



cXML Guida per l'utente

VERSIONE 1.1

APRILE 2000

Con il presente documento, la Ariba, Inc. (Ariba) garantisce all'utente un diritto e una licenza perpetui, non esclusivi, esenti da royalty e validi in tutto il mondo ad utilizzare la specifica cXML (la "Specifica") con qualsiasi copyright della Ariba incluso nella Specifica per utilizzare, copiare, pubblicare, modificare e distribuire la Specifica. La Ariba garantisce inoltre una licenza esente da royalty in conformità ai diritti di proprietà intellettuale applicabili della Ariba ad implementare ed utilizzare le direttive relative a schemi e tag cXML incluse nella Specifica, al fine di creare programmi informatici in ottemperanza a tali direttive. Una condizione della presente licenza sarà l'accettazione da parte dell'utente di non rivendicare alcun diritto di proprietà intellettuale nei confronti della Ariba o di altre compagnie per l'implementazione della Specifica. La Ariba si riserva espressamente tutti gli altri diritti che potrebbe avere sul materiale e sul contenuto della Specifica. La Ariba nega espressamente qualsiasi garanzia riguardo alla Specifica, incluse la garanzia che la Specifica o l'implementazione di questa non violino i diritti di altri. La Specifica viene fornita nello stato in cui si trova, senza garanzie espresse o implicite. In caso di pubblicazione, copia o distribuzione della Specifica, l'utente è tenuto ad allegare questa nota sul copyright; tuttavia, se la Specifica viene modificata, il nome della specifica modificata non può includere il termine "cXML". Qualora l'utente presenti alla Ariba commenti e suggerimenti e la Ariba modifichi il cXML in base a tali informazioni, la Ariba sarà proprietaria della versione modificata del cXML.

Le informazioni contenute in questo documento sono soggette a cambiamento senza preavviso.

Sommario

Prefazione	v
Pubblico di lettori e prerequisiti	v
Quali capitoli leggere	v
 Capitolo 1	
Introduzione al linguaggio cXML	1
Caratteristiche del cXML	1
Cataloghi	2
PunchOut	3
Ordini d'acquisto	4
Tipi di applicazione che utilizzano il cXML	5
Applicazioni di acquisizione	5
Piattaforme di rete per il commercio	5
Cataloghi PunchOut	6
Sistema di ricezione degli ordini	6
Strategia per la distribuzione dei contenuti	6
Convalida basata sulle DTD	7
Come ottenere le DTD cXML	8
Esecuzione della convalida	8
Transazione Profile	8
Utilità XML	9

Capitolo 2

Implementazione di un sito PunchOut	11
Requisiti per il PunchOut	11
Società acquirenti	12
Fornitori	13
Sequenza degli eventi PunchOut	16
Fasi 1 e 2: richiesta PunchOut	16
Fase 3: scelta dei prodotti	17
Fase 4: estrazione	18
Fase 5: trasmissione di un ordine d'acquisto	19
Documenti PunchOut	21
Indice del catalogo PunchOut	21
PunchOutSetupRequest	23
PunchOutSetupResponse	27
PunchOutOrderMessage	28
Modifica delle pagine web	30
Pagina di apertura	31
Pagina di avvio	35
Pagina di invio	35
Pagina di ricezione	38
Suggerimenti per i siti web PunchOut	39
Consigli per l'implementazione	39
Cookie del compratore e del fornitore	40
Personalizzazione	40

Capitolo 3

Ricezione degli ordini d'acquisto cXML	43
Il processo degli ordini d'acquisto	43
Ricezione degli ordini d'acquisto	44
OrderRequest	44
OrderResponse	47
Accettazione degli allegati agli ordini	47

Appendice A	
Specifica del linguaggio cXML	49
Specifica del protocollo	50
Modello richiesta-risposta	50
Convenzioni XML	52
Elemento cXML envelope	52
Livelli di codifica	54
Header	57
Request	59
Response	59
Modello unilaterale (asincrono)	62
Elementi fondamentali	67
Entità tipo	67
Elementi di base	68
Transazione Profile	69
ProfileRequest	69
ProfileResponse	70
Definizione degli ordini	71
OrderRequest	71
Risposta a un documento OrderRequest	80
Transazione PunchOut	80
PunchOutSetupRequest	80
PunchOutSetupResponse	83
PunchOutOrderMessage	84
Modifiche successive allo stato	88
DocumentReference	88
StatusUpdateRequest	89
Definizioni per i cataloghi	90
Supplier	91
Index	93
Contract	95
Definizioni per la gestione degli abbonamenti	96
Dati del fornitore	96
Abbonamenti ai cataloghi	100
Definizioni per il recupero dei messaggi	103
GetPendingRequest	103
GetPendingResponse	104

Appendice B

Nuove funzioni del cXML 1.1105

Modifiche generali al cXML	105
Supporto multilingue avanzato	105
DTD centralizzate	106
Nuova transazione Profile	106
Nuovi codici di stato	107
Nuovo attributo type per i membri di un mercato	107
Modifiche agli elementi Extrinsic.	108
Nuovo elemento Contact	108
Supporto dell'attributo requisitionID.	109
Riepilogo delle variazioni apportate agli elementi Extrinsic. . .	110
Elementi Extrinsic a livello di intestazione	110
Miglioramenti delle transazioni PunchOut.	111
Miglioramenti del documento PunchOutSetupRequest	111
Elemento SelectedItem	111
Documenti PunchOutOrderMessage vuoti	112
Nuovo campo nascosto cXML-base64	112
Nuove caratteristiche degli ordini d'acquisto	113
Nuovo attributo lineNumber	113
Elementi Attachment dell'ordine d'acquisto	113
Nuovo attributo shipComplete.	114
Nuovo elemento ShortName	114
Nuova transazione di stato degli ordini d'acquisto	115
Nuovo elemento OrderReference	115
Nuova transazione StatusUpdateRequest	116
Nuovo elemento Followup.	116

Indice 117

Prefazione

Questo documento spiega come utilizzare il linguaggio cXML (commerce eXtensible Markup Language) per la comunicazione dei dati relativi al commercio elettronico.

Pubblico di lettori e prerequisiti

Questo documento si rivolge ai programmatori che devono creare applicazioni abilitate al cXML. Inoltre, si rivolge ai fornitori che intendono modificare i propri siti web per l'e-commerce per abilitarli al PunchOut.

Il cXML è un linguaggio aperto e versatile, adatto ai requisiti delle transazioni relative a:

- Cataloghi elettronici di prodotti
- Cataloghi PunchOut in cXML
- Applicazioni di acquisizione
- Comunità acquirenti

I lettori dovrebbero avere una conoscenza pratica dei concetti dell'e-commerce e dello standard HTTP per le comunicazioni sul web.

Il documento non descrive l'uso di particolari applicazioni di acquisizione o degli hub di rete per l'e-commerce.

Quali capitoli leggere

- **Dirigenti di aziende e-commerce:** per una panoramica delle caratteristiche del cXML, leggere Capitolo 1, "Introduzione al linguaggio cXML".
- **Programmatore web:** i programmatori web che devono implementare siti e-commerce dovrebbero leggere tutti i capitoli.

- **Amministratori di siti PunchOut:** i tecnici web esperti di siti web PunchOut dovrebbero leggere Appendice B, "Nuove funzioni del cXML 1.1".

Capitolo 1

Introduzione al linguaggio cXML

In questo capitolo viene descritto il linguaggio cXML (commerce eXtensible Markup Language) per le transazioni del commercio elettronico.

In questo capitolo viene fornita una panoramica del cXML e vengono trattati i seguenti argomenti:

- Caratteristiche del cXML
- Tipi di applicazione che utilizzano il cXML
- Strategia per la distribuzione dei contenuti
- Convalida basata sulle DTD
- Transazione Profile
- Utilità XML

Caratteristiche del cXML

Il cXML permette a compratori, fornitori, aggregatori e intermediari di comunicare tramite un unico linguaggio, aperto e standardizzato.

Il successo dei portali del commercio elettronico business-to-business (e-commerce B2B) dipende dall'uso di un protocollo flessibile e ampiamente diffuso. Il cXML è la chiave per offrire il massimo accesso a prodotti e servizi, poiché è un potente linguaggio ad alta definizione, progettato appositamente per il commercio elettronico business-to-business, ed è la scelta di un elevato numero di compratori e fornitori.

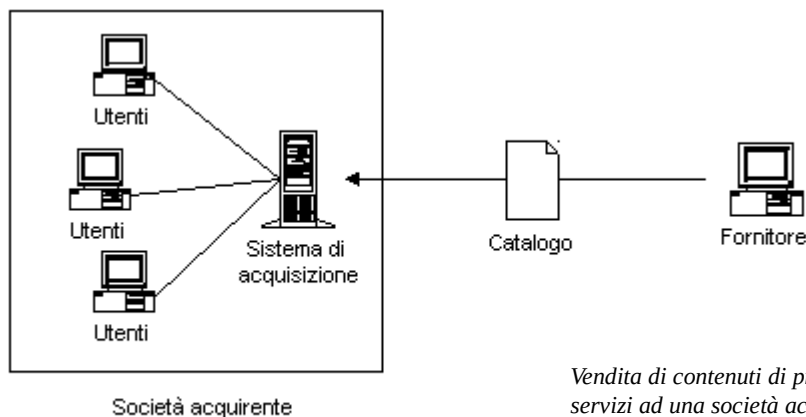
Le transazioni cXML sono costituite da *documenti*, ovvero da semplici file di testo caratterizzati da contenuti e formato ben definiti. La maggior parte dei tipi di documenti cXML sono simili ai documenti cartacei tradizionali, normalmente utilizzati nelle attività commerciali.

I principali tipi di documenti cXML sono descritti nelle sezioni riportate di seguito.

Cataloghi

I cataloghi sono file di testo utilizzati per trasmettere alle società acquirenti determinati prodotti e servizi. I cataloghi descrivono i prodotti e i servizi offerti e i relativi prezzi e sono il principale canale di comunicazione tra il fornitore e il cliente.

I cataloghi permettono alle aziende che utilizzano applicazioni di acquisizione di esaminare i prodotti e servizi offerti dai fornitori e acquistare quelli desiderati. Le applicazioni di acquisizione leggono i cataloghi e li memorizzano nei loro database. Una volta che la società acquirente ha approvato il catalogo di un fornitore, il contenuto del catalogo può essere visto dagli utenti, che possono scegliere gli articoli e aggiungerli alle proprie richieste di acquisto.



È possibile creare cataloghi per qualsiasi tipo di prodotto e servizio, indifferentemente dai metodi di determinazione delle quantità o dei prezzi e dal tipo di consegna.

Per ogni articolo, il catalogo riporta le informazioni di base e le altre informazioni opzionali che consentono l'uso delle funzioni di catalogo avanzate, quali le descrizioni in più lingue.

PunchOut

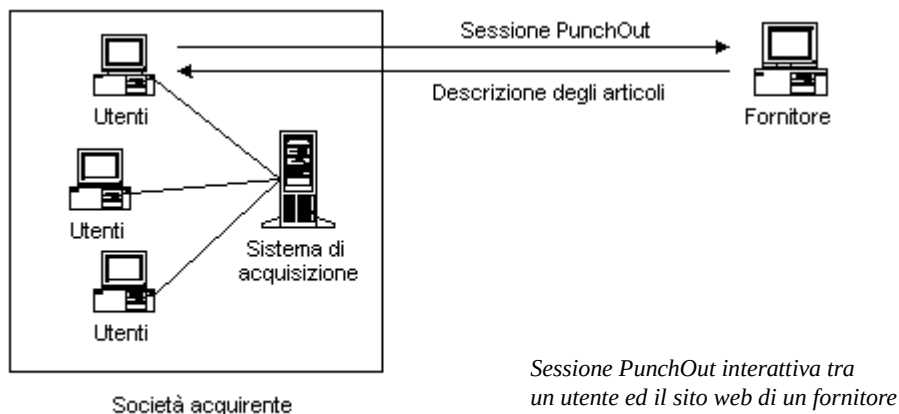
Il PunchOut offre un'alternativa ai file di catalogo statici. I siti PunchOut sono cataloghi dal vivo, interattivi, che vengono utilizzati sul sito web.

Chi dispone già di un sito web per l'e-commerce può apportarvi le modifiche necessarie affinché supporti il PunchOut. I siti PunchOut comunicano con i sistemi di acquisizione tramite Internet, utilizzando il linguaggio cXML.

Per ulteriori
informazioni:

Capitolo 2,
"Implementazione di
un sito PunchOut".

Nei siti PunchOut, le applicazioni di acquisizione visualizzano un pulsante al posto del prodotto o dei dettagli sul prezzo. Quando un utente fa clic su questo pulsante, il suo browser web visualizza le pagine del sito web locale. In base a come sono state implementate le pagine, l'utente può sfogliare i vari prodotti, specificare configurazioni e scegliere i metodi di consegna. Una volta terminate le operazioni, l'utente può fare clic su un pulsante che restituisce le informazioni all'applicazione di acquisizione. I prodotti configurati e i relativi prezzi vengono visualizzati nella richiesta di acquisto dell'utente.



Un sito web può offrire prodotti e prezzi precedentemente concordati in un contratto.

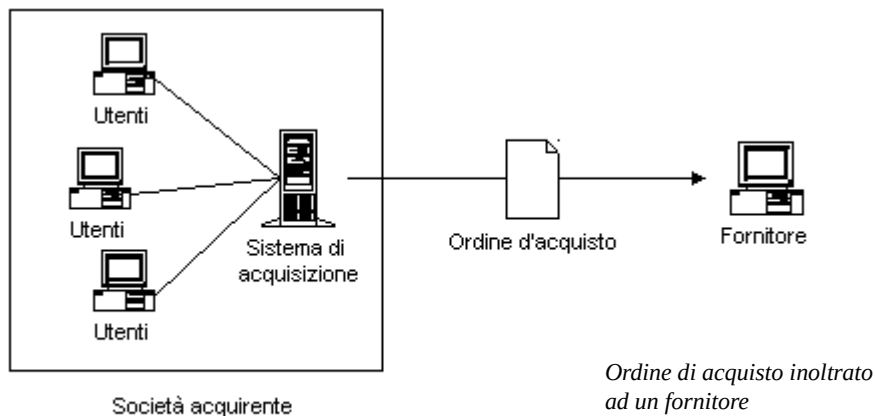
Ordini d'acquisto

Le società acquirenti inviano gli ordini d'acquisto ai fornitori per richiedere l'adempimento di un contratto.

Gli ordini possono essere instradati ai fornitori tramite una piattaforma di rete per l'e-commerce quale, ad esempio, Ariba Network.

Per ulteriori informazioni:

Capitolo 3, "Ricezione degli ordini d'acquisto cXML".



Il cXML è solo uno dei vari formati di ordine d'acquisto che possono essere utilizzati. Gli altri formati includono posta elettronica, fax ed EDI (X.12 Electronic Data Interchange). Tuttavia, il cXML è il formato ideale per gli ordini d'acquisto, in quanto è flessibile ed economico da implementare, supporta la maggiore gamma di dati e allegati e, in più, può essere utilizzato dai sistemi automatizzati.

Tipi di applicazione che utilizzano il cXML

Il cXML può essere utilizzato da qualsiasi applicazione e-commerce. Attualmente viene utilizzato dalle società acquirenti, dalle comunità di compratori orizzontali e verticali, dai fornitori e dai vendor di applicazioni.

I principali tipi di applicazioni che utilizzano il cXML sono descritte nelle sezioni riportate di seguito.

Applicazioni di acquisizione

Applicazioni di acquisizione, quali Ariba ORMS (Operating Resource Management System) e Ariba IBX (Internet Business eXchange), utilizzano il cXML per le transazioni esterne.

Ariba ORMS è un'applicazione di tipo enterprise che le società di grandi dimensioni mettono a disposizione dei propri dipendenti su una intranet aziendale.

Ariba IBX è un servizio basato su Internet che permette la creazione di comunità di compratori composte da società di piccole e medie dimensioni.

Queste applicazioni permettono alle comunità di utenti di acquisire prodotti e servizi personalizzati dai fornitori approvati dai propri responsabili per gli acquisti. Il processo è il seguente: le richieste di acquisto vengono approvate dai dirigenti delle comunità, gli ordini d'acquisto approvati vengono trasmessi ai fornitori utilizzando i vari canali disponibili, incluso il cXML su Internet.

Piattaforme di rete per il commercio

Le piattaforme di rete per il commercio, quale Ariba Network, sono servizi basati sul web che collegano compratori e fornitori.

Questi servizi web offrono funzioni quali la convalida dei cataloghi e la gestione dei file, la pubblicazione dei cataloghi e le sottoscrizioni agli abbonamenti, l'instradamento automatico degli ordini d'acquisto e la generazione della cronologia degli ordini d'acquisto.

Le comunicazioni tra questi servizi web, le applicazioni dei compratori e quelle dei fornitori possono avvenire interamente tramite cXML su Internet.

Cataloghi PunchOut

Per ulteriori informazioni:

Capitolo 2, "Implementazione di un sito PunchOut".

Come descritto nelle precedenti sezioni, i cataloghi PunchOut sono cataloghi interattivi, disponibili sui siti web dei fornitori. Questi cataloghi sono applicazioni per server web, scritte in un linguaggio di programmazione quale ASP (Active Server Pages), JavaScript o CGI, che gestisce le sessioni PunchOut del compratore.

I cataloghi PunchOut accettano le richieste PunchOut delle applicazioni di acquisizione, identificano la società acquirenti e visualizzano i prodotti e i prezzi appropriati in formato HTML. A questo punto, gli utenti possono selezionare gli articoli, configurarli e selezionare le opzioni desiderate.

Alla fine della sessione di PunchOut, il sito PunchOut invia le descrizioni delle scelte degli utenti, in formato cXML, alle applicazioni di acquisizione.

Sistema di ricezione degli ordini

Per ulteriori informazioni:

Capitolo 3, "Ricezione degli ordini d'acquisto cXML".

I sistemi di ricezione degli ordini sono applicazioni presenti sui siti dei fornitori, che accettano ed elaborano gli ordini d'acquisto inviati dalle società acquirenti. Tali sistemi possono essere costituiti da un qualsiasi sistema automatizzato, quale un sistema per la gestione dell'inventario oppure per l'evasione o l'elaborazione degli ordini.

Poiché l'estrazione delle informazioni dagli ordini d'acquisto cXML è molto semplice, è relativamente semplice anche creare gli adattatori che permettono ad un sistema di ricezione degli ordini di accettarle.

Strategia per la distribuzione dei contenuti

Le applicazioni di acquisizione presentano contenuti di prodotti e servizi agli utenti. I fornitori desiderano controllare il modo in cui i clienti possono visualizzare i loro prodotti e servizi, poiché la presentazione è di fondamentale importanza nei processi di vendita. Le società acquirenti, da parte loro, desiderano contenuti di facile accesso e consultazione per assicurare la massima conformità ai contratti.

Le società acquirenti e i fornitori possono scegliere da diversi metodi di distribuzione dei contenuti di prodotti e servizi. Il metodo da utilizzare dipende dall'accordo tra queste società e i fornitori e dalla natura dei prodotti e dei servizi trattati.

Nella tabella riportata di seguito vengono mostrati alcuni esempi di categorie di prodotti e servizi di tipo comune ed i relativi metodi di distribuzione.

Prodotto	Proprietà	Metodo di distribuzione del contenuto
Forniture per ufficio, Forniture interne	Contenuto statico, prezzo stabile	Cataloghi statici
Forniture per laboratorio, MRO (Maintenance, Repair, and Operations), Parti elettroniche	Per un uso ottimale, è richiesta la normalizzazione	PunchOut per un portale di prodotti verticale
Libri, Prodotti chimici	Numero elevato di articoli	PunchOut per un sito ospitato di un fornitore
Computer, Attrezzature di rete, Periferiche	Vasta scelta di configurazioni	PunchOut per uno strumento di configurazione di un fornitore
Servizi, Stampe	Contenuto ricco di attributi variabili	PunchOut per un modulo elettronico presso il sito di un fornitore

Le società acquirenti possono memorizzare il contenuto localmente, all'interno della società, oppure possono accedervi in remoto su Internet, tramite PunchOut. I cataloghi cXML supportano entrambi questi tipi di strategie.

Come mostrato nella tabella sopra riportata, il PunchOut offre una struttura flessibile su cui i fornitori, a seconda del prodotto o del cliente, possono offrire un contenuto personalizzato. L'obiettivo di questa strategia per i contenuti è di consentire a compratori e fornitori di scambiarsi i dati di catalogo nel modo più idoneo.

Convalida basata sulle DTD

Poiché il cXML è un linguaggio XML, è accuratamente definito da un gruppo di DTD (Document Type Definitions). Queste DTD sono dei file di testo che descrivono con precisione la sintassi e l'ordine degli elementi cXML. Le DTD permettono alle applicazioni di convalidare il cXML da leggere o scrivere.

La convalida dei documenti cXML da parte delle applicazioni cXML non è obbligatoria, ma è consigliabile.

Come ottenere le DTD cXML

Le DTD per tutte le versioni del cXML sono disponibili in diverse ubicazioni del sito cxml.org:

`http://xml.cXML.org/schemas/cXML/<versione>/cXML.dtd`

dove `<versione>` è il numero completo della versione del cXML;
ad esempio, 1.1.007.

Esecuzione della convalida

Le applicazioni possono utilizzare le DTD per convalidare i documenti cXML in entrata e in uscita. Le applicazioni per la convalida dell'XML sono disponibili sul web. Microsoft Internet Explorer 5 dispone di una funzione di convalida XML incorporata.

Per una gestione ottimale delle transazioni, è consigliabile convalidare tutti i documenti cXML ricevuti. Se vengono rilevati errori, è necessario emettere il codice di errore appropriato, per consentire al mittente di ritrasmettere il documento.

Per prestazioni ottimali, i client cXML non dovrebbero caricare le DTD ogni volta che analizzano i documenti cXML. Dovrebbero, piuttosto, controllare il numero di versione cXML nelle intestazioni dei documenti e recuperare le DTD che non sono state ancora memorizzate in locale.

Transazione Profile

La transazione Profile comunica le informazioni di base sui server cXML. Questa transazione è l'unica che deve essere supportata da tutti i server cXML.

Questa transazione consiste di due documenti, ProfileRequest e ProfileResponse. Questi due documenti recuperano le caratteristiche del server, inclusa la versione cXML supportata, le transazioni supportate e le opzioni per queste transazioni.

Nota: tutti i server cXML 1.1 **devono** supportare la transazione Profile.

I client possono utilizzare la transazione Profile per "effettuare il ping" dei server e verificare se sono disponibili.

ProfileRequest

Il documento ProfileRequest è privo di contenuto e viene instradato al server cXML specificato.

ProfileResponse

Il server risponde con un documento ProfileResponse, che elenca le transazioni supportate dal server cXML, la loro ubicazione e le eventuali opzioni, utilizzando un valore in formato di stringa.

Utilità XML

Le utilità per le operazioni di modifica e convalida dei file XML possono essere acquistate o scaricate gratuitamente dal web. L'elenco riportato di seguito descrive alcune di queste utilità:

- **Internet Explorer 5** della Microsoft. Un browser web che riconosce l'XML e può convalidare i file XML utilizzando le DTD.

www.microsoft.com/windows/ie/default.htm

- **XML Notepad** della Microsoft. Un semplice editor XML.

msdn.microsoft.com/xml/notepad/intro.asp

- **XML Authority** della Extensibility. Un editor DTD XML basato su Java, con viste grafiche e gerarchiche.

www.extensibility.com

- **XML Spy** della Icon Information Systems. Uno strumento per gestire DTD e file XML, con una vista a griglia, del codice sorgente o formattata per il browser.

www.icon-is.com

- **XMetaL** della Softquad Software. Uno strumento di creazione XML personalizzabile.

www.softquad.com

- **CLIP** della Techno2000 USA. Uno strumento di creazione XML facile da utilizzare, con istruzioni di guida per la modifica.

www.t2000-usa.com

- **XMLwriter** della Wattle Software. Uno strumento di creazione XML grafico, progettato per la gestione dei progetti XML.

www.xmlwriter.net

Oltre a quelli citati, altri strumenti XML sono disponibili sui seguenti siti web:

www.xmlsoftware.com

www.xml.com/pub/pt/Editors

Capitolo 2

Implementazione di un sito PunchOut

Il PunchOut permette agli utenti delle applicazioni di acquisizione di accedere ai listini prezzi di prodotti e servizi dei fornitori che risiedono su un sito web. In questo modo, non è necessario spedire interi cataloghi alle società acquirenti, ma è sufficiente inviare piccoli file che contengono la descrizione del materiale offerto, le categorie dei prodotti o i prodotti.

In questo capitolo viene spiegato come modificare un sito web affinché supporti il PunchOut e vengono trattati i seguenti argomenti:

- Requisiti per il PunchOut
- Sequenza degli eventi PunchOut
- Documenti PunchOut
- Modifica delle pagine web
- Suggerimenti per i siti web PunchOut

Requisiti per il PunchOut

Prima che le società acquirenti configurino le applicazioni di acquisizione per il PunchOut, o che i fornitori implementino siti web PunchOut, è necessario che entrambe le parti valutino attentamente i vantaggi e i requisiti del PunchOut.

Società acquirenti

L'impostazione e la verifica delle applicazioni di acquisizione compatibili con il cXML insieme ad un fornitore abilitato al PunchOut possono essere effettuate in meno di un giorno.

Poiché le barriere all'integrazione tecnica da parte delle società acquirenti sono decisamente basse, la decisione di utilizzare il PunchOut dovrebbe basarsi sulle prassi aziendali e sui tipi di beni acquistati. Per l'elenco dei beni che meglio si prestano all'uso del PunchOut, vedere "Strategia per la distribuzione dei contenuti" a pagina 6.

Problematiche di tipo aziendale

Le società acquirenti devono tenere presente quanto segue:

- Gli utenti che effettuano le richieste e quelli che le approvano dispongono dell'accesso a Internet? In caso negativo, sarebbe possibile concedere loro un accesso controllato?
- Le società acquirenti desiderano che i fornitori creino e gestiscano contenuti di catalogo (inclusi i prezzi)?
- Gli utenti che effettuano le richieste acquistano beni su Internet? In caso affermativo, questi beni richiedono uno strumento di configurazione da parte del fornitore o contengono attributi univoci che non possono conformarsi ad un modello statico di contenuti?
- La società acquirente utilizza aggregatori di contenuti per cataloghi (ad esempio, Aspect, TPN Register o Harbinger)?
- La società acquirente acquista servizi (ad esempio, consulenze, assistenza temporanea o manutenzione) via Internet?
- La società acquirente si rifornisce in linea?

Se la risposta ad una qualsiasi delle domande sopra esposte è sì, il PunchOut è la soluzione appropriata per la società acquirente.

