Windows App.config

Configuration config =

ConfigurationManager.OpenExeConfiguration(ConfigurationUserLevel.None);

if (ConfigurationManager.AppSettings["IntegrateWithPerforce"] != null)

ConfigurationManager.AppSettings["IntegrateWithPerforce"] =

e.Payload.IntegrateCheckBox.ToString();

else

config.AppSettings.Settings.Add("IntegrateWithPerforce",

e.Payload.IntegrateCheckBox.ToString());

config.Save(ConfigurationSaveMode.Modified);

config.AppSettings.Settings.Remove(key);

config.AppSettings.Settings.Add(key, value);

web.config

using System.Web.Configuration;

using System.Configuration;

Configuration config = WebConfigurationManager.OpenWebConfiguration("/");

string oldValue = config.AppSettings.Settings["SomeKey"].Value;

config.AppSettings.Settings["SomeKey"].Value = "NewValue";

config.Save(ConfigurationSaveMode.Modified);

Configuration config = configurationManager.OpenExeConfiguration(Application.ExecutablePath);

config.AppSettings.Settings.Add("Key","Value");

config.Save(ConfigurationSaveMode.Modified);

Configuration config = ConfigurationManager.OpenExeConfiguration(Application.ExecutablePath);

config.AppSettings.Settings["Key"]="Value";

config.Save(ConfigurationSaveMode.Modified);

Refresh Config

ConfigurationManager.RefreshSection("appSettings");

/\*

ConfigurationManager.AppSettings[key] = val;

Don't know why it will append the value with "," to old value, it not the case of modification.

value01,value02,value03.

solution: whatever modify or add , just remove key first , then add new value;

\*/

config.AppSettings.Settings.Remove(key);

//if (exists(key))

//ConfigurationManager.AppSettings[key] = val;

//else

config.AppSettings.Settings.Add(key, val);

config.Save(ConfigurationSaveMode.Modified);

// Important, if don't do this, it will always keep the first readed value

ConfigurationManager.RefreshSection("appSettings");

The sample entire class:

public class MetricsWinConfig

{

public Configuration config;

public MetricsWinConfig():this(Directory.GetCurrentDirectory())

{

}

public MetricsWinConfig(string execFilePath)

{

config = ConfigurationManager.OpenExeConfiguration(execFilePath);

}

public void set(string key, string val)

{

/\*

ConfigurationManager.AppSettings[key] = val;

Don't know why it will append the value with "," to old value, it not the case of

modification. value01,value02,value03.

solution: whatever modify or add , just remove key first , then add new value;

\*/

config.AppSettings.Settings.Remove(key);

//if (exists(key))

//ConfigurationManager.AppSettings[key] = val;

//else

config.AppSettings.Settings.Add(key, val);

config.Save(ConfigurationSaveMode.Modified);

// Important, if don't do this, it will always keep the first readed value

ConfigurationManager.RefreshSection("appSettings");

}

public string get(string key)

{

string ret\_val = string.Empty;

if(exists(key)) ret\_val = ConfigurationManager.AppSettings[key];

return ret\_val;

}

public bool exists(string key)

{

return ConfigurationManager.AppSettings[key]==null?false:true;

}

}

Usage: MetricsWinConfig config = new MetricsWinConfig();

if (config.exists("dbserver"))

this.txt\_dbserver.Text = config.get("dbserver");

else

config.set("dbserver", this.txt\_dbserver.Text);