

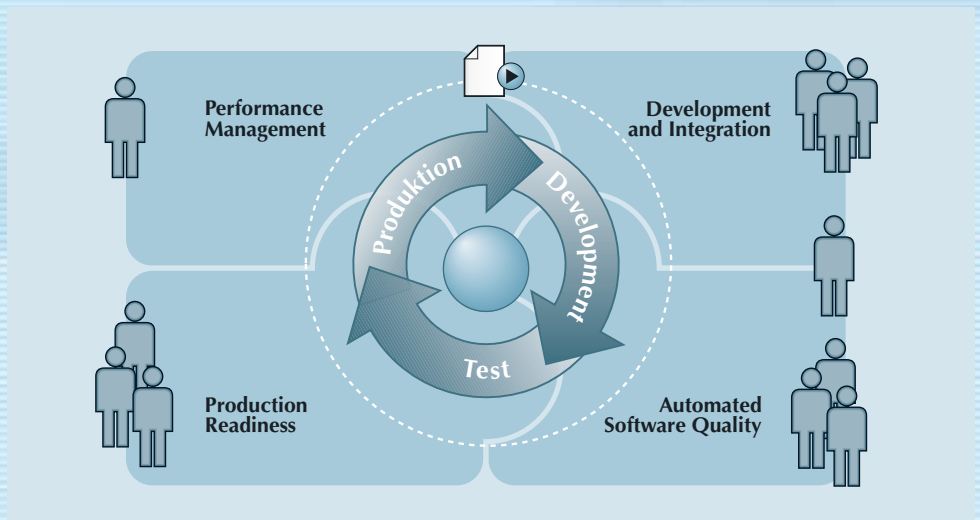
Qualitätssicherung und Testmethoden in der Produktivphase

Zyklus „Qualitätssicherung in der Software-Entwicklung“

Dienstag
23. März 2004
8.30 – 14.30 Uhr
Wien

- ▶ Automatisierte Testmethoden
 - ▶ Funktionales Testen
 - ▶ Last & Performance Tests
- ▶ Usability Tests
- ▶ Methoden (Projektmanagement etc.)
- ▶ Risikomanagement
- ▶ Best Practices
- ▶ Kostenkontrolle
- ▶ Zukunftstrends

Awareness-Event



Referenten: Christian Drescher (Compuware), Jutta Feigelbinder (Beko), Reinhard Haberfellner (QiDO Software Quality Services), Oliver Kuklok (SQS), Werner Novak (IBM), Erwin Schoitsch (ARC Seibersdorf), Georg Thurner (Triton)

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

Bei freiem
Eintritt!

In Kooperation mit:



Unsere Partner:



Qualitätssicherung und Testmethoden in der Produktivphase

Zyklus „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 gut zu machen, also geplante Projektzeiten nach Möglichkeit zu unterschreiten und damit allenfalls in früh(er)en 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 wesent-

lichen 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 sicher zu stellen, 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 nachfolgende 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 sicher gestellt, dass sich Oberflächlichkeiten und damit triviale Fehler einstellen, die bei der Umgehung der Tools entstanden.

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. So genannte 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.

Management von Testumgebungen

Ein erfolgreicher Softwaretest benötigt eine eigene, angemessene Testumgebung. Dabei ergeben die unterschiedlichen Anforderungen der operativen Testprozesse unterschiedlich aufgebaute Testumgebungen. Aus dem Entwicklertest heraus sind die Anforderungen an Testumgebungen völlig konträr zum Funktionstest oder Ergonomietest. Der Integrationstest oder Geschäftsprozessstest hat durch die weit gefasste Aufgabenstellung wiederum andere Anforderungen. Wie ist es möglich diese unterschiedlichen Testumgebungen je Testprozess und je Testdurchführung bereitzustellen? Wie findet man die „ideale“ Testumgebung zum durchzuführenden Testprozess? Dabei sind die Fragen des Fehlermanagements genauso zu betrachten, wie die Sicherstellung der richtigen Konfiguration im Test. Denn ein Projekt kann nur dann erfolgreich sein, wenn es nicht nur in „Budget and Time“ sondern auch in „Quality“ abschließen kann. Dabei ist das Wissen über das WAS (Testfälle), WO (Testumgebung), WIE (Testprozesse) und in WELCHER

Eventkonzept:

Bettina Hainschink
(Future Network)

Konfiguration die Hauptaufgabe des Managements von Testumgebungen im Softwaretest.

