



SIMPL GmbH, Universitätsstraße 38, 70569 Stuttgart
Katharina Görlach,
Peter Reimann
Universitätsstraße 38
70569 Stuttgart

Wolfgang Hüttig
Projektleiter SIMPL
Universitätsstraße 38
70569 Stuttgart

Angebot

17. Juli 2009

Sehr geehrte Frau Görlach, sehr geehrter Herr Reimann,

vielen Dank für Ihr Interesse an einem Angebot unserer Firma. Gerne würden wir ihre Vorstellungen im Rahmen eines Entwicklungsprojekts umsetzen.

Sie finden hierzu auf den folgenden Seiten unser ausführliches Angebot. Über eine Zusammenarbeit in diesem Projekt würden wir uns sehr freuen.

Bei Rückfragen können Sie sich selbstverständlich gerne jederzeit mit uns in Verbindung setzen.

Mit freundlichen Grüßen

Wolfgang Hüttig

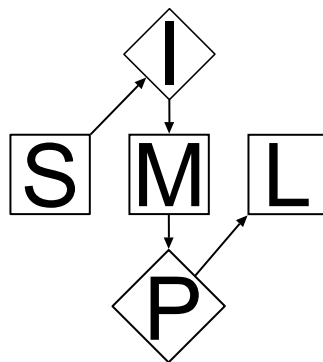
Angebot

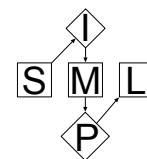
Version 1.0

17. Juli 2009

Verfasser:

Daniel Brüderle, Firas Zoabi, Michael Hahn



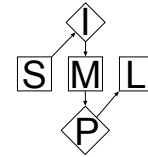


Inhaltsverzeichnis

1	Ausgangssituation	5
1.1	Externe Beteiligte	5
1.1.1	Kunden	5
1.1.2	Betreuer	6
1.2	Projektbeschreibung	6
1.3	Entwicklungsgrundlagen	6
2	Projektaufgabe	7
3	Unser Vorgehen	7
3.1	Interne Projektteam-Organisation	7
3.1.1	Rollen im Team	7
3.1.2	Kontaktdaten der Projektteam-Mitglieder	9
4	Nutzen	9
5	Leistungen im Rahmen des Angebots	10
5.1	Auszuliefernde Software-Einheiten	10
5.2	Software-Einheiten, die nicht zum Lieferumfang gehören	10
6	Terminplanung	11
6.1	Beschreibung der Phasen	12
6.1.1	Analyse	12
6.1.2	Einarbeitung	12
6.1.3	Spezifikation	13
6.1.4	Entwurf	13
6.1.5	Codierung	13
6.1.6	Test	13
6.1.7	Abnahmetest	13
6.1.8	Puffer	14



7	Unsere Konditionen	14
7.1	Risiken	14
7.2	Kostenschätzung	15
7.3	AGBs	16



1 Ausgangssituation

Zur Modellierung und Ausführung von Workflows jeglicher Art wird bisher die Sprache BPEL (Business Process Execution Language) verwendet. BPEL hat seine Stärke, wie der Name schon sagt, im Bereich der Geschäftsprozesse. Im wissenschaftlichen Bereich gelten andere Anforderungen und dafür ist BPEL nicht gut geeignet. Da aber auch im wissenschaftlichen Bereich immer öfter Workflows eingesetzt werden, sei es bei der Durchführung von komplexen Berechnungen im Rahmen von Simulationen oder bei der Ausführung von Experimenten, bedarf es einer Anpassung von BPEL an die wissenschaftlichen Bedürfnisse. Gekennzeichnet sind wissenschaftliche Workflows in der Regel durch große Datenmengen, die Heterogenität der Daten und somit dem Bedarf nach großer Rechenkapazität. Diesen Anforderungen soll durch die neu zu entwickelnde Software Rechnung getragen werden.

1.1 Externe Beteiligte

An dieser Stelle werden die externen Projektbeteiligten genannt. Dies sind die Kunden und die Betreuer vom Institut für Architektur von Anwendungssystemen (IAAS) und dem Institut für Parallele und Verteilte Systeme (IPVS).

