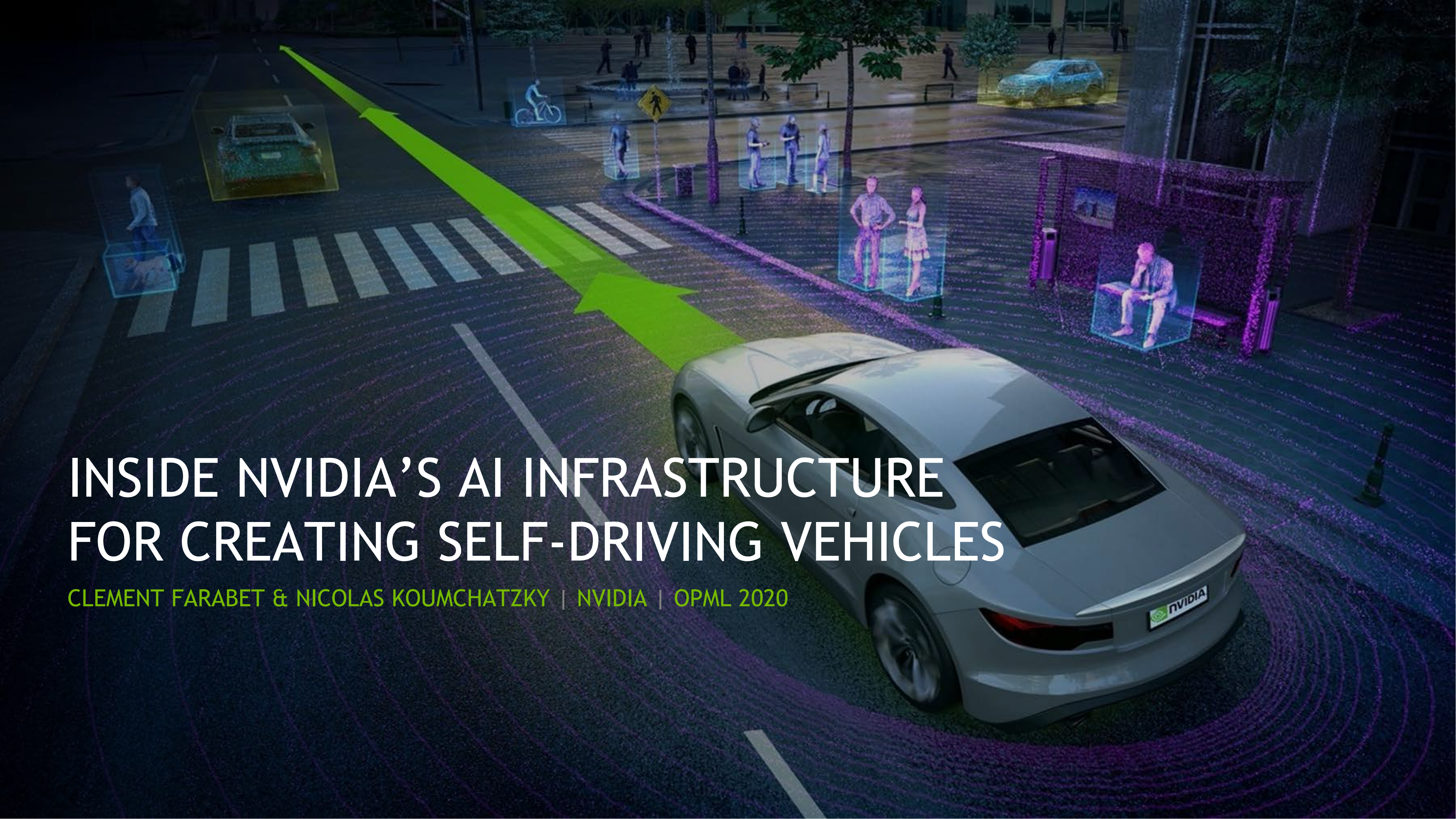
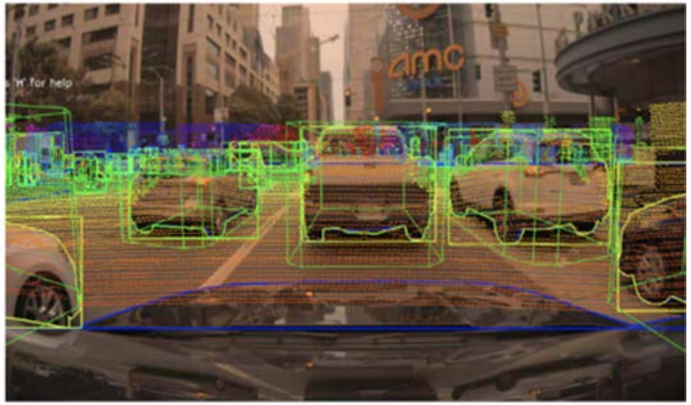


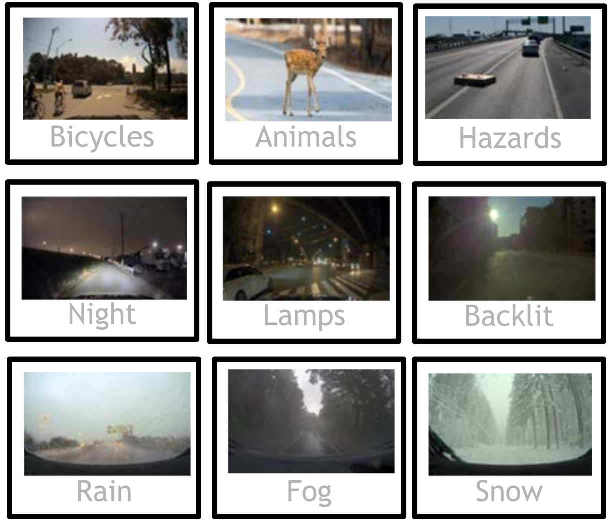
# INSIDE NVIDIA’S AI INFRASTRUCTURE FOR CREATING SELF-DRIVING VEHICLES

CLEMENT FARABET & NICOLAS KOUMCHATZKY | NVIDIA | OPML 2020

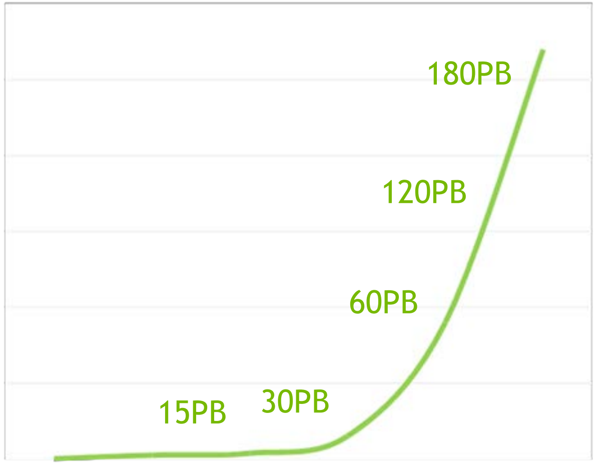
# More Functionality × More Conditions = MASSIVE Data



New features (i.e. lane keeping) require new data…



…and require more real examples to meet safety targets…



…resulting in exponential data and compute needs

# NVIDIA DRIVE DEVELOPMENT PLATFORM

## A complete platform to enable rapid & lifelong AV innovation

Building Autonomous Vehicles (AV) requires a tremendous investment:

