Google

Module 2 | **Lesson 1**

Data modeling with the DBO

Before you get started

This onboarding deck has interactive features and activities that enable a self-guided learning experience. To help you get started, here are two tips for viewing and navigating through the deck.

1 **View this deck in presentation mode.**
- To enter presentation mode, you can either:
  - Click the **Present** or **Slideshow** button in the top-right corner of this page.
  - Press **Ctrl+F5** (Windows), **Cmd+Enter** (macOS), or **Ctrl+Search+5** (Chrome OS) on your keyboard.
- To exit presentation mode, press the **Esc** key on your keyboard.

2 **Navigate by clicking the buttons and links.**
- Click the **Back** or **Next** buttons to go backward or forward in the deck. Moving forward, you'll find them in the bottom corners of every slide.
- Click blue text to go to another slide in this deck or open a new page in your browser.
- For the best learning experience, using your keyboard or mouse wheel to navigate is discouraged.

Ready to get started? Let's go!

Lesson 3

Get ready to data model

What you'll learn about:

- "Good" building configuration files
- Greenfield and brownfield sites
- Recommended data modeling workflow
- Data modeling resources

By the end of this lesson, you'll be able to:

- Reference "good" examples of building configuration files.
- Understand the difference between a greenfield and brownfield site.
- Recall our recommended data modeling workflow.
- Find helpful tooling, documentation, and references for data modeling with the DBO.

Back

Next

Data modeling and the building configuration file

You'll use data modeling to create a **building configuration file**.

What's a building configuration file?

The **building configuration file** is a file that maps real-world devices to the Digital Building Ontology (DBO). Also known as a building config file or simply building config for short, these files are an important part of every digital building project that uses the DBO.

Building configs make a building's data useful and recognizable across any deployment. They encode all of the relevant information about a building and its installed equipment including:

- What equipment exists
- What types they apply
- Which equipment connects to other equipment
- What equipment serves which zone
- And so on

Usually, one building config file contains all the relevant info for one building. You may find you'll need multiple building configs depending on your project's requirements and specific data modeling needs, which is supported.

Example

Here's a look at what a building config file might look like.

```
1   US-MTV-1234:
2     id: FACILITIES/buildings/36166722673234
3     type: FACILITIES/BUILDING
4
5   EF - 3 Restroom / Bldg:
6     cloud_device_id: '2804802894218214135'
7     connections:
8       US-MTV-SB55:
9       - CONTAINS
10    id: CDM/2804808941814135
11    translation:
12      run_command:
13        present_value: data.binary-output_1.present-value
14        states:
15          'OFF': inactive
16          'ON': active
17      run_status:
18        present_value: data.binary-input_1.present-value
19        states:
20          'OFF': inactive
21          'ON': active
22    type: HVAC/FAN_SS
```

Back

Next

Data modeling and the building config file (continued)

You'll use data modeling to create a **building configuration file**.

What's a "good" building configuration file?

A "good" building config is one that's error-free and fully conforms to the DBO. Haphazard data modeling isn't acceptable!

The Digital Building Project GitHub repository provides several samples from "good" building configs from the HVAC namespace including:

- Air handling units (AHU)
- Chilled and condensing water systems (CHWS and CWS)
- Fan coil units (FCU)
- Heating water systems (HWS)

These samples should serve as precedent in your own data modeling work.

Example

Here's a look at what a building config file might look like.

```
1   US-MTV-1234:
2     id: FACILITIES/buildings/36166722673234
3     type: FACILITIES/BUILDING
4
5   EF - 3 Restroom / Bldg:
6     cloud_device_id: '2804802894218214135'
7     connections:
8       US-MTV-SB55:
9         - CONTAINS
10    id: CDM/2804808941814135
11    translation:
12      run_command:
13        present_value: data.binary-output_1.present-value
14        states:
15          'OFF': inactive
16          'ON': active
17      run_status:
18        present_value: data.binary-input_1.present-value
19        states:
20          'OFF': inactive
21          'ON': active
22    type: HVAC/FAN_SS
```

Back

Next

Data modeling and the building config file (continued)

You'll use data modeling to create a **building configuration file**.

Example

Here's a sample of what you might start and end with over the course of your project.

This is an air handling unit (AHU) that serves as an individual zone as a stand-alone device.

The BMS screenshot is something you might reference to construct a building config.

