Exceeded Your Validation Cost Budget? Now What?

Let's take a look at CEL cost analysis

Joel Speed

Principal Software Engineer

Red Hat

What we'll discuss today

- What is CEL and why would you want to use it?
- What and why is a validation cost budget?
- Rule Costs and Cardinality
- Avoiding high validations costs in your CRDs

Joel Speed - @JoelASpeed

Principal Software Engineer and Team Lead

Focusing on cloud integrations (CAPI, CCM)

API reviewer for OpenShift

Kube CEL contributor

What is CEL?

And why do we care?

Common Expression Language (CEL)

By Google

The Common Expression Language (CEL) implements common semantics for expression evaluation, enabling different applications to more easily interoperate.

- C-like and non-turing complete
- Lightweight and fast execution
- Extensible

```
account.balance >= transaction.withdrawal
    || (account.overdraftProtection
    && account.overdraftLimit >= transaction.withdrawal  - account.balance)
```


You can play with CEL using the CEL playground at https://playcel.undistro.io/

And CEL is used in Kube?

For CRD Validation...

Introduced in KEP-2876, CRD X-Kubernetes-Validations:

- Self-Contained CRDs
- Simplify CRD development
- First introduced as beta in K8s 1.25, GA in 1.29

```
immutableField:
  description: immutableField is a field that is immutable once the object has been created.
  type: string
  x-kubernetes-validations:
  - rule: self == oldSelf
    message: immutableField is immutable
```


You can read more about CEL in Kube in the docs https://kubernetes.io/docs/reference/using-api/cel/

And CEL is used in Kube?

... and Admission Control

Introduced in KEP-3488, ValidatingAdmissionPolicy:

- Add policy to any kind in Kube
- An in-tree alternative to webhooks
- First introduced as beta in K8s 1.28, GA in 1.30

```
validations:
  # all requests should have a node-name claim, this prevents impersonation of the SA.
  - expression: "has(request.userInfo.extra) && ('authentication.kubernetes.io/node-name' in
request.userInfo.extra)"
    message: "this user must have a \"authentication.kubernetes.io/node-name\" claim"
  # all requests should originate from the CNM owner's node
  - expression: "object.metadata.name ==
request.userInfo.extra[\"authentication.kubernetes.io/node-name\"][0]"
    messageExpression: "'updates to Node ' + string(object.metadata.name) + ' may only be effected
from the cloud node manager running on the same node'"
```


You can read more about CEL in Kube in the docs https://kubernetes.io/docs/reference/using-api/cel/

Great, so what else can it do?

Some more interesting examples

- Validate co-dependency between fields

```
has(self.type) && self.type == 'RequiredMember' ?  has(self.requiredMember) : !has(self.requiredMember)
```