**Continuous Engineering** – For data collection, training, simulation, validation, testing, and deployment

**Strong Infrastructure, Tools and Methodology** – Comprehensive platform not commercially available today

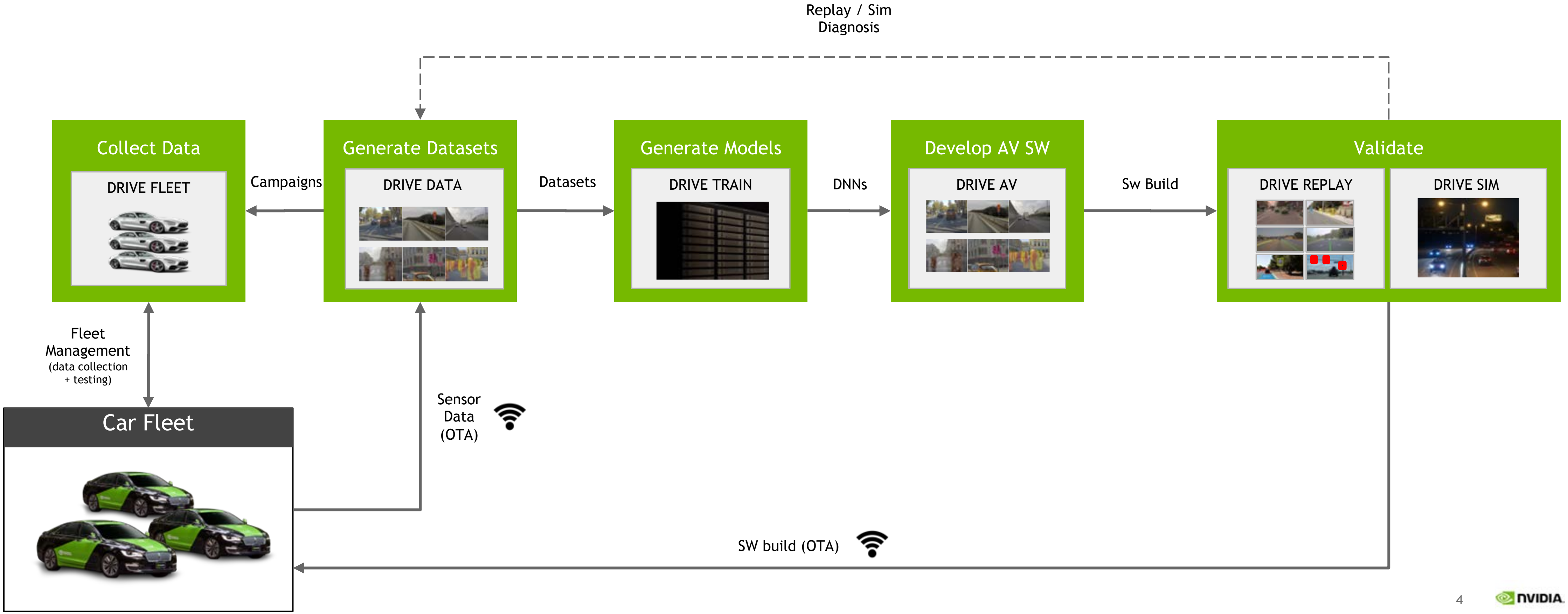
**New Algorithms & SW** – Target computer needs to be SW defined and programmable

**Unified Fleet Management** – One base architecture to enable agility of development, bug fixing

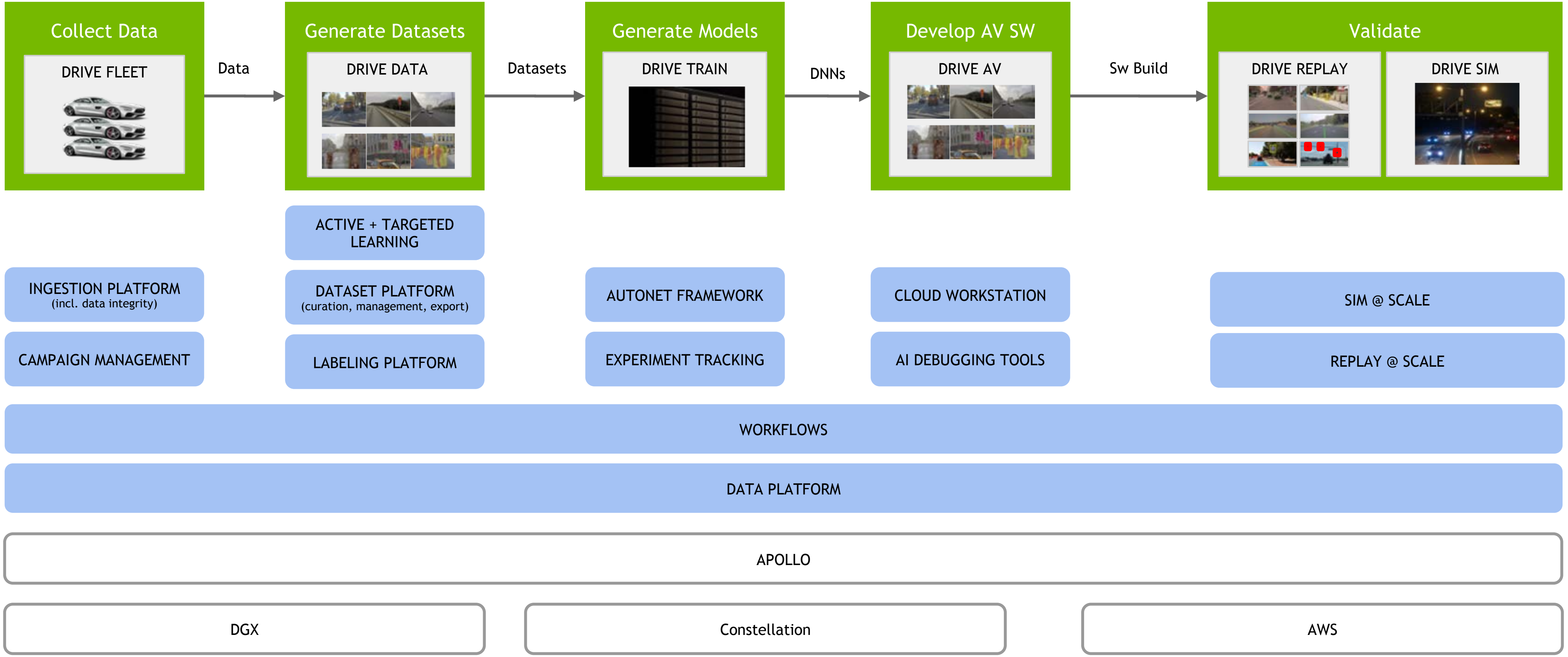
***NVIDIA's open platform, DRIVE, enables to iterate faster and for the entire product lifecycle***

# BUILDING AN AV REQUIRES A PROCESS

## It requires a data driven approach, rich tools to develop and validate (simulation), Hybrid Cloud + Target Hardware – and a strong methodology!



# MAGLEV FOR AV DEVELOPMENT



Collect Data

DRIVE FLEET

Data

Generate Datasets

DRIVE DATA

Datasets

Generate Models

DRIVE TRAIN

DNNs

Develop AV SW

DRIVE AV

Sw Build

Validate

DRIVE REPLAY

DRIVE SIM

ACTIVE + TARGETED LEARNING

INGESTION PLATFORM
(incl. data integrity)

CAMPAIGN MANAGEMENT

DATASET PLATFORM
(curation, management, export)