1.1.1 Kunden

Dipl.-Inf. Katharina Görlach

Institut für Architektur von Anwendungssystemen
Universitätsstraße 38, 70569 Stuttgart (Zimmer 1.328)
Telefon: +49 (0)711 7816-333
Fax: +49 (0)711 7816-472
E-Mail: katharina.goerlach@iaas.uni-stuttgart.de

Dipl.-Inf. Peter Reimann

Institut für Parallele und Verteilte Systeme
Universitätsstraße 38, 70569 Stuttgart (Zimmer 2.467)
Telefon: +49 (0)711 7816-445
Fax: +49 (0)711 7816-424
E-Mail: Peter.Reimann@ipvs.uni-stuttgart.de



1.1.2 Betreuer

Dipl.-Math. Michael Reiter

Institut für Architektur von Anwendungssystemen

Universitätsstraße 38, 70569 Stuttgart (Zimmer 1.037)

Telefon: +49 (0)711 7816-416

Fax: +49 (0)711 7816-472

E-Mail: michael.reiter(@)iaas.uni-stuttgart.de

Dipl.-Inf. Marko Vrhovnik

Institut für Parallele und Verteilte Systeme

Universitätsstraße 38, 70569 Stuttgart (Zimmer 2.467)

Telefon: +49 (0)711 7816-242

Fax: +49 (0)711 7816-424

E-Mail: Marko.Vrhovnik(@)ipvs.uni-stuttgart.de

1.2 Projektbeschreibung

Das Projekt wird als Time & Material Projekt durchgeführt und die Aufwände und Fortschritte des Projekts werden in regelmäßigen Abständen dem Kunden mitgeteilt. Dies erfolgt spätestens nach dem Abschluss einer Phase. Bei wichtigen Phasen auch innerhalb der Phasen, wenn es für sinnvoll erachtet wird.

1.3 Entwicklungsgrundlagen

Die Entwicklung von SIMPL erfolgt nach erprobten softwaretechnischen Prinzipien. Dadurch wird eine hohe Qualität des Produkts gewährleistet. Wichtige Qualitäten sind dabei die Wartbarkeit und die Erweiterbarkeit des Systems, da es über einen längeren Zeitraum bestehen soll und in der Zukunft um die Anbindung weiterer Datenquellen, Konzepte für den Datenzugriff und den Umgang mit weiteren Datenformaten ergänzt werden soll. Ein weitere wichtige Qualität ist die Skalierbarkeit des Systems, die eine sehr flexible Infrastruktur erlauben muss, da die Computersysteme auf denen SIMPL



später ausgeführt wird, in ihrer Leistung sehr weit auseinander gehen, d.h. vom normalen Desktop-Computer bis zum Supercomputer kann und soll alles möglich sein.

Robustheit und Usability der Software sind für uns innerhalb des Entwicklungsprozesses von großem Interesse. Die Usability sollte sich vor allem an unerfahrene und Nutzer mit wenig Kenntnissen richten und dafür die bestmögliche Transparenz liefern, um die Verwendung von SIMPL für alle Benutzergruppen zu ermöglichen.

2 Projektaufgabe

Das Entwicklungsteam soll ein erweiterbares, generisches Rahmenwerk erstellen, welches den Zugriff auf nahezu beliebige Datenquellen ermöglichen soll. Bei den Datenquellen kann es sich beispielsweise um Sensornetze, Datenbanken und Dateisysteme handeln. Der Schwerpunkt soll klar auf wissenschaftlichen Workflows beruhen. Über das Rahmenwerk sollen beliebige Datenmanagement-Funktionen in einen BPEL-Prozess eingebunden werden können. Dafür werden bereits vorhandene Konzepte evaluiert und für unsere Zwecke erweitert oder angepasst. Dabei wird auch die Sprache BPEL für unsere Zwecke erweitert. Für eine möglichst hohe Flexibilität soll ein dynamischer Ansatz gewählt werden, so dass während der Laufzeit des Systems die Datenquellen festgelegt werden können. Weiterhin sollte auch die Möglichkeit bestehen die Datenquellen statisch anbinden zu können. Der Kunde besteht darauf, dass eine BPEL-Engine sowie ein Modellierungstool um diese gewünschten Funktionen erweitert bzw. angepasst werden. Die BPEL-Prozesse sollen mit dem entsprechenden Modellierungstool modelliert und mit der BPEL-Engine ausgeführt werden können.

3 Unser Vorgehen

3.1 Interne Projektteam-Organisation

3.1.1 Rollen im Team

Projektleiter: Wolfgang Hüttig (Stellvertreter: Michael Hahn)

Der Projektleiter ist der zentrale Ansprechpartner für den Kunden. Er koordiniert sämtlichen Schriftverkehr und die Kommunikation zwischen den Kunden, den Betreuern und dem Team. Er ist auch an erster Stelle für das Projekt verantwortlich.