Problematiche di tipo tecnico

Le società acquirenti devono soddisfare i seguenti requisiti tecnici:

- Accesso diretto a Internet: gli utenti all'interno delle società acquirenti devono disporre dell'accesso diretto a Internet. Il PunchOut si basa su sessioni normali dei browser web, in cui l'utente interagisce con i siti web dei fornitori. Queste comunicazioni hanno luogo tramite una normale infrastruttura intranet/Internet e non tramite l'applicazione di acquisizione.
- Connessione affidabile a Internet: l'accesso a Internet deve essere sempre operativo e affidabile. Se gli utenti non possono acquistare i prodotti a causa della mancata disponibilità di Internet, è probabile che gli acquisti effettuati siano inaffidabili.
- Listini prezzi concordati con i fornitori PunchOut: gli agenti addetti agli acquisti devono avere concordato dei listini prezzi con i fornitori PunchOut. I siti web PunchOut permettono l'accesso solo a società acquirenti conosciute e autorizzate.

Fornitori

Il termine *fornitore* nel contesto del PunchOut è più ampio rispetto alla sua normale accezione. Il protocollo PunchOut è stato progettato come struttura flessibile in grado di trasmettere dati su quasi ogni tipo di prodotto o servizio offerto da qualsiasi fornitore, distributore, aggregatore o produttore.

Tra i prodotti e i servizi inclusi ci sono:

- Computer offerti direttamente dal produttore o da un rivenditore
- Sostanze chimiche e reagenti offerti da un aggregatore
- Forniture per ufficio offerte da un distributore
- Contratti di assistenza offerti da un'agenzia

Alcuni fornitori dispongono di un sito web transattivo, in grado di ospitare contenuti e ricevere ordini d'acquisto. Chi dispone di questa capacità e desidera implementare il PunchOut deve tenere presente sia le proprie prassi aziendali che le proprie risorse tecniche.

Problematiche di tipo aziendale

I fornitori devono tenere presente quanto segue:

- Il fornitore vende prodotti o servizi tramite Internet? In caso affermativo, offre ai clienti contenuti personalizzati sul proprio sito web?
- I prodotti e i servizi del fornitore rientrano in una delle categorie PunchOut descritte nel grafico contenuto in "Strategia per la distribuzione dei contenuti" a pagina 6. Per riassumere, queste categorie includono:
 - Prodotti altamente configurabili (quali i computer)
 - Molti articoli di una linea (quali i libri)
 - Attributi di prodotti univoci (quali i prodotti chimici)
 - Dati normalizzati (quali le forniture MRO)
- Il fornitore preferisce ricevere gli ordini d'acquisto e/o i pagamenti tramite il sito web?

Se la risposta ad una qualsiasi delle domande sopra esposte è sì, il PunchOut è la soluzione appropriata per il fornitore.

Problematiche di tipo tecnico

I fornitori devono soddisfare i seguenti requisiti tecnici:

- Connessione affidabile a Internet: l'infrastruttura dei server web e la connessione a Internet devono essere estremamente affidabili. Se gli utenti non possono accedere ad un contenuto remoto, è probabile che si rivolgano ad un altro fornitore.
- Amministratori competenti del sito web: il sito web PunchOut e le applicazioni di supporto richiedono una manutenzione e una modifica periodiche. Le esigenze degli utenti e le offerte di prodotti possono cambiare, quindi è necessario disporre di personale competente per modificare l'infrastruttura PunchOut.
- Supporto delle transazioni di base: i siti web PunchOut non devono necessariamente supportare tutte le funzioni cXML, ma solo le transazioni riportate di seguito:
 - Transazione Profile
 - PunchOutSetupRequest
 - PunchOutSetupResponse
 - PunchOutOrderMessage

Stima delle attività

Nella tabella riportata di seguito viene mostrata la stima delle attività richieste per integrare il PunchOut cXML, sulla base delle stime dei fornitori:

Livello dell'infrastruttura preesistente	Tempo stimato per il completamento
Sito transattivo con infrastruttura XML	3 settimane con staff IT interno 3-4 settimane con fornitori esterni
Sito transattivo privo di infrastruttura XML	4 settimane con staff IT interno 4-5 settimane con fornitori esterni

Il linguaggio XML

Il primo passo per cominciare ad utilizzare il PunchOut è capire l'XML. L'XML è un linguaggio che descrive i linguaggi. I documenti cXML sono basati sulle DTD (Document Type Definitions) XML. Le DTD sono dei modelli che possono essere utilizzati per definire i modelli di contenuto all'interno dei documenti cXML (ad esempio, il giusto ordine e la corretta nidificazione degli elementi) e i tipi di dati degli attributi.

Informazioni sull'XML

L'XML (eXtensible Markup Language) è uno standard per la trasmissione dei dati tra le varie applicazioni Internet. I documenti XML contengono dati in forma di coppie tag-valore. La struttura dell'XML è simile a quella dell'HTML (Hypertext Markup Language), anche se le applicazioni Internet possono estrarre e utilizzare i dati dei documenti XML con più facilità rispetto ai documenti HTML.

Con la diffusione delle applicazioni Internet, l'XML diventerà il linguaggio prevalente.

Per implementare un sito web PunchOut, è necessario sapere come creare, analizzare, eseguire query, ricevere e trasmettere i dati XML a/da una origine remota.

Gli strumenti di base per l'elaborazione dei documenti XML sono gli analizzatori XML. Questi analizzatori possono essere ottenuti gratuitamente dalla Microsoft e da altre società (ad esempio, un analizzatore XML è incluso in Microsoft Internet Explorer 5). Per un elenco degli strumenti XML, vedere "Utilità XML" a pagina 9.

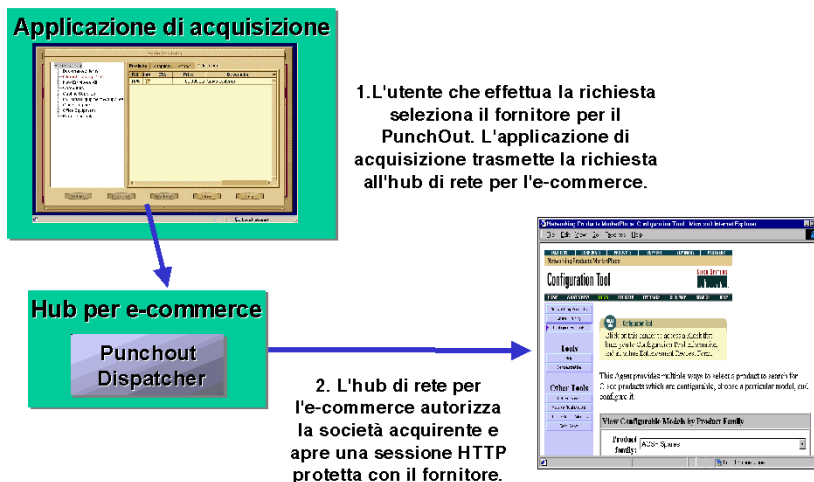
Sequenza degli eventi PunchOut

Una sessione PunchOut è costituita da una serie di fasi:

Fasi 1 e 2: richiesta PunchOut

Gli utenti eseguono il login ad un'applicazione di acquisizione e aprono nuove richieste di acquisto. Trovano gli articoli desiderati ricercandoli nei cataloghi locali in base al tipo di prodotto, al fornitore o alla descrizione del prodotto. Una volta selezionato un articolo PunchOut, l'applicazione di acquisizione apre una nuova finestra del browser e collega gli utenti ai rispettivi account sul sito web del fornitore.

Nella figura riportata di seguito vengono mostrate le fasi di una richiesta PunchOut:



Funzionamento della richiesta PunchOut Quando un utente fa clic su un articolo PunchOut, l'applicazione di acquisizione invia un documento cXML PunchOutSetupRequest ad un hub di rete per l'e-commerce. L'hub, in qualità di terza parte autorizzata, accetta la richiesta, verifica la società acquirente e passa la richiesta al sito web PunchOut del fornitore.

Nota: tutti i documenti cXML inviati tramite Internet possono viaggiare su connessioni sicure, quali le connessioni HTTPS con cifratura SSL (Secure Socket Layer) 3.0.

Lo scopo di questa richiesta è quello di notificare al sito web del fornitore l'identità del compratore e di comunicare l'operazione che deve essere eseguita. Tra le operazioni supportate, sono incluse le seguenti:

- **create:** Avvia una nuova sessione PunchOut
- **edit:** apre nuovamente una sessione PunchOut per consentirne la modifica.
- **inspect:** apre nuovamente una sessione PunchOut per consentirne l'ispezione (ma non la modifica dei dati)

Dopo avere ricevuto una richiesta, il sito web risponde con un documento PunchOutSetupResponse contenente un indirizzo Internet che indica all'applicazione di acquisizione da dove avviare la sessione del browser sul sito web del fornitore.

L'applicazione di acquisizione apre una nuova finestra del browser, che visualizza una sessione collegata ad un account sul sito web del fornitore. L'account può essere specifico di una regione, di un'azienda, di un reparto o di un utente.

Fase 3: scelta dei prodotti

Gli utenti scelgono gli articoli dell'inventario del fornitore utilizzando tutte le funzioni e i servizi disponibili sul sito web di quest'ultimo.



3. L'utente che effettua la richiesta utilizza il sito del fornitore per trovare e configurare i prodotti.

A seconda del tipo di prodotto o di cliente, le funzioni possono includere:

- Strumenti di configurazione per la creazione di prodotti personalizzati (ad esempio, computer, composti organici o prodotti personalizzati)
- Motori di ricerca per trovare i prodotti desiderati nei cataloghi di grandi dimensioni.
- Viste di dati normalizzati per confrontare i prodotti in base al prezzo, alle funzioni o alla disponibilità (ad esempio, i prodotti MRO).
- Viste degli attributi univoci di prodotti particolari (ad esempio, stampe, sostanze chimiche e reagenti, oppure servizi).

- Controllo in tempo reale dei prezzi, dell'inventario e della disponibilità
- Calcolo automatico di tasse e prezzi di trasporto, in base alla destinazione, alla dimensione o alla quantità di articoli per la spedizione (non necessario durante la sessione PunchOut).

Funzionamento della scelta dei prodotti Una volta che l'applicazione di acquisizione ha indirizzato gli utenti al sito web del fornitore, questi possono effettuare gli acquisti allo stesso modo di quando accedono direttamente al sito web del fornitore, pertanto non è necessario modificare nessuno dei servizi o delle funzioni sopra elencati.

Fase 4: estrazione

Il sito web calcola il costo totale degli articoli scelti dagli utenti, includendo tasse, spese di spedizione e sconti specifici per clienti. A questo punto, gli utenti fanno clic sul pulsante "Estrai" del sito web per inviare il contenuto del carrello della spesa alle richieste di acquisto all'interno dell'applicazione di acquisizione.

Nella figura riportata di seguito vengono mostrate le fasi di un'estrazione:



Come funziona l'estrazione Quando gli utenti fanno clic sul pulsante "Estrai", il sito web del fornitore invia all'applicazione di acquisizione un documento cXML PunchOutOrderMessage contenente i dettagli sui prodotti e i prezzi. Il fornitore può anche inviare dei cookie nascosti che, in seguito, possono associare gli articoli a specifiche sessioni di acquisto.

Di fatto, il fornitore ha inviato un preventivo per gli articoli richiesti; non ha ancora ricevuto un ordine d'acquisto, quindi non può registrare l'ordine.

Se, in seguito, gli utenti hanno bisogno di modificare un articolo contenuto nella richiesta di acquisto, il fornitore può consentire loro di rieseguire il PunchOut al suo sito web. L'applicazione di acquisizione restituisce il contenuto originario del carrello della spesa al sito web, affinché gli utenti possano modificarlo. Dopo l'estrazione, il sito web restituisce gli articoli alla richiesta di acquisto.

Il sito web del fornitore è la fonte di informazioni per tutti gli articoli PunchOut. Le modifiche alla quantità o l'aggiunta di altri articoli alla richiesta di acquisto possono influire sulle tasse o sulle spese di spedizione, quindi richiedono che il calcolo venga nuovamente eseguito sul sito web. Per questo motivo, le modifiche agli articoli originari devono essere apportate sul sito web e non nell'applicazione di acquisizione. Da qui, l'esigenza di eseguire nuovamente il PunchOut. Per eseguire nuovamente il PunchOut è sufficiente inviare una richiesta PunchOutSetupRequest specificando "edit" come tipo di operazione.

Fase 5: trasmissione di un ordine d'acquisto

Una volta che il contenuto del carrello della spesa è stato passato dal sito web alla richiesta di acquisto dell'utente, ha inizio il processo di approvazione da parte dell'applicazione di acquisizione. Quando la richiesta di acquisto viene approvata, l'applicazione la converte in un ordine d'acquisto e la restituisce al sito web affinché venga evasa. È possibile trasmettere i dati sulla carta d'acquisto insieme all'ordine oppure emettere la fattura relativa all'ordine al momento della spedizione.

Nella figura riportata di seguito viene mostrata la trasmissione di un ordine d'acquisto:



Funzionamento della trasmissione di un ordine d'acquisto L'applicazione di acquisto invia tutti gli ordini d'acquisto all'hub per l'e-commerce in formato cXML. L'hub, a questo punto, instrada gli ordini al fornitore, utilizzando il metodo scelto da quest'ultimo. Quando il fornitore conferma la ricezione dell'ordine d'acquisto, di fatto registra l'ordine.

Per i fornitori PunchOut, il metodo di instradamento degli ordini dovrebbe essere il cXML, per i seguenti motivi:

- Gli ordini d'acquisto cXML fanno sì che i cookie incorporati del fornitore possano essere ritrasmessi a quest'ultimo. Poiché un cookie può contenere qualsiasi tipo di dati, è possibile che gli altri metodi di instradamento, quali fax, posta elettronica ed EDI, non eseguano facilmente il relativo mapping.
- I fornitori PunchOut supportano già lo standard cXML, quindi l'accettazione degli ordini d'acquisto cXML non è particolarmente onerosa.

Per una descrizione dettagliata degli ordini d'acquisto, vedere Capitolo 3, "Ricezione degli ordini d'acquisto cXML."

Documenti PunchOut

Sono disponibili quattro tipi di documenti cXML:

- Indice del catalogo PunchOut
- PunchOutSetupRequest
- PunchOutSetupResponse
- PunchOutOrderMessage

Indice del catalogo PunchOut

Gli indici dei cataloghi PunchOut sono file che elencano gli articoli PunchOut e puntano ad un sito web PunchOut.

Nell'esempio riportato di seguito viene mostrato un indice del catalogo PunchOut:

<i>Tipo di documento cXML e indirizzo Internet della DTD</i>	<code><?xml version="1.0" encoding="UTF-8"?></code>
	<code><!DOCTYPE Index SYSTEM "http://xml.cxml.org/schemas/cXML/1.10,007/cXML.dtd"></code>
	<code><Index></code>
	<code> <SupplierID domain="DUNS">83528721</SupplierID></code>
	<code> <IndexItem></code>
	<code> <IndexItemPunchout></code>
	<code> <ItemID></code>
<i>L'identificativo assegnato dal fornitore all'articolo PunchOut</i>	<code> <SupplierPartID>5555</SupplierPartID></code>
	<code> </ItemID></code>
	<code> <PunchoutDetail></code>
	<code> <Description xml:lang="en-US">Desk Chairs</Description></code>
	<code> <Description xml:lang="fr-FR">Chaises de Bureau</Description></code>
<i>Indirizzo Internet del sito web PunchOut (pagina di apertura)</i>	<code> <URL>http://www.workchairs.com/punchout.asp</URL></code>
	<code> <Classification domain="UNSPSC">5136030000</Classification></code>
	<code> </PunchoutDetail></code>
	<code> </IndexItemPunchout></code>
	<code> </IndexItem></code>
	<code></Index></code>

SupplierID identifica la società del fornitore. È possibile utilizzare qualsiasi dominio di identificazione, ma quelli consigliati sono DUNS (Dun & Bradstreet Universal Naming System) e NetworkID. Per ulteriori informazioni sui numeri DUNS, vedere www.dnb.com.

Description specifica il testo che l'applicazione di acquisizione visualizza nei cataloghi dei prodotti. Se la descrizione è fornita in più lingue, l'applicazione di acquisizione visualizza la versione indicata dal codice per la lingua dell'utente.

Classification specifica il gruppo di prodotti a cui appartiene l'articolo. Tutti i prodotti e i servizi devono essere mappati e standardizzati secondo lo schema UNSPSC. Per gli indici dei cataloghi PunchOut, la classificazione determina la posizione dell'articolo PunchOut nei cataloghi visualizzati. Per un elenco dei codici UNSPSC, vedere www.unspsc.com.

Creazione e pubblicazione degli indici dei cataloghi

Il fornitore può creare questi cataloghi e metterli a disposizione dei propri clienti pubblicandoli su un hub per l'e-commerce. Il gestore dei cataloghi della società acquirente può scaricarli e memorizzarli per consentirne l'uso con le applicazioni di acquisizione.

Gli utenti vedono il contenuto degli indici dei cataloghi PunchOut a fianco di normali articoli di catalogo statici.

Livello di dettaglio degli articoli PunchOut

Il fornitore può creare cataloghi a livello di negozio, di settore o di prodotto.

- I cataloghi a livello di negozio riportano tutti i prodotti e i servizi offerti dal fornitore. Per trovare l'articolo desiderato, l'utente deve effettuare una ricerca.
- I cataloghi a livello di settore riportano prodotti e servizi correlati.
- I cataloghi a livello di prodotto elencano un solo tipo di prodotto o servizio. Per trovare l'articolo desiderato, l'utente non deve effettuare alcuna ricerca.

Per determinare il livello del catalogo degli articoli PunchOut, considerare fattori quali il proprio modello aziendale, l'aspetto dei prodotti e le offerte di servizio, nonché la struttura del sito web PunchOut.

Maggiore è la quantità di strumenti di ricerca e configurazione presenti sul sito web del fornitore, più elevato potrà essere il numero di articoli PunchOut negli indici dei cataloghi.

PunchOutSetupRequest

Per avviare una sessione PunchOut, l'utente seleziona un articolo PunchOut. L'applicazione di acquisizione genera un documento PunchOutSetupRequest e lo invia ad un hub per l'e-commerce, che lo inoltra al sito web PunchOut del fornitore.

Nell'esempio riportato di seguito viene mostrato un documento PunchOutSetupRequest:

		<pre><?xml version="1.0"?> <!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd"> <cXML version="1.1.007" xml:lang="en-US" payloadID="933694607118.1869318421@jlee" timestamp="2000-08-15T08:36:47- 07:00"> <Header> <From> <Credential domain="DUNS"> <Identity>65652314</Identity> </Credential> </From> <To> <Credential domain="DUNS"> <Identity>83528721</Identity> </Credential> </To> <Sender> <Credential domain="AribaNetworkUserId"> <Identity>sysadmin@ariba.com</Identity> <SharedSecret>abracadabra</SharedSecret> </Credential> <UserAgent>Ariba ORMS 6.1</UserAgent> </Sender> </Header> <Request> <PunchOutSetupRequest operation="create"> <BuyerCookie>1CX3L4843PPZO</BuyerCookie> <Extrinsic name="CostCenter">610</Extrinsic> <Extrinsic name="User">john_smith</Extrinsic> <BrowserFormPost> <URL>https://aribaorms:26000/punchout.asp</URL> </BrowserFormPost> <SupplierSetup> <URL>http://www.workchairs.com/punchout.asp</URL> </SupplierSetup> <SelectedItem> <ItemOut quantity="1"> <ItemID> <SupplierPartID>5555</SupplierPartID> </ItemID> </ItemOut> </SelectedItem> </PunchOutSetupRequest> </Request> </cXML></pre>
Autore (società acquirente)		
Destinatario (fornitore)		
Precedente entità di trasmissione (in questo caso, Ariba Network)		
Tipo di richiesta		
Destinazione del documento		
Articolo selezionato dall'utente		

```

        </SelectedItem>
      </PunchOutSetupRequest>
    </Request>
  </cXML>

```

Gli attributi payloadID e timestamp all'inizio vengono utilizzati dai client cXML per tenere traccia dei documenti e rilevare i documenti duplicati.

Gli elementi From, To e Sender permettono al sistema ricevente di identificare e autorizzare le parti. Gli elementi From e To di un documento non cambiano mentre, durante il viaggio che il documento compie per arrivare a destinazione, l'elemento Sender viene modificato dai nodi intermedi (ad esempio, Ariba Network).

Operazioni create, edit e inspect

L'attributo operation specifica il tipo di sessione avviata dal compratore, ovvero create, edit o inspect.

- Le sessioni create generano nuovi carrelli della spesa, che corrispondono a nuove richieste di acquisto.
- Le sessioni edit aprono nuovamente carrelli della spesa precedentemente creati per consentirne la modifica. L'applicazione di acquisizione invia i dati dell'articolo come parte del documento PunchOutSetupRequest. Il sito web PunchOut può utilizzare questi dati per creare una nuova istanza del carrello creato durante la sessione originaria.
- Le sessioni inspect aprono nuovamente i carrelli della spesa precedentemente creati per consentirne esclusivamente l'ispezione. Analogamente a quanto succede con l'operazione edit, l'applicazione di acquisizione invia i dati dell'articolo come parte del documento PunchOutSetupRequest. Tuttavia, dopo avere creato la nuova istanza del carrello della spesa, il sito web PunchOut non ne consente la modifica del contenuto.

Nell'esempio riportato di seguito viene mostrata una richiesta edit:

```

<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" xml:lang="en-US"
payloadID="933695135608.677295401@jlee" timestamp="2000-08-15T08:45:35-07:00">
  <Header>
    <From>
      <Credential domain="DUNS">
        <Identity>65652314</Identity>
      </Credential>
    </From>
    <To>
      <Credential domain="DUNS">

```



```

        <Identity>83528721</Identity>
      </Credential>
    </To>
    <Sender>
      <Credential domain="AribaNetworkUserId">
        <Identity>sysadmin@ariba.com</Identity>
        <SharedSecret>abracadabra</SharedSecret>
      </Credential>
      <UserAgent>Ariba ORMS 6.1</UserAgent>
    </Sender>
  </Header>
  <Request>
    <PunchOutSetupRequest operation="edit">
      <BuyerCookie>1CX3L4843PPZO</BuyerCookie>
      <Extrinsic name="CostCenter">610</Extrinsic>
      <Extrinsic name="User">john_smith</Extrinsic>
      <BrowserFormPost>
        <URL>https://aribaorms:26000/punchout.asp</URL>
      </BrowserFormPost>
      <SupplierSetup>
        <URL>http://www.workchairs.com/punchout.asp</URL>
      </SupplierSetup>
      <ItemOut quantity="2">
        <ItemID>
          <SupplierPartID>220-6338</SupplierPartID>
          <SupplierPartAuxiliaryID>E000028901
          </SupplierPartAuxiliaryID>
        </ItemID>
      </ItemOut>
    </PunchOutSetupRequest>
  </Request>
</cXML>

```

Se l'utente ha avviato la sessione edit selezionando un articolo di catalogo, il documento PunchOutSetupRequest dovrebbe contenere l'elemento SelectedItem, come per la sessione create.

Autenticazione da parte di un hub per l'e-commerce

Tutti i documenti PunchOutSetupRequest passano attraverso un hub per l'e-commerce che li deve autenticare e deve ricercare l'indirizzo Internet del sito web PunchOut del fornitore. Le fasi sono le seguenti:

1. L'hub riceve il documento PunchOutSetupRequest dall'utente.
2. L'hub verifica l'ID del compratore (From e Shared Secret) utilizzando l'account e-commerce del compratore. Inoltre, identifica il fornitore a cui deve essere inviata la richiesta (To).

3. L'hub ricerca la parola chiave (Shared Secret) nell'account del fornitore e la inserisce nell'elemento Sender.
4. L'hub localizza nell'account del fornitore l'indirizzo Internet del suo sito web PunchOut e invia il documento PunchOutSetupRequest all'indirizzo in questione.
5. Il sito web riceve il documento cXML e capisce che è autenticato poiché contiene la parola chiave del fornitore.
6. Il sito web utilizza le informazioni contenute nell'elemento From per identificare il richiedente a livello di azienda (ad esempio, acme.com).
7. Il fornitore può utilizzare le informazioni relative a Contact e i dati estrinseci contenuti nel corpo della richiesta per identificare in modo univoco l'utente (ad esempio, John Smith in Finance, presso acme.com).

I documenti PunchOutSetupRequest e PunchOutSetupResponse passano attraverso l'hub per l'e-commerce per l'autenticazione. Il documento PunchOutOrderMessage (che restituisce il contenuto del cestino della spesa all'applicazione di acquisizione) viaggia direttamente tra il sito web e l'utente tramite HTTPS o HTTP standard.

Indirizzo Internet dell'elemento SupplierSetup ed elemento SelectedItem

Nei precedenti release del cXML, l'elemento SupplierSetup forniva l'unico modo per specificare l'indirizzo Internet del sito web PunchOut. A partire dal release cXML 1.1, l'hub per l'e-commerce è in grado di conoscere già tale indirizzo.

Sempre a partire dal release cXML 1.1, le applicazioni di acquisizione possono utilizzare l'elemento SelectedItem per specificare il PunchOut a livello di negozio, settore o prodotto.

L'uso dell'elemento SupplierSetup non è più consigliabile; tuttavia, un sito web PunchOut deve gestire entrambi i metodi fino a quando il sito web PunchOut non riconosce l'elemento SelectedItem.

Dati degli elementi Contact e Extrinsic per l'identificazione degli utenti

Il documento PunchoutSetupRequest può contenere nell'elemento Contact informazioni dettagliate sull'utente che il sito web può utilizzare per autenticare e localizzare l'utente, quali:

- Nome dell'utente e ruolo ricoperto
- Indirizzo di posta elettronica

In più, il documento PunchOutSetupRequest può contenere anche i dati *estrinseci* che possono essere utilizzati per identificare gli utenti in modo ancora più preciso, quali:

- Centro di costo e account secondario dell'utente
- Area
- Supervisore
- Valuta predefinita

Le società acquirenti configurano le proprie applicazioni di acquisizione per l'inserimento dei dati estrinseci e dell'elemento Contact. Il fornitore deve chiedere ai propri clienti quali sono i dati che questi si aspettano di ricevere.

PunchOutSetupResponse

Dopo aver ricevuto un documento PunchOutSetupRequest, il sito web invia un documento PunchOutSetupResponse. Il documento PunchOutSetupResponse ha due diverse funzioni:

- Indica se la richiesta inviata con PunchOutSetupRequest ha avuto esito positivo.
- Fornisce all'applicazione di acquisizione l'indirizzo Internet della pagina di avvio del sito del fornitore.

Il documento contiene un elemento <URL> che specifica l'indirizzo Internet della pagina di avvio da passare al browser web dell'utente che deve avviare la sessione interattiva di esplorazione. Questo indirizzo deve contenere le informazioni sullo stato necessarie per collegarsi a una sessione sul sito web quali, ad esempio, l'identità del richiedente e il contenuto dell'elemento BuyerCookie.

