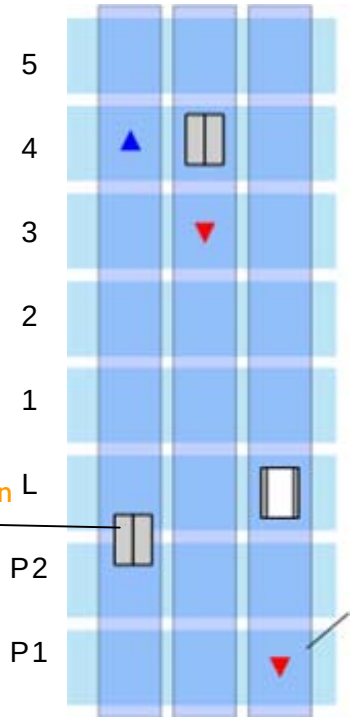
How to display floor locations

While the elevator application is running, cabin locations and destination markers must be moved to the correct locations within each shaft icon.

GUI Domain



Update cabin position

Set destination

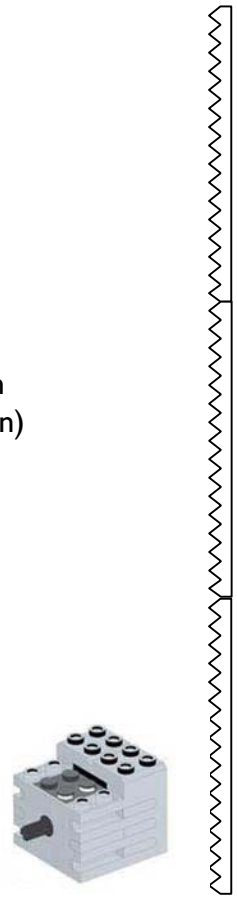
Transport Domain

We need a way to translate positions in the transport coordinate system into locations on our display coordinate system.

Place_icon
(icon_id, position)

Update_position
(load_id, position)

This technical note defines the algorithm necessary to perform this translation.



