

Online-Handelsplattformen als verteilte Systeme am Beispiel einer Reverse-Pricing-Plattform

Martin Bernhardt, Oliver Hinz

Professur für Electronic Commerce
Johann Wolfgang Goethe-Universität Frankfurt am Main
Mertonstr. 17-25
60054 Frankfurt am Main
bernhardt@wiwi.uni-frankfurt.de
ohinz@wiwi.uni-frankfurt.de

Abstract: Verschiedene Online-Handelsplattformen bieten Unternehmen im Internet einen alternativen Vertriebskanal für den Verkauf von Produkten und Dienstleistungen. Als Transaktionsintermediär müssen diese Plattformen dabei häufig eine große Anzahl komplexer und teilweise heterogener Prozesse steuern. Viele dieser Prozesse können modularisiert und von Web Services übernommen werden. Anhand einer Fallstudie der Online-Handelsplattform uPrice.de, die den innovativen Reverse-Pricing-Mechanismus einsetzt, soll eine solche kostenoptimierende Modularisierung demonstriert werden. Die vorgestellten Web Services können dabei sowohl von externen Anbietern bezogen als auch selbst vermarktet werden.

1 Einleitung

Für viele Unternehmen haben sich Online-Handelsplattformen als alternativer Vertriebskanal zum Verkauf von Produkten und Dienstleistungen etabliert. Häufig lassen sich hierdurch neue Käufersegmente erreichen, die über den traditionellen Offline-Vertriebskanal nicht oder nur unzureichend angesprochen werden können [GGD02]. Der Einsatz innovativer Preismechanismen im Internet unterstützt dabei zusätzlich die Segmentierung des Marktes, ohne vorhandene Vertriebskanäle zu stark zu kannibalisieren. Neben verschiedenen Auktionsmechanismen gelangen im Internet auch neue, bislang unbekannte Preismechanismen wie das so genannte „Reverse Pricing“ zum Einsatz [SSS03; HT03; Ch03; Fa04].

Zusätzlich zur Preisfindung muss eine Online-Handelsplattform eine Reihe weiterer Prozesse (z.B. Dateneingabe, Transaktionsabwicklung) steuern. Viele dieser Prozesse sind standardisierbar und müssen nicht vom Kernsystem einer solchen Plattform selbst ausgeführt werden. Alternativ bieten sich Web Services zur Ausführung solcher Prozesse und zur Modularisierung einer Plattform-Architektur an. Diese können dabei entwe-

der vom Plattformbetreiber selbst angeboten oder als Dienstleistung Dritter in Anspruch genommen werden.

Als Web Services werden in diesem Beitrag Softwarekomponenten bezeichnet, die die Fähigkeit besitzen, Funktionen im Internet bereitzustellen und auszuführen. Dieser ursprünglich aus dem Distributed Computing stammende Ansatz ermöglicht es, durch die Verwendung von Standard-Internettechnologien Unabhängigkeit von Plattform und Programmiersprache zu erreichen [KW02]. Die Kommunikation zwischen Client und Server erfolgt dabei über XML-basierte Nachrichten. Während Konsument und Anbieter eines Web Services lose gekoppelt sind, bleiben Details der Implementierung des Web Services verborgen [Be01]. Auf die einzelnen, im Umfeld von Web Services entwickelten Standards wie WSDL (Web Service Description Language), UDDI (Universal Description, Discovery and Integration), WSCI (Web Service Choreography Interface) und schließlich SOAP (Simple Object Access Protocol) soll in diesem Beitrag nicht näher eingegangen werden. Stattdessen wird nachfolgend die Architektur einer durch Web Services modularisierten Anwendung in einer Fallstudie erläutert, wobei der Fokus auf der aggregierten Sichtweise liegt und technische Details sowie Sicherheitsaspekte in den Hintergrund treten.

Durch die Modularisierung einzelner Prozesse tragen Web Services zu einfacherer System-Pflege, zur Reduktion von Redundanzen und schlussendlich zur Optimierung der Kosten eines Plattformbetreibers bei [COU02]. Web Services lassen sich dabei nicht nur zur Übernahme verschiedener interner Prozesse einer Plattform selbst einsetzen, sondern auch gegenüber dritten Parteien als zusätzliche Dienstleistung vermarkten. Die Skalierbarkeit einer solchen Erlösquelle könnte hierbei zu einer Steigerung des Deckungsbeitrages führen. Geschäftsmodelle in diesem Zusammenhang sind in [BS03] zu finden.

Dieser Beitrag stellt die Architektur der Online-Handelsplattform [uPrice.de](http://www.uprice.de) (www.uprice.de), die im Rahmen des vom Bundesministerium für Bildung und Forschung (BMBF) geförderten Forschungsschwerpunktes „Internetökonomie“ entwickelt wurde, vor. Ziel des Beitrags ist, anhand dieser Fallstudie zu demonstrieren, wie durch die Auslagerung verschiedener Prozesse in Web Services die Grundlage einer flexiblen und skalierbaren Infrastruktur gelegt werden kann.

