

# JTL Wawi Inputbox Tool

## Inhaltsverzeichnis

JTL Wawi Inputbox Tool.....	1
1. Beschreibung.....	2
2. Technische und fachliche Voraussetzungen.....	2
Technisch.....	2
Fachlich.....	2
3. Installation.....	2
4. Programm Idee.....	3
5. Programmeinrichtung - Verbindungsprofil anlegen.....	4
6. Kommandozeile – Parameter.....	6
7. Datenstruktur Datenbank.....	8
8. Workflow Beispiel Artikel TARIC Code per Textbox.....	9
9. Andere Workflows.....	12
10. Individuelle Eingabeformulare & mehrere Felder.....	13
11. Einschränkungen, Seiteneffekte.....	13
11. Kontakt.....	14

# 1. Beschreibung

Die JTL Wawi ermöglicht das Ausführen von Workflows, bei denen verschiedene Aktionen gestartet werden können.

Bisher war es jedoch nicht möglich, im Rahmen solcher Aktionen Benutzereingaben abzufragen und diese im Workflow zu verwenden. Eine Eingabemöglichkeit hat JTL nicht vorgesehen.

Die Wawi Inputbox ist eine Tool, welches genau diese Funktion bietet.

# 2. Technische und fachliche Voraussetzungen

## Technisch

- Windows 10/11, Server
- ab Wawi 1.3
- Zugriff auf die Wawi Datenbank. Schreibrechte sind nötig.
- Windows Adminrechte zur einmaligen Installation der Software

## Fachlich

Um die Arbeitsweise dieses Programms zu verstehen, sind IT Kenntnisse in folgenden Bereichen nötig

- Detaillierter JTL Wawi Administration und Betrieb
- SQL Datenbanken und SQL Abfragen
- Batchprogrammierung, Kommandozeile, Parameterübergabe
- Erkennen von Abhängigkeiten im Programm- und Workflowablauf

Bitte wenden Sie sich zur Installation und Einrichtung ggf. an einen Servicepartner oder den Programmautor.

# 3. Installation

Laden Sie die Installationsdatei herunter und führen Sie sie mit Administratorrechten aus.

Eine Windows Warnung können Sie ignorieren.

Die Installationsdatei ist vom Autor digital signiert.

Nach der Installation finden Sie das Programm Wawi Inputbox im Startmenü.

## 4. Programm Idee

Wenn Sie in einem Workflow eine Eingabe des Benutzers abrufen wollen, so wird im ersten Schritt des Workflows die Wawi Inputbox per Kommandozeile gestartet.

Dem Programm wird als Parameter ein zuvor definiertes Datenbank Verbindungsprofil und der Name des aktuellen Wawi Benutzers mitgeben, damit das Tool weiß, wie es sich mit der Datenbank verbinden kann.

Über weitere Parameter werden Informationen über die Art der Abfrage an den Benutzer übermittelt, zum Beispiele eine Frage bzw. Überschrift, damit der Benutzer weiß, was abgefragt wird.

Auch der Typ der gültigen Daten kann per Parameter definiert werden, z.B. Text, mehrzeiliger Text, Ganzzahlen, Dezimalzahlen oder eine Checkbox.

Die Eingabe des Benutzers wird auf Gültigkeit geprüft und anschließend als Ergebnis in eine eigene Tabelle in der Datenbank geschrieben.

Die Wawi Inputbox wird geschlossen und der erste Schritt des Workflows ist abgeschlossen.

Im zweiten Schritt des Workflows wird über eine eigene Erweiterte Bedingung zunächst geprüft, ob der Benutzer die Eingabe mit dem OK Button bestätigt hat. Falls nicht, wird der Workflow nicht ausgeführt.

Hat der User jedoch mit OK bestätigt, so wird der Workflow gestartet.

Es kann per SQL die eben durch die Wawi Inputbox in die Datenbank geschriebene Benutzereingabe wieder ausgelesen und als Rückgabewert innerhalb des Workflows genutzt werden.

