

Schnittstellenspezifikation

VDDS-media 1.3



Schnittstelle zum Austausch von Multimedia-Daten
zwischen Praxisverwaltungssoftware und Röntgen-/Kamerasoftware

Inhaltsverzeichnis

Vorbemerkungen	3
Leistungsumfang	3
Stufenweise Realisierung aus Sicht der PVS	4
Stufenweise Realisierung aus Sicht der BVS	4
Allgemeine Voraussetzungen	5
1. VDDS-Konfigurationsdatei (VDDS_MMI.INI)	6
1.1. PVS-Einträge	7
1.2. BVS-Einträge	7
2. Übergabe der Patientendaten an die BVS	8
2.1. Übergabedatei	8
2.2. PATID-Server	10
3. Übergabe der Bildinformationen an die PVS	11
3.1. Übergabedatei der PVS an die BVS	12
3.2. Übergabedatei der BVS an die PVS	12
3.3. PVS-Importmodul für Bildinformationen	13
4. Übergabe/Darstellung der Bilder	13
4.1. Bilddarstellung mit dem Viewer der PVS	14
4.2. Bilddarstellung mit dem Viewer der BVS	15
5. Optionale Integration eines Bildarchivs	15
5.1. Anmeldung des Archivs/ Kommunikationsmodell	15
5.2. Übergabe der Patientendaten an das Archiv	15
5.3. Übergabe der Bildinformationen an das Archiv und die PVS	16
5.4. Archiv-Importmodul für Bildinformationen	16
5.5. Übergabe und Darstellung der Bilder mit dem Viewer der PVS	16
5.6. Darstellung der Bilder mit dem Viewer der BVS oder des Archivs	16
6. Häufig gestellte Fragen/FAQ	16
Impressum	17

Anlage 1	18
VDDS-Konfigurationsdatei (VDDS_MMI.INI) - PVS-Einträge	18
Anlage 2	19
VDDS-Konfigurationsdatei (VDDS_MMI.INI) - BVS-Einträge	19
Anlage 3	21
Übergabe der Patientendaten an die BVS	21
Anlage 4	23
PATID Server / IEXPORT-Modul	23
Anlage 5	25
Übergabedatei der PVS an die BVS	25
Anlage 6	26
Übergabedatei der BVS an die PVS	26
Anlage 7	29
Bilddarstellung mit dem Viewer der PVS	29
Anlage 8	30
Änderungen an der VDDS-media Schnittstelle in Version 1.3 gegenüber 1.2	30
Anlage 9	32
Synchronisationsdiagramme	32
Ablauf einer Kommunikation ohne Event	32
Ablauf einer Kommunikation mit Event	33

Sachwort-Erklärung:

Bildinformation	Daten, die ein Bild beschreiben und identifizierbar machen, z.B. Erzeugungsdatum, ID, Kommentar, Zahnangaben; Bildinformationen werden über die in den Anhängen beschriebenen INI-Dateien ausgetauscht und sind keine Kopien der Pixel eines Bildes
BVS	Bildverarbeitungssystem
ID	Eindeutiges Identifikationskennzeichen (nur numerisch)
Key	Schlüssel
Kopie der Bilddaten	Datei mit den Bildpunkten (Pixeln), die als Kopie vom Originalbild erzeugt wurde
Linux, Unix, Mac OSX	Betriebssysteme für PC's
PVS	Praxisverwaltungssystem
Thumbnail	Kleines, sehr schnell ladbares Vorschaubild
VDDS	Verband Deutscher Dentalsoftware Unternehmen e.V.
Viewer	Bildbetrachter
Workflow	Arbeitsablauf

Vorbemerkungen

Im Folgenden wird die patientenverwaltende Software mit **PVS** abgekürzt und die bildverwaltende Software (Röntgen-/Kamerasoftware, usw.) mit **BVS**. Es wird die Schnittstelle in der Version 1.3 beschrieben.

Die Dokumentation bezieht sich auf den Austausch von Patientendaten und Bildinformationen. Andere Multimedia-Objekte (Videodateien, Tondateien, Textdateien) können mit einbezogen werden, sofern sie vorliegen.

Ein Hauptziel bei der Entwicklung der Schnittstelle war es, den teilnehmenden Firmen bzw. Programmen eine für den freien Wettbewerb nötige maximale Freiheit in der anbietbaren Anwenderfunktionalität bei der Umsetzung eines standardisierten Datenaustausches zu ermöglichen.

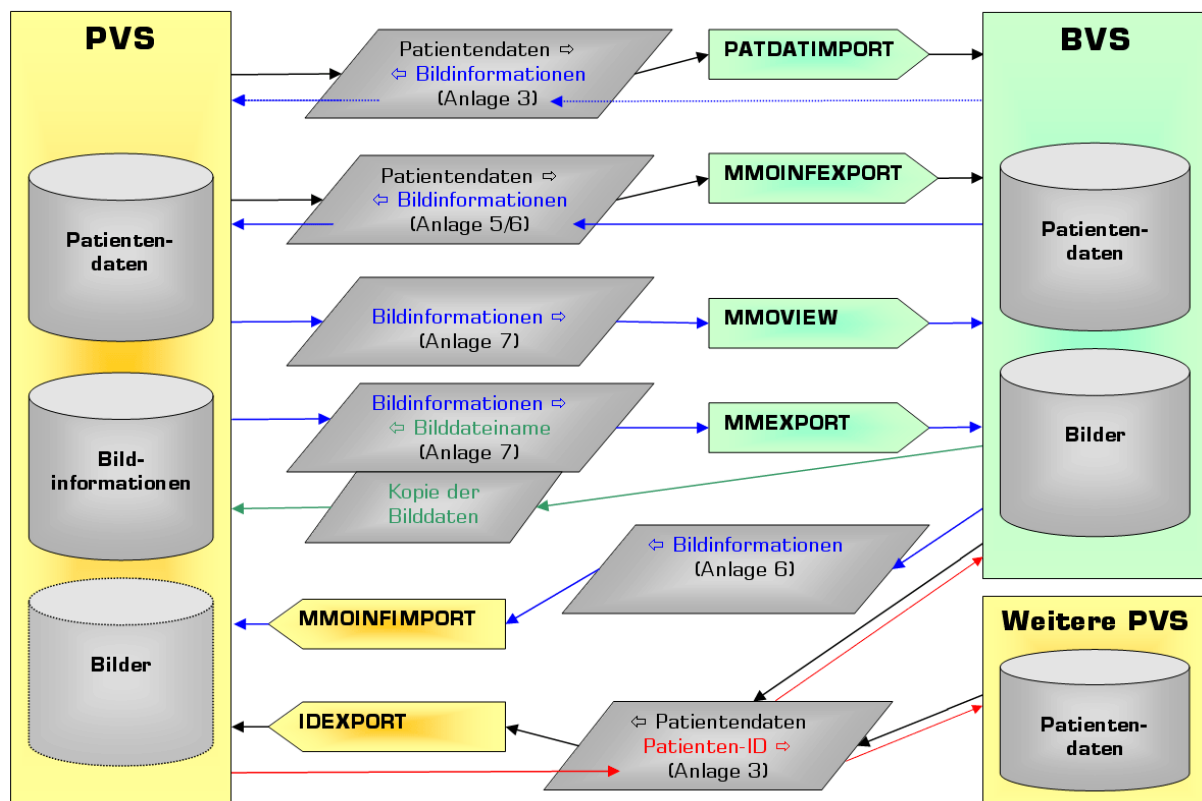


Die Pflege und Weiterentwicklung dieser Schnittstelle obliegt ausschließlich dem VDDS.

Leistungsumfang

- ☑ Es werden Patientendaten von der PVS an die BVS übergeben.
- ☑ Es werden Bilder/Bildinformationen (Zähne, Kommentar, Erstellungsdatum/Uhrzeit, Bildtyp) und andere Multimedia-Objekte wie Videos, Textdokumente, Tondateien von der BVS an die PVS übergeben.
- ☑ Die Schnittstelle ist betriebssystemunabhängig.
- ☑ Beim Anwender entsteht kein zusätzlicher Installationsaufwand.
- ☑ Es können mehrere PVS/BVS miteinander kommunizieren.
- ☑ Optionale Integration eines Bildarchivs.

Übersichtsdiagramm VDDS-media



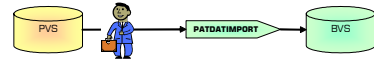
Stufenweise Realisierung aus Sicht der PVS



Die PVS kann ihren Anwendern unterschiedliche Ausbaustufen der Schnittstelle zur Verfügung stellen:

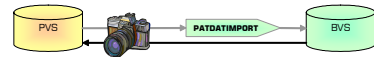
Stufe 1:

Die PVS ruft das Importmodul der BVS zur Übergabe der Patientendaten auf.



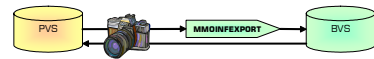
Stufe 2:

Die PVS wertet die in der Patientendaten-Datei zurückgelieferten Bildinformationen aus.



Stufe 3:

Die PVS ruft das Modul zum Export der Bildinformationen der BVS auf.



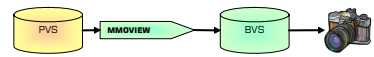
Stufe 4:

Die PVS ruft das Modul für den Export von Bildkopien auf und stellt diese dann im eigenen Viewer dar.



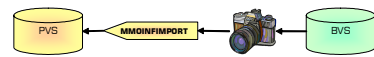
Stufe 5:

Die PVS ruft den Viewer der BVS auf.



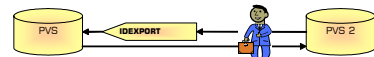
Stufe 6:

Die PVS stellt ein eigenes Modul für den Import von Bildinformationen zur Verfügung.



Stufe 7:

Die PVS stellt ein Modul für den Export von Patientendaten zur Verfügung (PVS ist als ID-Server geeignet).



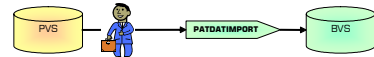
Stufenweise Realisierung aus Sicht der BVS



Die BVS kann ihren Anwendern unterschiedliche Ausbaustufen der Schnittstelle zur Verfügung stellen:

Stufe 1:

Die BVS stellt ein Importmodul für Patientendaten zur Verfügung.



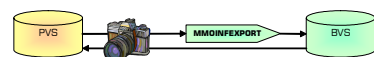
Stufe 2:

Die BVS liefert die Bildinformationen in der Patientendaten-Datei zurück.



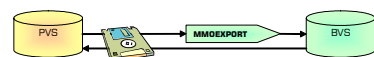
Stufe 3:

Die BVS stellt ein Exportmodul für Bildinformationen zur Verfügung.



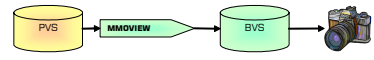
Stufe 4:

Die BVS stellt ein Modul für den Export von Bildkopien zur Verfügung.



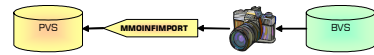
❑ **Stufe 5:**

Die BVS stellt einen Viewer zur Verfügung, den die PVS aufrufen kann.



❑ **Stufe 6:**

Die BVS ruft das Importmodul für Bildinformationen der PVS auf, wenn sich Bildinformationen ändern oder ein neues Bild erstellt wird.



Welche Stufen realisiert sind, kann die PVS/BVS anhand der Einträge in der VDDS_MMI.INI erkennen und die entsprechenden Module aufrufen. Da der Implementierungsaufwand nicht hoch ist, wird empfohlen seitens der BVS alle Stufen zu realisieren und seitens der PVS die Stufen 1, 2, 3 und 6 sowie wahlweise die Stufe 4 oder 5 (siehe Synchronisationsdiagramme in Anlage 9 auf Seite 32).