Nell'esempio riportato di seguito viene mostrato un documento PunchOutSetupResponse:

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" xml:lang="en-US" payloadID="933694607739"
timestamp="2000-08-15T08:46:00-07:00">
  <Response>
    <Status code="200" text="success"></Status>
    <PunchOutSetupResponse>
      <StartPage>
        <URL>
          http://xml.workchairs.com/retrieve?reqUrl=20626;Initial=TRUE
        </URL>
      </StartPage>
    </PunchOutSetupResponse>
  </Response>
</cXML>
```

PunchOutOrderMessage

Dopo che un utente ha selezionato gli articoli sul sito web del fornitore, li ha configurati e ha fatto clic sul pulsante "Estrai", il fornitore invia un documento PunchOutOrderMessage per comunicare all'applicazione di acquisizione del compratore il contenuto del cestino della spesa. Questo tipo di documento può contenere molti più dati degli altri, poiché deve essere in grado di descrivere il contenuto di qualsiasi tipo di cestino della spesa e non deve necessariamente seguire il modello richiesta-risposta; l'uso di questo documento è spiegato di seguito in modo più dettagliato.

Nell'esempio riportato di seguito viene mostrato un documento PunchOutOrderMessage:

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" xml:lang="en-US" payloadID="933695160894"
timestamp="2000-08-15T08:47:00-07:00">
  <Header>
    <From>
      <Credential domain="DUNS">
        <Identity>83528721</Identity>
      </Credential>
    </From>
    <To>
      <Credential domain="DUNS">
        <Identity>65652314</Identity>
      </Credential>
    </To>
    <Sender>
      <Credential domain="workchairs.com">
```

```

        <Identity> website 1</Identity>
    </Credential>
    <UserAgent>Workchairs cXML Application</UserAgent>
</Sender>
</Header>
<Message>
    <PunchOutOrderMessage>
        <BuyerCookie>1CX3L4843PPZO</BuyerCookie>
        <PunchOutOrderMessageHeader operationAllowed="edit">
            <Total>
                <Money currency="USD">763.20</Money>
            </Total>
        </PunchOutOrderMessageHeader>
        <ItemIn quantity="3">
            <ItemID>
                <SupplierPartID>5555</SupplierPartID>
                <SupplierPartAuxiliaryID>E000028901
            </SupplierPartAuxiliaryID>
            </ItemID>
            <ItemDetail>
                <UnitPrice>
                    <Money currency="USD">763.20</Money>
                </UnitPrice>
                <Description xml:lang="en">
                    <ShortName>Excelsior Desk Chair</ShortName>
                    Leather Reclining Desk Chair with Padded Arms
                </Description>
                <UnitOfMeasure>EA</UnitOfMeasure>
                <Classification domain="UNSPSC">5136030000
            </Classification>
            </ItemDetail>
        </ItemIn>
    </PunchOutOrderMessage>
</Message>
</cXML>

```

L'elemento BuyerCookie permette all'applicazione di acquisizione di associare un documento PunchOutOrderMessage al documento PunchOutSetupRequest che lo ha generato. Pertanto, il sito web deve restituire questo elemento ogni volta che lo incontra. L'uso dell'elemento BuyerCookie per tenere traccia delle sessioni PunchOut non è consigliabile, in quanto l'elemento è diverso per ognuna delle sessioni di creazione, ispezione o modifica.

Il campo `SupplierPartAuxiliaryID` ha la stessa funzione di un cookie del fornitore, in quanto consente di trasmettere dati aggiuntivi, quali il numero di preventivo o un altro documento cXML. L'applicazione di acquisizione restituisce questi dati al fornitore nelle successive sessioni `PunchOutSetupRequest` edit o inspect, e nell'ordine d'acquisto cXML che ne risulta. Il fornitore può utilizzare questo cookie per associare gli articoli presenti in una richiesta di acquisto agli articoli corrispondenti che si trovano in un carrello della spesa sul proprio sito web.

`UnitOfMeasure` descrive la modalità di confezionamento o di spedizione del prodotto. Deve essere conforme ai codici usati per le unità di misura UN/CEFACT. Per l'elenco dei codici UN/CEFACT, visitare il sito web all'indirizzo www.unece.org/cefact.

L'elemento `Classification` riporta il codice prodotto UNSPSC (United Nations Standard Product and Service Code, codici standard di prodotti e servizi delle Nazioni Unite) per ognuno degli articoli selezionati. Questi codici vengono utilizzati dai sistemi back-end utilizzati dalle società acquirenti e dei fornitori per la gestione amministrativa e la generazione dei rapporti. Per un elenco dei codici UNSPSC, vedere www.unspsc.org.

Modifica delle pagine web

Per ricevere o inviare i tre documenti PunchOut cXML, può rendersi necessaria la modifica o la creazione di quattro pagine sul sito web:

- Pagina di apertura
- Pagina di avvio
- Pagina di invio
- Pagina di ricezione

Per spiegare come è possibile implementare queste pagine, verranno impiegati degli esempi di codice ASP (Active Server Page) utilizzati con l'analizzatore XML di Microsoft Internet Explorer 5. L'implementazione di queste pagine varia a seconda dell'ambiente di sviluppo utilizzato dal fornitore (ad esempio, CGI, JavaScript o WebObjects).

Pagina di apertura

La pagina di apertura riceve tutti i documenti PunchOutSetupRequest autenticati dall'hub per l'e-commerce. Questa pagina legge il flusso HTTP inviato dall'hub e convalida la richiesta cXML incorporata in questo flusso, confrontandola con la DTD cXML. Nel caso delle ASP, vengono richiamati dei metodi dall'analizzatore XML di Internet Explorer 5.

Dopo la convalida, la pagina di apertura estrae gli elementi dal documento, al fine di:

1. Identificare l'utente e determinare dove indirizzarlo.
2. Comporre un documento PunchOutSetupResponse e restituirlo al mittente.

La pagina di apertura dovrebbe contenere i seguenti dati, che vengono poi utilizzati dalla pagina di avvio:

- Identità del richiedente (Sender)
- Identificativo della lingua dell'utente (xml:lang), per poter offrire un contenuto localizzato
- Tipo della richiesta, ovvero creazione, modifica o ispezione (create, edit o inspect)
- Dati estrinseci che permettono di identificare in modo più preciso l'utente e la relativa ubicazione

L'esempio riportato di seguito mostra una pagina di apertura. Questo codice non utilizza un analizzatore XML per generare dinamicamente un documento PunchOutSetupResponse, ma impiega un modello XML statico in cui vengono inseriti i dati relativi agli articoli. **Il codice viene fornito solo a scopo illustrativo.**

```
script language=JScript RUNAT=Server>
function elementValue(xml, elem)
{
    var begidx;
    var endidx;
    var retStr;

    begidx = xml.indexOf(elem);
    if (begidx > 0) {
        endidx = xml.indexOf('</', begidx);
        if (endidx > 0)
            retStr = xml.slice(begidx+elem.length,
                               endidx);
        return retStr;
    }
    return null;
}
```

```

    }

    function twoChar( str )
    {
        var retStr;
        str = str.toString();
        if ( 1 == str.length ) {
            retStr = "0" + str;
        } else {
            retStr = str;
        }
        return retStr;
    }

    function timestamp( dt )
    {
        var str;
        var milli;
        str = dt.getFullYear() + "-" + twoChar( 1 + dt.getMonth() ) + "-";
        str += twoChar( dt.getDate() ) + "T" + twoChar( dt.getHours() ) + ":";
        str += twoChar( dt.getMinutes() ) + ":" + twoChar( dt.getSeconds() ) + ".";
        milli = dt.getMilliseconds();
        milli = milli.toString();
        if ( 3 == milli.length ) {
            str += milli;
        } else {
            str += "0" + twoChar( milli );
        }
        str += "-08:00";
        return str;
    }

    function genProlog( cXMLvers, randStr )
    {
        var dt;
        var str;
        var vers, sysID;
        var nowNum, timeStr;
        if ( 1.1 > parseFloat( cXMLvers ) ) {
            vers = "1.0";
            sysID = "cXML.dtd";
        } else {
            vers = "1.1.007";
            sysID = "http://xml.cXML.org/schemas/cXML/" + vers + "/cXML.dtd";
        }
        dt = new Date();
        nowNum = dt.getTime();
        timeStr = timestamp( dt );
    }

```



```

str = '<?xml version="1.1.007" encoding="UTF-8"?>\n';
str += '<!DOCTYPE cXML SYSTEM "" + sysID + "">\n';
str += '<cXML version="" + vers + "" payloadID="" + nowNum + ". ";
str += randStr + '@' + Request.ServerVariables("LOCAL_ADDR");
str += "" timestamp="" + timeStr + "">';
return str;
}
</script>
REM Create data needed in prolog.
%<
Randomize
randStr = Int( 100000001 * Rnd )
prologStr = genProlog( "1.0", randStr )
Response.ContentType = "text/xml"
Response.Charset = "UTF-8"
%>
<%
REM This receives the PunchOutSetup request coming from the e-commerce hub.
REM It takes the ORMSURL and buyercookie, attaches them to the Start Page URL,
REM and sends the response back to the requester.
REM punchoutredirect.asp?bc=2133hfefe&url="http://workchairs/com/..&redirect="
Dim ret
Dim punch
Dim statusText
Dim statusCode
Dim cookie
Dim url
Dim xmlstr
Dim fromUser
Dim toUser
cookie = ""
url = ""
xmlstr = ""
dir = ""
path = Request.ServerVariables("PATH_INFO")
dir = Left(path, InstrRev(path, "/"))
if IsEmpty(dir) then
    dir = "/"
end if

REM This command reads the incoming HTTP cXML request
xml = Request.BinaryRead(Request.TotalBytes)
for i = 1 to Request.TotalBytes
    xmlstr = xmlstr + String(1,AscB(MidB(xml, i, 1)))
Next
cookie = elementValue(xmlstr, "<BuyerCookie>")
url = elementValue(xmlstr, "<URL>")
fromUser = elementValue(xmlstr, "<Identity>")
newXMLStr = Right(xmlstr, Len(xmlstr) - (Instr(xmlstr, "<Identity>") +
Len("<Identity>")))

```

```

        toUser = elementValue(newXMLStr, "<Identity>")
    %>
    REM This formats the cXML PunchOutSetupResponse
    <% if IsEmpty(cookie) then %>
    <%= prologStr %>
    <Response>
    <Status code="400" Text="Bad Request">Invalid Document. Unable to extract
    BuyerCookie.</Status>
    </Response>
</cXML>
<% else %>
<%= prologStr %>
<Response>
    <Status code="200" text="OK"/>
    <PunchOutSetupResponse>
    <StartPage>
    <URL>http://<%=
Request.ServerVariables("LOCAL_ADDR")%>/<%= dir%>/punchoutredirect.asp?bc=<%=
cookie%>&amp;url="<%= url%>"&amp;from=<%= fromUser%>&amp;to=<%=
toUser%>&amp;redirect=<%= StartPage%></URL>
    </StartPage>
    </PunchOutSetupResponse>
    </Response>
</cXML>
<%end if%>

```

La pagina di apertura deve restituire l'indirizzo Internet di una StartPage, ovvero di una pagina di avvio, che sia univoco nell'ambito della sessione PunchOut. In più, questo indirizzo deve essere valido solo per un periodo di tempo limitato. La disattivazione dell'indirizzo Internet, infatti, limita la possibilità che utenti non autorizzati accedano alla pagina di avvio.

Tenere presente che è necessario implementare la funzionalità delle successive sessioni di modifica e ispezione (edit e inspect). Un utente non può modificare i dettagli degli ordini relativi agli articoli PunchOut (ad esempio, la quantità) utilizzando l'applicazione di acquisizione, ma devono effettuare nuovamente il PunchOut con una sessione di modifica. Per agevolare l'utente, le sessioni di ispezione successive alla ricezione dell'ordine da parte del fornitore dovrebbero indicare lo stato dell'ordine.

Pagina di avvio

Lo scopo della pagina di avvio è quello di collegare il richiedente al relativo account presente sul sito web del fornitore. Dalla pagina di avvio del sito web, l'utente può cominciare ad effettuare gli acquisti. Se questa pagina è già presente sul sito web del fornitore, deve essere modificata affinché ricerchi la password ed il nome dell'utente nel documento PunchOutSetupRequest.

L'accesso alla pagina di avvio deve essere consentito solo agli utenti autorizzati. Se gli utenti vengono autenticati durante la fase di estrazione, le informazioni riservate relative a prezzi o condizioni non sono protette.

Se si utilizzano cookie del browser HTTP per tenere traccia delle preferenze dell'utente, è consigliabile eliminarli dopo avere inviato ai compratori il documento PunchOutOrderMessage. Questa prassi permette di evitare che alcuni privilegi, quale l'uso di alcune funzioni, possano essere concessi ad utenti non autorizzati.

Pagina di invio

Lo scopo della pagina di invio è quello di inviare il contenuto del carrello della spesa di un utente. L'utente, come descritto in precedenza, deve prima riempire il carrello della spesa, quindi deve fare clic sul pulsante "Estrai" sulla pagina del sito web del fornitore.

Di seguito viene mostrata una semplice implementazione ASP di questa funzione. Questo codice non utilizza un analizzatore XML per generare dinamicamente un documento PunchOutOrderMessage, ma impiega un modello XML statico in cui vengono inseriti i dati relativi agli articoli. **Il codice viene fornito solo a scopo illustrativo.**

Di seguito è riportata una parte di una pagina del sito web del fornitore, relativa ad un prodotto:

```
<!--#include file="punchoutitem.inc"-->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<!-- saved from
url=(0093)https://secure1.shore.net/wbird/cgi/vsc.cgi/wbird/houses/urban.htm?L+wbird+w
adt4101+928011405 -->

<TABLE border=0>
  <TBODY>
    <TR>
      <TD><IMG src="UrbanHouses_files/uhjm.gif"> </TD>
      <TD><STRONG>Jefferson Memorial</STRONG>- A birdfeeder with a
rotunda! This famous American monument will be a unique addition to any garden or yard.
It attracts small to medium sized birds and its dimensions are 11" x 9 1/2" x 8" H.
```

```

        </TD>
    </TR>
</TBODY>
</TABLE><BR>
-Jefferson Memorial<STRONG>
$139.95</STRONG><BR>
<% AddBuyButton 139.95,101,"Bird Feeder, Jefferson Memorial",5 %>
<BR>
<HR>

```

La funzione AddBuyButton restituisce il documento PunchOutOrderMessage all'utente.

L'elenco che segue è il file di inclusione (punchoutitem.inc) sopra citato:

```

<%
REM This asp is included in items.asp, which specifies the item parameters, formats
REM a cXML document, and allows the user to proceed with a checkout of the item.
function CreateCXML(toUser, fromUser, buyerCookie, unitPrice, supPartId, desc)
%>
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE cXML SYSTEM
"http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" payloadID="<%= Now &"@"&
Request.ServerVariables("LOCAL_ADDR")%>"> timestamp="<%= Now
%>">
    <Header>
        <From>
            <Credential domain="ariba.com">
                <Identity><%= toUser%></Identity>
            </Credential>
        </From>
        <To>
            <Credential domain="ariba.com">
                <Identity><%= fromUser%></Identity>
            </Credential>
        </To>
        <Sender>
            <Credential domain="ariba.com">
                <Identity><%= toUser%></Identity>
            </Credential>
            <UserAgent>PunchoutSite</UserAgent>
        </Sender>
    </Header>
    <Message>
        <PunchOutOrderMessage>
            <BuyerCookie><%= buyerCookie%></BuyerCookie>
            <PunchOutOrderMessageHeader
operationAllowed="edit">
                <Total>
                    <Money currency="USD"><%=

```

```

        unitPrice%></Money>
    </Total>
</PunchOutOrderMessageHeader>
<ItemIn quantity="1">
    <ItemID>
        <SupplierPartID><%= supPartId%></SupplierPartID>
        <SupplierPartAuxiliaryID><%= supPartAuxId%>
        </SupplierPartAuxiliaryID>
    </ItemID>
    <ItemDetail>
        <UnitPrice>
            <Money currency="USD"><%= unitPrice%>
            </Money>
        </UnitPrice>
        <Description xml:lang="en"><%= desc%>
        </Description>
        <UnitOfMeasure>EA</UnitOfMeasure>
        <Classification
            domain="SupplierPartID"><%= supPartId%>
        </Classification>
    </ItemDetail>
</ItemIn>
</PunchOutOrderMessage>
</Message>
</cXML>
<% end function

function AddBuyButton(unitPrice, supPartId, supPartAuxId, desc)

toUser = Session("toUser")
fromUser = Session("fromUser")

buyerCookie = Session("buyercookie")
url = Session("urlToPost")
if not IsEmpty(buyerCookie) then
    %>
    <FORM METHOD=POST ACTION=<%= url%>>
    <INPUT TYPE=HIDDEN NAME="cxml-urlencoded" VALUE="<% CreateCXML
toUser, fromUser, buyerCookie, unitPrice, supPartId, supPartAuxId, desc%>">
    <INPUT TYPE=SUBMIT value=BUY>
    </FORM>
<%else%>
    </p>
    <%
end if
end function
%>

```

La funzione `AddBuyButton` contiene il comando FORM POST che restituisce all'utente il documento `PunchOutOrderMessage` URL-Encoded.

HTTP

Per inviare un documento `PunchOutOrderMessage`, viene utilizzato il metodo di *codifica HTML*, un modello di trasporto diverso dal tradizionale modello HTTP richiesta-risposta, che facilita l'integrazione tra il sito web del fornitore e l'applicazione di acquisizione dell'utente. Questo metodo, inoltre, permette alle società acquirenti di ricevere i dati XML anche se non dispongono di un server web con accesso tramite firewall.

Anziché inviare un documento `PunchOutOrderMessage` direttamente all'applicazione di acquisizione, il sito web lo codifica come campo nascosto HTML Form e lo invia all'indirizzo Internet specificato nell'elemento `BrowserFormPost` del documento `PunchOutSetupRequest`. Il nome di questo campo nascosto deve essere `cxml-urlencoded` oppure `cxml-base64` (questi nomi non distinguono tra lettere maiuscole e lettere minuscole).

Questa codifica permette di creare una pagina web per l'estrazione, che contiene il documento cXML. Quando un utente fa clic sul pulsante "Estrai", il sito web presenta i dati all'applicazione di acquisizione (ma non all'utente) in forma di richiesta HTML Form Submit.

Annullo della sessione PunchOut

Il fornitore può aggiungere alle pagine del sito web un pulsante "Annulla" che l'utente può utilizzare per annullare la sessione di PunchOut. Questo pulsante invia un documento `PunchOutOrderMessage` vuoto che indica all'applicazione di acquisizione che non verrà restituito alcun articolo e che è necessario eliminare gli articoli PunchOut esistenti dalla richiesta di acquisto. Questo pulsante permette anche di effettuare alcune operazioni di gestione del sito web, come svuotare il carrello della spesa e chiudere la sessione dell'utente.

Pagina di ricezione

La pagina di ricezione accetta gli ordini d'acquisto cXML inviati al fornitore dalle società acquirenti. Questa pagina può essere simile alla pagina di apertura illustrata in precedenza.

Per ulteriori informazioni sulla ricezione degli ordini d'acquisto, vedere Capitolo 3, "Ricezione degli ordini d'acquisto cXML."

Suggerimenti per i siti web PunchOut

Se si desidera implementare un sito web PunchOut, è necessario tenere presente quanto segue.

Consigli per l'implementazione

Di seguito sono riportati alcuni consigli per implementare un sito web PunchOut:

- Studiare la specifica cXML.
- Utilizzare un analizzatore XML e convalidare i documenti utilizzando la DTD cXML.
- Utilizzare la proprietà `xml:lang=` per identificare le varie lingue e offrire agli utenti un contenuto localizzato.
- Utilizzare la credenziale `From` per identificare le società acquirenti.
- Inviare un indirizzo Internet univoco e temporaneo per l'indirizzamento della sessione.
- Evitare di lasciare attivi i cookie del browser.
- Evitare di utilizzare un numero eccessivo di dati estrinseci per i compratori.
- Per ogni articolo, utilizzare i codici UNUOM (United Nations Units of Measure, unità di misura delle Nazioni Unite) ed UNSPSC (United Nations Standard Product and Service Codes, codici standard di prodotti e servizi delle Nazioni Unite).
- Offrire un valore reale ai propri clienti. Presentare disponibilità dei prodotti, stato degli ordini e offerte speciali.
- Il processo di estrazione deve essere facile e intuitivo. Idealmente, l'utente dovrebbe comprare facendo clic solo su tre pulsanti.
- Codificare le successive sessioni di modifica e ispezione. L'utente non può modificare i dettagli di un ordine che contiene articoli PunchOut (ad esempio, la quantità) utilizzando direttamente l'applicazione di acquisizione, ma deve effettuare nuovamente il PunchOut con una sessione di modifica.
- Per agevolare l'utente, le sessioni di ispezione dovrebbero indicare lo stato dell'ordine.
- Verificare la funzionalità del sito web PunchOut, anche con le applicazioni di acquisizione dei clienti.
- Le transazioni PunchOut non producono ordini, ma preventivi. Per l'accettazione degli ordini, è necessario implementare una pagina di ricezione degli ordini d'acquisto cXML.

Cookie del compratore e del fornitore

I cookie del compratore e del fornitore permettono a entrambi di creare per i propri sistemi back-end nuove istanze dei dati sugli articoli.

- Il fornitore deve restituire il cookie del compratore (BuyerCookie) ricevuto. Non deve modificare questo elemento.
- Il fornitore deve utilizzare l'apposito cookie (SupplierPartAuxiliaryID).

Il cookie del compratore ha una funzione analoga a quella del numero di richiesta di acquisto: permette al sistema delle società acquirenti di gestire la relazione tra una richiesta di acquisto e un cestino della spesa.

Il cookie del fornitore, invece, ha una funzione analoga a quella di un numero di preventivo: permette al sistema del fornitore di gestire la relazione tra un cestino della spesa e la richiesta di acquisto e l'ordine d'acquisto del compratore. Le applicazioni di acquisizione restituiscono al fornitore il cookie del fornitore precedentemente ricevuto, durante le successive sessioni PunchOut di modifica o ispezione, nell'ordine d'acquisto che ne deriva. Il sito web del fornitore può avvalersi del cookie del fornitore per evitare di fare vedere al compratore dati specifici del fornitore.

Personalizzazione

L'intestazione di un documento PunchOutSetupRequest identifica sempre la società acquirente, ma la richiesta può contenere anche i dati sul contatto e i dati estrinseci (ad esempio, il centro di costo e l'ubicazione dell'utente, oppure la categoria dei prodotti) per stabilire l'indirizzo Internet dinamico per collegarsi all'utente.

Anche se i dati estrinseci non vengono inviati da tutte le società acquirenti, permettono di personalizzare un negozio sul web andando oltre il semplice livello organizzativo. Ad esempio, è possibile offrire un negozio diverso per ogni centro di costo della stessa società acquirente (oppure per ogni categoria di prodotti o ogni utente).

Il fornitore può anche memorizzare e visualizzare i precedenti preventivi di un utente e autorizzare l'utente a riutilizzarli, o a controllare lo stato degli ordini o, ancora, a creare rapporti sulle attività passate. Per evitare problemi di sicurezza, è consigliabile memorizzare la cronologia dei preventivi solo a livello di utente.

Nella pianificazione di un sito web PunchOut altamente dinamico e personalizzato, è necessario considerare il tempo e gli sforzi necessari per implementarlo. Si tratta di trovare il giusto rapporto tra personalizzazione e complessità: un sito web dinamico richiede più tempo per essere implementato e gestito, ma può offrire più valore agli utenti. Si consiglia di iniziare con un sito web PunchOut semplice, da arricchire e migliorare successivamente.

Capitolo 3

Ricezione degli ordini d'acquisto cXML

In questo capitolo viene descritto come impostare un sito web affinché possa ricevere ordini d'acquisto in formato cXML. Inoltre, viene spiegato come inviare messaggi sullo stato degli ordini d'acquisto alle società acquirenti o ad un particolare mercato.

Il processo degli ordini d'acquisto

Le applicazioni di acquisizione convertono le richieste di acquisto approvate in uno o più ordini d'acquisto. Un ordine d'acquisto è la richiesta formale che una società acquirente inoltra ad un fornitore per richiedere l'adempimento di un contratto.

Il cXML è semplicemente un formato per la trasmissione degli ordini d'acquisto. Gli altri formati utilizzati sono la posta elettronica, i fax e l'EDI (X.12 Electronic Data Interchange). Il cXML è il formato ideale, poiché permette di automatizzare con facilità il processo di evasione degli ordini. La struttura ben definita del cXML consente ai sistemi di elaborazione degli ordini di interpretare facilmente gli elementi contenuti in un ordine d'acquisto e di instradare l'ordine ai reparti di spedizione, fatturazione o vendita appropriati senza richiedere l'intervento da parte dell'utente o comunque con un intervento minimo.

Inoltre, il metodo per l'instradamento degli ordini cXML consente la trasmissione di qualsiasi cookie (SupplierPartAuxiliaryID) e allegato all'ordine d'acquisto del fornitore.

Per configurare un account su un hub di rete per l'e-commerce, è necessario specificare l'indirizzo Internet a cui dovranno essere inviati tutti gli ordini d'acquisto cXML. Una volta ricevuto un ordine d'acquisto, questo viene inviato al sistema di gestione interna degli ordini che lo evade secondo la normale prassi. Il sito web deve restituire all'hub un messaggio di conferma OrderResponse, che indica al compratore che il suo ordine è stato ricevuto ed analizzato correttamente.

Per ricevere gli ordini d'acquisto cXML non è necessario disporre di un sito web PunchOut; la ricezione degli ordini cXML e il PunchOut sono due caratteristiche diverse. Tuttavia, l'infrastruttura e le applicazioni richieste per utilizzare il PunchOut sono le stesse di quelle necessarie per la ricezione degli ordini d'acquisto cXML.

Ricezione degli ordini d'acquisto

Esistono due tipi di documenti cXML che vengono utilizzati per comunicare gli ordini d'acquisto. L'applicazione di acquisizione invia un documento OrderRequest al quale si risponde con un documento OrderResponse. Questi documenti passano attraverso l'hub di rete per l'e-commerce.