Im dritten und letzten Schritt wird die Wawi Inputbox noch einmal aufgerufen, um den Eintrag aus dem ersten Schritt wieder aus der Tabelle zu entfernen.

## 5. Programmeinrichtung - Verbindungsprofil anlegen

Beim ersten Programmstart wird Ihnen ein Fenster angezeigt, in dem Sie ein neues Verbindungsprofil zu Ihrer Microsoft SQL Wawi Datenbank einrichten. Dies ist nötig, damit sich die Software mit der Datenbank verbinden kann.

Wenn Sie mehrere Mandanten innerhalb Ihrer Datenbank haben, können Sie je Mandant ein Profil anlegen, um auf dessen Daten zugreifen zu können.

Es können auch Verbindungsprofile zu unterschiedlichen Servern eingerichtet werden.

### Name

Der Name des Profils kann beliebig gewählt werden und dient Ihnen innerhalb der Software zur Unterscheidung verschiedener Profile.

**Der Profilname wird später als -p Parameter zum Aufruf des Profils verwendet.**

**Wenn Sie die Wawi Inputbox auf verschiedenen Rechnern zur Arbeit mit der gleichen Datenbank verwenden wollen, vergeben Sie immer identische Profilnamen für die gleichen Datenbankzugänge. Dies ist wichtig.**

### Server

Falls Ihr Datenbankserver unter einen bestimmten Port erreichbar ist, so muss die Portnummer getrennt mit einem Komma mit angegeben werden.

Beispiel:

meinserver.de\JTL-WAWi,50000

### Datenbank Name

JTL Standard für den Hauptmandanten ist eazybusiness, bzw., Mandant\_1, Mandant\_2 usw. für weitere Mandanten.

### Datenbank Benutzer

JTL Standard ist sa. Wenn Sie andere Benutzer verwenden, muss dieser mindestens Leserechte auf der Datenbank haben.

**Passwort**

JTL Standard ist sa04jT14

**Lizenzname und Schlüssel**

Tragen Sie hier Ihren Lizenzinformationen ein, um das Programm frei zu schalten.

**Verbindungstest und Speichern**

Testen Sie nach Eingabe aller Daten die Verbindung.

Nach erfolgreichem Test können Sie das Profil speichern und die Software nun ist einsatzbereit.

## 6. Kommandozeile – Parameter

Die wawi-inputbox.exe im Installationsordner wird zum Betrieb als Eingabeaufforderung mit Parametern aufgerufen.

**Kann der übergebene Wert Leerzeichen enthalten, so ist der Inhalt von Anführungszeichen zu umschließen.**

Die Parameter im Einzelnen sind:

### **-p=**

Profilname

Hier wird der Profilname angegeben, der bei der Profileinrichtung vergeben wurde.

Enthält der Name Leerzeichen, so ist er mit Anführungszeichen zu umschließen.

### **-u=**

Wawi Benutzername

Da mit der Wawi unterschiedliche Benutzer gleichzeitig arbeiten können und auch gleichzeitig Workflows in Ihrer Wawi ausführen können, muss je Nutzer unterschieden werden können.

Im Workflow steht der Benutzername mit {{ Vorgabe.AngemeldeterBenutzer }} zur Verfügung. Der Parameter -u= wird also stets im Workflow belegt mit

-u="{{ Vorgabe.AngemeldeterBenutzer }}"

### **-f=**

Frage bzw. Prompt für die Benutzereingabe

Legen Sie hier fest, welche Frage oder Anweisung dem Nutzer im Eingabedialog angezeigt werden soll.

