Die Rückkehr der Mailbox

Die Kombination aus Mailboxprinzip und Internettechniken vereinfacht den Datenaustausch – gerade für KMU

Heinz W. Werntges
Fachbereich DCSM, FH Wiesbaden

Die Situation:
Der automatisierte Datenaustausch zwischen Geschäftspartnern gehört neben dem Mapping zu den Kernfunktionen des klassischen EDI. Value Added Network Services (VANS) verwendeten häufig das Mailboxprinzip und eine Hub-and-Spoke Topologie zur Realisierung ihrer Dienste: Geschäftspartner (die „spokes") stellten – meist mit proprietärer Technik – eine Verbindung zum VAN-Server (dem „hub") her, luden ausgehende Daten auf diesen Server und eingehende Daten vom Server. Die Adressierung erfolgte implizit über Sender/Empfänger-Kennungen in den standardisierten (!) Daten selbst, im Gegensatz etwa zur separaten Adressierungsebene in E-Mail.

In Europa wurden VANS später von Verbundnetzen auf Basis des OSI-Standards X.400 ergänzt und teils verdrängt. Die Standardisierung und der genormte Verbund aller wichtigen X.400-Anbieter (ADMDs) sorgte für Konkurrenz unter den Anbietern und damit für günstigere Volumengebühren als bei typischen VANS. Datenaustausch per X.400 erwies sich als sicher, flexibel, robust und einfach administrierbar.

Im Vergleich zu Übertragungskosten im Internet sind aber auch die X.400-Volumenkosten sehr hoch, so dass schon früh der Wunsch aufkam, die anspruchsvollen Anforderungen von EDI auch mittels Internettechniken zu erfüllen. Ein Resultat, EDI-INT AS2, wird aktuell als günstige Alternative zu X.400 propagiert.

Leider tauscht man bei AS2 einen Nachteil durch einen anderen aus: Ist eine AS2-Verbindung erst in Betrieb, sind die Volumenkosten zwar fast vernachlässigbar, dafür aber entstehen hohe Einmalkosten bei der komplexen Einrichtung. Der Betrieb von AS2 erfordert ferner dedizierte Hardware und sorgfältige Firewall-Konfiguration. Während größere AS2-Nutzer die Einmalkosten schnell amortisieren können, stehen KMU vor einem Dilemma: VANS und X.400 werden von ihren Kunden nicht mehr gewünscht, aber AS2 ist zu teuer und kompliziert in der Einrichtung. Die Folge ist meist eine Outsourcing-Strategie, wobei Anbieter von AS2-Diensten wieder die laufenden Kosten in die Höhe treiben.

Der Vorschlag:

Wir kombinieren die Einfachheit des Mailboxing-Prinzips und der Hub-and-Spoke Topologie mit allgemein verfügbaren Internet-Standards und Konventionen zur Verwaltung des „hubs" per Browser-Interface durch die Handelspartner selbst. Die verbleibenden Tätigkeiten des „hub"-Betreibers sind so gering, dass der Dienst besonders kostengünstig angeboten werden kann. Neue Partner lassen sich in wenigen Minuten einrichten, der Betrieb eigener Server seitens der Handelspartner entfällt.

DETAILS

Anmeldung

Neue Miglieder eines Verbundes melden sich am „hub" an, indem sie ein geeignetes Antragsformular per Web Browser auf den Internetseiten des „hubs" ausfüllen.

Pflichtfelder sind Firmenname, Liste der mit der beantragten Mailbox assoziierten GLNs (mindestens eine) dieser Firma, administrativer Kontakt (Klartextname, Anschrift, Telefon-/Fax-Nummer, E-Mail-Adresse, Einmal-Passwort (s.u.)), optional ist ein Firmenkurzname (für Tracking & Tracing).
Der Server sollte diese Daten nun validieren (möglichst automatisch, z.B. durch Abgleich der GLN-Liste mit dem Firmennamen mittels GEPIR und durch Prüfung auf Kollision mit bereits registrierten GLNs). Bei positivem Ergebnis wird dem neuen Mitglied eine Mailbox und ein Benutzerkonto eingerichtet, dessen Zugangsdaten ihm per Web Server zugänglich gemacht werden. Aus Sicherheitsgründen erhält der neue Benutzer diese Daten nicht direkt per E-Mail, sondern er erhält per E-Mail nur eine Web-Adresse, wo er seine Zugangsdaten – gesichert über https – gegen Eingabe seines Einmal-Passworts von der Anmeldung abrufen kann. Bei automatisierter Validierung benötigt der Server nur Sekunden, bei höheren Ansprüchen an die Validierung neuer Mitglieder ist eventuell noch ein manueller Prozess z.B. auf Basis eines Telefon- oder Fax-Kontaktes erforderlich.

