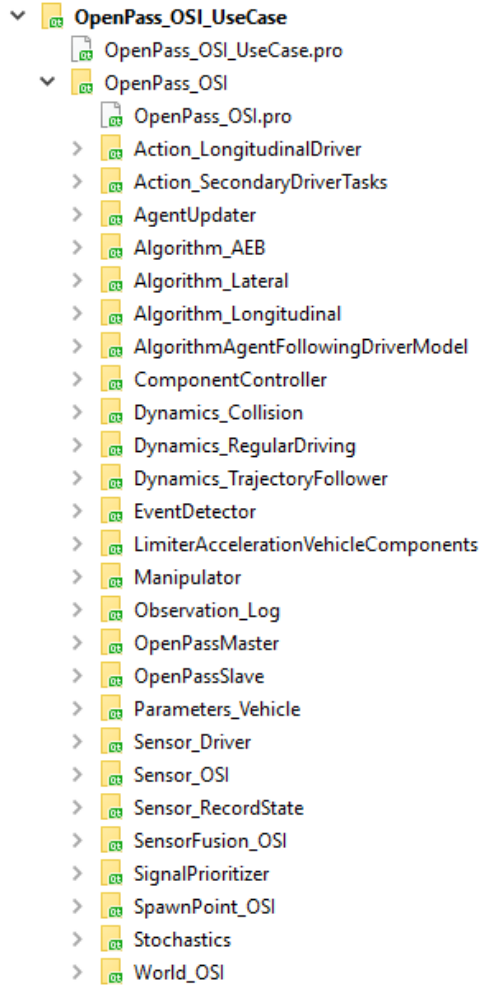
Dipl.-Ing. Konstantin Blenz
Fakultät Verkehrswissenschaften „Friedrich List“
Institut für Automobiltechnik – IAD
Lehrstuhl für Kraftfahrzeugtechnik – LKT

Status Integration Fahrerverhaltensarchitektur in openPASS v0.6

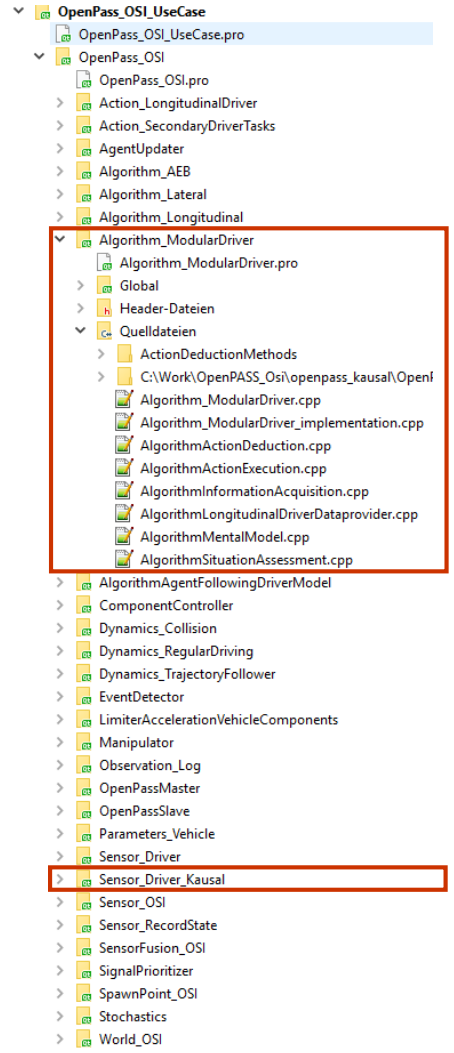
Dresden, 26.09.2019

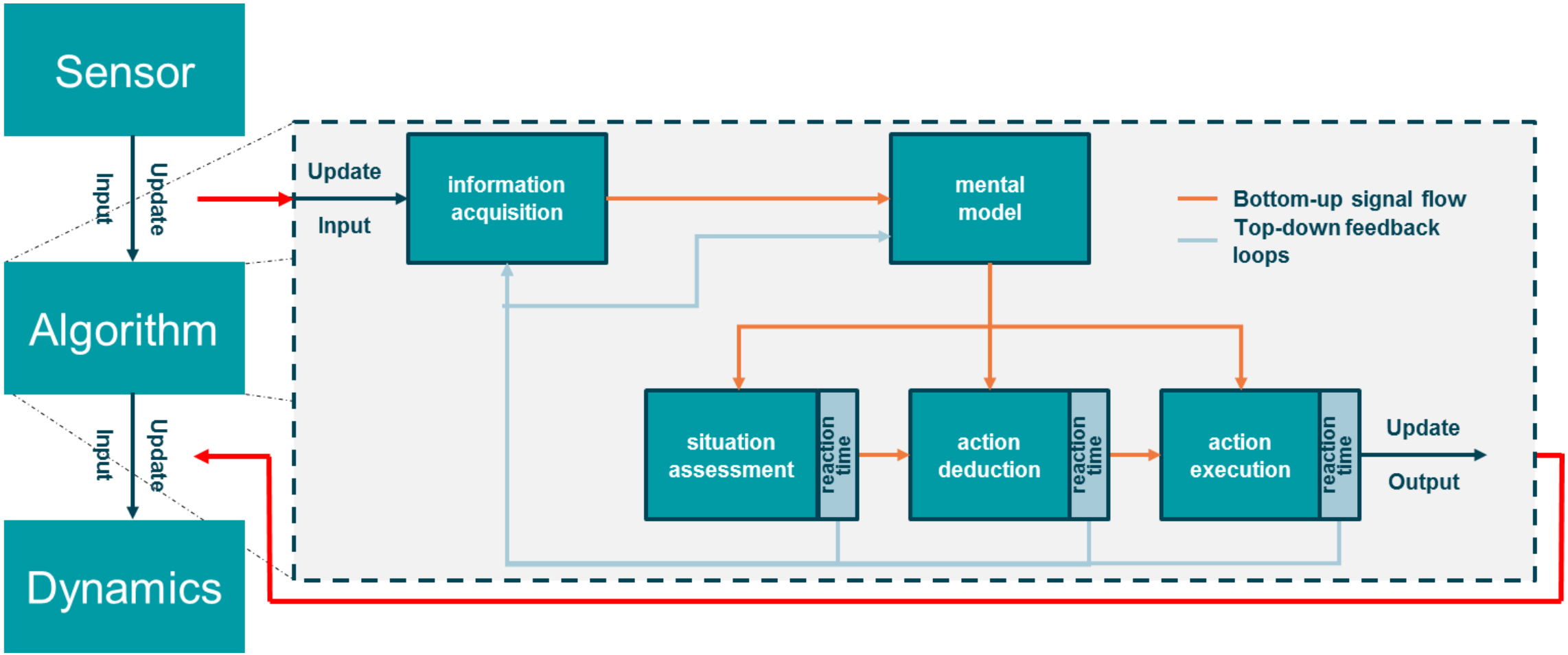
Integration v0.6



openPASS/intech

KAUSAL 2019




Sensor
Input
Update
Algorithm
Input
Update
Dynamics
Update
Input
information acquisition
mental model
Bottom-up signal flow
Top-down feedback loops
situation assessment
reaction time
action deduction
reaction time
action execution
reaction time
Update
Output

Sensor

Input

Update

Algorithm

Dynamics

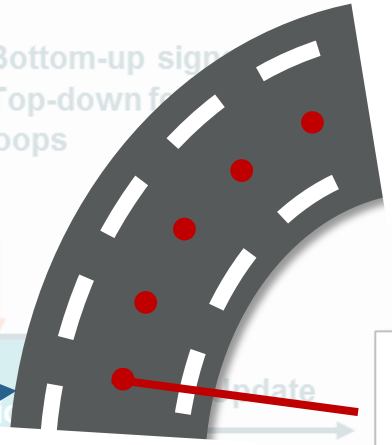
Ego (Output)

Moving Objects (Output)

Environ-ment (Output)