Allgemeine Voraussetzungen

- ☑ Alle hier aufgeführten Im- und Exportmodule müssen bei erfolgreicher Ausführung **0** und im Fehlerfall einen Wert größer gleich **1** zurückliefern (so kann ein Modul zu internen und Wartungszwecken unterschiedliche Fehlercodes liefern). Dies muss sowohl im **ERRORLEVEL** als auch in den Übergabedateien erfolgen, falls die PVS/BVS den **ERRORLEVEL** nicht auswerten kann.
- ☑ Benötigt eine Software nur die Patientendaten, müssen nur die unter „Übergabe der Patientendaten an die BVS“ (Abschnitt 2.) genannten Punkte erfüllt werden.
- ☑ Die Im- und Exportmodule sollten kleine Programme sein (<500kb), damit ein Aufruf schnell erfolgen kann (es sei denn, es handelt sich um die BVS selbst, die als Viewer aufgerufen wird).
- ☑ Nach dem Installieren der PVS und der BVS sollte keine weitere Einrichtung der Schnittstelle mehr erforderlich sein. Die PVS sollte dem Anwender eine Auswahl der unter VDDS/BVS eingetragenen Programme anbieten, damit er entscheiden kann, welche Schnittstellen er am jeweiligen Arbeitsplatz aktivieren möchte. Die BVS sollte dem Anwender eine Auswahl der unter VDDS/PVS eingetragenen Programme anbieten, damit der Anwender entscheiden kann, an welche PVS die BVS ihre Bildinformation zurückliefern muss. Wie diese Auswahl in der BVS/PVS gestaltet wird, obliegt dem jeweiligen Programm (z.B. schon bei der Installation oder im Programm konfigurierbar).



Bei Vorhandensein eines Archivs, muss die Kommunikation immer über das Archiv laufen d.h. die PVS/BVS ruft nur noch die Module des Archivs auf (siehe Abschnitt 5. „Optionale Integration eines Bildarchivs“ ab Seite 15).

- ☑ Für den Fall, dass die Module aufwendige Datenbanktreiber laden müssen, deren Ladezeiten über ein akzeptables Maß hinausgehen, müssen die Module nach dem ersten Aufruf ein Programm im Speicher installieren, welches dort verbleibt und die Befehle der Module ausführt.
- ☑ Im Netz müssen die Im- und Exportmodule von allen Stationen aufrufbar sein, an denen Bilder angesehen bzw. übertragen werden sollen, d.h. dass auch die benötigten Datenbanktreiber an diesen Stationen installiert sein müssen, wenn die Module sie benötigen. Alternativ können die Im/Exportmodule über entsprechende Netz-

protokolle mit einem Serverprogramm kommunizieren. Das bedeutet, dass z.B. auf der Station, wo die BVS installiert ist, zusätzlich ein Serverprogramm läuft. Dieses Programm bearbeitet Anfragen der von der PVS an verschiedenen Stationen aufgerufenen BVS-Im- und Exportmodule.

- ☑ Alle Übergabedateien (.ini) sollten in einem temporären Verzeichnis auf dem lokalen Rechner liegen, damit Stationsunabhängigkeit gewahrt bleibt. Liegen die Dateien im Netz, muss sichergestellt sein, dass jede Station diese anders benennt. Für das Löschen der temporären Dateien im Rahmen dieser Schnittstellen-Beschreibung ist jeweils der Erzeuger selbst verantwortlich.
- ☑ Es ist zu beachten, dass in einer Praxis mehrere BVS- und PVS-Systeme mit jeweils mehreren Hauptbehandlern/Praxisnummern installiert sein können. Eine PVS muss daher z.B. alle vom Anwender gewünschten BVS-Exportmodule der Reihe nach aufrufen, um die gesamten Bilder eines Patienten zu erhalten, es sei denn, ein Archiv ist installiert. Umgekehrt muss die BVS alle Importmodule der von Anwender gewünschten PVS-Systeme aufrufen.
- ☑ Ist eine PVS gleichzeitig BVS, muss sie ihre Multimedia-Objekte auch als BVS zur Verfügung stellen. Die PVS/BVS muss dafür die entsprechenden Im- und Exportmodule definieren und in die VDDS_MMI.INI (Abschnitt 1.) eintragen.
- ☑ Bei Windows/DOS-Betriebssystemen müssen alle verwendeten Pfade in der kurzen 8.3-Notation geführt werden.
- ☑ Als Zeichensatz ist grundsätzlich ANSI vorausgesetzt bzw. zu verwenden.
- ☑ Für Key's gilt, dass sie **case-insensitive** sind.
- ☑ In den Zuweisungen der INI-Dateien dürfen keine Leerzeichen nach dem „=" vorkommen.

1. VDDS-Konfigurationsdatei (VDDS_MMI.INI)

In der VDDS_MMI.INI tragen sich alle PVS/BVS mit ihren Namen und Modulen ein. Alle Moduleinträge beinhalten den kompletten Pfad zum Aufruf des Moduls. Die Konfigurationsdatei steht in einem von allen Programmen zugänglichen Verzeichnis des Betriebssystems.

Für Windows ist dies das Hauptverzeichnis des Betriebssystems welches die API-Funktion „*GetWindowsDirectory*“ zurückliefert (Achtung: bei Terminal Server 2003 werden hier u.U. benutzerspezifische Pfade geliefert). DOS-Programme können die Umgebungsvariablen `windir`, `winbootdir` oder `systemroot` verwenden, müssen aber den Pfad auf die Datei editierbar machen.

Bei Linux, Unix und Mac OSX-Systemen wird das Verzeichnis `/etc/vdds` verwendet. Für Teilnehmer an der Schnittstelle gilt folgendes Vorgehen:

- ☑ Anlegen einer Gruppe `vdds` durch das erste zu installierende Programm
- ☑ Anlegen des Verzeichnisses `/etc/vdds` durch das erste zu installierende Programm
- ☑ Anlegen der `VDDS_MMI.INI` Datei in `/etc/vdds` mit Gruppenzugehörigkeit `vdds` durch das erste zu installierende Programm (mit Zugriffsrechten `664/rw-rw-r--`: Lesen und Schreiben für root, user/group und nur lesen für alle anderen)
- ☑ Die Benutzer des Programms der Gruppe `vdds` zufügen

1.1. PVS-Einträge



Detaillierte Dateninformationen sind der Anlage 1 „VDDS-Konfigurationsdatei (VDDS_MMI.INI) - PVS-Einträge“ Seite 18 zu entnehmen.

Die Ausschnitte aus den INI-Dateien werden mit Hintergrundfarbe gelb dargestellt, wenn sie sich auf die VDDS_MMI.INI beziehen. Andere INI-Dateien haben einen grauen Hintergrund.

```
[PVS]
NAME1=PVS-Sektion 1
NAME2=PVS-Sektion 2
```

```
[PVS-Sektion 1]
NAME=
MMOINFIMPORT=
MMOINFIMPORT_OS=
MMOINFIMPORT_EVENT=
THUMBNAILS=
THUMBNAILSX=
THUMBNAILSX=
THUMBNAILSX=
VERSION=
STAGES=
```

```
[PVS-Sektion 2]
...
```

1.2. BVS-Einträge



Detaillierte Dateninformationen sind der Anlage 2 „VDDS-Konfigurationsdatei (VDDS_MMI.INI) - BVS-Einträge“ auf Seite 19 zu entnehmen.

Will sich eine PVS/BVS neu in die VDDS_MMI.INI eintragen, muss sie zuerst die PVS/BVS-Sektion nach einer noch nicht vergebenen Namen-ID durchsuchen (im Beispiel Anlage 2, Seite 19: NAME3) und dann ihre Schnittstellendaten in eine eigene Sektion, deren Name im Key NAME3 steht, einschreiben.

```
[BVS]
NAME1=BVS-Sektion 1
NAME2=BVS-Sektion 2
```

```
[BVS-Sektion 1]
NAME=
PATDATIMPORT=
PATDATIMPORT_OS=
PATDATIMPORT_EVENT=
MMOINFEXPORT=
MMOINFEXPORT_OS=
MMOINFEXPORT_EVENT=
MMOEXPORT=
MMOEXPORT_OS=
MMOEXPORT_EVENT=
MMOVVIEW=
MMOVVIEW_OS=
MMOVVIEW_EVENT=
PVSLIMIT=PVS-Sektion
SUPPORTSELECT=
SUPPORTNEW=
SUPPORTINFO=
SUPPORTDEPTH=
SUPPORTTHUMBNAILS=
STAGES=
VERSION=
```

```
[BVS-Sektion 2]
...
```



Die Sektionsnamen sollten nicht abgekürzt werden und den Firmen- und Produktnamen enthalten, (z.B. PRAXISSOFT_WINZAHN) damit eine Namensgleichheit ausgeschlossen werden kann.



Wenn ein Windows-Programm ein DOS-Modul aufruft, muss sichergestellt sein, dass kein DOS-Fenster sichtbar wird, es sei denn es handelt sich um den Viewer. Das kann durch entsprechende Parameter im CreateProcess-Befehl der Windows-API sichergestellt werden. Siehe Abschnitt FAQ auf Seite 17. Wird eine PVS/BVS deinstalliert, müssen sämtliche Einträge des Programms aus der VDDS_MMI.INI gelöscht werden.

Bitte beachten Sie:



Die Vorgehensweise kann Sprünge verursachen - NAME2 wurde gelöscht und es gibt dann nur noch NAME1 und NAME3. Eine PVS/BVS darf sich nicht auf eine „nahtlose“ Vergabe der Nummern verlassen (um diese z.B. als Ersatz des Sektionsnamens zu speichern)!



Sollte eine BVS seinen PVSLIMIT Eintrag gesetzt haben, so ist es bei nicht mehr vorhandenem PVS für die VDDS-media-Schnittstelle gesperrt. Andere PVS kommunizieren zwar sowieso nicht mit ihm, das BVS sollte dann aber einen entsprechenden Hinweis ausgeben. Die Patientennummern dürfen auf keinen Fall einem anderen PVS zugordnet werden! Hier sollte der BVS-Hersteller dann eine Auslagerung der Daten und eine Neukonfiguration vorsehen.

2. Übergabe der Patientendaten an die BVS



Die PVS ruft ein von der BVS bereitgestelltes Importmodul auf, dem als Parameter die Patientendaten-Übergabedatei übergeben wird. Das Importmodul legt den Patienten neu an, bzw. aktualisiert die Daten in der Patientendatenbank der BVS. Anschließend startet das Importmodul die BVS, wenn sie nicht schon geladen ist, und aktiviert den übergebenen Patienten.

Die Übergabedatei wird von der PVS verwaltet, hat das Format einer Windows-Ini-Datei und sollte auf der lokalen Festplatte in einem temporären Verzeichnis liegen.

2.1. Übergabedatei



Detaillierte Dateninformationen sind der Anlage 3 „Übergabe der Patientendaten an die BVS“ auf Seite 21 zu entnehmen.

Folgende Keys werden nur gefüllt, wenn eine sofortige Übergabe der Bildinformationen des Patienten gewünscht wird. Ist der Key **MAKEMMOS** auf 1 gesetzt, fordert die PVS nach Beendigung der BVS sofort Informationen über evtl. zum Patienten gespeicherte Bilder an, sie wartet also auf die Bildaufnahme.

Die BVS trägt dann vor ihrer Beendigung die Bildinformationen in die Patientendatenübergabedatei in der Form ein, wie in Abschnitt 3.1 auf Seite 12 beschrieben.

