Teleoperating Home Devices

Zeddy Chirombe i.z.chirombe@wustl.edu

David Kebo Houngninou houngninou@wustl.edu

Gene Kwan Lee gkl1@wustl.edu

ABSTRACT

Currently, people resort to calling their neighbors, friends, relatives and even strangers for intervention; whenever emergencies knock on their residence doors while they are far from home. We have all been required at some point in time to be in more than one place at a time, and may have experienced how frustrating the circumstances can be. For instance, on your way home from work or other places, you are caught in traffic, in a huge storm and even though you may have storm protectors for your house, the benefit is only realized with you being physically present at home. Say, you left you garage door open, and you are walking your son/daughter home from school, when a tornado touches down, wouldn't it be great if you could activate garage door closure from any place to protect your investment? Another scenario; your flight is just touching down and you are many miles away from home, when you suddenly realized that you left garden horse running, wouldn't it be great if you could actually shut that valve off from any place you are? As we can all see, these scenarios are converging or demanding of us to be in more than one place at time. Consequently, we are proposing a wireless home control system that we feel will close the gap, with the hope that everyone can be very many miles away from home but never be away from home.

General Terms

Performance, Design, Reliability, Security

Keywords

TinyOS, TelosB, Servomotor, Wireless, Home automation, HomeShield, Storm Protector, Teleoperation,

1. INTRODUCTION

Looking at devastating effects or total destruction caused by hurricane Katrina, Rita, etc., it's almost 10 years and Louisiana is still a rabble city. Even though most property owners carried insurance, the insurance careers could not afford to rebuild the entire city, without going under themselves. According to FEMA estimate, more than hundred billion dollars would be needed to bring New Orleans back to its first world status.

Although we do not seek to discount the fact that much, work would need to be done after any catastrophic disaster; we argue that the degree of home destruction is way over the top. We seek to inspire the world to invest in structurally sound but very simple shield initiative that we feel has the potential to drastically reduce the intensity of damage for both residential and business structures from weather related damages. We do not feel our idea will cure all, but definitely will reduce the impact, thereby saving the economy billions along the way.

Problem Description

Researching on structural engineering, particularly on how residential structures falter, and not necessarily why they fall, we quickly realized that the degree of destruction to any physical structure varies indirectly with the structure's ability to withstand changes in the atmospheric pressures outside. These structures are just but victims of aerodynamics, which for long has had very little attention. As wind speed increases, so as the pressure outside. Soon or later, window panes on residential structures fail to stand the inward pressure forcing them to curve in. Eventually, they break, leaving trail through which more wind floods the house to a point of blowing the whole house top off, sending fast moving debris into the air. From the debris, those two-by-four wooden blocks will become the agents in destruction of even more windows within the subdivision, compounding the damage many folds over. The end result is the scenery FEMA is not very enthusiastic about.

Hurricane Katrina in 2005 was coined the costliest natural disaster, as well as one of the five deadliest hurricanes ever recorded in the history of the United States. It was the sixth strongest overall among recorded Atlantic hurricanes. At least 1,836 people lost their lives in the actual hurricane and in the subsequent floods, making it the deadliest U.S. hurricane since the 1928 Okeechobee hurricane; total property damage was estimated at $81 billion. Hurricane Katrina had already killed at least 60 people along its track through Florida, before making an unforgettable pound on Louisiana, Mississippi, and Alabama.

Although the structural community agrees with the danger at hand, many homeowners can argue, with enough advance warning, manually bolting some strong plywood well ahead of time, they can protect their homes. But, as we all know, there are way too many factors that quickly bust that theory. First, not all neighbors will be around to respond to the warning call. Second, not all neighbors will be willing participants to climb up their walls to activate the protection system. Third, even though you get a solid buy in from all stockholders, what are the changes these participant will be able to manually protect all twenty plus windows of their homes, before the danger strikes? Besides, you only have a matter of seconds to act when say a Tornado touches down; else, your very own life is in total danger.

Proposed Solution

With the status quo premise as described above, our proposal is built on the idea that only if our homes can be intelligent enough to activate their own protective shields, to protect their windows from outside atmospheric pressures, then can they withstand the a big chunk of the destructive force and minimize by great margins the wrath of mother nature.

Even though the ideal situation is a Home System that can SENSE the world, PLAN the course of action, and then ACT, our research, is focusing on the human to do the sensing, planning, as well as activating/deactivating the home shield, or any other devices paired with your home system. Although the system we are proposing can be used for literally any home activity, we chose Storm Protectors for our proof of concept.