Seine zentralen Aufgaben liegen in der Verwaltung des Projekts, d.h. er plant, bewertet und kontrolliert, und verteilt alle Tätigkeiten und Ereignisse während des Projekts.



Architekt: Michael Schneidt (Stellvertreter: Daniel Brüderle)

Der Architekt ist zuständig für die Architektur des zu entwickelnden Softwaresystems und die Beschreibung der grundlegenden Komponenten und deren Zusammenspiel innerhalb des Softwaresystems. Er trifft die grundlegenden Entscheidungen über die Art und den Aufbau der Software-Architektur und überwacht deren Einhaltung während der Codierung. Der Architekt gewährleistet damit eine einfache Integration der einzelnen Software-Komponenten und ermöglicht die einfache Wartbarkeit und Erweiterbarkeit des Softwaresystems.

Administrator: René Rehn (Stellvertreter: Michael Schneidt)

Der Administrator ist zuständig für die gesamte Infrastruktur auf Software- sowie auch auf Hardware-Ebene für die Projektarbeit. Er ist für den Betrieb und die Wartung des Servers und den Betrieb der darauf ausgeführten Software verantwortlich. Weiterhin sorgt er für die Bereitstellung der nötigen Software-Infrastruktur, d.h. er liefert die Software, wie z.B. Werkzeuge für Entwicklung, Test und Entwurf, die zur Realisierung des Projekts benötigt wird.

Qualitätsmanagement: Michael Hahn, Firas Zoabi

Der Qualitätsmanager sorgt dafür, dass alle Dokumente und auch Software-Komponenten eine bestimmte Qualität besitzen. Diese Qualität soll durch die Definition und Einhaltung von Richtlinien, durch die ständige Durchführung von Qualitätskontrollen und durch die Erstellung von einheitlichen Dokumentvorlagen erreicht werden. So soll während des gesamten Projektverlaufs die Qualität der Dokumente und Software-Einheiten maximiert werden.

Entwickler: Alle

Als Entwickler arbeiten alle Teammitglieder. Ein Entwickler arbeitet dabei im Rahmen der vom Projektleiter erteilten Aufgaben an Dokumenten und Software-Komponenten.

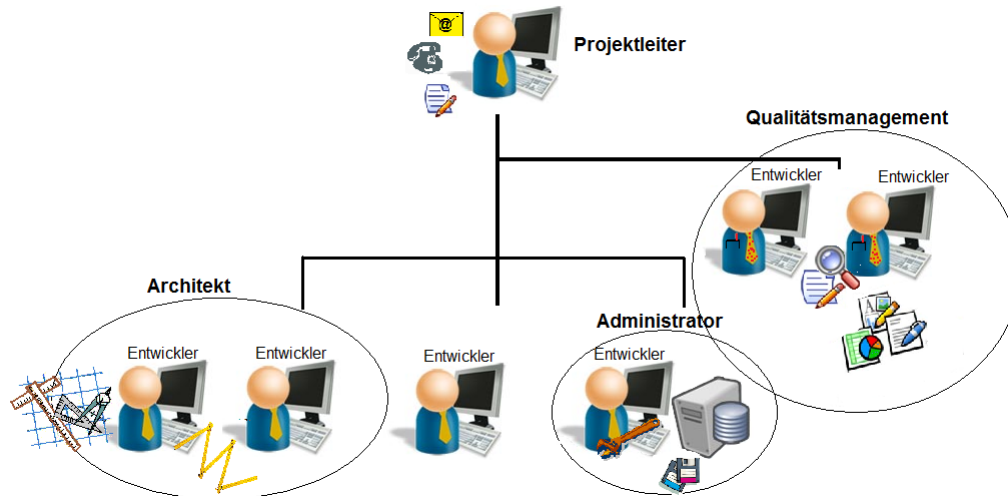
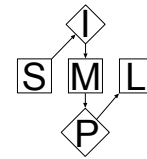


