

Beschreibung der Datenübertragung an die SQR-BW (Version 2.7)

Inhalt

Inhalt.....	1
1. Voraussetzungen	2
2. Verbindungsaufbau und Datenübertragung	2
3. Namenskonventionen für Datenlieferungen	3
4. Ergebnisse der Verarbeitung	3
5. Sicherung der Original-Datei	4
6. Beispiele für den Ablauf der Datenübertragung mit Filezilla	4
6.1 Notarzt-Daten.....	4
6.2 RTW-Daten (Notfalleinsätze und Krankentransporte)	5
6.3 Leitstellendaten.....	6
7. Anhang.....	7
7.1 Einrichtung und Konfiguration des FTP Zugangs.....	7
7.2 Beispiel für die Konfiguration des FTP-Programms Filezilla	7
7.3 Besonderheiten	8
7.4 Aufbau der Result XML-Datei	8
7.5 Beispiel einer Result-Datei bei erfolgreicher Verarbeitung	9
7.6 Beispiel einer Result-Datei mit Warnungen und Fehlern.....	10
7.7 Download-Links gängiger FTP Programme.....	11

1. Voraussetzungen

Folgende Voraussetzungen müssen erfüllt sein, damit eine Datenübertragung an die SQR-BW erfolgen kann:

Technische Voraussetzungen	
1	Internetverbindung
2	Firewall: Das Protokoll FTP/SFTP/FTPS sowie der Port 21 (Protokolltyp FTP) bzw. der Port 22 (Protokolltyp SFTP) müssen freigegeben sein. Für den passiven Modus werden die Ports zwischen 4000 und 4256 verwendet und müssen daher ebenfalls freigegeben sein.
3	Software: FTP Client (z. B. Filezilla oder WinSCP)

Organisatorische Voraussetzungen	
4	Zugangsberechtigung: Jeder zur Datenübertragung registrierte Nutzer hat einen eigenen, geschützten Bereich auf dem SQR FTP-Server. Die Zugangsdaten müssen bei der SQR-BW angefordert werden.
5	Der Dateiname muss der Namenskonvention entsprechen (s. u.).

2. Verbindungsaufbau und Datenübertragung

Verbindungsparameter zum SQR FTP-Server:

Verbindungsparameter FTP	
FTP-Server	213.221.124.206
Port	21
Protokoll	FTP
Verschlüsselung	explizites FTP über TLS
Benutzer/Passwort	Benutzerdaten wie von SQR-BW zugeteilt

Verbindungsparameter SFTP	
FTP-Server	213.221.124.206
Port	22
Protokoll	SFTP
Benutzer/Passwort	Benutzerdaten wie von SQR-BW zugeteilt

Sobald eine Verbindung zu dem SQR-FTP-Server besteht, können die jeweiligen Dateien übertragen werden. Wichtig für die automatisierte Verarbeitung der Daten auf dem SQR-Server ist, dass die Daten in das **korrekte Verzeichnis** abgelegt werden.

Nach erfolgreichem Verbindungsaufbau wird für **NA-Standortordner** automatisch das Verzeichnis geöffnet, in welches die Daten zu übertragen sind (aktuell „/daten/mind3.1“).

Für Lieferungen in **RD-Standortordner** ist es notwendig, noch den entsprechenden Unterordner auszuwählen. Es wird nach erfolgreichem Verbindungsaufbau das Verzeichnis „/daten“ geöffnet, daraufhin muss entweder der Ordner „/mind3.1“ für RD-Daten oder „/mikd“ für KTP-Daten ausgewählt werden (siehe auch folgende Tabelle).

