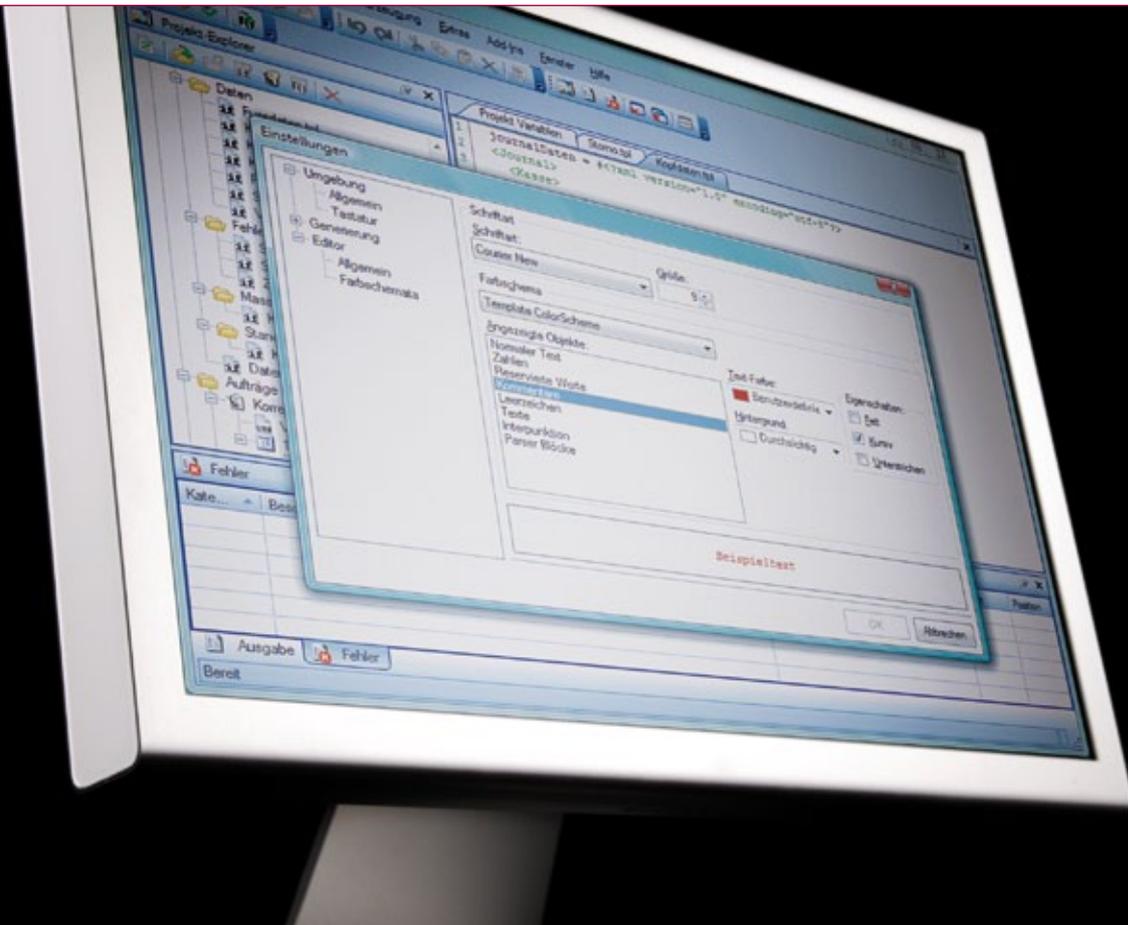


**LÖSUNGEN AUS DER PRAXIS**



**WILLKOMMEN  
DER TEST**

## Ihre Vorteile mit Q-up in Kürze:

- Bis zu 90 % weniger Zeit als bei manueller Eingabe
- Simulation realistischer Zeitabläufe in Echtzeit oder komprimiert
- Berücksichtigung beliebig vieler Eingabequellen, die parallel Daten erzeugen
- Simultanes Erzeugen der Daten in verschiedenen Zielformaten: parallel oder sequenziell
- Erzeugen gezielter Datenqualitäten und -quantitäten: korrekte sowie fehlerhafte
- Reproduzierbarkeit aller einmal erzeugten Daten
- Schnittstellen zur Integration eigener Funktionen

# WENIGER ZEIT MEHR DATEN IN DER ZUKUNFT DATENERZEUGUNG



## DER TESTDA

Q-up ist nicht einfach ein neuer Testdatengenerator. Dieses Programm kann deutlich mehr als andere! Es spart bis zu 90 % Zeit- und Personalkosten im Vergleich zur manuellen Erstellung von Testdaten. Gleichzeitig ist die Datenqualität sehr viel höher als bei herkömmlicher Testdaten-Software.

Mit Q-up ist es erstmals möglich, „fachliche Logik“ abzubilden. Das heißt: Durch die Simulation realistischer Situationen aus Ihrem Unternehmensalltag wie z. B. der Möglichkeit, Zeitabläufe widerzuspiegeln und beliebig viele Arbeitsplätze zu berücksichtigen, erzielen Sie höherwertige Testergebnisse. Gleichzeitig bietet Q-up zum ersten Mal die Möglichkeit, potenzielle Schwachstellen in Ihrer Datenbank durch den gezielten Einsatz fehlerhafter Daten aufzuspüren. So vermeiden Sie bereits im Vorfeld die Folgekosten unentdeckter Fehlerquellen.

Mithilfe der für Q-up neu entwickelten Schablonentechnik erzeugen Sie komplexe Datenstrukturen in beliebiger Quantität. Diese innovative Technik macht das Durchführen von Testprojekten nicht nur wesentlich einfacher, sondern dadurch auch sicherer und weitaus schneller. Sogar Änderungen in einzelnen

Schablonen während des Testprojekts sind damit im Handumdrehen erledigt. Weiterer Vorteil der Schablonentechnik: Die Daten sind jederzeit reproduzierbar. Dies ist besonders dann relevant, wenn gesetzliche Bestimmungen die Nachvollziehbarkeit oder Archivierung der Tests erfordern.

Q-up bietet maßgeschneiderten Service für jede Aufgabenstellung: Der Funktionsumfang kann durch Schnittstellen beliebig erweitert werden. Assistenten, wie der implementierte MSSQL-Assistent, helfen Ihnen, Arbeitsabläufe noch effizienter zu gestalten. Die Ausgabe kann gleichzeitig für verschiedene Ziele wie Datenbanken, Dateien oder Message-Queues erfolgen.