LABELING PLATFORM

AUTONET FRAMEWORK

EXPERIMENT TRACKING

CLOUD WORKSTATION

AI DEBUGGING TOOLS

SIM @ SCALE

REPLAY @ SCALE

WORKFLOWS

DATA PLATFORM

APOLLO

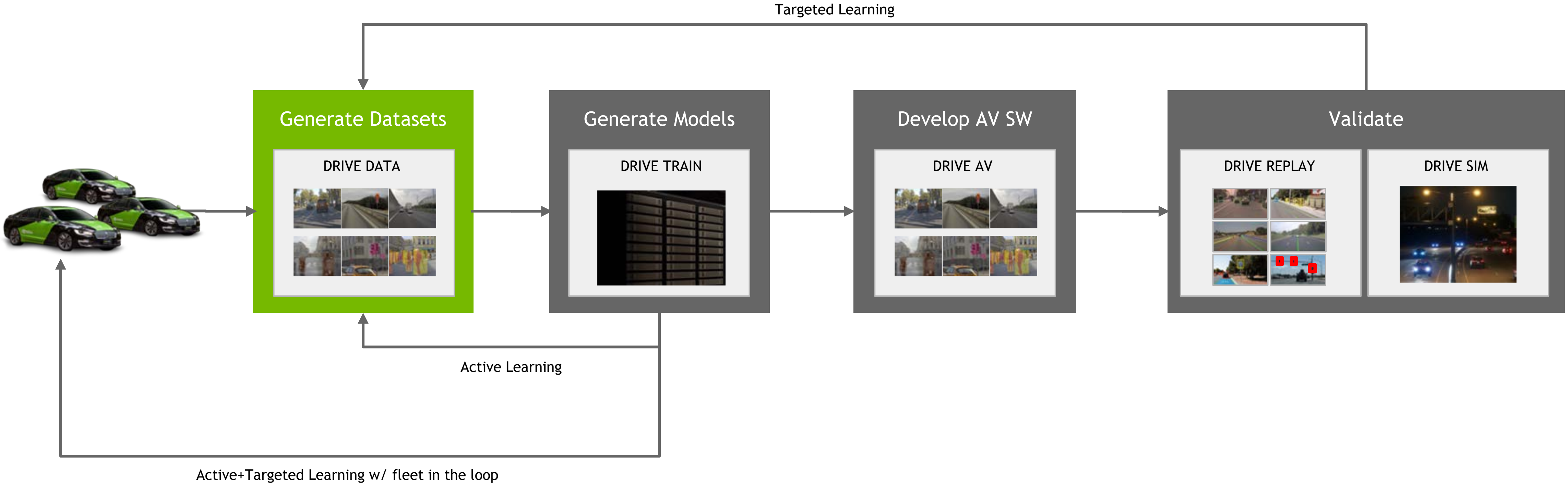
DGX

Constellation

AWS

# DATA DRIVEN AI DEVELOPMENT

## One of the biggest challenges to develop AI is to build the right datasets



Targeted Learning

Generate Datasets

DRIVE DATA

Generate Models

DRIVE TRAIN

Develop AV SW

DRIVE AV

Validate

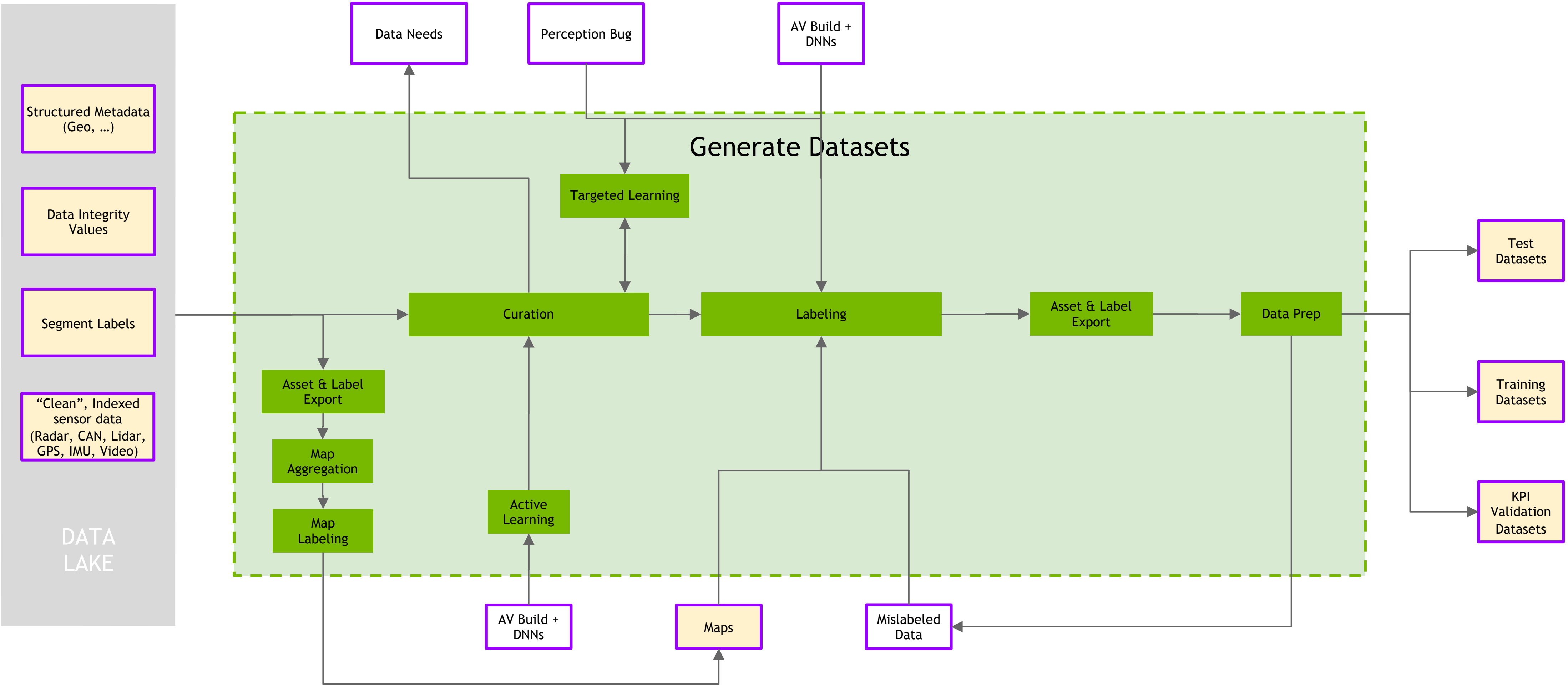
DRIVE REPLAY

DRIVE SIM

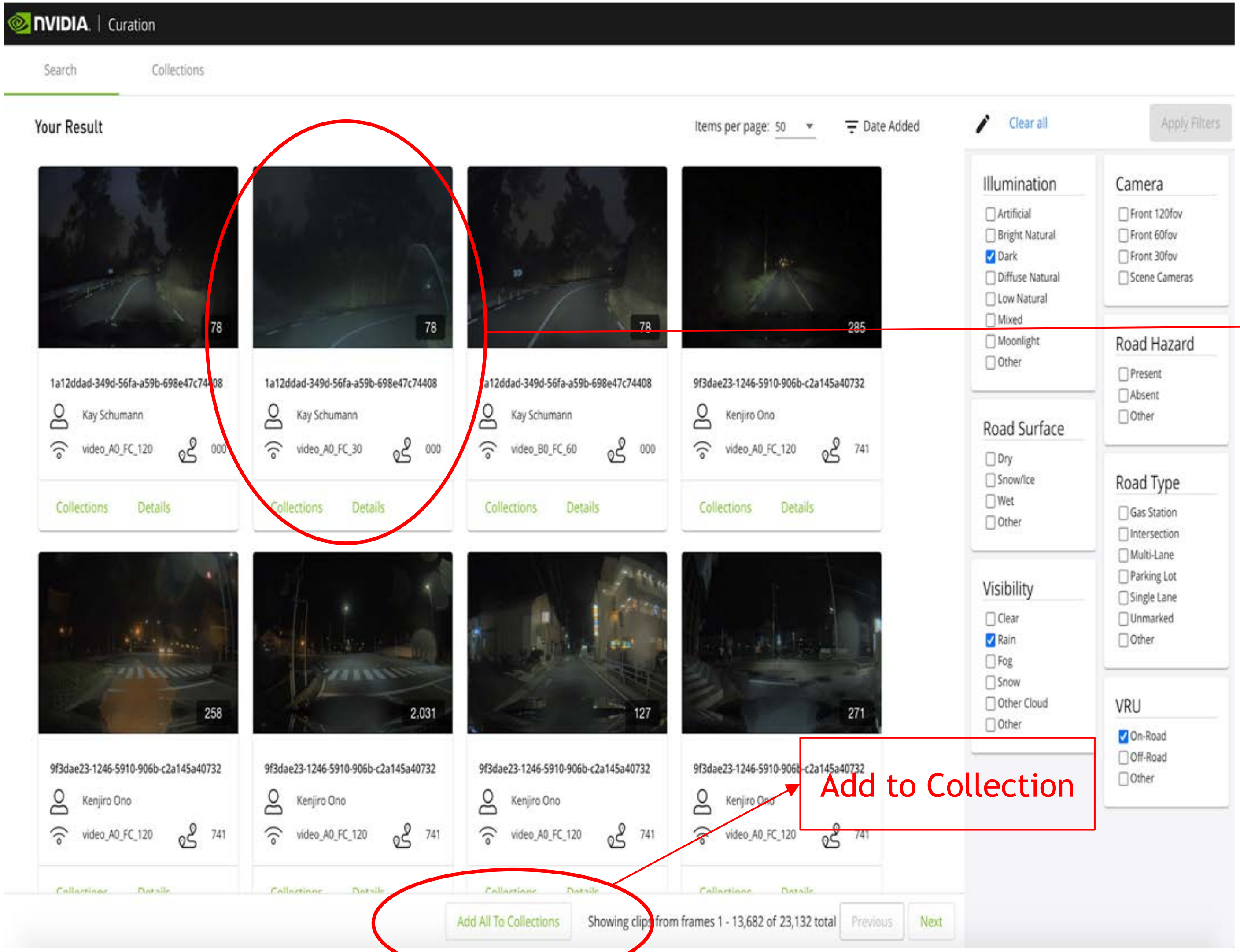
Active Learning

Active+Targeted Learning w/ fleet in the loop

# GENERATE DATASETS: PROCESS



# Curation UI -- Search



NVIDIA | Curation

Search    Collections

Your Result

Items per page: 50    Date Added

Clear all    Apply Filters

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_A0_FC_120    000
Collections    Details

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_A0_FC_30    000
Collections    Details

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_B0_FC_60    000
Collections    Details

285
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120    741
Collections    Details

258
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120    741
Collections    Details

2,031
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120    741
Collections    Details

127
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120    741
Collections    Details

271
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120    741
Collections    Details

Add All To Collections    Showing clips from frames 1 - 13,682 of 23,132 total    Previous    Next

Illumination
- □ Artificial
- □ Bright Natural
- ☑ Dark
- □ Diffuse Natural
- □ Low Natural
- □ Mixed
- □ Moonlight
- □ Other