MAKEMMOS=0

direkte Bildaufnahme-/Informationsanforderung (**1=ja**)

DATE=20010501

Bildinformation aller Bilder inklusive und ab diesem Datum übermitteln (im Format **CCYYMMDD**, „SELECT“ oder „NEW“ gemäß Abschnitt 3.1 (Seite 12) bzw. wenn leer dann alle Bilder)

[PATIENT]
PVS=PVS-Sektion
BVS=BVS-Sektion
PRXNR=
DOCTOR=
PATID=
PATSHOWNR=
MODELLNR=
FROMPVS=PVS-Sektion
LASTNAME=
FIRSTNAME=
TITLE=
BIRTHDAY=
SEX=
SALUTATION=
STREET=
CITY=
ZIP=
HOMEPHONE=
EMPLOYER=
PROFESSION=
EMAIL=
CELLULAR=
INSURANCESTATUS=
INSURANCE=
INSURANCENUMBER=
POLICYNUMBER=
ERRORLEVEL=
READY=
MAKEMMOS=
DATE=
INFO=

Will die PVS der BVS Patientendaten nur bekanntgeben ohne das die BVS startet und den Patienten aktiviert (Infomodus), kann der folgende Key genutzt werden:

INFO=1

Infomodus (**1=Ja**): BVS aktualisiert nur die Daten und startet nicht

Will die PVS zusammen mit den Bildinformationen auch Vorschaubilder haben, wenn die BVS dies ebenfalls unterstützt (**SUPPORTTHUMBNAILS=1** in der BVS-Sektion), so können die folgenden Keys genutzt werden:

THUMBNAILS=1

Anforderung von Vorschaubildern zusammen mit den Bildinformationen, wenn dies die BVS unterstützt

THUMBNAILSX=320

Gewünschte maximale Breite der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)

❑ **THUMBNAILSY=200**

Gewünschte maximale Höhe der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)

Bei den Vorschaubildern muss die BVS die Proportionen des Bildes beibehalten und eine Windows-BMP-Datei mit einer Farbtiefe von 24-Bit erzeugen (vgl. Abschnitt 3.2 auf Seite 12).

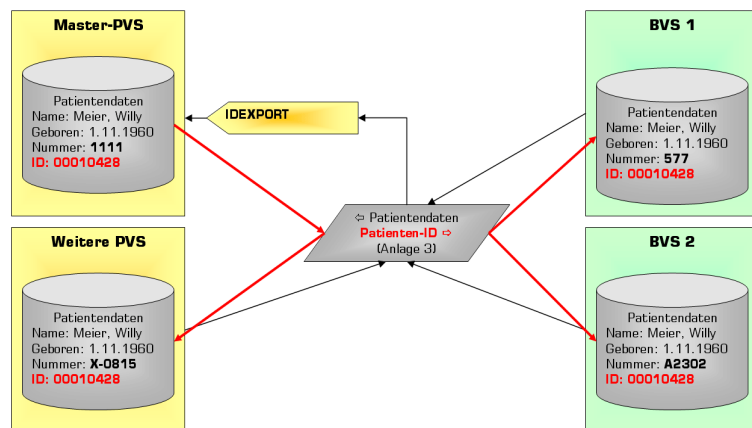
- ☑ Die PVS muss vor dem Einschreiben der Daten in die Patientendaten-Übergabedatei diese löschen, damit keine „alten“ Daten aus vorhergegangenen Patienten bestehen bleiben, falls der neue Patient nicht alle Felder belegt.
- ☑ Die Feldlängen (bis auf die Sektionsnamen und die Patientenummer) sind nicht bestimmt. D.h. die PVS schreibt so viele Zeichen in einen Key (z.B. STREET) wie ihr vorliegen und die BVS übernimmt so viele, wie sie speichern kann.
- ☑ Der Name der PVS-Sektion muss mit übergeben werden, damit die BVS in einem System mit mehreren PVS die Patienten trotz gleicher PATID voneinander unterscheiden kann. Speichert die BVS den Namen der PVS-Sektion nicht zum Patienten, kann sie nicht in einem System mit mehreren PVS eingesetzt werden. Das wird den PVS durch den Eintrag **PVSLIMIT** in der BVS-Sektion der VDDS_MMI.INI mitgeteilt.
- ☑ Beim Einsatz eines Archivs muss der Eintrag **[Patient]->BVS** immer den Sektionsnamen der gewünschten BVS enthalten, es sei denn, es wird eine Kommunikation mit dem Archiv gewünscht. Es wird allerdings immer das Importmodul des Archivs aufgerufen.
- ☑ Im Fehlerfall gibt das Importmodul der BVS die entsprechende Fehlermeldung aus und generiert einen Fehlercode für den **ERRORLEVEL**.
- ☑ Die BVS muss nach dem Einlesen der Patientendaten-Übergabedatei und der Verarbeitung der Daten das **ERRORLEVEL** setzen und den READY-Flag sowohl im System als auch in der Übergabedatei setzen. Dabei ist zu beachten, dass das Setzen des READY-Flag immer die letzte Aktion sein muss.
- ☑ Bei der Installation unter Windows trägt die BVS den kompletten Pfadnamen des Importmoduls in die VDDS_MMI.INI ein (siehe Abschnitt 1. auf Seite 6).
- ☑ Handelt es sich bei dem Importmodul um die BVS-Software selbst, sollte diese standardmäßig den Patienten übernehmen und aktivieren. Ist das nicht möglich weil diese sich z.B. gerade in einem Aufnahmezustand befindet, der nicht unterbrochen werden kann, muss diese den Aufruf durch einen **ERRORLEVEL>0** ablehnen.
- ☑ Beim Infomodus speichert die BVS einfach nur die Patientendaten ohne sich zu starten und den Patienten zu aktivieren. Auf diese Weise können Änderungen an Patientendaten (vom Anwender unbemerkt bzw. ihn nicht störend) der BVS mitgeteilt werden.

2.2. PATID-Server

In einer modernen Zahnarztpraxis läuft eine wachsende Anzahl von Softwareprodukten parallel. Die meisten davon nehmen Bezug auf die Patientendaten. Die eindeutige Identifikation eines Patienten innerhalb der gesamten Praxis ist eine notwendige Voraussetzung für ein erfolgreiches und sicheres Zusammenspiel der einzelnen Komponenten. Da in der Regel jede Software die Patienten individuell nummeriert, ist ein zentraler Abgleich notwendig. Hierzu muss eine PVS als ID-Server deklariert werden. Diese PVS muss eine zusätzliche Funktionalität zur Verfügung stellen, mit der ein für den Patienten in der gesamten Praxis eindeutiger Identifier an die anderen Kommunikationspartner übermittelt wird.

```
[PVS]
NAME1=PVS-Sektion 1
NAME2=PVS-Sektion 2
IDSERVER=PVS-Sektion 1

[PVS-Sektion 1]
...
IDEXPORT=
IDEXPORT_OS=
IDEXPORT_EVENT=
...
```



Die Nutzung der Patientennummer der PVS ist nicht in jedem Falle möglich, da KFO-PVS teilweise die Möglichkeit bieten, Patienten während der Behandlung umzumerken. Aus diesem Grund wurde auch der Eintrag **PATSHOWNR** eingeführt, um die Nummer vom Identifier unterscheiden zu können. Wenn in der PVS die Nummer eine feste Zuordnung erfährt und auch bei Löschung eines Patienten nicht wieder verwendet wird, kann diese natürlich als Identifier dienen.

Bei Verwendung eines ID-Servers muss zum Zeitpunkt der Kommunikation über den Eintrag **FROMPVS** auf den ID-Server verwiesen werden.



Der ID-Server muss ein **IDEXPORT-Modul** zur Verfügung stellen, das von beliebigen Schnittstellenteilnehmern aufgerufen werden kann. Das **IDEXPORT-Modul** erhält eine INI-Datei als Parameter (Anlage 4 auf Seite 23).

```
[PATIENT]
PVS=PVS oder BVS-Sektion
LASTNAME=
FIRSTNAME=
BIRTHDAY=
SEX=
ERRRLEVEL=
READY=

[PATIDS]
Count=1

[PATID1]
PRXNR=
DOCTOR=
PATID=
PATSHOWNR=
MODELLNR=
LASTNAME=
FIRSTNAME=
TITLE=
BIRTHDAY=
SEX=
SALUTATION=
STREET=
CITY=
ZIP=
HOMEPHONE=
EMPLOYER=
PROFESSION=
EMAIL=
CELLULAR=
INSURANCESTATUS=
INSURANCE=
INSURANCENUMBER=
POLICYNUMBER=
```

3. Übergabe der Bildinformationen an die PVS



Die PVS ruft ein von der BVS bereitgestelltes Exportmodul auf, dem als Parameter der Name der Bildübergabedatei übergeben wird. Diese Datei enthält beim Aufruf die PATID und ein Datum bzw. den gewünschten Modus. Das Exportmodul speichert jetzt alle Bildinformationen der Bilder, welche nach dem übergebenen Datum bzw. aufgrund des gewählten Modus hinzugekommen oder verändert worden sind, in dieser Datei. Für das Datum und den Modus gelten folgende Regeln:

1. Ein Datum muss im Format CCYYMMDD übergeben werden. Die BVS soll dann Informationen zu allen Bildern liefern, die inklusive und nach dem angegebenen Datum aufgenommen oder verändert worden sind. Diese Funktion ist Pflicht.
2. Ist das Datum leer, werden die Bildinformationen aller Bilder übermittelt. Diese Funktion ist Pflicht.
3. Lautet der Eintrag beim Datum „NEW“, so soll die BVS dem Anwender ermöglichen, direkt neue Bilder aufnehmen zu können. Die BVS soll dann nur Informationen über diese neuen Bilder liefern, oder keine, falls der Anwender den Vorgang abgebrochen hat. Diese Funktion ist optional.
4. Lautet der Eintrag beim Datum „SELECT“, so soll die BVS dem Anwender gestatten, aus den bereits vorhandenen Bildern des Patienten eine unbestimmte Anzahl auszuwählen. Es sind dann nur die Informationen der ausgewählten Bilder an die PVS zu übermitteln. Zusätzlich kann dem **SELECT**-Befehl nach einem Slash noch ein Text folgen, der als Titel zur Auswahl benutzt werden kann (z.B. „Gutachterauswahl“). Die **SELECT**-Funktion ist optional, die Titelangabe/-auswertung optional zu **SELECT**.

Die PVS hat drei Möglichkeiten die Beendigung des BVS-Exportmoduls zu erkennen:

1. Die PVS pollt auf das **READY**-Flag bis dieses auf **1** gesetzt ist.
2. Die PVS setzt das **PVSIMP**-Flag auf **1** und setzt seine Ausführung fort. Die BVS wird dann automatisch das PVS-Importmodul für die Bildinformation aufrufen. Zur Sicherheit, das die BVS die Befehle auch verstanden hat, sollte es aber dennoch auf das **READY**-Flag warten und den **ERRORLEVEL** auswerten. Insbesondere im Hinblick auf die von der BVS optional zu unterstützenden SELECT- und NEW-Befehle.
3. Wenn das Modul ein EVENT in der VDDS_MMI.INI definiert hat, kann die PVS nach dem Aufruf des Moduls auf dieses warten (die PVS erhält dazu mit OpenEvent ein Handle vom OS und wartet dann per WaitForSingleObject. Dieser Aufruf liefert *non-signaled* so lange das Modul läuft und *signaled* wenn es fertig ist. Dann ist nur noch ein CloseHandle nötig).



Übersichtsdiagramme der Synchronisation sind in der Anlage 9 auf Seite 32 enthalten.