Datenquelle	Zielverzeichnis	Verzeichnis bei Verbindungsaufbau
MIND3.1-Notarzt Daten	/daten/mind3.1	/daten/mind3.1
MIND3.1-RTW-Daten (Notfalleinsätze)	/daten/mind3.1	/daten
MIKD-RTW-Daten (Krankentransporte)	/daten/mikd	/daten
Leitstellen-Daten	/daten	/daten

3. Namenskonventionen für Datenlieferungen

Notarzt- und RTW-Daten-Notfalleinsätze: <MIND31>_<JAHR>_<Projekt-ID>_<Freitext>.xml

RTW-Daten-Krankentransporte: <MIKD>_<JAHR>_<Projekt-ID>_<Freitext>.xml

Wenn zum Beispiel ein Notarzt-Standort mit Projekt-ID „12345“ eine Datei mit MIND3.1 Daten aus dem 1. Quartal 2018 liefert, sollte diese wie folgt benannt werden: „MIND31_2018_12345_LieferungQ1.xml“.

Leitstellendaten: <SQR_LST_XXX>_<Freitext>.csv

Der Dateiname beginnt immer mit dem Präfix „SQR_LST_XXX“ (der Platzhalter „XXX“ ist durch die Landkreis-kennung der abliefernden Leitstelle in Form des KFZ-Kennzeichens zu ersetzen). Nach dem Präfix kann jede beliebige, für Dateinamen gültige Zeichenfolge verwendet werden. Wenn zum Beispiel die Leitstelle Stuttgart eine Datei mit Daten des 1. Halbjahrs 2018 liefert, kann diese wie folgt benannt werden: „SQR_LST_S_1.HJ2018.csv“.



Bitte achten Sie darauf, dass Sie je Lieferung im Abschnitt <Freitext> des Dateinamens einen **eindeutigen Eintrag** wählen. Die Lieferung von Dateien mit komplett identischen Dateinamen ist aus IT-technischen Gründen grundsätzlich zu vermeiden.

4. Ergebnisse der Verarbeitung

Bei erstmaliger Übermittlung von Daten oder erstmaliger Lieferung in einem neuen Format werden diese auf Spezifikationskonformität überprüft. Wenn die Daten den Anforderungen entsprechen, werden alle folgenden in dem o. a. Verzeichnis abgelegten Daten desselben Formats vom SQR-Server automatisiert verarbeitet. Hierbei werden zunächst die Inhalte verifiziert und nach erfolgreicher Prüfung in das verarbeitende System übernommen. Die Verarbeitungsergebnisse jeder Datei werden auf dem SQR FTP-Server in einer Result XML-Datei in folgenden Verzeichnissen bereitgestellt:

Datenquelle	Datenverzeichnis
MIND3.1-Notarzt Daten	/daten/mind3.1/result
MIND3.1-RTW-Daten (Notfalleinsätze)	/daten/mind3.1/result
MIKD-RTW-Daten (Krankentransporte)	/daten/mikd/result
Leitstellen-Daten	/daten/result

Der Name der Result-Datei entspricht dem ursprünglichen Namen der gelieferten Datei mit der Erweiterung **".status.xml"**. Aufbau der XML-Datei und Beispiele befinden sich im Anhang.


Nach erfolgter Verarbeitung wird eine automatisch generierte E-Mail erzeugt, in welcher über die Anzahl der übermittelten Datensätze und ggf. vorhandene Auffälligkeiten bzgl. der Primärschlüssel informiert wird.



Eine automatisierte Verarbeitung von Leitstellendaten erfolgt derzeit noch nicht. Diese wird eingeführt, sobald alle Leitstellen spezifikationskonforme Daten übermitteln.