Das Konto

Ein Konto wird mit einem Benutzernamen auf Betriebssystem-Ebene identifiziert. Zulässige Kontonamen sind daher vom Betriebssystem vorgegeben. Die erste GLN der gemeldeten Liste, eventuell mit einem festen Buchstaben-Präfix, bietet sich hier an.
Im Home-Verzeichnis eines solchen Benutzers befinden sich folgende Unterverzeichnisse:

in	Enthält eingehende Dateien von Geschäftspartnern
out	In dieses Verzeichnis werden zu versendende Dateien geladen
stat	Hier erzeugt das System Dateien mit Statussätzen zu versendeten Dateien
err	Verzeichnis zum Sammeln fehlerhafter Dateien. Das System verschiebt Dateien aus „out" hierhin, wenn sie z.B. Syntaxfehler oder unbekannte Empfängercodes enthalten.
tmp	Verzeichnis zur Aufnahme temporärer Dateien, sollte im Ruhezustand stets leer sein.

Konfiguration

Neue Mitglieder erscheinen mit ihren GLNs umgehend in der Liste der registrierten Mitglieder des Servers im Web. Sie verwenden nun diese Liste, um (nach Anmeldung an der Web-Schnittstelle) die GLNs auszuwählen, deren Besitzer an sie senden dürfen. Ihnen stehen ebenfalls die administrativen Kontakte der Mitglieder zur Verfügung, so dass Sie etwa per E-Mail Kontakt mit ihren Geschäftspartnern aufnehmen können, um diese um die Konfiguration ihrer GLNs zu ersuchen.

Datenaustausch

Mitglieder haben grundsätzlich nur Zugang zu den Verzeichnissen ihrer eigenen Mailbox („chroot"). Die Grundlage bildet das weit verbreitete und auf vielen Plattformen kostenlos verfügbare Secure Shell-Protokoll „ssh", das für Verschlüsselung der Daten während des Austauschs über das Internet sorgt. Der Zugang ist möglich mittels „ssh" (für administrative Zwecke), insbesondere aber mittels „scp" und „sftp". Über den mit ssh getunnelten Einsatz von „rsync" ist noch zu entscheiden, weitere Protokolle zum Datenaustausch zwischen Mitglied und Mailbox sind denkbar.

Senden

Mitglieder versenden Dateien, indem sie diese im Mailboxverzeichnis „out" erscheinen lassen. Eine Minute nach der letzten Änderung in einer dort erschienenen Datei (default-Einstellung, konfigurierbar) wird das System versuchen, diese zuzustellen. Es wird empfohlen, Dateien zunächst in den Ordner „tmp" zu transferieren und dann mit einem „atomaren" Schritt nach „out" zu verschieben, um jegliches Risiko auszuschließen, dass der Server Dateien zustellt, die noch nicht vollständig übertragen wurden.

Dateinamen sind im Fall von UN/EDIFACT-Daten beliebig wählbar, wobei der Absender darauf achten sollte, dass diese Namen möglichst eindeutig gewählt werden, sodass sie im EDI Tracking & Tracing (s.u.) nicht zu Missverständnissen führen. Der Server ermittelt die zur Zustellung benötigten Sender- und Empfängerkennungen aus den Composites S002 und S003 des UNB-Segments, in denen GLNs erwartet werden. Die Seriennummer wird DE 0020 (Interchange Control Reference, ICR) entnommen.
Zur Zustellung von Nicht-EDIFACT-Dateien gilt folgende Dateinamenskonvention:
<s>_<r>_<n>
<s> repräsentiert die Ziffernfolge der Sender-GLN, <r> die der Empfänger-GLN, und <n> die einer Seriennummer (analog zum Element DE0020 im EDIFACT-Fall, d.h mit Format „an..14", wobei als Nicht-Ziffern hier nur [A-Za-z] zugelassen werden).
Es ist möglich, auch UN/EDIFACT-Dateien dieser Namenskonvention zu unterwerfen. Dabei ist dann aber auf Übereinstimmung mit den entsprechenden UNB-Inhalten zu achten. Im Zweifelsfall hat der UNB-Inhalt Vorrang.
Die Kombination <s>, <r> und <n> bildet die Grundlage des EDI Tracking & Tracings und darf daher nur einmal verwendet werden. Dies gilt ebenso für die entsprechenden Inhalte des UNB-Segments von UN/EDIFACT Interchanges, wo diese Regel bereits etabliert ist und z.B. als Grundlage für Functional Acknowledgment mittels CONTRL-Nachrichten dient.

