

WhitePaper Systemfunktionen

1 Einführung

CodX PostOffice beinhaltet diverse Systemfunktionen, welche im Hintergrund arbeiten und den Betrieb sicherstellen. Diese Systemfunktionen sind für die Datenpflege, die Migration oder das Logging zuständig.

Dieses WhitePaper beschreibt diese Systemfunktionen.

2 Inhaltsverzeichnis

1	Einführung	1
2	Inhaltsverzeichnis	2
3	Installation Datenbanken	3
4	Produktivsystem / Testsystem	3
4.1	Produktivsystem	3
4.2	Testsystem.....	3
4.3	Datenbank als Testsystem konfigurieren	3
4.4	Einschränkungen eines Testsystems	3
4.5	Datenbank als Produktivsystem konfigurieren	4
5	Startup-Prozess CodX PostOffice.....	4
5.1	Startup-Prozess.....	4
5.2	Startverzögerung	4
6	DB-Update	5
7	Verwendung von Proxy	5
7.1	http-Proxy.....	5
7.2	ftp-Proxy	5
8	Save-Sendung-Server.....	5
8.1	Verarbeitung von unvollständigen Daten	5
8.2	Reihenfolge der Speicherung der Daten	5
9	Systemüberwachung CxWatchDog	6
9.1	Überwachungen	6
9.2	Alarme	7
10	Kontakt	7

3 Installation Datenbanken

CodX PostOffice setzt drei SQL-Datenbanken ein, welche für den Betrieb zuständig sind:

- ▲ PostOffice: Stammdaten und Live-Daten von *CodX PostOffice*
- ▲ Image: Bilder und Dokumente
- ▲ Archiv: Archiv-Daten

Diese Datenbanken können auf unterschiedlichen Datenbankservern installiert sein. Dies ist insbesondere bei sehr grossen Installationen sinnvoll.

Diese Datenbanken müssen kreiert, die Benutzer und Rollen angelegt und die Datenbankstruktur installiert werden. Zudem müssen noch diverse Stammdaten eingefügt werden. Optional können im gleichen Schritt auch Daten von einer anderen Datenbank übernommen werden.

4 Produktivsystem / Testsystem

In einer Installation von *CodX PostOffice* sind meist zwei getrennte Systeme installiert:

- ▲ Produktivsystem
- ▲ Testsystem

4.1 Produktivsystem

Das Produktivsystem wird für den produktiven Einsatz von *CodX PostOffice* verwendet.

4.2 Testsystem

CodX PostOffice kann als Testsystem konfiguriert werden. In der Regel wird dafür eine Kopie der produktiven Datenbank gezogen und diese auf getrennten Servern (Datenbank-Server, Applikations-Server) installiert werden. Auf dem Testsystem muss nicht zwingend dieselbe Version wie dem Produktivsystem installiert werden.

Das Testsystem kann anschliessend unabhängig vom Produktivsystem für Tests, Schulungen usw. verwendet werden.

4.3 Datenbank als Testsystem konfigurieren

Ob ein System als Testsystem fungiert, ist in der Datenbank hinterlegt. Es gibt verschiedene Möglichkeiten, die Datenbank als Testsystem zu konfigurieren:

- ▲ DBRestore-Tool: Wird die Datenbank des Testsystems mit dem DBRestore-Tool installiert, so kann die Option 'Testsystem' gleich bei der Installation angegeben werden.
- ▲ Einstellung in Config.xml: Wird im Config.xml die Einstellung *Testsystem* vorgenommen, so prüft *CodX PostOffice* beim nächsten Start, ob die Datenbank tatsächlich als Testsystem konfiguriert ist. Ist dies nicht der Fall, so wird diese nach Rückfrage an den Benutzer als Testsystem gesetzt.
- ▲ Einstellung im UI von *CodX PostOffice*. In den Einstellungen von *CodX PostOffice* kann die Datenbank auch als Testsystem geschaltet werden. Der Nachteil dabei ist, dass *CodX PostOffice* erst gestartet werden muss, bis die Einstellung vorgenommen werden kann. Deshalb sind die oben beschriebenen Optionen zu bevorzugen.

4.4 Einschränkungen eines Testsystems

In *CodX PostOffice* laufen verschiedene Aufgaben verteilt auf mehreren Computern gleichzeitig. Wenn nun ein Produktivsystem und ein Testsystem gleichzeitig in Betrieb sind, so besteht die Gefahr, dass sich die beiden Systeme in die Quere kommen. So ist es gegebenenfalls möglich, dass ein Import-Job auf dem Testsystem schneller startet als auf dem Produktivsystem und somit die Daten unter Umständen ins Testsystem und gar

nicht ins Produktivsystem importiert werden. Es sind auch Prozesse und Jobs denkbar, welche zu Datenverlust führen können.

Beim Konfigurieren der Datenbank als Testsystem werden alle kritischen Hintergrundprozesse ausgeschaltet oder die Einstellungen so geändert, dass diese dem Produktivsystem nicht in die Quere kommen.