5. Sicherung der Original-Datei

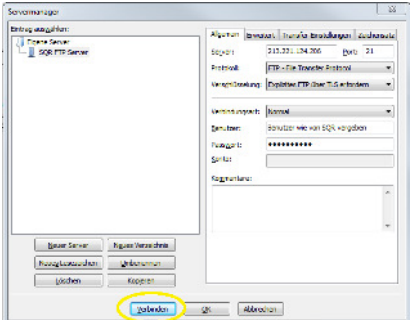
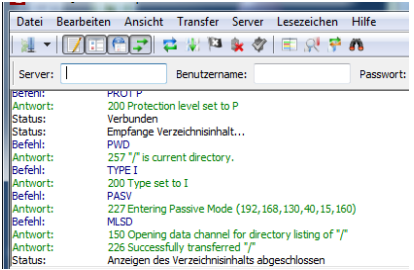
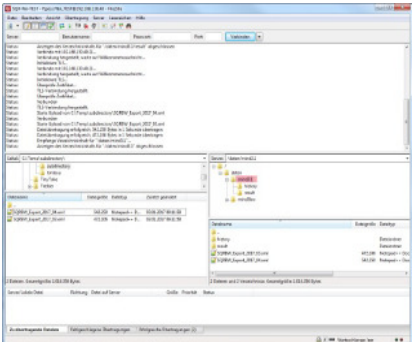
Alle verarbeiteten Dateien werden in ihrem Original-Zustand in einem History-Verzeichnis abgelegt. Dateien, die erfolgreich importiert werden konnten, erhalten hierbei den Zeitstempel ihrer Verarbeitung und die Erweiterung **".hist"**. Fehlerhafte Dateien werden mit der Erweiterung **".ERR"** gekennzeichnet. Dateien nach MIND3.1 Spezifikation, die zum Teil Protokolle enthalten, welche bei der Validierung Warnungen hervorrufen, werden mit der Erweiterung **„.PART“** gekennzeichnet.

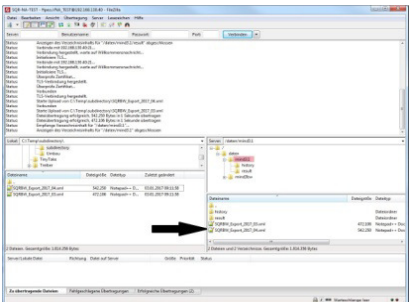
 Es liegt in der Verantwortung der einzelnen Nutzer, die Ergebnisse der Verarbeitung regelmäßig zu prüfen und die fehlerhaften Daten erneut zu senden.

Datenquelle	Datenverzeichnis
MIND3.1-Notarzt Daten	/daten/mind3.1/history
MIND3.1-RTW-Daten -Notfalleinsätze	/daten/mind3.1/history
MIKD-RTW-Daten -Krankentransporte	/daten/mikd/history
Leitstellen-Daten	/daten/history

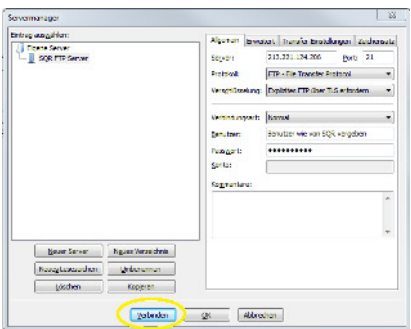
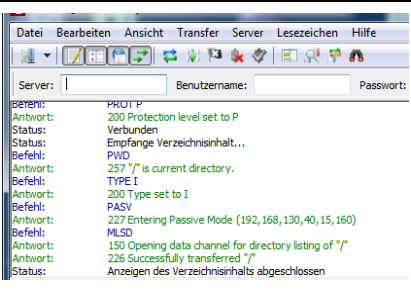
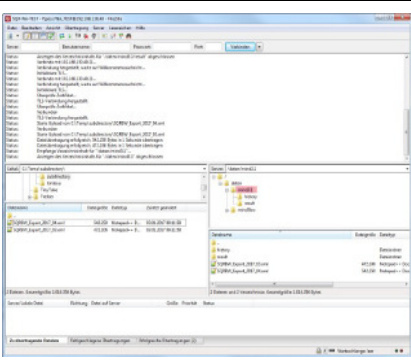
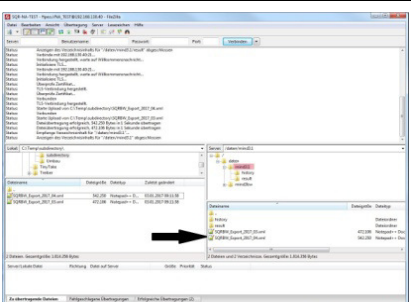
6. Beispiele für den Ablauf der Datenübertragung mit Filezilla

