eingehenden Nachrichten im jeweiligen Routing-System in die richtige Anwendung des Geschäftspartners leiten und Anhänge übertragen zu können. Bei beiden Kommunikationswegen werden die gleichen SOAP-Umschläge verwendet. Die Beschreibungen der von der Arbeitsgruppe festgelegten SOAP- und QDX-Umschläge erfolgt direkt im vorliegenden Leitfaden.

Abschließend sei darauf hingewiesen, dass die Geschäftspartner auch mit dem Vorliegen dieses Leitfadens noch zusätzliche, eigene Vorgaben definieren müssen, damit die Kommunikation funktioniert (z.B. welche Kommunikationsalternative wird eingesetzt, welche Alternativen sind möglich, welche der optionalen QDX-Datenfelder müssen zwingend übertragen werden; mit welchem Inhalt und bis zu welcher Datenfeldlänge usw.)

Das VDA-QMC dankt den Mitarbeitern der folgenden Unternehmen für Ihr persönliches Engagement und die geleistete Unterstützung:

- AUDI AG
- Axway
- Daimler AG
- Harman/Becker Automotive Systems GmbH
- iPoint-systems GmbH
- Mieschke Hofmann und Partner Gesellschaft für Management- und IT-Beratung mbH
- msg systems ag
- SEEBURGER AG
- SupplyOn AG
- T-Systems AG
- Volkswagen AG

Oberursel, Dezember 2008

Qualitäts-Management Center im Verband der Automobilindustrie e.V. (VDA)

1 Einleitung

Das vorliegende Dokument beschreibt den technischen Austausch von QDX-Dokumenten. Es soll in erster Linie dazu dienen, QDX-Schnittstellen in allen Unternehmen einheitlich zu implementieren. Hierzu wurden zwei Kommunikationsverfahren (OFTP und Webservice) ausgewählt, die zum Austausch von QDX-Dokumenten verwendet werden sollen. Des Weiteren soll dieses Dokument die Implementierung einer QDX-Schnittstelle transparenter gestalten und somit vereinfachen. Das Dokument ist wie folgt strukturiert:

- Kapitel 2 leitet in die Thematik ein, erläutert die Kommunikationstypen und betrachtet die Themen Routing und Vorgangsreferenz. Dieses Grundverständnis ist notwendig, um die Definition der Kommunikationsverfahren nachvollziehen zu können. Das Kapitel ist allgemein gehalten und für alle QDX-Prozesse anwendbar, lediglich die Beispiele beziehen sich auf den Beanstandungsprozess bzw. auf die Erstbemusterung.
- Kapitel 3 beschreibt die beiden Kommunikationsverfahren, die für den Austausch von QDX-Dokumenten definiert wurden. Weitere Kommunikationsverfahren sollen nicht verwendet werden. Zur Nachvollziehbarkeit empfiehlt es sich, Kapitel 2 vorher zu lesen. Das Kapitel beschreibt ausschließlich den Beanstandungsprozess um die QDX-Dokumente QDXComplaint und QDXReport8D.
- Kapitel 4 beschreibt den „Umschlag“, in dem die QDX-Dokumente transportiert werden. Der Umschlag beinhaltet die Informationen, die für das Routing benötigt werden. Das Kapitel beschreibt ausschließlich die QDX-Dokumente QDXComplaint und QDXReport8D (und die Dokumente, die zudem für den Austausch der beiden benötigt werden).
- Kapitel 5 beinhaltet die WSDL-Definition, XML-Schemata und Beispieldateien. Die WSDL-Definition und die XML-Schemata sollen die Implementierung eines Webservices transparenter und somit einfacher machen. Die Beispieldateien helfen bei der Erstellung der jeweiligen QDX-Dokumente.

1.1 Abgrenzung

Im Rahmen des vorliegenden Dokuments werden der QDX-Integrationsprozess und der damit verbundene technische Austausch von QDX-Dokumenten beschrieben. Die Beschreibung der Geschäftsprozesse ist nicht Gegenstand dieses Dokuments. Diese erste Version beschäftigt sich primär mit dem Beanstandungsprozess und den dazugehörigen QDX-Dokumenten QDXComplaint und QDXReport8D. Eine Erweiterung auf die anderen Qualitätsprozesse und derer QDX-Dokumente ist aber problemlos möglich.

Die Spezifikation basiert auf der QDX-Version 2.0, die aber zum Zeitpunkt der Erstellung der Spezifikation noch nicht fertig gestellt und veröffentlicht wurde. Die in Kapitel 4.4 beschriebenen QDX-Dokumente werden erst in die neue QDX-Version 2.0 einfließen, ebenso die Änderung der Referenzierung von Attachments – wie in Kapitel 4.5 beschrieben. Eine Umsetzung dieser Spezifikation unter Verwendung von QDX-Version 1.2 ist aber grundsätzlich möglich. Hierbei ist zu beachten, dass die neuen QDX-Dokumente aus Kapitel 4.4 noch nicht im QDX-Standard existieren. Darüber hinaus ist für die Referenzierung von Attachments ein Workaround notwendig¹.

Die Themen Signierung, Security, Monitoring und Priorisierung sind nicht Bestandteil der Spezifikation. Signierung und Security sollen zu einem späteren Zeitpunkt noch einmal betrachtet werden, zurzeit besteht aber keine Anforderung aus dem Geschäftsprozess heraus. Für Priorisierung besteht ebenfalls keine Anforderung. Das Monitoring des Datenaustausches ist dagegen Aufgabe der Systeme, die die QDX-Dokumente austauschen. Daher wird hier auf eine Beschreibung verzichtet.

1.2 Ausblick

Dieses Dokument bezieht sich in der ersten Version nur auf den Beanstandungsprozess. Dies soll zunächst einmal in der Praxis verwendet werden. In einer zweiten Version sollen die Erfahrung aus der Praxis einfließen und die Definitionen entsprechend überarbeitet werden.

¹ Für die Content-ID muss das Feld „URL“ verwendet werden.

Diese Spezifikation kann aber auch ohne Weiteres von den weiteren in QDX beschriebenen Qualitätsprozessen adaptiert werden. Die Struktur der QDX-Dokumente und die Funktionsweise der Webservice-Methoden sind lediglich auf die Bezeichnungen der jeweiligen QDX-Dokumente anzupassen. Es ist aber darauf zu achten, wer der Initiator des Prozesses (siehe Kapitel 2.1) ist.

Der Lebenszyklus von IT-Technologien und IT-Standards wird immer kürzer. Das vorliegende Dokument hat sich zum Ziel gesetzt, diejenige Kommunikationstechnologie zu beschreiben, die sich in der (Automotive-) Branche durchgesetzt hat. Aus diesem Grund kann die Spezifikation um neue Kommunikationstechnologien erweitert werden – allerdings unter der Maßgabe, die Anzahl der beschriebenen Kommunikationstechnologien möglichst gering zu halten. Um die Erweiterung um weitere Kommunikationstechnologien zu erleichtern, wurde in Kapitel 2 auf eine Beschreibung einer konkreten Kommunikationstechnologie verzichtet. Stattdessen wurden hier die beiden möglichen Kommunikationstypen beschrieben.

2 QDX-Integrationsprozess

Der QDX-Integrationsprozess soll die CAQ-Systeme der Kunden und der Lieferanten miteinander verknüpfen, sodass beim Austausch von Informationen zwischen Geschäftspartnern keine Medienbrüche entstehen. Der QDX-Integrationsprozess beschreibt daher die Kommunikation zwischen den CAQ-Systemen der Geschäftspartner. Zunächst einmal ist zwischen den Initiatoren des jeweiligen Geschäftsprozesses zu unterscheiden, denn davon ist abhängig, welcher der beiden Geschäftspartner Informationen für das Routing oder für das Mapping auf einen Vorgang speichern muss².

Initiator des Geschäftsprozesses: Der Initiator initiiert den Geschäftsprozess und wartet auf eine Antwort seines Geschäftspartners. Sowohl Kunde als auch Lieferant können der Initiator des Geschäftsprozesses sein:

- **Initiator Kunde:** Ist der Kunde der Initiator des Geschäftsprozesses (z.B. beim Beanstandungsprozess), so muss der Lieferant eine Antwort (8D-Report) auf den Vorgang (Beanstandung) des Kunden an das Kundensystem senden. Dazu muss der Lieferant die Vorgangsnummer und die Identifikation des Kundensystems angeben².
- **Initiator Lieferant:** Ist der Lieferant wie z.B. bei der Erstbemusterung der Initiator des Geschäftsprozesses, so muss der Kunde eine Antwort (Produktionsprozess- und Produktfreigabe – PPF) auf den Vorgang (Produktionsprozess- und Produktfreigabe Bericht – PPF-Bericht) des Lieferanten an das Lieferantensystem senden. Dazu muss der Kunde die Vorgangsnummer und die Identifikation des Lieferantensystems angeben².

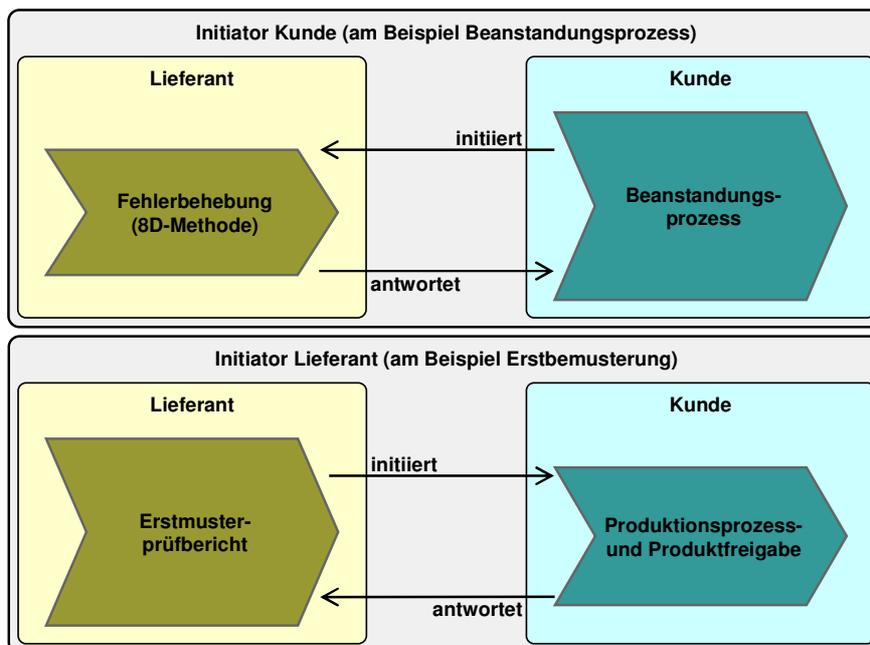


Abbildung 2-1: Initiator des Geschäftsprozesses

2.1 Routing

Die Kommunikation zwischen den CAQ-Systemen der beiden Geschäftspartner gestaltet sich einfach, wenn beide CAQ-Systeme direkt miteinander kommunizieren. Allerdings sind sogenannte Routing-Systeme gerade bei großen Unternehmen üblich, die die Kommunikation zwischen internen und externen Systemen steuern. Dies hat zur Folge, dass die Kommunikation wesentlich komplexer wird. Zum Beispiel benötigen diese Routing-Systeme

² Welche Informationen von einem Geschäftspartner beim Datenaustausch anzugeben sind, können in den Kapiteln 2.1 (Routing) und 2.1.2 (Vorgangsreferenz) nachgelesen werden.

Informationen, anhand derer sie die QDX-Dokumente weiterleiten können. Nachfolgend werden kurz die möglichen Szenarien erläutert:

Szenario 1: Beide CAQ-Systeme kommunizieren direkt miteinander. Dies ist der einfachste Fall, da hierbei keine weiteren Informationen notwendig sind.

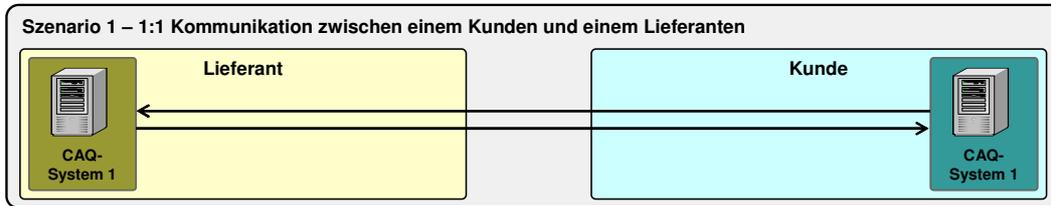


Abbildung 2-2: Szenario 1 - 1:1 Kommunikation zwischen einem Kunden und einem Lieferanten

Szenario 2: Auf Kundenseite wird die Kommunikation zwischen den internen und externen Systemen durch ein Routing-System gesteuert. Zum Versenden der QDX-Dokumente an den Lieferanten wird die Lieferantenummer benötigt. Um eingehende QDX-Dokumente an das richtige CAQ-System weiterleiten zu können, wird mindestens die Standortbezeichnung (KD-Nr.) und i. d. R. eine Systemidentifikation benötigt.

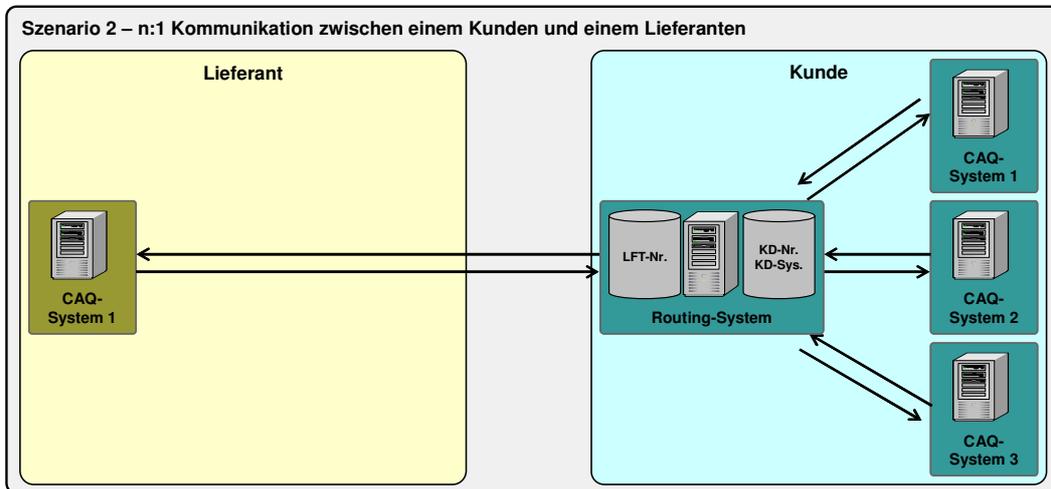


Abbildung 2-3: Szenario 2 - n:1 Kommunikation zwischen einem Kunden und einem Lieferanten

Szenario 3: Analog zu Szenario 2 wird hier auf Lieferantenseite die Kommunikation zwischen den internen und externen Systemen durch ein Routing-System gesteuert. Um beim Empfang von QDX-Dokumenten diese an das richtige CAQ-System weiterleiten zu können, wird mindestens die Lieferantennummer und evtl. eine Systemidentifikation benötigt. Zum Versand von QDX-Dokumenten vom Lieferanten an den Kunden wird lediglich die Standortbezeichnung (KD-Nr.) benötigt.

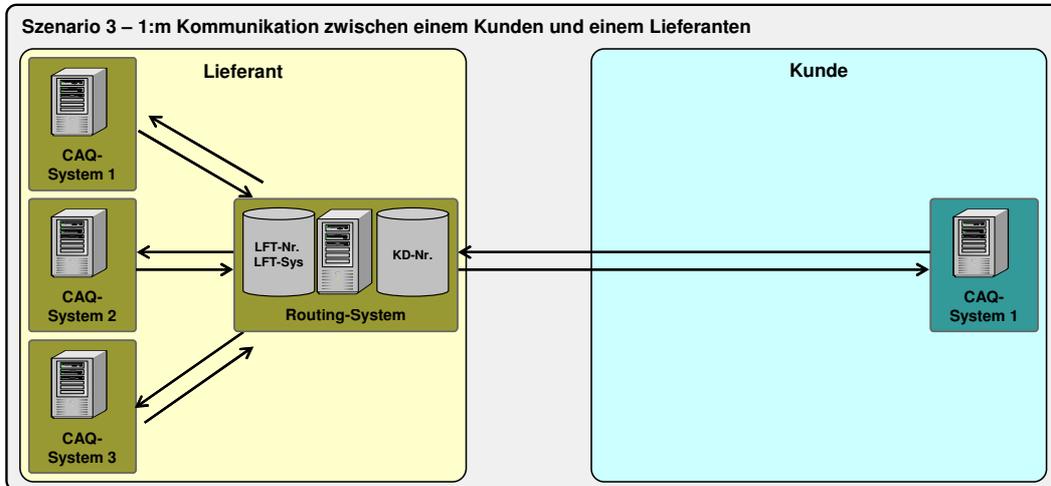


Abbildung 2-4: Szenario 3 - 1:m Kommunikation zwischen einem Kunden und einem Lieferanten

Szenario 4: Dieses Szenario stellt eine Mischung der Szenarien 2 und 3 dar – beide Geschäftspartner setzen ein Routing-System zur Steuerung der Kommunikation ein. Für ausgehende Dokumente ist die Nummer des Geschäftspartners (KD- bzw. LFT-Nr.) ausreichend, für eingehende Dokumente wird i. d. R. zusätzlich eine Systemidentifikation benötigt.

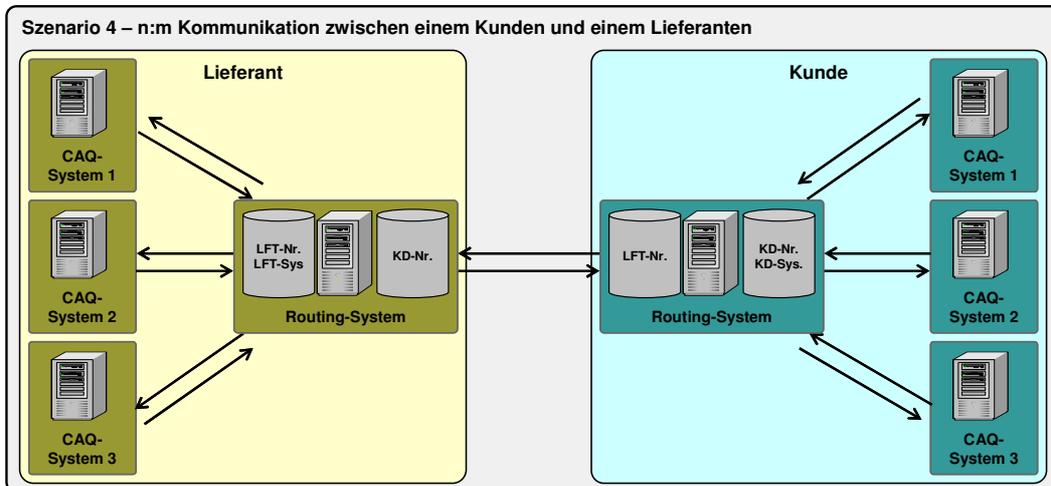


Abbildung 2-5: Szenario 4 - n:m Kommunikation zwischen einem Kunden und einem Lieferanten

Aufgrund der Komplexität mancher Szenarien müssen bei der Datenübertragung zusätzliche Informationen angegeben werden, anhand derer Routing-Systeme entscheiden können, welches Zielsystem der korrekte Empfänger der Daten ist. In Abhängigkeit vom Initiator des jeweiligen Geschäftsprozesses müssen die Geschäftspartner unterschiedliche Informationen angeben.

