WCF(Windows Communicaton Foundation)

Gursharan Singh Virdi

WCF?

- **Microsoft's platform for SOA architecture**.
  - Used for building distributed and interoperable applications.
- **Unifies .Asmx,.Net Remoting and Enterprise Services stacks.**
  - A single programming model for all.
  - Configurations drive protocols,message formats and process allocations.

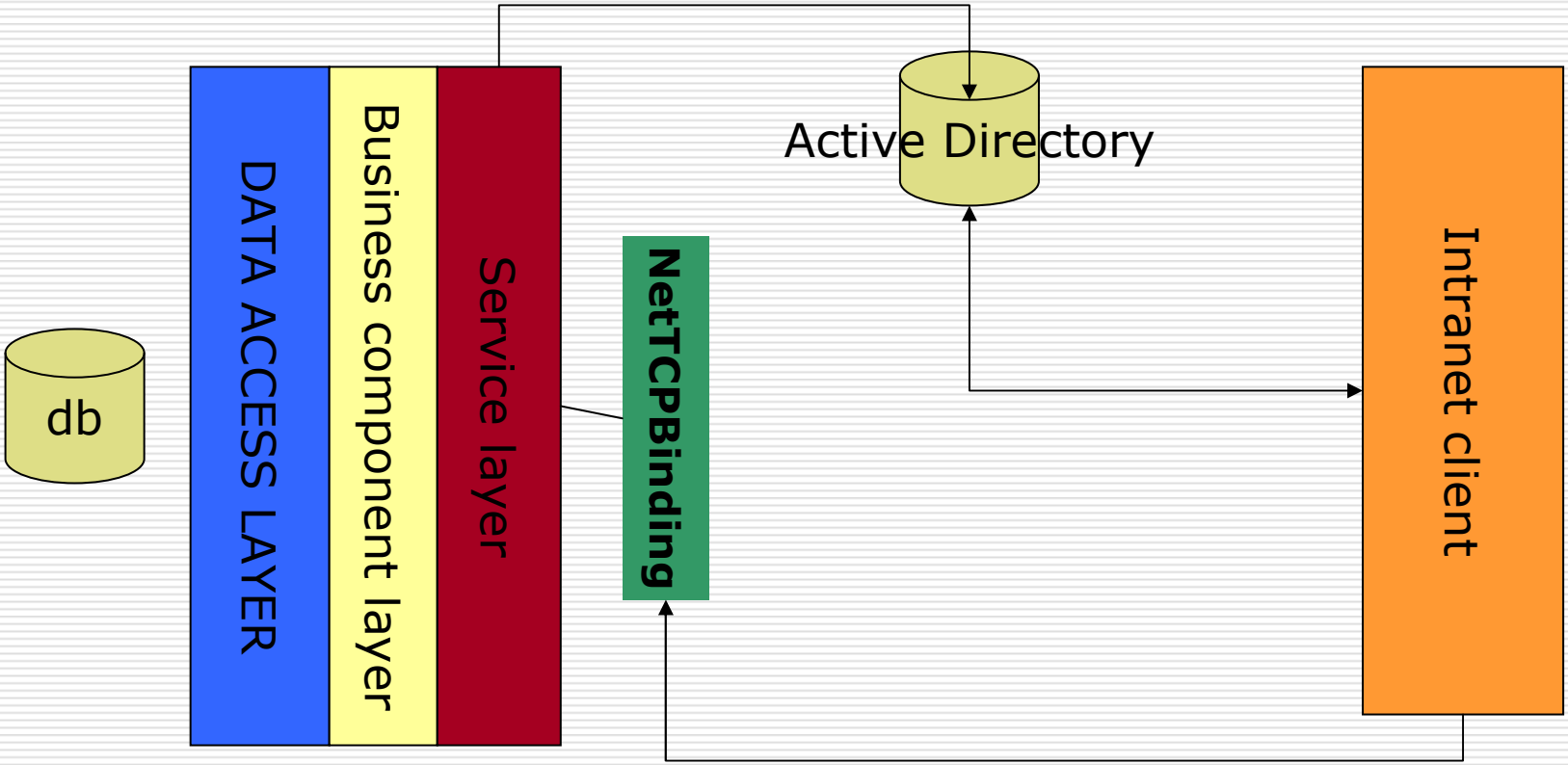
WCF?

