

20./21. Okt. 2004  
Wien

# Awareness-Event

Bei freiem  
Eintritt!

## Zielgruppe

- ▶ Mittleres Management aus Unternehmen der IT-Industrie
- ▶ Führungskräfte von Software-Entwicklungsabteilungen der IT Industrie
- ▶ QA-Mitarbeiter aus all diesen Unternehmen
- ▶ Testverantwortliche

# Zyklus „Qualitätssicherung in der Software-Entwicklung“

## Risikomanagement und agiles Programmieren

Mittwoch, 20. Oktober 2004, 15.30 – 18.30 Uhr

## Qualitätssicherung und Performance Testen

Donnerstag, 21. Oktober 2004, 8.30 – 13.00 Uhr

## Testen von Embedded Systems

Donnerstag, 21. Oktober 2004, 13.30 – 18.00 Uhr

IBM, 1020 Wien, Obere Donaustraße 95

**Referenten:** Kurt Aigner (Compuware), Martin Alt (IBM Software Group), Johannes Bergmann (software quality lab), Martin Fritz (OBJECTIS), Thomas Gruber (ARC), Martin Heilig (Beko), Peter Hruschka (Atlantic System Guild), Anastasios Kyriakopoulos (SQS), Christian Munser (Triton), Gerd Nanz (SQS), Mike Nemeth (Triton), Erich Prem (eutema), Erwin Schoitsch (ARC Seibersdorf)

**Moderation:** Erwin Schoitsch (ARC Seibersdorf), Alexander Wiesmayr (Future Network)

**Podiumsdiskussion:** „Die Bedeutung der Qualitätssicherung für die österreichische Software-Entwicklung und Wirtschaft“

Weiterführende  
Fachseminare  
im Heftinneren

In Kooperation mit:



COMPUWARE



OBJECTIS



SOFTWARE GUILD SYSTEMS

WORKING FOR YOUR SUCCESS  
TRITON

eu|te|ma



COMPUTERWELT

Unsere Partner:

AUSTRIAN RESEARCH CENTERS

presstext.austria

CONECT  
CONSULTING  
EVENTS  
COMMUNICATIONS  
TRAINING

# Qualitätssicherung in der Software-Entwicklung

**Ausgehend von den etablierten Methoden (V-Modell, ISO 12207) für die Software-Entwicklung und der IT-Infrastructure Library (ITIL) für den Betrieb von IT-Lösungen liegen die Schwerpunkte dieser Informationsveranstaltung in der Betriebsphase. In diesem Abschnitt des Lebenszyklus einer Software entsteht der Bedarf an Änderungen und Erweiterungen in der Funktionalität sowie der Fehlerbehebung. Diese Anforderungen aus geplanten Changes und deren Verifikation stellen sowohl die Fachabteilung als auch die Entwicklungsabteilung vor immer gleiche Anforderungen: Wie kann ich die durchgeführten Changes verifizieren und diese anschließend geordnet in den Betrieb übernehmen?**

**Es werden Methoden für effizientes Testen der Changes vorgestellt und durch Best-Practice-Beispiele ergänzt. Besonderes Augenmerk liegt auf dem Kosten-/Nutzenaspekt internen und externen Testens und dem Einsatz geeigneter Werkzeuge.**

Je länger eine geplante Projektlaufzeit, desto sicherer kann man von Projektverzögerungen und angespannten Budgets ausgehen – spätestens gegen Ende der Laufzeit eines Projektes. Dabei wird der Zeit- und Kostenaufwand für das Testen und die Fehlerbehebung im eigentlichen Code immer wieder unterschätzt; bei gewissenhafter Vorgehensweise können sich in dieser Projektphase weitere Projektverzögerungen ergeben. Es gilt daher in jeder Phase eines Projektes, innerhalb des geplanten Zeitrahmens zu bleiben oder – besser noch – Zeit gutzumachen, also geplante Projektzeiten nach Möglichkeit zu unterschreiten und damit allenfalls in früh(ere)n Projektphasen bereits entstandene Verzögerungen aufzuholen. Insbesondere gilt dies für die Codierungs- und Testphase: die bis dahin kumulierten Verzögerungen führen dazu, dass das Testen oftmals zu kurz kommt.

Klar strukturierte organisatorische Abläufe liefern einen wesentlichen Baustein zur Qualitätssicherung von Projekten, sei es auf dem Mainframe oder in einer Client-Server-Umgebung. Insbesondere die den gesamten Lebenszyklus eines Projektes/einer Software begleitenden Tools für Versionskontrolle, Change Management, Build-Unterstützung und Testen sichern einen geordneten Projektverlauf und Klarheit über den Projektfortschritt.

Eine gesicherte Code-Qualität (der Code entspricht den Konventionen und die Funktionalität deckt sich mit den Requirements) zu liefern, zeugt von der Professionalität und Leistungsfähigkeit eines Unternehmens. Diese Professionalität sicherzustellen, ist ein internes Problem der Entwicklungsabteilung und muss intern gelöst werden – denn mangelhafte Qualität kann nicht auf den Auftraggeber abgewälzt werden – der ja zu Recht den Anspruch erhebt, gute Basisqualität zu erhalten, auch wenn vielleicht seine Requirements nicht optimal spezifiziert sind und sich über die Projektlaufzeit dynamisch ändern. Diese interne Sicherstellung der Qualität des Codes kann völlig unabhängig von den Requirements erfolgen und bedarf keiner Auseinandersetzung mit dem Auftraggeber – ganz im Gegensatz zum Abnahmetest.

