

CAP Data Layer Guidelines

Jacob Westfall
jake@jpw.biz

The following guidelines on using data layers with CAP were developed based on my experience creating numerous layers for various CAP systems. Version 1.1 of CAP is assumed throughout this document and included is an example layer and sample message. These guidelines are still a work in progress and any comments are certainly welcome.

What is a Layer?

A CAP data layer is formed from the optional elements from the CAP standard. The use, formatting, and processing actions of these elements is outlined in a layer specification. Layers are used to increase the information processing capabilities of CAP systems.

The two primary rules for creating CAP data layers are:

- A layer must adhere to the CAP standard. No usage, formatting, or processing functions should deviate from what is acceptable under this standard.
- Layers must co-exist with other layers. No usage, formatting, or processing functions should be present that could cause conflicts to arise or prevent other layers from being added to a message.

Consider a pizza as an analogy. You start with the crust (CAP), the foundation upon which your pizza is made. Toppings (layers) are then added to your pizza. The cheese, sauce, pepperoni, and mushroom toppings are all added on top of the crust and on top of each other to create the ideal pizza satisfying everyone (interoperable). When the pizza gets delivered you might like it just the way its made, but your friend doesn't like ham so he picks off the ham (filters layer elements) and eats what remains. The ability to customize and account for different needs, while still adhering to standards, is key to creating a good pizza and a good layer.

Defining a Layer Specification

New layers should be defined by creating a layer specification document to be made available to all those who are expected to use this layer. Coordination and re-use of layers between similar agencies and purposes should be encouraged. In addition to the spec document, some examples (normal use and special cases) should be published, and if possible an XML schema that can be used for layer validation. Here are some suggested sections in a layer specification:

- ◆ Introduction

Outlines the purpose of the layer, its intended use, and audience. Ownership and maintenance of the layer should also be addressed here as well as any references.

- ◆ Interpretation

There are two types of layer interpretations, strict and loose. A strict layer spec needs all required elements to be present. The layer will fail validation otherwise. Caution must be taken when creating strict layers to ensure that message creators receive notification of validation failures. Messages that fail should not be discarded, but set aside as exceptions for follow-up processing to prevent otherwise valid messages containing potentially important information from being lost.

Loose interpretations allow for missing layer elements by providing a path for functionality degradation based on what is or isn't present in the message.

The layer interpretation section should define what type this new layer is and how validation problems are addressed.

- ◆ Elements

It can be argued that a layer should be defined for any messages that use elements beyond those required to form a valid CAP message. However the most common elements used in defining a layer are addresses, code, eventCode, parameter and geocode.

Each optional element that forms part of the layer should be listed with the following characteristics defined:

- Use – How is this element used by the layer? Is it required, recommended, or optional? Should it be used only in special cases or for all messages created using this layer?
- Format – In what format should the element values appear? If there is a list of values for this element, they should be shown. Does the format change depending on any special cases?
- Processing – What processing takes place, both on the creating and receiving systems based on this element?

◆ Limitations and Compatibility Concerns

Quoting the CAP Cookbook, "Be strict about what you send, and tolerant about what you receive.", receiving CAP systems should not impose any limitations or restrictions on a CAP message through the use of a layer. However there may be cases where there are limitations on the functionality supported by a creation system, especially for those layers being used to translate from legacy formats, and they should be defined here. How do these limitations impact on the messages and the layer in general? What compatibility workarounds need to be put in place? Can any conflicts arise between the limitations, the layer, and the CAP spec?

Are there any compatibility concerns with this layer, perhaps in relation to another layer? For example, another layer may use an identical element and value list and so it could act as a substitute for an element in this layer or potentially cause a problem.

◆ Implementation Notes

Additional notes regarding implementation of this layer. Perhaps notes on processing, special use cases, and any other information.

Layer Creation Recommendations

- ◆ Avoid embedding data in text fields

Strongly avoid putting any data in the free form text elements such as source, note, event, description, and instruction. Don't try to embed data or require some form of text parsing, instead use the elements that are provided for this purpose, code and parameter. For example, don't add embedded data like "Wind=North" to the bottom of a description element, use a parameter instead.

- ◆ Use an appropriate CAP element instead of custom parameters

Always try to find an existing CAP element to hold some data before creating a custom parameter. For example, instead of creating a custom parameter called "PhoneNumber" try using the contact element instead.

- ◆ Use a geo-spatial value in conjunction with all geocodes

Geocodes are a useful element for filtering and processing, however they are dependent on a receiving system being aware of the code list and are deprecated in the CAP standard. To ensure that you are creating interoperable messages that can be used by receiving systems that are not aware of your layer, always include a geo-spatial value with a geocode.

- ◆ Avoid overloading the message with limited use parameters

Custom parameters should be of value for all receivers and not be too specific to any one system, they should add value to the message. For example, a weather message with the wind speed during a storm as a parameter is useful to many systems. A parameter that holds a sync code for the wind measurement device, meant to be read by a single receiving system is not. Instead consider using an XML namespace extension and a custom schema if you must include this data with a CAP message.

Border Crossing Wait Times CAP Data Layer

Author: Jacob Westfall

Updated: Aug 16, 2008

Version: 1.1

Introduction

The Border Crossing Wait Times data layer is used for CAP messages that are created to warn users about delays at MEX/USA/CDN border crossings. Wait times published by the US Border Patrol are used and alerts are available for public distribution. Wait times are categorized by trucks/commercial traffic versus cars/passenger vehicles and the destination either into or out of the US. Jacob Westfall is responsible for the creation and updating of this layer specification.