In the building config sample, notice there are several DBO modeling concepts present:

- Connections
- Translation mappings
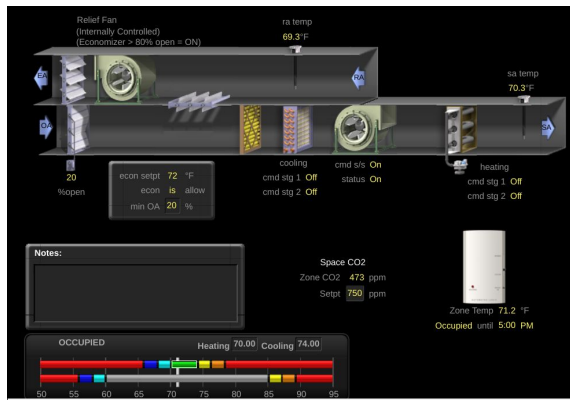- Entity types
- Fields

BMS screenshot

Image source: Google's WebCTRL instances.
WebCTRL is a building automation system owned by Automated Logic.

Building config

```
BLDG-1:
  type: FACILITIES/BUILDING

ZONE-1:
  connections:
    BLDG-1: CONTAINS
    AHU-1: FEEDS
  type: FACILITIES/ZONE

AHU-1:
  cloud device id: 1234
  connections:
    BLDG-1: CONTAINS
  type: HVAC/AHU_DFSS_...
  translation:
    compressor run command :
      present_value: points.comp_cmd.present_value
      units:
        key: pointset.points.comp_cmd.units
        values:
          percent: '%'
    discharge air temperature_sensor :
      present value:
points.supply_air_temperature_sensor.present_value
      units:
        key: pointset.points.supply_air_temperature_sensor.units
        values:
          degrees_celsius: degC
    ...
```

Back

Note: The building config sample is an abridged version of the full code. The … indicates abbreviations in the code. Moving forward, you'll notice most samples in this module are shortened using ….
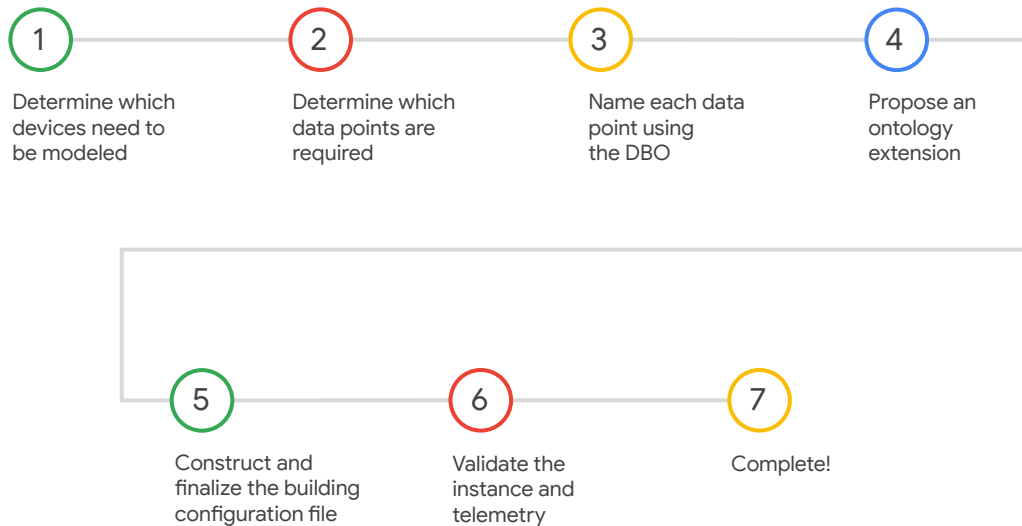
Next

Recommended workflow

Data modeling with the DBO is an art, not a science.

Much of your data modeling efforts will depend on your ability to accurately identify the equipment and systems that need to be modeled and how they can be described using the DBO. However, not everything you are likely to run into is currently described explicitly in DBO, and the building config will need to be tailored specifically to the site you are integrating.

Because of this, we can only recommend a workflow to guide your efforts. In general, data modeling begins with documents containing information, like mechanical schedules or controls submittals, about equipment installed (or being installed) in a building. The end result is a validated building configuration file that encodes this information in a common model.

1 Determine which devices need to be modeled

2 Determine which data points are required

3 Name each data point using the DBO

4 Propose an ontology extension

5 Construct and finalize the building configuration file

**6** Validate the instance and telemetry

7 Complete!

Back

Next

Google

Greenfield and brownfield sites

This module will focus primarily on the recommended workflow for modeling a **greenfield site**.

What's a greenfield site?