Jede Unterstützung im Design und Coding kann helfen, den nachfolgenden Testaufwand zu reduzieren bzw. banale Fehler in den Programmen auszumerzen – automatische Code-Review (Code-Inspection) hilft, triviale Fehler zu vermeiden und einen Minimalstandard des Codes sicherzustellen. Selbst beim Einsatz von Codegeneratoren ist nicht immer sichergestellt, dass sich Oberflächlichkeiten und damit triviale Fehler einstellen, die bei der Umgehung der Tools entstanden sind.

Neben der laufenden Code-Review (Code-Inspection) kann in der Phase des eigentlichen Testens der Funktionalität die Effizienz durch den Einsatz von Tools gefördert werden. Sogenannte Testtools zur Testautomatisierung reduzieren die Testdurchlaufzeiten und beschleunigen den Testfortschritt beträchtlich. Für die Methodik ist es dabei unerheblich, ob dieses Testen durch eigene Mitarbeiter abgewickelt wird, oder ob dieser Bereich der Entwicklung von einem externen Dienstleister zugekauft wird.

## RISIKOMANAGEMENT

**Risikomanagement in IT-Projekten – Damit IT-Management nicht zum permanenten „Katastrophenmanagement“ entartet!**

Risikomanagement ist das Gegenteil von Katastrophenmanagement und, im weiteren Sinne, von schlechtem Management.

In kaum einer Branche ist der Fall, dass ein Projekt dramatisch schief läuft, Termine und Kosten überschritten werden, und dennoch die Beteiligten von übertriebenem Optimismus nur so strotzen, bis es zu spät ist, so häufig wie in der IT-Softwarebranche. Weitläufig bekannt sein dürften die verschiedenen Statistiken großer staatlicher US-Auftraggeber, wonach nur ein geringer Teil der Großprojekte erfolgreich ist (DoD, USA: 30% misslingen komplett, 70% der restlichen Projekte überschreiten ihr Budget um 189% oder liegen im Termin 222% zurück, d. h. nur 21% der Projekte sind in Ordnung). Die Ursachen sind vielseitig, besonders kritisch sind unvollständige Anforderungsspezifikationen (Benutzer, System, Umfeld) und fehlendes Risikomanagement. Der Beitrag gibt einen Überblick über die grundsätzlichen Besonderheiten und Probleme des Software-Projektmanagements („Mythos Mannmonat“) und die wesentlichen Aspekte des Risikomanagements, welche an einfachen Beispielen und einer „Checkliste“ zur Rolle des Riskmanagers als Komplementär des Projektmanagers verdeutlicht wird.

## Risiko Management und IT-Security

Ziel des Risikomanagements ist es, potenzielle Probleme zu identifizieren, bevor sie eingetreten sind, und nach Bedarf Korrekturmaßnahmen zu planen und umzusetzen.

Durch die verstärkte Nutzung webbasierter Geschäftsprozesse und den Einsatz mobiler Lösungen steigen die Sicherheitsanforderungen der Unternehmen. Die daraus entstehenden IT-Sicherheitsrisiken können unter Umständen sehr schnell zu Geschäftsrisiken werden und erhebliche finanzielle

Schäden und Wettbewerbsnachteile nach sich ziehen. Diese Geschäftsrisiken müssen erkannt, bewertet und kontrolliert werden, damit die richtigen Maßnahmen rechtzeitig eingeleitet werden können. IT-Security ist ein essentieller Bestandteil von vertrauenswürdigen Geschäftsprozessen.

BEKO Engineering und Informatik unterstützt Sie in der Identifizierung von Geschäftsrisiken und hilft Ihnen eine ganzheitliche und Risiko optimierte IT-Sicherheitsstrategie zu entwickeln und umzusetzen

#### Risiko Management

- ▶ Prozessrisiken und -kontrollen
- ▶ Authentifizierung, SSO, Biometrik
- ▶ Berechtigungsverwaltung und -administration
- ▶ Datenschutz und Datensicherheit

#### IT-Security

- ▶ Technische Sicherheitsrisiken
- ▶ Netzwerk- und Kommunikationssicherheit
- ▶ Verzeichnisdienste (LDAP usw.)
- ▶ PKI-Infrastruktur
- ▶ Interface-Security

### Agile Methoden – Risikogetriebene Vorgehensmodelle

Seit Software-Engineering Ende der 70er-Jahre zum Thema wurde haben wir mehr und mehr über Entwicklungsprozesse und Methoden gelernt. Unsere Vorgehensmodelle sind ständig gewachsen – solange bis ein normaler Entwickler die Flut der Informationen nicht mehr beherrschen kann. Mit eXtreme Programming entstand dann ein radikal anderer Ansatz zur Systementwicklung, der im Frühjahr 2002 von einer Gruppe führenden Methodenentwickler zu „agilen Methoden“ relativiert wurden. Das Agile Manifest war geboren. Es führt uns zur Rückbesinnung auf längst bekannte Grundwerte für erfolgreiche Softwareentwicklung.

In diesem Vortrag lernen Sie dieses Wertesystem und die Maximen agiler Methoden kennen. Anhand einiger Vertreter wie XP, SCRUM, ARTE oder Crystal werden die neuen Herangehensweisen demonstriert. Eine Klärung der Voraussetzungen für den erfolgreichen Einsatz rundet den Vortrag ab.

## PERFORMANCE MANAGEMENT

### Die Bedeutung der Anforderungsspezifikation für die Qualität von Software-Systemen

Die Grundsteine für die Qualität und Effizienz in der Umsetzung und im Betrieb von Software-Systemen werden ganz wesentlich

durch die Qualität in der Spezifikation der entsprechenden Software-Systeme gelegt.