**Kurz: Q-up stellt einen Meilenstein in der Testdatenerstellung dar. Überzeugen Sie sich selbst!**

# STENGGENERATOR MIT IQ

## Sie sind in der Geschäftsleitung?

### Q-up sichert Ihnen Wettbewerbsvorteile:

- durch signifikante Zeit- und Kostenersparnis
- durch qualitativ hochwertigere Tests
- durch Archivierung und Bereitstellung rechtsrelevanter Datensätze



## Sie leiten die QS oder Entwicklung?

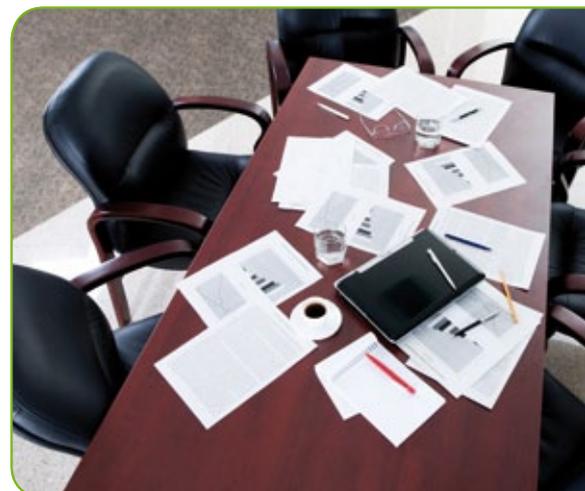
### Q-up hilft Ihnen bei der Problemlösung:

- durch Integration fachlicher Logik
- durch effektiveres Arbeiten
- durch Flexibilität im Datenimport und -export

## Sie arbeiten in der QS oder Entwicklung?

### Q-up unterstützt Sie bei Ihren Aufgaben:

- durch zeitsparende und sichere Schablonentechnik
- durch Schnittstellen zur Anpassung an Ihr System
- durch Assistenten zum effektiveren Arbeiten





## Q-UP - GENIAL EINFACH

Wie so oft liegt die Genialität in der Einfachheit. Q-up ist so strukturiert, dass Projekte möglichst schnell bearbeitet, archiviert, gewartet und ergänzt werden können.

### Wesentliche Bestandteile sind:

**PROJEKTE**  
**SCHABLONEN**  
**AUFTRÄGE**

### Ergänzt wird der Funktionsumfang durch:

**FUNKTIONSEDITOR**  
**ASSISTENTEN**  
**SCHNITTSTELLEN**

In einem Projekt werden Schablonen und Aufträge verwaltet. Welche konkreten Daten durch diese Schablonen generiert werden, wird durch die Aufträge definiert. Schablonen müssen also nur einmal erzeugt oder geändert werden, um für verschiedene Aufträge genutzt werden zu können. Änderungen an der Datenstruktur betreffen nicht mehr Zehntausende oder gar -tausende bereits vorhandener und für die laufenden Tests verwendeter Dateien, sondern im günstigsten Fall nur eine einzige Schablone.

# DAS HERZSTÜCK VON Q-UP: DIE SCHABLONEN

Die Schablone definiert die zu erzeugenden Daten, deren Format und Inhalt. Schablonen enthalten auch die Verarbeitungslogik für die Erzeugung von Daten und sie spezifizieren das Ziel (Datei, Datenbank usw.), in das die Daten geschrieben werden sollen.

Dabei können Daten sogar veränderliche Größen enthalten, wie z. B. eine laufende Nummer, einen Buchungsbetrag oder einen Saldo, die als Variablen definiert werden. Mithilfe von Funktionen kann innerhalb der Schablone auch eine Aktion ausgeführt werden, wie Berechnungen oder ein Datenabgleich.

Eine enorme Arbeitserleichterung stellt das Erzeugen von Schablonendaten per Kopieren und Einfügen von bereits existierenden Dateien dar. Diese werden im Editor für die gewünschte Datenausgabe einfach modifiziert, wie das Beispiel unten zeigt.

Darüber hinaus enthalten Schablonen Anweisungen, die festlegen, wie die Daten und deren veränderliche Größen verarbeitet werden sollen. So werden Anweisungen verwendet, um Variablen zu initialisieren, deren Inhalte zu verändern oder die erzeugten Daten in ein Ziel zu schreiben (z. B. eine Datei oder eine Datenbank).

## Aus einer bestehenden XML-Datei ...

```
<?xml version="1.0" encoding="UTF-8"?>
<Buchung>
  <Kassenummer>12</Kassenummer>
  <Filialnummer>1000</Filialnummer>
  <Journalnummer>1</Journalnummer>
  <Journalatznummer>1</Journalatznummer>
  <Betrag>55,00</Betrag>
  <Saldo>1250,30</Saldo>
</Buchung>
```

## ... wird in wenigen Schritten eine Schablone.

```
// Erhöhen der Journalsatznummer:
//
journalsatz += 1;

// Aufruf der Funktion zum Schreiben von Text in eine Datei:
//
WriteFile(

// 1. Argument für WriteFile(): Formatierter Dateiname,
// bestehend aus den in Format() angegebenen Argumenten:
//
Format("{0}-{1}-{2}-{3}.xml", filiale, kasse, journal, journalsatz),

// 2. Argument für WriteFile(): Zu schreibende Daten in
// Form eines Textblocks:
//
```

### ANWEISUNGEN

```
#<?xml version="1.0" encoding="UTF-8"?>
<Buchung>
  <Kassenummer>@kasse@</Kassenummer>
  <Filialnummer>@filiale@</Filialnummer>
  <Journalnummer>@journal@</Journalnummer>
  <Journalatznummer>@journalsatz@</Journalatznummer>
  <Betrag>@betrag = FRand() * 500 - 1000@</Betrag>
  <Saldo>@saldo += betrag@</Saldo>
</Buchung>#);
```