Nach Beendigung des Exportmoduls liest die PVS bzw. das PVS-Importmodul die Bildübergabedatei aus und übernimmt die Daten in die eigene Bilddatenbank.

3.1. Übergabedatei der PVS an die BVS



Die Bildinformations-Übergabedatei hat den Aufbau einer Windows-Ini-Datei und sollte auf der lokalen Festplatte in einem temporären Verzeichnis liegen. Sie wird von der PVS erzeugt und enthält beim Aufruf der BVS die von der PVS eingeschriebenen Daten.

Detaillierte Dateninformationen sind der Tabelle in Anlage 5 auf Seite 25 zu entnehmen.

```
[PATID]
PATID=
PATSHOWNR=
MODELLNR=
FROMPVS=PVS-Sektion
PRXNR=
DATE=
PVSIMP=
PVS=PVS-Sektion
BVS=BVS-Sektion
THUMBNAILS=
THUMBNAILSX=
THUMBNAILSY=
READY=
ERRORLEVEL=
```

3.2. Übergabedatei der BVS an die PVS



Die Bildinformationen werden in derselben Datei zurückgeliefert, in der auch die Anfragedaten stehen (Abschnitt 3.1). Diese werden nicht gelöscht, sondern die BVS hängt ihre Bildinformationen an die Datei an. Über den Eintrag **[MMOS]->COUNT** erhält die PVS die Anzahl der zurückgelieferten Bilder. Die Bildinformationen stehen in Sektionen mit konstanten Namen (**MMO1, MMO2, MMO3, ...**).

Der Aufbau der Bildübergabedatei nach dem Aufruf des BVS-Export-Moduls mit den Daten von zwei Bildern ist der Tabelle in Anlage 6 auf Seite 26 ersichtlich.

Bei der Installation trägt die BVS den kompletten Pfadnamen des Exportmoduls für die Bildinformation in die VDDS_MMI.INI ein (Abschnitt 1. auf Seite 6).

Jedem zurückgelieferten Bild wird von der BVS eine Bild-ID (max. 200 alphanumerische Zeichen) zugeordnet, unter der später eine Kopie des Bildes angefordert werden kann. Hat die BVS keine Bild-ID, kann auch der Dateiname und Pfad des Bildes als alphanumerische BILD-ID verwendet werden, sofern dieser eindeutig und nicht länger als 200 Zeichen ist. Dieser wird von der PVS nicht als Pfad ausgewertet, sondern nur als alphanumerische Zeichenkette gespeichert, mit der später eine Bildkopie angefordert werden kann.

```
[PATID]
PATID=
PATSHOWNR=
MODELLNR=
FROMPVS=PVS-Sektion
PRXNR=
DATE=
PVSIMP=
READY=
ERRORLEVEL=
PVS=PVS-Sektion
BVS=BVS-Sektion

[MMOS]
COUNT=2

[MMO1]
MMOID=
PRXNR=
TYPE=
TYPENR=
TEETH=
PREGNANCY=
XRAYCOMMENT=
XRAYMS=
XRAYCURRENT=
XRAYVOLTAGE=
EXT=
COLORTYPE=
DATE=
TIME=
COMMENT=
THUMBNAIL=

[MMO2]
...
```

Die PVS muss vor dem Einschreiben der Daten in die Bildübergabedatei diese löschen, falls vorhanden. Das muss geschehen, damit keine „alten“ Daten aus vorhergegangenen Patienten bestehen bleiben.

Unter EXT wird die Extension des Objekts eingetragen. Bisher sind 4 Objekttypen zur Übergabe vorgesehen (TIF-Bilder, AVI=Videos, WAV=Tondateien, DOC=Bild/Text-Dokumente). Die PVS übernimmt nur die Objekte, welche sie selber verwalten möchte.

- ☑ Will die BVS der PVS mitteilen, dass ein Objekt gelöscht wurde, muss sie den Key **TYPENR** auf **0** für „DELETE“ setzen. Die restlichen Felder, bis auf **MMOID** können dann leer bleiben. Diese Funktionalität ist nur im Zusammenhang mit dem PVS-Importmodul möglich, welches beim Löschen eines Bildes aufgerufen wird. Daher sollten alle PVS ein solches Importmodul zur Verfügung stellen bzw. die Stufe 6 realisieren (siehe Abschnitt “Stufenweise Realisierung aus Sicht der PVS” auf Seite 4).
- ☑ Bei der Erzeugung der Vorschaubilder (Thumbnail-Dateien) muss die BVS die Proportionen des Bildes beibehalten und diese als Windows-BMP-Datei mit einer Farbtiefe von 24-Bit bereitstellen. Bei nicht direkt als Einzelbild darstellbaren Dokumentformaten (AVI, WAV, DOC) muss eine entsprechende Standardbitmap bereitgestellt werden. Für das Löschen der bereitgestellten Thumbnail-Dateien ist die PVS verantwortlich.

3.3. PVS-Importmodul für Bildinformationen



Die PVS stellt zusätzlich ein Importmodul zur Verfügung, welches die Bildinformations-Übergabedatei (Anlage 6 auf Seite 26) übernehmen kann. Die BVS ruft die Importmodule der vom Anwender festgelegten PVS (Abschnitt “Stufenweise Realisierung aus Sicht der PVS” auf Seite 4) auf, wenn sich Bildinformationen geändert haben bzw. ein neues Bild hinzugekommen ist. Dieses dient dazu, die Bildinformation in der PVS immer aktuell zu halten, bzw. von gelöschten Objekten Kenntnis zu erhalten.

Hat eine PVS die Informationen bereits durch den synchronen Aufruf (**MAKEMMOS=1** aus Abschnitt 2.1. Seite 8 bzw. **PVSIMP=0** Anlage 6, Seite 26) erhalten, so braucht die BVS diese PVS nicht erneut zu „benachrichtigen“.

Das PVS-Import-Modul wird mit der Bildinformations-Übergabedatei als Parameter aufgerufen und setzt dann das **ERRORLEVEL** und den **READY**-Flag. Im Fehlerfall gibt das Importmodul die entsprechende Fehlermeldung und bietet ggf. eine Wiederholung an, um insbesondere beim Einsatz des optionalen Bildarchivs keine Bilder zu verlieren. Wünscht die PVS auch Vorschaubilder (**WANTTHUMBNAILS=1** in der PVS-Sektion) so muss die BVS diese mit erzeugen, wenn die BVS dies ebenfalls kann (**SUPPORTTHUMBNAILS=1** in der BVS-Sektion). Die PVS ist für das Löschen der Bilder verantwortlich.

Bei der Installation trägt die PVS den kompletten Pfadnamen des Importmoduls für die Bildinformation in die VDDS_MMI.INI ein (Abschnitt 1. Seite 6).



Der Aufruf aller vorhandenen PVS-Importmodule seitens der BVS ist eine Pflichtfunktion der BVS. Sollte ein Archiv installiert sein, so ruft die BVS nur das Importmodul des Archivs auf und das Archiv versorgt dann alle vorhandenen PVS mit den neuen Bildinformationen (vgl. Abschnitt 5. ab Seite 15).

4. Übergabe/ Darstellung der Bilder

Die Übergabe/Bilddarstellung erfolgt jeweils mittels Viewer der PVS oder der BVS.

4.1. Bilddarstellung mit dem Viewer der PVS



Die PVS ruft ein von der BVS bereitgestelltes Exportmodul auf (Anlage 2 „VDDS-Konfigurationsdatei (VDDS_MMI.INI) - BVS-Einträge“, Seite 19), dem als Parameter die Bildrückübergabedatei übergeben wird. Die PVS schreibt beim Aufruf die BILD-ID's aller gewünschten Bilder in die Datei.

Detaillierte Dateninformationen sind der Tabelle in Anlage 7 auf Seite 29 zu entnehmen.

Die PVS hat zwei Möglichkeiten die Beendigung des BVS-Exportmoduls zu erkennen:

1. Die PVS pollt auf das READY-Flag bis dieses auf **1** gesetzt ist.
2. Wenn das Modul ein EVENT in der VDDS_MMI.INI definiert hat, kann die PVS nach dem Aufruf des Moduls auf dieses warten (die PVS erhält dazu mit OpenEvent ein Handle vom OS und wartet dann per WaitForSingleObject.

```
[MMOIDS]
PVS=PVS-Sektion
COUNT=2
MMOID1=
MMOID2=
COLORDEPTH=
GRAYSCALE=
THUMBNAILS=
THUMBNAILSX=
THUMBNAILSX=
READY=
ERRORLEVEL=

[MMOPATH]
MMOID1=
MMOID2=
...
```

Dieser Aufruf liefert `non-signalEd` so lange das Modul läuft und `signalEd` wenn es fertig ist. Dann ist nur noch ein CloseHandle nötig).



Übersichtsdiagramme der Synchronisation sind in der Anlage 9 auf Seite 32 enthalten.

Im Fehlerfall gibt das Exportmodul die entsprechende Fehlermeldung aus und setzt den **ERRORLEVEL** auf den Fehlerwert sowie den Eintrag **[MMOIDS]->COUNT** auf **0** damit die PVS über den Fehler informiert wird.

Die Kopien der Bilder müssen von der BVS im TIFF-Format höchster Qualität, mit einer Farbtiefe nicht höher als im **COLORDEPTH/GRAYSCALE**-Eintrag angegeben, abgelegt werden und eventuelle Veränderungen z.B. an Helligkeit, Kontrast, Spiegelung usw. bereits enthalten. Die PVS zeigt dieses Bild jetzt mit ihrem eigenen Viewer an. Die Bildkopien werden von der PVS nicht gespeichert, sondern lediglich zur Anzeige verwendet und anschließend gelöscht.

Sollte das Archiv die Bilder anfordern (**PVS-Sektion=PVS_ARCHIV**), so kann das BVS auch unveränderte Originale liefern. Das Archiv darf Bilder natürlich nicht löschen.

Bei der Erzeugung der Vorschaubilder (Thumbnail-Dateien) muss die BVS die Proportionen des Bildes beibehalten und diese als Windows-BMP-Datei mit einer Farbtiefe von 24-Bit bereitstellen. Bei nicht direkt als Einzelbild darstellbaren Dokumentformaten (AVI, WAV, DOC) **muss** eine entsprechende Standardbitmap bereitgestellt werden. Für das Löschen der bereitgestellten Thumbnail-Dateien ist die PVS verantwortlich.

Bei der Installation trägt die BVS den kompletten Pfadnamen des Exportmoduls für die Bilder in die VDDS_MMI.INI ein (Abschnitt 1. Seite 6).

4.2. Bilddarstellung mit dem Viewer der BVS



Die PVS ruft ein von der BVS bereitgestellten Viewer auf, dem als Parameter die Bild-ID übergeben wird. Dieser übernimmt dann als eigenständiges Programm die Anzeige. Bei dem Viewer kann es sich auch um die BVS selbst handeln, die den zum Bild gehörenden Patienten aktiviert und das Bild darstellt. Ob und wie der Viewer/die BVS dem Anwender dann noch zusätzlich die Auswahl weiterer Bilder bzw. Funktionen gestattet, bleibt diesem überlassen. Im Fehlerfall gibt der Viewer die entsprechende Fehlermeldung aus. Eine Rückmeldung des Fehlers an die PVS ist nicht vorgesehen, kann aber über den **ERRORLEVEL** des Viewermoduls vorgenommen werden. Insbesondere kann das Viewermodul auch beendet werden, sobald der Patient aktiviert worden ist. Bei der Installation trägt die BVS den kompletten Pfadnamen des Viewers in die VDDS_MMI.INI ein (Abschnitt 1. Seite 6).