Oft wird hier aus Kostengründen gespart oder es werden nur funktionale Teile spezifiziert und andere wesentliche Bereiche einer Spezifikation werden nicht berücksichtigt. Fehler und Mängel in der Spezifikationsphase sind meist die teuersten Fehler!

Im Vortrag werden die erwähnten Problembereiche und Themen aufgegriffen und es wird auf die frühen Phasen von Software-Entwicklungs- oder -Beschaffungs-Projekten eingegangen. Der inhaltliche Teil des Vortrags wird begleitet durch praktische Beispiele aus der Tätigkeit des Vortragenden als Projektleiter und Berater sowie Spezifikations-Beispiele aus der inhaltlichen Analyse von zahlreichen Pflichtenheften aus öffentlichen Ausschreibungen der letzten Jahre.

### Maximale Qualität durch Methode – Risk Based Testing

#### Europaweite Studie bestätigt: Mangelnde SW-Qualität verursacht hohe Kosten

64 Prozent der Teilnehmer an einer Umfrage von Compuware verzichten darauf, das Ausfallrisiko bei der Einführung einer neuen Software-Anwendung zu quantifizieren. Andererseits sagen 74 Prozent, dass die Kosten aufgrund schlechter Softwarequalität für ihr Unternehmen bis zu 500.000 Euro im Jahr betragen. Bei Unternehmen mit mehr als 5.000 Mitarbeitern schätzen 15 Prozent die Kosten sogar auf mehr als 1 Million Euro im Jahr. Compuware befragte 358 IT-Manager in großen europäischen Unternehmen.

#### Risikobasiertes Testen als Lösung

Mehr als 55 Prozent der befragten IT-Manager gaben an, dass das Management ihnen beim Test einer Anwendung nicht mitteilt, welche Teile geschäftskritisch für das Unternehmen sind. Daher können die Test-Teams keine Strategie für das Testen der riskanteren Elemente der Anwendung entwickeln. Dies zeigt sich daran, dass 45 Prozent der Befragten sagen, sie versuchen so viel wie möglich von der Anwendung zu testen, um eine hundertprozentige Verlässlichkeit zu erreichen.

#### Die Methode wird zur Strategie

Risikobasiertes Testen erlaubt es der IT-Abteilung, das Ausfallrisiko einer Anwendung und die Auswirkung eines solchen Ausfalls auf das Unternehmen exakt einzuschätzen. Diese Einschätzung kann an das höhere Management weitergegeben werden. Mit den Informationen des Test-Teams kann das Management eine fundierte Entscheidung treffen, ob eine Anwendung zu einem bestimmten Zeitpunkt in den Produktivbetrieb gehen soll.

### Case Study: Test Management in BASEL II

Mit BASEL II hat der Basler Bankenausschuss der Finanzwelt ein Regelwerk gegeben, das weitreichende Auswirkungen auf die Kreditverzinsung sowie Eigenkapitaldeckung der Finanzinstitute hat. Bei großen Finanzdienstleistern sind Projekte dieser Größenordnung ohne umsichtiges Test Management kaum oder nur schwer zu bewältigen. Dabei treten neben allgemein zu berücksichtigenden Punkten wie Testorganisation, Auswahl der Testwerkzeuge, Bereitstellung der Testumgebung vermehrt Fragen nach Wirtschaftlichkeit, Testeffektivität, Kommunikation und Rollenverhalten in den Vordergrund. Begleitet durch technische Anforderungen nach guter Performance, Ausfallsicherheit u.a. entsteht in Summe eine spezielle Herausforderung an das gesamte Projekt- und Testteam.

Mit der Case Study „Test Management in BASEL II“ soll diesen Fragen und Themen Raum gegeben werden. Anhand von aktuellen Beispielen aus der jüngsten Projektpraxis werden die spezielle Ausgangslage analysiert, der methodische Testansatz näher beleuchtet sowie allgemeine Schlussfolgerungen für effektives Test Management gezogen.

OBJENTIS Software Integration GmbH wurde 1999 gegründet und ist mit seinen strategischen Geschäftsfeldern im Bereich der angewandten Qualitätssicherung und deren essentiellen Schwerpunkt Software-Test erfolgreich im deutschsprachigen Raum positioniert. In den letzten Jahren wurden von OBJENTIS Testmethodiken entwickelt, die von einer eigenen Tool-Suite QAS *Quality Assurance Studio* prozessbegleitend unterstützt werden.

### User warten nicht – Praxisbeispiel: Einsatz Open Source Last- & Performance Testtool

Zuverlässigkeit und Stabilität sind ein zentral wichtiger Aspekt der erlebten Qualität moderner Softwaresysteme.

Lange Wartezeiten und überlastete Systeme demotivieren nicht nur Mitarbeiter sondern verärgern Kunden, welche auf Auskunft und schneller Bearbeitung ihrer Anliegen vertrauen.

Triton skizziert in diesem Grundsatzreferat zunächst die Einsatzbereiche und Zeitpunkte, zu welchen Sie Last- & Performance-Tests durchführen müssen. In groben Zügen erfahren Sie von Experten, wie Last- & Performancetests aufgesetzt und die Ergebnisse aus der Durchführung interpretiert werden.

Im Anschluss zu diesem eher theoretischen Teil werden die Aussagen anhand

eines Beispiels aus der Praxis näher erläutert und vertieft.