Die logische Entwicklung: SW-Qualität → IT-Performance → Business-Value – Mit einem Praxisbeispiel einer neutralen Internetplattform

Die Präsentation wird neue, technologie-unterstützte Vorgehensweisen – so genannte „Best Practices“ – zur Risikominimierung und Performancemaximierung bei Software-Implementierungs- und -Entwicklungsprojekten bei gleichzeitigen Kosteneinsparungen aufzeigen. End-to-End-Applikationperformance – Testen, Analyse, Tuning und Monitoring unter Berücksichtigung der gesamten Infrastruktur (Client, Network, Server, Datenbanken) sowie der Software-Anwendungen selbst – wirkt sich unmittelbar auf die Business Performance aus. – IT bringt Business.

Anhand eines Beispiels aus der Praxis (Web-Applikation, betrieben in heterogener Umgebung; LAN und WAN, Datenbank: Oracle ...) wird ausgehend von einem nicht zufrieden stellenden Lasttestergebnis mit der Suche nach der oder den Ursache(n) begonnen. Dies erfolgt nach einem methodischen Vorgehen. Die Möglichkeit der Korrelation von Messdaten aus der Infrastruktur wie auch aus der Applikation und der Datenbank (End-to-End), führt rasch ans Ziel.

Wie in der Praxis, ist zu sehen wie eine Fehlerursache weitere Fehler vorerst „verdeckt“. Es wird auch deutlich wie unterschiedlich sich „Performance-Killer“ aus Entwicklungssicht bzw. aus Produktionssicht darstellen und auswirken.

Ebenfalls wird gezeigt wie, lange vor dem Lasttest, Standortbestimmungen hinsichtlich des Performaneverhaltens der Applikation in der späteren „Produktionsumgebung“ möglich sind. Damit wird proaktiv und messbar ein laufendes „Performance-Profiling“ parallel zum Entwicklungsprozess im Rahmen der Software-Qualitätssicherung möglich. Negative Überraschungen im Rahmen eines Lasttests vor dem „going live“ können damit drastisch reduziert werden.

Die dabei verwendete Software-Technologie ist dabei live zu sehen.

Entlang des „roten Fadens“ den das Praxisbeispiel vorgibt, wird ein spannender Ablauf gezeigt mit dem sich IT-Profis schnell identifizieren werden.

Abgerundet wird die Präsentation mit einem Praxisbericht aus einem realen Performanceprojekt eines Betreibers einer neutralen Internetplattform für die österreichische Versicherungsbranche.

Die Lösung des Steuerungsproblems in der Testautomatisierung

Viele Applikationen beinhalten komplexe Controls (Tabellen, Grids, TreeViews, ...) als zentrale Steuerelemente. Diese komplexen Controls sind für die Abwicklung von Bearbeitungsvorgängen oft von entscheidender Bedeutung. Vorbedingung für eine effiziente Testautomatisierung ist eine leistungsfähige technische Steuerung insbesondere für diese komplexen Controltypen.

Die Triton Unternehmensberatung ist Experte für die Realisierung komplexer technischer Steuerungen. Im Laufe praktischer Umsetzungsprojekte im Testbereich hat Triton umfassendes Know-how im Windows- und Java-Umfeld aufgebaut. Im Rahmen seiner Projekte trifft Triton immer wieder auf Testautomatisierungsprojekte, welche an der geschilderten Problematik einer unzureichenden Controlsteuerung zu scheitern drohen.

Anhand eines konkreten Kundenprojektes wird die von Triton entwickelte Vorgehensweise zur Erstellung leistungsfähiger Controlsteuerungen Schritt für Schritt vorgestellt. Dadurch wird das Publikum in die Lage versetzt, die entscheidenden technischen Risiken einer Testautomatisierung qualifiziert abschätzen zu können.

Warum Testautomation oft scheitert?

Durch immer engere Terminvorgaben, weniger Ressourcen und der Anforderung generell die Entwicklungskosten zu senken, wird häufig das Schlagwort „Testautomation“ als Lösung genannt. Das führt dazu, dass knapp vor dem Beginn des Testens ein Testautomationstool angeschafft wird. Dadurch soll Personalaufwand reduziert, das „lästige“

Testen verkürzt werden – insgesamt steht im Raum „Der Testaufwand wird nicht so bedeutend sein, da ein automatisches Werkzeug verwendet wird“.

An Hand eines Praxisbeispiels wird mittels der BEKO-Dienstleistung Testmentoring aufgezeigt, welche Voraussetzungen für die Einführung von Testautomation nötig sind.

Der Einsatz von Testautomation wird methodisch betrachtet, um Testautomation erfolgreich zu implementieren.

