



F.O.A.M.

Free Object Access Method

CO.BRA (Common BRAin) Universal Agent

CO.BRA: un nome per due prodotti.

CO.BRA oltre ad indicare la specifica implementazione di FOAM, scritta in C++ su piattaforma Windows NT e 2000 Server, conformemente agli standard POSIX in modo da garantirne la compatibilità anche con Linux ed i principali Unix commerciali, è anche il nome dell' "agent" software: un piccolo modulo che implementa il protocollo FOAM e ne mette a disposizione le funzionalità a programmi esterni.

Nei primi anni '60 del secolo scorso, J.C.R. Licklider, direttore dell'I.P.T.O. (Information Processing Techniques Office) ed uno dei "padri" di Internet, introdusse il concetto secondo il quale i computer potevano essere uno strumento in grado di ampliare le capacità intellettive degli esseri umani estendendone il proprio sistema neuronale.

La sigla CO.BRA è l'acronimo di COMmon BRAin: l'ottica è quella di considerare la Rete come una sorta di cervello comune, un repository di informazioni e servizi cui tutti (utilizzando gli opportuni protocolli ed eventuali privilegi di accesso) possono attingere.

CO.BRA supporta appieno tutte le tecnologie Object Oriented:

- programmazione per componenti;
- polimorfismo;
- derivazione;
- information hiding.

Grazie al polimorfismo, CO.BRA libera il programmatore dalla stesura di codice per la comunicazione in rete, infatti eredita tutte le caratteristiche di FOAM: trasparenza verso la rete, fault tolerance, load balancing.

Tramite CO.BRA si costruisce l'infrastruttura di rete cui poggia l'intera comunicazione aziendale, combinando al modello *client-server* il sempre più utilizzato modello *peer-to-peer*, tramite una linea di prodotti scalabile da un minimo di due fino a migliaia di postazioni collegate tra loro in rete geografica. Ogni "agent" può essere configurato dinamicamente, a run-time, per funzionare da server, client o come partner paritetico. In architettura peer-to-peer ogni agent dialoga con qualsiasi suo omologo, realizzando un vero e proprio sistema nervoso digitale distribuito.

CO.BRA possiede capacità di fault tolerance e load balancing. Queste capacità derivano dalla possibilità di tenere attive più istanze di ciascun oggetto, che possono girare sullo stesso sistema o su diversi sistemi. Quando un oggetto non può fornire un servizio che gli è stato richiesto (ad esempio perché vi sono troppi thread attivi, o poca memoria libera, troppi utenti, ecc.), COBRA attiva il meccanismo di load balancing e dirige le richieste ad un altro oggetto. Allo stesso tempo, COBRA riesce a garantire un sistema completamente fault tolerant. In caso di malfunzionamento hardware, COBRA gira le richieste ad un altro oggetto presente in rete e che offre gli stessi servizi. Qualora in rete non dovesse trovarsi nessun altro Application Server, la natura asincrona del protocollo garantisce comunque la sopravvivenza ed il funzionamento dei client. In questo caso, le richieste pendenti saranno erogate man mano che gli Application Server sono ripristinati. Il tutto, ovviamente, in maniera completamente trasparente sia per il programmatore, sia per l'utente finale.

CO.BRA Universal Agent: dal protocollo al prodotto.

CO.BRA Universal Agent nasce per rispondere ad esigenze tanto semplici quanto importanti:

- dare la possibilità ad applicazioni "legacy" di aprirsi al mondo della elaborazione distribuita;
- fornire ai programmatori non avvezzi alle problematiche della comunicazione, la possibilità di accedere in maniera completamente trasparente.

Uno degli obiettivi dei prodotti "middleware" è, infatti, quello di nascondere la complessità della programmazione distribuita. CO.BRA Universal Agent centra questo obiettivo con una soluzione elegante e che può essere utilizzata da tutti i linguaggi di programmazione che offrono un semplice supporto alle "socket" TCP/IP.

Il compito dell'agent è quello di dialogare con suoi simili, distribuiti in rete, realizzando, di fatto, una comunicazione "peer-to-peer", oppure richiedere servizi ad Anaconda Internet Application Server, realizzando una comunicazione client/server. Uno stesso agent può essere dinamicamente configurato, a "run-time" per svolgere ora l'una, ora l'altra forma di comunicazione.

La **Figura 1** mostra un esempio di comunicazione peer-to-peer.



Figura 1: comunicazione peer-to-peer

Mentre nella **Figura 2** è mostrata la comunicazione con Anaconda Internet Application Server.



Figura 2: comunicazione in configurazione client/server con Anaconda Internet Application Server

CO.BRA Universal Agent è disponibile in due versioni:

- end-user version, destinata agli utenti finali;
- developers kit, destinata agli sviluppatori software.

Dal punto di vista delle funzionalità, le due versioni sono completamente equivalenti, differendo solo dal punto di vista estetico: la versione end-user (**Figura 3**), infatti, non ha una vera e propria interfaccia grafica (in questo modo risulta completamente trasparente per l'utente finale), ma una semplice finestra dalla quale si può opzionalmente modificare il Port Number di "ascolto".

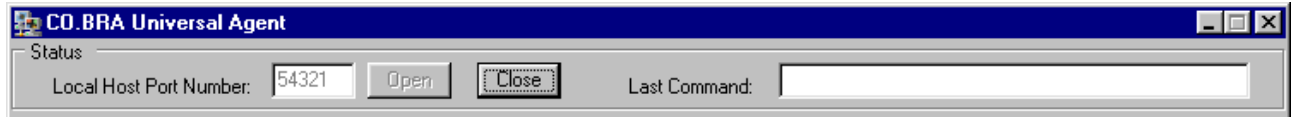


Figura 3: l'interfaccia grafica di CO.BRA Universal Agent end-user version

La developers version, invece, ha un'interfaccia grafica (**Figura 4**) che aiuta visivamente lo sviluppatore software a capire come i dati "viaggiano" tra applicazioni remote.

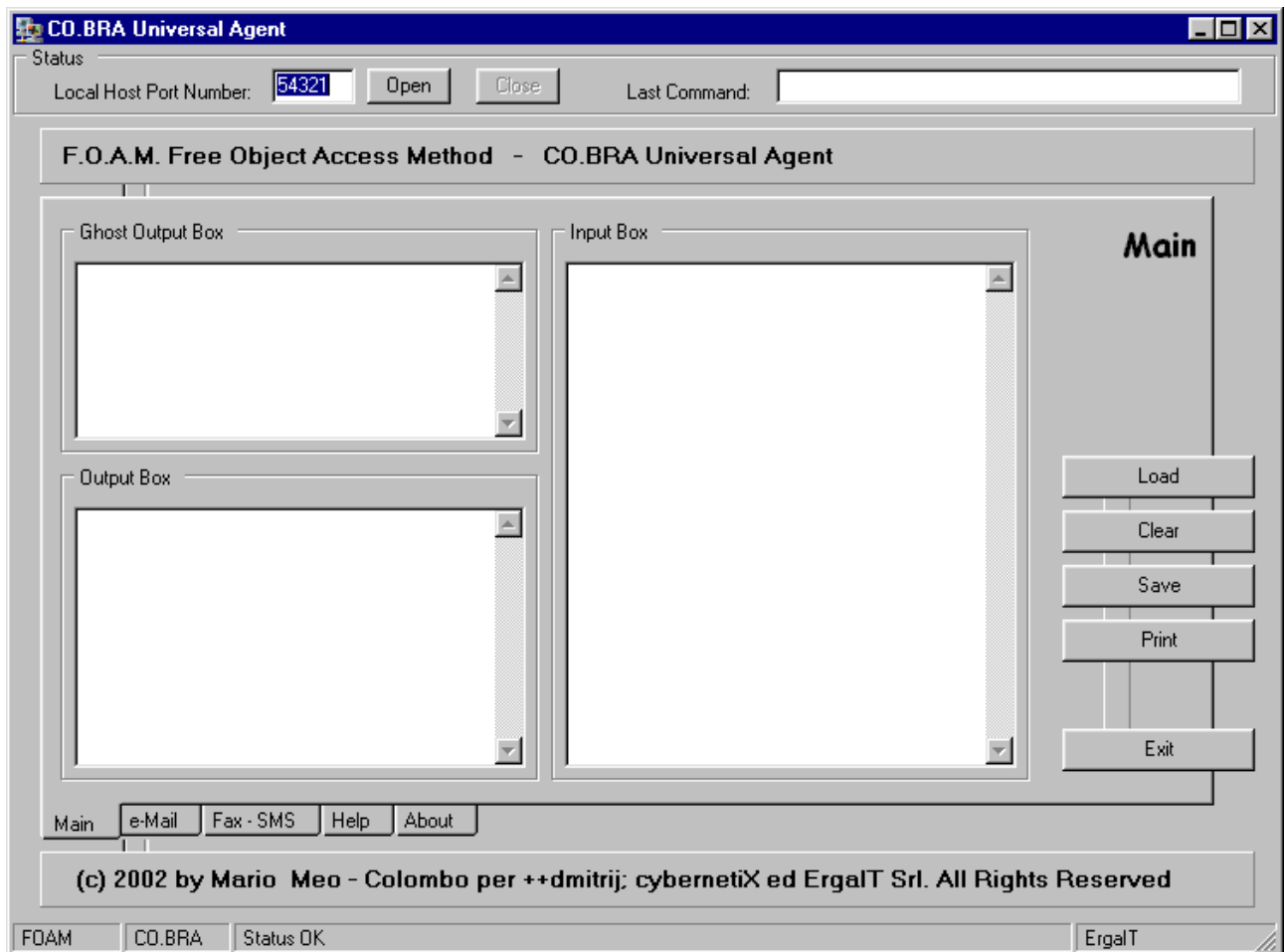


Figura 4: l'interfaccia grafica di CO.BRA Universal Agent developers version