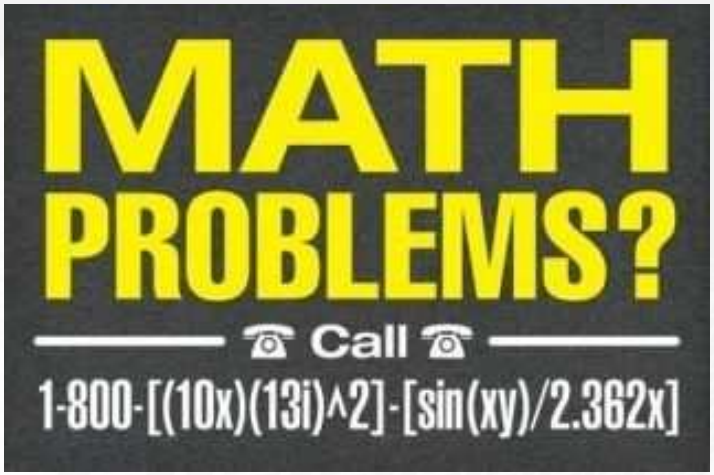
# Develop Web of Things




***Yes, we scan…***

http://en.wikipedia.org/wiki/Web_of_Things

# Web of No Things


Tricorder…

not (A v B) = not A ∧ not B

push a Session or pull a Service ?

Redundant DNA Helix

Hertz App or DigiCam UseCase?