### DATEN

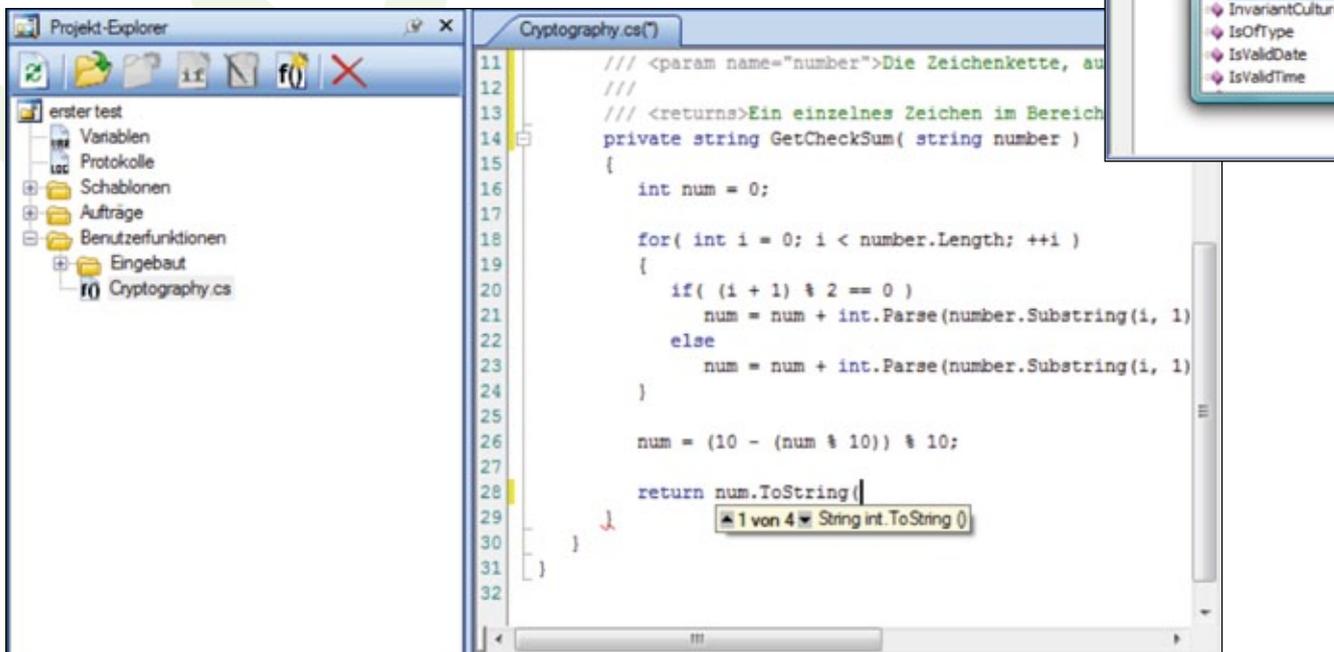
Feste Werte der Ausgangsdatei werden durch „@“ in Variablen umgewandelt.

FRand() gibt immer einen zufälligen Wert zwischen 0 und 1 (0 - 1) zurück. Der Ausdruck FRand()\*500-1000 errechnet also einen zufälligen Wert zwischen -500 und +500.

## Q-UP IST INDIVIDUELL

Unterschiedliche Testszenarios bedürfen auf die jeweilige Aufgabe zugeschnittene Daten. Deshalb bietet Q-up die Möglichkeit, mithilfe von Schnittstellen und Assistenten den Funktionsumfang beliebig zu erweitern.

Bei Projektvariablen handelt es sich um benutzerdefinierte Variablen, die innerhalb der Schablonen verwendet werden können. Diese sind auftragsübergreifend gültig und können in jedem Kontext benutzt werden. Sie werden in einem Editorfenster (siehe rechts) wie eine Schablone bearbeitet.