- **Service Oriented**
  - Built for service oriented design.
  - Simplifies the SOA approach.
- **Loosely coupled**
  - Not bound to a particular protocol ,encoding format or hosting environment.
  - Almost everything is configurable.

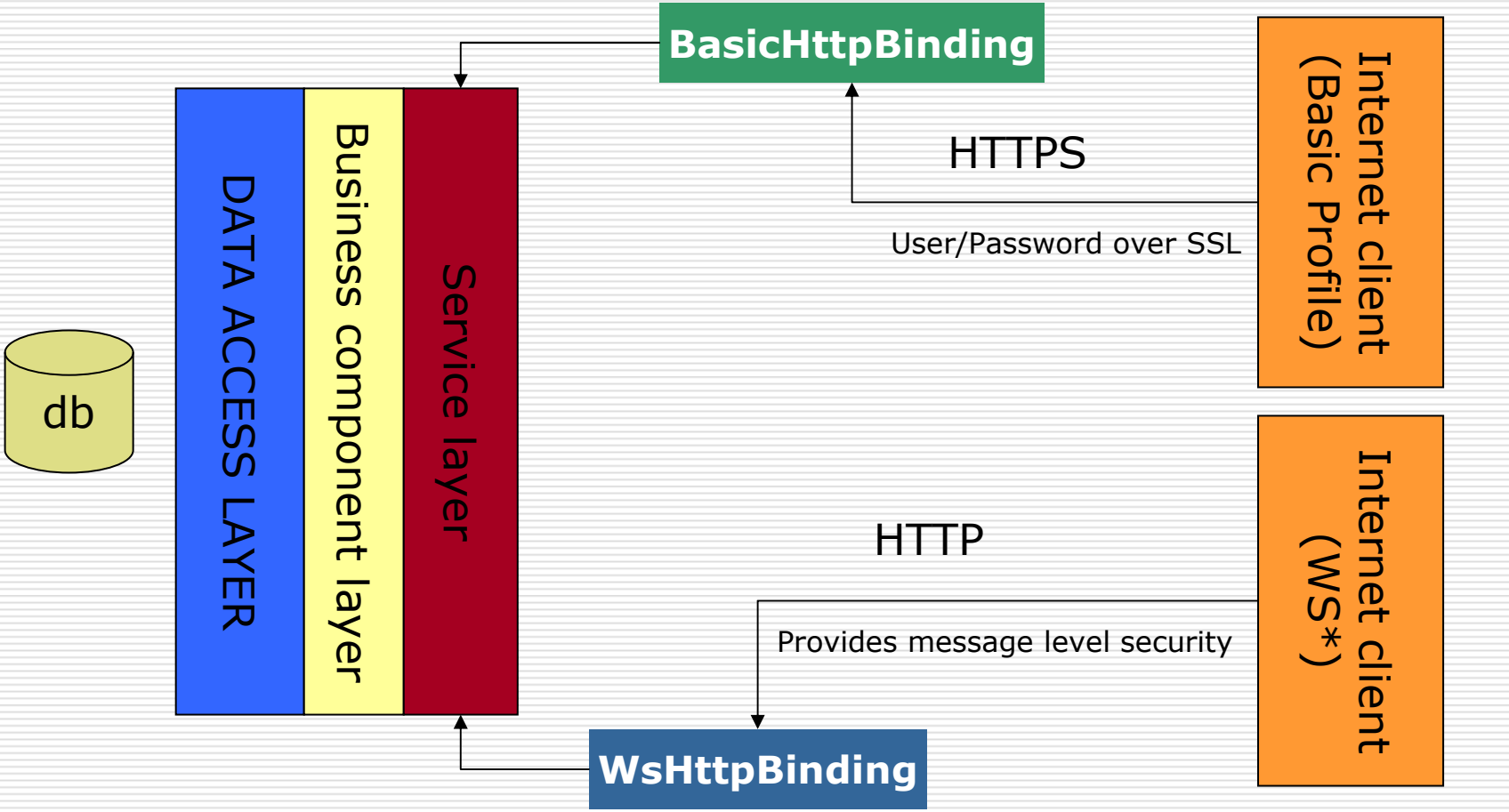
WCF?

- **Interoperable**
  - Supports core webservice standards.
  - Extensible to quickly adapt to new protocols and updates.
- **Integration**
  - Integrates with earlier microsoft distributed technologies/environments like COM,Enterprise services,MSMQ.

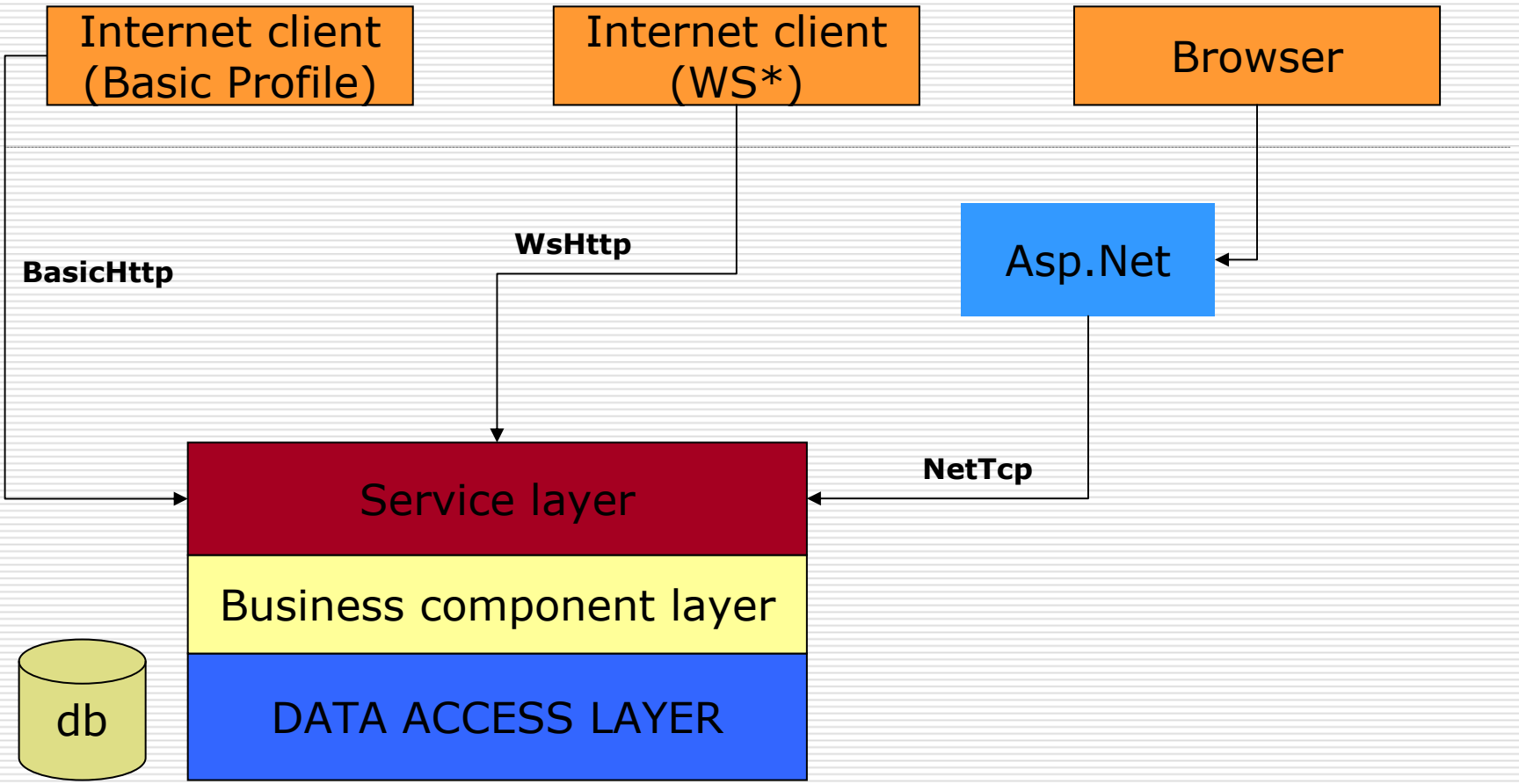
WCF CLIENT SERVER ENVIRONMENT



WCF: Webservices Architecture



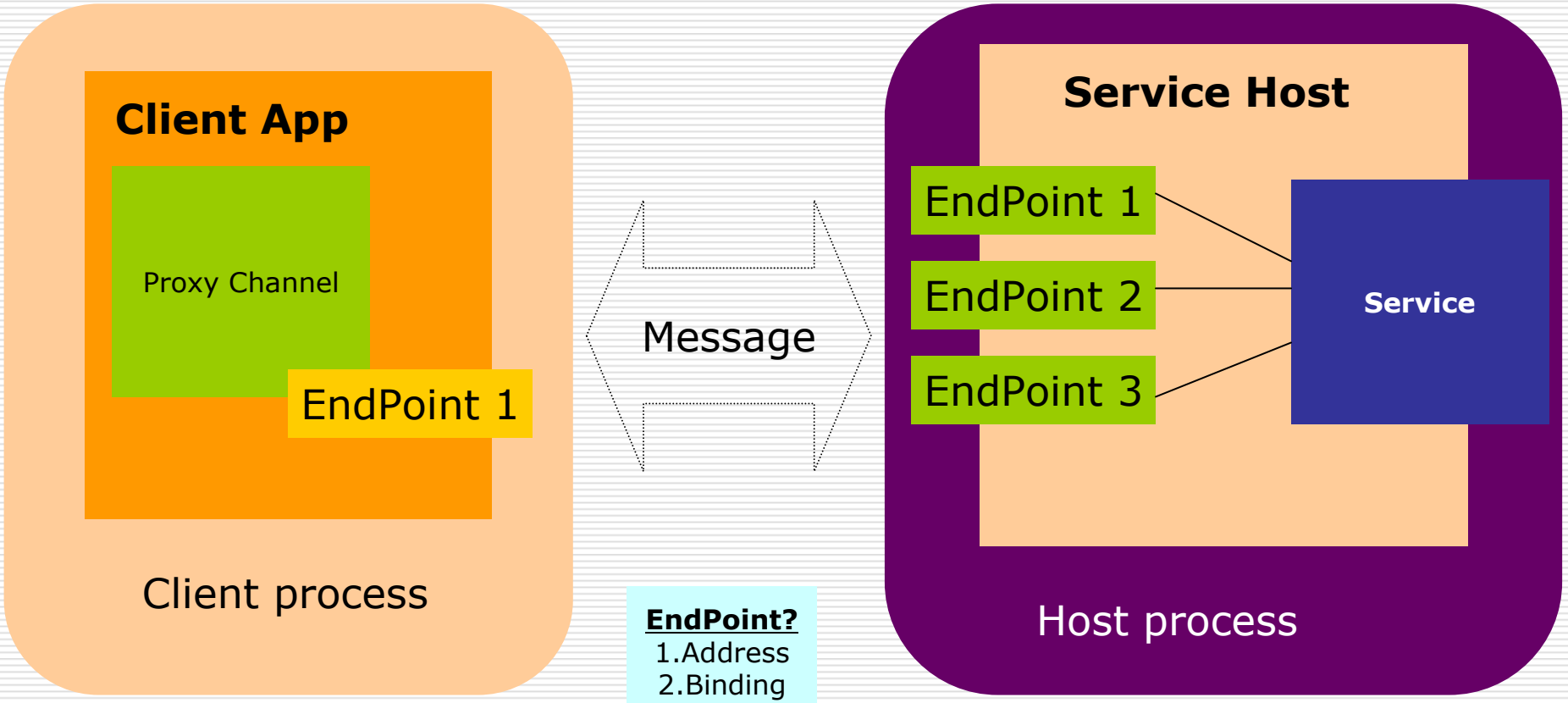
WCF: Consolidated Architecture for Distributed Apps



WCF: How to create a Service?