Zunächst wird in Kapitel 2 die Funktionsweise des von der Online-Handelsplattform uPrice.de eingesetzten Preismechanismus Reverse Pricing beschrieben und Unterschiede gegenüber bekannten Auktionsmechanismen erörtert. Kapitel 3 bildet mit einem Überblick über die Architektur des Gesamtsystems sowie einer Beschreibung der eingesetzten Web Services den Kern des Beitrages. Kapitel 4 beschließt den Beitrag mit einer Zusammenfassung.

2 Dynamische Preismechanismen im Internet

Dynamische Preismechanismen werden seit langer Zeit in Branchen eingesetzt, die sich durch eine fixe oder nur mit großen Kosten veränderliche Kapazität bei gleichzeitiger

Nichtlagerbarkeit der Produkte kennzeichnen lassen. Als klassische Beispiele werden hierfür insbesondere die Flug-, die Telekommunikations- oder die Energiebranche häufig herangezogen [EK03]. Im Internet hingegen haben die Digitalisierung von Prozessen und die unabhängig von physischen Aufenthaltsorten beteiligter Transaktionspartner ermöglichte Interaktivität zu einer starken Senkung der Transaktionskosten beigetragen [Ba98]. Dies hat einerseits zum Einsatz bereits bekannter dynamischer Preismechanismen in neuen Branchen geführt (z.B. Online-Auktionen) und andererseits die Entwicklung neuer und in der Offline-Welt bislang unbekannter Preismechanismen vorangetrieben (z.B. „Co-Shopping“ oder „Reverse Pricing“).

Neben einer Ausrichtung auf unterschiedliche Kundensegmente und Produkte lassen sich Online-Handelsplattformen anhand des eingesetzten Preismechanismus als zentralem Element klassifizieren. Da die in diesem Beitrag vorgestellte Online-Handelsplattform uPrice.de den innovativen Reverse-Pricing-Mechanismus einsetzt, soll eine Beschreibung der Funktionsweise gemeinsam mit einer Abgrenzung zu den im Internet häufig eingesetzten Auktionsmechanismen zur besseren Verständlichkeit des Preismechanismus auf uPrice.de beitragen.

2.1 Funktionsweise des Reverse Pricing

Als dynamischer Preismechanismus ermöglicht das Reverse Pricing sowohl dem Käufer als auch dem Verkäufer eines Produktes Einflussnahme auf den endgültigen Preis einer Transaktion. Während der Verkäufer durch die Bestimmung einer geheimen Preisschwelle einen Mindestpreis für diese Transaktion vorgibt, bestimmt ein Käufer die endgültige Höhe des Transaktionspreises durch die Abgabe eines Gebotes. Liegt dieses Gebot oberhalb der geheimen Preisschwelle des Verkäufers, wird die Transaktion in Höhe des vom Käufer genannten Gebotes ausgeführt. Reicht ein zunächst abgegebenes Gebot eines Käufers hingegen nicht aus, um die geheime Preisschwelle des Verkäufers zu übertreffen, kann ein Käufer bei entsprechender Gestaltung des Mechanismus weitere Gebote abgeben und sein ursprüngliches Gebot erhöhen.

Ein Verkäufer hat dabei die Möglichkeit, das Reverse Pricing anhand verschiedener Design-Variablen seinen Bedürfnissen gemäß anzupassen. Design-Variablen beschreiben dabei diejenigen Eigenschaften, anhand derer sich eine konkrete Ausprägung des Reverse Pricing bestimmen lässt (z.B. die maximal mögliche Anzahl an Geboten eines Käufers oder ein Mindestinkrement für zusätzliche Gebote). [Be04] geht detailliert auf verschiedene Design-Variablen ein.

2.2 Abgrenzung des Reverse Pricing zu Online-Auktionsmechanismen

Die große Anzahl an im Internet verfügbaren Auktionsplattformen wie eBay.de (www.ebay.de), hood.de (www.hood.de) oder Azubo.de (www.azubo.de) erschwert generelle Aussagen über den Online-Auktionsmarkt aufgrund der vorherrschenden Heterogenität eingesetzter Auktionsmechanismen [PSV03]. Es lässt sich jedoch eine Reihe

von Eigenschaften identifizieren, die vielen der Auktionsseiten gemein sind [Lu99] und anhand derer sich diese von dem auf uPrice.de eingesetzten Reverse-Pricing-Mechanismus abgrenzen lassen.

Im Gegensatz zu Auktionsmechanismen besteht beim Reverse Pricing **keine unmittelbare Preiskonkurrenz** zwischen den Bietern. Um eine Transaktion zum Abschluss zu bringen, müssen diese lediglich die ihnen unbekannte Preisschwelle eines Anbieters überbieten. Gebote anderer Bieter hingegen haben keinen direkten Einfluss auf den Preis einer Transaktion. Lediglich im Fall einer begrenzt verfügbaren Produktkapazität bei gleichzeitig vorhandener großer Nachfrage könnten sich Bieter dazu veranlasst sehen, ein Gebot möglichst früh abzugeben, um den Geboten anderer Bieter zuvor zu kommen. Dieses Verhalten wäre vergleichbar mit einem Szenario einer holländischen Auktion mit fallenden Preisen (z.B. bei Azubo.de).