Road Surface
- □ Dry
- □ Snow/Ice
- □ Wet
- □ Other

Visibility
- □ Clear
- ☑ Rain
- □ Fog
- □ Snow
- □ Other Cloud
- □ Other

Camera
- □ Front 120fov
- □ Front 60fov
- □ Front 30fov
- □ Scene Cameras

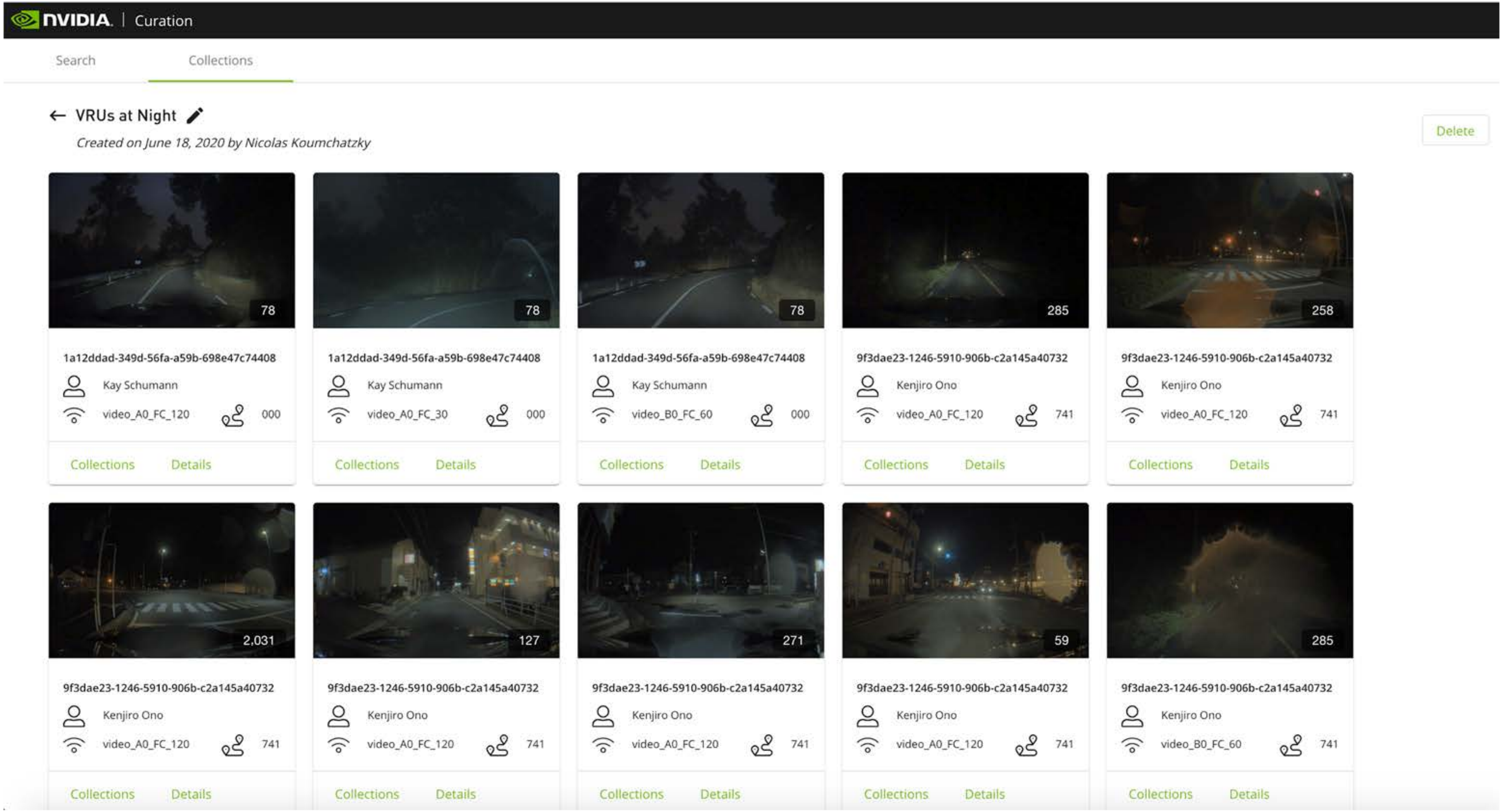
Road Hazard
- □ Present
- □ Absent
- □ Other

Road Type
- □ Gas Station
- □ Intersection
- □ Multi-Lane
- □ Parking Lot
- □ Single Lane
- □ Unmarked
- □ Other

VRU
- ☑ On-Road
- □ Off-Road
- □ Other

Add to Collection

# Curation UI -- Collections



← VRUs at Night

*Created on June 18, 2020 by Nicolas Koumchatzky*

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_A0_FC_120 000

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_A0_FC_30 000

78
1a12ddad-349d-56fa-a59b-698e47c74408
Kay Schumann
video_B0_FC_60 000

285
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

258
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

2,031
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

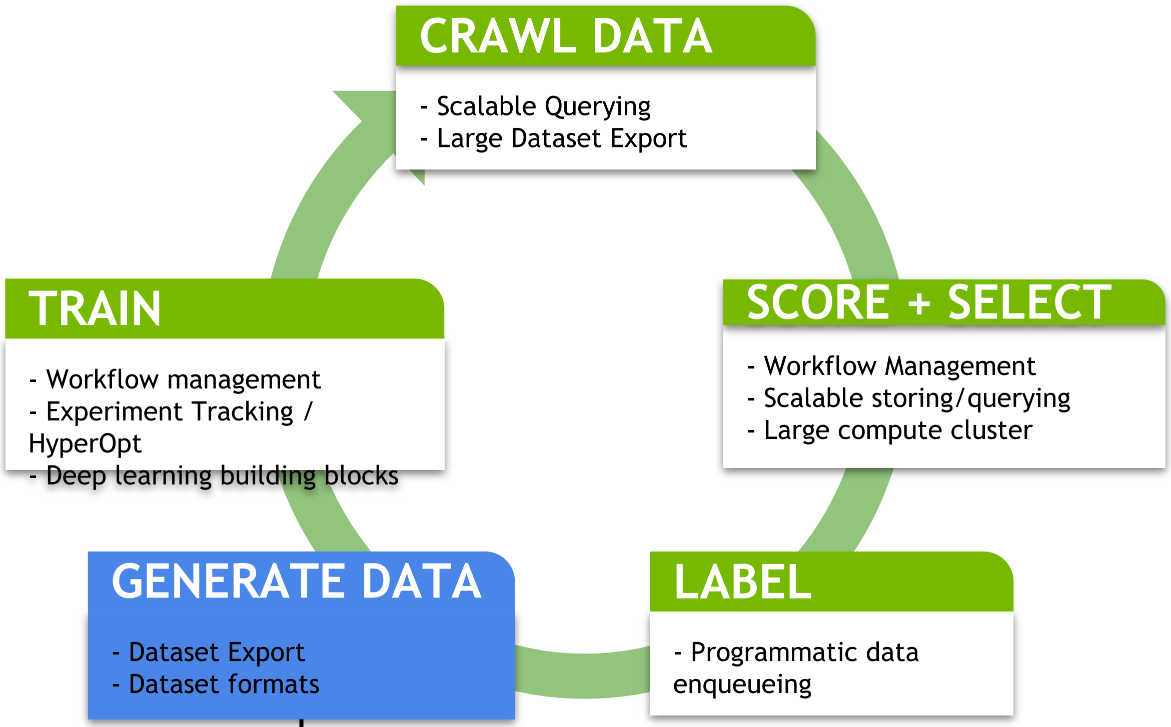
127
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

271
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

59
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_A0_FC_120 741

285
9f3dae23-1246-5910-906b-c2a145a40732
Kenjiro Ono
video_B0_FC_60 741

# ACTIVE LEARNING WITH MAGLEV



CRAWL DATA

- Scalable Querying
- Large Dataset Export

SCORE + SELECT

- Workflow Management
- Scalable storing/querying
- Large compute cluster

LABEL

- Programmatic data enqueueing

GENERATE DATA

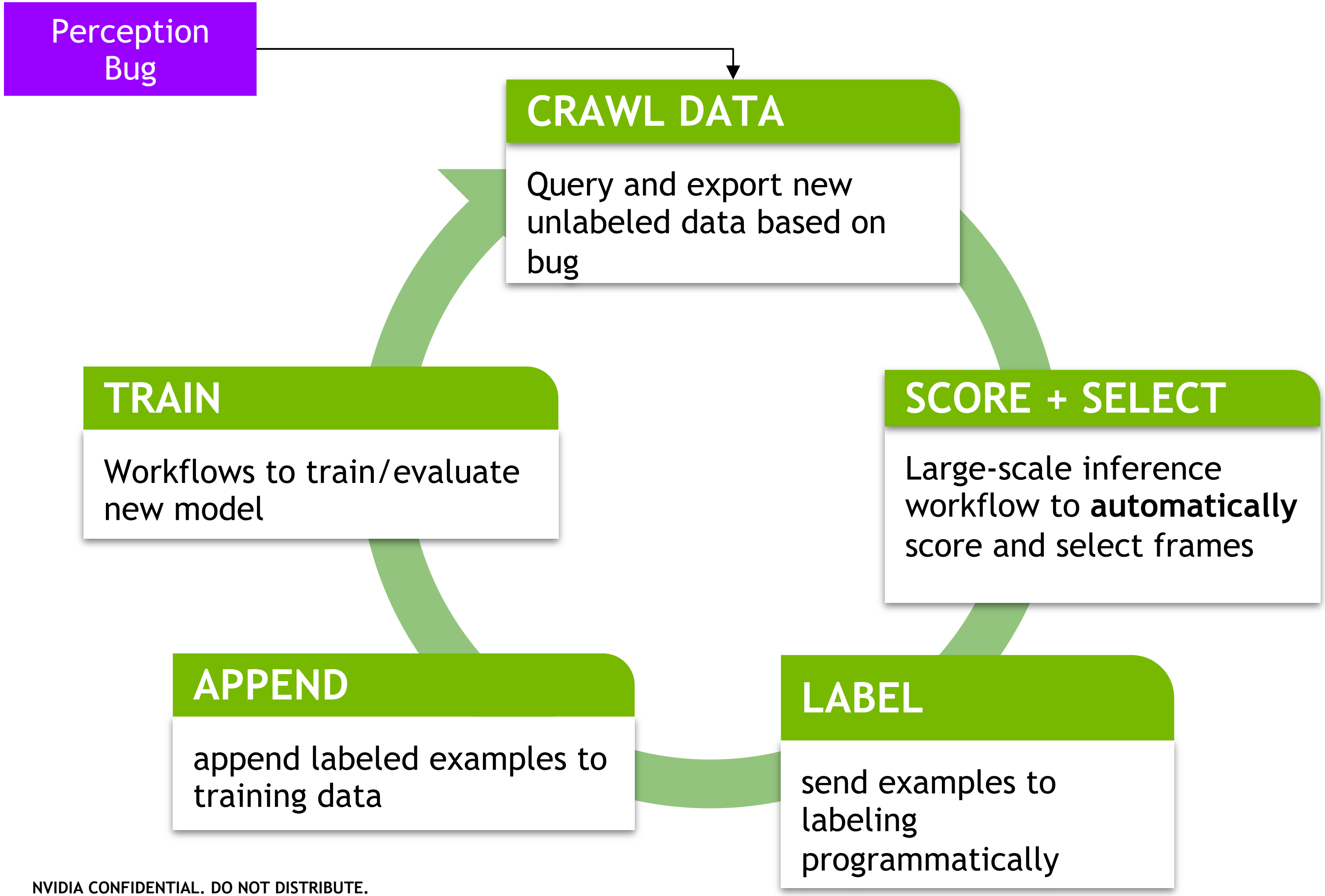
- Dataset Export
- Dataset formats

TRAIN

- Workflow management
- Experiment Tracking / HyperOpt
- Deep learning building blocks

# TARGETED LEARNING

## Similar cycle as active learning, but mine data based on “seed” (=bug)



Perception Bug

CRAWL DATA

Query and export new unlabeled data based on bug

SCORE + SELECT

Large-scale inference workflow to **automatically** score and select frames

LABEL

send examples to labeling programmatically

APPEND

append labeled examples to training data

TRAIN

Workflows to train/evaluate new model

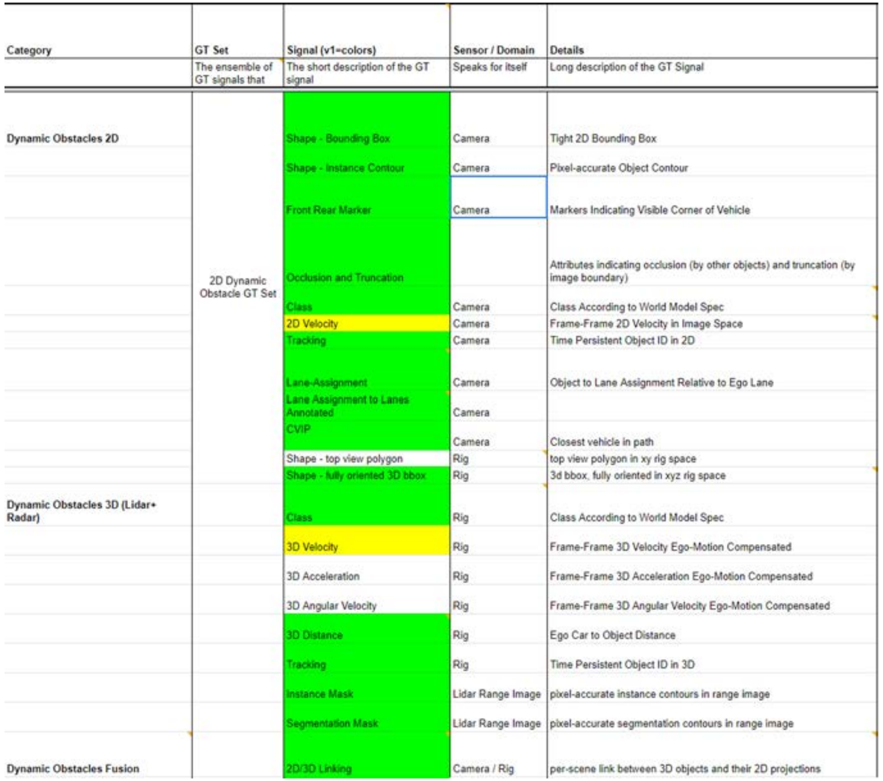
Similar cycle as for active learning, but:

Relies on bug to seed search through data lake

# DIFFICULTY OF GROUND TRUTH PRODUCTION

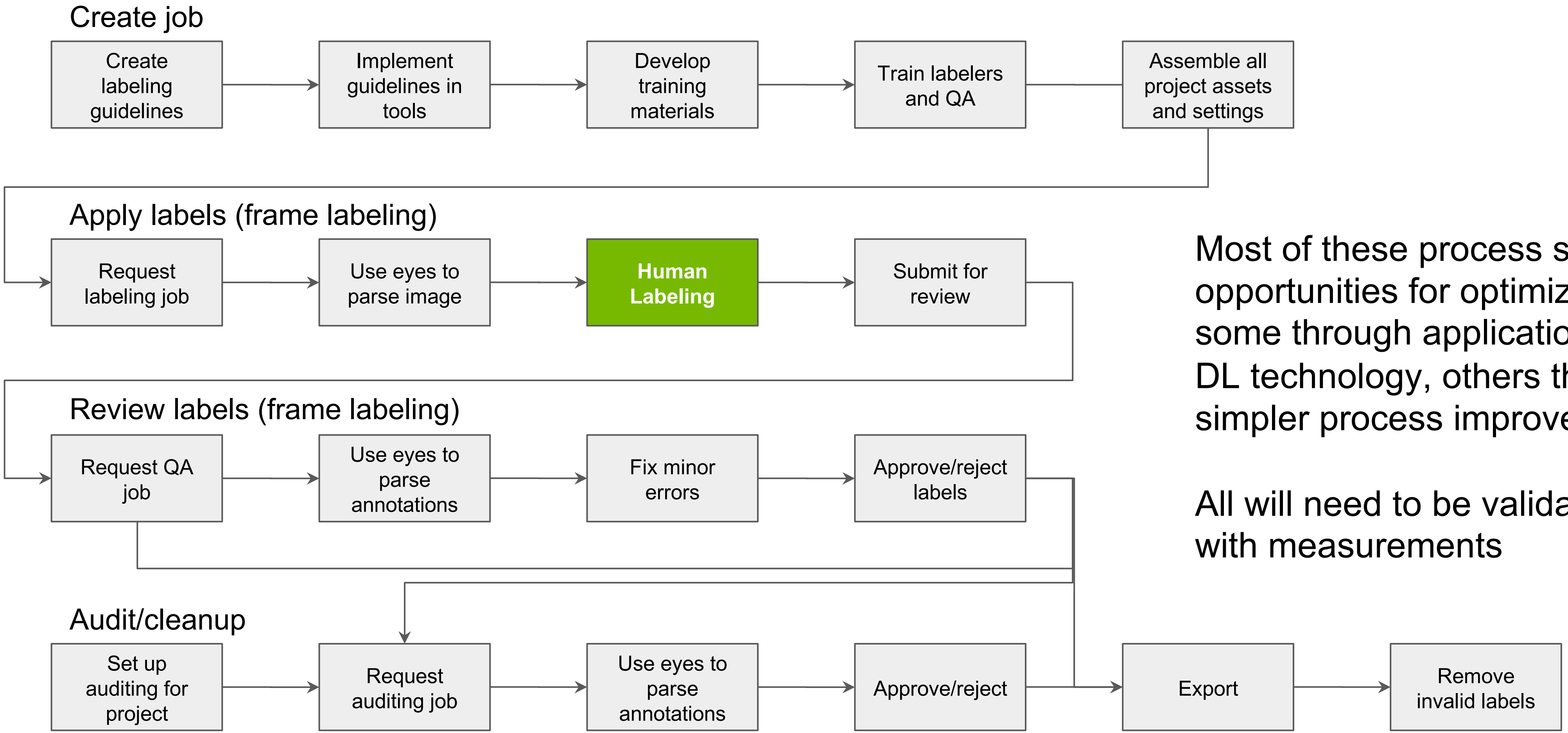
## Many Different GT Signals Need to be Supported