6.1 Notarzt-Daten

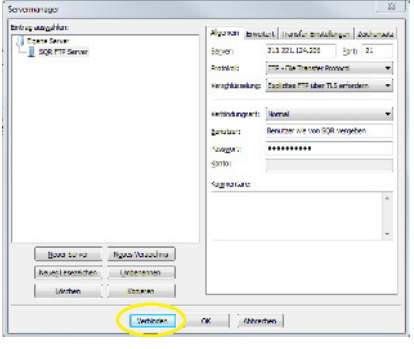
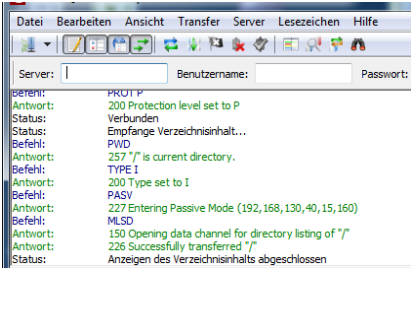
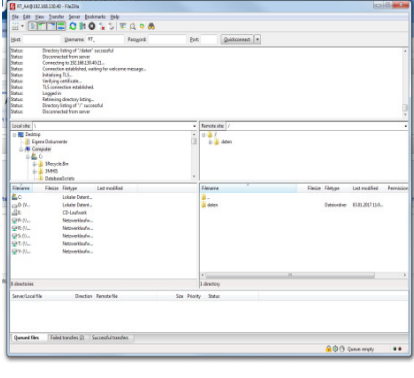
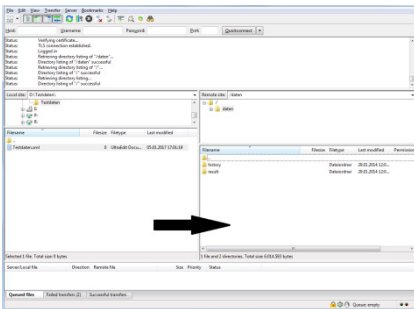
<p>1a</p>	<p>Verbindungsaufbau.</p> <p>Datei → Servermanager → Auswahl des "SQR FTP Servers"</p> <p>Button "Verbinden"</p>	
<p>1b</p>	<p>Ein erfolgreicher Verbindungsaufbau sieht in Filezilla wie folgt aus</p>	
<p>2</p>	<p>Nach dem Verbindungsaufbau sollte das Serververzeichnis sichtbar sein, in welches die Daten zu übertragen sind.</p>	

<p>3</p>	<p>Auswahl der zu übertragenden Daten auf dem lokalen Verzeichnis und Transfer auf den SQR File-Server.</p> <p>Mit Filezilla können hierbei die Ursprungs-Dateien markiert und in das nebenstehende rechte Fenster "gezogen" werden.</p>	
----------	--	---

6.2 RTW-Daten (Notfalleinsätze und Krankentransporte)


<p>1a</p>	<p>Verbindungsaufbau.</p> <p>Datei → Servermanager → Auswahl des "SQR FTP Servers"</p> <p>Button "Verbinden"</p>	
<p>1b</p>	<p>Ein erfolgreicher Verbindungsaufbau sieht in Filezilla wie folgt aus</p>	
<p>2</p>	<p>Nach dem Verbindungsaufbau ist der Inhalt des Serververzeichnis „daten“ sichtbar, in welchem entsprechend der Datenquelle der jeweilige Unterordner ausgewählt werden muss.</p>	
<p>3</p>	<p>Auswahl der zu übertragenden Daten auf dem lokalen Verzeichnis und Transfer auf den SQR File-Server.</p> <p>Mit Filezilla können hierbei die Ursprungs-Dateien markiert und in das nebenstehende rechte Fenster "gezogen" werden.</p>	