Eines der kennzeichnenden Merkmale des Reverse Pricing ist, dass auf uPrice.de Informationen weder über bereits erzielte Preise noch über die geheime Preisschwelle eines Verkäufers preisgegeben werden. Die hieraus resultierende **Preisintransparenz** erlaubt es, Produkte über einen solchen Vertriebskanal auch zu günstigeren Preisen zu verkaufen und somit neue Käufersegmente anzusprechen, ohne hierdurch die Preisstrukturen eines anderen Vertriebskanals zu kannibalisieren. Dies ist insbesondere bei den häufig anzutreffenden englischen Auktionen wie bei eBay.de oder hood.de nicht möglich. Informationen über das aktuelle Höchstgebot sind hier ständig verfügbar.

Schließlich eignet sich Reverse Pricing insbesondere für den **Abverkauf einer großen Anzahl identischer Produkte**, da hierfür lediglich ein einziges Angebot erstellt werden muss. Zwar bieten verschiedene Auktionsseiten mittlerweile so genannte „Power-“ oder „Dutch-Auktionen“ an (z.B. eBay.de, BesteAuktion.de), jedoch wird bei diesen für n Einheiten eines Produktes ein uniformer Preis durch das n -höchste Gebot eines Bieters determiniert. Im Gegensatz hierzu ermöglicht das Reverse Pricing kundenindividuell differenzierte Preise und somit eine bessere Abschöpfung der Konsumentenrente für einen Anbieter.

3 Modularisierung eines Gesamtsystems durch Web Services

Am Beispiel der Online-Handelsplattform uPrice.de soll aufgezeigt werden, wie einzelne Prozesse aus dem Kernsystem einer Gesamt-Architektur ausgegliedert und stattdessen modular von Web Services durchgeführt werden können. Dabei spielt es keine Rolle, ob die Web Services vom Plattformbetreiber selbst implementiert und schließlich gegen Entgelt Dritten zur Verfügung gestellt werden oder vom Plattformbetreiber als Service eingekauft werden.

uPrice.de tritt als reiner Transaktionsintermediär im Markt auf. Die Online-Handelsplattform dient folglich dazu, Prozesse ausschließlich bis zum Zustandekommen eines Kaufvertrages zu unterstützen. Vor- und nachgelagerte Prozesse wie die Herstellung oder Bezahlung von Produkten und Dienstleistungen hingegen werden gegenwärtig

von uPrice.de nicht angeboten. Kernprozess von uPrice.de ist dabei die Generierung eines Transaktionspreises anhand des in Kapitel 2.1 erläuterten Reverse-Pricing-Mechanismus. Nachfolgend soll zunächst die modularisierte Architektur des Gesamtsystems aufgezeigt und dann näher auf einzelne Module eingegangen werden.

3.1 Modularisierte Architektur einer Handelsplattform mit Web Services

Nicht alle Prozesse einer Handelsplattform eignen sich zur Ausgliederung und müssen daher teilweise im Kernsystem verbleiben. Datenschutzrechtliche Gründe machen es beispielsweise notwendig, die Verwaltung personenbezogener Daten innerhalb des Kernsystems durchzuführen. Angebots- und Produktdaten hingegen lassen sich problemlos ausgliedern. Auch für den Prozess der Preisfindung ist eine modulare Durchführung von einem außerhalb des Kernsystems angesiedelten Spezialsystem denkbar. Abbildung 1 veranschaulicht die Architektur einer durch Web Services modularisierten Handelsplattform anhand des Beispiels der Online-Handelsplattform uPrice.de. Zahlreiche Prozesse werden eigenständig von Web Services ausgeführt und können daher modular in eigene oder fremde Systeme (z.B. Online-Shops oder weitere Handelsplattformen) eingebunden werden.