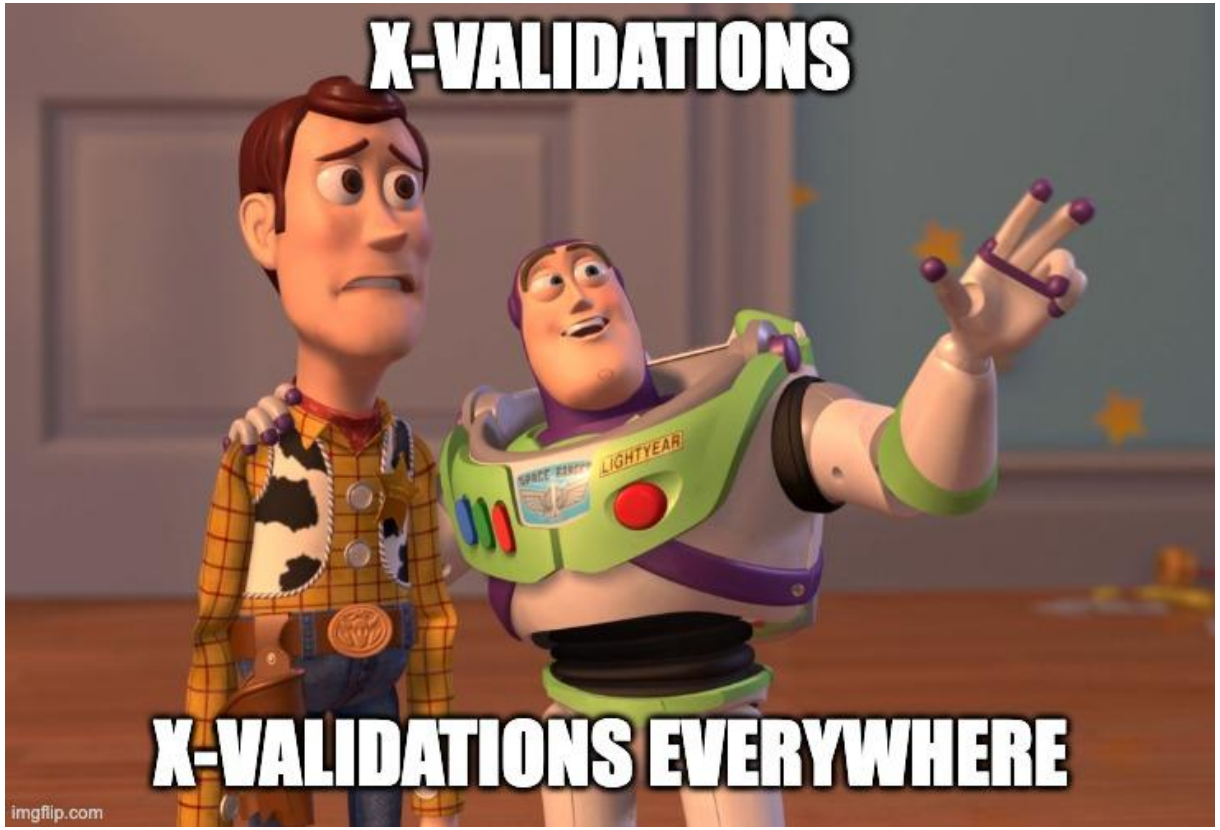
- Validate a URL is secure `isURL(self) && url(self).getScheme() == 'https'`
- Check that two IPs are distinct IP families

```
size(self) == 2 && isIP(self[0]) && isIP(self[1]) ? ip(self[0]).family() != ip(self[1]).family() : true
```

- Items cannot be removed from a list `oldSelf.all(e, (e in self))`
- Check a quantity is at least 8Gi `isQuantity(self) && quantity(self).isGreaterThan(quantity('8Gi'))`

Sounds great! Let’s validate all the things!