- Start with creating a CLR Type.
- For a functionality to be exposed in a service it needs to run in a host process and this process is the ServiceHost.
- Host Process takes the service and exposes the endpoints.
- Endpoints are locations from where the service is available for access from outside the service boundary.
- Endpoints open communication channels for external access using predefined protocols.
- Service host is opened with certain number of endpoints with predefined protocols.It then runs and waits to service incoming messages.

WCF: Service Host/Client Architecture



WCF: Communicating and Hosting

- Create a WCF Service contract and implement it.
- Host the service using a service host type to construct and open communication channels based on the end points defined.
- Endpoints will have an address, a protocol and operations to understand.
- Clients need port number, metadata and the protocol to send and receive messages to /from the service host.
- Clients need to construct proxy that can send messages to communicate with the endpoints.
- In short the communication involves address, binding and contract.

WCF: Sample App(Service)

- The whole process of creating a simple WCF Service can be broken down into 3 distinct tiers:
  - **Service**
    - Interface which has the [ServiceContract()] attribute

      This is the API Layer which basically exposes the methods to the outside world.The methods exposed to the public will be marked with the [OperationContract] tag in the ServiceContract layer.

      Without the [OperationContract] tag the methods will added to the ServiceContract and hence not accessible to the outside world.

    - We implement this ServiceContract on a viable type/class for it to be useful.

    - [ServiceContract] siginfies that if this interface is implemented in a type/class then the type or class can be hosted as a service.

    - **The Service should not be coupled with the host as we may want the service to be hosted in different ways like on IIS,Windows Service or a StandAlone Application.

WCF: Sample App(HOST)

- What is a **HOST**?

  The host is an executable process providing an application domain for the service to load.

- In order to enable access to the host we need to open or create a communication channel to it.

  For this the host needs to :

  - import System.ServiceModel namespace
  - Create an Endpoint 1. Which Protocol 2.Which Address 3.Which Operations
  - Inshort an Address,Binding,Contract needed to be provided.

  - Address : "net.tcp://localhost:9005/Customer“
  - Binding : netTcpbinding()
  - Contract : Interface name

WCF: Sample App(Client)

- What does the **CLIENT** need to access the host?

  The clients needs the metadata which is the service contract.
- It then creates proxy using the ChannelFactory of the type of the ServiceContract which takes the Binding type and service Endpoint Address eg.