- Many different input sensors
  - Standard sensors
  - LIDAR
  - High precision GPS
  - Etc.
- 100+ Output signals used
  - Static vs. dynamic
  - Obstacles
  - Traffic lights
  - Maps
  - Etc.
- Different outputs for different use cases
  - 2D - Projected back to image space on all sensors
  - 3D - World coordinates
  - Etc.



| Category | GT Set | Signal (v1=colors) | Sensor / Domain | Details |
|---|---|---|---|---|
| | The ensemble of GT signals that | The short description of the GT signal | Speaks for itself | Long description of the GT Signal |
| Dynamic Obstacles 2D | 2D Dynamic Obstacle GT Set | Shape - Bounding Box | Camera | Tight 2D Bounding Box |
| | | Shape - Instance Contour | Camera | Pixel-accurate Object Contour |
| | | Front Rear Marker | Camera | Markers Indicating Visible Corner of Vehicle |
| | | Occlusion and Truncation | | Attributes indicating occlusion (by other objects) and truncation (by image boundary) |
| | | Class | Camera | Class According to World Model Spec |
| | | 2D Velocity | Camera | Frame-Frame 2D Velocity in Image Space |
| | | Tracking | Camera | Time Persistent Object ID in 2D |
| | | Lane-Assignment | Camera | Object to Lane Assignment Relative to Ego Lane |
| | | Lane Assignment to Lanes Annotated | Camera | |
| | | CVIP | Camera | Closest vehicle in path |
| | | Shape - top view polygon | Rig | top view polygon in xy rig space |
| | | Shape - fully oriented 3D bbox | Rig | 3d bbox, fully oriented in xyz rig space |
| Dynamic Obstacles 3D (Lidar+ Radar) | | Class | Rig | Class According to World Model Spec |
| | | 3D Velocity | Rig | Frame-Frame 3D Velocity Ego-Motion Compensated |
| | | 3D Acceleration | Rig | Frame-Frame 3D Acceleration Ego-Motion Compensated |
| | | 3D Angular Velocity | Rig | Frame-Frame 3D Angular Velocity Ego-Motion Compensated |
| | | 3D Distance | Rig | Ego Car to Object Distance |
| | | Tracking | Rig | Time Persistent Object ID in 3D |
| | | Instance Mask | Lidar Range Image | pixel-accurate instance contours in range image |
| | | Segmentation Mask | Lidar Range Image | pixel-accurate segmentation contours in range image |
| Dynamic Obstacles Fusion | | 2D/3D Linking | Camera / Rig | per-scene link between 3D objects and their 2D projections |

# E2E LABELING WORKFLOW



Create job

Create labeling guidelines → Implement guidelines in tools → Develop training materials → Train labelers and QA → Assemble all project assets and settings

Apply labels (frame labeling)

Request labeling job → Use eyes to parse image → **Human Labeling** → Submit for review

Review labels (frame labeling)

Request QA job → Use eyes to parse annotations → Fix minor errors → Approve/reject labels

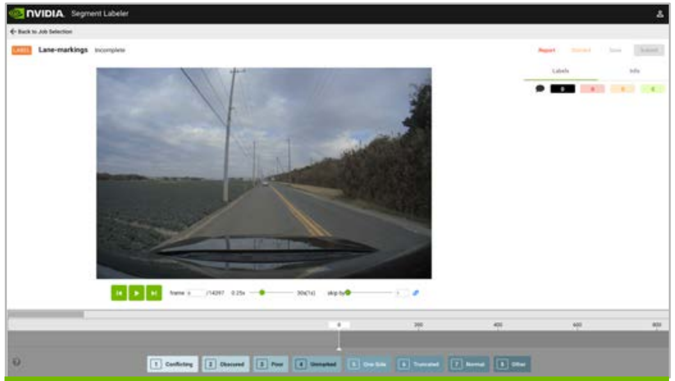
Audit/cleanup

Set up auditing for project → Request auditing job → Use eyes to parse annotations → Approve/reject → Export → Remove invalid labels

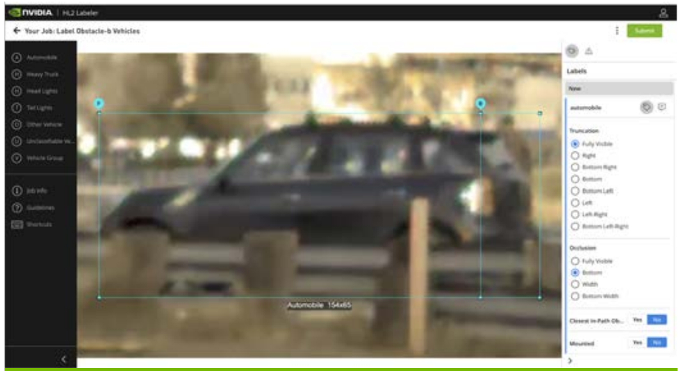
Most of these process steps for opportunities for optimization, some through applications of DL technology, others through simpler process improvements

All will need to be validated with measurements

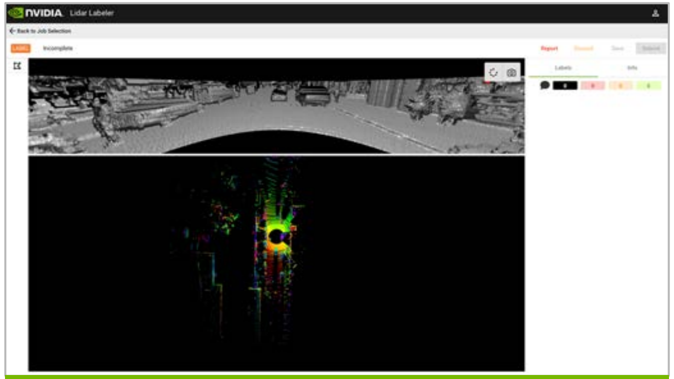
# LABELING TOOLS



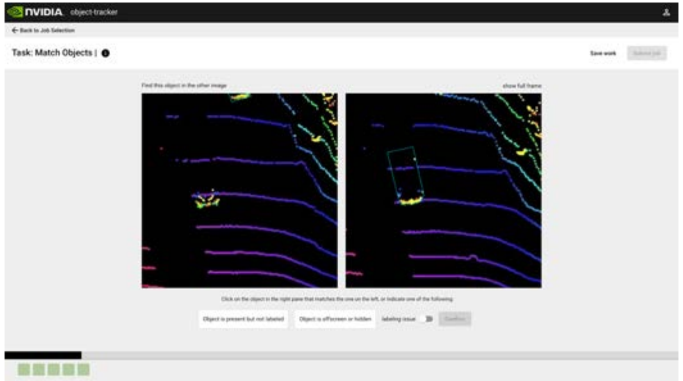
Segment Labeler



2D Image Labeling



LIDAR Labeling

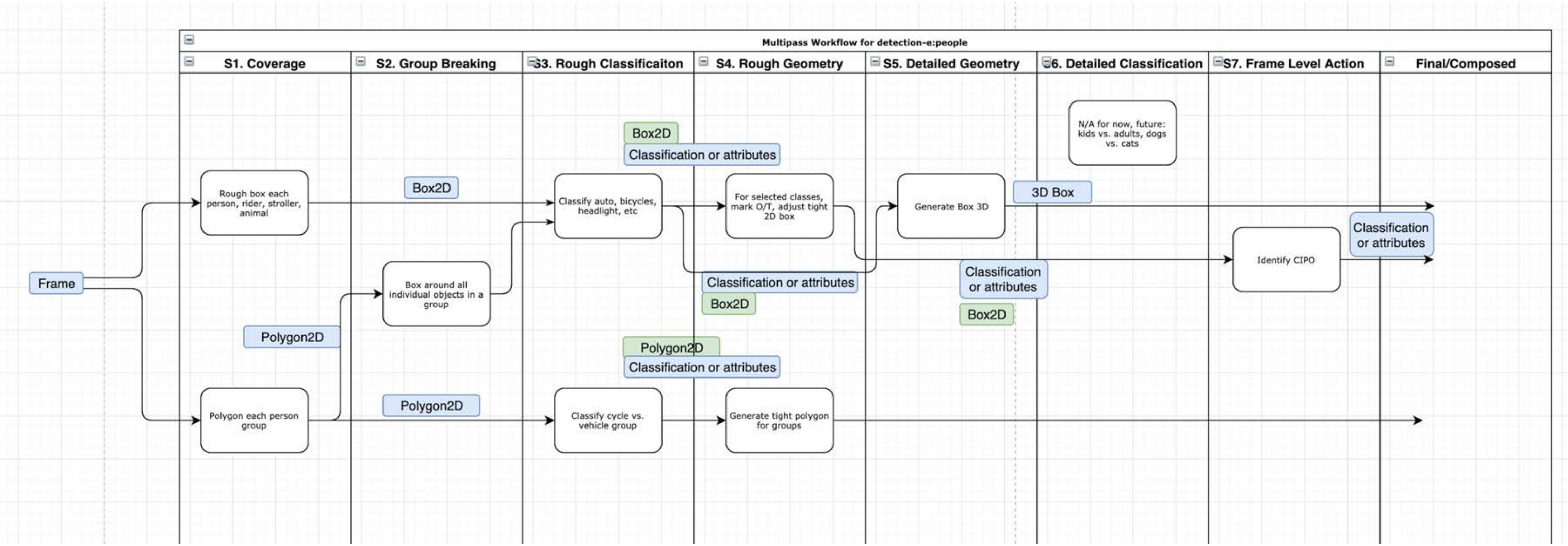


Click on the object in the right pane that matches the one on the left, or indicate one of the following