Ein erster Versendetest ist sofort nach Erhalt der Kontodaten möglich, denn die eigenen GLNs sind implizit stets auch gültige Absender-GLNs für den Empfang. Neue Mitglieder können also ohne weitere Konfiguration an sich selbst senden und damit ihre Sende- und Empfangs-Programme testen.

Zustellung

Der Server besucht periodisch (z.B. einmal pro Minute) die out-Verzeichnisse aller Mitglieder und sucht nach versandbereiten Dateien. Sender- und Empfänger-GLN werden extrahiert und auf Gültigkeit und Zulässigkeit geprüft. Je nach Syntax (etwa: UN/EDIFACT) und Inhalt (z.B. Rechnungsdaten nach EANCOM-02 und EDI-AK-Handel) können Validierungstests folgen.
Im Fall einer Ablehnung wird die Datei in den Ordner „err" verschoben und eine E-Mail mit der Fehlerursache an den zuständigen Administrator gesendet. Optional werden strukturierte Statussätze im Ordner „stat" angelegt.
Im Erfolgsfall wird die Datei in den Ordner „in" der zuständigen Empfänger-Mailbox kopiert. Eine positive Bestätigung per E-Mail geschieht nicht, wohl aber das Anlegen entsprechender Statussätze in „stat", wenn diese Option aktiviert wurde. Das Original wird aus dem Ordner „out" entfernt. In einer Ausbaustufe ist vorgesehen, es statt dessen in einen Unterordner „out/sent" zu verschieben und dort für eine einstellbare Zeit (z.B. eine Woche) vorzuhalten. Durch einfaches Verschieben einer Datei von „out/sent" nach „out" wäre so eine erneute Übertragung möglich.

Im EDI Tracking & Tracing werden alle Vorgänge protokolliert, die der Server ausführt bzw. entdeckt. Dazu mehr an anderer Stelle.

Empfangen

Eingehende Dateien erscheinen im Ordner „in". Sie unterliegen stets der in „Senden" beschriebenen Dateinamenskonvention <s>_<r>_<n>, also auch im UN/EDIFACT-Fall und wenn der Sender einen anderen Dateinamen wählte.
Diese Dateien sollten vom Anwender (per „scp" bzw. „sftp") auf den eigenen Rechner geladen und dann gelöscht werden. Der Löschvorgang wird aus Sendersicht als Statusfolge „Received", „Deleted" gewertet. Anwender sollten Dateien im Ordner „in" nicht selbständig umbenennen oder erzeugen, nur lesen oder löschen.
In einer Ausbaustufe könnte der Löschvorgang ersetzt werden durch ein Verschieben in einen Unterordner „in/received", in dem die Dateien eine einstellbare Zeit (default: eine

Woche) verbleiben, bevor der Server sie automatisch löscht. Auch eine Archivfunktion (mit Kompressionsschritt und Langzeit-Aufbewahrung) bietet sich an.

Protokollierung (EDI Tracking & Tracing)

Der Server verfolgt Dateien während ihres gesamten „Lebenszyklus". Sie werden eindeutig identifiziert über die Kombination aus Sender- und Emfänger-GLN und ICR. Alle Statusangaben mit Übereinstimmung in dieser „Signatur" werden zusammengefasst.
Dateien durchlaufen Ereignisse im Lebenszyklus zu wohlbekannten Zeitpunkten. Einem Ereignis entspricht ein Statuswechsel. Folgende Statuswerte werden registiert:

*	created	Anfangszustand. normalerweise Zeitpunkt des Erscheinens in „out" Angaben neben Syntax (E, I, andere) auch zu Dateigröße, Inhalt (Anzahl msgs bzw. groups), Konvertierzeitstempel aus UNB, ... ; noch auszugestalten
	incoming	Alternativer Anfangszustand, für Daten aus Gateways (Ausbaustufe)
	validated	Optional, Zeitpunkt durch Serverprozess bestimmt; evtl. mehr Details
*	rejected	Ende-Zustand bei Ablehnung, Zeitpunkt durch Serverprozess bestimmt
	transmitted	Zeitpunkt identisch mit dem von „delivered" bei Zustellung inner halb des Hubs, separater Zeitpunkt, falls versendet über Gateway (z.B. X.400, AS2)
*	delivered	Zeitpunkt der Zustellung ins Verzeichnis „in" (bzw. des MDN bei Gateways) Ende-Zustand, ab hier Verantwortung beim Empfänger.
	ndn	Nur bei Gateways: Fehler-/Ende-Zustand: Konnte nicht zugestellt werden. Möglichst mit Fehlerursache.
*	received	Zeitpunkt, zu dem der Server erstmals bemerkt, dass der Empfänger die Datei aus „in" entfernte.
	archived	optional, ggf. mit Angabe der Kompressionsmethode und der archiv ierten Dateigröße
	acknowledged	optional, erzeugt von CONTRL-Nachricht (Ausbaustufe, mit Statuscode)
*	deleted	Zeitpunkt, zu dem der Server erstmals bemerkt, dass der Empfänger die Datei aus „in" entfernte und sie auch nicht (mehr) in „in/received" existiert.

*) Pflichtwerte

(Noch zu diskutieren: „deleted" aus Sendersicht, Trennung in Sender- und Empfängersicht, demnach Verwaltung zweier Lebenszyklen: sender- und empfängerseitig? Optionales Verknüpfen beider? Status „translated" für optionale Übersetzer-Dienste vorsehen?)

T&T-Ereignisse werden in einer speziellen Datenbank gespeichert. Mitglieder können flexibel in „ihren" Statussätzen per Web-Schnittstelle recherchieren. Filter gestatten es etwa, Daten spezieller Geschäftspartner zu selektieren oder nur die noch nicht abgeholten Dateien oder nur Dateien mit Fehlerzuständen anzuzeigen. Das Konzept stammt aus [1].

Ausblick: EDI T&T könnte durch Anbindung externer Verarbeitungsschritte erweitert werden, um echte „end-to-end" Verfolgung von Anwendung zu Anwendung auch über Konverter und Unternehmensgrenzen hinweg zu erreichen. Prinzipiell ließen sich extern generierte Statussätze durchaus in die T&T-Datenbank des „hubs" einspielen, z.B. mittels Web Service-Schnittstelle (SOAP). Entsprechende „WS consumer" erforderten nicht unbedingt dedizierte Hardware seitens der Anwender, aber deren Erzeugung setzt viel Arbeit auf Anwenderseite voraus, und die Nutzung des Servers wäre nicht mehr so einfach. Diese tiefe Integration wird also eher ambitionierten Mitgliedern vorbehalten bleiben.

Statussätze

Alle in EDI T&T erzeugten Statussätze, die das Versenden einer Datei betreffen, können vom Server auch in strukturierter Form auf Dateiebene zur Verfügung gestellt werden. Per Konvention bildet jeder Statussatz eine Textzeile in einer Datei <s>_<r>_<n>.stat im Ordner „stat", z.B. im CSV-Format.
Alle Statussätze, die der Datei mit der angegebenen Signatur entsprechen, werden darin gesammelt.
Der Anwender lädt diese Statussätze auf seinen Rechner und löscht sie serverseitig, ähnlich wie eingehende Dateien aus „in". Da der Server hier neue Sätze anfügen könnte, während der Anwender die Datei gerade kopiert, sollten Statussatz-Dateien vom Anwender zunächst nach „tmp" verschoben und dann von dort kopiert werden, um sie anschließend zu löschen. Sollte der Server inzwischen eine Statusdatei gleichen Namens mit neuen Sätzen in „stat" angelegt haben, so wird diese später analog verarbeitet. Anwender sollten daher eingehende Statussätze sofort weiterverarbeiten, um Dateinamens-Kollisionen auszuschließen.

Die genaue Struktur der Statussätze ist noch zu erarbeiten.

Mitglieder müssen die Erzeugung von Statussätzen im Rahmen der Mailbox-Konfiguration erst aktivieren, sie ist standardmäßig ausgeschaltet.

