Team 6 Practicum Hand-Off

Table of Contents

Team members

- Alex MacDonald: Alexandra.MacDonald@oag-bvg.gc.ca
- Jason Albino: Jason.Albino@irb-cisr.gc.ca
- Pat Heard: patrick.heard@tpsgc-pwgsc.gc.ca
- Tracey Everitt: Tracey.Everitt@rcmp-grc.gc.ca

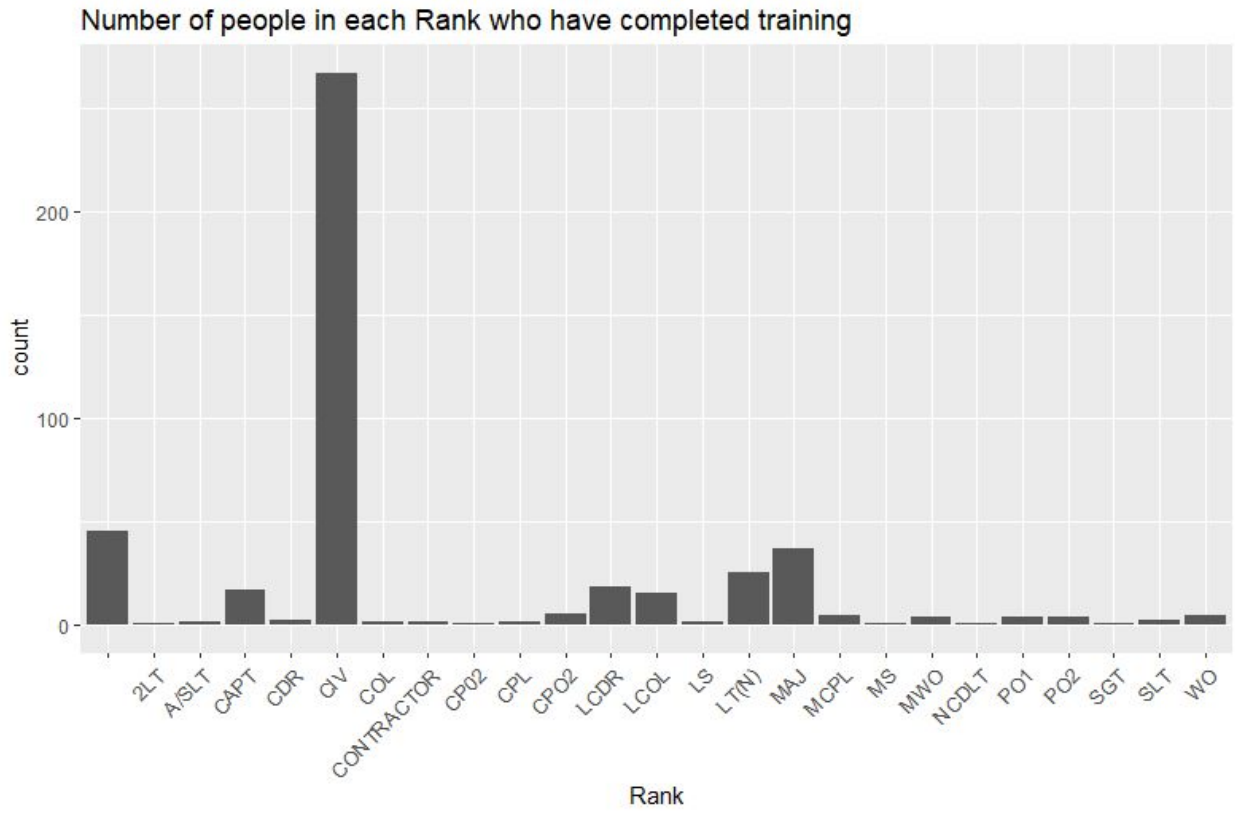
What we've done

Data Analysis:

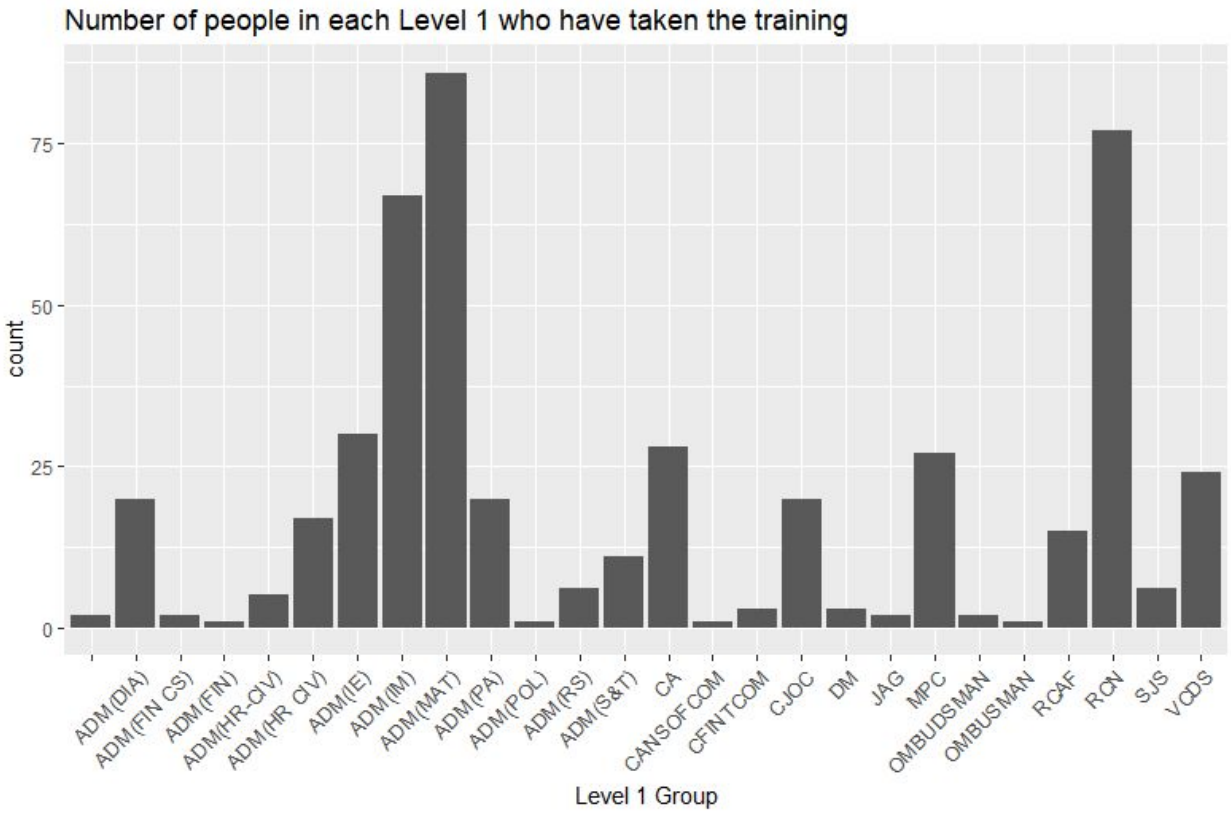
Prior to September 2018, individuals who took the training were not given a User ID. This causes some discrepancy between the users that appear in the different data sets.