Object Tracking

# LABELING WORKFLOWS

## Pluggable, composable workflows



Multipass Workflow for detection-e:people

S1. Coverage

S2. Group Breaking

S3. Rough Classificaiton

S4. Rough Geometry

S5. Detailed Geometry

S6. Detailed Classification

S7. Frame Level Action

Final/Composed

Frame

Rough box each person, rider, stroller, animal

Box2D

Classify auto, bicycles, headlight, etc

Box2D

Classification or attributes

For selected classes, mark O/T, adjust tight 2D box

Classification or attributes

Box2D

Generate Box 3D

3D Box

N/A for now, future: kids vs. adults, dogs vs. cats

Classification or attributes

Box2D

Identify CIPO

Classification or attributes

Polygon2D

Box around all individual objects in a group

Polygon each person group

Polygon2D

Classify cycle vs. vehicle group

Polygon2D

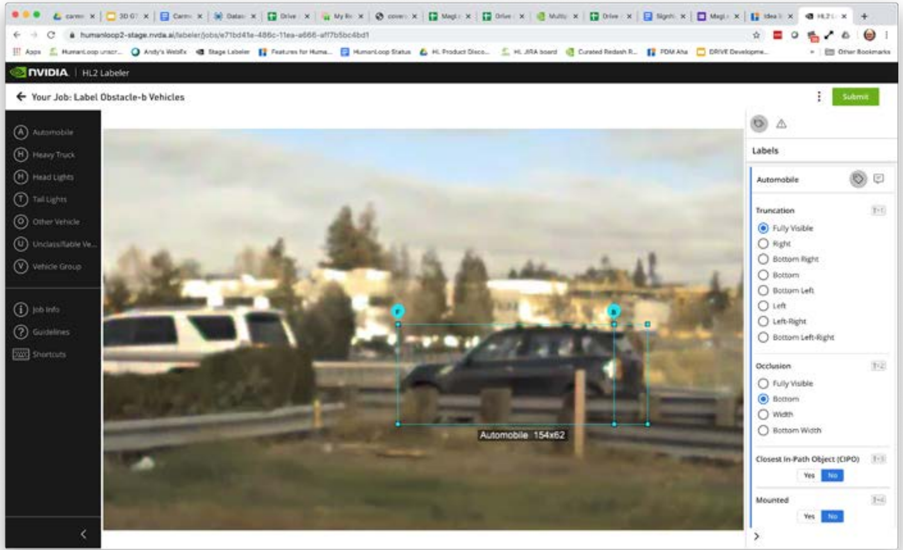
Classification or attributes

Generate tight polygon for groups

# LABELING: VERSIONED, TRACEABLE GUIDELINES

```
key: frame:obstacle-b:vehicles:v2
title: Obstacle-b Vehicles
owner: Tilman Wekel <twekel@nvidia.com>
description: |+
  Vehicles General Guidelines:
  - All vehicles in a frame must be boxed.
  - Do not include antennas/long-structures on top of vehicles in the
    BBOX for obstacle project.
  - Start with obvious "Automobiles" and "Heavy Truck" understand the
    differences from examples.
  - If a vehicle is not obviously an "Automobile" a “Heavy Truck”
    please label it as an “Other Vehicle”.
  - If a something is obviously a vehicle but is too far or hidden to
    determine the type, please use “Unclassifiable Vehicle”.
  - Please read the guidelines for “Vehicle Group” to speed up
    labeling in certain conditions.
  - Minimum Bounding Box size is 10px width and 6px height.
docurl: smb://netapp-pu/dnndf/AV-Guidelines/obstacle-detection-b-v2.pdf
status: DRAFT
tooltype: SHAPE2D
coveringtype: COVERING
appconfig:
  tasktypes:
    - name: single_pass
      title: Single Pass
  rejectionreasons:
    - name: normal_label
      title: Normal Label
      key: rejection-reasons:generic:normal-label:v1
      values:
        - name: annotation_too_loose
          title: Annotation too loose
        - name: annotation_too_tight
          title: Annotation too tight
        - name: incorrect_classifictation
          title: Classification incorrect
        - name: incorrect_attributes
          title: Attributes incorrect
        - name: other
          title: Other
          require_comment: true
    - name: bad_label
      title: Bad Label
      key: rejection-reasons:generic:bad-label:v1
      values:
        - name: wrong_classification
          title: Wrong Classification
        - name: other
          title: Other
          require_comment: true
```

Generate Labeling UI



Your Job: Label Obstacle-b Vehicles

Automobile

Truncation
● Fully Visible
○ Right
○ Bottom Right
○ Bottom
○ Bottom Left
○ Left
○ Left-Right
○ Bottom Left-Right

Occlusion
○ Fully Visible
● Bottom
○ Width
○ Bottom Width

Closest In-Path Object (CIPO)
Yes [✓ No]

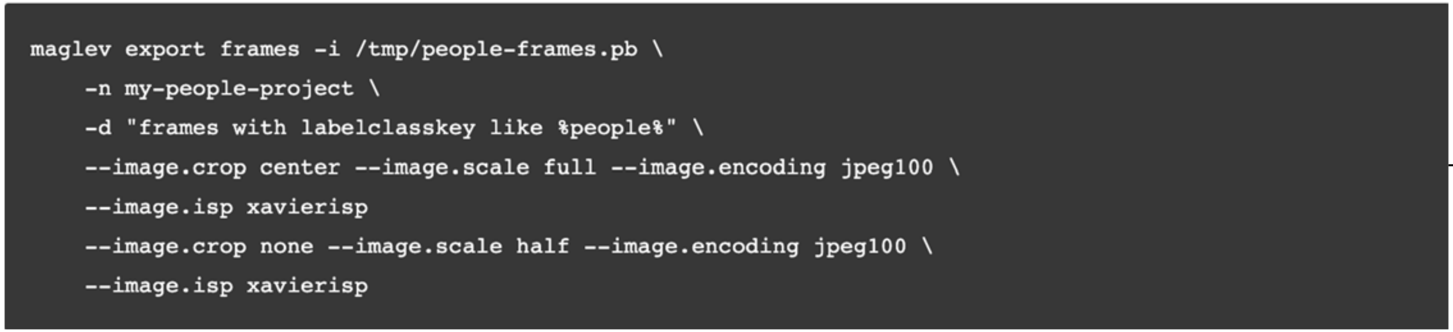
Mounted
Yes [✓ No]

Generate Labels

**Store Labels with Guidelines**

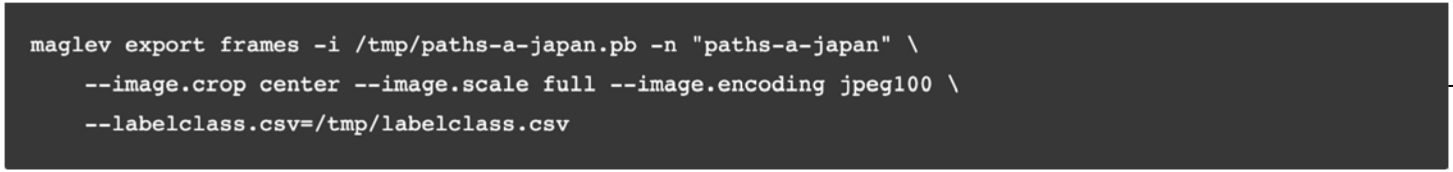
Generated labels linked to versioned guidelines

# DATASET EXPORT



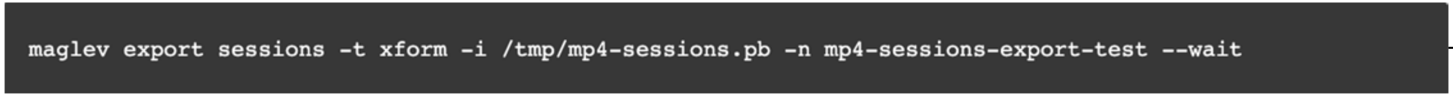
```
maglev export frames -i /tmp/people-frames.pb \
    -n my-people-project \
    -d "frames with labelclasskey like %people%" \
    --image.crop center --image.scale full --image.encoding jpeg100 \
    --image.isp xavierisp
    --image.crop none --image.scale half --image.encoding jpeg100 \
    --image.isp xavierisp
```

