

Database Views

Offices

OfficeID	LectID	Building	Room
1	1	Gates	178
2	1	Quad	1
3	5	Gates	189
4	3	EE	120

GatesOffices

```
CREATE VIEW GatesOffices AS
SELECT OfficeID, LectID, Room
FROM Offices
WHERE Building = 'Gates'
```

OfficeID	LectID	Room
1	1	178
3	5	189

Database Views

GatesOffices

```
CREATE VIEW GatesOffices AS
SELECT OfficeID, LectID, Room
FROM Offices
WHERE Building = 'Gates'
```

OfficeID	LectID	Room
1	1	178
3	5	189

Views do not store data—they are "virtual" tables.

If we query a view, tuples are obtained from the base table so that the query can be answered.

Database Views

GatesOffices

```
CREATE VIEW GatesOffices AS
SELECT OfficeID, LectID, Room
FROM Offices
WHERE Building = 'Gates'
```

OfficeID	LectID	Room
1	1	178
3	5	189

Views do not store data—they are "virtual" tables.

If we query a view, tuples are obtained from the base table so that the query can be answered.

```
SELECT OfficeID, Room
FROM GatesOffices
WHERE LectID = 1
```

OfficeID	Room
1	178

Database Views

We can rename the columns in the view if we want to.

```
CREATE VIEW GatesOffices(OId, LIId, RoomNum) AS
SELECT OfficeID, LectID, Room
FROM Offices
WHERE Building = 'Gates'
```

GatesOffices

OId	LIId	Room
1	1	178
3	5	189

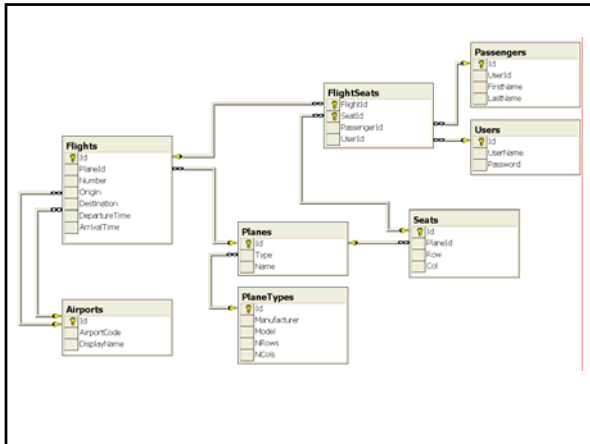
.NET DataView

- Represents a databindable, customized view of a DataTable for sorting, filtering, searching, editing, and navigation
- Associated with a single table
- Returns all columns
- Does not maintain its own copy of the data
- Can be configured in designer or programmatically
- Allows two controls to bind to the same table but show different versions of the data (e.g. one might show added rows and the other might show deleted rows)

.NET DataView

```
DataView dv = new DataView(Lecturers);
dv.RowFilter = "Name LIKE 'B%'";
dv.RowStateFilter = DataViewRowState.ModifiedCurrent;
dv.Sort = "Course ASC";
foreach (DataRowView rowView in dv)
    Console.WriteLine(rowView["Course"]);
```