5. Optionale Integration eines Bildarchivs



Das Bildarchiv ist ein Programm dessen Aufgabe die zentrale Verwaltung der Patientendaten, Originalbilder und Dokumente aller an der VDDS-media-Schnittstelle teilnehmenden PVS/BVS ist.

Das Archiv übernimmt die Sicherung und Bereitstellung der Bilder. Es bietet dem Anwender die Möglichkeit, sich alle zu einem Patienten gehörenden Bilder und Dokumente zusammenzustellen und anzusehen, obwohl diese aus verschiedenen BVS stammen. Ist ein Bildarchiv installiert, kommunizieren die PVS nicht mehr direkt mit den BVS bzw. umgekehrt, sondern alle Teilnehmer tauschen Daten ausschließlich mit dem Bildarchiv aus.

5.1. Anmeldung des Archivs/ Kommunikationsmodell



Das Archiv meldet sich bei der Installation in der VDDS_MMI.INI wie eine PVS **und** wie eine BVS an. Es muss alle Stufen der PVS/BVS realisiert haben, wobei bezüglich der PVS-Stufen 2 und 4 eine entsprechende BVS die Originalbilder liefern sollte, um doppeltes Speichern zu vermeiden. Das Archiv kann dann als ID-Server dieser BVS verstanden werden, da ein PVS die **MMOIDS** des Archivs erhält, sie aber unter dem Namen der BVS speichert.

Die Erstellung und Auswertung der Übergabedateien erfolgt wie zuvor definiert, nur abschließend muss die BVS/PVS das entsprechende Modul des Archivs anstelle das der PVS/BVS aufrufen. So erfolgt ein Aufruf einer BVS durch ein PVS oder umgekehrt immer über das Archiv. Das Archiv leitet die Daten gegebenenfalls weiter.

Für die PVS/BVS ist damit, bis auf die Nutzung anderer Module, die Nutzung des Archivs auf Wunsch völlig transparent. Die PVS/BVS können aber auch das Archiv über die **ARCHIV-Sektion** direkt ansprechen.

```
[BVS]
NAME1=BVS-Sektion 1
NAME2=BVS-Sektion 2
ARCHIV=ARCHIV

[ARCHIV]
NAME=
...
```

5.2. Übergabe der Patientendaten an das Archiv



Die PVS erzeugt die Übergabedatei, wie in Abschnitt 2.1 Seite 8 beschrieben und ruft das Importmodul des Archivs auf. Das Archiv speichert die Patientendaten samt des Sektionsnamens der aufrufenden PVS in seiner eigenen Datenbank. Anhand des **[PATIENT]->BVS**-Eintrags weiß das Archiv welche Bilder welcher BVS gewünscht sind, welches BVS-Importmodul danach aufgerufen werden muss oder von welcher BVS eine neue Bildaufnahme gewünscht ist.

Da das Archiv auch als BVS fungiert, kann es auch selbst direkt angesprochen werden. Setzt eine PVS z.B. bei der einfachen Übergabe der Patientendaten den Wert **[PATIENT]->BVS=ARCHIV-Sektion**, wäre dies ein Recherche-Aufruf nach allen zum Patienten gespeicherten Dokumenten.

5.3. Übergabe der Bildinformationen an das Archiv und die PVS



Die PVS erzeugt, wie in Abschnitt 3.1 auf Seite 12 beschrieben, eine Übergabedatei, ruft dann aber das Exportmodul des Archivs auf und erhält dann vom Archiv, wie im Abschnitt 3.2 auf Seite 12 beschrieben, die Bildinformationen der Bilder des Patienten der übergebenen BVS zurück. Damit die oben beschriebene Transparenz gewahrt bleibt, darf das Archiv nicht die Informationen über alle Bilder aller BVS zurückgeliefert, sondern nur die der jeweils unter **[PATID]->BVS** eingetragenen BVS. Möchte die PVS alle Bilder aller BVS zu einem Patienten haben, muss sie entweder:

- Übergabedateien für alle BVS erstellen, wobei es tatsächlich immer nur das Exportmodul des Archivs aufruft. Zur Sicherheit sollte das Archiv zusätzlich noch die PVS-Anfrage an die BVS weiterleiten um sicherzugehen dass sie auch wirklich alle Bildinformationen vorliegen hat.
- oder
- Das Archiv über die **ARCHIV-Sektion** direkt ansprechen.

Das Archiv sollte weiterhin prüfen, ob es schon die Originalbilder aller zurückgelieferten Bildinformationen vorliegen hat.

5.4. Archiv-Importmodul für Bildinformationen



Das Archiv muss dafür sorgen, dass es immer die aktuellen Bildinformationen zur Verfügung hat. Dafür muss das Archiv für alle BVS das im Abschnitt 3.3 auf Seite 13 beschriebene Importmodul zur Verfügung stellen. Das Archiv sollte weiterhin prüfen, ob es schon die Originalbilder aller zurückgelieferten Bildinformationen vorliegen hat.

Wird dieses Modul von einer BVS aufgerufen, muss das Archiv sofort das dazugehörige Bild anfordern und archivieren. Das Archiv erhält möglichst die Originaldateien der BVS. Wenn es sich um ein neues Bild handelt, fordert das Archiv dieses zur Archivierung sofort an, wie in Abschnitt 4.1 auf Seite 14 beschrieben.

5.5. Übergabe und Darstellung der Bilder mit dem Viewer der PVS

Die PVS ruft wie im Abschnitt 4.1 Seite 14 beschrieben, das Exportmodul des Archivs auf. Das Archiv stellt dann die gewünschten Bilder zur Verfügung, ggf. unter Weiterleitung der Anfrage an die BVS.

5.6. Darstellung der Bilder mit dem Viewer der BVS oder des Archivs

Die PVS ruft einen vom Archiv bereitgestellten Viewer auf, dem als Parameter die Bild-ID übergeben wird. Hat das Archiv keinen eigenen Viewer, ruft es den Viewer der BVS auf. Wenn das Archiv über einen eigenen Viewer verfügt, bringt es das Bild selbst zur Ansicht.

6. Häufig gestellte Fragen/FAQ

- BVS ist kein BVS**

Mittlerweile existieren Programme am Markt, die VDDS-media nur zur Übernahme von Patientendaten implementieren. Dazu geben sich diese Programme als BVS aus. Das Resultat kann je nach Konfiguration und Teilnehmern eine „gestörte“ Kommunikation sein, in der Fehlermeldungen ob der vermissten Bilddaten auftreten. Solche Programme sollten die Schnittstelle mindestens in der Version 1.2 implementieren und dann den Eintrag „**STAGES=1**“ setzen, damit „echte“ VDDS-media Programme den Workflow für diese „illegalen“ BVS kundenfreundlich steuern können.
- Darf eine PVS ihr Patientendaten-Exportmodul überhaupt in der VDDS_MMI.INI eintragen, wenn ein anderer ID-Server bereits vorhanden ist?**

Natürlich kann eine PVS sein Patientendaten-Exportmodul in der VDDS_MMI.INI eintragen, auch wenn eine andere PVS bereits ID-Server ist. Es darf natürlich nicht den Eintrag **IDSERVER** ungefragt verändern!

❑ **Warum lässt die Schnittstellenbeschreibung Netzwerkfunktionen komplett außer acht?**

Die Schnittstelle beschreibt nur lokale Kommunikation. Es steht aber jedem Teilnehmer frei, über die Module Netzwerkfunktionen abzubilden. So könnte ein lokal gestartetes Patientendaten-Importmodul einer BVS selbstverständlich auch die Datenübernahme bzw. Erfassung von Bilddaten auf einem „entfernten“ Rechner aktivieren.

❑ **Warum steht der Eintrag THUMBNAILS=1 in der VDDS_MMI.INI und als THUMBNAILS=1 nochmals in den Übergabedateien?**

Beim Aufruf eines Patientenimportmoduls eines BVS muss die PVS explizit angeben können, ob sie bei diesem konkreten Aufruf Thumbnails möchte oder nicht (Abschnitt 2.1 auf Seite 8 und 3.1 auf Seite 12). Der Eintrag in der VDDS_MMI.INI dient eigentlich nur der Information der BVS, ob das Importmodul der PVS auch Thumbnails möchte (Abschnitt 3.3 auf Seite 13). Das kann auch nur in der VDDS_MMI.INI stehen, da die BVS unter Umständen zuvor gar keinen Kontakt mit der PVS hat. Wenn also eine PVS per eigenen Importmodul keine Thumbnails haben möchte, so kann sie den Eintrag in der VDDS_MMI.INI weglassen und dennoch analog 2.1 (Seite 8) und 3.1 (Seite 12) bei direkten BVS-Aufrufen Thumbnails anfordern.

❑ **Wie können die schwarzen Fenster beim Aufruf einer DOS-Applikation aus einer Windows Applikation unterdrückt werden?**

```
PROCESS_INFORMATION PrInf;
STARTUPINFO         StartInf;

ZeroMemory(&StartInf, sizeof(STARTUPINFO));
StartInf.cb=sizeof(STARTUPINFO);
if (IsDos) {
    StartInf.dwFlags=STARTF_USESHOWWINDOW;
    StartInf.wShowWindow=SW_HIDE;
}
if (CreateProcess(NULL, lStr, NULL, NULL, FALSE,
    IsDos?DETACHED_PROCESS:CREATE_NEW_CONSOLE, NULL, NULL, &StartInf,
    &PrInf) {

...

if (PrInf.hProcess) {
    CloseHandle(PrInf.hProcess);
    CloseHandle(PrInf.hThread);
}
```

❑ **Der Pfad der eigenen Software enthält Leerzeichen. Muss für die VDDS-Kopplung ein eigenes Verzeichnis in 8.3 Notation angelegt werden?**

Nein. Lediglich ist es notwendig, alle Registrierungen in 8.3 Notation vorzunehmen. Dazu ist folgender Windows API-Ruf zu verwenden:

```
DWORD GetShortPathName(LPCTSTR lpszLongPath, LPCTSTR lpszShortPath,
    DWORD cchBuffer);
```

Damit kann eine Pfadangabe mit langen Dateinamen in 8.3 Notation umgewandelt werden.