Erfolgreiche Qualitätssteigerung durch frühzeitige Code-analyse"

Immer kürzere Releasezyklen bei steigenden Qualitätsanforderungen, verlangen nach effektiven und effizienten Tests der zu entwickelnden oder sich im Einsatz befindlichen Software-Applikation. Je früher im Entwicklungszyklus Tests durchgeführt und Bugs entdeckt werden, desto schneller kann darauf reagiert werden. Idealerweise werden daher Tests bereits durch den Entwickler durchgeführt und Bugs behoben.

Um den Entwickler hierbei zu unterstützen bietet IBM die „IBM Rational PurifyPlus“-Familie an. Dieser Werkzeugkasten, bestehend aus PureCoverage (Codeabdeckung), Quantify (Profiling) und Purify (Memory Check), bietet die Möglichkeit der Unterstützung bereits bei der automatischen Laufzeitanalyse.

Damit ist der Entwickler in der Lage die Codeabdeckung bei Testläufen zu messen, Performance-Profiling und das Erkennen von Speicherfehlern durchzuführen. All das funktioniert auch dann, wenn nur der Objektcode und nicht der Sourcecode zur Verfügung steht. In diesem Vortrag sehen Sie, wie man schwer erkennbare Bugs auffindet, Performance-Bottlenecks erkennt, Codeabdeckung von Testläufen evaluiert

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.

Workshop 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)

- TERMIN & ORT**
- 1. Termin: 17.-19. März 2004
 - 2. Termin: 21.-23. Juli 2004

Anmeldeschluss: jeweils 10 Tage vor Veranstaltungsbeginn
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
 - € 200,- zuzügl. 20% MWSt als Prüfungsgebühr

AGENDA

8.30 Registration

9.00 Management von Testumgebungen
Oliver Kuklok (SQS)

9.40 Die logische Entwicklung: SW Qualität – IT Performance – Business Value mit Praxisbericht
Christian Drescher (Compuware)

11.00 Die Lösung des Steuerungsproblems in der Testautomatisierung
Georg Thurner (Triton)

11.30 Pause

12.15 Warum Testautomation oft scheitert
Jutta Feigelbinder (Beko)

12.45 Erfolgreiche Qualitätssteigerung durch frühzeitige Codeanalyse
Werner Novak (IBM)

13.15 Risk Management in IT-Projekten
Erwin Schoitsch (ARC Seibersdorf)

14.00 6 gute Argumente gegen Code Reviews
Reinhard Haberfellner (QiDO Software Quality Services)

14.30 ca. Ende der Veranstaltung

Änderungen vorbehalten

TERMIN & ORT

Dienstag, 23. März 2004, 8.30 – 14.30 Uhr
IBM, 1020 Wien, Lassallestraße 1

GEBÜHREN

Bei freiem Eintritt!
Unkostenbeitrag für die Tagungsunterlagen:
€ 60,- zuzügl. 20% MWSt.

ZIELGRUPPE

- ▶ Mittleres Management von Unternehmen der IT-Industrie,
- ▶ Führungskräfte von Software-Entwicklungsabteilungen der IT Industrie
- ▶ Führungskräfte aus Unternehmen mit hoher IT-Nutzung und eigener IT-Entwicklung
- ▶ QA-Mitarbeiter aus all diesen Unternehmen
- ▶ Testverantwortliche

Future Network Event-Vorschau

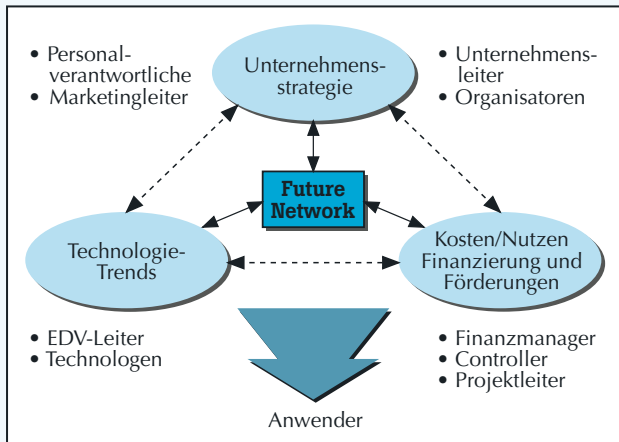
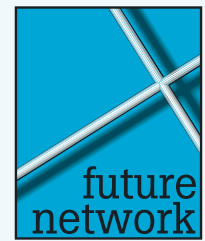
Kostenfreie Awareness Events