Erfahren Sie, wie mit einem Open Source Tool ein hochwertiger Last- & Performance-Test aufgesetzt werden konnte! Triton wird die Leistungsfähigkeit des genannten Testwerkzeugs darstellen und den Funktionalitäten kostenpflichtiger Werkzeuge gegenüberstellen. Dieser Vortrag wird Ihnen die Möglichkeit eröffnen, auch für Ihre Anwendungsbereiche abzuschätzen, inwieweit ein Open Source Werkzeug Ihre Bedürfnisse abdecken kann!

## TESTEN VON EMBEDDED SYSTEMS

### Viel Qualität auf kleinem Raum – Testen von Embedded Software

Schon heute finden sich 90% aller Prozessoren in eingebetteten Systemen (Embedded Systems), und ihr Anteil wird weiter steigen. Dies gilt auch oder gerade für sicherheitskritische Systeme wie Flugzeug, Eisenbahn oder Auto. Was aber die methodischen Software-Tests anbelangt, trägt das Qualitätsbewusstsein diesem Trend erst allmählich Rechnung. Zum einen stehen die Programmierer von Embedded Software vielfach in der spezifischen Tradition der Elektronik-Hardware-Entwickler, zum anderen schaffen auch die Ressourcen-begrenzten Microcontroller und die vielfach proprietären Compiler Probleme bei der Verwendung professioneller Testwerkzeuge. Zunehmend fordern aber Kunden und Behörden auch von eingebetteten Systemen die Zertifizierung nach Sicherheitsnormen.

Der Vortrag gibt zunächst einen Überblick über die von Systemsicherheitsnormen geforderten Software-Qualitätskriterien und schildert, durch welche Maßnahmen diese erfüllt werden. Danach wird die spezielle Problematik im Bereich der Embedded Systems angerissen und erläutert, wieso hier ein systematischer Test und damit eine Zertifizierung der Software vielfach schwieriger als am „normalen“ Computer ist. Schließlich werden verschiedene konventionelle Lösungsmöglichkeiten und auch neuere Ansätze aufgezeigt, wie man die Problematik der begrenzten Ressourcen und der proprietären Compiler beherrschbar und damit die Software zertifizierbar machen kann.

### Automatisierung der Tests für Software ohne grafische Oberfläche – Motivation und Lösungsansätze

Testen von Software kann – vor allem wenn es keine grafische Benutzeroberfläche gibt – ein frustrierender und mühseliger Vorgang

sein. Gerade aber im Bereich von dynamischen Tests lassen sich Vorgänge automatisieren. Dieser Beitrag beschäftigt sich mit den Möglichkeiten der Automatisierung von Software Tests für Applikationen und Embedded Echtzeit-Anwendungen.

Unter dynamischen Tests wird die Ausführung eines Programms mit bestimmten Eingangsdaten und dem anschließenden Soll-Ist-Vergleich der Ergebnisse verstanden. Zunächst wird im Beitrag erörtert, welche Testobjekte zu betrachten sind: Funktionen, Klassen, Echtzeitbetriebssystem Tasks, ein komplettes Embedded Target, vernetzte Systeme usw. Je nach Testobjekt finden sich verschiedene Arten der Testrealisierung, die im einzelnen vorgestellt werden. Dann wird zugeordnet, welche typischen Testarten (funktionale Tests, Robustness-, Performance und Lasttests) in welchen Testphasen realisiert werden. Ausgehend von dieser Darstellung manueller Testverfahren, werden automatisierte Testverfahren diskutiert. Testspezifikationen, Testskript-Sprachen spielen hierbei eine zentrale Rolle. Vorteile der Automatisierung bezüglich Wiederverwendbarkeit (Regressionstests), Eindeutigkeit und Nachvollziehbarkeit werden diskutiert. Auch die Anforderungen an Tests auf Embedded Targets werden hierbei berücksichtigt.

Industrielle Erfahrungen mit Test Automatisierung (aus den Bereichen Automotive, Telekommunikation und Aerospace) am Beispiel des Werkzeugs IBM Rational Test RealTime™ runden den Beitrag ab.

### Qualitätssicherung für Embedded Systems – Kostenersparnis, Zuverlässigkeit und Sicherheit durch Softwaretests und effiziente Prozesse

Software wird vermehrt in Systemen eingesetzt, bei denen der Benutzer oft gar nicht wahrnimmt, dass sie rechnergesteuert sind. Da die Softwarekomplexität solcher Embedded Systems immer größer wird, steigt damit auch die Notwendigkeit, Tests und weitere qualitätssichernde Maßnahmen zu intensivieren. Hinzu kommt, dass viele der betroffenen Anwendungen in sicherheitsrelevanten Bereichen eingesetzt werden und damit in den Zuständigkeitsbereich einschlägiger Normen fallen, die erweiterte Qualitätssicherungs-Aktivitäten und definierte Prozesse verlangen.

SQAS unterstützt diese Unternehmen den gestiegenen Anforderungen gerecht zu werden und sowohl die Sicherheit, als auch die Zuverlässigkeit ihrer Systeme zu verbessern.

Die SQS berät und unterstützt die Hersteller dabei, die notwendigen Verfahren und Methoden, die Planungssicherheit und Kostenkontrolle für ihr Projekt, sowie ihre

bestehenden Testprozesse stets zu verbessern. Hierzu gehören unter anderem folgende Prozesse:

- ▶ Spezifikationsreview
- ▶ Designreview
- ▶ Codereview
- ▶ Statischer/dynamischer Test
- ▶ Modultest
- ▶ Integrationstest
- ▶ Systemtest
- ▶ Echtzeittest
- ▶ Installationstest
- ▶ Abnahmetest
- ▶ Zulassungstest

Der erfolgreiche Weg muss nicht zwingend einhergehen mit dem kompletten Neuaufbau einer Teststrategie.