Impressum

Inhaltliche Bearbeitung:
Jürgen Weiland, Michael Brand

Redaktionelle Bearbeitung:
Bernd Künzel

Version: 1.3
Ausgabedatum: 31.08.2005
Letzte Überarbeitung: 23.01.2006

Kontakt:
jpw@compugroup.com
m.brand@computer-konkret.de

Die Datei hat den Aufbau einer (Windows-)Ini-Datei und sieht folgendermaßen aus:

[PVS]	
NAME1=PVS-Sektion 1	Sektionsname der ersten PVS (maximal 30 Zeichen)
NAME2=PVS-Sektion 2	Sektionsname der zweiten PVS (maximal 30 Zeichen)
ARCHIV=ARCHIV-Sektion	Sektionsname des PVS-Bildarchivs (immer= PVS_ARCHIV)
IDSERVER=PVS-Sektion	Sektionsname der PVS, die Patientendaten zentral zur Verfügung stellt (vgl. Abschnitt 2.2. „PATID-Server“ Seite 10)
[PVS-Sektion 1]	
NAME=PVS-Name	Name der bei Bedarf in der BVS angezeigt werden kann
MMOINFIMPORT=Pfad	kompletter Pfad des PVS-Importmoduls für Bildinformationen (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!) vgl. Abschnitt 3.3. „PVS-Importmodul für Bildinformationen“ auf Seite 13.
MMOINFIMPORT_OS=1	Betriebssystem des Moduls (1=Windows, 2=DOS, 3=Unix)
THUMBNAILS=1	Name eines Events unter Windows, das bei Modulende gefeuert wird. Kann vom Aufrufer statt Polling verwendet werden, um das Modulende zu erkennen. Keine Pflichtfunktion für das Modul! Vgl. Abschnitt 3.3. „PVS-Importmodul für Bildinformationen“ auf Seite 13.
MMOINFIMPORT_EVENT=Name	Flag für das PVS-Importmodul das angibt, ob die PVS beim Aufruf seines Importmoduls für Bildinformationen zusammen mit den Bildinformationen auch Vorschaubilder (Thumbnails) wünscht - vgl. Abschnitt 3.3. „PVS-Importmodul für Bildinformationen“ auf Seite 13.
THUMBNAILSX=320	Gewünschte maximale Breite der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
THUMBNAILSY=320	Gewünschte maximale Höhe der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
IDEXPORT=Pfad	kompletter Pfad des PVS-Exportmoduls für Patientendatenanfragen (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!)
IDEXPORT_OS=1	Betriebssystem des Moduls (1=Windows, 2=DOS, 3=Unix)
IDEXPORT_EVENT=Name	Name eines Events unter Windows, das bei Modulende gefeuert wird. Kann vom Aufrufer statt Polling verwendet werden, um das Modulende zu erkennen. Keine Pflichtfunktion für das Modul!
VERSION=1.3	Von der PVS unterstützte Schnittstellen-Version
STAGES=12567	Angabe der von der PVS unterstützten Stufen (HEX-Ziffern, aufsteigend). Vgl. Abschnitt “Stufenweise Realisierung aus Sicht der PVS” auf Seite 4.

PVS-Sektion 2 und ARCHIV-Sektion analog zu PVS-Sektion 1

Die Datei hat den Aufbau einer (Windows-)Ini-Datei und sieht folgendermaßen aus:

[BVS]	
NAME1=BVS-Sektion 1	Sektionsname der ersten BVS (maximal 30 Zeichen)
NAME2=BVS-Sektion 2	Sektionsname der zweiten BVS (maximal 30 Zeichen)
ARCHIV=ARCHIV-Sektion	Sektionsname des BVS-Bildarchivs (Immer=BVS_ARCHIV)
[BVS-Sektion 1]	
NAME=BVS-Name	Name der in der PVS auf einer Schaltfläche oder in einer Liste für das BVS angezeigt werden soll
PATDATIMPORT=Pfad	kompletter Pfad des Importmoduls für Patientendaten (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!)
PATDATIMPORT_OS=1	Betriebssystem des Moduls
PATDATIMPORT_EVENT=Name	Name eines Events unter Windows, dass bei Modulende gefeuert wird. Kann vom Aufrufer statt Polling verwendet werden, um das Modulende zu erkennen. Keine Pflichtfunktion für das Modul!
MMOINFEXPORT=Pfad	kompletter Pfad des Exportmoduls für Bildinformationen (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!)
MMOINFEXPORT_OS=1	Betriebssystem des Moduls
MMOINFEXPORT_EVENT=Name	Mögliches Windows-Event für das Modulende. Keine Pflichtfunktion für das Modul!
MMOEXPORT=Pfad	kompletter Pfad des Exportmoduls für Bildkopien (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!)
MMOEXPORT_OS=1	Betriebssystem des Moduls
MMOEXPORT_EVENT=Name	Mögliches Windows-Event für das Modulende. Keine Pflichtfunktion für das Modul!
MMOVIEW=Pfad	kompletter Pfad zum Aufruf des Viewers (bei Windows/DOS kurze 8.3-Pfade, nur Pfad mit Modul, keine Parameter!)
MMOVIEW_OS=1	Betriebssystem des Moduls
MMOVIEW_EVENT=Name	Mögliches Windows-Event für das Modulende. Keine Pflichtfunktion für das Modul!
PVSLIMIT=PVS-Sektion	wird von der BVS ausgefüllt, wenn diese bezüglich der Speicherung der Patientennummern nur eine PVS-unterstützt (alle PVSe die hier nicht angegeben sind dürfen diese BVS dann nicht mehr ansprechen, bzw. zur Auswahl anbieten)
SUPPORTSELECT=1	Flag das angibt, ob die BVS die individuelle Bildauswahl mit SELECT unterstützt (1=ja, 0=nein , vgl. Abschnitt 3.1. „Übergabedatei der PVS an die BVS“ Seite 12)

SUPPORTNEW=1	Flag das angibt, ob die BVS den Aufruf zum Erfassen neuer Bilder mit NEW unterstützt (1=ja, 0=nein , vgl. Abschnitt 3.1. „Übergabedatei der PVS an die BVS“ Seite 12)
SUPPORTINFO=1	Flag das angibt, ob der Infomodus bzgl. Patientendaten unterstützt wird (vgl. Abschnitt 2. „Übergabe der Patientendaten an die BVS“ Seite 8)
SUPPORTDEPTH=1	Flag das angibt, ob das MMOEXPORT-Modul wählbare Farbtiefen unterstützt (vgl. Abschnitt 4.1. “Bilddarstellung mit dem Viewer der PVS” auf Seite 14)
SUPPORTTHUMBNAILS=1	Flag das angibt, ob die BVS auch den Export von Thumbnails unterstützt (vgl. Abschnitt 3.1., 3.2 Seite 12 und 4.1. Seite 14). Ist dieses Flag gesetzt, muss die Funktionalität aller angegebenen Abschnitte umgesetzt sein.
STAGES=1234	Angabe der von der BVS unterstützten Stufen (HEX-Ziffern, aufsteigend). Vgl. Abschnitt “Stufenweise Realisierung aus Sicht der BVS” auf Seite 4.
VERSION=1.3	Von der BVS unterstützte Schnittstellen-Version

BVS-Sektion 2 und ARCHIV-Sektion analog zu BVS-Sektion 1. Die **ARCHIV-Sektion** enthält damit alle Einträge einer PVS- und einer BVS-Sektion.

Die Übergabedatei wird von der PVS verwaltet, hat das Format einer Windows-Ini-Datei und sollte auf der lokalen Festplatte in einem temporären Verzeichnis liegen.

[PATIENT]	
PVS=PVS-Sektion	Sektionsname der PVS muss übergeben werden (siehe Tabelle in der Anlage 1 auf Seite 18)
BVS=BVS-Sektion	Sektionsname der BVS (benötigt für das Archiv und die direkte Rückübergabe von Bildern, d.h. die PVS muss zu jedem übernommenen Bildinformationssatz den Namen der BVS mit abspeichern, um das Bild später anfordern zu können)
PRXNR=1	Praxisnummer bei Mehrfachpraxen (Standard 1)
DOCTOR=Dr. Schmerzfrei	Name des zur Praxisnummer gehörenden Hauptbehandlers / Arztes (zur optionalen Verwendung)
PATID=23	alphanumerische Patientenummer, maximal 12 Zeichen (Pflichtfeld: muss übergeben werden)
PATSHOWNR=2004-133	Unterstützung der im KFO-Bereich üblichen Umnummerierung von Patienten je nach Bearbeitungsstatus (zur optionalen Verwendung)
MODELLNR=23-101	Unterstützung der im KFO-Bereich üblichen Modellnummer (zur optionalen Verwendung)
FROMPVS=PVS-Sektion	Sektionsname der PVS die den Patienten verwaltet (vgl. Abschnitt 2.2 "PATID-Server" auf Seite 10 und Anlage 4 auf Seite 23)
LASTNAME=Beispiel	Nachname (Pflichtfeld: muss übergeben werden), Namenszusätze werden ebenfalls in diesem Feld mit übergeben
FIRSTNAME=Peter	Vorname (Pflichtfeld: muss übergeben werden)
TITLE=Prof. Dr.	Titel
BIRTHDAY=19800520	Geburtsdatum im Format CCYYMMDD (Kein Pflichtfeld mehr!)
SEX=M	Man, Woman (Pflichtfeld!)
SALUTATION=2	Mr.=1, Ms.=2, Miss.=3
STREET=Vogelsang 1	Strasse
CITY=Damp	Ort
ZIP=24351	Postleitzahl
HOMEPHONE=040/9171	Telefon privat
EMPLOYER=Praxissoft	Arbeitgeber
PROFESSION=Koch	Beruf
EMAIL=P.Beispiel@praxissoft.de	E-Mail-Adresse des Patienten
CELLULAR=0171/12345678	Mobiltelefon
INSURANCESTATUS=M	M=Mitglied, F=Familienmitglied, R=Rentner, P=Privat
INSURANCE=AOK	Name der Krankenkasse

INSURANCENUMBER =2234013	Nummer der Krankenkasse
POLICYNUMBER=32232	Mitgliedsnummer des Patienten bei der Krankenkasse
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK, ≥ 1=Fehler)
READY=0	die BVS setzt dieses Flag auf 1 , wenn die Übergabedatei verarbeitet wurde bzw. alle Bildinformationen geschrieben wurden (wenn MAKEMMOS auf 1 , s.u.)

Das IEXPORT-Modul beinhaltet folgende Parameter in der INI-Datei.

[Patient]	
PVS=PVS-Sektion oder	Sektionsname der PVS die Auskunft über Patienten haben möchte
BVS=BVS-Sektion	Sektionsname der BVS die Auskunft über Patienten haben möchte
LASTNAME=Meyer	Bekannter Name des oder der gefragten Patienten
FIRSTNAME=Willy	Bekannter Vorname des oder der gefragten Patienten
BIRTHDAY=19840123	Bekanntes Geburtsdatum des oder der gefragten Patienten (optional)
SEX=M	Bekanntes Geschlecht des oder der gefragten Patienten (optional)
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK , \geq 1=Fehler)
READY=0	Flag das angibt, ob die Datei bearbeitet wurde (0 bei Aufruf, 1 nach Beendigung)

Die PVS bzw. der ID-Server ergänzt dann die Patientendaten bzw. liefert alle Treffer zurück:

[Patient]	
PVS/BVS=PVS/BVS-Sektion	
LASTNAME=Meyer	
FIRSTNAME=Willy	
BIRTHDAY=19840123	
SEX=M	
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK , \geq 1=Fehler)
READY=0	die PVS setzt dieses Flag auf 1 , wenn die Übergabedatei verarbeitet wurde bzw. alle Patienteninformationen geschrieben wurden

[PATIDS]	
Count=1	Anzahl der übermittelten Patientendaten: „- 1 “ wenn zu viele Datensätze ermittelt wurden (64k Grenze bei INI-Dateien, Puffer bei DOS-Systemen)

[PATID1]	Patientendatensatz 1
PRXNR=1	Praxisnummer bei Mehrfachpraxen (Standard 1)
DOCTOR=Dr. Schmerzfrei	Name des zur Praxisnummer gehörenden Hauptbehandlers / Arztes (zur optionalen Verwendung)
PATID=23	alphanumerische Patientennummer, maximal 12 Zeichen (Pflichtfeld: muss übergeben werden)
PATSHOWNR=2004-133	Unterstützung der im KFO-Bereich üblichen Umnummerierung von Patienten je nach Bearbeitungsstatus (zur optionalen Verwendung)