### **-t=**

Typ der Eingabe

Legen Sie hier fest, welche Art Daten der Nutzer eingeben können soll. Mögliche Werte sind:

-t=INT           Ganzzahl, positiv, negativ, inklusive Null

-t=INT+          Ganzzahl, nur positiv, ohne Null

-t=INT0+         Ganzzahl, nur positiv, inklusive Null

-t=DEC           Dezimalzahl, positiv, negativ, inklusive Null

-t=DEC+          Dezimalzahl, positiv, ohne Null

-t=DEC0+         Dezimalzahl, nur positiv, inklusive Null

-t=TXT           Text einzeilig

-t=TXTML         Text mehrzeilig

-t=CHK           Checkbox, Parameter -cbt= ist Pflicht

-t=CMBLST        Combobox Auswahlliste ohne freie Eingabe, Parameter -lst= ist Pflicht

-t=CMBTXT        Combobox Auswahlliste mit zusätzlich freier Eingabe, Parameter -lst= ist Pflicht

-t=DAT           Datum

### **-v=**

Vorgabe (Der Parameter ist optional)

Legen Sie hier fest, ob und womit die Benutzereingabe vorbelegt sein soll.

Wird ein Text abgefragt, so können Sie einen Text vorgeben (in Anführungszeichen)

Wird eine Zahl abgefragt, so können Sie eine Zahl vorgeben.

Wird eine Checkbox abgefragt, so können Sie mit -v=1 die Checkbox standardmäßig setzen.

Wird eine Combobox angefragt, kann ein Element der Liste vorausgewählt werden.

**-cbt=**

CheckBox Text

Ist mit -t=CKB als Eingabe eine Checkbox festgelegt, so wird mit -cbt= die Beschriftung der Checkbox in Anführungszeichen übergeben.

**-lst=**

Liste auswählbarer Einträge

Der Eingabetyp Combobox Liste -t=CMBTXT und -t=CMBLST erfordern eine Liste möglicher Einträge, die dem Nutzer zur Auswahl angeboten werden.

Die einzelnen Elemente werden mit Pipe | getrennt. Es sind mindestens zwei Elemente nötig. Alle Elemente zusammen werden von Anführungszeichen umschlossen.

Beispiel: -lst="Hallo|JTL|Wawi"

Der Parameter kann mit dem Vorauswahlparameter -v= kombiniert werden, sofern die Vorauswahl ein Element der Liste ist.

Beispiel: -lst="Hallo|JTL|Wawi" -v="JTL"

Es wird das zweite Element JTL vorausgewählt.

**-mxl=**

Maximale Länge bei Text (Der Parameter ist optional)

Ist die Benutzereingabe mit -t=TXT oder -t=TXHTML auf einen Text festgelegt, so kann hier die maximale Länge der Eingabe beschränkt werden. Wird der Parameter nicht übergeben, liegt die maximale Länge bei 32767 Zeichen. Eine größere Länge ist auch per Parameter nicht möglich.

**-b=**

Breite des Eingabefensters (Der Parameter ist optional)

Die Breite wird in Pixeln angegeben. Ein kleinerer Wert als -b=290 ist nicht möglich.

**-h=**

Höhe des Eingabefensters (Der Parameter ist optional)

Die Höhe wird in Pixeln angegeben. Ein kleinerer Wert als -h=160 ist nicht möglich.

Wird mit -t=DAT ein Datum abgefragt, so liegt die minimal mögliche Höhe bei -h=300, weil das Kalender Steuerelement mehr Höhe benötigt.

**-reset**

Dieser Parameter entfernt den aktuellen Eintrag für den Benutzer aus der Datenbank.

Wird -reset aufgerufen, so sind als weitere Parameter nur -p=Profilnamen und -u=WawiBenutzer nötig.

## 7. Datenstruktur Datenbank

```
TABLE WawiHelfer.tWawiInputboxIO (  
    cWawiBenutzer VARCHAR(255) NOT NULL,  
    cFrage VARCHAR(1024) NOT NULL,  
    cID VARCHAR(64) NOT NULL,  
    cEingabeVarChar VARCHAR(MAX) NOT NULL,  
    nEinabeInt INT NOT NULL,  
    fEingabeDecimal DECIMAL(28,14),  
    dEingabeDatum DATETIME NULL,  
    nStatus INT NOT NULL,  
    dMod DATETIME NOT NULL  
)
```

Leider hat sich in der Struktur der Tabelle ein Tippfehler (nEinabeInt statt nEingabeInt) eingeschlichen.