Oft führen kleine Veränderungen im Testprozess, wie z.B. das Definieren und Einhalten von Testendkriterien, zu messbaren Fortschritten.

In der SQS haben bereits viele Hersteller – maßgeblich aus den Branchen Railway, Avionics und Automotive – einen kompetenten und zuverlässigen Ansprechpartner für ihre Qualitätssicherungs- und -managementaufgaben gefunden.

## QUALITÄTSSICHERUNG VON SOFTWARE IM BEREICH AUTOMOTIVE

### Wenn die Software streikt: Auto-Rückrufe auf Rekordniveau – Software-Qualitätssicherung noch keine eigene Disziplin

In letzter Zeit häufen sich diese für die Hersteller und auch Zulieferer sehr kostenintensiven Maßnahmen zur Fehlerbehebung. Beispiele sind nicht auslösende Airbags, falsche Einspritzsequenzen, Probleme mit dem Anti-Blockiersystem und Fahrzeuge, die liegen bleiben, weil die Steuerungs-Software zu spät meldet, dass der Tank leer ist. Alle diese Vorfälle sind auf Fehler in der Software des jeweiligen Systems zurückzuführen. Die Frage ist: was kann man trotz des bestehenden Kostendrucks unternehmen, damit es nicht soweit kommt und dabei sogar noch Kosten im Entwicklungsprozess einsparen?

Die Qualitätssicherung der Produktionsprozesse in der Automobilindustrie gilt allgemein als Vorreiter für andere Branchen. Ganz anders sieht es im Bereich der Softwareentwicklung aus. Hier sind die Prozesse oft noch nicht ausgereift, obwohl der Anteil an Software für die Funktionen eines Fahrzeugs drastisch zunimmt.

Die Lösung liegt in der konsequenten Einführung einer frühzeitigen und prozessbegleitenden Software-Qualitätssicherung.

Hierzu bietet die SQS AG als europäischer Marktführer im Bereich Software-Qualitätssicherung in enger Zusammenarbeit mit den Automobilherstellern und Zulieferern die folgenden Dienstleistungen an:

- ▶ Durchführung von Assessments und Prozessverbesserungen nach SPICE/ISO 15504
- ▶ Test- und Risikomanagement
- ▶ Testkonzeption und -durchführung
- ▶ Statische Analyse und Code-Review
- ▶ Unit-Test, Verifikation und Validierung
- ▶ Testautomatisierung und Toolintegration
- ▶ Software Change and Configuration Management
- ▶ Aufbau von QS-Know-how im Projekt/ Unternehmen

unter besonderer Berücksichtigung von

- ▶ MISRA, ISO 61508
- ▶ Anforderungen der Hersteller Initiative Software (HIS)
- ▶ AUTO-TMap®/TPI®

Mit den erzielten Ergebnissen halten Ihre Entwicklungsprozesse jedem Assessment stand:

- ▶ Vollständige Requirements
- ▶ Steigerung der Entwicklungssicherheit
- ▶ Höhere Effizienz der Entwicklungsprozesse
- ▶ Redundanzfreie Testfälle
- ▶ Maximale Testabdeckung durch Äquivalenzklassenanalyse
- ▶ Leichte Wartbarkeit der Testfälle
- ▶ Testfälle sind für Simulations- und Target-Tests nutzbar
- ▶ Höhere Sicherheit bei der Entwicklung von Steuergeräten
- ▶ Standardisierte QS-Prozesse
- ▶ Sichere Integration verschiedener Steuergeräte

### **FIT-IT Förderungen für Embedded Systems F&E**

Das Thema Embedded Systems wurde in Österreich und in ganz Europa als wichtiger Bereich für staatliche Förderinitiativen erkannt. Auch Österreich fördert Embedded-Systems-Forschung und -Entwicklung mit dem Programm „FIT-IT Embedded Systems“. Diese Initiative des BMVIT hat zum Ziel, neue Technologien auf dem dynamischen Gebiet eingebetteter Systeme zu fördern. Im Vortrag des Programm-Managers Dr. Prem werden die Ziele und Instrumente, aber auch die internationale Einbettung von FIT-IT präsentiert. Zusätzlich soll ein Ausblick auf andere österreichische und internationale Fördermöglichkeiten für Embedded Systems gegeben werden.

Informationen unter: [www.fit-it.at](http://www.fit-it.at)

# AGENDA

20. Oktober 2004

## **RISIKOMANAGEMENT**

15.30 **Registration**

16.00 **Risikomanagement in IT-Projekten – Damit IT-Management nicht zum permanenten „Katastrophenmanagement“ entartet!**  
Erwin Schoitsch (ARC Seibersdorf)

16.45 **Risiko Management und IT-Security**  
Mario Heilig (Beko)

17.30 **Agile Methoden – Risikogetriebene Vorgehensmodelle**  
Peter Hruschka (Atlantic Systems Guild)

18.30 **Ende der Veranstaltung**

21. Oktober 2004

## **PERFORMANCE TESTEN**

8.30 **Registration**

9.00 **Eröffnung**

9.15 **Die Bedeutung der Anforderungsspezifikation für die Qualität von Software-Systemen**  
Johannes Bergsmann (Software Quality Lab)

9.45 **Maximale Qualität durch Methode – Risk Based Testing**  
Kurt Aigner (Compuware)