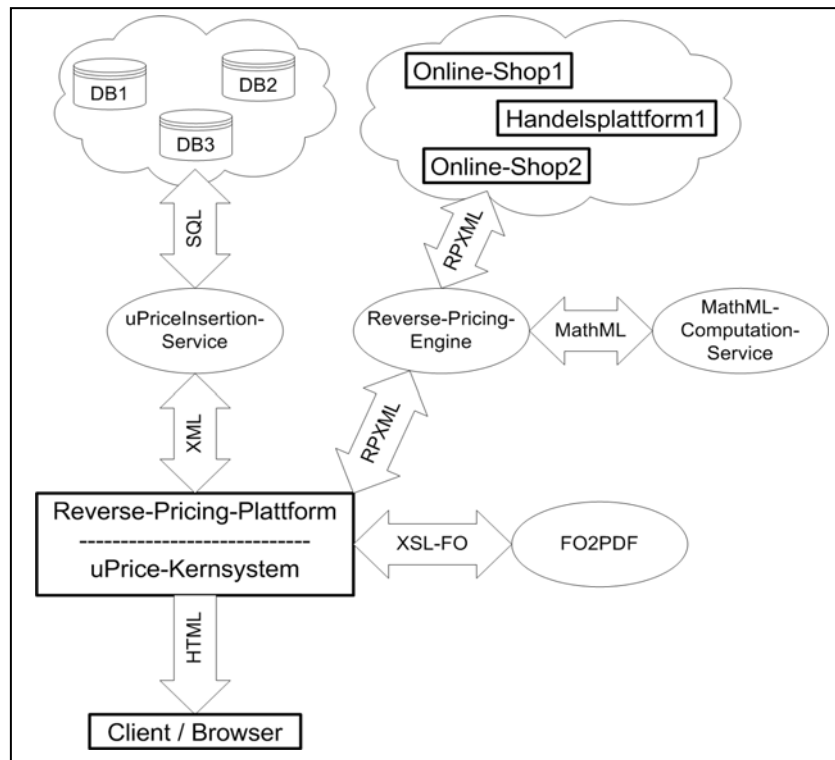


Abbildung 1: Modularisierte Architektur der Online-Handelsplattform uPrice.de mit Web Services

Die bei uPrice.de vorliegende modularisierte Architektur ermöglicht sowohl die Einstellung von Angebots- und Produktdaten über den *uPriceInsertion-Service* als auch die Auslagerung des Preismechanismus an einen Web Service, der so genannten *Reverse-Pricing-Engine*. Weitere Web Services, wie *FO2PDF* oder der *MathML-Computation-Service*, sind ebenfalls in die Architektur der Online-Handelsplattform integriert. Die letztendlich für den Käufer bestimmten Angebote können dann vom *uPrice-Kernsystem* dynamisch im Browser des Clients aus den Angebots- und Produktdaten sowie den Informationen der *Reverse-Pricing-Engine* zusammengestellt werden.

Gibt ein potentieller Käufer ein Gebot ab, wird dieses Gebot gemeinsam mit eindeutigen Bezeichnern für Produkt und Käufer sowie einigen weiteren Parametern an den Web Service der *Reverse-Pricing-Engine* übermittelt. Diese wertet die Parameter aus und prüft, ob das Gebot über der geheimen Preisschwelle des Verkäufers liegt. Ist dies nicht der Fall, liefert die *Reverse-Pricing-Engine* eine entsprechende Meldung an das *uPrice-Kernsystem* zurück. Ferner wird übermittelt, ob ein weiteres Gebot für diesen Käufer auf dieses Produkt möglich ist und unter welchen Bedingungen dies gegebenenfalls geschehen kann. Aufgrund dieser Informationen kann das *uPrice-Kernsystem*, wie in Abbildung 2 veranschaulicht, eine individualisierte Gebotsseite für den potentiellen Käufer erstellen.

Angebot: "Casio- Unisex Quarz Armbanduhr *NEU*"		Ihr Preis: € 0 . 00
Anzahl:	<input type="text" value="1"/>	Gebot abgeben
Anzahl bisheriger Gebote:	3	
Ihr letztes Gebot war:	€ 16,00	(Meine Gebote anzeigen!)
Dieses Angebot läuft am 31.07.2004, 12:00 aus.		
Sie können noch 2 Gebote abgeben.		
Dieses Gebot ...	Nachfolgende Gebote ...	
kostet Sie: [Hilfe]	0,50 €	kosten Sie: [Hilfe] 0,50 €

Abbildung 2: Individualisierte Gebotsseite nach einem abgewiesenen Gebot

Im Erfolgsfall entspricht das Gebot eines Käufers dem zu zahlenden Transaktionspreis. Eine Mail an die beiden an der Transaktion beteiligten Parteien wird generiert und in einigen Rubriken werden Gutscheine, so genannte „Vouchers“, der Mail als Attachment hinzugefügt. Für die Generierung dieser Vouchers steht der Web Service *FO2PDF* zur Verfügung, der in Kapitel 3.5 genauer beschrieben wird.

3.2 Einstellung von Produkt- und Angebotsdaten über den uPriceInsertion-Service

Um Angebote potenziellen Käufern präsentieren zu können, müssen zunächst Angebots- und Produktdaten vom System des Verkäufers in das *uPrice-Kernsystem* übertragen

werden. Die Eingabe dieser Daten kann bei uPrice.de im Normalfall über ein Webinterface geschehen. Bei der Übergabe einer Vielzahl von Katalogdaten bietet sich allerdings das automatisierte Einspeisen der Daten über einen Web Service an.