Abbildung 1: Rollen im Team

3.1.2 Kontaktdaten der Projektteam-Mitglieder

Wolfgang Hüttig: w.huettig(@)yahoo.de

Daniel Bruderle: daniel(@)brotherlee.de

Michael Schneidt: schneimi(@)studi.informatik.uni-stuttgart.de

René Rehn: Rene_Rehn(@)gmx.de

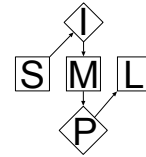
Firas Zoabi: Firas.Zoabi(@)web.de

Tu Xi: tu_xi(@)hotmail.com

Michael Hahn: hahnml(@)studi.informatik.uni-stuttgart.de

4 Nutzen

Durch das in diesem Projekt erstellte Rahmenwerk soll die Modellierung von wissenschaftlichen Workflows mit BPEL ermöglicht werden. Weiterhin werden der Eclipse BPEL-Designer, Apache ODE und BPEL selbst erweitert, um z.B. dynamische Datenquellenzugriffe und deren Überwachung möglich zu machen.



5 Leistungen im Rahmen des Angebots

5.1 Auszuliefernde Software-Einheiten

- Spezifikation
- Begriffslexikon
- Handbuch
- Dokumentation der Erweiterungen
- Testfälle
- Testprotokolle
- Installationspaket, das alle benötigten Software-Komponenten beinhaltet
- Eclipse Plug-In
- Quellcode

Alle Dokumente werden im PDF-Format geliefert.

5.2 Software-Einheiten, die nicht zum Lieferumfang gehören

Folgende Dokumente gehören nicht zur Lieferumfang:

- Anforderungskatalog
- Projektplan
- Entwurf
- Alle internen und organisatorischen Dokumente, wie z.B. Protokolle



6 Terminplanung

Das Studienprojekt läuft über zwei Semester und endet im Mai 2010. Jeder Projektteilnehmer hat dabei einen Arbeitsaufwand von ca. 480 Stunden zu leisten, dadurch entsteht ein Gesamtaufwand von ca. 3360 Stunden für alle 7 Teilnehmer. Die Projektphasen Analyse und Einarbeitung beanspruchen dabei 35%, Spezifikation und Entwurf (inkl. Fein-Entwurf) 24% und zuletzt die Codierung und der Test 33%. (siehe Abbildung 2)

Nach Berücksichtigung aller Einflussfaktoren von Seiten des Kunden und des Entwicklungsteams wird durch diese Planung vorausgesetzt, dass der Kunde das Endprodukt Mitte April 2010 testet und zwischen Anfang und Ende Mai 2010 nach Bedarf noch kleinere Änderungen durchgeführt werden.

Phase	Von / Bis	Aufwand
Analyse	13.05.2009-17.07.2009	658 h
Einarbeitung	17.07.2009-07.09.2009	526 h
Spezifikation	07.09.2009-31.10.2009	547 h
Entwurf	31.10.2009-24.12.2209	243 h
Codierung	24.12.2009-12.03.2010	779 h
Test	12.03.2010-15.04.2010	334 h
Abnahmetest	15.04.2010	—
Puffer	15.04.2010-13.05.2010	273 h

Tabelle 1: Phasen des Projekts

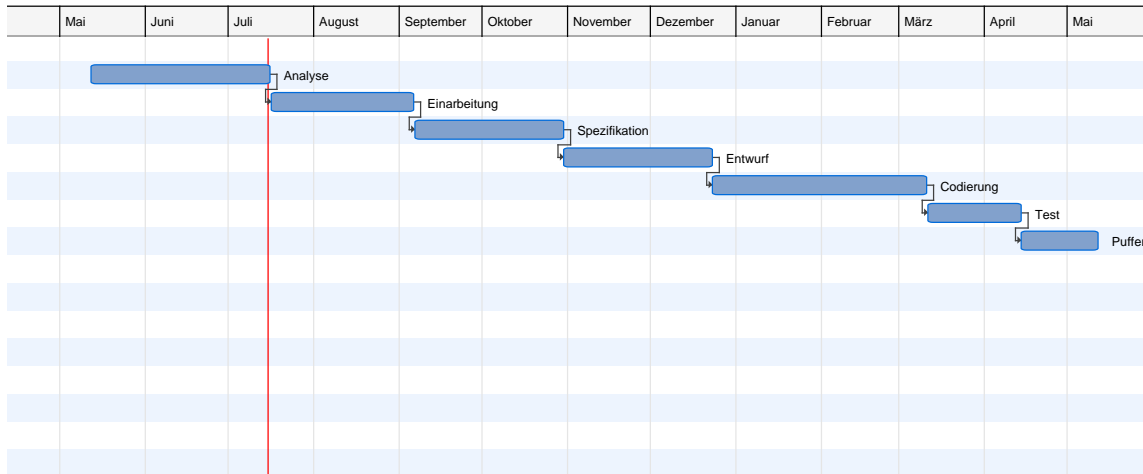
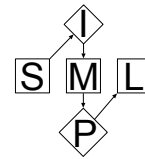


Abbildung 2: Terminplanung für die Entwicklungsphasen

6.1 Beschreibung der Phasen

In diesem Abschnitt wird jede der oben aufgeführten Phasen und die darin enthaltenen Aufgaben kurz näher erläutert.