A **greenfield site** is a new construction. When modeling one, the equipment and systems are newly installed and made to conform with your project's standards from the outset of construction. However, they will be installed and configured at the same time as when you're constructing the building config file. Care will be needed to make sure these are synchronized properly.
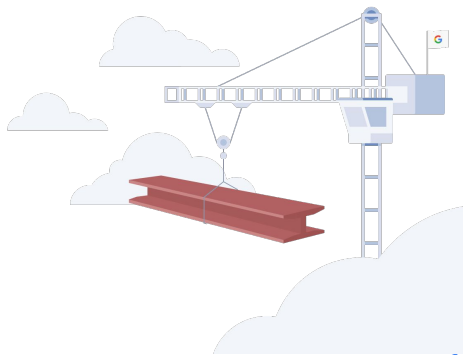
What's a brownfield site?

A **brownfield site** is an existing building. When modeling one, the equipment and systems are pre-determined. Integrating these systems can be straightforward, since all devices and their telemetry is well-understood from the outset. However, it can be challenging to conform everything to your project's standards, because what's in the building may not be common or fully translatable.
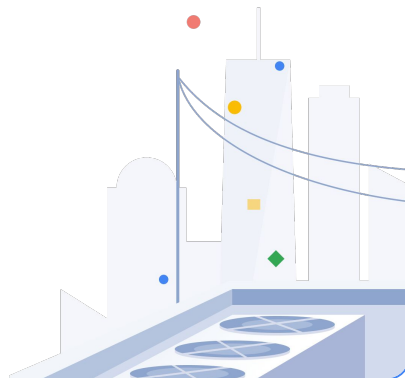
Back

Next

Recommended workflow

Here's the workflow we recommend following:
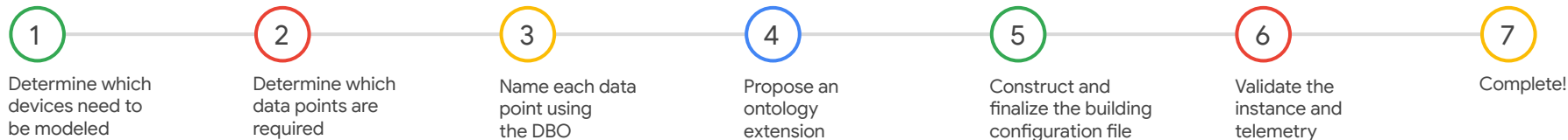*Click on each step in the workflow to reveal a brief description about it.*

① **1** — Determine which devices need to be modeled

② **2** — Determine which data points are required

③ **3** — Name each data point using the DBO

④ **4** — Propose an ontology extension

⑤ **5** — Construct and finalize the building configuration file

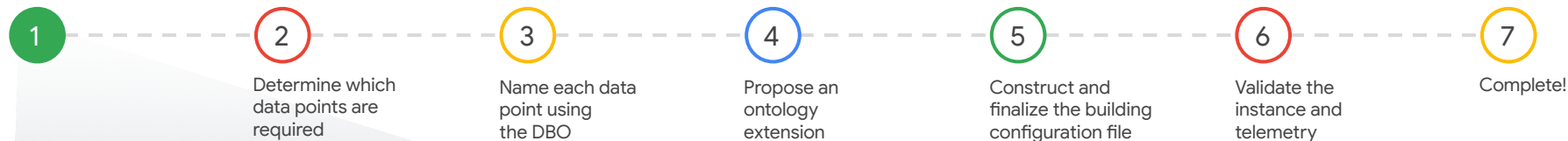⑥ **6** — Validate the instance and telemetry

⑦ **7** — Complete!

Back

Note: Data integration is another important workflow you'll undergo simultaneously with this one. It involves installing and registering UDMI-compliant devices and gateways. You'll learn about the data integration workflow in another module.

Next

Recommended workflow

Here's the workflow we recommend following:
Click on each step in the workflow to reveal a brief description about it.

1 → **2** Determine which data points are required → **3** Name each data point using the DBO → **4** Propose an ontology extension → **5** Construct and finalize the building configuration file → **6** Validate the instance and telemetry → **7** Complete!

Determine which devices need to be modeled

Throughout the process, you'll receive or request project documents containing your scope of work and important information about the equipment and systems installed in the building. From these, you'll identify the reporting devices from each namespace that need to be modeled and integrated into Google Cloud IoT Core.

Back

Note: Lesson 2 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
Click on each step in the workflow to reveal a brief description about it.

1 Determine which devices need to be modeled

2

3 Name each data point using the DBO

4 Propose an ontology extension

5 Construct and finalize the building configuration file

6 Validate the instance and telemetry

7 Complete!

Determine which data points are required

You'll individually examine each logical entity that needs to be modeled using the project documents you received. You'll differentiate between important data points that are required by your project from others that are optional.