In Tabelle 2-1 werden die Informationen definierter, die für das Routing benötigt werden. Die Verwendung der Informationen wird in den folgenden Kapiteln beschrieben.

Bezeichnung	Beschreibung
KD-Nr.	Kundennummer: Identifikation des Kunden. Es handelt sich dabei um die Nummer, die der

Bezeichnung	Beschreibung
	Kunde für sich selbst führt.
KD-Sys.	Kundensystem: Identifikation des CAQ-Systems des Kunden. Hier kann der Kunde eine frei wählbare Bezeichnung angeben.
LFT-Nr.	Lieferantennummer: Identifikation des Lieferanten. Es handelt sich dabei um die Nummer, die der Kunde für den Lieferanten vergeben hat.
LFT-Sys.	Lieferantensystem: Identifikation des CAQ-Systems des Lieferanten. Hier kann der Lieferant eine frei wählbare Bezeichnung angeben.

Tabelle 2-1: Beschreibung der Routing-Informationen

2.1.1 Initiator Kunde

Ist der Kunde der Initiator des Geschäftsprozesses, so ist die Lieferantennummer ausreichend, um die initiale Nachricht im Routing-System des Kunden korrekt weiterzuleiten. Das Routing-System auf Lieferantenseite benötigt aber unter Umständen weitere Informationen, da beispielsweise in einem Werk mehrere CAQ-Systeme im Einsatz sein können. Um den hohen Pflegeaufwand von Systemnummern aller Lieferanten auf Kundenseite zu vermeiden, wurde definiert, dass die Angabe der Kundennummer, der Systemnummer des Kunden und der Lieferantennummer für das Routing-System des Lieferanten ausreichend sein muss³. Andernfalls ist der Lieferant dafür verantwortlich, dass die Nachricht in das korrekte System weitergeleitet wurde.

Sendet der Lieferant in seiner Rolle als Geschäftspartner eine Antwort zurück, so ist die Kundennummer ausreichend, um die Nachricht in seinem Routing-System korrekt weiterzuleiten. Das Routing-System auf Kundenseite benötigt aber unter Umständen ebenfalls weitere Informationen, da auch hier an einem Standort mehrere CAQ-Systeme im Einsatz sein können. Im Unterschied zu vorher handelt es sich nun aber um eine Antwort auf einen Vorgang des Kunden und der Kunde hat hierzu bereits Informationen an den Lieferanten gesendet. Beim Versenden der initialen Nachricht hat der Kunde nun die Möglichkeit, die Systemnummer seines Quellsystems anzugeben. Diese ist vom Lieferanten zu speichern und beim Senden seiner Antwort mit anzugeben. Dadurch erhält das Routing-System auf Kundenseite alle notwendigen Informationen, um die Antwort an das richtige CAQ-System weiterzuleiten.

³ Es müssen aber nicht alle Informationen für das Routing verwendet werden.

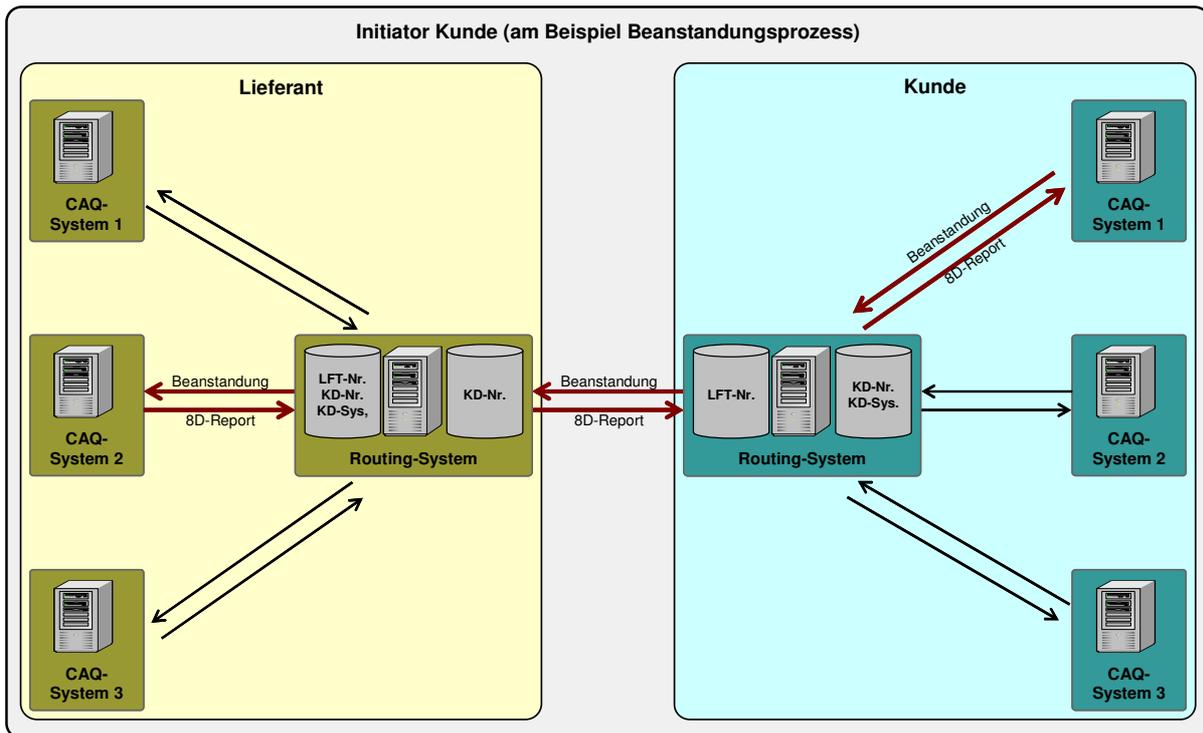


Abbildung 2-6: Darstellung des Routings, Kunde ist Initiator⁴

Bezeichnung	Beschreibung
KD-Nr.	Kundennummer: Identifikation des Kunden. Es handelt sich dabei um die Nummer, die der Kunde für sich selbst führt.
KD-Sys.	Kundensystem: Identifikation des CAQ-Systems des Kunden. Hier kann der Kunde eine frei wählbare Bezeichnung angeben. Kann vom Kunden optional verwendet werden. Wenn vom Kunden verwendet, dann ist die Angabe dieser Information für den Lieferanten bei einer Rückantwort verpflichtend.
LFT-Nr.	Lieferantennummer: Identifikation des Lieferanten. Es handelt sich dabei um die Nummer, die der Kunde für den Lieferanten vergeben hat.

Tabelle 2-2: Verwendbare Routing-Informationen

2.1.2 Initiator Lieferant

Ist der Lieferant der Initiator des Geschäftsprozesses, so ist die Kundennummer ausreichend, um die Nachricht im Routing-System des Lieferanten korrekt weiterzuleiten. Das Routing-System auf Kundenseite benötigt aber unter Umständen weitere Informationen, da beispielsweise an einem Standort mehrere CAQ-Systeme im Einsatz sein können. Um den hohen Pflegeaufwand der Systemnummern aller Lieferanten auf Kundenseite zu vermeiden, wurde definiert, dass der Lieferant neben der Kundennummer auch die Systemnummer des Kunden angeben muss. Diese Informationen sind für das Routing-System des Kunden ausreichend, um die Nachricht an das korrekte CAQ-System weiterzuleiten.

Sendet der Kunde in seiner Rolle als Geschäftspartner eine Antwort zurück, so ist die Lieferantennummer ausreichend, um die Nachricht im Routing-System auf Kundenseite korrekt weiterzuleiten. Das Routing-System

⁴ Der Datentopf zu den eigenen CAQ-Systemen hin beschreibt die Routing-Informationen, die für das Routing vom Routing-System zu den CAQ-Systemen benötigt wird. Der Datentopf zu dem anderen Geschäftspartner hin beschreibt dagegen die Routing-Informationen, die für das Routing vom eigenen Routing-System zum Routing-System des Geschäftspartners benötigt werden.

auf Lieferantenseite benötigt aber unter Umständen ebenfalls weitere Informationen, da auch hier in einem Werk mehrere CAQ-Systeme im Einsatz sein können. Im Unterschied zu vorher handelt es sich nun aber um eine Antwort auf einen Vorgang des Lieferanten. Beim Versenden der initialen Nachricht hat der Lieferant die Möglichkeit, die Systemidentifikation des Quellsystems mit anzugeben. Diese ist vom Kunden zu speichern und beim Senden der Antwort mit anzugeben. Dadurch erhält das Routing-System auf Lieferantenseite alle notwendigen Informationen, um die Antwort an das richtige CAQ-System weiterzuleiten.

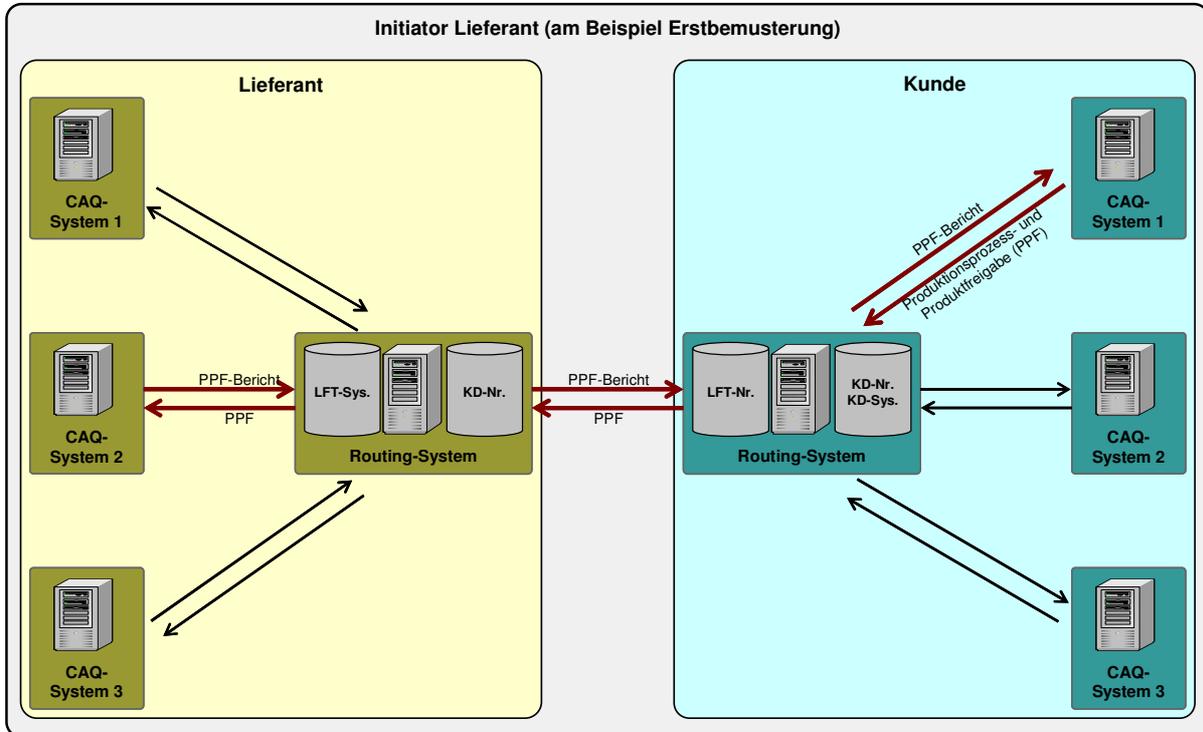


Abbildung 2-7: Darstellung des Routings, Lieferant ist Initiator⁴

Bezeichnung	Beschreibung
KD-Nr.	Kundennummer: Identifikation des Kunden. Es handelt sich dabei um die Nummer, die der Kunde für sich selbst führt.
LFT-Nr.	Lieferantennummer: Identifikation des Lieferanten. Es handelt sich dabei um die Nummer, die der Kunde für den Lieferanten vergeben hat.
LFT-Sys.	Lieferantensystem: Identifikation des CAQ-Systems des Lieferanten. Hier kann der Lieferant eine frei wählbare Bezeichnung angeben. Kann vom Lieferanten optional verwendet werden. Wenn vom Lieferanten verwendet, dann ist die Angabe dieser Information für den Kunden bei einer Rückantwort verpflichtend.

Tabelle 2-3: Beschreibung der Routing-Informationen

2.2 Vorgangsreferenz

Im Rahmen der von QDX unterstützten Qualitätsprozesse werden unterschiedliche QDX-Dokumente zwischen den Geschäftspartnern ausgetauscht. Hierbei handelt es sich meist um eine initiale Nachricht, die einen Prozess beim Geschäftspartner auslöst, sowie einer Antwort darauf, die den ausgelösten Prozess abschließt. Damit das empfangende System des Initiators des Geschäftsprozesses die Antwortnachricht zu dem richtigen Vorgang zuordnen kann, muss der andere Geschäftspartner Informationen angeben, die dem Initiator die Zuordnung ermöglicht.

Tabelle 2-4 beschreibt die Informationen, die der Geschäftspartner des Initiators speichern und bei einer Antwort mit angeben muss. Die ersten beiden Informationen sind allgemeingültig und daher immer zu speichern. Darüber

hinaus kann es erforderlich sein, weitere Informationen zu einem Vorgang zu speichern, da dieser aus Untervorgängen besteht – wie z. B. der Beanstandungsprozess oder die Erstbemusterung.

DocumentID	Identifiziert das vom Initiator gesendete Dokument (Beanstandung, PPF-Bericht etc.). Es ist die DocumentID zu verwenden, die der Initiator im Header des QDX-Dokuments angegeben hat.
RevisionID (optional verfügbar)	Identifiziert die Version des gesendeten Dokuments. Allerdings ist die Verwendung optional und dient zur Unterstützung der Kommunikation. Es ist die Versionsnummer zu verwenden, die der Initiator im Header des QDX-Dokuments angegeben hat.
RevisionDateTime	Der Zeitstempel der Version identifiziert zusammen mit der DocumentID eindeutig das gesendete Dokument. Anhand des Versionszeitstempels können Versionsstände der Dokumente genau identifiziert werden. Es ist der Zeitstempel zu verwenden, das der Initiator im Header des QDX-Dokuments unter RevisionDateTime angegeben hat.
ComplaintItemID (Beanstandungsprozess)	Identifiziert die einzelne Beanstandung beim Kunden und dient zur Unterscheidung der Positionen eines Prüfberichts (da meist die Prüfberichtsnummer als DocumentID angegeben wird). Es ist die ID zu verwenden, die der Initiator für das ComplaintItem vergeben hat.
BuyerProductItemID (Erstbemusterung)	Identifiziert das einzelne Teil beim Kunden (Teilenummer) und dient zur Unterscheidung der Positionen eines PPF-Berichts (ein PPF-Bericht kann mehrere Teile beinhalten). Die angegebene Teilenummer muss aber aus dem Nummernkreis des Kunden stammen.

Tabelle 2-4: Vom Geschäftspartner zu speichernde Vorgangsreferenzen

Die beschriebenen Elemente zur Identifizierung des Geschäftsdokuments müssen bei einer Rückantwort im Header des QDX-Dokuments unter „ReferenceDocument“ angegeben werden. Werden diese Informationen bei einer Antwort nicht mit angegeben, hat der Initiator keine Möglichkeit, die Antwort zu dem korrekten Vorgang zuzuordnen.

Darüber hinaus muss aber auch der Initiator des Geschäftsprozesses Informationen zur Rückantwort des Geschäftspartners speichern. Dies dient in erster Linie dazu, die erfolgreiche Verarbeitung einer erhaltenen Rückantwort qualifiziert bestätigen zu können. In Tabelle 2-5 werden diese Informationen detailliert beschrieben.

DocumentID	Identifiziert das vom Geschäftspartner gesendete Dokument (8D-Report, PPF etc.). Es ist die DocumentID zu verwenden, die der Geschäftspartner im Header des QDX-Dokuments angegeben hat.
RevisionID (optional verfügbar)	Identifiziert die Version des gesendeten Dokuments. Allerdings ist die Verwendung optional und dient zur Unterstützung der Kommunikation. Es ist die Versionsnummer zu verwenden, die der Initiator im Header des QDX-Dokuments angegeben hat.
RevisionDateTime	Der Zeitstempel der Version identifiziert zusammen mit der DocumentID eindeutig das gesendete Dokument. Anhand des Versionszeitstempels können Versionsstände der Dokumente genau identifiziert werden. Es ist der Zeitstempel zu verwenden, das der Initiator im Header des QDX-Dokuments unter RevisionDateTime angegeben hat.

Tabelle 2-5: Vom Initiator zu speichernde Vorgangsreferenzen

2.3 Kommunikationstypen

Für die Kommunikation ist zunächst einmal entscheidend, wer die Kommunikation initiiert. Hierbei wird zwischen zwei Kommunikationstypen unterschieden:

- **Aktive Kommunikation:** Derjenige Geschäftspartner ist der Initiator der Kommunikation, der Informationen an den anderen Geschäftspartner übermitteln möchte. Jeder Geschäftspartner übermittelt seine Informationen aktiv an den anderen Geschäftspartner und ist somit für den Aufbau der Kommunikation verantwortlich.
- **Passive Kommunikation:** Es ist aber auch möglich, dass diese Verantwortung nur von einem Geschäftspartner getragen wird und somit nur einer der beiden Geschäftspartner für den Aufbau der Kommunikation verantwortlich ist. Dabei nimmt der andere Geschäftspartner immer eine passive Rolle ein. Die passive Rolle kann aber nur vom Kunden angenommen werden, der Lieferant ist dagegen immer in der aktiven Rolle. Das bedeutet für den Lieferanten, dass er Informationen vom Kunden nur dann erhält, wenn er explizit beim Kunden nach neuen Informationen „fragt“. Tut er das nicht, erhält er keine neuen Informationen.

Der Kunde muss vor der Integration eines Geschäftsprozesses festlegen, ob er aktiv die Kommunikation aufbaut oder ob er sich im oben dargestellten Sinn immer passiv verhält. Die Festlegung ist für jeden Geschäftsprozess, in dem Qualitätsdaten über QDX ausgetauscht werden sollen, separat zu treffen. Innerhalb eines Geschäftsprozesses ist die Rolle der beiden Geschäftspartner konstant und darf sich nicht ändern. Die Festlegung erfolgt in einer individuellen Partnervereinbarung (vgl. Kapitel 5.4). Die Rolle des Lieferanten ist immer aktiv – im Gegensatz zu der des Kunden. Der Lieferant ist also immer in der Verantwortung, seine Informationen an den Kunden zu übermitteln.

Auf beide Kommunikationstypen (aktiv/passiv) wird in den beiden folgenden Kapiteln 2.3.1 (aktive Kommunikation) und 2.3.2 (passive Kommunikation) detailliert eingegangen.

Wie bereits in Kapitel 2.1 erwähnt werden in großen Unternehmen oft sogenannte Kommunikations- oder Routing-Systeme eingesetzt, die die Kommunikation zwischen internen und externen Systemen steuern. In diesem Falle werden die Daten vom Routing-System empfangen und anschließend an die Zielsysteme weitergeleitet; die Verarbeitung der Daten im Zielsystem erfolgt daher asynchron zum Empfang der Daten. Quittiert nun das empfangende System des Geschäftspartners den Erhalt der Daten, sagt dies nichts darüber aus, ob die gesendeten Daten auch im Zielsystem angekommen sind und dort verarbeitet werden konnten. Aus diesem Grund wird eine Verarbeitungsbestätigung eingeführt, mit der der Empfänger einer Nachricht den Erhalt und die korrekte Verarbeitung im Zielsystem bestätigt. Dies gewährleistet eine transparente Übertragung von Daten/Informationen.