Interpretation

A loose interpretation of this layer is used for validation. Both the eventCode and geocode elements are required for proper operation but the layer does degrade. If only the geocode is available this indicates that all eventCodes are deemed applicable. If no geocode is available, any geo-spatial values are to be used, and if none of these are available, then all geocodes are deemed applicable.

Elements

code

- Use: the use of code value is recommended since it denotes the layer type and version in use for this message
- Format: the format of the code value should be the short form name of this layer BCW and the version number, ex. "BCW=1.1"
- Processing: receiving systems upon seeing this code should load the appropriate layer validator and look for other layer elements

eventCode

- Use: the use of eventCode is required to indicate which category of traffic is delayed
- Format: the valueName for this element should be "BCW". The possible values are:
 - TravUSA – cars or travelers heading to USA
 - TravMEX – cars or travelers heading to Mexico
 - TravCDN – cars or travelers heading to Canada
 - ComUSA – trucks or commercial heading to USA
 - ComMEX – trucks or commercial heading to Mexico
 - ComCDN – trucks or commercial heading to Canada
- Processing: receiving systems can use this element for filtering or display values

parameter

- Use: this custom parameter is optional and indicates the reported delay time
- Format: the value for this element should be the wait time in minutes. The possible valueNames are:
 - BCW-TU – cars or travelers heading to USA
 - BCW-TM – cars or travelers heading to Mexico
 - BCW-TC – cars or travelers heading to Canada
 - BCW-CU – trucks or commercial heading to USA
 - BCW-CM – trucks or commercial heading to Mexico
 - BCW-CC – trucks or commercial heading to Canada
- Processing: receiving systems can use this element for filtering or display values

geocode

- Use: the use of geocode is required to indicate which border crossing is experiencing the delay in this message. Only one geocode should be used in an area block along with the corresponding geo-spatial value.
- Format: the valueName for this element should be "BCW". The possible values are:
 - Alexandria Bay - 1
 - Pacific Highway - 2
 - Peach Arch – 3
 - Lewiston Bridge – 4
 - Peace Bridge – 5
 - Whirlpool Bridge – 6
 - Ferry Point – 7
 - Milltown – 8
 - Champlain – 9
 - Derby Line – 10
 - Ambassador Bridge – 11

Windsor Tunnel – 12
Highgate Springs – 13
Houlton – 14
Jackman – 15
Norton – 16
Pembina – 17
Port Huron – 18
Sault Ste Marie – 19
Sumas – 20
Sweetgrass – 21

Andrade – 22
B&M – 23
Gateway – 24
Los Indios – 25
Veterans International – 26
Calexico East – 27
Calexico West – 28
Columbus – 29
Del Rio – 30
Douglas – 31
Eagle Pass I – 32
Eagle Pass II – 33
BOTA – 34
PDN – 35
Ysleta – 36
Fabens – 37
Fort Hancock – 38
Hidalgo – 39
Pharr – 40
Laredo I – 41
Laredo II – 42
Colombia Solidarity – 43
World Trade Bridge – 44
Lukeville – 45
Naco – 46
Deconcini – 47
Mariposa – 48
Morley Gate – 49
Otay Mesa Commercial – 50
Otay Mesa Passenger – 51
Presidio – 52
Progreso – 53
Rio Grande City – 54
Roma – 55
San Luis – 56

San Ysidro – 57
Santa Teresa – 58
Stanton DCL – 59
Tecate – 60

- Processing: receiving systems can use this element for filtering or display values

Limitations and Compatibility Concerns

None

Implementation Notes

- It is recommended that if a message is regarding multiple crossings, one crossing per info block should be used. This allows for a direct correlation between all elements within that info block and there is no confusion about which type of traffic for which crossing is delayed.
- It is recommended that if a message has multiple languages, individual info blocks per language should be used and the CAP language element be added. If the language element is missing then all info blocks will be considered the CAP default of en-US and processed accordingly.

Sample Message

```
<?xml version="1.0" encoding="UTF-8"?>
<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
  <identifier>BCW-0001</identifier>
  <sender>bcw@example.com</sender>
  <sent>2008-08-16T12:00:00-00:00</sent>
  <status>Actual</status>
  <msgType>Alert</msgType>
  <scope>Public</scope>
  <code>BCW=1.1</code>
  <info>
    <category>Transport</category>
    <event>Border Crossing Delay</event>
    <urgency>Expected</urgency>
    <severity>Moderate</severity>
    <certainty>Likely</certainty>
    <eventCode>
      <valueName>BCW</valueName>
      <value>TravMex</value>
    </eventCode>
    <headline>El Paso Bridge of the Americas border crossing
delay reported for travelers to Mexico</headline>
    <parameter>
      <valueName>BCW-TM</valueName>
      <value>60</value>
    </parameter>
    <area>
      <areaDesc>El Paso Bridge of the Americas</areaDesc>
      <geocode>
        <valueName>BCW</valueName>
        <value>34</value>
      </geocode>
      <polygon>31.769222526451045,-106.45151138305664
31.760830462497644,-106.45275592803955 31.76053855092474,-
106.45095348358154 31.768930641350558,-106.44992351531982
31.769222526451045,-106.45151138305664</polygon>
    </area>
  </info>
</alert>
```