The ability for a homeowner to tele-commute home from anyplace in the world where there is internet access, and be able to, on a push of a button, activate the home shield to close all storm windows or storm protectors, securing his/her house from weather dangers, amounts to improved convenience, comfort, and most importantly, billions of dollars that will be saved collectively from across the nation. If you follow money trails following every disaster, you will quickly find that Insurance companies pay a fortune in rebuilding homes, but in return, they pass on those costs back to the consumer through upward adjustment of yearly premiums. That said, limiting unnecessary drainage of funds from the financial sector, means more money and power in the hands of the consumers, which in theory is the primary driver for advancing and promoting principles that value human dignity worldwide.

Even though our idea is basically a conservation project, we would not have done much service if we had not think green or at least attempt to address global warming aspect of our work. From the onset, it may appear that we would need too much energy for the Home Shield System's upkeep, however the reverse is true. As it turns out, every entity in our Home Shield Wireless sensor network run on 2 1.5V AA batteries for at least 6month. This is what we would like to consider as the biggest green initiative for our day – using AA batteries to protect homes destruction from Tornados and Hurricanes.

Project Conceptual Overview

The flagship, or the basis to our "Think Green" initiative, is the low cost low power TelosB hardware platform that we use for all our sensing and wireless communication that is required within every home. Using this platform, we guarantee huge energy savings, while providing service just as good as all other projects that use main power supply

Also, because our TelosBs are running Tiny Operating System (TinyOS) built on NesC, making it an open source product suite targeted for embedded systems, our product is readily available for use by anyone with an interest in staying in total control of their home. Taking advantage of the rich sensors like temperature, humidity, light, in addition to other external sensors already coming standard with the TelosB, we can monitor the physical and environmental conditions of the entire house and use that data to make some intelligent choices.

Our project's primary focuses is two fields; 1) wirelessly, and in parallel, command all storm windows to OPEN/CLOSE at the push of a button. 2) giving the user the ability to check/change and activate home shield system states remotely is our pledge.

For our goal to be realized, we used the smart phone as the Internet gateway, which connects via secure shell host (SSH) to the home physical network, into a Linux server running the Base Station. We didn't have to build our own for the class, as there are many off the shelf Android-based products that can do just that. We chose ConnectBot, and it worked flawlessly. Once connected to the server, the user will issue through the Base Station TelosB mote, broadcasts to all client TelosBs interfacing with the Storm Protectors. These TelosB motes mounted on every Storm Protector will send a CLOSE/OPEN pulse to the servomotor, which will then drive the Storm protectors to OPEN/CLOSE position. This is just the quick summary of how the project is designed to work.

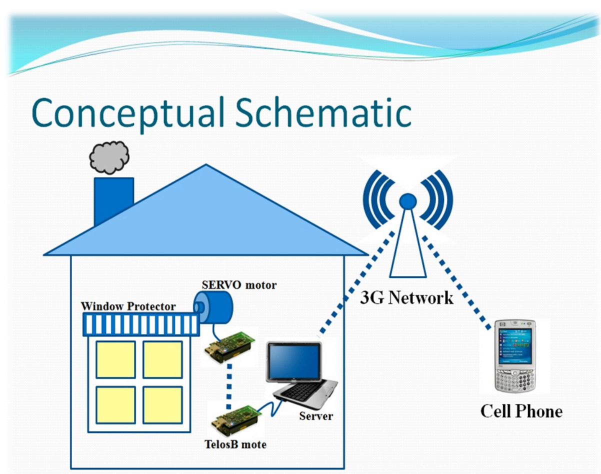


Diagram 1: End-To-End Project Layout

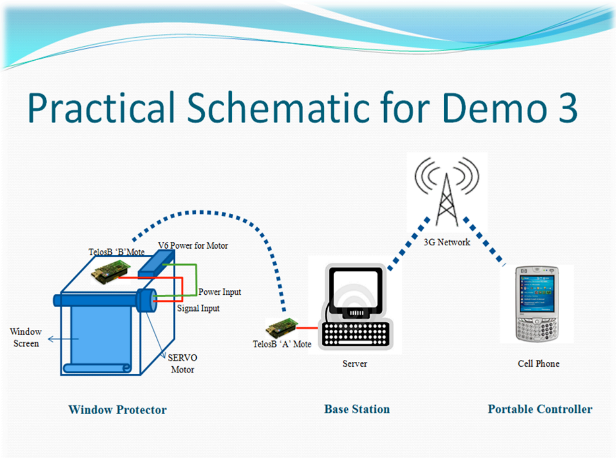


Diagram 2: End to end Project Cross-Sectional View

As part of the wireless network we use 3 TelosB motes: base-station mote, Storm Protector mote and sensing mote. **Base-station mote**: This mote is connected to the server laptop and sends/receives messages to/from the other nodes.
Storm Protector mote: This mote activates and controls the rotation of the servomotor.
Sensing mote: This mote monitors the motion of the gate, and detects when the gate reaches the top/bottom of the window.