Back
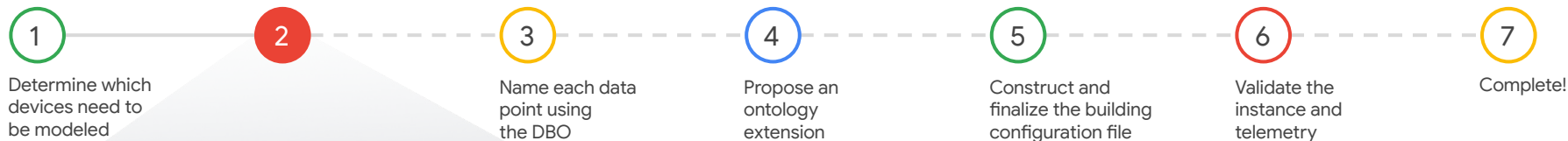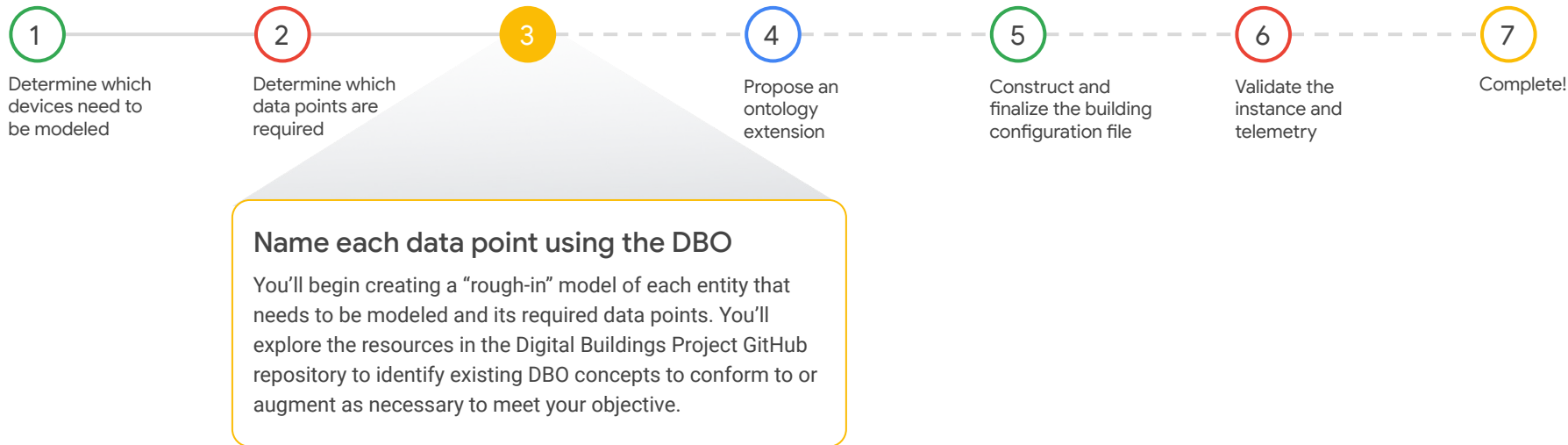
Note: Lesson 3 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
Click on each step in the workflow to reveal a brief description about it.

1 Determine which devices need to be modeled

2 Determine which data points are required

3

4 Propose an ontology extension

5 Construct and finalize the building configuration file

6 Validate the instance and telemetry

7 Complete!

Name each data point using the DBO

You'll begin creating a "rough-in" model of each entity that needs to be modeled and its required data points. You'll explore the resources in the Digital Buildings Project GitHub repository to identify existing DBO concepts to conform to or augment as necessary to meet your objective.

Back

Note: Lesson 4 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
Click on each step in the workflow to reveal a brief description about it.

1 Determine which devices need to be modeled

2 Determine which data points are required

3 Name each data point using the DBO

4

5 Construct and finalize the building configuration file

6 Validate the instance and telemetry

7 Complete!

Propose an ontology extension

You may find that the DBO doesn't have a concept that can describe parts of or all of the entity you're modeling. If this happens, then you'll need to propose an ontology extension to add new models to the DBO.