MODELLNR=23-101	Unterstützung der im KFO-Bereich üblichen Modellnummer (zur optionalen Verwendung)
LASTNAME=Beispiel	Nachname (Pflichtfeld: muss übergeben werden), Namenszusätze werden ebenfalls in diesem Feld mit übergeben
FIRSTNAME=Peter	Vorname (Pflichtfeld: muss übergeben werden)
TITLE=Prof. Dr.	Titel
BIRTHDAY=19800520	Geburtsdatum im Format CCYYMMDD
SEX=M	Man, Woman (Pflichtfeld!)
SALUTATION=2	Mr.=1, Ms.=2, Miss.=3
STREET=Vogelsang 1	Strasse
CITY=Damp	Ort
ZIP=24351	Postleitzahl
HOMEPHONE=040/9171	Telefon privat
EMPLOYER=Praxissoft	Arbeitgeber
PROFESSION=Koch	Beruf
EMAIL=P.Beispiel@praxissoft.de	E-Mail-Adresse des Patienten
CELLULAR=0171/12345678	Mobiltelefon
INSURANCESTATUS=M	M=Mitglied, F=Familienmitglied, R=Rentner, P=Privat
INSURANCE=AOK	Name der Krankenkasse
INSURANCE-NUMBER=2234013	Nummer der Krankenkasse
POLICYNUMBER=32232	Mitgliedsnummer des Patienten bei der Krankenkasse

Die Bildinformationsübergabedatei hat den Aufbau einer Windows-Ini-Datei und sollte auf der lokalen Festplatte in einem temporären Verzeichnis liegen. Sie wird von der PVS erzeugt und enthält beim Aufruf der BVS folgende von der PVS eingeschriebenen Daten:

[PATID]	
PATID=23	PATID des Patienten von dem die Bildinformationen angefordert werden
PATSHOWNR=2004-133	Unterstützung der im KFO-Bereich üblichen Umnummerierung von Patienten je nach Bearbeitungsstatus (zur optionalen Verwendung)
MODELLNR=23-101	Unterstützung der im KFO-Bereich üblichen Modellnummer (zur optionalen Verwendung)
FROMPVS=PVS-Sektion	Sektionsname der PVS die den Patienten verwaltet (vgl. ID-Server in Abschnitt 2.2. auf Seite 10)
PRXNR=1	Praxisnummer zu der das Bild gehört. Pflichtfeld.
DATE=20010501	Bildinformationen aller Bilder ab diesem Datum übermitteln (Datum im Format CCYYMMDD, wenn leer dann alle Bilder, wenn „NEW“ dann nur Aufnahme neuer Bilder, wenn „SELECT[/<Titel>]“ dann muss die BVS den Anwender eine Auswahl zusammenstellen lassen, wobei Sie den optionalen Titel zur Verdeutlichung – Überschrift o.ä. – verwenden kann)
PVSIMP=0	Flag das angibt, ob das PVS-Importmodul aufgerufen werden soll (1=ja), oder die Bildinformationen direkt zu übergeben sind (=0)
PVS=PVS-Sektion	Sektionsname der aufrufenden PVS (für einen Aufruf des PVS-Importmoduls wenn PVSIMP=1)
BVS=BVS-Sektion	Sektionsname der BVS (damit die PVS die Bilder einer BVS zuordnen kann, d.h. die PVS muss zu jedem übernommenen Bildinformationssatz den Namen der BVS mit abspeichern, um das Bild später anfordern zu können)
THUMBNAILS=1	Flag das angibt, ob zusammen mit den Bildinformationen auch Vorschaubilder gewünscht sind. Dabei muss die PVS berücksichtigen ob die BVS diese Funktion unterstützt (SUPPORTTHUMBNAILS=1 in der BVS-Sektion).
THUMBNAILSX=320	Gewünschte maximale Breite der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
THUMBNAILSY=200	Gewünschte maximale Höhe der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
READY=0	Flag das angibt, ob die Datei bearbeitet wurde (0 bei Aufruf, 1 nach Beendigung)
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK , \geq 1=Fehler)

Wurden bei **PVSIMP=1** die Daten in der Datei von der BVS korrekt „verstanden“ bzw. bei **PVSIMP=0** alle Bildinformationen gespeichert, muss das BVS-Exportmodul das **READY-Flag** auf **1** setzen und die Datei „freigeben“. Bei **PVSIMP=1** muss die BVS das **READY-Flag** auf **1** setzen und **[MMOS]->COUNT** auf **0**. Die Patienten- und Bildinformationen müssen in einer neuen Datei abgelegt und das PVS-Importmodul aufgerufen werden.

Beim Einsatz eines Archivs muss der Eintrag **[PATID]->BVS** immer den Sektionsnamen der gewünschten BVS enthalten, es sei denn es wird eine Kommunikation mit dem Archiv gewünscht. Es wird allerdings immer das Modul des Archivs aufgerufen.

Aufbau der Bildübergabedatei nach dem Aufruf des BVS-Export-Moduls mit den Daten von zwei Bildern:

[PATID]	
PATID=23	
PATSHOWNR=	
MODELLNR=	
FROMPVS=PVS-Sektion	
PRXNR=1	
DATE=20010501	
PVSIMP=0	
READY=1	Wird bei Beendigung des BVS-Exportmoduls auf 1 gesetzt
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK , \leq 1=Fehler)
PVS=PVS-Sektion	
BVS=BVS-Sektion	
[MMOS]	
COUNT=2	Anzahl der übermittelten Bildinformationen
[MM01]	Sektionsname für Bild 1
MMOID=1244	Bildnr=BildID bis zu 200 alphanumerische Zeichen möglich. Pflichtfeld.
PRXNR=1	Praxisnummer zu der das Bild gehört. Pflichtfeld.
TYPE=Kleinröntgenbild	Bildtyp im Klartext zur optionalen Verwendung z.B. Bissflügel, OPG, Video, usw.
TYPENR=1	Eindeutiger Bildtyp gemäß unten stehender Tabelle. Pflichtfeld
TEETH=12	(weitere Möglichkeiten „ 11,13 “ oder „ 23-27 “ oder leer, wenn z.B. TYPENR=3 ist)
PREGNANCY=0	Flag, ob Patientin schwanger ist (0=nein , 1=ja ; Nur bei weiblichen Patienten, kein Pflichtfeld)
XRAYCOMMENT=	Freier Text für Röntgenbegründung/Indikation nach Röntgenverordnung (Wenn PREGNANCY=1, dann Pflichtfeld!)
XRAYMS=	Belichtungszeit in ms (Milli-Sekunden) bei Röntgenbildern, sonst leer
XRAYCURRENT=	Strom in mA (Milli-Ampere) bei Röntgenbildern, sonst leer
XRAYVOLTAGE=	Spannung in kV (Kilo-Volt) bei Röntgenbildern, sonst leer
EXT=TIF	für Bilder immer TIF, andere mögliche Objekte: WAV,AVI,DOC
COLORTYPE=COLOR	„GRAYSCALE“, „COLOR“ oder „LINEART“. Leer, bei WAV oder Text-Dokumenten. Pflichtfeld.
DATE=20010430	Aufnahmedatum im Format CCYYMMDD
TIME=10:45	Aufnahmeuhrzeit (leer, wenn unbekannt)

COMMENT=kein Befund	max. 255 Zeichen möglich
THUMBNAIL=...\thumb1.bmp	Pfad zur Thumbnail-Datei, wenn angefordert (Windows 24-Bit BMP-Format, vgl. auch Hinweis am Ende dieses Abschnittes)
[MM02]	Sektionsname für Bild
MMOID=1245	
PRXNR=1	
TYPE=Video	
TYPENR=6	
COLORTYPE=COLOR	
EXT=TIF	
TEETH=	
DATE=20010430	
TIME=11:00	
COMMENT=	
THUMBNAIL=<Pfad>	

Im Fehlerfall gibt das Exportmodul die entsprechende Fehlermeldung aus und setzt sicherheitshalber neben dem ERRORLEVEL auch den Eintrag **[MMOS]->COUNT** auf **0** damit die PVS über den Fehler informiert wird. Der Abbruch einer Aktion wie z.B. bei NEW, wird dann über die beiden Werte **ERRORLEVEL=0** und **[MMOS]->COUNT=0** eindeutig identifiziert.

Für den Eintrag **TYPENR** sind derzeit folgende Werte zugelassen, wobei hier die möglichen Palettentypen nur zur „Existenzberechtigung“ des **COLORTYPE** Eintrags aufgeführt sind:

TYPENR	Typ	COLORTYPE
0	DELETE	leer (Info über ein gelöscht Bild/Dokument)
1	Kleinröntgenbild	GRAYSCALE
2	Bissflügel	GRAYSCALE
3	PSA	GRAYSCALE
4	FRS	GRAYSCALE
5	Status	GRAYSCALE
6	Video	COLOR / GRAYSCALE
7	Foto	COLOR / GRAYSCALE
8	Intraorales Bild	COLOR / GRAYSCALE
9	Videobild	COLOR / GRAYSCALE
10	Formular (AU, Rezept, Erfassungsschein)	COLOR / GRAYSCALE / LINEART
11	Gutachten	COLOR / GRAYSCALE / LINEART
12	Rechnung	COLOR / GRAYSCALE / LINEART
13	Schriftverkehr Patient	COLOR / GRAYSCALE / LINEART

14	Kostenplan	COLOR / GRAYSCALE / LINEART
15	Eigenlabor-Rechnung	COLOR / GRAYSCALE / LINEART
16	Fremdlabor-Rechnung	COLOR / GRAYSCALE / LINEART
17	Materialbeleg	COLOR / GRAYSCALE / LINEART
18	Modellfoto	COLOR / GRAYSCALE
19	PA-Röntgen	GRAYSCALE
20	Handröntgen	GRAYSCALE
21	Analyseergebnis	COLOR / GRAYSCALE / LINEART (tabellarisch)
22	Durchzeichnung	COLOR / GRAYSCALE / LINEART (graphisch)
23	Sonstige	COLOR / GRAYSCALE / LINEART
24	Unbekannt	COLOR / GRAYSCALE / LINEART (z.B. für Dokumente die bereits auf proprietäre Weise in eine Anwendung gekommen sind).
25	Eigenlabor-Rechnung (Privat)	COLOR / GRAYSCALE / LINEART
26	Eigenlabor-Rechnung (Kasse)	COLOR / GRAYSCALE / LINEART
27	Fremdlabor-Rechnung (Privat)	COLOR / GRAYSCALE / LINEART
28	Fremdlabor-Rechnung (Kasse)	COLOR / GRAYSCALE / LINEART
29	FR frontal	GRAYSCALE
30	Kiefergelenkaufnahme	GRAYSCALE

Die PVS schreibt beim Aufruf die BILD-ID's aller gewünschten Bilder in die Datei.

[MMOIDS]	
PVS=PVS-Sektion	PVS welches die Bilder anfordert (Pflichtfeld bei Archiv-Unterstützung).
COUNT=2	Anzahl der übermittelten BildID's
MMOID1=1244	BildID des ersten gewünschten Bildes
MMOID2=1245	BildID des zweiten gewünschten Bildes
COLORDEPTH=24	Maximale von der PVS unterstützte / gewünschte Farbtiefe in Bit, wenn die BVS dies unterstützt (ohne Angabe = 24 Bit)
GRAYSCALE=16	Maximal von der PVS unterstützten / gewünschten Graustufen in Bit, wenn die BVS dies unterstützt (ohne Angabe = 8 Bit)
THUMBNAILS=1	Flag das angibt, ob anstelle von Bildkopien nur Vorschaubilder gewünscht sind. Dabei muss die PVS berücksichtigen ob die BVS diese Funktion unterstützt (SUPPORTTHUMBNAILS=1 in der BVS-Sektion)
THUMBNAILSX=320	Gewünschte maximale Breite der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
THUMBNAILSY=200	Gewünschte maximale Höhe der Vorschaubilder in Pixel (wenn nicht angegeben, der BVS überlassen)
READY=0	wird bei Beendigung des BVS-Exportmoduls auf 1 gesetzt
ERRORLEVEL=0	Rückgabe der Bearbeitung (0=OK , \geq 1=Fehler)

Die BVS erzeugt die Kopien der Bilder bzw. die Thumbnail-Dateien in einem Verzeichnis auf der lokalen Festplatte des Rechners (z.B. im Temp-Verzeichnis des Betriebssystems) und trägt die kompletten Pfadnamen in der Sektion **[MMOPATH]** ein.