Why did we use a servomotor?
The choice of a servomotor for controlling the storm protector was driven by the low power consumption (5V DC) and the control using pulse width modulation PWM (See Figure 2). The servomotor is controlled with a pulse signal, where the rotational speed and direction are determined by the duration of that pulse. For obvious reasons, our first exposure to the servomotor apparatus, made a permanent appeal at first sight, making our design quest somewhat easier what we had initially thought.. We quickly abandoned our original idea of manually bread-boarding latching relays to control power to the motor as that made obsolete right from the onset.

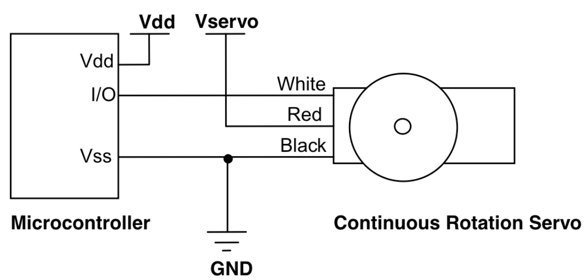


Figure 2: Power & control of the servomotor

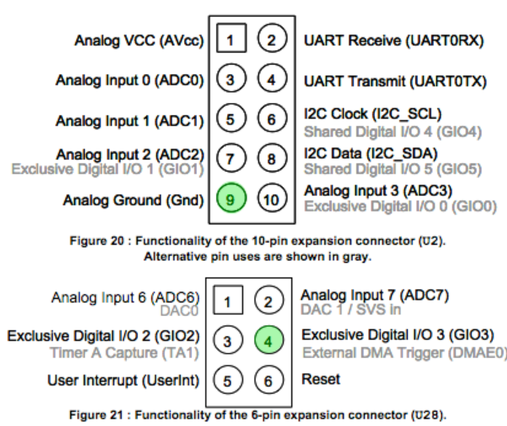


Figure 3: TelosB 6-pin expansion connector

1.1 Hardware

Components:

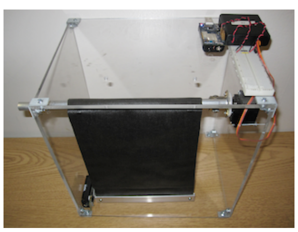


Figure 1a

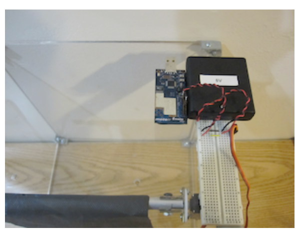


Figure 1b

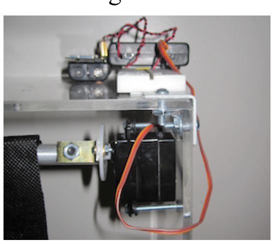


Figure 1c

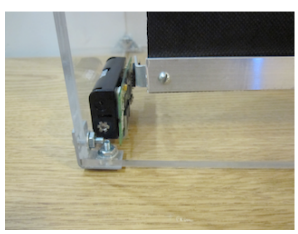


Figure 1d

Figure 1: System overview

The setup is a miniature storm protector built and scaled down for demo purpose only. The setup consists of:

- 01 Plexiglas enclosure
- 01 aluminum axle for rolling the protector up or down
- 01 continuous rotation servomotor (Futaba GWS S35)
- 01 5V DC Power source
- 03 TelosB/Tmote Sky Motes
- 01 laptop running as server
- 01 cellular phone with SSH access to the server

The TelosB mote has two expansion connectors: a 10-pin expansion connector (U2) and a 6-pin expansion connector (U28). For our control signal, we used pin 4: Exclusive Digital I/O on the 6-pin expansion connector and pin 9: Analog Ground on the 10-pin expansion connector (See Figure 3). Pin 4 outputs a pulse that can be set using the component HplMsp430GeneralIOC for general-purpose I/O. The interface in TinyOS provides the commands 'set' and 'clear', which sets a desired pin the high or low respectively. Using these two commands, the pulse width is easily adjusted to control the rotation of the servomotor. The servomotor has three possible states: no rotation, rotation clockwise, and rotation counterclockwise.

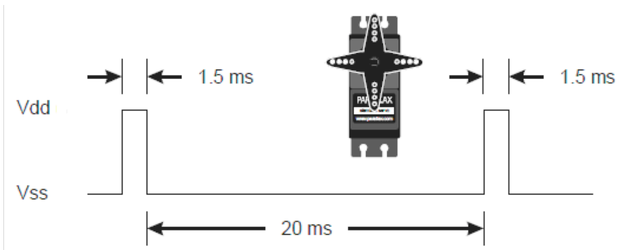


Figure 4: Servomotor motionless, pulse width = 1.5ms

When the pulse width generated by the output pin 4 is equal to 1.5ms, the servomotor is motionless.

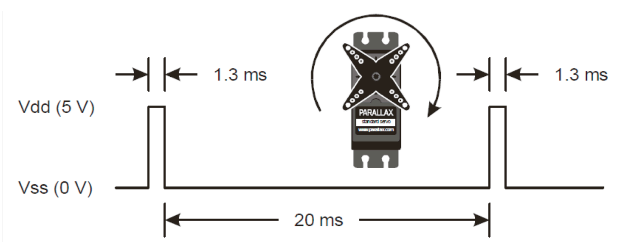


Figure 5: Servomotor clockwise rotation, pulse width = 1.3ms

As the length of the pulse decreases from 1.5 ms, the servomotor gradually rotates faster in the clockwise direction.

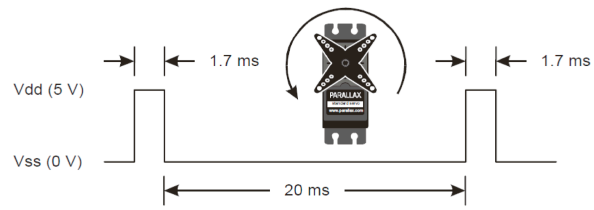


Figure 6: Servomotor counterclockwise rotation, pulse width = 1.7ms

As the length of the pulse increases from 1.5 ms, the servomotor gradually rotates faster in the counter-clockwise direction.

Note: For this application, the servomotor that we use is a 'continuous rotation servomotor', which means it can turn freely at any angle without any constraints.

1.2 Software

Using NesC programming language, we designed a homogenous component that runs on all TelosB Motes. The primary function of this mote, is to receive OPEN/CLOSE wireless messages, and then generate an equivalent pulse and sending it to the Servo. This pulse is what the Servo uses to drive or wind the storm protectors up/down. Now we know how to activate or deactivate the desired states, but where does the OPEN/CLOSE signals originate from? It turns out; we did not have to write our own access point, whose sole function would have been to receive human commands and broadcasting them to all client TelosB motes. This access point or basestation as the its authors called it, attaches to the serial interface to connect to the traditional Linux based servers, and uses the TelosB chipset for all its wireless needs.

In our case, we took Java as our weapon of choice. Apparently, NesC ships with an application called Message Interface Generator, MIG for short, whose purpose is to generate a communication stub/driver that bridges the gap between X86 chipset with the TelosB chipsets. The interface generated speaks high-level languages like C/C++, Java, etc, do the translation, and then speaks NesC on the other side. As such, after linking our java client program, with this interface, we can send our packets to the serial port. The base-station mote picks those packets and broadcast them to every node within its range. When the field TelosB motes sends back their ACKs, or status messages, the Base-station mote picks these up and push them through the serial port, where they will be picked up by the Base-station application which then forwards the message back to our Java program which runs our business logic. This is the program remote users interact with whenever they connect from their smart phones from anywhere in the world.

Because we understand how difficult it is to type so many parameters on a smart phone, we have provided a script, which sets the Linux environment up when a user connected, as well as connecting to the serial interface – getting you ready to send messages.

The remote user must connect to the Linux Server via SSH, and then execute "gate" script from the command line. This script automatically sets up and activates the java program running our business logic, and then avails a prompt to the user to enter operation commands – OPEN/CLOSE in this case. Once the user makes a selection, a packet is sent to the base-station, which broadcast that message to all Storm Protector nodes.