2.3.1 Aktive Kommunikation

Der Initiator des Geschäftsprozesses initiiert die Kommunikation und sendet seine Informationen (Daten) an den Geschäftspartner. Nach erfolgreicher Verarbeitung im Zielsystem sendet der Geschäftspartner eine Verarbeitungsbestätigung an den Initiator zurück. Anschließend erfolgt die Beantwortung der gesendeten Daten. Die Antwort wird dann vom Geschäftspartner an den Initiator zurück gesendet. Der Initiator sendet nach erfolgreicher Verarbeitung wieder eine Verarbeitungsbestätigung an den Geschäftspartner zurück.

Bei der „aktiven“ Kommunikation müssen also beide Geschäftspartner die erzeugten Daten in eigener Verantwortung an den anderen Geschäftspartner übermitteln. Lieferant und Kunde sind damit beide in der „Bringschuld“.

2.3.1.1 Initiator Kunde

Am Beanstandungsprozess soll die Kommunikation zwischen den beiden Geschäftspartnern verdeutlicht werden, wenn der Kunde der Initiator des Geschäftsprozesses ist.

Der Kunde sendet die Beanstandungsdaten an den Lieferanten. Konnten die Beanstandungsdaten auch vom Zielsystem des Lieferanten korrekt verarbeitet werden, sendet dieses eine Verarbeitungsbestätigung zurück an

den Kunden. Dieser Schritt ist verpflichtend, damit der Kunde die Information erhält, dass die gesendeten Daten auch tatsächlich im Zielsystem angekommen sind und verarbeitet werden konnten (speziell bei asynchroner Verarbeitung ist dieser Schritt unerlässlich). Die empfangenden Systeme quittieren jeweils den Erhalt der Daten (Beanstandungsdaten bzw. Verarbeitungsbestätigung)⁵.

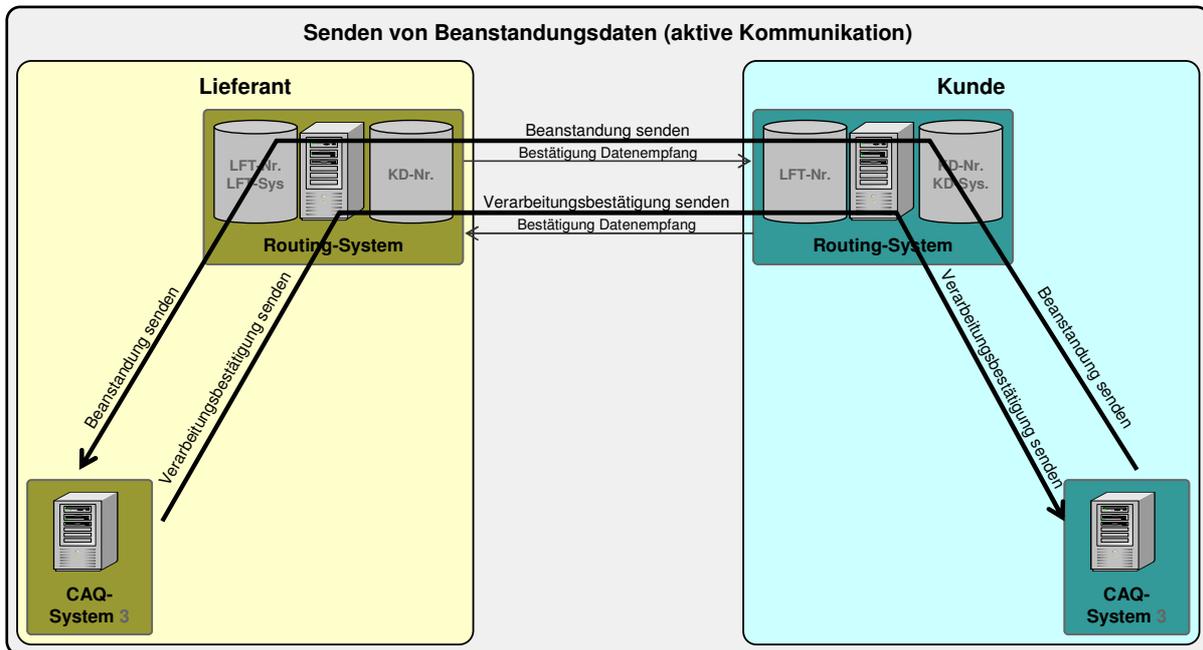


Abbildung 2-8: Aktive Kommunikation am Beispiel Beanstandungsprozess: Senden von Beanstandungsdaten

Die Antwort in Form eines 8D-Reports wird vom Lieferanten an den Kunden gesendet. Hat das Zielsystem des Kunden den 8D-Report erhalten und konnte diesen fehlerfrei verarbeiten, so wird vom Kunden eine Verarbeitungsbestätigung zurück an den Lieferanten gesendet. Dieser Schritt ist verpflichtend, damit der Lieferant die Information erhält, dass die gesendeten Daten auch tatsächlich im Zielsystem angekommen sind und verarbeitet werden konnten (speziell bei asynchroner Verarbeitung ist dieser Schritt unerlässlich). Der Erhalt der Daten (8D-Report bzw. Verarbeitungsbestätigung) wird jeweils von den empfangenden Systemen quittiert⁵.

⁵ Dies decken in der Regel die Protokolle ab.

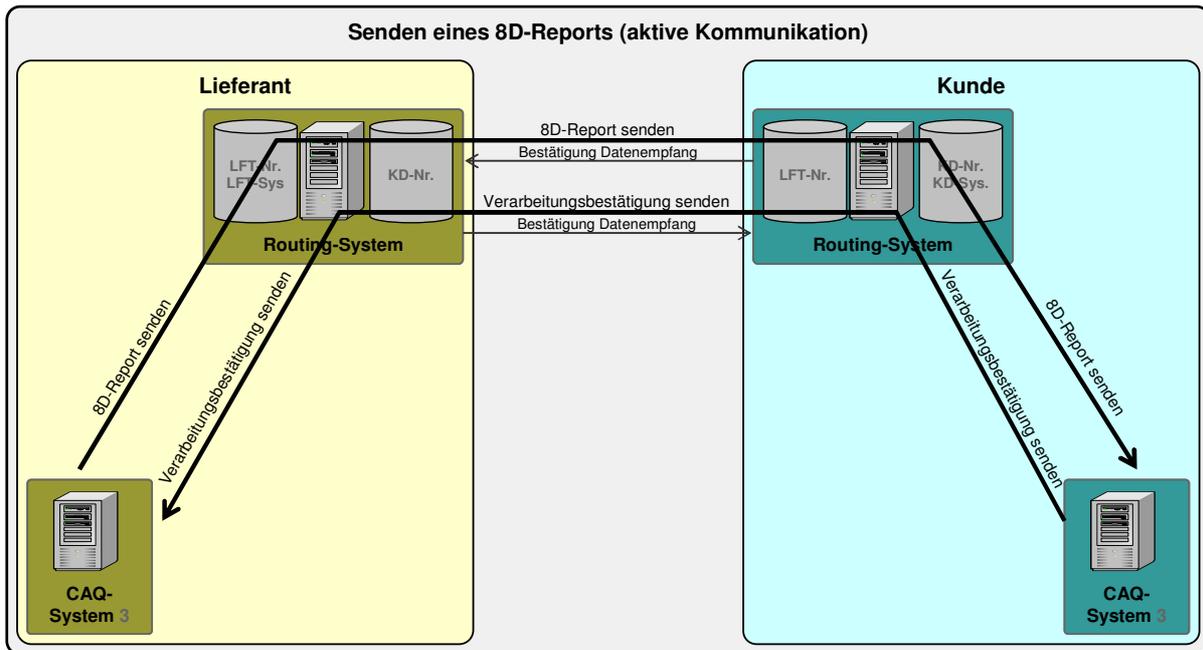


Abbildung 2-9: Aktive Kommunikation am Beispiel Beanstandungsprozess: Senden eines 8D-Reports

2.3.1.2 Initiator Lieferant

Am Erstbemusterungsprozess soll die Kommunikation zwischen den beiden Geschäftspartnern verdeutlicht werden, wenn der Lieferant der Initiator des Geschäftsprozesses ist.

Der Lieferant sendet den Erstmusterprüfbericht (PPF-Bericht) an den Kunden. Konnte der PPF-Bericht auch vom Zielsystem des Kunden korrekt verarbeitet werden, sendet dieses eine Verarbeitungsbestätigung zurück an den Lieferanten. Dieser Schritt ist verpflichtend, damit der Lieferant die Information erhält, dass die gesendeten Daten auch tatsächlich im Zielsystem angekommen sind und verarbeitet werden konnten (speziell bei asynchroner Verarbeitung ist dieser Schritt unerlässlich). Die empfangenden Systeme quittieren jeweils den Erhalt der Daten (PPF-Bericht bzw. Verarbeitungsbestätigung)⁵.

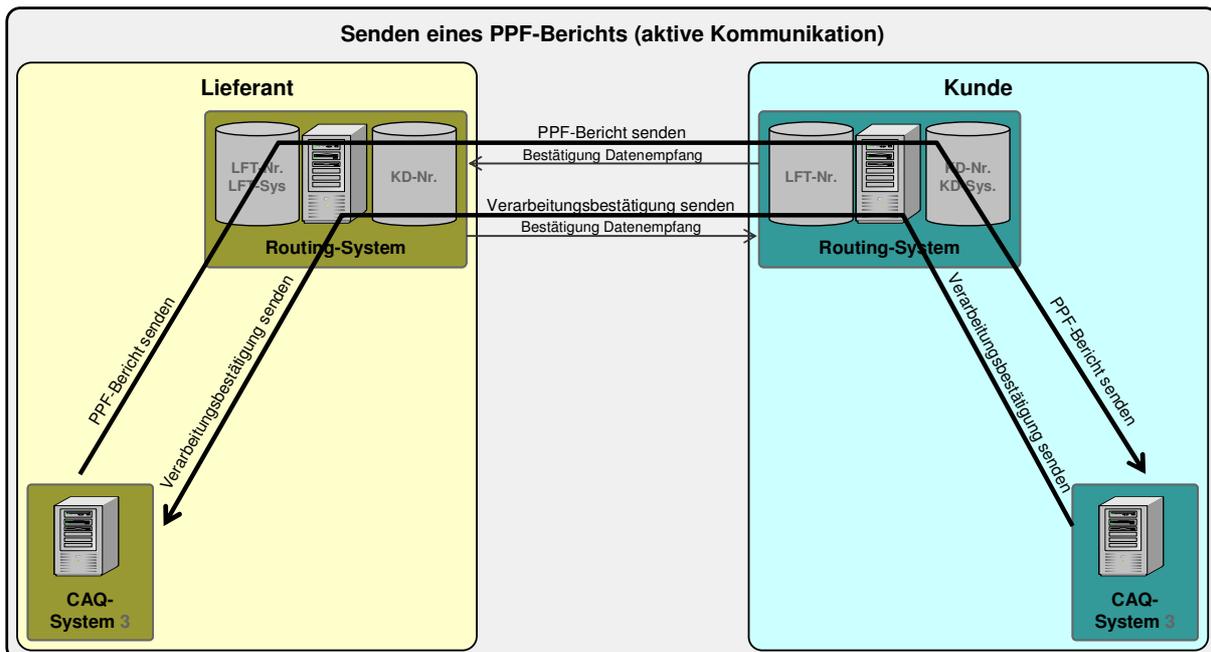


Abbildung 2-10: Aktive Kommunikation am Beispiel Erstbemusterung: Senden eines PPF-Berichts

Die Antwort in Form einer PPF wird vom Kunden an den Lieferanten gesendet. Hat das Zielsystem des Lieferanten die PPF erhalten und konnte diese fehlerfrei verarbeiten, so wird vom CAQ-System des Lieferanten eine Verarbeitungsbestätigung zurück an den Kunden gesendet. Dieser Schritt ist verpflichtend, damit der Kunde die Information erhält, dass die gesendeten Daten auch tatsächlich im Zielsystem angekommen sind und verarbeitet werden konnten (speziell bei asynchroner Verarbeitung ist dieser Schritt unerlässlich). Der Erhalt der Daten (PPF bzw. Verarbeitungsbestätigung) wird jeweils von den empfangenden Systemen quittiert⁵.

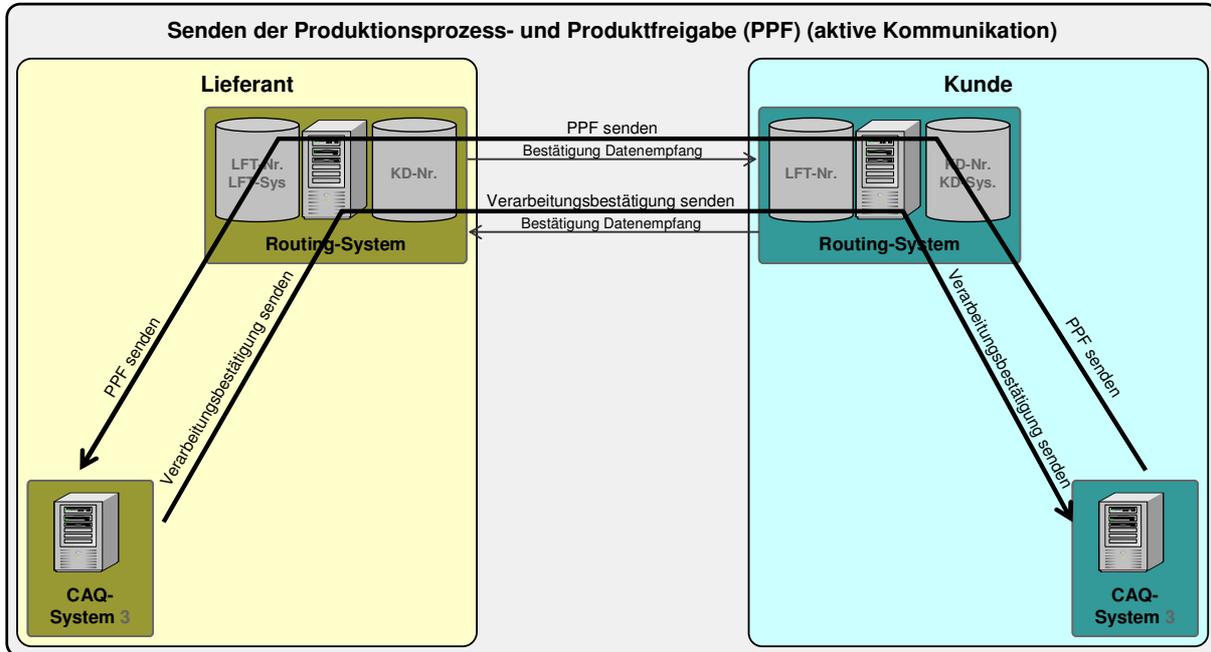


Abbildung 2-11: Aktive Kommunikation am Beispiel Erstbemusterung: Senden der PPF

2.3.2 Passive Kommunikation

Bei der passiven Kommunikation muss immer der Lieferant die Kommunikation mit dem Kunden initiieren, der Kunde verhält sich dagegen passiv. Möchte der Lieferant zunächst Daten an den Kunden senden, erfolgt dies analog der aktiven Kommunikation. Der Unterschied liegt darin, dass Daten vom Kunden an den Lieferanten übertragen werden sollen. Dies erfolgt nicht aktiv durch den Kunden, sondern muss durch den Lieferanten initiiert werden. Der Kunde verhält sich passiv und stellt die Daten nur zur Verfügung.

Aus Sicht des Prozessinitiators ergibt sich dadurch ein Unterschied beim Senden der initialen Nachricht wie auch beim Senden der Antwort.

Bei der „passiven“ Kommunikation ist also immer der Lieferant dafür verantwortlich, dass er die Daten des Kunden erhält und dass seine Daten an den Kunden übermittelt werden. Der Lieferant ist somit sowohl in der „Bring-“ als auch in der „Holschuld“.

Hinweis: Passive Kommunikation erfolgt typischerweise durch „Polle“⁶n der Gegenstelle des Geschäftspartners. Fällt die Entscheidung auf passive Kommunikation, so müssen die entsprechenden Systeme mit einer Grundlast an ständigen Systemzugriffen umgehen können.

2.3.2.1 Initiator Kunde

Ist der Kunde der Initiator des Geschäftsprozesses, so muss der Geschäftspartner (Lieferant) beim Initiator nachfragen, ob neue Informationen (Daten) zur Verfügung stehen und diese abrufen. Konnte der Lieferant die zur Verfügung gestellten Daten des Initiators im Zielsystem korrekt verarbeiten, sendet er eine

⁶ „Pollen“ wurde aus dem Englischen (to poll – befragen, abrufen) abgeleitet. Es beschreibt das regelmäßige Anfragen eines Clients an den Server, ob neue Daten/Informationen vorliegen.

Verarbeitungsbestätigung an den Initiator. Anschließend erfolgt die Beantwortung durch den Lieferanten, der die Antwort an den Initiator zurücksendet. Die Verarbeitungsbestätigung der Antwort muss nun vom Lieferanten abgerufen werden.

Am Beanstandungsprozess soll die Kommunikation zwischen den beiden Geschäftspartnern verdeutlicht werden.

Der Lieferant muss die Beanstandungsdaten des Kunden abrufen. Konnten die Beanstandungsdaten vom Zielsystem des Lieferanten korrekt verarbeitet werden, sendet dieses eine Verarbeitungsbestätigung zurück an den Kunden. Das empfangende System quittiert den Erhalt der Verarbeitungsbestätigung⁷.

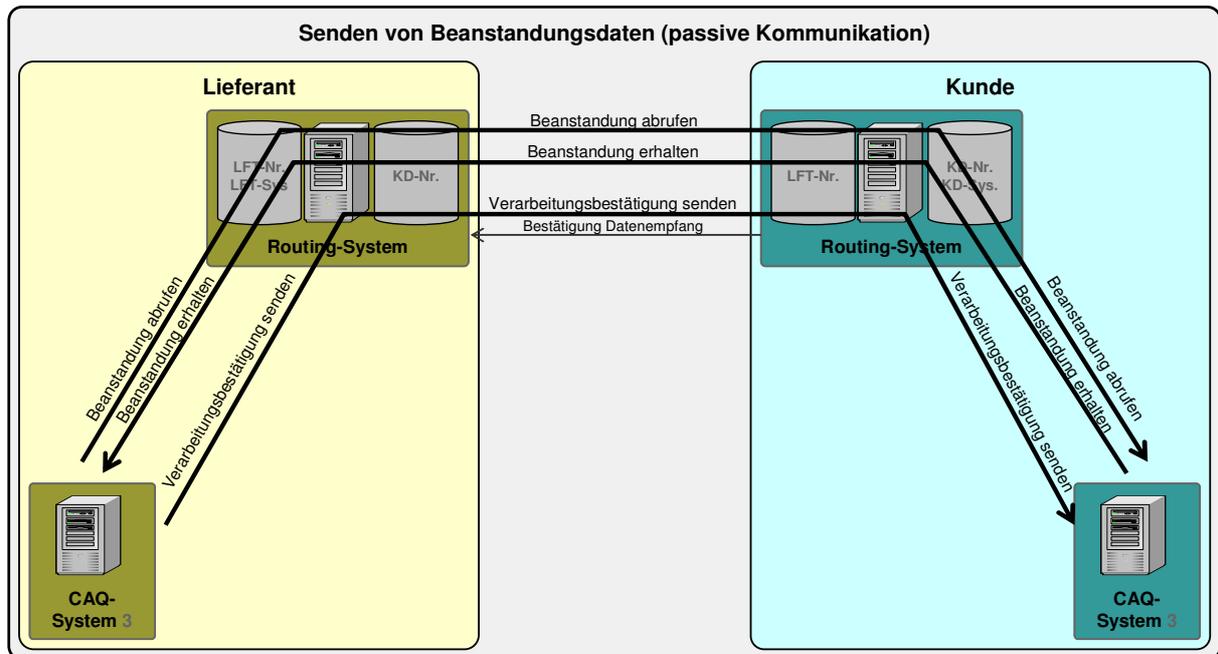


Abbildung 2-12: Passive Kommunikation am Beispiel Beanstandungsprozess: Senden von Beanstandungsdaten

⁷ Der Erhalt abgerufener Daten wird nicht im Rahmen des technischen Protokolls quittiert, da dies i. d. R. bereits die technische Antwort ist. Es werden also immer nur gesendete Daten quittiert.

Die Antwort in Form eines 8D-Reports wird vom Lieferanten an den Kunden gesendet. Die Verarbeitungsbestätigung, ob der 8D-Report korrekt im Zielsystem verarbeitet wurde, muss der Lieferant vom Kunden abrufen. Der Erhalt des 8D-Reports wird vom empfangenden System quittiert⁷.

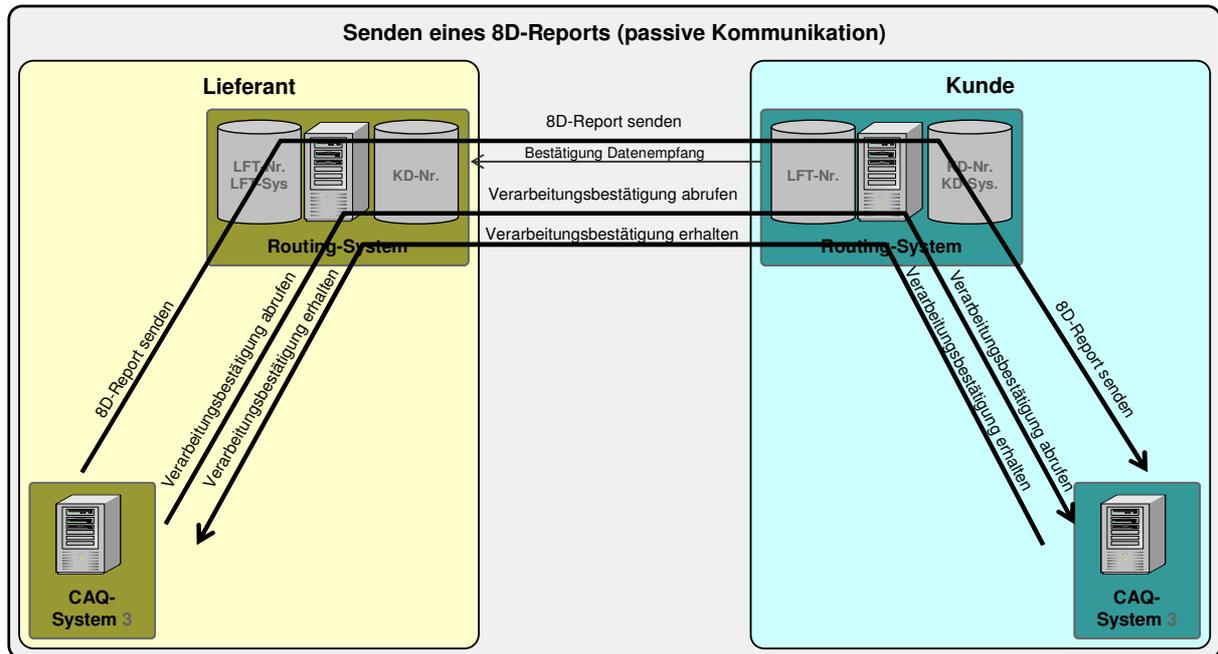


Abbildung 2-13: Passive Kommunikation am Beispiel Beanstandungsprozess: Senden eines 8D-Reports

2.3.2.2 Initiator Lieferant

Ist der Lieferant der Initiator des Geschäftsprozesses, so muss er zunächst seine Informationen (Daten) an den Geschäftspartner (Kunde) senden. Die Verarbeitungsbestätigung muss nun aber der Initiator abrufen. Anschließend erfolgt die Beantwortung durch den Kunden. Die Antwort muss der Initiator ebenfalls wieder abrufen. Die Verarbeitungsbestätigung der Antwort sendet der Initiator schließlich an den Kunden.

Am Erstbemusterungsprozess soll die Kommunikation zwischen den beiden Geschäftspartnern verdeutlicht werden.

Der Lieferant sendet den PPF-Bericht an den Kunden. Die Verarbeitungsbestätigung, ob der PPF-Bericht auch vom Zielsystem des Kunden korrekt verarbeitet wurde, muss der Lieferant vom Kunden abrufen. Das empfangende System des Kunden quittiert den Erhalt des PPF-Berichts⁷.

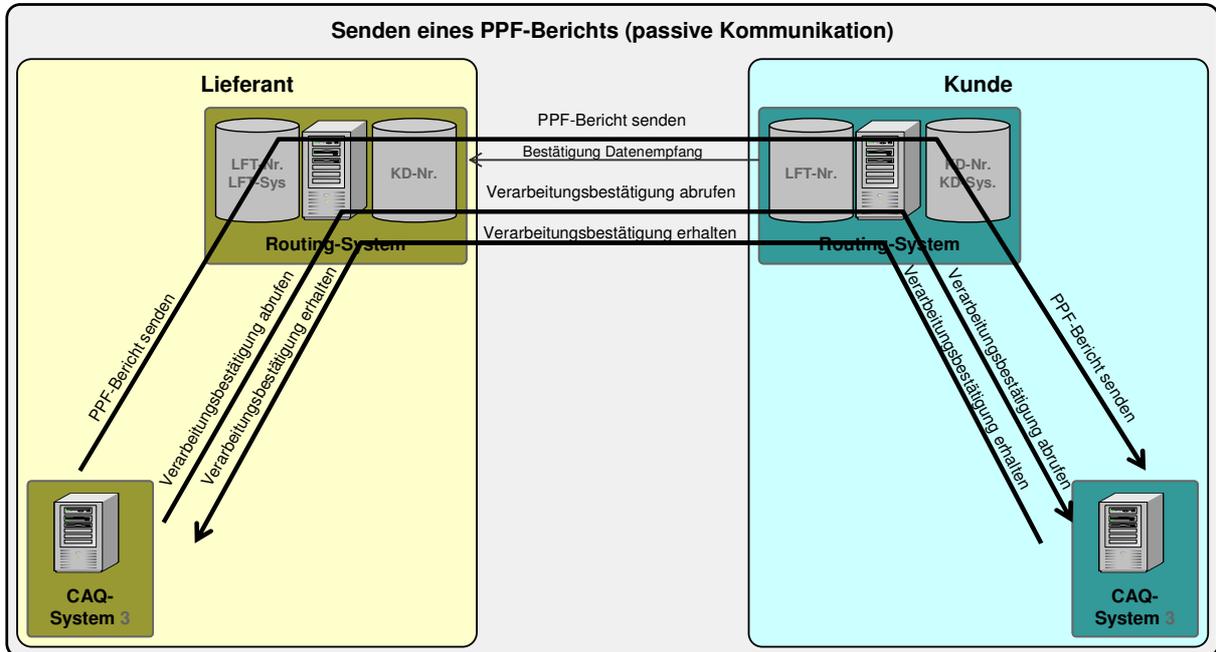


Abbildung 2-14: Aktive Kommunikation am Beispiel Erstbemusterung: Senden eines PPF-Berichts

Die Antwort in Form einer PPF muss der Lieferant vom Kunden abrufen. Konnte das Zielsystem des Lieferanten die PPF fehlerfrei verarbeiten, sendet der Lieferant eine Verarbeitungsbestätigung zurück an den Kunden. Der Erhalt der Verarbeitungsbestätigung wird vom empfangenden System des Kunden quittiert⁷.

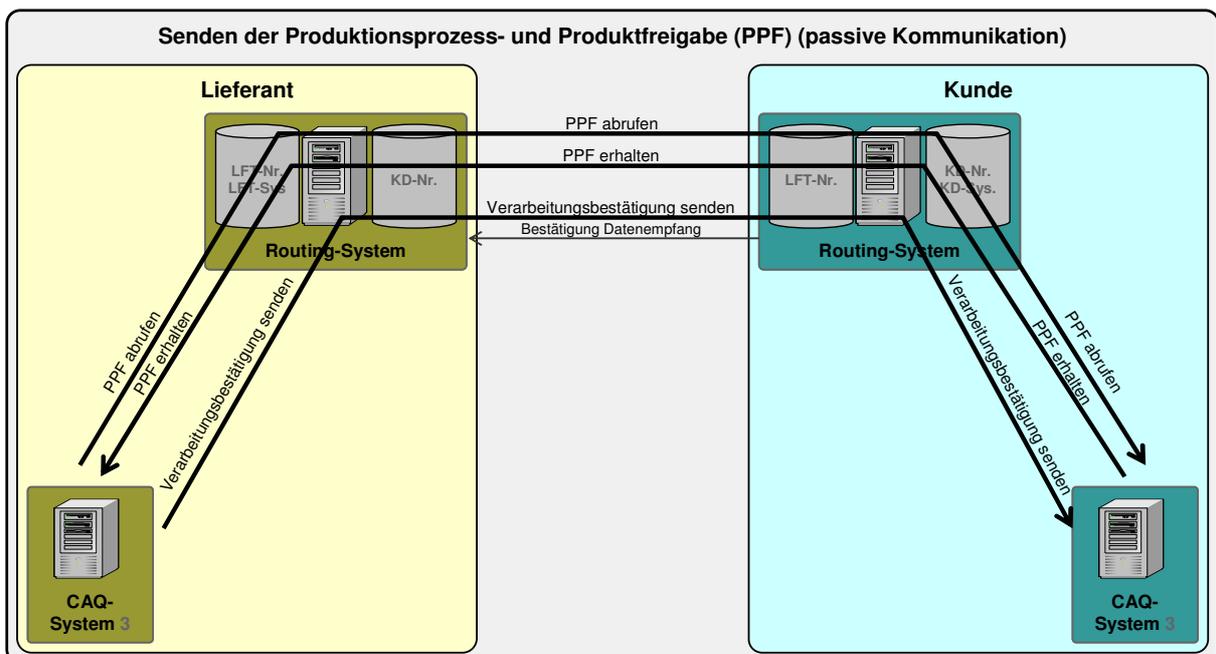


Abbildung 2-15: Aktive Kommunikation am Beispiel Erstbemusterung: Senden der PPF

3 Kommunikationsverfahren

Das Kommunikationsverfahren ist im Rahmen einer individuellen Partnervereinbarung festzulegen. In einer solchen Vereinbarung ist unter anderem das Netzwerk (Internet, ISDN, ENX) zu definieren, über welches die Kommunikation stattfinden soll.

Hat sich der Kunde für die „Aktive Kommunikation“ entschieden (siehe Kapitel 2.3.1), so ist OFTP zu verwenden, im Falle einer „Passiven Kommunikation“ (siehe Kapitel 2.3.2) Webservice.

Bei beiden Varianten sind die Spezifikationen des Kapitels 4 einzuhalten.

3.1 OFTP (Aktive Kommunikation)

Das **ODETTE-File-Transfer-Protocol** (OFTP) ist ein Protokoll zur elektronischen Datenübertragung zwischen Geschäftspartnern und bewährt sich seit über 20 Jahren. Es dient dem gesicherten Transport von Nachrichten zwischen zwei Partnern. Dabei lassen sich sowohl strukturierte als auch unstrukturierte Daten austauschen. Dieser Austausch kann im Punkt-zu-Punkt-Verfahren oder über Mehrwertnetz-Dienstleister sowohl über ISDN, TCP/IP als auch über X.25 genutzt werden. Die Kommunikation über TCP/IP umfasst dabei auch den Datenaustausch über ENX und das Internet. Letzteres gilt allerdings nur für den Einsatz der OFTP Version 2.0.

Das ODETTE-File-Transfer-Protocol gewährleistet dabei:

- **Datenunabhängigkeit:** Mit dem OFTP können alle Arten von Anwenderdaten übertragen werden.
- **Datenschutz:** Der Austausch eines Passwortes ist vorgeschriebener Bestandteil des Protokolls. Bei eingehendem Ruf kann der rufende Partner identifiziert werden. Damit wird nichtautorisierten Anrufern der Verbindungsaufbau automatisch verwehrt. Die ISDN- und TCP/IP-Schnittstelle bietet zudem die Rufnummerprüfung als Verfahren, um ausschließlich eingehende Rufe von bekannten Partnern entgegenzunehmen. Das OFTP 2.0 bietet zusätzlich noch die Möglichkeit der Verschlüsselung sowohl auf Datei- als auch auf Protokollebene.
- **Datensicherheit:** Zum Schutz vor Fehlern oder Datenverlusten ist eine automatische Rückmeldung vom Empfänger zum Sender integriert.
- **Komprimierung:** Das OFTP bietet die Möglichkeit, Daten zu komprimieren und komprimierte Daten weiterzuverarbeiten. Dieses Leistungsmerkmal verringert den Umfang der zu übertragenden Zeichen und damit auch die Übertragungskosten.
- **Wiederanlauf nach Übertragungsfehlern:** Nach einem Abbruch der Übertragung aufgrund von Übertragungsfehlern kann eine erneute Übertragung ab dem zuletzt korrekt gesendeten und empfangenen Datenpaket erfolgen. Diese Restart-Fähigkeit gewährleistet, dass lediglich die noch fehlenden Daten übertragen werden müssen, was die Übertragungsdauer verkürzt und damit die Übertragungskosten verringert.
- **Austausch von End-To-End-Responses (EERPs):** Das OFTP unterstützt den Austausch von End-to-End-Responses, wodurch die Vollständigkeit der Datenübertragung vom Empfänger bestätigt wird.

Zusätzlich zu den bisher genannten Leistungsmerkmalen gewährleistet OFTP 2.0:

- **Transaktionssicherheit und Vertraulichkeit:** Bei OFTP 2.0 werden sowohl auf Leitungsebene als auch auf Dateiebene Transaktionssicherheit und Vertraulichkeit gewährleistet. So können Sessions durch Verschlüsselung und Authentifizierung vollständig gesichert werden. Unbefugte Dritte haben somit keinen Zugriff auf Informationen aus der Übertragung. Bei Datenübertragungen über das Internet sichert die Transport Layer Security (TLS)⁸ die gesamte Session und sorgt auch für die Authentifizierung der Partner.

⁸ Transport Layer Security (TLS) ist ein Verschlüsselungsprotokoll für Datenübertragungen im Internet. TLS 1.0 und 1.1 sind die standardisierten Weiterentwicklungen von Secure Sockets Layer (SSL) 3.0.

Bei der Übertragung über andere Netzwerke wird die Identität der Partner zu Beginn einer OFTP-Session über Zertifikate geprüft. Auf Dateiebene können dann die Dateien selbst verschlüsselt in beide Richtungen übertragen werden.

- **Datenintegrität und Authentizität:** Datenintegrität und Authentizität gewährleistet OFTP 2.0 durch digitale Signaturen. Diese liefern den Nachweis, dass die Daten vom Sender stammen und nach dem Versand nicht verändert wurden. Hierfür signiert der Sender die Daten mit seinem Private Key. Mit dem Public Key lässt sich diese Signatur überprüfen. Zum Prüfen der Signatur benötigen beide Partner voneinander die Zertifikate mit Public Key.
- **Unleugbarkeit:** Das OFTP 2.0 unterstützt den Austausch von End-to-End-Responses (EERP) und negativen End-to-End-Responses (NERP), wodurch die Vollständigkeit der Datenübertragung vom Empfänger bestätigt oder die fehlerhafte Übertragung angezeigt wird. Die EERP dient damit auch als Empfangsnachweis. Außerdem können die Bestätigungsnachrichten digitale Signaturen enthalten.

3.1.1 Übertragung von QDX-Nachrichten per OFTP

Die Übertragung von QDX-Nachrichten per OFTP läuft grundsätzlich für alle Nachrichtentypen nach dem selben Schema ab, welches die nachstehende Abbildung veranschaulicht.

Auf die Unterschiede zwischen OFTP V1 und V2, insbesondere auf die Sicherheitsmechanismen von OFTP V2, wird an dieser Stelle bewusst nicht eingegangen, da sie für das Grundverständnis des Austausches von QDX-Nachrichten per OFTP nicht relevant sind.

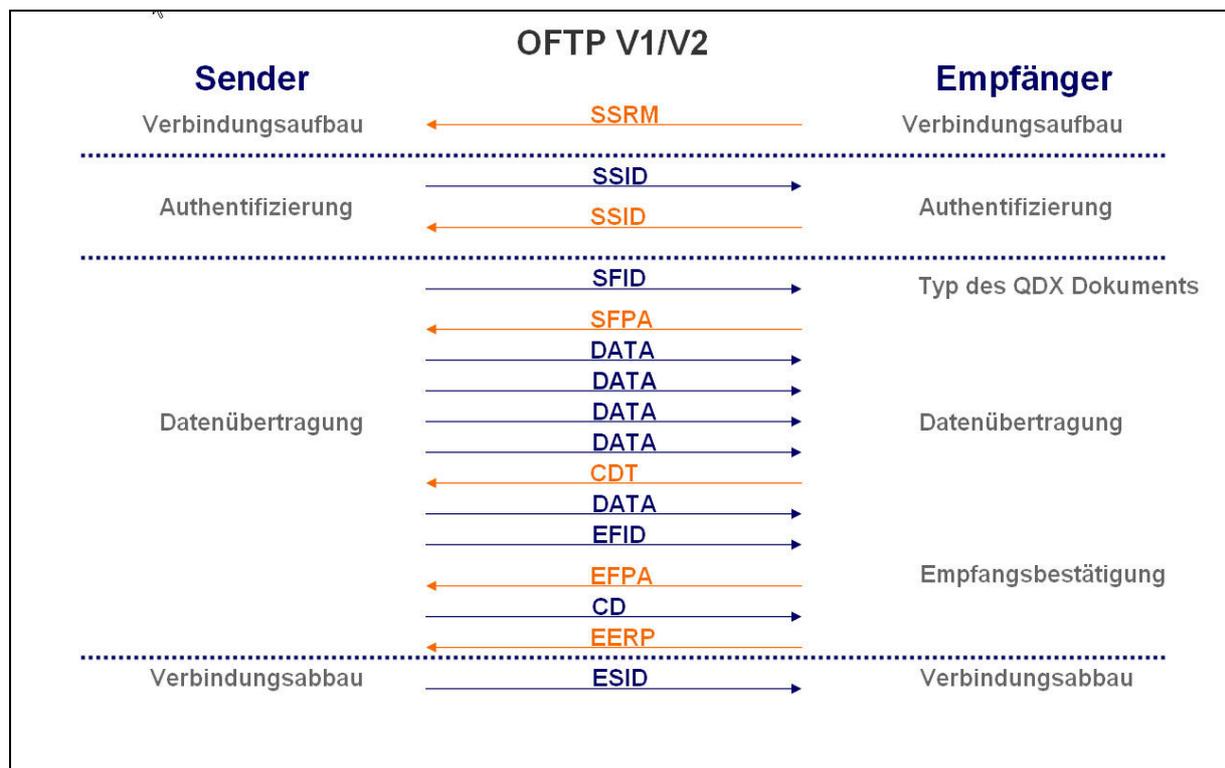


Abbildung 3.1-1 Übertragung von QDX-Nachrichten per OFTP

Bei den weiteren Ausführungen wird davon ausgegangen, dass zwischen den Beteiligten bereits eine Kommunikationsverbindung über OFTP besteht.

