

Kapitel 2

Änderungen in i-effect V1R7

*BASE

Neue Vorgangsverfolgung

Bei der Vorgangsverfolgung in i-effect handelt es sich um die Möglichkeit, mehrere asynchron verlaufende Prozess zu einem logischen Prozess zusammen zu fassen. Hierdurch werden alle zusammengehörigen Vorgänge mit Ihren beteiligten Objekten unter einem Ordnungsbegriff gruppiert.

Wird beispielsweise eine eMail empfangen, in der Anhänge gespeichert waren, könnten folgende Objekte auf dem System entstehen.

Erster Anhang aus eMail (eMailAnhang1.edi)

Zweiter Anhang aus eMail (eMailAnhang2.edi)

Archivdatei vom eMail Empfang (i-effectMailClient200909151338320.012-470_0.ieffect@menten.com.eml)

Bei einem ausgehenden Prozess könnten analog die folgenden Objekte entstehen.

Durch den Konverter erstellte EDIFACT Datei (4048454000005_ORDERS_2009-12-30-09.24.48.788000.edi)

Archivdatei vom eMail Senden(i-effectMailClient200909151338320.012-470_0.ieffect@menten.com.eml)

Werden andere Kommunikations- oder Funktionsmodule verwendet, kann die Liste der erstellten Objekte weiter anwachsen. So ist bei einer AS2 Kommunikation zusätzlich zu rechnen mit:

AS2 Header der gesendeten Nachricht (i-effect_AS2Client-200911191614372-7805@menten.header)

Empfangene MDN (i-effectAS2Server-20091124163003767-5036@as2.ieffect.mdn)

Auch bei einer in den Prozess geschalteten zusätzlichen Verarbeitung durch Komprimierung, Verschlüsselung, Signatur oder einer der anderen i-effect Funktionen entstehen Objekte, die alle unter einer gemeinsam zugeordneten Vorgangsnummer verwaltet werden.

Zur Einführung der zentralen Vorgangsverfolgung wurde ein Registrierungsdienst im *BASE Modul eingeführt, der neu erstellte Datei(en) in einem zentralen Repository registriert.

Hierbei wird über den eindeutig qualifizierenden Dateinamen und dessen Änderungs-Zeitstempel eine bereits bestehende Vorgangsnummer ermittelt und im weiteren Verlauf verwendet. Module die eine Eingabedatei verarbeiten, erfahren auf diesem Weg die bereits vorhandene Vorgangsnummer, unter der dieses Objekt bereits verarbeitet wurde und können dann weitere Dateien unter diesem Vorgang registrieren. So können Prozessketten, die über mehrere asynchron ablaufende ServerTask eingerichtet wurden, trotzdem einen Zusammenhang erstellen.

Ebenfalls über eine zentrale Schnittstelle registriert jedes Modul die in seinem Kontext zu einem Objekt bekannten Indexbegriffe bei dem zentralen Registrierungsdienst.

Sie können die Protokollierung für die Vorgangsverfolgung über das i-effect Menü 80 „Mit Programmmodulen arbeiten“ im *BASE Modul zentral ein- bzw. ausschalten.

*SERVER

Neuer *SERVER Workflow

Um auf Ereignisse innerhalb der i-effect Verarbeitungen mit spezifischen Abläufen reagieren zu können ist es notwendig diese Abläufe mit dem i-effect Server-Workflow erfassen zu können.

Bislang ist es möglich auf den Ausgang einer Server-Verarbeitung in den Stati *OK und *ERROR zu reagieren. Dies wurde durch die Modifikation des Server-Workflows auf *MSGID, *SQL, *DIAG und *REGEX erweitert:

- *MSGID** Ermöglicht es auf die internen i-effect Mitteilungs-IDs, welche in den verschiedenen Modulen von i-effect eindeutig vergeben sind, einen Workflow abzubilden
- *REGEX** Hierbei ist es möglich auf Textpassagen der i-effect Logbuch Meldungstexte zu reagieren. Dieses ermöglicht eine grösstmögliche Flexibilität hinsichtlich der Erstellung von Server-Verarbeitungen. So ist es nunmehr möglich auf ein einzelnes Wort - z.B. „Zertifikat“ - innerhalb eines Mitteilungstextes einen Workflow zu definieren.

***DIAG:** Diese Verarbeitung startet, wenn die entsprechende Serververarbeitung mit Status *DIAG beendet wurde und stellt eine Erweiterung zu *OK und *ERROR dar.

***SQL** Diese Verarbeitung arbeitet auf das i-effect Logbuch mit SQL Abfragen. Sie startet, wenn die entsprechende Serververarbeitung einen Logbucheintrag erzeugt hat, der mit der in diesem Eintrag hinterlegten SQL-WHERE-LIKE Bedingung zu einem Ergebnis führt. Zum Beispiel: SQL LIKE expression: %Mailvorgang erfolgreich gespeichert%

IFS *Monitor /*Schedule

Vor der Übergabe von Dateien an die Verarbeitung erfolgt nun eine Prüfung der Dateisperre mittels QPOLROR API. Dies verhindert die Verarbeitung von unvollständigen Dateien die noch von anderen Prozessen in Verwendung sind. Damit entfällt die Notwendigkeit die maximale Anzahl an aktiven Jobs in der EFFSEVER Jobqueue auf 1 zu setzen. Bisher war dies erforderlich um die Verarbeitung von z.B unvollständigen EDIFACT Dateien zu vermeiden die sich noch in der Konvertierung befinden.

