```
        /// </summary>
        /// <param name="id"></param>
        /// <returns></returns>
        public ProductDTO[] GetAllProducts(int id)
        {
            Product p = new Product();
            ProductDTO[] pa;
            //return all products
            p.GetAllProducts();
            pa = new ProductDTO[p.ProductList.Count];
            return pa;
        }
    }//end class
}//end namespace
```

Note that we don't need to use the `[WebMethod]` identifier here as it was used in ASMX services. Using WCF instead of ASMX services is recommended because WCF is a more complete and easy-to-use platform when compared to simple ASMX web services. Hence, WCF helps us write better services that are more useful in an SOA environment.

Summary

Enabling SOA in your .NET applications is not a difficult task. SOA implementations use object-oriented techniques internally, but provide a simple, platform-independent coarse-grained interface to be used by any client, independent of the technology used. Because XML is an open standard, using SOA guarantees that your system can talk to any other system, be it built in PHP, JAVA or any other platform.

SOA architecture should not be used everywhere without proper forethought. There is no need for SOA architecture when we are sure that the application is small and limited in scope as well as potential growth.

ASMX web services were one of the best ways to implement SOA-based architectures. But, Microsoft's WCF framework works best when one has to integrate all of the service-oriented techniques into one framework, as it is easy to use and involves less complexity. Considering the richness of WCF over ASMX, it is suggested that all services should be implemented in WCF only.

All of these topics are very important to understand, and will help us make the right choices when selecting an architectural platform for our web applications. By leveraging this knowledge and experience, we can develop scalable and robust solutions that are flexible enough to cater for future needs.