[MMOPATH]	
MMOID1=c:\windows\temp\picture1.tif	Pfad auf die Bildkopie zu [MMOIDS]->MMOID1
MMOID2=c:\windows\temp\picture2.tif	Pfad auf die Bildkopie zu [MMOIDS]->MMOID2

Die genaue Beschreibung entnehmen Sie bitte dem entsprechenden Teil der Schnittstellenbeschreibung.

Punkt	Abschnitt/Anlage	Seite	Änderung
	Vorbemerkungen	3	Hinweis auf die Schnittstellenversion 1.3.
	Stufe 7	4	Stufe 7 für PVS-System: Export von Patientendaten (ID-Server)
1.	VDDS-Konfigurierungsdatei	6	Umstellung der Beschreibung des zentralen Verzeichnisses für die VDDS_MMI.INI mit Berücksichtigung von Unix-basierten Systemen. Die Beschreibung und das Vorgehen stammt von Herrn Jan Schönauer von der Fa. Crosssoft. Er gibt dazu folgende Begründungen an: <ul style="list-style-type: none"> <input type="checkbox"/> das /etc Verzeichnis existiert unter allen Unices inkl. Mac OSX. <input type="checkbox"/> Die Gruppe bzw. das Schreibrecht für die Programme wird benötigt, damit zur Laufzeit der Programme Änderungen, Anpassungen und Korrekturen durchgeführt werden können (automatische Pflege der „eigenen“ Einträge, Behebung einer evt. Zerstörung ohne den Eingriff von Root).
1.	Anlage 1 - PVS-Einträge	18	<ul style="list-style-type: none"> <input type="checkbox"/> Expliziter Hinweis beim MMOINFIMPORT Modul, dass im Eintrag keine Parameter stehen dürfen. <input type="checkbox"/> Wegen der Einheitlichkeit: WANTHUMBNAILS in THUMBNAILS umbenannt, THUMBNAILSX und THUMBNAILSX für das PVS-Importmodul ergänzt. <input type="checkbox"/> IDSERVER: Sektionsname der PVS, die Patientendaten zentral zur Verfügung stellt (vgl. neuer Abschnitt 2.1 "PATID-Server" auf Seite 10). <input type="checkbox"/> IDEXPORT, IDEXPORT_OS, IDEXPORT_EVENT Einträge für die ID-Server Funktion der PVS (Stufe 7). <input type="checkbox"/> Versionseintrag auf 1.3 geändert. <input type="checkbox"/> STAGES Eintrag um mögliche Stufe 7 ergänzt
1.	Anlage 2 - BVS-Einträge	19	<ul style="list-style-type: none"> <input type="checkbox"/> Explizite Pfad Hinweise bei den Modulen und dass im Eintrag keine Parameter stehen dürfen. <input type="checkbox"/> Versionseintrag auf 1.3 geändert.
2.	Anlage 3 - Übergabedatei	21	<ul style="list-style-type: none"> <input type="checkbox"/> PATSHOWNR: Optionale Unterstützung der im KFO-Bereich üblichen Ummummerierung von Patienten je nach Bearbeitungsstatus. <input type="checkbox"/> MODELLNR: Optionale Unterstützung der im KFO-Bereich üblichen Modellnummer. <input type="checkbox"/> FROMPVS: Sektionsname der PVS die den Patienten verwaltet <input type="checkbox"/> SEX= muss wegen des MPG ein Pflichtfeld werden ("Schwangerschaftserkennung")!
2.1.	PATID-Server	10	Komplett neuer Abschnitt der den ID-Server beschreibt.
3.1.	Anlage 5 - PVS-Übergabedatei an die BVS	25	<ul style="list-style-type: none"> <input type="checkbox"/> PATSHOWNR: Optionale Unterstützung der im KFO-Bereich üblichen Ummummerierung von Patienten je nach Bearbeitungsstatus. <input type="checkbox"/> MODELLNR: Optionale Unterstützung der im KFO-Bereich üblichen Modellnummer. <input type="checkbox"/> FROMPVS=PVS-Sektion: Sektionsname der PVS die den Patienten verwaltet. <input type="checkbox"/> Workflow-Beschreibung am Ende der Tabelle: Behandlung der Dateien und Daten bei PVSIMP=1.

3.2.	Anlage 6 - BVS-Übergabedatei an die PVS	26	<ul style="list-style-type: none"> <input type="checkbox"/> PATSHOWNR, MODELLNR, FROMPVS und PRXNR=1 ergänzt <input type="checkbox"/> PREGNANCY-Flag, das angibt ob die Patientin schwanger ist <input type="checkbox"/> XRAYCOMMENT: Indikation/Begründung die angegeben sein muss, wenn PREGNANCY=1 ist. <input type="checkbox"/> OPG=PSA: OPG ist ein eingetragenes Zeichen bzw. ein Produkt, PSA (Panorama-Schichtaufnahme) wäre die allgemeine Bezeichnung
4.1.	Bilddarstellung mit dem Viewer der PVS	14	<p>Das BVS-Feld in der Sektion [MMOIDS] ist überflüssig und wird gestrichen: ohne Archiv-Anbindung muss die PVS wissen von welcher BVS es Bilder anfordert, da das Modul der BVS aufgerufen wird. Dort ist diese Information dann völlig überflüssig. Mit Archiv-Anbindung kann die PVS nur die ID's des Archivs kennen und damit auch nicht wissen von welcher BVS die Bilder sind, dort könnte dann also höchsten immer nur BVS=BVS_ARCHIV stehen. Aber auch diese Angabe ist für das Archiv-Exportmodul uninteressant.</p>

Ablauf einer Kommunikation ohne Event

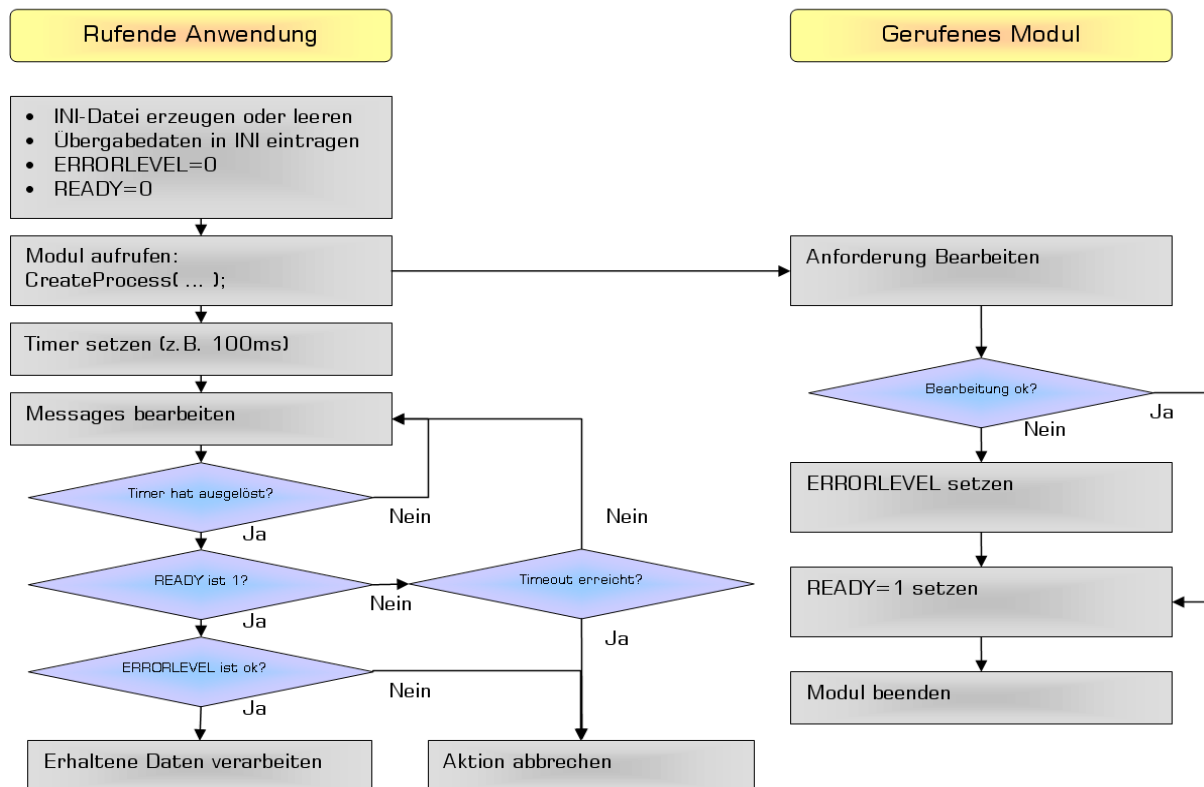
Die Synchronisation der Prozesse ist ein wichtiger Teil der Schnittstelle. Das unterschiedliche Verhalten der einzelnen Versionen von Windows gebietet es, einen näheren Blick auf diese Stelle zu werfen.

Empfehlenswert ist der Einsatz eines Timers, weil dieser in jedem Fall funktioniert. Wenn auf das Ende des Moduls gewartet wird, muss die rufende Anwendung sicherstellen, dass dem Modul ausreichend Rechenkapazität zur Verfügung gestellt wird, damit es überhaupt zu einem Ende kommen kann. Eine ungünstige Umsetzung kann die Reaktionszeit der Schnittstelle erheblich erhöhen.

Wird das Modul gerufen, erhält man unter Windows ein Prozesshandle. Theoretisch könnte man darauf warten, dass dieses Handle gefeuert wird, um das Ende des anderen Prozesses zu erkennen. Dies ist aber riskant, weil das gerufene Modul seinerseits einen anderen Prozess erzeugen und rufen und sich vor Ende des eigentlichen Moduls beenden kann. Der Fall ist nicht selten, die registrierte EXE für das Modul ruft die eigentliche Anwendung mit entsprechenden Parametern auf.

DOS-Anwendungen spielen eine immer geringere Rolle. Wenn sie auftreten, ist aber Vorsicht geboten. DOS-Prozesse benötigen ein permanentes Message-Bumping, d.h. die Zeit des Timers muss kürzer gewählt werden und anliegende Messages sind mit TranslateMessage und DispatchMessage weiterzuleiten.

Synchronisationen über WAIT-Funktionen sollten unbedingt mit MsgWaitForMultipleObjects erfolgen. WaitForSingleObject kann dazu führen, dass die Schnittstelle hängen bleibt, wenn das aufgerufene Modul eigene Fenster erzeugt, was die Schnittstelle nicht verbietet.



Ablauf einer Kommunikation mit Event

Unter Windows sollte die Möglichkeit von Events genutzt werden. Damit kann das gerufene Modul der rufenden Anwendung mitteilen, dass es mit der Bearbeitung fertig ist, ohne dass die rufende Anwendung auf das READY-Flag pollen muss.

Das Event muss von der rufenden Anwendung vor Aufruf des Moduls erzeugt werden. Anderenfalls könnte die Synchronisation fehlschlagen, weil die rufende Anwendung das Event früher abfragt, als das Modul dieses erzeugt.

Die Synchronisation sollte unbedingt mit `MsgWaitForMultipleObjects` erfolgen. `WaitForSingleObject` kann dazu führen, dass die Schnittstelle hängen bleibt, wenn das aufgerufene Modul eigene Fenster erzeugt, was die Schnittstelle nicht verbietet.