Bei einer Punkt-zu-Punkt Verbindung wird bereits auf Protokollebene, nämlich in der Start File ID (SFID), der Typ des zu übertragenden QDX-Dokumentes dem Empfänger mitgeteilt. Anhand dieser Information lässt sich auf dem Empfängersystem ein spezifischer Workflow für die weitere Verarbeitung der Nachricht starten.

Neben der Übermittlung von QDX-Nachrichten wird es in der Praxis auch erforderlich sein, zu einer QDX-Nachricht auch noch ein oder mehrere Attachments zu übertragen, die das vorliegende Qualitätsproblem veranschaulichen sollen.

In der vorliegenden Empfehlung werden beide Varianten der Übertragung von QDX-Daten per OFTP betrachtet:

- Übertragung von QDX-Nachrichten ohne Attachments
- Übertragung von QDX-Nachrichten mit Attachments

3.1.1.1 Übertragung von QDX-Nachrichten ohne Attachments

Sofern QDX-Nachrichten ohne Attachments übertragen werden, muss zwischen den Partnern lediglich der virtuelle Filename für diese Datenart vereinbart werden. Diese Aussage gilt nur für eine Punkt-zu-Punkt Verbindung. Werden die Daten über ein VAN übertragen, übermittelt der Betreiber oftmals nur einen festen virtuellen Dateinamen für alle Typen von Nachrichten. In diesem Fall müssen die für die Weiterverarbeitung notwendigen Routing-Informationen aus den Nutzdaten ermittelt werden.

3.1.1.2 Übertragung von QDX-Nachrichten mit Attachments

Sollen neben einer QDX-Nachricht auch Attachments (*.jpg, *.wav, *.bmp) übertragen werden, muss zwischen den Partnern nicht nur der virtuelle Filename vereinbart werden, sondern auch ob die QDX-Nachricht und Attachments in einer oder in mehreren Übertragungen (Sessions) transportiert werden sollen. Die Übertragung in mehreren Sessions bedingt einen höheren organisatorischen Aufwand beim Empfänger und wird deshalb ausgeschlossen. Die Übertragung von QDX-Nachricht und Attachments in einer Session erfordert eine vorherige Bündelung der Einzeldateien zu einem physischen Objekt (Daten-Bundle). Dies geschieht durch die Verwendung einer SOAP-Nachricht in Multipart MIME Format. Die Verwendung eines solchen SOAP-Envelopes für die Bündelung der zu übertragenden Daten ist vor allem dann empfehlenswert, wenn für die Weiterverarbeitung der Daten ein Webservice in Anspruch genommen werden soll. Konnten die Routing-Informationen bereits auf Protokollebene ermittelt werden, kann das Datenpaket an den Webservice übergeben werden. Andernfalls müssen die für das Routing benötigten Informationen dem SOAP Header entnommen werden. Darüber hinaus befinden sich im Payload neben der eigentlichen QDX-Nachricht auch noch die Referenzen zu den Attachments. Für weitere Informationen siehe auch Kapitel 4.5.

Die nachstehende Abbildung veranschaulicht den Inhalt der SOAP-Nachricht.

```

MIME-Version: 1.0
Content-type: multipart/mixed; boundary=mime-boundary; type=text/xml
-----mime-boundary-----
Content-type: text/xml; charset=utf-8
Content-Transfer-Encoding: 8bit
SOAP-Nachricht  
inkl. QDX-Dokument

<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <env:Header>
    <wsa:To env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">urn:vda.qdx:empfänger</wsa:To>
    <wsa:From env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">
      <wsa:Address>urn:vda.qdx:sender</wsa:Address>
    </wsa:From>
    <wsa:Action env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">urn:vda.qdx:qdx-dokument</wsa:Action>
  </env:Header>
  <env:Body>
    <qer:QDXEnvelopeResponse xmlns:qer="www.vda-qmc.de/qdx/QDXEnvelopeResponse_V2.0.xsd">
      <qer:Code>201</qer:Code>
      <qer:CodeDescription>Request succeeded</qer:CodeDescription>
      <qer:CodeDetails>Anfrage erfolgreich durchgeführt</qer:CodeDetails>
      <qdx:QDXComplaint xmlns:qdx="www.vda-qmc.de/qdx/QDXComplaint_V2.0.xsd">
        ...
      </qdx:QDXComplaint>
    </qer:QDXEnvelopeResponse>
  </env:Body>
</env:Envelope>
-----mime-boundary-----
Attachment 1
Content-type: image/jpeg
Content-ID: 1
Content-Transfer-Encoding: binary

...binäres jpg-Bild....
-----mime-boundary-----
Attachment 2
Content-type: video/mpeg
Content-ID: 2
Content-Transfer-Encoding: binary

...binäres mpeg-Video...

```

Abbildung 3.1-2 Struktur einer SOAP-Nachricht bestehend aus einem QDX-Complaint mit zwei Attachments

3.1.2 Übertragung von Beanstandungen per OFTP

Nach den allgemeinen Erläuterungen in den vorangegangenen Kapiteln zur Übertragung von QDX-Nachrichten soll in diesem Abschnitt der Prozess der Übertragung eines Complaints veranschaulicht werden.

Initiiert wird der Prozess durch den Kunden, indem dieser eine Beanstandung bzw. einen QDXComplaint an den Lieferanten überträgt. Der Lieferant bestätigt seinerseits die ordnungsgemäße Verarbeitung der Beanstandung in seinem CAQ-System durch Zurücksenden seiner Bestätigung bzgl. der Verarbeitung mittels. QDXAcknowledgeComplaint.

Sollte seitens des Lieferanten die Notwendigkeit bestehen, die Übertragung der Beanstandungsnachricht durch den Kunden zu wiederholen, kann er die erneute Übertragung durch Senden einer entsprechenden Nachricht (QDXResetDeliveryStatusComplaint) anfordern.

Für alle drei QDX-Nachrichtentypen ist ein gesonderter virtueller Dateiname zu vereinbaren.

Die nachstehende Abbildung veranschaulicht den beschriebenen Prozess.

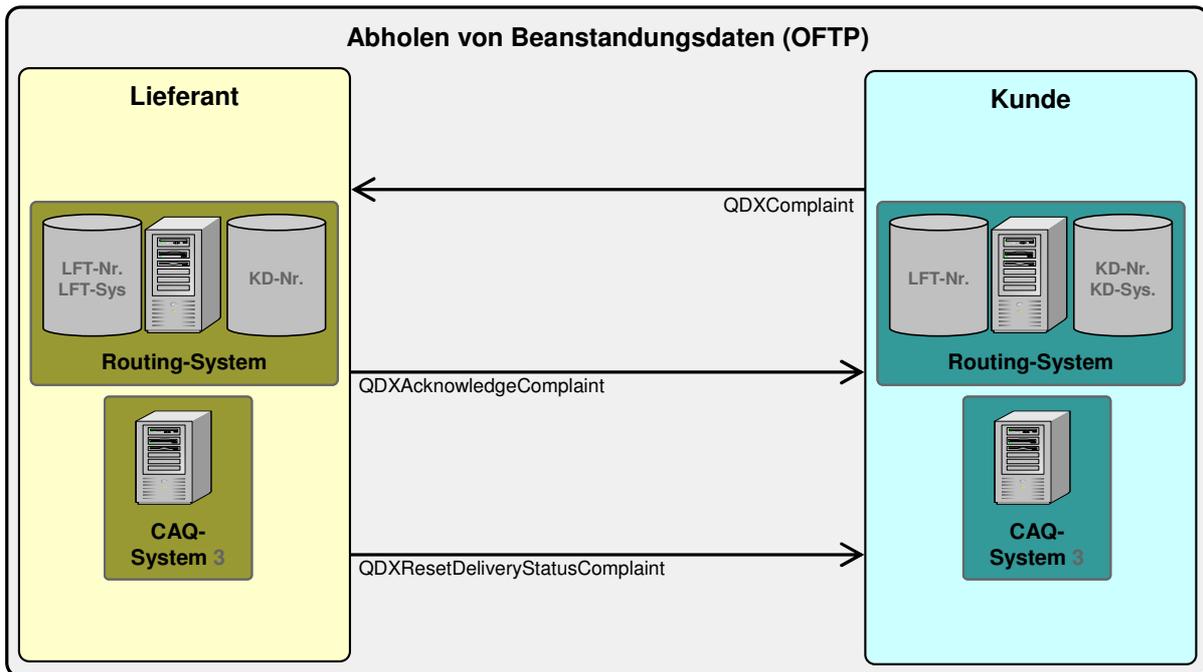


Abbildung 3.1-3 Übertragung einer Beanstandungsdaten per OFTP

3.1.3 Übertragung von 8D-Reports per OFTP

Als Antwort auf eine akzeptierte Reklamation sendet der Lieferant einen 8D-Report bzw. QDXReport8D an den Kunden. Der Kunde bestätigt die korrekte Verarbeitung des erhaltenen 8D-Reports in seinem CAQ-System durch Zurücksenden eines Acknowledgements bzw. QDXAcknowledgeReport8D an den Lieferanten.

Für die beiden QDX-Nachrichtentypen ist jeweils ein gesonderter virtueller Dateiname zu vereinbaren.

Die nachstehende Abbildung veranschaulicht den beschriebenen Prozess.

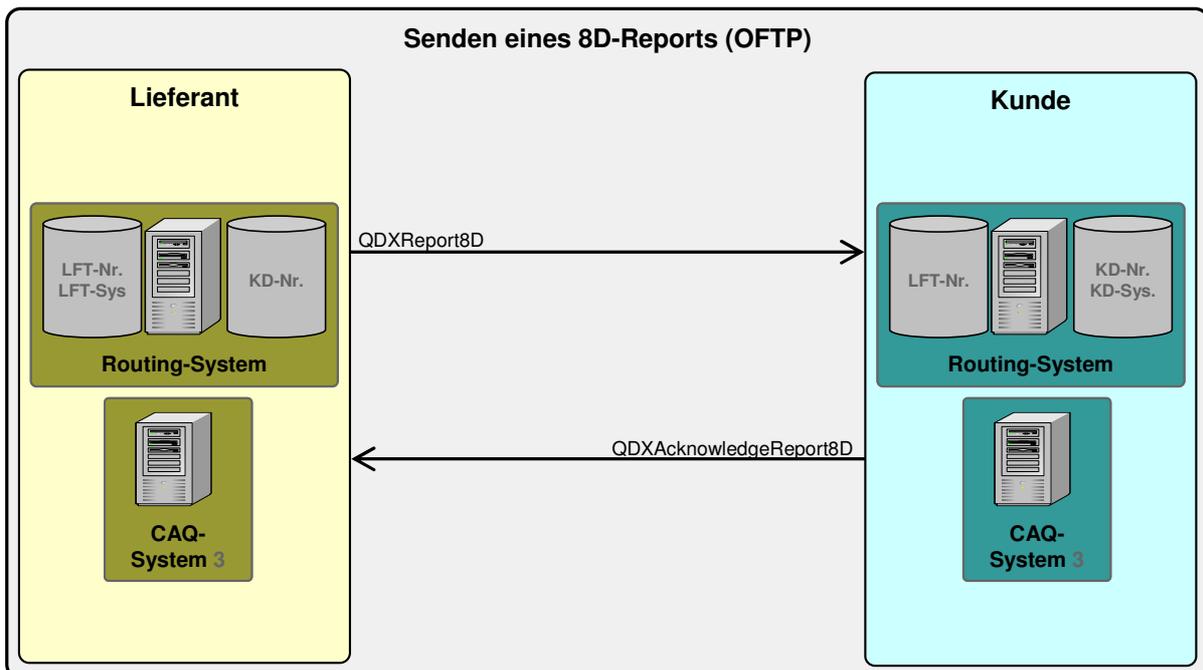


Abbildung 3.1-4 Senden eines 8D-Reports per OFTP

3.2 Webservices (Passive Kommunikation)

Die Kommunikation über Webservice entspricht im Ablauf der in Kapitel 2.3.2 beschriebenen „Passiven Kommunikation“, bei der der Aufbau der Kommunikation stets durch den Lieferanten erfolgt. Da der Lieferant beim Initiieren der Kommunikation die aufzurufenden Methoden kennen muss, werden in diesem Kapitel auch die Webservice-Methoden inklusive der Aufrufparameter definiert. Dieses Kapitel kann aus diesem Grund auch nicht allgemeingültig formuliert werden, weshalb hier die Beschreibung der Webservice-Methoden für den Beanstandungsprozess erfolgt.

Im Rahmen einer individuellen Partnervereinbarung sind Wartezeiten⁹ und Zugriffsintervalle auf den Webservice zu definieren. Zudem ist auch die Timeout-Zeit festzulegen, nach der ein Client die Anfrage abbricht¹⁰. Innerhalb der Timeout-Zeit muss der Kunde den Request beantworten und die gewünschten Daten zurücksenden (z. B. einen QDXComplaint inklusive aller Attachments).

Hinweis: Werden Beanstandungsdaten erst beim Aufruf der Webservice-Methode erzeugt, ist besonders auf das Einhalten der festgelegten Timeout-Zeit zu achten.

3.2.1 Authentisierung

Zur Authentisierung gegenüber dem Webservice wird das Basic-Authentication¹¹ Verfahren verwendet. Username und Passwort werden dabei codiert übertragen. Als Übertragungsprotokoll ist HTTPS zu empfehlen, wodurch eine verschlüsselte Verbindung aufgebaut wird und dadurch Username und Passwort vor unberechtigten Zugriffen von außen geschützt sind.

3.2.2 Abholen von Beanstandungen

Im Reklamationsprozess wird die initiale Nachricht – die Beanstandung bzw. der QDXComplaint – vom Kunden erzeugt. Aufgrund der passiven Kommunikation muss der Kommunikationsaufbau durch den Lieferanten erfolgen, der die Nachricht abrufen muss.

Zur Optimierung der Kommunikation wird der in Kapitel 2.3.2 beschriebene Ablauf um zwei Anfragen erweitert:

- Um Beanstandungen abzurufen muss der Lieferant vorab eine Liste aller von ihm abzuholender Beanstandungen abrufen.
- Der Lieferant kann das erneute Senden einer Beanstandung erwirken.

⁹ Zum Beispiel Wartezeiten zwischen dem Senden eines 8D-Reports und dem Abruf der Verarbeitungsbestätigung zu dem gesendeten 8D-Report)

¹⁰ Standardzeit für Timeout: 120 s.

¹¹ Weitere Informationen zur Basic-Authentication: <http://tools.ietf.org/html/rfc2617>

Nachfolgende Grafik gibt einen Überblick über die Webservice-Methoden und ihre logischen Reihenfolgen.

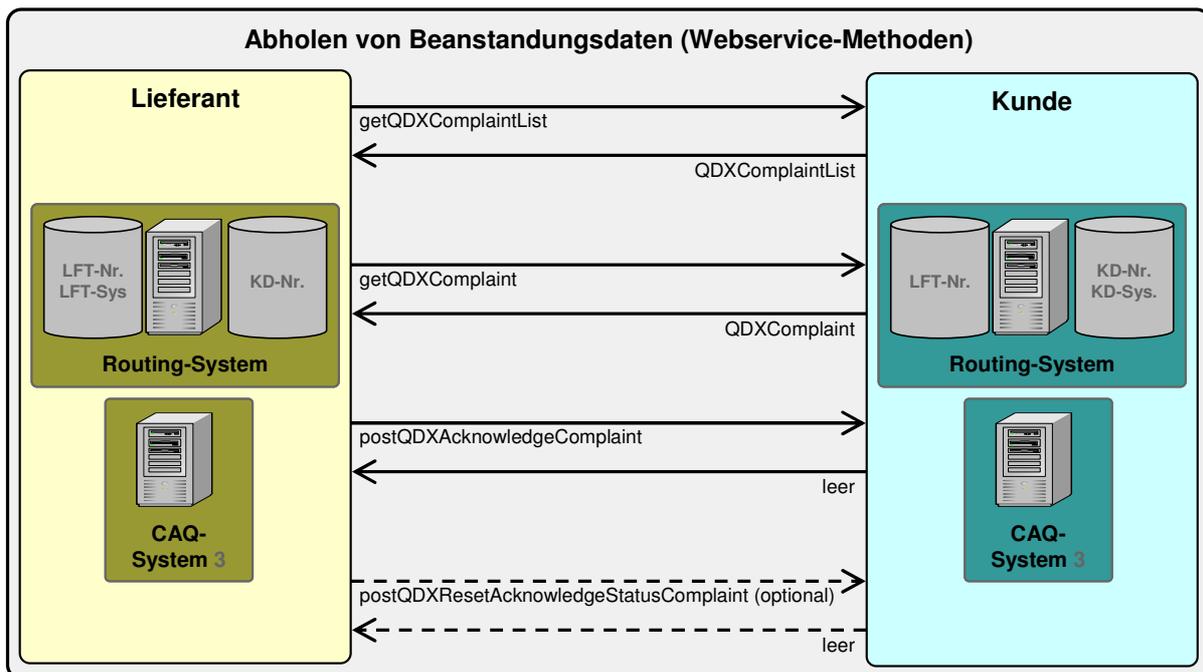


Abbildung 3-5: Webservice-Methoden zum Abrufen von Beanstandungen

- **getQDXComplaintList:** Mit dieser Methode muss der Lieferant (bzw. das CAQ-System) eine Liste aller von ihm abzuholender Beanstandungen abrufen. Die Liste beinhaltet die ID der Beanstandungen des Kunden. Anhand der ID kann der Lieferant die nachfolgend beschriebene Methode aufrufen.
- **getQDXComplaint:** Hiermit ruft der Lieferant seine abzuholenden Beanstandungen anhand der erhaltenen ID einzeln ab.
- **postQDXAcknowledgeComplaint:** Nach erfolgreicher maschineller Verarbeitung des QDXComplaints im CAQ-System des Lieferanten muss dies der Lieferant bestätigen. Dieser Schritt ist also verpflichtend und immer nach erfolgreicher Verarbeitung eines QDXComplaints auszuführen! Ein bestätigter QDXComplaint wird bei einem erneuten Aufruf von "getQDXComplaintList" nicht erneut aufgeführt.
- **postQDXResetAcknowledgeStatusComplaint:** Treten Gründe auf, weshalb ein bereits abgerufener und bestätigter QDXComplaint erneut vom Lieferanten abgerufen werden muss, so kann der Status des QDXComplaints mit dieser Methode zurückgesetzt werden.¹² Dieser Schritt kann bei Bedarf und optional vom Lieferanten ausgeführt werden.

Jede Webservice-Methode benötigt vom aufrufenden System Informationen, um die korrekten Daten ermitteln zu können. Diese Übergabeparameter werden als QDX-Dokument im SOAP-Body als HTTP-Request übergeben. Das Ergebnis des Methodenaufrufs wird als QDX-Dokument im Rahmen der HTTP-Response übermittelt. In der nachstehenden Tabelle werden pro Webservice-Methode die Übergabeparameter, das Ergebnis und die möglichen Statuscodes zugeordnet. Die für die Übergabeparameter und für das Ergebnis benötigten QDX-Dokumente werden in Kapitel 4.4 detailliert beschrieben, die Statuscodes in Kapitel 5.3.

¹² Eine Rücksetzung des Status eines QDXComplaints hat keine automatische (!) Auswirkung auf die Terminüberwachung. In der Regel wird die Terminüberwachung nicht angepasst, da hier tendenziell ein Verschulden des Lieferanten vorliegt.

Methode	Übergabeparameter	Ergebnis	Statuscodes
getQDXComplaintList	QDXComplaintListRequest	QDXComplaintList	200, 400
getQDXComplaint	QDXComplaintRequest	QDXComplaint	201, 401, 402, 403
postQDXAcknowledgeComplaint	QDXAcknowledgeComplaint	leer	202, 401, 402, 403, , 404, 405, 406
postQDXResetAcknowledgeStatusComplaint	QDXResetAcknowledgeStatusComplaint	leer	203, 401, 402, 403

Tabelle 3-1: Beschreibung der Webservice-Methoden zum Abholen von Beanstandungen

3.2.3 Senden eines 8D-Reports

Der 8D-Report stellt im Standard-Reklamationsprozess die fachliche Antwort dar und wird vom Lieferanten erstellt. Da diesmal die Kommunikation vom Dokumentenersteller – dem Lieferanten – aufgebaut wird, gestaltet sich die Kommunikation im Vergleich zum Abrufen von Beanstandungen einfacher.

Nachfolgende Grafik gibt einen Überblick über die Webservice-Methoden und ihre logischen Reihenfolge für den 8D-Report. Weitere Antworttypen (wie QDXShortConfirmation oder QDXFieldFailureResponse) können analog verwendet werden.

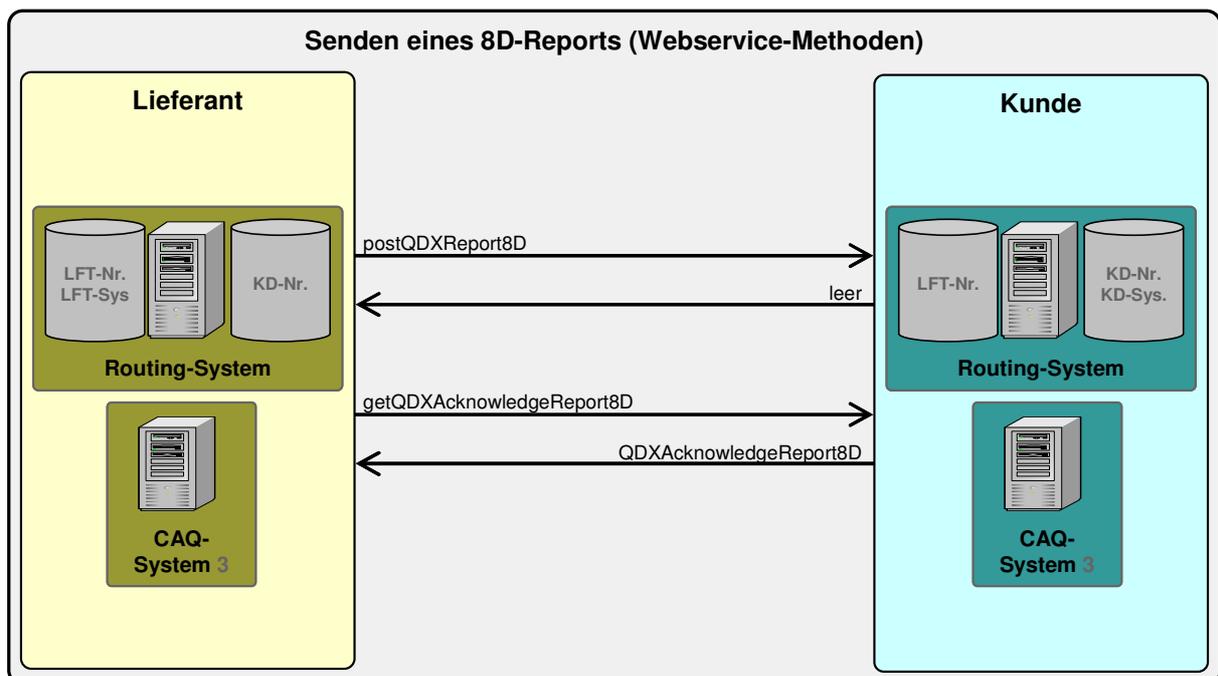


Abbildung 3-6: Webservice-Methoden zum Einstellen von 8D-Reports

- **postQDXReport8D:** Hiermit kann der Lieferant einen 8D-Report zu einer Beanstandung im CAQ-System beim Kunden einstellen.
- **getQDXAcknowledgeReport8D:** Um sicherzustellen, dass der 8D-Report vom CAQ-System des Kunden korrekt verarbeitet werden konnte, muss der Lieferant dies mit dieser Methode abfragen. Dieser Schritt ist verpflichtend und immer nach dem Senden eines QDXReport8Ds auszuführen! (Die Wartezeit zwischen dem Ausführen der beiden Methoden ist in einer individuellen Partnervereinbarung zu definieren – siehe Kapitel 5.4).

Jede Webservice-Methode benötigt vom aufrufenden System Informationen, um die korrekten Daten ermitteln zu können. Diese Übergabeparameter werden als QDX-Dokument im SOAP-Body als HTTP-Request übergeben. Das Ergebnis des Methodenaufrufs wird als QDX-Dokument im Rahmen der HTTP-Response übermittelt. In der

nachstehenden Tabelle werden pro Webservice-Methode die Übergabeparameter, das Ergebnis und die möglichen Statuscodes zugeordnet. Die für die Übergabeparameter und für das Ergebnis benötigten QDX-Dokumente werden in Kapitel 4.4 detailliert beschrieben, die Statuscodes in Kapitel 5.3.

Methode	Übergabeparameter	Ergebnis	Statuscode
postQDXReport8D	QDXReport8D	leer	204, 401, 402, 403
getQDXAcknowledgeReport8D	QDXAcknowledgeReport8DRequest	QDXAcknowledgeReport8D	205, 401, 402, 403

Tabelle 3-2: Beschreibung der Webservice-Methoden zum Senden der fachlichen Antwort

4 Format des Datenpakets

In den heutigen Systemlandschaften der größeren Unternehmen wird die Kommunikation mit externen Systemen oft durch ein zentrales Routing-System gesteuert. Es findet also keine direkte Kommunikation zwischen den beiden Endsystemen (CAQ-Systemen) statt, sondern es liegen weitere Systeme dazwischen. Um die QDX-Dokumente an das richtige CAQ-System weiterleiten zu können, müssten die Routing-Systeme in die QDX-Dokumente „schauen“ können, um dies anhand definierter Informationen herauszufinden. Um zu vermeiden, dass jedes Zwischensystem QDX „verstehen“ können muss, wurde festgelegt, dass die QDX-Dokumente innerhalb einer SOAP-Nachricht¹³ übertragen werden sollen – unabhängig von der Übertragungstechnik.

Die Metainformationen, die für die Verarbeitung bzw. für das Routing der Nachricht in Zwischensystemen benötigt werden, werden im Header der SOAP-Nachricht (auch SOAP-Envelope genannt) übertragen (genauere Beschreibung siehe Kapitel 4.1). Das eigentliche QDX-Dokument befindet sich im SOAP-Body (vergleiche Kapitel 4.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
  </env:Header>
  <env:Body>
  </env:Body>
</env:Envelope>
```

Abbildung 4-1: Aufbau SOAP-Envelope

SOAP wurde deshalb als Transportstruktur für die QDX-Nachricht gewählt, weil SOAP ein offener Standard des W3C¹⁴ ist und im Rahmen der Datenübertragung bei Webservices bereits als Standard angesehen werden kann. Zudem kann SOAP unabhängig von dem verwendeten Übertragungsprotokoll verwendet werden.

Die SOAP-Nachricht selbst wird im MIME-Format¹⁵ als Multipart-Message¹⁶ übertragen. Das bedeutet, dass die Nachricht in mehrere Bereiche aufgegliedert wird. Im ersten Teil der Nachricht befindet sich die SOAP-Nachricht mit den QDX-Dokumenten. In den optional folgenden Teilen der Multipart-Message werden mögliche Attachments (siehe Kapitel 4.5) übertragen. Nachfolgend der Aufbau bzw. die Struktur der übertragenen Daten:

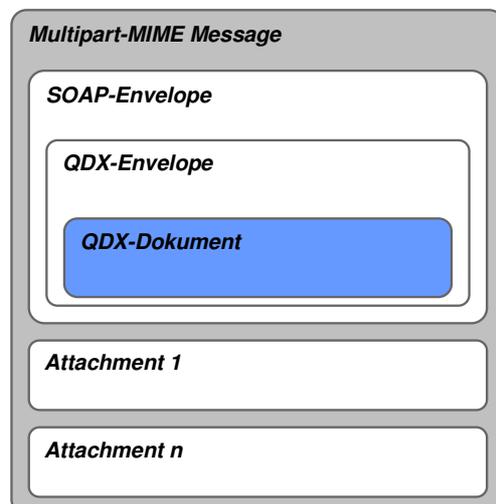


Abbildung 4-2: Struktur der übertragenen Daten

¹³ Vergleiche hierzu die Definition der verwendeten SOAP-Version 1.2 auf: <http://www.w3.org/TR/soap12/>

¹⁴ W3C = World Wide Web Consortium

¹⁵ Vergleiche hierzu die Definitionen unter: <http://tools.ietf.org/html/rfc2045>

¹⁶ Vergleiche hierzu die Definition unter: <http://tools.ietf.org/html/rfc2387>

Das Beispiel für eine Multipart-MIME Message sieht wie folgt aus:

```
MIME-Version: 1.0
Content-type: multipart/mixed; boundary=mime-boundary; type=text/xml

--mime-boundary
Content-type: text/xml; charset=utf-8
Content-Transfer-Encoding: 8bit

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:wsa="http://www.w3.org/2005/08/addressing">
  ...
</env:Envelope>

--mime-boundary
Content-type: image/jpg; name="bild.jpg"
Content-Transfer-Encoding: binary

...binäres jpg-Bild...

--mime-boundary
Content-type: video/mpeg; name="video.mpeg"
Content-Transfer-Encoding: binary

...binäres mpeg-Video...
```

Abbildung 4-3: Beispiel einer Multipart-MIME Message

4.1 Header

Im SOAP-Header werden Informationen übertragen, die für die weitere Verarbeitung in den Zwischensystemen (Routing-Systemen) unerlässlich sind. In erster Linie sind dies die Bezeichnung des Empfängers, die des Senders und der Typ des QDX-Dokuments. Letzteres ist für die CAQ-Systeme sinnvoll, die damit schon beim Lesen des SOAP-Envelopes wissen, welcher Dokumententyp verarbeitet werden soll.

Als Schema für die Übertragung der Metainformationen wurde der W3C-Standard „WS-Addressing“¹⁷ festgelegt. Der Standard WS-Addressing ist analog SOAP ein offener Standard und weit verbreitet – besonders bei der Umsetzungen von Webservices.

Die Daten innerhalb der Felder des WS-Addressing Standards sind in Form einer URI¹⁸ darzustellen. Als Schema dafür wurde URN¹⁹ festgelegt. Bei der zuständigen Organisation zur Verwaltung der Namensräume wurde dort der Namensraum „vda“ beantragt, innerhalb dessen der VDA²⁰ eigene Namensräume²¹ wie „qdx“ anlegen kann.

Die Daten im SOAP-Header müssen also wie folgt angegeben werden: „urn:vda:qdx:daten“.

¹⁷ WS-Addressing = Webservice-Addressing. Vergleiche hierzu die Definition auf: <http://www.w3.org/2002/ws/addr/>

¹⁸ URI = Uniform Resource Identifier

¹⁹ URN = Uniform Resource Name

²⁰ VDA = Verband der Automobilindustrie e.V.

²¹ iana.org

Daraus ergibt sich folgende Darstellung für den SOAP-Header:

```
<env:Header>
  <wsa:To env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">
    urn:vda:qdx:empfänger
  </wsa:To>
  <wsa:From env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">
    <wsa:Address>
      urn:vda:qdx:sender
    </wsa:Address>
  </wsa:From>
  <wsa:Action env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">
    urn:vda:qdx:qdx-dokument
  </wsa:Action>
</env:Header>
```

Abbildung 4-4: Aufbau SOAP-Header

- **Empfänger:** Als Empfänger ist die Kennnummer des Geschäftspartners einzutragen. Wird die Nachricht vom Kunden gesendet, steht hier die Lieferantenummer aus seinem Nummernkreis. Wird die Nachricht vom Lieferanten gesendet, so muss dieser die Kundennummer eintragen, die der Kunde bei sich führt.
Beispiel: 12345678A
- **Sender:** Als Sender ist die eigene Kennnummer einzutragen. Wird die Nachricht vom Kunden gesendet, steht hier seine Kundennummer aus seinem Nummernkreis. Wird die Nachricht vom Lieferanten gesendet, so muss dieser seine Lieferantenummer eintragen, die der Kunde für ihn führt.
Beispiel: 1234567800
- **QDX-Dokument:** Die Angabe des QDX-Dokuments erfolgt analog der Bezeichnung der Schemadateien.
Beispiel: QDXReport8D

Zudem kann die Angabe des Empfängers bzw. Senders um eine Systemkennung ergänzt werden. Die Systemkennung kann die intern bekannte Systemnummer oder ein beliebiger Wert sein, der dafür genutzt werden kann, QDX-Dokumente an das korrekte CAQ-System weiterzuleiten. Die Systemkennung wird vom Empfänger/Sender durch einen Punkt getrennt:

- **Systemkennung:** urn:vda:qdx:sender.systemkennung bzw. urn:vda:qdx:empfänger.systemkennung
Beispiel: urn:vda:qdx:12345678A.sys-id123

Die Angabe der Systemkennung ist optional. Wird aber mit dem SOAP-Umschlag eine Antwort auf einen Vorgang des Geschäftspartners gesendet, so ist hier die Systemkennung anzugeben, die mit der initialen Nachricht vom Geschäftspartner gesendet wurde (vergleiche hierzu Kapitel 2.1). Wurde dort keine Systemkennung angegeben, so kann diese auch in der Antwort leer bleiben.

4.2 Body

Der SOAP-Body enthält das QDX-Dokument. Allerdings wird das QDX-Dokument in einen weiteren Umschlag, den QDX-Envelope, gepackt. Der benötigte Inhalt des QDX-Envelopes hängt vom jeweiligen Integrationsprozess ab (aktive oder passive Kommunikation, siehe Kapitel 2). Der QDX-Envelope wird in Kapitel 4.3 in seinen Ausprägungen näher erläutert. Die WSDL hierzu befindet sich in Kapitel 5.1.

4.3 QDX-Envelope

Neben der Übertragung eines QDX-Dokuments ist es i. d. R. erforderlich, weitere Informationen für z. B. die Authentisierung mit zu senden. Aus diesem Grund werden das QDX-Dokument und die weiteren Informationen in einen eigenen Umschlag gepackt, den QDX-Envelope. Der Inhalt und damit der Aufbau des QDX-Envelopes hängen zunächst vom Integrationsprozess ab (siehe Kapitel 2). Bei einer aktiven Kommunikation ist ein QDX-Envelope ausreichend. Bei der passiven Kommunikation wird meist in einer Anfrage (Request) eine Methode

aufgerufen, deren Antwort (Response) die nötigen Informationen liefert. Daher wurden für die passive Kommunikation zwei QDX-Envelopes definiert.

Die Definitionen der zugehörigen XSD sind in Kapitel 5.2 zu finden.

4.3.1 QDXEnvelope (Aktive Kommunikation)

Für die aktive Kommunikation wird von den Elementen her derselbe QDX-Envelope verwendet, wie er für den HTTP-Request bei der passiven Kommunikation verwendet wird (siehe Kapitel 4.3.2). Um zu vermeiden, dass die Bezeichnung „Request“ bei OFTP zu Verwirrungen führt, wurde ein eigener QDX-Envelope mit der Bezeichnung QDXEnvelope definiert.

```
<env:Body>
  <qer:QDXEnvelope xmlns:qer="www.vda-qmc.de/qdx/QDXEnvelope_V2.0.xsd">
    <qdx:QDXReport8D xmlns:qdx="www.vda-qmc.de/qdx/QDXReport8D_V2.0.xsd">
      ...
    </qdx:QDXReport8D>
  </qer:QDXEnvelope>
</env:Body>
```

Abbildung 4-5: Aufbau QDXEnvelope

4.3.2 QDXEnvelopeRequest (Passive Kommunikation)

Das QDX-Dokument wird zusätzlich in einen QDX-Envelope gepackt. Bis auf das Dokument, welches für den HTTP-Request benötigt wird, enthält der Umschlag keine weiteren Informationen.

```
<env:Body>
  <qer:QDXEnvelopeRequest xmlns:qer="www.vda-qmc.de/qdx/QDXEnvelopeRequest_V2.0.xsd">
    <qdx:QDXReport8D xmlns:qdx="www.vda-qmc.de/qdx/QDXReport8D_V2.0.xsd">
      ...
    </qdx:QDXReport8D>
  </qer:QDXEnvelopeRequest>
</env:Body>
```

Abbildung 4-6: Aufbau QDXEnvelopeRequest

4.3.3 QDXEnvelopeResponse (Passive Kommunikation)

Zu der Antwort auf eine Anfrage müssen zum QDX-Dokument (oder anstelle dessen) Statusinformationen zur Verarbeitung der Anfrage übertragen werden. Daher enthält dieser QDX-Envelope einen Statuscode, der maschinell verarbeitet werden kann, eine Standardbeschreibung zum Statuscode sowie eine spezielle Beschreibung zum Statuscode, die ggf. eine Beschreibung zum verursachenden Fehler enthält. Das QDX-Dokument, als eigentliche Antwort auf die ursprüngliche Anfrage, wird mitgesendet, wenn die Anfrage erfolgreich war. Im Fehlerfall enthält der QDX-Envelope nur den Statuscode mit der Fehlerbeschreibung.

```
<env:Body>
  <qer:QDXEnvelopeResponse xmlns:qer="www.vda-qmc.de/qdx/QDXEnvelopeResponse_V2.0.xsd">
    <qer:Code>201</qer:Code>
    <qer:CodeDescription>Request succeeded</qer:CodeDescription>
    <qer:CodeDetails>Anfrage erfolgreich durchgeführt</qer:CodeDetails>
    <qdx:QDXComplaint xmlns:qdx="www.vda-qmc.de/qdx/QDXComplaint_V2.0.xsd">
      ...
    </qdx:QDXComplaint>
  </qer:QDXEnvelopeResponse>
</env:Body>
```

Abbildung 4-7: Aufbau QDXEnvelopeResponse

Alle Statuscodes sind in Kapitel 5.2.10 beschrieben.

4.4 Neue QDX-Dokumente

Zur Unterstützung des Integrationsprozesses (Kommunikationsprozesses) wurden weitere Methoden definiert, die u. a. sicherstellen, dass das QDX-Dokument im Zielsystem angekommen ist. Zudem müssen bei einer passiven Kommunikation diese und weitere Informationen zunächst abgefragt werden, bevor QDX-Dokumente gesendet und verarbeitet werden können. In diesem Kapitel werden die dafür notwendigen QDX-Dokumente definiert. Die Dokumente werden danach untergliedert, ob diese bereits die relevanten Informationen beinhalten oder lediglich zur Abfrage der Informationen benötigt werden.

Bei den QDX-Dokumenten, die fachliche Informationen transportieren, handelt es sich um die bereits existierenden QDX-Dokumente QDXComplaint, QDXReport8D, QDXShortConfirmation und QDXFieldFailureResponse. Die hier beschriebenen QDX-Dokumente beziehen sich nur auf die Verwendung von QDXComplaint und QDXReport8D. Sollen im Reklamationsprozess weitere QDX-Dokumente einbezogen werden, so ist zunächst zu prüfen, ob die Dokumente vom Kunden oder vom Lieferanten gesendet werden. Wird das neu verwendete Dokument vom Kunden gesendet, so sind die QDX-Dokumente, die zum QDXComplaint gehören (erkennbar an der Bezeichnung), zu adaptieren und entsprechend der Bezeichnung des neuen QDX-Dokuments anzupassen. Wird das neu verwendete Dokument vom Lieferanten gesendet, so ist analog mit den Dokumenten zu verfahren, die zum QDXReport8D gehören. Wird das neue Dokument gar von beiden Seiten genutzt, so sind alle beschriebenen Dokumente zu adaptieren und anzupassen.

Die Definition der Schemadateien zu den neuen QDX-Dokumenten ist in Kapitel 5.2 zu finden.

4.4.1 QDX-Dokumente mit technischen Informationen

Hier werden die QDX-Dokumente beschrieben, die Informationen zur technischen Verarbeitung an das empfangende CAQ-System transportieren.

4.4.1.1 QDXAcknowledgeComplaint

Das Dokument wird dazu verwendet, um den Erhalt und die korrekte Verarbeitung eines QDXComplaints im Zielsystem zu bestätigen. Hierdurch erhält der Sender der Beanstandung die Gewissheit, dass das QDX-Dokument beim Lieferanten angekommen ist und dieser mit der Fehlerbehebung beginnen kann.

Datenfeld	Pflicht/optional	Beschreibung
BuyerParty/ID	Pflicht	Kundennummer Zunächst muss angegeben werden, von welchem Kunden (bzw. Kundenwerk) eine Beanstandung bestätigt werden soll. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer.
BuyerParty/AdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht	Beanstandungsnummer Die Beanstandung, deren Verarbeitung bestätigt werden soll, muss hier angegeben werden. Hier ist die vom Kunden vergebene Nummer zu verwenden.
Complaint/RevisionID	optional	Versionsnummer Um die Beanstandung eindeutig zu identifizieren, kann die Versionsnummer optional mit angegeben werden.
Complaint/RevisionDateTime	Pflicht	Versionsdatum Um die Beanstandung eineindeutig identifizieren zu können, ist zusätzlich zur DocumentID das vom Kunden angegebene Versionsdatum der Beanstandung mit anzugeben.

Tabelle 4-1: Aufbau QDXAcknowledgeComplaint

4.4.1.2 QDXResetAcknowledgeStatusComplaint

Unter Umständen kann es notwendig sein, einen bereits verarbeiteten QDXComplaint erneut abzurufen. Mit diesem Dokument kann der Verarbeitungs- bzw. Sendestatus zurückgesetzt werden. Je nach Kommunikationsart (aktiv oder passiv) wird der QDXComplaint erneut vom Quellsystem gesendet bzw. kann vom Zielsystem erneut abgerufen werden.

In dem Dokument wird bewusst auf die Angabe einer Versionsnummer oder eines Versionsdatums verzichtet, da immer der aktuelle QDXComplaint gesendet wird.

Datenfeld	Pflicht/optional	Beschreibung
BuyerParty/ID	Pflicht	Kundennummer Zunächst muss angegeben werden, von welchem Kunden (bzw. Kundenwerk) die Beanstandung kommt, deren Bearbeitungsstatus zurückgesetzt werden soll. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer.
BuyerParty/AdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht	Beanstandungsnummer Die Beanstandung, deren Bearbeitungsstatus zurückgesetzt werden soll, muss hier angegeben werden. Hier ist die vom Kunden vergebene Nummer zu verwenden.

Tabelle 4-2: Aufbau QDXResetAcknowledgeStatusComplaint

4.4.1.3 QDXAcknowledgeReport8D

Analog der Bestätigung der fehlerfreien Verarbeitung eines QDXComplaints beim Lieferanten muss auch der Kunde die Verarbeitung eines QDXReport8D dem Lieferanten bestätigen. Damit hat der Lieferant die Gewissheit, dass sein 8D-Report im Zielsystem des Kunden angekommen ist und dort korrekt verarbeitet wurde. Dies kann unter Umständen für die Einhaltung von Fristen und damit für die Lieferantenbewertung relevant sein.

Ein Zurücksetzen des Bearbeitungsstatus' ist nicht vorgesehen. Sollte der 8D-Report erneut benötigt werden, so kann dieser im Rahmen eines aktualisierten QDXComplaints angefordert werden.

Datenfeld	Pflicht/optional	Beschreibung
SellerParty/ID	Pflicht	Lieferantennummer Der Lieferant, zu dessen 8D-Report die Verarbeitung bestätigt werden soll, muss angegeben werden. Hier muss der Kunde die von ihm für den Lieferanten vergebene Nummer eintragen.
SellerParty/AdditionalID	optional	Zusätzliche Lieferantennummer Zusätzlich zur Lieferantennummer kann noch eine weitere Nummer angegeben werden, die den Lieferanten identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht	Beanstandungsnummer Die Beanstandung zu dem 8D-Report, dessen Bearbeitungsstatus abgefragt wurde, ist anzugeben. Hier muss der Kunde seine eigene Nummer eintragen.
Report8D/DocumentID	Pflicht	8D-Reportnummer Der 8D-Report, dessen Verarbeitung bestätigt werden soll, muss angegeben werden. Hier ist die vom Lieferanten vergebene Nummer einzutragen.
Report8D/RevisionID	optional	Versionsnummer Die Versionsnummer des 8D-Reports kann zur eindeutigen Identifizierung mit angegeben werden.
Report8D/RevisionDateTime	Pflicht	Versionsdatum Das Versionsdatum des 8D-Reports ist zur eindeutigen Identifizierung zusätzlich zur DocumentID mit anzugeben.

Tabelle 4-3: Aufbau QDXAcknowledgeReport8D

4.4.1.4 QDXComplaintList

Dieses Dokument ist die einzige Ausnahme unter den QDX-Dokumenten mit technischen Informationen, denn es wird ausschließlich für die passive Kommunikation benötigt. Bei der aktiven Kommunikation wird der QDXComplaint direkt an den Lieferanten gesendet; bei der passiven Kommunikation muss der QDXComplaint vom Lieferanten abgerufen werden. Damit das anfragende System des Lieferanten weiß, wie viele und welche QDXComplaints abgerufen werden sollen, wird zunächst eine Liste mit den abzuholenden Beanstandungen abgerufen. Anhand der Liste weiß der Lieferant, wie viele QDXComplaints (und welche) abzurufen sind.

Datenfeld	Pflicht/optional	Beschreibung
BuyerParty/ID	Pflicht	Kundennummer Zunächst muss angegeben werden, zu welchem Kunden (bzw. Kundenwerk) Beanstandungen abgeholt werden können. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer
BuyerParty/AdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht (1..n)	Beanstandungsnummer Nummer der Beanstandung, die vom Lieferanten abgerufen werden kann. Hier ist die vom Kunden vergebene Nummer zu verwenden. Die Angabe der Version bzw. des Versionsdatum ist nicht vorgesehen, da immer der aktuellste QDXComplaint zur Verfügung gestellt wird. Das Datenfeld kann beliebig oft und muss mindestens einmal verwendet werden.

Tabelle 4-4: Aufbau QDXComplaintList

4.4.2 QDX-Dokumente zur Anfrage von Informationen

Im Rahmen der passiven Kommunikation ist es erforderlich, dass die Informationen (QDX-Dokumente) des Kunden vom Lieferanten angefragt werden müssen. Hierzu müssen Parameter übergeben werden, damit das verarbeitende System des Kunden das richtige QDX-Dokument mit den richtigen Informationen zurücksenden kann.

4.4.2.1 QDXComplaintListRequest

Dieses Dokument wird zur Abfrage der Liste der abzuholenden Beanstandungen (siehe Kapitel 4.4.1.4) eingesetzt.

Datenfeld	Pflicht/optional	Beschreibung
BuyerParty/ID	Pflicht	Kundennummer Der Lieferant muss angeben, von welchem Kunden (bzw. Kundenwerk) die Liste der abzuholenden Beanstandungen abgefragt werden soll. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer.
BuyerParty/AdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.

Tabelle 4-5: Aufbau QDXComplaintListRequest

4.4.2.2 QDXComplaintRequest

Für den Abruf eines QDXComplaints ist dieses Dokument zu verwenden.

Datenfeld	Pflicht/optional	Beschreibung
BuyerParty/ID	Pflicht	Kundennummer Der Lieferant muss angeben, von welchem Kunden (bzw. Kundenwerk) die Beanstandungen abgerufen werden soll. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer.
BuyerParty/AdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht	Beanstandungsnummer Nummer der Beanstandung, die abgerufen werden soll. Hier ist die vom Kunden vergebene Nummer zu verwenden. Die Angabe der Version bzw. des Versionsdatum ist nicht vorgesehen, da immer der aktuellste QDXComplaint zur Verfügung gestellt wird.

Tabelle 4-6: Aufbau QDXComplaintRequest

4.4.2.3 QDXAcknowledgeReport8DRequest

Um die Bestätigung zu erhalten, dass ein gesendeter 8D-Report im Zielsystem angekommen ist und korrekt verarbeitet wurde, muss dies mit diesem Dokument abgefragt werden.

Datenfeld	Pflicht/optional	Beschreibung
BuyerPartyID	Pflicht	Kundennummer Zunächst muss angegeben werden, von welchem Kunden (bzw. Kundenwerk) abgefragt werden soll, ob ein 8D-Report korrekt verarbeitet wurde. Die Kundennummer entspricht der vom Kunden für sich selbst vergebenen Nummer.
BuyerPartyAdditionalID	optional	Zusätzliche Kundennummer Zusätzlich zur Kundennummer kann noch eine weitere Nummer angegeben werden, die den Kunden identifiziert. Hier können übergreifende Nummern wie die DUNS-Nummer angegeben werden.
Complaint/DocumentID	Pflicht	Dokumentnummer Die Nummer des vom Kunden gesendeten QDXComplaints ist hier anzugeben. Es ist die vom Kunden vergebene Nummer zu verwenden. Hierbei handelt es sich meist um die Nummer des Prüfberichts oder der Reklamation.
Complaint/ComplaintItemID	Pflicht	Beanstandungsnummer Die Beanstandung zu dem 8D-Report, dessen Verarbeitungsstatus abgefragt wird, ist anzugeben. Hier muss der Lieferant die Nummer des Kunden eintragen.
Report8D/DocumentID	Pflicht	8D-Reportnummer Der 8D-Report, dessen Verarbeitung bestätigt werden soll, muss angegeben werden. Hier ist die vom Lieferanten selbst vergebene Nummer einzutragen.
Report8D/RevisionID	optional	Versionsnummer Die Versionsnummer des 8D-Reports kann zur eindeutigen Identifizierung mit angegeben werden.
Report8D/RevisionDateTime	Pflicht	Versionsdatum Das Versionsdatum des 8D-Reports ist zur eindeutigen Identifizierung zusätzlich zur DocumentID mit anzugeben.

Tabelle 4-7: Aufbau QDXAcknowledgeReport8Drequest

4.5 Attachments

In der Regel werden mit einem QDX-Dokument (teils größere) Attachments versendet. Alle Attachments zu einem QDX-Dokument werden innerhalb des Datenpakets in der MIME-Message angegeben. Hierbei spricht man auch von „SOAP with Attachments“, also von einer SOAP-Nachricht mit beliebig vielen Attachments.

Es kann sinnvoll sein, dass im Rahmen einer Partnervereinbarung die Größe der Attachments eingeschränkt wird (vergleiche hierzu Kapitel 5.4).

Die Referenzierung von Attachments erfolgt nicht auf Basis des Dateinamens sondern auf Basis einer innerhalb der MIME-Message eindeutigen ID. Hiermit können zwei Probleme umgangen werden:

- Identische Datei-Benennung für unterschiedliche Attachments.
- Verwendung von nicht ASCII-konformen Zeichen bei der Dateibenennung.

```
--mime-boundary
Content-type: image/jpg
Content-ID: 1
Content-Transfer-Encoding: binary

...binäres jpg-Bild....
```

Abbildung 4-8: Angabe von Attachments

Aufgrund dessen wurde auch die Angabe der Referenz in den QDX-Dokumenten angepasst; das Datenfeld „URL“ wurde umbenannt in „AttachmentID“, welches auf die Content-ID zu mappen ist.

```
<MimeReference>
  <AttachmentID>1</AttachmentID>
  <RevisionDateTime>2007-11-02T11:58:01.499+01:00</RevisionDateTime>
  <Filename>Bild1.jpg</Filename>
</MimeReference>
```

Abbildung 4-9: MIME-Referenz

Bei dem Versenden einer QDX-Nachricht gilt das „Brutto-Prinzip“ – es werden immer alle Informationen gesendet. Dieses Prinzip gilt auch für die Attachments, es werden also immer alle aktuellen Attachments übertragen. Wird ein Attachment in einer neuen Version einer QDX-Nachricht (z. B. QDXComplaint) nicht mehr referenziert bzw. mit gesendet, so wird die Verknüpfung zwischen dem Attachment und der aktuellen Version des QDX-Dokuments (hier des QDXComplaints) auch beim Geschäftspartner (hier beim Lieferanten) aufgelöst. Dem Lieferanten steht es frei, das Attachment auch ganz zu löschen. Allerdings ist dies im Rahmen einer lückenlosen Dokumentation/Archivierung nicht empfehlenswert.

²² Als Codierungsverfahren ist base64 zu verwenden. Vergleiche hierzu die Definitionen auf: <http://tools.ietf.org/html/rfc4648>

4.6 Zusammenfassung

Daraus ergibt sich folgende Darstellung einer SOAP-Nachricht mit QDX-Dokument im Multipart-MIME Format, inklusive Attachments. Ein Beispiel mit einem vollständigen QDX-Dokument ist in Kapitel 5.6 zu finden. Die einzelnen Elemente in der MIME-Message sind durch eine Trennlinie grafisch getrennt.

```
MIME-Version: 1.0
Content-type: multipart/mixed; boundary=mime-boundary; type=text/xml

-----mime-boundary
Content-type: text/xml; charset=utf-8
Content-Transfer-Encoding: 8bit
SOAP-Nachricht  
inkl. QDX-Dokument

<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <env:Header>
    <wsa:To env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">urn:vda.qdx:empfänger</wsa:To>
    <wsa:From env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">
      <wsa:Address>urn:vda.qdx:sender</wsa:Address>
    </wsa:From>
    <wsa:Action env:role="http://www.w3.org/2003/05/soap-envelope/role/next" env:relay="true">urn:vda.qdx:qdx-dokument</wsa:Action>
  </env:Header>
  <env:Body>
    <qer:QDXEnvelopeResponse xmlns:qer="www.vda-qmc.de/qdx/QDXEnvelopeResponse_V2.0.xsd">
      <qer:Code>201</qer:Code>
      <qer:CodeDescription>Request succeeded</qer:CodeDescription>
      <qer:CodeDetails>Anfrage erfolgreich durchgeführt</qer:CodeDetails>
      <qdx:QDXComplaint xmlns:qdx="www.vda-qmc.de/qdx/QDXComplaint_V2.0.xsd">
        ...
      </qdx:QDXComplaint>
    </qer:QDXEnvelopeResponse>
  </env:Body>
</env:Envelope>

-----mime-boundary
Attachment 1
Content-type: image/jpeg
Content-ID: 1
Content-Transfer-Encoding: binary

...binäres jpg-Bild....

-----mime-boundary
Attachment 2
Content-type: video/mpeg
Content-ID: 2
Content-Transfer-Encoding: binary

...binäres mpeg-Video...
```

Abbildung 4-10: SOAP-Nachricht im Multipart-MIME-Format

5 Anhang

5.1 WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:tns="urn:vda:qdx:QDXComplaintProcess_V2.0.wsdl"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:ns="urn:jai:qdxQDXEnvelopeRequest:2:0" xmlns:ns1="urn:jai:qdxQDXEnvelopeResponse:2:0"
xmlns:ns2="http://schemas.xmlsoap.org/soap/encoding/" name="QDXComplaintProcess"
targetNamespace="urn:vda:qdx:QDXComplaintProcess_V2.0.wsdl">
  <wsdl:types>
    <!-- Definition of request- and response-types-->
    <xsd:schema targetNamespace="urn:vda:qdx:QDXComplaintProcess_V2.0.wsdl"
elementFormDefault="qualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:envreq="urn:jai:qdxQDXEnvelopeRequest:2:0"
xmlns:envresp="urn:jai:qdxQDXEnvelopeResponse:2:0">
      <xsd:import namespace="urn:jai:qdxQDXEnvelopeRequest:2:0"
schemaLocation="../../XSD/QDXEnvelopeRequest_V2.0.xsd"/>
      <xsd:import namespace="urn:jai:qdxQDXEnvelopeResponse:2:0"
schemaLocation="../../XSD/QDXEnvelopeResponse_V2.0.xsd"/>
      <xsd:element name="QDXComplaintListRequest" type="envreq:QDXEnvelopeRequestType"/>
      <xsd:element name="QDXComplaintList" type="envresp:QDXEnvelopeResponseType"/>
      <xsd:element name="QDXComplaintRequest" type="envreq:QDXEnvelopeRequestType"/>
      <xsd:element name="QDXComplaint" type="envresp:QDXEnvelopeResponseType"/>
      <xsd:element name="QDXAcknowledgeComplaint"
type="envreq:QDXEnvelopeRequestType"/>
      <xsd:element name="QDXEnvelopeResponse" type="envresp:QDXEnvelopeResponseType"/>
      <xsd:element name="QDXResetAcknowledgeStatusComplaint"
type="envreq:QDXEnvelopeRequestType"/>
      <!-- Verwendung QDXEnvelopeResponse -->
      <xsd:element name="QDXReport8D" type="envreq:QDXEnvelopeRequestType"/>
      <!-- Verwendung QDXEnvelopeResponse -->
      <xsd:element name="QDXAcknowledgeReport8DRequest"
type="envreq:QDXEnvelopeRequestType"/>
      <xsd:element name="QDXAcknowledgeReport8D"
type="envresp:QDXEnvelopeResponseType"/>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="QDXComplaintListRequest">
    <wsdl:part name="parameter" element="tns:QDXComplaintListRequest"/>
  </wsdl:message>
  <wsdl:message name="QDXComplaintListResponse">
    <wsdl:part name="parameter" element="tns:QDXComplaintList"/>
  </wsdl:message>
  <wsdl:message name="QDXComplaintRequest">
    <wsdl:part name="parameter" element="tns:QDXComplaintRequest"/>
  </wsdl:message>
  <wsdl:message name="QDXComplaintResponse">
    <wsdl:part name="parameter" element="tns:QDXComplaint"/>
  </wsdl:message>
  <wsdl:message name="QDXAcknowledgeComplaintRequest">
    <wsdl:part name="parameter" element="tns:QDXAcknowledgeComplaint"/>
  </wsdl:message>
  <wsdl:message name="QDXAcknowledgeComplaintResponse">
    <wsdl:part name="parameter" element="tns:QDXEnvelopeResponse"/>
  </wsdl:message>
  <wsdl:message name="QDXResetAcknowledgeStatusComplaintRequest">
    <wsdl:part name="parameter" element="tns:QDXResetAcknowledgeStatusComplaint"/>
  </wsdl:message>
  <wsdl:message name="QDXResetAcknowledgeStatusComplaintResponse">
    <wsdl:part name="parameter" element="tns:QDXEnvelopeResponse"/>
  </wsdl:message>