Der Server

Als Hardware sollte ein „managed root server" genügen. Betriebssystem ist Linux. Dieses sollte vom Hosting-Dienst in Betrieb gehalten werden, zusammen mit der Hardware-Überwachung.
Der benötigte Festplatten-Platz hängt sehr vom Datenaufkommen und der Größe der Mitgliedergemeinde ab: Bei nur wenigen Mitgliedern (< 100) und einfachen Ansprüchen sollte praktisch jedes Rootserver-Angebot genügen. Steht dagegen der Austausch umfangreicher Daten (etwa vom Typ SLSRPT oder ENGDAT) mit langer mittlerer Verweildauer auf dem Server im Vordergrund, zeigt folgende Überschlagsrechnung den Bedarf auf:
Anzahl Mitglieder m * Anzahl Dateien pro Tag d * Anzahl Tage Verweildauer t * Bytes pro Datei s * 2 = Mittlerer Platzbedarf c: $c = 2 * m * d * t * s$
Für m = 100, d = 10, t = 7, s = 50 kB folgt c = 700 MB, bei m=1000 und s = 50 MB aber bereits c = 7 TB! Daraus ist zu schließen, dass das Datenvolumen anfangs keine große Rolle spielen dürfte, bei großem Erfolg des „hubs" aber durchaus dessen Kosten dominieren wird. Es liegt nahe, „quota"-Regelungen für die Mitglieder einzuführen, um ein drohendes Allmende-Problem zu vermeiden und die Mitglieder zum zeitnahen Löschen nicht mehr benötigter Daten anzuhalten.

Daemon

Die Kernfunktionen des Servers werden von einem einzigen daemon-Prozess wahrgenommen. Dieser kopiert Dateien im Rahmen der Zustellung, stößt Validierungen an, entfernt alte Dateien und protokolliert alle seine Aktivitäten in der T&T-Datenbank. Im Routinebetrieb gilt es, diesen Prozess zu überwachen, z.B. mit Watchdog-Mechanismen und Alarmierung eines zentralen Administrators.
Der Grad der Automatisierung dieses Prozesses und seiner Überwachung bestimmt maßgeblich die späteren Betriebskosten sowie die Verfügbarkeit der Plattform, daher sollte er besonders sorgfältig entworfen und implementiert werden.

Web Front-End

Hier kann auf Standardkomponenten zugegriffen werden, insbesondere auf Ruby on Rails als Framework zur Enwcklung der Web-basierten Anwendung und auf RDBMS wie MySQL oder PostgreSQL für die Datenhaltung. Im Betrieb wird die Hauptlast auf den EDI T&T-Anteilen liegen, während Anmeldung und Konfiguration nur gelegentlich genutzt werden.

Zusammenfassung und Ausblick

Sicherer und verlässlicher Austausch standardisierter EDI-Daten über das Internet lässt sich unter Benutzung allgemein verfügbarer Hilfsmittel wie scp/sftp und dem Prinzip der sich selbst verwaltenden Gemeinschaft sowie des Mailboxing-Verfahrens mit einer Hub-and-Spoke Topologie rasch und sehr preisgünstig erreichen. Hohe Einmalkosten wie bei EDI-INT AS2 entfallen. Auch die Betriebskosten des (einen!) Hubs sind sehr gering: Im Prinzip genügt ein "managed root-Server" und gelegentliches Nachsehen, ob Portal- und Hub-Software noch laufen. Manuelle Arbeit dürfte die (jeweils einmal erforderliche) Account-Vergabe machen, aber auch das wären nur Minuten pro Einzelfall.

Ein solcher Hub könnte von einem (dominierenden) Mitglied oder gleich von einem Interessenverband schlicht dauerhaft gesponsert und/oder durch die eigene IT-Abteilung betrieben werden, so dass auch keine Aufwände für Abrechnungen entstehen.

Mögliches Szenario:

8:00 Uhr: Konto / Mitgliedschaft beantragen
10:00 Uhr: E-Mai mit Hinweisen zum Erhalt der Konto-Daten/Passwort da (Wirkung: Überprüfung des Empfängers), optional: Kontrollrückruf oder -FAX zur Identitätsprüfung.
10:10 Uhr: Anmeldung am Portal, Konfiguration (eigene GLNs eintragen, GLNs der gewünschten Geschäftspartner freischalten, Partner informieren).
10:30 Uhr: Erstes Login per „ssh" bzw. erster Upload mit „sftp".

In einer Ausbaustufe ließen sich Hubs mit anderen Systemen per Gateway verbinden, z.B. über ein AS2-Gateway. Die Hub-Software wird im Rahmen von Diplomarbeiten entstehen und als Open Source allgemein verfügbar gemacht werden. Unterschiedlichste Benutzergruppen könnten also ganz einfach ihren eigenen Hub einrichten.

Literatur

[1] EDI Tracking & Tracing. H. Werntges, edi-change 03/2001, 26-34, Beuth-Verlag 2001.