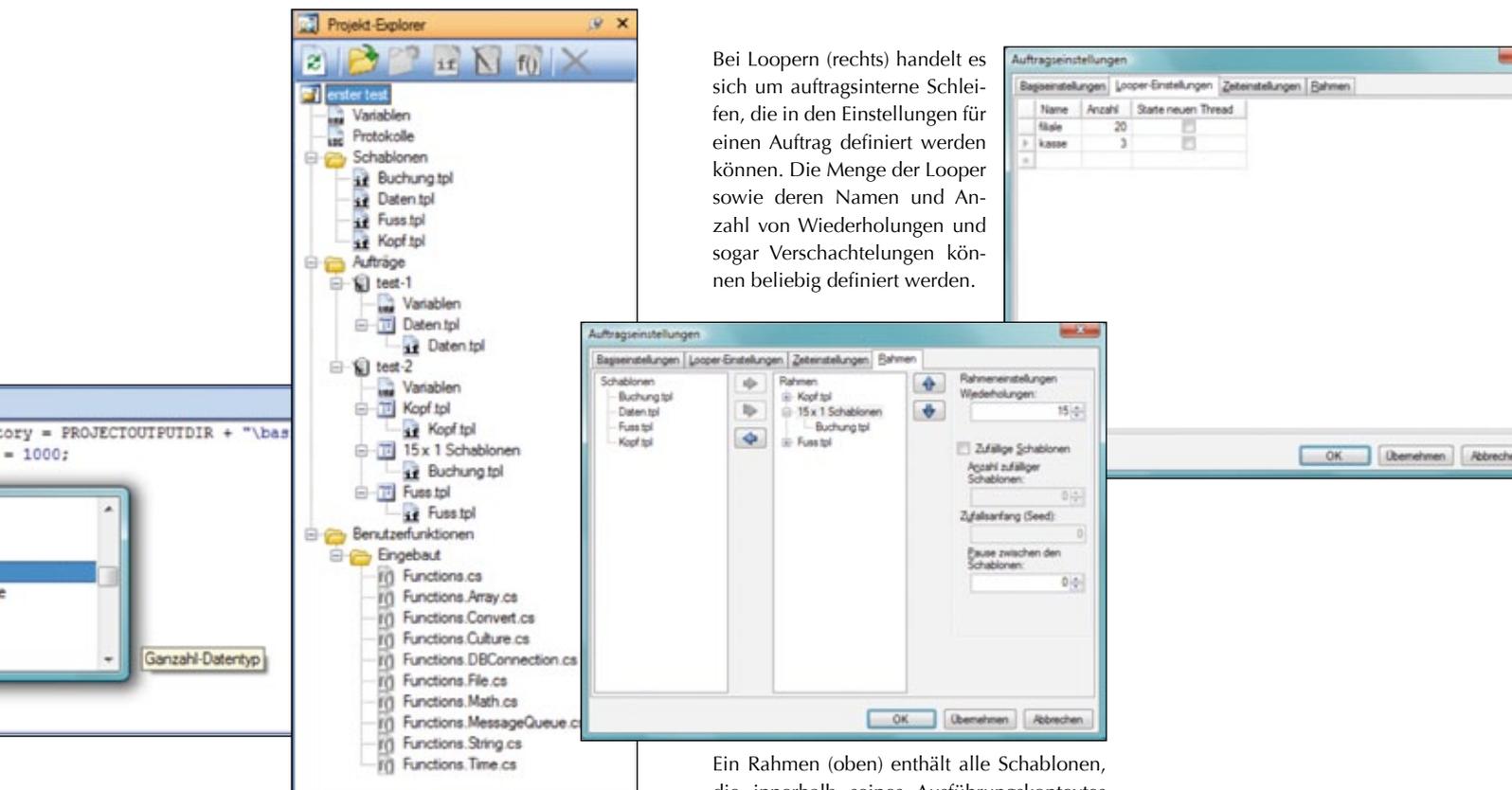
### Der Funktionseditor

Datenstrukturen werden immer komplexer. Deshalb ist es schwierig vorherzusagen, welche speziellen Berechnungen durchgeführt oder für welche ausgefallenen Ziele Testdaten erzeugt werden müssen. Daher besteht die Möglichkeit, den Funktionsumfang von Q-up zu erweitern.

Dazu kann z. B. über das Kontextmenü des Projekt-Explorers eine neue Funktion eingefügt werden. Es wird in diesem Fall ein Gerüst erzeugt, das mit der benötigten Funktionalität ausgefüllt werden kann. Die hinzugefügte Funktion kann anschließend in einer Schablone verwendet werden, so als wäre sie ein vordefinierter Bestandteil von Q-up.

### Vorteile

- Erweiterung der Möglichkeiten, die Schablonen bieten
- Problemloses Hinzufügen weiterer Datenquellen und -ziele (z. B. Webservices, SAP, Message-Queues von Drittanbietern wie RIPOSTE usw.)
- Vollständiger Zugriff auf den Funktionsumfang von .NET
- Einfache Anbindung und Nutzung der Programmbibliotheken von Drittanbietern
- Einfache Anbindung und Nutzung selbst entwickelter Programmbibliotheken



Bei Loopern (rechts) handelt es sich um auftragsinterne Schleifen, die in den Einstellungen für einen Auftrag definiert werden können. Die Menge der Looper sowie deren Namen und Anzahl von Wiederholungen und sogar Verschachtelungen können beliebig definiert werden.

Die meisten Verwaltungsarbeiten lassen sich bequem über das Kontextmenü des Projekt-Explorers (oben) ausführen. Schablonen können einfach durch Ziehen mit der Maus in Aufträge eingefügt oder darin verschoben und kopiert werden.

Ein Rahmen (oben) enthält alle Schablonen, die innerhalb seines Ausführungskontextes verarbeitet werden sollen. Dabei kann sehr detailliert angegeben werden, wie die Verarbeitung der im Rahmen enthaltenen Schablonen erfolgt.

## Assistenten

Zur einfacheren Verwaltung von Datenbanken ist in Q-up ein SQL-Datenbankassistent implementiert. Q-up bietet die Möglichkeit, der Umgebung über eine Programmierschnittstelle weitere Assistenten hinzuzufügen.

## Schnittstellen

Bei der Schnittstelle handelt es sich um Schnittstellenklassen, die von jedem .NET-Programm implementiert werden können. Diese Schnittstellen bieten die Möglichkeit, die Menüstruktur von Q-up um eigene Befehle zu erweitern sowie Schablonen, Aufträge und Rahmen hinzuzufügen, zu ändern und zu löschen.

## Lizenzmodelle

Auf [www.q-up-data.com](http://www.q-up-data.com) können Sie eine Testversion von Q-up herunterladen und kostenlos 30 Tage testen.

Diese Version können Sie durch Erwerb eines Lizenzkeys jederzeit in eine Vollversion umwandeln. So gehen Ihnen keine Daten verloren, die Sie bis zu diesem Zeitpunkt erstellt haben.

Vollversionen von Q-up gibt es als Professional-Lizenz für einen Arbeitsplatz sowie als Enterprise-Lizenz für mehrere Anwender.

Sie haben Fragen oder wünschen eine ausführliche Beratung? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr

## ANWENDUNGSBEISPIELE AUS DER PRAXIS



### Verarbeitung der Eingangsmeldungen in einem Callcenter

Die Software eines Callcenters verwaltet Daten, die sowohl automatisch als auch manuell von mehreren Arbeitsplätzen erfasst und in einer zentralen Datenbank analysiert, sortiert und weiterverarbeitet werden.

#### Fachliche Anforderungen

Es gibt Kategorien und Zustände, die in Abhängigkeit voneinander angegeben werden müssen. Zum Beispiel ob ein Fehler vorliegt und Unterstützung gegeben werden konnte, die zur Problemlösung beigetragen hat, oder eben nicht. Eine laufende Nummer wird automatisch beim Erstellen der Datei hochgezählt.

#### Ziel

Für den Test der Sortierung und der korrekten Weiterverarbeitung sollen Protokolle der Kategorien „Fehler“ und „Unterstützung“ erzeugt und in ein Übergabeverzeichnis gestellt werden. Zu jeder Kategorie müssen auch Protokolle mit den Zuständen „Gelöst“ und „Offen“ vorhanden sein.

Für Massentests müssen Protokolle mit allen oben angegebenen Kategorien und Zuständen für mindestens 20 Arbeitsplätze erzeugt werden, wobei für jeden Arbeitsplatz 200 Protokolle bereitgestellt werden müssen. Arbeitsplätze werden von dem zu testenden Programm über den im Protokoll befindlichen Namen identifiziert. Q-up muss die Namen für die 20 Arbeitsplätze selbstständig erzeugen und zuordnen.

## Von Q-up bereitgestellte Funktionen

- Schreiben von Daten in eine Datei
- Formatierung von Daten
- Simulation unterschiedlicher Arbeitsplätze durch Nutzung von Loopern
- Erzeugung unterschiedlicher Datenmengen mit denselben Schablonen durch Nutzung mehrerer Aufträge

## Aufwand

Bei manueller Erzeugung und einem optimistischen Durchschnittswert von zwei Minuten pro Protokoll benötigt man für die 4.000 Protokolle mehr als 130 Stunden, also mehr als 16 Arbeitstage!