All Users

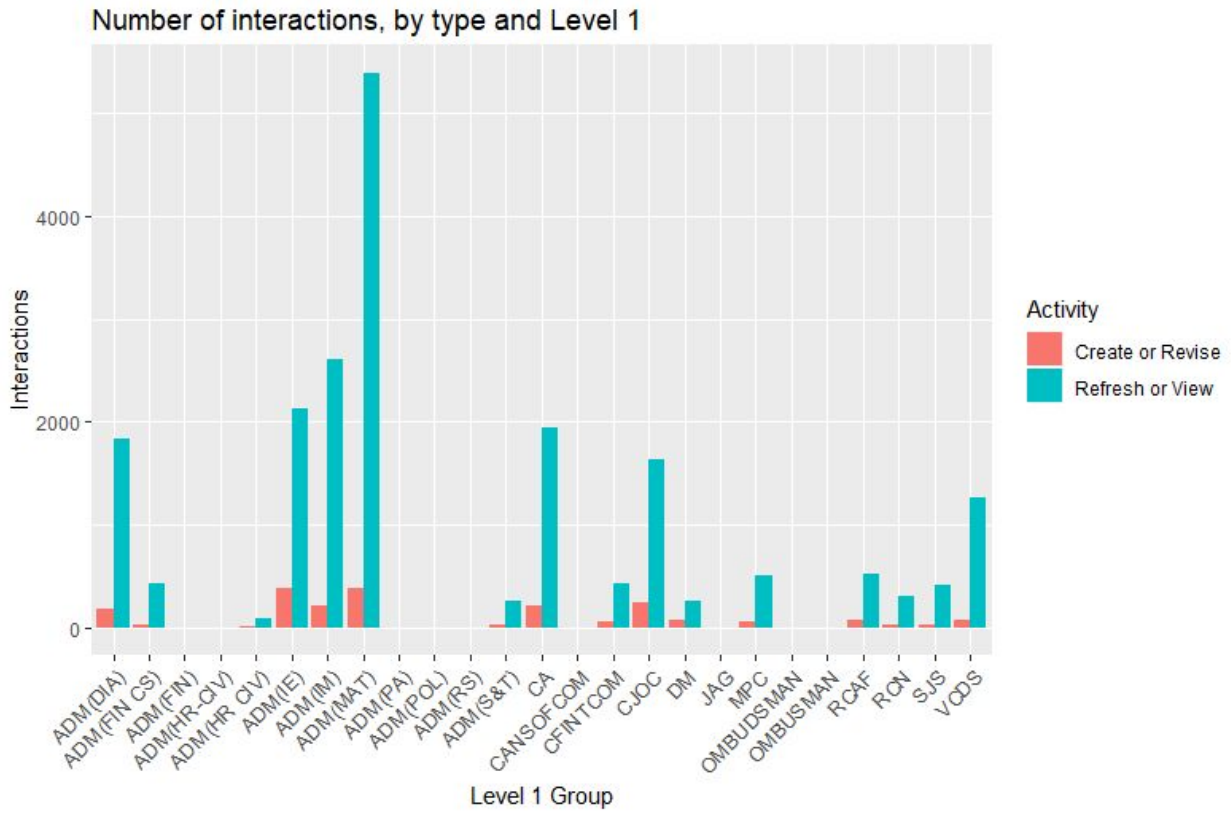
- 1148 users in the system, across User and Training data.
- 463 Users have done one of Create/Review or Refresh/View
    - Don't necessarily need the same training
- 1035 users have not done the training, not used the software, or both.
    - 113 users have done both
    - 671 users have only used the software
        - Individuals who took the course before September 2018 do not have a UserId in the training data
- 364 users only did the training
- Mostly civilians taking training



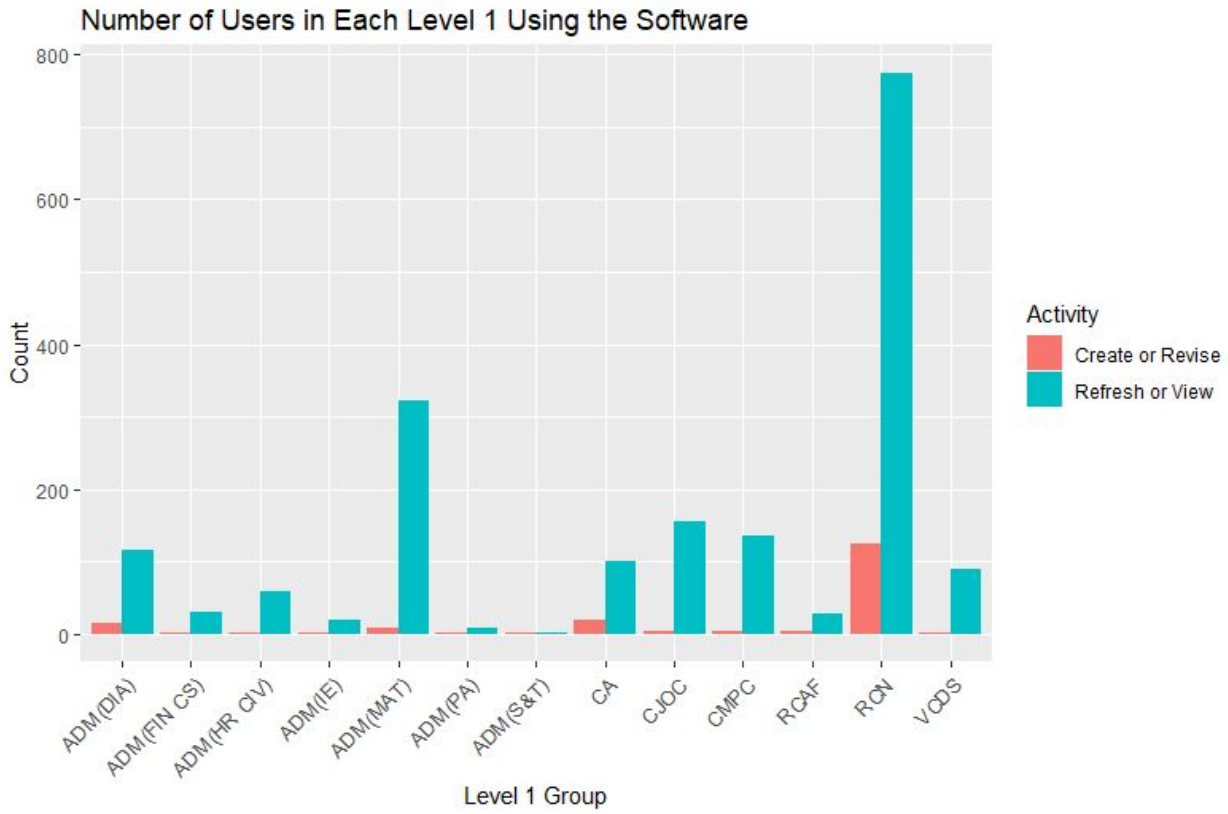
- Most people doing training come from these L1s:
    - ADM(MAT)
    - RCN



- Most interactions with the software:
    - Create and Review
        - ADM(IE)
        - ADM(MAT)
        - CJOC
    - Refresh and View
        - ADM(MAT)
        - ADM(IT)
        - (ADM(IE)

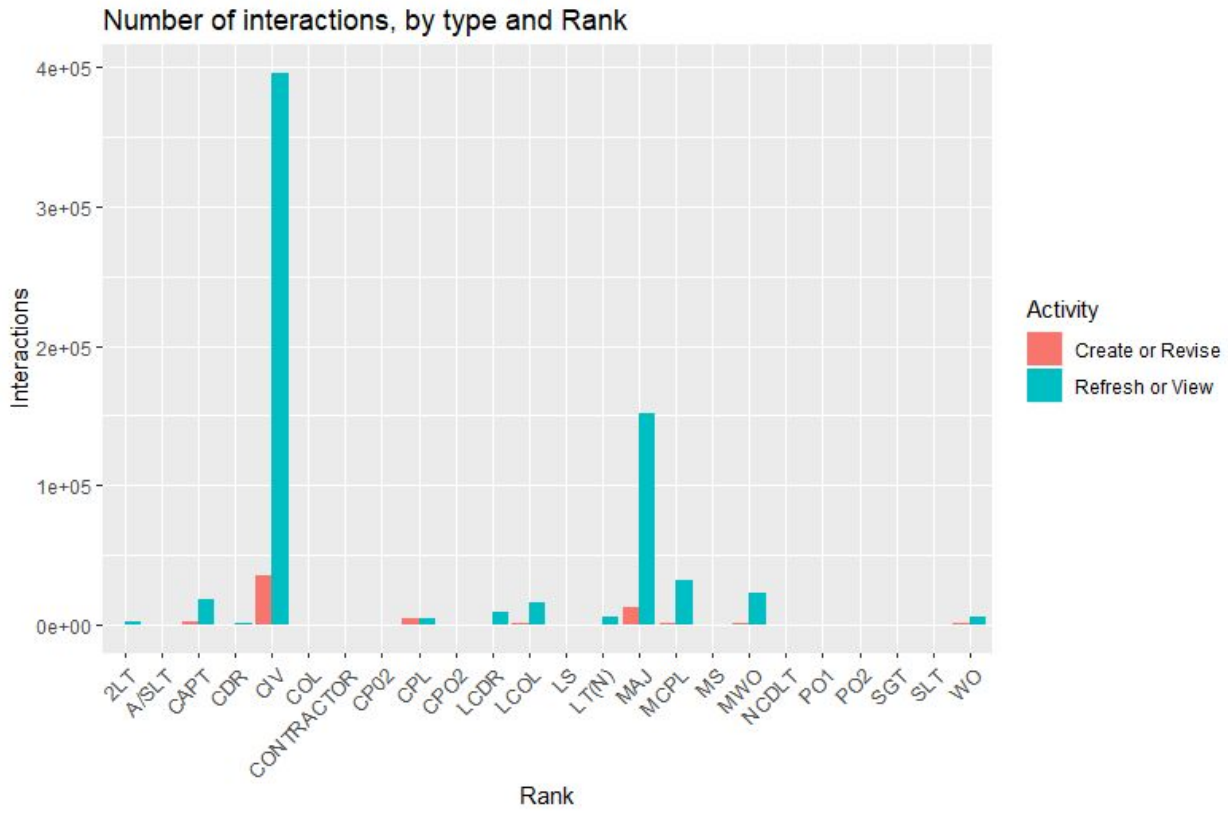


- Most users using the software:
    - Create and Review
        - RCN
    - Refresh and View
        - RCN

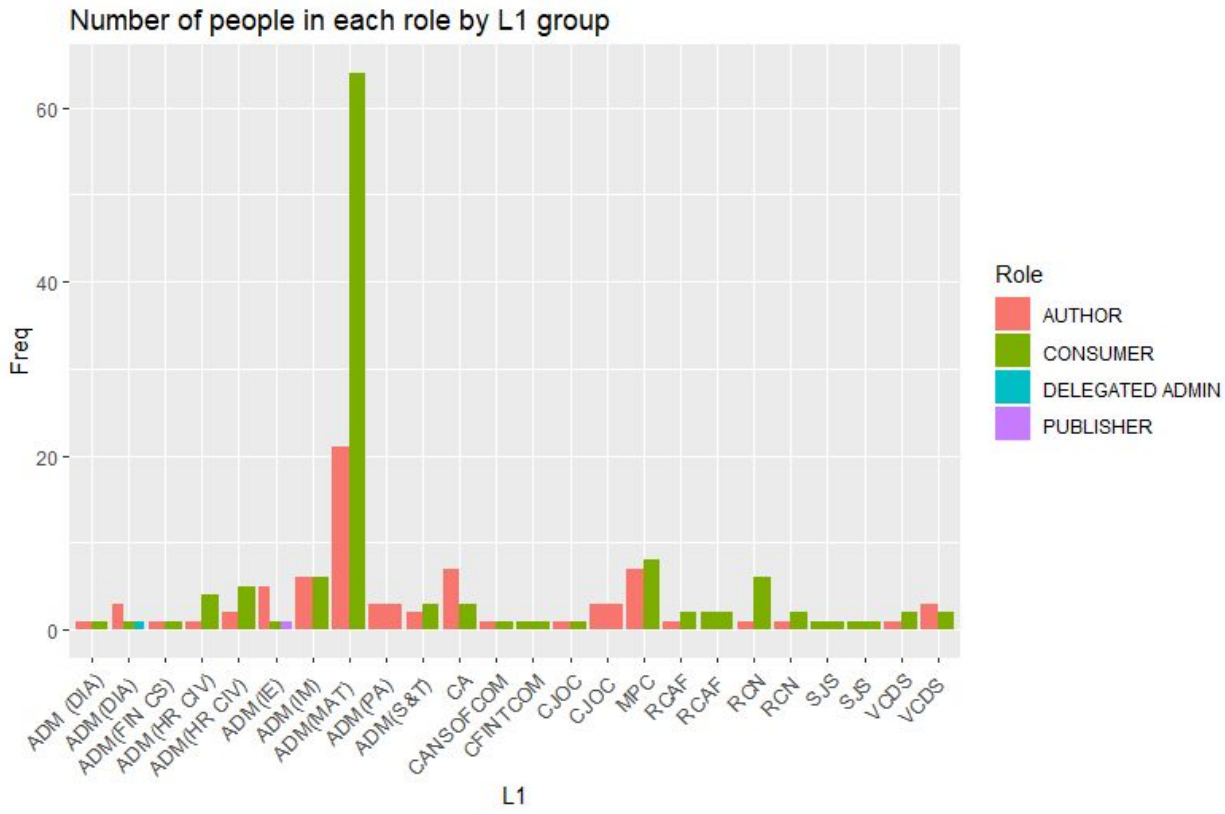


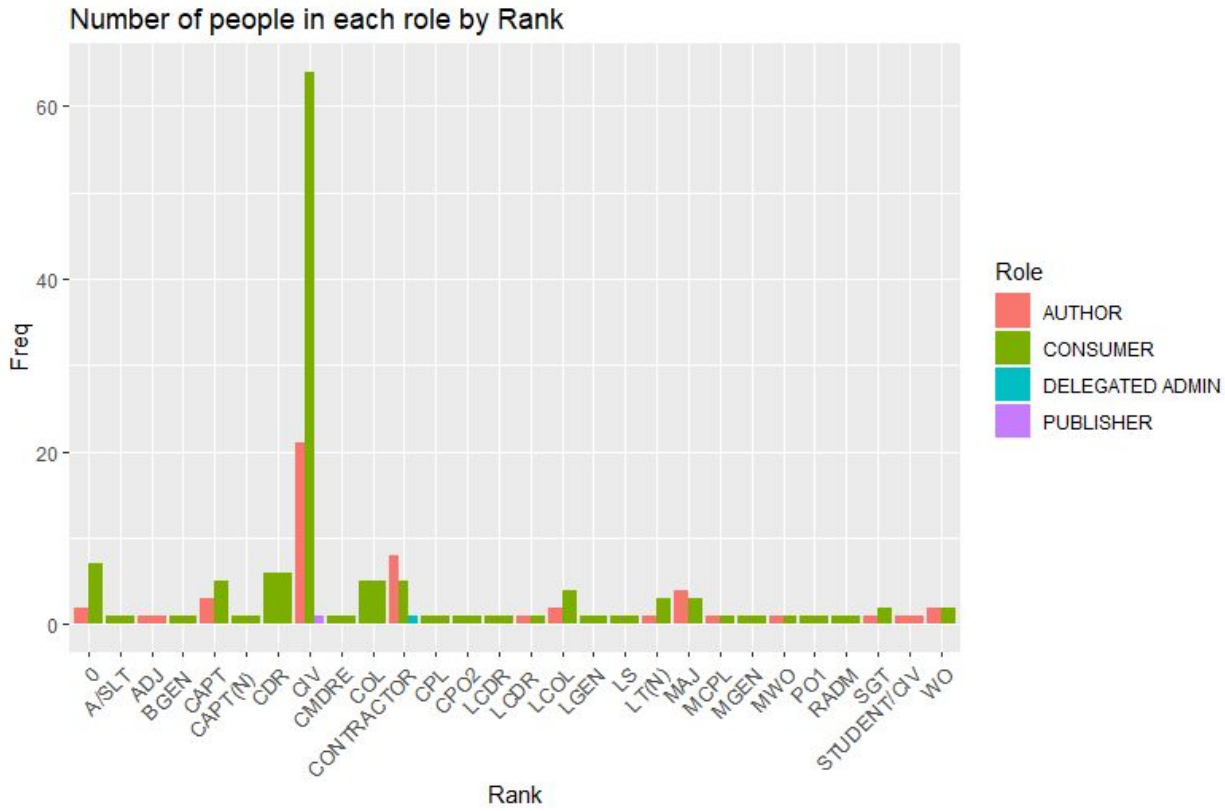
While RCN has the most *users*, most activity comes from ADM(MAT)

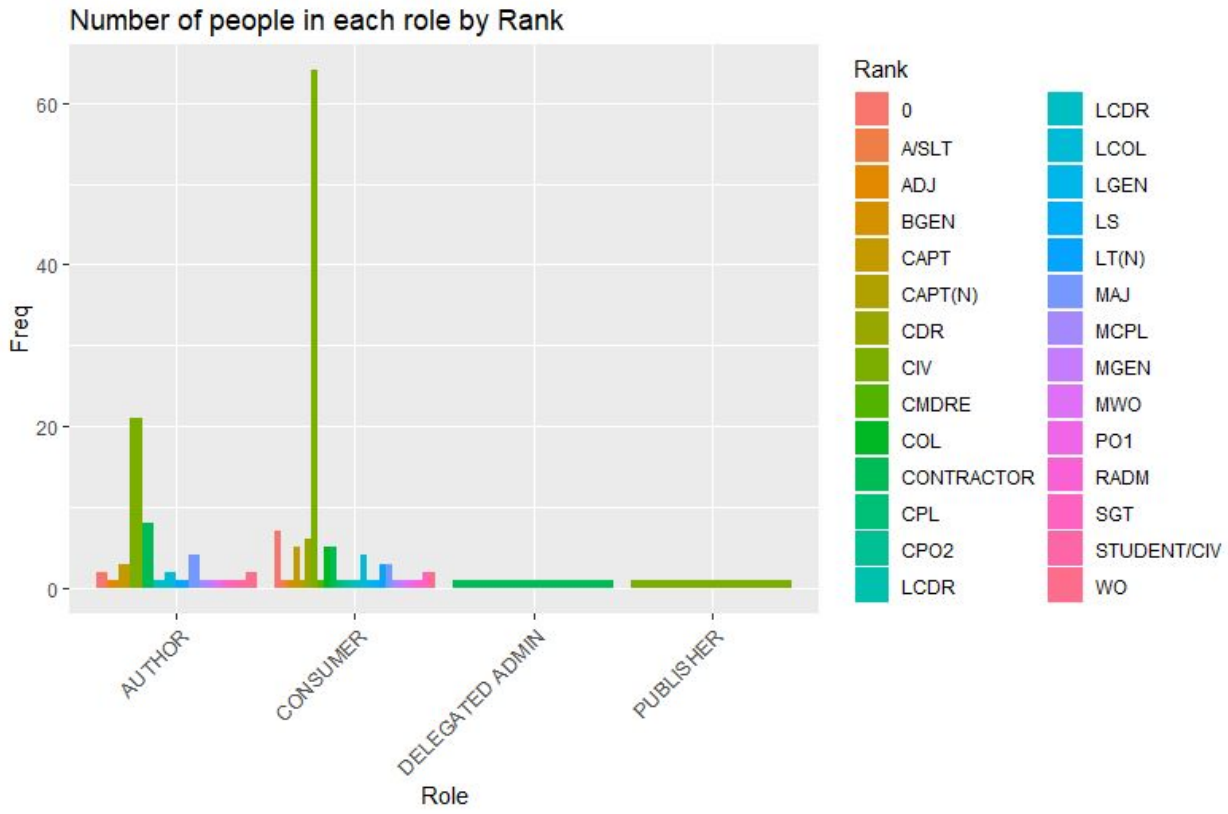
- The report data does not have Rank, making it difficult to identify exactly who should be taking the training.
    - 2537798 interactions with Create/Review missing rank
    - 20535254 interactions with Refresh/View missing rank
    - From the data that is available:
        - Civilians interact more with the software



- Most people are consumers of data. Most users are from ADM(MAT)
- Most users are civilians. Most users are consumers.


Number of people in each role by L1 group
Freq
60
40
20
0
ADM (DIA)
ADM(DIA)
ADM(FIN CS)
ADM(HR CIV)
ADM(HR CIV)
ADM(IE)
ADM(IM)
ADM(MAT)
ADM(PA)
ADM(S&T)
CA
CANSOFCOM
CFINTCOM
CJOC
CJOC
MPC
RCAF
RCAF
RCN
RCN
SJS
SJS
VCDS
VCDS
L1
Role
AUTHOR
CONSUMER
DELEGATED ADMIN
PUBLISHER


Number of people in each role by Rank
Freq
60
40
20
0
0
A/SLT
ADJ
BGEN
CAPT
CAPT(N)
CDR
CIV
CMDRE
COL
CONTRACTOR
CPL
CPO2
LCDR
LCDR
LCOL
LGEN
LS
LT(N)
MAJ
MCPL
MGEN
MWO
PO1
RADM
SGT
STUDENT/CIV
WO
Rank
Role
AUTHOR
CONSUMER
DELEGATED ADMIN
PUBLISHER



Users that have taken the training and use the software

Number of users that completed training by Rank (n = 113)

2LT	CAPT	CDR	CIV	CPL	LCDR	LCOL	LT(N)	MAJ	MCPL	MWO	WO	NA
1	6	1	71	1	2	5	1	11	3	3	1	7

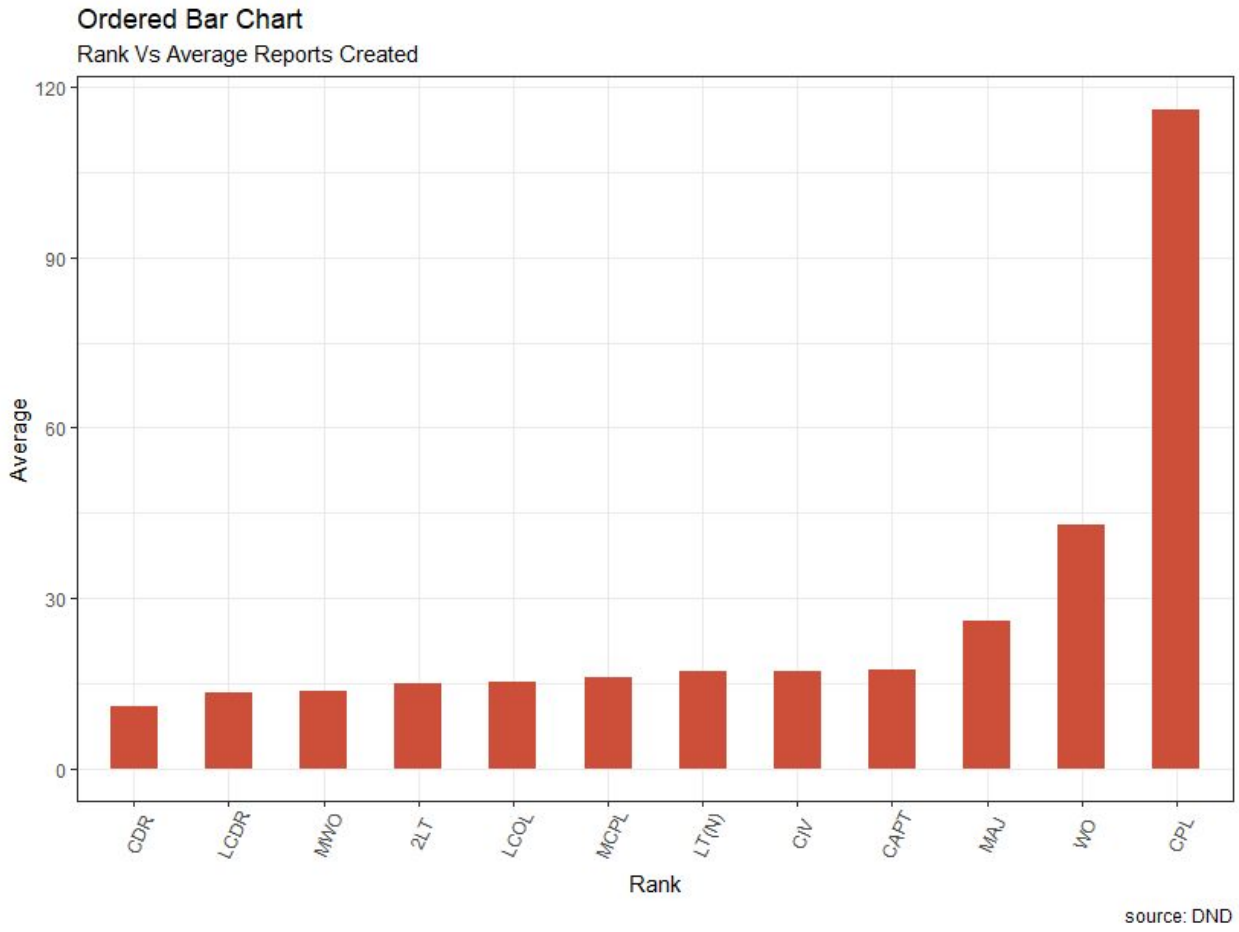
Number of users that completed training by L1 (n = 113)

ADM (DIA)	16
ADM(Fin CS)	2
ADM(HR Civ)	1
ADM(IE)	16

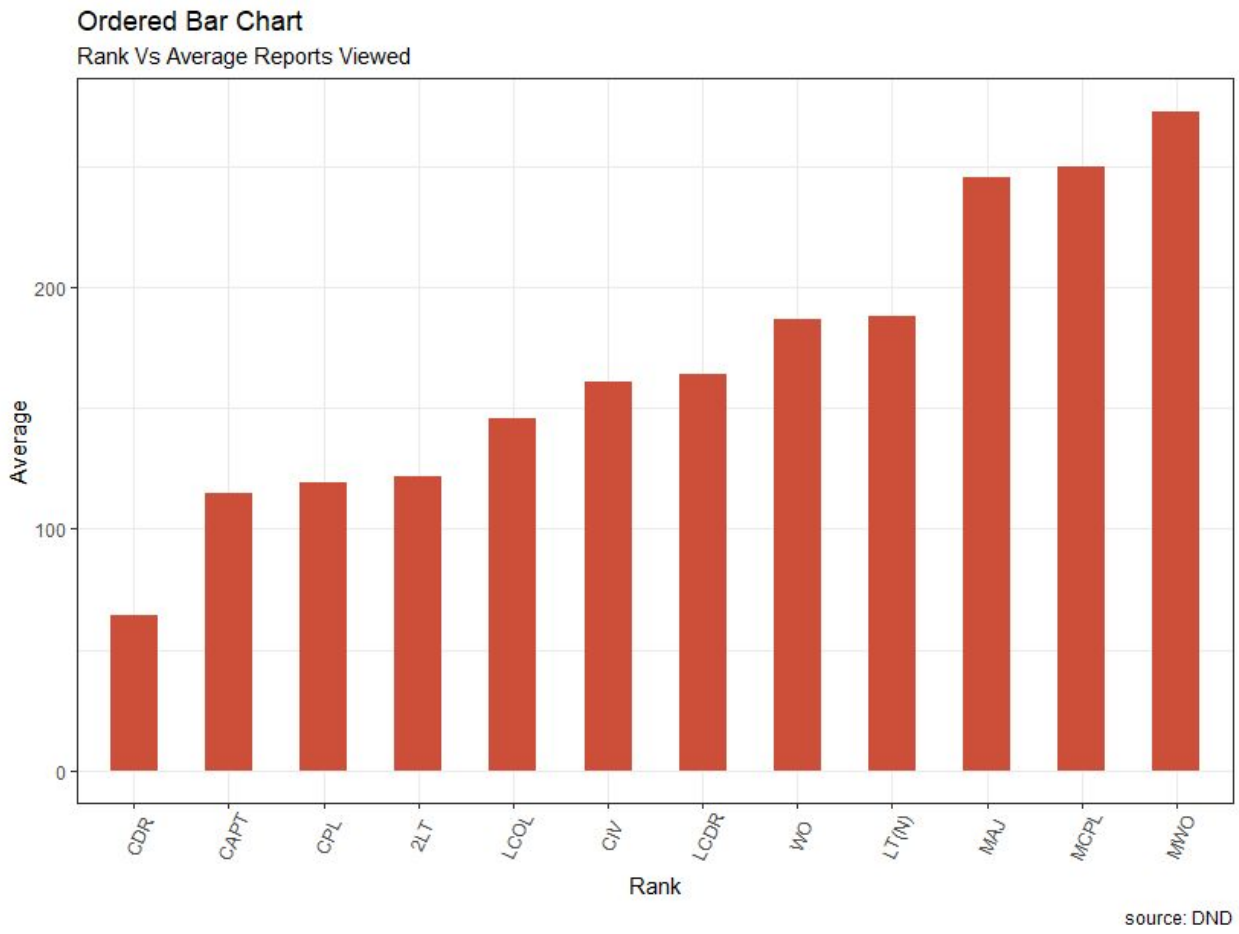
ADM(IM)	14
ADM(Mat)	24
ADM(S&T)	2
CA	14
CFINTCOM	2
CJOC	3
DM	2
MPC	4
RCAF	3
RCN	2
SJS	3
VCDS	5

Observations and findings

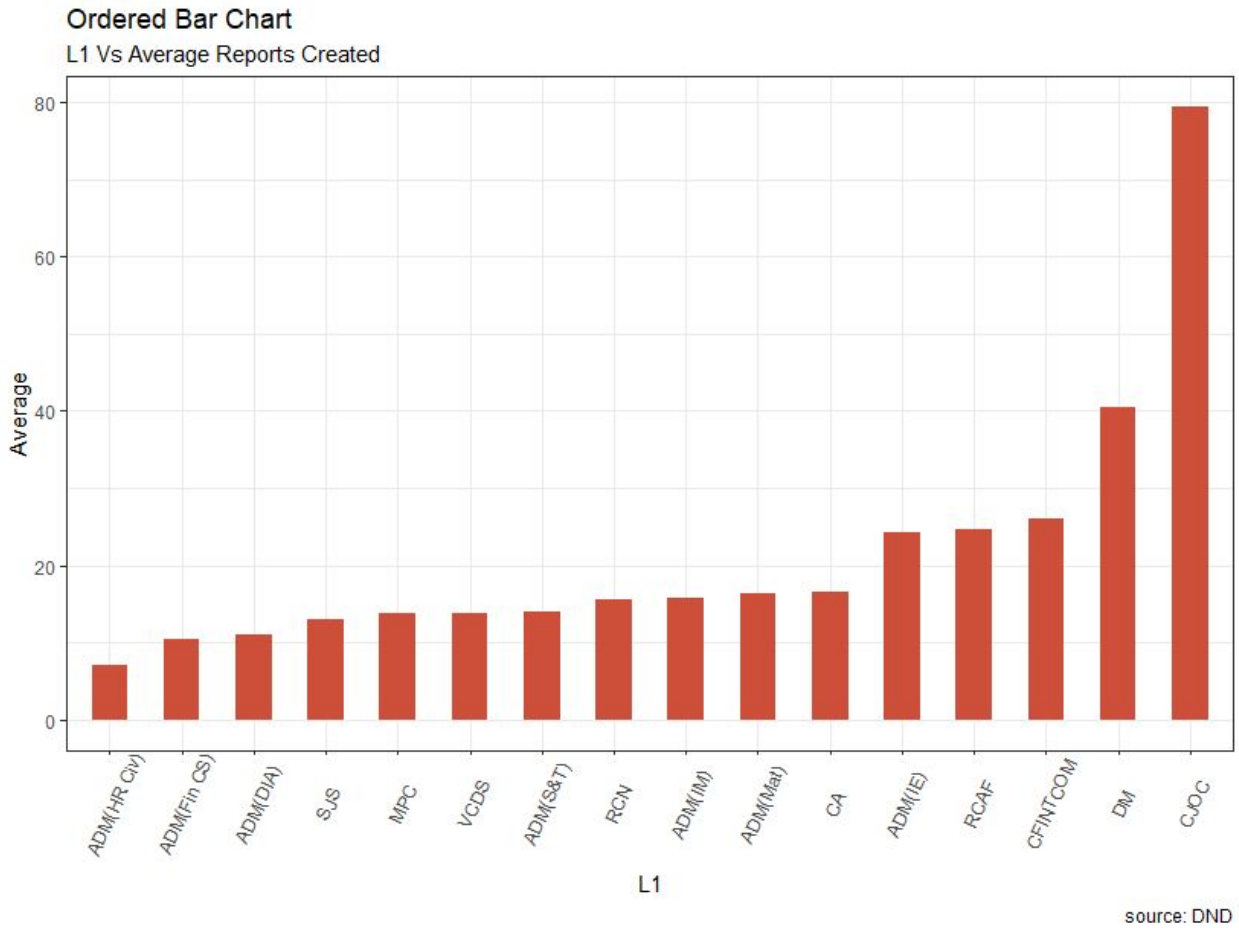
1. MAJORs should complete the training courses as they rank in the Top 3 in report creation and report views
2. CAPTAINs produce as much as CIVILIANS, with 6 captains having taken the course vs 71 civilians
3. CPLs and WO produce many reports, but are least represented in the training. It would be good to know whether their participation is an anomaly or could be regular participants.