Build Systems you can count on

http://en.wikipedia.org/wiki/Anti-pattern

http://en.wikipedia.org/wiki/Wireless_Sensor_Networks

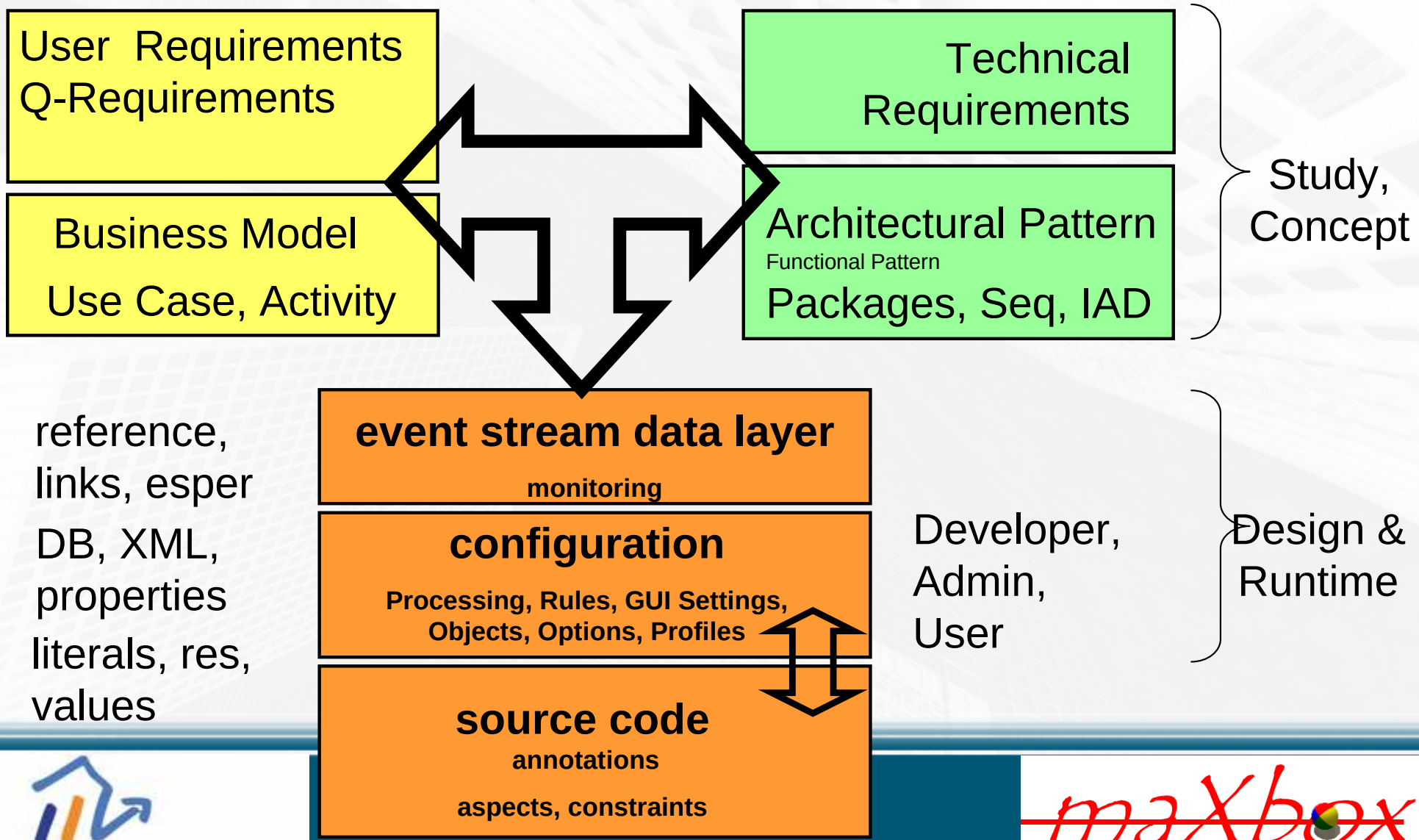
# Web of Words

Fest codiert in der Erden
Steht die App, aus Bits gebrannt.
Heute muß die Software werden,
Frisch, Entwickler! seid zur Hand.
maXbox

http://en.wikipedia.org/wiki/RFID

Chess Roboter Topic – First Web of Things

# WoT Code Layers



# How to start?

Do you want to work with “things” that are under your direct control?

Things:

- Easy to use with a wide community support are Arduino, Raspberry and SunSpot.
- Crossbow, Libellium, Sensinode, etc. are also possible solutions but may require more effort.
- For very specific solutions you may need to go for hardware design.

maXbox

# Programming the Things

- **Complex and time consuming process**
- tool chain: IDE, compiler, debugger
- microcontroller is programmed and executes the code, radio chip is not programmed but controlled by microcontroller, usually via SPI (Serial Peripheral Interface Bus) which sets/reads registers
- compiled code is loaded to the microcontroller using bootloader or JTAG (Joint Test Action Group)
- protocol stack may be precompiled and available through API or available as library, operating system (not needed for simple tasks), virtual machine (optional)

- http://www.mikrocontroller.net/articles/JTAG

# Decison Process

- **Before starting, the following questions should be answered:**
- **What is the scope or application?**
- •Monitoring measurements?

- **What is the scenario Use Case?**
- •A thing with embedded web service?
- •A set of things connected through a gateway?

- **What programming language or IDE?**
- •Options: C, Pascal, Java, C#, Processing, ADT

- **What is the publishing infrastructure?**
- •None, custom, third party.

maXbox

# Stream Thinking

```
procedure letStreamDataWork;
var biglist: TStringList;
begin
  biglist:= TStringlist.create;
  memo2.setfocus;
  repeat
    biglist.add('Value for mem[%d] enter : '
                +inttoStr(RandomRange(500,100000000)));
    biglist.add(getbigPI+getbigPI+getbigPI+getbigPI)
    with TJvMemoryInfos.Create(self) do begin
      writeln('Available Mem: '+FreeMemory);
      Free;
    end;
  until isKeypressed;
  writeln('elements of biglist '+intToStr(biglist.count))
  writeln('capacity of biglist '+intToStr(biglist.capacity))
  biglist.Free; //destruct
end;
```

maXbox

Kleiner Kommunikation.....

# About Rules

- **CA1303: Do not pass literals as localized parameters**
- `public void TimeMethod(int hour, int minute)`

  `{ if (hour < 0 || hour > 23) { MessageBox.Show( "The valid range is 0 - 23."); //CA1303 fires because the parameter for method Show is Text }`
- **CA1302: Do not hardcode locale specific strings**
- `static void Main()`

  `{ string string0 = "C:";`
- PMD: Avoid duplicate literals (string or numeric)

# Timeline - Lord of the Things

- **Description and Technologies**
- **Web 1.0** **Static** HTML pages (web as we first learned it) HTML, HTTP
- **Web 1.5** **Dynamic** HTML content (web as we know it) Client side (JavaScript, DHTML, Flash, …), server side (CGI, PHP, Perl, ASP/.NET, JSP, …)
- **Web 2.0** **Participatory** information sharing, interoperability, user-centered design, and collaboration on the World Wide Web (web of people) weblogs, social bookmarking, social tagging, wikis, podcasts, RSS feeds, many-to-many publishing, web services, … URI, XML, RDF, OWL, SparQL, …
- **Web 3.0** …definitions vary a lot – from Full Semantic Web to AI
- (web as we would need it) http://en.wikipedia.org/wiki/Web_3.0#Web_3.0
-
- **Web of Things** Everyday devices and objects are connected by fully integrating them to the Web. (web as we would like it) Well-accepted and understood standards and blueprints (such as URI, HTTP, REST, Atom, etc.) http://en.wikipedia.org/wiki/Web_of_Things
- Singularity **Web of Rings** (Thanks to Kurzweil)

# Timeline II

- **A Short History of Time**
- 1991 Application Program
- 1995 Application
- 1998 Applet
- 2010 App
- 2015 A

(Android, Arduino, Apache, ARM)

maXbox

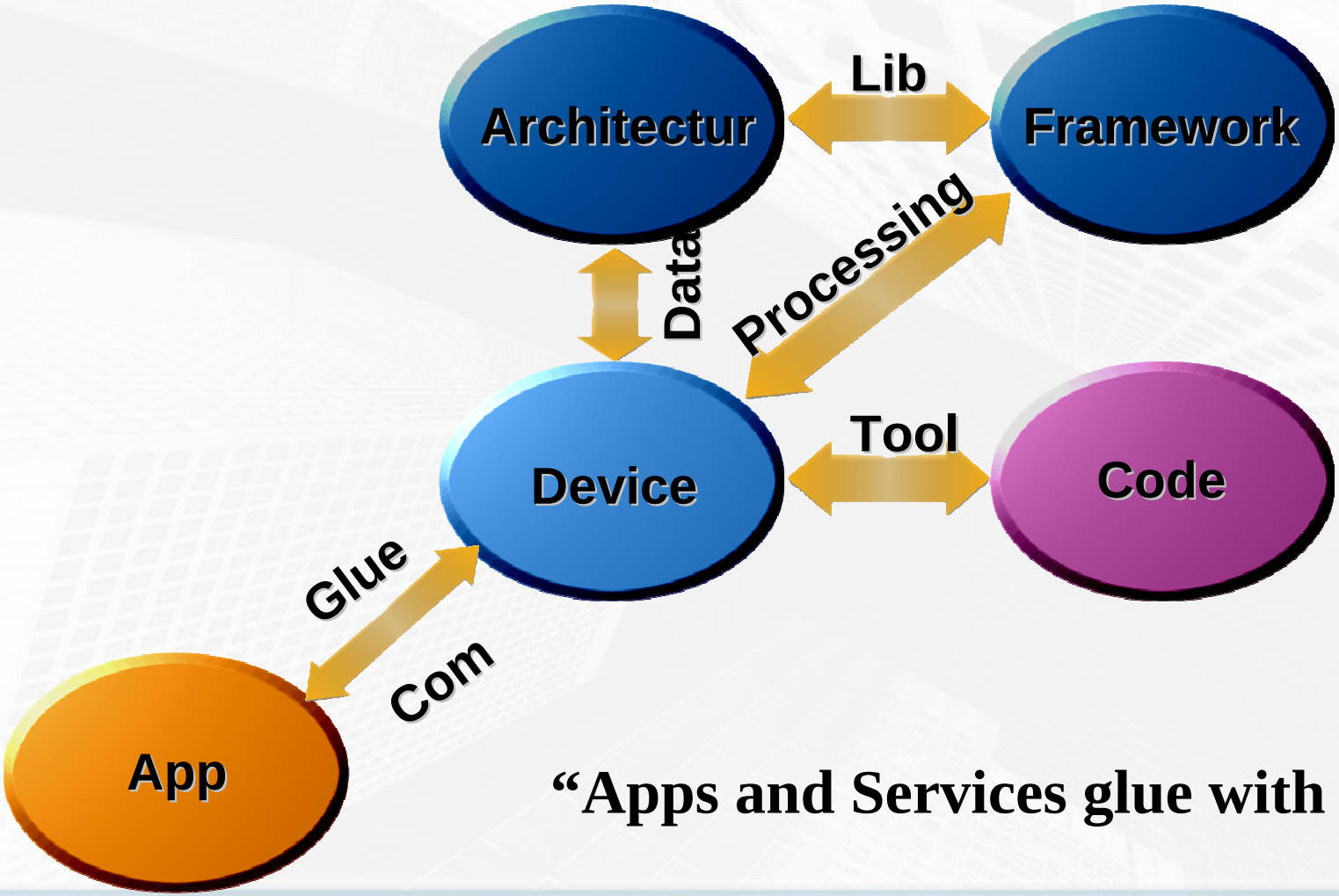
# WoT Use Case

Motivated by an increased interest of physical computing and embedded in automatic management of large systems

- •Power grids
- •Transport systems
- •Water distribution
- •Logistics
- •Industrial automation, 3D-Printer
- •Health, example Schiller Poster
- •Environmental intelligence
- •Academic, example maXbox
- •Distributed sensing infrastructure

maXbox

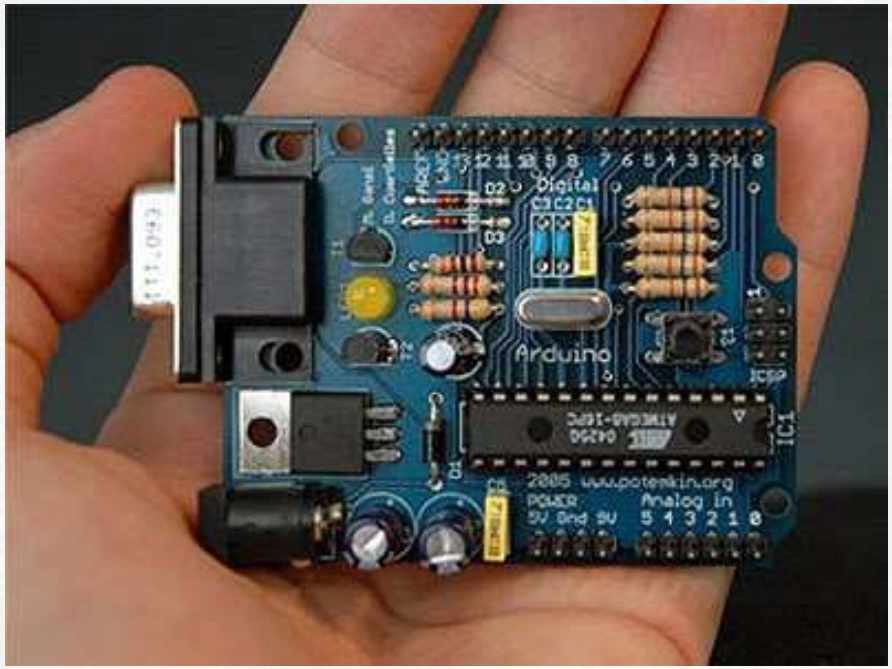
# Web of Things Environment



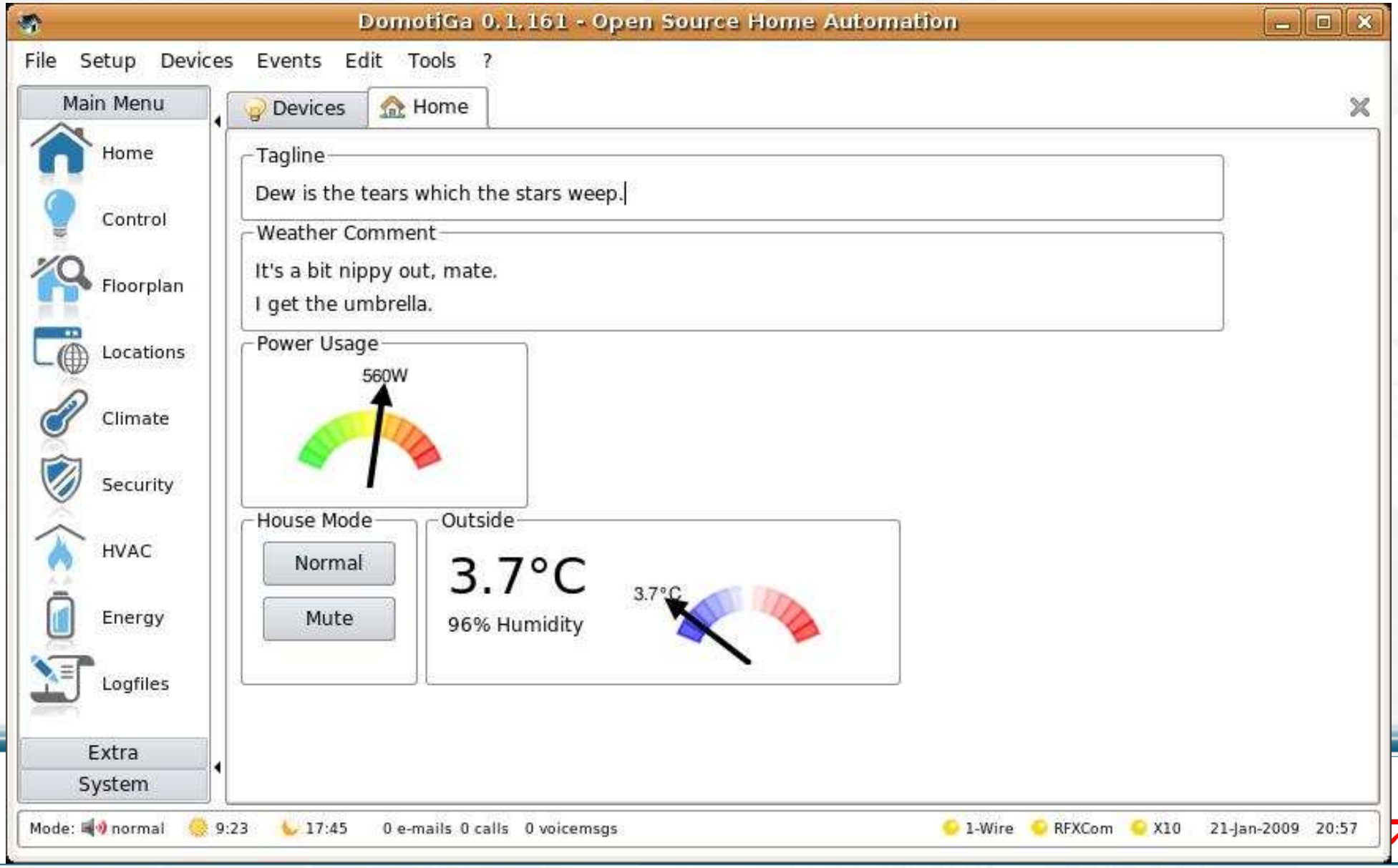

**“Apps and Services glue with Things”.**

# Solution

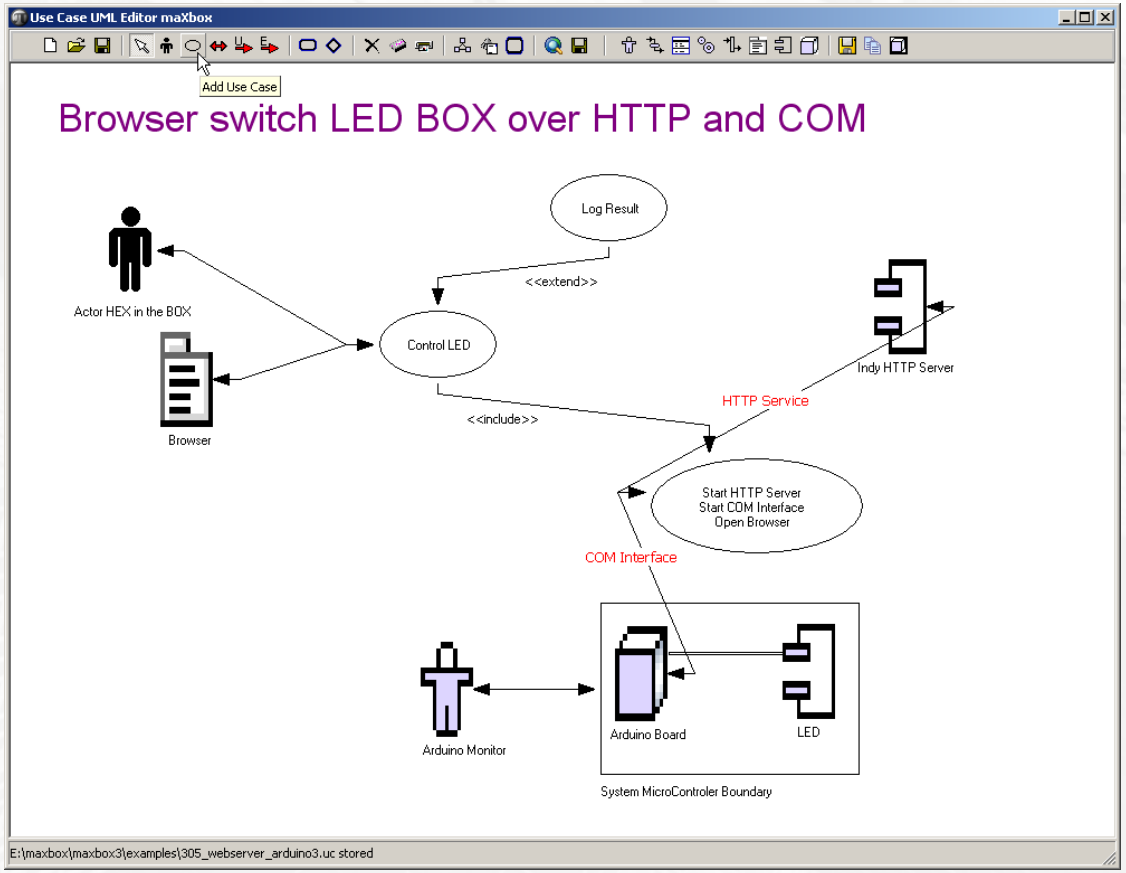
- Arduino Controller



# Solar Solution



# Solution




http://www.softwareschule.ch/examples/305_webserver_arduino3.txt

# Solution

```
76  with HTTPServer do begin
77    if Active then Free;
78    if not Active then begin
79      bindings.Clear;
80      bindings.Add;
81      bindings.items[0].Port:= APORT;
82      bindings.items[0].IP:= IPADDR; //'127.0.0.1';
83      Active:= true;
84      onCommandGet:= @HTTPServerGet;
85      PrintF('Listening HTTP on %s:%d.', [Bindings[0].IP,Bindings[0].Port]);
86    end;
```

http://www.softwareschule.ch/examples/305_webserver_arduino3.txt

maXbox

# Solution

```
54 if uppercase(localcom) = uppercase('/LED') then begin
55    cPort.WriteStr('1')
56    writeln(localcom+ ': LED on');
57    RespInfo.ContentText:= getHTMLContentString('LED is:  ON');
58 end else
59 if uppercase(localcom) = uppercase('/DEL') then begin
60   cPort.WriteStr('A');
61   writeln(localcom+ ': LED off');
62   RespInfo.ContentText:= getHTMLContentString('LED is:  OFF')
63 end;
```

361_heartbeat_wave.txt
http://en.wikipedia.org/wiki/Household_appliances

maXbox

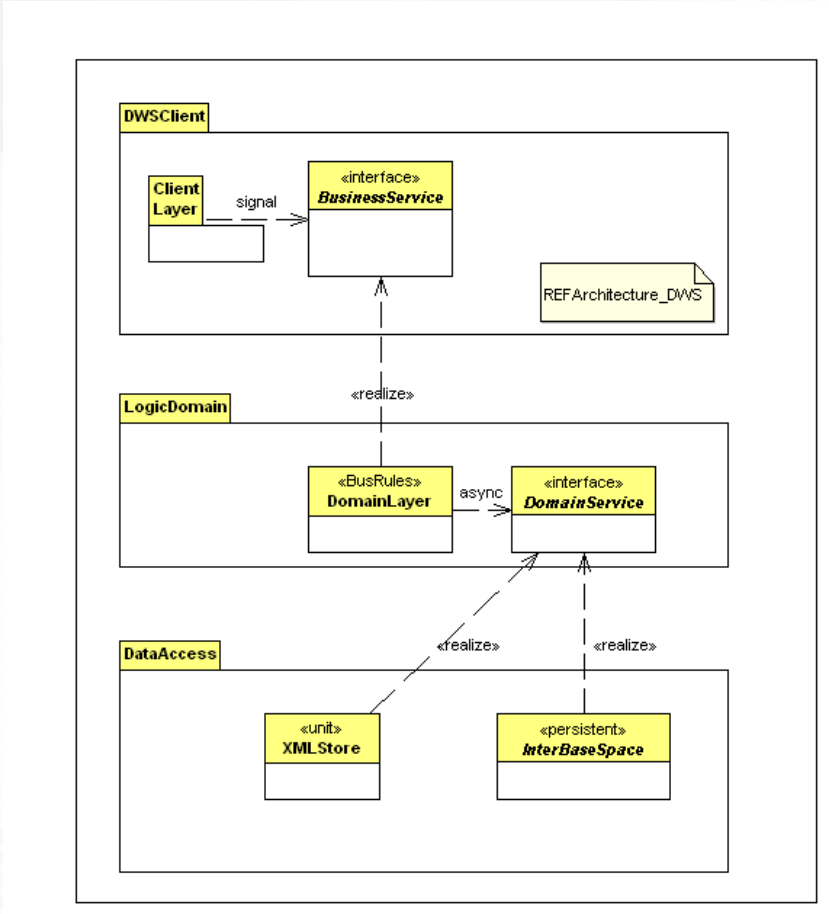
# Solution

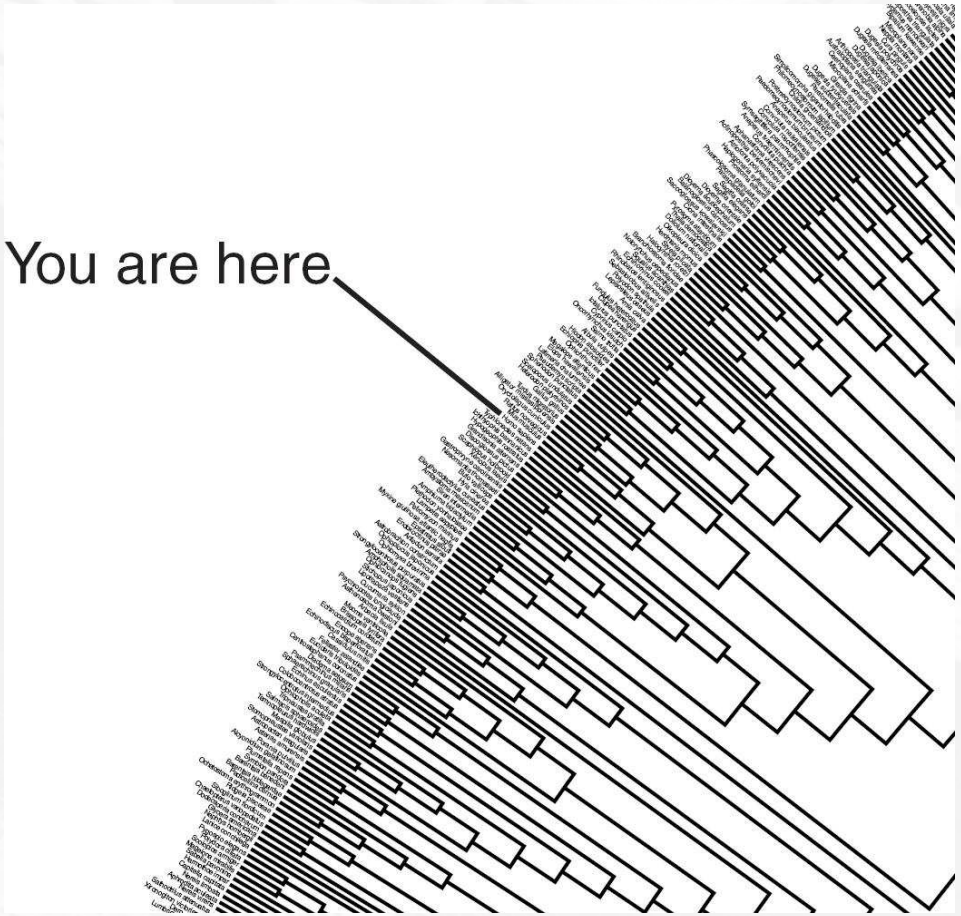
```
void setup() {
  // initialize digital pin as an output.
  pinMode(ledPin11, OUTPUT);
  Serial.begin(9600);

void loop () {
  val = Serial.read();      //read serial port
  if (val !=-1){
    if (val=='1'){
      digitalWrite(ledPin1,HIGH);
    }
    else if (val=='A'){
      digitalWrite(ledPin1,LOW);
      }
```

Tutorial: http://www.softwareschule.ch/download/maxbox_starter18.pdf