Sollen diese Prozesse auch auf dem Testsystem verfügbar sein, so müssen diese entsprechend umkonfiguriert und manuell aktiviert werden.

4.5 Datenbank als Produktivsystem konfigurieren

Soll ein Testsystem als Produktivsystem geschaltet werden, so kann dies über das User Interface der Einstellungen von *CodX PostOffice* vorgenommen werden.

Dabei ist darauf zu achten, dass kein anderes System als Produktivsystem läuft.

5 Startup-Prozess *CodX PostOffice*

Wird *CodX PostOffice* gestartet, so wird ein Startup-Prozess durchlaufen, welcher die Konfiguration von *CodX PostOffice* vornimmt und allenfalls ein Software-Update durchführt.

Für die Durchführung des Startup-Prozesses ist der 'StartupServer' von *CodX PostOffice* zuständig.

5.1 Startup-Prozess

Grundsätzlich werden folgende Schritte durchlaufen:

- ▲ Auswahl des Application Servers, falls mehrere vorhanden sind
- ▲ Konfiguration vom Application Server herunterladen
- ▲ Prüfen, ob Update notwendig ist. Durchführung des Updates, falls nötig
- ▲ ShutDown- und Startup-Message anzeigen, falls nötig
- ▲ Konfiguration des Systems gemäss Config.XML
- ▲ Anpassung der ServerList gemäss ServerList.XML
- ▲ Anpassung der Registry gemäss Registry.XML
- ▲ Ausführen von Startup-Scripts gemäss Startup.vbs
- ▲ Startverzögerung, falls gesetzt (siehe unten)
- ▲ Starten der eigentlichen Applikation *CodX PostOffice*

Die Konfigurationsdateien liegen unter '%programdata%\CodX\PostOffice'.

Die genaue Beschreibung des StartupServers, des Startup-Prozesses und der entsprechenden Konfigurationsdateien finden Sie in der AdminDoc. Suchen Sie nach dem Begriff 'StartupServer'.

5.2 Startverzögerung

In speziellen Konfigurationen ist es erwünscht, den Start von *CodX PostOffice* zu verzögern. Dies ist insbesondere dann der Fall, wenn zum Beispiel auf demselben PC wie *CodX PostOffice* auch noch der SQL-Server installiert ist. In diesem Fall sollte *CodX PostOffice* verzögert gestartet werden, damit der SQL-Server Zeit zum Starten hat. Dies ist auch der Fall, wenn die Desktop-Instanz von *CodX PostOffice* auf einen funktionierende Server-Instanz von *CodX PostOffice* angewiesen ist.

Die Einstellung der Startverzögerung kann in der Registry (über Startup-Server) vorgenommen werden. Details dazu siehe AdminDoc (nach 'StartupServer' suchen).

6 DB-Update

CodX PostOffice führt die Struktur der Datenbank mit jeder Version automatisch nach. Dies erfolgt durch den DBUpdateServer beim erstmaligen Start von *CodX PostOffice* mit der neuen Version.

Dabei werden die Live-Daten, Image-Daten und die Archiv-Daten aktualisiert.

7 Verwendung von Proxy

CodX PostOffice unterstützt Proxys für http- und ftp-Verbindungen. Die Proxy-Einstellungen müssen in *CodX PostOffice* entsprechend vorgenommen werden. Diese finden Sie unter den Einstellungen / Allgemein / System-Einstellungen.

Die in *CodX PostOffice* vorgenommenen Proxy-Einstellungen werden automatisch an jeden Client verteilt und angewendet. Diese müssen nicht bei jeder Arbeitsstation separat eingestellt werden.

7.1 http-Proxy

Es werden folgende Funktionen unterstützt:

- ▲ Verschiedene Authentifizierungsmethoden
- ▲ Einstellungen von Proxy-Name oder IP, Port, Benutzername und Passwort separat für *CodX PostOffice*
- ▲ Verwendung der Proxy-Einstellungen vom Internet Explorer
- ▲ Unterstützung von Proxy-Scripts (eigene oder diese von Windows)

Die Details zu diesen Einstellungen finden Sie in der Online-Help von *CodX PostOffice*.

7.2 ftp-Proxy

Es werden folgende Funktionen unterstützt:

- ▲ Unterstützung verschiedener Proxy-Typen
- ▲ Einstellung von Proxy-Name, Port, Benutzername und Passwort separat für *CodX PostOffice*

Die Details zu diesen Einstellungen finden Sie in der Online-Help von *CodX PostOffice*.

8 Save-Sendung-Server

Der *Save-Sendung-Server* (SSS) ist eine zentrale Komponente, welche die Verarbeitung und Speicherung von Sendungsdaten vornimmt. Da die Speicherung der Sendungsdaten komplex und Ressourcen-intensiv ist, erfolgt dies zentral auf dem App-Server.

