Embeddable Widgets Integration Guide

F13745-73
April 2026

Embeddable Widgets Integration Guide,

F13745-73

Contents

1

Oracle Utilities Opower Embeddable Widgets Integration Guide

Welcome to the Oracle Utilities Opower Embeddable Widgets Integration Guide. Use this information to learn how to embed widgets on your utility site. Have a question? Contact Your Delivery Team or visit My Oracle Support.

2

Getting Started with Embedding Widgets

Many Oracle Utilities Opower web products consist of widgets. Widgets are modular pieces of functionality and content that have a responsive layout and flexible configuration options. For example, the Bill Comparison and Data Browser are both widgets.

Most widgets can be embedded directly on a webpage. For example, you may want to embed a widget on a page within your own utility website or on a page within your instance of the Oracle Utilities Opower Digital Self Service standalone web portal. Widgets are embedded using HTML5 custom elements, a technology which allows widgets to be placed alongside other content on a webpage.

This guide provides recommendations on where and how to embed Oracle Utilities Opower widgets on your website. See Guidelines for Embedding Widgets for an overview of the factors to take into account, such as mobile breakpoints, CSS styling, and external links. You can also search or browse for the widget you are interested in to find recommendations specific to that widget.

Your Oracle Utilities Delivery Team is a useful resource in answering questions about embedding widgets and helping you with implementation. Contact Your Delivery Team if you have any questions.

3 Requirements

Single Sign-On: Oracle Utilities supports OpenID Connect-based single sign-on (SSO) for embedded widgets. Your Delivery Team will work with you to set it up.

> **Warning**
>
> Due to an updated handling of third-party cookies by the Safari browser, which is the native browser for iOS devices, SAML has been deprecated as an SSO integration method for embedded deployments of Digital Self Service - Energy Management widgets. Utilities that currently use SAML can work around the issue by asking users to enable cross-site tracking. Oracle Utilities also strongly recommends that utilities migrate to OpenID Connect since SAML is no longer technically supported. Contact Your Delivery Team if you have any questions or need guidance on this change. Standalone deployments of Digital Self Service - Energy Management can still use SAML-based SSO. For more information on configuring SSO, see the Oracle Utilities Opower SSO Configuration Guide.

Embedding Multiple Widgets: For Oracle Utilities Opower products that consist of multiple widgets, you must complete the embedding process for each embeddable component. For more information, see Guidelines for Embedding Widgets and Embedding a Widget Using Web Components.

Query Parameters: Widgets can add query parameters to the URL of the pages on which they are embedded. For example, the Ways to Save widgets pass relevant tip information in the query parameters when customers interact with the widgets. Ensure that your infrastructure does not block or remove these widget-specific query parameters. Query string parameters are automatically detected and passed to the embedded widgets, which means the parameters do not need to be passed in the embedded script tag.

4

Guidelines for Embedding Widgets

This section provides guidance for embedding Oracle Utilities Opower widgets into a utility website or customer portal. It includes both general implementation considerations and recommendations for selecting and arranging widgets to support different customer experiences.

- General Guidelines: Describes cross-widget implementation requirements such as layout constraints, authentication, styling, and linking behavior.
- Widget-Specific Guidelines: Provides recommendations for selecting widgets and designing page layouts, along with links to detailed guidance for individual widgets.

Start with the general guidelines to understand platform requirements, then use the widget-specific guidance to design your widget experience.

General Guidelines

This section provides guidance for embedding Oracle Utilities Opower widgets into a utility website or customer portal. It focuses on *cross-widget* implementation considerations such as layout constraints, authentication, styling, and linking behavior.

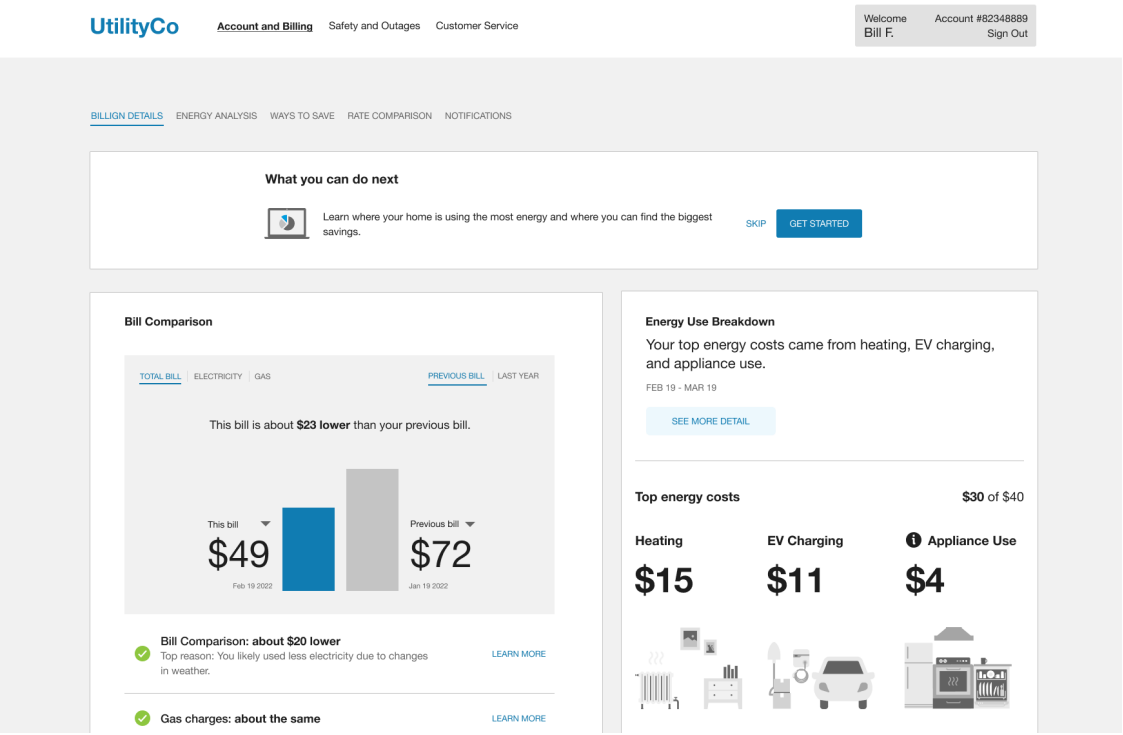
Use these guidelines to ensure that all embedded widgets:

- Display correctly across supported screen sizes and responsive breakpoints
- Load within supported layout and container width requirements
- Are accessible only to authenticated customers when required
- Maintain consistent styling without conflicts from host page CSS
- Support direct navigation from external links when needed

For guidance on selecting specific widgets and designing page layouts for different use cases, see Widget-Specific Guidelines.

Embedded Widgets Example

An example of widgets embedded on a page is shown below. This example includes a Next Best Action banner, a Bill Comparison, and a Home Energy Analysis v2 breakdown. Together these widgets provide a consolidated view of billing details for a customer.



Showing or Hiding Widgets for Different Customers

Not all widgets are applicable to all customers. For example, some widgets are available for residential customers but not non-residential customers. Other widgets can only show data for electricity use and so should not be displayed for gas-only customers. You may therefore want to create a list for yourself that identifies which widgets to show under different conditions. See Determining a Widget's Applicability for details.

Layouts and Minimum Widths

When considering the placement of widgets on your website, it is important to note that Oracle Utilities Opower products only support certain layouts. Widgets respond to the size of the container that includes an embedded widget, and each widget requires a minimum amount of width in each screen layout design. Breakpoint ranges determine when the widget designs transition to a different layout design. All embeddable widgets use the following default breakpoint ranges, described in pixels (px), and have the following minimum width requirements:

Breakpoint Range	Minimum Width
640px and lower	320px
641px - 1024px	600px
1025px and higher	600px

Authentication

Most widgets should be embedded on pages that are accessible only after the customer logs in to their utility account. Embedding widgets on a webpage that is accessible to pre-authenticated customers requires a redirect to fully authenticate customers before they can view the widget content.

Exceptions to this rule include widgets such as the optional Pre-Authenticated - EasyOpen Workflow for Home Energy Analysis and the Confirmation Message. See the descriptions for these widgets for more details.

> **Note**
>
> You can embed more than one widget on a single webpage. See Optimizing Widget Load Performance and Improving the Embedded Widget User Experience with Custom Events for tips on how to support this embedding strategy.

Styling Widgets

Styles from the webpage can affect the widgets that are embedded on it. Oracle Utilities uses the utility branding guidelines to configure styles such as colors which include the primary color, efficient neighbors color, and the you color. Utilities provide colors as part of the utility branding guidelines in the Oracle Utilities Opower Platform Configuration Guide.

Be aware of the following styling considerations related to embeddable widgets:

Use Namespaces for Properties: Use namespaces for properties to avoid overriding to widget styles. For example, it is recommended to use properties in the style of `.text-block {text-align:center}` rather than the more generic `td {align-center}`.

Avoid !important Values: Any styles defined as `!important` can override widget styles and could result in poor user experience.

CSS Isolation

Widgets use web components rather than iframes, which means that the widgets are exposed to the Cascading Style Sheets (CSS) properties applied to the webpages that host the widgets. The shadow DOM capability is employed to prevent CSS conflicts between widgets and their respective webpages. This method isolates the widgets from the webpage CSS when viewed in browsers that support shadow DOM.

For browsers that do not support shadow DOM, widgets employ pseudo-shadow DOM logic which interprets the webpage's CSS and attempts to ignore it for the widget. This method can still encounter CSS conflicts if the top level CSS shares a class name with the widget CSS, and the top level CSS has a specific rule related to that class. If CSS conflicts are encountered, contact your Delivery Team to identify the conflict and determine a resolution.

External Links to Widgets

If you plan to link to embedded widgets from external resources such as emails communications to customers, you can improve the user experience of these links by creating and linking to unique URLs for each widget experience. This technique is applicable when multiple widgets are embedded on the same webpage (URI) and are visually exclusive from other content. A common example of this is when widgets are embedded in different tabs of the same webpage.

Various techniques are available to uniquely identify specific content on a webpage, such as fragment identifiers, custom JavaScript, or single-page application routing. The utility is responsible for configuring and supporting any of these techniques to create unique links for the widget content embedded on their webpages. After the unique links are created, they can be used in customer communications and other external resources attempting to link directly to widget content.

For example, a customer email can use the unique link to link directly to a customer's Bill or Usage Forecast so that they see it immediately upon redirection, rather than linking to the top of the webpage which might not provide an immediate view of the Bill Forecast.

Widget-Specific Guidelines

Utilities can choose which Oracle Utilities Opower widgets to embed based on their customer experience goals and available data. At a minimum, consider including core widgets such as Bill Comparison, Data Browser, Home Energy Analysis, and Ways to Save to provide a comprehensive view of energy usage, insights, and practical tips for saving energy.

Additionally, when designing your widget experience, consider these principles:

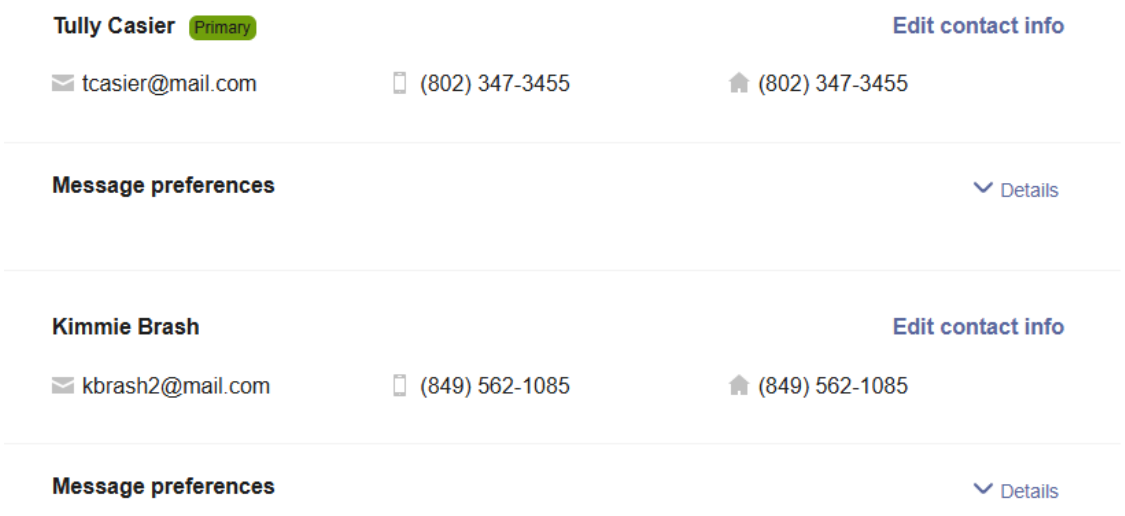
- Group complementary widgets (for example, bill insights and usage analysis) to create a cohesive experience
- Place high-value or summary widgets (such as banners or bill insights) on top-level or dashboard pages
- Use side-by-side layouts for widgets that support comparison or deeper exploration

See individual widget descriptions for detailed guidance, including recommended placement, supported configurations, and customer eligibility.

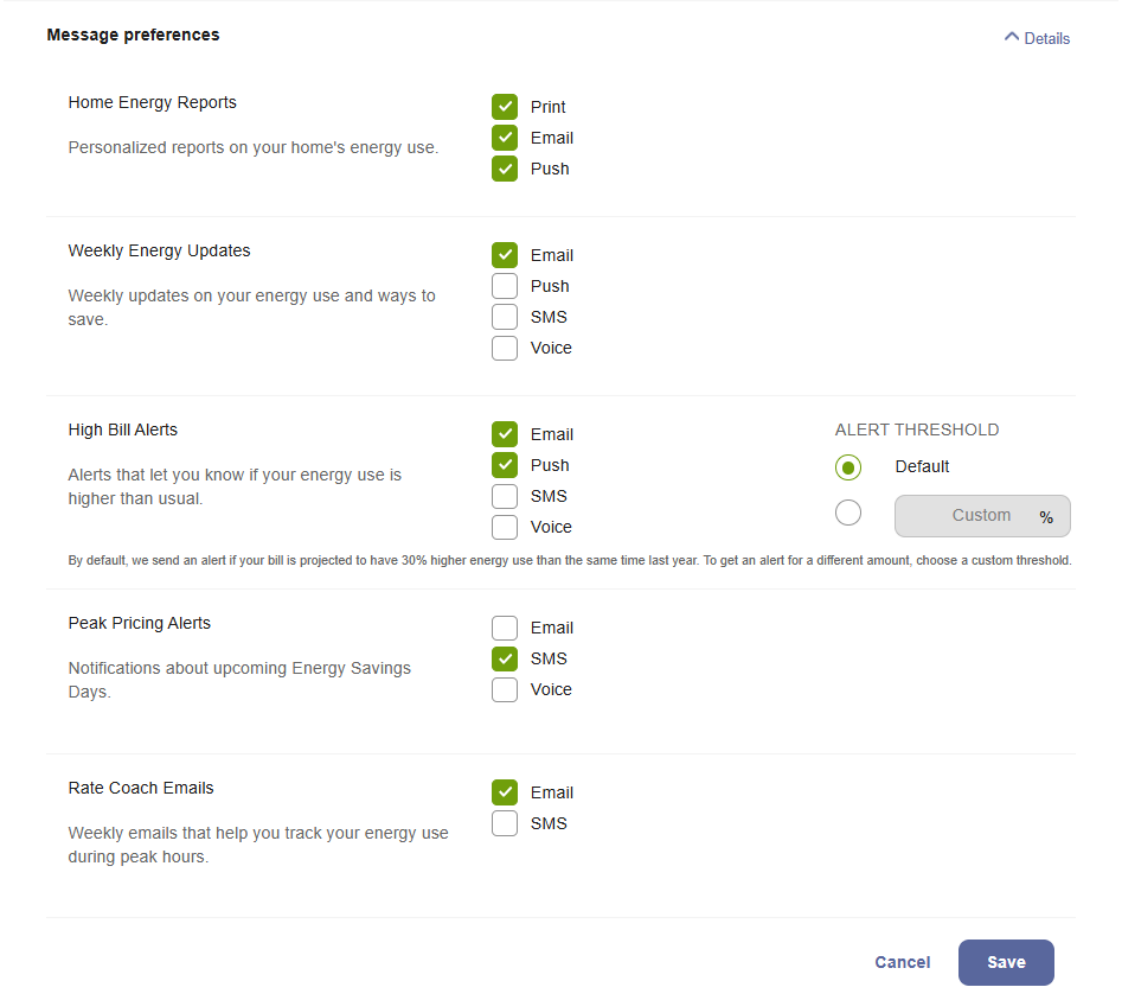
Account Center

The Account Center allows customers to view and change information about their utility account, their alert preferences, and the communications they receive from the utility. Users can edit the primary recipient's information, add additional recipients to the account, edit recipient details, and select alert preferences for each recipient. At a minimum, the Account Center displays information for the primary recipient.

Example: Recipient Information Collapsed



Example: Recipient Information Expanded



Embedding Guidelines

Widget Name: `widget-communication-preferences`

Full Embedding Code: `<opower-widget-communication-preferences opower-instance="widget-communication-preferences"></opower-widget-communication-preferences>`

- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content. Your Delivery Team will work with you to determine the ideal placement.
- Be aware that the number of recipients that a customer creates affects the vertical size of the widget. When more recipients are created, the vertical size of the widget expands. If the widget is embedded on a page with other widgets, then the expansion of the widget might push down any webpage content that is placed below it.

