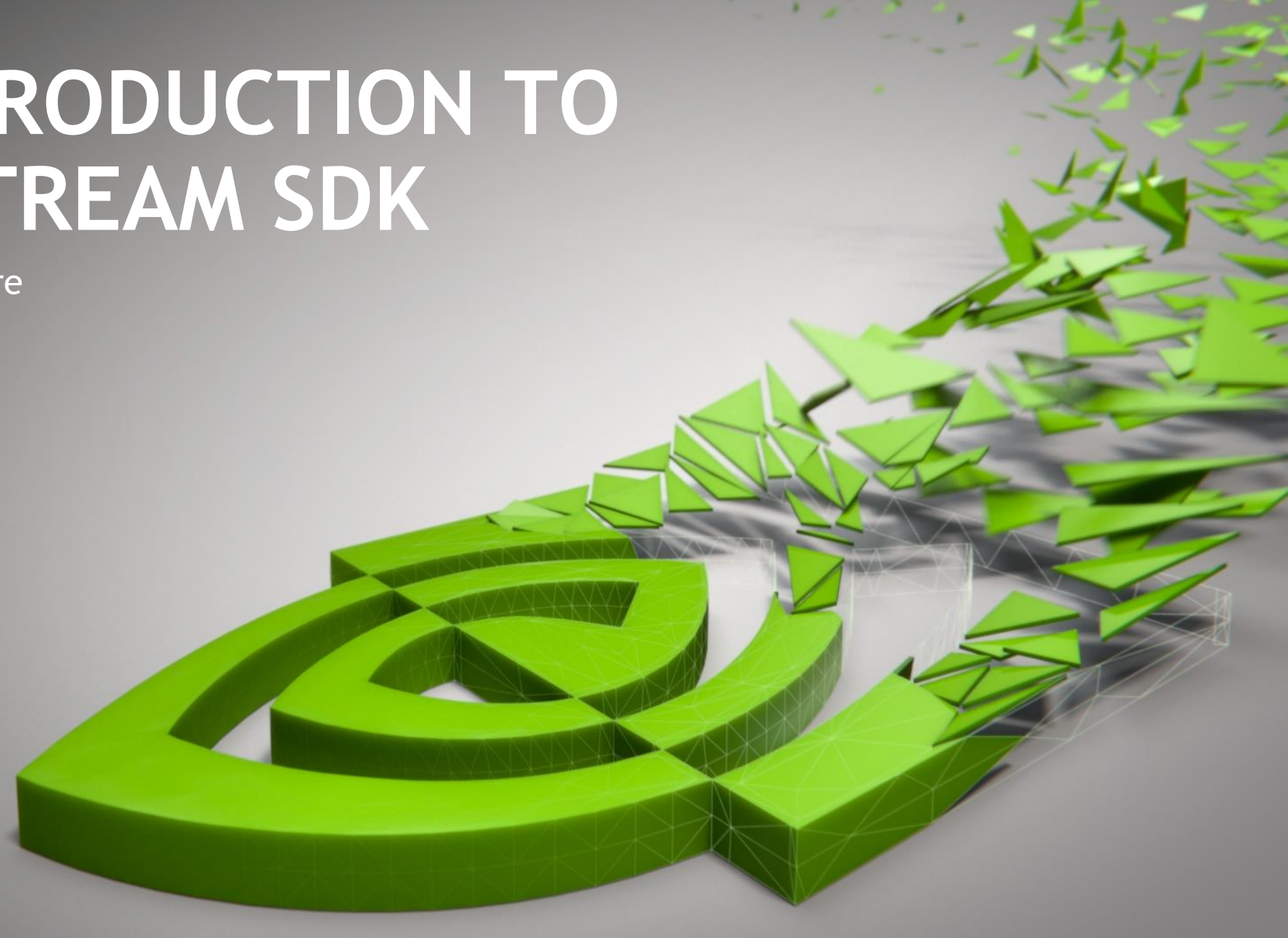
AN INTRODUCTION TO DEEPSTREAM SDK

Kaustubh Purandare

March 2018

NVIDIA



AGENDA

- Introduction to DeepStream SDK
- DeepStream SDK Basic Building Blocks
- Setup & Installation
- Application Examples
- Performance Analysis
- 3rd Party Integration
- Q&A

WHAT IS DEEPSTREAM SDK

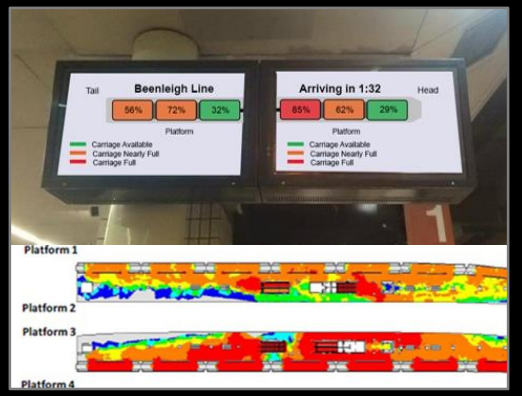
1) NVIDIA DeepStream simplifies the development of scalable intelligent video analytics (IVA) applications

2) Developers can now use this to quickly build new applications to transform video into valuable insight.

3) Applications for the DeepStream SDK include image classification, scene understanding, video categorization, content filtering etc..

DEEPSTREAM SDK FOR INTELLIGENT VIDEO ANALYTICS

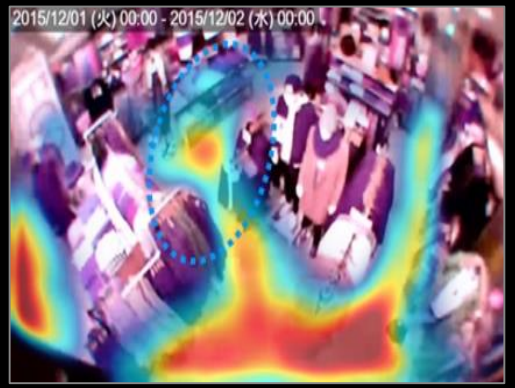
Access control



Public Transit

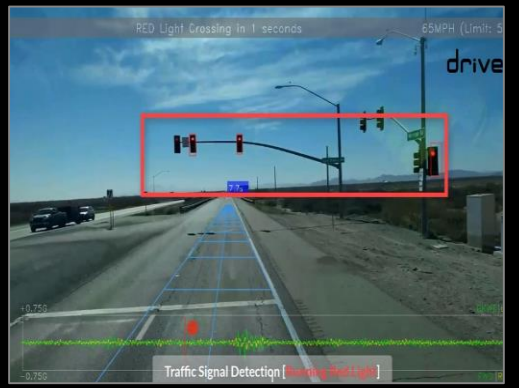
Parking Management

Traffic Engineering



Retail Analytics

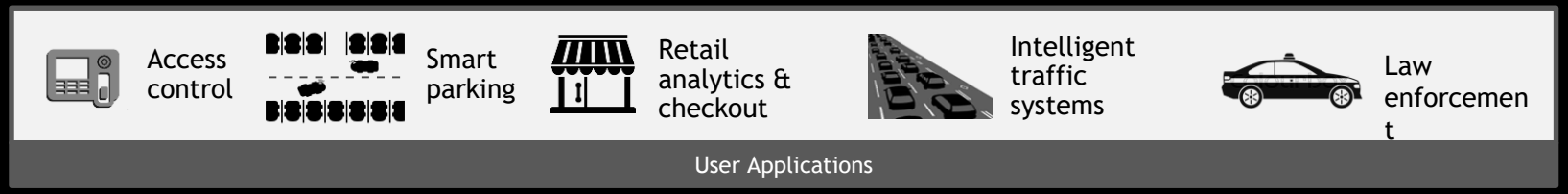
Securing Critical Infrastructure



In-Vehicle Analytics

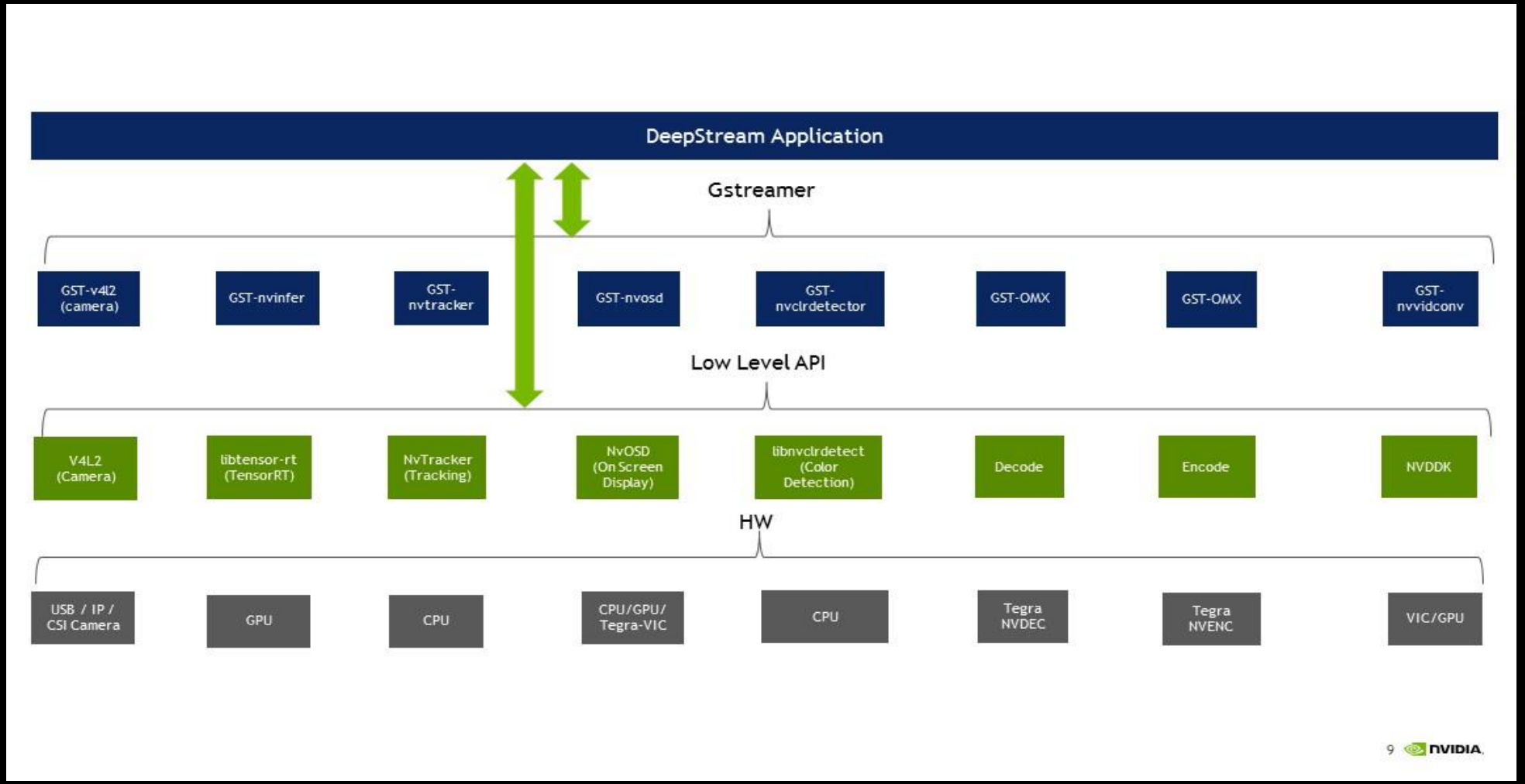
Law Enforcement

DEEPSTREAM SDK



* Early release and limited availability.

DEEPSTREAM SDK BUILDING BLOCKS



SETUP & INSTALLATION

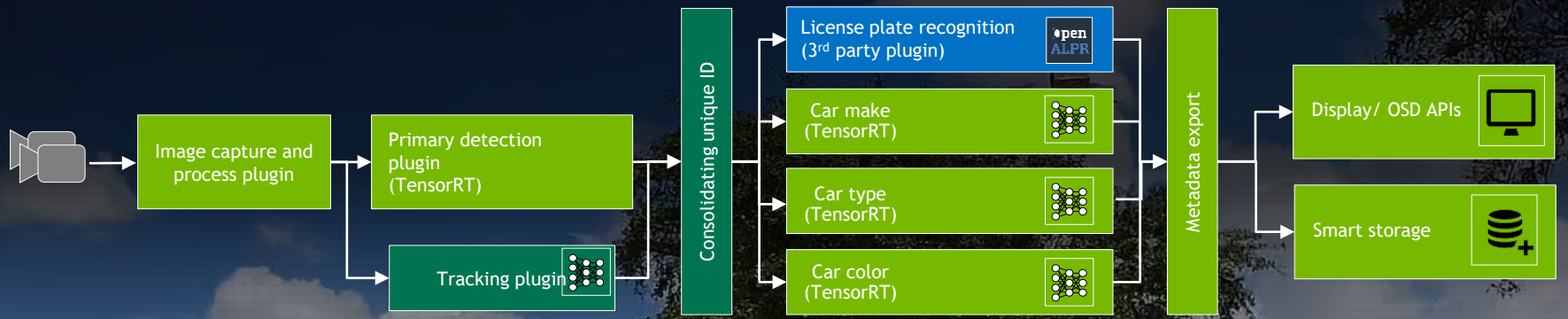
A] Jetson

1) Latest JetPack https://developer.nvidia.com/embedded/jetpack
2) DeepStream on Tegra SDK https://developer.nvidia.com/deepstream-jetson
3) Ubuntu 16.04 64-bit operating system (host)
4) Jetson TX1 / TX2 Development Platform

B] Running the Sample Application

```
nvgstiva-app -c <HOME_dir>/configs/<Config.txt> \
-i /home/nvidia/<path_to_input_stream>
```

SAMPLE APPLICATION



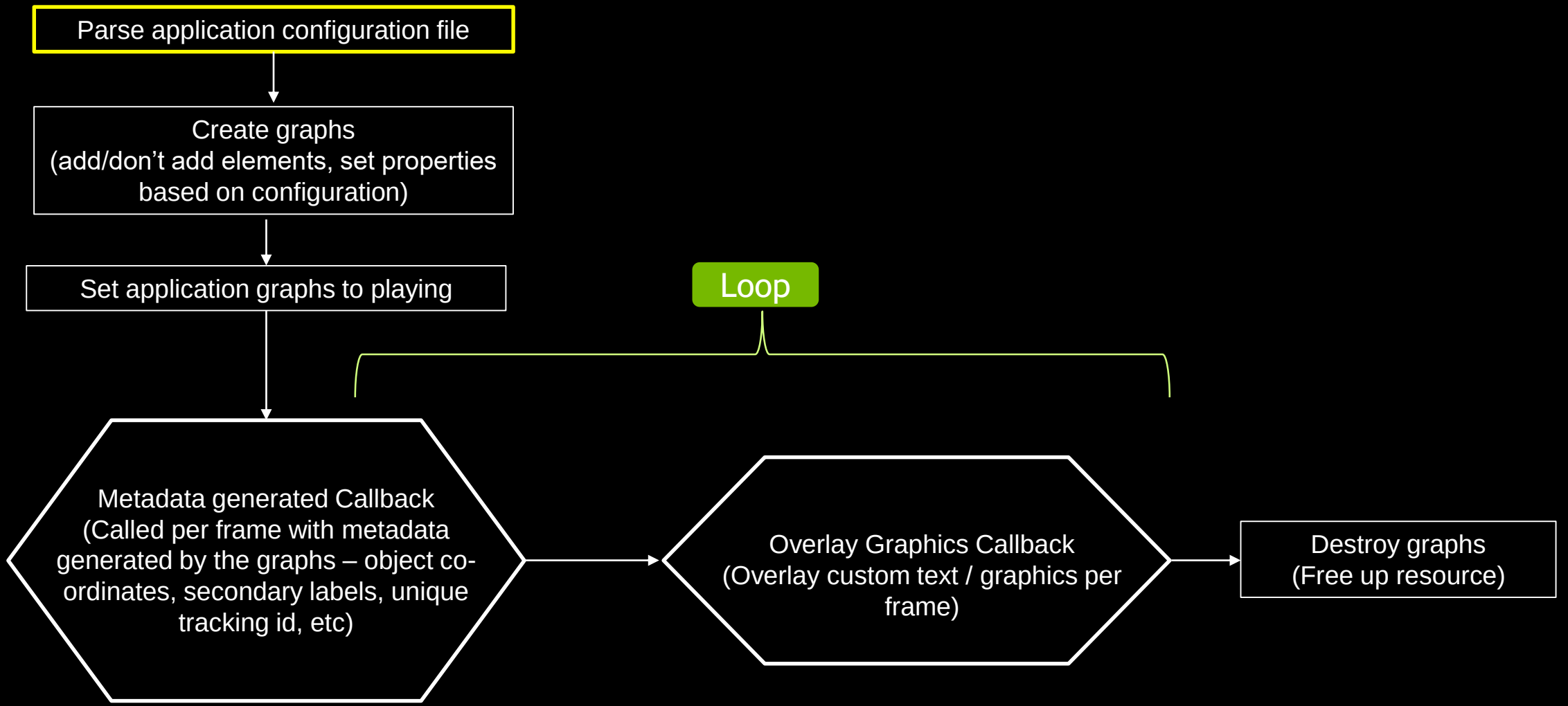
Vehicle 200 Vehicle 191 Vehicle Vehicle 215 Vehicle 2 suv black Mercedes DFGP309 Vehicle 183

Situational awareness -

- Identify cars, pedestrians, and two-wheelers
- Classify Make (Mercedes, BMW, Audi, …)
- Classify Type (SUV, Sedan, Truck, …)
- Identify Color (Black, Blue, …)
- Read license plate


Vehicle 1
Vehicle 2
Vehicle 3
Vehicle 4
Vehicle 5
Vehicle 6
17/10/05
11:53:29

APPLICATION FLOW

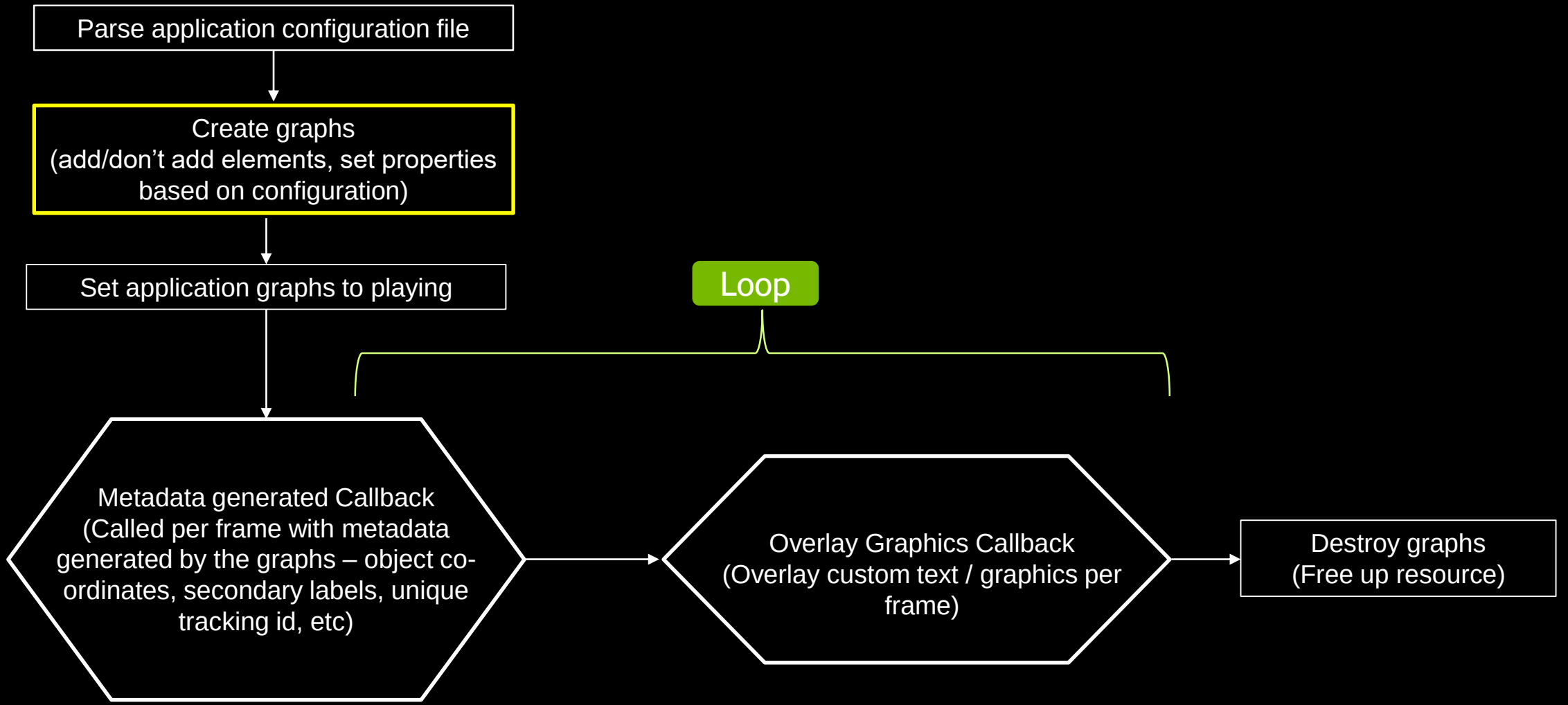


CONFIGURATION FILE

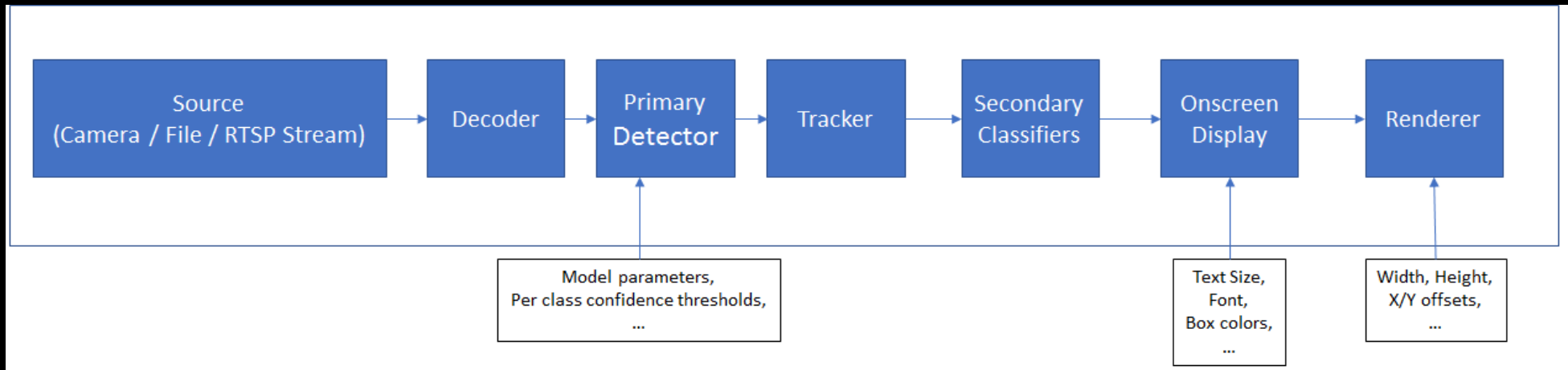
```
[application]
enable-perf-measurement=1
tracker-id=1
roi-marking=0
debug-mode=0
:
[source0]
enable=1
#Type - 1=CameraCSi 2=CameraV4L2 3=URI
uri=file:///home/ubuntu/00023.MTS
:
[sink0]
enable=1
#Type - 1=FakeSink 2=OverlaySink 3=EglSink 4=XvImageSink 5=File
:
[osd]
enable=1
osd-mode=2
border-width=2
text-size=15
:
[primary-gie]
enable=1
model-file=file:////home/ubuntu/Model/resnet/ResNet_*
:
```

- Organized as groups and key-value pairs
- Groups for graphs stages
- Key-value pair to configure each stage
- Very high granularity for configuring each stage

APPLICATION FLOW

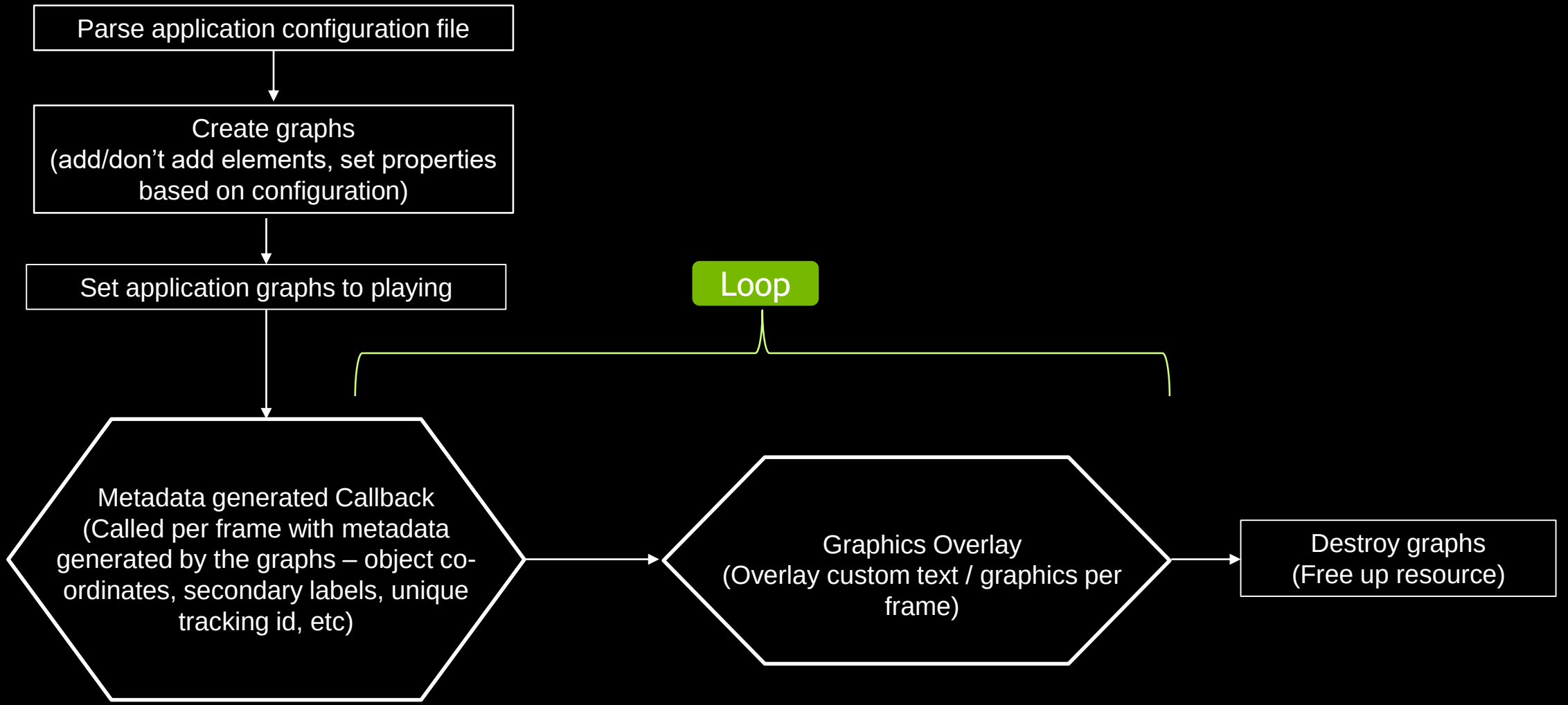


GRAPHS CREATED



- App adds elements if required based on the configuration
- Individual elements / stages are configured
- Links the elements
- Gstreamer framework performs caps negotiation, buffer allocations/deallocations transparent to the application
- Gstreamer pipeline and elements take care of zero buffer copies, buffer mangement

APPLICATION FLOW



CREATING SIMPLE GRAPHSS

1) Gstreamer provides powerful tool like `gst-launch` to create trial / experimental graphs as per use cases.

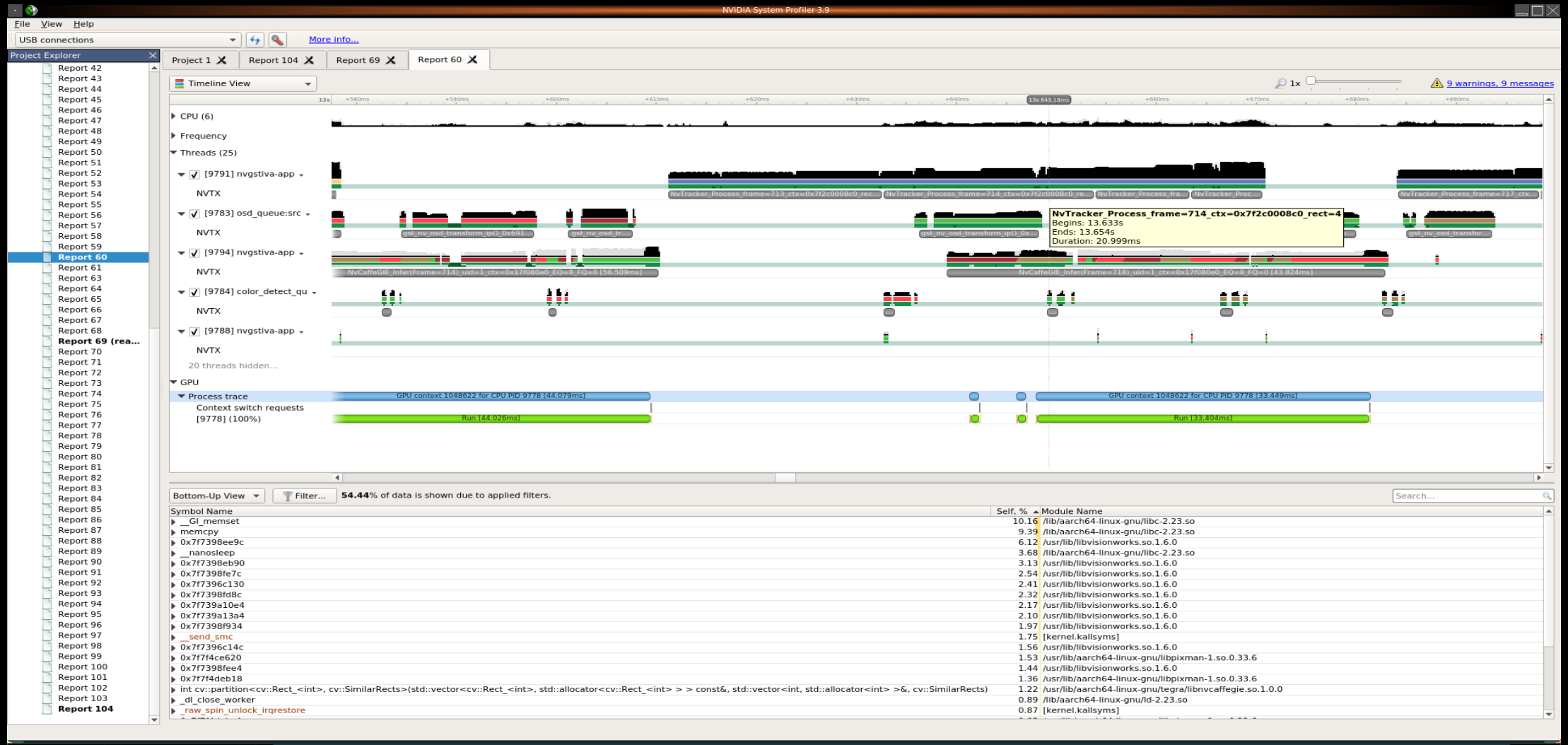
2) File stream with Primary object detection and OnScreen Display

   - `gst-launch-1.0 uridecodebin uri=file:///home/nvidia/video.mp4 ! nvinfer <primary-infer-properties> ! queue ! nvosd <osd-properties> ! nveglglessink`

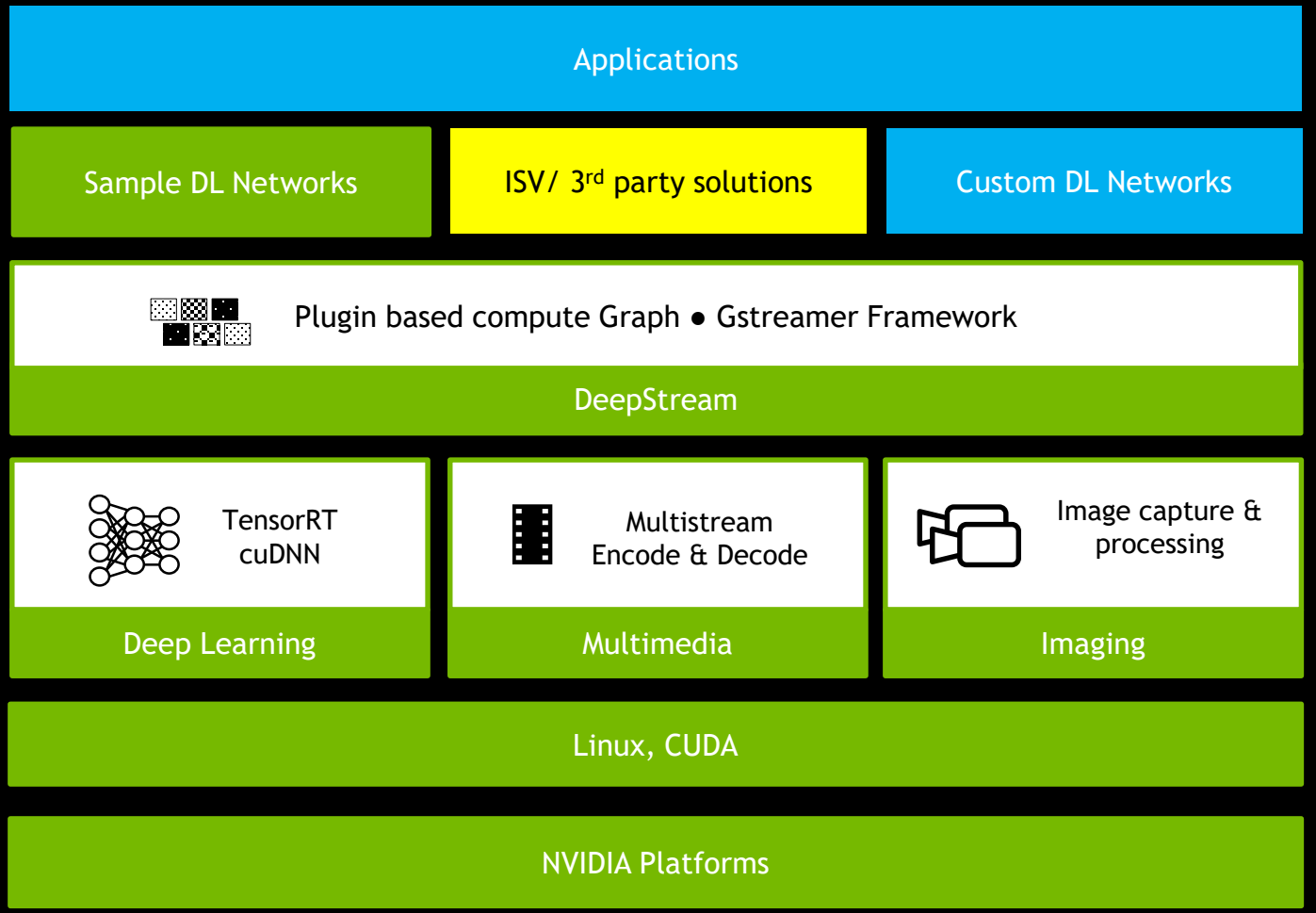
3) RTSP stream with primary object detection + tracking + secondary classification labels + OnScreen Display

   - `gst-launch-1.0 uridecodebin uri=rtsp://10.24.1.1/video0 ! nvinfer <primary-infer-properties> ! queue ! nvtracker ! queue ! nvinfer <secondary-infer-properties> ! queue ! nvosd <osd-properties> ! nveglglessink`

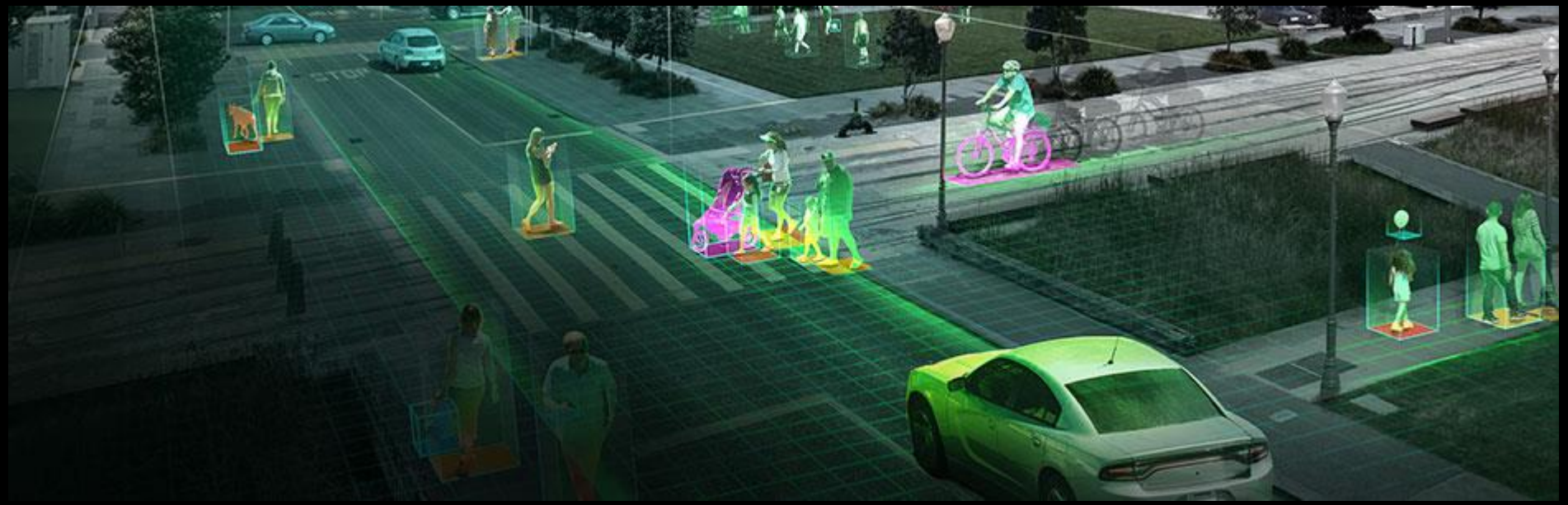
PERFORMANCE ANALYSIS



APPLICATION SOFTWARE STACK



START DEVELOPING WITH DEEPSTREAM



DeepStream Early Access program . Explore Metropolis . Intelligent Video Analytics Forums

THANK YOU