10.30 **Pause**

11.00 **Case Study: Test Management in BASEL II**  
Martin Fritz (OBJENTIS)

11.40 **User warten nicht – Praxisbeispiel: Einsatz Open Source Last- & Performance Testtool**  
Christian Munser, Mike Nemeth (Triton Unternehmensberatung GmbH)

12.20 **Performance Testen Erfolgsfaktoren und Hindernisse**  
Anastasios Kyriakopoulos (SQS)

12.50 **Schlussdiskussion: „Die Bedeutung der Qualitätssicherung für die österreichische Software-Entwicklung und Wirtschaft“**

13.15 **Mittagspause und Registration**

## **TESTEN VON EMBEDDED SYSTEMS**

14.00 **Viel Qualität auf kleinem Raum – Testen von Embedded Software**  
Thomas Gruber (ARCS Seibersdorf)

15.00 **Automatisierung der Tests für Software ohne grafische Oberfläche – Motivation und Lösungsansätze**  
Martin Alt (IBM Software Group)

15.30 **Qualitätssicherung für Embedded Systems – Kostenersparnis, Zuverlässigkeit und Sicherheit durch Softwaretests und effiziente Prozesse**  
Gerd Nanz (SQS)

16.00 **Pause**

16.30 **Qualitätssicherung von Software im Bereich Automotive – Anwenderbericht**

17.00 **FIT-IT Förderungen für Embedded Systems F&E**  
Erich Prem (eutema Technologie Management)

18.00 **ca. Ende der Veranstaltung**

# TERMIN & ORT

20. Oktober 2004, 15.30–18.30 Uhr

21. Oktober 2004, 8.30–18.00 Uhr

IBM, 1020 Wien, Obere Donaustraße 25

# GEBÜHREN

Bei freiem Eintritt!

Unkostenbeitrag für die Tagungsunterlagen:

€ 60,- zuzügl. 20% MWSt.

# ZIELGRUPPE

- ▶ Mittleres Management aus Unternehmen der IT-Industrie,
- ▶ Führungskräfte von Software-Entwicklungsabteilungen der IT Industrie
- ▶ QA- Mitarbeiter aus all diesen Unternehmen
- ▶ Testverantwortliche

# Workshop Automatisiertes Softwaretesten

Im Rahmen des Workshops wird gemeinsam mit den Teilnehmern ein schlankes automatisiertes Testportfolio an Hand einer Internet-Anwendung erstellt. Die Teilnehmer lernen die Potenziale und die Probleme in der Erstellung und im Betrieb automatisierter Testportfolios kennen.

Der inhaltliche Bogen spannt sich vom Aufzeichnen von einzelnen Testscripts bis zur Erstellung von vollwertigen automatisierten Testfällen, die über das Konzept der Dynamischen Scriptgenerierung zur Ausführung gebracht werden. Die typischen Problemsituationen im automatisierten Test (Steuerungsproblem, Wartungsproblem, etc.) werden diskutiert und beispielhaft gelöst.

Best Practices zur Erstellung effizienter Testfall-Kataloge und zur kostengünstigen Übernahme in die Automatisierung komplettieren das Themenspektrum, sodass der Gesamtprozess der Testautomatisierung im Rahmen des Workshops abgedeckt wird.

Die Teilnehmer sind nach dem Besuch des Workshops in der Lage, die verschiedenen technischen Ansätze zur Testautomatisierung im Hinblick auf ihre individuellen Anforderungen zu bewerten und die günstigste Lösung auszuwählen. Die Teilnehmer können automatisierte Tests für einfache Applikationen autark aufbauen und verfügen über ein breites Basiswissen, das in der Folge auch die Errichtung und den Betrieb großer automatisierter Testportfolios erlaubt.

## Voraussetzungen für die Teilnahme:

Die Teilnehmer müssen über Basiskenntnisse in der Programmierung (Visual Basic o.ä.) verfügen sowie über geübte Anwenderkenntnisse in Microsoft Office. Theoretische Kenntnisse zum Thema Software-Test sind von Vorteil, aber nicht zwingende Voraussetzung.

## AGENDA

1. Tag Testen und Testautomatisierung  
Know-how-Transfer (Theorie)  
Ansatz Capture/Replay  
Scriptsprachen und Bibliotheksaufbau  
Wartung von Scripts  
Dynamische Scriptgenerierung
2. Tag Testfall-Verwaltung im TOSCA Explorer  
Object Maps  
Scriptgenerator und fachliche Steuerung  
Komposite Control  
Erweiterung Testfall  
Dynamische Testautomatisierung  
Erstellung fachlicher Testfälle  
Verbindung mit dem TOSCA Explorer  
Zusammenfassung

## REFERENTEN

Dipl.-Ing. Wolfgang Platz, Georg Thurner  
(Triton Unternehmensberatung)

## TERMIN & ORT

3./4. November 2004  
it-versity Education Center der Ziegler & Partner GmbH  
1070 Wien, Schottenfeldgasse 69, 3. Stock

## GEBÜHREN

€ 990,- zuzügl. 20 % MWSt. für Mitglieder des  
Future Network  
€ 1.190,- zuzügl. 20 % MWSt. für Nichtmitglieder

# Vertiefungsseminare zum Thema Testen – Methoden und Verfahren

Die weiterführende Ausbildung zum „ASQF-Certified Tester, Advanced Level“ vertieft einzelne Themen und bietet dazu verschiedene aufgabenspezifische Module mit hohem Praxisanteil an, die zur intensiven und professionellen Durchführung von Testaufgaben in Projekten befähigen.