Der Schreibfehler wird auch mit zukünftigen Versionen nicht mehr korrigiert werden, weil das Tool und damit Abfragen in Workflows bei Kunden im Einsatz sind. Sorry ;-(

Die Felder cWawiBenutzer und cFrage entsprechen den Werten, die über die Parameter -u= und -f= angegeben werden.

Je nachdem, welcher Eingabetyp vom Nutzer angefragt wird, werden unterschiedliche Felder in der Tabelle mit der Nutzereingabe belegt.

### Texteingaben durch Textfeld oder Combobox

t=TXT, t=TXHTML, t=CMBLST und t=CMBTXT belegen cEingabeVarChar.

### Ganzzahlen

-t=INT, -t=INT+, -t=INT0+ belegen nEinabeInt.

### Dezimalzahlen

-t=DEC, -t=DEC+, -t=DEC0+ belegen fEingabeDecimal.

### Checkbox

t=CHK

Belegt nEinabeInt mit 1 belegt, wenn die Checkbox gesetzt ist und nEinabeInt mit 0, wenn die Box nicht gesetzt ist.

### Datum

-t=DAT

Belegt das Feld dEingabeDatum mit dem Datum und der Uhrzeit 00:00:00 dEingabeDatum wird bei allen anderen Eingabetypen DBNULL.

Das Feld nStatus wird 1, wenn der Benutzer den Dialog mit OK abschließt. Bricht er ab oder schließt die Wawi Inputbox ohne Eingabe, ist nStatus = 0.

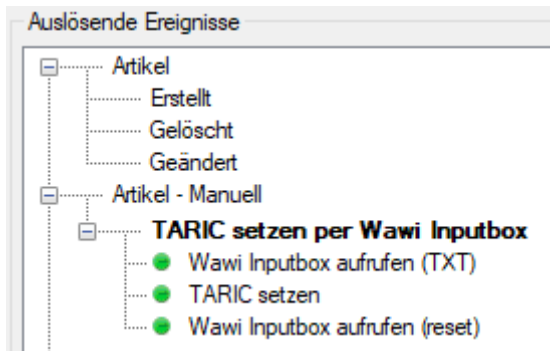
dMod enthält den Zeitpunkt der letzten Änderung, cID ist für zukünftige Zwecke reserviert.

Wird die Wawi Inputbox mit dem Parameter -reset und dazu gehörig -p=Profil und -u=WawiBenutzer gestartet, so wird der Datensatz, der cWawiNutzer = -u=WawiBenutzer entspricht aus der Tabelle gelöscht.



## 8. Workflow Beispiel Artikel TARIC Code per Textbox

Im Folgenden wird ein manueller Workflow für Artikel der den TARIC Code setzt gezeigt.



Es wurde bereits ein Wawi Inputbox Profil mit dem Namen JohnWawi wie in (5) beschrieben angelegt.

Im Bereich Workflows > Artikel – manuell wird ein neues Ereignis „TARIC setzen per Wawi Inputbox“ angelegt.

Darunter werden drei Workflows angelegt, die beim Auslösen des Ereignisses nacheinander ausgeführt werden.

### 1. Workflow „Wawi Inputbox aufrufen (TXT)“

Ausführen: C:\Program Files (x86)\wawi-inputbox\wawi-inputbox.exe -u={{ Vorgabe.AngemeldeterBenutzer }} -p="JohnWawi" -f="TARIC Code eingeben" -t=TXT -mxl=20 (auf Prozess warten)

Ausführen

Programm/Skript: C:\Program Files (x86)\wawi-inputbox\wawi-inputbox.exe Zurücksetzen ...

Datei auswählen

Parameter: nWawi" -f="TARIC Code eingeben" -t=TXT -mxl=20 Zurücksetzen ...

