# The Smart Zoomable Widget[[1]](#footnote-2) concept

# (The Fallacy of the LOD)

## Why the current ZUI cannot cope with everything?

*Zoomable User Interface (ZUI): The graphical user interface (GUI) is based on a graphical environment where users can change the scale of the viewed area in to see more. detail or less*

The current mouse zoom features operates in the following steps.

1. When the mouse is scrolled by the user, the program magnifies or minifies the canvas “Uniformly”.
2. When a special magnification is achieved, a LOD (*Level of Detail*) change kicks in that abruptly changes the current area to a more detailed one.
3. Go to step 1.

Let’s have an example of visualizing the Nasdaq market index data.

## Zooming Method 1.

Nasdaq (+1.56%)

If it is magnified only (without any LOD), with magnification = 3x, we got:

Nasdaq (+1.56%)

## Zooming Method 2.

We realize that the main point of ZUI is not magnifying the letters to incredible huge, illegible size, but the main point is to insert as much data into the available space as possible. Let’s take an example, when we want to show one further detail, the actual value of the Nasdaq composite. So, we realize that a LOD change is needed.

Original:

Nasdaq (+1.56%)

Magnification = 2x

Nasdaq (+1.56%)

With the next minor user zoom, the LOD jumps in (abruptly), and the fontsize jumps to a smaller one:

Nasdaq 2,348.56 (+1.56%)

And the user can zoom further, Magnification = 3x

Nasdaq 2,348.56 (+1.56%)

## Zooming Method 3.

What I propose to change is that the transition from one LOD to the other should be smooth.

Original:

Nasdaq (+1.56%)

Magnification = 1.3x

Nasdaq (+1.56%)

Magnification = 1.35x, the new data appears in very small size, the fontsize of the already existing data doesn’t change

Nasdaq 2,348.56 (+1.56%)

Magnification = 1.8x, the new data grows

Nasdaq 2,348.56 (+1.56%)

This kind of approach would limit the fontSize (or other size) of the elements to a maximum threshold. So, if the user zooms further, the fontsize will not grow to illegible huge size. There is no point doing that.  
  
And the user can zoom further, Magnification = 3x

Nasdaq 2,348.56 (+1.56%)

However, this concept is not the distretized LOD, that is commonly used. We cannot really say any more that LOD1 contains 2 controls and LOD2 contains 3 controls, because the whole process is continuous. Let’s see why. Let’s introduce another further visualized data: the value of change. (of the Nasdaq)

Magnification = 1.35x, the new data (“Nasdaq value”) appears in small size

Nasdaq 2,348.56 (+1.56%)

Magnification = 1.6x, the new data grows, (“Nasdaq value” is bigger)

Nasdaq 2,348.56 (+1.56%)

Magnification = 1.65x, the new data grows, (“Nasdaq value” is bigger, but doesn’t achieve its full size) AND “Nasdaq change” appears in very small size

Nasdaq 2,348.56 +36.28 (+1.56%)

Magnification = 2x, the new data grows (“Nasdaq value” achieves its full size, “Nasdaq change” is still growing)

Nasdaq 2,348.56 +36.28 (+1.56%)

Magnification = 2.5x, the new data grows

Nasdaq 2,348.56 +36.28 (+1.56%)

The conclusion is that with this method there isn’t a clear distinction between LOD1, LOD2 and LOD3, because there is only one Level of Detail, and this Level of Detail contains every widget (just maybe the size of the widget is too small to be visible). So, there is only one LOD.

Every control should be programmed to be smart enough to find out their size and position. I called them Smart Zoomable Widgets. That should be programmed with *meticulous* care.

I made a demonstration program to show the concept.

1. Widget: any UI element like Label, TextBox, Image, Button or any other Control [↑](#footnote-ref-2)