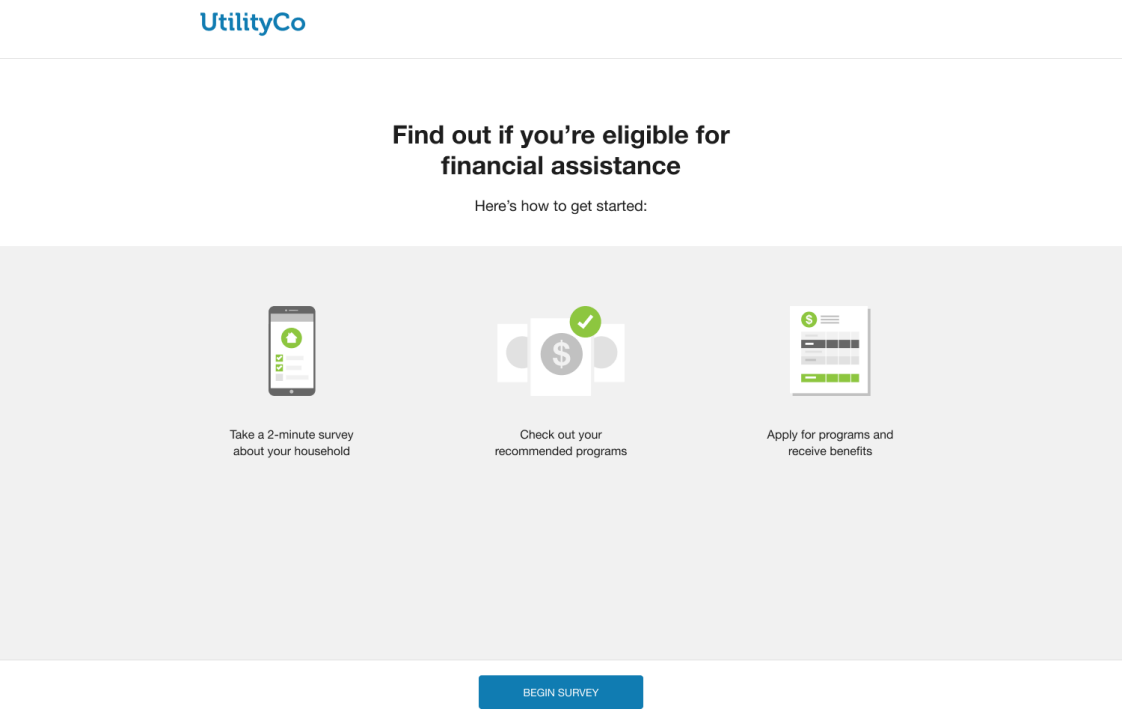
For more information about Account Center, including details about its data requirements and the overall user experience, see the resources below.

- Account Center for residential customers

- Account Center for business customers

Affordability Savings Hub

The Affordability Savings Hub helps customers understand and apply for financial assistance programs at their utility. Customers can use the Savings Hub to learn what programs they are eligible for, and get direct access to enrollment instructions and the tools needed to begin the enrollment process.



Authenticated Workflow

The Affordability Savings Hub survey is designed to be easy to understand and simple to complete. Customers should be able to answer all of the questions they are asked, even if they skip a question.

It is required that you embed the authenticated version of the survey on a separate webpage devoted to hosting the survey questions. This webpage must not include a footer, as well as exclude navigation panels on the sides of the webpage to provide an optimal user experience when completing the survey. Additionally, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-lmi-survey-programs opower-instance="widget-lmi-survey-
programs"></opower-widget-lmi-survey-programs>
```

Pre-Authenticated – EasyOpen Workflow

In addition to the standard authenticated workflow for Affordability Savings Hub, you can allow customers to access the survey without logging in to their utility account.

For example, customers who receive an email communication from Oracle Utilities Opower can follow a link from within the email to begin the survey, which includes a token to

automatically identify the customer. If the customer navigates directly to the survey rather than following the link in their email, they are prompted to provide their billing account number and their full name as it appears on their bill.

To support the embedding of the pre-authenticated workflow, you must duplicate the same embedding strategy described above, but embed the widget on a pre-authenticated page. Additionally, the widget must define an `opower-instance` attribute as part of the embedded tag.

```
<opower-widget-lmi-survey-programs opower-instance="widget-lmi-survey-
programs-easyopen-full"></opower-widget-lmi-survey-programs>
```

Embedding Guidelines

Widget Name: `widget-lmi-survey-programs`

Full Embedding Code: `<opower-widget-lmi-survey-programs opower-instance="widget-lmi-survey-programs"></opower-widget-lmi-survey-programs>`

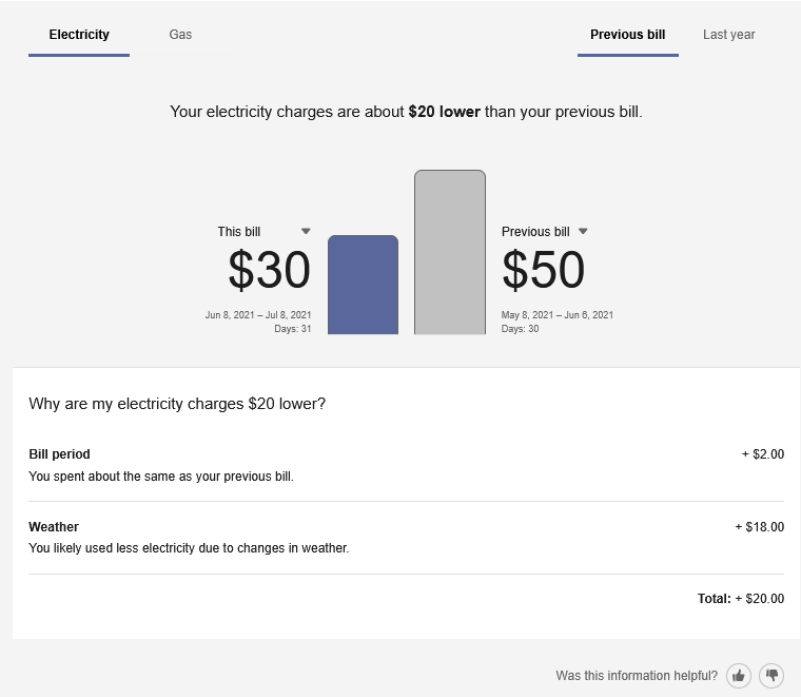
The Affordability Savings Hub consists of multiple sections, but is embedded as a single widget. There are several factors to consider when embedding it:

- The widget is a full-page experience, and so it must be embedded on its own page.
- It is recommended that you place the Savings Hub widget near the Home Energy Analysis widget or the billing section of your website. This is because customers will likely find a logical connection between signing up for an assistance program, learning more about where they use the most energy, and viewing their billing information. Your Delivery Team will work with you to determine the ideal placement.

For more information about Affordability Savings Hub, including details about its data requirements and the overall user experience, see Affordability Savings Hub - Customer Experience.

Bill Comparison

The Bill Comparison allows customers to compare their current bill to their previous bill and to the corresponding bill from the same time period the previous year. A statement indicates whether the customer is spending more, less, or about the same as the compared bill. The feature also highlights factors (such as weather or rate plan changes) that may have contributed to differences between the compared bills.



Embedding Guidelines

Widget Name: `widget-bill-compare-enhanced`

Full Embedding Code: `<opower-widget-bill-compare-enhanced opower-instance="widget-bill-compare-enhanced"></opower-widget-bill-compare-enhanced>`

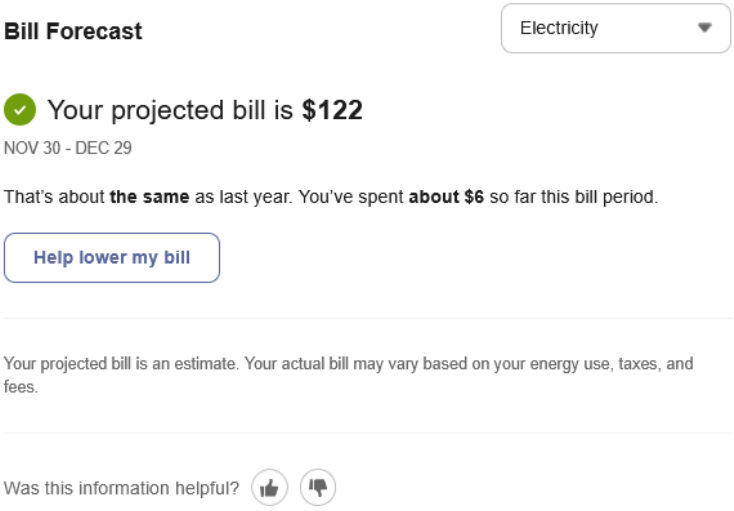
- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.
- Embed this widget directly on highly-visited areas of the utility website such as an account overview dashboard. User research has shown that customers investigating a billing question want to know whether their bill is normal compared to historical energy use. The Bill Comparison widget displays current and most recent bill costs by default, which enables quick validation as to whether their energy use is normal. Secondary links in the widget encourage further exploration through tools such as the Data Browser.

For more information about Bill Comparison, including details about its data requirements and the overall user experience, see the resources below.

- Bill Comparison for residential customers
- Bill Comparison for business customers

Bill or Usage Forecast

The Bill or Usage Forecast shows residential AMI customers their energy use or cost so far in the billing period, projected total energy use or cost for the period, and typical energy use or cost for the period based on their past energy use. The forecast informs customers before the end of the billing cycle if they are likely to have high energy use or cost compared to the same time period the previous year.



Embedding Guidelines

Widget Name: `widget-bill-forecast`

Full Embedding Code: `<opower-widget-bill-forecast opower-instance="widget-bill-forecast"></opower-widget-bill-forecast>`

- Due to its compact size, this widget does not adhere to the default widget breakpoints and minimum pixel widths. It is well-suited for embedding in sections that do not use the full width of the page.
- Embed the Bill Forecast on highly-visited areas of the utility website such as an account overview dashboard or billing pages. Customers that are motivated to take savings actions are directed to Ways to Save.

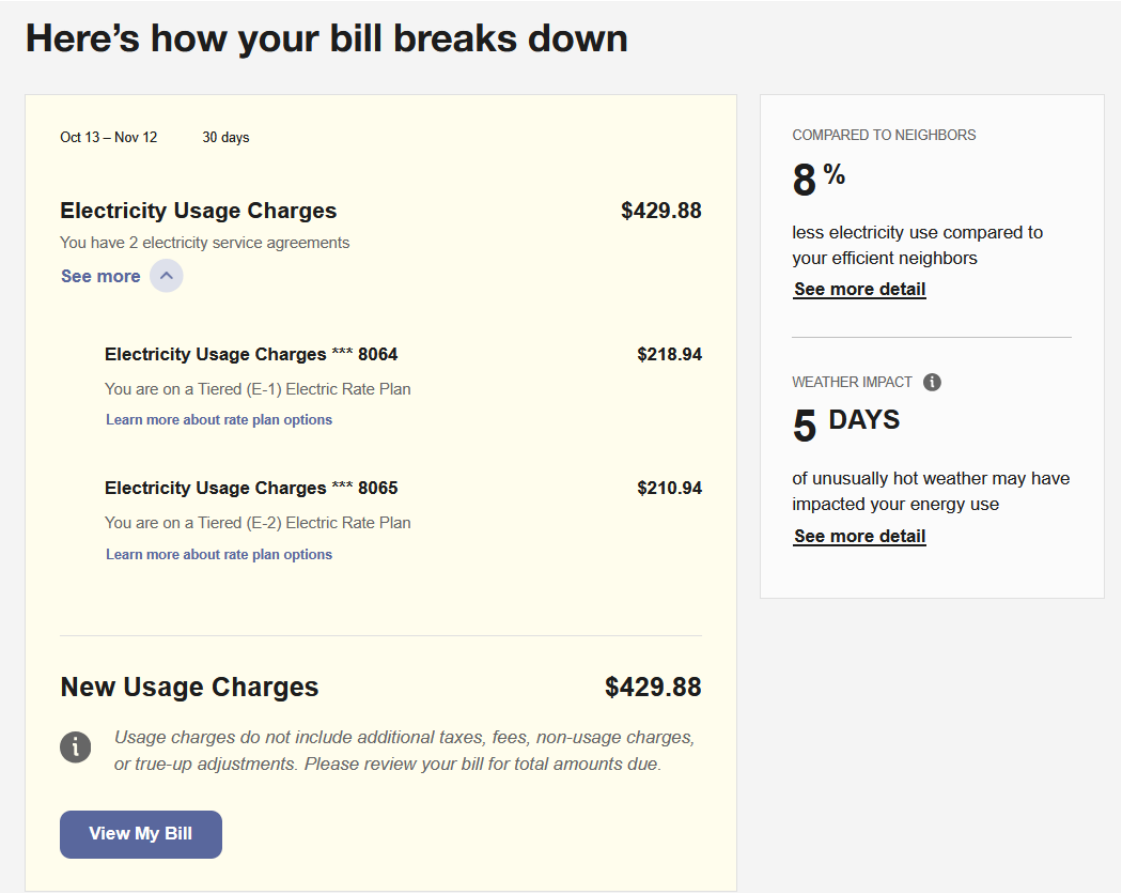
For more information about Bill Forecast, including details about its data requirements and the overall user experience, see the resources below.

- Bill Forecast for residential customers
- Bill Forecast for business customers

Bill Guide

The Bill Guide is a summary of billing insights that helps customers better understand their most recent billed usage charges. It includes a bill breakdown, weather insight, and neighbor comparison insight.

> **Note**
>
> Bill Guide is embedded as a single widget rather than as multiple separate widgets. Utilities must be on the latest technical framework for widgets in order to embed it.



Embedding Guidelines

Widget Name: `widget-bill-guide`

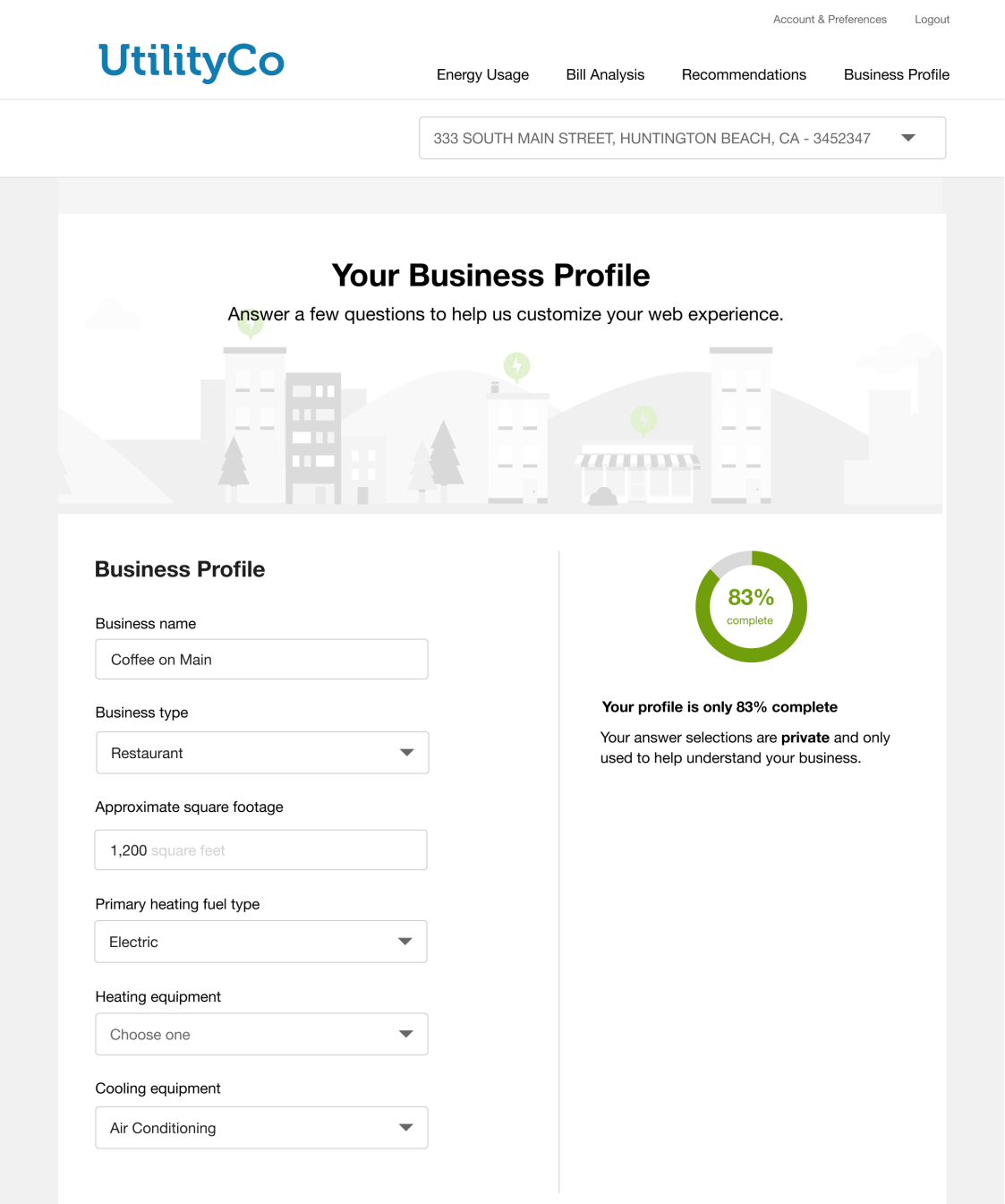
Full Embedding Code: `<opower-widget-bill-guide opower-instance="widget-bill-guide"></opower-widget-bill-guide>`

- Bill Guide is best suited for a prominent location on your website, since it presents a summary of key billing insights. It is recommended that you place it in a primary or top-level section on your website, such as the first among your tabs or pages related to energy efficiency.
- Bill Guide contains components and insights that are suitable for embedding on the same page as other widgets. For example, Bill Guide could be embedded on a dashboard-like page above other widgets or insights related to usage and weather trends.

For more information about Bill Guide, including details about its data requirements and the overall user experience, see the Bill Guide description in the product overview.

Business Profile

The Business Profile allows business customers to enter basic information about their businesses. This information can then be used in other features or widgets to provide valuable energy use insights.



Embedding Guidelines

Widget Name: `widget-business-profile`

Full Embedding Code: `<opower-widget-business-profile opower-instance="widget-business-profile"></opower-widget-business-profile>`

- The Business Profile widget is suitable for locations near any other profile features or account settings sections on your website. Your Delivery Team will work with you to determine the ideal placement. Since it includes a header and two vertical sections of content, it is well suited for using the full width of its own page.
- In addition to the standard authenticated workflow for the Business Profile, you can allow customers to access the profile without logging in to their utility account. For example,

customers who receive an email communication from Oracle Utilities Opower can follow a link from within the email to open the profile, which includes a token to automatically identify the customer. To support this workflow, simply embed the widget on a pre-authenticated page in your website using the widget name above. The same `opower-instance` attribute can be used for authenticated or pre-authenticated workflows. The full embedded tag is:

```
<opower-widget-business-profile opower-instance="widget-business-profile"></opower-widget-business-profile>
```

For more information about the Business Profile, see the Business Profile description in the relevant product overview.

Confirmation Message

The Confirmation Message displays web content to customers who respond to a feedback prompt (such as Customer Feedback) or a question-based module (such as Mini Home Energy Analysis) in an email communication. The content of the Confirmation Message varies depending on which feedback prompt or question is selected. Customers may be thanked for their input, asked a follow up question, or presented with targeted offers that enable deeper personalization in future experiences.

The screenshot below shows an example of a thank-you Confirmation Message that can be displayed to customers.

Embedding Guidelines

Widget Name: `widget-bite-sized`

Full Embedding Code: `<opower-widget-bite-sized opower-instance="widget-bite-sized"></opower-widget-bite-sized>`

- There are different versions of the Confirmation Message. The version that is displayed varies depending on which query parameters are passed in the URL clicked by the customer. Your Delivery Team will work with you to give an overview of the different versions of the widget and the query parameters available. You can also learn more by seeing Confirmation Message in the product overview.
- Ensure that URL query parameters are not removed when a customer arrives to the page where the widget is embedded. The widget uses those parameters to trigger the correct experience and successfully store data as part of the experience.

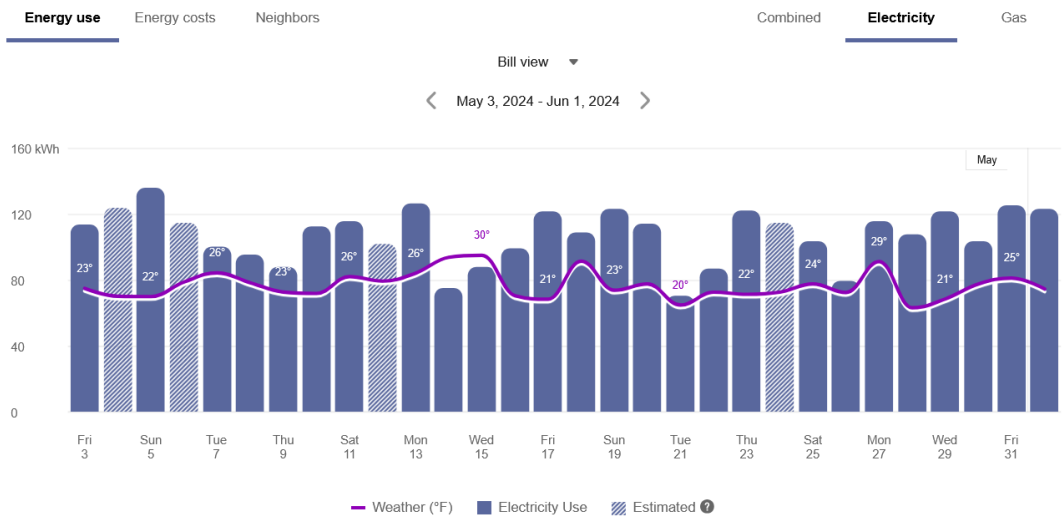
- Embed the widget on a pre-authenticated page so that customers do not need to enter their login credentials to see it. A pre-authenticated link is used to associate the customer's answer with their account and keep a record of the answer.
- The widget is meant to gather responses or feedback from customers, or to prompt them to take a survey or learn more about an offer. This requires them to focus and take action. With this in mind, consider embedding the widget on its own page so that there is little to no other content to distract them.

> **Note**
>
> If you are using widgets that are configured in the Opower Configuration Tool (OCT), then the Confirmation Message cannot be embedded on an Opower-hosted feedback webpage due to certain technical limitations. Contact Your Delivery Team to discuss possible alternatives or workarounds. The widget can still be embedded a utility-hosted webpage.

For more information about Confirmation Message, including details about its data requirements and the overall user experience, see Confirmation Message in the product overview.

Data Browser

The Data Browser is an interactive tool that allows customers to visualize and explore their energy use trends and costs, and make comparisons to useful benchmarks, such as weather and similar homes. One or more views for different kinds of energy or resource use are available in the feature. If applicable, customers can also use menus to switch between multiple accounts or service points.



Embedding Guidelines

Widget Name: `widget-data-browser`

Full Embedding Code: `<opower-widget-data-browser opower-instance="widget-data-browser"></opower-widget-data-browser>`

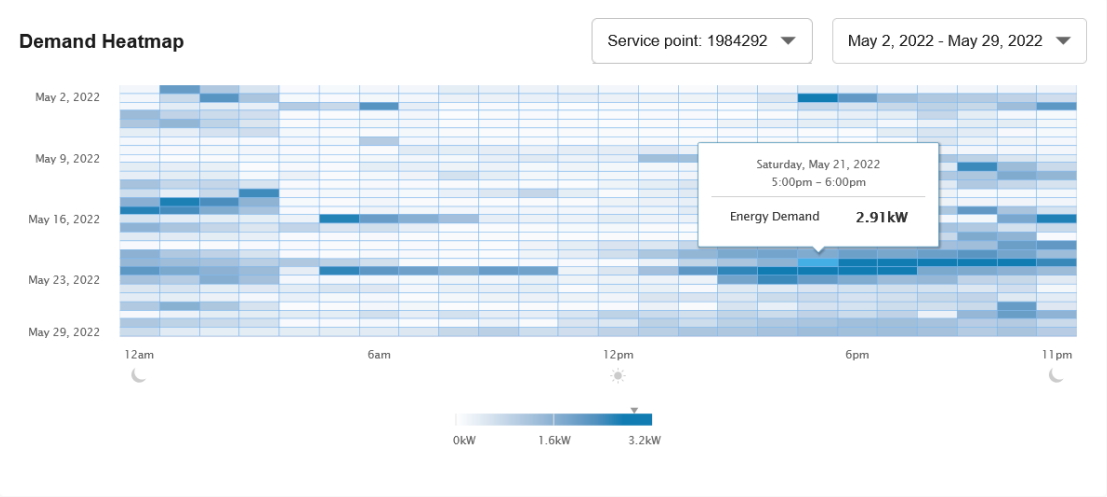
- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.
- Data Browser is intended for customers interested in further exploring their energy use or billing status. Other products and widgets (such as High Bill Alerts AMI, Weekly Energy Updates, Energy Use Overview, Bill or Usage Forecast, and Neighbor Comparison) can send customers directly to the Data Browser to perform this additional analysis. Since many customers are redirected to this widget, prominent utility site location is not as important a factor to drive user interaction with this widget.
- Green Button is commonly included along with the Data Browser, so consider embedding these two widgets on the same page.

For more information about Data Browser, including details about its data requirements and the overall user experience, see the resources below.

- Data Browser for residential customers
- Data Browser for business customers

Demand Heatmap

The Demand Heatmap displays a business customer's demand data in a color-coded grid, and indicates the level of energy demand during specific times, days, and weeks of the year. This information shows at what point in time a business draws the most energy from the grid, allowing business customers to consider how to lower their demand during those times in the future.



Embedding Guidelines

Widget Name: `widget-demand-heatmap`

Full Embedding Code: `<opower-widget-demand-heatmap opower-instance="widget-demand-heatmap"></opower-widget-demand-heatmap>`

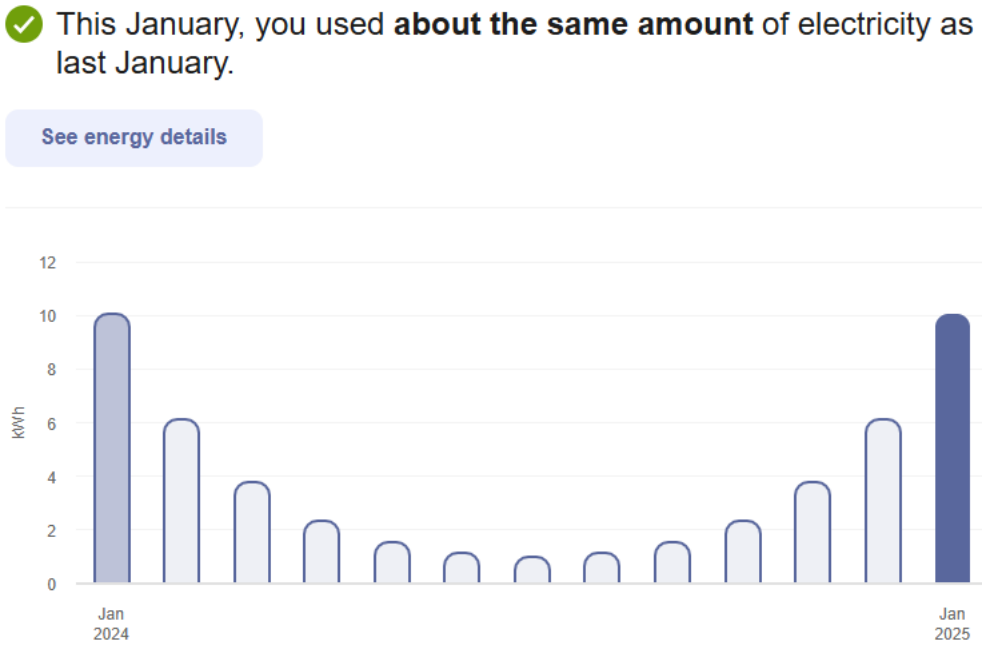
- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.

- Keep in mind that the Demand Heatmap will grow vertically to include more rows of data for customers who have bi-monthly or quarterly bills. This means that any content that appears beneath the widget may be pushed down further for customers who are on these longer billing cycles.

For more information about Demand Heatmap, including details about its data requirements and the overall user experience, see the Demand Heatmap description in the relevant product overview.

Energy Use Overview

The Energy Use Overview displays the most recent bill period's energy use in comparison to that of a bill from the same bill period for the previous year.



Embedding Guidelines

Widget Name: `widget-usage-overview`

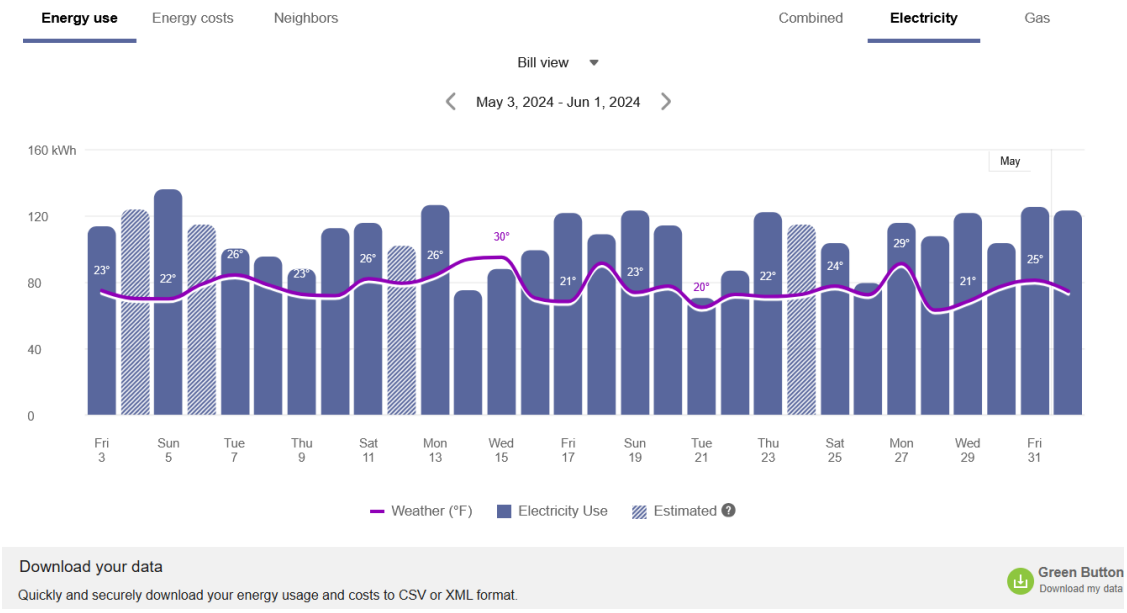
Full Embedding Code: `<opower-widget-usage-overview opower-instance="widget-usage-overview"></opower-widget-usage-overview>`

- This widget acts as a customer's energy dashboard and is well-suited for a primary location on the utility website, such as the initial landing page after the customer has been authenticated. User research and customer interviews have shown that utility customers prefer personal comparison information available with this widget.
- You can also consider embedding this widget on the same page as the Highest Energy Use Days widget. When viewed together, these widgets make it easy for customers to see current use, explore details over time, and recognize use trends and patterns.

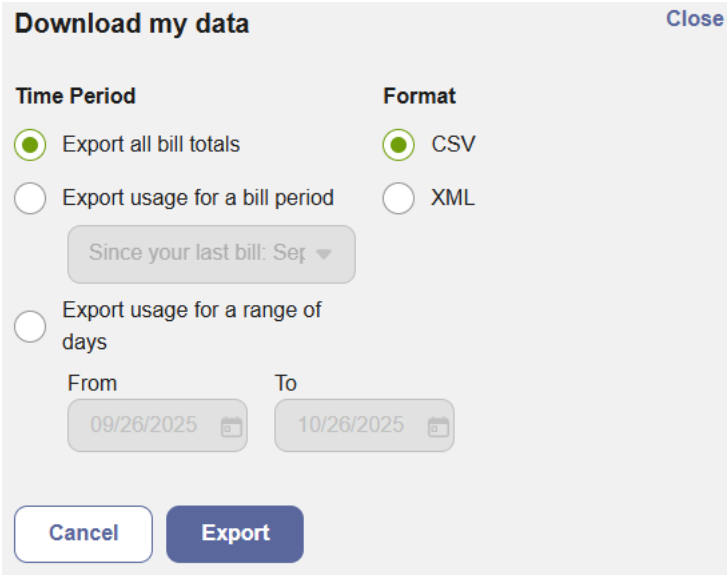
For more information about Energy Use Overview, including details about its data requirements and the overall user experience, see the Energy Use Overview description in the product overview.

Green Button

The Green Button allows customers to export their billing data into CSV or XML format. It is typically located beneath the Data Browser.



When a customer clicks the Green Button link, a **Download my data** section displays and allows the customer to choose between downloading their data in CSV or XML format.



Embedding Guidelines

Widget Name: `widget-usage-export`

Full Embedding Code: `<opower-widget-usage-export opower-instance="widget-usage-export"></opower-widget-usage-export>`

The Green Button is typically included under the Data Browser since it is natural for customers to see a connection between visualizing their data in a graph and then downloading more details about it. Consider embedding these two widgets on the same page when completing the steps in Embedding a Widget.

For more information about Green Button, including details about its data requirements and the overall user experience, see the resources below.

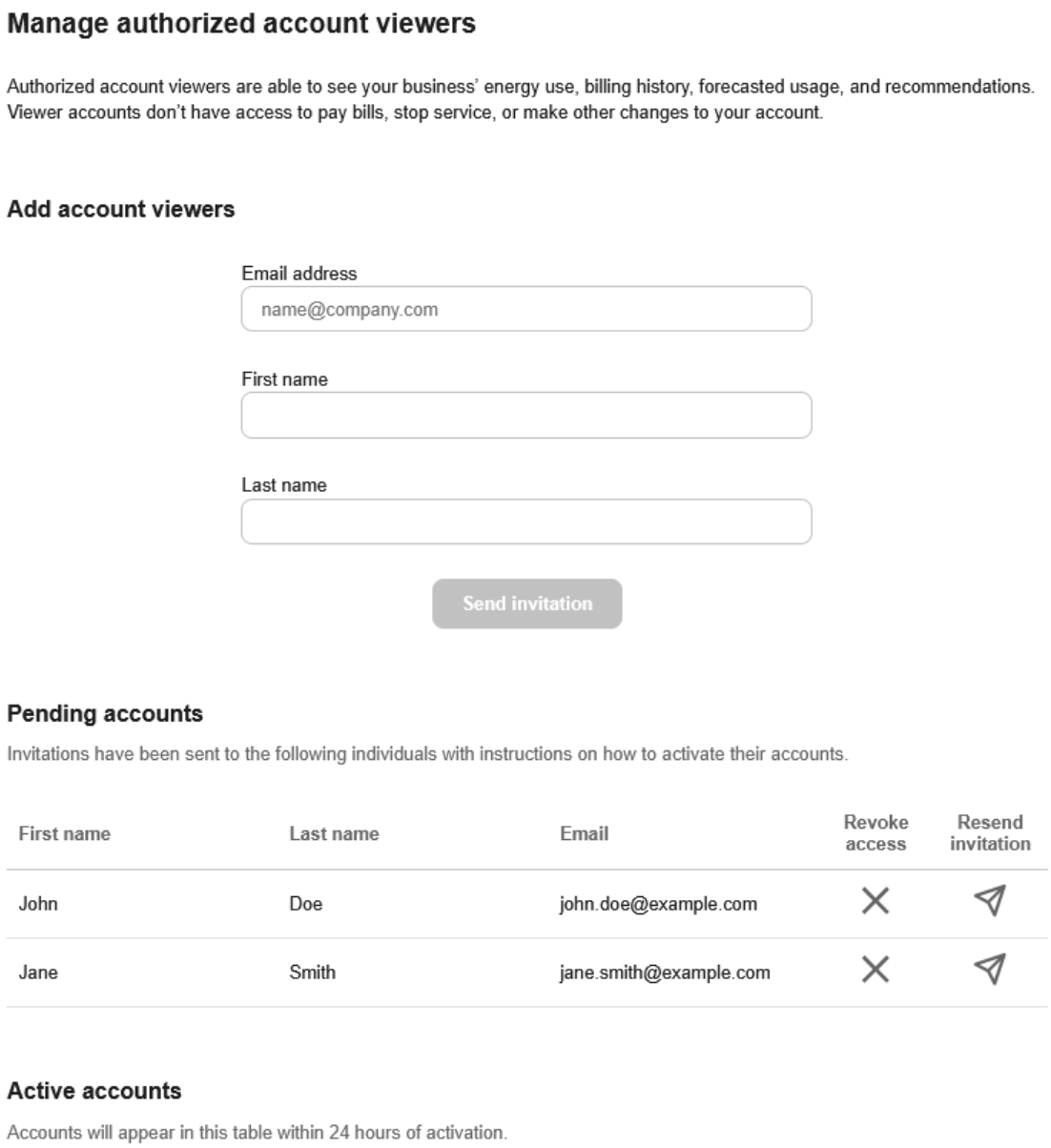
- Green Button for residential customers
- Green Button for business customers

Guest User Access

The Guest User Access feature allows primary utility business account holders to invite guest users to have access to energy insights from their Business Customer Engagement Digital Self Service - Energy Management billing account. This is particularly helpful for staff at large businesses who oversee multiple business locations and who want other users to have access to their energy use trends, insights, and tips on how to save energy.

The Guest User Access feature consists of several interrelated components: a widget to manage authorized account viewers, an email invitation, and a Guest User Portal. This page focuses on the widget used to manage authorized account viewers.

Managed Authorized Viewers Widget Example



Embedding Guidelines

Widget Name: `widget-guest-user-access`

Full Embedding Code: `<opower-widget-guest-user-access opower-instance="widget-guest-user-access"></opower-widget-guest-user-access>`

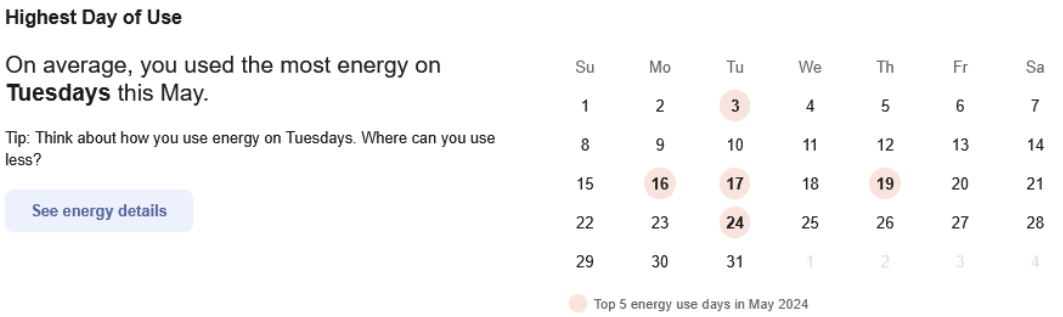
- Since this widget consists of several sections and columns of contact information, it is well suited for using the full width of its own page. (When the widget re-sizes on mobile screens, the first and last name columns are hidden, and the only contact information column that appears is the email column.)
- Since this widget focuses on inviting guest users to view energy usage information, consider placing its page in the same navigation area where other energy usage widgets

are listed, such as the Data Browser, Bill Comparison, or Bill or Usage Forecast. Your Delivery Team will work with you to determine the ideal placement in the navigation.

For more information about Guest User Access, including details about its requirements and the overall user experience, see the Guest User Access description in the relevant product overview.

Highest Energy Use Days

The Highest Energy Use Days calendar highlights the top five days of the month in which a customer used the most energy, helping them identify patterns in their energy use.



Embedding Guidelines

Widget Name: `widget-highest-day-of-use`

Full Embedding Code: `<opower-widget-highest-day-of-use opower-instance="widget-highest-day-of-use"></opower-widget-highest-day-of-use>`

- This widget is well-suited for a primary location on the utility website, such as the initial landing page after the customer has been authenticated. User research and customer interviews have shown that utility customers prefer personal comparison information available with this widget.
- You can also consider embedding this widget on the same page as the Energy Use Overview. When viewed together, these widgets make it easy for customers to see current use, explore details over time, and recognize use trends and patterns.

For more information about Highest Energy Use Day, including details about its data requirements and the overall user experience, see the Highest Energy Use Days description in the product overview.

Home Energy Analysis

The Home Energy Analysis is a visual, interactive survey that prompts customers to answer simple questions about their home attributes and energy habits. This information is used to provide customers with a more detailed breakdown of how they use energy.

There are two versions of the Home Energy Analysis. The way that the widgets are embedded varies slightly depending on which version is used. If you need help identifying which version is applicable to you, Contact Your Delivery Team.

- Home Energy Analysis v1

- Home Energy Analysis v2

Home Energy Analysis v1

The Home Energy Analysis v1 is a visual, interactive tool that prompts customers to answer simple questions about their home attributes and energy habits, and generates a breakdown of energy use in specific categories. The Home Energy Analysis consists of multiple widgets that must be embedded separately.

Pre-Survey

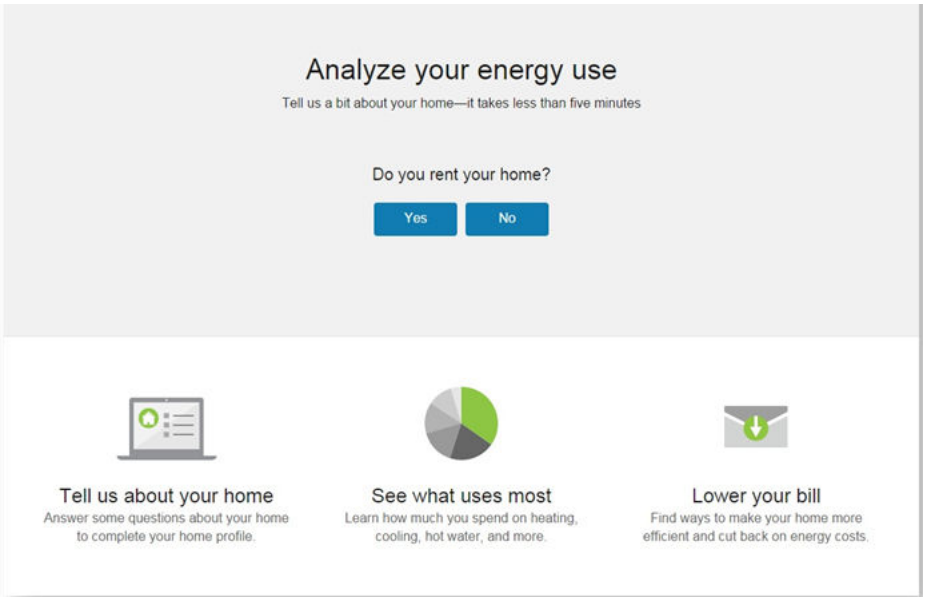
For customers who have not yet started the survey, the Home Energy Analysis displays information about the survey and provides the first question to begin the analysis. This messaging can include information on the estimated time it would take to complete the survey, as well as the benefits of completing the analysis. Customers who select to take the survey are directed to the full survey.

This experience is provided as part of the `widget-survey` widget. It must be placed alongside the post-survey experience (`widget-disaggregation`) to provide a single, authenticated location for customers to interact with the Home Energy Analysis. This pre-survey content is fully hidden for customers after they complete the survey.

To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-survey opower-instance="widget-survey-splash"></opower-widget-survey>
```

> **Note**
>
> If a customer starts but does not complete the survey, progress information and a link to complete the survey is displayed. This experience is provided through the embeddable widget and redirects customers to the survey, which must also be embedded on a separate page.



Survey

The survey is designed to be easy to understand and simple to complete. Customers should be able to answer all of the questions they are asked, even if their answer is to skip the question. It is required to embed the survey, provided with the `widget-survey` widget (the same widget used for the pre-survey experience), on a separate webpage devoted to hosting the survey questions. This webpage must not include a footer, as well as exclude navigation panels on the sides of the webpage to provide an optimal user experience when completing the survey.

To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-survey opower-instance="widget-survey-full"></opower-widget-
survey>
```

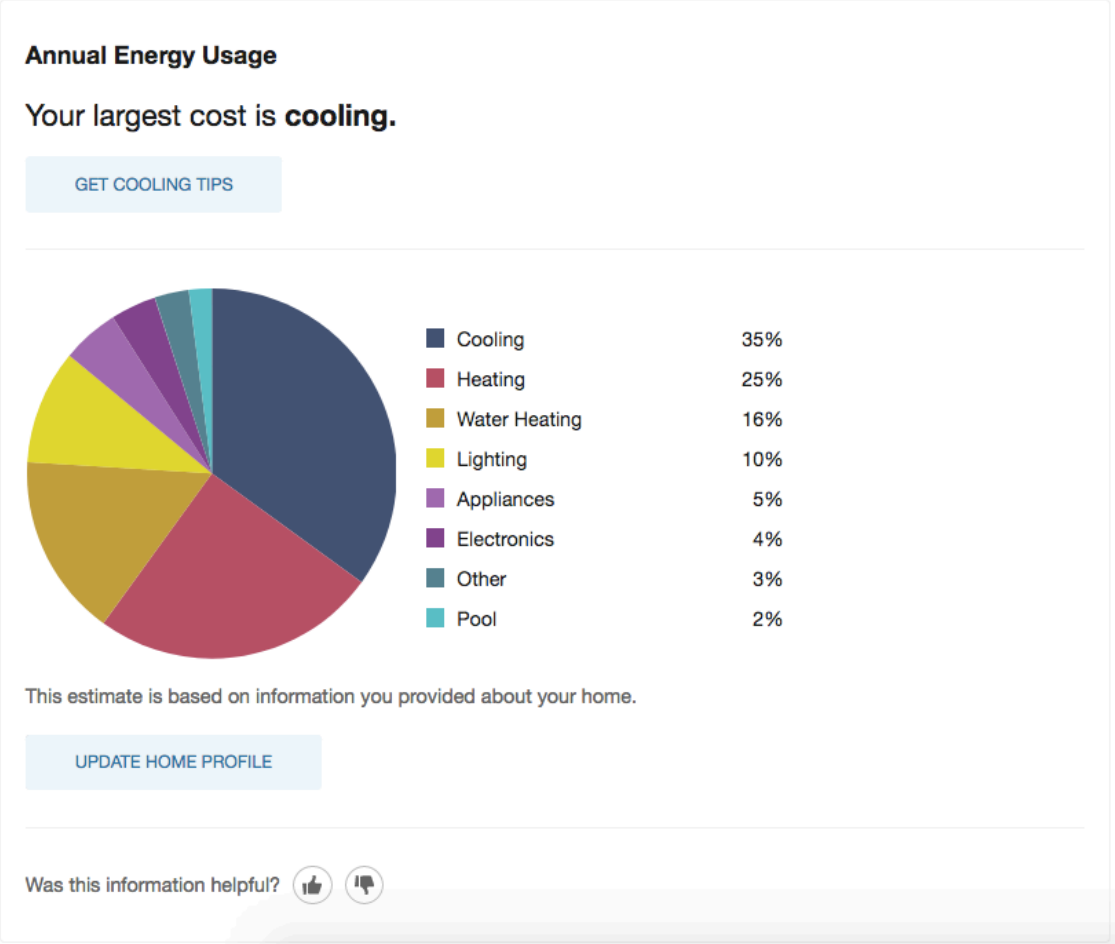
Disaggregation

After the customer answers the last question and finishes the survey, the Home Energy Analysis displays their energy use breakdown based on their responses. Customers can select each individual energy use category to see what percentage of their home's energy use it comprises, category-specific tips, and a list of what contributes to energy use for the category.

This experience is provided as part of the `widget-disaggregation` widget. It must be placed alongside the pre-survey experience to provide a single location for customers to interact with the Home Energy Analysis. This content is fully hidden for customers until they complete the survey. To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-disaggregation opower-instance="widget-disaggregation"></
opower-widget-disaggregation>
```

Embed this widget in proximity to billing information to provide further context and insights into a customer's energy costs.



Pre-Authenticated - EasyOpen Workflow

In addition to the standard authenticated workflow for the Home Energy Analysis, you can allow customers to access the survey without logging in to their utility account. For example, customers of Email Home Energy Reports can follow a link from within their email to begin the survey, which includes a token to automatically identify the customer. If the customer navigates directly to the survey rather than following the email link in their email, they are prompted to provide their billing account number and their full name as it appears on their bill.

To support the embedding of the EasyOpen workflow, you must duplicate the same embedding strategy described above, but embed the widgets on pre-authenticated pages. This includes one pre-authenticated page for the pre-survey and post-survey disaggregation experiences, and one pre-authenticated page for the full survey questions. All other requirements listed above apply for the pre-authenticated versions of the widgets. To use the pre-authenticated experience, the widgets must define an `opower-instance` attribute as part of the embedded tag. The full embedded tags are listed below:

- Pre survey:

```
<opower-widget-survey opower-instance="widget-survey-easyopen-splash"></
opower-widget-survey>
```

- Post-survey disaggregation:

```
<opower-widget-disaggregation opower-instance="widget-disaggregation-
easyopen"></opower-widget-disaggregation>
```

- Full survey questions:

```
<opower-widget-survey opower-instance="widget-survey-easyopen-full"></
opower-widget-survey>
```

Home Energy Analysis v2

The Home Energy Analysis v2 is a visual, interactive tool that prompts customers to answer simple questions about their home attributes and energy habits, and generates a breakdown of their top energy use in specific categories. The Home Energy Analysis consists of multiple widgets that must be embedded separately. Your Delivery Team will work with you to identify the optimal placement.

Pre-Survey

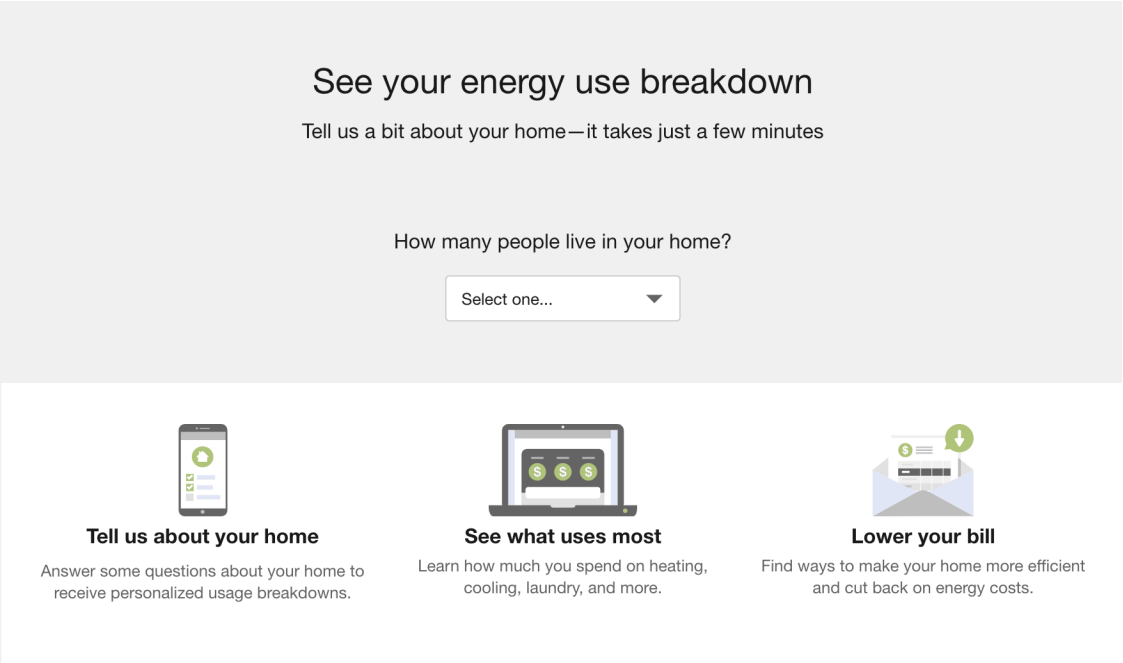
For customers who have not yet started the survey, the Home Energy Analysis displays information about the survey and provides the first question to begin the analysis. This messaging can include information on the estimated time it would take to complete the survey, as well as the benefits of completing the analysis. Customers who select to take the survey are directed to the full survey.

This experience is provided as part of the `widget-survey` widget. It must be placed alongside the disaggregation experience (`widget-usage-categories`) to provide a single, authenticated location for customers to interact with the Home Energy Analysis. This pre-survey content is fully hidden for customers after they complete the survey.

To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-survey opower-instance="widget-survey-splash"></opower-widget-
survey>
```

> **Note**
>
> If a customer starts but does not complete the survey, progress information and a link to complete the survey is displayed. This experience is provided through the embeddable widget and redirects customers to the survey, which must also be embedded on a separate page.



Survey

The survey is designed to be easy to understand and simple to complete. Customers should be able to answer all of the questions they are asked, even if their answer is to skip the question. It is required to embed the survey, provided with the `widget-survey` widget (the same widget used for the pre-survey experience), on a separate webpage devoted to hosting the survey questions. This webpage must not include a footer, as well as exclude navigation panels on the sides of the webpage to provide an optimal user experience when completing the survey.

To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-survey opower-instance="widget-survey-full"></opower-widget-survey>
```

Disaggregation

After the customer answers the last question and finishes the survey, the Home Energy Analysis displays their energy use breakdown based on their responses. Customers can view individual energy use categories to see what percentage of their home's energy use it comprises, find tips for their top three energy use categories, and access more ways to save.

This experience is provided as part of the `widget-usage-categories` widget. It must be placed alongside the pre-survey experience to provide a single location for customers to interact with the Home Energy Analysis. This content is fully hidden for customers until they complete the survey. To support this behavior, the widget must define an `opower-instance` attribute as part of the embedded tag. The full embedded tag definition is as follows:

```
<opower-widget-usage-categories opower-instance="widget-usage-categories"></opower-widget-usage-categories>
```

Embed this widget in proximity to billing information to provide further context and insights into a customer's energy costs.

Your energy use breakdown This bill

Your top energy costs came from water heating, EV charging, and cooling.

MAY 06 - JUN 07

Top energy costs **$100** of $148

Water heating	EV charging	Cooling
$45	$30	$25

Tips for reducing energy use

More energy costs (6)

$48 of $148

Select a category to see relevant tips.

Lighting	$20
Heating	$10
Appliance use	$5
Pool energy use	$5
Electronics	$4
Other energy use	$4

Your always-on use

About 12% of your electricity costs came from always-on energy use—the small amount of power appliances and electronics draw simply because they are plugged in. Learn more.

How do we determine your energy breakdown?

Cost breakdowns are based on your past energy use, smart meter data, and information you provided about your home. Charges and credits applied at the account level are excluded. Minimum bill charges may impact accuracy.

Pre-Authenticated - EasyOpen Workflow

In addition to the standard authenticated workflow for the Home Energy Analysis, you can allow customers to access the survey without logging in to their utility account.

For example, customers of Email Home Energy Reports can follow a link from within their email to begin the survey, which includes a token to automatically identify the customer. If the customer navigates directly to the survey rather than following the link in their email, they are prompted to provide their billing account number and their full name as it appears on their bill.

To support the embedding of the pre-authenticated workflow, you must duplicate the same embedding strategy described above, but embed the widgets on pre-authenticated pages. This includes one pre-authenticated page for the pre-survey and post-survey disaggregation experiences, and one pre-authenticated page for the full survey questions. All other requirements listed above apply for the pre-authenticated versions of the widgets. To use the pre-authenticated experience, the widgets must define an `opower-instance` attribute as part of the embedded tag. The full embedded tags are listed below:

- Pre-survey:

  ```
  <opower-widget-survey opower-instance="widget-survey-easyopen-splash"></
  opower-widget-survey>
  ```

- Post-survey disaggregation:

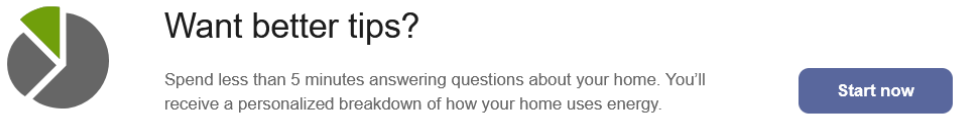
  ```
  <opower-widget-usage-categories opower-instance="widget-usage-categories-
  easyopen"></opower-widget-usage-categories>
  ```

- Full survey questions:

  ```
  <opower-widget-survey opower-instance="widget-survey-easyopen-full"></
  opower-widget-survey>
  ```

Home Energy Analysis Call-to-Action

The Home Energy Analysis Call-to-Action widget encourages customers to complete the Home Energy Analysis in order to receive better tips.



Embedding Guidelines

Widget Name: `widget-hea-cta`

Full Embedding Code: `<opower-widget-hea-cta opower-instance="widget-hea-cta"></opower-widget-hea-cta>`

- It is recommend that you embed this widget at the top of the Ways to Save page so that customers see a natural connection between the tips and the Home Energy Analysis. You can also embed it near other widgets that display tips, such as Tips Light and Tips List.
- Be aware that this widget is hidden for customers who have completed the Home Energy Analysis survey, so it must be embedded in a way that accounts for this behavior.
- You can also consider embedding this widget on an account overview dashboard to use minimal screen space while also directing customers to complete the survey.

Home Energy Analysis Light

The Home Energy Analysis (HEA) Light widget encourages customers to complete the HEA survey if they have not taken it already, and displays an energy use disaggregation for customers who have completed the survey. The widget is designed to be included in the Smart Dashboard or embedded on other utility-hosted webpages to promote the survey and lead customers to the full Home Energy Analysis experience.

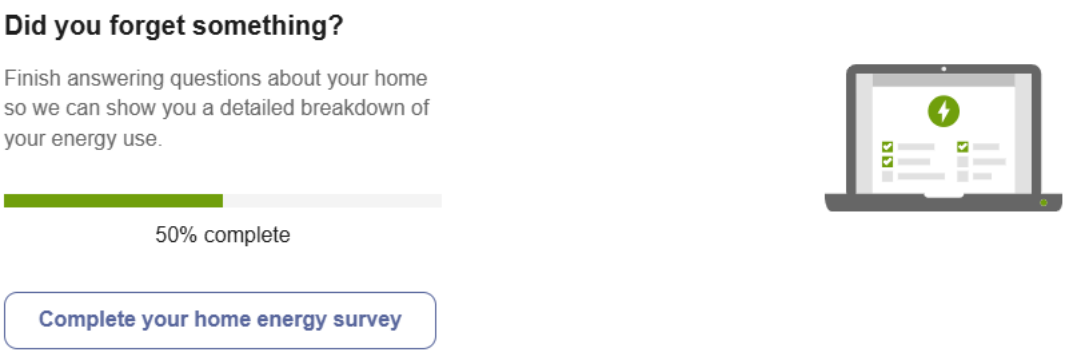
> **Note**
>
> There are two versions of Home Energy Analysis Light—version 1 and version 2—and there are significant differences in the user experience between them. The guidance on this page applies only to version 2 since only version 2 can be embedded. Contact Your Delivery Team if you have questions about which version of the widget you have, and to discuss options for upgrading if necessary.

Survey Prompt

The HEA Light survey prompt is shown if a customer has not started or not completed the survey. The appearance of the prompt varies depending on the survey status. If the customer has not started the survey, the widget displays a prompt to start it. If the customer has started but not completed the survey, the widget shows a progress bar and prompts the customer to complete it.

Example: Survey Not Started

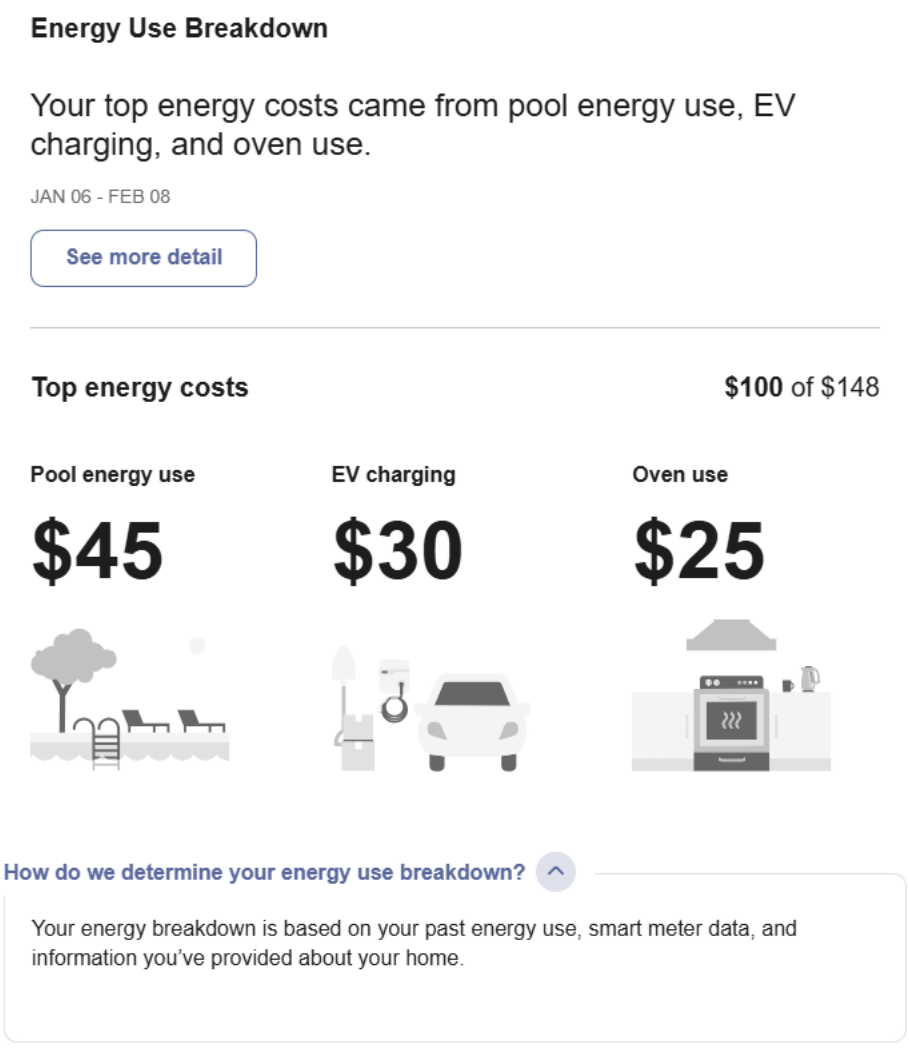
Example: Survey Started, Not Completed



Disaggregation

The HEA Light disaggregation displays a breakdown of the customer's top three categories of energy use. There are two pathways to displaying the disaggregation in the HEA Light widget:

- After the survey is completed, the HEA Light widget is updated to display the disaggregation.
- Before the survey is completed, the HEA Light widget shows a disaggregation so that customers have quicker access to the results. In this case, the survey prompt described above is hidden, and customers can still take the survey to update their answers.



Embedding Guidelines

Widget Name: `widget-hea-cta`

Full Embedding Code: `<opower-widget-hea-cta opower-instance="widget-hea-cta"></opower-widget-hea-cta>`

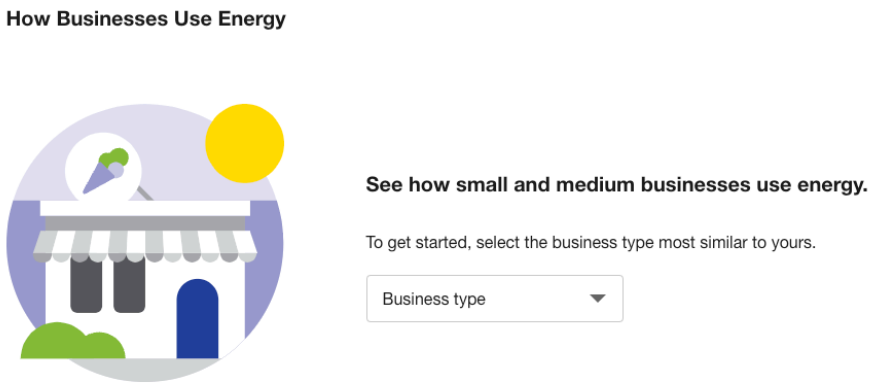
- It is recommend that you embed this widget at the top of the Ways to Save page so that customers see a natural connection between the tips and the Home Energy Analysis. You can also embed it near other widgets that display tips, such as Tips Light and Tips List.
- Be aware that this widget is hidden for customers who have completed the Home Energy Analysis survey, so it must be embedded in a way that accounts for this behavior.
- You can also consider embedding this widget on an account overview dashboard to use minimal screen space while also directing customers to complete the survey.

How Businesses Use Energy

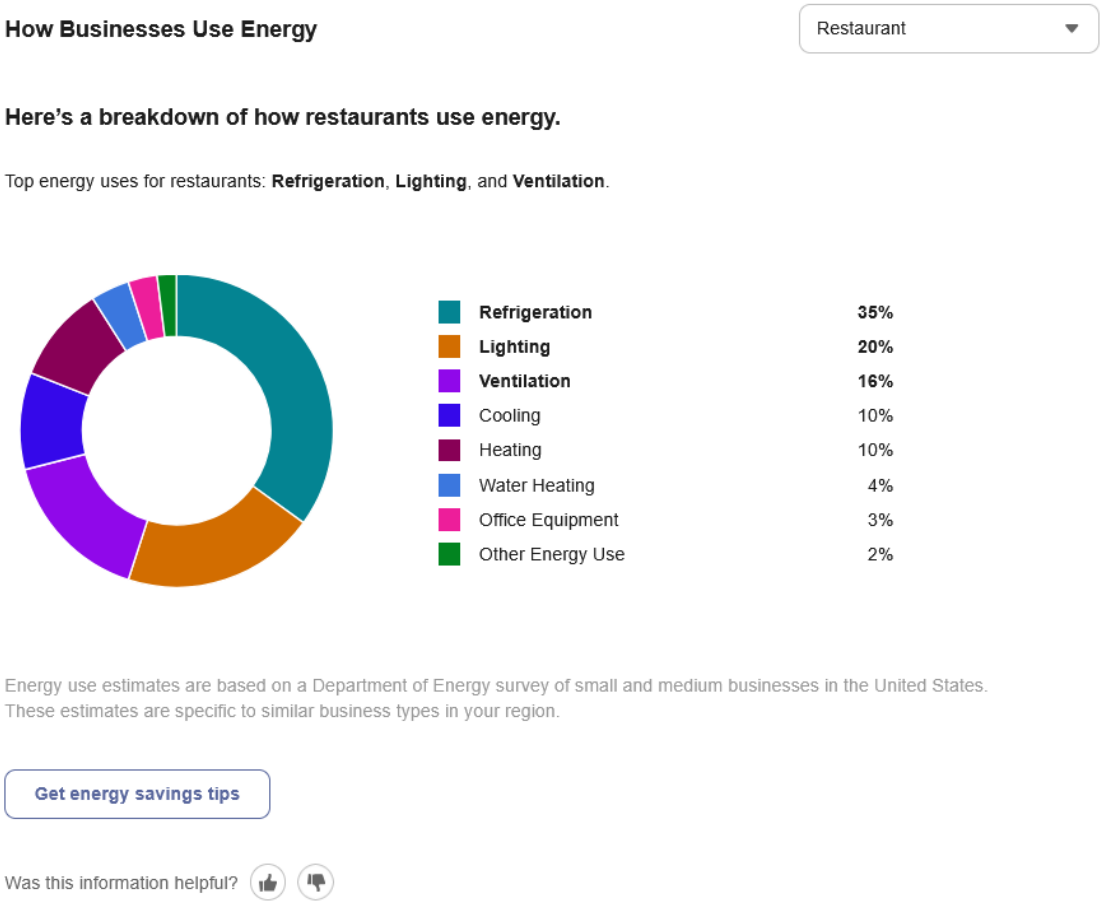
The How Businesses Use Energy feature allows business customers to see an estimated annual breakdown of energy use categories (such as heating, cooling, lighting, refrigeration,

and so on) based on primary business type or activity (such as education, healthcare, retail, or restaurant).

