CIRRUS BY PANASONIC V2X

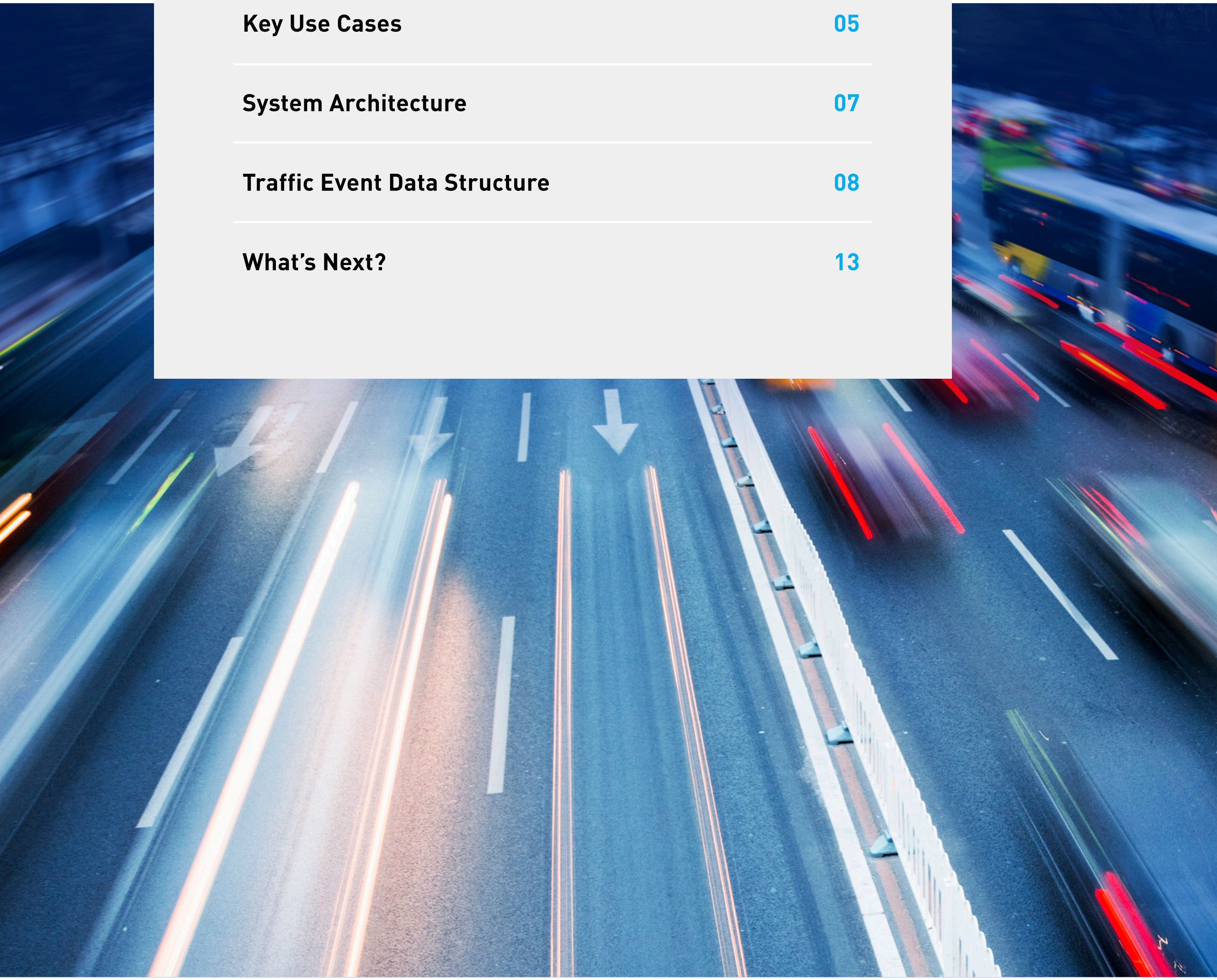
Panasonic

Delivering Cross Platform Driver Warnings at Scale

PREPARED BY

Lauren Cordova, CSEP and Kjeld Lindsted

Table of Contents



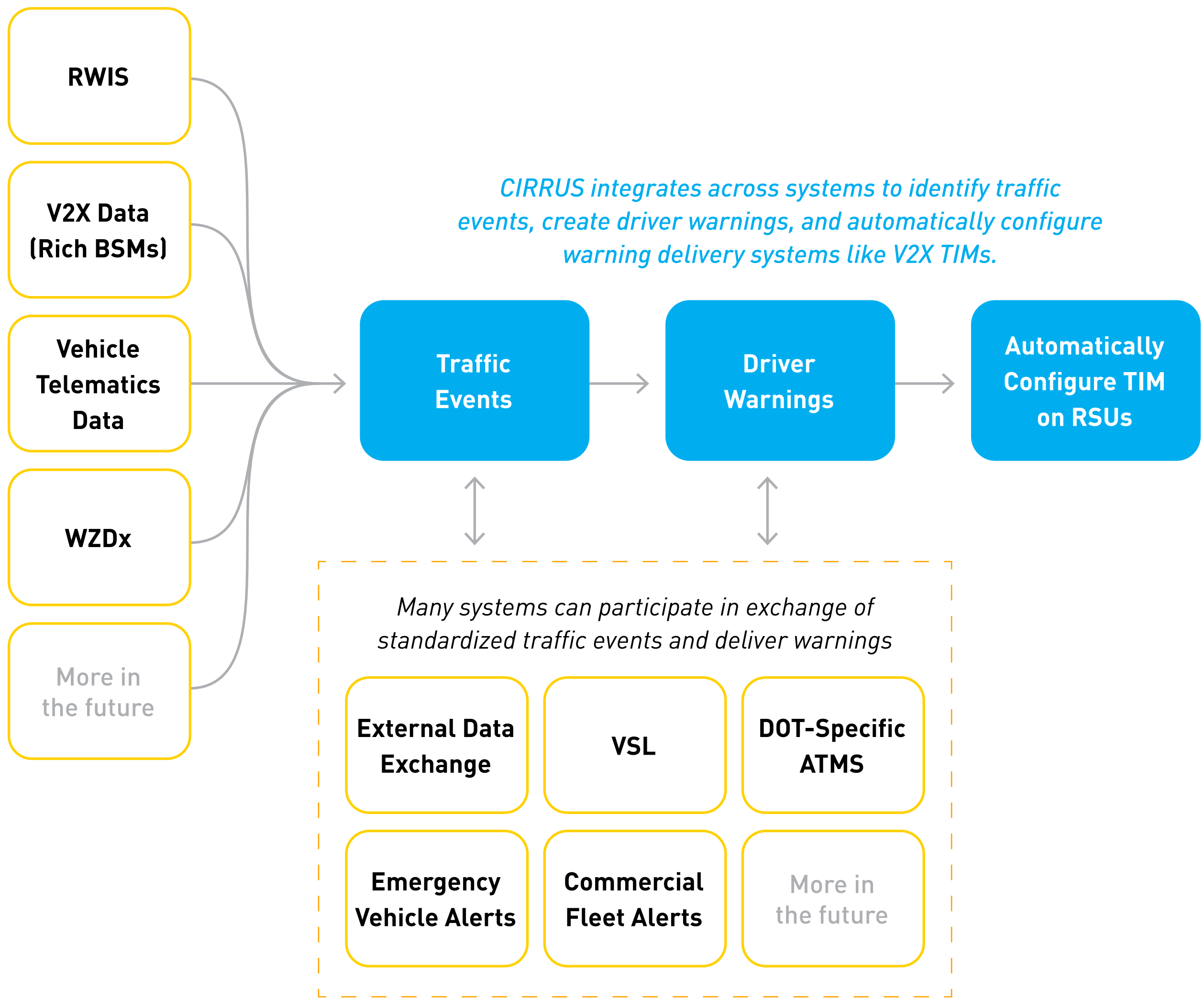
Delivering Cross Platform Driver Warnings at Scale

There are many different mechanisms and approaches to disseminating driver warnings, which can vary both in terms of available technologies, such as physical road signs or in-vehicle systems like V2X, as well as jurisdictional regulations on parameters such as distance in advance and contents of warnings. Because of these factors that can impact the way driver warnings are managed, a scalable, interoperable solution is the most effective approach to communicate warnings in a way that allows users to customize implementation for regional factors. A similar argument was described for identification of traffic events in a prior white paper, and the technical approach described below builds upon that interoperable architecture in issuing driver warnings from either events detected or other sources.