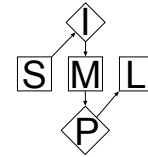
6.1.1 Analyse

Das Ziel der Analysephase ist es durch Kundengespräche die funktionalen und nicht-funktionalen Anforderungen zu erheben, die der Kunde an das zu entwickelnde Softwaresystem stellt. Das Ziel ist es einen Einblick in den momentanen Zustand des Arbeitsbereichs für den die Software entwickelt werden soll zu bekommen, um bereits realisierte Anforderungen aufzunehmen.

Am Ende der Analysephase wird aus den gewonnenen Erkenntnissen ein Anforderungskatalog erstellt, der als Referenz für die spätere Spezifikation der Software verwendet wird.

6.1.2 Einarbeitung

Da wenige bis gar keine Ansätze für Teilbereiche des Projekts wie z.B. die Verwendung von BPEL in Scientific Workflows existieren, müssen alle vorhandenen Ansätze evaluiert oder z.T. auch neue Ansätze geschaffen werden. Die Einarbeitungsphase dient dazu sich in die verschiedenen Themenbereiche einzuarbeiten und festzustellen in welchem Umfang Erweiterungsmaßnahmen notwendig sind.



6.1.3 Spezifikation

In dieser Phase wird die zu entwickelnde Software genau spezifiziert, d.h. in einem Dokument, der Spezifikation, werden alle funktionalen und nichtfunktionalen Anforderungen an die Software und ihre Schnittstellen präzise, vollständig und überprüfbar definiert. Ebenso werden spätere Abläufe im Umgang mit der Software und ihren Funktionen beschrieben. Die Spezifikation dient darüber hinaus als Referenz für viele nachfolgende Dokumente, wie z.B. das Handbuch und den Entwurf und sie ist unerlässlich für die Testphase und die spätere Wartung oder für spätere Erweiterungen der Software. Anhand der Spezifikation kann sich auch der Kunde davon überzeugen, dass wirklich seine Anforderungen umgesetzt werden.

6.1.4 Entwurf

In der Entwurfsphase wird die Architektur der Software festgelegt und in verschiedenen Granularitäten in den Dokumenten Grob-Entwurf (Architektur des Gesamtsystems) und Fein-Entwurf (Architektur der einzelnen Module) festgehalten.

6.1.5 Codierung

In der Codierungsphase werden die Anforderungen aus der Spezifikation mit der Architektur des Entwurfs umgesetzt. Am Ende dieser Phase existiert eine ausführbare Software.

6.1.6 Test

Die Testphase dient dazu die Software gegen die Spezifikation zu prüfen, d.h. alle Anforderungen aus der Spezifikation werden mit geeigneten Testfällen überprüft und die Software entsprechend korrigiert, falls sie nicht die gewünschten Ergebnisse liefert. Dazu wird eine Reihe von Testfällen definiert und deren Ausführung in einem Testprotokoll aufgezeichnet. Diese zwei Dokumente sind auch für den Kunden wichtig, da sie als Referenz für seine internen Tests während der Abnahme genutzt werden können.

6.1.7 Abnahmetest

Am Ende der Testphase erfolgt die Abnahme der Projekts und aller genannten Dokumente (siehe Kapitel 5.1).



Dem Kunden wird damit Zeit zur Verfügung gestellt sicherzustellen, dass das von uns gelieferte Produkt seinen Anforderungen entspricht. Ist das der Fall, dann gilt das Projekt als abgeschlossen, andernfalls werden die nötigen Nachbesserungen oder Änderungen durchgeführt.

6.1.8 Puffer

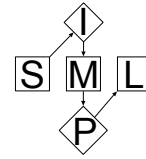
Ein extra Zeitfenster vor dem endgültigen Abgabetermin Ende Mai 2010. In dieser Phase können noch kleinere Änderungswünsche, die der Kunde nach dem Abnahmetest eventuell hat, umgesetzt werden.