Example of Business Type Selector



Example of Energy Breakdown



Embedding Guidelines

Widget Name: `widget-business-disaggregation`

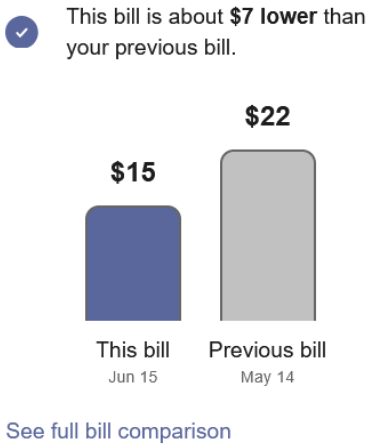
Full Embedding Code: `<opower-widget-business-disaggregation opower-instance="widget-business-disaggregation"></opower-widget-business-disaggregation>`

- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.
- This widget encourages customers to review tips on how to improve their energy efficiency, so consider placing the widget near to the Ways to Save widget.

For more information about this widget, including details about its data requirements and the overall user experience, see the How Businesses Use Energy description in the relevant product overview.

Mini Bill Comparison

The Mini Bill Comparison allows customers to compare their current bill to their previous bill, and to see whether they are spending more, less, or about the same. A link below the comparison directs customers to the full Bill Comparison. The Mini Bill Comparison is intended for use on dashboards or summary pages in a utility website for faster loading and reference.



Embedding Guidelines

Widget Name: `widget-mini-bill-compare`

Full Embedding Code: `<opower-widget-mini-bill-compare opower-instance="widget-mini-bill-compare"></opower-widget-mini-bill-compare>`

- This widget is well-suited for embedding in dashboards alongside other widgets for a unified view of energy and billing information. Consider embedding it near or alongside

other dashboard-friendly information, such as the customer's current balance, transactional tools your dashboard provides, and usage trends that customers can explore.

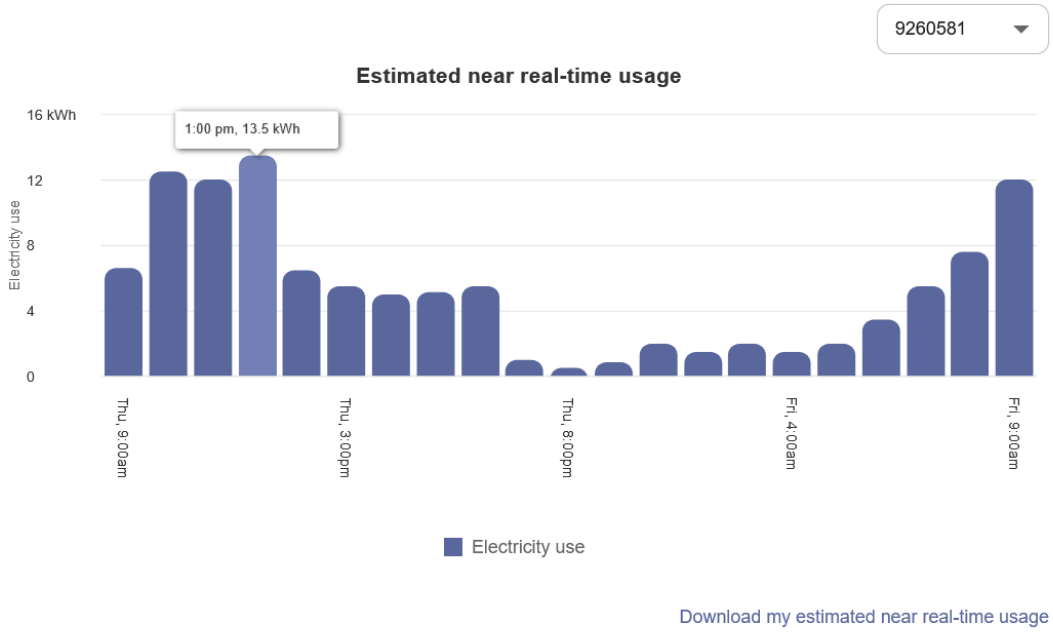
- Embed this widget on highly-visited areas of the utility website such as an account overview dashboard. User research has shown that customers investigating a billing question want to know whether their bill is normal compared to historical energy use. The Mini Bill Comparison widget displays current and most recent bill costs by default, which enables quick validation as to whether their energy use is normal.

For more information about Mini Bill Comparison, including details about its data requirements and the overall user experience, see the resources below.

- Mini Bill Comparison for residential customers

Near Real Time Usage

The Near Real-Time Usage widget displays unvalidated electric interval usage data for customers with AMI (Advanced Metering Infrastructure) meters. The data for this widget is delivered through an API built in partnership between the utility and Oracle Utilities Opower, and is designed to provide users with energy usage data and trends that are nearly current, as opposed to delayed or batch-processed data.



Embedding Guidelines

Widget Name: `widget-real-time-ami`

Full Embedding Code: `<opower-widget-real-time-ami opower-instance="widget-real-time-ami"></opower-widget-real-time-ami>`

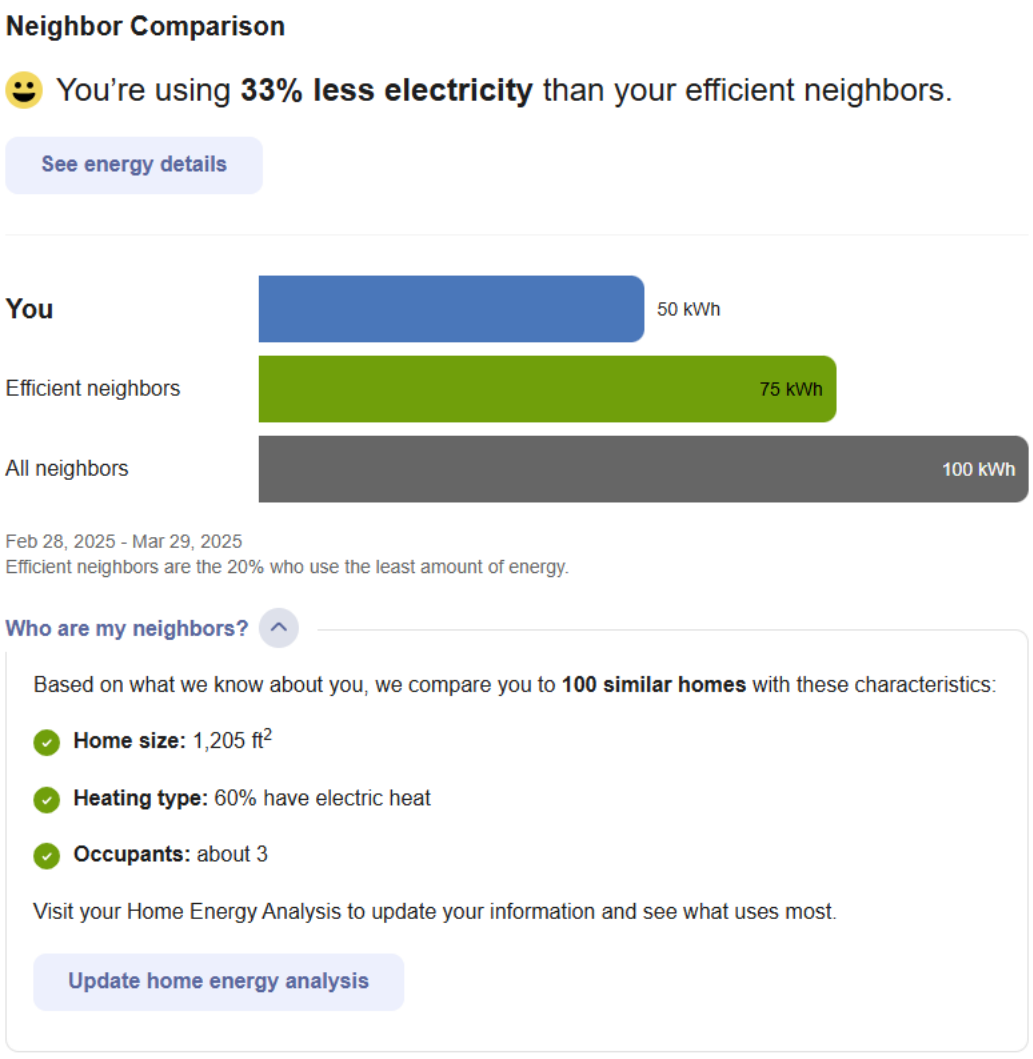
- Since this widget displays energy use trends over time as a bar graph visualization, it is recommended that you place it near the Data Browser widget.

- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.

For more information about the Near Real Time Usage widget, including details about its data requirements and overall user experience, see Near Real Time Usage in the product overview.

Neighbor Comparison

The Neighbor Comparison compares the customer (“You”) to two groups: “Efficient Neighbors” and “All Neighbors.” The results are displayed in a horizontal bar graph, and a message explains how the customer compares to their efficient neighbors. The customer can fall into one of three states: "Great," "Good," or “Using more than average". "Efficient Neighbors" are defined as the most efficient 20% of the customer's neighbors. An informational section below the bar graph provides details about the comparison.



Embedding Guidelines

Widget Name: `widget-neighbor-comparison`

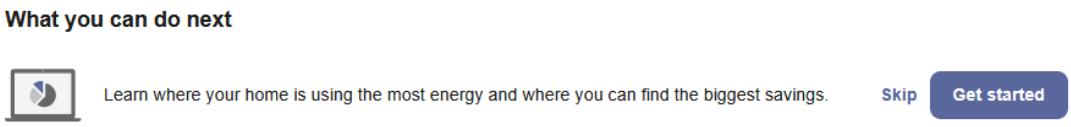
Full Embedding Code: `<opower-widget-neighbor-comparison opower-instance="widget-neighbor-comparison"></opower-widget-neighbor-comparison>`

- The Neighbor Comparison is ideally suited for presenting personalized energy information to highly-visited areas of the utility website, such as an account overview dashboard. Alternatively, it can provide customers further context about their energy use when it is included on pages related to billing information.
- Including the widget on highly-visited areas of the utility website is also beneficial since the Neighbor Comparison encourages customers to explore other insights. For example, customers interested in exploring historical trends in their energy use may be directed to Energy Use Details (also called the Data Browser) or Ways to Save. For customers exploring the "Who are my neighbors?" section, the widget motivates customers to complete the Home Energy Analysis survey.

For more information about Neighbor Comparison, including details about its data requirements and the overall user experience, see the Neighbor Comparison description in the product overview.

Next Best Action

Next Best Action provides dynamic and actionable web banners that can be shared with all customers, or targeted to specific customer groups. The web banners guide customers to the most valuable opportunities or solutions available and helpful tools to help them manage energy use and cost. Customer attributes and segmentation can be employed to determine the best banners to deliver timely announcements, notifications, and actions to each customer.



Embedding Guidelines

Widget Name: `widget-next-best-action`

Full Embedding Code: `<opower-widget-next-best-action opower-instance="widget-next-best-action"></opower-widget-next-best-action>`

- Next Best Action banners can be embedded anywhere on the page that can fit the responsive content. They are typically placed at the top or the bottom of a page.
- Next Best Action banners can be embedded on multiple pages. The embedding process described in Embedding a Widget must be completed for each applicable page.
- Next Best Action banners are targeted to specific customers. If a customer is not in an applicable group, the banner is hidden from view. This dynamic display must be considered when embedding Next Best Action.

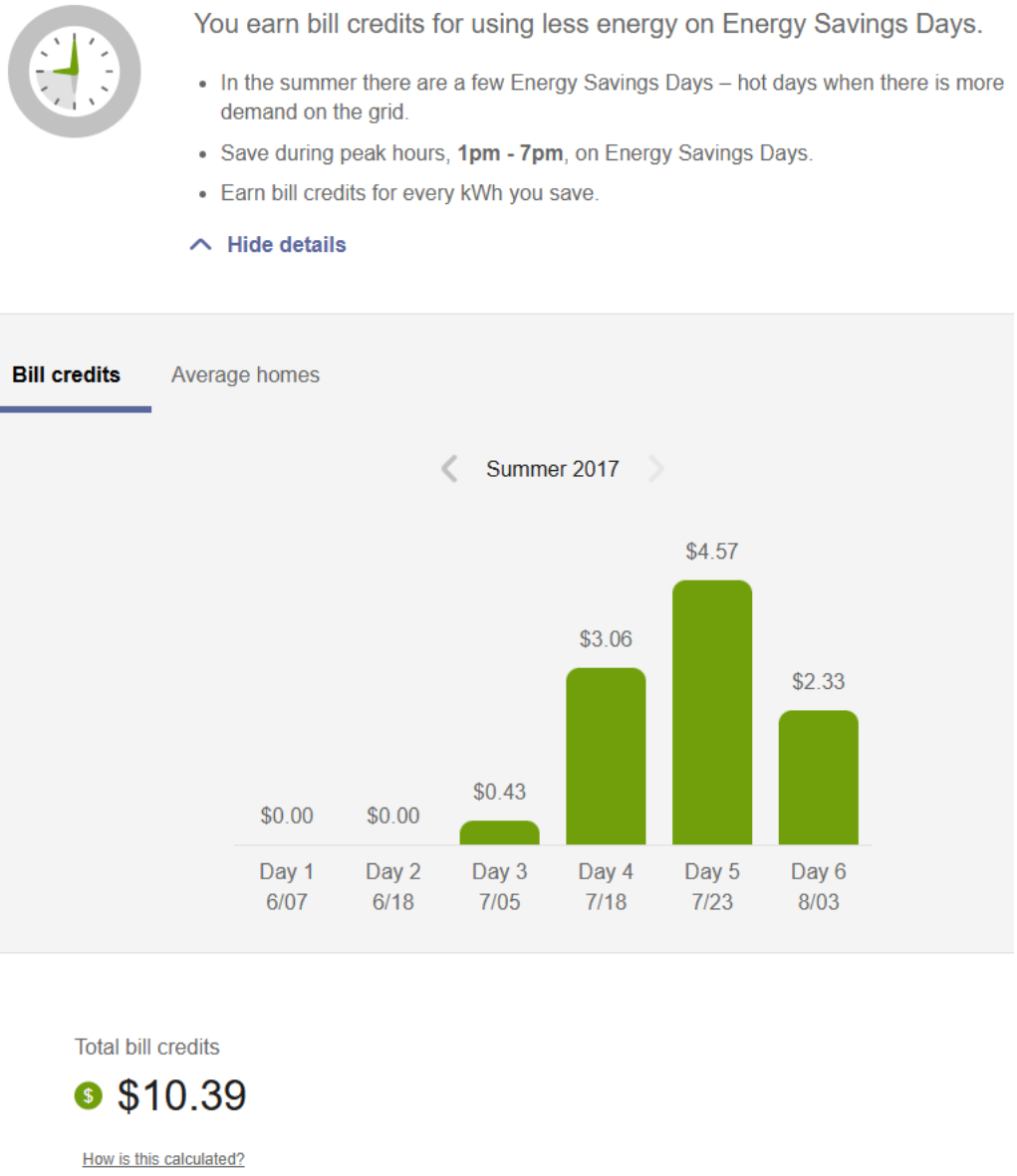
For more information about Next Best Action, including details about its data requirements and the overall user experience, see the resources below.

- Next Best Action for residential customers
- Next Best Action for business customers

Peak Time Rebates

Peak Time Rebates is an energy-savings initiative aimed at encouraging people to reduce energy during peak event days in the summer or winter seasons, thereby lowering energy demands on a large scale. This is accomplished by offering customers monetary credits towards their next bill when they reduce their energy use during peak events.

The Peak Time Rebate widget allows customers who are enrolled in Peak Time Rebates to view their historical peak event day rebate and energy saving information in a user-friendly graph.



