Prep:

* Warm up SQL. Clean database
* Visual Studio in admin mode + solution
* ZoomIt

DEMO 1:

* Add CV.dbml to Data.Core > Data Access
* Drag candidates table on to it.
* Show the content of it
* Talk about what we doing with data storage class
* Flesh out candidate class:

*internal DataStorage ORM { get; set; }*

*public int? ID { get; set; }*

*public byte[] CV { get; set; }*

*public string CandidateName { get; set; }*

*public bool Vetted { get; set; }*

*public List<string> ExcludedFor { get; set; }*

*public List<string> SkillsSet { get; set; }*

*public Candidate() { }*

*public Candidate(DataStorage dataStorage)*

*{*

*this.ORM = dataStorage;*

*}*

*internal void Save()*

*{*

*if (this.ORM == null)*

*{*

*throw new InvalidPropertyException("ORM has not been set", "ORM");*

*}*

*if (this.ID.HasValue)*

*{*

*var candidate = (from c in this.ORM.database.Candidates*

*where c.Id == this.ID.Value*

*select c).Single();*

*Mapper.Map(this, candidate);*

*}*

*else*

*{*

*var candidate = new DataAccess.Candidate();*

*Mapper.Map(this, candidate);*

*this.ORM.database.Candidates.InsertOnSubmit(candidate);*

*}*

*}*

Now add the following to the CV file in the view model:

*public void CreateCandidate(string candidateName, byte[] cv)*

*{*

*using (var dataStorage = new DataStorage())*

*{*

*var candidate = new Candidate(dataStorage)*

*{*

*CandidateName = candidateName,*

*CV = cv*

*};*

*candidate.Save();*

*dataStorage.Commit();*

*}*

*}*

Add CV.svc to web project and change it to

*<%@ ServiceHost Language="C#" Debug="true" Service="Data.Core.ViewModel.CV" %>*

DEMO 2:

Add new WCF Workflow Service: TermExtractor.xamlx

Change RecieveRequest name to process, click message and switch to parameters and don’t set anything.

Remove parameters on SendResponse

Add Service Reference > Discover and add CV

* Namespace: CV
* Do a build

Go to the toolbox and drag on a RetrieveNextUnvettedCandidate method

Variables:

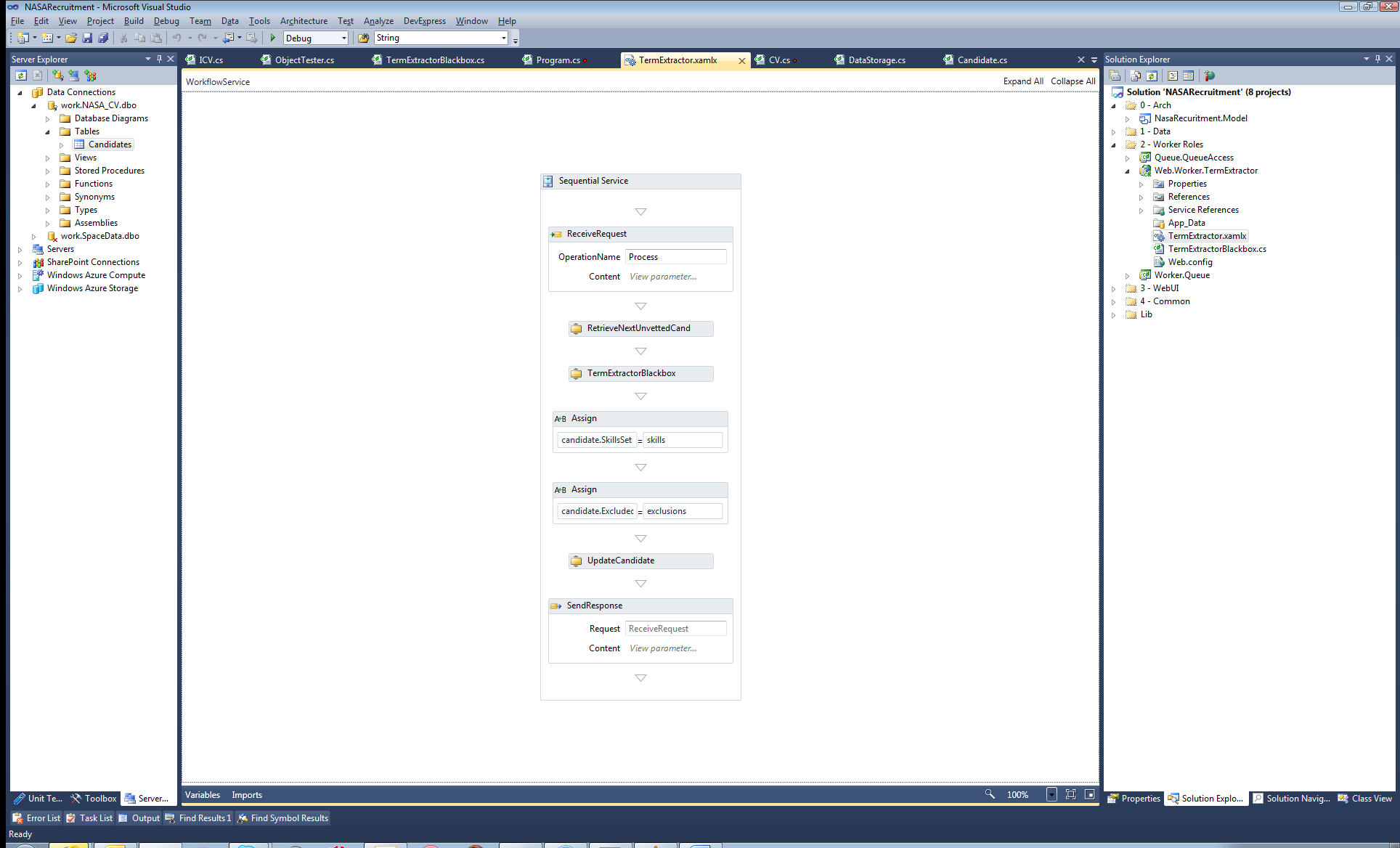
* Remove data
* Keep handle to talk about that
* Add candidate variable
* Add exclusions 7 skills (string array)
* Assign the retrieve candidate option to the candidate

Drag on a term extractor black box.

* Assign the values to the various variables we created before

Drag on two assigns to update the candidate options

Drag on an update candidate



DEMO 3:

Add term extractor web reference

*string message = queue.ReadNext();*

*if (!string.IsNullOrWhiteSpace(message))*

*{*

*Console.WriteLine("nom nom nom");*

*if (message == Messages.NewCandidateWorkMessage)*

*{*

*using (var client = new ServiceClient())*

*{*

*client.ProcessAsync();*

*}*

*}*

*}*

Update to async too

Demo 4:

* Add WIF
* Site: http://localhost:7970/
* Add a service reference to CV.
* Add following code to the home controller:

*using (var cvClient = new CVClient())*

*{*

*cvClient.Open();*

*cvClient.CreateCandidate(candidateName, fileData);*

*// no closing as dispose just calls that*

*}*

*using (var queue = new QueueAccess())*

*{*

*queue.Send(Core.Messages.NewCandidateWorkMessage);*

*}*