"A scalable, interoperable solution is the most effective approach to communicate warnings in a way that allows users to customize implementation for regional factors."

Key Use Cases

Utah, Wyoming, and Colorado have proactively engaged in cross-state driver warning applications, leveraging J2735 TIMs as a foundation for data sharing. All three states have deployed V2X RSUs and OBUs to transmit and display TIMs with 5.9GHz technology. Furthermore, Trihydro's SDX platform offers an integration to distribute TIMs via SiriusXM to reach additional vehicles that don't have V2X technology onboard. Finally, to extend communication to as many drivers as possible, other integrations are being considered including physical road signs in use cases such as VSL or VRU safety as well as non-V2X in vehicle alerting systems (AVL, HAAS Alert, MakeWay, etc.). Because requirements for such diverse warning dissemination varies system to system, an extensible architecture is essential to support cross-platform scale.



WZDx	Workzone Data Exchange	**TIM**	Traveler Information Message
ATMS	Advanced Traffic Management System	**BSM**	Basic Safety Message
RWIS	Road Weather Information Station	**RSU**	Roadside Unit
VSL	Variable Speed Limit		

Figure 1: Key components of the CIRRUS platform.

System Architecture

The diagram (left) shows key components of the CIRRUS by Panasonic platform, with integration points for scaling insights and impact through collaboration and partnership with many different industry initiatives. The traffic event feature manages all types of events in a common data structure and API. Events are then converted into Driver Warnings based on defined logic per event type. Pre-processed warnings may also be received from external systems, and applicable warnings are converted into a J2735 TIM configured onto the relevant RSUs based on factors such as location.

See Figure 1.

Like the two key assumptions used in developing Panasonic's traffic event architecture, the same methodology is extended to driver warnings. First, that there is value in all warning types being in a single, common data structure; and second, that there are an unknown number of data elements both for current warning types and certainly for new warning types that may be identified in the future, yet there is a desire to accommodate these new data elements while minimizing breaking changes. Therefore, the architecture proposed for driver warnings follows the same principles as the traffic event structure to deliver common high-level fields and include an extensible field for customized data element capture per warning.

Traffic Event Data Structure

Panasonic proposes the below structure to support both scale and advanced analytics over time. First, implement the known fields that are going to be necessary for any warning type. Just like with traffic events, this is going to be a unique identifier, source, type, location, time (start/end), and version.

Next, implement a customizable field that allows for metadata capture which may vary over time or by warning type. If the data is captured at this level, downstream processing can parse out metadata elements for tailored analyses, such as analysis of weather specific warnings/fields, without requiring any changes to the upstream primary data structure. Even if warnings implement new metadata fields with new versions over time, capturing the version, source, and metadata fields make the analytics straightforward with simple if/then logic statements. Users (or downstream applications) can then leverage the data and conduct studies over changing architecture without the hard limitations that generally come from breaking changes

*Below is the data structure — **FIELD** / **DESCRIPTION** — CIRRUS has deployed.*

warning_id TEXT
A system-generated unique ID for the warning. The warning_id can be used to locate all records associated with a single warning, in the case warnings are updated (e.g. extended) after the original warning is issued.

---

source TEXT
This element reports the organization or source identifying the event.

---

warning_type TEXT
The type of warning being issued (i.e. rain, snow, ice, crash ahead, etc.).

---

geojson JSON
Contains a JSON data structure that represents a geographic element, such as a Point or a Polygon. A generic GeoJSON data structure is comprised of a 'type' field which indicates what kind of geographic element it represents, and a 'coordinates' field, which can be a two-item list containing lat/long coordinates for a Point element, or a list of two item lists of lat/long coordinates which form a Polygon.

---

warning_metadata JSON
This element is a JSON field that provides metadata relevant to the warning_type, source, and version. **This may vary over time and is available for customization as new data elements are desired**. Because new or changing key/value pairs in this JSON structure may be implemented without a top-level architecture change, this field is key to unlocking efficient scale.

---

tmstp_utc TIMESTAMP
This is the timestamp when the warning record is received by the system.

---

start_tmstp_utc TIMESTAMP
This is the timestamp when the warning is first active.

---

end_tmstp_utc TIMESTAMP
This is the timestamp when the warning is scheduled to end.

---

version TEXT
The version of the logic reporting the warning, which may be used in historical analyses to parse data elements or evaluate effectiveness of updates to the logic over time, among other use cases.

To configure a TIM on a specific RSU when a warning is identified, CIRRUS populates the required metadata when warnings are received that are within pre-defined distances from deployed RSUs integrated to CIRRUS. The below fields are captured in the warning metadata and used in the CIRRUS application to configure the TIM on the RSUs identified.

*Below fields — **KEY** / **VALUE DESCRIPTION & SOURCE** — are captured in the warning_metadata and used in the CIRRUS application to configured the TIM on the RSUs identified.*

device_ids ARRAY
This metadata field consists of an array of RSU IDs to configure for the identified warning.

broadcast_start_utc TIMESTAMP
This is the timestamp the RSU should begin broadcasting the TIM associated with the driver warning (i.e. rsuSRMDeliveryStart per 4.1 spec). In V1, broadcast start will be set to the same as the start_tmstp_utc field above.

broadcast_end_utc TIMESTAMP
This is the timestamp the RSU should end broadcasting the TIM associated with the driver warning (i.e. rsuSRMDeliveryStop per 4.1 spec). In V1, broadcast end will be set to the same as the end_tmstp_utc field above.

broadcast_rate INTEGER32
This is the transmission rate the RSU should broadcast the TIM associated with the driver warning (i.e. rsuSRMTxInterval per 4.1 spec). In V1, broadcast rate will be set to 1Hz.

tim_raw TEXT
The **encode**d TIM (based on 2020 J2735 standard) associated with the warning. The TIM data elements will be populated per the below and encoded for use in downstream applications (rsuSRMPayload per 4.1 spec).

tim_decoded JSON
The decoded TIM (based on 2020 J2735 standard) captured as JSON for downstream analysis and/or systems that are unable to decode the raw TIM.

event_id TEXT
This is the event_id of the event that generates the warning, if the warning is sourced from an event upstream in the event data pipeline.

version TEXT
This is the warning version associated with the data record.


"CIRRUS believes this extensible, optimized architecture will provide value today and many years into the future."

What's Next

CIRRUS by Panasonic has deployed its architecture in multiple partnership applications, most notably with the Utah Department of Transportation (UDOT). UDOT has created a data community where users can access this data and provide feedback. CIRRUS believes this extensible, optimized architecture will provide value today and many years into the future through scale and expansion, and we welcome all user input as we continue to improve.

We invite others to join the movement to advance traffic event detection and analysis, and smooth impacts for everyone on the road. To join the community conversation and access the data, please visit **https://mobility.na.panasonic.com/contact**.

About CIRRUS by Panasonic

CIRRUS by Panasonic is a division of the Panasonic Corporation of North America's Smart Mobility Office. The CIRRUS platform enables instant and safe communication between vehicles, infrastructure, intersections, and the operations teams who manage them.

Formed in 2017, CIRRUS offers connected vehicle data and applications at scale and in the cloud. Our data collection and insights platform allows us to deliver unique intersection-specific insights, including data interaction replay (time travel). We incorporate input from edge devices, external systems, and cloud products for a scalable, complete, end-to-end solution. Learn more at **https://mobility.na.panasonic.com/cirrus**.

About the Authors

LAUREN CORDOVA, CSEP

Lauren is the Head of Technology for CIRRUS by Panasonic, leading data, analytics, cloud, and edge engineering functions to deliver advanced transportation solutions for government and fleet markets. Lauren's deep expertise in data architecture design have led to several industry-first approaches to managing connected vehicle data at scale with several patents pending in this industry. As a mechanical, systems, and data engineer, Lauren has 20 years of experience developing technology for NASA, medical, defense, energy, and transportation solutions.

KJELD LINDSTED

Kjeld is the Head of Product and Sales for CIRRUS by Panasonic, responsible for establishing CIRRUS requirements and for partnership development. Kjeld started his career in transportation technology with the City of Redlands, founding Smart Redlands to scale technology implementation in public works programs. In the years since, Kjeld has led connected vehicle programs for startups, government, and fortunes 500s and has more than 20 years expertise guiding teams through market changing innovations.