OrderRequest

Il documento OrderRequest è simile ad un ordine d'acquisto. Nell'esempio riportato di seguito viene mostrato il documento OrderRequest per un articolo:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" xml:lang="en-US" payloadID="93369535150910.10.57.136"
timestamp="2000-08-03T08:49:11+07:00">
  <Header>
    <From>
      <Credential domain="AribaNetworkUserId">
        <Identity>admin@acme.com</Identity>
      </Credential>
    </From>
    <To>
      <Credential domain="DUNS">
        <Identity>114315195</Identity>
      </Credential>
    </To>
    <Sender>
      <Credential domain="AribaNetworkUserId">
        <Identity>sysadmin@ariba.com</Identity>
        <SharedSecret>abracadabra</SharedSecret>
      </Credential>
      <UserAgent>Ariba Network V1.1</UserAgent>
    </Sender>
  </Header>
  <Request>
    <OrderRequest>
      <OrderRequestHeader orderID="DO102880"
orderDate="2000-08-03T08:49:09+07:00" type="new">
        <Total>
```

```

    <Money currency="USD">4688.00</Money>
  </Total>
  <ShipTo>
    <Address isoCountryCode="US" addressID="1000467">
      <Name xml:lang="en">Acme, Inc.</Name>
      <PostalAddress name="default">
        <DeliverTo>John Q. Smith</DeliverTo>
        <DeliverTo>Buyers Headquarters</DeliverTo>
        <Street>123 Main Street</Street>
        <City>Mountain View</City>
        <State>CA</State>
        <PostalCode>94089</PostalCode>
        <Country>United States</Country>
      </PostalAddress>
      <Email name="default">john_smith@acme.com</Email>
      <Phone name="work">
        <TelephoneNumber>
          <CountryCode isoCountryCode="US">1
          </CountryCode>
          <AreaOrCityCode>800</AreaOrCityCode>
          <Number>5555555</Number>
        </TelephoneNumber>
      </Phone>
    </Address>
  </ShipTo>
  <BillTo>
    <Address isoCountryCode="US" addressID="12">
      <Name xml:lang="en">Acme Accounts Payable</Name>
      <PostalAddress name="default">
        <Street>124 Union Street</Street>
        <City>San Francisco</City>
        <State>CA</State>
        <PostalCode>94128</PostalCode>
        <Country isoCountryCode="US">US</Country>
      </PostalAddress>
      <Phone name="work">
        <TelephoneNumber>
          <CountryCode isoCountryCode="US">1
          </CountryCode>
          <AreaOrCityCode>415</AreaOrCityCode>
          <Number>6666666</Number>
        </TelephoneNumber>
      </Phone>
    </Address>
  </BillTo>
  <Shipping>
    <Money currency="USD">12.34</Money>
    <Description xml:lang="it-IT">Pantofole antisdrucciolo</Description>
  </Shipping>
  <Tax>

```

```
        <Money currency="USD">10.74</Money>
        <Description xml:lang="en">CA State Tax</Description>
    </Tax>
    <Payment>
        <PCard number="1234567890123456" expiration="2002-03-12"/>
    </Payment>
</OrderRequestHeader>
<ItemOut quantity="2" >
    <ItemID>
        <SupplierPartID>220-3165</SupplierPartID>
        <SupplierPartAuxiliaryID>E000028901</SupplierPartAuxiliaryID>
    </ItemID>
    <ItemDetail>
        <UnitPrice>
            <Money currency="USD">2344.00</Money>
        </UnitPrice>
        <Description xml:lang="en">Laptop Computer Notebook Pentium® II
processor w/AGP, 300 MHz, with 12.1" TFT XGA Display
        </Description>
        <UnitOfMeasure>EA</UnitOfMeasure>
        <Classification domain="UNSPSC">43171801</Classification>
        <URL>http://www.supplier.com/Punchout.asp</URL>
        <Extrinsic name="ExtDescription">Enhanced keyboard</Extrinsic>
    </ItemDetail>
    <Distribution>
        <Accounting name="DistributionCharge">
            <Segment type="Account" id="7720"
description="Office Supplies"/>
            <Segment type="CostCenter" id="610"
description="Engineering Management"/>
        </Accounting>
        <Charge>
            <Money currency="USD">4688.00</Money>
        </Charge>
    </Distribution>
</ItemOut>
</OrderRequest>
</Request>
</cXML>
```

OrderResponse

Il documento OrderResponse conferma che l'ordine d'acquisto è stato ricevuto e correttamente analizzato. Non costituisce un impegno ad eseguire l'ordine, ma si limita ad informare che l'ordine è stato ricevuto ed è stato riconosciuto come documento cXML valido.

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" payloadID="8/3/2000 8:49:30 PM@205.180.14.45"
timestamp="2000-08-03T08:49:30+07:00">
  <Response>
    <Status code="200" text="OK"/>
  </Response>
</cXML>
```

Accettazione degli allegati agli ordini

Spesso, i compratori hanno bisogno di rendere più chiari gli ordini d'acquisto utilizzando promemoria, disegni o fax. A tale scopo, è possibile allegare file di qualsiasi tipo agli ordini d'acquisto cXML utilizzando le estensioni MIME (Multipurpose Internet Mail Extensions).

Il cXML contiene solo i riferimenti alle parti MIME esterne inviate in un unico elemento MIME envelope a più parti (con il documento cXML, in un messaggio di posta elettronica o con un fax).

L'hub di rete per l'e-commerce riceve gli allegati e li può inoltrare al fornitore, oppure li può memorizzare per consentirne il recupero in linea.

Per ulteriori informazioni sugli allegati agli ordini d'acquisto, vedere "Trasmissione degli allegati" a pagina 55.

Per ulteriori informazioni sullo standard MIME, visitare i siti web agli indirizzi riportati di seguito:

www.hunnysoft.com/mime
www.rad.com/networks/1995/mime/mime.htm

Appendice A

Specifica del linguaggio cXML

Questa appendice descrive il protocollo e i formati di dati del linguaggio cXML (commerce eXtensible Markup Language). Contiene tutte le informazioni necessarie a implementare qualsiasi transazione supportata sia dal punto di vista del sistema client che da quello del sistema server. Tratta approfonditamente sia le interazioni del protocollo sia i documenti commerciali.

Inoltre, sono riportati degli esempi che illustrano le implementazioni effettive e chiariscono come usare il cXML.

Questa appendice contiene le sezioni seguenti:

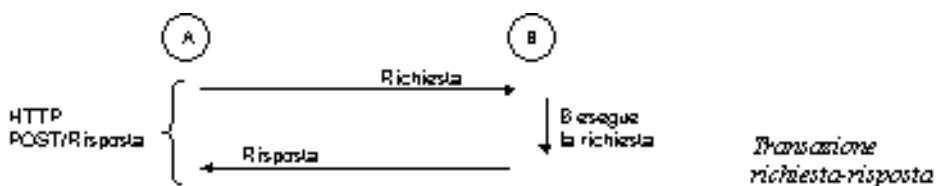
- Specifica del protocollo
- Elementi fondamentali
- Transazione Profile
- Definizione degli ordini
- Transazione PunchOut
- Modifiche successive allo stato
- Definizioni per i cataloghi
- Definizioni per la gestione degli abbonamenti
- Definizioni per il recupero dei messaggi

Specifica del protocollo

Esistono due modelli di comunicazione per le transazioni cXML: richiesta-risposta e unilaterale. Questi due modelli consentono una semplice implementazione poiché le operazioni richieste sono descritte dettagliatamente. Sono necessari entrambi i modelli in quanto in alcune situazioni un solo modello potrebbe non essere appropriato.

Modello richiesta-risposta

Le transazioni di tipo richiesta-risposta possono essere eseguite solo con una connessione HTTP. Nella figura riportata di seguito vengono mostrati le fasi di un'interazione richiesta-risposta tra le parti A e B:



Questa transazione è composta dalle fasi seguenti:

1. A avvia una connessione HTTP/1.x con B utilizzando un indirizzo Internet prestabilito che rappresenta l'indirizzo di B.
2. A usa un'operazione POST per inviare il documento cXML mediante la connessione HTTP.
3. A attende una risposta mediante la connessione HTTP.
4. B dispone di un server che supporta il protocollo HTTP/1.x e che spedisce la richiesta HTTP alla risorsa specificata dall'indirizzo Internet usato al punto 1. Questa risorsa può essere costituita da qualsiasi posizione valida conosciuta dal server HTTP di B come, ad esempio, un programma CGI o una pagina ASP.
5. La risorsa di B identificata al punto 4 legge il contenuto del documento cXML e associa la richiesta al gestore in grado di soddisfarla.
6. Il gestore di B della richiesta cXML esegue le azioni indicate dalla richiesta e prepara la risposta in un documento cXML.

7. B invia la risposta cXML ad A mediante la connessione HTTP stabilita al punto 1.
8. A legge la risposta cXML e la restituisce al processo che ha avviato la richiesta.
9. A chiude la connessione HTTP stabilita al punto 1.

Questo processo viene quindi ripetuto per altri cicli richiesta-risposta.

Per semplificare le operazioni descritte nei punti precedenti, i documenti cXML sono suddivisi in due parti distinte:

- Intestazione: contiene le informazioni sull'indirizzamento e l'autenticazione.
- Dati della richiesta o della risposta: contiene una specifica richiesta o risposta e le informazioni da trasmettere.

Entrambi questi elementi vengono trasmessi in un elemento envelope principale. Nell'esempio riportato di seguito viene mostrata la struttura di un documento cXML di tipo richiesta:

```
<cXML>
  <Header>
    Informazioni sull'intestazione...
  </Header>
  <Request>
    Informazioni sulla richiesta...
  </Request>
</cXML>
```

Nell'esempio riportato di seguito viene mostrata la struttura di un documento cXML di tipo risposta:

```
<cXML>
  <Response>
    Informazioni sulla risposta...
  </Response>
</cXML>
```

La struttura della risposta non utilizza l'elemento Header poiché la risposta viene trasmessa usando la stessa connessione della richiesta.

Convenzioni XML

Il cXML utilizza gli elementi per descrivere le varie voci che spesso caratterizzano i documenti commerciali tradizionali. Gli elementi vengono utilizzati anche per descrivere le normali suddivisioni e le relazioni tra tali suddivisioni, ad esempio un indirizzo.

Il cXML è caratterizzato anche da molti attributi.

Nel cXML, tutti i nomi di elemento e di attributo utilizzano parole intere e si avvalgono delle maiuscole anziché dei trattini per separare le parole. I nomi di elemento iniziano con la maiuscola mentre i nomi di attributo con la minuscola. Ad esempio:

Elementi:	Sender, Credential, Payment, ItemDetail
Attributi:	version, payloadID, lineNumber, domain

Elemento cXML envelope

L'elemento envelope rappresenta l'inizio della struttura del documento cXML e contiene tutti gli altri elementi. L'elemento cXML è presente in ogni transazione cXML.

Nell'esempio riportato di seguito viene mostrato un elemento cXML specificato per intero:

```
<cXML version="1.1.007" xml:lang="en-US"
  payloadID=1234567.4567.5678@test.ariba.com
  timestamp="1999-03-31T18:39:09-08:00">
```

Gli attributi di cXML sono i seguenti:

version (facoltativo)	Specifica la versione del protocollo cXML. Per determinare l'attributo version dalla DTD di riferimento è possibile utilizzare anche un analizzatore di convalida XML. Comunque, tutti i documenti cXML dovrebbero includere esplicitamente l'attributo version come ausilio alle applicazioni che usano analizzatori che non effettuano la convalida.
xml:lang (facoltativo)	La lingua usata per tutto il testo non codificato contenuto in questo documento. Il destinatario dovrebbe rispondere o visualizzare le informazioni nella stessa lingua o in una simile. Ad esempio, un client che specifica xml:lang="en-UK" in una richiesta, potrebbe ricevere i dati "en" in risposta.
payloadID	Un numero univoco nello spazio e nel tempo, usato a scopi di registrazione per identificare i documenti che potrebbero andare perduti o che presentano problemi. Questo valore non deve cambiare per poterlo usare in altri tentativi. L'implementazione consigliata è: dataora.ID processo.numero casuale@nomehost.
timestamp	La data e l'ora in cui è stato inviato il messaggio nel formato ISO 8601. Questo valore non deve cambiare per poterlo usare in altri tentativi. Il formato è AAAA-MM-GGThh:mm:ss-hh:mm (ad esempio, 1997-07-16T19:20:30+01:00).

Attributo xml:lang per la specifica della lingua

L'attributo xml:lang è usato anche con molti elementi per il testo non codificato (ad esempio Description e Comments). Mentre la specifica XML consente alla lingua di un elemento di assumere il valore predefinito specificato per qualsiasi elemento principale, questi valori predefiniti generano query inefficaci della struttura del documento. Il cXML tenta di mantenere gli identificativi della lingua insieme alle stringhe interessate.

Gli attributi xml:lang presenti in tutto il protocollo cXML non influiscono sui dati formattati come i numeri, le date e gli orari. Come descritto di seguito a proposito dell'attributo timestamp, questi diversi valori sono formattati secondo i rispettivi tipi di dati. Le stringhe più lunghe (e le pagine web a cui fanno riferimento) non destinate all'elaborazione della macchina potrebbero contenere un formato di data o numerico specifico di una lingua che corrisponde a un attributo xml:lang vicino.

Indicazioni temporali e altri tipi di dati

L'attributo timestamp (e tutte le date e gli orari nel cXML) devono essere formattati secondo lo standard ISO 8601 descritto nella nota W3C (Word Wide Web Consortium) intitolata "Date and Time Formats" e disponibile all'indirizzo www.w3.org/TR/NOTE-datetime-970915.html.

I timestamp richiedono almeno la specifica della data e dell'ora (quest'ultima completa dei minuti e dei secondi). Le frazioni dei secondi sono facoltative. Questo protocollo richiede che gli orari siano espressi nell'ora locale, considerando il fuso orario secondo l'UTC (Coordinated Universal Time ovvero Ora di Greenwich). Non è possibile usare l'indicatore "Z" del fuso orario.

Ad esempio, 2000-04-14T013:36:00-08:00 corrisponde a 14 aprile 2000, 13:36 (ora del Pacifico).

Ulteriori riferimenti per i formati di data, ora e altri tipi di dati usati dal cXML sono:

- Il sito XML Data Types Reference della Microsoft
msdn.microsoft.com/xml/reference/schema/datatypes.asp
- La proposta originale XML Data al W3C (Word Wide Web Consortium)
www.w3c.org/TR/1998/NOTE-XML-data-0105

Livelli di codifica

L'elemento cXML è il corpo di un normale documento XML. L'inizio di un documento deve essere costituito da:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd">
<cXML version="1.1.007" xml:lang="en-US"
    payloadID="0c300508b7863dcclb_13550"
    timestamp="2000-01-09T01:36:05-08:00">
  ...
```

Questo documento viene normalmente trasmesso mediante il protocollo HTTP con il tipo MIME `text/xml` e un parametro `charset` corrispondenti al sistema di codifica nel documento. Poiché l'HTTP è un protocollo a otto bit puro, è possibile utilizzare qualsiasi sistema di codifica supportato dall'analizzatore del destinatario indipendentemente dal sistema di codifica del trasferimento del contenuto, ad esempio il sistema base64 o un sistema che utilizza stringhe racchiuse tra virgolette. Tutti gli analizzatori XML supportano il sistema di codifica UTF-8 che include tutti i caratteri Unicode. Pertanto le applicazioni dovrebbero usare questo sistema quando trasmettono documenti cXML.

Nota: In base alla norma RFC 2376 "XML Media Types", il parametro MIME `charset` sostituisce qualsiasi codifica specificata nella dichiarazione XML. Inoltre, il sistema di codifica predefinito per il tipo `text/xml` è `us-ascii` e non UTF-8 come citato nella sezione 4.3.3 di XML Specification. Per ragioni di chiarezza, i documenti cXML dovrebbero includere una codifica esplicita nella dichiarazione XML. Gli elementi MIME envelope dovrebbero usare un parametro `charset` corrispondente per il tipo `text/xml` o `application/xml`.

La trasmissione HTTP di un documento cXML può includere le intestazioni MIME e HTTP seguenti:

```
POST /cXML HTTP/1.0
Content-type: text/xml; charset="UTF-8"
Content-length: 1862
Accept: text/html, image/gif, image/jpeg, *, q=.2, */*; q=.2
User-Agent: Java1.1
Host: localhost:8080
Connection: Keep-Alive

<?xml version="1.0" encoding="UTF-8"?>
...
```

Trasmissione degli allegati

Quando si invia una `OrderRequest` che fa riferimento a file esterni, questi file possono trovarsi su un server accessibile dal fornitore o possono essere trasmessi con il documento cXML. Nel secondo caso è necessario usare l'elemento MIME envelope a più parti. Un requisito cXML per questo elemento envelope (oltre quelli fondamentali descritti nella norma RFC 2046 "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types") consiste nella specifica delle intestazioni Content-ID in ogni file allegato.

Nell'esempio riportato di seguito viene mostrato come deve essere strutturato un documento cXML con un'immagine JPEG allegata e senza le intestazioni HTTP mostrate sopra:

```
POST /cXML HTTP/1.0
Content-type: multipart/mixed; boundary=testo univoco

--testo univoco
Content-type: text/xml; charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
...
--testo univoco
Content-type: image/jpeg
Content-ID: <uniqueCID@cxml.org>
...
--testo univoco--
```

Questa struttura è quanto un analizzatore MIME del destinatario deve poter elaborare. Le applicazioni che impiegano il tipo descritto nella norma RFC 2387 "The MIME Multipart/Related Content-type" ottengono una quantità maggiore di informazioni se la struttura del documento è più dettagliata:

```
POST /cXML HTTP/1.0
Content-type: multipart/related; boundary=testo univoco;
    type="text/xml"; start=<uniqueCIDmain@cxml.org>

--testo univoco
Content-type: text/xml; charset="UTF-8"
Content-ID: <uniqueCIDmain@cxml.org>

<?xml version="1.0" encoding="UTF-8"?>
...
--testo univoco
Content-type: image/jpeg
Content-ID: <uniqueCID@cxml.org>
...
--testo univoco--
```

Gli analizzatori MIME del destinatario che non riconoscono il tipo `multipart/related` devono considerare identici i due esempi sopra riportati. Ogni parte della trasmissione MIME può avere anche una codifica `Content-transfer` e usarlo. Ciò non è necessario per la trasmissione HTTP. Le intestazioni `Content-description` e `Content-disposition` sono facoltative nel protocollo cXML sebbene forniscano informazioni utili.

Per ulteriori informazioni su come allegare file esterni agli ordini d'acquisto, vedere "Attachment" a pagina 75.

Header

L'elemento Header contiene informazioni sull'indirizzamento e l'autenticazione e resta invariato indipendentemente da quale specifica, Request o Response, sia contenuta nel corpo del messaggio cXML. Le applicazioni necessitano dei dati identificativi del richiedente ma non della convalida dell'esattezza di tali dati.

Nell'esempio riportato di seguito viene mostrato l'elemento Header:

```
<Header>
  <From>
    <Credential domain="AribaNetworkUserId">
      <Identity>admin@acme.com</Identity>
    </Credential>
  </From>
  <To>
    <Credential domain="DUNS">
      <Identity>012345678</Identity>
    </Credential>
  </To>
  <Sender>
    <Credential domain="AribaNetworkUserId">
      <Identity>sysadmin@ariba.com</Identity>
      <SharedSecret>abracadabra</SharedSecret>
    </Credential>
    <UserAgent>Ariba Network 1.1</UserAgent>
  </Sender>
</Header>
```

Gli elementi From e To corrispondono agli elementi From e To nei messaggi di posta SMTP; rappresentano la provenienza e la destinazione logiche dei messaggi. Sender rappresenta chi apre la connessione HTTP e invia il documento cXML.

Sender contiene l'elemento Credential che consente al destinatario di autenticare il mittente. Ciò assicura un'autenticazione affidabile senza dover ricorrere a un'infrastruttura che usi certificati digitali, sia per il mittente sia per il destinatario, a chiave pubblica. Per consentire al mittente di eseguire richieste, è sufficiente che il destinatario specifichi un nome utente e una password.

Inizialmente, gli elementi Sender e From sono uguali. Tuttavia, se il documento cXML passa attraverso degli hub di rete per l'e-commerce, l'elemento Sender cambia ad indicare il mittente corrente.

From

Questo elemento identifica l'autore della richiesta cXML. Può contenere più di un elemento Credential che consente ai richiedenti di identificarsi usando vari metodi. L'uso di più credenziali è analogo all'invio degli indirizzi SMTP e X.400 in un messaggio di posta elettronica.

To

Questo elemento identifica la destinazione della richiesta cXML. Analogamente all'elemento From, può contenere più di un elemento Credential per meglio identificare la destinazione.

Sender

Questo elemento consente al destinatario di identificare e autenticare chi ha aperto la connessione HTTP. La funzione di autenticazione dell'elemento Credential è più potente che negli elementi From o To in quanto il destinatario deve autenticare chi gli richiede di eseguire un'operazione.

Credential

Questo elemento contiene i valori di identificazione e autenticazione usati nei messaggi cXML.

Gli attributi di Credential sono i seguenti:

domain	Specifica il tipo di credenziali. Questo attributo consente di inserire nei documenti più tipi di credenziali per più domini di autenticazione. Ad esempio, per i messaggi inviati ad Ariba Network il dominio solitamente è AribaNetworkUserId o DUNS.
type (facoltativo)	Le richieste inviate o provenienti da un mercato identificano sia il mercato sia l'azienda che ne fa parte negli elementi Credential From o To. In tal caso, le credenziali per il mercato utilizzano l'attributo type impostato sul valore "marketplace".

Credential contiene un elemento Identity e può contenere un elemento SharedSecret o DigitalSignature. L'elemento Identity indica a chi fa riferimento Credential mentre gli elementi facoltativi di autenticazione verificano l'identità della parte.

L'elemento SharedSecret è usato quando l'elemento Sender dispone di una combinazione nome utente/password riconosciuta dal richiedente.

L'elemento `DigitalSignature` può essere usato se le due parti concordano sull'uso di un formato e di un'autorità di certificazione comune. L'attributo `type` in un elemento `DigitalSignature` specifica il tipo di certificato usato.

Nota: Non utilizzare elementi di autenticazione in documenti inviati tramite una comunicazione unilaterale. Questo metodo di trasporto esegue l'instradamento utilizzando i browser degli utenti. In questo modo, gli utenti possano visualizzare l'origine del documento, compresi gli elementi `Credential`.

Request

I client inviano richieste di operazioni. Per ogni elemento `envelope` cXML è consentito un solo elemento `Request` in modo da semplificare le implementazioni del server. Questo limite è necessario in quanto durante la lettura dei documenti cXML non deve verificarsi alcun annullamento del multiplexing. L'elemento `Request` può contenere virtualmente qualsiasi tipo di dati XML.

L'attributo di `Request` è il seguente:

deploymentMode (facoltativo)	Indica se la richiesta è una richiesta di verifica o di produzione. I valori consentiti sono "production" (valore predefinito) o "test".
--	--

Response

I server inviano le risposte per informare i client dei risultati delle operazioni. Poiché il risultato di alcune richieste potrebbe essere privo di dati, l'elemento `Response` potrebbe contenere solo un elemento `Status`. Un elemento `Response` può contenere anche dati a livello di applicazione. Negli scenari `PunchOut`, ciò equivale a un elemento `PunchOutSetupResponse`.

Status

Questo elemento riporta l'esito positivo o negativo di un'operazione di richiesta.

Gli attributi di Status sono i seguenti:

code	Il codice dello stato della richiesta. Questo codice è conforme al modello dei codici di stato HTTP. Ad esempio, 200 indica la riuscita di una richiesta.
text	Il testo del messaggio di stato. Questo testo aiuta l'utente a leggere i file di log ed è costituito da normali stringhe in lingua inglese.
xml:lang (facoltativo)	La lingua dei dati nell'elemento Status. Facoltativo per la compatibilità con il formato cXML 1.1. Potrebbe essere obbligatorio nelle versioni future del cXML.

Gli attributi dell'elemento Status indicano gli eventi relativi alla richiesta.

L'elemento Status può contenere qualsiasi dato necessario al richiedente. Il codice di stato 200/OK potrebbe non riportare alcun dato. Comunque si raccomanda che per il codice di stato 500/Internal Server Error vengano presentati gli errori effettivi dell'applicazione o dell'analisi XML. Questo errore consente di eseguire un debugging del server e una verifica dell'interoperabilità più approfonditi.

I server non dovrebbero includere altri elementi di risposta, come ad esempio un elemento PunchOutSetupResponse, a meno che il codice di stato non sia compreso nella classe 200 (ad esempio 200/OK).

La specifica HTTP 1.1 include molti codici di stato che non sono validi per il cXML. Poiché in gran parte dei casi il livello del cXML è superiore a quello dell'HTTP, molti errori, come 404/Not Found, vengono gestiti dal metodo di trasporto. I codici di stato 200/OK e 500/Internal Server Error sono molto frequenti. Gli errori di convalida nell'analisi di un documento di richiesta normalmente generano un errore di trasporto come l'errore HTTP 400/Bad Request.

Nella tabella riportata di seguito sono inclusi gli altri codici HTTP utilizzabili:

Il cXML comprende pochissimi codici di stato non HTTP:

- 550 – Impossibile raggiungere il server cXML successivo per completare una transazione che richiede le connessioni upstream. Questo codice può essere restituito da un hub intermedio quando il sito di un fornitore non è raggiungibile. Se le connessioni upstream riescono, gli hub intermedi dovrebbero restituire gli errori direttamente al client.
- 551 – Impossibile inoltrare la richiesta a causa di errori di configurazione del fornitore. Ad esempio, l'autenticazione di un hub intermedio per un fornitore non è riuscita. I client non possono correggere questo errore ma il problema può essere risolto prima del successivo tentativo del client.
- 560 – Errore temporaneo del server. Ad esempio, può verificarsi quando il server non è operativo perché è in corso la manutenzione. Il client deve riprovare più tardi.

Stato	Testo	Significato
200	OK	Il server è riuscito a eseguire la richiesta sebbene la risposta restituita potrebbe contenere avvisi o errori.
201	Accepted	Alcune elaborazioni potrebbero non essere ancora state completate. Come indicato in "StatusUpdateRequest" a pagina 89, il client deve attendere le successive transazioni StatusUpdate se questo stato viene restituito in risposta a una OrderRequest.
204	No Content	Tutte le informazioni della richiesta sono valide e sono state riconosciute. Il server non dispone di dati del tipo richiesto per la risposta. In un PunchOutOrderMessage, questo stato indica che la sessione PunchOut è terminata senza modificare il "carrello della spesa" o la richiesta di acquisto del client.
400	Bad Request	Richiesta inaccettabile per il server sebbene sia stata analizzata correttamente.
401	Unauthorized	Le credenziali fornite nella richiesta (elemento Sender) non sono state riconosciute dal server.
402	Payment Required	La richiesta deve includere un elemento Payment completo.
403	Forbidden	L'utente non dispone di privilegi sufficienti a eseguire questa richiesta.
406	Not Acceptable	Un alias per il codice 400: la richiesta è inaccettabile per il server sebbene sia stata analizzata correttamente.
409	Conflict	Lo stato corrente o i dati interni del server hanno impedito la richiesta di un'operazione di aggiornamento. Una richiesta identica potrebbe ripetersi in futuro, soprattutto dopo l'esecuzione di un'operazione.
412	Precondition Failed	Non è stata soddisfatta una condizione necessaria di una richiesta (ad esempio una sessione PunchOut per una modifica PunchOutSetupRequest). Normalmente questo stato indica che il client ha ignorato alcune parti di una trasmissione precedente da un server come, ad esempio, l'attributo operationAllowed di un elemento PunchOutOrderMessageHeader).
417	Expectation Failed	Non è stata soddisfatta una condizione relativa a una risorsa implicita nella richiesta. Ad esempio, SupplierDataRequest ha richiesto delle informazioni su un fornitore sconosciuto al server. Questo stato può indicare la perdita di informazioni sul client o sul server.
500	Internal Server Error	Il server non è stato in grado di completare la richiesta.
501	Not Implemented	Il server non è riuscito ad implementare una particolare richiesta. Ad esempio è possibile che PunchOutSetupRequest o l'operazione richiesta non siano supportate. Normalmente questo stato indica che il client ha ignorato il profilo del server.

Quando ricevono codici non riconosciuti, i client cXML devono gestirli in base alla classe a cui appartengono. Pertanto i client vecchi devono considerare tutti i nuovi codici 2xx come 200, 4xx come 400 e 5xx come 500. Ciò consente ulteriori espansioni del protocollo cXML e dei codici specifici del server senza compromettere l'interoperabilità.

Modello unilaterale (asincrono)

Al contrario delle transazioni richiesta-risposta, i messaggi unilaterali non sono limitati al metodo di trasporto HTTP.

I messaggi unilaterali vengono utilizzati nelle situazioni in cui un canale HTTP non è appropriato (operazione di tipo richiesta-risposta sincrona). Nella figura seguente viene riportato un esempio di come A e B possono comunicare tramite messaggi invece di usare una transazione richiesta-risposta.



Un tipico scenario potrebbe essere il seguente:

1. A formatta e codifica un documento cXML in base al metodo di trasporto riconosciuto da B.
2. A invia il documento usando il metodo di trasporto riconosciuto. A non aspetta, e non può aspettare, una risposta da B.
3. B riceve il documento cXML e lo decodifica dalle informazioni del flusso di trasporto.
4. B elabora il documento.

Nel modello unilaterale, A e B *non* utilizzano un ciclo richiesta-risposta esplicito. Ad esempio, nel corso della trasmissione di messaggi unilaterali, possono arrivare messaggi da altre provenienze e possono aver luogo altre conversazioni.

Per specificare completamente una transazione unilaterale, è necessario documentare anche il metodo di trasporto usato per il messaggio. Per le transazioni cXML che usano l'approccio unilaterale, vengono specificati il metodo di trasporto e il sistema di codifica. Un esempio di transazione unilaterale è `PunchOutOrderMessage`.

La struttura dei messaggi unilaterali è simile al modello richiesta-risposta:

```
<cXML>
  <Header>
    Informazioni sull'intestazione...
  </Header>
  <Message>
    Informazioni sul messaggio...
  </Message>
</cXML>
```

L'elemento Header viene considerato esattamente come nel caso del modello richiesta-risposta. Anche l'elemento cXML è identico a quello descritto precedentemente. La differenza tra un messaggio unilaterale e un messaggio richiesta-risposta è nell'uso dell'elemento Message al posto dell'elemento Request o Response. La sezione seguente descrive dettagliatamente l'elemento Message.

Message

Questo elemento contiene tutte le informazioni a livello del corpo in un messaggio cXML. Può contenere un elemento Status facoltativo, identico a quello di un elemento Response e che può essere usato nei messaggi per rispondere coerentemente ai messaggi di richiesta.

Gli attributi di Message sono i seguenti:

deploymentMode (facoltativo)	Indica se la richiesta è una richiesta di verifica o di produzione. I valori consentiti sono "production" (valore predefinito) o "test".
inReplyTo (facoltativo)	Specifica il messaggio al quale questo messaggio risponde. Il contenuto dell'attributo inReplyTo corrisponderà al contenuto dell'attributo payloadID di un messaggio ricevuto precedentemente. Ciò può essere usato per costruire una conversazione bilaterale con molti messaggi.

L'attributo inReplyTo può fare riferimento anche all'attributo payloadID di un precedente documento di richiesta o risposta. Quando una transazione di tipo richiesta-risposta inizia una "conversazione" con più interazioni unilaterali, il primo messaggio può includere l'attributo payloadID della relativa richiesta o risposta più recente che va nell'altra direzione. Ad esempio, un messaggio che contiene un elemento PunchOutOrderMessage può includere un attributo inReplyTo che contiene il payloadID di PunchOutSetupRequest che ha avviato la sessione PunchOut (il BuyerCookie incluso nei documenti PunchOut esegue una funzione simile all'attributo inReplyTo).

Metodi di trasporto

Per i messaggi unilaterali generalmente si utilizzano due metodi di trasporto: HTTP e URL-Encoded. Si tratta di due tra i migliori metodi attualmente disponibili e molti altri saranno supportati nel futuro.

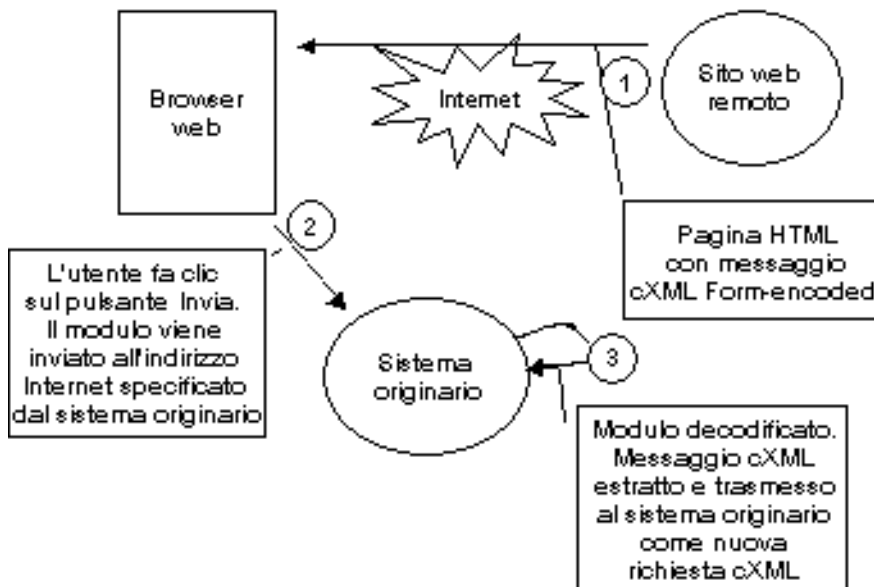
HTTP

Il metodo HTTP viene utilizzato per la comunicazione unilaterale e consente alle applicazioni di acquisizione di reperire informazioni. La transazione `GetPendingRequest`, trattata a pagina 103, utilizza la comunicazione HTTP unilaterale.

URL-Encoded

Questo metodo di trasporto è esemplificato dalla modalità di esecuzione della transazione `PunchOutOrderMessage`. Il metodo URL-Encoded consente l'integrazione tra un sito web remoto e le applicazioni di acquisizione. Inoltre consente di evitare di ricorrere a un server in ascolto sul sistema del compratore direttamente accessibile mediante Internet.

Il messaggio cXML `PunchOutOrderMessage` non viene inviato direttamente all'applicazione di acquisizione dal sito web remoto ma viene codificato come campo nascosto HTML Form e inviato all'indirizzo Internet specificato nell'elemento `BrowserFormPost` di `PunchOutSetupRequest`. Quando l'utente fa clic su Estrai, il sito web invia i dati all'applicazione di acquisizione come richiesta HTML Form Submit. Il diagramma riportato di seguito illustra queste fasi:



Di seguito viene descritta la semantica di codifica e decodifica.

Codifica tramite elemento Form

Il documento `PunchOutOrderMessage` è URL-Encoded (secondo la specifica HTTP) e assegnato al campo nascosto FORM denominato `cXML-urlencoded`. All'elemento HTML FORM vengono assegnati gli attributi METHOD con valore POST e ACTION costituiti dall'indirizzo Internet passato all'elemento `BrowserFormPost` di `PunchOutSetupRequest`. Ad esempio:

```
<FORM METH.OD=POST
  ACTION="http://workchairs.com:1616/punchoutexit">
  <INPUT TYPE=HIDDEN NAME="cXML-urlencoded"
    VALUE="URL-Encoded PunchOutOrderMessage document">
  <INPUT TYPE=SUBMIT VALUE="Proceed">
</FORM>
```

L'esempio precedente potrebbe essere contenuto anche in altre tag HTML della pagina per descrivere in dettaglio il contenuto del cestino della spesa.

Nota: Quando i server web inviano il campo cXML-urlencoded, la codifica non è ancora URL. Questo sistema di codifica è necessario solo quando il documento viene inviato dai browser web - quando gli utenti fanno clic su Estrai nell'esempio precedente. I browser web soddisfano questo requisito. Il server web deve codificare in HTML solo il valore del campo, le virgolette di escape e altri caratteri speciali in modo che il documento possa essere visualizzato correttamente dall'utente.

Il nome cXML-urlencoded non distingue tra lettere maiuscole e lettere minuscole.

Per i dati cXML-urlencoded, l'analizzatore del destinatario non può assumere un parametro charset oltre il valore predefinito per il tipo text/xml. Nell'elemento HTTP POST non viene inserita alcuna informazione sul sistema di codifica dei caratteri. Il server web del destinatario non può determinare la codifica della pagina HTML che contiene il campo nascosto. Pertanto il documento inviato in questo modo deve utilizzare una codifica dei caratteri us-ascii. Qualsiasi carattere, compresi quelli URL-Encoded come "%XX", individuato nel documento di origine XML deve appartenere al gruppo "us-ascii". Gli altri simboli Unicode possono essere codificati utilizzando le entità carattere nel documento di origine.

Sistema di codifica Base64

Il campo nascosto cXML-base64 facilita la gestione di documenti internazionali. I documenti cXML che contengono i simboli esterni al gruppo "us-ascii" devono usare questo campo invece del campo nascosto cXML-urlencoded. Il nuovo campo è equivalente al precedente, ma consente la codifica Base64 dell'intero documento durante il trasporto; pertanto non vengono più utilizzate le codifiche HTML del browser e URL-Encoded del server web. Il sistema di codifica Base64 è descritto nella norma RFC 2045 "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies".

Il nome cXML-base64 non distingue tra lettere maiuscole e lettere minuscole.

Il sistema Base64 mantiene la codifica dei caratteri di un documento cXML nel percorso che parte dal sito del fornitore, passa per il browser e arriva al server web del destinatario nel client. Sebbene insieme alle informazioni inviate non vi sia alcun parametro charset, il documento decodificato (dopo la rimozione del codice relativo al trasferimento) può essere considerato di tipo application/xml. Questo sistema di codifica consente all'analizzatore del destinatario di rispettare qualsiasi attributo encoding specificato nella dichiarazione XML. Per questo campo, come per qualsiasi documento application/xml, il sistema di codifica dei caratteri predefinito è UTF-8.

Uno di questi campi nascosti, cXML-urlencoded o cXML-base64, deve risultare nei dati inviati all'applicazione di acquisizione. Sebbene i destinatari debbano prima di tutto ricercare cXML-base64 nei dati, è inutile inviare entrambi i campi.

Decodifica ed elaborazione dell'elemento Form

L'applicazione di acquisizione, che aveva fornito la codifica URL appropriata, riceve un elemento HTML FORM POST che contiene i dati Form come descritto sopra. Il processore Form POST prima di tutto ricerca la variabile cXML-base64, estrae il valore e ne decodifica il contenuto Base64. Se tale campo non esiste nei dati, il processore Form POST ricerca la variabile cXML-urlencoded, estrae il messaggio cXML URL-Encoded e lo decodifica. Il contenuto decodificato del campo viene quindi elaborato come se fosse stato ricevuto attraverso un normale ciclo HTTP di tipo richiesta-risposta.

Il tipo del documento dopo la decodifica varia, con diverse possibilità di codifica dei caratteri:

- La variabile cXML-urlencoded è di tipo text/xml senza alcun attributo charset. Pertanto è limitata al sistema di codifica dei caratteri us-ascii. L'analizzatore del destinatario deve ignorare qualsiasi attributo encoding nella dichiarazione XML del documento cXML poiché il browser potrebbe aver cambiato il sistema di codifica.
- La variabile cXML-base64 è di tipo application/xml e pertanto può avere qualsiasi sistema di codifica dei caratteri indicato dall'attributo encoding della dichiarazione XML contenuta, eventualmente presente.

La differenza principale tra questa transazione e una normale transazione di tipo richiesta-risposta consiste nel fatto che non può essere generata alcuna risposta dato che non esiste alcuna connessione HTTP per mezzo della quale inviare la risposta.

Elementi fondamentali

Nella specifica cXML vengono utilizzati le entità e gli elementi seguenti. La maggior parte delle definizioni costituiscono il vocabolario di base con il quale vengono descritti i documenti commerciali di primaria importanza. Vengono definiti anche gli elementi e le entità tipo comunemente usati per rappresentare gli oggetti a livello inferiore.

Entità tipo

La maggior parte di queste definizioni provengono dalla nota XML-Data in W3C (World Wide Web Consortium). Viene definito anche un numero minore di entità che non provengono dalla nota XML-Data. Questi tipi sono trattati anche in "Elemento cXML envelope" a pagina 52.

isoLangCode

Un codice per la lingua ISO dello standard ISO 639.

isoCountryCode

Un codice per la nazione ISO dello standard ISO 3166.

xmlLangCode

Un codice per la lingua definito dalla specifica XML 1.0 (disponibile all'indirizzo www.w3.org/TR/1998/REC-xml-19980210.html). Nella maggior parte dei casi, questo codice include un codice per la lingua ISO 639 e può includere un codice per la nazione ISO 3166 separati da un trattino. Al contrario di quanto previsto dalla raccomandazione XML, nel cXML non devono essere usati i codici per lingua IANA o privati. È possibile utilizzare i codici secondari IANA e privati ma devono essere inseriti dopo un codice per la nazione ISO 3166 valido.

Come formato del codice per la lingua cXML si consiglia di usare `xx[-YY[-zzz]]*?`, dove `xx` rappresenta un codice per la lingua ISO 639, `YY` un codice per la nazione ISO 3166 e `zzz` un codice secondario IANA o privato. Si consiglia di utilizzare sempre un codice per la nazione. Normalmente, per specificare il codice per la lingua si utilizzano le lettere minuscole mentre per specificare il codice per la nazione quelle maiuscole sebbene questa convenzione non sia necessaria per il raffronto dei codici.

unitOfMeasure

UnitOfMeasure descrive la modalità di confezionamento o di spedizione del prodotto. Deve essere conforme ai codici usati per le unità di misura UN/CEFACT. Per l'elenco dei codici UN/CEFACT, visitare il sito web all'indirizzo www.unece.org/cefact.

URL (indirizzo Internet)

Indirizzo Internet (URL - Uniform Resource Locator) secondo la definizione dello standard HTTP/1.1.

Elementi di base

Questi elementi, usati in tutta la specifica, sono sia generici, come Name ed Extrinsic, sia specifici, come Money.

Transazione Profile

I documenti ProfileRequest e ProfileResponse devono essere supportati dalle implementazioni del server cXML 1.1. Questa transazione può essere utilizzata per recuperare le caratteristiche del server, inclusa la versione cXML supportata, le transazioni e le opzioni per queste transazioni.

La risposta dovrebbe elencare tutte le richieste supportate in un particolare sito web e non necessariamente tutte quelle supportate dall'azienda. I fornitori che possono ricevere i documenti OrderRequest e inviare vari messaggi o avviare transazioni di tipo richiesta-risposta descrivono il proprio supporto OrderRequest nella transazione Profile.

La transazione Profile può essere utilizzata per "effettuare il ping" di un server all'interno del protocollo cXML.

ProfileRequest

Questo elemento è privo di contenuto. Viene semplicemente instradato al server cXML appropriato utilizzando l'elemento Header. Il server risponde con un singolo documento ProfileResponse, come descritto di seguito. Le uniche parti dinamiche di questa risposta sono costituite dagli attributi payloadId e timestamp dell'elemento cXML. In questo caso particolare, il fornitore non è tenuto a fornire risposte in più lingue.

Di seguito viene riportato un esempio di questo tipo di richiesta:

```
<Request>
  <ProfileRequest />
</Request>
```

ProfileResponse

Questo elemento contiene l'elenco delle transazioni supportate, della loro posizione e di qualsiasi opzione supportata. Nel caso non sia ancora stata definita alcuna opzione, l'elemento ProfileResponse potrebbe essere di questo tipo:

```
<ProfileResponse effectiveDate="2001-03-03T12:13:14-05:00">
  <Option name="Locale">1</Option>
  ...
  <Transaction requestName="PunchOutSetupRequest">
    <URL>http://www.workchairs.com/cXML/PunchOut.asp</URL>
    <Option name="operationAllowed">create inspect</Option>
    <Option name="dynamic pricing">0</Option>
    ...
  </Transaction>
  ...
</ProfileResponse>
```

Un documento ProfileResponse ricevuto da un fornitore corrente potrebbe essere come il seguente:

```
<ProfileResponse effectiveDate="2000-01-01T05:24:29-08:00">
  <Transaction requestName="OrderRequest">
    <URL>http://workchairs.com/cgi/orders.cgi</URL>
  </Transaction>
  <Transaction requestName="PunchOutSetupRequest">
    <URL>http://workchairs.com/cgi/PunchOut.cgi</URL>
  </Transaction>
</ProfileResponse>
```

L'attributo di ProfileResponse è il seguente:

effectiveDate	La data e l'ora in cui questi servizi sono resi disponibili. Non è possibile utilizzare date future.
----------------------	---

Option

Il valore per un'opzione definita (per tutto il servizio o per una specifica transazione). Nessuna opzione è stata ancora definita.

L'attributo di Option è il seguente:

name	Il nome di questa opzione. Questo attributo non dovrebbe essere visualizzato direttamente poiché il profilo è destinato all'utilizzo da parte della macchina.
-------------	---

Transaction

Le descrizione di una transazione supportata da questo servizio. La definizione del profilo corrente indica le ubicazioni a cui inviare specifiche richieste. Nelle versioni future del cXML saranno aggiunte definizioni Option e verranno estese le informazioni sul profilo in modo da comprendere ulteriori dati sulle richieste supportate.

L'elemento Transaction deve contenere un elemento URL.

L'attributo di Transaction è il seguente:

requestName	Una richiesta specifica che il server accetta all'indirizzo Internet indicato. I valori possono essere:	
	ProfileRequest PunchOutSetupRequest GetPendingRequest SupplierListRequest SupplierDataRequest	OrderRequest StatusUpdateRequest SubscriptionListRequest SubscriptionContentRequest

Definizione degli ordini

I documenti per gli ordini cXML sono costituiti da un elemento OrderRequest e da una risposta generica. L'elemento OrderRequest è analogo a un ordine d'acquisto. La risposta rappresenta la conferma che il fornitore ha ricevuto l'ordine d'acquisto. Non costituisce l'impegno ad eseguire l'ordine ma solo la conferma che l'ordine è stato ricevuto correttamente.

OrderRequest

Nell'esempio riportato di seguito viene mostrata la struttura dell'elemento OrderRequest:

```
<OrderRequest>
  <OrderRequestHeader ... >
    ...
  </OrderRequestHeader>
  <ItemOut ... >
    ...
  </ItemOut>
  <ItemOut ... >
    ...
  </ItemOut>
</OrderRequest>
```

OrderRequestHeader

Nell'esempio riportato di seguito viene mostrato dettagliatamente l'elemento OrderRequestHeader:

```
<OrderRequestHeader orderID="DO1234"
  orderDate="1999-03-12T13:30:23+8.00"
  type="new"
  requisitionID="R1234">
  <Total>
    <Money currency="USD">12.34</Money>
  </Total>
  <ShipTo>
    <Address>
      <Name xml:lang="en">Acme Corporation</Name>
      <PostalAddress name="Headquarters">
        <DeliverTo>Joe Smith</DeliverTo>
        <DeliverTo>Mailstop M-543</DeliverTo>
        <Street>123 Anystreet</Street>
        <City>Sunnyvale</City>
        <State>CA</State>
        <PostalCode>90489</PostalCode>
        <Country isoCountryCode="US">USA</Country>
      </PostalAddress>
    </Address>
  </ShipTo>
  <BillTo>
    <Address>
      <Name xml:lang="en">Acme Corporation</Name>
      <PostalAddress name="Finance Building">
        <Street>124 Anystreet</Street>
        <City>Sunnyvale</City>
        <State>CA</State>
        <PostalCode>90489</PostalCode>
        <Country isoCountryCode="US">USA</Country>
      </PostalAddress>
    </Address>
  </BillTo>
  <Shipping>
    <Money currency="USD">12.34</Money>
    <Description xml:lang="en-US">FedEx 2-day</Description>
  </Shipping>
  <Tax>
    <Money currency="USD">12.34</Money>
    <Description xml:lang="en">CA State Tax</Description>
  </Tax>
  <Payment>
    <PCard number="1234567890123456" expiration="1999-03-12"/>
  </Payment>
  <Contact role="purchasingAgent">
```



```
<Name xml:lang="en-US">Mr. Smart E. Pants</Name>
<Email>sepants@acme.com</Email>
<Phone name="Office">
  <TelephoneNumber>
    <CountryCode isoCountryCode="US">1</CountryCode>
    <AreaOrCityCode>800</AreaOrCityCode>
    <Number>555-1212</Number>
  </TelephoneNumber>
</Phone>
</Contact>
<Comments xml:lang="en-US">
  Commenti codificati in XML.
</Comments>
<Followup>
  <URL>http://acme.com/cgi/orders.cgi</URL>
</Followup>
</OrderRequestHeader>
```

Gli attributi di OrderRequestHeader sono i seguenti:

orderID	L'identificativo per questo ordine. Analogo al numero dell'ordine d'acquisto.
orderDate	La data e l'ora in cui l'ordine è stato piazzato, nel formato ISO 8601.
type (facoltativo)	Il tipo della richiesta: new (valore predefinito), update o delete.
requisitionID (facoltativo)	L'identificativo della richiesta di acquisto del compratore per tutto l'ordine. Deve essere uguale all'attributo orderID e può non essere incluso. Non è necessario includerlo se è stato specificato l'attributo requisitionID in un qualsiasi elemento ItemOut.
shipComplete (facoltativo)	Una preferenza in alternativa alle spedizioni parziali. L'unico valore consentito è "yes". In base all'impostazione predefinita, gli articoli vengono spediti non appena sono disponibili. Poiché gli ordini possono includere articoli con elementi ShipTo diversi, se si utilizza shipComplete="yes", solo i gruppi di articoli che hanno le stesse destinazioni di spedizione dovrebbero essere trattenuti finché non vengono completati.

Gli elementi OrderRequestHeader e ItemOut (quando esteso con ItemDetail) contengono informazioni simili. Laddove OrderRequestHeader include informazioni globali sulla fatturazione (BillTo) e sul pagamento (Payment), ItemOut descrive invece i singoli articoli (in ItemID, ItemDetail e Distribution).

Non utilizzare le informazioni riportate nell'elemento `OrderRequestHeader` come impostazione predefinita per gli elementi specifici dell'articolo. Gli elementi `ShipTo`, `Shipping`, `Contact` e ogni elemento `Extrinsic`, eventualmente presenti, devono essere riportati o con ogni `ItemOut` o in `OrderRequestHeader`. Gli elementi `Comments` e `Tax` possono essere usati contemporaneamente ad entrambi i livelli. Tuttavia, i vari elementi `Comments` non devono contenere le stesse informazioni e l'elemento `Tax` a livello di intestazione deve contenere un totale per l'ordine.

Total

Questo elemento contiene l'importo monetario totale dell'ordine. Vi è contenuto l'elemento `Money`.

ShipTo/BillTo

Questi elementi contengono gli indirizzi delle entità `Ship To` e `Bill To` riportate nel documento `OrderRequest`.

Un ordine deve essere fatturato a una singola entità. Pertanto l'elemento `BillTo` viene riportato solo in `OrderRequestHeader`. Gli articoli di un ordine possono essere inviati a più destinazioni. Analogamente all'elemento `Shipping` (vedere la sezione successiva), l'elemento `ShipTo` può essere riportato in `OrderRequestHeader` o in singoli elementi `ItemOut`.

Shipping

Questo elemento descrive la modalità di spedizione degli articoli nella richiesta e il costo dell'operazione. Se l'elemento `Shipping` è presente in `OrderRequestHeader`, non deve essere riportato nei singoli elementi `ItemOut`. Se, invece, non è presente in `OrderRequestHeader`, deve essere riportato negli elementi `ItemOut`.

Tax

Questo elemento contiene le tasse associate all'ordine. È presente se la società acquirente calcola le tasse. Se contenuto in un elemento `OrderRequestHeader`, `Tax` descrive le tasse totali per un ordine. Gli elementi `Tax` a livello di articolo possono descrivere i singoli importi delle tasse.

Payment

Questo elemento descrive il mezzo di pagamento usato per pagare gli articoli richiesti. Nell'esempio precedente, l'elemento `Payment` contiene un elemento `PCard` che codifica una carta d'acquisto nel documento cXML. Per il futuro saranno definiti e supportati altri mezzi di pagamento.

Contact

Le informazioni sui contatti utilizzate dal fornitore per il follow-up dell'ordine. Questo elemento identifica una persona e fornisce l'elenco di tutte le modalità per mettersi in contatto con la persona o l'entità. L'unico elemento obbligatorio è il nome (Name) del contatto. È possibile usare, anche più volte, PostalAddress (non consigliato per un'immediata risoluzione dei problemi dell'ordine), Email, Phone, Fax e URL.

Le società acquirenti possono scegliere se usare questo elemento per identificare il richiedente originale, l'amministratore del sistema delle applicazioni di acquisizione o altri contatti incaricati della risoluzione dei problemi relativi agli ordini. Per un ordine, le informazioni di Contact possono essere diverse sia da quelle di BillTo sia da quelle di ShipTo.

Gli attributi di Contact sono i seguenti:

role (facoltativo)	La qualifica della persona nel processo di acquisizione. I valori possono essere: endUser, administrator, purchasingAgent, technicalSupport, customerService o sales.
------------------------------	---

Lo stesso elemento Contact role non deve essere riportato sia a livello di intestazione sia a livello di articolo.

Non esiste un ruolo predefinito in quanto il contenuto dell'elemento Contact può variare. Pertanto le applicazioni cXML considerano un elemento Contact privo dell'attributo role come un ulteriore ruolo.

Comments

Un testo comprensibile che qualsiasi compratore può inviare all'interno degli ordini d'acquisto. L'uso di questa stringa di dati non è destinato ai sistemi automatizzati nei siti dei fornitori.

L'elemento Comments può contenere un elemento Attachment per includere file esterni.

Attachment

L'elemento Comments consente di allegare file esterni per aggiungere informazioni agli ordini d'acquisto. L'elemento Attachment è riportato nell'elemento Comments e contiene solo un riferimento alla parte MIME esterna dell'allegato. Tutti gli allegati dovrebbero essere inviati in una singola trasmissione a più parti con il documento OrderRequest. Anche se ciò non fosse possibile, il contentID fornito dall'elemento Attachment deve essere utilizzabile per recuperare l'allegato.

Per dettagli sul trasferimento dei file allegati, vedere "Trasmissione degli allegati" a pagina 55.

L'elemento Attachment contiene un singolo indirizzo Internet con schema "cid:". Di seguito è riportato un esempio di un file allegato a un documento cXML:

```
<Comments>
  <Attachment>
    <URL>cid: uniqueCID@cxml.org</URL>
  </Attachment>
  Please see attached image for my idea of what this
  should look like
</Comments>
```

L'elemento Comments è presente in molti punti del protocollo cXML ma può contenere l'elemento Attachment solo nei documenti OrderRequest.

Followup

Specifica l'indirizzo Internet a cui devono essere inviati i documenti StatusUpdateRequest futuri. Si tratta dell'ubicazione di input per qualsiasi documento successivo che fa riferimento al documento OrderRequest corrente.

Extrinsic

Questo elemento contiene informazioni MRI (Machine-Readable Information) relative all'ordine ma non definite dal protocollo cXML. Mentre l'elemento Comments contiene delle informazioni per l'utente, gli elementi Extrinsic contengono dati che probabilmente appariranno in documenti successivi. A questo livello, Extrinsic amplia la descrizione di tutti gli articoli contenuti nell'ordine d'acquisto. Alcune informazioni di tipo Extrinsic possono anche descrivere l'ordine d'acquisto globalmente senza conseguenze sul significato degli elementi ItemOut contenuti.

Ogni specifico elemento Extrinsic può essere riportato solo una volta negli elenchi associati all'elemento OrderRequestHeader e a singoli elementi ItemOut (negli elementi ItemDetail contenuti). Lo stesso nome non deve essere presente negli elenchi OrderRequestHeader e in nessun elenco associato agli elementi ItemOut. Se in tutti gli elenchi ItemOut sono ripetuti lo stesso nome e lo stesso valore Extrinsic, devono essere spostati nell'elemento OrderRequestHeader.

L'elemento Extrinsic può essere riportato anche negli elementi IndexItem, PunchOutSetupRequest e ContractItem. Questi contesti sono descritti più avanti in questo documento.

ItemOut

Nell'esempio riportato di seguito viene mostrato un semplice elemento ItemOut valido.