Alle Clients (z.B. Erfassungsstationen) und Module (z.B. SSI) speichern die Daten temporär in den *Save-Sendungs-Buffer*. Dieser Buffer wird anschliessend vom SSS asynchron verarbeitet.

8.1 Verarbeitung von unvollständigen Daten

Es gibt den Fall, dass die Sendungsdaten im ersten Schritt unvollständig sind und später in weiteren Schritten durch den entsprechenden Client vervollständigt werden.

Der SSS erkennt diese unvollständigen Datensätze und stellt diese zurück. Damit werden diese zu einem späteren Zeitpunkt nochmals verarbeitet. Dies wird mehrfach bis zu einem einstellbaren Maximum vorgenommen.

8.2 Reihenfolge der Speicherung der Daten

In den Einstellungen des *Save-Sendungs-Server* kann die Reihenfolge der Speicherung der Sendungsdaten bestimmt werden. Damit können bestimmte Module oder Erfassungsstationen (z.B. Sortiermaschinen) priorisiert werden. Diese Funktion benötigt die Lizenz LIC_F_SSS_SAVE_ORDER.

Es können beliebig viele Kriterien in die Reihenfolge mit aufgenommen werden. Einzelne Kriterien können auch mehrfach mit anderen Parametern aufgenommen werden. Damit lassen sich komplexe Priorisierungen für verschiedene Prozesse einstellen.

Ist keine spezifische Reihenfolge eingestellt oder die entsprechende Lizenz nicht vorhanden, so werden die Sendungen in der Reihenfolge der Speicherung in den Save-Sendung-Buffer abgearbeitet. (E#25148)

8.2.1 Fallbeispiele

Sortiermaschine	Werden mehrere Sortierläufe mit unterschiedlichen Sortiermaschinen durchgeführt, so müssen die Sendungsdaten der Grobsortierung für alle nachstehenden Sortierläufe bereitstehen. In diesem Fall kann der Sorter-UPOC der Erfassungsmaschine mit hoher Priorität eingetragen werden.
Sendungs-Editor	Werden manuelle Änderungen mit dem Sendungs-Editor vorgenommen, werden diese auch durch den <i>Save-Sendungs-Server</i> verarbeitet. Damit die Änderungen auch während der Produktion priorisiert verarbeitet werden, kann dies mit dem entsprechenden Eintrag mit 'Erfasst durch' eingestellt werden.
Bezirksstruktur	Wird in der Produktion unterschieden in Tag- und Nachtzustellung, so können die Sendungen der Nachtzustellungen mit der Priorisierung der entsprechenden Bezirksstruktur bevorzugt werden.

9 Systemüberwachung CxWatchDog

CodX PostOffice beinhaltet eine eigenständige Systemüberwachung *CxWatchDog*. Damit werden interne, wie externe Komponenten überwacht und nötigenfalls Alarm geschlagen.

9.1 Überwachungen

CxWatchDog kann folgende Überwachungen ausführen

- ▲ **Echo Ping:** ICMP-Ping
- ▲ **Laufwerk Speicherplatz:** Es kann das Lokale oder die Laufwerke anderer Rechner überwacht werden. Laufwerk: z.B. c:\
- ▲ **Prozess Speicherverbrauch:** Es kann der Speicherverbrauch lokaler Prozesse überwacht werden. Prozessname cxframer.exe
- ▲ **Datenbank Datendateigrösse:** Es wird die totale Grösse der Datendatei(en) überwacht.
- ▲ **Datenbank Logdateibelegung:** Es wird der Füllgrad in Prozent der Logdatei überwacht
- ▲ **Datenbank Abfrage:** Es wird eine Stored Procedur mit definierter Syntax aufgerufen und dessen Antwort ausgewertet.
- ▲ **Windows Script Abfrage:** Es wird ein beliebiges Visual Basic Script ausgeführt und dessen Antwort ausgewertet.
- ▲ **LogParser Abfrage:** Es wird ein Microsoft LogParser Abfrage ausgeführt und dessen Antwort ausgewertet.
- ▲ **CxDatalogger Überwachung:** Es kann ein CxDatalogger Filter definiert werden, so dass beim Auftreten dieses Log Eintrages ein Alarm generiert wird.
- ▲ **Nur Alarm:** Via Software-Interface kann die Anwendung einen Alarm auslösen
- ▲ **Ping:** Der CxWatchDog überwacht, ob die Anwendung regelmässig ein Ping ausführt

Die einzelnen Überwachungen und die zugehörigen Alarme werden bei der Konfiguration des Systems entsprechend eingestellt.

9.2 Alarme

Löst eine Überwachung ein Alarm aus, so kann der Alarm auf verschiedene Arten ausgelöst werden:

- ▲ ShellExecute
- ▲ eMail
- ▲ SMS (via CxMessage)
- ▲ Elektrischer Kontakt

10 Kontakt

CodX Software AG
Sinslerstrasse 47
6330 Cham
Schweiz

+41 41 798 11 22
info@codx.ch
www.codx.ch