Compass

Revision	Author	Date	Reviewer	Approved By	Changelog	JIRA Story
0.2	Lorin Atzberger - (p)	20 Aug 2020	☐ Li, Mingzhou ☐ Adrian Tut ☑ Zhang, Mingming	☐ Li, Mingzhou ☐ Adrian Tut ☑ Zhang, Mingming	• Modified the oriented texts to also work for texts along a path. • Removed the anchors from the batch splitting by changing the positioning math. • Moved the material changes to TextMaterial::update • Move the request.info.coordinateSpace code in ViewStyle.cpp • Increment resources_version from '11.6' to '12.6'.Zhang, Mingming	~~ATLAS-1000~~ - [Kipawa K2] - Support for offroad compass CLOSED ~~ATLAS-1958~~ - Add oriented text support CLOSED
0.1	Zhang, Mingming	12 Aug 2020	☐ Li, Mingzhou ☑ Adrian Tut ☐ Razvan Oprea ☐ Wu, Chong ☐ Zhou, Jun ☐ Wu, Chong	☐ Li, Mingzhou ☑ Adrian Tut ☐ Razvan Oprea ☐ Wu, Chong ☐ Zhou, Jun ☐ Wu, Chong	Support for offroad compass and the sub-task: Add oriented text support	~~ATLAS-1000~~ - [Kipawa K2] - Support for offroad compass CLOSED ~~ATLAS-1958~~ - Add oriented text support CLOSED

Table of Contents

Introduction

The user would like to be able to see the compass pointing north when navigating offroad so that he/she is aware of what is the direction where driving, it asks to support the feature of the compass on the map.

Requirements

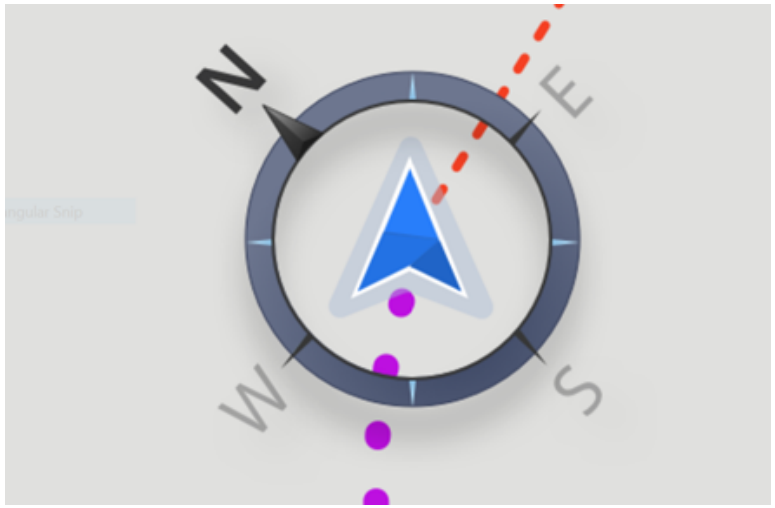
1. Overview page: https://spaces.telenav.com:8443/display/TELENAVEU/K2+2020+Integration+Scope
2. Updated Kipawa Flows: https://spaces.telenav.com:8443/display/ARP/UPDATED+-+Kipawa+Flows
3. Zeplin: https://app.zeplin.io/project/5ca7c52c65b9d234eaf55c34/screen/5cfef2c368714a15d2824204

Compass is a feature to off-road, the design by UX is as below:

Breadcrumbs and compass when the user drives off-road

Design

Draw in world coordinate, not only support in 2D, but also in 3D.



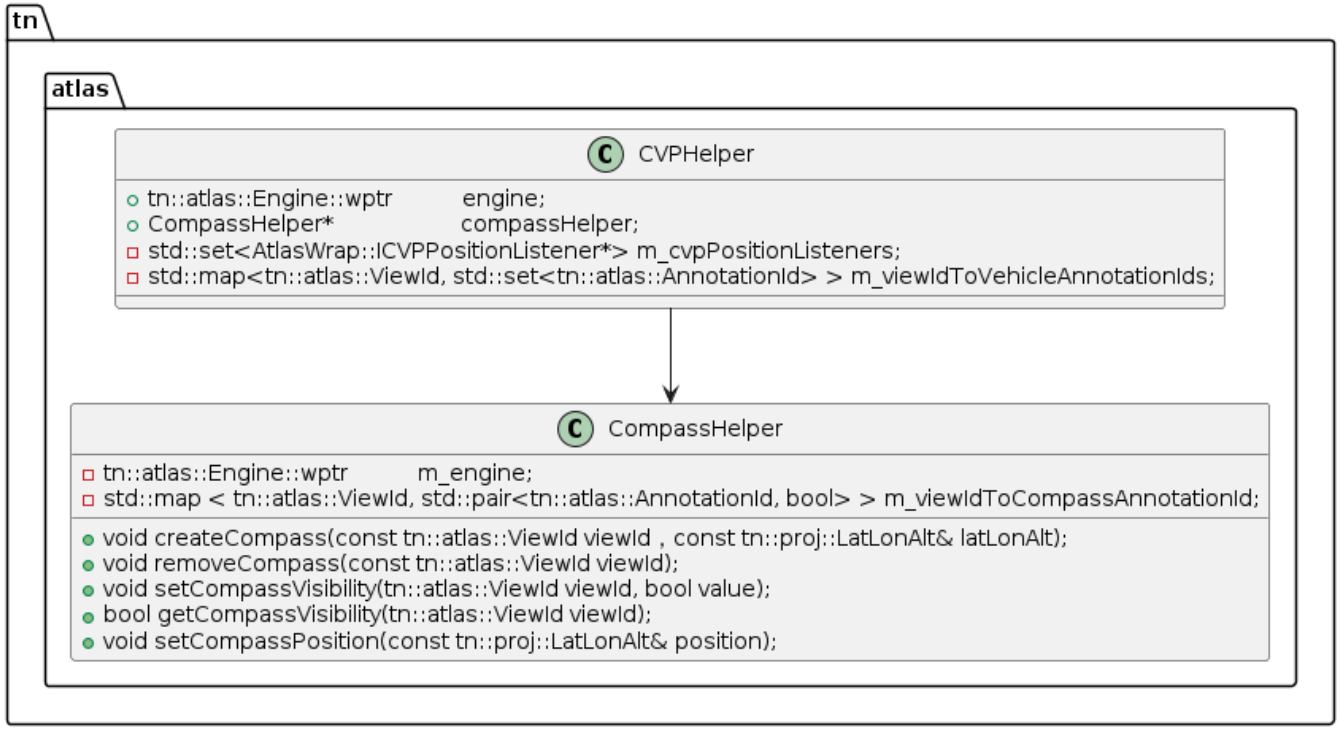
The rendering of compass follows below rules:

1. Ring always point North, regardless of the heading of the vehicle
2. Compass render size on screen remains unchanged across different zoom levels.

The "Compass" is composed of 2 parts, which can be configured individually:

- The hollow ring, configurable in size and color
- The configurable glyphs for the four directions (default is N/S/E/W), configurable in size and color

Compass



Oriented text

Shader

Add three new shaders to support ORIENTED text mode, and change only happens in the vertex shader, we will use the MVP matrix to implement the orientation.

https://bitbucket.telenav.com/projects/NAV/repos/atlas-resources/pull-requests/205/overview

```
eOrientedTextGlyph,
eOrientedTextShadow,
eOrientedTextOutline,
```

the core code is like:

pos = a_position* u_view_settings.x // position multiplied by the scale (screen to world space scale) to keep the same size when camera zoom in/out

gl_Position = pos * mvp // pos * Proj * View * Trans

Text Vertex Shader Comparison

Origin

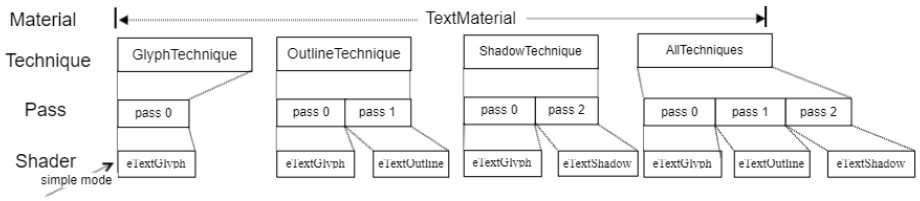
```
void main()
{
    v_uv      = a_uv;
    v_opacity = a_opacity;
#if PASS_MODE == 3
    vsOut(UNIFORM_BLOCK(ub_proj,u_proj_matrix) * vec4(a_position +
u_shadow_offset, 0.0, 1.0));
#else
    vsOut(UNIFORM_BLOCK(ub_proj,u_proj_matrix) * vec4(a_position, 0.0, 1.0));
#endif
}
```

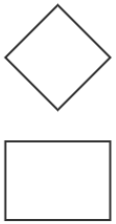
New

```
void main()
{
    v_uv      = a_uv;
    v_opacity = a_opacity;

#ifdef USE_ORIENTATION
        highp vec2 pos = a_position;
        pos *= UNIFORM_BLOCK(ub_vs,u_view_settings).x;
        pos.y *= -1.0;

#if PASS_MODE == 3
    vsOut(UNIFORM_BLOCK(ub_mvp,u_mvp_matrix) * vec4(pos + u_shadow_offset,
0.0, 1.0));
#else
        vsOut(UNIFORM_BLOCK(ub_mvp,u_mvp_matrix) * vec4(pos, 0.0, 1.0));
#endif        // PASS_MODE

#else

#if PASS_MODE == 3
    vsOut(UNIFORM_BLOCK(ub_proj,u_proj_matrix) * vec4(a_position +
u_shadow_offset, 0.0, 1.0));
#else
    vsOut(UNIFORM_BLOCK(ub_proj,u_proj_matrix) * vec4(a_position, 0.0, 1.0));
#endif // PASS_MODE

#endif  // USE_ORIENTATION
}
```

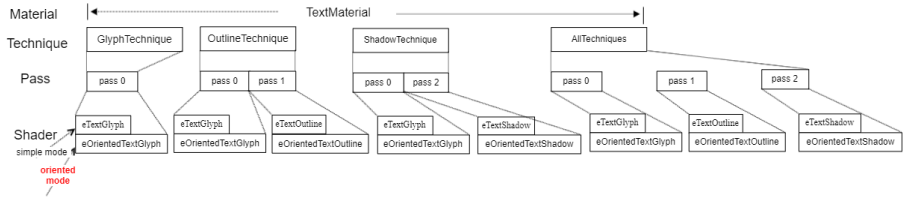
Material

Origin TextMaterial, each pass only supports one text shader, a simple mode that renders text in screen space.





Now TextMaterial, each pass can support two text shaders, except the existed simple mode, add an oriented mode that renders text in world space.



when text.placement is "oriented", text render will set the ActiveShader as the oriented one in each pass of TextMaterial's ActiveTechnique.

Limitations and Constraints

NOTE:

Add oriented text support in the world space(3D), now used in the POIFeature.

For LineFeature, it already has a strategy to support the oriented text(road name), which is that computing path angle of text for rotation it,
it always draws in screen space.

Modified the oriented texts to also work for texts along a path. Lorin Atzberger - (p)20 Aug 2020

User Interfaces

Public API

Interface Signature	Usage	Sample Code

`virtual bool ITnMapEngine::SetBool(ViewId viewId, eParameterBool param, bool value);`	Turn on/off compass display dynamically
	
	The style is according to the compass configuration in **newstyle.tss**
		`...`
		`// Turn compass ON`
		`m_engine->SetBool(m_viewId, ITnMapEngine::eParameterBool_ShowCompass, true);`
		`...`
		`// Turn compass OFF`
		`m_engine->SetBool(m_viewId, ITnMapEngine::eParameterBool_ShowCompass, false);`

`virtual bool GetBool(ViewId viewId, eParameterBool param, bool& value) const;`	Get a boolean of compass status(ON/OFF)	`// get compass status(ON/OFF)`
		
		`bool isShowCompass;`
		
		`m_engine->GetBool(m_viewId, ITnMapEngine::eParameterBool_ShowCompass, isShowCompass);`

Configuration

Add a new annotation named "compass" in newstyle.tss

```
layer_order

{

...
breadcrumb,
compass,
all-models,
smart-bubble,
cvp,

...

}

.....
```

```
layer<annotation> compass[annotation-data="compass"]
{
collision-enabled: disabled;
icon-image: "compass.png";
icon-image: stepped(time-of-day, [0:"compass.png", 1:"compass_night.png"]);
icon-placement: "oriented";
icon-layer-type: "3DImportant";
icon-size: animated(zoom, [14: 60pt, 18: 60pt]);
text-font: @text_regular;
text-placement: "oriented";
text-layer-type: "3DImportant";
text-layer-type: "2DImportant";
text-valignment: "center";
text-opacity: 1.0;
text-color: #000000;
[annotation-data="compass"]
{
text:"N";
text-size: 18;
text-color: animated(time-of-day, [0:#000000, 1:#ffffff]);
text-position-offset: [0,40];
}
[annotation-data="compass"]
{
text: "W";
text-size: 13;
text-opacity: 0.7;
text-color: animated(time-of-day, [0:#000000, 1:#ffffff]);
text-position-offset: [-35,0];
}
[annotation-data="compass"]
{
text: "E";
text-size: 13;
text-opacity: 0.7;
text-color: animated(time-of-day, [0:#000000, 1:#ffffff]);
text-position-offset: [35,0];
}
[annotation-data="compass"]
{
text: "S";
text-size: 13;
text-opacity: 0.7;
text-color: animated(time-of-day, [0:#000000, 1:#ffffff]);
text-position-offset: [0,-35];
}
};

....
```

Change Impact

Performance and Memory Impact

the impact is low, there is only one compass texture + 4 direction glyphs("WSNE") , the vertices number is 4(compass)+ 4*4("WSNE")

Backward Compatibility Impact

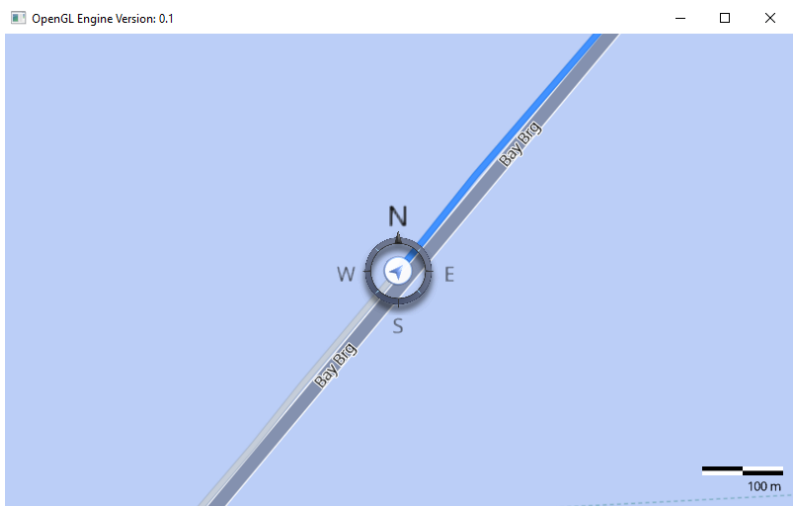
no impact, it is a new feature in ATLAS.

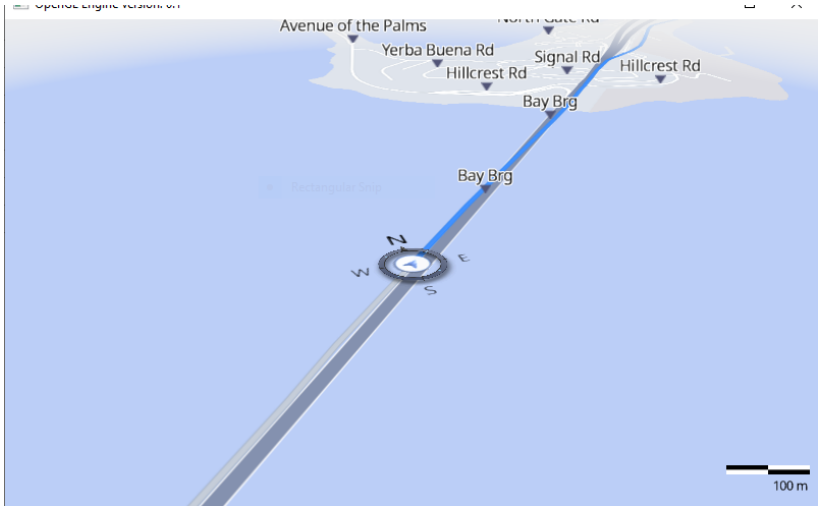
Increment resources_version from '11.6' to '12.6'.

Examples

use the default configuration

2D **3D**

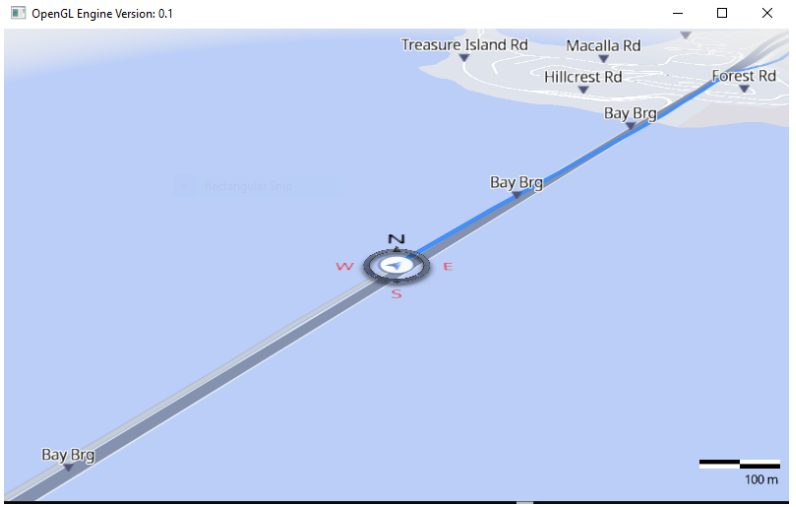




use the different color

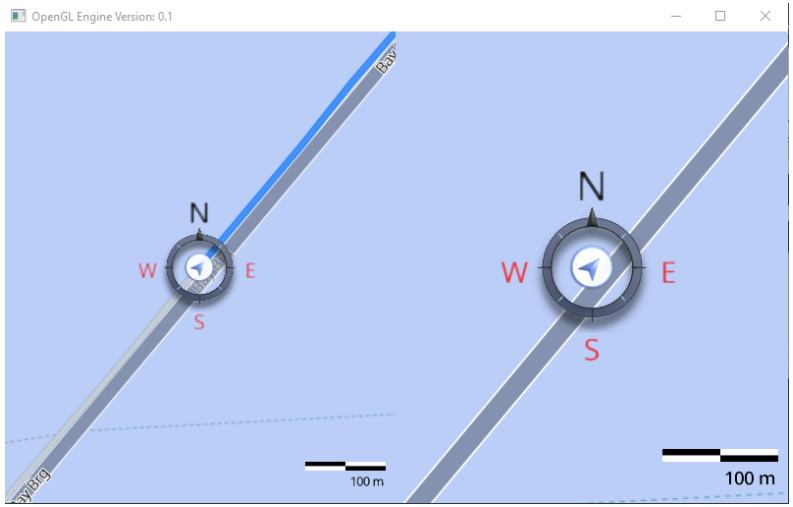
```
[annotation-data="compass"]
{
text:"N";
text-size: 18;
text-position-offset: [0,40];
}
[annotation-data="compass"]
{
text: "W";
text-size: 13;
text-opacity: 0.7;

text-color:#ff0000;
text-position-offset: [-35,0];
}
[annotation-data="compass"]
{
text: "E";
text-size: 13;
text-opacity: 0.7;
text-color:#ff0000;
text-position-offset: [35,0];

}
[annotation-data="compass"]
{
text: "S";
text-size: 13;
text-opacity: 0.7;
text-color:#ff0000;

text-position-offset: [0,-35];
}
```

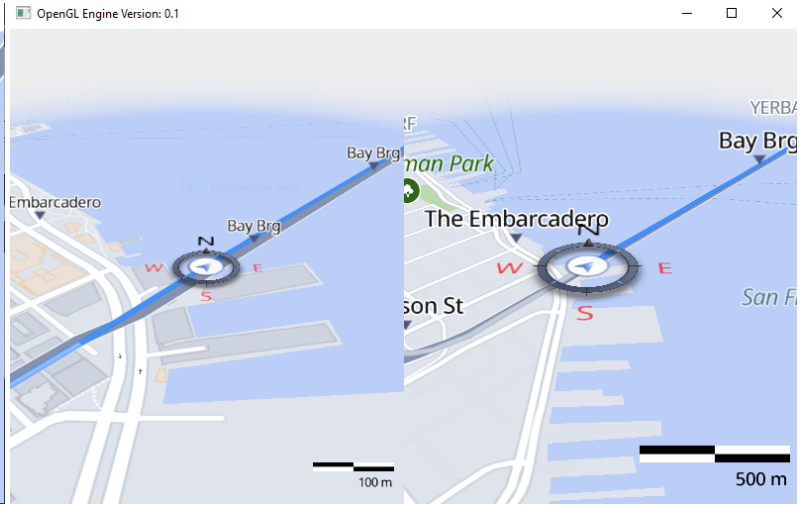
2D **3D**



show compass on the multi-views

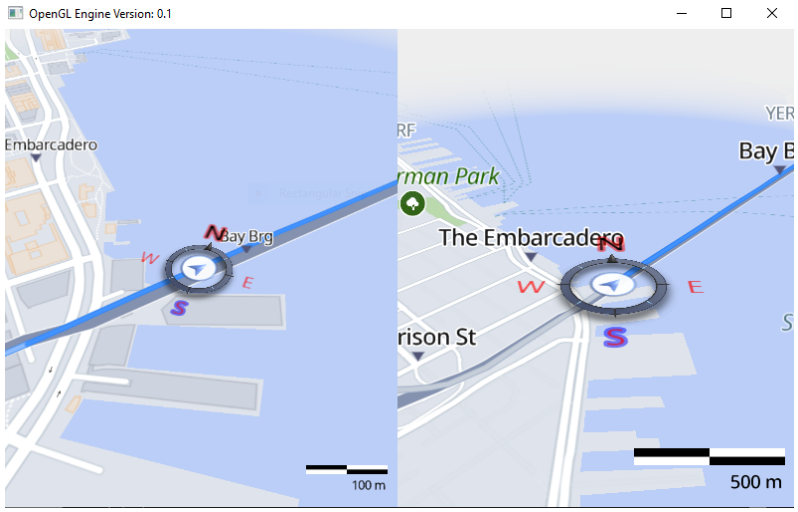
2D **3D**



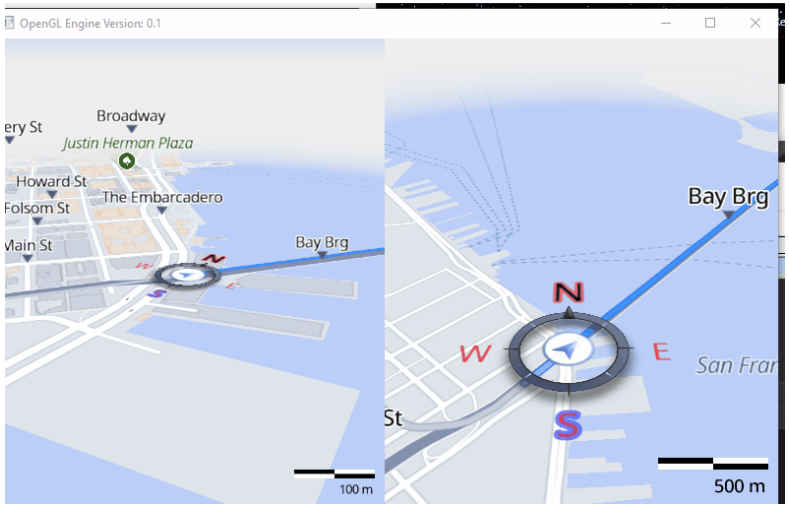


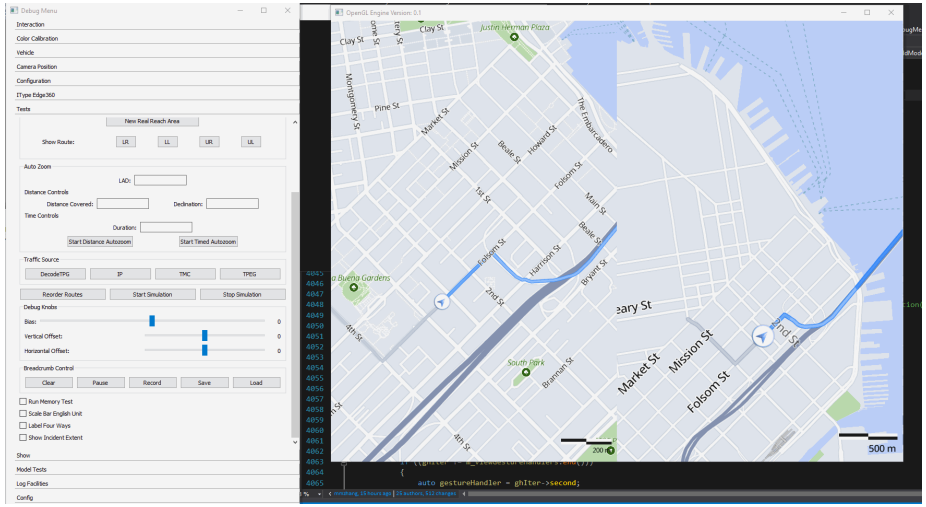
3D outline text

```
[annotation-data="compass"]
{
text:"N";
text-size: 18;
text-color: animated(time-of-day, [0:#000000, 1:#ffffff]);
text-outline-width: 2;
text-outline-color: animated(time-of-day, [0:#ff0000, 1:#161616]);
text-outline-opacity: 0.5;
text-position-offset: [0,40];
}
```



The video:


OpenGL Engine Version: 0.1
Broadway
Justin Herman Plaza
Howard St
Folsom St
The Embarcadero
Bay Brg
100 m
N
W
E
S
500 m


Debug Menu
Interaction
Color Calibration
Vehicle
Camera Position
Configuration
IType Edge360
Tests
Auto Zoom
Distance Controls
Time Controls
Traffic Source
Debug Knobs
Breadcrumb Control
Run Memory Test
Scale Bar English Unit
Label Four Ways
Show Incident Extent
Show
Model Tests
Log Facilities
Config
OpenGL Engine Version: 0.1
Justin Herman Plaza
Market St
Mission St
Folsom St
500 m