6.3 Leitstellendaten

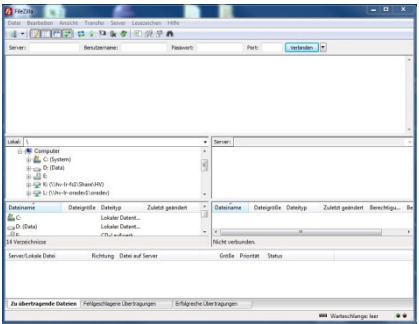
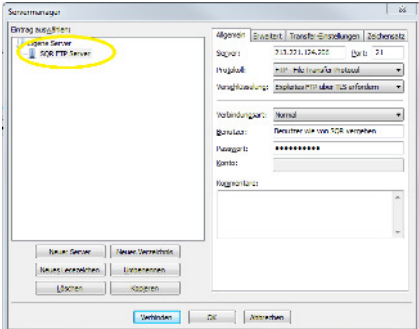
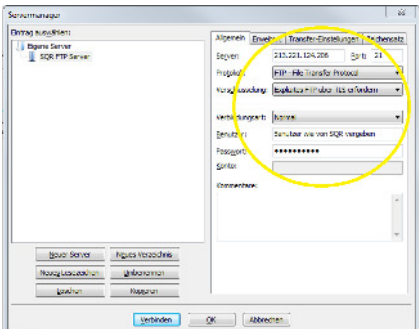
<p>1a</p>	<p>Verbindungsaufbau.</p> <p>Datei → Servermanager → Auswahl des "SQR FTP Servers"</p> <p>Button "Verbinden"</p>	
<p>1b</p>	<p>Ein erfolgreicher Verbindungsaufbau sieht in Filezilla wie folgt aus</p>	
<p>2</p>	<p>Nach dem Verbindungsaufbau sollte mindestens ein Serververzeichnis mit dem Namen "daten" sichtbar sein, in welches die Daten zu übertragen sind.</p>	
<p>3</p>	<p>Auswahl der zu übertragenden Daten auf dem lokalen Verzeichnis und Transfer auf den SQR File-Server.</p> <p>Mit Filezilla können hierbei die Ursprungs-Dateien markiert und in das nebenstehende rechte Fenster "gezogen" werden.</p>	

7. Anhang

7.1 Einrichtung und Konfiguration des FTP Zugangs

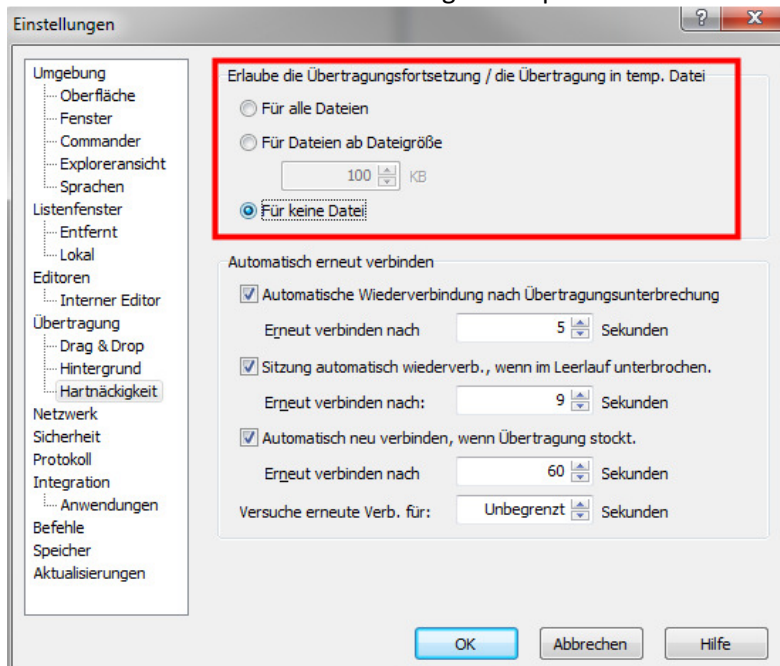
 Die Einrichtung bzw. Konfiguration des FTP Programms wird hier beispielhaft mit dem FTP Client Filezilla vorgenommen. Eine Download-Liste befindet sich unter 7.6 auf Seite 10.