Der *uPriceInsertion-Service* besteht wie aus Abbildung 3 hervorgeht aus drei Vorgängen: Mit dem Vorgang `validateXML` wird das übergebene XML-Dokument gegen eine XML Schema Definition (XSD) validiert. Zudem werden einige weitere notwendige Bedingungen abgeprüft, die nicht über XSD abgebildet werden können. Der Vorgang `uploadImages` ermöglicht das Aufspielen binärer Bilddateien unter Angabe von Login und Passwort. Abschließend können die Produkt- und Angebotsdaten mit dem Vorgang `insertProducts` in das System eingespeist und online gestellt werden.

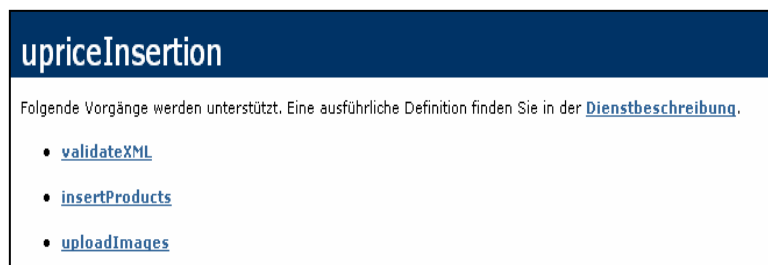


Abbildung 3: Vorgänge des uPriceInsertion-Service

Der Web Service zerlegt die ankommenden Daten in einen Produktteil und in Informationen, die zur Preisfindung anhand des Reverse-Pricing-Mechanismus benötigt werden. Diese Informationen sind in RPXML spezifiziert und geben an, wie der Reverse-Pricing-Mechanismus ausgestaltet werden soll. Die *Reverse-Pricing-Engine* wertet diese RPXML-Strukturen aus; Kapitel 3.3 vermittelt hierüber einen genaueren Einblick.

uPrice.de stellt einen Client zur Verfügung, der den oben genannten Web Service einbindet und die verschiedenen Vorgänge in einer logischen Reihenfolge anstößt. Zunächst wird das übergebene XML-Dokument wie oben beschrieben validiert. Im Fehlerfall werden genaue Fehlermeldungen ausgegeben.

Entspricht das XML-Dokument den Spezifikationen, werden die lokalen Bilddateien, die zur Produktbeschreibung dienen, aus den `<image_path>`-Tags ermittelt (vgl. Abbildung 4). Die binären Daten werden an das *uPrice-Kernsystem* übertragen. Als Ergebnis wird für jedes übergebene Bild eine Referenz zurückgegeben, mit der die lokalen `<image_path>`-Tags ersetzt werden. Abschließend werden die Produkte mit dem Vorgang `insertProducts` eingestellt.

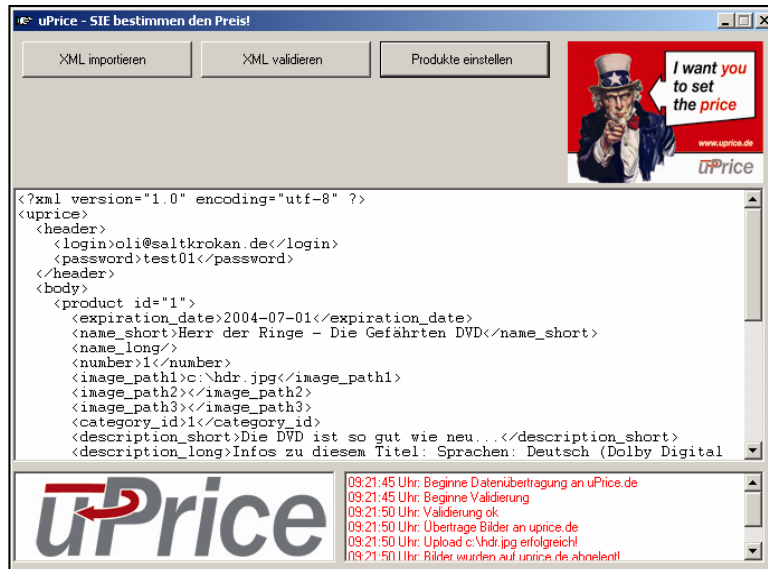


Abbildung 4: Web Service-Client zum Einstellen von Daten

Mit dieser Client-Applikation und dem Web Service von uPrice.de können Verkäufer die meist schon vorhandenen Katalogdaten ohne viel Aufwand an die Plattform senden. Lediglich das Design des Reverse-Pricing-Mechanismus muss vom Verkäufer eingestellt werden. Zu diesem Zweck stellt die Online-Handelsplattform uPrice.de zahlreiche Informationen über verschiedene Design-Alternativen als Hilfestellung zur Verfügung.

3.3 Beschreibung des Reverse-Pricing-Mechanismus mit RPXML

Zur Übermittlung von Produkt- und Angebotsdaten über einen Web Service wird eine standardisierte Beschreibungssprache benötigt. Mit ihrer Hilfe lassen sich detaillierte Informationen über das angebotene Produkt und den dazu verwendeten Preismechanismus von einem Client auf einen Server übertragen. Im Electronic Business existieren bereits zahlreiche Standardmodelle, die zur Übermittlung solcher Daten eingesetzt werden können (z.B. ebXML, cXML, RosettaNet). Einen Überblick über verschiedene Standards im Electronic Business verleiht [WHB01].