# The End is Flexibility

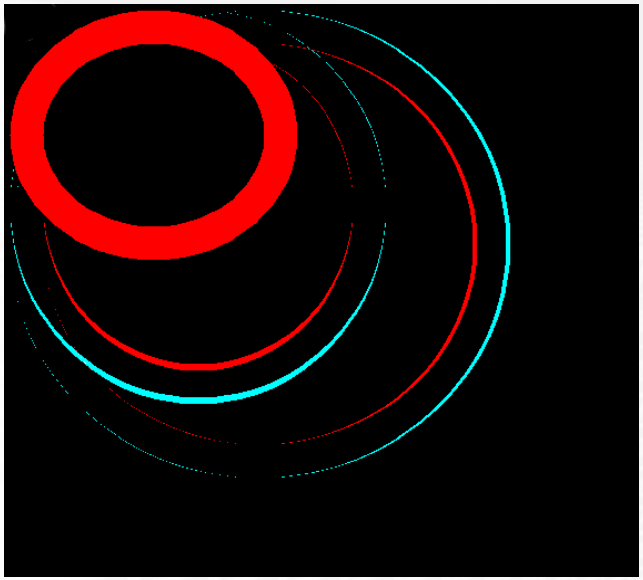
## Personal Dependency Inversion – Programming for Change



# Thanks! Links to Rights

the source is the code



http://www.softwareschule.ch/maxbox.htm
http://sourceforge.net/projects/maxbox
http://sourceforge.net/apps/mediawiki/maxbox/
http://en.wikipedia.org/wiki/Arduino
http://www.softwareschule.ch/download/webofthings2013.pdf

HTTP://SENSORLAB.IJS.SI
http://carolinafortuna.com/web-of-things-tutorial/

# Code the World

hack the earth