7.2 Beispiel für die Konfiguration des FTP-Programms Filezilla

<p>1</p>	<p>Filezilla starten</p>	
<p>2</p>	<p>Servermanager aufrufen, neuen Server anlegen</p> <p>Datei → Servermanager → Button: "Neuer Server"</p> <p>Anschließend Server ggf. umbenennen (z.B. "Neuer Server" nach "SQR FTP Server")</p>	
<p>3</p>	<p>Verbindungsinformationen eintragen</p> <p>Server: 213.221.124.206</p> <p>Port: 21</p> <p>Protokoll: FTP - File Transfer Protokoll</p> <p>Verschlüsselung: Explizites FTP über TLS</p> <p>Verbindungsart: Normal</p> <p>Benutzer: <wie von SQR-BW mitgeteilt></p> <p>Passwort: <wie von SQR-BW mitgeteilt></p> <div style="background-color: #ffffcc; padding: 5px; margin-top: 10px;"> <p>Um die Verbindung zu testen, kann auch der folgende Benutzer verwendet werden:</p> <p style="text-align: center;">Benutzer: gast Passwort: gast</p> <p>Hiermit ist aber kein File-Upload möglich. Es können lediglich die Verzeichnis-Inhalte gelesen werden.</p> </div>	
<p>4</p>	<p>Konfiguration mit dem Button "OK" speichern</p>	

7.3 Besonderheiten

Bei dem Client-Tool WinSCP muss folgende Option in den Einstellungen geändert werden:



Dieser Parameter steht standardmäßig auf „Für Dateien ab Dateigröße“. Dies muss wie oben gezeigt angepasst werden.

7.4 Aufbau der Result XML-Datei

```
<?xml version="1.0" encoding="UTF-8"?>
  <!--W3C Schema für die Status-Datei eines Daten-Importes -->
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="result">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="datasource"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="datasource">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="id"/>
          <xs:element ref="file"/>
          <xs:element ref="details"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <!-- Eindeutige ID der Datei, hiermit lässt sich der Ursprung der Daten nachvollziehen -->
    <xs:element name="id">
      <xs:simpleType>
        <xs:restriction base="xs:string"/>
      </xs:simpleType>
    </xs:element>
    <!-- Original-Dateiname -->
    <xs:element name="file">
      <xs:simpleType>
        <xs:restriction base="xs:string"/>
      </xs:simpleType>
    </xs:element>
    <xs:element name="details">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="status"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:schema>
```



```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- Status der Verarbeitung -->
<xs:element name="status">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="message" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="stack" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="value"/>
            <xs:element ref="timestamp"/>
            <xs:element ref="elements"/>
            <xs:element ref="elementsAll"/></xs:sequence>
        </xs:complexType>
    </xs:element>
<!-- Message die bei der Verarbeitung der Datei aufgetreten ist -->
<xs:element name="message">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="type"/>
            <xs:element ref="text"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- Typ des Textes -->
<xs:element name="type">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="Info"/>
            <xs:enumeration value="Error"/>
            <xs:enumeration value="Warning"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Fehler-/Hinweistext -->
<xs:element name="text" type="xs:string"/>
<xs:element name="elements" type="xs:integer"/>
<xs:element name="elementsAll" type="xs:integer"/><!-- Stack-Trace -->
<xs:element name="stack" type="xs:string"/>
<!-- Verarbeitungsstatus der Datei -->
<xs:element name="value">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="ERROR"/>
            <xs:enumeration value="WARNING"/>
            <xs:enumeration value="DONE"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- Zeitstempel der Verarbeitung -->
<xs:element name="timestamp">
    <xs:simpleType>
        <xs:restriction base="xs:dateTime"/>
    </xs:simpleType>
</xs:element>
</xs:schema>

