Improving code sharing between councils

MHCLG Local Digital Fund Discovery report March 2020

London Borough of Croydon
Brighton & Hove City Council
Bracknell Forest Council
Oxford City Council

Our partners and team

CROYDON
www.croydon.gov.uk

Brighton & Hove
City Council

Bracknell
Forest
Council

dxw digital

OXFORD
CITY
COUNCIL

agile{collective}

(Some of) our team

Ert Erol
Product Lead, dxw

Will Callaghan
Delivery Lead, Croydon

John Waterworth
Research Lead, dxw

Aaron Hirtenstein
Governance Lead, Agile Collective

Finn Lewis
Technical Lead, Agile Collective

Stephen Cox
Drupal Lead, Agile Collective

Andrew Katz
IP and Licensing Lead, Moorcrofts

Summary

Summary

→ Common problems and needs make sharing attractive for councils

→ It is hard for councils to continually improve their websites

→ The codebase we’ve got currently needs work to get it into a shareable state

Summary

- In order to collaborate, councils need a lightweight framework, almost a “piece of paper”
- Collaboration will provide alignment, but it also should not slow us down (different models for Alpha for different capabilities)
- Councils expect a clear product direction/ roadmap and will need help with support

The councils we interviewed want our Drupal code sharing efforts to succeed...

> “We’ve all got the same underlying challenges. And we all need to do more with less.

> “If we open sourced this tomorrow, I’d have no problem with it.

> “We want to collaborate, but we also must be able to do our thing from time to time.

Our discovery work with councils

MHCLG asked us to narrow the scope of our initial bid. Rather than look at code sharing in general, we agreed to focus on the barriers to sharing our Drupal 8 code

Problem statement

How can we help councils co-develop, share and maintain open-source Drupal code for their citizen-facing websites?

Sprint 1

Going broad with primary research with Councils to determine the right focus for subsequent sprints

Sprint 2

Analysing the research, working on the code and theme audits, and researching governance and decision making models

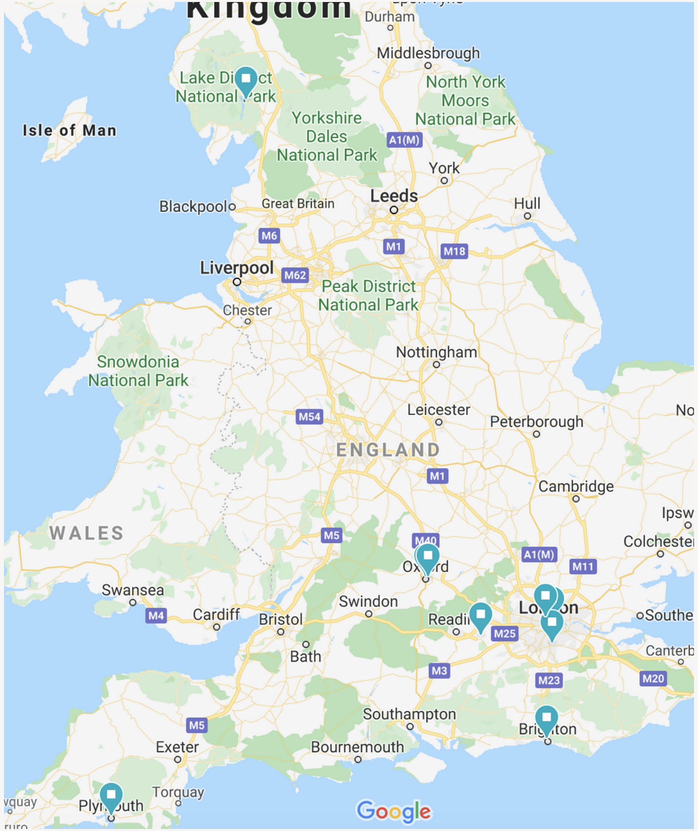
Sprint 3

Developing a model to explain what might enable or block a council from adopting a shared Drupal codebase, preparing the proposal around technical architecture, finalising governance and decision making models, drafting an MoU

We conducted primary research with councils, desk research, and technical audits to understand council needs, work out how they collaborate, and prove what’s technically possible

We did 10 face to face and video interviews with councils

- A mix of unitary authorities and city, county and district councils
- Some were on Drupal, others were considering a switch
- Two had in house development teams, others relied on agencies. Some had no dev capacity at all



The questions we asked

How do councils build, operate and maintain their citizen-facing websites?

What do they know about the existing code sharing work, what excites them and and what questions or concerns they have?

How might they make the decision to join and what kind of sharing arrangement would they envisage?

What we found

Finding

1

It’s hard for councils to continually improve their websites

Councils face significant pressure to improve the service they provide to the public through their websites, within already tight budget constraints.

They want desperately to stop reinventing the wheel and to break out of the wasteful cycle of procuring a new, and often not very good, website every few years.

Finding

2

Common problems and needs make sharing attractive

Councils are working on common problems and their citizens have common needs. They believe that collaborating and sharing is a good way to do better within the resources they have.

As well as sharing code or components councils value being part of a supportive community that shares ideas and ways of working.

Finding

3

Open source, and Drupal, can work well for councils

Open source technologies can be an acceptable choice for councils, with around 30 per cent using an open source platform.

Drupal is seen as an exemplar open source technology and community. It supports the low-code, configurable approach that many councils are taking.

Some councils distrust open source, and default to procuring proprietary solutions. A codebase created and used by leading councils should increase confidence.

Finding
4

Code sharing can work for councils with different starting points and levels of capability

Code sharing can work for councils who take the complete codebase to build a new website, and for councils who take components to add to their existing website.

Code sharing can work for councils who buy in their Drupal development, as well as for councils who have their own capability.

Finding

5

Councils expect ‘a piece of paper’ but not a weighty agreement

The Local Digital Declaration offers a starting point for code sharing, but lacks concrete mechanisms to make it happen.

Councils expect code to be shared within a simple and transparent agreement, that is open to councils joining and leaving.

Asking for money means procurement, which can create barriers and slow adoption.

Finding

6

Councils expect a clear product direction and roadmap

All councils expect open and effective product management with a clear roadmap. Councils looking to contribute code also expect to be involved in decisions.

Councils believe that their common needs will reduce conflict over priorities and features.

Councils may cluster around particular technologies, such as search, forms and low-code platforms.

Finding

7

Councils are concerned about receiving and providing support

Councils with less technical capability are concerned about the quality of the codebase and the level of support and documentation they may receive.

Councils who expect to contribute to the codebase are concerned about providing support and the cost of meeting additional coding and testing standards.

Finding

8

None of the councils wanted to buy ‘Local Gov Drupal as a service’

We asked councils if they would prefer a managed service of code, hosting and support, versus joining a ‘code sharing club’ which would need some development expertise.

None of the councils wanted to buy the service, including those who weren’t able to build for themselves. Instead, all saw this as an opportunity to skill up and take ownership of their sites.

We think there are councils who want to buy this service, and will try to identify them during an Alpha phase.

Finding

9

Teams need approval from their senior leadership

Adopting an open source codebase developed in collaboration with other councils will need the approval of senior leadership.

Collaborating and sharing with other councils is generally seen as ‘a good thing’. But poor experiences with shared services agreements can make councils wary.

A clear proposition and benefits case will help teams gain the approval they need.

Finding
10

Councils have website challenges that code sharing alone won’t solve

Sharing a high quality codebase can bring significant benefits both for councils and for the people using their websites.

But councils have significant challenges with product ownership and management, content strategy and design, and performance measurement and analysis.

While a shared codebase won’t solve these problems, a supportive community around that codebase can help councils share and adopt better practices.

We created a ‘switching forces’ diagram to summarise things that might enable or block a council from adopting a shared Drupal codebase

Switching forces for councils

► Push

- Not meeting needs of public
- Technical challenges
- Tight resource constraints

► Pull

- Stop reinventing the wheel
- Multiply development effort
- Be part of a community

Inertia

- Limited capability
- Divided ownership
- Tight resource constraints

Anxiety

- Burden on contributors
- Support and documentation
- Distrust of open source

The need to rebuild is the most pressing push

The need to rebuild their current website, because of obsolete technology and problems meeting user needs, creates the most common push.

For councils already using Drupal, the need to upgrade to Drupal 8 (a major release) can be a significant push.

Pull

Adopting something working for other councils is the biggest pull

The opportunity the build on code that is already working for other councils is the biggest pull.

The Drupal codebase allows councils to multiply their development effort, both by sharing development and testing efforts, and by moving to lower-code approaches.

Councils are also keen to join a community of like minded and supportive organisations.

Inertia

Councils with the most problems can also have the most inertia

Councils can struggle to serve the public with their websites because of limited capability, tight resource constraints and divided ownership.

But those same problems can also make it harder for them to agree on and implement better approaches.

Anxiety

Code quality and support are the greatest anxieties

Councils are concerned about the quality of the codebase, both as a risk and as an additional cost.

Less capable councils are concerned about getting support they may need, and more capable councils are concerned about ending up providing that support.

We created an ‘adoption paths’ model to characterise different reasons why councils might adopt a shared Drupal codebase

Adoption paths for councils

	Problem website	OK website
Drupal	1. Adopt complete codebase	2. Take components
Not Drupal	3. Adopt Drupal and codebase	4. Not an initial target group

1

Existing Drupal users with problem websites can use the complete codebase to restart

Councils who are already using Drupal can restart with the shared codebase.

This might be to help upgrade from Drupal 7 to 8, or to avoid a costly rebuild of a non-responsive and inaccessible site.

2

Existing Drupal users with good websites can take components from codebase to add features

Some councils using Drupal are broadly happy with their current website, both in how it is built and how it is working for the public and for staff.

The shared codebase already has components that those councils are interested in using, and will have more over time.

3

Councils replacing a problem website can use the codebase to kickstart a move to Drupal

Some councils with problem websites are looking to move to a new content management system and to new low-code approaches.

Having a good codebase created and used by other councils makes Drupal a more attractive option.

These findings provide us a good steer to set the boundaries around our Discovery, and any potential Alpha work

The next piece in the puzzle was to find out what is technically possible with the codebase that BHCC and Croydon currently share...

Our technical discovery work

We focused on 3 important questions...

Is the current code in a shareable state?

What might the proposed architecture look like?

How might different organisations contribute and commit to the same codebase?

We audited the code and content architecture, then talked about what we found with developers in Croydon and Brighton. Finally we made some recommendations

What we found

1

The code is generally written, but it’s not yet ideal for sharing

It’s clear that the custom Drupal code and configuration was not originally written to be shared and not architected to be fully modular.

Sharing the code in its current state will limit the number of councils who will be able to benefit.

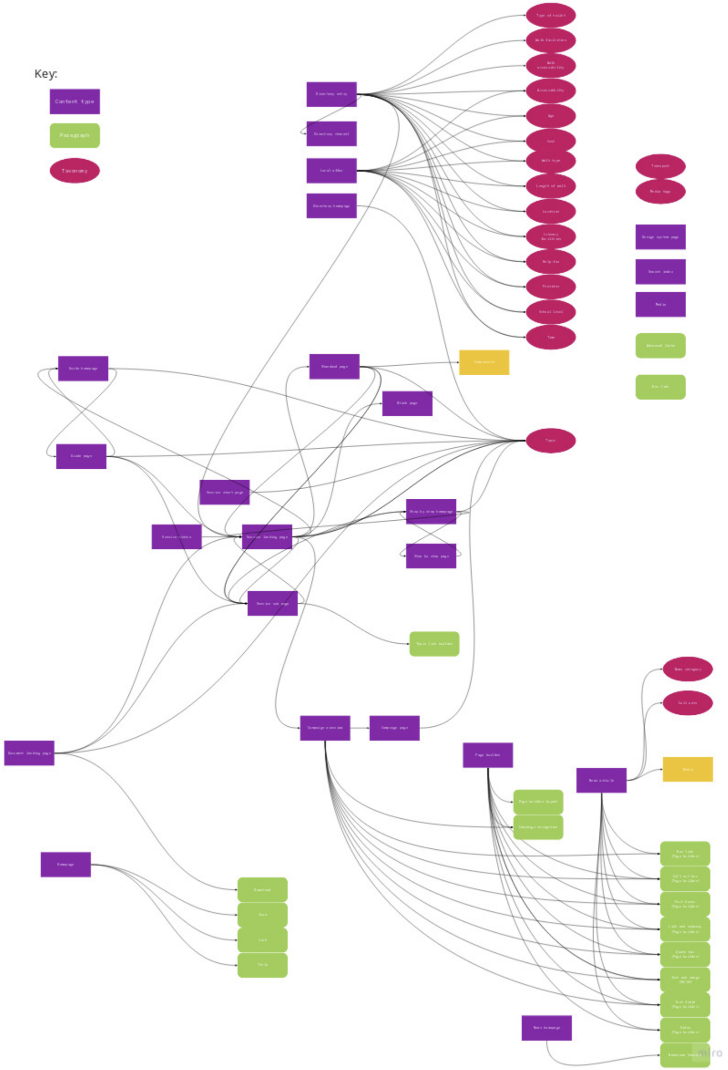
2

The dependencies need untangling

The architecture of the content entities currently includes many dependencies, some of which are circular.

Entity map

2


Key:
Content type
Paragraph
Taxonomy
Directory entry
Directory channel
Local offer
Directory homepage
Type of toilet
Walk facilities
Walk accessibility
Accessibility
Age
Cost
Walk type
Length of walk
Location
Library facilities
Help for
Provider
School level
Time
Transport
Media tags
Design system page
Search index
Media
Advanced links
Box link
Standard page
Components
Block page
Topic
Guide homepage
Guide page
Service start page
Service status
Service landing page
Step by step homepage
Step by step page
Service sub page
Topic list builder
Campaign overview
Campaign page
Document landing page
Page builder
Page builder layout
Campaign navigation
News article
News category
Call outs
Media
Box link (Page builder)
Call out box (Page builder)
Fact boxes (Page builder)
Link and summary (Page builder)
Quote box (Page builder)
Text and image (50/50)
Text field (Page builder)
Video (Page builder)
News homepage
Homepage
Download
Icon
Link
Title
miro

Entity map

Is the current code in a shareable state?

What might the proposed architecture look like?

How might different organisations contribute and commit to the same codebase?

Proposal

1

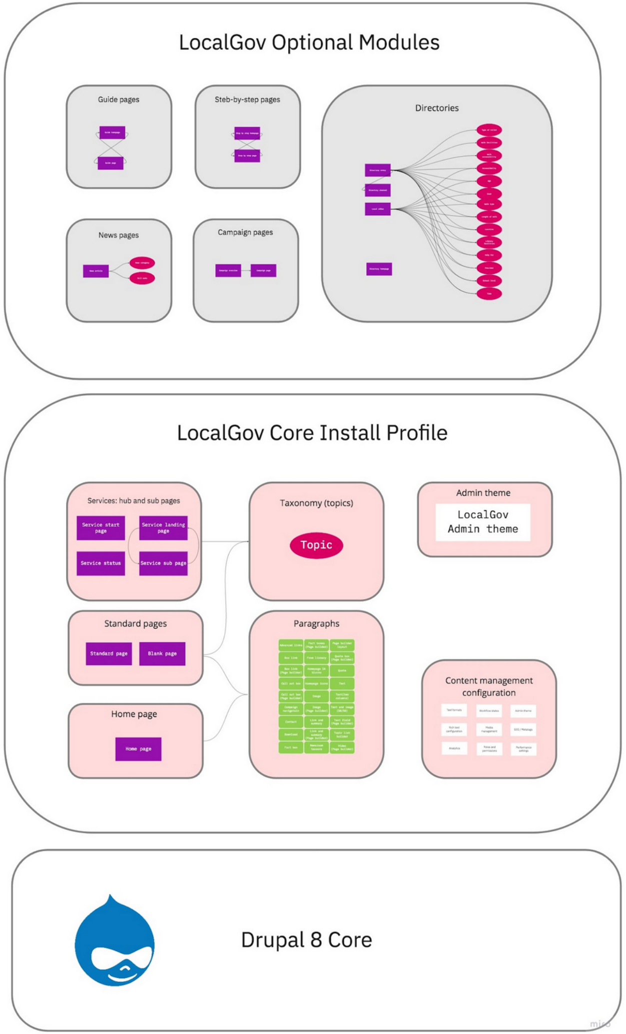
Architect LocalGov Drupal as a Drupal distribution

Architect LocalGov Drupal as a Drupal distribution, to include:

- Drupal Core
- LocalGov Core and base theme
- Modular architecture of optional features

Proposal

1



LocalGov Optional Modules

Guide pages

Guide homepage

Guide page

Steb-by-step pages

Step by step homepage

Step by step page

News pages

News article

News category

Call outs

Campaign pages

Campaign overview

Campaign page

Directories

Directory entry

Directory channel

Local office

Directory homepage

Type of toilet

Walk facilities

Walk accessibility

Accessibility

Age

Cost

Walk type

Length of walk

Location

Library facilities

Help for

Provided

School level

Time

LocalGov Core Install Profile

Services: hub and sub pages

Service start page

Service landing page

Service status

Service sub page

Taxonomy (topics)

Topic

Admin theme

LocalGov Admin theme

Standard pages

Standard page

Blank page

Paragraphs

Advanced links	Fact boxes (Page builder)	Page builder layout
Box link	From library	Quote box (Page builder)
Box link (Page builder)	Homepage IA blocks	Quote
Call out box	Homepage icons	Text
Call out box (Page builder)	Image	Text(two columns)
Campaign navigatoin	Image (Page builder)	Text and image (50/50)
Contact	Link and summary	Text field (Page builder)
Download	Link and summary (Page builder)	Topic list builder
Fact box	Newsroom teasers	Video (Page builder)

Home page

Home page

Content management configuration

Text formats	Workflow states	Admin theme
Rich text configuration	Media management	SEO / Metatags
Analytics	Roles and permissions	Performance settings

Drupal 8 Core

Proposal

2

Make changes to content architecture

To support flexibility and modularity, changes to the content architecture should be made, including:

→ remove unnecessary dependencies

→ include common content types required by all councils in the LocalGov Core Drupal distribution

→ ensure optional content types and features are fully modular

Proposal
3

Make recommended changes to custom code

To avoid increasing future technical debt, the existing shared codebase should be refactored to:

- → rationalise dependencies
- → establish consistent naming conventions
- → remove unnecessary complexity
- → adhere to best practices in Drupal coding standards

Is the current code in a shareable state?

What might the proposed architecture look like?

How might different organisations contribute and commit to the same codebase?

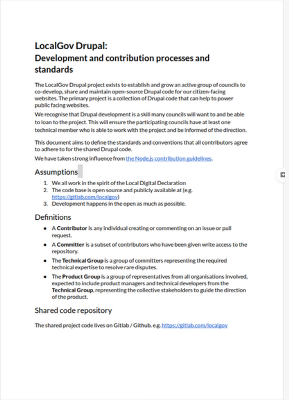
Proposal

4

Adopt and promote clear contribution guidelines

This sets expectations for people contributing to the project, including:

→ how to contribute
→ peer review
→ coding standards
→ documentation
→ automated testing



LocalGov Drupal:
Development and contribution processes and standards

The LocalGov Drupal project exists to establish and grow an active group of councils to co-develop, share and maintain open-source Drupal code for our citizen-facing websites. The primary project is a collection of Drupal code that can help to power public facing websites.

We recognise that Drupal development is a skill many councils will want to and be able to loan to the project. This will ensure the participating councils have at least one technical member who is able to work with the project and be informed of the direction.

This document aims to define the standards and conventions that all contributors agree to adhere to for the shared Drupal code.

We have taken strong influence from the Node.js contribution guidelines.

Assumptions

1. We all work in the spirit of the Local Digital Declaration
2. The code base is open source and publicly available at (e.g. https://gitlab.com/localgov)
3. Development happens in the open as much as possible.

Definitions

- A **Contributor** is any individual creating or commenting on an issue or pull request.
- A **Committer** is a subset of contributors who have been given write access to the repository.
- The **Technical Group** is a group of committers representing the required technical expertise to resolve rare disputes.
- The **Product Group** is a group of representatives from all organisations involved, expected to include product managers and technical developers from the **Technical Group**, representing the collective stakeholders to guide the direction of the product.

Shared code repository

The shared project code lives on Gitlab / Github. e.g. https://gitlab.com/localgov

Development and contribution processes and standards

Proposal

5

Make use of open source automation services

For example, Gitlab.com allows you to run tests for free

GitLab Open Source Program

Our top tiers are free for all Open Source projects

Our governance and licensing work

Again, we asked 3 important questions...

What’s the most appropriate model for us?

How can organisations join, contribute and leave?

How might we take decisions in a larger group? (product + tech group)

We spent time fleshing out possible licensing and governance options...

[Governance and Licensing models]

These decisions are important because they lay the foundations for collaboration and ensure that councils feel comfortable joining the project

We researched the governance and licensing of similar projects. Then we discussed different options and agreed what was "good enough for now"

Comparison of open source models

What we found

Conclusion

1

There is no need for a formal incorporated body at this stage

In the future, once the project has achieved a certain level of maturity and size, it can move to a **more formal foundation** that allows for different types of members (e.g. commercial) and a financial model (e.g. subscription for members).

Conclusion

2

Agree a Memorandum of Understanding (MoU)

Instead of a more formally constituted body.

The MoU needs to be **lightweight** enough that people can easily understand what they are signing up to.

It also links to separate documents covering product management and technical processes / standards – **“how” this will work in practice.**

Conclusion

3

Licensing models are complicated

Agreeing the **purpose** and **values** of the project enabled the team to find agreement.

Specifically, working in the open and publishing under open source licences.

From these, the most appropriate licence became clear: **open source GPL (version 2 or later)** following Drupal’s licence.

What’s the most appropriate model for us?

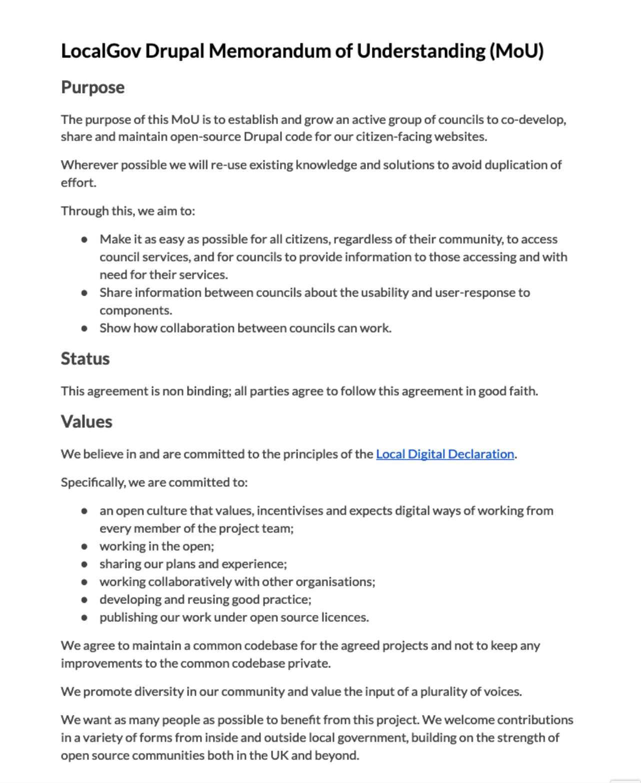
How can organisations join, contribute and leave?

How might we take decisions in a larger group? (product + tech group)

We drafted a Memorandum of Understanding (MoU) and completed an initial round of testing with partner councils. It covered the following...

MoU contents

- → Purpose
- → Values
- → Intellectual property and licensing
- → Liability
- → Governance
- → Joining and Leaving
- → Commitment (time)
- → Finance



LocalGov Drupal Memorandum of Understanding (MoU)

Purpose

The purpose of this MoU is to establish and grow an active group of councils to co-develop, share and maintain open-source Drupal code for our citizen-facing websites.

Wherever possible we will re-use existing knowledge and solutions to avoid duplication of effort.

Through this, we aim to:

- Make it as easy as possible for all citizens, regardless of their community, to access council services, and for councils to provide information to those accessing and with need for their services.
- Share information between councils about the usability and user-response to components.
- Show how collaboration between councils can work.

Status

This agreement is non binding; all parties agree to follow this agreement in good faith.

Values

We believe in and are committed to the principles of the Local Digital Declaration.

Specifically, we are committed to:

- an open culture that values, incentivises and expects digital ways of working from every member of the project team;
- working in the open;
- sharing our plans and experience;
- working collaboratively with other organisations;
- developing and reusing good practice;
- publishing our work under open source licences.

We agree to maintain a common codebase for the agreed projects and not to keep any improvements to the common codebase private.

We promote diversity in our community and value the input of a plurality of voices.

We want as many people as possible to benefit from this project. We welcome contributions in a variety of forms from inside and outside local government, building on the strength of open source communities both in the UK and beyond.

Memorandum of Understanding

1

Purpose

To establish and grow an active group of councils to co-develop, share and maintain open-source Drupal code for our citizen-facing websites.

2

Values

Based on and inspired by the Local Digital Declaration:

- an open culture that values, incentivises and expects digital ways of working from every member of the project team
- working in the open
- sharing our plans and experience
- working collaboratively with other organisations
- developing and reusing good practice
- publishing our work under open source licences.

Local Digital Declaration

3

Intellectual Property and licensing

Code and other materials committed to the project will be automatically licensed to the other participants on an agreed open source licence, but, subject to that licence, each contributor will retain ownership of their copyright and other intellectual property.

Drupal is released under the GPLv2 (or later version), therefore by default, all Drupal code will be published under GPLv2 (or later version).

4

Liability

All code and other materials are provided to the project “as is”.

No party bears liability for any contribution. Code and other materials are provided without any warranty or support or other liabilities.

What we found

5

Governance

The structure will evolve as the project grows and will always aim to be “good enough for now” to enable it to be fit for purpose.

Two main groups that form the administrative body:

- **Product Group** – project governance and product roadmap
- **Technical Group** – development processes, standards, security, committer rights

6

Joining and leaving

New local councils may be admitted by agreement from the existing members.

Any party may quit at any time without notice, but all parties may retain access to the common codebase in the repository on the basis of the open source licence.

7

Commitment

Councils who join the Product Group agree to commit some time to further the aims of the project. The Product Group welcomes commitment from a range of skills, including:

- → Dedicated developer time
- → Management / admin time
- → User research
- → Product management
- → Testing

8

Finance

There is no plan to accept financial donations or hold money at this stage.

At this time we do not envisage any money changing hands between councils.

Our draft MoU was received favourably by every council who read it...

> “Don't think I've ever enjoyed reading an MOU before, but I did like this one.”

> “We didn’t want to be burdened with paper. This nicely covers the minimum we need.”

> “Very good doc. I like that it's concise.”

What’s the most appropriate model for us?

How can organisations join, contribute and leave?

How might we take decisions in a larger group? (product + tech group)

We researched decision making in open source projects, and modelled how the group could work together to make decisions

Apache Foundation

OpenJS Foundation

Apperta

Conclusion

1

Most open source projects use a form of consensus decision making

Consensus	→	“Everyone must agree”
Consensus seeking	→	“If no agreement, we take a vote”
Lazy consensus	→	“Silence means assent”

2

Any process needs to be scalable but grounded in the same principles

- Clear
- Transparent
- Inclusive
- Do-ocratic
- Equivalent

3

Decisions vary in importance

Depending on the potential impact and the area they affect.

Considering the following questions helps to give an indication:

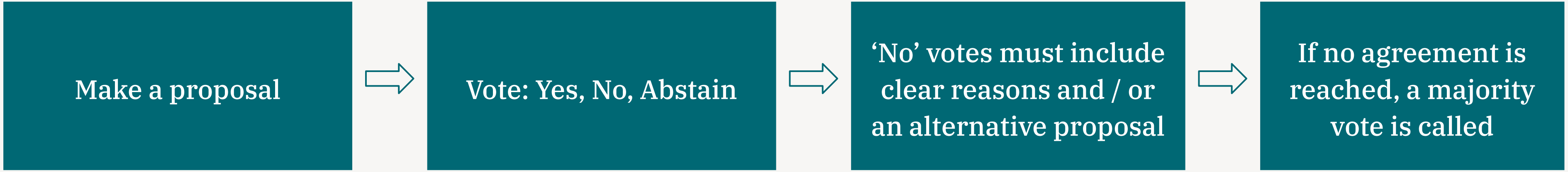
1. Do we need to invest a lot? (time and people)
2. If there is a positive outcome, would that benefit a lot of people radically?
3. If there is a negative outcome, how bad would that be for the project?

Based on our research, we have proposed a combined approach to collaborative decision making

1

Consensus seeking

For important decisions that have the potential to fundamentally alter the nature of the project e.g. membership criteria.



Make a proposal

Vote: Yes, No, Abstain

‘No’ votes must include clear reasons and / or an alternative proposal

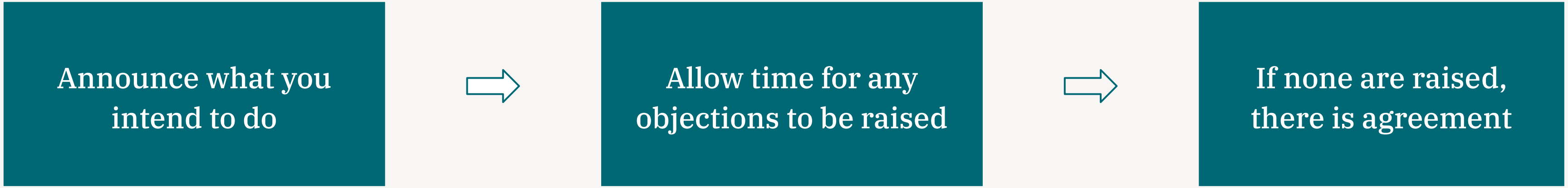
If no agreement is reached, a majority vote is called

2

Lazy Consensus

For minor decisions that will add value but at low risk.

“If no-one objects by tomorrow morning, I’ll assume lazy consensus and do it.”



Announce what you intend to do

Allow time for any objections to be raised

If none are raised, there is agreement

Being open and transparent in collaboration is important, but not at the expense of empowering people to do work

Making product decisions as a group

As more councils join the project, making decisions on the immediate and future priorities of the product will require some principles

We can organise these discussions under two main categories

Sprint planning
(immediate priorities)

Roadmap planning
(soon and mid term
priorities)

In Sprint planning, the Product Group decides on what to focus on in the next 14 days

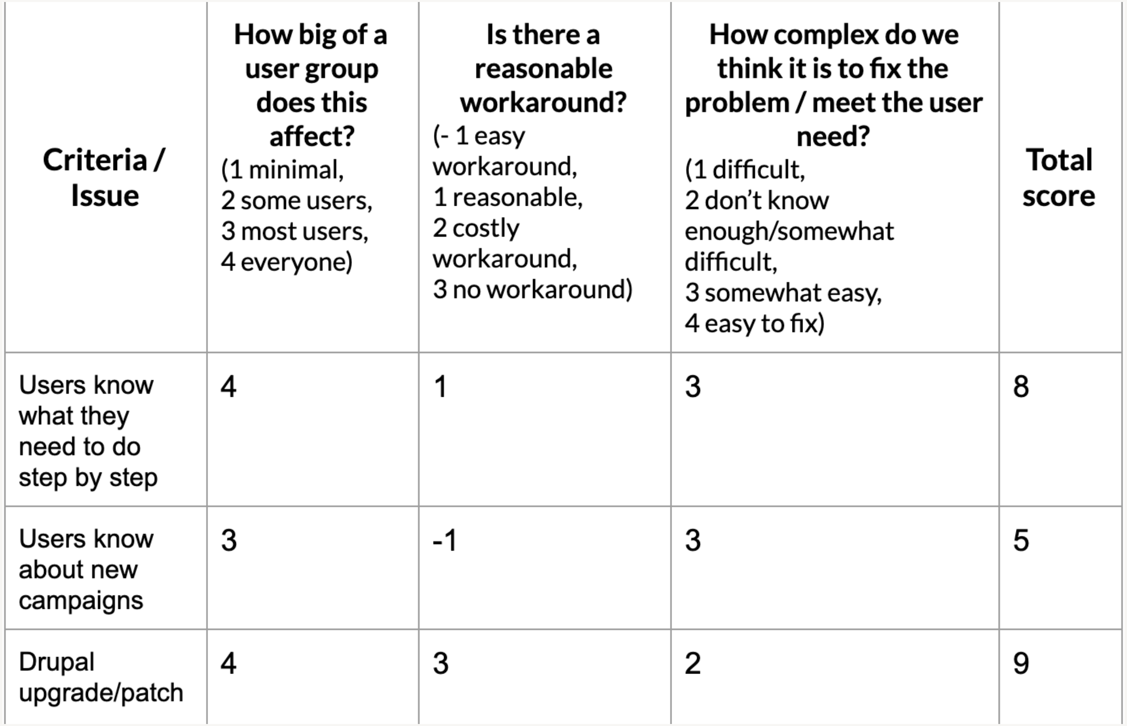
In cases where there might be too many stakeholders deciding on these priorities, we can use a prioritisation matrix.

There are three important factors that affect how we prioritise a problem area or a feature: How big of a user group does this problem affect, is there a reasonable workaround for this, and how complex do we think the work is.

Sprint planning

Here is an example of how that matrix might look.

The higher the score, the more of a priority that issue is for the immediate sprint work.



Criteria / Issue	**How big of a user group does this affect?** (1 minimal, 2 some users, 3 most users, 4 everyone)	**Is there a reasonable workaround?** (- 1 easy workaround, 1 reasonable, 2 costly workaround, 3 no workaround)	**How complex do we think it is to fix the problem / meet the user need?** (1 difficult, 2 don't know enough/somewhat difficult, 3 somewhat easy, 4 easy to fix)	Total score
Users know what they need to do step by step	4	1	3	8
Users know about new campaigns	3	-1	3	5
Drupal upgrade/patch	4	3	2	9

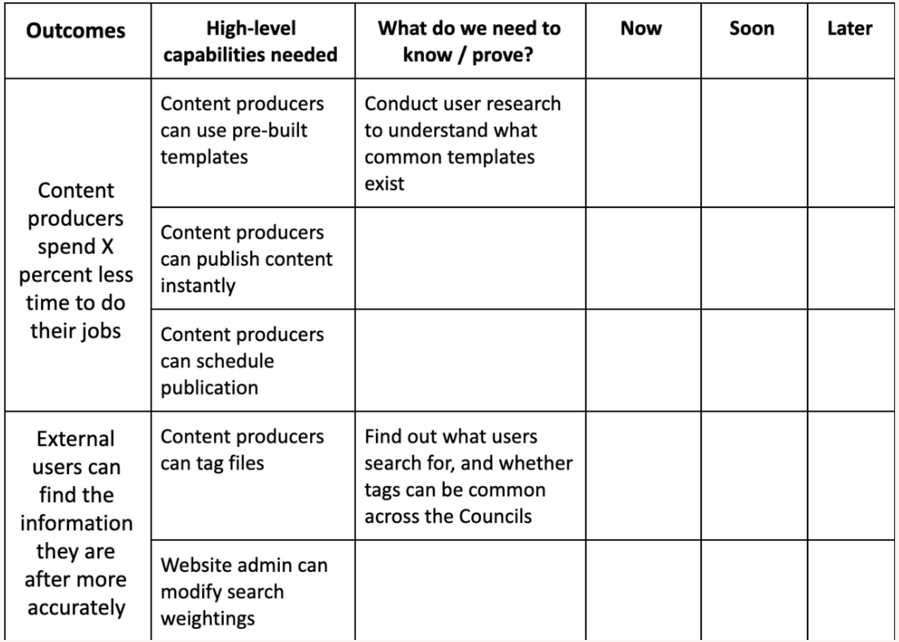
In Roadmap planning, the Product Group agrees on how the product should evolve

The Roadmap is not a time-based plan that the team follows to the letter, as it’s more of a team artefact that facilitates conversations on:

- prioritisation
- dependencies
- future direction

Roadmap planning

The product roadmap must present a series of outcomes, as in a specific value offering either to the citizens, or to the Council users who produce content



Outcomes	High-level capabilities needed	What do we need to know / prove?	Now	Soon	Later
Content producers spend X percent less time to do their jobs	Content producers can use pre-built templates	Conduct user research to understand what common templates exist			
	Content producers can publish content instantly				
	Content producers can schedule publication				
External users can find the information they are after more accurately	Content producers can tag files	Find out what users search for, and whether tags can be common across the Councils			
	Website admin can modify search weightings				

Roadmap planning

The Product Group should meet the first time face-to-face and preferably spend a day on mapping out what outcomes they would like to generate

We propose that after that the Group meets once a month to review the roadmap, add/ remove items based on what might be coming out of user research in that month, and re-publish the updated roadmap.

The decisions taken at the roadmapping sessions must take place after the Group had a good amount of time to discuss all candidates, and weigh them up against the vision statement

This is the “**decide after discussion**” model, and will require arriving at an informed **consensus** at the end of the discussion.

For example, if our vision is to help the biggest number of councils in the UK with an open-source Drupal website, then the desired outcome can be to reach a certain amount of Councils by a certain date

That means we might need new capabilities to support the onboarding of new councils, as opposed to adding more features for the existing members.

Or if our vision is to enable the content publishers of the current members do their job more quickly, then the capabilities we’ll need are more of a priority compared to new things we might need to onboard new members

Benefits case

Several times during this Discovery we’ve discussed creating a ‘code sharing club’ versus a third party providing the Drupal code, hosting and technical support as a service

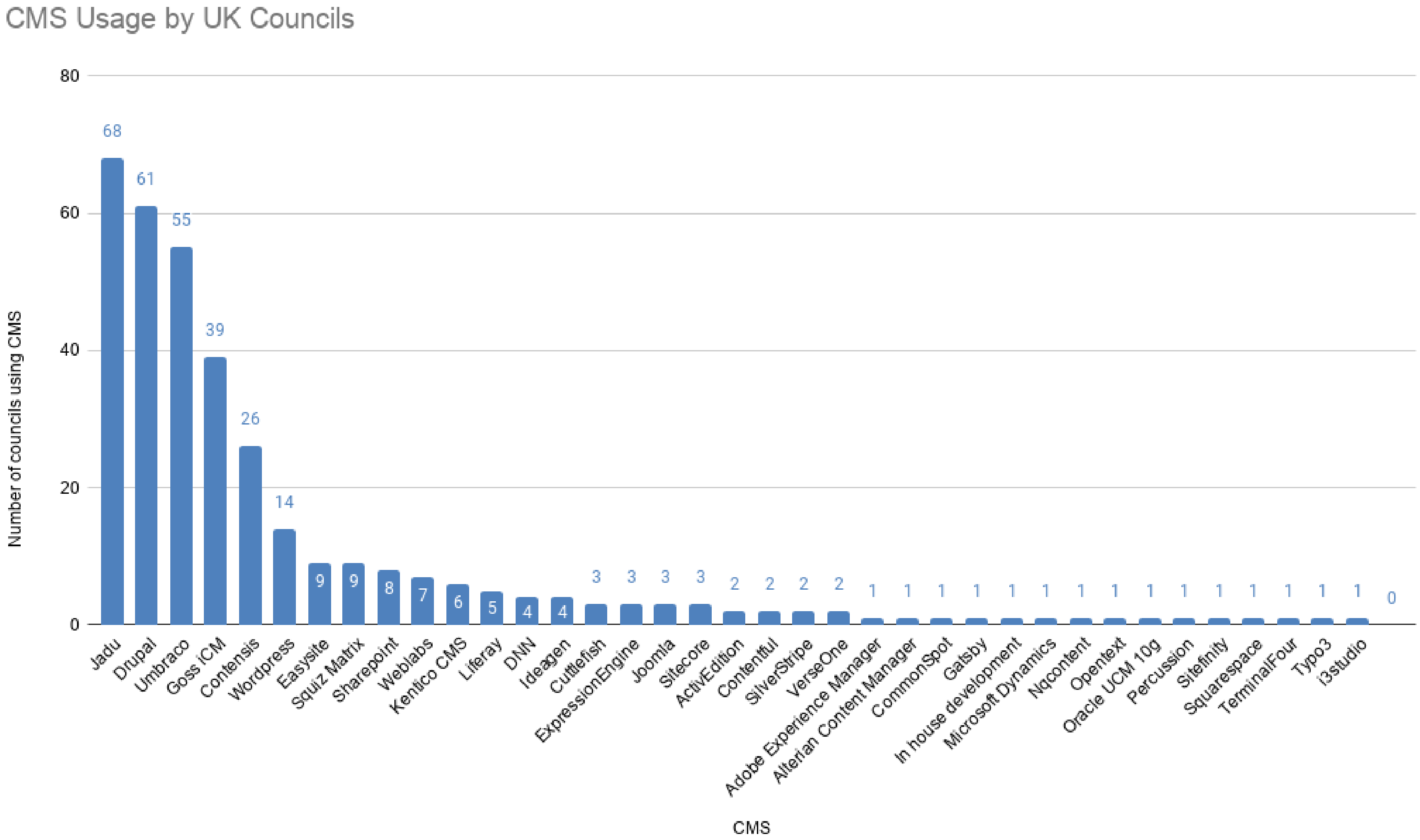
The councils we spoke to, when given the choice, preferred to skill up and take ownership of their sites rather than buying the service from a supplier

As such, our Discovery has focused on the mechanics of creating a club. This doesn’t rule out creating a service however, and the demand for this should be explored at Alpha

We’re reasonably certain about the benefits a club would provide, and in Alpha we’ll attempt to estimate the impact of ‘Local Gov Drupal as a Service’

Code sharing club

Drupal is the second most popular publishing platform solution for UK councils, with 61 organisations using it. These are the most likely candidates to join the club



CMS Usage by UK Councils

There are 4 ways councils could save time and money by joining the club

1. Launch new council sites more quickly
2. Build fewer standalone “microsites”
3. Reduce unnecessary customer contact
4. Sharing knowledge and skills makes us better

Launch council sites more quickly

Croydon launched site in ⅓ time taken by Brighton & Hove

Council type	Cost	Cashable benefit
Switchers from off the shelf solutions to Drupal	One example: £65,000 one off set up cost and £16,000 per year to host and maintain	Likely switch to agency, so Drupal club could be more expensive
One off Drupal build using an agency	Typically £100,000 (some we've spoken to are spending twice that)	£67,000 per council (assuming ⅔ saving made by Croydon)
Ongoing Drupal dev work with in-house team	Typically £175,000 - 200,000 per year	£117,000 per council (assuming ⅔ saving)

61 UK councils use Drupal - assume 51 use agencies, rest in house

51 using agencies = 51 x £67,000 = **£3,417,000**
10 in-house teams = 10 x £117,000 = **£1,170,000**

(over 3-4 year period, typical time to refresh a council website)

Build fewer standalone microsites

“Flat pages” eg landing pages, content pages, site search. Does not include services such as Modern Gov, consultations, planning portal

Typical number of microsites per council	Potential savings
5	10 days design and dev time = £5000 Accessibility testing = £2000 Pen testing = £6000 Typical saving per site = £13,000 Typical saving per council = 5 x £13,000 = £65,000
8 councils in the Alpha = **£520,000** 61 councils using Drupal = **£3,965,000** (over 3-4 year period, typical time to refresh a council website)	

Anecdotally, our improved websites have reduced customer contact, but this is very difficult to measure. The right kind of data is hard to come by, and time to reflect and iterate is often lacking

A better publishing platform is a good start, but it’s not enough. Councils also need to embrace Content Design, and fund teams rather than projects so they can respond to changing user needs

Already, having a forum to share knowledge and skills is making us better builders and publishers. In our interviews with councils, they were interested in this just as much as sharing code

LocalGov Drupal as a Service

We had planned to start this work immediately following the Discovery project. Due to the COVID-19 situation, we will pick this up in detail at Alpha

Recommendations for next steps

There’s a worthwhile Alpha here

- Code and theme refactoring
- Testing Discovery concepts at scale
  - Code flexibility with councils
  - MOU and licensing
  - Collaboration sessions
- Build a “shop window” so others can see features and try them
- Initial investigations into “Local Gov Drupal as a Service”

CROYDON
www.croydon.gov.uk

Brighton & Hove
City Council

Bracknell
Forest
Council

OXFORD
CITY
COUNCIL

Cumbria
County Council

City of Westminster

LONDON
COUNCILS

THE ROYAL BOROUGH OF
KENSINGTON
AND CHELSEA

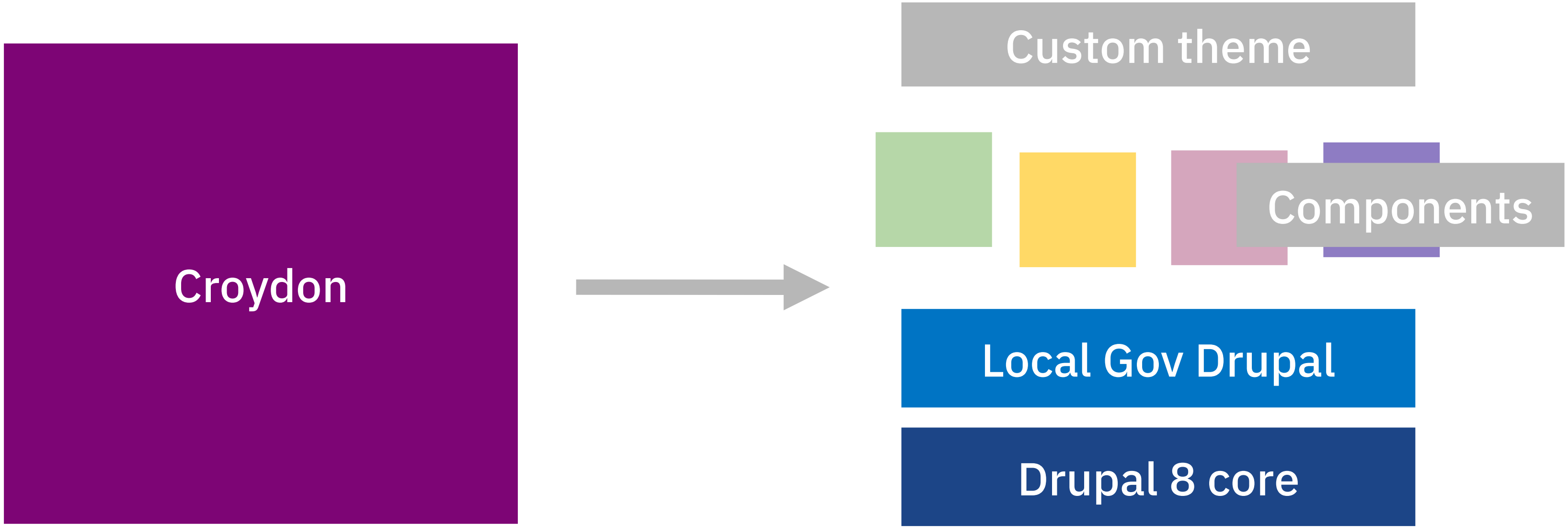
Technical and governance supplier

Research and product supplier

2 sprints (4 weeks)

Code and theme refactoring to make it easier to install, share and adapt

Existing code needs to be broken into components and dependencies removed



Croydon

Custom theme

Components

Local Gov Drupal

Drupal 8 core

4 sprints (8 weeks)

Test the code and theme flexibility, MOU, licensing and collaboration sessions

Councils need different approaches

Refactor existing sites

CROYDON
www.croydon.gov.uk

Brighton & Hove
City Council

All components
Custom theme

Cumbria
County Council

LONDON
COUNCILS

OXFORD
CITY
COUNCIL

THE ROYAL BOROUGH OF
KENSINGTON
AND CHELSEA

City of Westminster

Selected components
Reworked code from site
Custom theme

Bracknell
Forest
Council

Detail here

Separately from Alpha, we’re going to parcel up the Discovery assets, such as the MOU, for others to try and comment on

Project contacts

#localgovdrupal
on Local Digital Slack

@willguv
will.callaghan@croydon.gov.uk

@agilecollective
@dxw