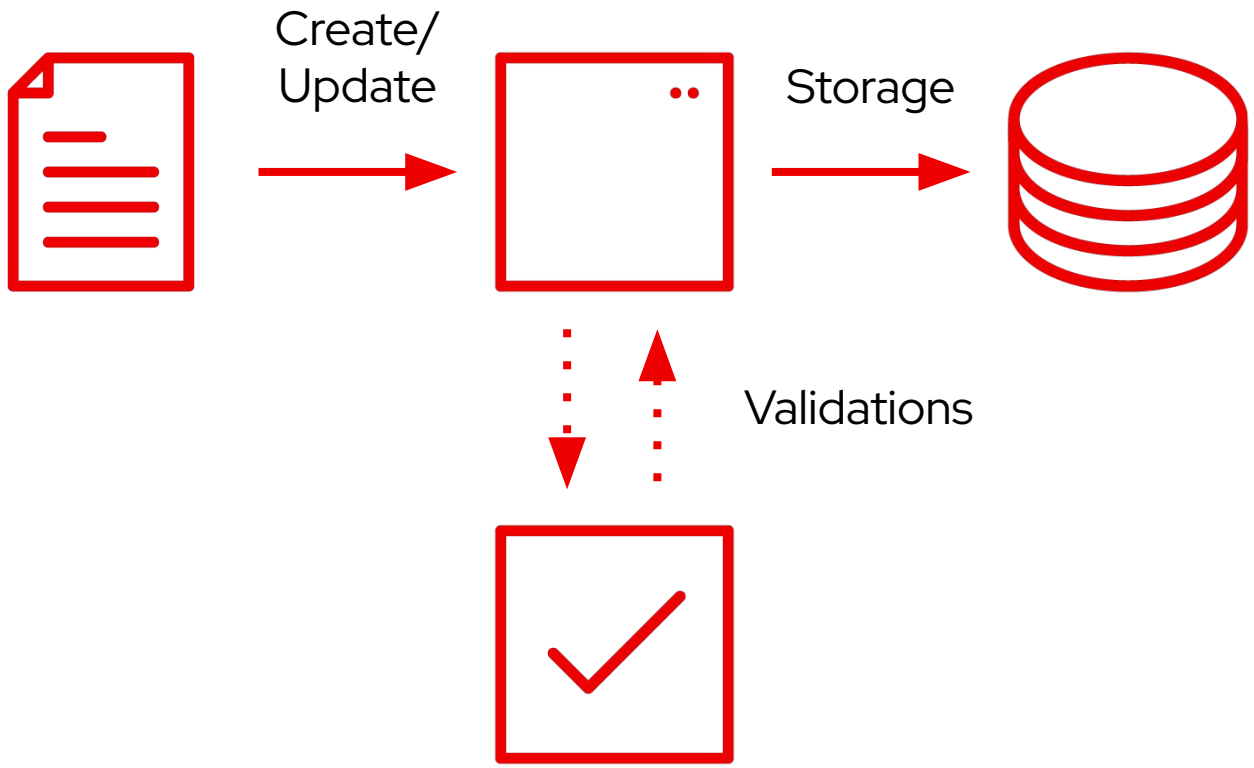
There’s always a catch



Red Hat

These validations are not free...

Unfortunately



Validations are synchronous

All validations are executed as part of the admission flow, during create an update operations on resources

Red Hat

Resource Constraints and execution safety

Preserving the API server

Estimated Cost

At CRD admission, calculate the worst case runtime cost of a validation.

Runtime Cost

At CR admission time, count the cost and error if the runtime cost budget is exceeded.

```
self.matches('^(((?:[a-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0-9-]*[a-zA-Z0-9])(?:\\.(?:[a
-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0-9-]*[a-zA-Z0-9]))+(?::[0-9]+)?)|(localhost(?::[0-
9]+)?))(?:(?:/[a-z0-9]+(?:(?:(?:[._]|__|[-]*)[a-z0-9]+)+)?)+)?$|^(((?:[a-zA-Z0-
9]|[a-zA-Z0-9][a-zA-Z0-9-]*[a-zA-Z0-9])(?:\\.(?:[a-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0
-9-]*[a-zA-Z0-9]))+(?::[0-9]+)?)|(localhost(?::[0-9]+)?))(?:(?:/[a-z0-9]+(?:(?:
(?:[._]|__|[-]*)[a-z0-9]+)+)?)+)?$')
```

```
spec.validation.openAPIV3Schema.properties[spec].properties[scopes].items.x-kub
ernetes-validations[0].rule: Forbidden: estimated rule cost exceeds budget by
factor of more than 100x (try simplifying the rule, or adding maxItems,
maxProperties, and maxLength where arrays, maps, and strings are declared)
```

Uh oh

This validation is too complex.

The aim was to validate an image specification.

The API server rejected the CRD update.

So how does the API server calculate the cost?

A deeper look at estimated costs

Per call and per resource limits

Important numbers

10,000,000

The “per rule” cost limit.

100,000,000

The total cost limit, per custom resource or validating admission policy as a whole.

Source:
https://github.com/kubernetes/apiextensions-apiserver/blob/67f4a1dade1d801f1c7c89298137c6d189807f08/pkg/apis/apiextensions/validation/validation.go#L57-L60

Two factors of the calculation

Rule Cost x Cardinality = Total Cost

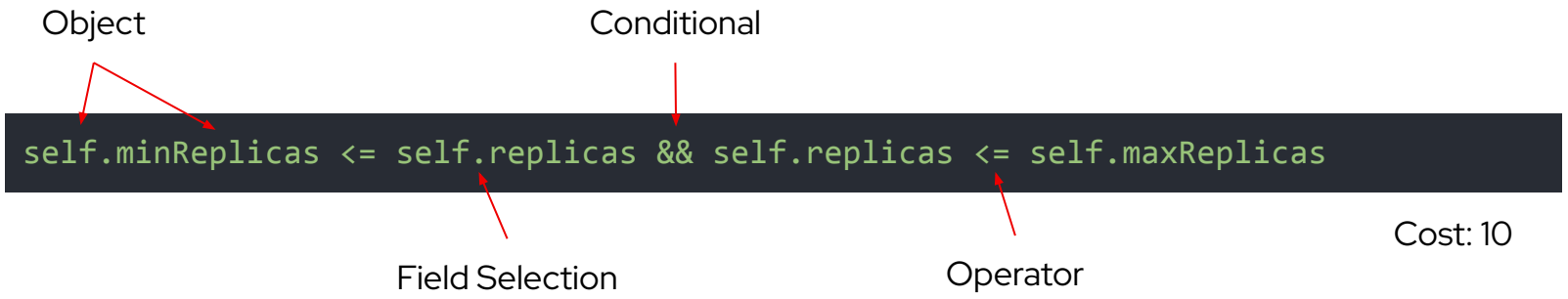
Rule Cost

What is the cost of executing the rule once.

Cardinality

How many times is the rule executed.

Source: https://github.com/kubernetes/kubernetes/blob/master/staging/src/k8s.io/apiserver/pkg/apis/cel/config.go



Rules have objects, functions, conditionals and operators

Accessing objects and simple operators and conditionals have fixed costs.

Functions get more complicated.

Some functions create objects.

Red Hat

```
self: kubecon-cloudnativecon-north-america.events.linuxfoundation.org
```

```
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*$')
```

$$\lceil (\text{len(self)} + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (63 + 1) \times 0.1 \rceil \times \lceil 64 \times 0.25 \rceil + 1 = 113$$

Proportional Cost Functions

String operations such as ‘contains’, ‘matches’, ‘startsWith’, ‘endsWith’.

Source: https://github.com/google/cel-go/blob/0213a8eb4051448f7e03f8283c4a2b511ad23eae/checker/cost.go#L592-L606

```
self:
- kubecon-cloudnativecon-north-america.events.linuxfoundation.org
- kubecon-cloudnativecon-india.events.linuxfoundation.org
- kubecon-cloudnativecon-europe.events.linuxfoundation.org
```

```
self.all(x,
  x.matches(
    '^[a-z0-9]([-a-z0-9]*[a-z0-9])?(.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*$'
  )
)
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

116 + 100 + 100 + 2 = 318

Iterations are a product of length by rule cost

Iterators such as ‘all’, ‘exists’, ‘exists_one’.

Map and Filter

‘map’ and ‘filter’ have similar time complexity, but also allocate space.

But Joel, all of these costs are really small?

```
self:
- <string>
- <string>
- <string>
- ...
```

```
self.all(x,
  x.matches(
    '^[a-z0-9]([-a-z0-9]*[a-z0-9])?(.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*$'
  )
)
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

??? × (??? + 3) + 2 = ???

At cost estimation

The actual size of the values is unknown.

It must be estimated.

```
self: <string>
```

```
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*$')
```

$$\lceil (\text{len(self)} + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (??? + 1) \times 0.1 \rceil \times \lceil 64 \times 0.25 \rceil + 1 = ???$$

For the string case

Without knowing the length of the string, how do we compute the size?

```
myString:
  description: myString is …
  type: string
  maxLength: 256
  x-kubernetes-validations:
  - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*
/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

$$\lceil (\text{len(self)} \times 4 + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (256 \times 4 + 1) \times 0.11 \times \lceil 109 \times 0.25 \rceil + 1 = 2885$$

At CRD admission

The API server inspects the schema, and uses the ‘maxLength’ property of the field to estimate the worst case cost.

Source: https://github.com/kubernetes/apiserver/blob/1ea7befac323763d28173ed613a9e73ef1ef9c16/pkg/cel/common/schemas.go#L162-L166

```
myString:
  description: myString is …
  type: string
  maxLength: ???
  x-kubernetes-validations:
  - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*
/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

$$\lceil (\text{len(self)} + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (??? \times 4 + 1) \times 0.11 \rceil \times \lceil 109 \times 0.25 \rceil + 1 = ???$$

What about when maxLength is unset?

