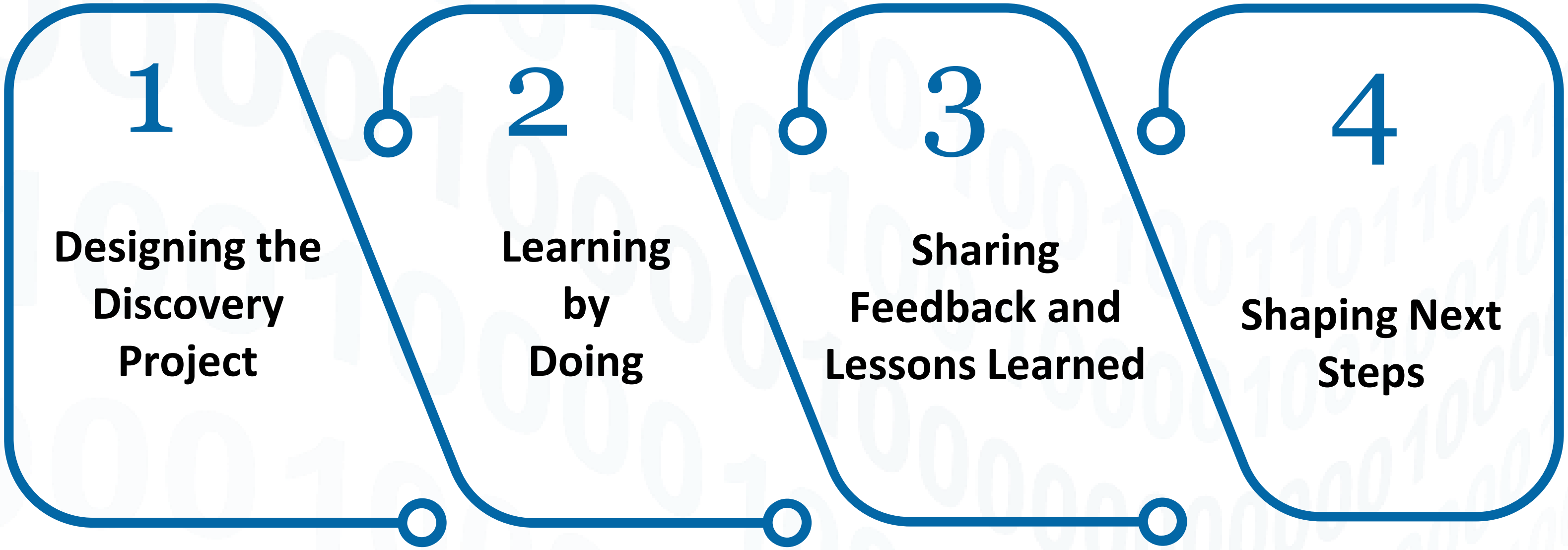
Rules As Code

Discovery Project and Possibilities for the Future

CIAJ-ICAJ Legislative
Drafting Conference
September 10, 2020

Presentation Outline

This presentation will summarize the **Rules as Code Discovery Project**, offer lessons we learned, and present a series of options for next steps


1
Designing the Discovery Project
2
Learning by Doing
3
Sharing Feedback and Lessons Learned
4
Shaping Next Steps

Project Design: Learning Objectives

This discovery project supported two main objectives: **experience the rules as code process** and **test** whether the process **can produce better rules**

1 Experience the rules as code process in the Government of Canada context

- Assess the feasibility of rules as code
- Build capacity and experience within the public service
- Build an evidence base
- Document lessons learned

2 **Begin to test** whether rules as code **produces better rules**

- Improve the quality of rules
- Support a more consistent interpretation
- Reduce costs of compliance
- New possibilities to improve service delivery

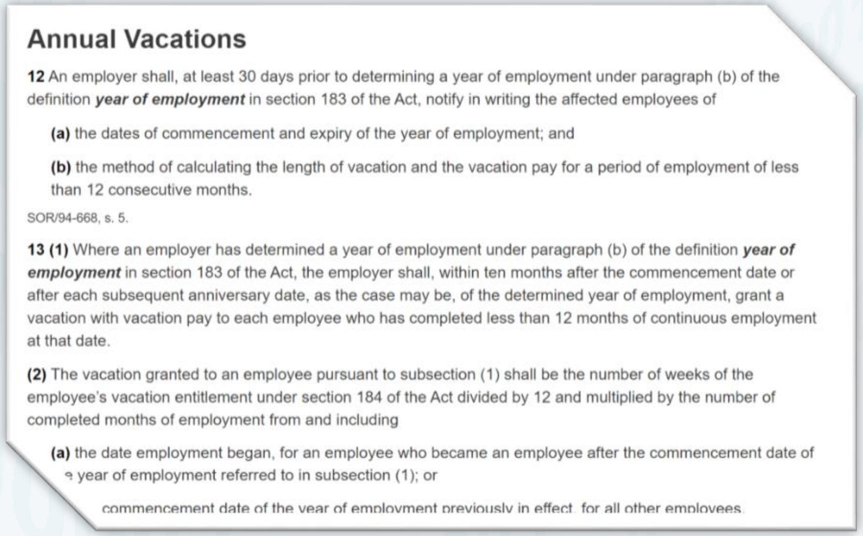
Project Design: Use Case

We selected sections 12 and 13 of the Canada Labour Standards Regulations for our use case. This regulation details vacation pay entitlements to eligible employees

Our process included:

- Identifying **key stages** of the process
- Observing of the benefits and challenges of working in a **multi-disciplinary team**
- Reflecting on what **types of rulesets** would be **best suited** to rules as code
- Learning about **opportunities to improve the drafting process**
- Collecting **lessons learned** and identify **opportunities for future learning**

This regulation has a narrow focus and prescriptive rules. It is a good test case to demonstrate the value of rules as code



Annual Vacations

12 An employer shall, at least 30 days prior to determining a year of employment under paragraph (b) of the definition ***year of employment*** in section 183 of the Act, notify in writing the affected employees of

(a) the dates of commencement and expiry of the year of employment; and

(b) the method of calculating the length of vacation and the vacation pay for a period of employment of less than 12 consecutive months.

SOR/94-668, s. 5.

13 (1) Where an employer has determined a year of employment under paragraph (b) of the definition ***year of employment*** in section 183 of the Act, the employer shall, within ten months after the commencement date or after each subsequent anniversary date, as the case may be, of the determined year of employment, grant a vacation with vacation pay to each employee who has completed less than 12 months of continuous employment at that date.

(2) The vacation granted to an employee pursuant to subsection (1) shall be the number of weeks of the employee's vacation entitlement under section 184 of the Act divided by 12 and multiplied by the number of completed months of employment from and including

(a) the date employment began, for an employee who became an employee after the commencement date of e year of employment referred to in subsection (1); or

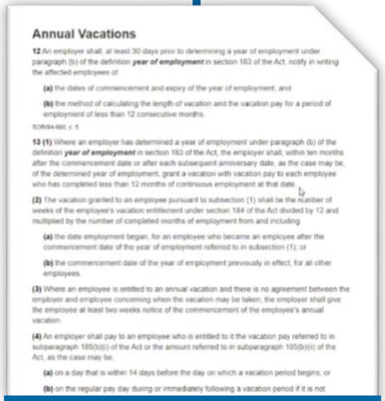
commencement date of the year of employment previously in effect, for all other employees.

Ultimately, **working through the process yielded the greatest insights**

Learning by Doing: Process Overview

We worked through a process to convert the regulation to code

Fall 2019



Annual Vacations

12 An employer shall, at least 30 days prior to determining a year of employment under paragraph (b) of the definition ***year of employment*** in section 183 of the Act, notify in writing the affected employees of

(a) the dates of commencement and expiry of the year of employment; and

(b) the method of calculating the length of vacation and the vacation pay for a period of employment of less than 12 consecutive months.

SOR/94-668, s. 5

13 (1) Where an employer has determined a year of employment under paragraph (b) of the definition ***year of employment*** in section 183 of the Act, the employer shall, within ten months after the commencement date or after each subsequent anniversary date, as the case may be, of the determined year of employment, grant a vacation with vacation pay to each employee who has completed less than 12 months of continuous employment at that date.

(2) The vacation granted to an employee pursuant to subsection (1) shall be the number of weeks of the employee's vacation entitlement under section 184 of the Act divided by 12 and multiplied by the number of completed months of employment from and including

(a) the date employment began, for an employee who became an employee after the commencement date of the year of employment referred to in subsection (1); or

(b) the commencement date of the year of employment previously in effect, for all other employees.

(3) Where an employee is entitled to an annual vacation and there is no agreement between the employer and employee concerning when the vacation may be taken, the employer shall give the employee at least two weeks notice of the commencement of the employee's annual vacation.

(4) An employer shall pay to an employee who is entitled to it the vacation pay referred to in subparagraph 185(b)(i) of the Act or the amount referred to in subparagraph 185(b)(ii) of the Act, as the case may be,

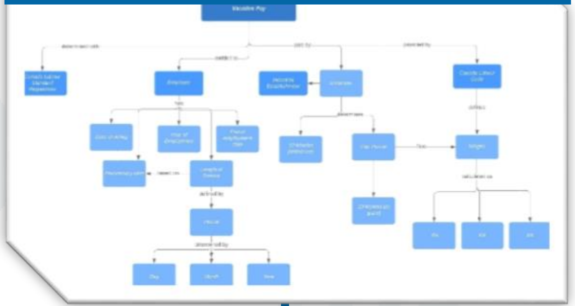
(a) on a day that is within 14 days before the day on which a vacation period begins; or

(b) on the regular pay day during or immediately following a vacation period if it is not

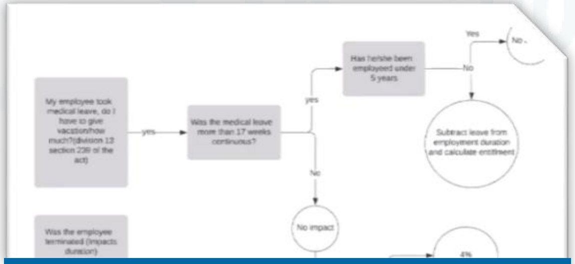
Ruleset Selected

Feb. 2020

Concept Model Created



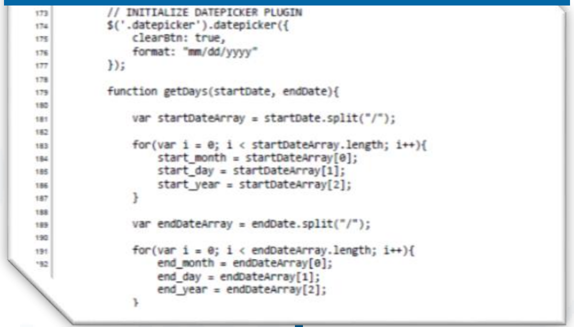
Feb. 2020


My employee took medical leave, do I have to give vacation/how much?(division 13 section 239 of the act)
yes
Was the medical leave more than 17 weeks continuous?
yes
Has he/she been employeed under 5 years
Yes
No
No
Subtract leave from employment duration and calculate entitlement
No
No impact
Was the employee terminated (impacts duration)
4%

Decision Tree Created

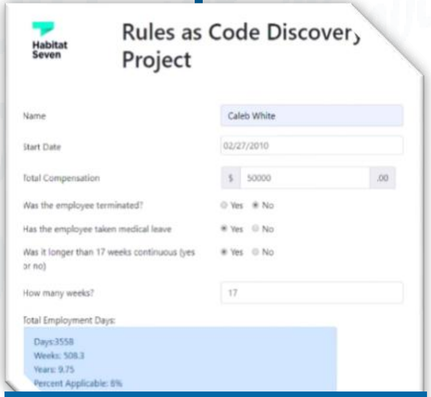
Mar. 2020

Ruleset Converted to Code



```
// INITIALIZE DATEPICKER PLUGIN
$('.datepicker').datepicker({
    clearBtn: true,
    format: "mm/dd/yyyy"
});

function getDays(startDate, endDate){

    var startDateArray = startDate.split("/");

    for(var i = 0; i < startDateArray.length; i++){
        start_month = startDateArray[0];
        start_day = startDateArray[1];
        start_year = startDateArray[2];
    }

    var endDateArray = endDate.split("/");

    for(var i = 0; i < endDateArray.length; i++){
        end_month = endDateArray[0];
        end_day = endDateArray[1];
        end_year = endDateArray[2];
    }
```

Mar. 2020



Habitat Seven

Rules as Code Discovery Project

Name	Caleb White
Start Date	02/27/2010
Total Compensation	$ 50000 .00
Was the employee terminated?	○ Yes ● No
Has the employee taken medical leave	● Yes ○ No
Was it longer than 17 weeks continuous (yes or no)	● Yes ○ No
How many weeks?	17

Total Employment Days:

Days:3558
Weeks: 508.3
Years: 9.75
Percent Applicable: 8%

Prototype Developed

Government of Canada / Gouvernement du Canada

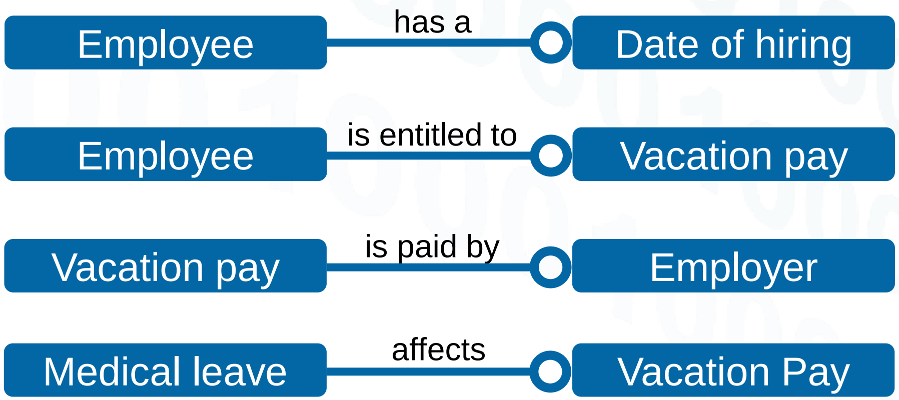
Step 1: Concept Model

First, the team held a series of workshops to **identify key concepts** in the regulation and **define the relationships between them**

Key Terms* from the Vacation Pay Regulation

- Employee
- Employer
- Date of hiring
- Length of service
- Termination
- Vacation
- Vacation pay
- Medical leave

Relationships* Between Key Terms



Employee — has a — Date of hiring

Employee — is entitled to — Vacation pay

Vacation pay — is paid by — Employer

Medical leave — affects — Vacation Pay

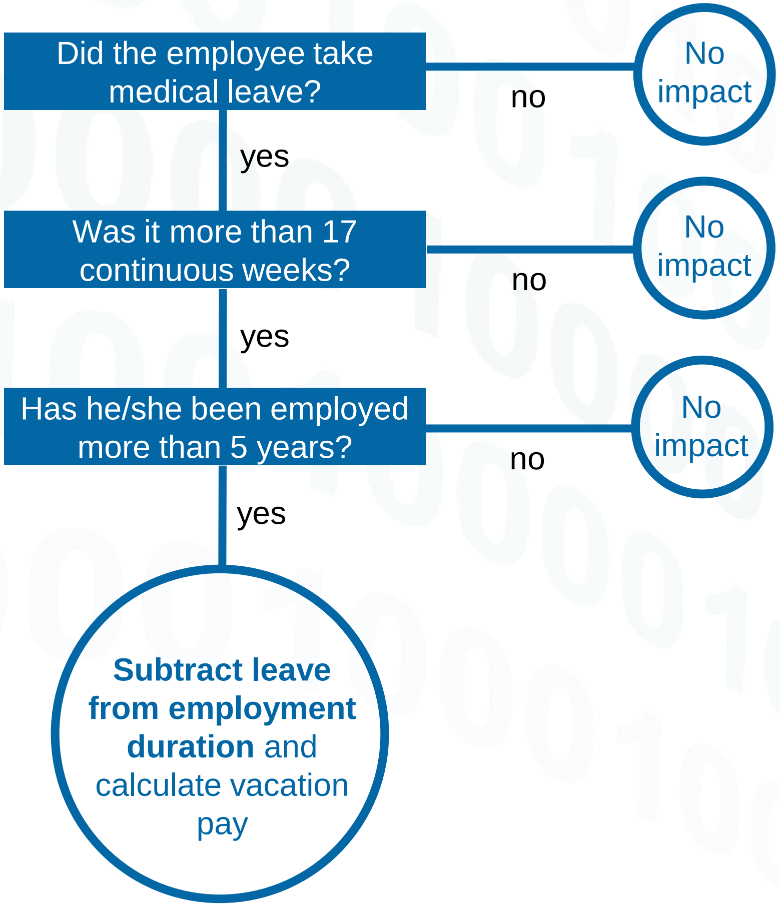
Key Insights

- Converting rules to code is not as simple as handing a ruleset to a programmer - the process matters
- Subject-matter experts played an important role to help our team interpret subtleties of the rules and how different concepts related