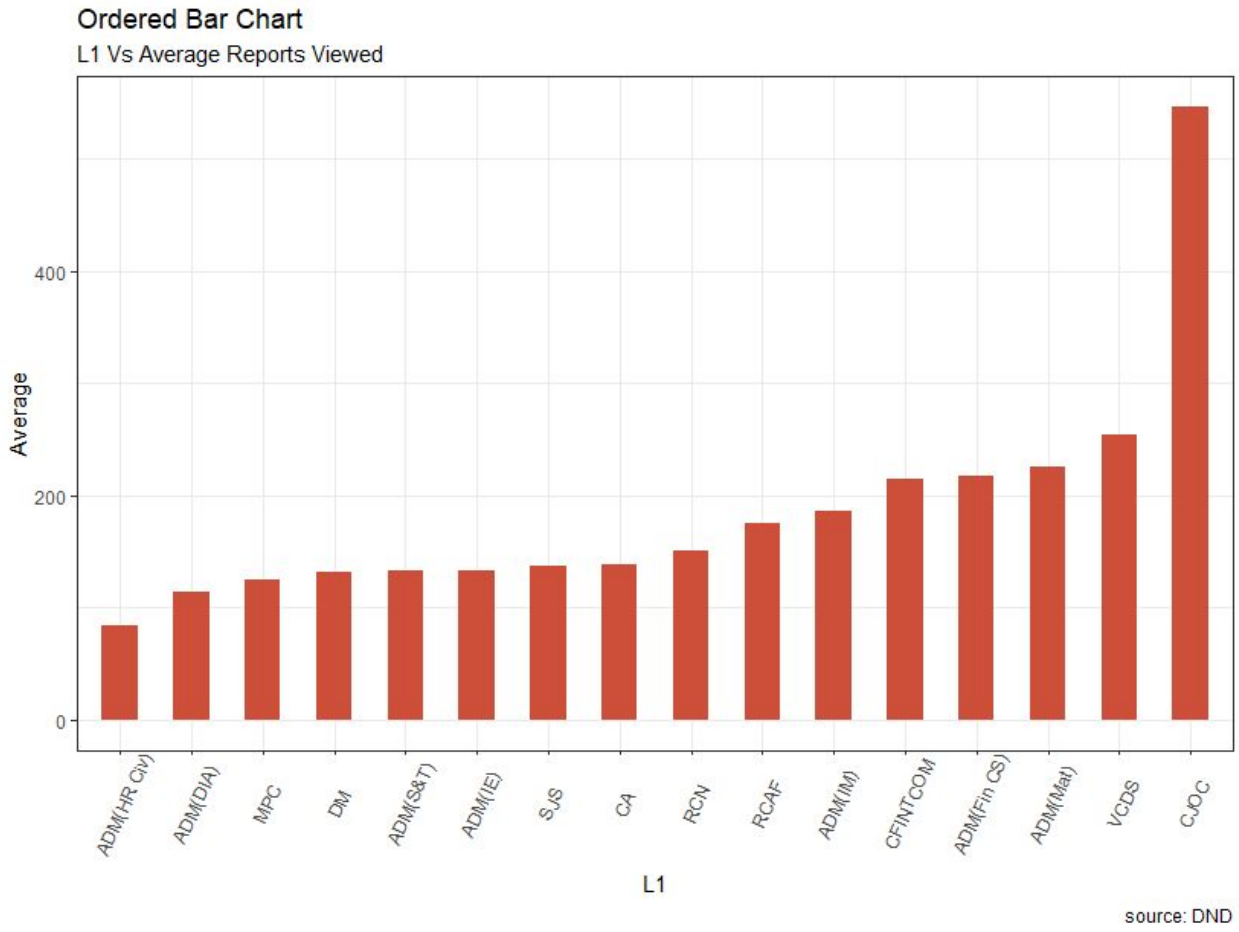
4. MWO and MCPL could be receptive to training since they view these reports heavily and could benefit from training by creating a feedback loop between viewers and creators to improve quality of reports and/or pull better insights.



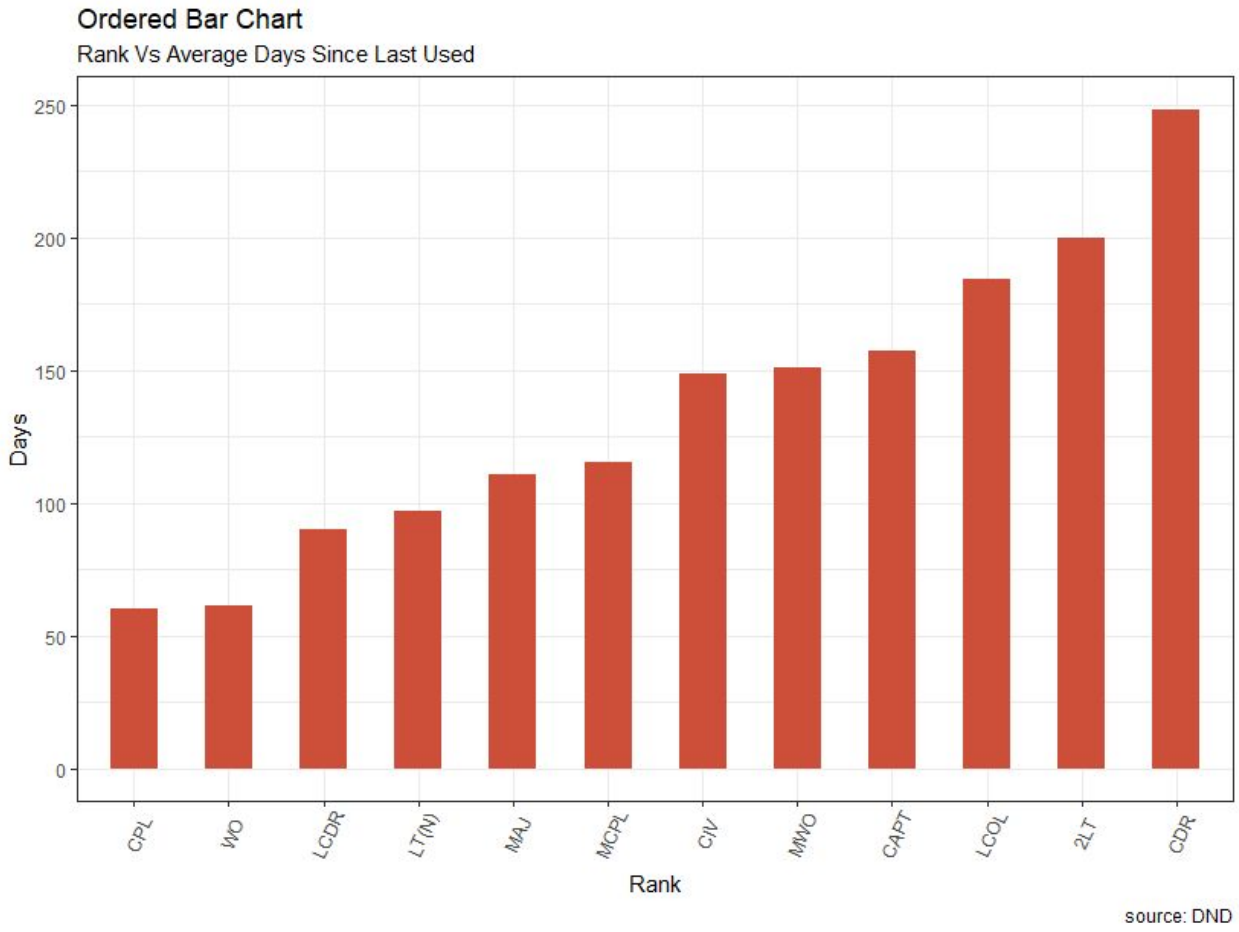
5. The L1 groups that create the most reports are the least represented. The Top 3 creators are CJOC, DM, CFINTCOM with each sending 3, 2 and 2 learners, respectively. Workload distribution could be more equitable if more representatives from these groups attend training.



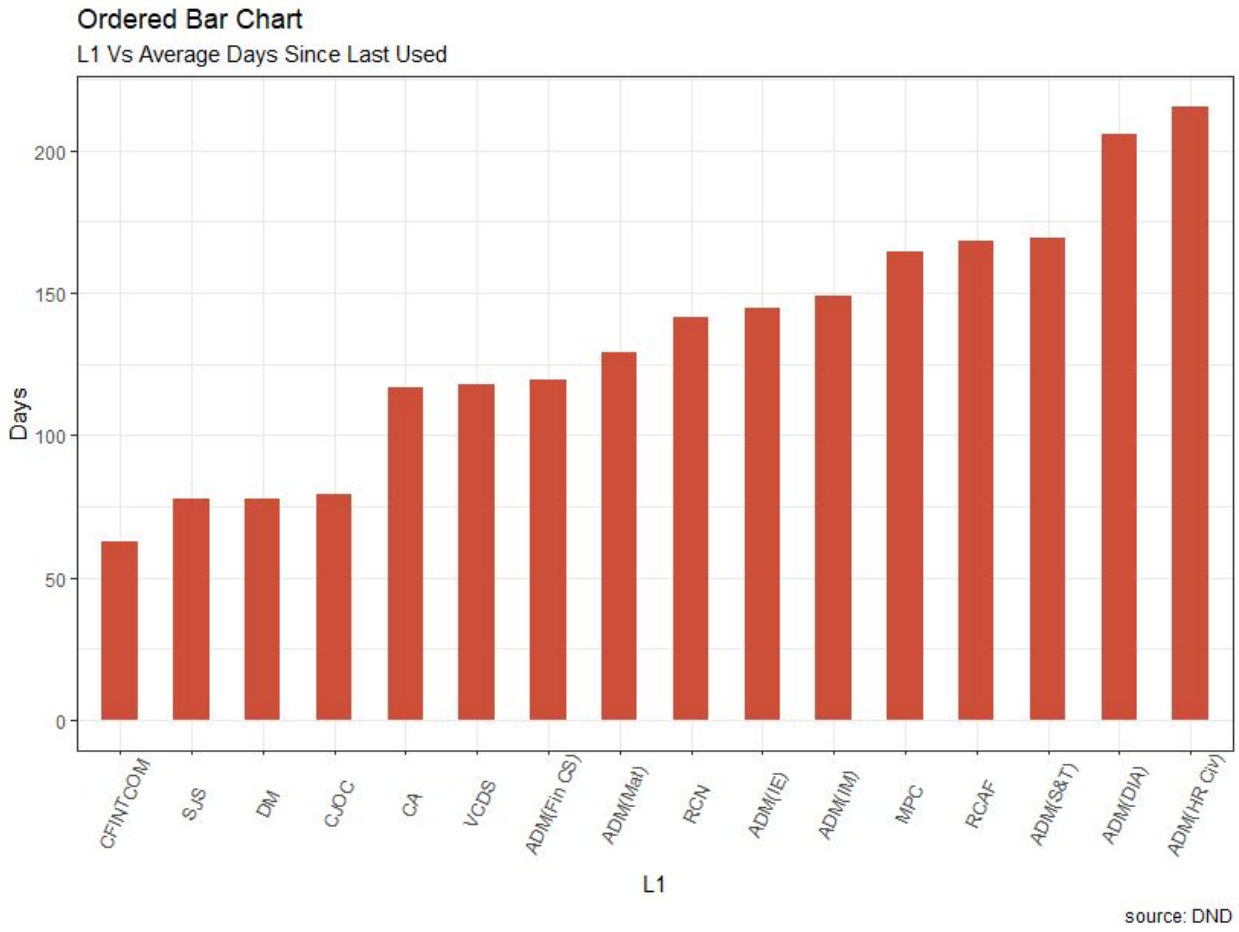
6. Two of the Top 3 L1 groups in report viewing have little representation among the learners. Those groups are CJOC and VCDS. They could be groups targeted for training.



7. LCOLs probably shouldn't take the course since over 180 days elapsed, on average, elapsed between their last use and 28 May 2019. Similarly, for CDRs, LCOLs and 2LTs.



8. ADM(DIA) might not apply the training regularly, and as a result, might not be the best group to complete the training



Assumptions used in making observations:

- Report creation is a proxy for expertise
- Reports created aren't primarily the tasks of senior officers
- Report viewing is a proxy for use and application of training
- Number of days between last use and May 28 2019 is a proxy for how regularly the platform is used

Framework

The ***Data Literacy Comptencies.xlsx*** document includes a draft of a competency framework with proficiency-level based performance statements to evaluate data literacy, as well as the competency definitions and three defined data user types.

The document also identifies related competencies where there may be overlap that may need to be addressed to synthesized these competencies into the organizational competency dictionary.

Prototype

Web page prototypes for the evaluation app and data dashboards were created to show the user interface (UI) and user experience (UX) design of the product. These prototypes can be seen on Github:

- **Student dashboard**: shows how data visualization could be used to help a student understand their progress in a course.
- **Quiz**: a quick knowledge check that a student would take before a course or module to assess their current understanding of the topic. This would allow instructors to tailor the delivery of the course.

Going forward with the project, these prototypes could be used for:

- **User testing** to ensure that key features are obvious and simple to use. This would also help identify features that have been missed or that do not work well.
- **Accessibility testing** to make sure the application works well with assistive technologies and meets WCAG 2.0 AA guidelines.
- **Data collection as a front-end for the evaluation app**. The team's assumption is that with more data before, during, and after training, it will be easier to measure if training is having the desired impact. To test this assumption, we need to start collecting and analyzing data as people use the application in a training environment.

How the prototype is built

The prototypes are simple Node.js applications that were built using:

- **Bootstrap:** front-end CSS/JS framework.
- **Chart.js:** graphing JS library.
- **Webpack:** build tool for creating the CSS/JS bundles used by the web pages.

The code is available on the team's Github repository in the /prototype folder, with instructions about running the project in the README.md.

Next steps for the project

Going forward, the next steps for developing the evaluation app would be:

1. **Perform user testing:** This would be used to validate that the web pages that have been developed are easy to use and do not impede the student's experience on a course.
2. **Create a quiz service:** The quiz service would serve questions to the students and capture data on their interactions. This data would then be analyzed to assess how the student is progressing in the course.
3. **Create a data analysis service:** This service would be responsible for analyzing the data collected by the quiz server and determining a student's performance in the course.
4. **Design data visualizations:** Data visualizations would be needed to communicate the analysis to the different audiences (student, training team and student's organization).

In terms of a development team, the following positions are recommended:

- **Front-end developer:** responsible for the user experience, accessibility and development of all front-end screens that the user interacts with. Would also be required to call out to back-end API endpoints to load and save data.
- **Back-end developer:** responsible for creating testable back-end service code that drive the front-end screens. They would be creating the quiz and data analysis service.
- **DevOps:** manage the CI/CD pipeline and infrastructure provisioning. A background in test automation would also be helpful to support the developers in creating automated tests.
- **Security:** ensure that the service is secure and not vulnerable to attack. This role could be part of the DevOps role if the right person was found (DevSecOps role).
- **Data scientist:** analyze the data, design visualizations and identify gaps in the data.
- **Education specialist:** responsible for creating course content and ensuring education best practices are followed.
- **AI/ML specialist:** not required initially, but could research topics such as question/assignment grading and student classification based on results (e.g. what course level is "right" for a student based on their performance).

Data collection

The main assumption of this project is that by collecting and analyzing more student data, it will be possible to determine if training is having a positive impact. The types of data we can collect are based on the question and assignment types of the course.

From a grading perspective, questions with a single correct answer (multiple choice, true/false) are easier to analyze since they are objective and can be marked by software. However, to gauge concept understanding, performance based questions that are observed by an instructor give the most insight into a student's knowledge. As such, it's important that the evaluation app offers the following options:

- Questions with a single correct answer, and
- Subjective, interactive assignments that can be assessed by the instructor.

The reason for the two types is to strike a balance between ease-of-use, effort required and overall student assessment.

The different assessment types, question types and data that can be collected is available on Github. The key piece is the data that can be collected as the user interacts with the evaluation app. This is because the app will be able to collect both explicit and implicit data about the student's performance:

Explicit data:

- Answer
- Quality of the answer (in the case of non-binary questions)
- Quality of answer compared to other questions within the same module.

Implicit data:

- Length of time between question load and answer entered (but not submitted).
- Length of time between question load and answer submitted.
- Total length of time to complete series of questions or assignment.
- Length of time to answer questions when familiar topic.
- Length of time to answer questions when unfamiliar topic.
- Length of time to answer simple vs. complex questions.
- Number of times revised answer.
- Number of times revisited question.

This data will be trivial to capture and should, over time, give insight into how students are performing on a given subject. It's also likely that as student data is collected, new types of implicit data will be recognized and captured by the evaluation app.

Course creation

In order to pilot the evaluation app, course content will need to be created. The quickest way to do this would be to design supporting course content for an existing course that's already being delivered. Once the supporting course content is ready, use of the evaluation app could be phased into the delivery of that course.

In terms of how the supporting course content could work, here are some recommendations:

- **Before the course:** Create a short quiz asking the student about their familiarity with the concepts (Likert scale). The quiz should include at least 3 questions designed to test the student's current understanding. These questions should start easy and gradually become more difficult.
- **During the course:** Throughout the course, similar short quizzes should be given frequently to the students. These quizzes would assess the student's current sentiment towards the material and test knowledge. It would also be helpful to deliver assignments and longer, performance based tasks in the app to create a richer dataset.
- **After the course:** Once the course is finished a final series of questionnaires should be delivered to assess the student's feelings about their knowledge. Ideally, these questionnaires can be delivered periodically (immediately after the course, a month after the course, etc.). This would allow the questionnaires to ask questions about how the training knowledge is being applied in the student's job.

Please note, that it is imperative that the student interactions with the evaluation app be painless and quick. Otherwise, there's a high chance that students will not use the app.

Where we are

- Competency model
- Prototype
- Learner dashboard

Next Goals

- Competencies - Synthesize competencies into competency dictionary (determine domains and overlap) and revise performance statements as required.
- Back end
- Questions
- User testing
- Review with stakeholders

Recommendations

- Collect usage data that will give a full picture of who is using the software, in what capacity, to what end. This will provide a clearer picture of who should be taking what kind of training.
    - For example, the report data did not have Rank, so it was difficult to create a full picture of which Ranks were completing which tasks, unless those users were also in the training data or the user data.

- The user data didn’t have L1 or Rank.
- How often are users accessing the software in what capacity? Someone who is accessing the software once a month to review a report has different training needs than someone using the software daily to create reports.
    - Currently, the software reports the last use only

APPENDIX

DIGITAL ACADEMY Practicum Team 6 - Solution Evolution

Problem Framing

Through discussions with our solutions leaders and interviews with other stakeholders, it was clear that there were many problems that could be solved or questions that could be answered including:

- How to measure the effectiveness of a course (WebI Report Writing Course)
- How to help make data-informed decisions
- How to identify the learners who should take the course (skills and/or job-related tasks)
- How to measure life-long learning and the development of skills
- How to increase data literacy skills

Root cause analysis

It was evident from the discussions that no matter how we framed the problem, not having sufficient evidence--or being able to determine if there is sufficient evidence--to make decisions was at the root. Whether decisions are operational or training, determining how to best allocate resources is imperative for DND and requires the proper data and analysis.

While the problems are rooted in the areas of training, performance, and talent management, we realized we needed to go beyond the current capabilities of the systems that manage these functions.

Ideation

Four initial ideas were generated:

1. Data visualization tool for learner training data
2. AI tool that measures true KPIs
3. Learning evaluation and feedback tool
4. Wearable device that measures learning brain activity (among other physical and situational attributes)

Convergence

The solution that was selected was a mash-up of idea numbers 1 & 3. Our tool will collect numerous data on all aspects of learning and also provide feedback through visualizations.

To complete the solution concept, the following questions needed to be answered:

- What are the measures already captured and what does it tell us?
- What are all the potential learning and performance measures?
- What are the gaps between the current and potential measures?
- When should these measures be captured?
- How should these measures be captured?
- What is in it for the end-user (learner) and the organization?
- What constraints have to be considered?

Solution

It was determined that we needed to change the way training evaluation is done in terms of what type of data is collected, when and how:

What	**Reaction level survey** questions on training courses provides the data on learners feelings about the course content, instructors, satisfaction etc. **Comprehension questions or micro activities** provide a gauge on learning knowledge and skills (evaluated against the appropriate competencies) *__Integrated__ activities can be assessed by time to complete as well as other inputs (up to/not excluding instructor feedback) **Job-related (task) questions** give an accurate depiction of a learner's day-to-day activities Feedback on **skills gaps**, recommendations for improvements, and
When	**Pre-training** **During training** **Post-training** (*often the same as pre-training including survey on skill level and confidence)
How	Learners are **prompted** to answer questions and complete short activities Since learners are prompted frequently, a maximum of **2-3 questions** per interaction A **template** interface to integrate training lesson questions (both survey and comprehension) with general skills questions and measures *Learners are prompted to answer many of the pre-training related questions post-training as well to measure increases in knowledge
Why	**Real-time** (data and visualizations) **Interactive**

	Performs both **data collection and data visualization** Data collection supports: training course competency mapping, training (design) evaluation, learner competency proficiency level, learner job task information, user adoption of tools (e.g. WebI), reaction level feedback, transfer level assessment, (possible Results/business level assessment, remediation or fast-track) *Future: data collected is opportunity to use machine learning (AI)*
Who	**Learner** inputs **Admin/Instructor** inputs (integrate training) Learner-specific outputs **Organization** outputs

Using the agile approach to guide our development process, and the WebI reporting writing course data, we fleshed out what our solution would look like in terms of the interactions and visualizations and incorporated this into our pitch presentation. What needed to be considered was what could be accomplished in the timeframe of the practicum? What was feasible for deliverables?

The design and development of a working prototype was not feasible. While we were able to build the front-end user interaction mock-ups, the development of the backend was out of scope. Instead, our team considered the following deliverables for our solutions leaders a win:

- Draft version of a competency framework to evaluate data literacy skills
- All data visualizations created
- Document recommendations for development steps to build the solution (prototype) as well as ideation process/evolution
- Click-through simulation to better communicate the user experience

Q. Is this solution a major shift or a risk? Where does this solution fit between improvement and transformation?

While the standard learning evaluation method of the systems approach to training has provided results that are actionable, the evidence was not holistic in measurement, not timely enough, and not easily visualized without a considerable level of effort.

While the first iteration of this solution may not be completely transformational, it is a considerable improvement in measures and a major shift in the way evaluation data is collected and presented. However, if the use of artificial intelligence is incorporated into this solution in later iterations, it could truly become transformational in the learning industry.

Summary

In the lifelong learning process the proposed solution:

- Collects multiple evaluation data types at smaller intervals
- Provides benchmarks, evaluation standardization, progress assessment
- Prepares user-friendly visualizations
- Presents up-to-date learner data including during training to allow instructors to make mid-course changes to support learner and organizational objectives
- Integrates with course content to allow mapping to competencies (for evaluation of training goals and desired performance levels)
- Utilizes transfer-level data to measure barriers to increasing skill level as well as ROI
- Assesses user-adoption for tools and processes