→ Export HWISP transcoded frames with transformations



```
maglev export frames -i /tmp/paths-a-japan.pb -n "paths-a-japan" \
    --image.crop center --image.scale full --image.encoding jpeg100 \
    --labelclass.csv=/tmp/labelclass.csv
```

→ Export frames with labels

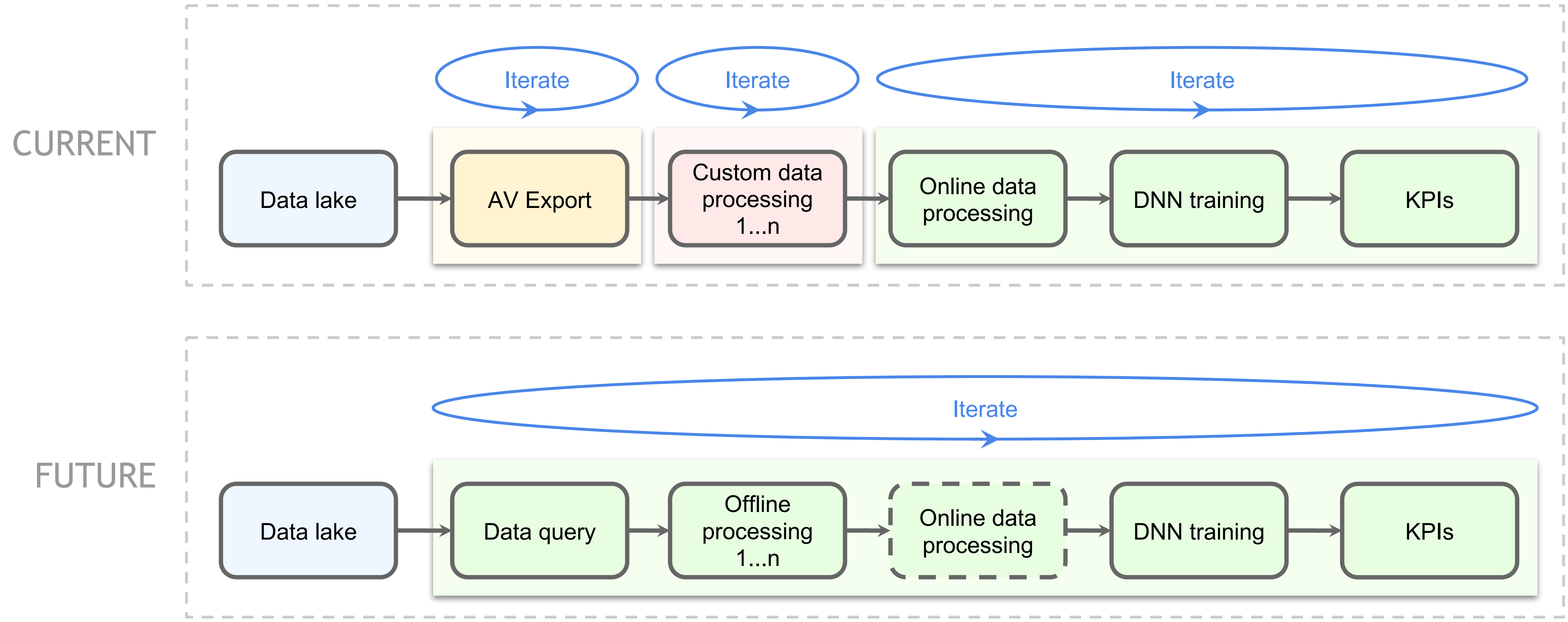


```
maglev export sessions -t xform -i /tmp/mp4-sessions.pb -n mp4-sessions-export-test --wait
```

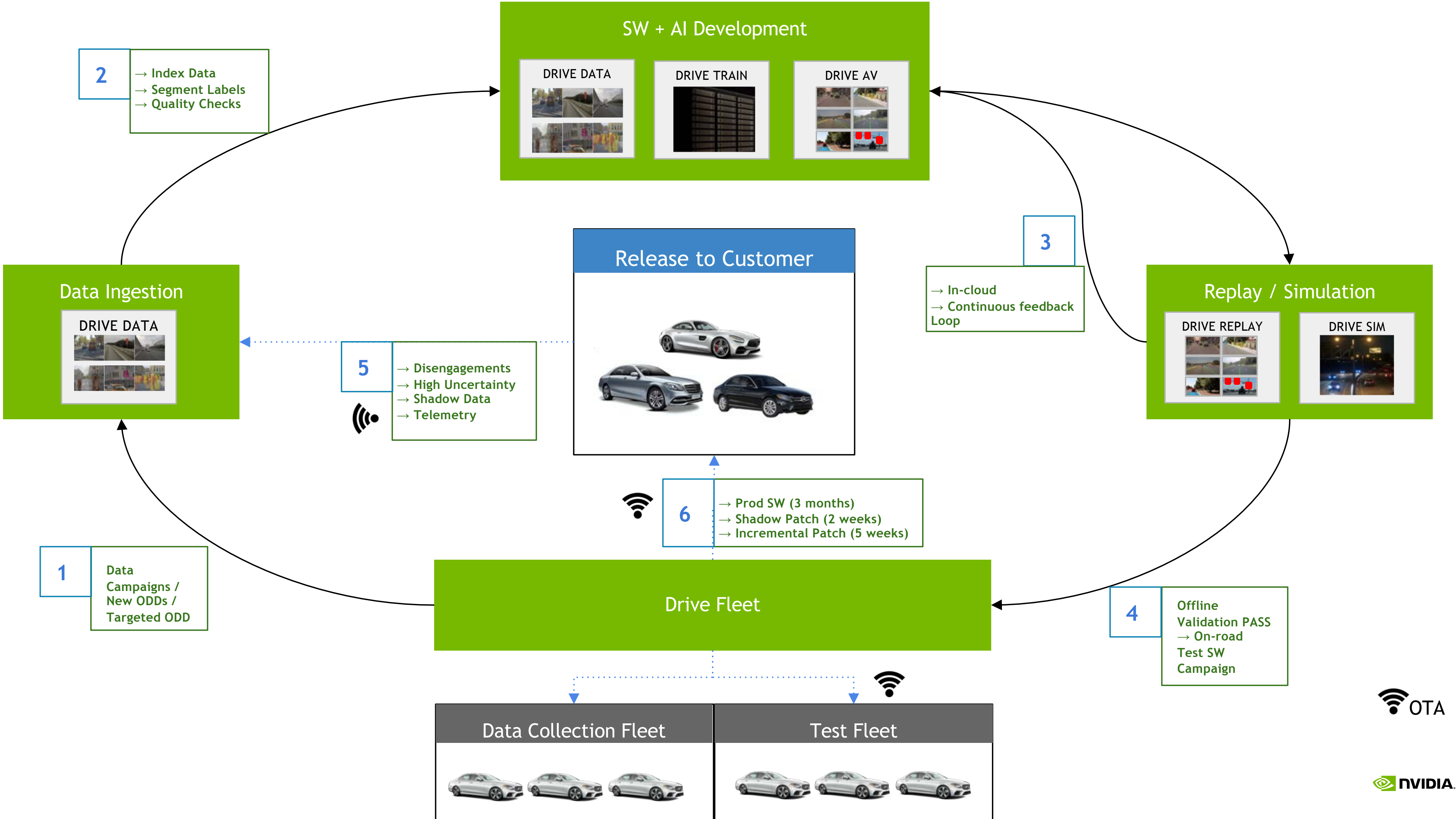
→ Export Sessions

# DATA PREP

## Unified & extensible data preparation



# DEPLOYMENT AT FLEET SCALE



SW + AI Development

DRIVE DATA

DRIVE TRAIN

DRIVE AV

Data Ingestion

DRIVE DATA

Release to Customer

Replay / Simulation

DRIVE REPLAY

DRIVE SIM

Drive Fleet

Data Collection Fleet

Test Fleet

1

Data Campaigns / New ODDs / Targeted ODD

2

→ Index Data
→ Segment Labels
→ Quality Checks

3

→ In-cloud
→ Continuous feedback Loop

4

Offline Validation PASS
→ On-road Test SW Campaign

5

→ Disengagements
→ High Uncertainty
→ Shadow Data
→ Telemetry

6

→ Prod SW (3 months)
→ Shadow Patch (2 weeks)
→ Incremental Patch (5 weeks)

OTA

NVIDIA

# THANK YOU

nvidia.com/en-us/self-driving-cars/drive-labs
twitter.com/nvidiaDRIVE
twitter.com/nvidiaAI
twitter.com/nkoumchatzky
twitter.com/clmt