Wenn man sich von vornherein im Klaren darüber ist, wie man Schablonen und Aufträge für diese Szenarien organisiert, benötigt ein versierter Q-up-Nutzer für die Erzeugung der Schablone und der Aufträge etwa zehn Minuten.

Ein einzelner Erzeugungsvorgang für die 4.000 Protokolle dauert um die zwei Minuten.

Dieses Szenario demonstrieren wir ausführlich und mit einem praktischen Beispiel auf unserer Internetpräsenz [www.q-up-data.com](http://www.q-up-data.com).

Sie haben weitere Fragen? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr



## ANWENDUNGSBEISPIELE AUS DER PRAXIS



### Verarbeitung von Kassenjournalen

Handelskonzerne mit mehreren Filialen verfügen häufig über einen zentralen Server mit einem Programm, das aus allen eingehenden Filialen (teils mit mehreren Kassen) die eingehenden Daten prüft. Dabei werden inkorrekte Journale aussortiert und für eine Korrektur bereitgestellt. Alle anderen werden für die Verarbeitung an ein Buchungssystem weitergeleitet. Das Programm, das diese Prüfungen durchführt, soll getestet werden. Dazu müssen korrekte und fehlerhafte Journale bereitgestellt werden.

### Fachliche Anforderung

In diesem Szenario müssen neben erheblichen fachlichen Informationen auch buchhalterische Regeln beachtet werden. Saldostände müssen logisch addiert werden, Buchungszeiten müssen fortlaufend sein, ebenso wie Journalnummern. Die verschiedenen Buchungstypen wie Kassenöffnung, Kassenschluss, Verkauf, Pfandrückgabe, Storno müssen unter buchhalterischen Aspekten berücksichtigt werden.

### Ziel

**Es könnten u. a. folgende Testszenarien bedient werden:**

1. Korrekte Journale für eine beliebige Kasse in einer beliebigen Filiale. In dem erzeugten Journal sollen, neben der Kassenöffnung und dem Kassenschluss, Buchungen in bestimmter Reihenfolge enthalten sein. Die Beträge und Artikelmen gen können zufällig ausgewählt werden.
2. Journale für eine beliebige Kasse in einer beliebigen Filiale, in denen aus buchhalterischer Sicht fehlerhafte Daten enthalten sind.
3. Für Massentests können jeweils korrekte Journale für mehrere Kassen aus beliebig vielen Filialen erzeugt werden. In den erzeugten Journalen sollen neben der Kassenöffnung und dem Kassenschluss jeweils 100 zufällige Buchungen enthalten sein.

Da die Journale letztendlich in einer zentralen Datenbank landen, in denen Abhängigkeiten bestehen, die sich nicht einfach auflösen lassen, muss dafür gesorgt werden, dass die für einen bestimmten Test erzeugten Testdaten schlüssig zu den Daten einer vorhergehenden Datenerzeugung sind. Das bedeutet, dass für eine Kasse, die in einem vorhergehenden Erzeugungslauf verwendet wurde, die relevanten Daten (Journalnummer und Saldo) für diesen Erzeugungslauf korrekt wiederhergestellt und fortgeschrieben werden müssen.

## Von Q-up bereitgestellte Funktionen

- Schreiben von Daten in eine Datei
- Lesen von Daten aus einer Datei
- Erzeugung zufälliger und korrekter Datenmengen
- Erzeugung zufälliger Werte
- Erzeugung gezielt fehlerhafter Daten
- Organisation von Schablonen durch Nutzung von Unterverzeichnissen
- Kombination mehrerer Schablonen, um einen Datenblock auf unterschiedliche Weise zusammenzustellen
- Erzeugung unterschiedlicher Datenmengen mit denselben Schablonen durch Nutzung mehrerer Aufträge
- Erzeugung von Datenmengen für mehrere Kassen in mehreren Filialen durch Nutzung von Loopern
- Bedingte Verarbeitung von Anweisungen in Schablonen (Anweisungen werden nur ausgeführt, wenn bestimmte Kriterien erfüllt sind)
- Schleifenverarbeitung in Schablonen (Anweisungen werden in einer Schleife mehrfach ausgeführt)
- Speicherung und Wiederherstellung anschlussrelevanter Daten

## Aufwand

Bei manueller Erzeugung von z. B. 1.600 Journalen und einem optimistischen Durchschnittswert von zehn Minuten pro Journal (es müssen immerhin für jedes Journal bei 100 enthaltenen Buchungen die korrekten Salden und Journalnummern eingetragen werden) benötigt man über 260 Stunden, also mehr als 32 Tage.

Aufgrund der hohen Komplexität der zu erzeugenden Daten benötigt ein geübter Q-up-Nutzer rund drei Stunden. Selbst wenn ein Neuling für die Lösung dieses komplexen Problems mehrere Tage in Anspruch nimmt, ist die Effizienz bei Weitem größer als bei einer manuellen Erzeugung.

Benötigt man diese Daten später wieder mit anderen Journalnummern, fallen bei manueller Erzeugung jedes Mal wieder die Arbeiten für die Anpassung der Journalnummern (und logischerweise auch der Salden) an. Mit Q-up muss man nur das Projekt laden und die Datenerzeugung anstoßen, um einen weiteren Satz aufeinanderfolgender Journale zu erzeugen.

Der Erzeugungsvorgang für die beispielsweise angenommenen 1.600 Journale (diese enthalten insgesamt 163.200 Buchungen) dauert ca. 20 Minuten. In dieser Zeit wurden 172.800 Schablonen verarbeitet.

Dieses Szenario demonstrieren wir ausführlich und mit einem praktischen Beispiel auf unserer Internetpräsenz [www.q-up-data.com](http://www.q-up-data.com).

Sie haben weitere Fragen? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr



## ANWENDUNGSBEISPIELE AUS DER PRAXIS



### Codierungssoftware im Bereich Logistik

Logistikunternehmen mit halbautomatischen Hochregallagern arbeiten inzwischen häufig mit Softwarelösungen, die den Status einer Palette anhand eines Barcodes, der an der Palette angebracht wird, identifiziert. Jede Aktion, die mit dieser Palette ausgeführt wird (aufs Prüfband legen, umpacken, einlagern usw.), wird vom beteiligten System oder vom Personal mithilfe eines (Hand-)Computers in eine Message-Queue übertragen.