Auf Prozess warten: ☒

Kommandozeile benutzen: ☐

(Kommandozeile wird nicht sichtbar)

Ausführungsverzeichnis: Nicht ändern

#### Bedingung: Keine Aktion Ausführen

Programm/Skript: wawi-inputbox.exe im Installationsordner  
C:\Program Files (x86)\wawi-inputbox auswählen.

**Wichtig: Benutzen Sie unbedingt den Knopf Datei auswählen und geben Sie die exe nicht von Hand an, sonst funktioniert es nicht (Wawi Bug).**

Parameter:

```
-u="{{ Vorgabe.AngemeldeterBenutzer }}" -p="JohnWawi" -f="TARIC Code eingeben" -  
t=TXT -mxl=20
```

Der -p Parameter muss entsprechend Ihrem Profilnamen angepasst werden.

**Auf Prozess warten: AKTIV**

## 2. Workflow „TARIC setzen“

Zunächst wird eine eigene Erweiterte Eigenschaft mit dem Namen „Wawi-Inputbox-OK“ angelegt, indem über Erweiterte Eigenschaften verwalten der Editor aufgerufen wird.

Bedingungen

☒ Alle Bedingungen erfüllt ☐ Eine Bedingung erfüllt ☐ Keine Bedingung erfüllt ⓘ

▼ Wawi-Inputbox-OK gleich True

Erweiterte Eigenschaften\Wawi-Inputbox-OK Gleich Wahr

Neue

Felder durchsuchen

- Artikel
- Globale Texte
- Datum
- Erweiterte Eigenschaften
  - Wawi-Inputbox-OK

Erweiterte Eigenschaften verwalten

Legen Sie eine neue Eigenschaft an und vergeben Sie den Namen. Als Rückgabebetyp muss oben rechts im Editor Boolean gewählt werden.

JTL Erweiterte-Eigenschaften bearbeiten

Hier können Sie die Erweiterte-Eigenschaften bearbeiten. » Dokumentation

Artikel

- Wawi-Inputbox-OK

Name: Wawi-Inputbox-OK Rückgabebetyp: Boolean

```
1 {% capture q %}SELECT nStatus FROM WawiHelfer.tWawiI
2 {% assign Status = q | DirectQueryScalar -%}
3 {% if Status == 1 -%}
4 TRUE
5 {% else -%}
6 FALSE
7 {% endif -%}
```

Variablen

- Vorgang
- Datum
- Firma
- Globale Texte
- Tabulator
- Verzeichnisse
- Vorgabe

Der Code im mittleren Bereich lautet:

```
{% capture q %}SELECT nStatus FROM WawiHelfer.tWawiInputboxIO WHERE cWawiBenutzer =
'{{ Vorgabe.AngemeldeterBenutzer }}'{% endcapture -%}
{% assign Status = q | DirectQueryScalar -%}
{% if Status == 1 -%}
TRUE
{% else -%}
FALSE
{% endif -%}
```

Technisch wird also geprüft, ob der Aufruf der Wawi Inputbox mit OK oder anderweitig beendet wurde und als Boolean zurück gegeben.

Wurde die Erweitert Eigenschaft angelegt, wird sie zur Prüfungsbedingung = Wahr benutzt

## Workflow Aktion

### Wert setzen

Variable: Artikel\Sonstiges\Taric-Code

Der Code für Wert lautet:

```
{% capture q %}SELECT cEingabeVarChar FROM WawiHelfer.tWawiInputboxIO WHERE  
cWawiBenutzer = '{{ Vorgabe.AngemeldeterBenutzer }}'{% endcapture -%}  
{% assign EingabeVarChar = q | DirectQueryScalar -%}  
{{ EingabeVarChar }}
```

Die Wawi Inputbox wurde im ersten Schritt mit dem Eingabetyp -t=TXT, also Text betrieben. Das Ergebnis wird in cEingabeVarChar in die Datenbanken geschrieben.