```

Abbildung 5-1: WSDL Complaint Process Part 1

```

<wsdl:message name="QDXReport8DRequest">
  <wsdl:part name="parameter" element="tns:QDXReport8D"/>
</wsdl:message>
<wsdl:message name="QDXReport8DResponse">
  <wsdl:part name="parameter" element="tns:QDXEnvelopeResponse"/>
</wsdl:message>
<wsdl:message name="QDXAcknowledgeReport8DRequest">
  <wsdl:part name="parameter" element="tns:QDXAcknowledgeReport8DRequest"/>
</wsdl:message>
<wsdl:message name="QDXAcknowledgeReport8DResponse">
  <wsdl:part name="parameter" element="tns:QDXAcknowledgeReport8D"/>
</wsdl:message>
<wsdl:portType name="QDXComplaintProcessPortType">
  <wsdl:operation name="getQDXComplaintList">
    <wsdl:documentation>Operation is used for requesting a list of available QDXComplaints
    </wsdl:documentation>
    <wsdl:input message="tns:QDXComplaintListRequest"/>
    <wsdl:output message="tns:QDXComplaintListResponse"/>
  </wsdl:operation>
  <wsdl:operation name="getQDXComplaint">
    <wsdl:documentation>Operation is used for requesting a dedicated QDXComplaint
    </wsdl:documentation>
    <wsdl:input message="tns:QDXComplaintRequest"/>
    <wsdl:output message="tns:QDXComplaintResponse"/>
  </wsdl:operation>
  <wsdl:operation name="postQDXAcknowledgeComplaint">
    <wsdl:documentation>Operation is used for acknowledging of a QDXComplaint
    </wsdl:documentation>
    <wsdl:input message="tns:QDXAcknowledgeComplaintRequest"/>
    <wsdl:output message="tns:QDXAcknowledgeComplaintResponse"/>
  </wsdl:operation>
  <wsdl:operation name="postQDXResetAcknowledgeStatusComplaint">
    <wsdl:documentation>Operation is used for resetting the processing status of a QDXComplaint
    </wsdl:documentation>
    <wsdl:input message="tns:QDXResetAcknowledgeStatusComplaintRequest"/>
    <wsdl:output message="tns:QDXResetAcknowledgeStatusComplaintResponse"/>
  </wsdl:operation>
  <wsdl:operation name="postQDXReport8D">
    <wsdl:documentation>Operation is used for transmitting a QDXReport8D</wsdl:documentation>
    <wsdl:input message="tns:QDXReport8DRequest"/>
    <wsdl:output message="tns:QDXReport8DResponse"/>
  </wsdl:operation>
  <wsdl:operation name="getQDXAcknowledgeReport8D">
    <wsdl:documentation>Operation is used for requesting the acknowledgement of a QDXReport8D
    </wsdl:documentation>
    <wsdl:input message="tns:QDXAcknowledgeReport8DRequest"/>
    <wsdl:output message="tns:QDXAcknowledgeReport8DResponse"/>
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="QDXComplaintProcessBinding" type="tns:QDXComplaintProcessPortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="getQDXComplaintList">
    <soap:operation soapAction="urn:vda:qdx:QDXComplaintListRequest"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>

```

Abbildung 5-2: WSDL Complaint Process Part 2

```

<wsdl:operation name="getQDXComplaint">
  <soap:operation soapAction="urn:vda:qdx:QDXComplaintRequest"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="postQDXAcknowledgeComplaint">
  <soap:operation soapAction="urn:vda:qdx:QDXAcknowledgeComplaint"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="postQDXResetAcknowledgeStatusComplaint">
  <soap:operation soapAction="urn:vda:qdx:QDXResetAcknowledgeStatusComplaint"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="postQDXReport8D">
  <soap:operation soapAction="urn:vda:qdx:QDXReport8D"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="getQDXAcknowledgeReport8D">
  <soap:operation soapAction="urn:vda:qdx:QDXAcknowledgeReport8DRequest"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="QDXComplaintProcessService">
  <wsdl:port name="QDXComplaintProcessPort" binding="tns:QDXComplaintProcessBinding">
    <soap:address location="https://www.msg-systems.de"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

Abbildung 5-3: WSDL Complaint Process Part 3

5.2 Definition neuer XSD

Nachfolgend die Definition der neuen QDX-Dokumente.

5.2.1 QDXAcknowledgeComplaint

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ackco="urn:jai:qdxQDXAcknowledgeComplaint:2:0"
  targetNamespace="urn:jai:qdxQDXAcknowledgeComplaint:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXAcknowledgeComplaint schema for QDX-version 2.0
      This document ist used for acknowledgement of a QDXComplaint.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXAcknowledgeComplaint" type="ackco:QDXAcknowledgeComplaintType"/>
  <xsd:complexType name="QDXAcknowledgeComplaintType">
    <xsd:sequence>
      <xsd:element name="BuyerParty">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
                      type="xsd:string" use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Complaint">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="ComplaintItemID" type="xsd:string"/>
            <xsd:element name="RevisionID" type="xsd:string" minOccurs="0"/>
            <xsd:element name="RevisionDateTime" type="xsd:dateTime"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-4: XSD QDXAcknowledgeComplaint

5.2.2 QDXAcknowledgeReport8D

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ack8d="urn:jai:qdxQDXAcknowledgeReport8D:2:0"
  targetNamespace="urn:jai:qdxQDXAcknowledgeReport8D:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXAcknowledgeReport8D schema for QDX-version 2.0
      This document ist used for acknowledgement of an 8D-Report.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXAcknowledgeReport8D" type="ack8d:QDXAcknowledgeReport8DType"/>
  <xsd:complexType name="QDXAcknowledgeReport8DType">
    <xsd:sequence>
      <xsd:element name="SellerParty">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
                      type="xsd:string"
                      use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Report8D">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="RevisionID" type="xsd:string" minOccurs="0"/>
            <xsd:element name="RevisionDateTime" type="xsd:dateTime"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-5: XSD QDXAcknowledgeReport8D

5.2.3 QDXAcknowledgeReport8DRequest

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ack8dreq="urn:jai:qdxQDXAcknowledgeReport8DRequest:2:0"
targetNamespace="urn:jai:qdxQDXAcknowledgeReport8DRequest:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXAcknowledgeReport8DRequest schema for QDX-version 2.0
      This document ist used for requesting the acknowledgement of an 8D-Report.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXAcknowledgeReport8DRequest"
type="ack8dreq:QDXAcknowledgeReport8DRequestType"/>
  <xsd:complexType name="QDXAcknowledgeReport8DRequestType">
    <xsd:sequence>
      <xsd:element name="BuyerParty">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
type="xsd:string" use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Complaint">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="ComplaintItemID" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Report8D">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="RevisionID" type="xsd:string" minOccurs="0"/>
            <xsd:element name="RevisionDateTime" type="xsd:dateTime"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-6: XSD QDXAcknowledgeReport8Drequest

5.2.4 QDXEnvelope

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:env="urn:jai:qdxQDXEnvelope:2:0"
targetNamespace="urn:jai:qdxQDXEnvelope:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXEnvelope schema for QDX-version 2.0
      This envelope ist used for for any transmission (active communication).
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXEnvelope" type="env:QDXEnvelopeType"/>
  <xsd:complexType name="QDXEnvelopeType">
    <xsd:sequence>
      <xsd:any namespace="##other"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-7: XSD QDXEnvelope

5.2.5 QDXEnvelopeRequest

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:envreq="urn:jai:qdxQDXEnvelopeRequest:2:0"
targetNamespace="urn:jai:qdxQDXEnvelopeRequest:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXEnvelopeRequest schema for QDX-version 2.0
      This envelope ist used for any request (passive communication).
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXEnvelopeRequest" type="envreq:QDXEnvelopeRequestType"/>
  <xsd:complexType name="QDXEnvelopeRequestType">
    <xsd:sequence>
      <xsd:any namespace="##other"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-8: XSD QDXEnvelopeRequest

5.2.6 QDXEnvelopeResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:envresp="urn:jai:qdxQDXEnvelopeResponse:2:0"
targetNamespace="urn:jai:qdxQDXEnvelopeResponse:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXEnvelopeResponse schema for QDX-version 2.0
      This envelope ist used for any response to a request (passive communication).
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXEnvelopeResponse" type="envresp:QDXEnvelopeResponseType"/>
  <xsd:complexType name="QDXEnvelopeResponseType">
    <xsd:sequence>
      <xsd:element name="Code">
        <xsd:simpleType>
          <xsd:restriction base="xsd:integer">
            <xsd:pattern value="[0-9]{3}"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="CodeDescription"/>
      <xsd:element name="CodeDetails"/>
      <xsd:any namespace="##other" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-9: XSD QDXEnvelopeResponse

5.2.7 QDXComplaintList

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:coli="urn:jai:qdxQDXComplaintList:2:0"
targetNamespace="urn:jai:qdxQDXComplaintList:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXComplaintList schema for QDX-version 2.0
      This document ist used for transmitting a list of retrievable QDXComplaints.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXComplaintList" type="coli:QDXComplaintListType"/>
  <xsd:complexType name="QDXComplaintListType">
    <xsd:sequence>
      <xsd:element name="ComplaintList" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="BuyerParty">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="ID" type="xsd:string"/>
                  <xsd:element name="AdditionalID" minOccurs="0">
                    <xsd:complexType>
                      <xsd:simpleContent>
                        <xsd:extension
                          base="xsd:string">
                          <xsd:attribute
                            name="schemeAgencyID"
                            type="xsd:string"
                            use="required"/>
                        </xsd:extension>
                      </xsd:simpleContent>
                    </xsd:complexType>
                  </xsd:element>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Complaint">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID"
              type="xsd:string"/>
            <xsd:element name="ComplaintItemID"
              type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

Abbildung 5-10: XSD QDXComplaintList

5.2.8 QDXComplaintListRequest

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:colireq="urn:jai:qdxQDXComplaintListRequest:2:0"
targetNamespace="urn:jai:qdxQDXComplaintListRequest:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXComplaintListRequest schema for QDX-version 2.0
      This document ist used for requesting the list of available QDXComplaints (QDXComplaintList).
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXComplaintListRequest" type="colireq:QDXComplaintListRequestType"/>
  <xsd:complexType name="QDXComplaintListRequestType">
    <xsd:sequence>
      <xsd:element name="BuyerParty" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
                      type="xsd:string" use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

Abbildung 5-11: XSD QDXComplaintListRequest

5.2.9 QDXComplaintRequest

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:coreq="urn:jai:qdxQDXComplaintRequest:2:0"
targetNamespace="urn:jai:qdxQDXComplaintRequest:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXComplaintRequest schema for QDX-version 2.0
      This document ist used for requesting a dedicated and retrievable QDXComplaint.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXComplaintRequest" type="coreq:QDXComplaintRequestType"/>
  <xsd:complexType name="QDXComplaintRequestType">
    <xsd:sequence>
      <xsd:element name="BuyerParty">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
                      type="xsd:string" use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Complaint">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="ComplaintItemID" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-12: XSD QDXComplaintRequest

5.2.10 QDXResetAcknowledgeStatusComplaint

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:rasc="urn:jai:qdxQDXResetAcknowledgeStatusComplaint:2:0"
targetNamespace="urn:jai:qdxQDXResetAcknowledgeStatusComplaint:2:0">
  <xsd:annotation>
    <xsd:documentation>
      QDXResetAcknowledgeStatusComplaint schema for QDX-version 2.0
      This document ist used for resetting the acknowledge status of a QDXComplaint.
      Schema definition created by Verband der Automobilindustrie e.V.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="QDXResetAcknowledgeStatusComplaint"
type="rasc:QDXResetAcknowledgeStatusComplaintType"/>
  <xsd:complexType name="QDXResetAcknowledgeStatusComplaintType">
    <xsd:sequence>
      <xsd:element name="BuyerParty">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="ID" type="xsd:string"/>
            <xsd:element name="AdditionalID" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute name="schemeAgencyID"
type="xsd:string" use="required"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Complaint">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DocumentID" type="xsd:string"/>
            <xsd:element name="ComplaintItemID" type="xsd:string"/>
            <xsd:element name="RevisionID" type="xsd:string" minOccurs="0"/>
            <xsd:element name="RevisionDateTime" type="xsd:dateTime"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Abbildung 5-13: XSD QDXResetAcknowledgeStatusComplaint

5.3 Statuscodes für QDXEnvelopeResponse

5.3.1 Statuscodes für den Erfolgsfall

Statuscode	Statusbeschreibung	Bedeutung
200	Request of QDXComplaintList succeeded	Das Abrufen der Liste abzuholender Beanstandungen war erfolgreich.
201	Request of QDXComplaint succeeded	Das Abrufen einer Beanstandung war erfolgreich.
202	Transmission of QDXAcknowledgeComplaint succeeded	Das Übertragen der Verarbeitungsbestätigung einer Beanstandung war erfolgreich.
203	Transmission of QDXResetAcknowledgeStatusComplaint succeeded	Das Übertragen der Rücksetzung eines Verarbeitungsstatus' war erfolgreich.
204	Transmission of QDXReport8D succeeded	Das Übertragen eines 8D-Reports war erfolgreich.
205	Request of QDXAcknowledgeReport8D succeeded	Das Abrufen der Verarbeitungsbestätigung eines 8D-Reports war erfolgreich.

Tabelle 5-1: Statuscodes für den Erfolgsfall

5.3.2 Statuscodes für den Fehlerfall

Statuscode	Statusbeschreibung	Bedeutung
400	No QDXComplaints available	Es liegen keine Beanstandungen zur Abholung bereit.
401	The requested QDXComplaint is not available	Die angefragte Beanstandung steht nicht zur Verfügung.
402	Unknown customer identification	Die angegebene Kundennummer ist nicht bekannt.
403	Unknown additional customer identification	Die angegebene „Zusätzliche Kundennummer“ ist nicht bekannt.
404	Acknowledgement the specified QDXComplaint is not possible	Eine Bestätigung der angegebenen Beanstandung ist nicht möglich.
405	Unknown revision of the QDXComplaint	Die Version der Beanstandung ist nicht bekannt.
406	Unknown revision date of the QDXComplaint	Das Versionsdatum der Beanstandung ist nicht bekannt.
407	Unknown QDXReport8D	Der angegebene 8D-Report ist unbekannt.
408	Unknown revision of the QDXReport8D	Die Version der Beanstandung ist nicht bekannt.
409	Unknown revision date of the QDXReport8D	Das Versionsdatum der Beanstandung ist nicht bekannt.

Tabelle 5-2: Statuscodes für den Fehlerfall

5.4 Optionale individuelle Partnervereinbarungen

In der folgenden Tabelle werden die im Rahmen dieser Spezifikation notwendigen individuellen Partnervereinbarungen aufgeführt.

Thema	Einschränkung	Kapitel
Festlegung der Rolle des Kunden im Kommunikationsaufbau.	aktiv, passiv	2
Festlegung des Kommunikationsverfahrens.	OFTP, Webservice	3
Webservice: Wartezeiten definieren: - Zwischen dem Einstellen eines 8D-Reports und dem Abruf der Verarbeitungsbestätigung zu dem eingestellten 8D-Report (Wenn nicht anders vereinbart muss die Verarbeitungsbestätigung innerhalb von 90 Sekunden verfügbar sein). - Zwischen dem Zurücksetzen des Verarbeitungsstatus' eines QDXComplaints und dem erneuten Abruf des QDXComplaints (Wenn nicht anders vereinbart muss der Complaint innerhalb von 90 Sekunden verfügbar sein).	x Sekunden, Minuten, Stunden	3.1
Ist keine Verarbeitungsbestätigung verfügbar, muss der Prozess zwischen Kunde und Lieferant definiert werden: - zweites Mal abrufen - dann Eskalation und bilaterale Klärung zwischen Lieferant und Kunde (Lieferant ist hierfür verantwortlich).		
Webservice: Zugriffsintervalle für das Abrufen und das Einstellen von QDX-Dokumenten definieren.	nach x Minuten, zu bestimmten Uhrzeiten, Zeitfenster	3.1
Attachments einschränken.	Größe, Anzahl	4.5
Codierungsverfahren für Attachments muss definiert werden (Empfehlung: Base64).		

Tabelle 5-3: Individuelle Partnervereinbarungen

5.5 Glossar

Begriff	Beschreibung
8D-Report	Der 8D-Report wird zur Dokumentation der Fehlerbehebung durch den Lieferanten verwendet.
Beanstandung	Eine Beanstandung stellt eine Abweichung von der Spezifikation dar.
CAQ	Computer Aided Quality Assurance
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
Integrationsprozess	Unter einem Integrationsprozess ist ein systemübergreifender Prozess zu verstehen, dessen Kommunikation zwischen den Systemen aus XML-Nachrichten basiert.

Kommunikationssystem	Siehe Routing-System.
MIME	M ultipurpose I nternet M ail E xtensions
Multipart MIME	Eine Multipart-MIME Message enthält mehrere Teile (Bodyparty), in denen Daten (auch Binärdaten) übertragen werden.
OFTP	O DETTE F ile T ransfer P rotocol
Payload	Payload bezeichnet die Nutzdaten, die während einer Kommunikation zwischen zwei Geschäftspartnern ausgetauscht werden.
QDX	Q uality D ata eX change
Reklamationsprozess	Unter Reklamationsprozess ist der Prozess aus der Automobilindustrie zu verstehen, in dem der Kunde Mängel an gelieferten Teilen (Kaufteile) gegenüber dem Lieferanten reklamiert und dieser die Fehlerbehebung durchführt. Die Fehlerbehebung wird als 8D-Report dokumentiert.
Routing-System	Ein System, welches die Kommunikation mit internen und/oder externen Systemen steuert. Ein Routing-System leitet lediglich die Daten weiter, eine Validierung oder eine Verarbeitung der Daten erfolgt nicht.
SOAP	Netzwerkprotokoll zum Austausch von Daten auf Basis von XML. SOAP statt ursprünglich für S imple O bject A ccess P rotocol, ist aber seit 2003 offiziell kein Akronym mehr.
VDA	V erband d er A utomobilindustrie e.V.
VDA-QMC	Q uality M anagement C entre des VDA
WSDL	W eb s ervice D escription L anguage
XSD	X ML S cheme D efinition

Abbildung 5-14: Glossar

5.6 Download

Die Schemadateien, die Beispieldateien sowie die WSDL können unter folgender Adresse heruntergeladen werden: <http://www.gefeg.de/qdx/QDX2.0>