Back

Note: Lesson 5 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
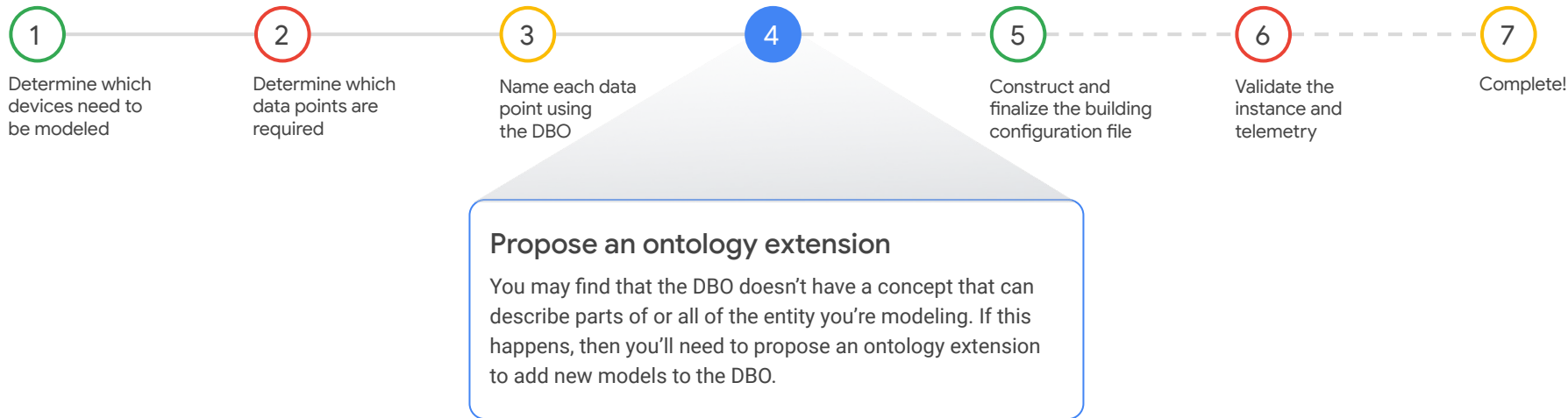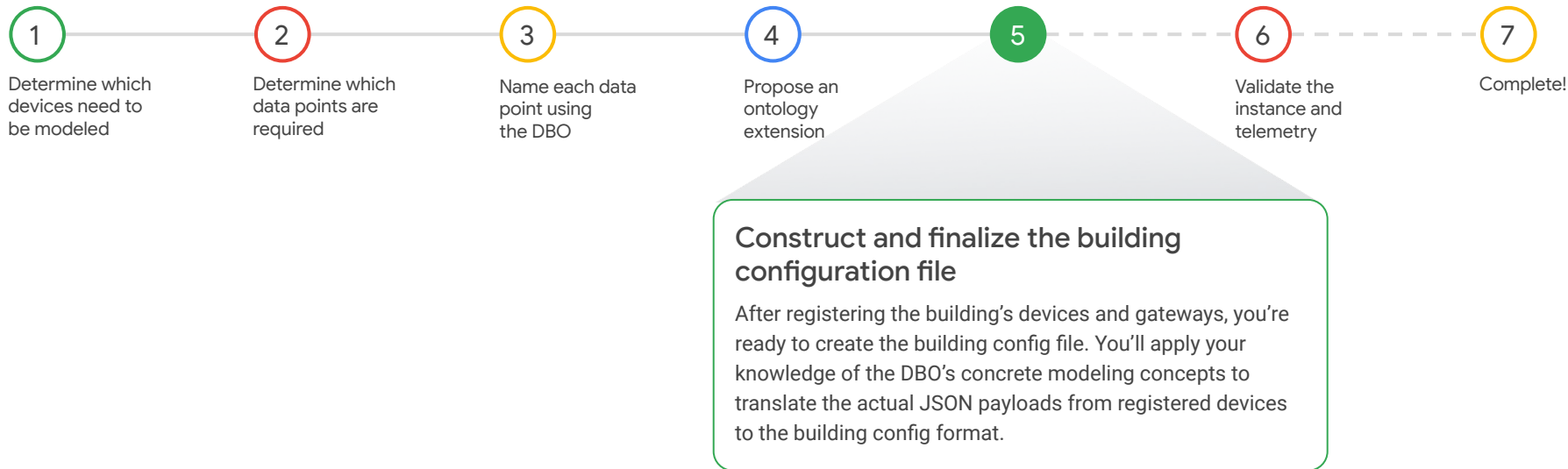*Click on each step in the workflow to reveal a brief description about it.*

1 — Determine which devices need to be modeled

2 — Determine which data points are required

3 — Name each data point using the DBO

4 — Propose an ontology extension

5 —

6 — Validate the instance and telemetry

7 — Complete!