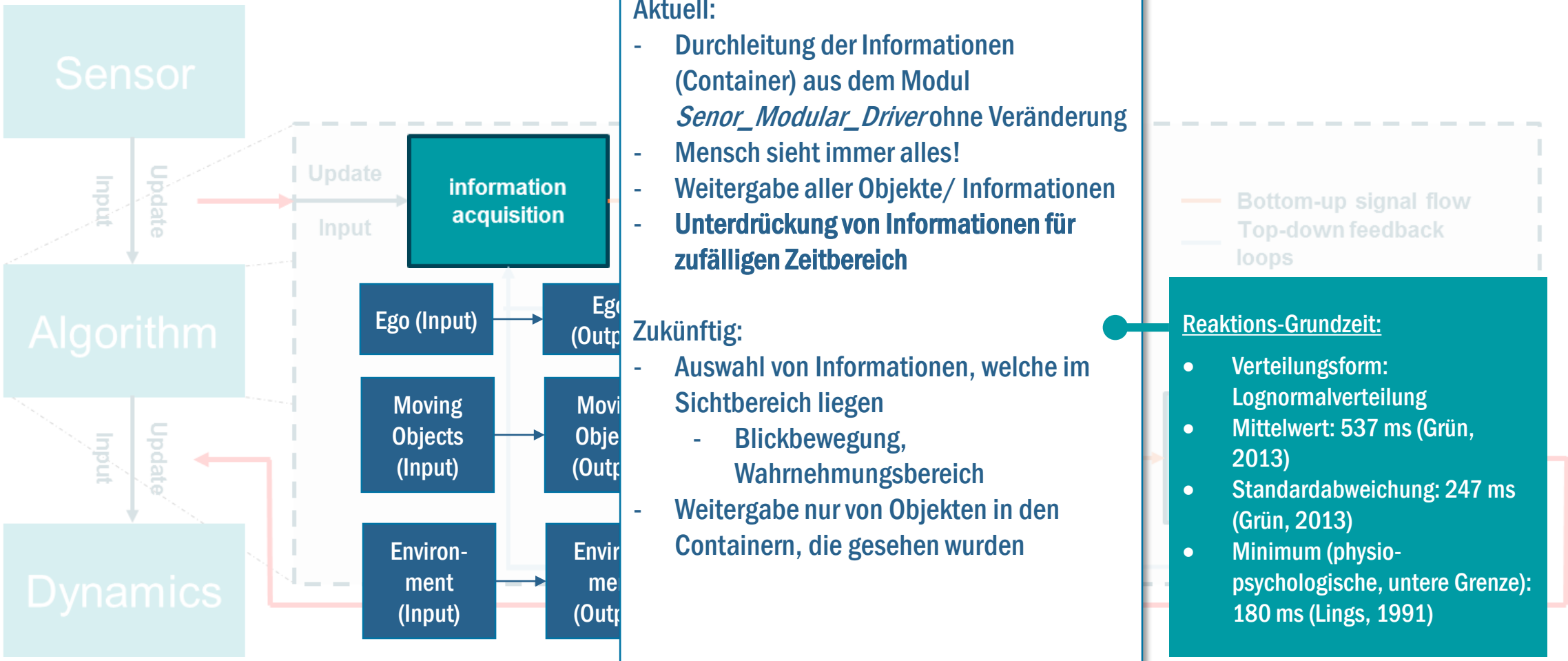
State.id = agent->GetId();
State.pos.xPos = agent->GetPositionX();
State.pos.yPos = agent->GetPositionY();
.
.
.

- TrafficRuleInformations
  - Trafficsigns in range
- RoadGeometryInformations
  - Lanes: Ego, Left, Right
    - Curvature, width, distanceToEndOfLane, laneid
    - MentalModelLanes
      - LaneSections



Tuples:
- x
- y
- yaw
- s

```
<structSignal<StaticEnvironmentData> const>(out_StatEnvironment_Container);
<structSignal<EgoData> const>(out_EgoData_Container);
<structSignal<std::vector<SurroundingMovingObjectsData>>
const>(out_SourrMovingObj_Container);
```



Aktuell:

- Durchleitung der Informationen (Container) aus dem Modul *Senor_Modular_Driver* ohne Veränderung
- Mensch sieht immer alles!
- Weitergabe aller Objekte/ Informationen
- **Unterdrückung von Informationen für zufälligen Zeitbereich**

Zukünftig:

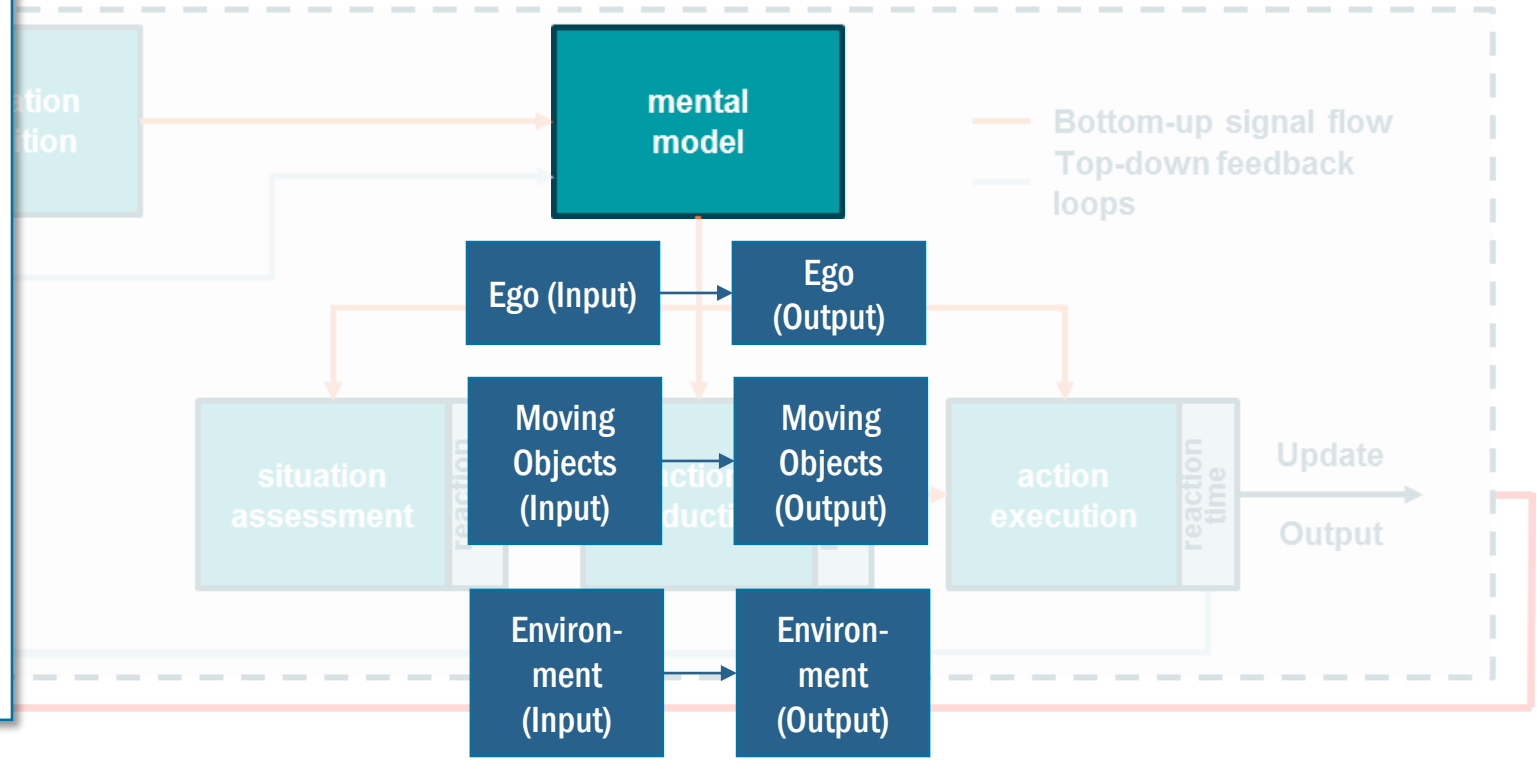
- Auswahl von Informationen, welche im Sichtbereich liegen
  - Blickbewegung, Wahrnehmungsbereich
- Weitergabe nur von Objekten in den Containern, die gesehen wurden

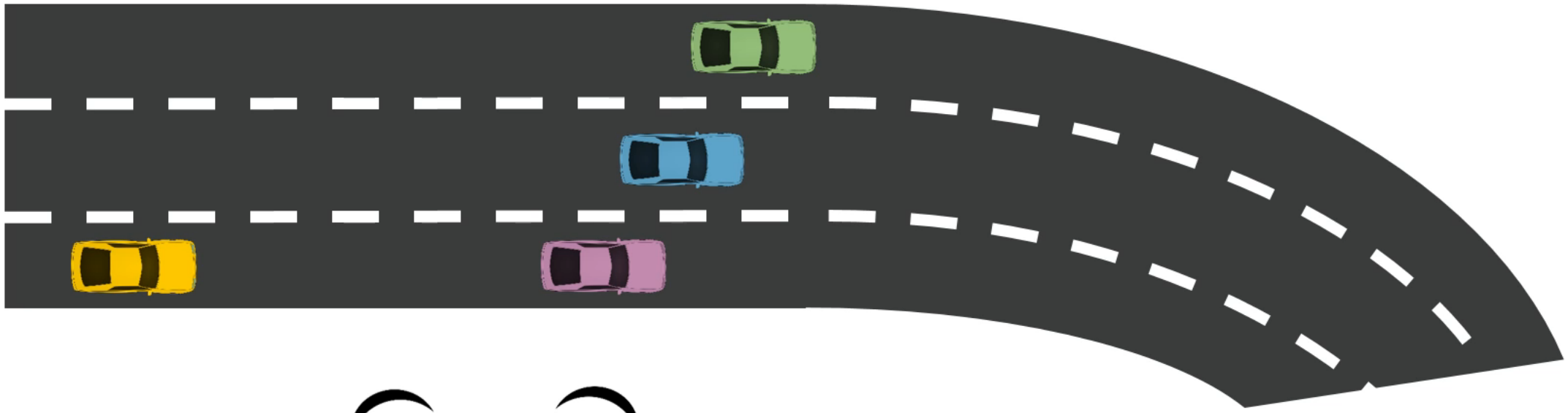
<u>Reaktions-Grundzeit:</u>

- Verteilungsform: Lognormalverteilung
- Mittelwert: 537 ms (Grün, 2013)
- Standardabweichung: 247 ms (Grün, 2013)
- Minimum (physio-psychologische, untere Grenze): 180 ms (Lings, 1991)

Aktuell:
- Extrapolation von Objekten, die in der Vergangenheit gesehen wurden entlang des jeweiligen Fahrstreifens
- Dynamische Extrapolation von Agenten außerhalb der Lanes „Ego, Left, Right“ (siehe Sensor)
- Weitergabe nur von Informationen, die der Fahrer kennt

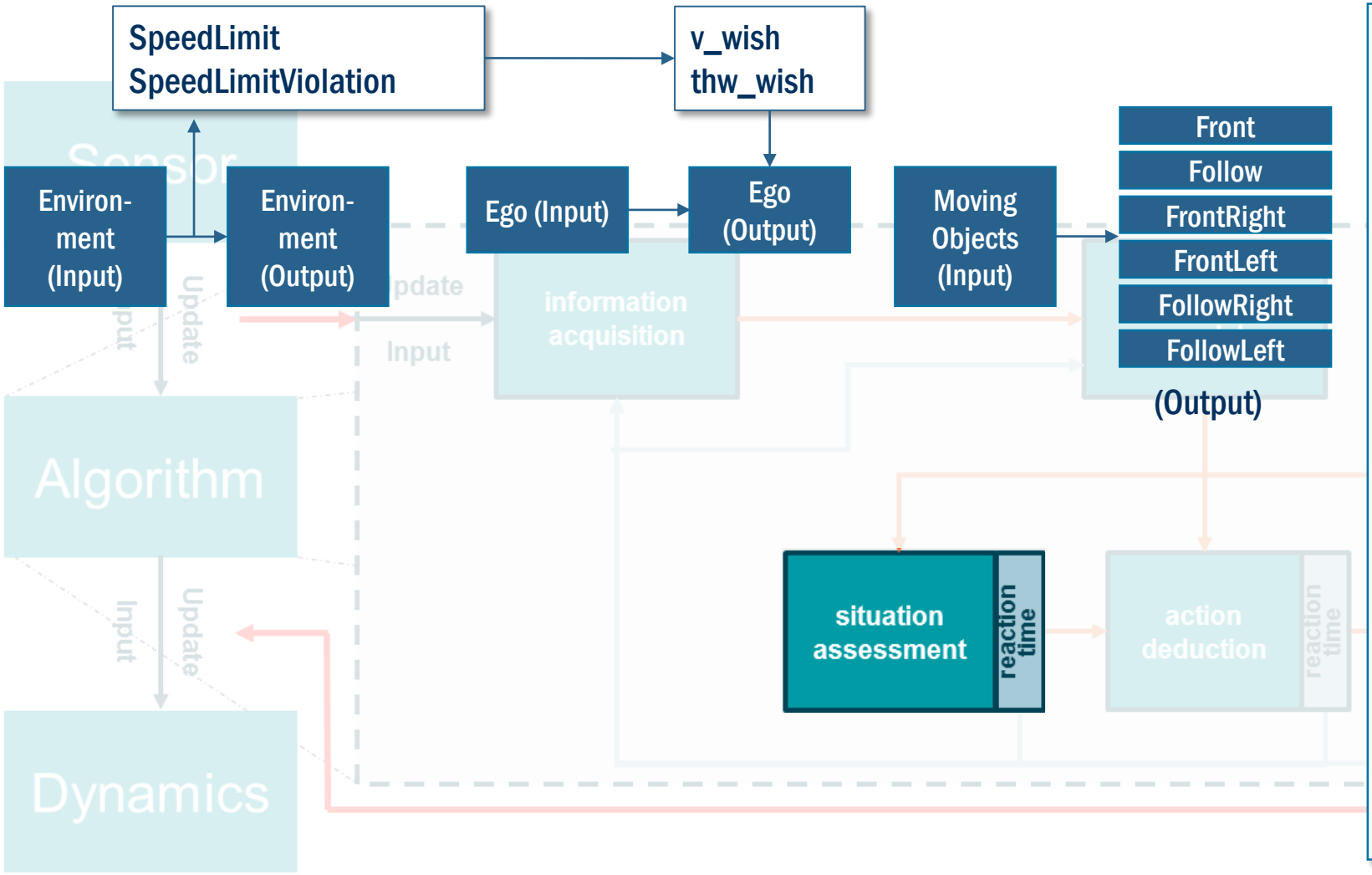
Zukünftig:
- Extrapolation von speziellen Bewegungsmustern (z.B. Spurwechsel)





Ego

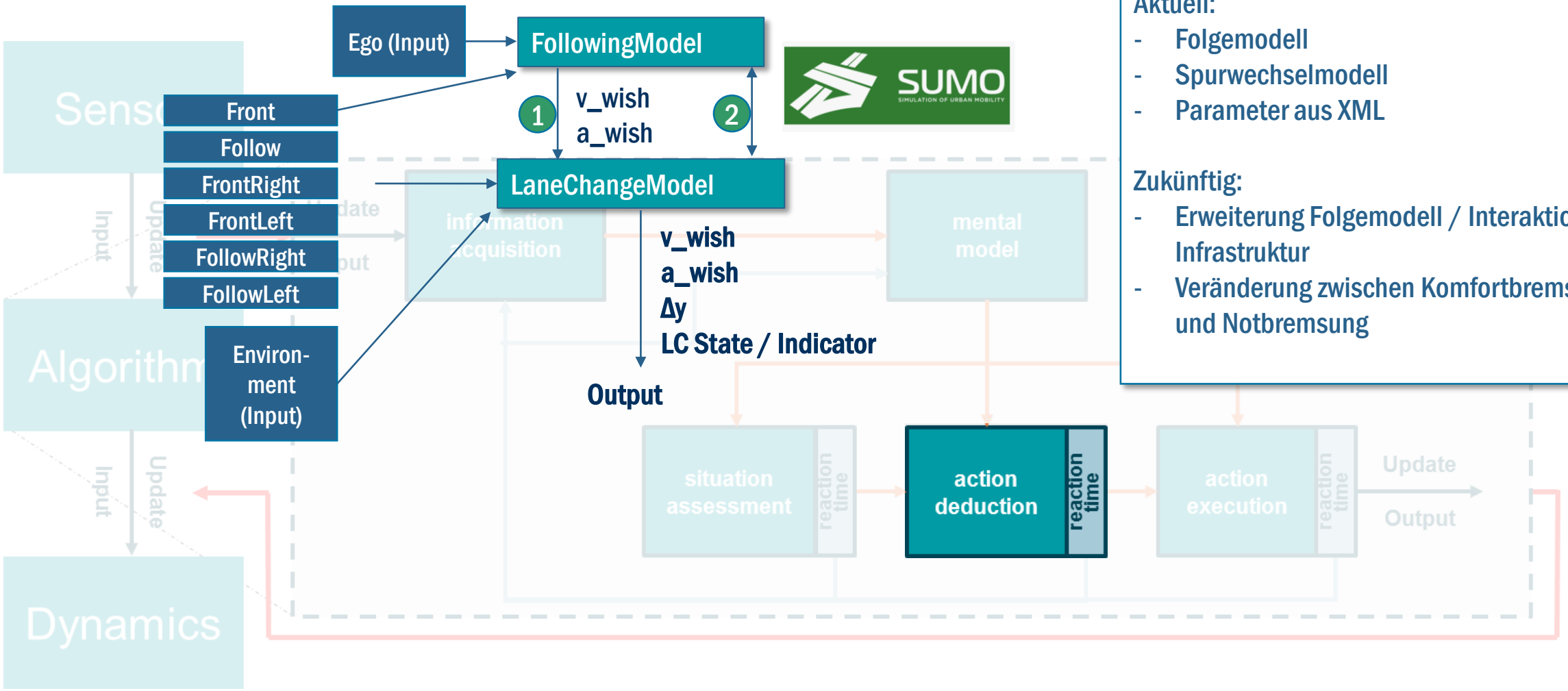
Umgebungsverkehr



Aktuell:
- Einordnung vom Signal vom Typ „MaximumSpeedLimit“
- Festlegung einer Wunschgeschwindigkeit auf Basis der Eingaben in „ProfilesCatalog.xml“ und des Speedlimits + eines stochast. Verstoßwertes
- Festlegung einer Wunschzeitlücke
- Auswahl der den Agenten umgebenden Fahrzeuge, basierend auf deren Spur, sekundärer Spur und Blickersignale
- Schnittstelle zum „Logging“ der Kritikalität

Zukünftig:
- Anpassung der Wunschzeitlücke
- Bewertung der Umwelt (Wetter)
- Konkrete Bewertung der Kritikalität zur Handlungsableitung aus Sicht des Fahrers (z.B. zur Notbremsung)

Fahrerverhaltensmodellierung
Action Deduction



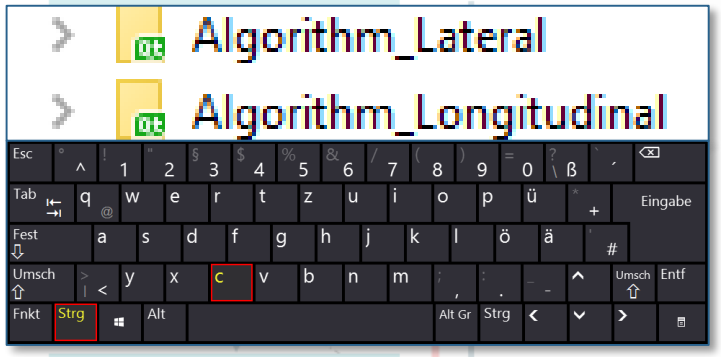
Aktuell:
- Folgemodell
- Spurwechselmodell
- Parameter aus XML

Zukünftig:
- Erweiterung Folgemodell / Interaktion Infrastruktur
- Veränderung zwischen Komfortbremsung und Notbremsung

Fahrerverhaltensmodellierung
Action Deduction

```
if (driverParameters->GetParametersString().count("AlgorithmLongitudinalModule") > 0)
{
    if (driverParameters->GetParametersString().at("AlgorithmLongitudinalModule")!="Dummy")
    {
    const auto& algorithmLongitudinalModuleName = driverParameters->GetParametersString().at("AlgorithmLongitudinalModule")
    GatherComponent(algorithmLongitudinalModuleName, agentBuildInformation.agentType);
    }
}
```

```
<DriverProfile Name="ModularDriver">
    <String Key="Type" Value="AlgorithmModularDriver"/>
    <String Key="SensorDriverModule" Value="Sensor_Modular_Driver"/>
    <String Key="AlgorithmLateralModule" Value="Dummy"/>
    <String Key="AlgorithmLongitudinalModule" Value="Dummy"/>
```



Aktuell:
- Verwendung (Kopie) der Inhalte aus openPASS v0.6 als Funktionsaufruf
- Keine Komponenten-erstellung im *dynamic-AgentTypeGenerator*

Zukünftig:
- Action Execution auch außerhalb vom Driver-Model denkbar