Floor Display Scaling Algorithm

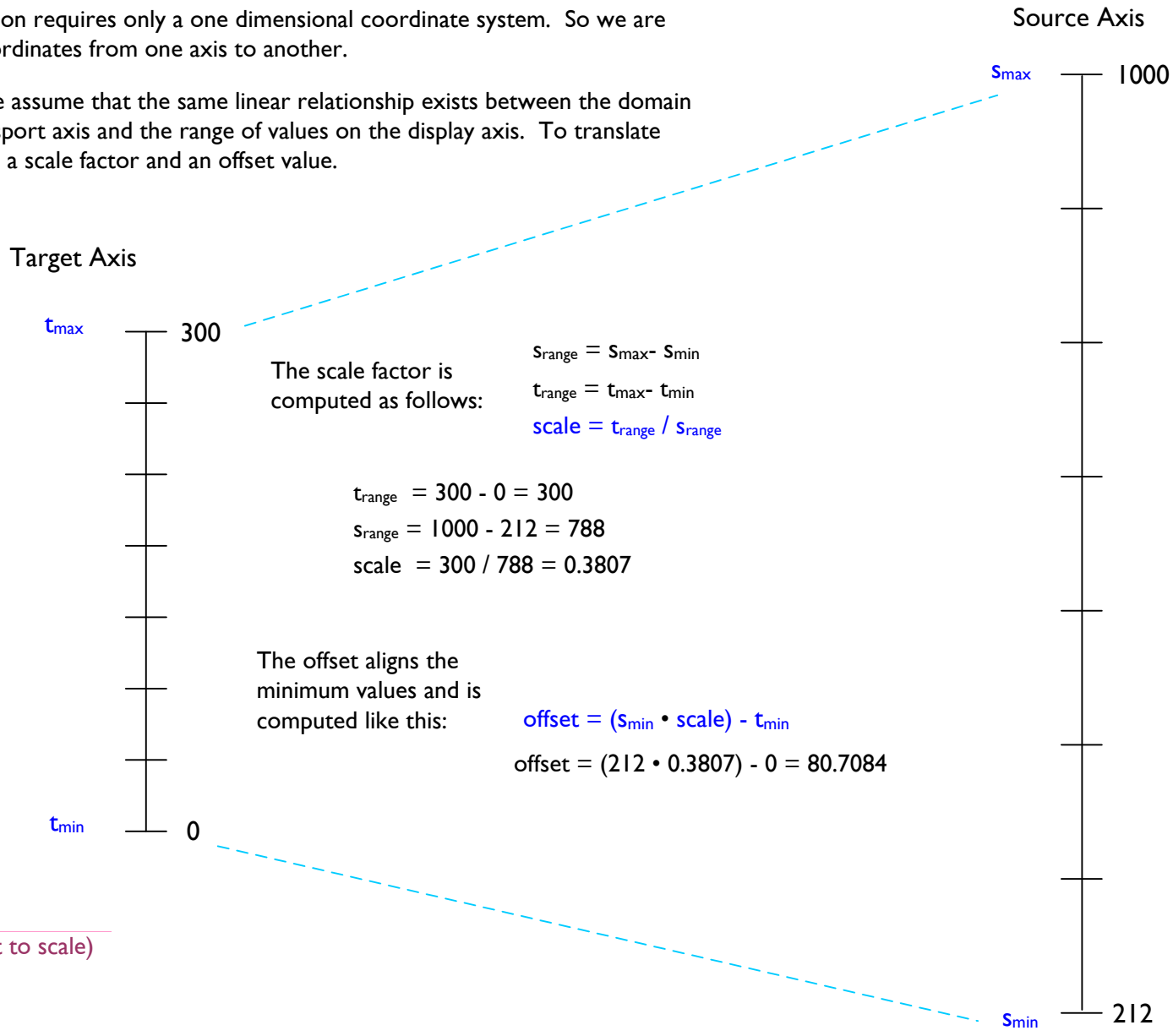
Leon Starr

mint.elev2.tn.4

The source and target axes

The elevator application requires only a one dimensional coordinate system. So we are simply translating coordinates from one axis to another.

In the simple case, we assume that the same linear relationship exists between the domain of values on the transport axis and the range of values on the display axis. To translate coordinates, we need a scale factor and an offset value.



The scale factor is computed as follows:

$s_{range} = s_{max} - s_{min}$

$t_{range} = t_{max} - t_{min}$

$scale = t_{range} / s_{range}$

$t_{range} = 300 - 0 = 300$

$s_{range} = 1000 - 212 = 788$

$scale = 300 / 788 = 0.3807$

The offset aligns the minimum values and is computed like this:

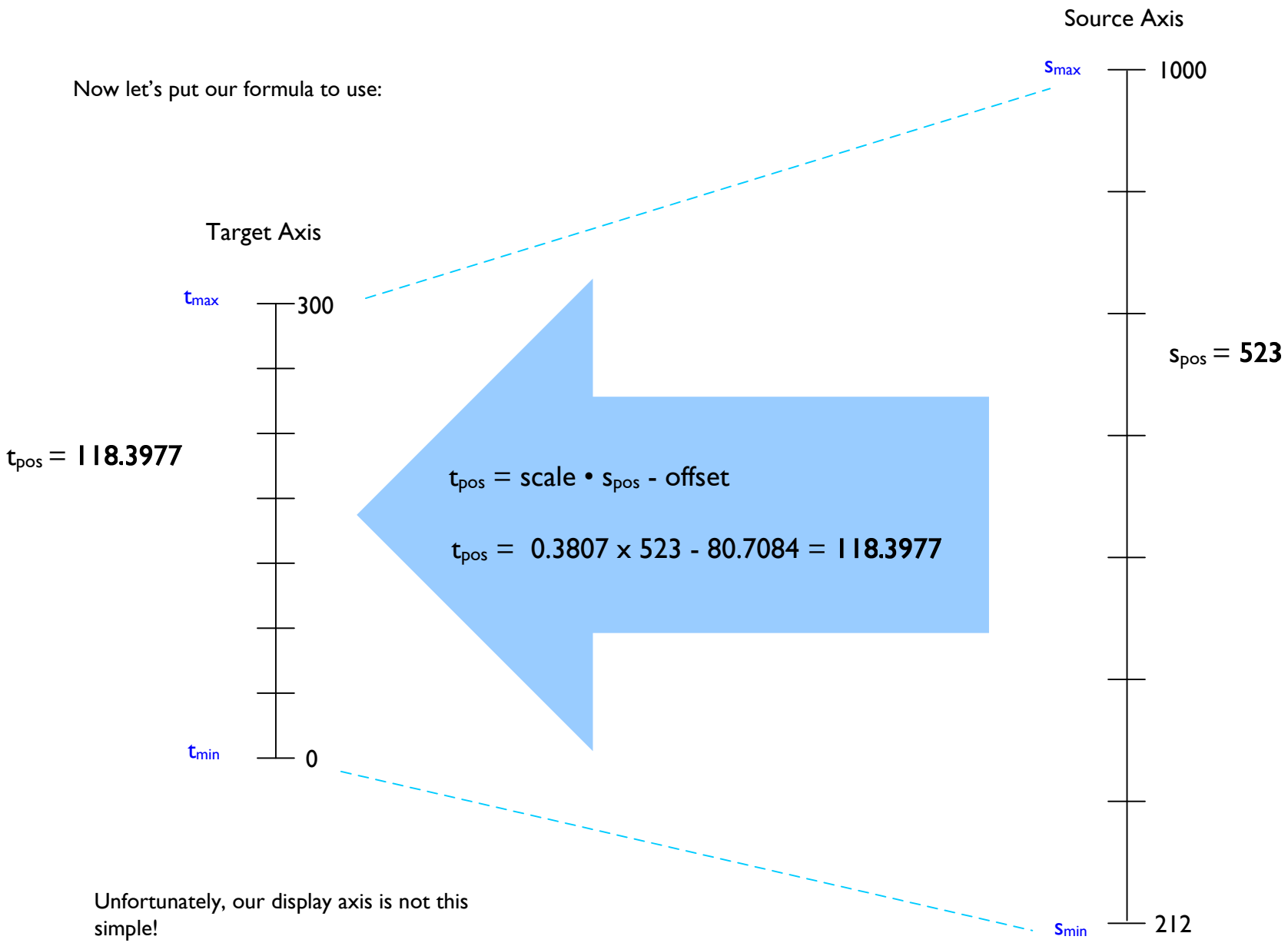
$offset = (s_{min} \bullet scale) - t_{min}$

$offset = (212 \bullet 0.3807) - 0 = 80.7084$

(Coordinate axes not to scale)