The API server must make a worst case estimate for the basis of the calculation.

```
myString:
  description: myString is ...
  type: string
  maxLength: 786431.5 (3145726 bytes)
  x-kubernetes-validations:
  - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*
/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

$$\lceil (\text{len(self in bytes)} + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (3145726 + 1) \times 0.11 \times \lceil 109 \times 0.25 \rceil + 1 = 8{,}808{,}045$$

Fallback to the max request size

When the length of the string is unset, estimate it as (3 × 1024 × 1024) - 2

The quotes are excluded, hence -2.

This time the size is in bytes, remove the rune conversion factor.

Source: https://github.com/kubernetes/apiserver/blob/0265ad61bc30c07af2b35dcba65990b8946d73e5/pkg/cel/common/schemas.go#L240-L242

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/((
[A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

??? × (2885 + 3) + 2 = ???

Looking at the list again

This time we have a length limit on the string.

So we know the per step cost is 2885.

```
myListOfString:
  description: myListOfString is ...
  maxItems: 1024
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/((
[A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

1024 × (2885 + 3) + 2 = 2,957,314

Adding maxItems

Adding the maxItems property allows an effective worst case estimate.

```
myListOfString:
  description: myListOfString is …
  maxItems: ???
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/((
[A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

??? × (2885 + 3) + 2 = ???

Without maxItems

Without the maxItems we again cannot calculate the worst case cost.

```
myListOfString:
  description: myListOfString is ...
  maxItems: 1048575
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/((
[A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

len(self) × (<per step cost> + <loop cost>) + <cost of args>

1048575 × (2885 + 3) + 2 = 3,028,284,602

Fallback to estimated maximum list items

This time, estimate the maximum number of items based on the maximum number of copies of the minimum serialised object.

((3×1024×1024) - 2) ÷ (<min size> + 1)

A minimum serialised string is “”.

```
myListOfString:
  description: myListOfString is ...
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.key.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Z
a-z0-9])$'))
  items:
    type: object
    properties:
      key:
        description: key is a unique identifier for the list map entry
        type: string
        maxLength: 256
      value:
        description: value is the value for the key
        type: string
        maxLength: 256
      required:
      - key
      - value
```

<min serialised object size> + <required fields min size>

for each required field: len(name) + <min serialised size> + 4

2 + (3 + 2 + 4) + (5 + 2 + 4) = 22

Minimum Size for objects

The minimum size for an object is calculated taking into account all required fields and their minimum serialised size.

An object with no required fields would serialise to '{}'.

Source: https://github.com/kubernetes/apiserver/blob/0265ad61bc30c07af2b35dcba65990b8946d73e5/pkg/cel/common/schemas.go#L90-L135

CEL Cost Calculations

```
myListOfString:
  description: myListOfString is ...
  maxItems: 136770
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.key.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Z
a-z0-9])$'))
  items:
    type: object
    properties:
      key:
        description: key is a unique identifier for the list map entry
        type: string
        maxLength: 256
      value:
        description: value is the value for the key
        type: string
        maxLength: 256
      required:
      - key
      - value
```

len(self) × (<per step cost>+ <loop cost>) + <cost of args>

136770 × (2886 + 3) + 2 = 395,128,532

Estimated min size is greater

Since the estimated min size for the object is greater than the string, the estimated worst case of the number of list items is much lower.

395 million vs 3 billion

N.B. The per step cost here is one more than before to account for the additional field selection in x.key.matches vs x.matches before

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  items:
    type: string
    maxLength: 256
    x-kubernetes-validations:
    - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*
/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

$$\lceil (\text{len(self)} \times 4 + 1) \times \langle \text{string traversal cost factor} \rangle \rceil \times \lceil \text{len}(\langle \text{regex} \rangle) \times \langle \text{regex traversal cost factor} \rangle \rceil + \langle \text{cost of args} \rangle$$

$$\lceil (256 \times 4 + 1) \times 0.1 \rceil \times \lceil 109 \times 0.25 \rceil + 1 = 2885$$

But what about moving the validation into the items?

This time the validation is executed at the item level, rather than using the ‘self.all’ semantic at the list level.

We saw this rule earlier and the cost was 2885.

That was smart, but now it is time to talk cardinality

How many times will a rule execute

A String Validation

What's the cardinality?

```
openAPIV3Schema:
  description: "TheKind is for testing."
  type: object
  properties:
    myString:
      description: myString is ...
      type: string
      x-kubernetes-validations:
      - rule: ... <-- What is my cardinality?
```

A List Validation

What's the cardinality?

```
openAPIV3Schema:
  description: "TheKind is for testing."
  type: object
  properties:
    myListOfStrings:
      type: array
      maxItems: 1000
      x-kubernetes-validations:
      - rule: ... <-- What is my cardinality?
      items:
        type: string
```

A String Validation Within A List

What's the cardinality?

```
openAPIV3Schema:
  description: "TheKind is for testing."
  type: object
  properties:
    myListOfStrings:
      type: array
      maxItems: 1000
      items:
        type: string
        x-kubernetes-validations:
        - rule: ... <-- What is my cardinality?
```

A String Validation Within A List #2

What's the cardinality?

```
openAPIV3Schema:
  description: "TheKind is for testing."
  type: object
  properties:
    myListOfStrings:
      type: array
      items:
        type: string
        X-kubernetes-validations:
        - rule: ... <-- What is my cardinality?
```

```
myListOfString:
  description: myListOfString is ...
  maxItems: 1048576
  type: array
  x-kubernetes-list-type: atomic
  items:
    type: string
    maxLength: 256
    x-kubernetes-validations:
    - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*
/(([A-Za-z0-9][-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

Rule Cost: ⌈(256 × 4 + 1) × 0.11×⌈109 × 0.25⌉ + 1 = 2885

Total Cost: 1048576 × 2885 = 3,025,141,760

When the rule has unbounded cardinality

This time calculate the maximum possible times the rule could execute, and multiple by the rule cost.

(3×1024×1024) ÷ (<min size> + 1)

Note this formula is slightly different to before.

Source: https://github.com/kubernetes/apiextensions-apiserver/blob/a1e0ff9923b68731a7735f1c4168b6b3d0bd1027/pkg/apiserver/schema/cel/compilation.go#L351-L360

Which one do you prefer?

These validations achieve the same thing

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  items:
    type: string
    maxLength: 256
    x-kubernetes-validations:
    - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\
\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][
-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[
a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][-A-
Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

Which one do you prefer?

These validations achieve the same thing

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  items:
    type: string
    maxLength: 256
    x-kubernetes-validations:
    - rule:
self.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\
\.[a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][
-A-Za-z0-9_.]*)?[A-Za-z0-9])$')
```

3,025,141,760

```
myListOfString:
  description: myListOfString is …
  type: array
  x-kubernetes-list-type: atomic
  x-kubernetes-validations:
  - rule: self.all(x,
x.matches('^[a-z0-9]([-a-z0-9]*[a-z0-9])?(\\\\.[
a-z0-9]([-a-z0-9]*[a-z0-9])?)*/(([A-Za-z0-9][-A-
Za-z0-9_.]*)?[A-Za-z0-9])$'))
  items:
    type: string
    maxLength: 256
```

3,028,284,602

Be careful though

This may not always be true...

$1048576 \times 2885 = 3{,}025{,}141{,}760$

$\downarrow$

$(x + 1) \times y = 3{,}025{,}141{,}760$

$1 \times (1048575 \times (2885 + 1 + 2)) = 3{,}028{,}284{,}602$

$\downarrow$

$x \times (y + 3) = 3{,}028{,}284{,}602$

$\downarrow$

$(x + 1) \times y = x \times (y + 3)$

$\downarrow$

$xy + y = xy + 3x$

$\downarrow$

$y = 3x$

$\downarrow$

<rule cost> = 3<calculated cardinality>

The formula here breaks down when the minimum size of the object is not 1 lower than a factor of 3*1024*1024 -> (xy = xy + 3y => 0 = 3y => y=0)

So what can I afford with an unbounded field?

Please don't actually do this

An unbounded array of strings

<unbounded list of string cost> × <rule cost> = 10,000,000

↓

10000000 ÷ 1048576 = <rule cost> = 9.537 ~ 9

Check if an unbounded string matches a regex

⌈(<unbounded string estimate> + 1) × 0.11 × ⌈len(<regex>) × 0.25⌉ + 1 = 10,000,000

↓

(10,000,000 - 1) ÷ 314573 ÷ 0.25= <regex length> = 127.156 ~ 127

Round down to stay within the budget

Wrapping up

What was that formula again?

Function cost quick reference

What do common functions cost

	Examples	Approximate Formula (args cost omitted)
String Traversal	url, isURL, ip, isIP, lowerAscii, upperAscii, substring, trim, isQuantity	⌈<size of input> × <string traversal cost factor>⌉
Double Traversal	split, join, replace	2 × ⌈<size of input> × <string traversal cost factor>⌉
String Matchers	matches, find, findAll	⌈<size of input> × <string traversal cost factor> + 1⌉ × ⌈<size of regex> × <regex traversal cost factor>⌉
Accessing Properties	On authorization: path, group, resource, name On URLs: getScheme, getHost, getPort	1 (cost is constant)
Macros	all, exists, exists_one, map, filter	len(array) × (<per step cost>+ <loop cost>)
Fixed cost special cases	containsIP, containsCIDR	See implementation e.g. (16 + 16) × <string traversal cost factor>

Note: <string traversal cost factor> is 0.1, <regex traversal cost factor> is 0.25
CEL Cost Estimator: https://github.com/google/cel-go/blob/b43274dcbde54e3b2d50e24852dc68e3bca61b9a/checker/cost.go#L537-L661
Kubernetes Cost Estimator: https://github.com/kubernetes/apiserver/blob/78628824cd629c15057a1804df50c992011e779c/pkg/cel/library/cost.go#L264-L536

Formula quick reference

Other functions from the cost calculations

	Formula
Worst case string	3 × 1024 × 1024 - 2 = 3145726
Schema maxLength to bytes	<maxLength> × 4 (worst case rune to byte conversion)
Worst case array length	(3 × 1024 x 1024 - 2) ÷ (<min item size> + 1)
Worst case object size	<min serialised object size> + <required fields min size>
Required field min size	len(name) + <min serialised size> + 4
Worst case for unbounded cardinality	(3 × 1024 x 1024) ÷ (<min item size> + 1)

OK, so what do I need to know to avoid blowing my budget?

You promised you'd tell us

```
// +kubebuilder:validation:MaxLength:=
```

Any string field must set a maximum length to avoid large factors in proportional function costs

```
// +kubebuilder:validation:MaxItems:=
```

Arrays must set a maximum length to prevent a high factor for range counts, and to avoid unbounded cardinality

But what if I'm not using CEL, or I don't know what limit to set?

You need to set the length limit to something sensible. Think about the use case, pick the smallest size that is more than you think could ever be needed.

Joel's general caps; maxItems: 256 and maxLength: 4096

Checking your work

How to check your cost, before you ship

- https://book.kubebuilder.io/reference/envtest
  - Integration test - apply the CRD directly to an API server to find issues
- https://github.com/openshift/crd-schema-checker
  - Uses the same libraries to validate costs of rules from CRD yaml
  - Playing with exposing more detailed analysis, like cardinality and rule cost, unbounded parents

```
ERROR: "MustNotExceedCostBudget": ^.properties[spec].properties[platformSpec].properties[baremetal].properties[apiServerInternalIPs].items:
Forbidden: estimated rule cost exceeds budget by factor of 1.8x (try simplifying the rule, or adding maxItems, maxProperties, and maxLength where
arrays, maps, and strings are declared)
Warning: "MustNotExceedCostBudget": ^.spec.platformSpec.baremetal.apiServerInternalIPs: Array has unbounded maxItems. It will be considered to
have 1048575 items. Consider adding a maxItems constraint to reduce the raw rule cost.
Warning: "MustNotExceedCostBudget": ^.spec.platformSpec.baremetal.apiServerInternalIPs[*]: Field has unbounded cardinality. At least one,
variable parent field does not have a maxItems or maxProperties constraint: ^.spec.platformSpec.baremetal.apiServerInternalIPs. Falling back to
CEL calculated worst case of 1048576 executions.
info: "MustNotExceedCostBudget": ^.spec.platformSpec.baremetal.apiServerInternalIPs[*]: Rule 0 raw cost is 17. Estimated total cost of 17825792.
The maximum allowable value is 10000000.
```

Thank you

Enjoy the rest of your KubeCon!

Talk feedback ^

Find me on:

- X @JoelASpeed
- LinkedIn joel-speed
- GitHub @JoelSpeed

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat