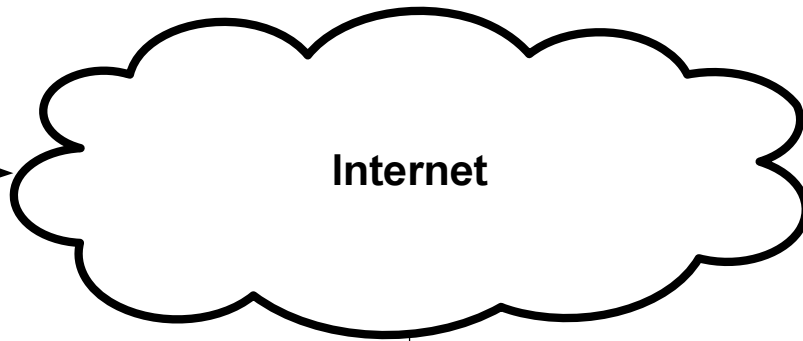
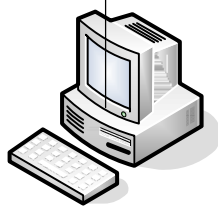
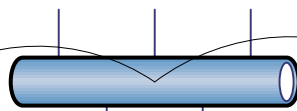


End-User  
End-user accesses customer website via the internet



Webserver  
The webserver contains the website developed by the customer in ASP.net, Coldfusion, Perl, etc.  
If in ASP.net, the project must reference the Mult-e-commerce .dll  
If the environment supports java, the .clas file must be referenced.

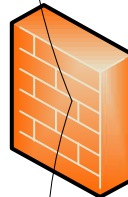


Webserver/e-comm network  
The webserver and the e-comm server should be on the same backbone, i.e. in the same facility.

E-comm server  
The e-comm server contains the Progress database, and the software that is called from either the .dll or the .clas api.  
It also includes replication software that captures all updates that come to the database via the web, and writes a record in replication tables to send to MultiPub.  
It also contains software which updates the e-comm database with all changes that come directly into MultiPub (via manual data entry or processing).  
The e-comm server does not need to have a webserver running; however, you may if you choose to execute the api from the e-comm server instead of the webserver.

VPN  
The webserver, e-comm server and MultiPub server should be on the same VPN

Firewall  
The firewall should be placed between the e-comm server and the MultiPub server. It is designed so that all ports are closed except for calls coming from the e-comm server



MultiPub server  
The MultiPub server contains the MultiPub software and database, and replication software that captures all updates that come to the database via MultiPub, and writes a record in the replication tables to send to Mult-e-commerce.  
It also includes software to apply the Mult-e-commerce changes to MultiPub as well as processes to handle exceptions.