Construct and finalize the building configuration file

After registering the building's devices and gateways, you're ready to create the building config file. You'll apply your knowledge of the DBO's concrete modeling concepts to translate the actual JSON payloads from registered devices to the building config format.

Back

Note: Lesson 6 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
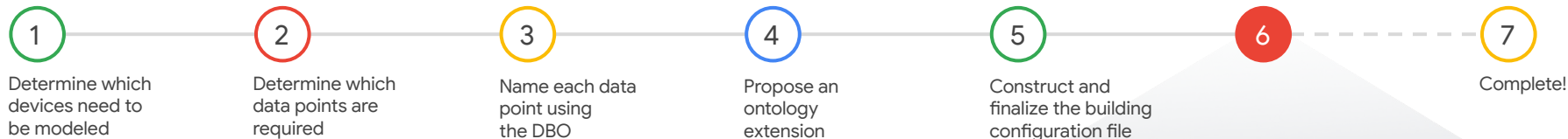*Click on each step in the workflow to reveal a brief description about it.*

1 — Determine which devices need to be modeled

2 — Determine which data points are required

3 — Name each data point using the DBO

4 — Propose an ontology extension

5 — Construct and finalize the building configuration file

6

7 — Complete!

Validate the instance and telemetry

As you construct the building config, it's critical to ensure it fully conforms to the DBO. You'll need to validate the instance and the mapped telemetry frequently before completing it for your project.

Back

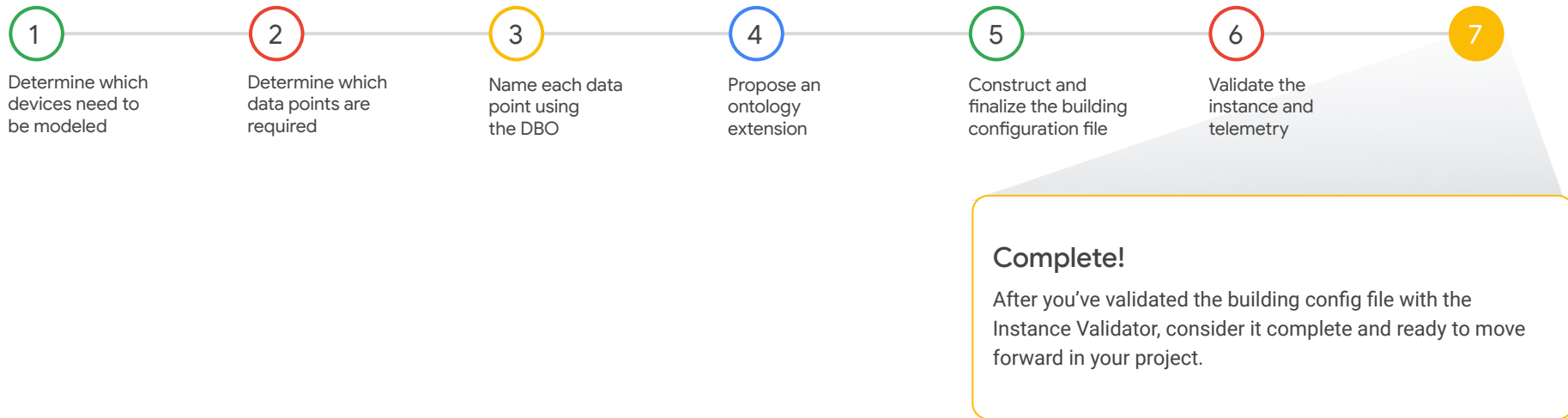**Note:** Lesson 7 of this module will focus on this workflow step.

Next

Recommended workflow

Here's the workflow we recommend following:
*Click on each step in the workflow to reveal a brief description about it.*

1 — Determine which devices need to be modeled

2 — Determine which data points are required

3 — Name each data point using the DBO

4 — Propose an ontology extension

5 — Construct and finalize the building configuration file

6 — Validate the instance and telemetry

7

Complete!

After you've validated the building config file with the Instance Validator, consider it complete and ready to move forward in your project.

Back

Next

Resources

The Digital Buildings Project GitHub repo has all the resources you need for data modeling with the DBO.