Embedding Guidelines

Widget Name: `widget-peak-time-rebate`

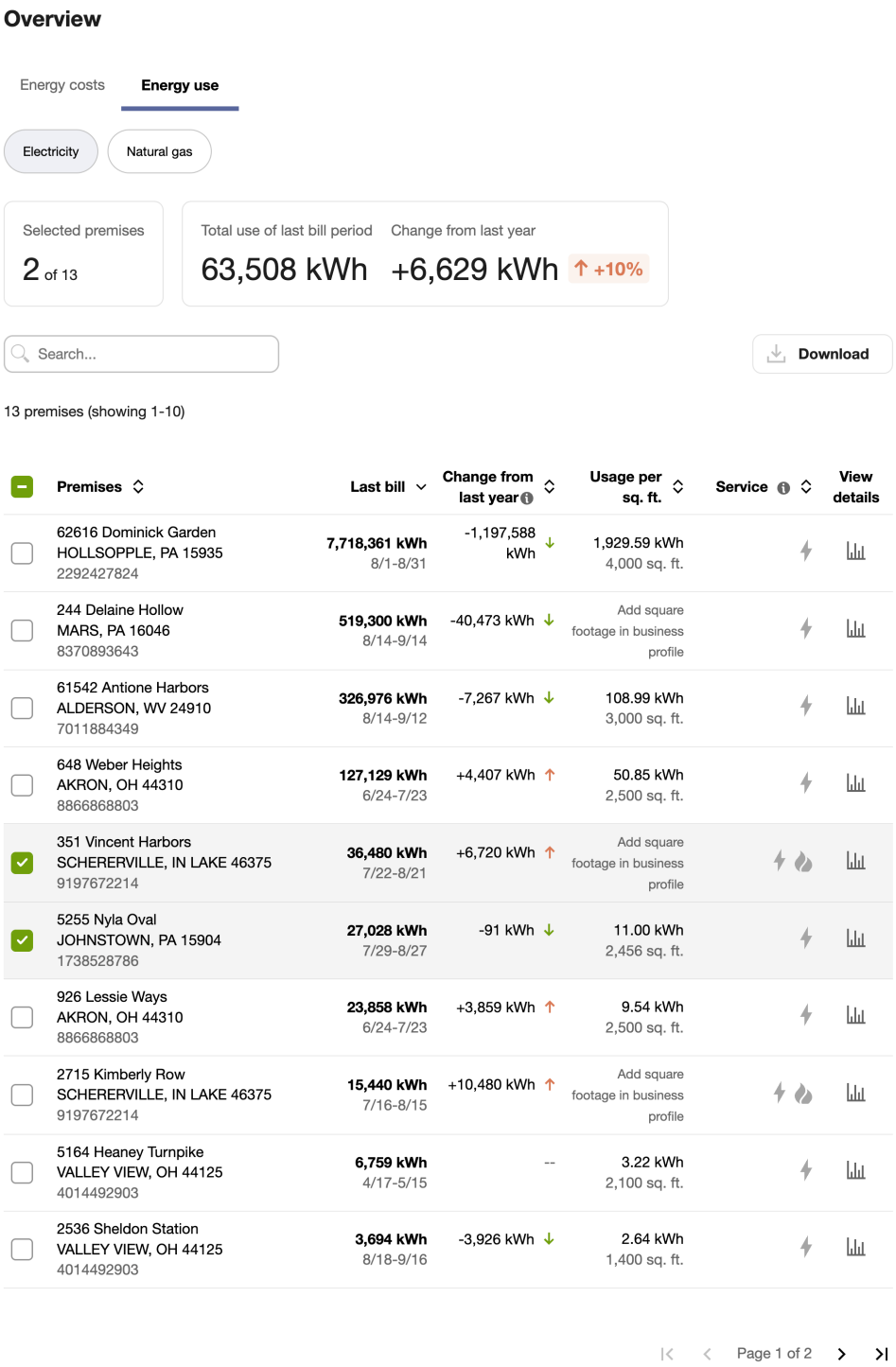
Full Embedding Code: `<opower-widget-peak-time-rebate opower-instance="widget-peak-time-rebate"></opower-widget-peak-time-rebate>`

The Peak Time Rebates widget is intended for customers interested in exploring their energy use patterns. Consider embedding it in a location near the Data Browser or near another area where customers can see their energy use trends or information about any peak time credits they may have earned. Your Delivery Team will work with you to determine the ideal placement.

For more information about the Peak Time Rebates web widget, including details about its data requirements and the overall user experience, see the Peak Time Rebates description in the product overview.

Portfolio View

The Portfolio View allows business customers to see aggregated energy use and cost information across multiple premises. Business customers can use this information to quickly determine if any of their premises need attention.



Embedding Guidelines

Widget Name: `widget-portfolio-view`

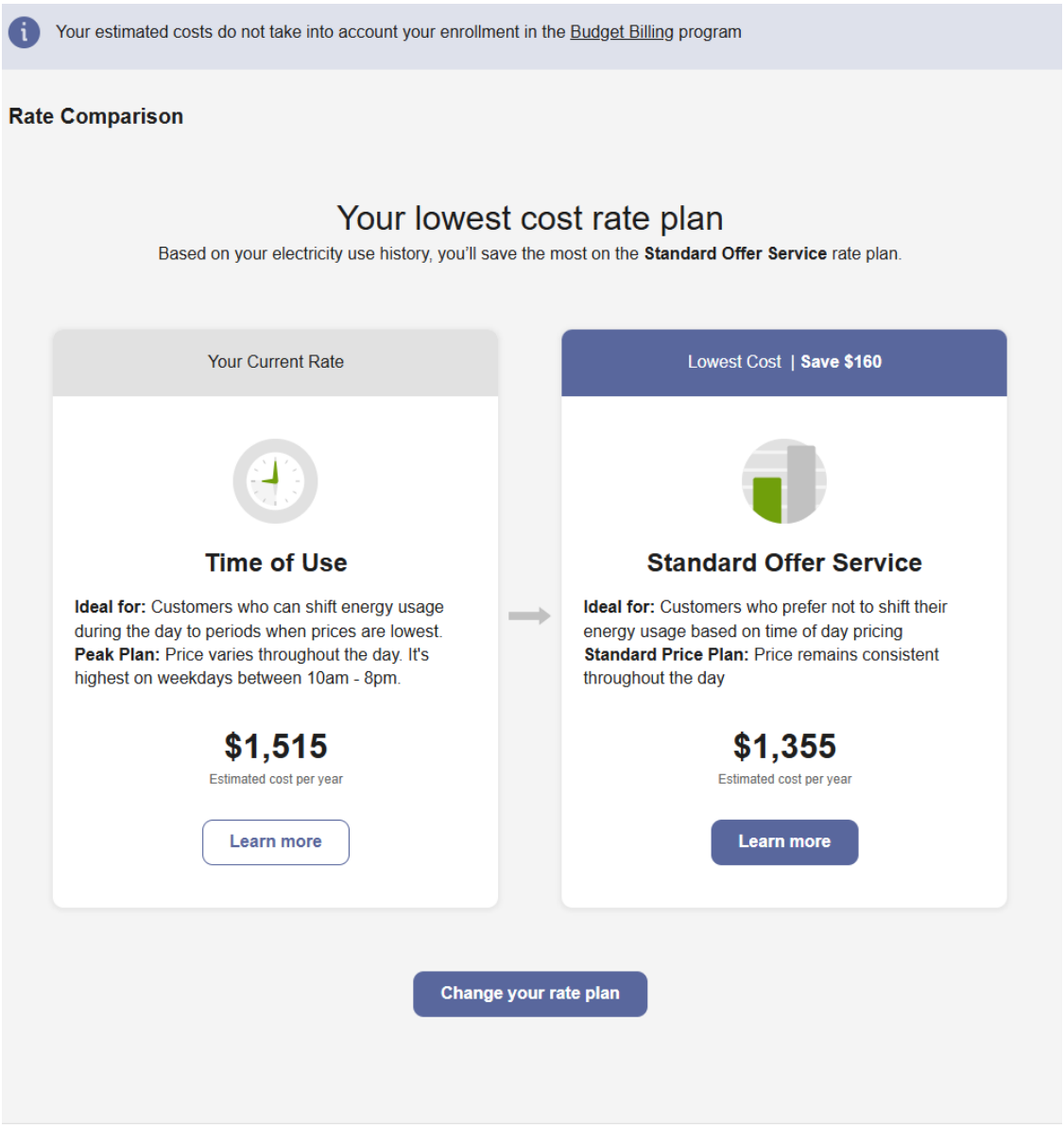
Full Embedding Code: `<opower-widget-portfolio-view opower-instance="widget-portfolio-view"></opower-widget-portfolio-view>`

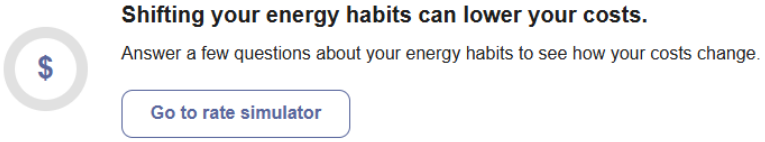
- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.
- This widget presents a dashboard of billing and premise information for business customers, along with links to more detailed energy usage information in other widgets. With this in mind, consider embedding this widget in an introductory or top-level section of your website, since it is meant to be an overview which then leads to more in-depth resources.

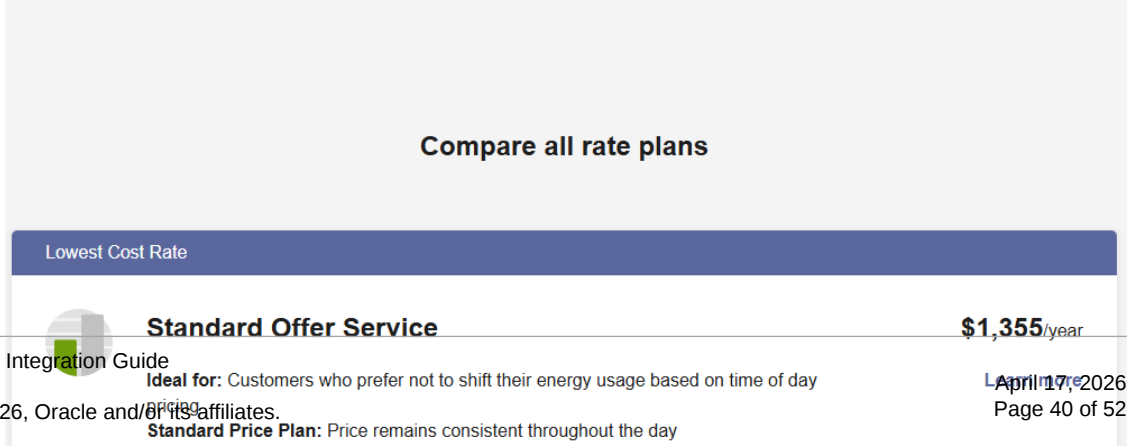
For more information about the Portfolio View, including details about its data requirements and the overall user experience, see the Portfolio View description in the relevant product overview.

Rate Comparison

The Rate Comparison allows customers to see an overview of rate plans they are eligible for, estimated cost information, and details about each rate plan option. Customers can compare rate plans in terms of estimated annual costs and quickly identify their cheapest rate. Further rate analysis is available through additional rate details and the ability to perform what-if scenarios representing energy use choices the customer can make.







Embedding Guidelines

Widget Name: `widget-rates`

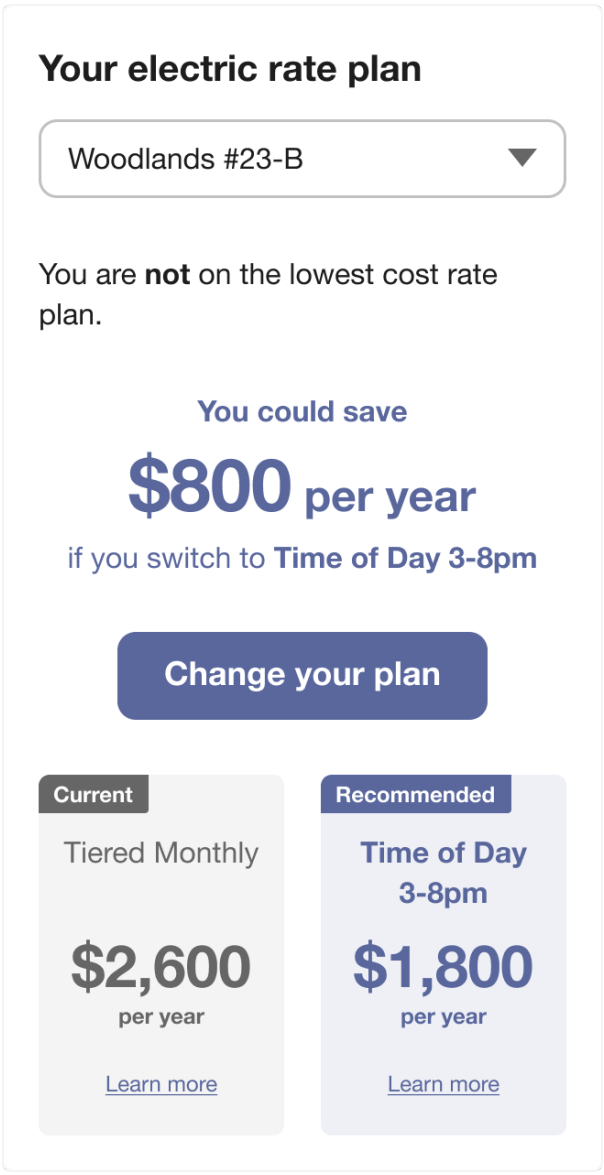
Full Embedding Code: `<opower-widget-rates opower-instance="widget-rates"></opower-widget-rates>`

The Rate Comparison widget consists of three components: Rate Comparison, Rate Details, and an optional Rate Simulator. The `widget-rates` widget must be embedded on a separate webpage devoted to hosting the widget's components. Users can interact with links within their Rate Comparison which redirect the users to review Rate Details or complete the Rate Simulator survey.

> **Note**
>
> - If other areas of a utility's site include direct links to the Rate Details component, the webpage that hosts this widget must be enabled to allow the `rate-id` query parameter to be passed in, so that the correct rate can be displayed.
> - If you have an existing embedded configuration of Rate Comparison that uses multiple widgets, you must update your configuration to utilize the single widget-rates widget which supports all components of the Rate Comparison. You Delivery Team can assist you in this process.

Rates Light

The Rates Light widget is a dashboard-friendly tool designed to help customers quickly view and compare utility rate plans. Customers can use the widget to understand at a glance whether they are on the best rate, and if not, which plan is recommended. The widget directs customers to the detailed Rate Comparison Feature for a more in-depth comparison. The widget directs customers to the detailed Rate Comparison Feature for a more in-depth comparison.



Embedding Guidelines

Widget Name: `widget-rates-light`

Full Embedding Code: `<opower-widget-rates-light opower-instance="widget-rates-light"></opower-widget-rates-light>`

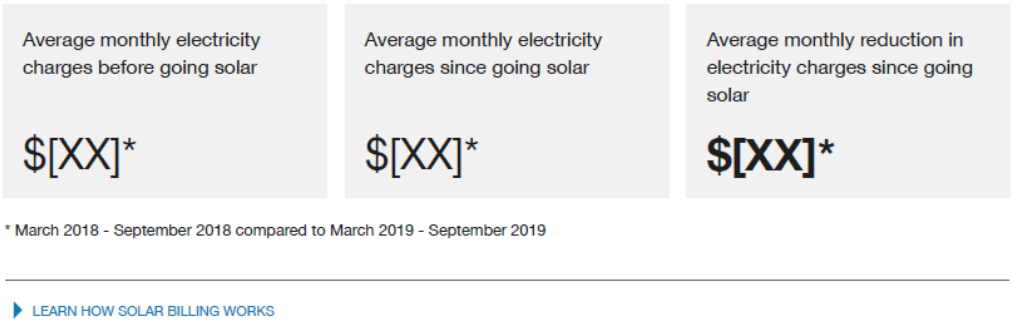
- This widget is well-suited for embedding in dashboards alongside other widgets for a unified view of energy, billing, and rate plan information. Consider embedding it near or alongside other dashboard-friendly information, such as the customer's current balance and usage trends that customers can explore.
- Embed this widget on highly visited areas of the utility website, such as an account overview dashboard. While the Rate Comparison tool is available to help customers determine if they are on the least expensive rate given their usage, many customers will not navigate far enough into the website to find this tool. Embedding the Rates Light widget in prominent dashboard locations helps ensure customers can quickly access rate

comparison information when it is most relevant, and drives traffic to the Rate Comparison tool.

For more information, including details about the data requirements and the overall user experience, see the Rates Light Overview description in the *Oracle Utilities Opower Rates Engagement Cloud Service Overview*.

Solar Savings Insight

The Solar Savings Insight is designed to provide a simplified view of how much money a customer has saved on their electricity bill since becoming a solar customer. It is displayed beneath the Data Browser for solar customers who have already seen the Solar Welcome Experience - How Solar Works.