```

7.5 Beispiel einer Result-Datei bei erfolgreicher Verarbeitung

```

<?xml version="1.0" encoding="UTF-8"?><result>
  <datasource>
    <fileId>9fcc60ec-44a7-xxxx-977a-e89014894f22</fileId>
    <file>MIND31_2017_12345_Lieferung_01.xml</file>
    <details>
      <status>
        <value>DONE</value>
        <timestamp>2017-01-07T14:14:19</timestamp>
        <elements>539</elements>
        <elementsAll>539</elementsAll>
      </status>
    </details>
  </datasource>
</result>

```

Zeile	Erläuterung
3	Jede Datei erhält beim Import eine eindeutige ID. Somit lässt sich nachverfolgen, aus welcher Ursprungsdatei die Daten stammen.
4	Original-Dateiname
7	Status der Verarbeitung. Folgende Möglichkeiten <ul style="list-style-type: none"> • DONE: erfolgreich verarbeitet • ERR: Fehler bei der Verarbeitung aufgetreten • WARN: Warnung bei der Verarbeitung aufgetreten
8	Zeitstempel der Verarbeitung

7.6 Beispiel einer Result-Datei mit Warnungen und Fehlern

```
<?xml version="1.0" encoding="UTF-8"?>
<result>
  <datasource>
    <fileId>2fbaf1cc-1e97-4c0e-ba3e-842e99bb01b3</fileId>
    <file>testdaten_1projid-1plausiFehler-PRSS4-6.xml</file>
    <details>
      <status>
        <message>
          <type>Warning</type>
          <text>Detailprüfung Auftrag 6
            Fehler (8a-1): Bei Patientenkontakt muss die Zeit des Eintreffens beim Patienten angegeben
            werden!
          </text>
        </message>
        <message>
          <type>Error</type>
          <text>MIND Daten zu Einsatz/Auftrag unvollständig. Datasource/Primaerschluessel/Leitstelle
            2fbaf1cc-1e97-4c0e-ba3e-842e99bb01b3/6/XX
          </text>
        </message>
        <value>WARNING</value>
        <timestamp>2017-06-21T14:59:20</timestamp>
        <elements>2</elements>
        <elementsAll>3</elementsAll>
      </status>
    </details>
  </datasource>
  ...
  <stack>java.lang.Thread.run(Unknown Source)</stack>
  <stack>org.jboss.threads.JBossThread.run(JBossThread.java:122)</stack>
  <value>ERROR</value>
  <timestamp>2017-06-21T15:09:00</timestamp>
  ...
</result>
```

Zeile	Erläuterung
8	Jede bei der Verarbeitung aufgetretene Nachricht wird durch ein <message> Tag protokolliert
9	Typ der Message (Info, Error, Warning)
10	Fehlertext mit Angabe der Zeile die den Fehler enthalten hat. Hier wurde z. B. trotz Patientenkontakt kein Zeitpunkt für das Eintreffen beim Patienten übermittelt.
27	Sollte ein Fehler-Stack vorhanden sein, wird dieser unter dem Tag <stack> abgelegt. Hier können ggf. weitere Details zur Fehlerermittlung herausgelesen werden, insbesondere wenn es sich um schwerwiegende Fehler handelt.

7.7 Download-Links gängiger FTP Programme

FTP Programm	Download-Link
Filezilla	http://www.filezilla.de/download.htm
WinSCP	http://winscp.net/eng/download.php
CuteFTP	http://www.cuteftp.com/cuteftp/