Back

Next

Google
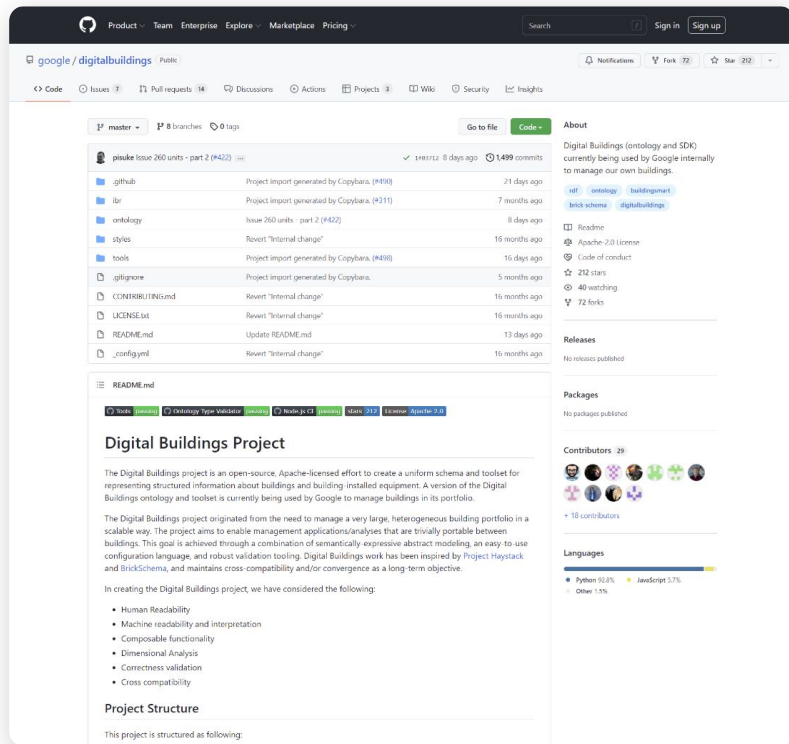
Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
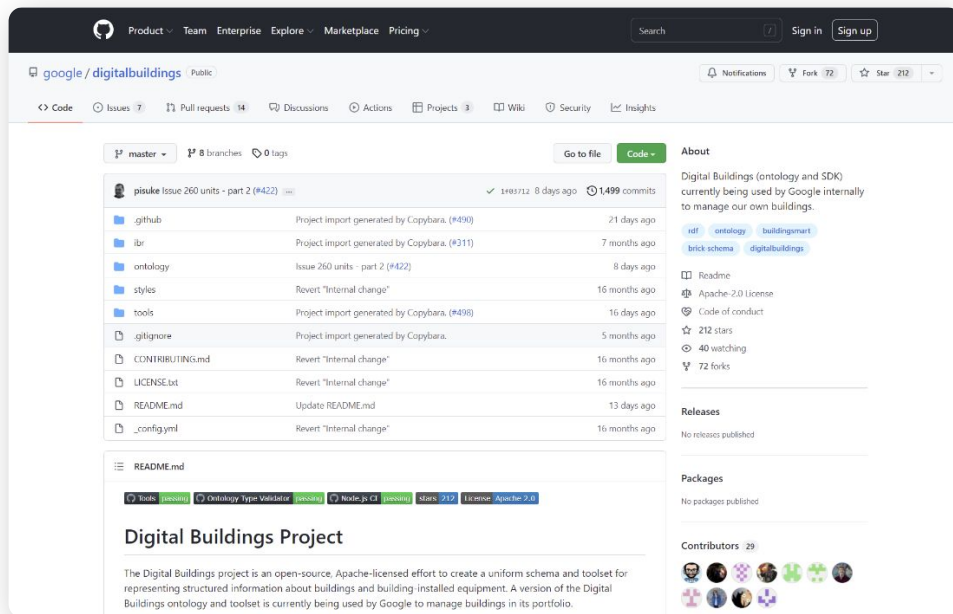*Click on each type of resource to reveal a list of links.*

DBO documentation

Building config samples

Global and child namespaces

Tooling

GitHub resources

Back

Next

Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
Click on each type of resource to reveal a list of links.

DBO documentation

Building config samples

Global and child namespaces

Tooling

GitHub resources

DBO documentation

The following docs can be found in [digitalbuildings / ontology / docs](#):

- [Ontology Overview](#)
  Describes the structure and principles of the Digital Buildings Ontology (DBO).

- [Frequently Asked Questions (FAQ)](#)
  Answers common questions about using the DBO to model building equipment.

- [Abstract Model](#)
  Describes the conventions used in the DBO abstract model.

- [Building Configuration](#)
  Describes the building configuration format for mapping concrete assets.

- [Ontology Configuration](#)
  Explains how to write building configuration files using what's provided in the global and child namespaces.

- [HVAC Model](#)
  Outlines the best practices, general types, and abstract types for data modeling in the HVAC namespace.

Back

Next

Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
Click on each type of resource to reveal a list of links.