Use current values of modified rows



In your assignment handout, we suggest code like this, where `airportView` is a `DataGridView`:

```
private void FilterAirportView(string columnName, string filterString)
{
    airportView.Table = airlineDataSet.Airports;
    airportView.RowFilter = columnName + " LIKE '" + filterString + "'";
    airportView.Sort = columnName + " ASC";
    airportsBindingSource.DataSource = airportView;
    airportsBindingSource.DataMember = null;
}

...

AirlineDataSet.AirportsRow originRow =
    (AirlineDataSet.AirportsRow)
        ((DataRowView)airportsBindingSource.Current).Row;

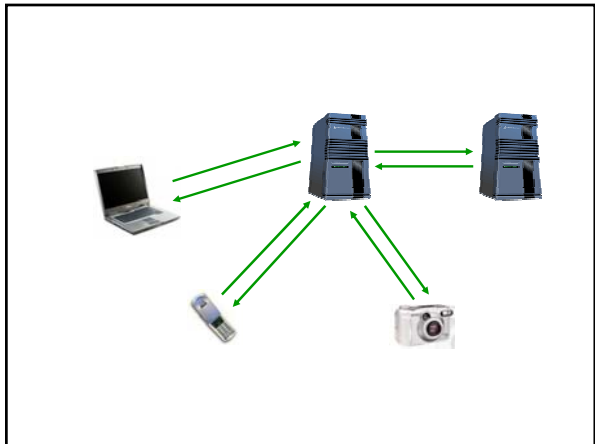
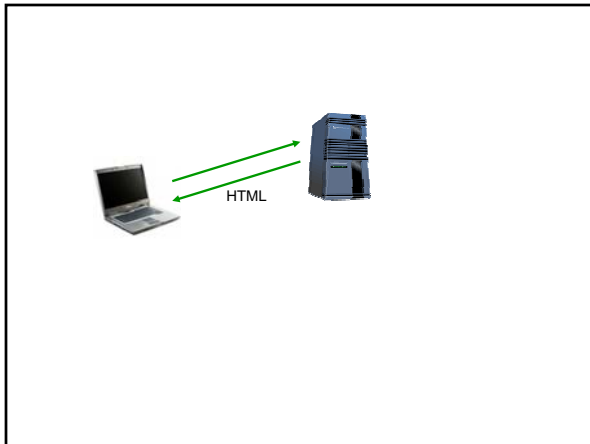
tbOrigin.Text = originRow.AirportCode;
```

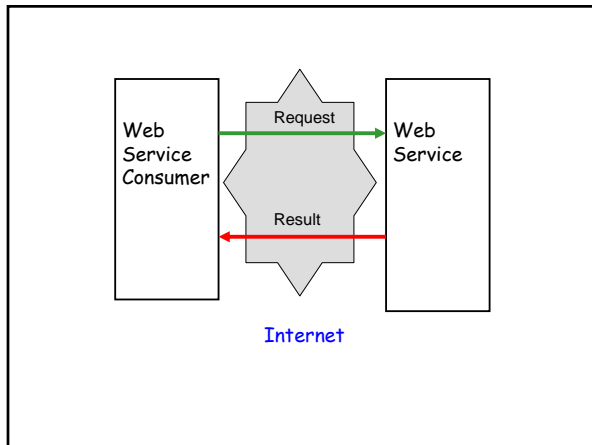
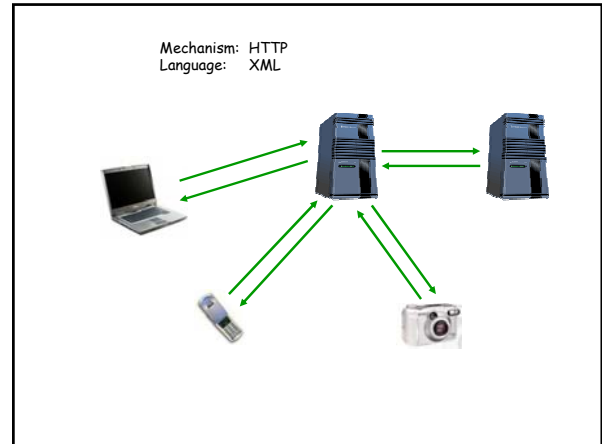
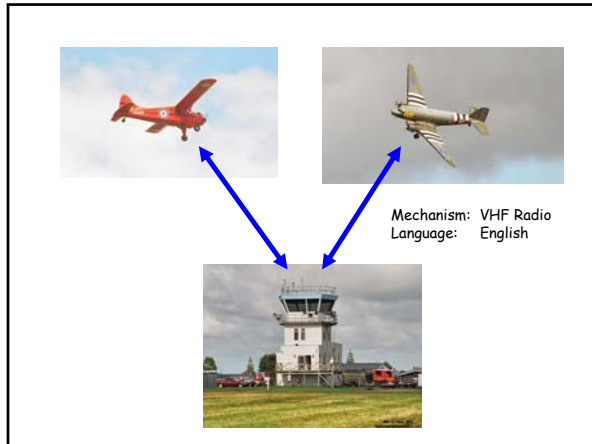
Getting Started