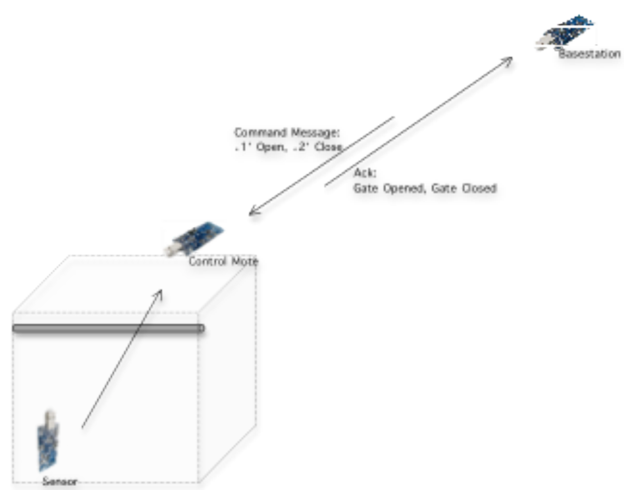
Storm Protector nodes are continuously listening for incoming packets from the base-station. Once a packet is received from the base-station, this node activates the servomotor for the corresponding action. As described in the section above, the direction of the servomotor rotation depends on the pulse width generated by the Storm Protector mote. We defined two arbitrary messages for a rotation: '1' means clockwise (closing the gate) and '2' means counterclockwise (opening the gate). When the control node receives "1' from the base-station, it sets the PWM to 2.1 ms, which automatically rotates the servomotor clockwise. When the control node receives '2' from the base-station, it sets the PWM to 9 ms, which automatically rotates the servomotor counterclockwise.

As discussed earlier, a servo has only three commands Rotate, LEFT, RIGHT, and STOP. So, our main challenge was how to tell the servo when to stop. As such, our first attempt to stop the servomotor after a full rotation up or down was to use a timer in the program running on the Storm Protector mote. The servomotor would run for a fixed time in second, and stop after the timer elapses. This approach was unsuccessful because the RPM of the servomotor is not constant, and causes the servomotor to rotate faster or slower sometimes depending on factors beyond our control. From power intensity to varying speed irrespective of power capacity.

To solve this issue, we introduced a third mote, with a built-in photo sensor to monitor the motion of the servomotor. The sensing node is placed at the bottom of the Plexiglas box. When the protector reaches the bottom of the window, the photo sensor on the mote detects that the moving object has reached the bottom and sends a message to the Storm Protector mote to stop the rotation. The second purpose of the sensing mote is safety. If an object is placed right below the windows, the photo sensor detects the object due to the change in luminosity. Next, the sensing mote sends a message to the control mote, and the servomotor doesn't move until the surface is cleared.

We also implemented a set of acknowledgments at the control node level. When the window is fully closed or fully opened, the mote sends an acknowledgement message to the base-station including the current state of operation. The base-station forwards that message to the server via serial. The acknowledgements received from the control node are appended to a timestamp and stored in a data file for records

Since every Storm Protector unite operates on 3 Telosb Motes; 1) receives broadcasts from the Base-station, and pass active signals to the server. The other two motes are for detecting when to stop the servo when the protector is fully closed or retracted. As you can see, there is a very high chance of cross talk. To mitigate that, we made use of the Node Id feature in the TelosB infrastructure, to isolate communication to a specific set. The diagram below illustrates the communication flow between the three major parts of the system.



The communication between the base-station and the Storm Protector mote is bidirectional.

The communication between the Storm Protector mote and the sensing mote is unidirectional.

2. Related work

In general, home automation is a very active research area currently. For instance, [1] "A ZigBee-Based Home Automation System" by Khusvinder, Yang, Yao, and Lu, alluded to the fact that many home initiatives have been dealt with for some time now, but adoption thereof has been far too slow. One of the biggest changes that push most people off the cab is wireless security, which is in addition to the questionable value you get from home automation, versus the good old. Yes, we can see why people can be skeptical, but is that the only side of the coin?

Almost when every day, there are some weather based disasters happening somewhere in the continental US. There is hardly a day that goes by without hearing about some property damage. Check [2-4] for more weather related destruction. If that is not enough, then try reflecting on Hurricane Katrina's TV footages or check [5-7] for your references.