Um Angebotsdaten aus dem Bereich Reverse Pricing übermitteln zu können, werden hingegen komplexe Strukturen benötigt, die von bisherigen Standards nur unzureichend bereitgestellt werden. RPXML, eine auf XML-basierende Beschreibungssprache für Reverse Pricing, löst dieses Problem durch eine Erweiterung bestehender Sprachelemente um ein so genanntes `<reversePricing>`-Tag, das stellvertretend für die in vielen Standards vorhandenen Elemente zur Beschreibung eines fixen Preises (z.B. `<article_price_details>` in BMEcat [BME03]) eingesetzt werden kann [BH04]. Unterhalb dieses Tags bieten verschiedene, hierarchisch angeordnete Tags in RPXML die Möglichkeit, den eingesetzten Reverse-Pricing-Mechanismus exakt zu beschreiben. Neben

grundlegenden Elementen wie der geheimen Preisschwelle und der maximal möglichen Anzahl an Geboten eines Käufers lassen sich hierüber insbesondere Informationen zu verschiedenen Restriktionen wie z.B. minimale Erhöhungsbeträge oder Zeitverzögerungen zwischen zwei aufeinander folgenden Geboten eines Käufers festlegen. Eine XML Schema Definition für RPXML ist als Download verfügbar [RPX04].

Neben der standardisierten Beschreibung entsprechender Angebotsdaten wird RPXML im vorgestellten Gesamtsystem insbesondere zur Datenübertragung zwischen dem Web Service der *Reverse-Pricing-Engine* und dem *uPrice-Kernsystem* als Client eingesetzt (vgl. Abbildung 1). Somit wird die Information über den Preismechanismus vollständig aus dem Kernsystem ausgelagert und eine Modularisierung erreicht. Hierdurch können auch andere Clients (z.B. Online-Shops) den Web Service der *Reverse-Pricing-Engine* nutzen, um den Reverse-Pricing-Mechanismus im eigenen Angebot zu integrieren.

3.4 Dynamische Erweiterung von RPXML mit dem MathML-Computation-Service

Mit Hilfe der durch RPXML festgelegten Beschreibungssprache ist zunächst die Abbildung einer Grundfunktionalität des Reverse Pricing möglich. Neben der Spezifikation des gewünschten Mindestpreises können Verkäufer eine Reihe von Tags definieren und den Mechanismus gemäß der von ihnen gewünschten Eigenschaften ausgestalten. Darüber hinaus gestattet RPXML auch die Berücksichtigung dynamischer Einflussgrößen zur Gestaltung eines entsprechenden Reverse-Pricing-Mechanismus. Die Abhängigkeit der geheimen Preisschwelle von zur Laufzeit verfügbaren Einflussgrößen, z. B. der Restlaufzeit eines Angebots oder dem durch die Anzahl der Pageviews gemessenen Interesse an einem Angebot, lässt sich hierdurch realisieren. Zur Beschreibung solcher funktionaler Abhängigkeiten wird im Rahmen von RPXML die Mathematical Markup Language (MathML) [Ca03] eingesetzt.

Ist die Erweiterung von RPXML um dynamische und zur Laufzeit erzeugte Parameter erwünscht, kann die *Reverse-Pricing-Engine* auf den *MathML-Computation-Service* zurückgreifen. Dieser Web Service wertet die übergebenen MathML-Fragmente in einem zweistufigen Verfahren aus, indem zunächst die abhängigen Variablen um mitgelieferte Parameterwerte ersetzt und schließlich die in MathML spezifizierten Ausdrücke berechnet werden. Als Rückgabewert dient stets das Formelergebnis, das sich im Kontext des Gesamtsystems z. B. als dynamisch berechneter Erhöhungsbetrag einer geheimen Preisschwelle interpretieren lässt. Um ein erfolgreiches Gebot abzugeben, müssten Käufer dann im nächsten Gebotsschritt eine um diesen Betrag erhöhte Preisschwelle überbieten.

3.5 Automatische Generierung von PDF-Dateien mit FO2PDF

Die Online-Handelsplattform uPrice.de übernimmt neben der Rolle als Intermediär auch einige Service-Dienstleistungen. Für Käufer und Anbieter aus der Restaurant- oder Tourismusbranche werden direkt beim Abschluss einer Transaktion „Vouchers“ generiert,

die beim Verkäufer bezahlt und eingelöst werden können. Um Fälschungssicherheit zu gewährleisten, arbeitet uPrice.de mit TAN-Listen. Zudem sollen die Vouchers ansprechend gestaltet und nicht wie eine Text- oder HTML-Datei leicht editierbar sein. Dazu werden die Vouchers als PDF-Dateien erstellt, die dem Käufer direkt am Bildschirm zum Ausdruck ausgegeben oder mit der Bestätigungsmail an Käufer und Verkäufer als Attachment versendet werden.