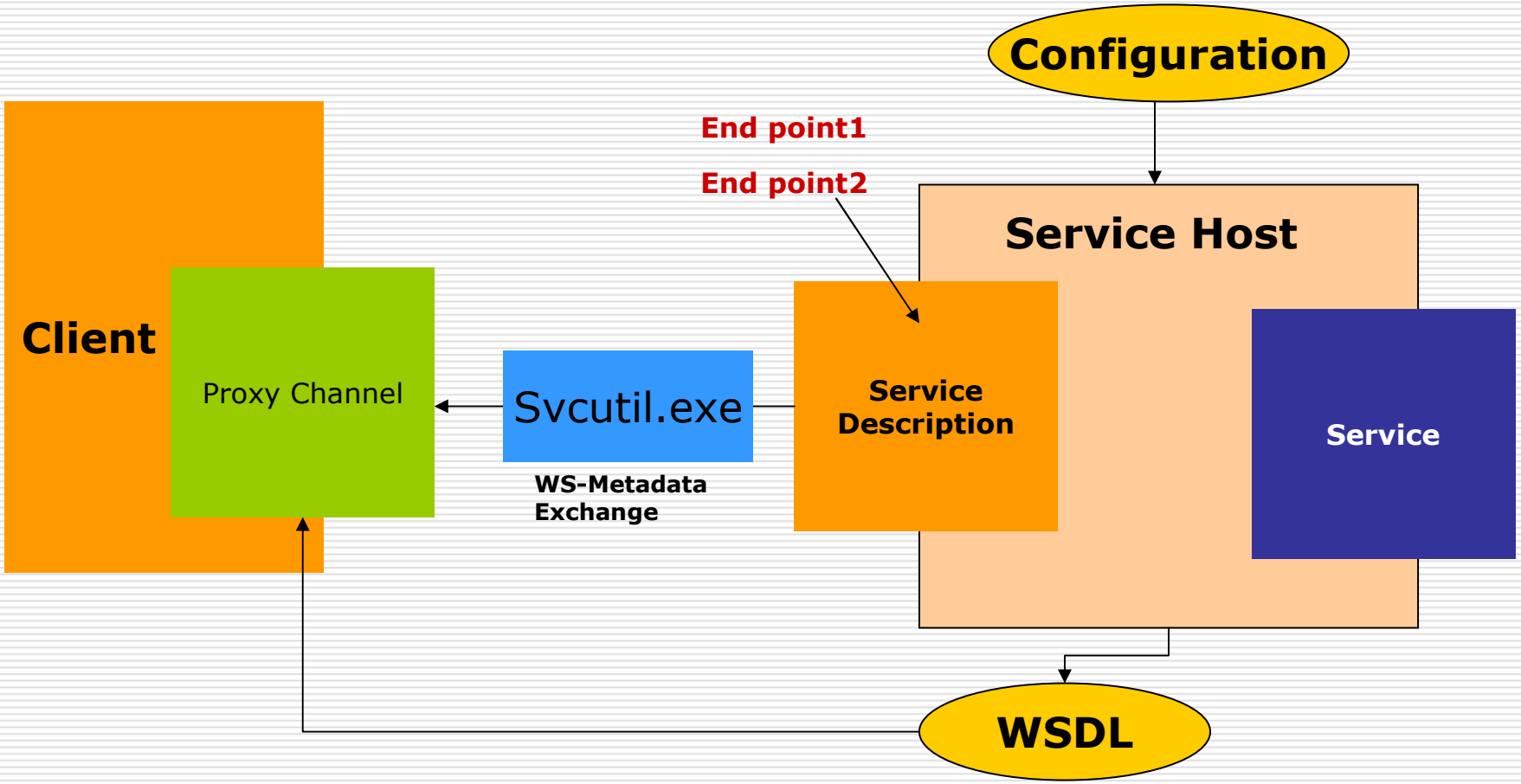
```
ICustomerService proxy =ChannelFactory<ICustomerService>
    .CreateChannel(new NetTcpBinding(),
     new EndpointAddress("net.tcp://localhost:9005/Customer"));
```

****Refer to the source code attachment for the full example.**

WCF : Contracts

- Clients should share contracts,not code.
  - WSDL provides interoperable contract for a WCF service.
  - The Service Endpoints.
  - Bindings and Operations.
  - Message and Type Defination.
  - Policies.

Contracts and Metadata: WCF Design Overview



Contracts and Metadata

- **Contracts never change**
  - Save the WSDL in a .wsdl file once the service is public to end users.
- **Policies may change**
  - Service description and related policies can be accessed dynamically using metadata exchange (WS-MetadataExchange , WS-MEX, MEX).
- **Clients rely on MEX or WSDL to consume services.**
  - To generate proxy code
  - To generate protocol configuration.
- **Standardize naming conventions**
  - Naming conventions matter for scalable client integrations.

Contracts and Metadata

- **WSDL is one representation on the service description**
- **It describes :**
  - The location where the service is located and can be reached (endpoints).
  - The protocols supported and policy requirements (bindings).
  - The operations available at each address (service contracts).
  - The messages supporting each operation (in/out/fault).
  - The schema for each message (including message contract/data contract/serializable types).

Contract and Metadata(Service Contract)

- **ServiceContract attribute**
  - Describes service operations.
  - Exposes signatures,namespaces,message exchange patterns,fault contracts,known types.
- **OperationContract attribute**
  - Operations exposed as part of the public service contract.

Contract and Metadata

- **Always apply ServiceContract attribute to interface**
  - Remove coupling to service implementation.
  - Service may implement > 1 contract.
  - Provide meaningful namespace.
  - Use single naming convention for internal and external services.
  - Consider versioning implications of namespaces.
  - Consider Webservice specification Uri conventions.