With these devastating scenarios at hand, we are saying, yes, there are some disadvantages to using home automation, but we feel the benefit far outweighs the risk. If you are afraid of home a automation hacker, think of it this way, aren't you better of dealing with a human hacker as opposed to the destructive force that comes with a tornado or hurricane destroyer? We encourage our readers to choose their devil, but to do so wisely.

3. Lessons learned

Combining TelosB Motes with Servomotor proved to us that these two independently designed apparatus are quite modular and worked together seamlessly. This combination worked fine every time we put them to work.

From our previous project experience with the TelosB radio, where multiple senders and multiple receivers ended up with very bad packet loss and packet collision, we didn't think TelosB devices were that great. However, in this implementation, the TelosBs have been extremely reliable. In our test bed for this project, the maximum range we tested was 20 meters, and we figure Collection Tree Protocol can be employed should these low power mote can't provide coverage for the entire house. But, if the Base-station is stationed in the middle of the house, then you are probably going to get much of the coverage you need.

Next we noticed that the RPM of the servomotor did not accurately match the specifications of the datasheet. The specifications listed 50 RPM with a linear response. However we noticed that this value varied depending on the charge of the DC power supply. When the 5V DC battery holder includes brand new batteries, the high current causes the servomotor to rotate faster.

The biggest advantage for using a servomotor, is the high torque that it can generate from very low power, to close and open our storm protectors seamlessly, a fine control using PWM and low-power consumption

This project involved the use of the photo sensor on one of the mote. We had to make sure that the calibration of this sensor was correct according to the environment in which the system is tested.

Last but not least, there was very little to no interference from WIFI, or from other home based wireless devices. In fact, as part of our experiment we had to have a WIFI access point that we used as the gateway through which our smart phone could send commands to our Base Station to operate the Storm Protecting Shield. We did not see any negative side effects from our system.

3.1 Conclusion & future work

Having established that damage residential structures suffer are mostly due to harsh weather conditions which starts with window breakage caused by fly debris, or they give in to the outside atmospheric pressures induced by hurricanes and tornados, or some other strong wind patterns, we feel our project idea can offer the support our homes need to withstand some of these dangers.

Considering a home is the biggest asset middle class people ever own, we feel that our simple idea has the ability to protect the most treasured asset, and there by reducing speculative practice by insurance companies. As far as the technology goes, there is no doubt that running a Storm Shield System on few 1.5 AAA battery power for 6+ months is the best most cost effective, efficient, and best alternative to current product suites. The fact that we were able to craft a fully functional prototype in few weeks, that show how basic our idea is, and can be mass produced for cheap, and can be an inspiration to many of those "do it yourself" fellows out there.

Future Work

In this project, our main goal was to give the user the flexibility to be in more than one place at once. Remotely teleoperating home device was for long the unreachable ideal. Having placed all that control in the users palm, by means of syncing up our smart phones with home based system, and be able to effect change remotely, and act based on intelligent feedback therefrom, we may not want to settle there. The next phase would be to link up, in addition to the storm protectors, all other wireless sensor network in our homes.

Temperature Sensor Net: Wouldn't you want your system to notify you whenever your home temperature falls outside a health range, and be able to work with you on alternative routes to explore, or for you to just intervene as you normally would if you were home?

Your Security System thinks there is an intruder, wouldn't it be nice for you to peruse the data that was used to derive the intrusion state, and be able to capture some vital info right from the palm of your hand, irrespective of your physical location?

REFERENCES

[1] Khusvinder Gill, Shuang-Hua Yang, Fang Yao, and Xin Lu "A ZigBee-Based Home Automation System,
[2] Associated Press; CHARLOTTE, N.C. - 3/28/2010 "Violent storms damage N.C. homes" http://www.msnbc.msn.com/id/36077021/ns/weather/
[3] CBSNEWS; BELLEMONT, Arizona. - 10/07/2010

"Tornadoes Rip through Arizona, Destroy Homes"
http://www.cbsnews.com/stories/2010/10/07/national/main6934988.shtml
[4] Associated Press; EUSTIS, Florida. - 9/21/2007
"Possible tornado damages 50 Florida homes"
http://www.msnbc.msn.com/id/20903232/ns/weather/
[5] Hurricane Katrina causing deaths and flooding
http://en.wikipedia.org/wiki/Hurricane_Katrina
[6] Hurricane Rita most intense tropical cyclone ever seen in the Gult of Mexco
http://en.wikipedia.org/wiki/Hurricane_Rita
[7] Texas hurricanes in history
http://www.usatoday.com/weather/hurricane/history/whtexas.htm