DBO documentation

Building config samples

Global and child namespaces

Tooling

GitHub resources

Building config samples

The following samples from the HVAC namespace can be found in [digitalbuildings / ontology / docs](#):

- [Air handling units (AHU)](#)
  Includes single zone and multi-zone AHUs and an AHU as part of a dual duct system.

- [Chilled and condensing water systems (CHWS and CWS)](#)
  Includes an air-cooled and water-cooled variable primary CHWS.

- [Fan coil units (FCU)](#)
  Includes a stand-alone FCU, an FCU and VAV tandem zone, and an FCU with local exhaust control.

- [Heating water systems (HWS)](#)
  Includes a variable primary HWS.

Back

Next

Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
Click on each type of resource to reveal a list of links.

DBO documentation
Building config samples
Global and child namespaces
Tooling
GitHub resources

Global and child namespaces

The DBO namespaces can be found in [digitalbuildings / ontology / yaml / resources](#):

Global namespace

- [Subfields](#)
- [Units](#)
- [Fields](#)
- [States](#)
- [Entity types](#)
- [Connections](#)

Child namespaces

- [Carson](#)
- [Electrical](#)
- [Facilities](#)
- [Gateways](#)
- [HVAC](#)
- [Info Tech](#)
- [Lighting](#)
- [Meters](#)
- [Physical Security](#)
- [Safety](#)

Back

Next

Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
Click on each type of resource to reveal a list of links.

DBO documentation
Building config samples
Global and child namespaces
Tooling
GitHub resources

Tooling

The following tools can be found in [digitalbuildings / tools](#):

- [Ontology Validator](#)
  Used to validate an ontology extension to ensure it conforms to the DBO.

- [Instance Validator](#)
  Used to validate a building configuration file to ensure it conforms to the DBO.

- [Ontology Explorer](#)
  Used to ask basic questions of what's curated within the DBO.

- [GUID Generator](#)
  Used to generate and assign GUIDs to entities in a building config file.

Back

Next

Resources

The [Digital Buildings Project GitHub repo](#) has all the resources you need for data modeling.
Click on each type of resource to reveal a list of links.

DBO documentation

Building config samples

Global and child namespaces

Tooling

GitHub resources

GitHub resources

Need general help with GitHub? Outside of the Digital Buildings repo, [GitHub Docs](#) has a collection of help articles for working and collaborating with GitHub. The following should be useful for DBO data modeling:

- [Fork a repo](#)
  Outlines how to fork (or create a copy of) the Digital Buildings GitHub repo for you to freely experiment and modify without affecting the published version.

- [Contributing to projects](#)
  Includes information about forking, cloning, changes, pull requests, and more to help you get started with GitHub projects like the Digital Buildings Project.
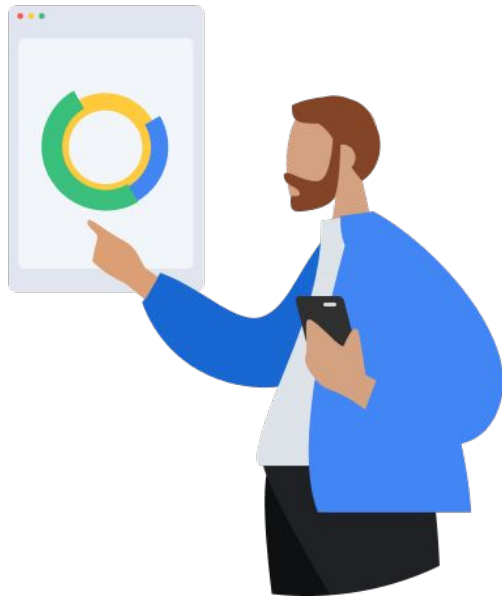
Back

Next

Lesson 1 summary

Let's review what you learned about:

- "Good" building configuration files
- Greenfield and brownfield sites
- Recommended data modeling workflow
- Data modeling resources

Now you should be able to:

- Reference "good" examples of building configuration files.
- Understand the difference between a greenfield and brownfield site.
- Recall the recommended data modeling workflow.
- Find helpful tooling, documentation, and references for DBO data modeling.

Back

Next

You completed Lesson 1!

Now's a great time to take a quick break before starting Lesson 2.

Ready for Lesson 2?

Let's go!

Back

Google

Have questions?

For future reference, keep these contacts and resources easily accessible for technical and procedural questions.

Key contacts
- For DBO questions: Trevor (tsodorff@) or Charbel (charbelk@)
- For UDMI: udmi-discuss@

Helpful resources
Bookmark these resources for future reference.

- Digital Buildings Project GitHub
  Contains source code, tooling, and documentation for the DBO.