- 10. März 04: Geschäftsprozesse neu gestalten, managen und optimieren**
Geschäftsprozessmanagement, Qualitätsmanagement von Geschäftsprozessen, Prozessorientierte Organisationsgestaltung und Projektmanagement, Best Practices, Einführung von Prozessketten über mehrere Unternehmen
- 23. März 04: Qualitätssicherung und Testmethoden in der Produktivphase**
Automatisierte Testmethoden, Usability Tests, Best Practices, Kostenkontrolle, Zukunftstrends

Kostenpflichtige Workshops und Strategieseminare:

- 10./11. Februar 04: ITIL Foundation Training mit Zertifizierungsprüfung**
Referentin: Ullrike Buhl (FCS Consulting & Training)
ITIL ermöglicht die geschäftsprozessorientierte, kostenoptimierte, benutzerfreundliche Erbringung und Überwachung der IT-Services und hilft bei der Neuordnung des IT-Geschäftsbereiches
- 2./3. März 04: Controlling der IT-Kosten**
Referent: Harald Schnetzer (team4e.com technology & management consulting GmbH)
Elemente des IT-Controlling, IT-Programme identifizieren und qualifizieren, Methoden des Investitionscontrollings, das IT-Projektcontrolling-Cockpit, transparente Leistungsverrechnung, IT-Kennzahlen, IT-Leistungen und IT-Kosten transparent machen, Wertbeitrag der IT zum Gesamterfolg erkennbar machen
- 9.–12. März 04: Optimale Supportprozesse nach ITIL- ITSM 1**
Referenten: Ullrike Buhl (FCS Consulting & Training),
Markus Lindinger (Masters consulting)
- 17.–19. März 04: ISTQB-Certified Tester, Foundation Level**
Referent: Anastasios Kyriakopoulos (SQS)
Grundlagen des Software-Testens
Testen während des Software-Lebenszyklus: Testmodelle, dynamischer Test, statischer Test, Testmanagement, Werkzeuge zur Testunterstützung
- 18./19. März 04: Strategisches IT-Management, „Erstellung einer IT-Strategie“ für Geschäftsführer und IT Manager**
Referenten: Christoph F. Strnadl (Atos Origin, Management Consultant),
Wolfgang Muffat (Consultant, Trainer und Coach)
Begriffserklärung – Vision Strategie, Mission, Goals, Objectives, Ziele, Wert der IT, Geschäftsstrategie und Prozesse Business Driver & IT Driver, IT-Analyse, IT-Projekte und IT-Strategie
- 25./26. März 04: Geschäftsprozessorganisation**
Referent: Holger Kahl (Consultant)
Anstöße zur Geschäftsprozessorganisation, Entwicklung der Organisationsformen: Von der funktionalen Organisation zur Prozessorganisation, aktuelle Managementkonzepte

Nähere Informationen unter: http://www.future-network.at/events_1.asp



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** (Arthur D. Little)
- Rudolf Mrstik** (AUA)
- Dr. Rupert Nagler** (Information Design Institute)
- Prof. Helmut Schauer** (Universität Zürich)
- Johannes Werner** (Consultant)
- Sonja Haberl** – Finanzreferentin
- Mag. Bettina Hainschink** – Generalsekretärin

Unsere Partner:



Institut für Informatik der Universität Zürich

AUSTRIAN RESEARCH CENTERS
presstext.austria

Web powered by:



Weitere Future Network Events finden Sie unter <http://www.future-network.at>

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: Falls Sie nach erfolgter Anmeldung doch nicht am Event teilnehmen können, bitten wir Sie, uns unbedingt rechtzeitig Bescheid zu geben, damit wir Ihren Platz an einen anderen Interessenten weitergeben können.

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 und Testmethoden in der Produktivphase“ am 23. März 2004 kostenfrei an
- Ich hätte gerne die Tagungsunterlagen zum Preis von € 60,- + 20% MWSt.
- **Anmeldeschluss: 22. März 2004**
- Bitte nehmen Sie mich in den E-Mail-Verteiler auf –
Veranstaltungsprogramme als PDF, Textfile

Angebot für Nicht-Mitglieder:

- Zugleich mit der Bestellung der Tagungsunterlagen können Sie eine kostenfreie Einzelmitgliedschaft für das Jahr 2004 erhalten, mit der Sie unter anderem auf über 150 Papers im Future Network Webarchiv zugreifen können.
- Bitte fordern Sie den Aufnahmeantrag an!
- Ich bin bereits Future Network Mitglied

- 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)

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: