The Wikirate Project

D6.6.3 Annual development review and WikiRate platform prototype v3

D6.6.3 Annual development review and WikiRate platform prototype v3

Ethan McCutchen | Decko Commons | 8/5/2017

Dissemination level	Public
Contractual date of delivery	Month 42 | March 31, 2017
Actual date of delivery	Month 44 | May 8, 2017
Work package	WP6 | Architecture and Development
Deliverable number	D6.6.3 | Final Development Review
Type	Other
Approval status	PMB Final Draft
Version	V1.0
Number of pages	65
File name	D6_6_3_20170508_V10_DC_Final_Annual_Review.docx

Abstract
A review of the current state of the WikiRate.org platform, with an emphasis on features implemented since the 24-month review. Following a high-level exploration of implemented features, each WP6 task is considered in order. Links to code repositories are provided.

Co-funded by the European Union

History

Version	Date	Reason	Revised by
0.1	26/4/17	Initial framing	Ethan McCutchen
0.2	1/5/17	Further framing and initial content	Ethan McCutchen
0.3	2/5/17	Complete Draft	Ethan McCutchen
0.4	3/5/17	Executive Summary and data updates	Ethan McCutchen
0.5	3/5/17	Fixed errors spotted by Philipp Kühl	Ethan McCutchen
1.0	7/5/17	Finalize	Ethan McCutchen

Author list

Organization	Name	Contact information
Decko Commons	Ethan McCutchen	ethan@decko.org

Executive Summary

Since the 24-month review, there has been a considerable increase in the use of WikiRate.org as a research tool. In accordance, WikiRate.org's design (WP2) and development (WP6) efforts have focused on supporting research-oriented user scenarios.

A new homepage introduces WikiRate as a research platform, invites users into the site by introducing three primary content types (Companies, Topics, and Metrics), and conveys the community's breadth by explaining contributions from different user domains. Updated search and browsing functionality reflects the emphasis on the primary types and seek to welcome users into these content areas. Both Company and Topic pages have been streamlined, de-emphasizing or hiding older qualitative functionality (Reviews, Notes, etc) and focusing instead on their respective relationships with *Metrics*.

Metrics and their Answers have been made considerably more engaging and navigable in the project's final year, both with elegant browsing patterns that support in-place expansions of Answer details and with powerful new sorting and filtering functionality.

To support research of more Answers, we overhauled our project management functionality and now offer *Projects*, which track progress in researching Answers for a fixed set of Companies and Metrics. Projects link to Research Pages, which enable users to find and report related Answers.

Thanks to Source imports that link Sources to specific Companies and Report Types, Research Pages are often able to simplify the process of finding source material dramatically. This supports a new pattern in which project organisers gather source material, and other project participants use that material to find WikiRate Answers. Project organisers can be individuals or Research Groups, which support collaboration on Metric design and Project research.

A new *Badges* system promotes use of these new and improved features. Integrated into a User dashboard as part of redesigned profile pages, badges recognise and incentify contributions and also provide a simple visualisation of a user's contributions.

Fully functional but not yet central to outreach, *Calculated Metrics* complete WikiRate's implementation of its *Rating Concept* (WP4). Formulae, Scores, and WikiRatings each provide distinct interface for mathematically reusing Answers.

In addition to implementation of the above features T6.1 *(Implementation of Features of WP2 and integration of WP5)* efforts included enhancing WikiRate performance through the use of lookup tables (*answers* and *counts*) and concept simplification.

T6.2 *(Improvements to Wagn platform)* further optimised performance with improvements to memory object usage, card fetching, file handling, events, and views. The platform's security, code organisation, and documentation were among other improvements.

The most significant new contribution to *WikiRate System Administration* (T6.3) in the past year has been the integration of New Relic as a performance-monitoring tool. New Relic data drove most of the improvements deployed in T6.1 and T6.2.

T6.4 (*Application Programming Interface (API) and Plug-ins*) did not receive heavy attention this year, though highly useful JSON and CSV links will soon be integrated into the Answer result interface.

All code produced from WP6 is available at GitHub.

Table of Contents

Table of Figures

Introduction

This report will present Year 3 design and development results of WP6 *(Architecture and Development)*. It will first present the WikiRate.org website in its present form, underscoring changes in the project's third year, and then present the underlying technological changes.

This report will generally *not* seek to evaluate quality of the technology or suggest future implementation directions. Those tasks fall instead to *Deliverable 7.7.6 Final Technical Evaluation*. High-level strategic recommendations, similarly, fall within the purview of Deliverable 7.7.3. This report will limit its recommendations to small-scale interface and functionality improvements.

Because there is no annual report for *WP2 (Interactive Design),* which determines the user requirements and technical requirements to be fulfilled by WP6 we have in previous years presented this report as an integrated presentation of the functionality and its rationale, proceeding task by task. In the 24-month review it was suggested that:

> *the report on development should take a more systematic approach (e.g. not mixing UI development with conceptual changes in the conceptual model).*

For that reason, the structure has been updated to begin with a high-level presentation of features implemented, followed by a lower level presentation of the underlying development work. It is hoped that this structure will also succeed in addressing another reviewer request that the report:

> *be more self- contained adding, e.g., screenshots for making it a reasonable contribution for a public readership.*

The report will include abundant images for contextualization purposes and will generally seek to make the subject matter more approachable. No mockup images are included in this report; unless otherwise noted, all screen captures are from live or staging versions of the website.

Implemented Features

In this section we will traverse all major site functionality, emphasizing changes made in Year 3.

Homepage

The homepage was reconceived and simplified in Year 3.

In keeping with the increased focus on WikiRate as a research hub rather than merely a data hub, the top banner now adds an emphasis on research.

The upper body of the homepage is organised around what the consortium decided to emphasise as WikiRate's three *main* content types: *Companies, Topics*, and *Metrics*. The tables on the right introduce the interrelationships of these three content types. On landing, the vertical Companies tab is selected, and users are shown a sample company and some sample metrics for that company. They can scroll through companies, look into a specific value, or browse other companies.

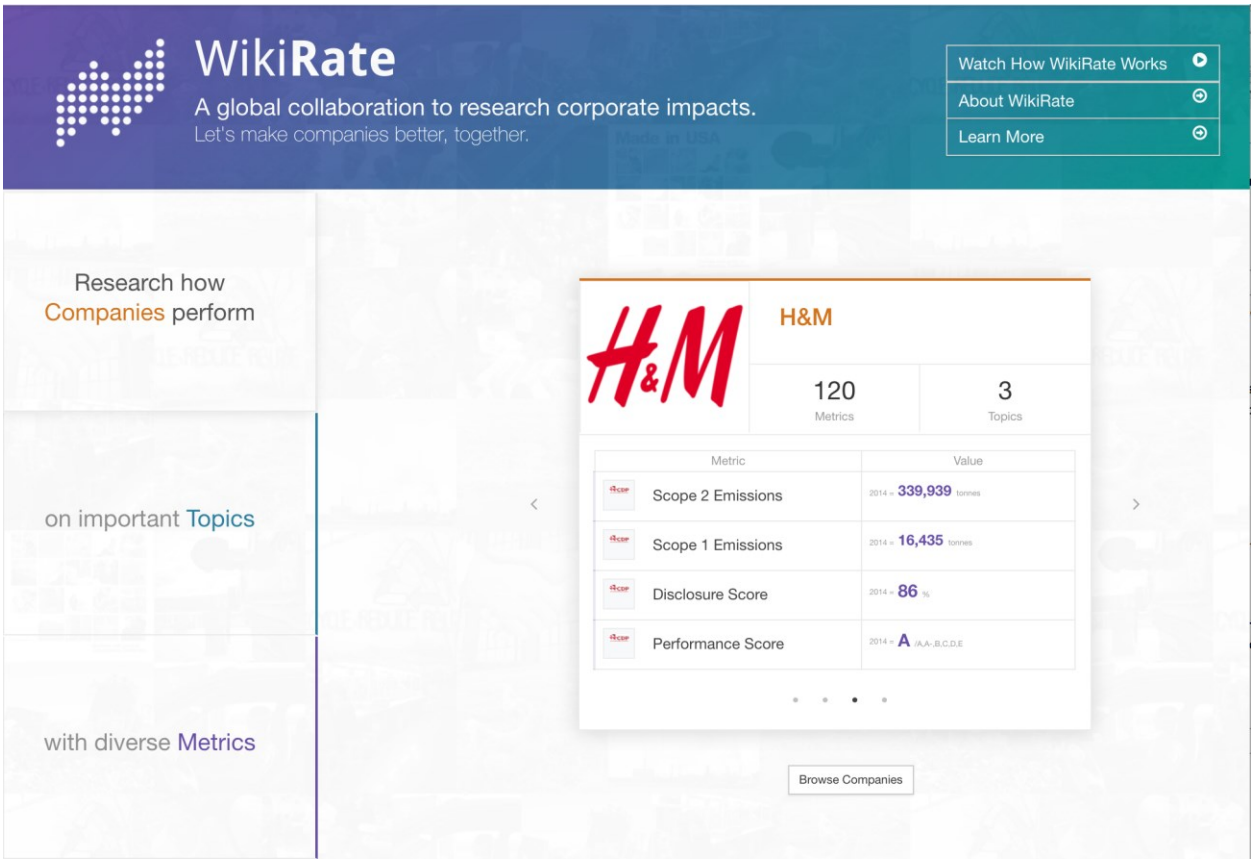


Figure 1 Homepage, Upper Body, Companies

Companies are the primary subject matter of WikiRate.org and are familiar to most users, and therefore they were deemed to comprise the most appropriate content type for initial emphasis.

If the Topics tab is selected, the user is shown a sample topic. Topics, like companies, are chosen for familiarity.

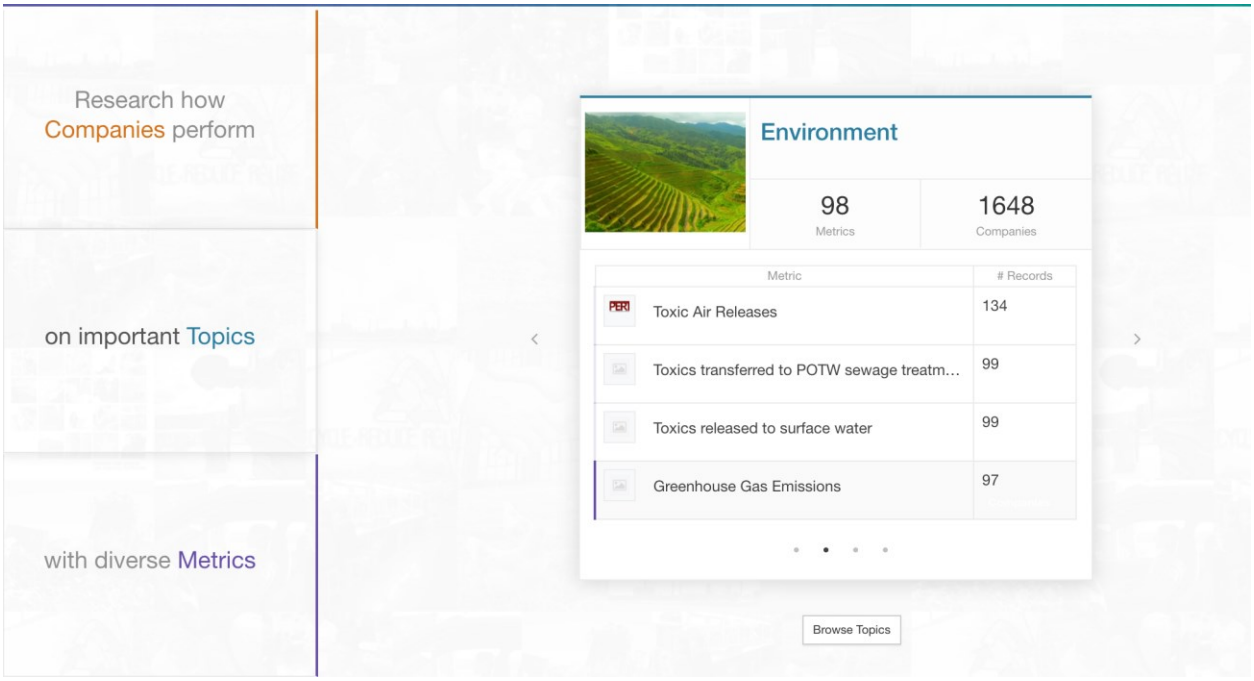


Figure 2 Homepage, Upper Body, Topics

Metrics are generally the least familiar of WikiRate's primary content types and are introduced in the context of familiar companies and topics in an effort both to ground them in known quantities and to prepare the user for their role as the central organising unit of WikiRate research.

The Metrics tab takes this introduction a step further and introduces the key concepts of metric designers (indicating that the metrics are not directly designed by WikiRate) and *questions*, which serve to make metrics more concrete and digestible.

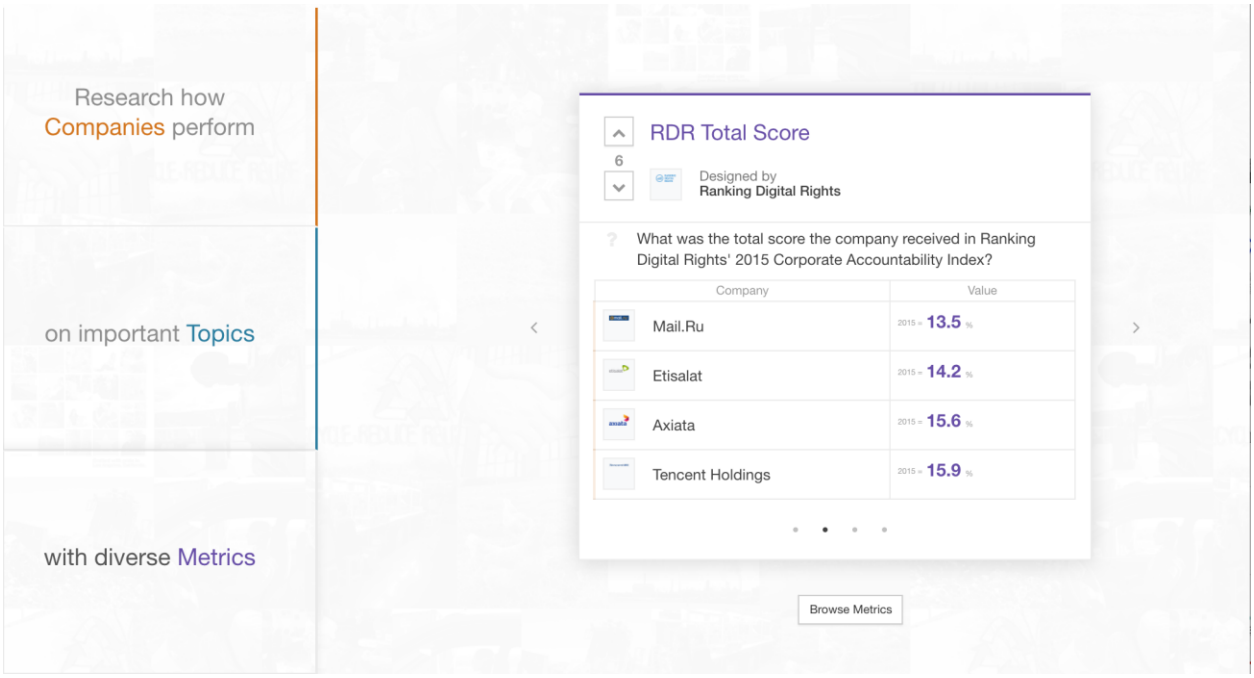


Figure 3 Homepage, Upper Body, Metrics

Early versions of the homepage presented similar information but were navigated by scrolling rather than tabs. The new version supports a greater feel of integration and simpler more readable messaging. It also allows for more emphasis to be given to the second half of the page, which emphasises WikiRate's value to current and future target audiences.

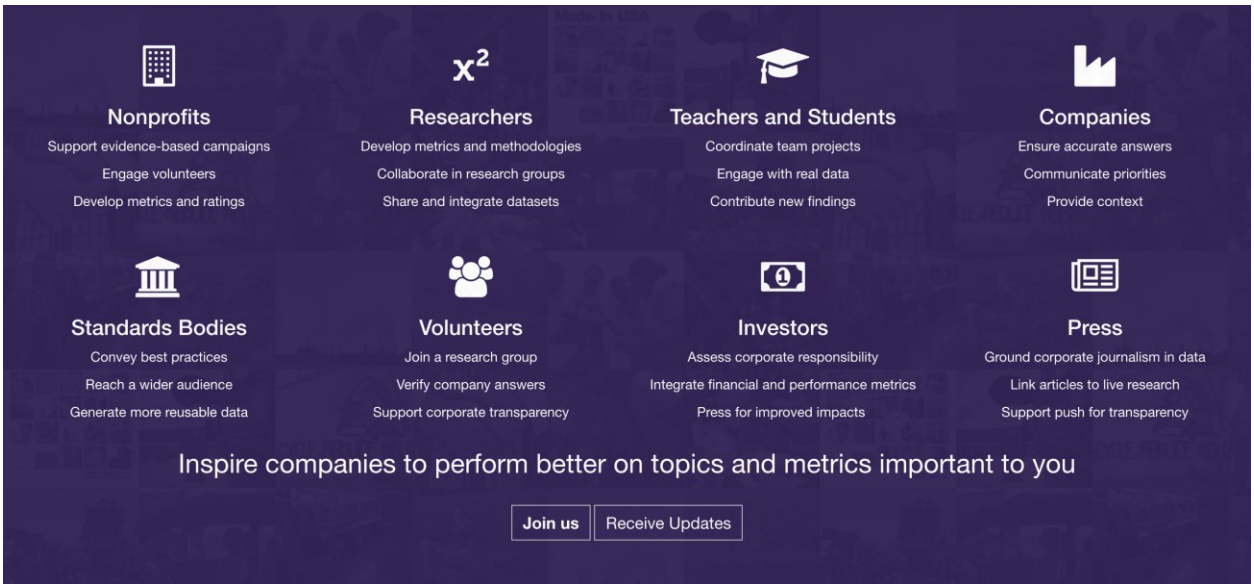


Figure 4 Hompage, Lower Body

The lower section serves to introduce the breadth of the community, and even a cursory inspection conveys the integrative nature.

The key impetus for this section was the need to differentiate audiences and provide links to landing pages for each "user domain" as we have termed the eight groupings. At the time of writing, the landing pages have not yet been completed and activated, but there is no further technical or design work required to make use of this key strategic user flow.

While WikiRate.org remains predominantly a research tool, the primary user scenarios involve use of sufficient screen space to extract data from source material. Mobile technology, for the meantime, remains a secondary interface, but we have still endeavored throughout year 3 to make the mobile interface friendlier. The new homepage, for example, received mobile-specific design and implementation attention:

Figure 5 Homepage, Mobile

Notably absent from the homepage (and generally de-emphasized throughout the site) are WikiRates' qualitative components: *Notes* and *Reviews*. This is because, while WikiRate's metric research functionality has proven useful to our current main audiences (CSOs and academics) in conducting teaching and research, Notes and Reviews have not. A potential timeline for the return of emphasis on qualitative functionality is discussed in D8.8.5.

Browse and Search

Top Menu

This year, the website's top menu was simplified and reorganised in an effort to reduce noise and to make key functionality easier to find. The top menu's position remains fixed even when pages are scrolled. This means navigation is always available, and it helps retain awareness of the WikiRate context even when, as on Source pages, much of the content is served from remote sites.

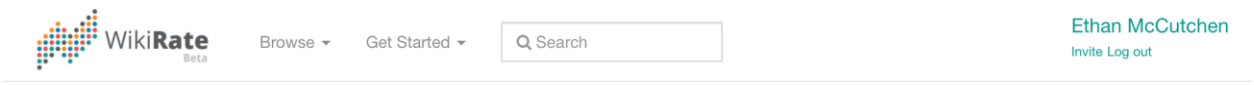


Figure 6 Top Menu

The *Browse* dropdown re-emphasises the three main content types (Companies, Metrics, and Topics) and simplifies navigation for returning users. The *Get Started* dropdown is, obviously, intended to lower the barrier to entry for new users.

Most the content referred to by the Get Started menu has been updated this year; some of it has been overhauled entirely. However, because that content makes use of default Decko content editing functionality and is not directly implemented as part of WP6 (or, to date, designed as part of WP2), it will not be reviewed here.

The browse and search functionality, however, is very much in the domain of WP6 and will be discussed in the subsections that follow.

Unified browsing

At the start of year 3, the various browse pages (Browse Companies, Browse Topics, Browse Metrics, etc.) each had their own distinct look, depriving the site of a sense of continuity. That has been addressed in an effort to make these key browse pages visually compelling and consistent.

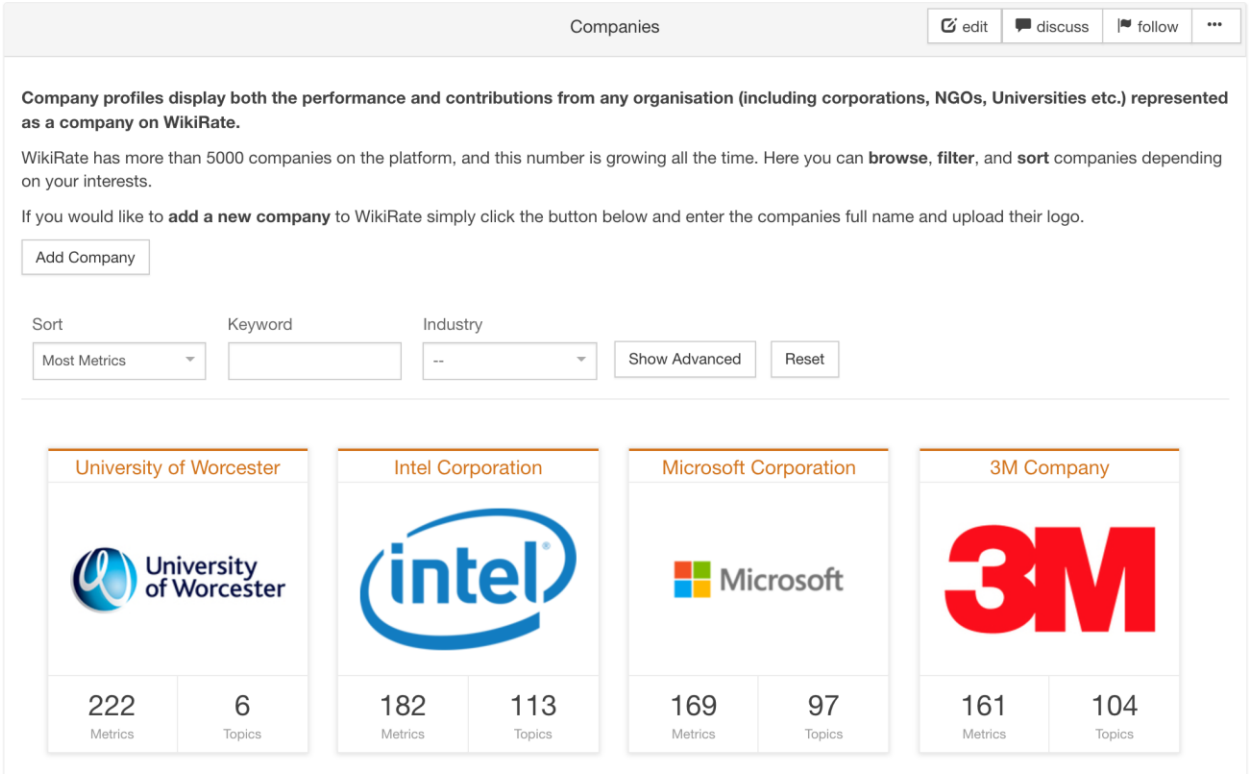


Figure 7 Browse Companies

The *Browse Company* items show the core pattern: for each company item we present the name, an image (logo), and the connections to the other two major content types (Metrics and Topics).

We have presented the most common filtering options: Keyword and Industry. Clicking "show advanced" would also reveal filtering by Project and Topic.

Because of the de-emphasis of Notes and Reviews, a Company and Topic are now considered connected if the Company has an answer for a Metric tagged with that topic.

The Browse Topics page has an almost identical interface. The headers reiterate the coloration pattern introduced on the homepage (Orange = Companies, Blue = Topics, Purple = Metrics), and Topic counts are replaced with Company counts, but otherwise a topic item looks just like a company item.

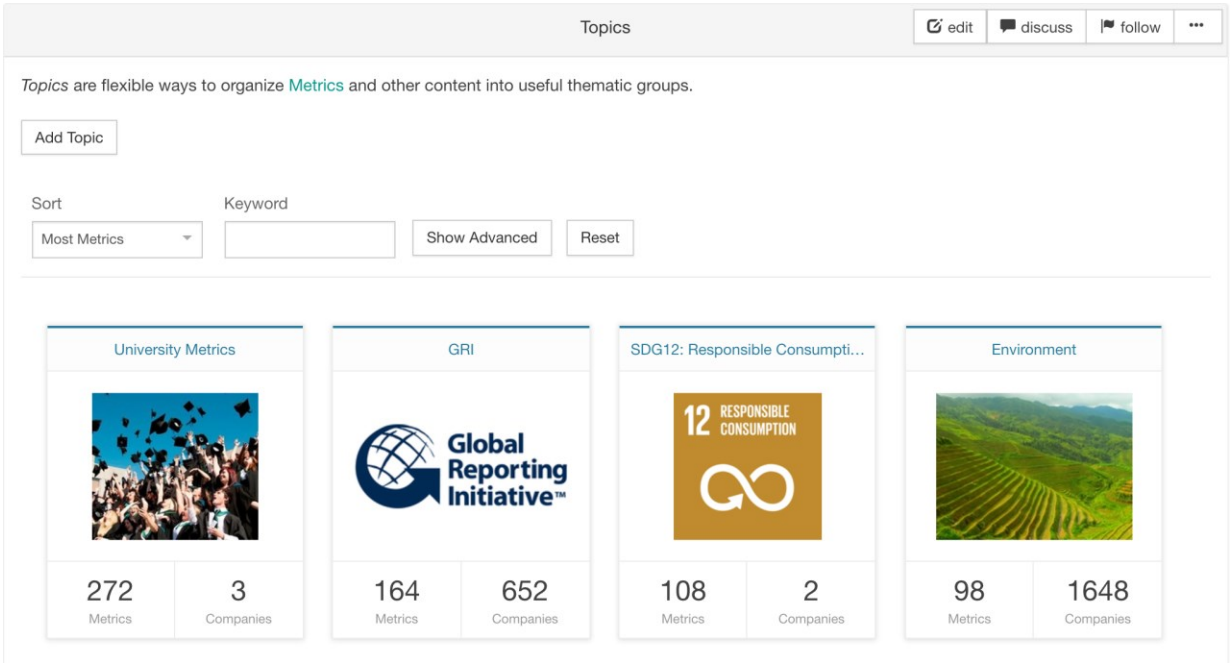


Figure 8 Browse Topics

Metrics deviate slightly from the pattern, because they have no primary image, are voted on directly, and are difficult to understand without a bit more context, particularly the metric's *designer* and *question*. But, clearly, the overall pattern is the same.

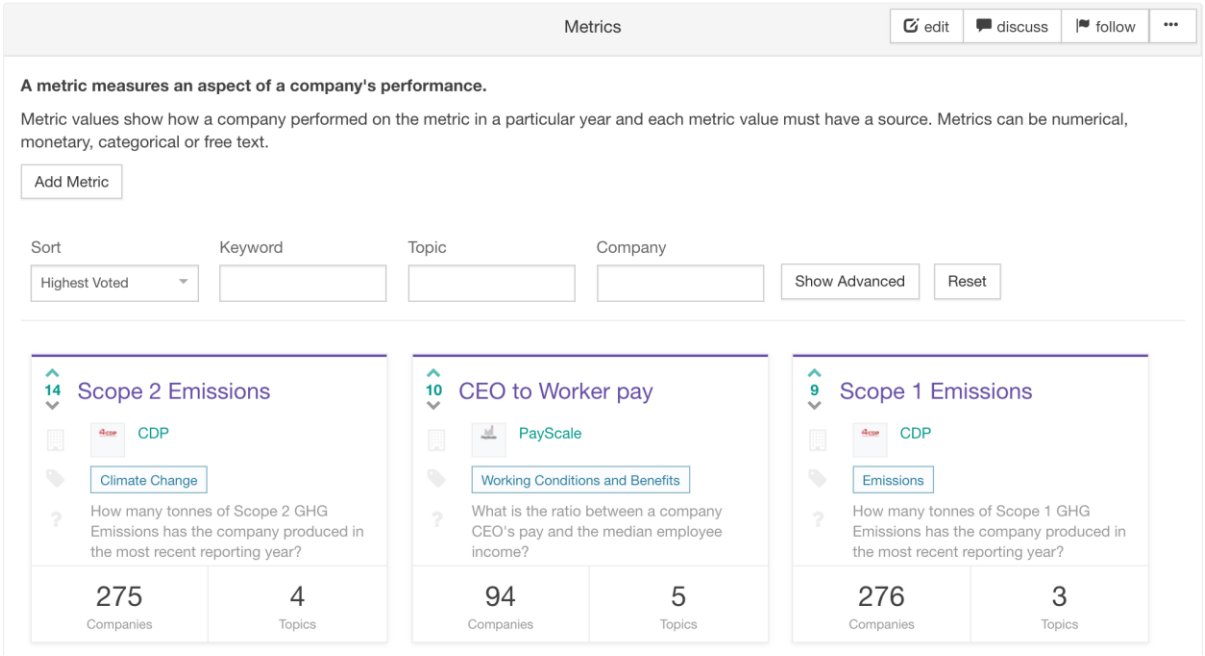


Figure 9 Browse Metrics

Recent Changes

Another item on the "Browse" dropdown that received attention is the *Recent Changes* list. Recent Changes lists are a standard wiki community pattern that allows visitors to review recent activity on the site.

Prior to this year, the Recent Changes list showed every individual card change, despite the fact that a single form submission can involve edits to many different cards. This made the changes quite difficult to navigate.

The new interface reflects the underlying representation of *acts* (a form submission is a single act), *actions* (an action creates, updates, or deletes a given card), and *changes* (alters a card's type, name or content). Each of the grey list items below involves a single act. An act can be expanded to see all the actions associated with that act, and an action can be expanded to see full details of its changes.

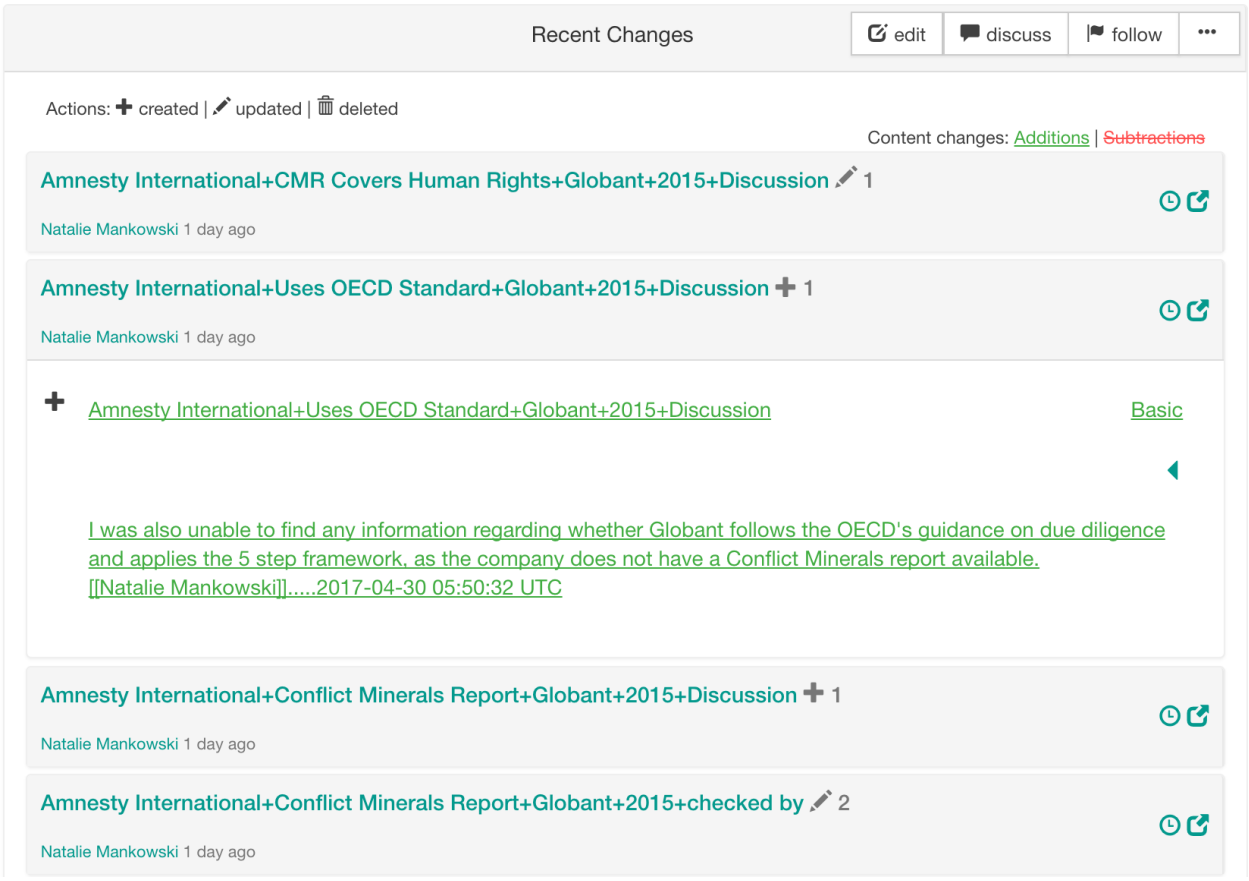


Figure 10 Recent Changes

Search Results

Search results were mentioned in the 24-month review report as an area of particular concern:

> *The search UI should be re-considered, especially the result list. The current version is not very user friendly, since the individual result items are not contextualized well. A better ranking or grouping of the result list, e.g. in different types of resources might also help.*

In keeping with the new strategic emphasis on Companies, Metrics, and Topics, we decided to focus the search page on delivering results for those three types. Search result items have the same appearance as browse items for the respective types.

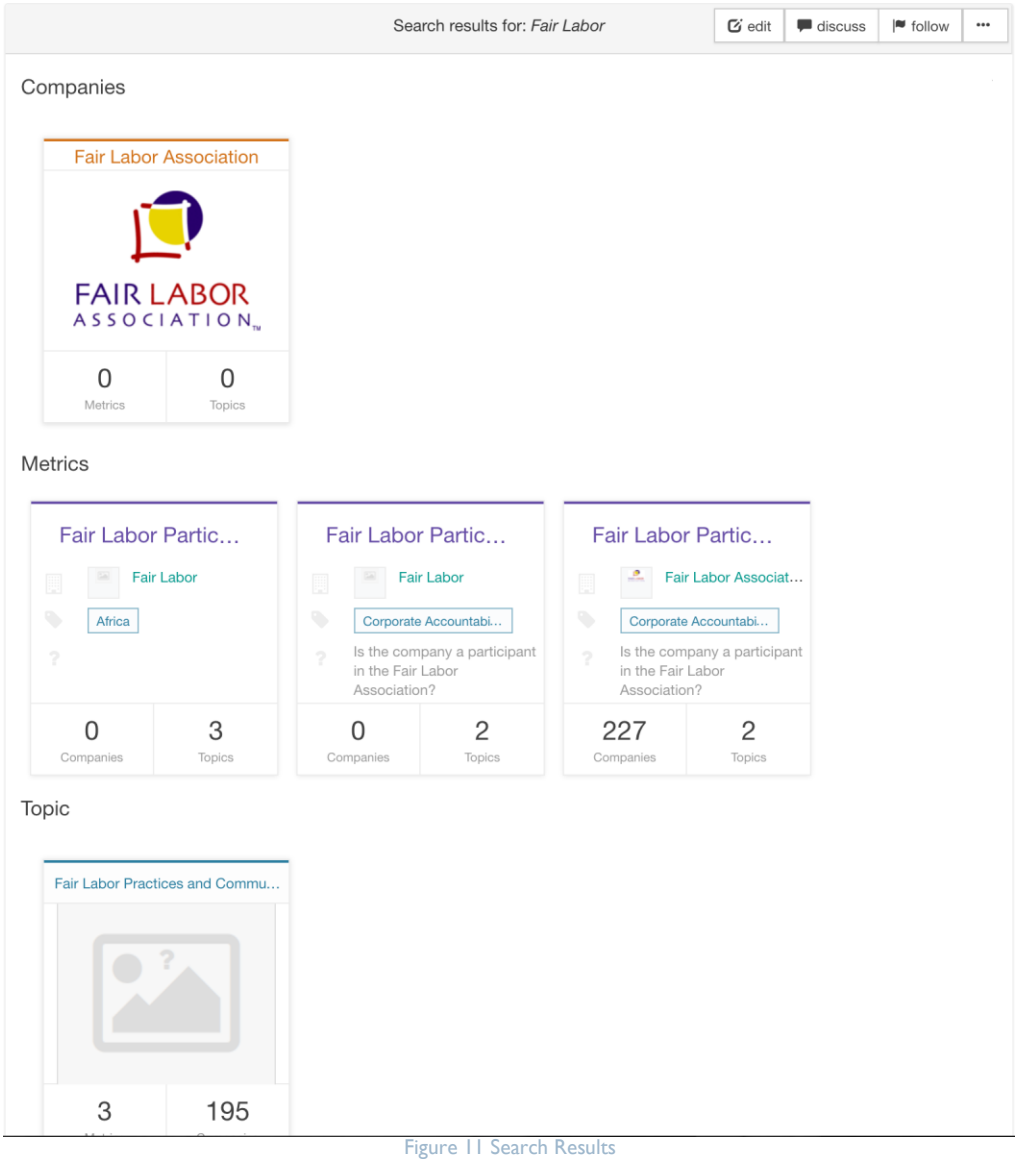


Figure 11 Search Results

Research Metrics

Research Metrics (sometimes referred to in previous reporting as *simple* metrics) are differentiated from *Calculated Metrics* (previously *compound* metrics) in that their values are stored from direct editing or imports, whereas calculated metric values are computed by combining other values.

Research Metrics were first introduced just prior to year 2 reporting and have evolved considerably in the last year.

Metric Page

The metric page has received quite a bit of designer and developer attention, because the rapidly growing database has created new challenges for making the data navigable and usable.

The latest version of the metric page puts primary focus on the answer data, relegating almost all metadata to the right side. The left side contains the Metric's vote count, title, question, sorting/filtering, and distribution visualisation before listing the company records.

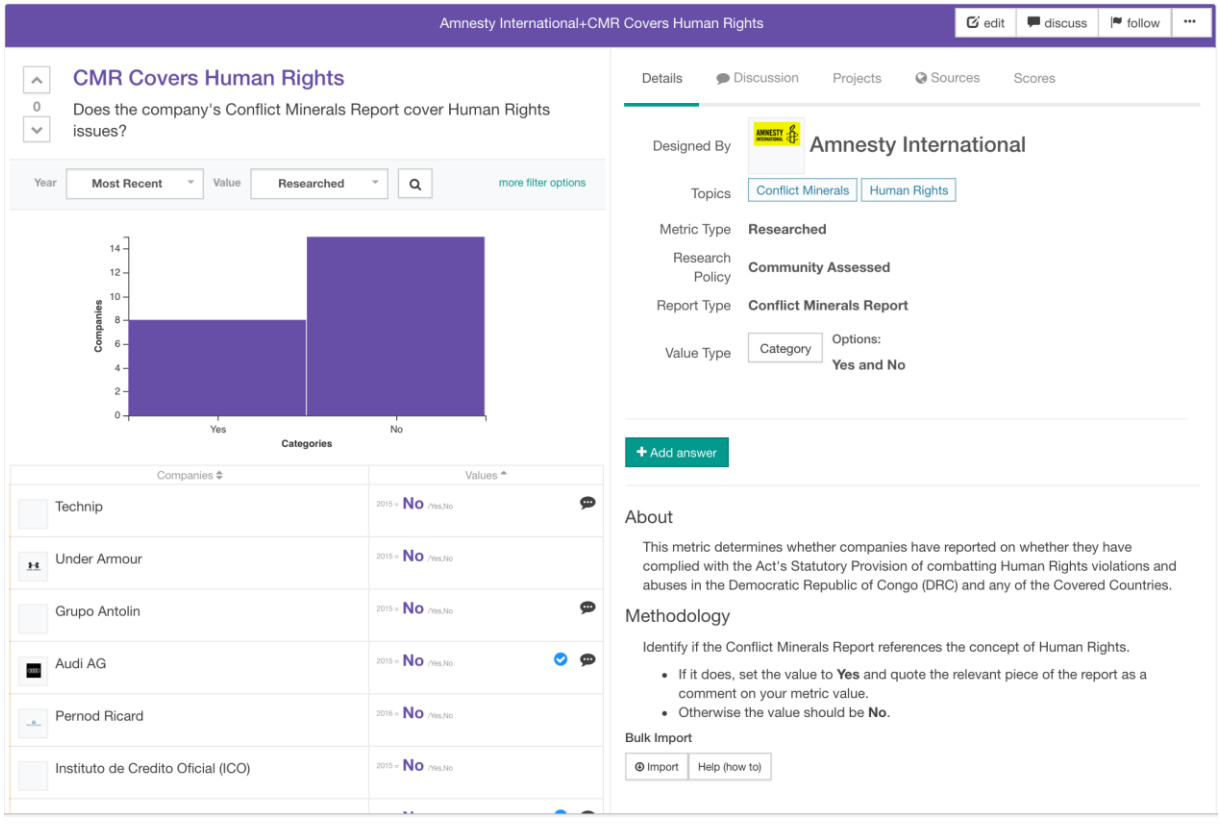


Figure 12 Research Metric page

Though the design seeks to draw users' attention first to the metric's questions and answers on the left, we will first review the right-side metadata here before following the normal navigation routes away.

(Note: because Metrics are such a central feature, because they're new, and because tabs are a recurring pattern, we will present every tab here. We won't be so pedantic with every content type!)

Details Tab

The meanings of the basic *Research Metric* metadata fields are described in *Deliverable 4.4.2 Ratings Concept Evaluation*. To that we can here add that the rationale for their new presentation is that by condensing the short fields into a short view at the top of a *Details* tab, we can often present the *Methodology* above the fold (where the *About* section is not too long) and can avoid hiding any valuable information or interface too far down the page.

Whereas the *About* and *Methodology* fields are typically edited in place, the short fields can all be edited either (A) in place or (B) within the edit form that is presented when one clicks the "Edit" button at the upper right:

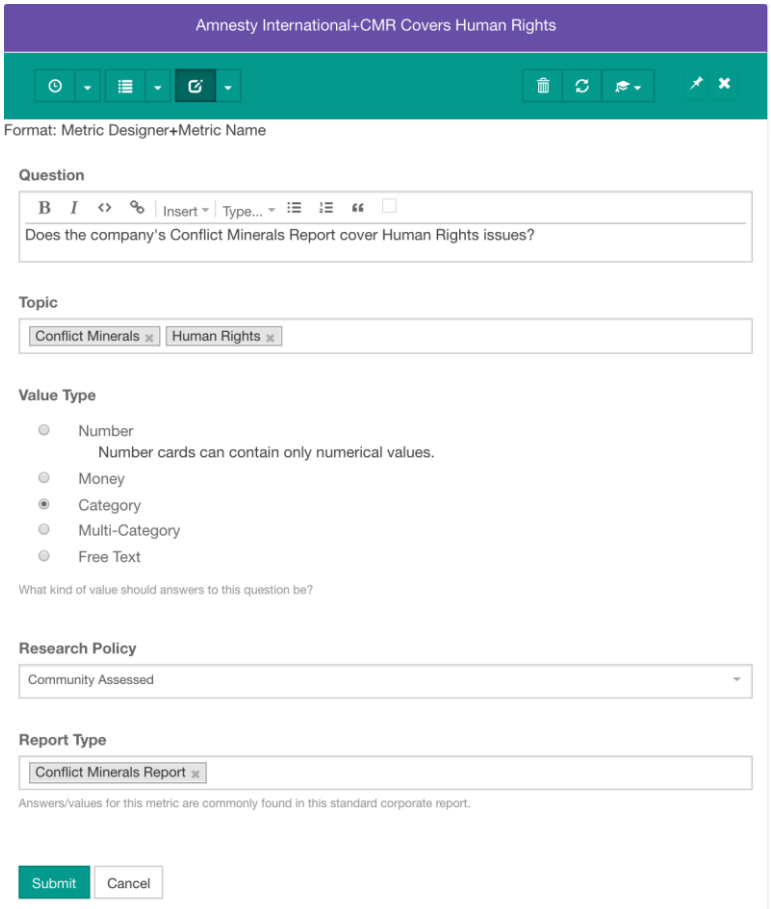


Figure 13 Research Metric, Edit view

Discussion Tab

The discussion tab on Metrics pages is included here more for consistency than any exciting feature, though it worth noting here that discussions on WikiRate cards are cards themselves and can be accessed through the "discuss" card menu item whether or not they are featured on the page. This uniformity creates an opportunity for deeper integrations of discussions into WikiRate user patterns – an opportunity expected to receive greater emphasis later this year.

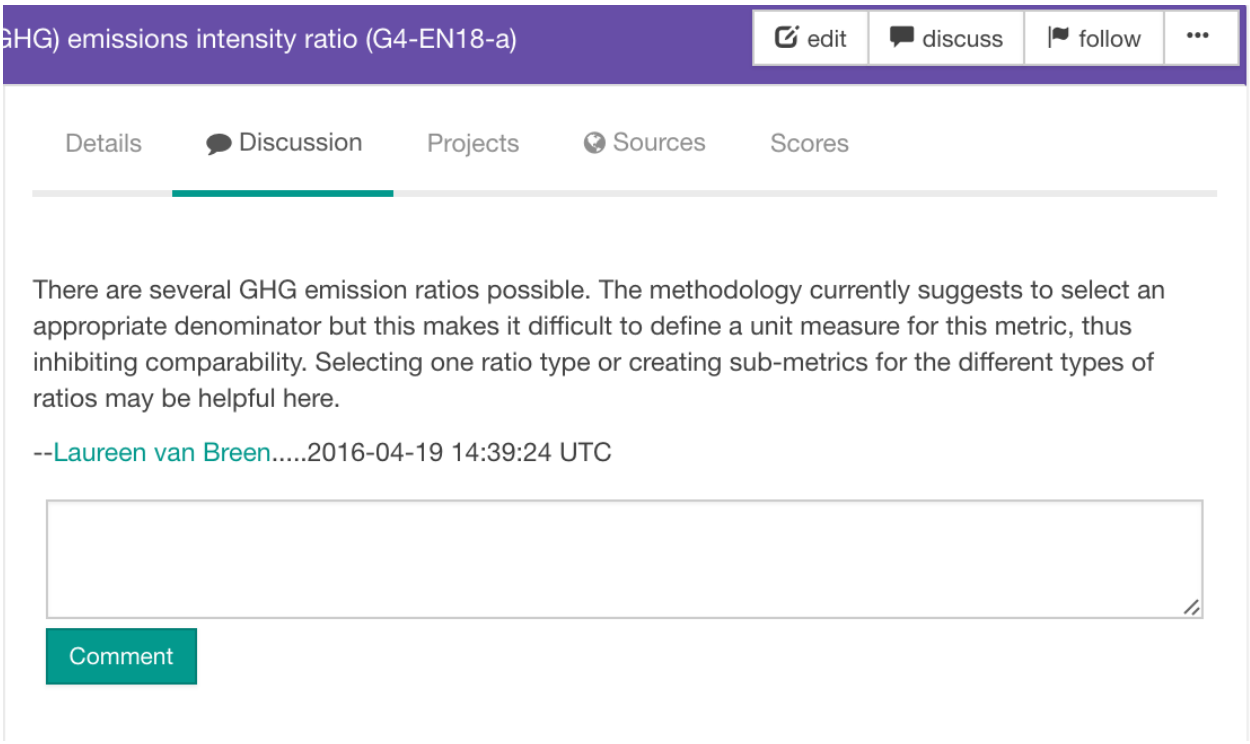


Figure 14 Research Metric, Discussion Tab

Projects Tab

As will be discussed in the *Projects* section, each project is connected to a specific list of Metrics and Companies. This tab shows all the projects connected to a given Metric:

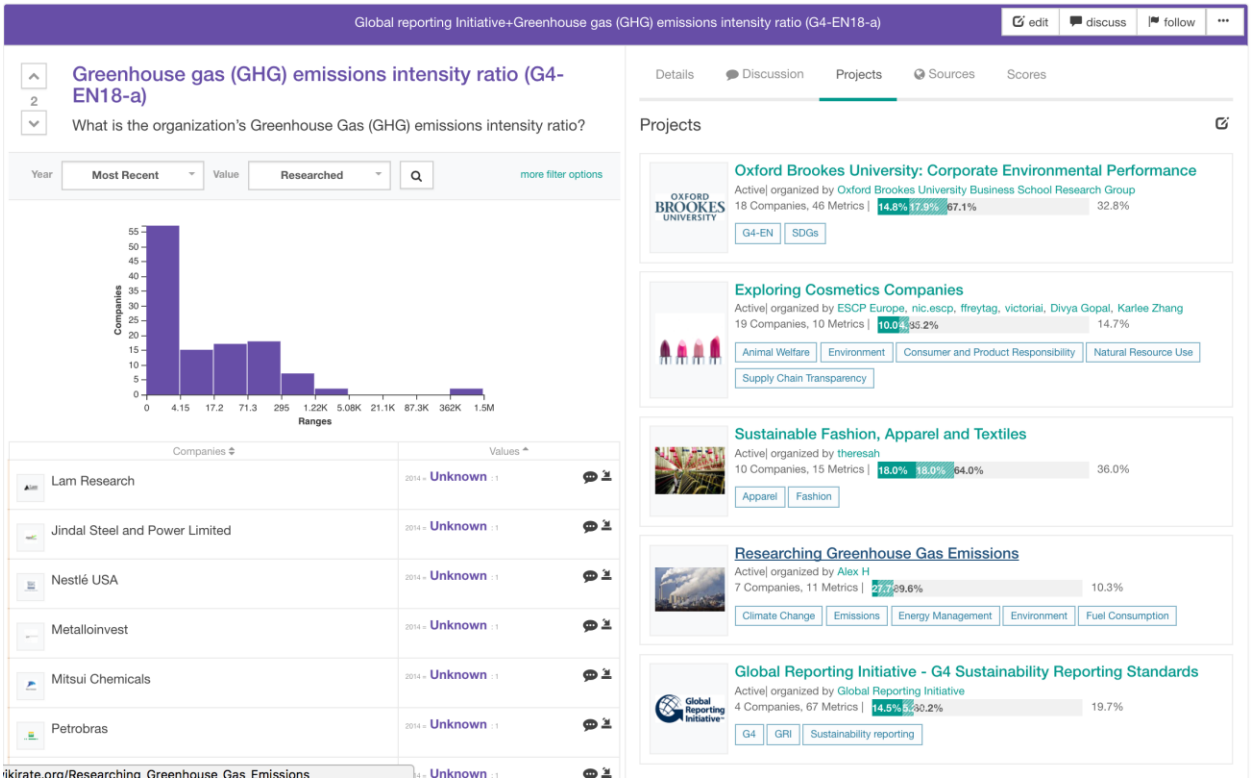


Figure 15 Research Metric, Projects Tab

This metric is particularly interesting, in that it shows how a single metric can be part of many different projects.

Sources Tab

The *Sources* tab shows all sources cited by any Answer to the current Metric.

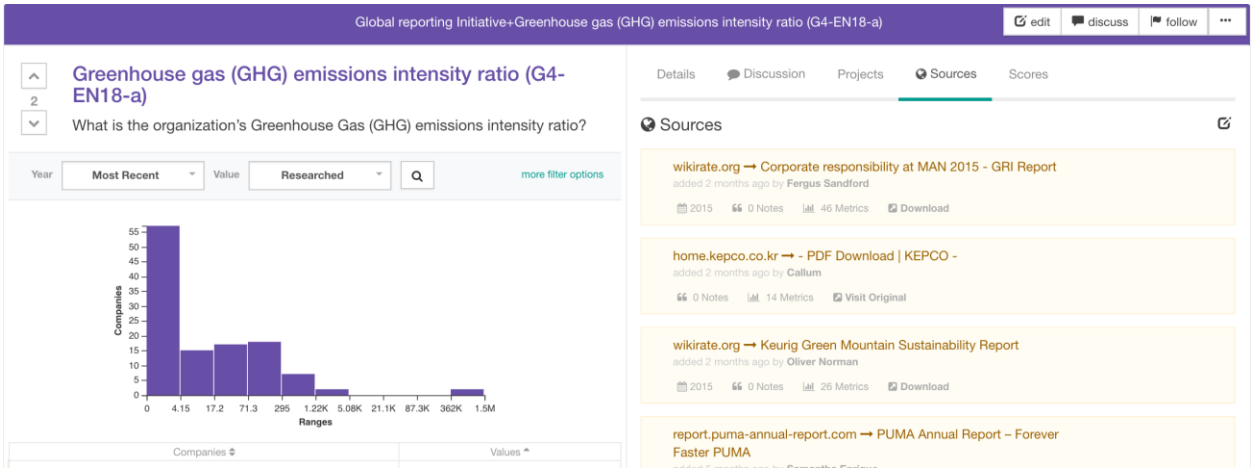


Figure 16 Research Metric, Sources Tab

Every metric answer must have a source. While it's possible for a source to support many answers, typically this tab could have quite a lot of sources for a well-populated metric. When navigating to a specific source, it is typically much easier to follow from a specific answer (see *Expanding Records and Answers* below). For now, however, this tab is maintained because it provides a quick overview of the kinds of sources used to research a given metric.

Scores Tab

This tab is part of the Calculated Metrics framework as such is not in heavy use as of yet.

For Research Metrics to be used in WikiRatings, they must be scored (normalised to a 10-point scale). Different researchers may score the same metric in different ways. At present the tab simply shows the users who have scored the metric and links to the Score in question.

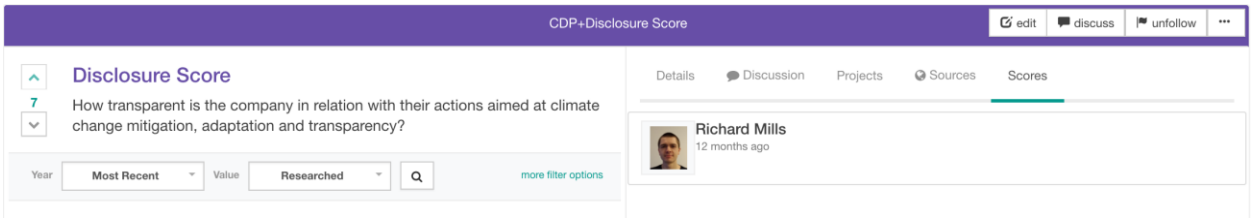


Figure 17 Research Metric, Scores Tab

Sorting and Filtering

A primary concern in year 3 was making WikiRate's increasing abundant data easy to use and navigate. The sorting and filtering on Metric pages was particularly important, because it really enables metrics to be used as a comparison tool.

In the collapsed view, results may be filtered by *year* or by *value.* By default, the results only show *researched* values, but other filters include values that have not yet been researched. Such values prompt users to add answers.

Clicking "more filter options" adds the additional options to sort by Company, Industry and/or Project.

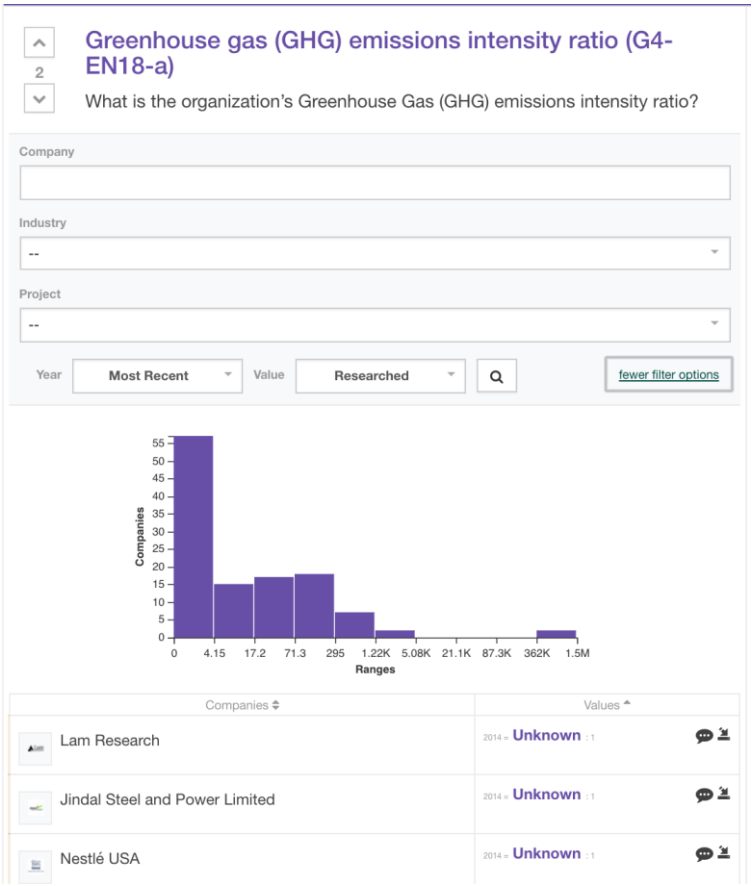


Figure 18 Metric, Sorting and Filtering

Changing the filters will update not only the results but the distribution visualization as well. And the visualization is itself part of the filtering interface: clicking on a given bar of the visualization will limit search results to those included by that bar (and will present a new graph with the breakdown within the applicable range).

Expanding Records and Answers

To clarify terminology, a given company-metric combination can have only one *Record*, but a Record can contain a different *Answer* for each year.

When viewing a Metric page, only one answer per company is shown at a time on the left side. However, by clicking on *Answer*, one can see the full *Record* for that company/metric. Eg, the following shows a view in which the user has clicked on the "General Electric" on the left to expand the full record for General Electric's foreign tax expense benefit on the right.

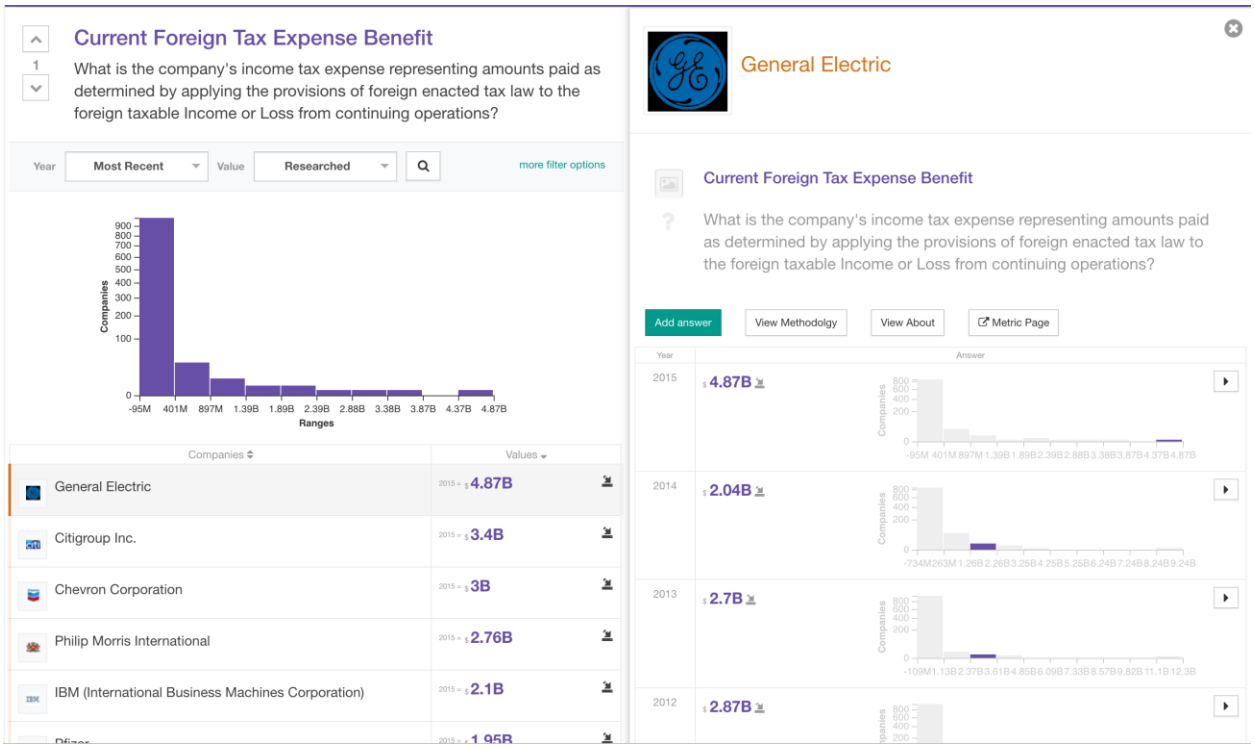


Figure 19 Research Metric, Expanding Records

The interface allows users to gain at a glance both the Company's historical performance for and market context for the answer on a year-by-year basis. Additional Answers can be added directly from this context via the green "Add Answer" button.

Clicking the arrow on the right of existing Answers will expand them to reveal additional information, including who added or updated the current value, whether a *double check* has been performed or requested, the source citation, and any comments specific to this answer.

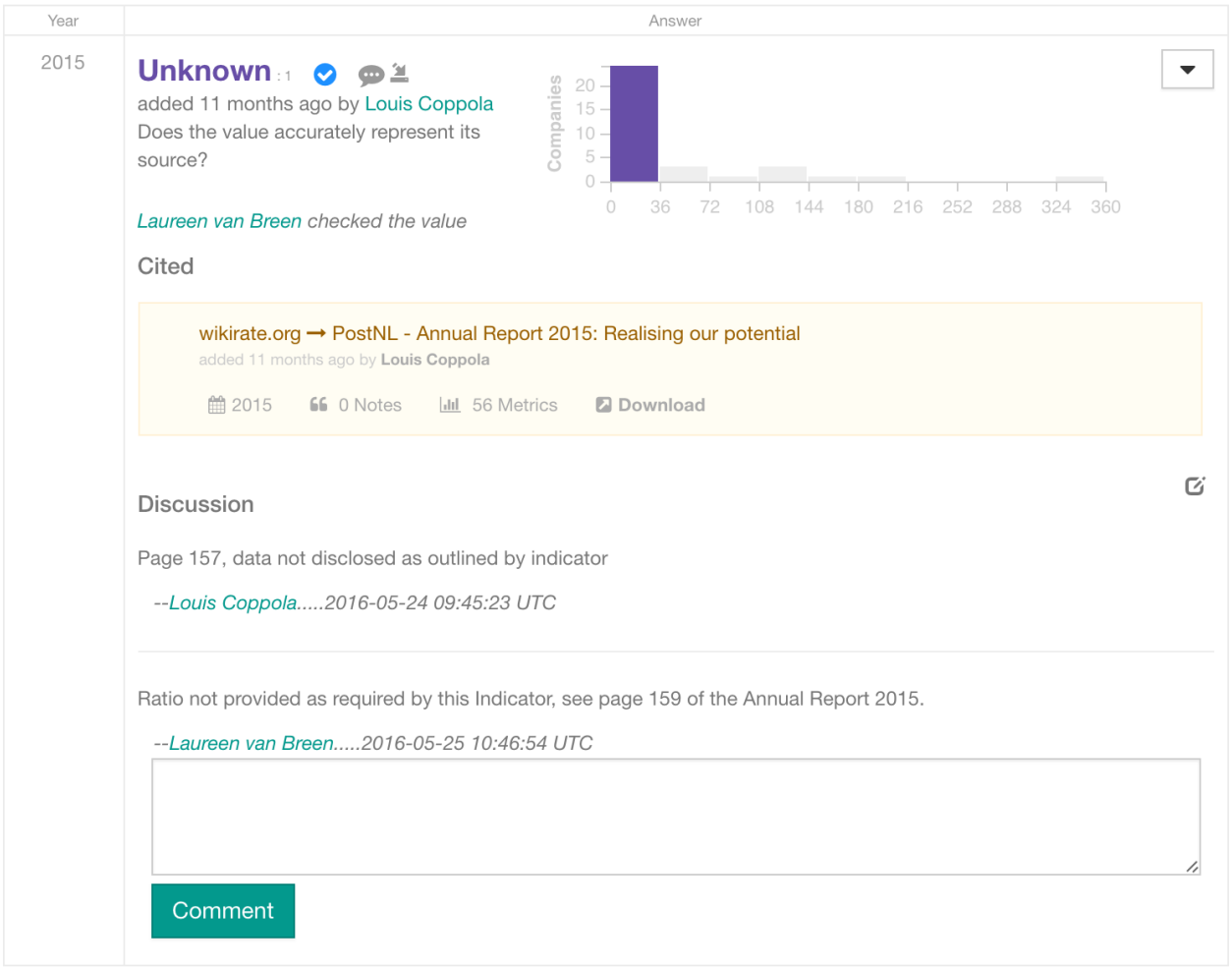


Figure 20 Research Metric, Expanded Answer

The icons on this answer, which also appear on the listing view, indicate that the record has been double checked (the blue check) and commented on (the speech bubble). The third and newest icon labels values added by import.

If a value has not been double checked, a user can request that someone else do so; this functionality was offered in support of users who feel doubt about a value they have added, in hopes that it might give them more confidence to give a good faith effort at a response even without 100% confidence in its accuracy. *Check requests* are indicated by checks in red circles, as in the following listing:

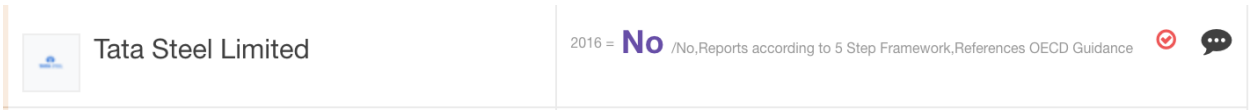


Figure 21 Double Check Requested

Companies

Companies are WikiRate.org's most seminal subject matter, and it is a long-term ambition of the project that standard web searches for company names should return WikiRate pages second only to the company's own webpage. Company pages did change considerably this year, but because their changes mirror and were largely driven by metrics, we will not go into the same level of detail.

Like Metric pages, Company pages are dominated on the left by Answers and on the right by metadata. There are no visualizations on the left, of course, because each metric is different. But expanding an Answer on a company page gives essentially the same view as expanding one on a metric page, enabling the navigation pattern of metric ➔ company ➔ metric ➔ company…

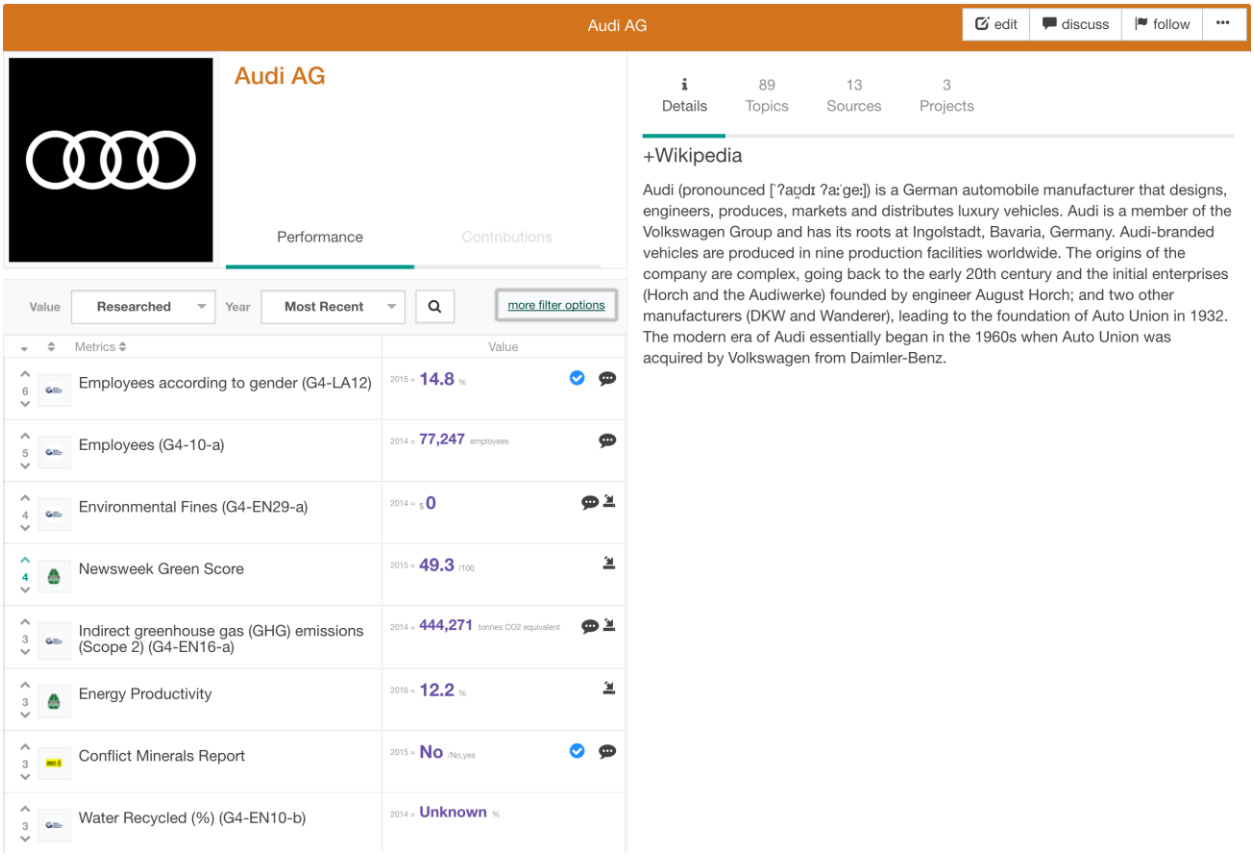


Figure 22 Company Page

At present, the *Details* tab contains only a Wikipedia article, which is dynamically loaded and reflects the most recent Wikipedia content. Other remote-service details-tab integrations (funded separately) are scheduled for release in coming months.

The *Sources* and *Projects* tabs directly mirror those shown on Metric pages. Metrics, which are directly tagged with topics, do not have a Topic tab. The metric counts link to Topics page with its filter open to filter by the current company.

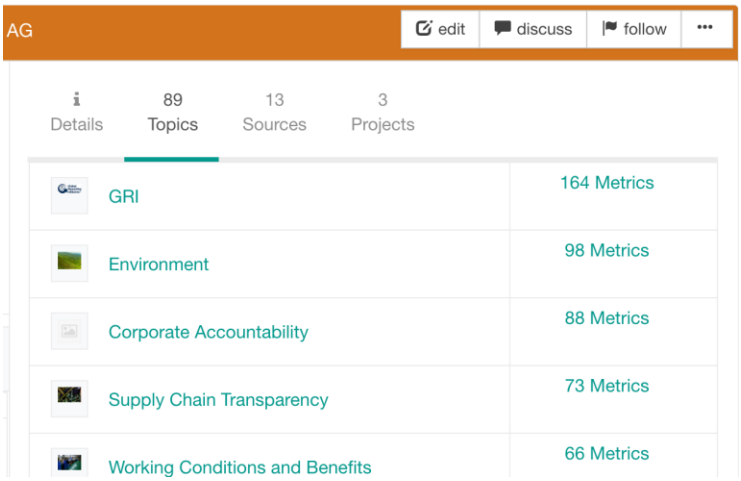


Figure 23 Company Page, Topics Tab

The sorting and filtering also has subtle differences from the Metric pages stemming from the additional fields on metric pages. Company pages can filter for topic, research policy, metric votes, and metric type and can sort by metric votes or designer.

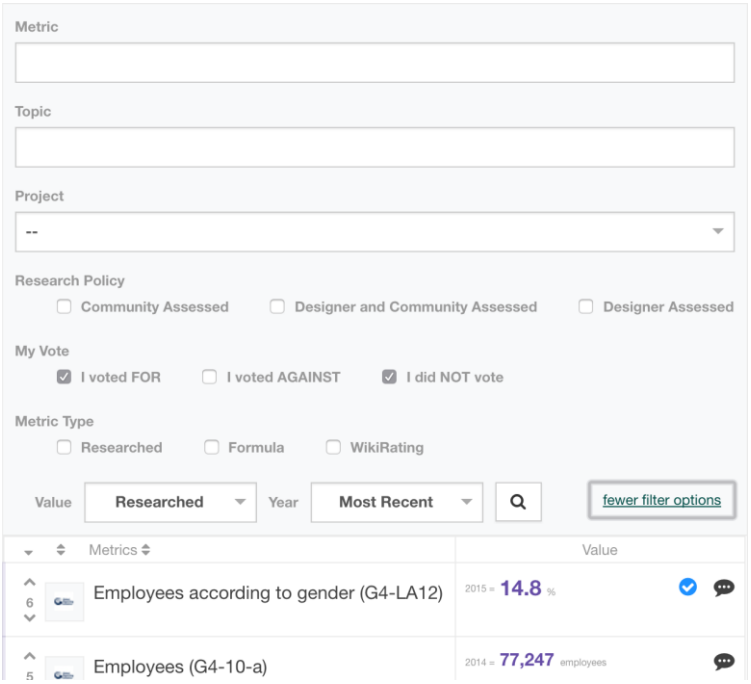


Figure 24 Company Page, Filtering

The most significant difference between Companies and Metrics, however, is that Companies can contribute to WikiRate. We have plans to add official corporate accounts that can do most anything normal users can. However, at present the only things companies can do *as companies* (as opposed to as individuals who work for them) are design metrics and organise projects/research groups.

Figure 25 Company Page, Contribution Tab

Note that the same Company can both frame research (through metric design) and be researched (as a subject of metric answers).

Projects

In Year 2 we introduced a rapidly assembled set of project management functionality organised into Campaigns, Projects, and Tasks. The revised *Projects* are so drastically simplified as to be unrecognisable to any fans of the old system, had any existed. Now Projects simply monitor answers at the intersection of a list of metrics and a list of companies.

Progress can thus be easily calculated and viewed on a project-wide basis. The Project page features a version of this bar with a legend and short description that conveys the project's scope.

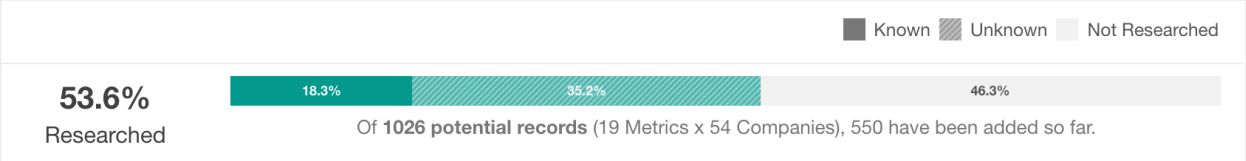


Figure 26 Project progress bar

This same project-wide progress bar appears on listings of the project throughout the site, eg on Company pages, Metric pages, and Profile pages. Additional progress bars on the Progress page show progress on a per-metric basis, indicating how much of the research on the project's companies have been completed:

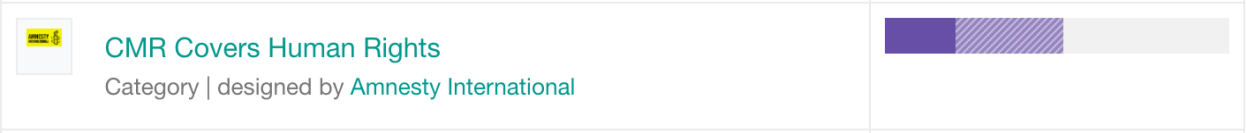


Figure 27 Project Page, metric progress bar

And, finally, there are project bars showing how many of the project's metrics have been addressed for a given company:

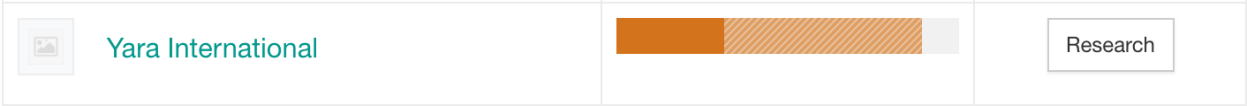


Figure 28 Project Page, company progress bar

The "Research" button here is the primary call to action on Project pages. It leads to the *Research Page* interface described in the ensuing section.

The following provides a high-level view of a Project pages in which overall progress, progress by metric, and progress by companies.

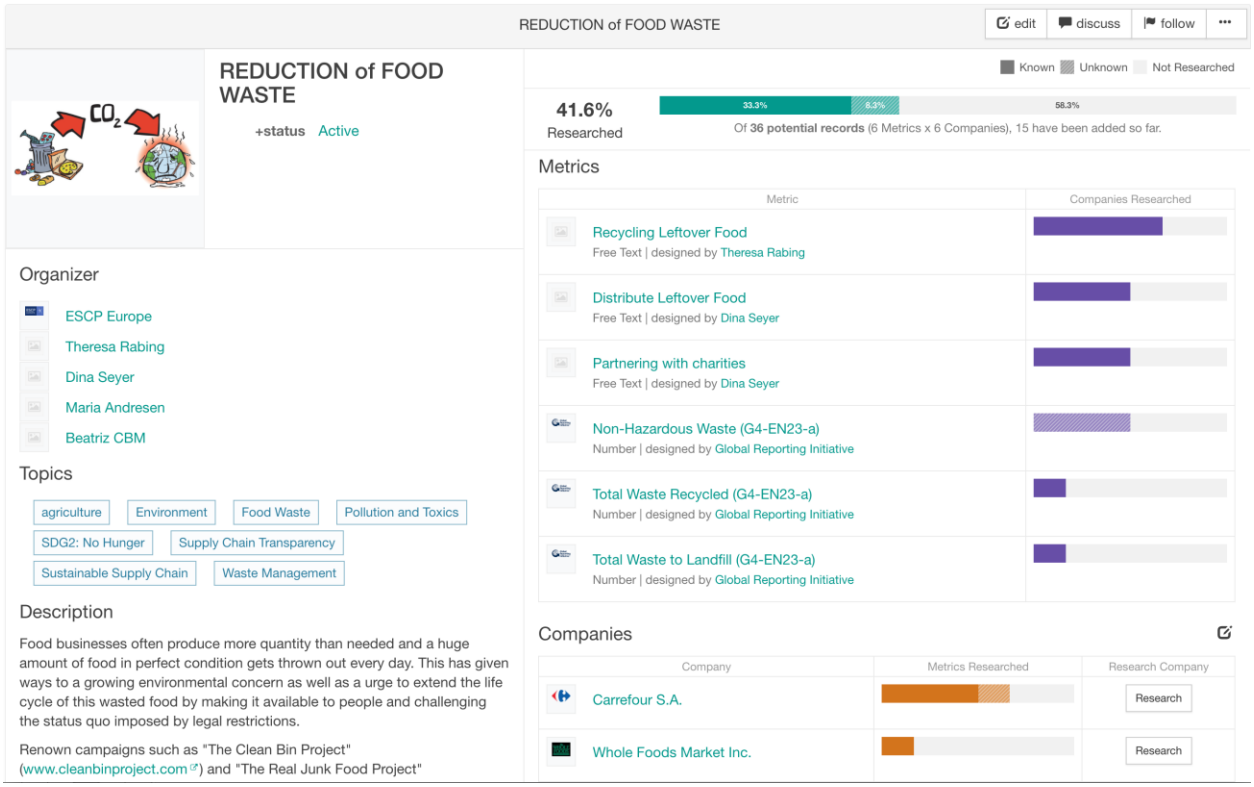


Figure 29 Project Pages

However, the emerging pattern is that most Projects tackle large lists of related metrics, and in doing so push company progress bars below the fold. Given that the "Research" buttons are the primary call to action, this is a significant problem and is already creating an impetus to revise the Project page layout.

Also motivating an upgrade is the need for better year handling. Note that a company/metric is currently considered "researched" when there is an Answer for *any* year. Projects will soon need the capacity to specify target years, particularly because one likely pattern is for organisers to research the same metrics annually.

Research Pages

WikiRate.org has two engines for generating data: one is imports, and the other is the Research Page interface. This interface can be reached from many places. When reached from the "Add Answer" button on a Research page, it looks like this:

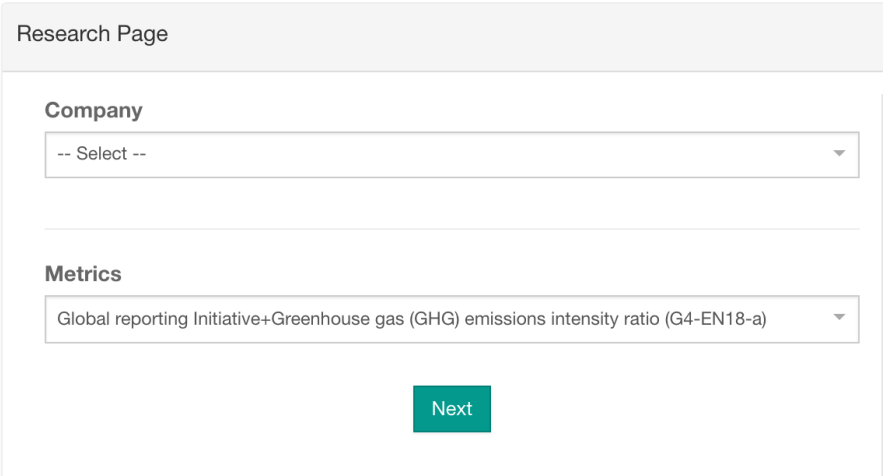


Figure 30, Research Page from Metric

When reached from the "Add Answer button on a Company page, the other field is prepopulated. Once a company and metric have both been chosen – either via this interface or by an "Add Answer" button on a *Not Researched* value result on a Metric or Company page – the user is taken to the following interface:

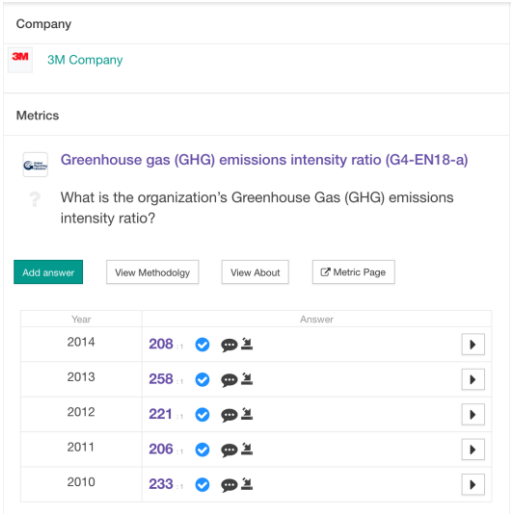


Figure 31 Research Page, View Record

This particular record already has answers for 2010-2014. Expanding an Answer provides a view that looks very much like an expanded answer on a company or metric page. But the page's core functionality is reached through the "Add Answer" button, which presents the following form:

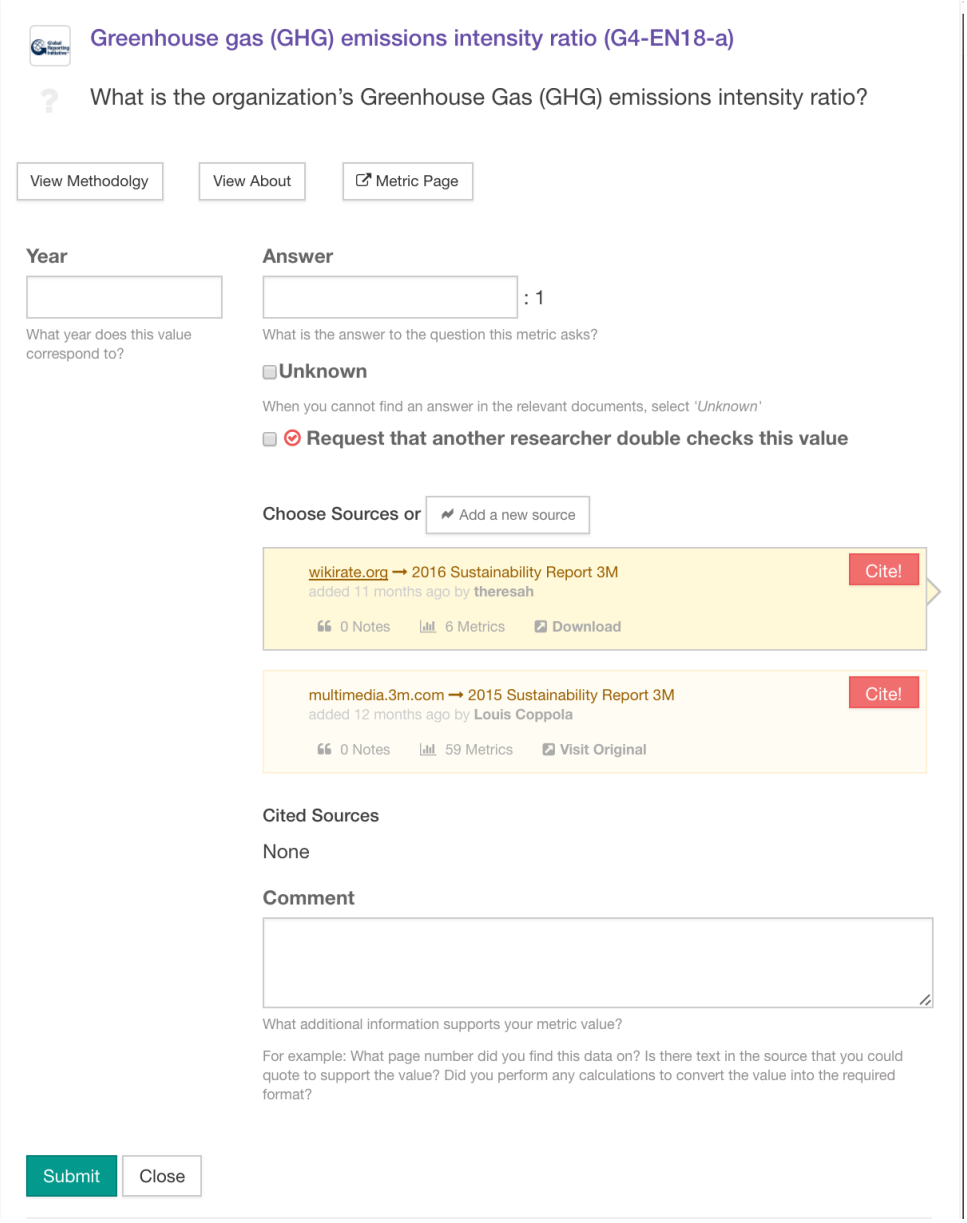


Figure 32 Research Page, Add Answer Form

The form mimics the typical answer layout on the site, with years to the left and answers atop the right. Options to confirm an answer as Unknown or to request a double check aid in data quality. And sources already appear as they do in citations.

The above forms are actually only a partial view of the webpage. The right side of the webpage is reserved for sources, so that the source and the form being populated from it can be viewed at once:

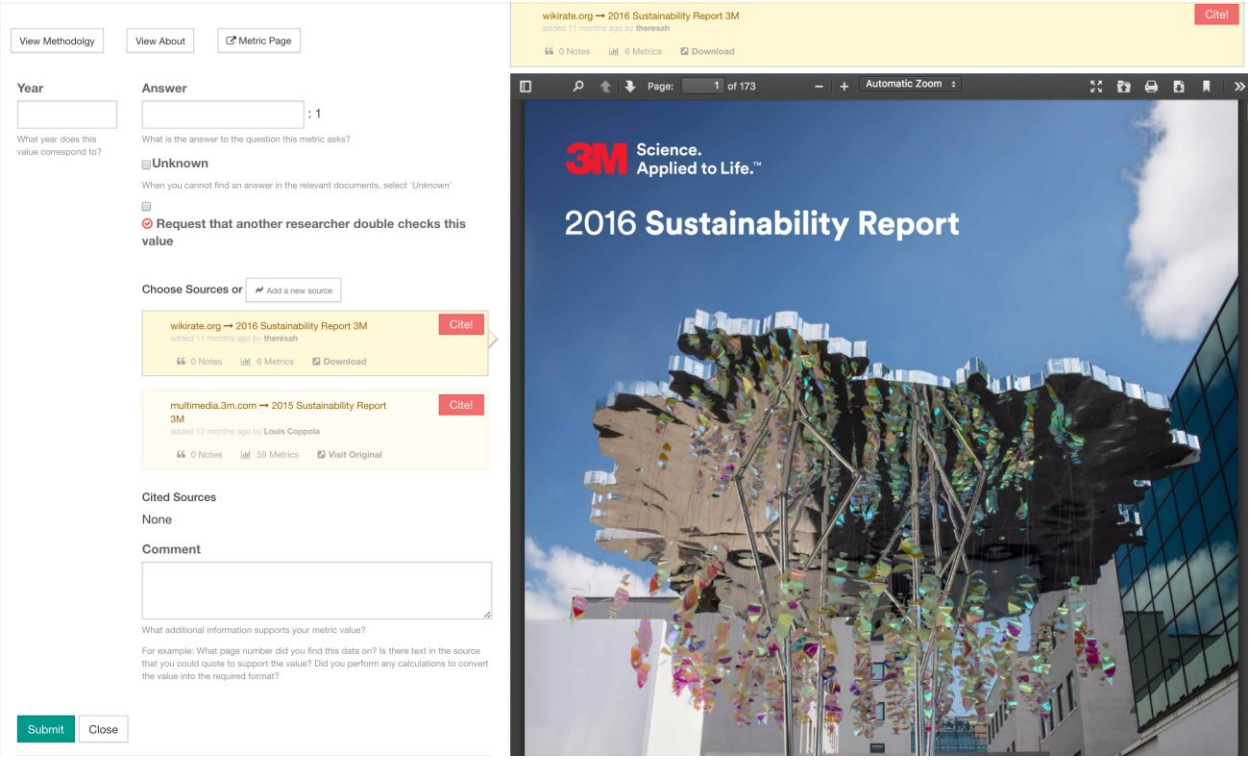


Figure 33 Research Page with Source

Clicking on a "Cite!" button will attach the source to the form.

One regular problem with the current interface is that many key data sources (such as the US Securities and Exchange Commission) block the remote inclusion of their webpages. To date WikiRate does not store backup copies of all web sources, but we plan to change that practice. Capacity to render our own version of source material would resolve this problem.

The Research page adds one more level of sophistication when reached from the "Research" buttons on project pages. In these cases, the page covers not a single record but records at the intersection of (A) a given company and (B) all the metrics associated with the project.

For example, clicking on the *Research* button beside the company *Sherwin-Williams* on the *Investigating Mineral Sourcing Practices* project will take you to a page with the following in the left column:

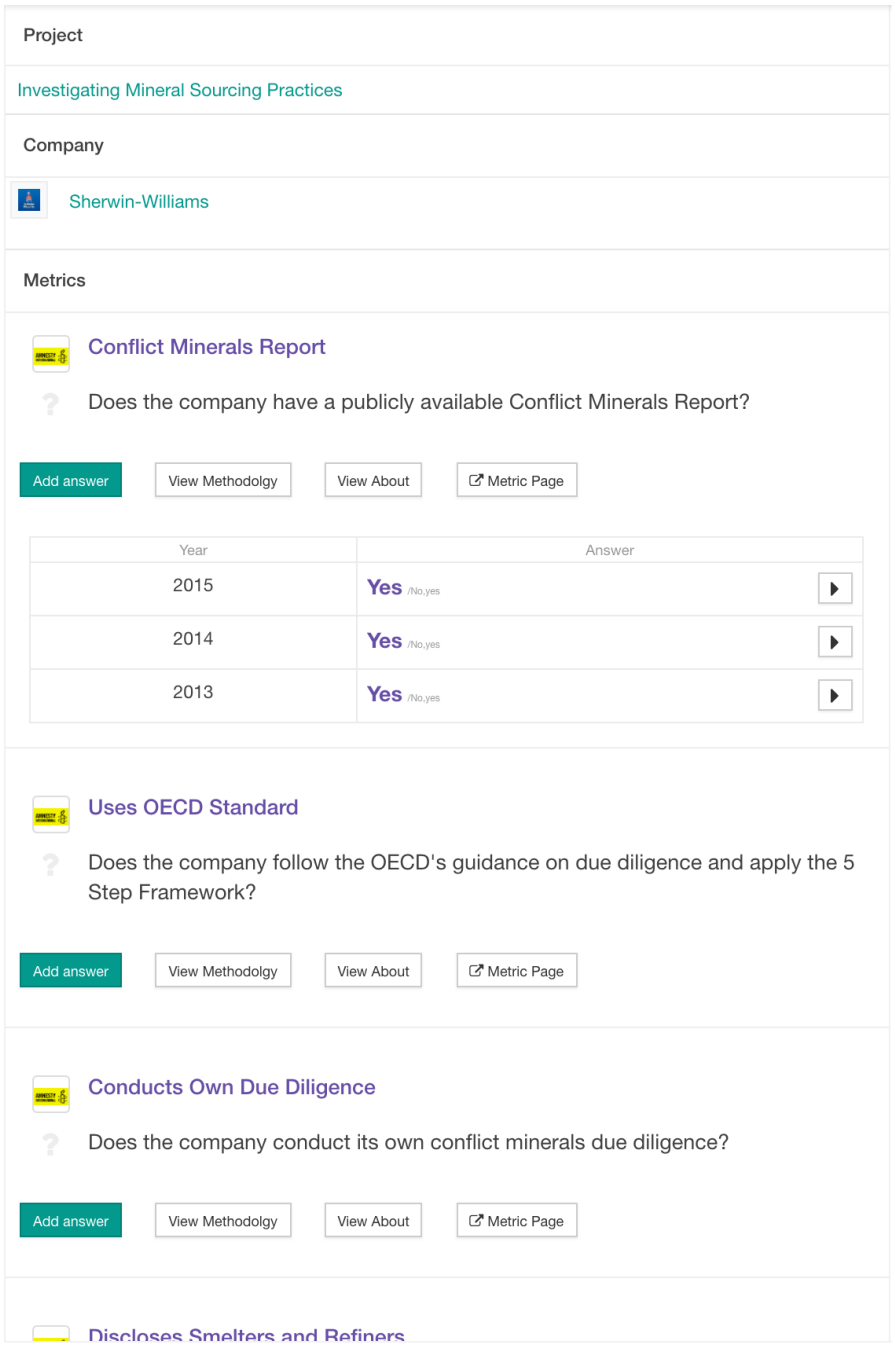


Figure 34 Research Page from Project

The top of the page identifies the Project and Company. What follows is a list of records for Metrics associated with the Project. Some have existing answers, some don't. All support adding new answers, viewing methodology, etc.

Sources

While *Sources* still play a vital role in WikiRate's data ecosystem, Source pages are less emphasised than they once were as a starting place for adding data. This is because of the emphasis on using machine intelligence to do much of the heavy loading of finding/organising sources, identifying their report type, and connecting them to specific companies. CERTH has spearheaded this effort, and in doing so has made Projects much more attractive to researchers who can avoid much of the time and vicissitudes of data gathering.

Source Pages have not changed dramatically in year 2, but do have a look and feel more consistent with the rest of the site:

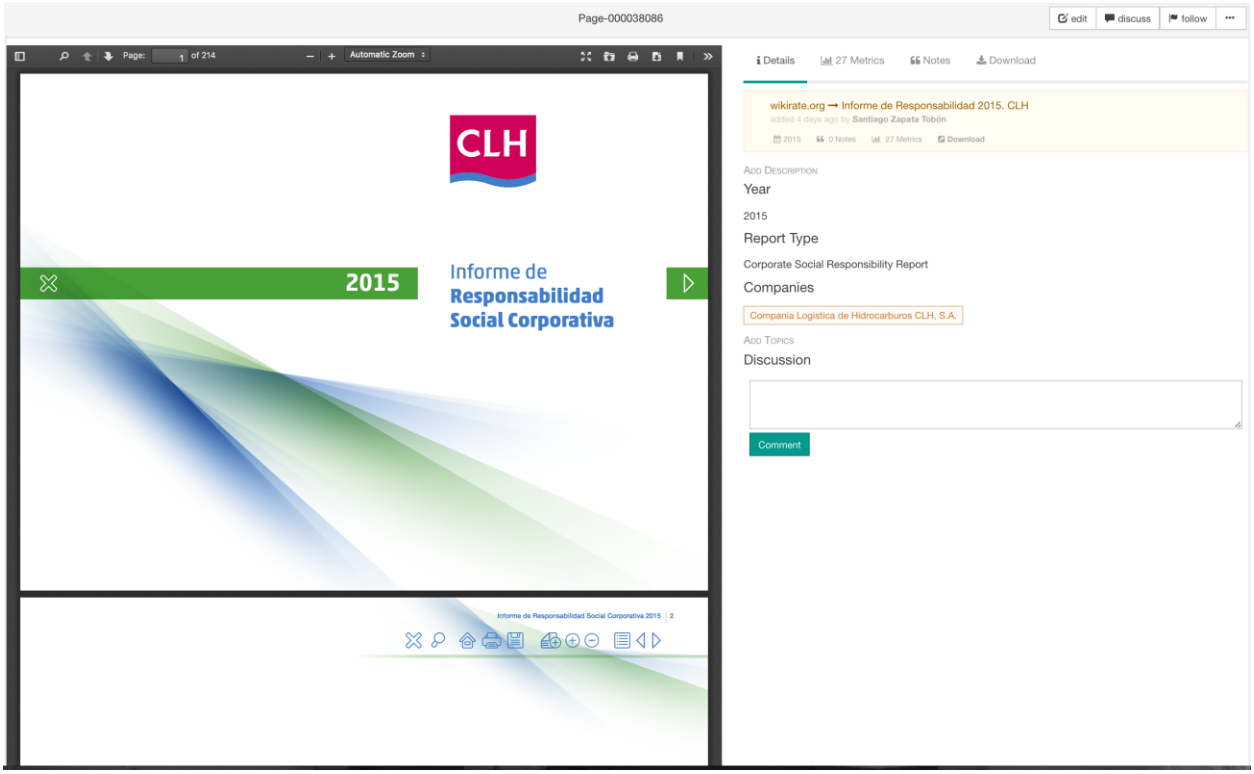


Figure 35 Source Page

The "27" beside the metrics tab indicates that 27 metrics answers have been populated so far from this single source, which was, in fact *not* part of an automated import. Sources can be added manually from the *Research Page* interface or from *Browse Sources*.

Imports/ Integration of WP5

The import interfaces are currently reserved for admins, though it is expected that they will someday be open to broader use. As such, very little emphasis has been put on making the import interface *discoverable*, but considerable time has been spent making it *usable*, particularly to the WP5 partners who have prepared ample amounts of data for WikiRate.org.

In short, the process for both metric value and source imports is:

1. Upload a CSV import file
2. Review the interpreted data (especially company mappings) to ensure data quality
3. Finalise the import.

In year 2, imports were a significant source of performance problems. At the time, imports were conducted within a database transaction, meaning that during imports, the database was "locked", preventing any other data alterations to be transacted until the import was complete. Given that imports could involve many thousands of rows, and each row could involve quite a few cards, imports could take a long time, and the impact was significant.

A Decko upgrade made it possible to conduct imports outside of transactions (See *Task 6.2: Improvements to Decko platform*) shortly after the last review, and imports have been successfully conducted alongside active research projects ever since. Imports are still slower than ideal, and further optimizations are planned. Happily, the inconvenience is now limited to the importer, but we would prefer to eliminate the inconvenience altogether, and this must be done before imports can be employed more broadly.

In interface terms, the review phase is naturally the most involved, as data are presented in a form designed to facilitate potential integrity problems. The most challenging component is matching companies of datasets that have not yet been explicitly mapped to WikiRate unique identifiers. Each row of the Import files is color-coded to convey the quality of the company match:

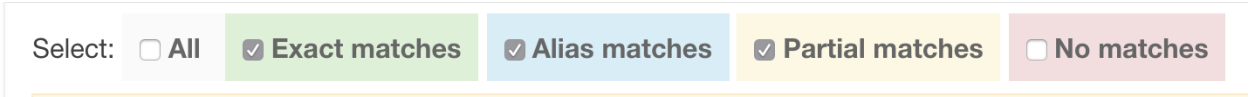
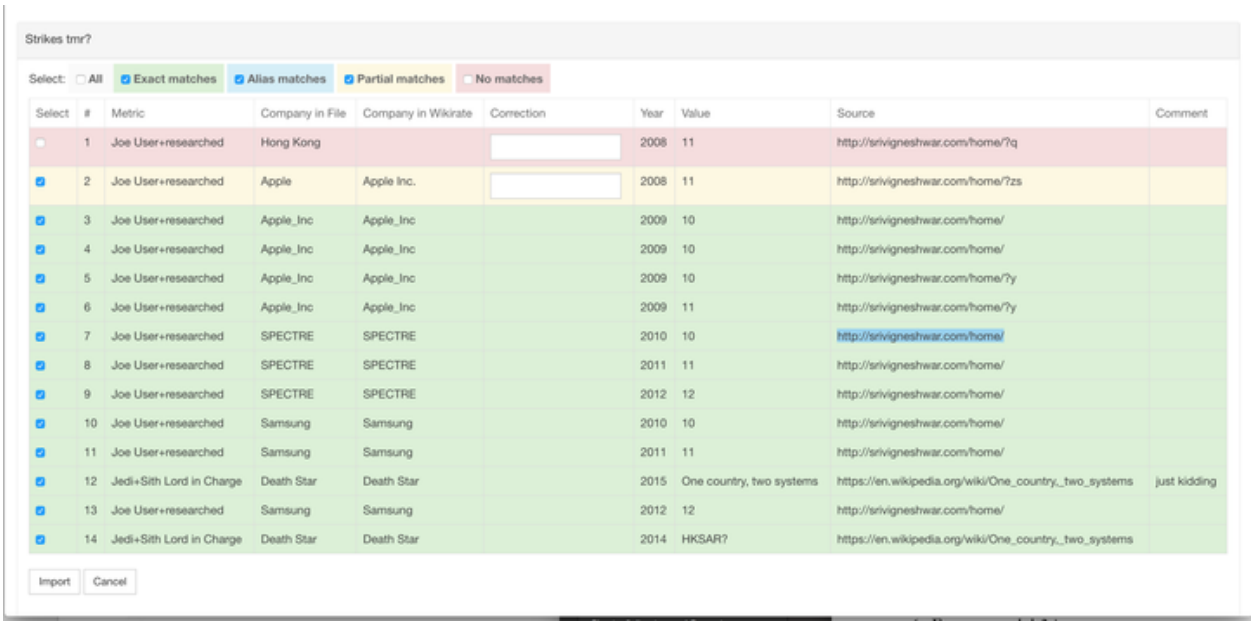


Figure 36 Import Legend

Exact matches and aliases matches tend to be accepted immediately, and "no matches" tend to be left out of the import in question. Partial matches require the most attention.

The following screenshot is from our automated import test, as such clues as "Joe User" and "Death Star" may have implied. It shows one "no match", one "partial match", and twelve "exact matches".



Select	#	Metric	Company in File	Company in Wikirate	Correction	Year	Value	Source	Comment
	1	Joe User+researched	Hong Kong			2008	11	http://srivigneshwar.com/home/?q	
	2	Joe User+researched	Apple	Apple Inc.		2008	11	http://srivigneshwar.com/home/?zs	
	3	Joe User+researched	Apple_Inc	Apple_Inc		2009	10	http://srivigneshwar.com/home/	
	4	Joe User+researched	Apple_Inc	Apple_Inc		2009	10	http://srivigneshwar.com/home/	
	5	Joe User+researched	Apple_Inc	Apple_Inc		2009	10	http://srivigneshwar.com/home/?y	
	6	Joe User+researched	Apple_Inc	Apple_Inc		2009	11	http://srivigneshwar.com/home/?y	
	7	Joe User+researched	SPECTRE	SPECTRE		2010	10	http://srivigneshwar.com/home/	
	8	Joe User+researched	SPECTRE	SPECTRE		2011	11	http://srivigneshwar.com/home/	
	9	Joe User+researched	SPECTRE	SPECTRE		2012	12	http://srivigneshwar.com/home/	
	10	Joe User+researched	Samsung	Samsung		2010	10	http://srivigneshwar.com/home/	
	11	Joe User+researched	Samsung	Samsung		2011	11	http://srivigneshwar.com/home/	
	12	Jedi+Sith Lord in Charge	Death Star	Death Star		2015	One country, two systems	https://en.wikipedia.org/wiki/One_country,_two_systems	just kidding
	13	Joe User+researched	Samsung	Samsung		2012	12	http://srivigneshwar.com/home/	
	14	Jedi+Sith Lord in Charge	Death Star	Death Star		2014	HKSAR?	https://en.wikipedia.org/wiki/One_country,_two_systems	

Figure 37 Metric Value Import

The input box in each row allows for a company name correction mid-import.

Topics

As discussed, Topics join Companies and Metrics as one of WikiRate's three most prominent content types. Their current role is to provide a flexible means of grouping related content and to create a low-barrier-to-entry route into engaging with Metrics. They were also originally conceived as a way of organising qualitative content in a format friendly to side-by-side comparison (eg, two companies might each have a Review for the same Topic).

For now, given the strong strategic emphasis on advancing metric research, their greatest value is in (a) conveying WikiRate's scope to new users, and (b) providing a navigational tool for existing users to find related metrics.

The following example topic illustrates how Topics may be used in support of organising related metrics.

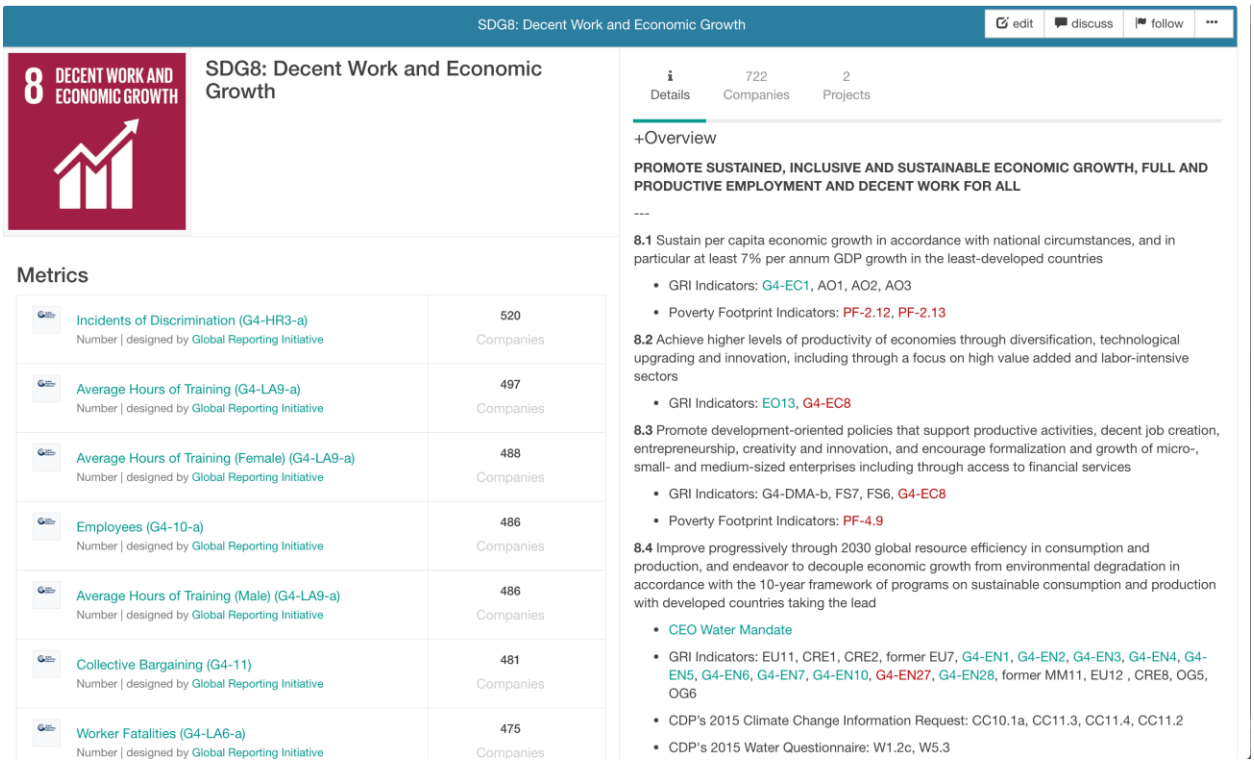


Figure 38 Topic Page

The layout is new and consistent with the other primary content types, and the expected connections to metrics and companies are easy to navigate.

However, despite their elevated status in introducing the site's concepts, the de-emphasis of qualitative content has actually led to Topics pages becoming considerably simpler in Year 3, and their current role is, put briefly, conceptual and navigational glue.

Profiles and Badges

The user profile page had a wholesale redesign in year 2 in order to support our badge system as conceived in WP3. A key goal for us was to integrate the badges richly with usable dashboard interface (and not to separate "game" area).

It is readily visible from the profile chosen that the badge system is based on gold, silver, and bronze level badges.

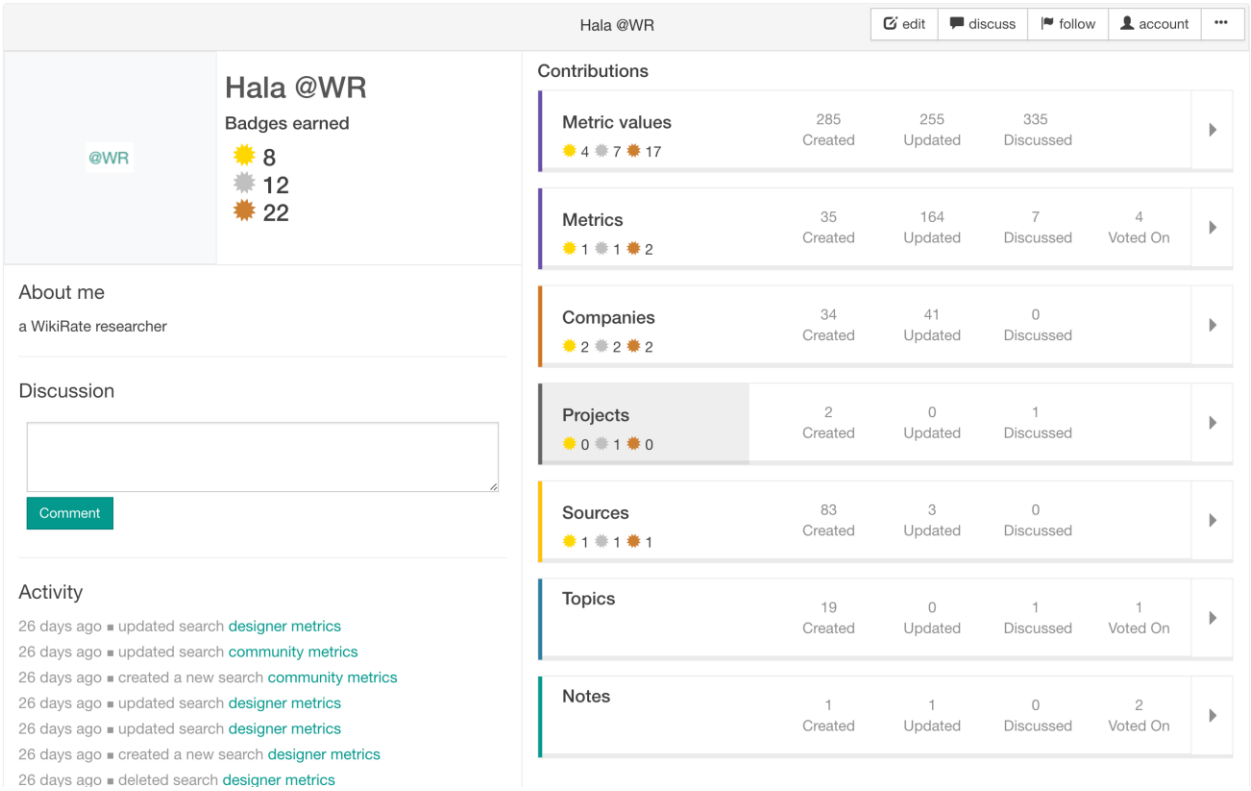


Figure 39 User Profile page

These are broken down into contribution sections (metric values, metrics, etc), each of which offers summary numbers for basic actions.

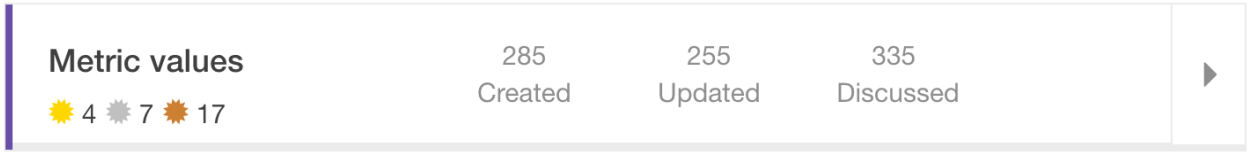


Figure 40 Profile page, contribution section

Each section has summary counts that double as tabs, and clicking on a tab opens more specific details. The arrow on the right aims to make this more discoverable and opens up the "Create" tab, which itself has several subtabs. University course evaluation user scenarios drove the subtab design.

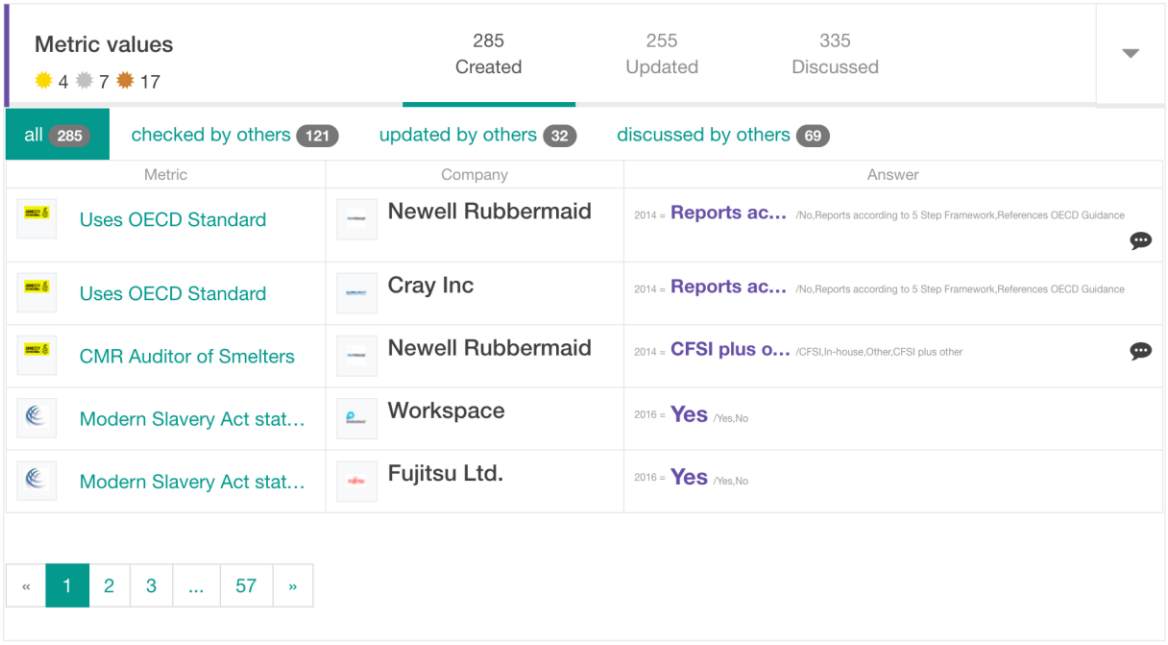


Figure 41 Profile page, contribution section, "Create" tab

Each section lists items in a form appropriate to its type. Here is a view of the Projects section:

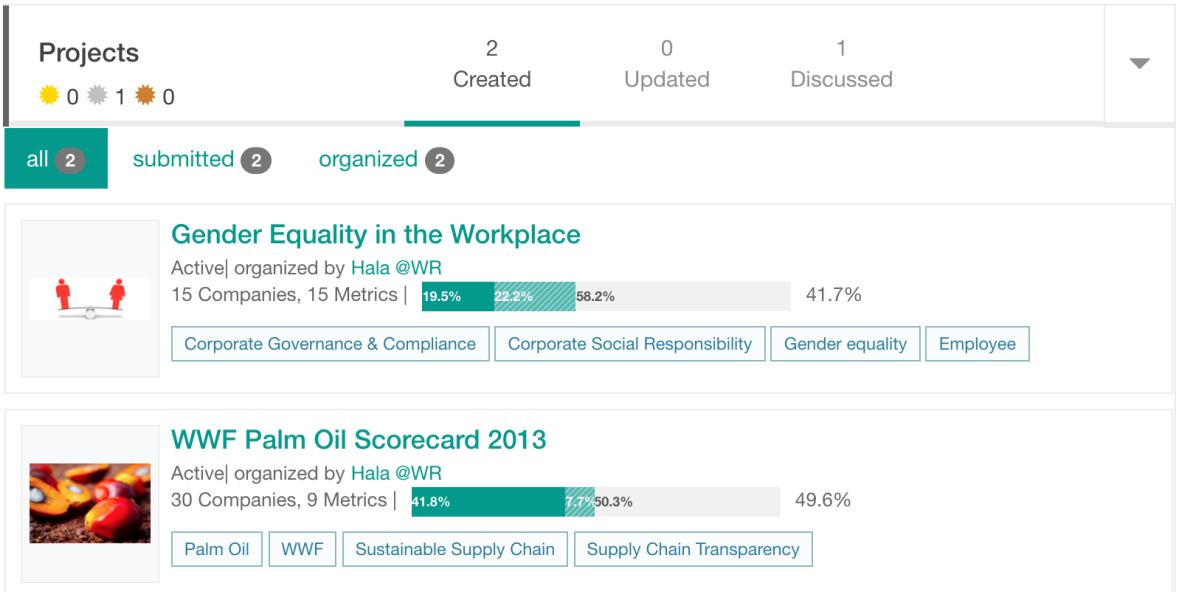
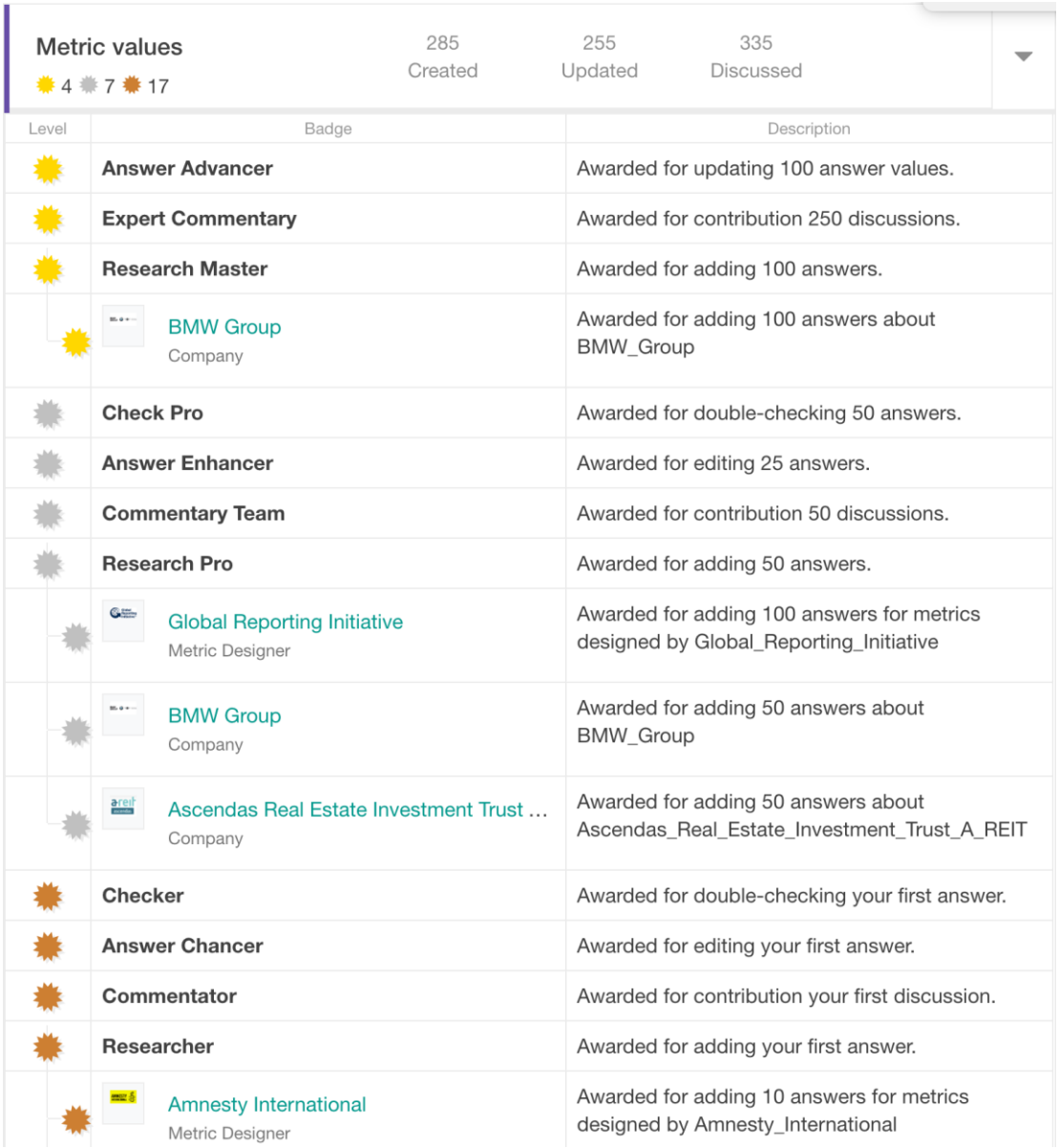


Figure 42 Profile page, Projects contribution section

Where applicable, the leftmost tab opens a badge report, indicating all of the badge the user has received for the contribution type.



Metric values
4 7 17

285 Created
255 Updated
335 Discussed

Level	Badge	Description
	Answer Advancer	Awarded for updating 100 answer values.
	Expert Commentary	Awarded for contribution 250 discussions.
	Research Master	Awarded for adding 100 answers.
	BMW Group Company	Awarded for adding 100 answers about BMW_Group
	Check Pro	Awarded for double-checking 50 answers.
	Answer Enhancer	Awarded for editing 25 answers.
	Commentary Team	Awarded for contribution 50 discussions.
	Research Pro	Awarded for adding 50 answers.
	Global Reporting Initiative Metric Designer	Awarded for adding 100 answers for metrics designed by Global_Reporting_Initiative
	BMW Group Company	Awarded for adding 50 answers about BMW_Group
	Ascendas Real Estate Investment Trust ... Company	Awarded for adding 50 answers about Ascendas_Real_Estate_Investment_Trust_A_REIT
	Checker	Awarded for double-checking your first answer.
	Answer Chancer	Awarded for editing your first answer.
	Commentator	Awarded for contribution your first discussion.
	Researcher	Awarded for adding your first answer.
	Amnesty International Metric Designer	Awarded for adding 10 answers for metrics designed by Amnesty_International

Figure 43 Profile page, metric value badges

Badges are sorted by level (gold, silver, bronze) and *affinity* badges (for contributions to specific Companies, Metric Designers, etc) are represented with a tree diagram, indicating that they are affinity variations of the parent badge.

When a new badge is earned, the user receives an inline notification. It was decided not to present these notifications by modal windows, because it was hoped that they would cause as little disruption as

possible to the research process. Notifications appear in green boxes above the content in question and can be closed at any time:

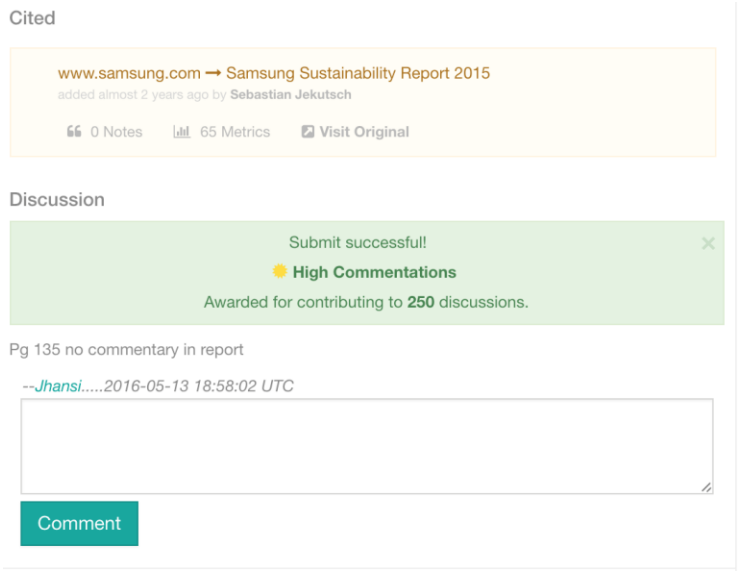


Figure 44 Badge award notification

Each Badge has its own card, and clicking on a badge lists all the users with that badge.

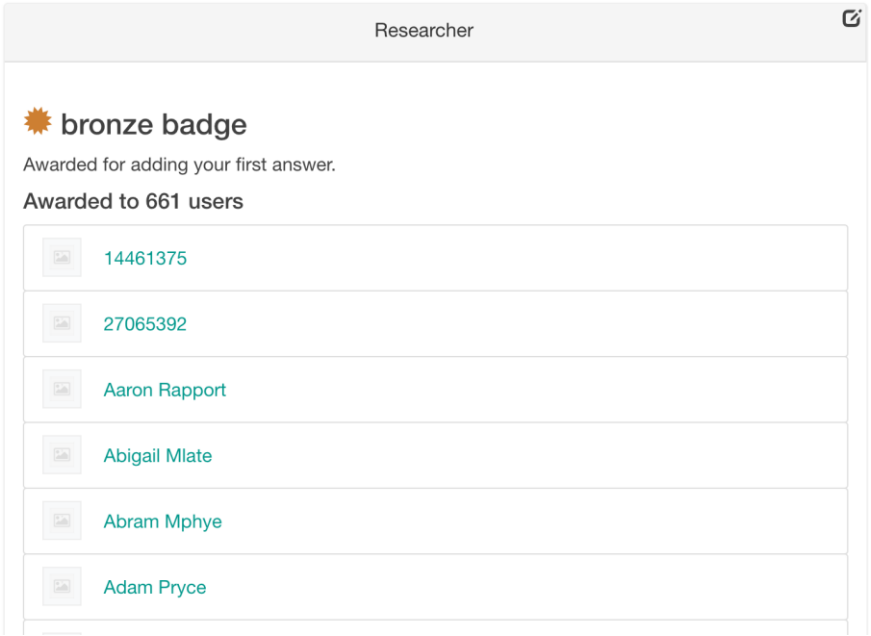


Figure 45 Badge page

Research Groups

Research Groups were conceived as a solution to several seemingly unrelated problems:

- Individuals were often wary of having the role of Metric Designer, because the Metric Designer's name was presented as part of the metric's full name. Presenting an individual's name (and often image) in identifying a metric brought unwanted attention and was seen as diminishing credibility.
- University teachers using WikiRate as part of a course wanted convenient views for reviewing student contributions.
- There was no easy way to group related projects being carried out by the same teams.
- In general, WikiRate.org offered few tools for formalizing collaborations.

Research Groups solved these problems by creating an easy way to form groups of researchers. As a named group, they can design metrics and organize projects, and the Research Group page can serve as a hub of group activity.

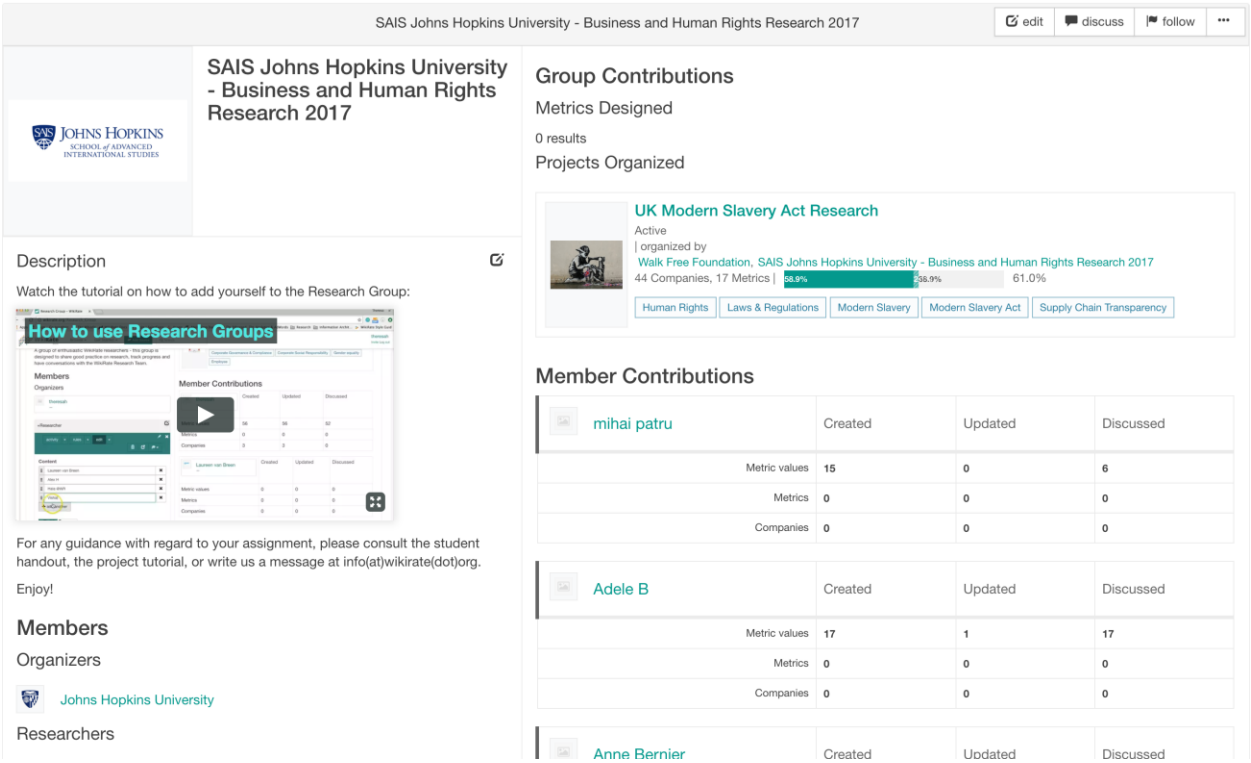


Figure 46 Research Group

Research Groups are still quite new and merit further interface improvements before being featured more prominently, eg. via the Browse menu. That said, even in their nascent, rapidly implemented form they are already helping resolve each of the challenges they were designed to address.

Calculated Metrics

WikiRate's metrics framework is discussed at length in Deliverable 4.4.2 (Rating Concept Evaluation). We will not reprise its rationale here, but will look a bit more closely at the current state of the implementation and (for context) exploitation.

Calculated Metrics are currently considered to be in *beta*. They have been live on the production site for about a year now and are fully functional and available for use, but they are not being promoted or prominently linked to, nor is there abundant help text. The consensus in the consortium is that the functionality should first be proven by (and improved upon based on) strong initial use cases.

We are currently moving forward with 2-3 research partners at various stages of the process of developing a set of calculated metrics, either by designing new metrics or implementing an existing system on WikiRate.org. Work on the metric implementation is being performed in close collaboration with the WikiRate team, who are closely tracking interface needs and opportunities. Based on these pilots, we will update the interface to support discoverability and critical workflows and will begin engaging a wider audience.

Our metric framework conceives of four different types of metric, as illustrated below.

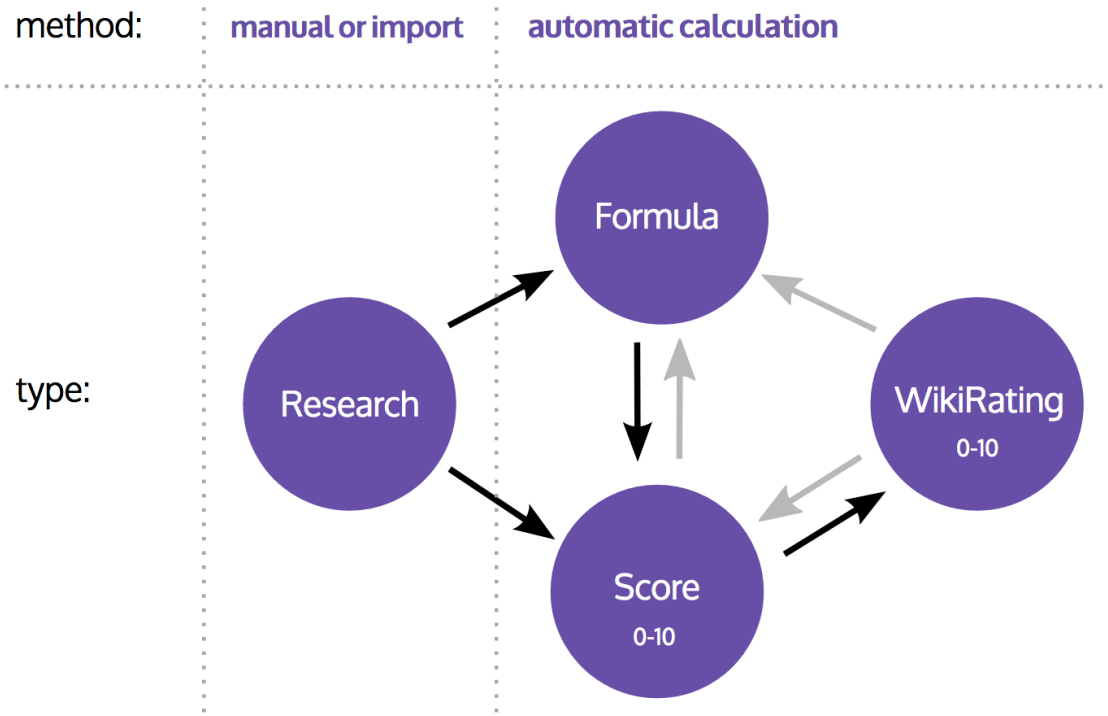


Figure 47 Metric Types Diagram

The arrows were intended to convey data flows, but they simultaneously convey interface flows.

When creating a new metric, one is offered the following interface:

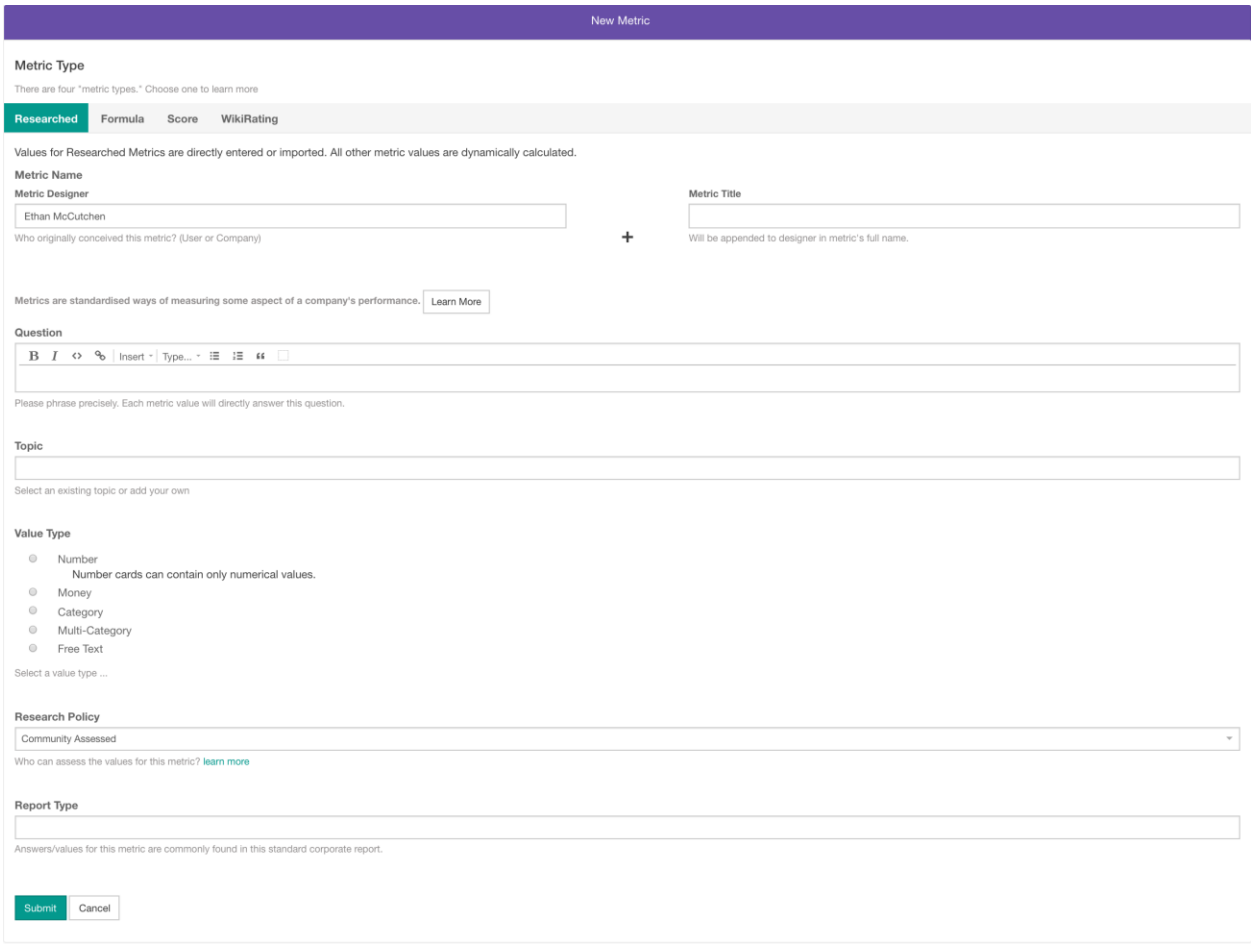


Figure 48 Researched, new metric form

By default, one is presented with the interface for creating Research metrics, including many familiar research-specific fields (research policy, report type, etc). But if, instead of the current green "Researched" tab, another is chosen; the user is presented with an interface for creating a calculated metric. We will here consider each Calculated Metric type one by one.

Formulas

Formula metrics support users in combining metric values mathematically with an open (non GUI) formula interface.

Below is the form for creating a new Formula card:

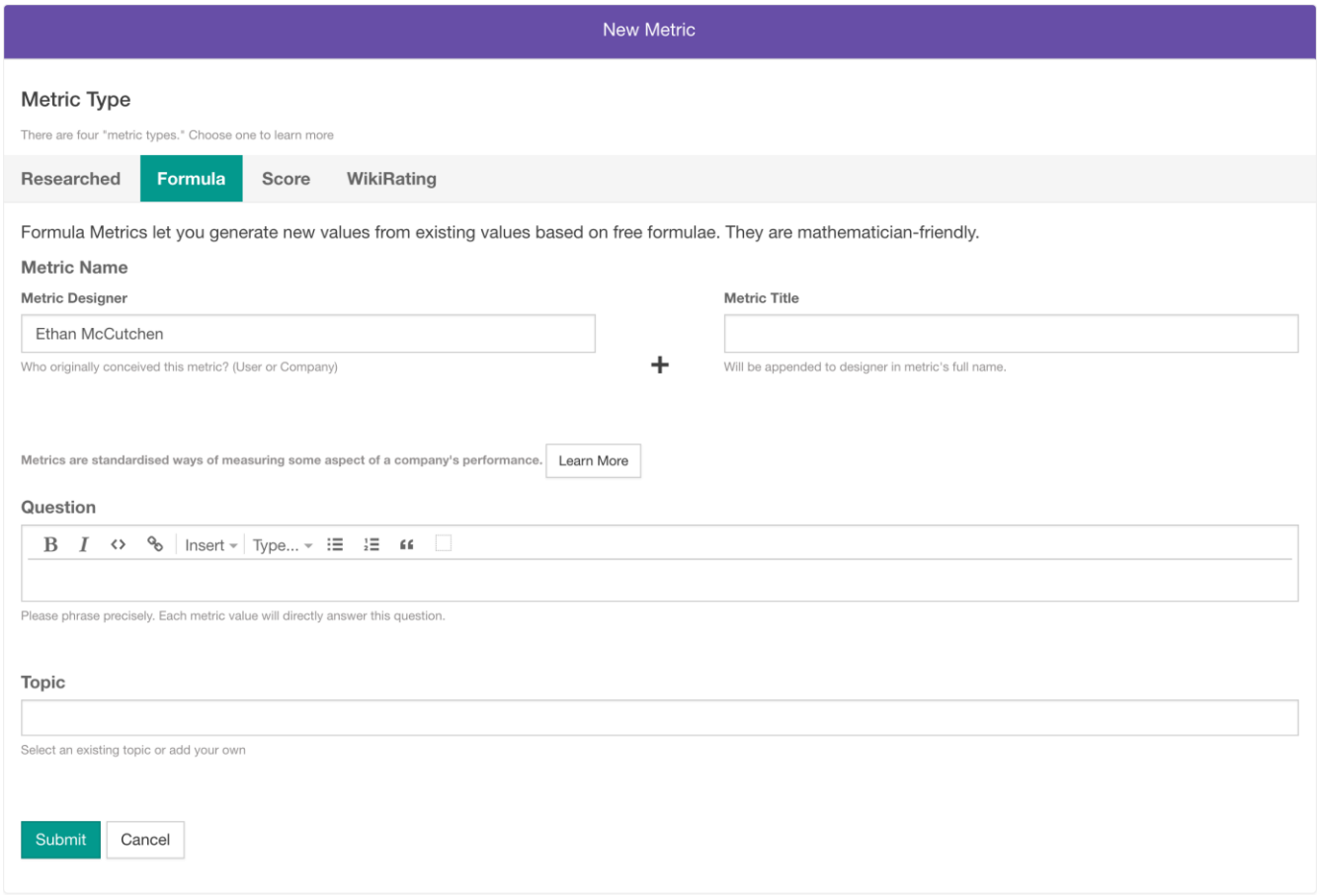


Figure 49 Formula, new metric form

Note that there are no fields beyond those shared with all metrics in the initial form.

Having created the metric, the user can then add a Formula via the following Spartan interface:

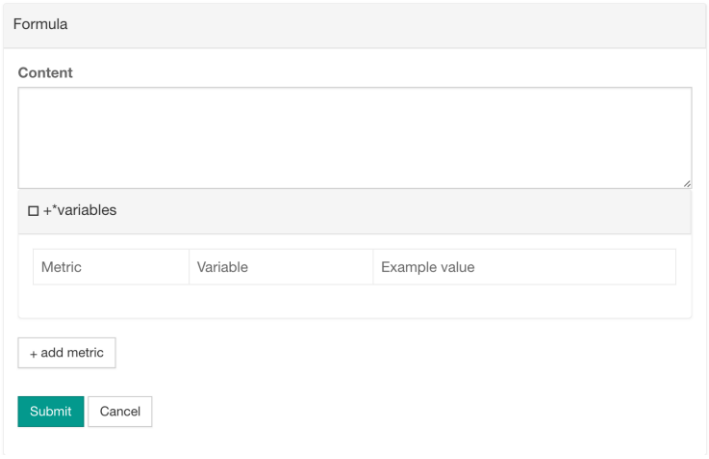


Figure 50 Formula metric, formula editor

By clicking "add metric" on the formula editor, you bring up the following interface, which lets you add a metric or metrics to use as variables in the formula.

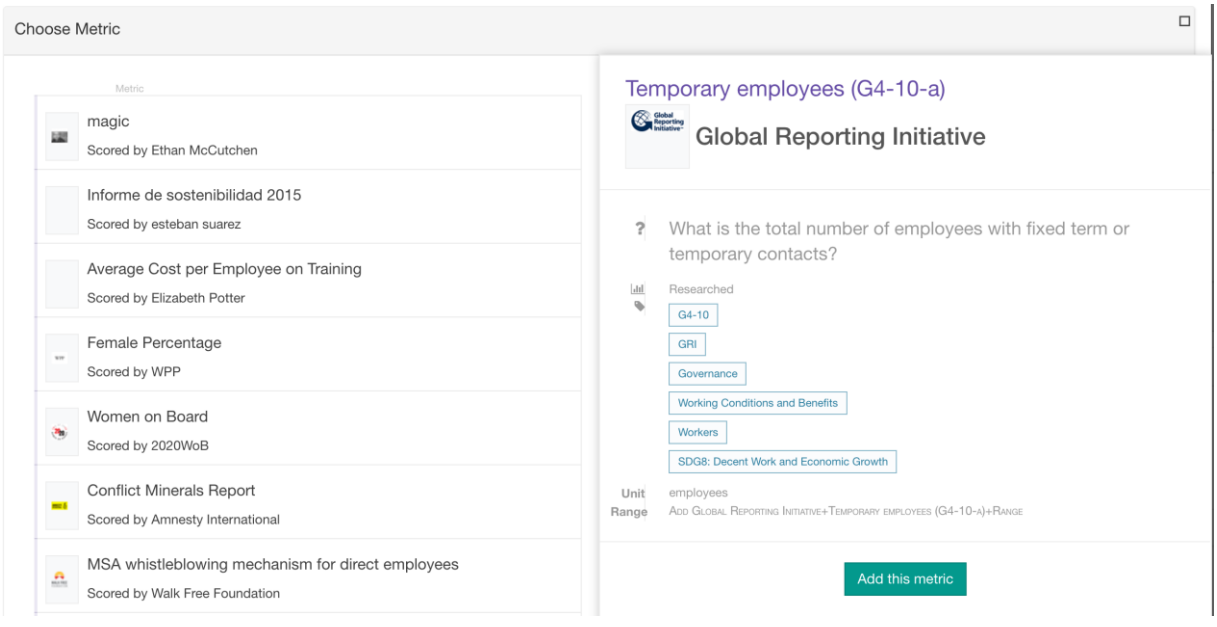


Figure 51 Formula editor, add metric

Having selected metrics, they can be combined in the editor, as in the following simple example. Metrics are contained in double curly brackets and can be referred to by their full names or by the variable name provided (eg. M0, M1,…).

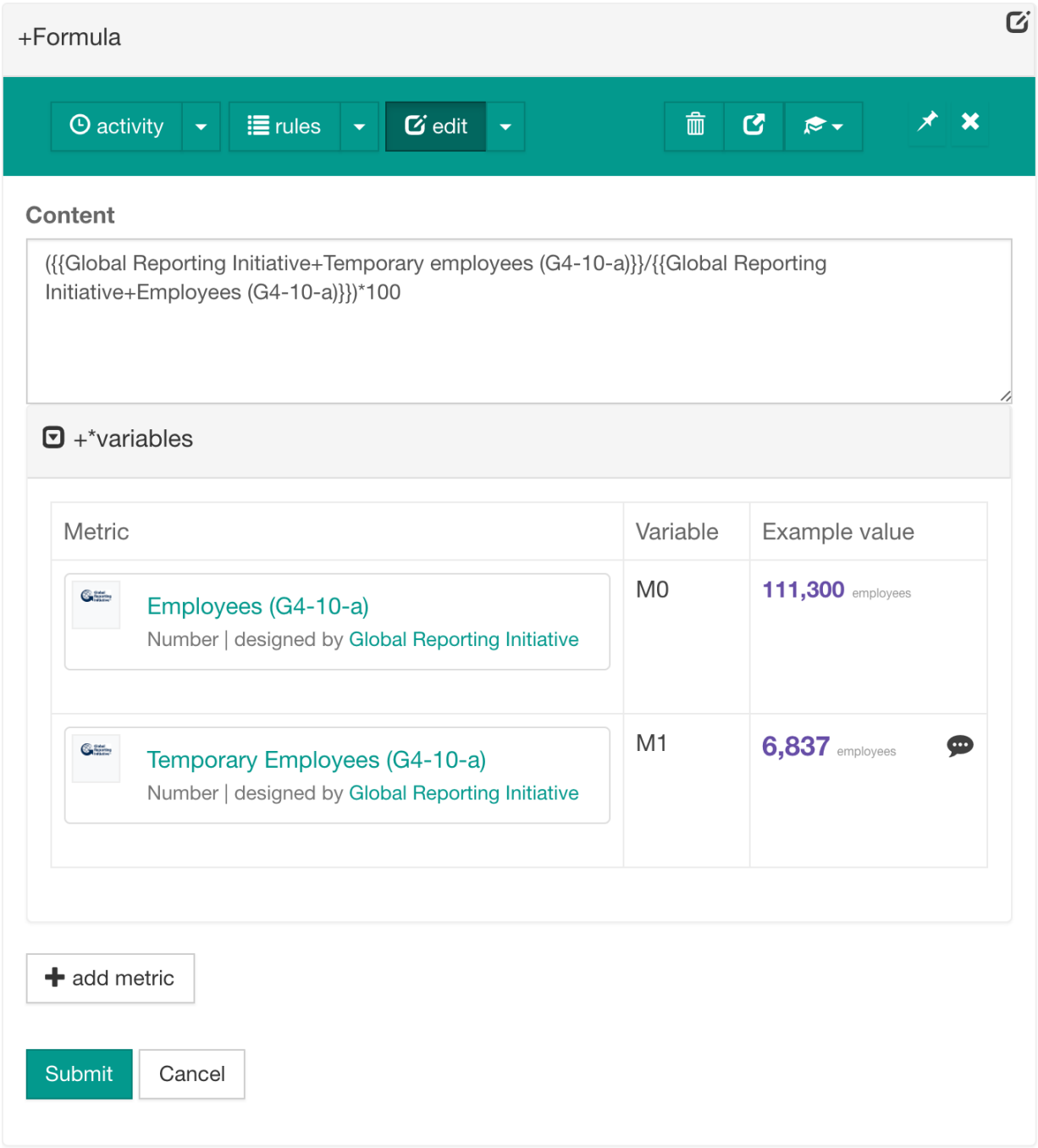
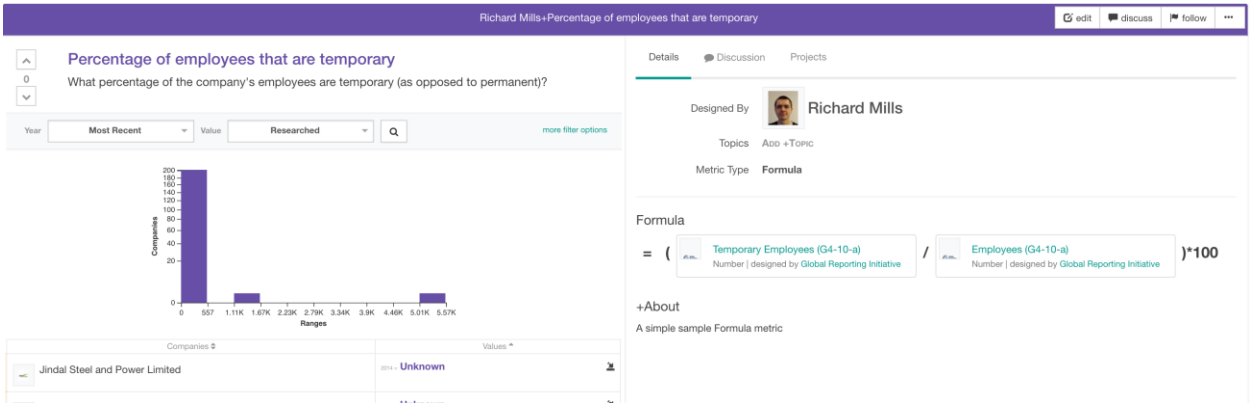


Figure 52 Formala editor with metrics

When the formula is saved, WikiRate presents a view of the formula with the metric variables replaced by small metric listings, and it immediately calculates values for every company for whom all variable metrics have been researched:



Those new answers will now appear on any applicable company page. Expanding the answer shows the variable values with which the final answer is calculated.

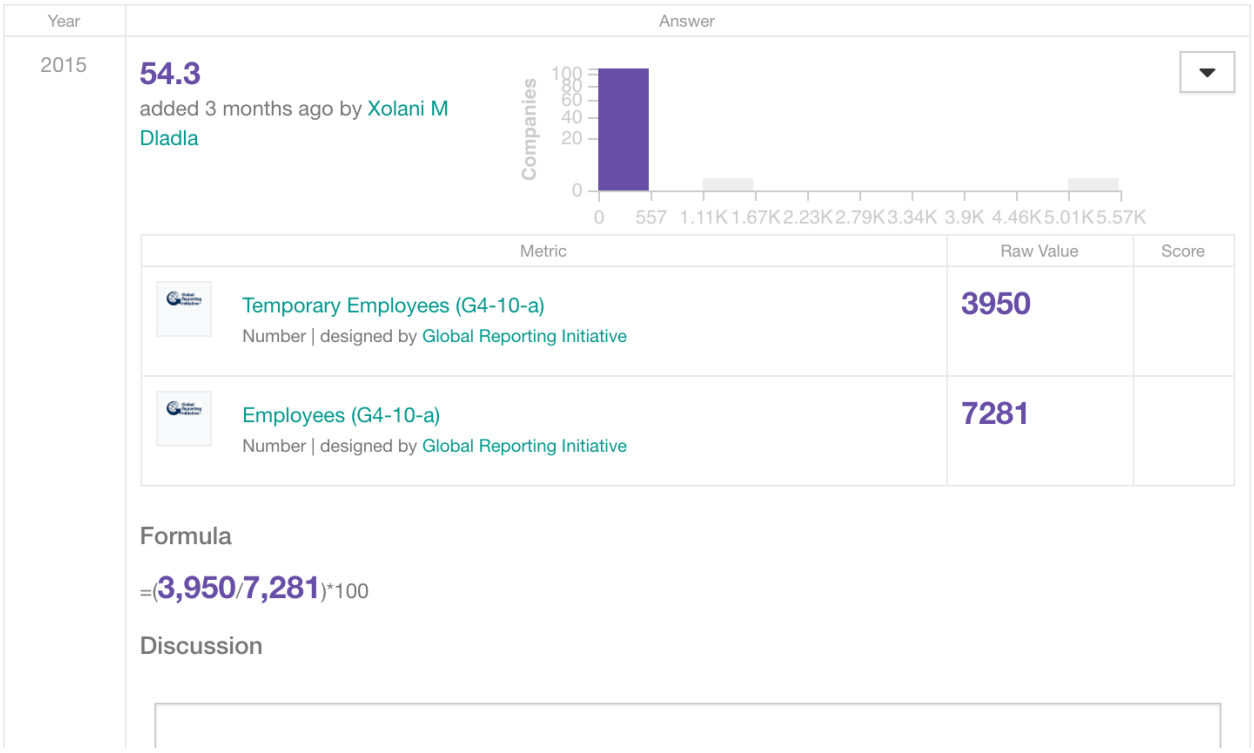


Figure 53 Formula metric, expanded answer

Scores

The new metric interface for Scores explains their purpose concisely "A Score Metric normalizes another metric's values to a 0-10 scale":

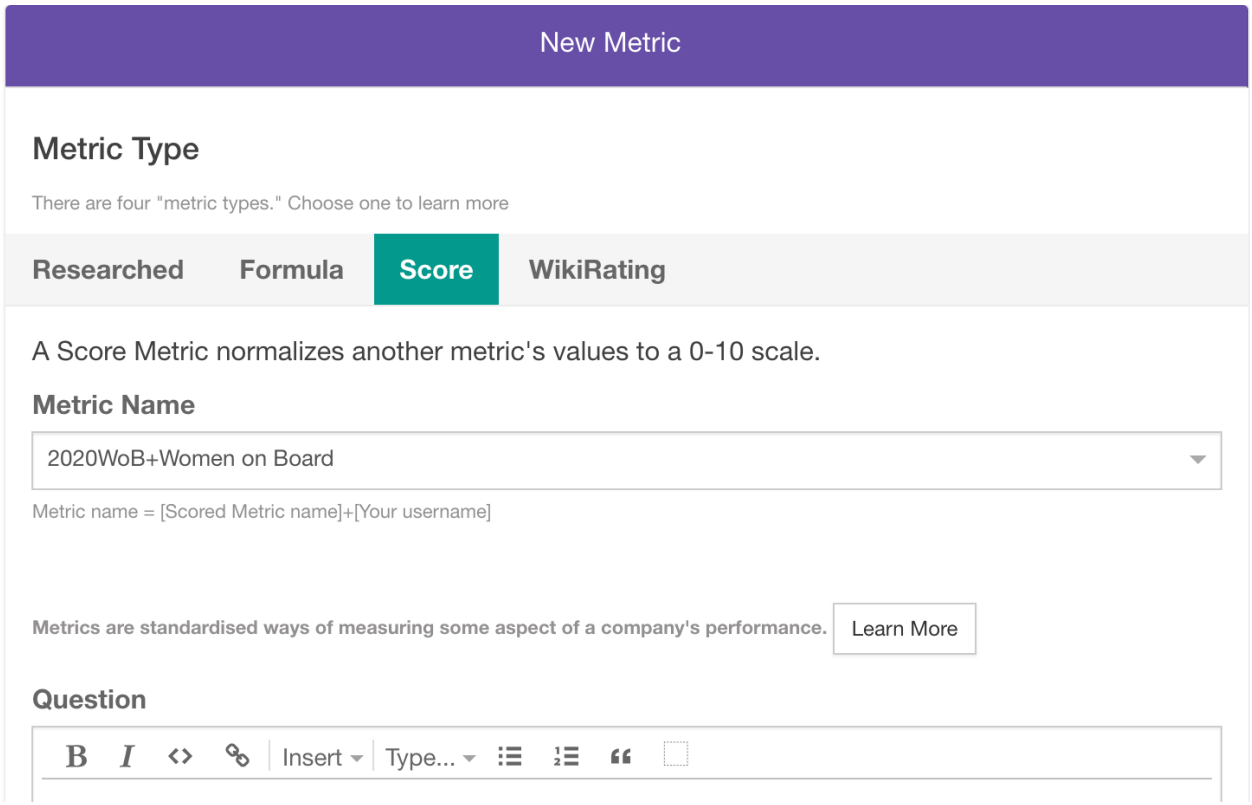


Figure 54 Score, add metric form

The naming interface is different from the Formula interface, because scores break the standard naming pattern: their names combine the name of the scored metric and the scorer's name. Thus the interface provides a dropdown to choose the metric to be scored and automatically adds the scorer's name.

Generally speaking, Scores offer lots of mathematical value – specifically in preparing Researched and Formula metric values for use in WikiRatings – but very little new information, and for that reason they are generally displayed less prominently throughout the site than the other three types. At present, for example, they are not listed in answer lists on company and metric pages. It has been proposed that they should appear in such listings as optional expansions of the metrics they score.

The following sample score shows how a categorical metric can be scored by the direct numerical translation of answer categories into 0-10 numeric values:

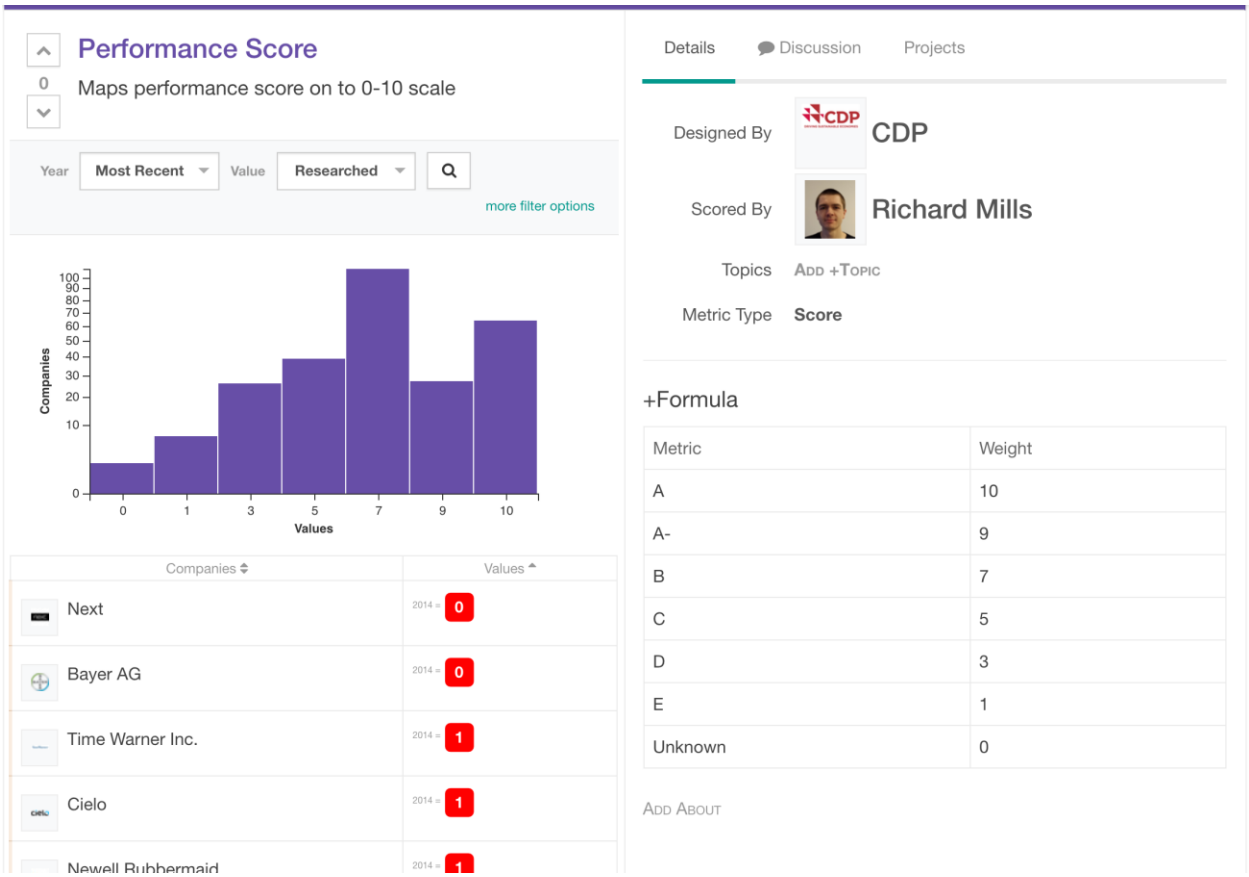


Figure 55 Score, categorical metric values

By contrast, numerical metric values are scored with ranges. Note that the metric credits the original designer and the scorer separately. Also important is the fact that scores generally imply a value judgment, in which 10 is most preferred and 0 is least. This is visualized in use of traffic light colors (red, yellow, green) in the values. The above view is sorting lowest to highest. A high score looks like this:

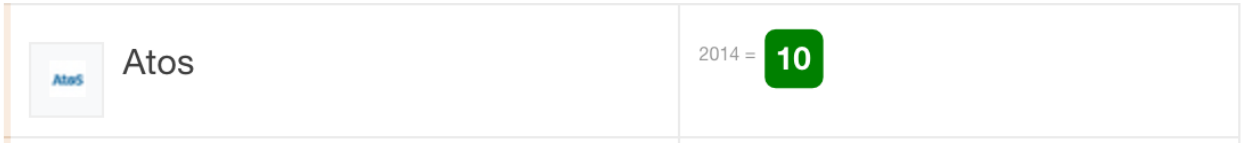


Figure 56 Score answer listing, high score

There is considerable room for maturation in Scores. In particular, we expect them to grow more sophisticated in their handling of unknown and unresearched values.

WikiRatings

WikiRatings are the top of the Calculated Metrics food chain. They are designed to be very accessible – easy to understand, navigate, and even create. However, for them to be meaningful, the community must first complete a lot of legwork on Research, Formula, and Score metrics.

The new metric interface for WikiRatings is identical to the Formula interface, except for the help text, which reads "WikiRating Metrics combine Scores with simple weighted averages. No heavy math involved." Scores, in turn, derive meaning from Research and Formula Metrics. The more Research metrics are populated, the closer WikiRate is to being able to start generating meaningful WikiRatings.

The following example WikiRating was WikiRate.org's first:

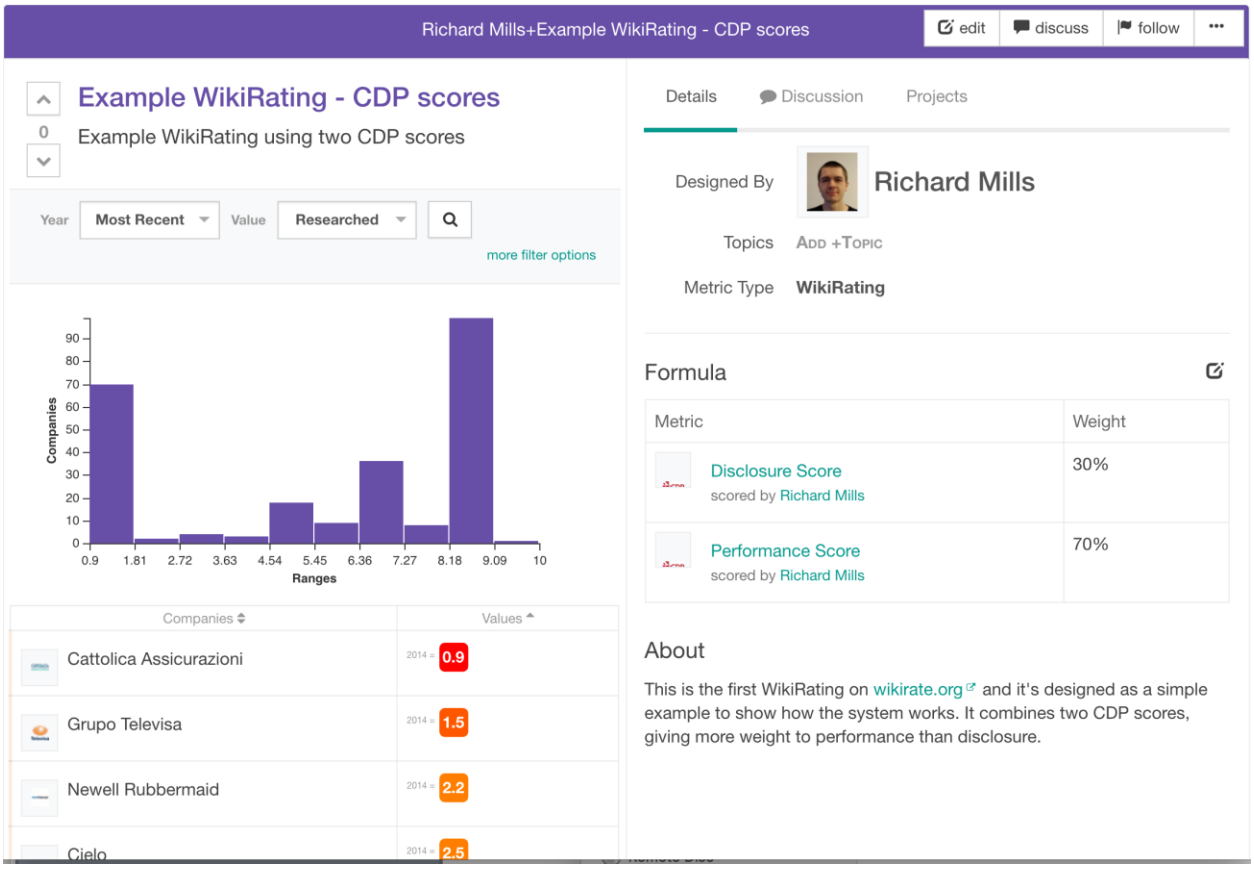


Figure 57 WikiRating page

This WikiRating looks much like other Calculated metrics, including the traffic-light-colored values of Scores and the listings of variable metrics of Formulae. But a WikiRating's formula is much more proscribed. The edit interface is also far more approachable:

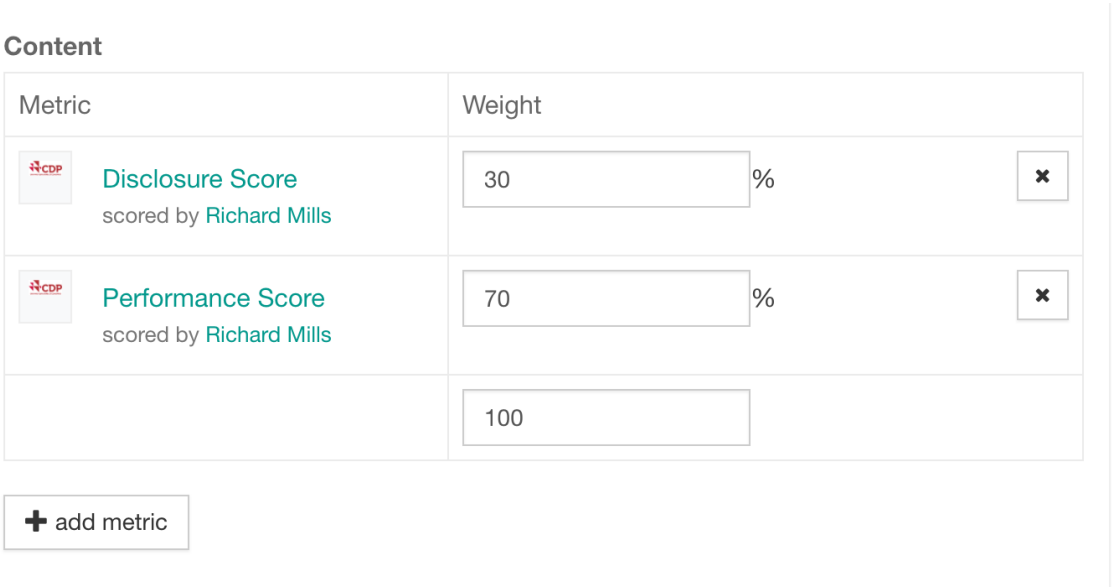


Figure 58 WikiRating page, edit formula

The weights must add up to 100 for the form to validate. The "add metric" interface is the same as for Formulae, except that naturally only Scores are available. Expanded WikiRating answers are similarly readable:

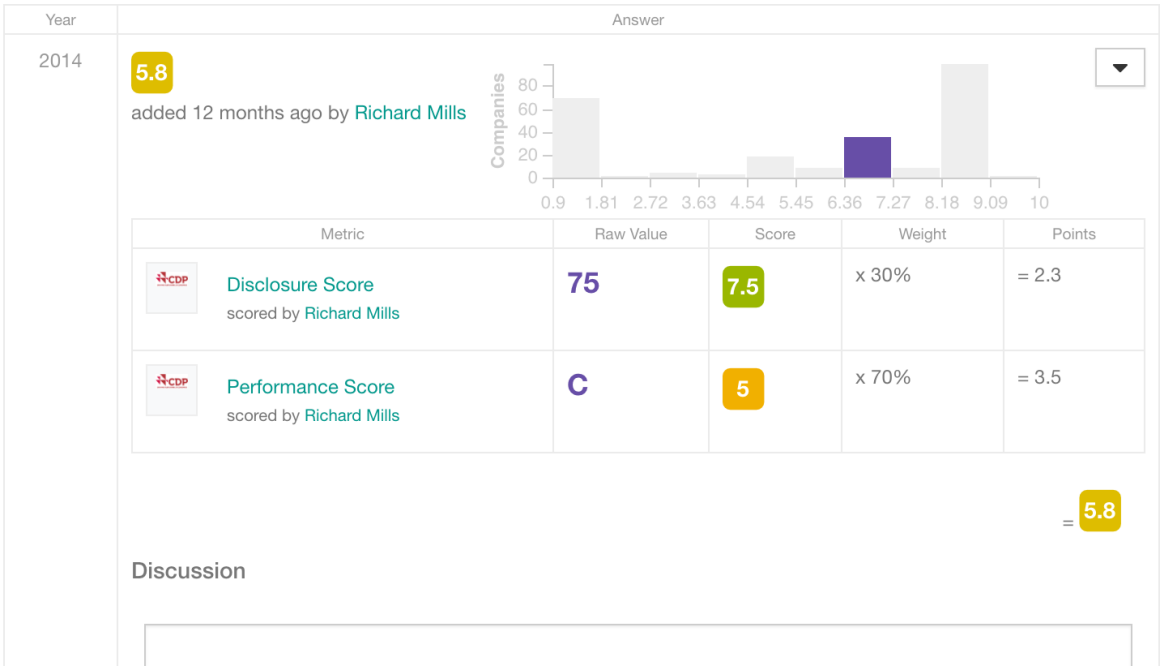


Figure 59 WikiRating answer, expanded

Task 6.1: Implementation of Features of WP2 and integration of WP5

The previous 50-odd pages present the *results* of Task 6.1. The *source code* produced is referred to in the Source Code Repositories section below. The *evaluation* of the project's technical development is designated to D7.7.6. All that's left for this first of four sections reviewing WP6 tasks is a bit of the view from the middle. First we'll look a bit at how WikiRate-specific code translates Decko concepts into the year two features discussed above. Then we'll introduce a couple of WikiRate-specific optimizations.

Card/code representation of WikiRate Concepts

In year 3, we remained faithful to the Decko credo that *everything is a card*, even as we pushed interface patterns well beyond Decko defaults.

Cardtypes

WikiRate can now boast over 1.2 million cards of over 50 different Cardtypes. Some of the types of particular interest in Year 3 (with counts as of time of writing) include:

- **Badge** (29), the core of WikiRate gamification. We will continue to add new badges in an effort to make WikiRate research more engaging.
- **Company** (9204). We continue to pace ourselves with companies and import only those for which we have multiple sources of data, in order to avoid diluting the data set with poorly covered, little known companies.
- **Metric** (1000), the most focal cardtype in year 3, and the most complex in terms of code. Metric types (Research, Formula, Score, and WikiRatings) are *not* implemented as Cardtypes, but as values of a metric field. New metric type *Set Patterns* were added to make it easy to add code specific to a given metric type.
- **Metric Value** (222K) will be renamed to "Answer" in keeping with shared team terminology. Values are just one part of Answers; other parts include sources, discussions, flags, etc. When you consider that an Answer can comprise four or five cards its not hard to see how WikiRate has passed a million cards.
- **Metric Value Import File** (555). These cards, which inherit code from simple Decko file cards, are used to manage the import of company/metric/year answers.
- **Project** (73) cards have become a more central (and popular) feature for WikiRate than expected at the beginning of the year.
- **Report Type** (8) cards are not elaborated upon above, but they are a key bit of glue in supporting source suggestions. This drastic simplification of our initial source suggestion mechanism is effective because of the prevalence and topical regularity of a few different kinds of corporate reports. WikiRate did not previously take advantage of this pattern.

- **Research Group** (26). We expect these cards could become considerably more powerful with a thoughtful upgrade.
- **Source Import File** (72). These files drove the source imports that enabled the new focus on Research pages with source suggestions

Mods

Decko organises code into *mods,* which are easily shared card code libraries. WikiRate code is currently organised into fifteen of them. Five of them are sufficiently generalised for broader use that they are maintained outside the WikiRate repository.

- *airbrake* – a decko integration with the error tracking service
- *logger* – code for customizing performance, debugging, and other logging
- *new_relic* – a decko integration with the performance monitoring service
- *pdfjs* – used by WikiRate for rendering PDFs
- *voting* – generalized up-down voting support

The other ten are currently still part of the WikiRate repository and are thus available for broader use, but not all have clear broad application, and those that do could use some attention to generalisation before being promoted as easy plug-ins.

- *badges* – one of the most obvious candidates for generalization, this code governs the new gamification system
- *caching* – in this case "caching" mostly refers to database caching, in which the results of complex processing are stored in cards or other optimization structures. (memory-based caching is also heavily used, but that is mostly governed directly by the decko platform)
- *filter* – code governing sorting and filtering
- *layout* – a newer mod, this provides a shared API for parallel wikirate structures and has helped to standardize the look and feel of the site
- *metric_answer_lookup* – this mod governs the answer table optimization described in the ensuing *WikiRate Optimizations* section
- *metrics* – a significant mod governing the handling of metrics, answers, etc.
- *profiles* – the complexity of the contribution sections warranted the formation of a new mod
- *utility* – a bit of a "miscellaneous" bucket for potentially reusable code but without enough conceptual cohesion to promote as a major mod (yet)
- *wikirate* – the mother mod, in ways, most custom wikirate structures begin here and then get their own mod once they show signs of reusability. Companies, Topics, Projects, Notes, imports, etc are all part of this mod.
- *wikirate_source* – handles sources and source types (files, websites, text, etc)
-

WikiRate Optimizations

WikiRate's performance, both past and future, is evaluated in detail in D7.7.6 (*Final Technical Evaluation*). Here we cover only a few WikiRate-specific changes intended to optimize its performance.

All of them address MySQL database optimizations that are necessitated by the combination of (a) the capacity of Card Queries to make complex SQL queries traversing card relationships easily to generate, and (b) the size of WikiRate's card table, which limits the depth of complexity that is pragmatic for such queries. Simply put, card queries made it a bit too easy to write SQL too complex for big databases. These optimizations obviated those complex queries.

Answers Table

Consider the sorting and filtering options on Metric and Company pages. Industry, Topic, Value, Research Policy, Year, etc – almost all of them are stored in separate cards. Finding a list of answers for queries of this complexity is painfully slow. Our initial solution to this problem was to break the query into multiple queries. Our *better* solution, now in use, was to create a lookup table. (Here we nod to Cambridge University's Bartosz Kiełczewski, who hinted early in the project that lookup tables would eventually be important optimization tools for WikiRate).

The *answers* table is updated whenever an answer is created, updated, or deleted, and makes prohibitively complex queries quite simple. Here is the structure of the *answers* table:

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
answer_id	int(11)	YES	UNI	NULL	
metric_id	int(11)	YES	MUL	NULL	
designer_id	int(11)	YES	MUL	NULL	
company_id	int(11)	YES	MUL	NULL	
record_id	int(11)	YES	MUL	NULL	
policy_id	int(11)	YES	MUL	NULL	
metric_type_id	int(11)	YES	MUL	NULL	
year	int(11)	YES		NULL	
metric_name	varchar(255)	YES		NULL	
company_name	varchar(255)	YES		NULL	
designer_name	varchar(255)	YES		NULL	
title_name	varchar(255)	YES		NULL	
record_name	varchar(255)	YES		NULL	
value	varchar(1024)	YES	MUL	NULL	
numeric_value	decimal(30,5)	YES	MUL	NULL	
updated_at	datetime	YES		NULL	
imported	tinyint(1)	YES		NULL	
latest	tinyint(1)	YES		NULL	
checkers	varchar(255)	YES		NULL	
check_requester	varchar(255)	YES		NULL	
creator_id	int(11)	YES		NULL	

Figure 60 answers table description

Even a WikiRate user who has never seen a MySQL database table description before could readily recognise many of the fields as familiar WikiRate structures (answer, metric, designer, company, etc). Contrast that with this *cards* table description:

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
left_id	int(11)	YES	MUL	NULL	
created_at	datetime	NO	MUL	NULL	
updated_at	datetime	NO	MUL	NULL	
current_revision_id	int(11)	YES		NULL	
name	varchar(255)	NO	MUL	NULL	
creator_id	int(11)	NO		NULL	
updater_id	int(11)	NO		NULL	
right_id	int(11)	YES	MUL	NULL	
key	varchar(255)	NO	UNI	NULL	
trash	tinyint(1)	NO		NULL	
references_expired	int(11)	YES		NULL	
codename	varchar(255)	YES		NULL	
read_rule_class	varchar(255)	YES		NULL	
read_rule_id	int(11)	YES	MUL	NULL	
type_id	int(11)	NO	MUL	NULL	
db_content	mediumtext	YES		NULL	

Figure 61 cards table description

This optimization sets the table for scaling answer filtering, which we expect only to grow in its importance to the site.

Counts Table

A more general lookup table is the counts table. In fact, it would be difficult to make a more generic looking table:

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
left_id	int(11)	YES	MUL	NULL	
right_id	int(11)	YES	MUL	NULL	
value	int(11)	YES		NULL	

Figure 62 counts table description

The purpose of this table is to reduce the need for repeatedly performing mysql counting operations, for example to generate the many stats produced on profile pages (5 topics created, 10 updated, etc). Our previous solution for reducing such expensive operations was to store the counts in cards. While fine for a small site, the card count optimization scaled poorly, because it added many more cards to the cards table and involved more self-joining queries. Combined with an elegant cache-clearing solution that insures the counts table is repopulated when relevant database changes occur, this table has proven a very effective tool for increasing query speeds while actually reducing overall database complexity.

Simplified Relationships

Speaking of reducing complexity, one of the most effective tools for improving performance and reducing bugs has been eliminating unnecessary conceptual complexity.

For example, the previous project management system involved an interaction of Campaigns, Projects, and Tasks, each of which was represented in Decko as a separate cardtype. As one example of the complexity created, Campaigns showed a list of participants, who were defined as anyone having either (a) edited a project task associated with one of the campaign's projects (b) discussed one of the campaign's project, or (c) edited a value for which both the company AND metric were associated with on of the campaign's projects. While Decko is an exceptional tool for rapidly building sites that traverse such relationships even without any custom coding, these kinds of card queries cannot scale without the help of lookup tables.

The simplification of Projects obviated the need for such optimizations, and the new approach has the added benefit of actually appealing to users.

Similarly, Topics and Companies previously suffered from an overly complicated relationship. Topics could be related to companies through sources, notes, or metrics. Now that they are considered related only through metrics, the relationship is much simpler and can be queried with the help of the *answers* table. Should WikiRate revert to greater complexity, a new lookup table will be in order.

It is worth noting that much of the code for lookup tables has been generalized so that it will be easier to embrace this recurring optimization pattern as needed.

Task 6.2: Improvements to Decko platform

Note that we refer to the platform as "Decko" throughout for simplicity's sake, even though the rebranding from *Wagn* to *Decko* will not formally be complete until the Decko 1.0 release. That release, which was previously scheduled for this year, is now scheduled for Autumn 2017. Precedence was given instead to developing functionality vital to WikiRate.org

This year generally involved fewer significant overhauls to the Decko Framework, given that the first two years included significant architectural milestones like the first gem release, the new history system, and the completion of the events API. Nonetheless, WikiRate needs did drive significant improvements to the platform.

Performance Improvements

Memcached

The first and simplest performance bug identified after the deployment of New Relic monitoring (as discussed in D7.7.6) was the overuse of Memcached SET calls in tracking card sets for future clearing. A quick refactor had an immediate and powerful impact.

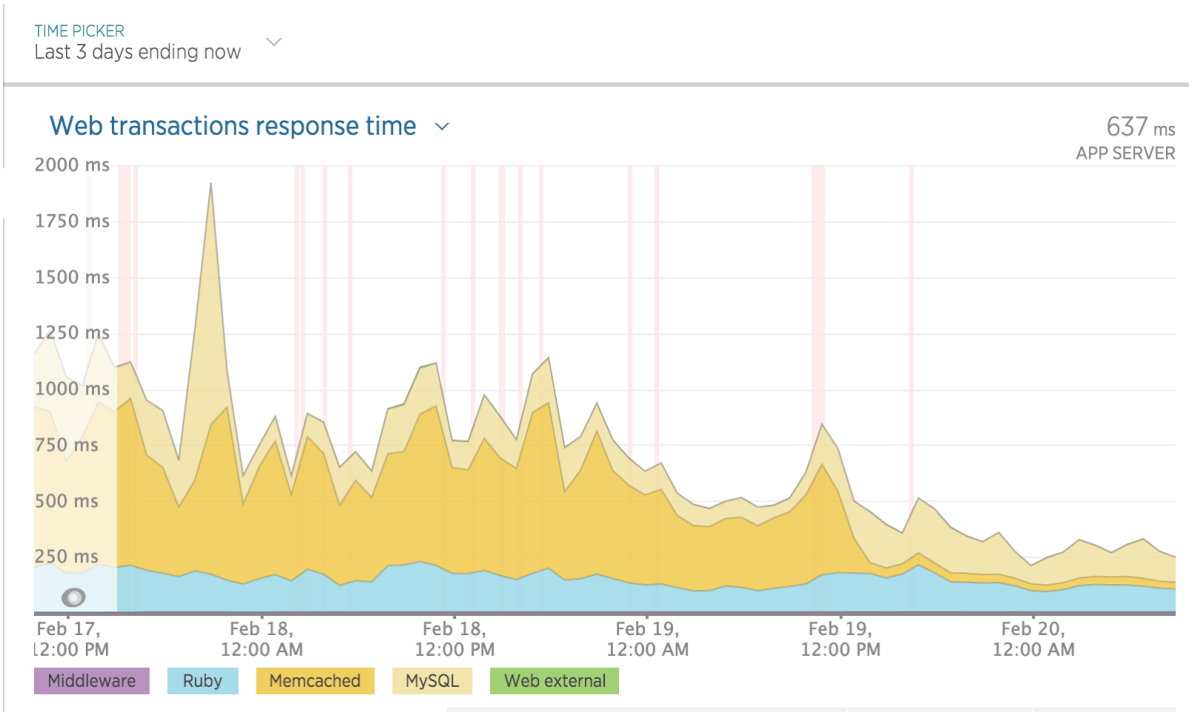


Figure 63 New Relic performance, Memcached SET bug fix

The patch was deployed on the afternoon of February 19th. The pink vertical lines indicate critical performance problems; with this one fix, they virtually disappeared, as the wide "Memcached" band shrank to its expected width (a change that proved durable).

Cookie-based Session Storage

The next opportunity identified with New Relic's help was the burden placed on MySQL by our database-driven session handling. When this was replaced with encrypted cookie-based session storage, we saw further improvements, as shown in this view of database usage:

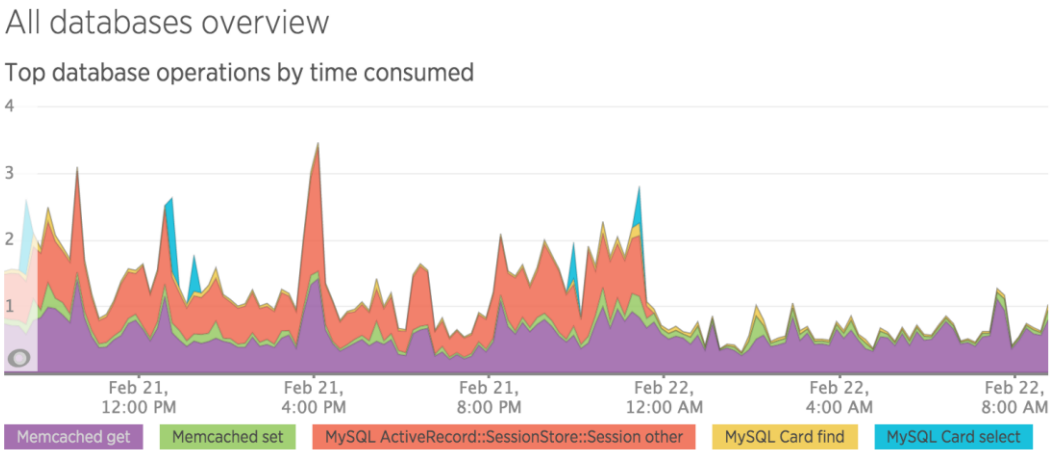


Figure 64 NewRelic, removing Sessions from MySQL

Notice the complete disappearance of the red band representing session-related database requests.

File handling

Because many early Decko use cases involved significant security needs, the platform was originally developed to check permissions before allowing the webserver to render files. Not only did this add many web requests to Decko's workload – including permission processing for every image request – but it also required the files to be served from the same server as the application code. This entanglement would make a multi-server architecture unfeasible.

In year 3, Decko's file handling was upgraded to support both the old high-security configuration and the new high-speed configuration. After the upgrade was deployed, WikiRate files were moved to a cloud server. This had the immediate impact of reducing WikiRate server's load and the long-term effect of readying WikiRate for a multi-server deployment when needed. (More on those plans in D7.7.6).

Card Fetching

Whereas card queries are Decko's core tool for searching lists of cards, *Card.fetch* is the primary API for retrieving cards one at a time. The simple API is backed by a sophisticated integration of memory (Memcache) and database lookups, and includes retrieval of *virtual cards* (cards formed by combining other cards that do not appear directly in the database). The card fetching algorithm was refactored and optimized in year 3 and has been markedly faster and more reliable.

Events upgrades

Decko's *Events* API gives developers access to trigger code at different phases/stages of a card action (create, read, update, delete). Two key changes this year had significant performance impact:

1. *Card actions without transactions*. Previously all actions were conducted within database transactions. This has the benefit of making it possible to roll back changes in case an error happens at any stage. It also has the cost of locking the database for the duration of the action. This was particularly a problem for imports, which often involved many changes and thus locked the database for inexcusably long periods, blocking regular contributions to the site. By making it possible to conduct complex card actions outside of transactions, this significant performance risk was resolved.
2. *Delayed job handling.* Some card actions trigger events that can take time but do not need to be completed prior to completing a web request. For example, if Ramona edits a project, an email is sent notifying anyone "following" that project, but there is no need to make Ramona wait for the email to be sent before she carries on with her work. Decko now handles delayed jobs, so that such events can be postponed until after the request is completed. This does not necessarily reduce overall server load, but it certainly improves the user experience with a faster response time, and it also organizes lower priority events into a single delayed jobs queue.

View Upgrades

The performance improvements mentioned above are fully deployed and activated. The new view handling, while fully deployed, involves designed performance improvements that are not yet activated.

The inspiration for year 3's significant refactor of Decko's view handling is the fact that:

1. WikiRate does far too much duplicate rendering
2. Page caching proposals have generally been rejected because of the extent to which pages vary based on context
3. Views are the building blocks of card formatting
4. Views are often reused throughout the site
5. Card updates can alter views without altering the entire page

A new *view caching* design was thus proposed that would provide a key element of a long-term scaling strategy. It entailed a new more object-oriented approach to card views, and a stub-based caching

approach that embraced Decko's nesting patterns. The view API has been upgraded, and in the process nearly every significant view was reviewed, updated to use the new API, and, where possible, improved.

Before the caching can be activated, however, contextual names must be refactored into views. Having been chastened by previous premature deployments, we will test the new caching system extensively on our staging server before deploying it live.

Security

Decko introduced two security-related improvements this year:

1. *Token-based authentication*, which will allow remote services to use WikiRate's API without requiring the default session-based authentication, which is convenient for browsers but inconvenient for application integrations
2. *Updated captcha functionality*, making use of the latest mouse-movement-based ReCaptcha confirmations for signups and other actions permitted to visitors not logged in.

Code Organization and Documentation

To make the Decko code more approachable to other developers, we undertook a reorganization of major Ruby Classes and established a clean namespace hierarchy. In the process, we added extensive new documentation, following Yardoc conventions to support automated generation of inline code documentation.

We also added search commenting, making it easier to understand and debug logged queries.

Other

- Decko now supports a new WYSIWYG (What You See Is What You Get) editor called ProseMirror. This streamlined editor is the new default rich text editor and has supplanted the less wieldy TinyMCE on WikiRate.org.
- The API for links / paths was upgraded and
- Automated Heroku deployment makes it easy for users to install and try out Decko
- The *Recent Changes* upgrades described in the Implemented Features section above were implemented on the platform level and would therefor apply to all Decko instances.

Task 6.3: WikiRate system administration

Performance and Error Monitoring

The most significant upgrade to WikiRate's system administration in year 2 was the addition of NewRelic as a performance-monitoring tool. Almost all the performance enhancements implemented in T6.1 and T6.2 were driven by NewRelic data, and NewRelic was used to measure and fine-tune their effect. The tool has been central to restoring WikiRate to strong performance after its very poor performance at the 24-month review.

NewRelic has been configured to monitor WikiRate servers and to send notifications if they are under duress. We also use it to track bug frequency, as a supplement to our use of Airbrake for a similar purpose.

Server Architecture

Our server architecture is largely unchanged from year 2. D7.7.6 (*Final Technical Evaluation*) discusses the appropriateness of this architecture and the timeline for upgrading to a multi-server architecture.

WikiRate.org is currently provided by a single Hetzner EX40 dedicated root server as follows:

- high-performance Intel® Core i7-4770 Quad-Core processor
- 32 GB DDR3 RAM.
- two 2 TB SATA 6 Gb/s

The same server is used for our internal documentation site (also using Decko) and our public project site (wikirate.eu).

The development server is identical. It is used for testing, staging and demoing WikiRate changes. Both servers are running:

- Ubuntu 12.04.5 LTS
- Ruby 2.3.0
- Passenger 5.0.27
- Mysql 5.6.30
- Apache 2.2.22

Development Environments

As in previous years, developers at Grass Commons and Cambridge developed primarily on laptops and then coordinated code changes with the rest of the development team through GitHub pull requests.

Each development team member also maintains an open Slack window for WikiRate, which integrates:

- Instant messages and group chats with team members (including nontechnical team members)
- GitHub updates and pull request notifications
- Updates to PivotalTracker stories
- Server Error notifications from Airbrake
- Notifications regarding automated testing from Semaphore

All developers continue to work collegially and to share resources openly. Daily meetings (standups) assure consistent coordination, and biannual developer retreats are used for intensive planning, designing, developing, and team building.

Deployment and Backup Tools

We continue to use Capistrano to manage all our deployments to both the development and the production server. With Capistrano we can run full updates with a single command that manages:

- WikiRate code updates and data migrations
- Decko code updates and data migrations
- Updates to dependent libraries
- Clearing tmp directories
- Restarting Passenger

We have also created Capistrano commands to:

- generate backups
- refresh the development copy of WikiRate.org from a backup of the production site
- copy and install a backup to a local development environment

Additional daily backups are generated on the production server and copied to the development server using cron jobs.

While authentication details are naturally kept private, all Capistrano commands are made public in WikiRate's github repository for reuse/modification in comparable environments.

Testing Environments

Both Decko and WikiRate are tested using Rspec and Cucumber. Continuous integration testing is performed using Semaphore, which automatically runs applicable test suites whenever pull requests are submitted for Decko or WikiRate. These pull requests are also automatically tested for code quality and adherence to style guidelines using Code Climate.

Task 6.4: Application Programming Interface (API) and Plug-ins

Decko features two primary API's: the Mods API and the RESTful Web API. The former is the means by which Decko's functionality can be altered or extended, and the latter is the means of interacting with Decko data over the web.

5.1 Mods API

Decko's mods API is the primary means by which functionality *not* distributed in default Decko instances is created or customized for WikiRate.org.

There is not a bright line between platform improvements (T6.1) and mods API improvements (T6.4), and the aforementioned events and view API improvements could clearly be listed here. However, whereas those changes were motivated predominantly by platform need, the templating API improvements were very directly driven by mod developer needs and are thus more clearly in the domain of T6.4.

Early in WikiRate's development, the dev team tended to follow the traditional Decko pattern of emphasizing traditional card configuration (in databases) over "card code" configuration. The two approaches follow similar organisational patterns, but while database configuration is easier for quick, live code changes, card code configuration is more conducive to code review and structured releases.

As the project's database and deployment infrastructure have scaled, it has become increasingly expedient to emphasize card configuration. This has led to three key patterns:

1. Supporting smooth transitions from database to code configuration. This not only supports updates to old WikiRate code, it also provides for a general Decko development patterns in which collaborators can immediately establish a site and begin configuring it and using it without involving code changes, and then later mature the site and engage a developer in fluid upgrades as needed.
2. The development of view-framing helper methods that make it easier to make use of the mobile-friendly Bootstrap framework integrated into Decko in year 2
3. HAML templates, which provide for cleaner view code by allowing complex views to be represented in separate .haml files.

5.2 RESTful Web API

The core model of Decko's (and thus WikiRate's) RESTful Web API has been well established since Year 1. However, new JSON and CSV formats – in Decko terms new "views" available via the same RESTful construct –have been devised and will soon feature prominently in WikiRate's answer search interface.

Source Code Repositories

WikiRate code is distributed between three Github repositories

1. The primary WikiRate repository at **https://github.com/wagn/wikirate**
2. The Decko framework repository at **https://github.com/wagn/wagn**
3. A repository for commonly shared mods at **https://github.com/wagn/mods**