Die jeweilige Aktion wird zusammen mit dem eingescannten Barcode der Palette in eine Message-Queue auf einem zentralen Server eingestellt. Dieser Dienst prüft, ob die gewählte Aktion auf die angegebene Palette überhaupt ausgeführt werden darf und leitet dann entsprechende weitere Schritte ein.

Dieser Dienst soll getestet werden. Dazu ist es notwendig, entsprechende Nachrichten, die vom Dienst verarbeitet werden sollen, in die Message-Queue zu schreiben.

#### Fachliche Anforderungen

Als Aktion ist nur eine begrenzte Anzahl an vorgegebenen Texten wie „Prüfen“, „Einlagern“, „Auslagern“ usw. erlaubt. Die Nummer der Palette muss in der Datenbank enthalten sein. Da diese Nummer in der Datenbank als Ganzzahl gespeichert ist, muss sie ggf. mit Nullen aufgefüllt werden.

In der Datenbank befinden sich auch Prozeduren, die z. B. eine neue Palette in die Datenbank einfügen und eine ID dieser Palette zurückliefern.

#### Ziel

Das Ziel ist, dem zu testenden Windows-Dienst Nachrichten zu senden, die jeden erdenklichen Weg (korrekte wie unkorrekte Aktionsfolgen) abdecken, um die Funktion des Dienstes sicherzustellen (C1-Abdeckung).

## Von Q-up bereitgestellte Funktionen

- Schreiben von Daten in eine Message-Queue
- Schreiben von Daten in eine Datenbank
- Formatierung von Daten
- Organisation von Abdeckungstests
- Erzeugung von Datenmengen ganz ohne Schablonen (nur über die Auftragsvariablen)

## Aufwand

Bei einem Test mit konventionell erzeugten Daten, der nur einmal ausgeführt wird, und bei dem z. B. 20 Pfade bedient werden müssen, ist im günstigsten Fall eine Zeitspanne von zehn Stunden durchaus realistisch. Aber für jeden weiteren Test müssen wieder neue Paletten erzeugt und in die entsprechenden Textvorlagen eingetragen werden. Mit Q-up ist der Aufwand für jeden weiteren Test praktisch gleich Null.

Der Aufwand für das Erzeugen der Aufträge, die Definition der notwendigen Variablen im Projekt sowie das Anordnen der Anweisungen in den Auftragsvariablen dauert für einen Q-up-Fachmann etwa 35 Minuten.

Ein Vergleich zur manuellen Erzeugung solcher Testdaten lässt sich zu diesem Beispiel nur sehr schwer anbringen, denn das Schreiben von entsprechenden Nachrichten in verschiedene Textdateien (mit Dummy-Palettennummern) würde genauso lange dauern. Aber ohne Q-up sind das Erzeugen einer neuen Palette und das Eintragen der Textdateien in eine Message-Queue nur mit erheblichem Aufwand möglich.

Dieses Szenario demonstrieren wir ausführlich und mit einem praktischen Beispiel auf unserer Internetpräsenz [www.q-up-data.com](http://www.q-up-data.com).

Sie haben weitere Fragen? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr



## ANWENDUNGSBEISPIELE AUS DER PRAXIS



### **Zeit- und datumsabhängige Daten**

Es gibt Testszenarien, in denen man darauf angewiesen ist, sogenannte Livedaten zu erzeugen. Das heißt, dass bestimmte Datenmengen über einen der Realität entsprechenden Zeitraum erzeugt werden müssen. Das ist z. B. dann der Fall, wenn von einem zu testenden Programm Datums- und Zeitangaben auf bestimmte Relationen zueinander geprüft werden müssen. Oder ein zu testendes Programm muss eingehende Daten in Abhängigkeit des tatsächlichen Zeitpunkts des Dateneingangs auf unterschiedliche Weise verarbeiten.

### **Fachliche Anforderungen**

Die Zeitangaben in Protokollen werden von dem zu testenden Programm berücksichtigt und müssen von Protokoll zu Protokoll plausibel sein. Das bedeutet, die im Protokoll angegebene Dauer darf nicht länger sein als der Zeitraum, der zwischen diesem und dem folgenden Protokoll liegt (wenn ein Protokoll um 8:00 Uhr erzeugt wurde und die Dauer mit sechs Minuten angegeben ist, kann das folgende Protokoll nicht bereits um 8:02 erzeugt worden sein).

### **Ziel**

Für den beschriebenen Test dürfen die zu erzeugenden Protokolle nicht innerhalb kürzester Zeit auf den Server kopiert, sondern sollen zu realen Zeiten erzeugt werden, weil das zu testende Programm unter realen Bedingungen beobachtet werden soll. Das heißt, dass ein Protokoll die tatsächliche aktuelle Uhrzeit enthalten muss und das Folgeprotokoll erst einige Minuten später (aber wegen der Dauer auch nicht zu früh) erzeugt werden darf.

## Von Q-up bereitgestellte Funktionen

- Schreiben von Daten in eine Datei
- Formatierung von Daten
- Simulation unterschiedlicher Arbeitsplätze durch Nutzung von Loopern
- Erzeugung unterschiedlicher Datenmengen mit denselben Schablonen durch Nutzung mehrerer Aufträge
- Erzeugen von Datenmengen, deren Zeitangaben einer realen Situation entsprechen
- Erzeugen von Datenmengen über einen gedehnten Zeitraum.
- Paralleles Erzeugen von Datenmengen durch Nutzung von Threads

## Aufwand

Die Erzeugung der Schablone und der Aufträge zur Generierung von 4.000 Datensätzen nimmt etwa zehn Minuten in Anspruch. Müsste man die z. B. 4.000 Protokolle manuell erzeugen, und es wird ein sehr optimistischer Durchschnittswert von einer Minute pro Protokoll zugrunde gelegt, sind das mehr als 65 Stunden, also mehr als acht Tage.

Das Erzeugen des Auftrags für die Livedaten hat mit Q-up ebenfalls keine Minute in Anspruch genommen. Der Datenerzeugungsvorgang selbst dauert entsprechend der vorgegebenen Zeitdauer.

