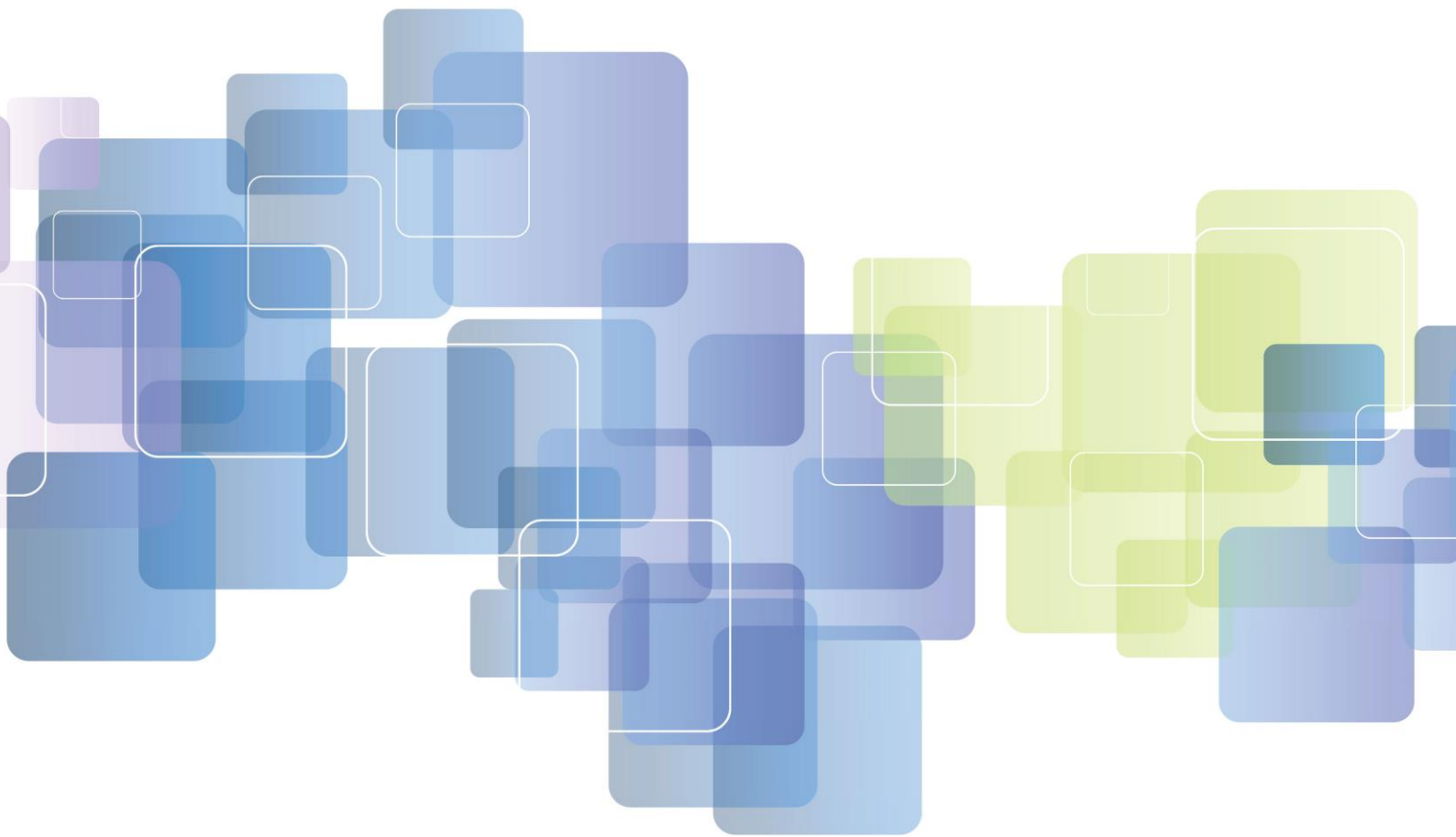


 **TecnetDati**

L'irresistibile ascesa di UML

Emilio C. Porcelli



Un linguaggio "universale" per la modellazione dei sistemi software. Diventerà lo standard di cui si avverte tanto la mancanza?

Forse ne avete già sentito parlare e forse no, ma se le cose andranno com'è nelle aspettative generali finirete inevitabilmente col fare i conti con lui. Il suo nome è UML, che sta per Unified Modeling Language, e giovedì 16 gennaio (era il penultimo giorno utile, ma venerdì 17 viene evidentemente considerato un giorno infausto anche al di là dell'oceano) è stato sottoposto all'approvazione dell'Object Management Group (OMG), il consorzio che conta oltre 500 aderenti in tutto il mondo - tra loro tutti i maggiori vendor - e al quale dobbiamo già significative realizzazioni come CORBA. Nei prossimi mesi OMG dovrà decidere se adottare UML come standard. Ma cos'è esattamente UML? Ecco come ce lo presentano i suoi autori, Grady Booch, Ivar Jacobson e James Rumbaugh, tre personaggi che hanno fatto la storia più recente dell'Analisi e del Disegno Object-Oriented: "E' un metodo per specificare, vi-sualizzare e documentare gli artefatti di un sistema OO in via di sviluppo. UML rappresenta l'unificazione dei metodi Booch-93, OMT-2 e OOSE, così come delle migliori idee di altri metodologi. Attraverso l'unificazione di questi metodi leader, UML costituisce la base per uno standard de facto nel dominio dell'analisi e del disegno OO fondato su un ampio spettro di esperienze utente". Ma da dove nasce l'esigenza di un linguaggio "universale" per la modellazione dei sistemi orientati agli oggetti e, a ben vedere, non solo agli oggetti? Idee come queste non vengono dal nulla e infatti la vicenda di UML ha i tempi e i modi di una rappresentazione teatrale. Ci sono protagonisti e comprimari, si svolge in più atti e l'ultimo atto - come vedremo - dev'essere ancora scritto.

Atto I - Esplosione dei metodi

Cinque anni fa si potevano contare sulle dita di una sola mano, oggi il loro numero è passato a cinquanta e più. L'ultimo quinquennio ha infatti segnato l'esplosione dei metodi - ma da noi ci si ostina a chiamarli "metodologie" - per l'Analisi e il Disegno OO. Va da sè che le differenze tra tutta questa pletora di metodi spesso sono solo di facciata: in molti casi semplici questioni di terminologia, di notazioni e poco più. Saranno anche superficiali, ma restano pur sempre differenze fastidiose e a volte perniciose. Pensiamo per esempio ai tanti modi con cui in un diagramma delle Classi - il cuore del progetto di una qualsiasi applicazione OO - può essere rappresentata la cardinalità "a molti": c'è chi adotta lo strano glifo introdotto anni fa dal solito James Martin, simbolo che Oltre Oceano chiamano familiarmente "crow feet", termine che noi possiamo tranquillamente tradurre "a zampa di gallina"; chi invece usa una freccina con due punte come Shlaer & Mellor, oppure con una punta sola come Bachman; c'è anche chi, come James Rumbaugh prima di ravvedersi, indicava la molteplicità con un bel pallino e l'elenco potrebbe andare avanti ancora per un po'. Sono solo piccolezze - dirà qualcuno. Ma le conseguenze qualche volta si fanno sentire e questa storiellina ce lo può dimostrare. La consociata italiana di una certa grande azienda chimica tedesca si era dotata per i suoi sviluppi software interni di un accreditato strumento CASE di origine

statunitense. Non solo, i modelli prodotti venivano anche usati per illustrare i modi in cui veniva condotto il business qui in Italia e comunicarli al quartier generale laggiù in Germania. I tedeschi - si sa - è gente che alla precisione ci tiene. Immaginate il loro sconcerto nello scoprire, naturalmente solo a cose fatte, che tra le notazioni usate dai loro colleghi italiani per descrivere i modelli del business e quelle adottate in Germania, che seguivano lo standard Isotec, non c'era corrispondenza uno a uno. La difformità delle notazioni può anche voler dire inconsistenza delle semantiche. E infatti in quella grande azienda chimica i modelli non riuscivano a parlare con i modelli. E' solo un esempio del caos che regna nel mondo dei metodi e dei modelli, e di riflesso anche in quello del CASE. L'assenza di uno standard riconosciuto rende rischioso ogni investimento e infatti - anche se questa non è la sola ragione - il mercato del CASE, dopo una vampata iniziale, di questi tempi segna visibilmente il passo.

Atto II - Confluenza dei Metodi

Poi, fortunatamente, la tendenza si è invertita e così all'esplosione dei metodi ha fatto seguito la loro confluenza. Tutto comincia nell'ottobre del 1994, quando James Rumbaugh, una vita spesa in General Electric, raggiunge Grady Booch alla Rational Software, la società californiana di cui Booch è chief scientist. Dal loro incontro, dopo esattamente un anno di gestazione, nasce l'Unified Method Versione 0.8, frutto della confluenza dei metodi Booch-93 e OMT-2 di Rumbaugh. Siamo così arrivati all'ottobre del 1995 e quello stesso mese Objectory AB, la società in cui opera Ivar Jacobson, viene acquisita dalla Rational. Jacobson abbandona la natia Svezia per approdare in California, anche lui alla corte di Booch. Ancora alcuni mesi di lavoro - è il giugno dello scorso anno - e in UML, ora in Versione 0.9, confluisce anche OOSE (sta per Object-Oriented Software Engineering), il metodo di Jacobson. Con l'occasione l'Unified Method cambia anche nome e viene ribattezzato Unified Modeling Language. Scopriremo poi il perchè. Va da sè che l'incontro tra Booch, Jacobson e Rumbaugh non è stato casuale. I loro metodi infatti erano complementari: OOSE, basato com'è sui Casi d'Uso, aveva già dato buona prova di sè per il BPR e l'analisi dei requisiti; OMT-2 risultava invece particolarmente espressivo nella progettazione dei Sistemi Informativi ad alta intensità di dati; infine Booch- 93 era più orientato alle fasi di disegno e di costruzione delle applicazioni, specialmente quelle di contenuto ingegneristico. C'era poi il responso del mercato. Secondo i dati di IDC nel 1995 sulle proposte metodologiche di Booch, Jacobson e Rumbaugh si era concentrato il 55,7% della spesa complessiva per strumenti di supporto all'Analisi e al Disegno OO, valutata in 127,5 milioni di dollari. Quello stesso anno Rational registrava un incremento sensazionale (+ 110%) del suo giro d'affari rispetto all'anno precedente. Senza contare che i nostri tre Amici (o "Amigos" come suona il loro indirizzo di e-mail alla Rational) si erano già conosciuti, si erano annusati e si erano piaciuti. Ma UML è più della semplice convergenza dei metodi di Booch, Jacobson e Rumbaugh. Presentato come "Metodo della Terza Generazione", è in realtà un metodo eclettico che, oltre alle idee dei suoi autori, accoglie quelle di numerosi altri metodologi:

- David Harel per i diagrammi di stato
- Bertrand Meyer per le pre-e postcondizioni
- Sally Shlaer & Stephen Mellor per il Ciclo di Vita degli Oggetti
- Rebecca Wirfs-Brock per responsabilità e collaborazioni

Sono quanto di meglio aveva saputo proporre fino ad allora il mondo tumultuoso dei metodi per l'OO. Ma il cammino di UML non si ferma qui.

Atto III - Convergenza sul Metodo

I primi a convergere su UML sono stati i metodologi. Alla sua successiva elaborazione sfociata nella Versione 1.0, quella sottoposta appunto all'approvazione dell'Analysis & Design Task Force RFP-1 di OMG, ha infatti collaborato una fitta schiera di personaggi eccellenti. Tre nomi su tutti: Peter Coad, James Odell e Ed Yourdon. Poi è venuta la volta dei vendor. I primi farsi avanti sono Microsoft e Hewlett-Packard che il 22 luglio dello scorso anno hanno annunciato di unirsi alla Rational nel proporre l'adozione di UML come standard da parte di OMG. Poi è stata la volta di Oracle, Texas Instruments e Unisys. Attualmente i "co-submitter" di UML sono ben otto, per non parlare poi di una folta schiera di altri "supporter". Con quali motivazioni? Secondo Bob Muglia che alla Microsoft è a capo di un settore sicuramente strategico come quello degli strumenti per lo sviluppo "UML coniuga le pratiche più avanzate dell'Analisi e del Disegno OO con le tecnologie COM e ActiveX che sono già state adottate da centinaia di migliaia di sviluppatori in tutto il mondo". E per Ken Jacobs che in Oracle coordina le politiche per i database server "UML è perfettamente in linea con la nostra strategia che prevede il supporto della modellazione degli oggetti e un disegno delle applicazioni basati su Oracle 8, il database server che sta per uscire sul mercato". E così, grazie a UML, Microsoft e Oracle una volta tanto hanno finito col trovarsi sulla stessa barca. Adesso qualcuno si chiederà: e IBM? C'è anche lei. La sua adesione a UML è avvenuta in sordina; niente clamori, niente dichiarazioni ufficiali. Ma per Big Blue parlano alcuni fatti. Attraverso Rose, lo strumento CASE della Rational, UML si è fatto strada nel progetto "San Francisco", un'iniziativa IBM alla quale partecipano 75 produttori indipendenti di software sparsi in tutto il mondo con l'obiettivo di mettere a punto una serie di framework applicativi multi-piattaforma - dalla contabilità generale alla gestione degli ordini, al magazzino -, i primi che siano mai stati scritti in Java. Senza contare che all'evoluzione di UML ha anche contribuito John Vlissides del Thomas J. Watson Research Center IBM di Hawthorne. Con Eric Gamma, Richard Helm e Ralph Johnson è uno degli autori di "Design Pattern: Elements of Reusable Object-Oriented Software" (Addison-Wesley, 1995), un libro che ha aperto una nuova dimensione alla nozione di riusabilità del software. UML e Processi Software A questo punto è giunto il momento di chiedersi come mai UML, che era stato inizialmente battezzato Unified Method, ha successivamente rinunciato all'appellativo di "metodo" per assumere quello di "linguaggio" e per di più "universale". Ecco la spiegazione che ce ne danno i suoi Autori: "Un solo processo universale buono per tutti gli stili dello sviluppo non sembra possibile e tanto meno desiderabile. Ciò che va bene per un progetto è probabilmente sbagliato per un altro. Tuttavia UML può essere usato per esprimere gli artefatti di tutti questi processi, in particolare i Modelli prodotti". Ma ciò non significa che UML trascuri i processi. In realtà quello che i suoi Autori hanno in mente è un processo: *

- basato sui Casi d'Uso
- incentrato sull'architettura
- iterativo e incrementale.

E aggiungono: "L'esperienza insegna che i dettagli di questo processo di tipo generale vanno adattati alle peculiarità della cultura dello sviluppo o del dominio applicativo di ciascuna organizzazione". Proprio perché, non sposa una causa metodologica precostituita UML si può fregiare dell'appellativo di universale: "Universale nel senso che può esprimere significati di natura e che si prestano a scopi anche assai diversi, esattamente come un linguaggio di programmazione, non naturale, può essere usato in modi molto diversi".

IV Atto: il responso di OMG

L'ultimo atto della vicenda UML resta ancora tutto da scrivere. Infatti a candidarsi come "linguaggio universale" di modellazione non c'è solo UML. Tra tutti coloro che bussano alla sua porta, su chi si appunterà la scelta di OMG? La risposta non tarderà a farsi conoscere. Infatti, a differenza di certi pachidermi come ISO e ANSI, in tutta la sua breve storia OMG ha sempre dimostrato di sapersi muovere agilmente. Quando si conoscerà il suo responso? Probabilmente entro il prossimo mese di giugno. Ma, a dar retta a quanto accade sull'altra sponda dell'oceano, i giochi sembrerebbero già fatti. Il mondo dei vendor di strumenti CASE è entrato in fibrillazione e la parola d'ordine è UML. I vendor tradizionali si affannano a riposizionare la loro offerta per adattarla al nuovo verbo, mentre sul mercato si sta affacciando tutta una nuova generazione di produttori e di prodotti. Sulla Rete c'è già chi offre tool che vantano una completa aderenza a UML e costano solo 500 dollari. Manterranno davvero tutto ciò che promettono? Anche da noi le cose si stanno muovendo, anche qui i vendor di strumenti CASE danno segni di vita. Per tutta una serie di fortunate anche se non fortuite circostanze chi vi scrive si è trovato a essere uno dei non molti depositari in Italia della scienza su UML. E così ho trascorso questi ultimi mesi nelle aule di mezza Italia a parlare di UML e dintorni a un pubblico qualche volta di soli vendor. Posso quindi soddisfare una curiosità che forse è rimasta a chi ha avuto la pazienza di seguirmi fino in fondo. Qual è il simbolo adottato da UML per specificare la cardinalità a molti? Non più pallini o frecce, per non parlare di zampe di gallina. E' * e cioè un normalissimo asterisco. Più semplice di così...

IL COSA e i DOVE di UML

La descrizione di UML Versione 1.0 è contenuta in una serie di cinque documenti:

UML Summary - Una introduzione a UML e ai restanti documenti.

UML Semantics - E' la descrizione formale del metamodello che è alla base di UML. Il metamodello è illustrato usando le notazioni di UML corredate da una concisa descrizione in linguaggio naturale (leggi inglese).

UML Notation Guide - Descrive ed esemplifica la notazione di UML.

UML Process-Specific Extentions - Illustra le estensioni di UML alla descrizione dei processi del business.

UML Glossary - Definisce i termini usati per descrivere UML. Oltre alla terminologia specifica di UML comprende anche termini tratti dagli standard OMG che fanno riferimento a essa.

Tutti questi documenti sono disponibili in Rete al sito della Rational:
<http://www.rational.com>.

E per finire una buona notizia! In Rete la versione italiana di [UML Glossary](#) c'è già.



Tecnet Dati s.r.l.
C.so Svizzera 185 -
10149 - Torino (TO), Italia
Tel.: +39 011 7718090 Fax.: +39 011 7718092
P.I. 05793500017 C.F. 09205650154
www.tecnetdati.com