## Referenten:

Anastasios Kyriakopoulos (SQS)  
und weitere

Bei Buchung über Future Network  
(Anmeldung auf der Rückseite des  
Programms) 3 % Ermäßigung!

## ASQF®-Certified Tester, Advanced Level: Funktionale Testtechniken

7.10. – 8.10. 04, Wien

Vorstellung und Übung systematischer und nicht-systematischer Testtechniken für den funktionalen Systemtest basierend auf den klassischen Black-Box-Methoden wie z.B. der Äquivalenzklassenanalyse, Ursache-Wirkungsgraph-Analyse, Grenzwertanalyse sowie der Geschäftsprozessanalyse.

### Teilnahmegebühr:

€ 950,- zuzügl. 20% MWSt.  
zuzügl. Prüfungsgebühr: € 170,- zuzügl.  
20% MWSt.

## ASQF®-Certified Tester, Advanced Level: Testmanagement

12.10. – 15.10. 2004, Wien

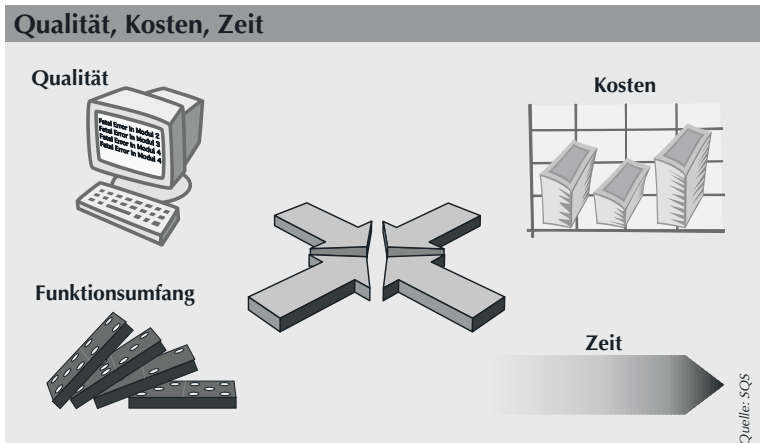
Vorstellung der wesentlichen Bereiche und Aspekte des Testmanagements mit Diskussion und praktischen Übungen: Testkonzepterstellung, Testplanung, Testüberwachung und Testfortschrittskontrolle, Risikomanagement, Fehlermanagement, Teamorganisation, Testprozessverbesserung

### Teilnahmegebühr:

€ 1.700,- zuzügl. 20% MWSt.  
zuzügl. Prüfungsgebühr: € 170,- zuzügl.  
20% MWSt.

# ISTQB® - Certified Tester, Foundation Level

- ▶ Grundlagen des Software-Testens
- ▶ Testen während des Software-Lebenszyklus:  
Testmodelle
- ▶ Dynamischer Test
- ▶ Statischer Test
- ▶ Testmanagement
- ▶ Werkzeuge zur Testunterstützung



Leitung: Anastasios Kyriakopoulos (SQS)

## Was ist der ISTQB®-Certified-Tester, Foundation Level?

Das Austrian Testing Board arbeitet mit dem ISTQB® (International Software Testing Qualifications Board) zusammen. Erarbeitet hat dieses Angebot eine Arbeitsgruppe, in der im SW-Testumfeld erfahrene Softwarefirmen und Softwarespezialisten vertreten sind (u. a. SQS). Trainingsinhalt und Qualifikation wurden kompatibel zu bestehenden europäischen Standards formuliert. Damit wird in Österreich ein Aus- und Weiterbildungsschema für SW-Tester nach international anerkannten Standards angeboten.

## Zielgruppe:

Personen mit dem Aufgabenschwerpunkt Software-Testen und Software-Qualitätsmanagement,

die grundlegende Fähigkeiten und Kenntnisse auf diesem Gebiet erwerben und dies durch ein national und international anerkanntes Zertifikat nachweisen wollen.

## Was ist das Austrian Testing Board?

Das ATB ist die österreichische Vereinigung des International Software Qualifications Board und gewährleistet den einheitlichen Standard von ISTQB-konformen Tester-Kursen in Österreich.

## Was kann der Teilnehmer nach Besuch des Kurses?

Nach dem Kurs kennt der Teilnehmer die Grundprinzipien des Software-Prüfens und Software-Testens. Er kann den generellen Testprozess anwenden und hat Aktivitäten und

## AGENDA

Grundlagen des SW-Testens  
Testen während des Lebenszyklus  
Dynamischer Test  
Statischer Test  
Testmanagement  
Testunterstützungswerkzeuge

## LEITUNG

Anastasios Kyriakopoulos (SQS)

## TERMIN & ORT

18.–20. Oktober 2004

CON.ECT Eventcenter, Kaiserstraße 14/2, 1070 Wien

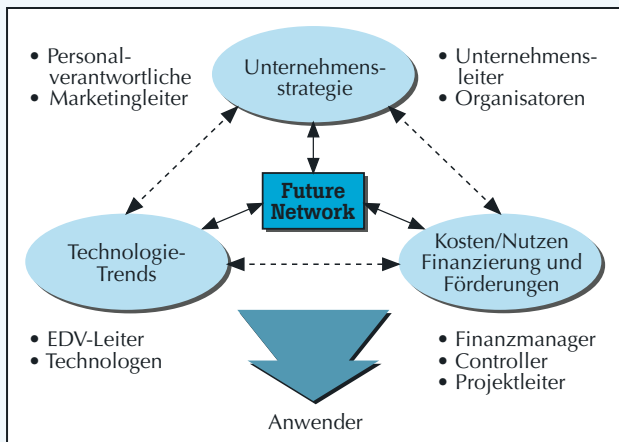
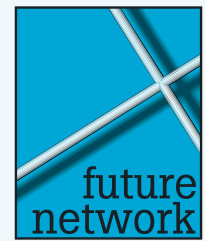
## GEBÜHREN