*These are not exhaustive lists

Step 2: Decision Tree

Next, we converted the regulations into a decision tree to determine the **effect of certain variables on vacation pay entitlement**. Examples of key questions related to medical leave are depicted below


Did the employee take medical leave?
no
No impact
yes
Was it more than 17 continuous weeks?
no
No impact
yes
Has he/she been employed more than 5 years?
no
No impact
yes
Subtract leave from employment duration and calculate vacation pay

Key Insights

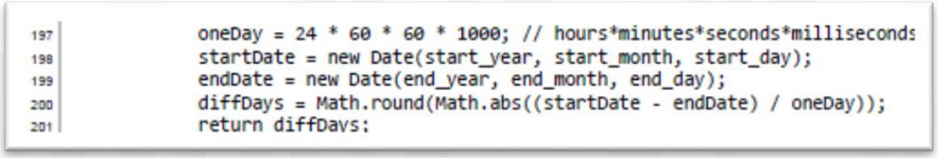
- Rules as code are **best suited to prescriptive rulesets** like vacation pay, though even these can be challenging. In this case we had to consult experts to understand technicalities of how medical leave affected the vacation pay calculation

- Using a decision tree helped our team **close the gap between how the regulation was drafted and how it would be applied in practice** by testing how the rule would perform in different scenarios

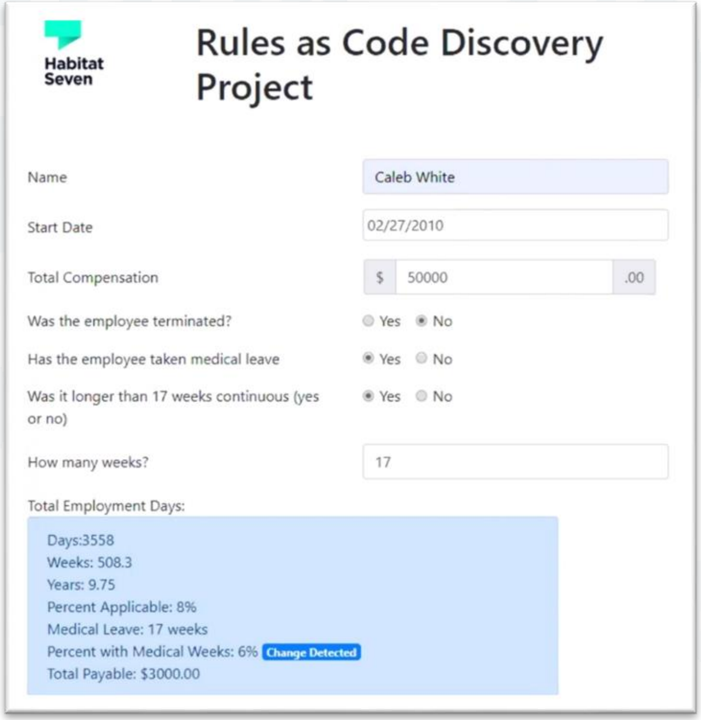
Step 3: Coding and Prototype Development

1 2 3 4

After completing the models, we **coded the regulation** and **developed a simple prototype** based on the coded rules



```
        oneDay = 24 * 60 * 60 * 1000; // hours*minutes*seconds*milliseconds
        startDate = new Date(start_year, start_month, start_day);
        endDate = new Date(end_year, end_month, end_day);
        diffDays = Math.round(Math.abs((startDate - endDate) / oneDay));
        return diffDays;
```



Habitat Seven

Rules as Code Discovery Project

Name	Caleb White
Start Date	02/27/2010
Total Compensation	$ 50000 .00
Was the employee terminated?	○ Yes ● No
Has the employee taken medical leave	● Yes ○ No
Was it longer than 17 weeks continuous (yes or no)	● Yes ○ No
How many weeks?	17

Total Employment Days:

Days:3558
Weeks: 508.3
Years: 9.75
Percent Applicable: 8%
Medical Leave: 17 weeks
Percent with Medical Weeks: 6% Change Detected
Total Payable: $3000.00

https://rulesascode.habitatseven.work

Key Insights

- Encoding rules and building a prototype was the 'easy part' of this project. The **greatest source of value is the process** to identify concepts, relationships, and decision points
- Though not tested in this project, future projects will face technical challenges to develop an **application programming interface** (API) - an interface that could allow a business to link encoded rules directly to their business systems

Lessons Learned: Processes

1 2 3 4

We demonstrated how rules as code could be applied in the Government of Canada context, we could reflect on their **use cases** and **potential benefits**

Learning Objective

1 Experience the rules as code process in the Government of Canada context

Assess the feasibility of rules as code

Build capacity and experience within the public service

Build an evidence base

Document lessons learned

Key Insights

This process is best suited to rulesets that are prescriptive and able to be quantified – in other words, this process works best when we can reduce rules to yes/no, true/false, and if-this-then-that statements

We expect parallel drafting of rules as text and as code will add time to the drafting process, but we see the potential for **efficiency and time savings** after a rule is converted to code (e.g. quicker deployment of a ruleset)

Lessons Learned: Processes (continued)

Working through the process together built internal capacity and rules as code experience across the regulatory community

Learning Objective

1 Experience the rules as code process in the Government of Canada context

Assess the feasibility of rules as code

Build capacity and experience within the public service

Build an evidence base

Document lessons learned

Key Insights

- Public servants who participated in the project learned new skills and improved their understanding of how rules of code could be applied in practice. This **capacity** can be leveraged to equip departments pursue future rules as code initiatives

- The **multi-disciplinary** team we assembled included public servants with different skillsets, expertise, and perspectives. The multi-disciplinary team was an asset to the project, with team members asking challenging questions and **considering the project from multiple perspectives**

Lessons Learned: Better Rules

The rules as code process offers opportunities to draft higher quality rules

Learning Objective

2 **Begin to test** whether rules as code **produces better rules**

Improve the quality of rules

Support more consistent interpretation

Reduce costs of compliance

New possibilities to improve service delivery

Key Insights

One of the biggest potential benefits we see is better **connecting how rules are drafted and how they are implemented**

We simulated how our ruleset could be applied in our prototype application. Based on our test, we could make corrections and improve accuracy. If rules were encoded in parallel, **live testing could help drafters write better rules**

Our experience showed us that rules as code will likely **change both the drafting process and the substance of the rules we draft**

Lessons Learned: Better Rules (continued)

Encoding rules offers opportunities to **improve service delivery** and facilitate greater regulatory **compliance**

Learning Objective

2 **Begin to test the hypothesis** that rules as code **produces better rules**

Improve the quality of rules

Support more consistent interpretation

Reduce costs of compliance

New possibilities to improve service delivery

Key Insights

After working through the process to interpret the meaning and intention of the vacation pay regulation, we see significant value to encoding a rule that others could access as an **authoritative source** and accurate interpretation of a complex rule

The scope of this project did not include a public-facing element or the development of an API to link encoded rules to other business applications. The team recommends that future projects test these capabilities to consider effects on **compliance costs and service delivery**

Future Learning

The demonstration project was effective. It quickly and cheaply demonstrated that it is possible to convert legal text into machine-usable code. Building on this success, there is **more to learn** as we explore how rules as code could be applied

- Coded rules will likely be considered to have "non-official" legal standing. What does that mean for the practical applicability of coded rules?
- What approaches, vocabularies, languages, and platforms should be standardized as we move forward?
- How would a rules as code process change the process of drafting a new ruleset if done in parallel? What about documents incorporated by reference?
- What would it look like to demonstrate an entire use case (including development of an API and a public-facing end product)?
- What capacity and skillsets would be required for drafters and for regulators to be able to effectively work alongside someone with technical coding expertise?

Future Opportunities

We have opportunities to take on other work related to rules as code. Some of these opportunities include:

Rules as Code This Year

1) **Simulate a process where a new ruleset is developed in parallel** with a rules as code process to study how drafting timelines could be affected and whether rules as code could improve the quality of the ruleset

2) Cover the **entire use case for a ruleset**, including the development of an API and a public-facing component

3) **Test** the rules as code method on a **different kind of ruleset** (e.g. policy or standard)

Other Rules as Code Work

1) Assemble **guidance** and other resources for the Government of Canada Regulatory Community, including lessons learned, use cases, and advice on methods and techniques

2) Contribute to the development of **common standards, frameworks, and guidelines** for rules as code in the Government of Canada

Questions?

Scott McNaughton
Senior Project Lead
Innovation and Policy Services Branch
Canada School of Public Service
Scott.Mcnaughton@canada.ca
@scott_pm