Zur Generierung von PDF-Dateien eignet sich der Formatting Objects Processor (FOP) [FOP04]. Als Teil des Apache XML-Projekts [APA04] ist FOP kostenlos. Neben PDF als primärem Ausgabeformat kann FOP aus einer XSL-Formatting Object (XSL-FO)-Baumstruktur verschiedene Ausgabeformate wie z.B. PCL, PS, SVG, AWT, MIF und TXT erzeugen.

Das Eingabeformat XSL-FO ist ein Bestandteil der Extensible Stylesheet Language (XSL)-Empfehlung des W3-Konsortiums und kombiniert Cascading Stylesheets (CSS) und Document Style Semantics and Specification Language (DSSSL) in einer gut strukturierten und stabilen Spezifikation von Formatierungseinheiten zur Seitengestaltung.

Zur Generierung der Vouchers wird, wie in Abbildung 5 zu sehen, der Web Service *FO2PDF* eingesetzt, der eine entsprechende XSL-FO-Baumstruktur als Eingabe entgegennimmt und den erstellten Gutschein als PDF-Dokument zurückliefert.

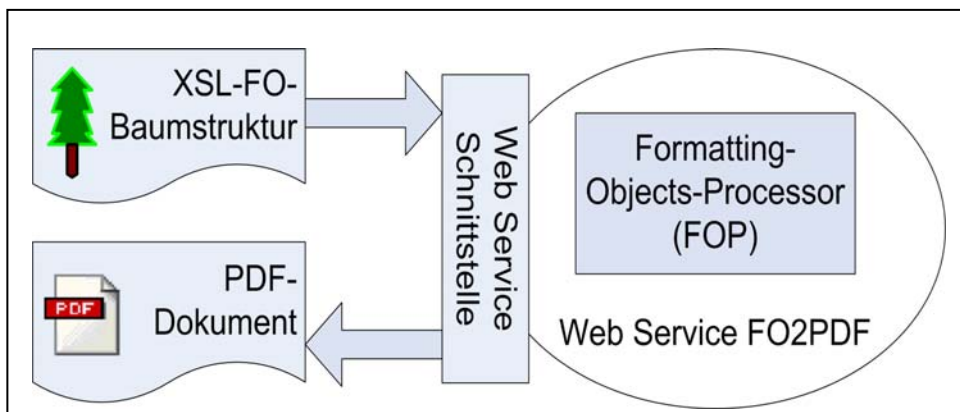


Abbildung 5: Der Web Service FO2PDF

Da es sich bei PDF-Dokumenten um binäre Daten handelt und diese nicht ohne weiteres in einem XML-Strom übertragen werden können, ist eine Erweiterung des Protokolls notwendig. Um binäre Dateien mit SOAP-Nachrichten verschicken zu können, gibt es die Empfehlung „SOAP Messages with Attachments“ des W3C [BTN00]. Als Alternative dazu hat sich Direct Internet Message Encapsulation (DIME), das bei der Internet Engineering Task Force (IETF) als Draft vorliegt und von Microsoft und IBM unterstützt wird, etabliert [Ni02].

4 Zusammenfassung

Die vorliegende Fallstudie zeigt, wie Web Services an vielen Stellen sinnvoll eingesetzt werden können. Besonders Prozesse, die für eine externe Vermarktung von Interesse sind, eignen sich zur Implementierung als Web Service. So könnte eine Berechnung von mathematischen Ausdrücken, wie sie im *MathML-Computation-Service* durchgeführt wird, für viele Anwendungsbereiche von Interesse sein.

Speziell die in diesem Beitrag vorgestellte *Reverse-Pricing-Engine* ist darauf ausgelegt, die hinterlegte Preisfindungslogik nicht nur auf der Online-Handelsplattform uPrice.de einzusetzen, sondern eine Verwendung auch für andere Plattformen und Onlineshops zu ermöglichen und damit zu einer weiteren Verbreitung des Reverse Pricing beizutragen. Das in der *Reverse-Pricing-Engine* gekapselte Know-how soll im weiteren Verlauf des Gesamtprojekts als Grundlage eines Application Service Providers (ASP) dienen.

Ein großer Vorteil der vorgestellten Architektur ist die Interoperabilität durch den Einsatz von Web Services. So ist der in Kapitel 3.5 beschriebene Formatting Objects Processor zwar in Java implementiert, kann über die Web Service-Schnittstelle vom in ASP.NET erstellten *uPrice-Kernsystem* aber problemlos eingebunden werden.

Anhand der Fallstudie von uPrice.de demonstriert dieser Beitrag wie durch die Ausgliederung von Prozessen in Web Services eine höhere Modularisierung der gesamten Architektur erreicht werden kann. Die hierdurch ermöglichte Steigerung der Wartbarkeit und Flexibilität kann letztendlich zu Kostenoptimierungen führen.