Mapping formula, simple case

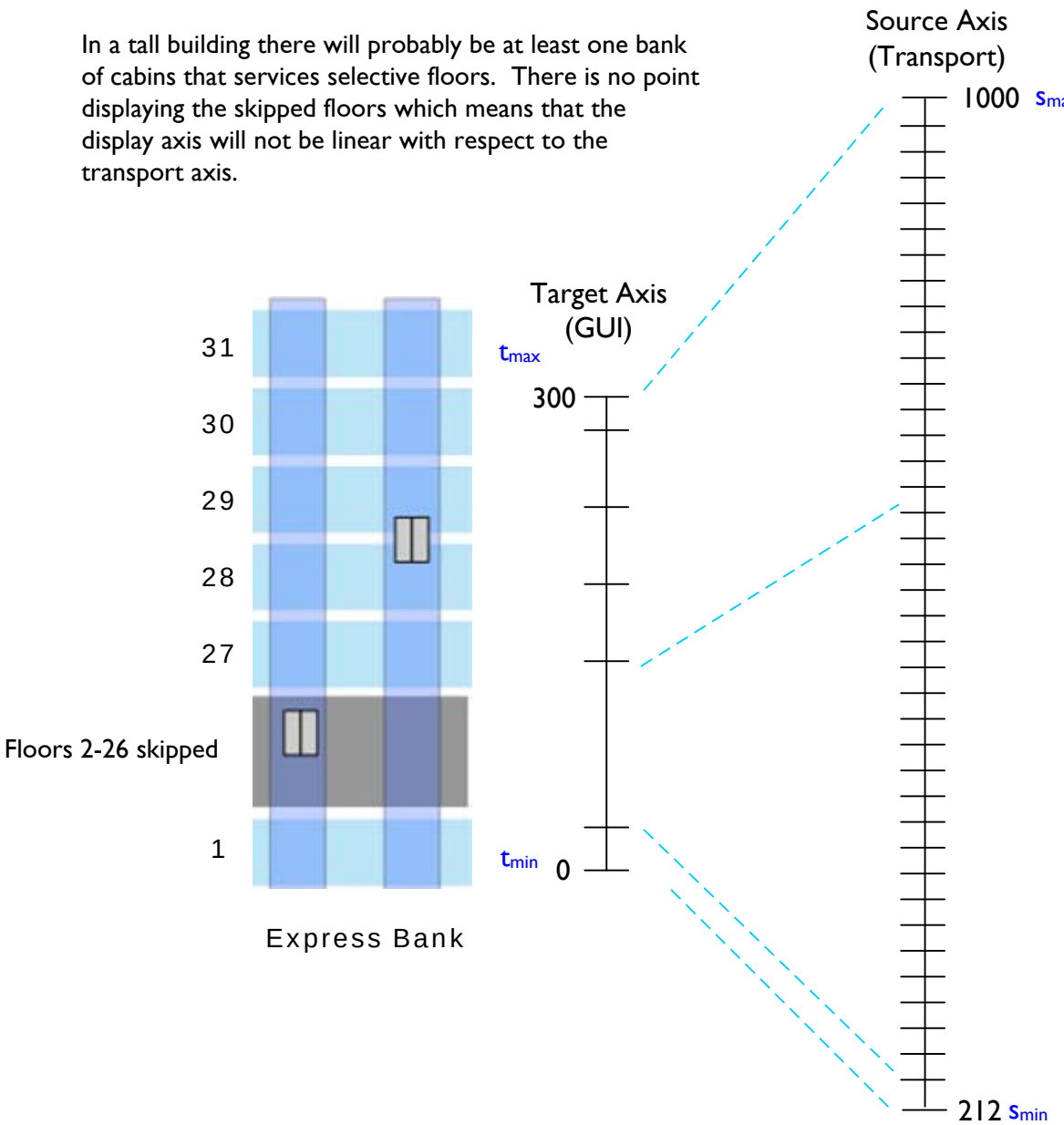
Now let's put our formula to use:



Unfortunately, our display axis is not this simple!

A more complicated mapping

In a tall building there will probably be at least one bank of cabins that services selective floors. There is no point displaying the skipped floors which means that the display axis will not be linear with respect to the transport axis.