7 Unsere Konditionen

7.1 Risiken

Aufgrund der geforderten sehr breiten Verwendungsmöglichkeiten im Rahmen der wissenschaftlichen Workflows besteht ein hohes Unsicherheitspotential im Bezug auf die Implementierungsdauer und auf sich ändernde Anforderungen.

Risiko	%	Kosten	Gegenmaßnahmen
Mitarbeiter fällt aus	20	2 MM	Wissen teilen, Zeitarbeit
Produkthaftung	5	1-10 MM	Versicherung abschliessen
Änderung der Anforderungen	50	1-5 MM	Flexible Struktur herstellen, regelmäßige Rücksprache mit Kunden
Qualität nicht ausreichend	30	3 MM	längere Testphase
Werkzeuge nicht einsetzbar	20	5 MM	Organisatorischer Slack schaffen
Infrastruktur Probleme	20	1 MM	einheitliche, zertifizierte Hardware
Anforderungen werden unterschätzt	40	5 MM	Schulungen
Software nicht lauffähig	20	6 MM	Prototyping
“Wildwuchs” der Änderungen	10	2 MM	Versionsmanagement
Entwicklungsverzug	30	6 MM	Überstunden
Kundenwünsche unrealistisch	30	1-5 MM	Anforderungen reduzieren.



7.2 Kostenschätzung

Unsere internen Berechnungen nach der Funktion-Point-Methode ergab folgende Kostenschätzung :

Gesamtentwicklungszeit :

19 MM (Mann-Monate)

Entwicklungszeit pro Mitarbeiter :

19 MM / 7 Mann = 2,71 Monate Entwicklungszeit pro Mitarbeiter

Zuschlag Entwicklungszeit (wg. Krankheit / Urlaub) :

2,71 Monate * 1,10 = 2,981 Monate Entwicklungszeit pro Mitarbeiter

Mitarbeiterstunden :

Die Mitarbeiter haben alle eine 5 Tage Woche mit jeweils 8 Stunden, können also im Monat 160 Stunden arbeiten.

Bei einer effektiven Entwicklungszeit von gerundet 3 Monaten (2,981) wird jede Person 480 Stunden

arbeiten.

Stundensätze

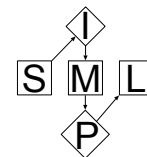
Entwickler : 69 € zzgl. MwSt.

Projektleiter : 99 € zzgl. MwSt.

Gesamtkostenschätzung :

6 Entwickler x 69 €/Stunde x 160 Stunden x 3 Monate = 198.720 €

1 Projektleiter x 99 €/Stunde x 160 Stunden x 3 Monate = 47.520 €



Gesamt : 246.240 € zzgl 19 % MwSt. = 293.025 €

7.3 AGBs

Allgemeine Geschäftsbedingungen (AGB) der SIMPL GmbH

1. Geltungsbereich und Vertragsschluss

Diese allgemeinen Geschäftsbedingungen gelten für alle Vertragsbeziehungen und vorvertraglichen Verhandlungen mit unseren Kunden, unabhängig von Art und Umfang der Leistung im Rahmen laufender und zukünftiger Geschäftsverbindungen. Allgemeine Geschäftsbedingungen des Kunden werden ausdrücklich nicht Vertragsinhalt, auch wenn ihnen seitens der Firma SIMPL GmbH nicht ausdrücklich widersprochen wird. Für den Fall das der Kunde die nachfolgenden allgemeinen Geschäftsbedingungen nicht gelten lassen will, hat er dies vorher schriftlich der Firma SIMPL GmbH anzuzeigen. Mündliche, telefonische und telegrafische Vereinbarungen bedürfen zu ihrer Wirksamkeit unserer schriftlichen Bestätigung. Unsere Angebote sind freibleibend und unverbindlich; sie dienen allein der Aufforderung des Kunden zur Abgabe eines Auftrages; dessen Annahme wir uns vorbehalten.

2. Preise

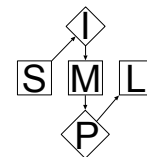
Alle Preise, auch diejenigen in der Auftragsbestätigung sind freibleibend. Für den Fall wesentlicher Änderungen der den Preis bestimmender Faktoren vor endgültiger Abwicklung der Bestellung bleibt eine entsprechende Anpassung an diese Änderungen vorbehalten.

3. Auftragsannullierung

Auftragsannullierungen bedürfen unser ausdrücklichen schriftlichen Zustimmung.

4. Lieferung

Alle von der Firma SIMPL GmbH genannten Liefertermine sind unverbindliche Liefertermine , es sei denn, dass ein Liefertermin ausdrücklich schriftlich vereinbart wird. Verlangt der Kunde nach Auftragserteilung Änderungen oder Ergänzungen des Auftrages oder treten sonstige Umstände ein, die der Firma SIMPL GmbH eine Einhaltung des Liefertermins unmöglich machen, obwohl die Firma SIMPL GmbH diese Umstände nicht zu vertreten hat, so verschiebt sich der Liefertermin um einen angemessenen Zeitraum. Wird die Firma SIMPL GmbH an der rechtzeitigen Vertragserfüllung z. B. durch Beschaffungs-, Fabrikations- oder Lieferstörungen bei ihr oder ihrem Zulieferanten gehindert, so gelten die allgemeinen Rechtsgrundsätze mit der Maßgabe, dass der Kunde nach Ablauf von einem Monat eine Nachfrist von 6 Wochen setzen kann. Ist die Nichteinhaltung eines verbindlichen Liefertermins nachweislich auf Krieg, Aufruhr,



Streik oder Aussperrung oder aus sonstige nach allgemeinen Rechtsgrundsätzen von der Firma SIMPL GmbH nicht zu vertretende Umstände zurückzuführen, so wird die Lieferfrist angemessen verlängert. Der Kunde kann vom Vertrag zurücktreten, wenn die Firma SIMPL GmbH nach Ablauf der verlängerten Frist eine angemessene Nachfrist zur Lieferung nicht erfüllt. Der Rücktritt hat schriftlich zu erfolgen. Wird der Firma SIMPL GmbH die Vertragserfüllung aus den vorgenannten Gründen ganz oder teilweise unmöglich, so wird sie von ihrer Lieferpflicht befreit.

5. Zahlung und Verzug

Hardware und Software werden auch im Rahmen eines Gesamtauftrages sofort in Rechnung gestellt. Alle Rechnungen der Firma SIMPL GmbH sind innerhalb von 10 Tagen ab Rechnungsdatum zahlbar. Maßgebend ist das Datum des Einganges der Zahlung bei der Firma SIMPL GmbH. Kommt der Kunde mit fälligen Zahlungen mehr als 7 Tage in Verzug, so ist die Firma SIMPL GmbH berechtigt, Verzugszinsen in Höhe von 5% über dem jeweiligen Referenzzinssatz der Europäischen Zentralbank zu berechnen. Weitere Ansprüche bleiben unberührt.

6. Rücktrittsrecht

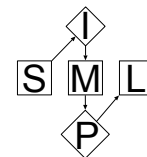
Hängt die Liefermöglichkeit von der Belieferung durch einen Vorlieferanten ab und scheitert diese Belieferung aus Gründen, die wir nicht zu vertreten haben, so sind wir zum Rücktritt vom Vertrag berechtigt. Erhalten wir nach Vertragsschluss Kenntnis von Tatsachen, die auf fehlende Kreditwürdigkeit des Kunden schließen lassen, so sind wir zum Rücktritt berechtigt, wenn der Käufer trotzdem zur Zug-um-Zug-Leistung oder Sicherheitsleistung nicht bereit ist.

7. Patent- und Urheberrechte

Die Firma SIMPL GmbH bleibt Inhaberin aller Urheber-, und Nutzungsrechte an der dem Kunden überlassenen Software, einschließlich der gesamten Dokumentation, auch wenn der Kunde sie verändert. Ohne unsere schriftliche Zustimmung dürfen weder die Software noch die Unterlagen, weder mittelbar noch unmittelbar, Dritten zugänglich gemacht werden.

8. Überlassung von Software

Zwischen der SIMPL GmbH und dem Kunden wird in einem gesonderten Vertragswerk, dem Rahmenvertrag, folgendes geregelt: Vertragsgegenstand, Vertragsdauer, Leistungsumfang, Nutzungsbedingungen für die beschriebene Software, Nutzung von Fremdsoftware, Beratungs- und Pflegeleistungen, Mitwirkungspflichten des Kunden, die Abnahme durch den Kunden, die Vergütung, die Gewährleistung, Haftung, Geheimhaltung und Datenschutz. Welche Leistungen im einzelnen von SIMPL GmbH zu erbringen sind ergibt sich aus den Leistungsscheinen und dem Software-Pflegevertrag, die als Anlagen zu



dem Rahmenvertrag genommen werden. Die SIMPL GmbH gewährt dem Kunden das nicht weiter übertragbare und nicht ausschließliche Recht, die in dem Lizenzschein aufgeführte spezifische Software und das Dokumentationsmaterial auf unbestimmte Zeit zu nutzen.