Literaturverzeichnis

- [APA04] Apache.org: The Apache XML Project, <http://xml.apache.org/>, 2004, Abruf am 01.07.2004.
- [Ba98] Bakos, Y.: The Emerging Role of Electronic Marketplaces on the Internet. *Communications of the ACM* 41, 1998; S. 35-42.
- [Be01] Bettag, U.: Aktuelles Schlagwort: Web-Services. *Informatik-Spektrum* 24 (5), 2001; S. 302-304.
- [Be04] Bernhardt, M.: Classification of Design Options in Reverse Pricing Mechanisms. In (Bichler, M.; Holtmann, C.; Kirn, S.; Müller, J. P.; Weinhardt, C., Hrsg.): *Coordination and Agent Technology in Value Networks*. GITO, Berlin, 2004; S. 29-43.
- [BH04] Bernhardt, M.; Hinz, O.: *RPXML – Standardisierung von Reverse-Pricing-Mechanismen*. Arbeitspapier, Johann Wolfgang Goethe-Universität Frankfurt, Frankfurt am Main, 2004.
- [BME03] eBusiness Standardization Committee: BMEcat, Version 1.2, <http://www.bmecat.org>, 2003, Abruf am 07.07.2004.
- [BS03] Boles, D.; Schmees, M.: Kostenpflichtige Web-Services. In (Uhr, W.; Esswein, W., Schoop, E., Hrsg.): *Wirtschaftsinformatik 2003, Medien - Märkte - Mobilität, Band I*. Physica-Verlag, Heidelberg, 2003; S. 385-404.
- [BTN00] Barton, J. J.; Thatte, S.; Nielsen, H. F.: *SOAP Messages with Attachments*, <http://www.w3.org/TR/SOAP-attachments/>, 2000, Abruf am 01.07.2004.

- [Ca03] Carlisle, D. et al.: Mathematical Markup Language (MathML) Version 2.0 (Second Edition) W3C Recommendation 21 October 2003. <http://www.w3.org/TR/2003/REC-MathML2-20031021/>, 2003, Abruf am 07.07.2004.
- [Ch03] Chernev, A.: Reverse Pricing and Online Price Elicitation Strategies in Consumer Choice. *Journal of Consumer Psychology* 13 (1&2), 2003; S. 51-62.
- [COU02] Council on Technology Services: Final Report of the COTS Web Services Workgroup, <http://www.cots.state.va.us/Web%20Services/Web%20Services%20Final%20Report%2010-1-2002.pdf>, 2002, Abruf am 07.07.2004.
- [EK03] Elmaghraby, W.; Keskinocak, P.: Dynamic Pricing in the Presence of Inventory Considerations: Research Overview, Current Practices, and Future Directions. *Management Science* 49, 2003; S. 1278-1309.
- [Fa04] Fay, S.: Partial Repeat Bidding in the Name-Your-Own-Price Channel. *Marketing Science*, zur Veröffentlichung angenommen.
- [FOP04] Formatting Objects Processor: FOP, <http://xml.apache.org/fop/index.html>, 2004, Abruf am 01.07.2004.
- [GGD02] Geyskens, I.; Gielens, K.; Dekimpe, M. G.: The Market Valuation of Internet Channel Additions. *Journal of Marketing* 66 (2), 2002; S. 102-119.
- [HT03] Hann, I.-H.; Terwiesch, C.: Measuring the Frictional Costs of Online Transactions: The Case of a Name-Your-Own-Price Channel. *Management Science* 49 (11), 2003; S. 1563-1579.
- [KW02] Kuschke, M.; Wölfel, L.: *Web Services kompakt*. Spektrum Akademischer Verlag, Heidelberg, 2002.
- [Lu99] Lucking-Reiley, D.: Using Field Experiments to Test Equivalence between Auction Formats: Magic on the Internet. *American Economic Review* 89 (11), 1999; S. 1063-1080.
- [Ni02] Nielsen, H. F. et al.: Direct Internet Message Encapsulation (DIME), <http://msdn.microsoft.com/library/en-us/dnglobspec/html/draft-nielsen-dime-02.txt>, 2002, Abruf am 07.07.2004.
- [PSV03] Pinker, E. J.; Seidmann, A.; Vakrat, Y.: Managing Online Auctions: Current Business and Research Issues. *Management Science* 49, 2003; S. 1457-1484.
- [RPX04] Reverse Pricing in XML (RPXML) Version 1.01. <http://www.reverse-pricing.com/downloads/rpxml.xsd>, 2004, Abruf am 08.07.2004.
- [SSS03] Spann, M.; Skiera, B.; Schäfers, B.: Reverse-Pricing-Verfahren und Möglichkeiten zur Messung von individuellen Suchkosten und Zahlungsbereitschaften. *Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung*, zur Veröffentlichung angenommen.
- [WHB01] Weitzel, T.; Harder, T.; Buxmann, P.: *Electronic Business und EDI mit XML*. dpunkt, Heidelberg, 2001.