€ 1.350,- zuzügl. 20 % MWSt für Mitglieder des Future Network

€ 1.450,- zuzügl. 20 % MWSt für Nichtmitglieder

€ 250,- zuzügl. 20 % MWSt als Prüfungsgebühr

Weitere Informationen und Anmeldung auch unter <http://www.future-network.at/>



## Vorstandsmitglieder des Future Network:

- Michael Vesely (Consultant)
- Dipl.-Ing. Wolfgang Apfelbaum (Apfelbaum Business Consulting)
- Dipl.-Ing. Dr. Franz Barachini (BIC AUSTRIA)
- Vstd.-Dir. Ing. Bernhard Graf (Basler Versicherung)
- Ing. Johann Ehm (OMV)
- Ing. Mag. Heinz Janecska (Bundesrechenzentrum)
- Rudolf Mrstik (AUA)
- Dr. Rupert Nagler (Information Design Institute)
- Prof. Helmut Schauer (Universität Zürich)
- Johannes Werner (Consultant)
- Sonja Haberl – Finanzreferentin
- Mag. Bettina Hainschink – Generalsekretärin

**ANMELDUNG:** Nach Erhalt Ihrer Anmeldung senden wir Ihnen eine Anmeldebestätigung. Diese Anmeldebestätigung ist für eine Teilnahme am Event erforderlich. Das Future Network behält sich vor, Mitglieder vorzuziehen.

**STORNIERUNG:** Bei Stornierung der Anmeldung zu einem Fachseminar bis zum Anmeldeschluss fällt eine Stornogebühr in der Höhe von 10% der Teilnahmegebühr an. Bei Abmeldung nach diesem Termin wird die gesamte Gebühr fällig. Wenn Sie einen Ersatzteilnehmer melden, entfällt natürlich die Stornogebühr.

**ADRESSÄNDERUNGEN:** Wenn Sie das Unternehmen wechseln oder wenn wir Personen anschreiben, die nicht mehr in Ihrem Unternehmen tätig sind, teilen Sie uns diese Änderungen bitte mit. Nur so können wir Sie gezielt über unser Veranstaltungsprogramm informieren.

- ▶ Ich melde mich zum **Awareness Event „Qualitätssicherung in der Softwareentwicklung“** am 20./21. Oktober 2004 kostenfrei an
  - am Mittwoch, 20. 10. 2004, 15.30–18.30 Uhr
  - am Donnerstag, 21. 10. 2004, 8.30–13.00 Uhr
  - am Donnerstag, 21. 10. 2004, 13.30–18.00 Uhr
- Ich hätte gerne die Tagungsunterlagen zum Preis von € 60,- + 20% MWSt.

▶ **Anmeldeschluss: 18. 10. 2004**

- Bitte nehmen Sie mich in den E-Mail-Verteiler auf – Veranstaltungsprogramme als  PDF,  Textfile

### Workshops:

- ▶ Ich melde mich zum Workshop „ISTQB®-Certified Tester, Foundation Level“ am 18.–20. Okt. 2004 an:
  - als Mitglied des Future Network zum Preis von € 1.350,- + 20% MWSt.
  - als Nichtmitglied zu € 1.450,- + 20% MWSt.
  - Ich lege auch die Prüfung ab: Prüfungsgebühr € 250,- + 20% MWSt.
- ▶ Ich melde mich zum Workshop „Automatisiertes Softwaretesten“ am 3./4. Nov. 2004 an
  - als Mitglied des Future Network zum Preis von € 990,- + 20% MWSt.
  - als Nichtmitglied zu € 1.190,- + 20% MWSt.
- Ich melde mich zum Workshop „ASQF®-Certified Tester, Advanced Level: Funktionale Testtechniken“ am 7.–8. 10. 2004 zum Preis von € 950,- + 20% MWSt. an (zuzügl. Prüfungsgebühr von € 170,- + 20% MWSt.) und erhalte 3% Ermäßigung.
- Ich melde mich zum Workshop „ASQF®-Certified Tester, Advanced Level: Testmanagement“ am 12.–15. 10. 2004 zum Preis von € 1.700,- + 20% MWSt. an (zuzügl. Prüfungsgebühr von € 170,- + 20% MWSt.) und erhalte 3% Ermäßigung.
- ▶ **Anmeldeschluss** jeweils 3 Tage vor Workshopbeginn

An  
Future Network  
Kaiserstraße 14/2  
1070 Wien

Tel.: +43/1/522 36 36-37  
Fax: +43/1/522 36 36-10  
E-Mail: office@future-network.at  
http://www.future-network.at

Firma:	
Titel:	Vorname:
Nachname:	
Funktion:	
Straße, Hausnummer:	
PLZ:	Ort:
Telefon:	Telefax:
E-Mail:	
Ort, Datum:	
Unterschrift, Firmenstempel:	

- ▶ Ich erkläre mich mit der elektronischen Verwaltung meiner ausgefüllten Daten und der Nennung meines Namens im Teilnehmerverzeichnis einverstanden.
  - ▶ Ich bin mit der Zusendung von Veranstaltungsinformationen per E-Mail einverstanden.
- (Nichtzutreffendes bitte streichen)

**Begrenzte Teilnehmerzahl!**