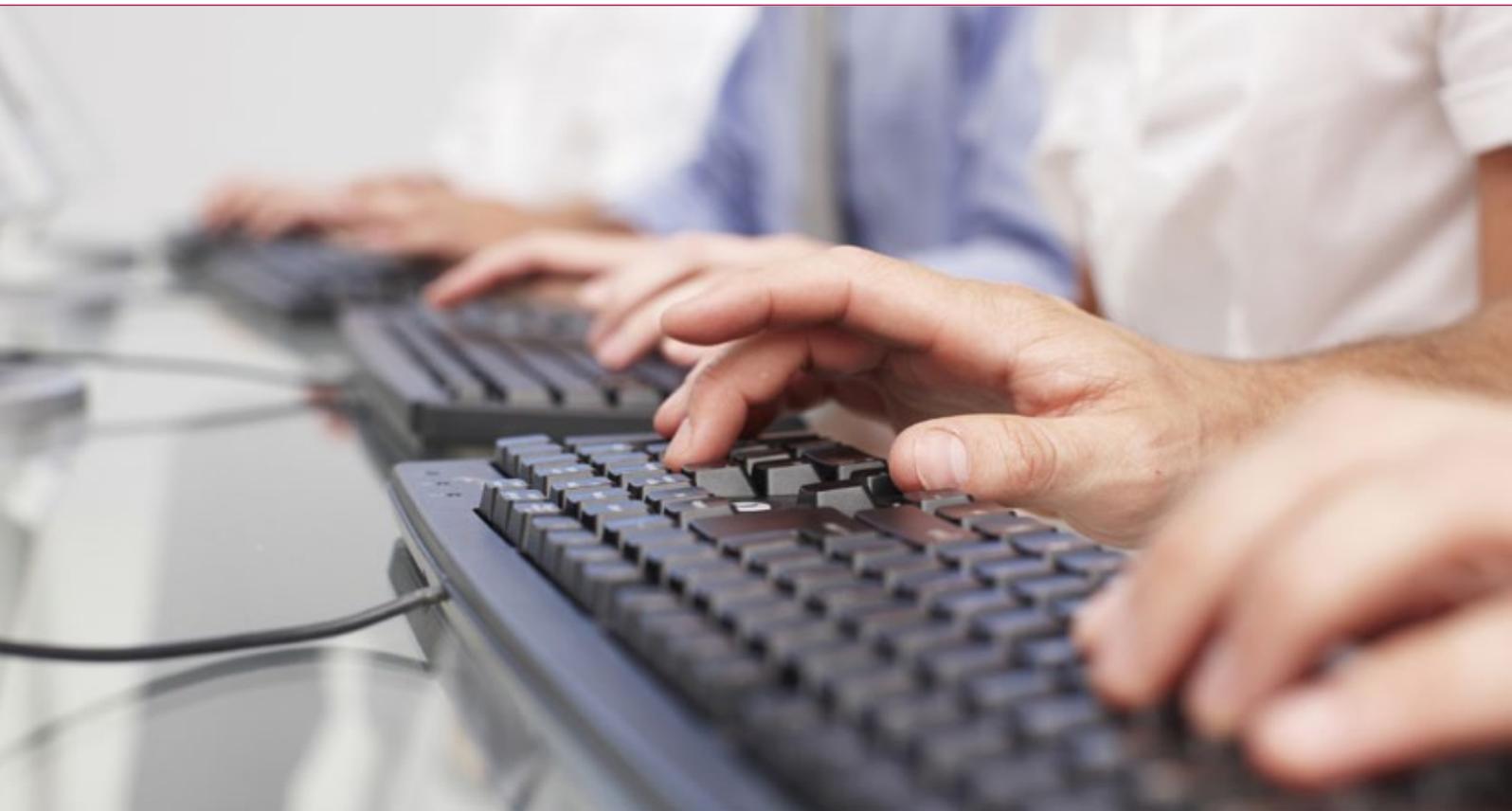
Dieses Szenario demonstrieren wir ausführlich und mit einem praktischen Beispiel auf unserer Internetpräsenz [www.q-up-data.com](http://www.q-up-data.com).

Sie haben weitere Fragen? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr





## EXZELLENTER SERVICE UND SUPPORT. DENN DAS MACHT DEN UNTERSCHIED.

Damit Sie von Anfang an erfolgreich mit Q-up arbeiten können, sind wir jederzeit an Ihrer Seite. Nutzen Sie unsere umfangreichen Service- und Supportleistungen:

- Eingehende telefonische Beratung vor dem Kauf
- Livepräsentation über Webkonferenz
- Hilfestellung und Beratung bei der Installation
- Schulung im Umgang mit der Anwendung
- Schnelle Hilfe durch kompetentes Hotline-Personal
- Kostenlose Patches/Updates innerhalb eines Releases

Sie haben Fragen oder wünschen eine ausführliche Beratung? Nutzen Sie unsere kostenlose Service-Hotline\*:

**0800 7873282**

\*aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr

Q-up ist ein sehr umfangreiches und mächtiges Werkzeug. Um alle Funktionen und Optionen bestmöglich nutzen zu können, bieten wir regelmäßig Schulungen durch unsere qualifizierten Mitarbeiter an. Sie lernen neben der grundlegenden Herangehensweise an die Projektbearbeitung auch Tricks und Kniffe für ein effizientes Arbeiten.

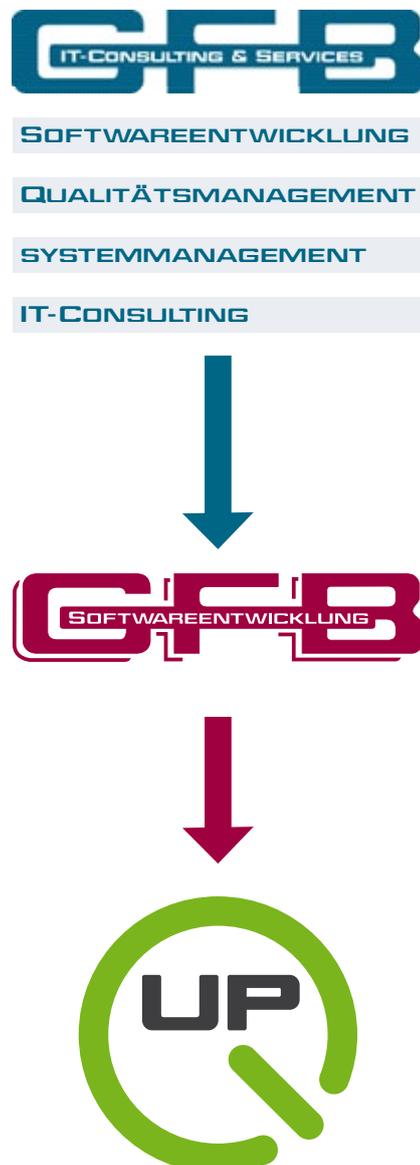
Auf [www.q-up-data.com](http://www.q-up-data.com) können Sie sich im Forum mit Fachleuten austauschen und eigene Entwicklungen für Schnittstellen, Assistenten und Funktionen up- und downloaden.

## VERLASSEN SIE SICH AUF ZEHN JAHRE ERFAHRUNG

Q-up ist ein Produkt der GFB Softwareentwicklungsgesellschaft. Diese ist 100%ige Tochter der GFB EDV Consulting und Services GmbH. Sie wurde gegründet, um die Probleme von Unternehmen noch gezielter durch geeignete Software lösen zu können.

In die Entwicklung des Q-up-Testdatengenerators flossen zehn Jahre Erfahrung aus dem Bereich Qualitätsmanagement mit ein.

Das Ergebnis: Im Vergleich zu manuellen Tests arbeitet Q-up bis zu 90 % effizienter und generiert dabei qualitativ höherwertige Daten.



## TECHNISCHE DETAILS



### Systemvoraussetzungen:

- Windows® XP, Vista, 7, Server 2003, Server 2008
- 32 und 64 bit
- Mindestens 50 MB Festplatte
- Mindestens 512 MB Arbeitsspeicher
- Grafikauflösung mindestens 1024 x 768 Pixel
- .NET Framework 3.5

### Funktionen

- Dateiverarbeitung (Erstellen/Lesen/Schreiben/Löschen)
- Verzeichnisverarbeitung (Erstellen/Löschen)
- Mathematik (Trigonometrie/Logarithmen/ Exponentialrechnung/Runden/Zufallszahlen)
- Datum/Uhrzeit (Ermitteln/Zusammensetzen/Teile extrahieren/Gültigkeitsprüfung)
- Datenkonvertierung (von einem Datentyp in einen anderen, z. B. Zahlen nach Zeichenketten und umgekehrt)
- Kulturinformationen (kulturabhängige Formatierung von Daten)
- Datenbanken (Verbindungsherstellung/Ausführen von SQL-Anweisungen)
- Windows-Message-Queues (Erstellen/Lesen/Schreiben/ Löschen)
- Zeichenketten (Formatierung/Teile einfügen/Teile extrahieren/Teile entfernen/Umwandlung u. v. m.)
- Verschiedenes (Protokollierung/Ablaufverfolgung/ Betriebssystemaufrufe)

### Normen/Standards

- SQL, ODBC, .NET, C#, Windows
- Bedienung an MS-Office und Visual Studio .NET angelehnt

### Schnittstellen

- .NET
- ODBC-fähige Datenbanken

### Programmfunktionen

Erzeugung von Testdaten

- Erzeugen von semantisch korrekten Datenmengen
  - Wertebereiche
  - Wertelisten
  - Relationen
  - Eindeutige (Indizes), fortlaufende und zufällige Werte
- Erzeugen von korrekten Anschlussdaten
  - Testdaten als Anschlussdaten eines vorhergehenden Tests
  - Kein ständiges Leeren und Neubefüllen von Datenbanken oder Message-Queues durch Fortschreiben bestimmter Werte
- Erzeugen von gezielt fehlerhaften Daten
- Schablonen (für die Erzeugung konkreter Daten)
  - Ausdrücke
  - Funktionen
  - Variablen
  - Bedingungen
  - Schleifen
  - Syntaxprüfung, um Schablonenfehler vor der Erzeugung konkreter Daten zu ermitteln
- Aufträge (Kombination von Schablonen)
  - Beliebige Zusammenstellung von Schablonen u. Aufträgen
  - Erzeugen von Daten eines bestimmten oder aller Aufträge

### Bezugsquellen und Support:

GFB Softwareentwicklungsgesellschaft mbH

Obere Zeil 2 • 61440 Oberursel

Tel.: +49 (0) 6171 69410-0

Fax.: +49 (0) 6171 69410-11

Q-up-Hotline\*: 0800 7873282

info@gfb-softwareentwicklung.de

www.gfb-softwareentwicklung.de

\*Kostenlos aus dem deutschen Festnetz, Mo.-Fr. 10-13 Uhr und 14-17 Uhr

Beliebige Quell- und Zielformate

- Datenbanken (über ODBC)
- Message-Queues
- Dateien (XML, SQL-Skripte, Textdateien, CSV usw.)

Protokollierung

- Protokollierung aller Verarbeitungs-/ Erzeugungsschritte
- Frei wählbarer Detailgrad der Protokollierung (von Fehlern bis hin zu Debug-Informationen)

Erweiterbarkeit

- Benutzerdefinierte Funktionen
  - Voller Zugriff auf .NET
  - Erzeugung für Daten in speziellen Zielformaten
  - Lesen von Daten aus speziellen Zielformaten
  - Einbindung von Bibliotheken von Drittanbietern
- Add-Ins
  - Vordefinierte Assistenten zum Erzeugen von XML- und Datenbankdaten
  - Benutzerdefinierte Assistenten zum Erzeugen von spezialisierten Schablonen und Aufträgen

Oberfläche

- Werkzeuggesten (Toolbars)
  - Frei schwebend oder anklebbar
  - Frei wählbare Symbolgrößen (klein, mittel, groß)
- Integrierte Editoren für Schablonen und Funktionen
  - Syntaxhervorhebung
  - Code-Gliederung
  - Änderungsverfolgung
  - Zeilennummerierung
  - Code-Vervollständigung (IntelliSense)
- MDI (Multi-Dokument-Schnittstelle)
  - Mehrere Schablonen/Funktionen parallel bearbeiten
  - Fenster wahlweise als konventionelle MDI-Fenster oder als Tabulatoren darstellbar
- Ziehen und Ablegen (Drag and Drop)
  - Zusammenstellung von Aufträgen per Ziehen und Ablegen
- Frei schwebende Werkzeugfenster
  - Fehlerliste
  - Ausgabe (Protokollierung)
  - Projektbaum
  - Beliebige positionierbar (frei schwebend oder angeklebt)
  - Automatisch in den Hintergrund tretend
- Kontextsensitive Hilfe
  - Aufrufbare Onlinehilfe zu jedem Fenster (F1)
  - Kurzhinweise (Tooltips) für Befehle, Dialogelemente und Fenster
- Kontextsensitive Fehlerbeschreibungen
  - Per Doppelklick im Fehlerfenster zur Fehlerursache springen