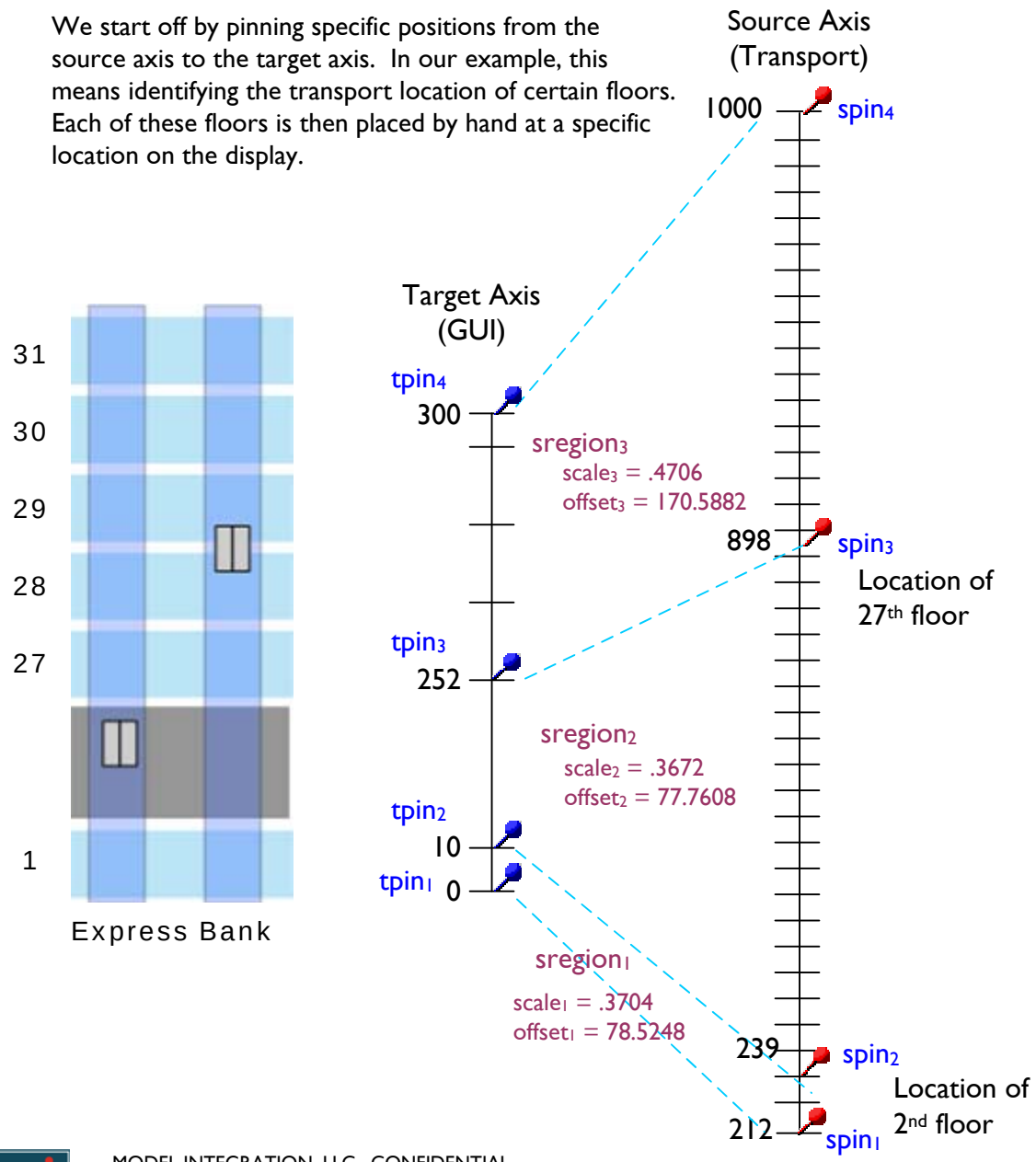


A position on the transport axis translates to a position on the display axis differently depending on location. In this example, transport positions are scaled more tightly in the region of skipped floors than in the region of serviced floors in the bank display. Therefore, if the real world elevator cabin moves at constant speed, the cabin icon will appear to move slower in the skipped-floor region of the display.

The mapping problem is easily solved if we partition the display axis into multiple scale regions. Each of these regions will have its own scale factor and offset.

Scale regions

We start off by pinning specific positions from the source axis to the target axis. In our example, this means identifying the transport location of certain floors. Each of these floors is then placed by hand at a specific location on the display.



We start off by pinning specific positions from the source axis (spins) onto selected positions on the target axis (tpins). In our example, this means identifying the location of certain floors on the transport axis. A display location is chosen for each of these floors on the display axis. Note also that the min and max values of the source axis must be pinned to the min and max values of the target axis.

A gap between each pair of adjacent spin positions $spin_n$ and $spin_{n+1}$ scales to a gap between a pair or corresponding $tpin_n$ and $tpin_{n+1}$ positions. This mapping is called a "scale region".

For each scale region we need to compute a local scale and offset value. These will be used to map source positions to target positions. The formula presented on page 2 is adapted as follows:

$sregion_n$

$$scale_n = (tpin_{n+1} - tpin_n) / (spin_{n+1} - spin_n)$$

$$offset_n = (spin_n \bullet scale_n) - tpin_n$$

For example, we can compute the scale and offset values for sregion3 as follows:

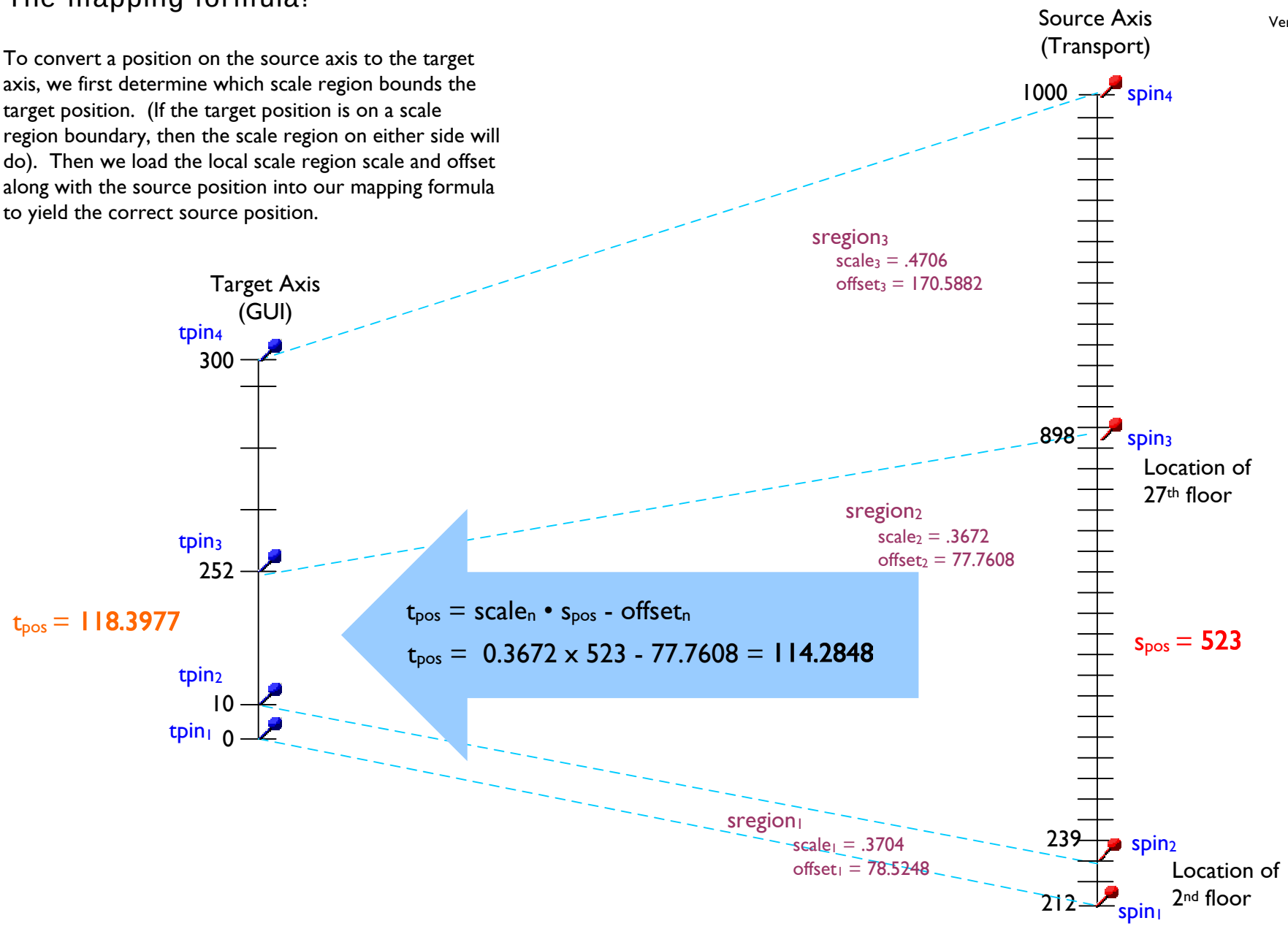
$sregion_3$

$$scale_3 = (300 - 252) / (1000 - 898) = 0.4706$$

$$offset_3 = (898 \bullet 0.4706) - 252 = 170.5882$$

The mapping formula!

To convert a position on the source axis to the target axis, we first determine which scale region bounds the target position. (If the target position is on a scale region boundary, then the scale region on either side will do). Then we load the local scale region scale and offset along with the source position into our mapping formula to yield the correct source position.



Floor Display Scaling Algorithm

Leon Starr
mint.elev2.tn.4