9. Eigentumsvorbehalt

Die Ware bleibt bis zum vollständigen Ausgleich unserer sämtlichen, auch künftig entstehenden Forderungen gegen den Kunden aus der Geschäftsverbindung - bei Herausgabe von Wechseln oder Schecks bis zu deren vorbehaltloser Einlösung - unser Eigentum.

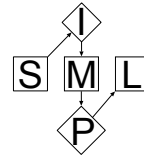
10. Gewährleistung

Die Gewährleistungsfrist beim Handelsgeschäft unter Kaufleuten beträgt 12 Monate. Sie beginnt mit der Auslieferung der Software an den Lizenznehmer. Der Kunde ist verpflichtet die gelieferte Software unverzüglich zu prüfen.

11. Haftung

SIMPL GmbH haftet wie folgt: Der Höhe nach unbegrenzt für Schäden, die auf Rechtsmängeln, zugesicherten Eigenschaften, Arglist oder auf vorsätzlichem oder grob fahrlässigem Verhalten der gesetzlichen Vertreter oder leitenden Angestellten von Intera beruhen. Unbegrenzt ist auch die Haftung nach dem Produkthaftungsgesetz. Für Schäden die aus der (leicht) fahrlässigen Verletzung wesentlicher Vertragspflichten (Kardinalpflichten) resultieren, die vorsätzlich oder grob fahrlässig durch Erfüllungsgehilfen oder leitende Angestellte verursacht wurden oder die auf anfänglichem Unvermögen beruhen, ist die Haftung von SIMPL GmbH wie folgt begrenzt: - Auf Schäden, mit deren Entstehen man typischerweise rechnen musste, sowie - auf die Hälfte der Lizenzvergütung für die betreffende Vertragssoftware, sofern es sich um Schäden aus der Überlassung eines Vertrags- softwareproduktes handelt, bzw. auf die Hälfte der im letzten Kalenderjahr für den betreffenden Service geleisteten Servicegebühr, sofern es sich um Schäden aus den vertraglichen Serviceleistungen handelt; - Auf den typischen Wiederherstellungsaufwand, der bei regelmäßiger und gefahrenstprechender Anfertigung von Sicherungskopien eingetreten wäre, wenn es sich um Schäden handelt, die auf dem Verlust von Daten beruhen. Im übrigen ist die Haftung ausgeschlossen. Intera trägt keinerlei Verantwortung für Inhalte, die der Kunde unter Verwendung der Software in das Internet und/oder sein Intranet stellt. Dies gilt auch dann, wenn Intera den Kunden bei der Auswahl der Inhalte beraten hat. Der Kunde wird Intera von jeglichen Ansprüchen freistellen, die gegen Intera wegen der angeblichen oder tatsächlichen Rechtswidrigkeit von mit unserer Software verarbeiteten Inhalten gestellt werden. Sämtliche Kosten, die Intera in diesem Zusammenhang entstehen, wird der Kunde erstatten. Die vorstehenden Regelungen gelten auch zugunsten der Mitarbeiter von Intera.

12. Rechtswahl, Erfüllungsort und Gerichtsstand



Unsere gesamten Geschäftsbeziehungen mit unseren Kunden unterliegen ausschließlich dem Recht der Bundesrepublik Deutschland. Die Anwendung des UN-Kaufrechts wird ausdrücklich ausgeschlossen. Der Erfüllungsort ist Stuttgart. Der Gerichtstand ist Stuttgart.

13. Salvatorische Klausel

Sollten eine oder mehrere der vorstehenden Vertragsbedingungen, oder daneben etwa abgeschlossener individueller Vereinbarungen ganz oder teilweise ungültig sein, berührt das die Wirksamkeit der übrigen Bedingungen nicht. Beide Vertragspartner sind in diesem Falle verpflichtet, eine unwirksame durch eine wirksame Bedingung zu ersetzen, die dem wirtschaftlichen Zweck der unwirksamen am nächsten kommt.

Stuttgart, den 15.07.2009