Wird mit der Inputbox hingegen eine Zahl oder CheckBox abgefragt, so ist entsprechend nEingabeInt oder fEingabeDecimal auszuwerten.

Für die Abfrage eines Datums wird dDatumEingabe ausgewertet.

Workflow bearbeiten

Name: TARIC setzen Alle [i] Ausführungsplan

Bedingungen

☒ Alle Bedingungen erfüllt ☐ Eine Bedingung erfüllt ☐ Keine Bedingung erfüllt [i]

▼ Wawi-Inputbox-OK gleich True [trash]

Erweiterte Eigenschaften\Wawi-Inputbox-OK Gleich Wahr

[Neue Bedingung](#)

Aktionen

▼ Werte setzen: Sonstiges.TaricCode = {% capture q %}SELECT cEingabeVarChar FROM WawiHelfer.tWawiInputboxIO WHERE cWawiBenutzer = '{{ Vorgabe.AngemeldeterBenutzer }}'{% endcapture -%}' [trash]

Werte setzen Variable: Artikel\Sonstiges\Taric-Code

Wert: {% capture q %}SELECT cEingabeVarChar Zurücksetzen ...

Feld leeren: ☐

### 3. Workflow „Wawi Inputbox aufrufen (reset)“

Aktionen

▼ Ausführen: C:\Program Files (x86)\wawi-inputbox\wawi-inputbox.exe -u={{ Vorgabe.AngemeldeterBenutzer }} -p="JohnWawi" -reset (auf Prozess warten)

Ausführen

Programm/Skript: C:\Program Files (x86)\w Zurücksetzen ...

Datei auswählen

Parameter: -u="JohnWawi" -reset Zurücksetzen ...

Auf Prozess warten: ☒

Kommandozeile benutzen: ☐

(Kommandozeile wird nicht sichtbar)

Ausführungsverzeichnis: Nicht ändern

#### Bedingung: Keine

#### Aktion Ausführen

Programm/Skript: wawi-inputbox.exe im Installationsordner

C:\Program Files (x86)\wawi-inputbox auswählen.

**Wichtig: Benutzen Sie unbedingt den Knopf Datei auswählen und geben Sie die exe nicht von Hand an, sonst funktioniert es nicht (Wawi Bug).**

Parameter:

-u="{{ Vorgabe.AngemeldeterBenutzer }}" -p="JohnWawi" -reset

Der -p Parameter muss entsprechend Ihrem Profilnamen angepasst werden.

**Auf Prozess warten: AKTIV**

Durch diesen dritten und letzten Workflow wird das Ergebnis, welches die Wawi Inputbox im ersten Workflow in der Datenbank geschrieben hat und welches im zweiten Workflow zur Änderung der Taric Nummer benutzt wurde nun wieder aus der Datenbank entfernt.

Dies ist wichtig, damit für den nächsten Start der Wawi Inputbox wieder saubere Startbedingungen hergestellt sind.

## 9. Andere Workflows

Die Wawi Inputbox kann beliebig in Workflows genutzt werden, die aus der Wawi heraus (nicht jedoch über den Worker) aufgerufen werden. Zu beachten ist dabei grundsätzlich die Dreiteiligkeit des Aufrufs.

1. Aufruf der Wawi Inputbox mit gewünschtem Eingabetyp.
2. Auswerten der Eingabe per SQL. Zuvor muss über eine eigene Erweiterte Eigenschaft als Workflow Bedingung per SQL auf nStatus=1 geprüft werden, ob der Benutzer die Inputbox mit OK bestätigt hat oder nicht.
3. Bedingungsloser, erneuter Aufruf der Wawi Inputbox mit dem Parameter -reset, um die nicht mehr benötigten Daten zu entfernen.

## 10. Individuelle Eingabeformulare & mehrere Felder

Auf Anfrage lassen sich auch individuell nach Ihren Anforderungen Eingabeformulare mit beliebigen Eingabefeldern realisieren. Eine Plausibilitätsprüfung ist möglich.

Hier ein Beispiel, um Artikelmenge, Bezeichnung und Nettopreis abzufragen.

The screenshot displays the Wawi Inputbox application interface. On the left, a tree view shows the database structure. The main area shows a SQL query: `SELECT TOP (1000) [cWawiBenutzer] ... FROM [eazybusiness].[WawiHelfer].[tWawiInputboxIO]`. On the right, a dialog box titled "Bitte geben Sie die Positionsdaten ein" contains three input fields: a numeric field with "12", a text field with "Wundertüte extra groß", and a decimal field with "5.99". Below the dialog, a table titled "Ergebnisse" displays the following data:

	cWawiBenutzer	cFrage	cID	cEingabeVarChar	nEinabeInt	fEingabeDecimal	dEingabeDatum	nStatus
1	John	Bitte geben Sie die Positionsdaten ein	EX1_fAnzahl		0	12.000000000000000	NULL	1
2	John	Bitte geben Sie die Positionsdaten ein	EX1_cBeschreibung	Wundertüten extra groß	0	0.000000000000000	NULL	1
3	John	Bitte geben Sie die Positionsdaten ein	EX1_fPreisNetto		0	5.990000000000000	NULL	1

Wenn Sie Bedarf an einer solchen Lösung haben, nehmen Sie Kontakt mit mir auf.

## 11. Einschränkungen, Seiteneffekte

Es darf zu keinen Überschneidungen von ablaufenden Workflows im Kontext des gleichen Wawi Benutzers kommen.

Wawi Inputbox schreibt immer nur ein Resultat pro Benutzer in die Tabelle WawiHelfer.tWawiInputboxIO.

Wird die Wawi Inputbox ein zweites Mal bzw. erneut gestartet, während ein Workflow noch mit den Rückgabewerten des ersten Starts arbeitet, so kann es hier zu Überschneidungen mit falschen Ergebnissen kommen.

Ein Workflow muss immer zu Ende laufen und nach seinem Ende die Wawi Inputbox mit dem Parameter -reset gelaufen sein, bevor erneut Daten über einen weiteren Start abgefragt werden dürfen.

Da die Ergebnisse je Wawi Benutzer in der Datenbank geschrieben werden, dürfen keine zwei Wawis mit dem selben Benutzer aktiv sein und das Wawi Inputbox Tool aufrufen.

Scheitert der -reset Aufruf der Wawi Inputbox als letzter Schritt eines Workflows und scheitert auch der erneute Aufruf der Wawi Inputbox als erster Teil eines neuen Workflows, so stehen in der Datenbank noch die Rückgabewerte des letzten Aufrufs.

Grund für das Scheitern könnte z.B. ein nicht geändertes Wawi Wawi Inputbox Verbindungsprofil nach Passwortänderung oder sonstige Verbindungsprobleme mit der Datenbank sein.

Um hier eine weitere Sicherheitsmaßnahme einzuziehen, kann der Zeitstempel der tatsächlich in die Tabelle WawiHelfer.tWawiInputboxIO geschriebenen, letzten Eingabe im Feld dMod vor der Ausführung des Workflows geprüft werden.

Auch ein Timeout lässt sich hierüber realisieren.

Soll die Wawi Inputbox auf verschiedenen Rechnern bzw. Windows Installationen oder in verschiedenen Windows Benutzeraccounts auf dem gleichen Rechner mit ein und derselben Wawi Datenbank benutzt werden, so müssen in allen Installationen identische Namen für die Wawi Inputbox Profile benutzt werden, weil die Profile fest mit dem -p= Parameter in den Workflows namentlich aufgerufen werden.

## **11. Kontakt**

[www.wawi-helfer.de](http://www.wawi-helfer.de)

Dipl. Ing.(BA) John-Patrick Shaw

Archibaldweg 22

10317 Berlin, Deutschland

USt.-Identifikationsnummer: DE225980050

eMail: [info@wawi-helfer.de](mailto:info@wawi-helfer.de)

Tel: 030 51069123 (eMail bevorzugt)