```
<ItemOut quantity="1">
  <ItemID>
    <SupplierPartID>5555</SupplierPartID>
  </ItemID>
</ItemOut>
```

Gli attributi di ItemOut sono i seguenti:

quantity	Il numero degli articoli desiderato. Per alcune unità di misura è possibile usare le frazioni. Il valore potrebbe essere già stato controllato dal fornitore durante una sessione PunchOut. Non deve mai essere negativo.
lineNumber (facoltativo)	La posizione di questo articolo all'interno di un ordine. Questo valore ordinale aumenta di uno per ogni ItemOut in un "nuovo" OrderRequest. I client devono specificare sempre questo attributo in un documento OrderRequest sebbene possa non rivelarsi utile in altri contesti ItemOut.
requisitionID (facoltativo)	L'identificativo della richiesta di acquisto del compratore per questa voce d'ordine. Non deve essere incluso se è stato specificato requisitonID nell'elemento OrderRequestHeader.
requestedDeliveryDate (facoltativo)	La data in cui è stata richiesta la consegna dell'articolo. Questo attributo consente di specificare date di consegna a livello di articolo nel documento OrderRequest. Deve essere nel formato ISO 8601.

L'attributo lineNumber resta costante per tutti gli articoli indipendentemente dagli aggiornamenti dell'ordine. L'eliminazione di articoli da un ordine non cambia l'attributo lineNumber degli articoli rimanenti. Ai nuovi articoli vengono assegnati numeri più alti di quelli già inclusi nell'ordine. Una modifica apportata a un articolo esistente, ad esempio l'aumento di una quantità, non ha effetti sull'attributo lineNumber di quell'articolo.

Nell'esempio riportato di seguito viene mostrato un elemento ItemOut più complesso.

```
<ItemOut quantity="2" lineNumber="1"
  requestedDeliveryDate="1999-03-12">
  <ItemID>
    <SupplierPartID>1233244</SupplierPartID>
    <SupplierPartAuxiliaryID>ABC</SupplierPartAuxiliaryID>
  </ItemID>
  <ItemDetail>
    <UnitPrice>
      <Money currency="USD">1.34</Money>
    </UnitPrice>
  </ItemDetail>
</ItemOut>
```

```

</UnitPrice>
<Description xml:lang="en">hello</Description>
<UnitOfMeasure>EA</UnitOfMeasure>
<Classification domain="SPSC">12345</Classification>
<ManufacturerPartID>234</ManufacturerPartID>
<ManufacturerName xml:lang="en">foobar</ManufacturerName>
<URL>www.bar.com</URL>
</ItemDetail>
<ShipTo>
  <Address>
    <Name xml:lang="en">Acme Corporation</Name>
    <PostalAddress name="Headquarters">
      <Street>123 Anystreet</Street>
      <City>Sunnyvale</City>
      <State>CA</State>
      <PostalCode>90489</PostalCode>
      <Country isoCountryCode="US">USA</Country>
    </PostalAddress>
  </Address>
</ShipTo>
<Shipping>
  <Money currency="USD">1.34</Money>
  <Description xml:lang="en-US">FedEx 2-day</Description>
</Shipping>
<Tax>
  <Money currency="USD">1.34</Money>
  <Description xml:lang="en">foo</Description>
</Tax>
<Distribution>
  <Accounting name="DistributionCharge">
    <Segment type="G/L Account" id="23456"
      description="Entertainment"/>
    <Segment type="Cost Center" id="2323"
      description="Western Region Sales"/>
  </Accounting>
  <Charge>
    <Money currency="USD">.34</Money>
  </Charge>
</Distribution>
<Distribution>
  <Accounting name="DistributionCharge">
    <Segment type="G/L Account" id="456"
      description="Travel"/>
    <Segment type="Cost Center" id="23"
      description="Europe Implementation"/>
  </Accounting>
  <Charge>
    <Money currency="USD">1</Money>
  </Charge>
</Distribution>

```

```

        <Comments xml:lang="en-US">
            Commenti codificati in XML.
        </Comments>
    </ItemOut>

```

L'elemento ItemDetail consente di inviare ulteriori dati al fornitore invece che solo l'identificativo univoco dell'articolo rappresentato da ItemID.

Gli elementi ShipTo, Shipping, Tax, Contact, Comments e Extrinsic (alcuni sono nidificati in ItemDetail) sono identici a quelli che possono essere presenti nell'elemento OrderRequestHeader. Questi elementi consentono di rappresentare i dati, relativi ad esempio alla spedizione, al tipo di spedizione e al costo associato, in base all'articolo. Utilizzare questi elementi a livello di OrderRequestHeader o a livello ItemOut ma non ad entrambi i livelli.

Distribution

La distribuzione suddivide il costo di un articolo tra più parti. I fornitori restituiscono l'elemento Distribution nelle fatture per facilitare il processo di riconciliazione del compratore.

Accounting

L'elemento Accounting raggruppa gli elementi Segments per identificare l'entità a cui addebitare l'importo.

L'attributo di Accounting è il seguente:

name	Il nome di questa combinazione di elementi Accounting.
------	--

Gli attributi di Segment sono i seguenti:

type	Un nome che identifica questo elemento Segment rispetto agli altri presenti nell'elemento Accounting.
id	L'identificativo univoco per questo tipo di elemento Segment. Questo valore potrebbe essere il codice dell'account effettivo se il tipo è "Cost Center".

Charge

Questo elemento specifica l'importo da addebitare all'entità rappresentata dall'elemento Accounting.

Risposta a un documento OrderRequest

Si tratta della parte risposta della transazione richiesta-risposta sincrona. Nell'esempio riportato di seguito viene mostrata la risposta a un documento OrderRequest:

```
<cXML version="1.1.007" payloadID="9949494" xml:lang="en"
  timestamp="1999-03-12T18:39:09-08:00">
  <Response>
    <Status code="200" text="OK"/>
  </Response>
</cXML>
```

Come mostrato sopra, si tratta di una semplice risposta. In questo caso non vi sono elementi "OrderResponse" poiché i soli dati che devono essere restituiti al richiedente sono quelli della parte Status della risposta.

La risposta indica al richiedente che la parte remota della connessione HTTP ha analizzato correttamente il documento OrderRequest e avviato le necessarie operazioni. Non contiene conferme a livello di ordine come, ad esempio, quali articoli possono essere spediti o quali consegne devono essere differite.

Transazione PunchOut

Le definizioni dei messaggi PunchOut sono messaggi di tipo richiesta-risposta contenuti negli elementi Request e Response. Tutti i messaggi seguenti devono essere implementati dai fornitori per supportare il PunchOut.

PunchOutSetupRequest

PunchOutSetupRequest e PunchOutSetupResponse costituiscono la coppia richiesta-risposta utilizzata per impostare una sessione PunchOut in un sistema remoto. Il client utilizza questi elementi per identificare l'applicazione di acquisizione, inviare informazioni sull'impostazione e ricevere una risposta che indica da dove avviare una sessione di esplorazione HTML sul sito web remoto.

Un elemento PunchOutSetupRequest è contenuto nell'elemento Request. Nell'esempio riportato di seguito viene mostrato un elemento PunchOutSetupRequest.

```
<PunchOutSetupRequest operation="create">
  <BuyerCookie>34234234ADFSDF234234</BuyerCookie>
  <Extrinsic name="department">Marketing</Extrinsic>
  <BrowserFormPost>
    <URL>http://or.ms.acme.com:1616/punchoutexit</URL>
  </BrowserFormPost>
</PunchOutSetupRequest>
```



```
</BrowserFormPost>
<SelectedItem>
  <ItemID>
    <SupplierPartID>54543</SupplierPartID>
  </ItemID>
</SelectedItem>
<SupplierSetup>
  <URL>http://workchairs.com/cxml</URL>
</SupplierSetup>
</PunchOutSetupRequest>
```

L'attributo di PunchOutSetupRequest è il seguente:

operation	Specifica il tipo di PunchOutSetupRequest, ovvero creazione, ispezione o modifica ("create", "inspect" o "edit").
------------------	---

Questo elemento contiene anche gli elementi seguenti: BuyerCookie, Extrinsic, BrowserFormPost, Contact, ShipTo, SelectedItem, SupplierSetup e un elemento di elenco ItemOut. Solo l'elemento BuyerCookie è obbligatorio. La struttura degli elementi Extrinsic, Contact e ShipTo è descritta più dettagliatamente in "OrderRequestHeader" a pagina 72 mentre l'elemento ItemOut è descritto in "ItemOut" a pagina 77. Poiché le sessioni PunchOut hanno luogo prima di un'ordinazione, in questo contesto, all'esterno di un OrderRequest, non è necessario includere gli elementi Distribution e Comments e gli attributi lineNumber, requisitionID e requestedDeliveryDate di un elemento ItemOut in quanto non aggiungono informazioni significative a PunchOutSetupRequest.

Un elenco ItemOut descrive un carrello della spesa esistente, cioè gli articoli di una sessione PunchOut precedente. L'operazione inspect avvia una sessione PunchOut di sola lettura, stabilita sia dal client sia dal server, per visualizzare i dettagli degli articoli elencati. Anche l'operazione edit inizia dal carrello della spesa esistente, descritto usando l'elenco ItemOut, ma consente di apportare modifiche. Il supporto per l'operazione edit implica il supporto per l'operazione inspect (vedere "PunchOutOrderMessageHeader" a pagina 85 e "Carrelli della spesa vuoti" a pagina 85).

BuyerCookie

Questo elemento trasmette le informazioni che non sono trasparenti al sito web remoto ma che devono essere restituite all'autore per eseguire tutte le successive operazioni PunchOut. Questo elemento consente all'applicazione di acquisizione di confrontare più richieste PunchOut in sospeso.

BrowserFormPost

Questo elemento rappresenta la destinazione dei dati in PunchOutOrderMessage. Contiene un elemento URL il cui impiego è spiegato in modo più dettagliato nella definizione di PunchOutOrderMessage. Se non è usato il metodo URL-Encoded, non è necessario includere questo elemento.

Extrinsic

Questo elemento facoltativo contiene ulteriori dati che il richiedente desidera passare al sito web esterno. Questo esempio mostra come passare i dati relativi al reparto dell'utente che ha avviato l'operazione PunchOut. La specifica cXML non definisce il contenuto degli elementi Extrinsic che va invece concordato e implementato da ogni autore e da ogni sito web remoto.

Gli elementi Extrinsic consentono di fornire ulteriori informazioni MRI (Machine-Readable Information). Ampliano il protocollo cXML in modo da supportare quelle funzioni che non sono richieste da tutte le implementazioni. In questo contesto, i dati nuovi descrivono più in dettaglio l'utente che ha avviato la richiesta PunchOut.

L'elemento Extrinsic può essere riportato anche negli elementi OrderRequestHeader, ItemDetail e ContractItem. Questi contesti sono descritti più avanti in questo documento.

SelectedItem

Questo elemento facoltativo indica gli articoli per i quali l'utente intende eseguire il PunchOut per l'acquisto. Contiene un solo elemento ItemID obbligatorio, che consente all'utente di passare dal catalogo locale al sito web del fornitore.

Di solito questo elemento è presente nelle operazioni create. Le applicazioni di acquisizione che consentono agli utenti di eseguire il PunchOut direttamente dall'elenco di un fornitore dovrebbero ignorare l'elemento SelectedItem.

L'elemento SelectedItem dovrebbe essere presente nel caso di operazioni edit e inspect solo se l'utente ha scelto di tornare al sito web del fornitore mentre visualizzava nuove informazioni nel catalogo locale invece che gli articoli in una richiesta di acquisto esistente. In entrambi i casi, il carrello della spesa corrente deve essere presente nell'elenco ItemOut.

I fornitori possono creare propri cataloghi in modo che l'elemento SelectedItem consenta di eseguire il PunchOut a livello di negozio, settore o prodotto. Se l'articolo è specificato dettagliatamente nel catalogo, la ricerca compiuta dall'utente nel sito web del fornitore risulta più semplice.

SupplierSetup

Questo elemento facoltativo specifica l'indirizzo Internet al quale inviare il documento PunchOutSetupRequest. Non risulta necessario quando l'hub di rete per l'e-commerce conosce l'indirizzo Internet PunchOut del fornitore.

PunchOutSetupResponse

Dopo che il sito web remoto ha ricevuto un documento PunchOutSetupRequest, risponde con un documento PunchOutSetupResponse, come mostrato nell'esempio riportato di seguito:

```
<PunchOutSetupResponse>
  <StartPage>
    <URL>
      http://premier.workchairs.com/store?23423SDFSDF23
    </URL>
  </StartPage>
</PunchOutSetupResponse>
```

StartPage

Questo elemento contiene un elemento URL che specifica l'indirizzo Internet da passare al browser per avviare la sessione di esplorazione PunchOut richiesta in PunchOutSetupRequest. Questo indirizzo deve contenere le informazioni sullo stato necessarie per collegarsi a una sessione sul sito web remoto quali, ad esempio, l'identità del richiedente e l'elemento BuyerCookie.

A questo punto, l'utente che ha avviato la richiesta PunchOutSetupRequest esplora il sito web esterno e seleziona gli articoli da trasferire all'applicazione di acquisizione utilizzando un messaggio PunchOutOrderMessage.

PunchOutOrderMessage

Questo elemento invia il contenuto del cestino della spesa remoto all'autore di un messaggio PunchOutSetupMessage. Può contenere una quantità maggiore di dati rispetto agli altri messaggi poiché deve essere in grado di rappresentare per intero il contenuto di qualsiasi tipo di cestino della spesa nel sito web esterno. Questo messaggio non deve necessariamente attenersi al modello richiesta-risposta.

Il sito web remoto genera un messaggio PunchOutOrderMessage quando l'utente esegue un'estrazione. Questo messaggio comunica il contenuto del cestino della spesa remoto all'applicazione di acquisizione; ad esempio:

```
<PunchOutOrderMessage>
  <BuyerCookie>34234234ADFSDF234234</BuyerCookie>
  <PunchOutOrderMessageHeader operationAllowed="create">
    <Total>
      <Money currency="USD">100.23</Money>
    </Total>
  </PunchOutOrderMessageHeader>
  <ItemIn quantity="1">
    <ItemID>
      <SupplierPartID>1234</SupplierPartID>
      <SupplierPartAuxiliaryID>
        ulteriori informazioni su questo articolo
      </SupplierPartAuxiliaryID>
    </ItemID>
    <ItemDetail>
      <UnitPrice>
        <Money currency="USD">10.23</Money>
      </UnitPrice>
      <Description xml:lang="en">
        Learn ASP in a Week!
      </Description>
      <UnitOfMeasure>EA</UnitOfMeasure>
      <Classification domain="SPSC">12345</Classification>
    </ItemDetail>
  </ItemIn>
</PunchOutOrderMessage>
```

Questi elementi sono trattati nelle sezioni seguenti.

BuyerCookie

Questo elemento è lo stesso che è stato passato nella richiesta PunchOutSetupRequest originale. Deve essere restituito in questo punto per consentire all'applicazione di acquisizione di confrontare il messaggio PunchOutOrderMessage con una richiesta PunchOutSetupRequest precedente.

PunchOutOrderMessageHeader

Questo elemento contiene informazioni su tutto il contenuto del cestino della spesa trasferito. L'unico elemento obbligatorio è Total, che rappresenta il costo totale degli articoli aggiunti alla richiesta di acquisto. È possibile utilizzare anche gli elementi Shipping e Tax, che rappresentano l'importo e la descrizione delle spese di spedizione o delle tasse calcolate sul sito web remoto. Anche l'elemento ShipTo è facoltativo e specifica le informazioni sull'indirizzo Ship-To selezionate dall'utente sul sito web remoto o passate nella richiesta PunchOutSetupRequest originale. Tutti gli importi monetari sono contenuti nell'elemento Money che specifica sempre la valuta nel formato standard.

L'attributo di PunchOutOrderMessageHeader è il seguente:

operationAllowed	Specifica le operazioni consentite per PunchOutSetupRequest ovvero creazione, ispezione o modifica (create, inspect o edit).
-------------------------	--

Questo attributo controlla se l'utente può avviare le transazioni PunchOutSetupRequest successive che contengono i dati di questo PunchOutOrderMessage. Se è stato specificato operationAllowed="create", questi articoli possono essere contenuti solo in un documento OrderRequest successivo. Diversamente, l'applicazione di acquisizione potrà ispezionare e modificare il carrello della spesa in un secondo momento, avviando le successive transazioni PunchOutSetupRequest mediante le operazioni appropriate e gli elementi ItemID corrispondenti all'elenco ItemIn in questo PunchOutOrderMessage. Il supporto per l'operazione edit implica il supporto per l'operazione inspect. L'applicazione di acquisizione può utilizzare sempre gli articoli in un documento OrderRequest successivo.

Carrelli della spesa vuoti

Il messaggio PunchOutOrderMessage può contenere l'elenco degli articoli corrispondenti a un carrello della spesa sul sito web del fornitore. Indica sempre la fine di una sessione PunchOut interattiva. I paragrafi che seguono illustrano in dettaglio alcuni casi in cui nel messaggio PunchOutOrderMessage è assente l'elenco degli articoli. Questi messaggi consentono ai client di riprendere immediatamente l'azione quando l'utente esce dal sito web del fornitore.

Se nella richiesta PunchOutSetupRequest originale è stata specificata l'operazione inspect, l'applicazione di acquisizione deve ignorare l'elenco degli articoli del messaggio PunchOutOrderMessage. In questo caso il sito del fornitore non dovrebbe restituire alcun elemento ItemIn. Se un messaggio PunchOutOrderMessage non contiene alcun elemento ItemIn ed è stata specificata l'operazione create, alla richiesta di acquisto non deve essere aggiunto alcun articolo. Il sito del fornitore o l'utente hanno annullato la sessione PunchOut senza creare un carrello della spesa. Se è stata specificata l'operazione edit e il messaggio PunchOutOrderMessage non contiene alcun elemento ItemIn, è necessario eliminare dalla richiesta di acquisto dell'applicazione di acquisizione gli articoli esistenti della sessione PunchOut.

Il codice di stato "204/No Content" indica la fine di una sessione senza modifiche al carrello della spesa. Anche in questo caso il messaggio PunchOutOrderMessage, sempre necessario per BuyerCookie, non dovrebbe contenere elementi ItemIn. Questa codifica viene gestita esattamente come negli altri casi di carrelli vuoti, descritti sopra, a meno che l'operazione specificata non sia edit. In tal caso, l'utente ha annullato la sessione senza apportare alcuna modifica e quindi non dovrebbe essere effettuata alcuna modifica alla richiesta di acquisto nell'applicazione di acquisizione.

ItemIn

Questo elemento aggiunge un articolo dal cestino della spesa ad una richiesta di acquisto di un'applicazione di acquisizione. Può contenere vari elementi ma solo due sono obbligatori: ItemID e ItemDetail.

Gli attributi di ItemIn sono i seguenti:

quantity	Il numero di articoli selezionati dall'utente sul sito web remoto. Poiché il sito del fornitore può applicare le regole per le unità parziali, il protocollo consente di utilizzare quantità frazionarie. Non deve mai essere negativo.
lineNumber (facoltativo)	La posizione di questo articolo in un ordine. Poiché normalmente le sessioni PunchOut hanno luogo prima dell'ordinazione e il server non può in nessun caso controllare il posizionamento degli articoli in un ordine, questo attributo non è importante in un messaggio PunchOutOrderMessage.

Gli elementi facoltativi sono ShipTo, Shipping e Tax che equivalgono agli elementi specificati precedentemente in PunchOutOrderMessage.

Ad eccezione degli elementi Distribution e Comments e degli attributi requisitionID e requestedDeliveryDate disponibili nell'elemento ItemOut, le strutture di ItemIn e ItemOut corrispondono. Il sistema acquirente originario può eseguire la conversione diretta tra gli elenchi ItemIn e ItemOut quando avvia un'operazione inspect o edit. I fornitori possono eseguire la conversione, cancellando le estensioni dell'elenco disponibili

nell'elemento `ItemOut`, quando eseguono un'operazione edit. Il sistema acquirente originario può eseguire la conversione diretta e aggiungere altre informazioni e commenti sulla spedizione e sulla distribuzione quando avvia una transazione `OrderRequest`. I dati di `ItemDetail`, con l'eventuale eccezione degli elementi `Extrinsic`, contenuti negli elementi `ItemIn` non devono essere rimossi quando si converte l'elemento `ItemIn` nell'elemento `ItemOut`.

ItemID

Questo elemento identifica in modo univoco l'articolo nel sito web remoto. È l'unico elemento che deve essere restituito al sito web remoto per identificare di nuovo l'articolo trasferito.

`ItemID` contiene due elementi: `SupplierPartID` e `SupplierPartAuxiliaryID`. Solo `SupplierPartID` è obbligatorio. Grazie all'elemento `SupplierPartAuxiliaryID`, le informazioni sulla complessa configurazione del metodo di trasporto del sito web remoto e quelle sulla partita di merci potranno riconoscere più precisamente l'articolo quando questo sarà presentato al sito web remoto.

Se nell'elemento `SupplierPartAuxiliaryID` sono contenuti caratteri speciali, ad esempio ulteriori elementi XML non definiti nel protocollo cXML, è necessario eseguirne correttamente il riconoscimento. Poiché le informazioni di `SupplierPartAuxiliaryID` passano nelle applicazioni e tornano al fornitore originario, non è sufficiente utilizzare un sottoinsieme interno che contiene altri elementi XML.

ItemDetail

Questo elemento contiene informazioni descrittive sull'articolo che le applicazioni di acquisizione presentano agli utenti. Il contenuto di un elemento `ItemDetail` può essere complesso ma i requisiti minimi sono semplici: `UnitPrice`, `Description`, `UnitOfMeasure` e `Classification`.

Nel contesto di un elemento `ItemIn`, gli elementi `Extrinsic` contenuti in un `ItemDetail` operano allo stesso modo di quelli presenti in un elemento `Index` (in particolare in un `IndexItemAdd`).

Descrizione

Questo elemento descrive l'articolo in forma di testo. Poiché questo testo potrebbe superare i limiti di lunghezza di una piccola tabella di articoli, o di altre interfacce utente limitate, e potrebbero verificarsi dei troncamenti casuali, l'elemento `Description` contiene un elemento `ShortName` facoltativo. In tutti i campi soggetti a limitazione dovrebbe essere presentato dai client l'elemento `ShortName`, se specificato, invece del testo troncato in `Description`. Se non viene specificato alcun elemento `ShortName`, i client troncheranno il testo contenuto in `Description`.

Ad esempio:

```
<Description xml:lang="en-US">
  <ShortName>Big Computer</ShortName>
  This wonder contains three really big disks, four CD-Rom drives, two Zip drives, an
  ethernet card or two, much more memory than you could ever use, four CPUs on two
  motherboards. We'll throw in two monitors, a keyboard and the cheapest mouse we can
  find lying around.
</Description>
```

viene visualizzato come "Big Computer" laddove lo spazio è limitato, mentre se lo spazio è sufficiente viene visualizzato il testo "Big Computer: This wonder ... lying around", eventualmente suddiviso su due campi.

Modifiche successive allo stato

Dopo il completamento della transazione OrderRequest, i fornitori e i server intermedi potrebbero aver bisogno di comunicare altre informazioni al sistema acquirente. Le transazioni descritte in questa sezione vengono usate a questo scopo. Queste transazioni hanno in comune la semantica e alcuni elementi.

Analogamente alla transazione di risposta a una richiesta OrderRequest (vedere "Risposta a un documento OrderRequest" a pagina 80), nessuna di queste transazioni include uno specifico elemento Response. Il documento restituito contiene invece una risposta pressoché vuota (solo un elemento Status). Di seguito viene riportato il formato di ogni documento restituito:

```
<cXML version="1.1.007" payloadID="9949494@supplier.com"
  timestamp="2000-01-12T18:39:09-08:00" xml:lang="en-US">
  <Response>
    <Status code="200" text="OK"/>
  </Response>
</cXML>
```

Viene restituito il codice "200" solo se l'operazione è stata completata regolarmente.

DocumentReference

L'elemento DocumentReference contiene informazioni sufficienti ad associare la richiesta di aggiornamento a un documento particolare. Riporta un attributo obbligatorio del documento precedente e aggiunge un identificativo facoltativo generato dal fornitore. Ad esempio:


```

<DocumentReference
  payloadID="0c300508b7863dcclb_14999"
  orderID="DO4321" />

```

DocumentReference non contiene elementi ma ha gli attributi seguenti:

payloadID	Un numero univoco rispetto allo spazio e al tempo, usato a scopi di registrazione per identificare i documenti. Questo valore non deve cambiare per poterlo usare in altri tentativi. L'implementazione consigliata è: dataora.ID processo.numero casuale@nomehost Ricavato direttamente dall'elemento cXML del documento OrderRequest.
orderID	L'identificativo per questo ordine. Il numero corrente dell'ordine d'acquisto. Ricavato direttamente dall'elemento OrderRequestHeader del documento OrderRequest.

StatusUpdateRequest

Questa transazione informa un nodo precedente circa le modifiche dello stato di elaborazione di un ordine. Quando un hub intermedio riesce a trasmettere un documento OrderRequest, può informare il mittente originario o un hub precedente dell'esito positivo dell'operazione. Per il compratore potrebbero essere significative anche tutte le fasi di elaborazione e di inserimento nelle code a cui è soggetta la richiesta presso un fornitore o un hub.

Questa richiesta aggiorna lo stato di elaborazione di un singolo documento OrderRequest. Ad esempio:

```

<cXML version="1.1.007" xml:lang="en-US"
  payloadID="0c30050@supplier.com"
  timestamp="2000-01-08T23:00:06-08:00">
  <Header>
    Informazioni su instradamento, identificazione e autenticazione.
  </Header>
  <Request>
    <StatusUpdateRequest>
      <DocumentReference
        payloadID="0c300508b7863dcclb_14999"
        orderID="DO4321" />
      <Status code="200" text="OK" xml:lang="en-US">Forwarded
        to supplier</Status>
    </StatusUpdateRequest>
  </Request>
</cXML>

```

Questa richiesta contiene solo un elemento `DocumentReference` e un elemento `Status`. Entrambi sono obbligatori. L'elemento `Status` può comunicare un errore del metodo di trasporto successivo, rilevato da un hub intermedio. La semantica di questo elemento è identica a quella di un elemento `Status` che potrebbe essere stato restituito nella risposta HTTP iniziale a un documento `OrderRequest`.

Il codice 200/OK è particolarmente importante quando i documenti vengono memorizzati e inoltrati. Questo codice indica che un fornitore ha iniziato l'elaborazione di un documento `OrderRequest` o che un hub ha inoltrato il documento. Dopo aver ricevuto il codice 200/OK, per il destinatario non dovrebbero essere previsti altri documenti `StatusUpdateRequest`.

I fornitori e gli hub che utilizzano la transazione `StatusUpdate` devono restituire il codice 201/Accepted quando un documento `OrderRequest` viene inserito nella coda per successive elaborazioni. Dopo che è stato inviato il codice 200/OK (in una risposta immediata a un documento `OrderRequest` o a una transazione `StatusUpdateRequest` successiva), il server non dovrebbe inviare ulteriori transazioni `StatusUpdate` per quell'ordine. Gli errori successivi di elaborazione potrebbero dar luogo ad eccezioni a questa regola.

Definizioni per i cataloghi

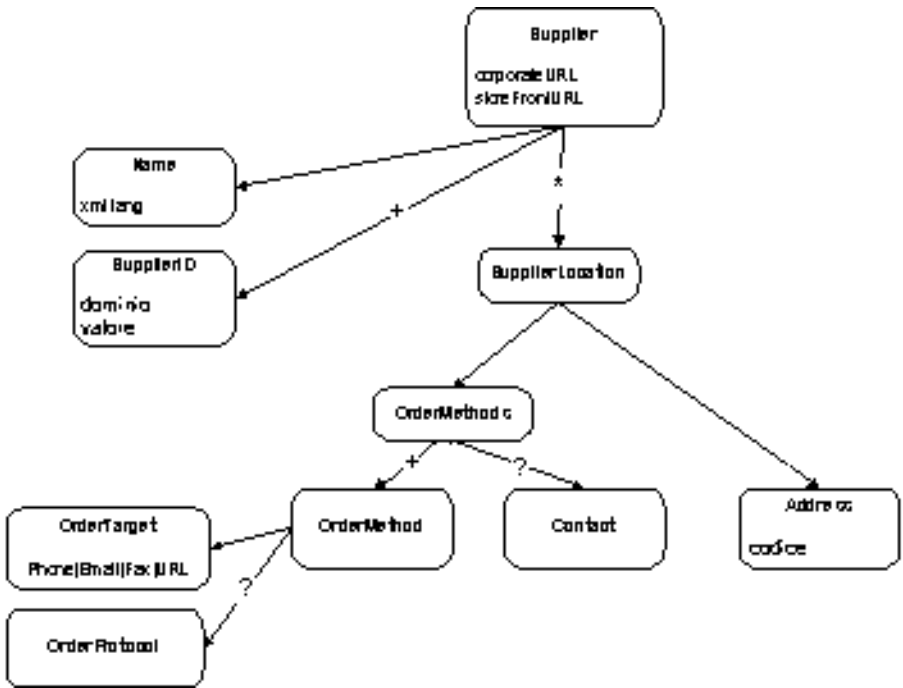
Le definizioni per i cataloghi cXML sono costituite da tre elementi principali: `Supplier`, `Index` e `Contract`. Questi tre elementi consentono di descrivere i dati che devono essere memorizzati in modo permanente o nella memoria cache in un hub o in un sistema di acquisizione di una società acquirente.

- `Supplier`: contiene i dati generali sul fornitore; ad esempio, l'indirizzo, il contatto e le informazioni sull'ordinazione.
- `Index`: descrive i dati sull'inventario dei beni e dei servizi del fornitore; ad esempio, le descrizioni, i numeri parte e i codici di classificazione.
- `Contract`: contiene i dati sugli aspetti flessibili dell'inventario, concordati tra il compratore e il fornitore; ad esempio un prezzo.

Tenere presente che l'elemento `Index` utilizza diversi elementi secondari per descrivere gli articoli negli inventari dei fornitori. I fornitori possono sia inviare le informazioni sui prezzi nelle memorie cache dei sistemi dei compratori che eseguire il `PunchOut` delle informazioni per permettere ai compratori di conoscere i prezzi personalizzati e altre informazioni visitando i siti Web remoti.

Supplier

L'elemento Supplier racchiude un fornitore di beni e servizi. Gli elementi obbligatori sono Name e SupplierID. Inoltre, può descrivere altre informazioni, quali l'indirizzo e le informazioni sull'ordinazione.



Gli attributi di Supplier sono i seguenti:

corporateURL (facoltativo)	Indirizzo Internet del sito web del fornitore.
storeFrontURL (facoltativo)	Indirizzo Internet del sito web per la consultazione o gli acquisti.

Nell'esempio riportato di seguito viene mostrata la sintassi dell'elemento Supplier:

```
<Supplier>
  <SupplierID domain="InternalSupplierID">29</SupplierID>
  <SupplierID domain="DUNS">76554545</SupplierID>
  <SupplierLocation>
    <Address>
```

```

        <Name xml:lang="en-US">Main Office</Name>
        <PostalAddress>
            ...
        </PostalAddress>
        <Email>bobw@workchairs.com</Email>
        <Phone name="Office">
            ...
        </Phone>
        <Fax name="Order">
            ...
        </Fax>
        <URL>http://www.workchairs.com/Support.htm</URL>
    </Address>
    <OrderMethods>
        <OrderMethod>
            <OrderTarget>
                <URL>http://www.workchairs.com/cxmllorders</URL>
            </OrderTarget>
        </OrderMethod>
        <Contact>
            <Name xml:lang="en-US">Mr. Smart E. Pants</Name>
            <Email>sepants@workchairs.com</Email>
            <Phone name="Office">
                ...
            </Phone>
        </Contact>
    </OrderMethods>
</SupplierLocation>
</Supplier>

```

SupplierLocation

Alcuni fornitori svolgono le proprie attività commerciali da più di una sede. Per specificare ciascuna sede, è possibile utilizzare l'elemento *SupplierLocation*. Questo elemento racchiude anche gli elementi che specificano il modo in cui la sede svolge l'attività commerciale e le modalità di accettazione degli ordini. L'elemento *SupplierLocation* contiene un elemento *Address* e un gruppo di elementi *OrderMethods*.

OrderMethods e OrderMethod

L'elemento *OrderMethods* consente di specificare uno o più elementi *OrderMethod* per uno specifico elemento *SupplierLocation*. La posizione dell'elemento *OrderMethods* nell'elenco è significativa: il primo elemento corrisponde al metodo di ordinazione preferito, il secondo corrisponde al metodo con priorità successiva, e così via.

L'elemento `OrderMethod` racchiude le informazioni sull'ordinazione specificando la destinazione dell'ordine (ad esempio, telefono, fax o indirizzo Internet) e, facoltativamente, un protocollo che specifica ulteriormente il metodo di ordinazione per la destinazione, ad esempio "cxml" per una destinazione di tipo indirizzo Internet.

Index

Questo è l'elemento principale per l'aggiornamento dei cataloghi nei sistemi di acquisizione delle società acquirenti.

Un elemento `Index` è associato ad un singolo fornitore. L'elemento `Index` permette di specificare un elenco di ID del fornitore, ciascuno dei quali è considerato un sinonimo di tale fornitore.

L'elemento `Index` contiene uno o più elementi `IndexItem` e, facoltativamente, un gruppo di elementi `SearchGroup` per la definizione dei dati per la ricerca parametrica degli articoli. L'elemento `IndexItem` contiene elementi che consentono di aggiungere o eliminare articoli dal catalogo nella memoria cache della società acquirente. Nell'esempio riportato di seguito viene mostrata la sintassi dell'elemento `Index`:

```
<Index>
  <SupplierID> ... </SupplierID>
  ...
  <IndexItem>
    <IndexItemAdd>
      <IndexItemDetail>
        ...
      </IndexItemDetail>
    </IndexItemAdd>
    ...
    <IndexItemDelete>
      ...
    </IndexItemDelete>
    ...
    <IndexItemPunchout>
      ...
    </IndexItemPunchout>
  </IndexItem>
</Index>
```

IndexItem, IndexItemAdd, IndexItemDelete e IndexItemPunchout

L'elemento `IndexItem` racchiude un elenco di articoli in un indice. Può contenere tre tipi di elementi:

- `IndexItemAdd`: consente di inserire un nuovo articolo nell'indice o di aggiornarne uno esistente. Contiene gli elementi `ItemID`, `ItemDetail` e `IndexItemDetail`.

- **IndexItemDelete**: consente di eliminare un articolo dall'indice. Contiene un elemento **ItemID** che identifica l'articolo.
- **IndexItemPunchout**: consente di inserire un articolo per cui eseguire il PunchOut sul sito web del fornitore. Contiene gli elementi **PunchoutDetail** e **ItemID**. È simile all'elemento **IndexItemAdd**, ad eccezione del fatto che non richiede le informazioni sui prezzi. I compratori possono consultare i dettagli sugli articoli in tempo reale sul sito web del fornitore.

ItemID

L'elemento **ItemID** identifica in modo univoco gli articoli di un fornitore. Contiene l'elemento **SupplierPartID** e, facoltativamente, l'elemento **SupplierPartAuxiliaryID**.

SupplierPartAuxiliaryID

Se l'elemento **SupplierPartID** non identifica in modo univoco l'articolo, il fornitore deve utilizzare l'elemento **SupplierPartAuxiliaryID**. Questo elemento consente di specificare una chiave "ausiliaria" che identifica la parte univoca quando utilizzata insieme agli elementi **SupplierID** e **SupplierPartID**. Ad esempio, un fornitore potrebbe utilizzare lo stesso elemento **SupplierPartID** per un articolo che prevede prezzi diversi a seconda del tipo di unità di misura, ad esempio "EA" (cadauno) e "BOX" (scatola). In questo caso, è consigliabile aggiungere l'elemento **SupplierPartAuxiliaryID** per i due articoli con valore "EA" e "BOX".

È possibile utilizzare l'elemento **SupplierPartAuxiliaryID** anche come cookie del fornitore, per permettere ai fornitori di fare riferimento a dati o configurazioni complessi. Questo elemento può contenere tutti i dati che consentono ai fornitori di ricostruire il percorso di riferimento per i loro articoli presenti nei loro computer (un contenitore o un cookie di dati che hanno senso solo per il fornitore). Per ulteriori informazioni, vedere "Cookie del compratore e del fornitore" a pagina 40.

ItemDetail

L'elemento **ItemDetail** contiene le informazioni dettagliate su un articolo oppure tutti i dati che un utente potrebbe voler vedere su un articolo oltre al dato fondamentale rappresentato nell'elemento **ItemID**. Deve contenere un elemento **UnitPrice**, un elemento **UnitOfMeasure**, uno o più elementi **Description** e un elemento **Classification**. Facoltativamente, può contenere un elemento **ManufacturerPartID**, un elemento **ManufacturerName**, un elemento **URL** e un numero qualsiasi di elementi **Extrinsic**. Per ulteriori informazioni, vedere "ItemDetail" a pagina 87.

Nell'ambito di un elemento **IndexItemAdd**, l'elemento **Extrinsic** consente di estendere le informazioni su un particolare articolo. Queste estensioni non devono essere trasmesse ad un fornitore tramite una transazione **OrderRequest**, in quanto il fornitore può recuperare gli stessi dati utilizzando l'elemento **ItemID** univoco.

IndexItemDetail

L'elemento IndexItemDetail contiene elementi specifici per l'indice che definiscono ulteriori aspetti di un articolo, quali gli elementi LeadTime, ExpirationDate, EffectiveDate, SearchGroupData o TerritoryAvailable.

PunchoutDetail

L'elemento PunchoutDetail è simile all'elemento ItemDetail, ad eccezione del fatto che richiede solo uno o più elementi Description e un elemento Classification. Può contenere anche gli elementi URL, ManufacturerName, ManufacturerPartID, ExpirationDate, EffectiveDate, SearchGroupData, TerritoryAvailable e Extrinsic. Non contiene informazioni sulla determinazione di prezzi personalizzata, sul tempo di approvvigionamento e sulle unità di misura.

Contract

Un elemento Contract rappresenta un listino prezzi stabilito tra un fornitore e un compratore per i beni o i servizi descritti nell'indice del fornitore. Questo elemento consente al fornitore di "sostituire" gli attributi degli articoli (ad esempio, i prezzi) riportati nell'indice con nuovi valori concordati con il compratore. Inoltre, consente ai fornitori e ai compratori di creare segmenti per questi nuovi valori sulla base di una "chiave segmento" concordata e significativa per la società acquirente, ad esempio il nome di un impianto o di un centro di costo.

Gli attributi di Contract sono i seguenti:

effectiveDate	Data e ora di validità del contratto, in formato ISO 8601.
expirationDate	Data e ora di scadenza del contratto, in formato ISO 8601.

L'elemento Contract contiene uno o più elementi ItemSegment, ad esempio:

```
<Contract effectiveDate="2000-01-03T18:39:09-08:00"
  expirationDate="2000-07-03T18:39:09-08:00">
  <SupplierID domain="InternalSupplierID">29</SupplierID>
  <ItemSegment segmentKey=Plant12>
    <ContractItem>
      <ItemID>
        <SupplierPartID>pn12345</SupplierPartID>
      </ItemID>
      <UnitPrice>
        <Money currency=USD>40.00</Money>
      </UnitPrice>
    </ContractItem>
    ...
  </ItemSegment>
</Contract>
```

ItemSegment

L'elemento ItemSegment racchiude un elenco di elementi ContractItem di un determinato "segmento", dove un segmento rappresenta una divisione arbitraria degli articoli di un listino prezzi concordato tra un fornitore e un compratore.

L'attributo di ItemSegment è il seguente:

segmentKey (facoltativo)	Stringa concordata utilizzata per creare segmenti di prezzo personalizzati.
------------------------------------	---

ContractItem

Un elemento ContractItem contraddistingue un particolare articolo che sostituisce un articolo dell'indice. Contiene un elemento ItemID che identifica in modo univoco l'articolo dell'indice che deve essere sostituito nel sistema di acquisizione. Può contenere un numero qualsiasi di elementi Extrinsic che specificano il valore di sostituzione per l'attributo dell'articolo dell'indice interessato.

Definizioni per la gestione degli abbonamenti

Gli intermediari, quali gli hub di rete per l'e-commerce, possono gestire i fornitori e i cataloghi dei fornitori utilizzati dai sistemi di acquisizioni delle società acquirenti. Questi intermediari possono fornire collegamenti diretti tra i sistemi di acquisizione e i sistemi dei fornitori. In questa sezione vengono descritte le definizioni degli elementi per la gestione dei dati del fornitore e dei contenuti dei cataloghi. Queste definizioni si basano su molte delle definizioni trattate in precedenza relativamente al modulo cXML di tipo richiesta-risposta, ai messaggi unilaterali e ai cataloghi.

Dati del fornitore

Le definizioni per la gestione dei dati del fornitore sono costituite principalmente dagli elementi SupplierListRequest, SupplierListResponse, SupplierDataRequest, SupplierDataResponse e SupplierChangeMessage. Negli esempi riportati di seguito, dove l'intermediario è Ariba Network, viene mostrata la sintassi di questi elementi.

SupplierListRequest

SupplierListRequest elenca i fornitori con cui il compratore ha stabilito rapporti commerciali.

```
<Request>
  <SupplierListRequest/>
</Request>
```

SupplierListResponse

L'elemento SupplierListResponse elenca i fornitori con cui il compratore ha stabilito rapporti commerciali.

```
<Response>
  <Status code="200" text="OK"/>
  <SupplierListResponse>
    <Supplier corporateURL=http://www.workchairs.com
      storeFrontURL="http://www.workchairs.com">
      <Name xml:lang="en-US">Workchairs, Inc.</Name>
      <Comments xml:lang="en-US">this is a cool company</Comments>
      <SupplierID domain="DUNS">123456</SupplierID>
    </Supplier>
    <Supplier corporateURL=http://www.computersRus.com
      storeFrontURL="http://www.computersRus.com">
      <Name xml:lang="en-US">Computers R us</Name>
      <Comments xml:lang="en-US">another cool company</Comments>
      <SupplierID domain="DUNS">123456789</SupplierID>
    </Supplier>
  </SupplierListResponse>
</Response>
```

SupplierDataRequest

L'elemento SupplierDataRequest richiede i dati sul fornitore.

```
<Request>
  <SupplierDataRequest>
    <SupplierID domain="DUNS">123456789</SupplierID>
  </SupplierDataRequest>
</Request>
```

SupplierDataResponse

L'elemento SupplierDataResponse contiene i dati sul fornitore.

```
<Response>
  <Status code="200" text="OK"/>
  <SupplierDataResponse>
    <Supplier corporateURL=http://www.workchairs.com
      storeFrontURL="http://www.workchairs.com">
      <Name xml:lang="en-US">Workchairs, Inc.</Name>
      <Comments xml:lang="en-US">this is a cool company</Comments>
      <SupplierID domain="DUNS">123456</SupplierID>
      <SupplierLocation>
        <Address>
          <Name xml:lang="en-US">Main Office</Name>
          <PostalAddress>
            <DeliverTo>Bob A. Worker</DeliverTo>
            <Street>123 Front Street</Street>
            <City>Toosunny</City>
            <State>CA</State>
            <PostalCode>95000</PostalCode>
            <Country isoCountryCode="US">USA</Country>
          </PostalAddress>
          <Email>bobw@workchairs.com</Email>
          <Phone name="Office">
            <TelephoneNumber>
              <CountryCode
                isoCountryCode="US">1</CountryCode>
              <AreaOrCityCode>800</AreaOrCityCode>
              <Number>5551212</Number>
            </TelephoneNumber>
          </Phone>
          <Fax name="Order">
            <TelephoneNumber>
              <CountryCode
                isoCountryCode="US">1</CountryCode>
              <AreaOrCityCode>408</AreaOrCityCode>
              <Number>5551234</Number>
            </TelephoneNumber>
          </Fax>
          <URL>http://www.workchairs.com/Support.htm</URL>
        </Address>
        <OrderMethods>
          <OrderMethod>
            <OrderTarget>
              <URL>http://www.workchairs.com/cxmlorder</URL>
            </OrderTarget>
            <OrderProtocol>cXML</OrderProtocol>
          </OrderMethod>
        </OrderMethods>
      </SupplierLocation>
    </SupplierDataResponse>
  </Response>
```

```

        </SupplierLocation>
    </Supplier>
</SupplierDataResponse>
</Response>

```

SupplierChangeMessage

Questo elemento consente di notificare le modifiche apportate ai dati del fornitore.

```

<Message>
  <SupplierChangeMessage type="new">
    <Supplier corporateURL=http://www.workchairs.com
      storeFrontURL="http://www.workchairs.com">
      <Name xml:lang="en-US">Workchairs, Inc.</Name>
      <Comments xml:lang="en-US">this is a cool company</Comments>
      <SupplierID domain="DUNS">123456</SupplierID>
      <SupplierLocation>
        <Address>
          <Name xml:lang="en-US">Main Office</Name>
          <PostalAddress>
            <DeliverTo>Bob A. Worker</DeliverTo>
            <Street>123 Front Street</Street>
            <City>Toosunny</City>
            <State>CA</State>
            <PostalCode>95000</PostalCode>
            <Country isoCountryCode="US">USA</Country>
          </PostalAddress>
          <Email>bobw@workchairs.com</Email>
          <Phone name="Office">
            <TelephoneNumber>
              <CountryCode
                isoCountryCode="US">1</CountryCode>
              <AreaOrCityCode>800</AreaOrCityCode>
              <Number>5551212</Number>
            </TelephoneNumber>
          </Phone>
          <Fax name="Order">
            <TelephoneNumber>
              <CountryCode
                isoCountryCode="US">1</CountryCode>
              <AreaOrCityCode>408</AreaOrCityCode>
              <Number>5551234</Number>
            </TelephoneNumber>
          </Fax>
          <URL>http://www.workchairs.com/Support.htm</URL>
        </Address>
        <OrderMethods>
          <OrderMethod>
            <OrderTarget>

```

```
        <URL>http://www.workchairs.com/cxmlorder</URL>
      </OrderTarget>
    <OrderProtocol>cXML</OrderProtocol>
  </OrderMethod>
</OrderMethods>
</SupplierLocation>
</Supplier>
</SupplierChangeMessage>
</Message>
```

Abbonamenti ai cataloghi

Di seguito vengono riportate le definizioni per la gestione degli abbonamenti ai cataloghi. L'intermediario utilizzato negli esempi è Ariba Network.

Subscription

Questo elemento cattura i metadati relativi ad un singolo abbonamento al catalogo. Di seguito vengono elencati gli elementi secondari:

- InternalID: ID interno univoco di riferimento per l'intermediario
- Name: nome dell'abbonamento
- ChangeTime: data e ora in cui sono state apportate le ultime modifiche all'abbonamento
- SupplierID: ID del fornitore proprietario del catalogo
- Format: formato del catalogo
- Description: descrizione del catalogo

```
<Subscription>
  <InternalID>1234</InternalID>
  <Name xml:lang="en-US">Q2 Prices</Name>
  <Changetime>1999-03-12T18:39:09-08:00</Changetime>
  <SupplierID domain="DUNS">123456789</SupplierID>
  <Format version="2.1">CIF</Format>
  <Description xml:lang="en-US">The best prices for software</Description>
</Subscription>
```

SubscriptionListRequest

Questo elemento richiede l'elenco corrente degli abbonamenti ai cataloghi del compratore.

```
<Request>
  <SubscriptionListRequest/>
</Request>
```

SubscriptionListResponse

Questo elemento elenca i cataloghi correnti a cui il compratore è abbonato.

```
<Response>
  <Status code="200" text="OK"/>
  <SubscriptionListResponse>
    <Subscription>
      <InternalID>1234</InternalID>
      <Name xml:lang="en-US">Q2 Prices</Name>
      <Changetime>1999-03-12T18:39:09-08:00</Changetime>
      <SupplierID domain="DUNS">123456789</SupplierID>
      <Format version="2.1">CIF</Format>
      <Description xml:lang="en-US">The best prices for software</Description>
    </Subscription>
    <Subscription>
      <InternalID>1235</InternalID>
      <Name xml:lang="en-US">Q2 Software Prices</Name>
      <Changetime>1999-03-12T18:15:00-08:00</Changetime>
      <SupplierID domain="DUNS">555555555</SupplierID>
      <Format version="2.1">CIF</Format>
      <Description xml:lang="en-US">The best prices for software</Description>
    </Subscription>
  </SubscriptionListResponse>
</Response>
```

SubscriptionContentRequest

Questo elemento richiede il contenuto di un catalogo a cui il compratore è abbonato. La richiesta include gli elementi InternalID e SupplierID del catalogo.

```
<Request>
  <SubscriptionContentRequest>
    <InternalID>1234</InternalID>
    <SupplierID domain="DUNS">123456789</SupplierID>
  </SubscriptionContentRequest>
</Request>
```

SubscriptionContentResponse

Questo elemento contiene il contenuto di un catalogo. Il formato del catalogo può essere sia CIF (Catalog Interchange Format) che cXML. Se il formato del catalogo è CIF, viene utilizzato il sistema di codifica base64 e il contenuto viene incluso nell'elemento CIFContent. Se il formato del catalogo è cXML, vengono inclusi gli elementi Index e Contract.

```
<Response>
  <Status code="200" text="OK"/>
  <SubscriptionContentResponse>
    <Subscription>
      <InternalID>1234</InternalID>
      <Name xml:lang="en-US">Q2 Prices</Name>
      <Changetime>1999-03-12T18:39:09-08:00</Changetime>
      <SupplierID domain="DUNS">123456789</SupplierID>
      <Format version="3.0">CIF</Format>
      <Description xml:lang="en-US">The best prices for software
    </Description>
    </Subscription>
    <SubscriptionContent filename="foobar.cif">
      <CIFContent>
        <!-- base64 encoded data -->
        ABCDBBDBDBDBDB
      </CIFContent>
    </SubscriptionContent>
  </SubscriptionContentResponse>
</Response>
```

SubscriptionChangeMessage

Questo elemento segnala al sistema di acquisizione di una società acquirente che un catalogo a cui si è abbonati è stato modificato.

```
<Message>
  <SubscriptionChangeMessage type="new">
    <Subscription>
      <InternalID>1234</InternalID>
      <Name xml:lang="en-US">Q2 Prices</Name>
      <Changetime>1999-03-12T18:39:09-08:00</Changetime>
      <SupplierID domain="DUNS">123456789</SupplierID>
      <Format version="2.1">CIF</Format>
    </Subscription>
  </SubscriptionChangeMessage>
</Message>
```

L'attributo di SubscriptionChangeMessage è il seguente:

type	Il tipo di modifica: new, delete o update.
------	--

Definizioni per il recupero dei messaggi

Alcune società acquirenti non dispongono di punti di ingresso HTTP per la ricezione dei messaggi cXML dall'esterno dei propri firewall aziendali. Per ovviare a questo problema, è possibile utilizzare la specifica cXML.

In questa sezione vengono introdotte le definizioni che permettono ai sistemi di origine di inserire i messaggi in una coda qualora i sistemi di destinazione non siano in grado di accettare direttamente i messaggi HTTP. In questo modo, i sistemi di destinazione possono recuperare i messaggi quando lo ritengono opportuno.

GetPendingRequest

Questo elemento consente di recuperare una serie di messaggi in attesa del richiedente. L'elemento `MessageType` e gli attributi `lastReceivedTimestamp` e `maxMessages` controllano il tipo di messaggio e il numero di messaggi recuperati.

lastReceivedTimestamp (facoltativo)	Timestamp dell'ultimo messaggio ricevuto.
maxMessages (facoltativo)	Numero massimo di messaggi in una singola risposta che il richiedente può gestire.

Una volta ricevuta la richiesta, il destinatario restituisce i messaggi più vecchi, dei tipi specificati e con il valore timestamp uguale o successivo a quello specificato. Se esistono più messaggi corrispondenti ai criteri specificati, può essere restituito il numero di messaggi indicato dall'attributo `maxMessages`. Il sistema di gestione della coda elimina tutti i messaggi in attesa, dei tipi specificati e con il valore timestamp precedente a quello specificato.

```
<Request>
  <GetPendingRequest lastReceivedTimestamp="1999-03-12T18:39:09-08:00"
    maxMessages="5">
    <MessageType>SubscriptionChangedMessage</MessageType>
  </GetPendingRequest>
</Request>
```

GetPendingResponse

Questo elemento contiene uno o più messaggi in attesa del richiedente.

```

<Response>
  <Status code="200" text="OK"/>
  <GetPendingResponse>
    <cXML version="1.1.007" xml:lang="en-US"
      payloadID="456778@ariba.com"
      timestamp="1999-03-12T18:39:09-08:00">
      <Header>
        <From>
          <Credential domain="AribaNetworkUserId">
            <Identity>admin@ariba.com</Identity>
          </Credential>
        </From>
        <To>
          <Credential domain="AribaNetworkUserId">
            <Identity>admin@acme.com</Identity>
          </Credential>
        </To>
        <Sender>
          <Credential domain="AribaNetworkUserId">
            <Identity>admin@ariba.com</Identity>
          </Credential>
          <UserAgent>Ariba.com</UserAgent>
        </Sender>
      </Header>
      <Message>
        <SubscriptionChangeMessage type="new">
          <Subscription>
            <InternalID>1234</InternalID>
            <Name xml:lang="en-US">Q2 Prices</Name>
            <Changetime>1999-03-12T18:39:09-08:00</Changetime>
            <SupplierID domain="DUNS">123456789</SupplierID>
            <Format version="2.1">CIF</Format>
          </Subscription>
        </SubscriptionChangeMessage>
      </Message>
    </cXML>
  </GetPendingResponse>
</Response>

```

Appendice B

Nuove funzioni del cXML 1.1

Il cXML 1.1 contiene le categorie di nuove funzioni riportate di seguito:

- Modifiche generali al cXML
- Modifiche agli elementi Extrinsic
- Miglioramenti delle transazioni PunchOut
- Nuove caratteristiche degli ordini d'acquisto
- Nuova transazione di stato degli ordini d'acquisto

Per una descrizione approfondita degli elementi o degli attributi descritti in questa appendice, vedere Appendice A, "Specifiche del linguaggio cXML"..

Modifiche generali al cXML

Le modifiche descritte di seguito hanno un impatto su vaste aree del linguaggio cXML. Questi miglioramenti facilitano l'uso del cXML a livello internazionale e assicurano la compatibilità tra le versioni del cXML. Inoltre, la nuova transazione Profile consente ai client cXML di eseguire query sulle caratteristiche dei server cXML.

Supporto multilingue avanzato

Per assicurare la congruenza e migliorare il supporto multilingue, gli elementi cXML, Status e ManufacturerName dispongono ora di un attributo `xml:lang` facoltativo.

Esempio:

```
<Status  
  xml:lang="en-US"  
  code="200"  
  text="OK">  
</Status>
```

Per ulteriori
informazioni:

"Status" a pagina 59.

Questo attributo consente di specificare la lingua che i client cXML devono utilizzare nelle risposte e che i siti web PunchOut devono visualizzare agli utenti.

DTD centralizzate

Nelle versioni precedenti, non esisteva alcuna ubicazione centralizzata per le DTD (Document Type Definition) cXML, pertanto gli analizzatori cXML non potevano recuperarle automaticamente. Ora, le DTD di tutte le versioni del cXML sono disponibili in ubicazioni congruenti sul sito web [xml.org](http://xml.cxml.org).

Per le DTD cXML, visitare il seguente indirizzo:

<http://xml.cxml.org/schemas/cXML/<versione>/cXML.dtd>

Per ulteriori informazioni:

"Convalida basata sulle DTD" a pagina 7.

dove *<versione>* è il numero completo della versione del cXML, ad esempio, 1.1.007.

Per prestazioni ottimali, i client cXML non dovrebbero caricare le DTD ogni volta che analizzano i documenti cXML. Dovrebbero, piuttosto, caricarle nella memoria cache locale. Dopo aver inserito i dati in un indirizzo Internet in [//xml.cxml.org/schemas/cXML](http://xml.cxml.org/schemas/cXML), non è necessario cambiare tale posizione. La posizione delle DTD non verrà mai cambiata, ma verranno aggiunti nuovi rami.

Nuova transazione Profile

La nuova transazione Profile comunica le informazioni di base sui server cXML. Questa transazione consiste di due nuovi documenti, ProfileRequest e ProfileResponse e recupera le caratteristiche dei server, incluse la versione cXML supportata, le transazioni supportate e le opzioni di tali transazioni.

Per ulteriori informazioni:

"Transazione Profile" a pagina 69

Nota: Tutti i server cXML 1.1 **devono** supportare la transazione Profile.

I client possono anche utilizzare la transazione Profile per effettuare il "ping" dei server e verificare se sono disponibili.

ProfileRequest

Il documento ProfileRequest è privo di contenuto. Viene semplicemente instradato al server cXML appropriato utilizzando le credenziali specificate nell'elemento Header.

```
<cXML version="1.1.007" payloadID="9949494"
  xml:lang="en-US" timestamp="2000-03-12T18:39:09-08:00">
  <Header>
```

Informazioni su instradamento, identificazione e autenticazione.

```
</Header>
<ProfileRequest />
</cXML>
```

Il server risponde con un singolo documento ProfileResponse, come illustrato di seguito.

ProfileResponse

Il documento ProfileResponse elenca le transazioni supportate dal server cXML, la loro ubicazione e le eventuali opzioni utilizzando un valore in formato di stringa.

```
<ProfileResponse effectiveDate="2000-01-01T05:24:29-08:00">
  <Transaction requestName="OrderRequest">
    <URL>http://workchairs.com/cgi/orders.cgi</URL>
  </Transaction>
  <Transaction requestName="PunchOutSetupRequest">
    <URL>http://workchairs.com/cgi/PunchOut.cgi</URL>
  </Transaction>
</ProfileResponse>
```

Nuovi codici di stato

Per ulteriori informazioni:

"Status" a pagina 59

La specifica cXML 1.1 include nuovi codici di stato per le transazioni che assicurano comunicazioni client-server più accurate. Inoltre, la specifica contiene delle descrizioni migliori per i codici di stato HTTP e cXML esistenti.

Nuovo attributo type per i membri di un mercato

L'elemento Credential dispone del nuovo attributo type che consente di specificare se il mittente o il destinatario sono membri di un mercato. È possibile che si utilizzino più mercati e ogni mercato può avere requisiti diversi per le credenziali.

Per ulteriori informazioni:

"Credential" a pagina 58

Per il nuovo attributo type è possibile specificare un solo valore: marketplace. Utilizzarlo per differenziare le credenziali dei membri di mercato da quelle di un fornitore o di un compratore abituale. L'elemento Credential privo dell'attributo type identifica le aziende non associate a un mercato.

Nelle richieste dirette a o provenienti da un mercato devono essere identificati sia il mercato che l'azienda membro tramite gli elementi To o From di Credential.

Modifiche agli elementi Extrinsic

Il cXML 1.1 introduce ulteriori elementi e attributi per i dati contenuti precedentemente negli elementi Extrinsic. Queste aggiunte spostano nella specifica di base le informazioni precedentemente inviate utilizzando l'elemento Extrinsic.

Il cXML 1.1 introduce anche il supporto dell'elemento Extrinsic a livello di intestazione.

Nuovo elemento Contact

Gli elementi OrderRequestHeader, PunchOutSetupRequest e ItemOut possono ora contenere elementi Contact facoltativi, che riportano una persona o un gruppo da contattare per ottenere ulteriori informazioni.

Per ulteriori
informazioni:

"Contact" a pagina 75

```
<ItemOut quantity="1">
  <ItemID>
    <SupplierPartID>5555</SupplierPartID>
  </ItemID>
  <ItemDetail>
    <UnitPrice>
      <Money currency="USD">134.00</Money>
    </UnitPrice>
    <Description xml:lang="en">
      <ShortName>Office Chair</ShortName>
      Black leather, with adjustable arms, adjustable height and back angle.
    </Description>
    <UnitOfMeasure>EA</UnitOfMeasure>
    <Classification domain="UNSPSC">12345</Classification>
  </ItemDetail>
  <Contact role="customerService">
    <Address>
      <Name xml:lang="en-US">Joe Bob Emmet</Name>
      <Email>joebob@workchairs.com</Email>
      <Phone name="Office">
        <TelephoneNumber>
          <CountryCode isoCountryCode="US">1</CountryCode>
          <AreaOrCityCode>800</AreaOrCityCode>
          <Number>5551212</Number>
        </TelephoneNumber>
      </Phone>
      <Fax name="Order">
        <TelephoneNumber>
          <CountryCode isoCountryCode="US">1</CountryCode>
          <AreaOrCityCode>408</AreaOrCityCode>
          <Number>5551234</Number>
        </TelephoneNumber>
      </Fax>
    </Address>
  </Contact>
</ItemOut>
```

```
        </Fax>
      </Address>
    </Contact>
  </ItemOut>
```

L'attributo `role` specifica la qualifica o il titolo della persona da contattare. I ruoli consentiti sono `endUser`, `administrator`, `purchasingAgent`, `technicalSupport`, `customerService` o `sales`.

Supporto dell'attributo `requisitionID`

L'attributo `requisitionID` è ora pienamente supportato. Si tratta di un attributo facoltativo degli elementi `OrderRequest` e `ItemOut` che identifica la richiesta di un articolo da parte di un compratore.

Per ulteriori informazioni:

"OrderRequestHeader"
a pagina 72

```
<OrderRequest>
  <OrderRequestHeader
    orderID="DO1234"
    orderDate="2000-03-12T13:30:23+8.00"
    type="new"
    requisitionID="R4321">
    <Total>
      <Money currency="USD">12.34</Money>
    </Total>
    <ShipTo>
      ...
    </ShipTo>
  </OrderRequestHeader>
  <ItemOut>
    ...
  </ItemOut>
</OrderRequest>
```

Non includere l'attributo `requisitionID` a livello `ItemOut` se lo si utilizza a livello `OrderRequestHeader`.

Riepilogo delle variazioni apportate agli elementi Extrinsic

La tabella riportata di seguito riporta le modifiche apportate agli elementi Extrinsic più comunemente utilizzati.

Elemento Extrinsic precedente	Elemento o attributo cXML esistente o nuovo
Requested Ship Date	attributo ItemOut requestedDeliveryDate
ship complete	nuovo attributo shipComplete
ReqNumber	attributo esistente requisitionID
Requisition #	attributo esistente requisitionID
Name	nuovo elemento Contact
Phone	nuovo elemento Contact
E-mailAddress	nuovo elemento Contact
Buyer Name	nuovo elemento Contact
Buyer Phone	nuovo elemento Contact
OriginalRequester	nuovo elemento Contact
Requester Phone Number	nuovo elemento Contact
ETA	attributo esistente requestedDeliveryDate

Elementi Extrinsic a livello di intestazione

In precedenza, gli elementi Extrinsic dell'ordine d'acquisto potevano essere specificati soltanto a livello di articolo di una linea di prodotti. Ora possono essere inseriti ovunque all'interno del documento OrderRequest.

Utilizzare questa nuova caratteristica per i dati Extrinsic che interessano l'intero ordine d'acquisto.

All'interno di un documento OrderRequest, gli elementi Extrinsic che hanno lo stesso nome non possono essere specificati sia a livello di intestazione che a livello di voce d'ordine.

Miglioramenti delle transazioni PunchOut

Le transazioni PunchOut sono state migliorate per assicurare un supporto migliore degli acquisti ai livelli di settore e di prodotto. Ora il cXML supporta anche le sessioni PunchOut annullate.

Miglioramenti del documento PunchOutSetupRequest

Il documento PunchOutSetupRequest che i compratori inviano è stato migliorato in modo da aumentare la flessibilità delle transazioni PunchOut. L'indirizzo Internet specificato nel documento PunchOutSetupRequest non è più affidabile, pertanto i server cXML in futuro ignoreranno questo elemento.

Il nuovo metodo utilizza l'ID del fornitore (presente nell'elemento Credential). Gli hub di rete per l'e-commerce ricevono il documento PunchOutSetupRequest, leggono l'ID del fornitore, rilevano l'indirizzo Internet del sito web PunchOut nelle informazioni dell'account del fornitore e inviano il documento PunchOutSetupRequest all'indirizzo Internet appropriato. In questo modo, è l'hub di rete per l'e-commerce, e non il compratore, a specificare l'indirizzo Internet del sito web PunchOut, con una procedura molto più flessibile.

Gli hub di rete per l'e-commerce consentono ai fornitori di memorizzare gli indirizzi Internet dei propri siti web PunchOut.

Elemento SelectedItem

Nell'ambito del miglioramento del documento PunchOutSetupRequest, il cXML ora dispone di un supporto migliore per il PunchOut a livello di negozio, settore o prodotto. Il nuovo elemento SelectedItem facoltativo in questo documento consente ai fornitori di specificare il PunchOut per un intero negozio o per qualunque sottoinsieme di offerte di prodotti. Le applicazioni di acquisizione possono includere l'elemento SelectedItem nei documenti PunchOutSetupRequest, che i siti PunchOut possono utilizzare per determinare quali prodotti visualizzare agli utenti. Se non si utilizza l'elemento SelectedItem, i fornitori dovranno presentare per intero le loro offerte di prodotti (a livello di negozio).

Per ulteriori informazioni:

"SelectedItem" a pagina 82

Un elemento SelectedItem contiene un elemento ItemOut che a sua volta contiene un elemento ItemID. Ad esempio:

```
<SelectedItem>
  <ItemID>
    <SupplierPartID>5555</SupplierPartID>
  </ItemID>
</SelectedItem>
```

Come contenuto dell'elemento `SelectedItem`, le applicazioni di acquisizione utilizzano l'elemento `ItemID` (`SupplierPartID` e `SupplierPartAuxiliaryID`) dell'indice del catalogo PunchOut. Non sono richieste modifiche al catalogo.

Inizialmente le applicazioni di acquisizione devono inviare sia il nuovo elemento `SelectedItem` che il vecchio indirizzo Internet PunchOut nel documento `PunchOutSetupRequest`. Gli hub di rete per l'e-commerce utilizzano il vecchio indirizzo Internet soltanto per i fornitori che non hanno ancora memorizzato le destinazioni degli indirizzi Internet PunchOut.

Documenti PunchOutOrderMessage vuoti

Ora il cXML consente l'uso di documenti `PunchOutOrderMessage` vuoti per consentire agli utenti di terminare le sessioni di acquisto PunchOut senza scegliere alcun articolo. In precedenza, il documento `PunchOutOrderMessage` restituito doveva necessariamente contenere almeno un articolo.

I fornitori possono implementare il pulsante "Cancel" per generare un documento `PunchOutOrderMessage` vuoto. Questo consente sia al sito PunchOut che all'applicazione di acquisizione di riconoscere che un utente ha annullato una sessione di acquisto e conseguentemente di eliminare il carrello della spesa, rimuovere gli articoli dalla richiesta ed eseguire altre attività di gestione.

Nuovo campo nascosto cXML-base64

Il nuovo campo nascosto cXML-base64 supporta i documenti internazionali tramite l'elemento `FORM POST` restituito dai siti PunchOut. I documenti cXML che contengono simboli esterni al gruppo "us-ascii" devono usare questo campo nascosto invece di quello nascosto cXML-urlencoded. Il nuovo campo è equivalente a quello precedente, ma consente la codifica Base64 dell'intero documento durante il trasporto; pertanto, non vengono più utilizzate le codifiche HTML del browser e URL-Encoded del server web.

Per ulteriori informazioni:

"URL-Encoded" a pagina 64

Il sistema Base64 mantiene il sistema originale di codifica dei caratteri di un documento cXML. Sebbene insieme alle informazioni inviate non vi sia alcun parametro "charset", il documento decodificato (dopo la rimozione del codice relativo al trasferimento) può essere considerato di tipo "application/xml". Questo sistema di codifica consente all'analizzatore del destinatario di rispettare qualsiasi attributo "encoding" specificato nella dichiarazione XML. Per questo campo, come per qualsiasi documento "application/xml", il sistema di codifica dei caratteri predefinito è UTF-8.

Nuove caratteristiche degli ordini d'acquisto

I documenti degli ordini di acquisto cXML sono stati migliorati al fine di supportare le caratteristiche richieste.

Nuovo attributo lineNumber

lineNumber è un nuovo attributo facoltativo dell'elemento ItemOut. Specifica la posizione di un articolo all'interno dell'ordine d'acquisto del compratore. Viene scelto quando l'ordine viene piazzato e pertanto non è solitamente attinente alle sessioni PunchOut.

Per ulteriori informazioni:
"ItemOut" a pagina 77

```
<ItemOut quantity="2" lineNumber="1"
  requestedDeliveryDate="2000-03-12">
  <ItemID>
    <SupplierPartID>11223344</SupplierPartID>
  </ItemID>
  <ItemDetail>
    . . .
  </ItemDetail>
</ItemOut>
```

Per potere identificare le modifiche, è necessario che i numeri di riga relativi agli articoli restino invariati negli ordini modificati.

Elementi Attachment dell'ordine d'acquisto

Spesso, i compratori hanno bisogno di rendere più chiari gli ordini d'acquisto utilizzando promemoria, disegni o fax. A tale scopo, ora le applicazioni di acquisizione consentono di allegare file di qualsiasi tipo agli ordini d'acquisto cXML utilizzando le estensioni MIME (Multipurpose Internet Mail Extension).

Il cXML contiene solo i riferimenti alle parti MIME esterne inviate in un unico elemento MIME envelope a più parti (con il documento cXML, in un messaggio di posta elettronica o con un fax).

Il nuovo elemento Attachment contiene i riferimenti agli allegati:

Per ulteriori informazioni:
"Trasmissione degli allegati" a pagina 55

```
<Comments>
  <Attachment><URL>cid: uniqueCID@buyer.com</Attachment>
  Please see attached image for my idea of what this should look like
</Comments>
```

Gli hub di rete per l'e-commerce ricevono gli allegati e li possono inoltrare al fornitore oppure memorizzarli per consentirne il recupero in linea.

Per ulteriori informazioni sullo standard MIME, visitare i siti web agli indirizzi riportati di seguito:

www.hunnysoft.com/mime
www.rad.com/networks/1995/mime/mime.htm

Nuovo attributo shipComplete

shipComplete è un nuovo attributo facoltativo dell'elemento OrderRequestHeader che richiede ai fornitori di evadere un ordine soltanto quando sono disponibili tutti gli articoli. Questo attributo evita le spedizioni parziali di articoli.

Per ulteriori
informazioni:

"OrderRequestHeader"
a pagina 72

```
<OrderRequest>
  <OrderRequestHeader
    orderID="DO1234"
    orderDate="2000-03-12T13:30:23+8.00"
    type="new"
    requisitionID="R4321"
    shipComplete="yes">
  <Total>
    <Money currency="USD">12.34</Money>
  </Total>
  <ShipTo>
    . . .
  </ShipTo>
</OrderRequestHeader>
<ItemOut>
  . . .
</ItemOut>
</OrderRequest>
```

Nuovo elemento ShortName

ShortName è un nuovo elemento facoltativo all'interno degli elementi Description degli articoli.

```
<Description xml:lang="en-US">
  <ShortName>Big Computer</ShortName>
  This wonder contains three really big disks, four CD-ROM drives, two Zip drives, an
  Ethernet card, much more memory than you could ever use, four CPUs on two
  motherboards. We'll throw in two monitors, a keyboard and the cheapest mouse we
  can find lying around.
</Description>
```

Per ulteriori informazioni:

"ItemDetail" a pagina 87

ShortName è un nome abbreviato dell'articolo che può essere composto al massimo da 50 caratteri (sono consigliati 30 caratteri), che viene riportato negli elenchi dei prodotti per gli utenti. In precedenza, le descrizioni lunghe venivano troncate in punti imprecisati in quanto esisteva soltanto un elemento Description.

Le applicazioni di acquisizione e altri client cXML dovrebbero visualizzare l'elemento ShortName al posto del testo riportato nell'elemento Description che potrebbe venire troncato in tutti i campi con limiti di spazio. Se non viene fornito nessun elemento ShortName, i client cXML continueranno a troncare il testo riportato nell'elemento Description.

Gli autori di cataloghi non dovrebbero utilizzare l'elemento ShortName per duplicare le informazioni del testo riportato nell'elemento Description. Al contrario, dovrebbero utilizzare ShortName per classificare il prodotto e Description per fornire una descrizione dei dettagli del prodotto.

Il formato di catalogo CIF 3.0 è stato migliorato al fine di supportare l'elemento ShortName. Il nome del campo CIF è Short Name.

Nuova transazione di stato degli ordini d'acquisto

Il cXML 1.1 introduce una transazione per l'invio dello stato degli ordini d'acquisto agli hub di rete per l'e-commerce.

La transazione si basa sul nuovo elemento OrderReference, che associa l'aggiornamento dello stato al documento OrderRequest del compratore ricevuto più di recente.

Nuovo elemento OrderReference

Il nuovo elemento OrderReference associa un aggiornamento dello stato a un determinato documento OrderRequest. Ripete gli attributi richiesti degli elementi OrderRequestHeader e cXML dal documento OrderRequest e aggiunge un ID facoltativo generato dal fornitore.

Per ulteriori informazioni:

"DocumentReference" a pagina 88

```
<OrderReference
  payloadID="0c300508b7863dcclb_14999"
  timestamp="2000-01-08T14:36:05-07:00"
  orderID="DO4321"
  orderDate="2000-01-08T13:56:23-07:00"
  supplierOrderID="27-33-00-08-01"
/>
```

La nuova transazione StatusUpdateRequest utilizza questo elemento.

Nuova transazione StatusUpdateRequest

Gli enti associati per l'elaborazione degli ordini, quali service provider EDI o fax, inviano la nuova transazione StatusUpdateRequest agli hub di rete per l'e-commerce per impostare lo stato degli ordini d'acquisto. Questa transazione agisce sull'indicatore di stato degli ordini presso gli hub, che è visibile sia ai compratori che ai fornitori. Inoltre, i fornitori possono inviare questa transazione per consentire ai compratori di controllare lo stato dell'elaborazione dei documenti all'interno dell'organizzazione del fornitore.

Per ulteriori informazioni:

"StatusUpdateRequest" a pagina 89

Nota: Questa transazione comunica alle parti interessate le modifiche allo stato delle consegne e dell'elaborazione dei *documenti degli ordini d'acquisto*, ma non indica lo stato di spedizione degli articoli.

Una modifica risulta di particolare interesse: quando un hub intermedio inoltra correttamente un documento OrderRequest, può informare il mittente originario o un hub precedente dell'esito positivo dell'operazione. Per il compratore potrebbero essere significative anche tutte le fasi di elaborazione e di inserimento nelle code a cui è soggetta la richiesta presso un fornitore o un hub.

Nuovo elemento Followup

L'elemento Followup si trova all'interno dell'elemento OrderRequestHeader e specifica l'indirizzo Internet al quale dovranno essere inviati i documenti StatusUpdateRequest. Si tratta dell'ubicazione di input per qualsiasi documento successivo che fa riferimento al documento OrderRequest.

Per ulteriori informazioni:

"Followup" a pagina 76

Indice

Symboli

/ItemDetail 94

A

Accounting, elemento 79
allegati agli ordini d'acquisto 47
allegati MIME 47, 55
Attachment, elemento 76

B

BillTo, elemento 74
BrowserFormPost, elemento 82
BuyerCookie, elemento 81, 84

C

carattere, sistema di codifica 55
Charge, elemento 79
Classification, elemento 22
code, attributo 60
codice per la lingua, nell'intestazione
 cXML 31
codifica tramite elemento Form 65
Comments, elemento 75
compratore, cookie 29, 40
Contact, elemento 75
Contract, elemento 95
convalida del cXML 7
cookie, compratore e fornitore 29, 40
corporateURL, attributo 91
Credential, elemento 58
cXML, elemento 52
cxml.org, sito web 8

cXML-base64, campo nascosto 38, 66
cXML-urlencoded, campo nascosto 38, 65

D

data e ora, formato 54
deploymentMode, attributo 59, 63
Description, elemento 21, 87
Distribution, elemento 79
DocumentReference, elemento 88
domain, attributo 58
DTD (Document Type Definitions) 7

E

EDI (X.12 Electronic Data Interchange) 4
editor per XML 9
effectiveDate, attributo 70, 95
expirationDate, attributo 95
Extrinsic, elemento 27, 40, 76, 82

F

Followup, elemento 76
fornitore, cookie 29, 40
From, elemento 24
From, To e Sender (elementi) 58

G

GetPendingRequest, elemento 103
GetPendingResponse, elemento 104

H

Header, elemento 57
HTML, codifica tramite elemento Form 38,
65

I

id, attributo 79
Index, elemento 93
IndexItemAdd, elemento 93
IndexItemDelete, elemento 94
IndexItemDetail, elemento 95
IndexItemPunchout, elemento 94
indirizzo Internet dell'elemento
SupplierSetup 26
inReplyTo, attributo 63
IsoCountryCode, elemento 68
IsoLanguageCode, elemento 68
ItemDetail, elemento 87
ItemID, elemento 87
ItemIn, elemento 86
ItemOut, elemento 77
ItemSegment, elemento 96

L

lastReceivedTimestamp, attributo 103
lineNumber, attributo 77, 86
lingua, nell'intestazione cXML 31

M

maxMessages, attributo 103
Message, elemento 63

O

operation, attributo 24, 81
operationAllowed, attributo 85
ora e data, formato 54
orderDate, attributo 73
orderID, attributo 73, 89

OrderMethods, elemento 92
OrderRequest, elemento 71
OrderRequestHeader, elemento 72
ordini d'acquisto 43–47
allegati 47

P

pagina di apertura 31
pagina di avvio 35
pagina di invio 35
pagina di ricezione 38
payloadID, attributo 24, 53, 89
Payment, elemento 74
ping dei server, con la transazione Profile 8
preventivi 19
Profile, transazione 8
ProfileRequest, elemento 69
ProfileResponse, elemento 70
PunchoutDetail, elemento 95
PunchOutOrderMessage 28
PunchOutOrderMessage, elemento 84
PunchOutOrderMessageHeader,
elemento 85
PunchOutSetupRequest 23
PunchOutSetupRequest, elemento 80
PunchOutSetupResponse 27
PunchOutSetupResponse, elemento 83

Q

quantity, attributo 77, 86

R

registrazione degli ordini 20
Request, elemento 59
requestedDeliveryDate, attributo 77
requestName, attributo 71
requisitionID, attributo 73, 77
Response, elemento 59
role, attributo 75

S

segmentKey, attributo 96
 SelectedItem, elemento 26, 82
 Sender, elemento 24
 Sender, From e To (elementi) 58
 shipComplete, attributo 73
 Shipping, elemento 74
 ShipTo, elemento 74
 ShortName, elemento 87
 sistema di codifica del carattere 55
 sistema, URL-Encoded 38
 StartPage, elemento 83
 Status, elemento 59
 StatusUpdateRequest, elemento 89
 storeFrontURL, attributo 91
 strumenti per XML 9
 Subscription, elemento 100
 SubscriptionContentRequest, elemento 101
 SubscriptionContentResponse, elemento 102
 SubscriptionListRequest, elemento 101
 SubscriptionListResponse, elemento 101
 Supplier, elemento 91
 SupplierChangeMessage, elemento 99
 SupplierDataRequest, elemento 97
 SupplierDataResponse, elemento 98
 SupplierID, elemento 21
 SupplierListRequest, elemento 97
 SupplierListResponse, elemento 97
 SupplierLocation, elemento 92
 SupplierPartAuxiliaryID, elemento (cookie del fornitore) 30, 40, 94
 SupplierSetup, elemento 83

T

Tax, elemento 74
 timestamp, attributo 24, 53
 To, elemento 24
 To, From e Sender (elementi) 58
 Total, elemento 74
 Transaction, elemento 71
 type, attributo 58

U

unità di misura 30, 68
 URL, elemento 68
 URL-Encoded 65
 utilità per XML 9

V

version, attributo 53

X

XML 15
 xml:lang 31
 xmllanguageCode, elemento 68



www.cxml.org