Airline.sql
Airports.xml
PlaneTypes.xml

What is a Web Service?

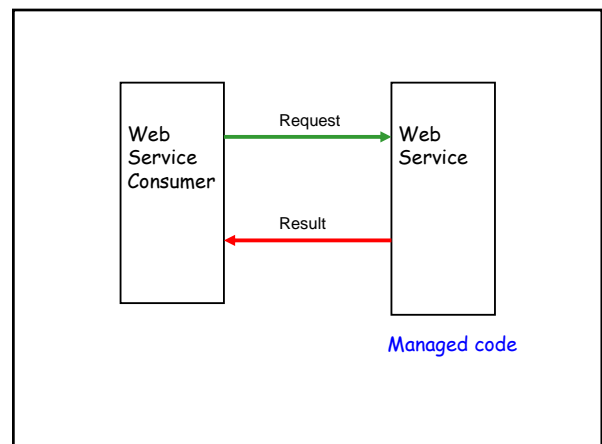
A Web Service is programmable application logic accessible via standard web protocols.

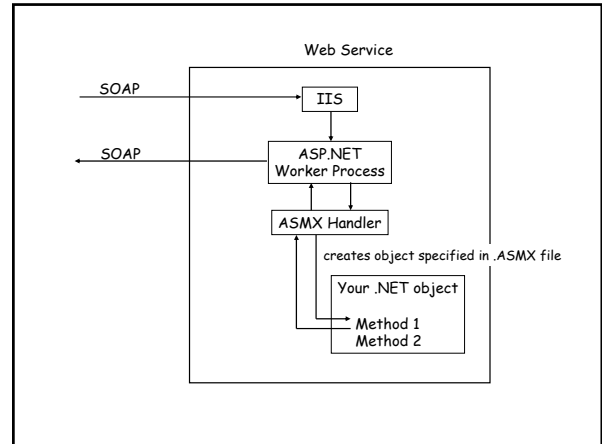
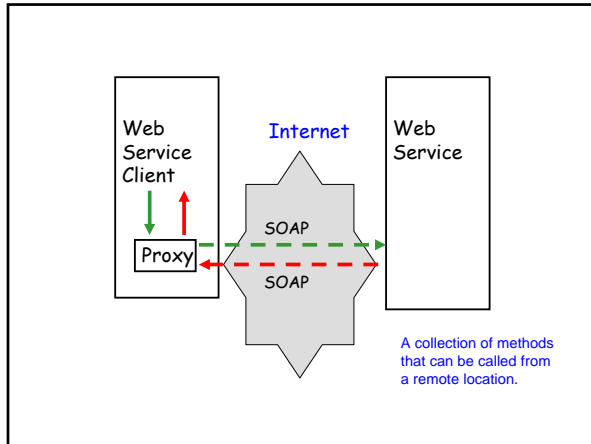




- ### Types of Web Services
- Data providers, e.g., a service providing stock quotes
 - Business-to-business process integration, e.g., purchase orders
 - Enterprise application integration

- ### Some Web Service issues
- Describing the service—how do clients know how it works?
 - Network—how can we avoid firewall issues?
 - Development tools—how can we build services in any language?
 - Plumbing—how can we simplify making all of this work?





Comparison to the Server/Browser Model

Web page	Web Service
Has a UI	No UI
Interacts with users	Interacts with applications
Works with web browser clients	Works with any type of client