Embedding Guidelines

Widget Name: `widget-solar-savings`

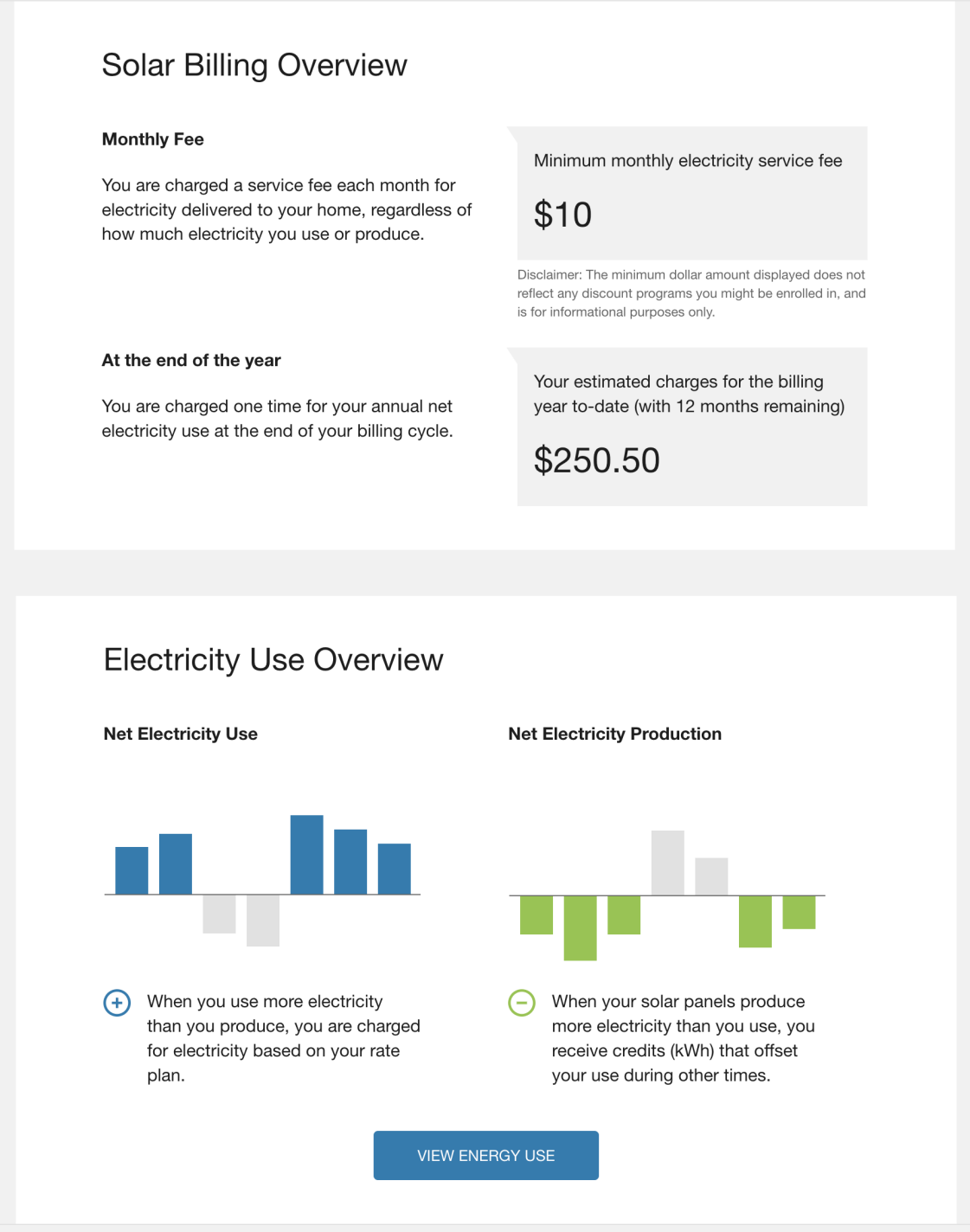
Full Embedding Code: `<opower-widget-solar-savings opower-instance="widget-solar-savings"></opower-widget-solar-savings>`

- The Solar Savings widget is designed to be embedded beneath the Data Browser. It takes up the same width as the Data Browser and will push down any content that is located beneath it.
- It is recommended that you pair the Solar Savings widget with the version of the Solar Welcome Experience - How Solar Works widget that appears beneath the Data Browser. If you do this, make sure that Solar Savings appears above Solar Welcome Experience - How Solar Works, since it is a natural transition for customers to view their solar savings insights followed by an explanation of those insights.
- The Solar Savings widget is often paired with the Green Button widget below the Data Browser. It is recommended that you embed the Solar Savings widget to appear above the Green Button widget.

For more information about the Solar Savings Insight, including details about its data requirements and the overall user experience, see the Solar Savings Insight description in the *Distributed Energy Resources Customer Engagement Cloud Service Overview*.

Solar Welcome Experience - How Solar Works

The Solar Welcome Experience - How Solar Works widget presents a collection of insights and graphics to help customers understand their positive and negative energy use, and interpret their solar bills. It is displayed for customers who navigate to the **Electricity** view in the Data Browser for the first time after starting a solar program.



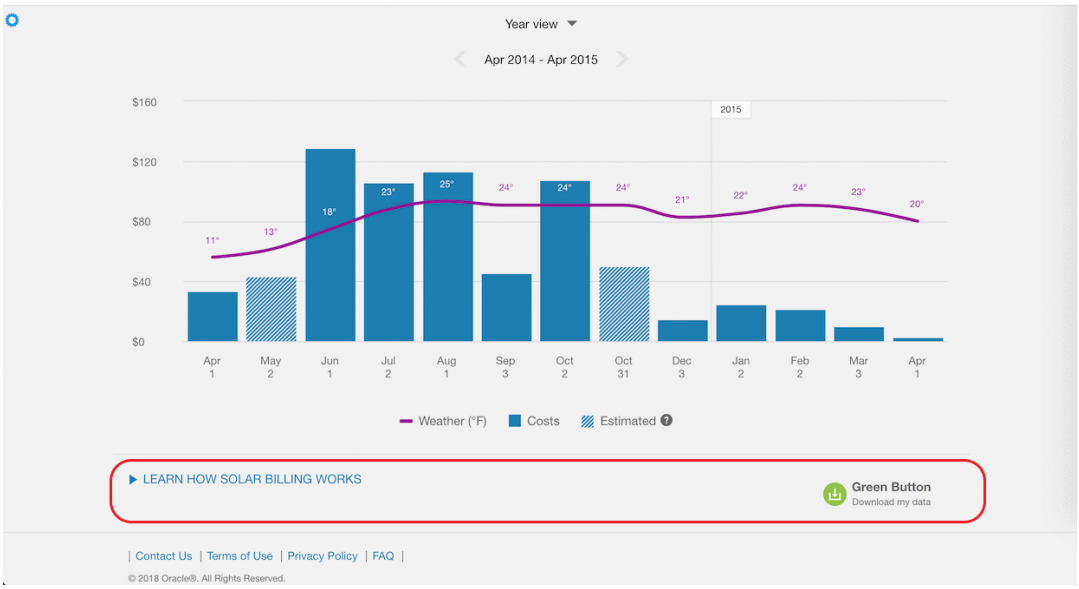
On subsequent visits, the full-page experience is hidden, and customers can find the widget by clicking the **Learn How Solar Billing Works** menu beneath the Data Browser.

Embedding Guidelines

Widget Name: `widget-how-solar-billing-works`

Full Embedding Code: `<opower-widget-how-solar-billing-works opower-instance="widget-how-solar-billing-works"></opower-widget-how-solar-billing-works>`

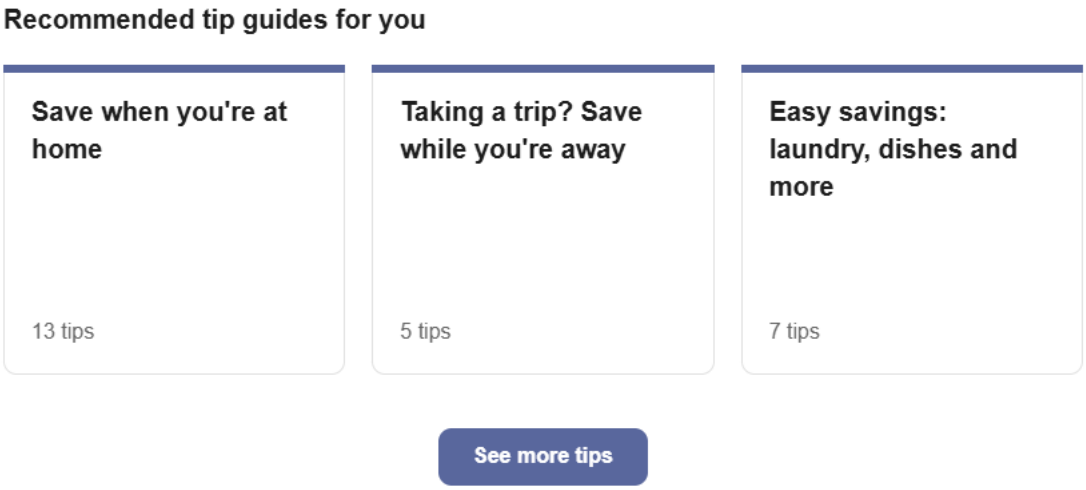
- The Solar Welcome Experience - How Solar Billing Works widget is designed to be embedded on the same page as the Data Browser. Keep in mind that it appears as a full-page experience the first time a customer browses to the **Electricity** view of the Data Browser after starting a solar program. On subsequent visits, the widget appears beneath the Data Browser within a collapsible menu. When the menu is expanded, the widget takes up the same width as the Data Browser and pushes down any content that is located beneath it.
- Consider pairing the Solar Welcome Experience - How Solar Works widget with the Solar Savings Insight. If you do this, make sure that Solar Savings appears above Solar Welcome Experience - How Solar Works, since it is a natural transition for customers to see their solar savings insights followed by an explanation of those insights.
- The How Solar Works widget is often paired with the Green Button widget so that both appear below the Data Browser. It is recommended that you embed How Solar Works to appear inline with Green Button.



For more information about Solar Welcome - How Solar Works, including details about its data requirements and the overall user experience, see the Solar Welcome Experience description in the *Distributed Energy Resources Customer Engagement Cloud Service Overview*.

Tips Light

Tips Light displays the top three tip guides for the customer's household. These tip guides promote customer interest in tips and lead them to more detailed tip information.



Embedding Guidelines

Widget Name: `widget-tips-light`

Full Embedding Code: `<opower-widget-tips-light opower-instance="widget-tips-light"></opower-widget-tips-light>`

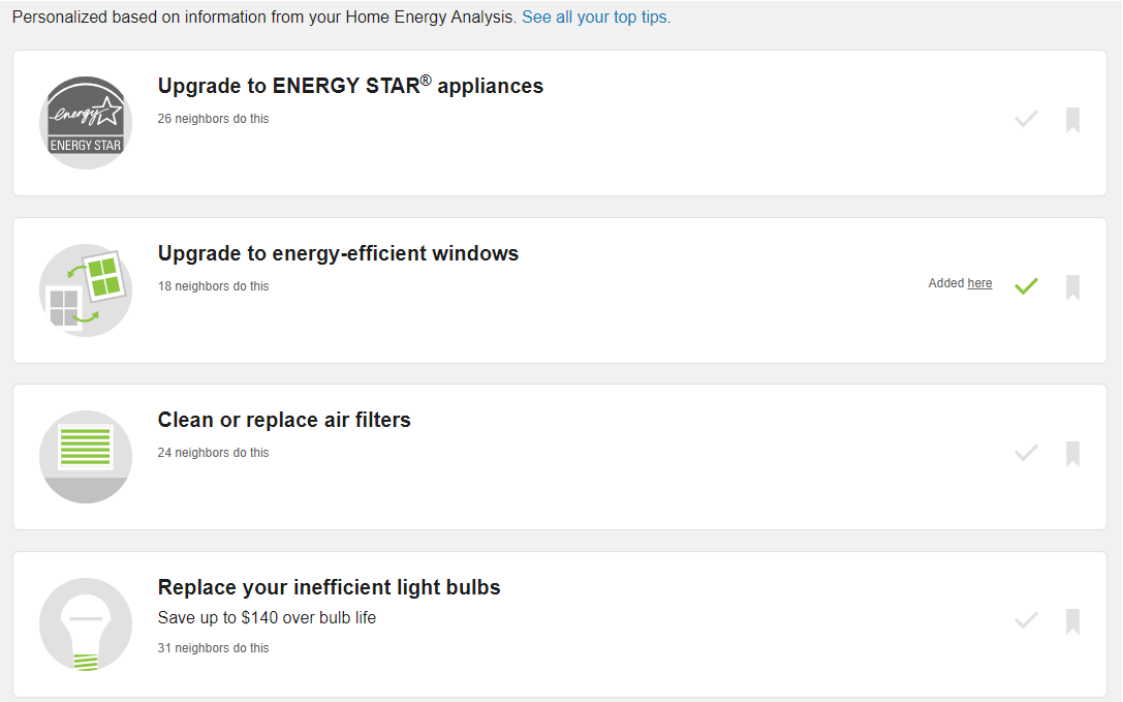
This widget acts as a summary of practical steps for customers to take after seeing a summary of their energy use. It is well-suited for a location near other summary widgets such as Energy Use Overview or Highest Energy Use Days. When viewed together, these widgets make it

easy for customers to see current use and take action. Your Delivery Team will work with you to determine the ideal placement.

For more information about the Tips Light, including details about its data requirements and the overall user experience, see the Tips Light description in the product overview.

Tips List

The Tips List widget displays the top five energy efficiency tips for a customer's household. The priority and order of the tips is determined based on data that is available for the customer. The purpose is to promote customer interest in tips and guide them to more energy efficient behavior.



Embedding Guidelines

Widget Name: `widget-ways-to-save`

Full Embedding Code: `<opower-widget-ways-to-save opower-instance="widget-ways-to-save"></opower-widget-ways-to-save>`

The Tips List widget is typically paired with the Home Energy Analysis or the Bill Comparison. Your Delivery Team will work with you to determine the ideal placement.

> **Note**
>
> This widget is part of the Ways to Save widget described in Ways to Save. This widget can be configured to display the Tips List user experience by default rather than the Ways to Save experience.

For more information about the Tips List, including details about its data requirements and the overall user experience, see the Tips List description in the product overview.

Ways to Save

Ways to Save presents a personalized selection of energy saving tips. Customers can browse through the available tips to learn how they can save energy.

When embedding the Ways to Save widget, a single widget includes all components in an experience that is hosted on a single webpage. A query parameter `ou-wts-state` is automatically passed to direct users to the correct component. The host webpage allows this query parameter to be passed. The parameter lets customers browse tip guides, browse the list of tips in those guides, and review details of a particular tip. An example of this browsing workflow is described below.

Tip Guides

Ways to Save displays a collection of tip guides to customers. Tips are organized into behavior-oriented, seasonal, and end-use guides that provide customers with relevant cost-savings recommendations.

Top 2 tips for this season

Install window shades such as blinds or shutters

Set your thermostat to 78°F in the summer

Save up to $1,200 per year

See all seasonal tips

Get more energy-saving advice

Easy to do | Cost type | Available rebates | Seasonal | Appliances and more

Top tips for homes like yours

71 tips

Want more relevant tips for your home? Take the survey

Ways to save at no cost to you

27 tips

High cooling bills? Here's how to cut back

9 tips

Heating tips to help you save

6 tips

Save every day with home appliances

12 tips

Show all

Tips you've saved

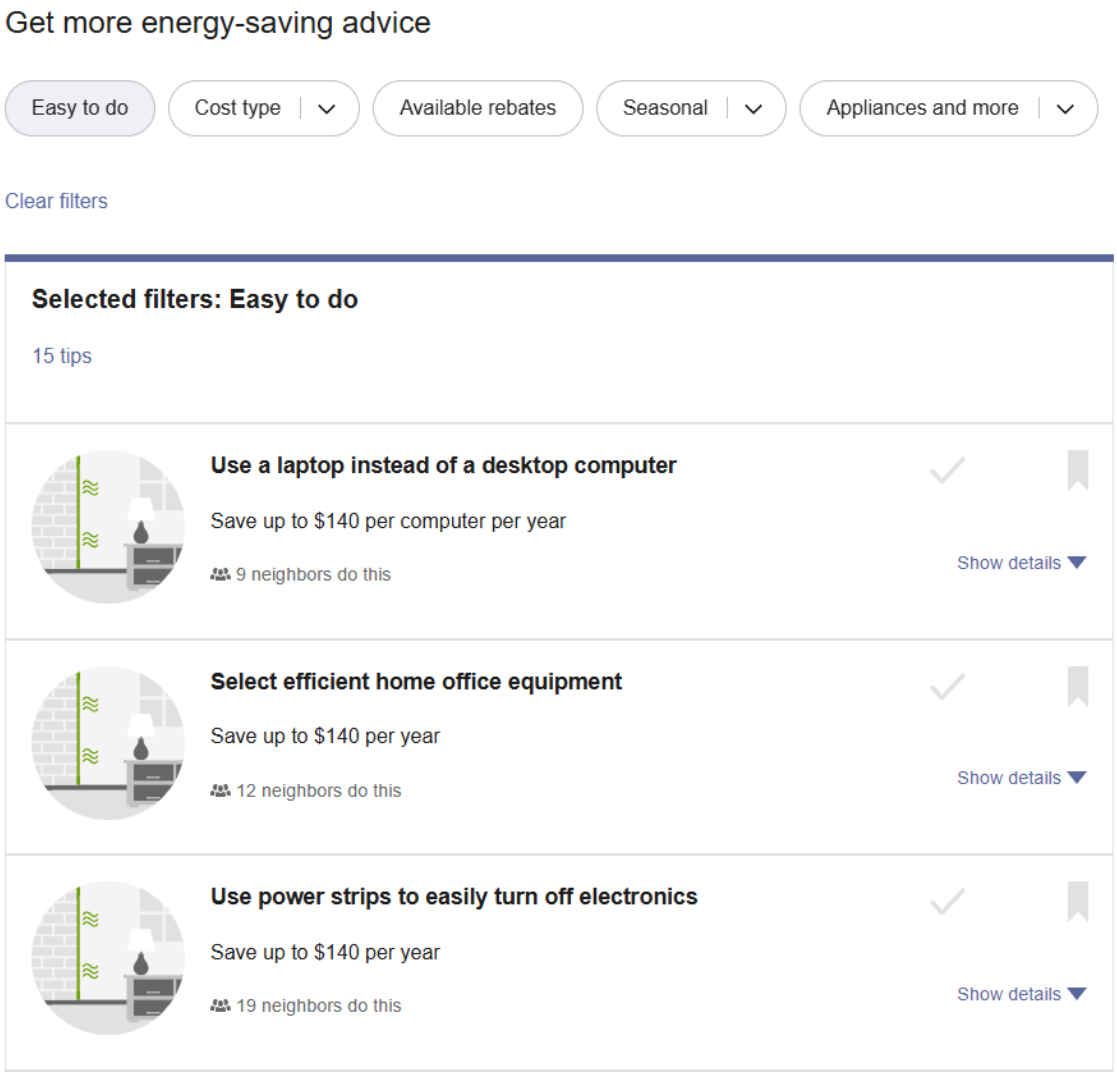
1 tip

Tips you've completed

1 tip

Tips List

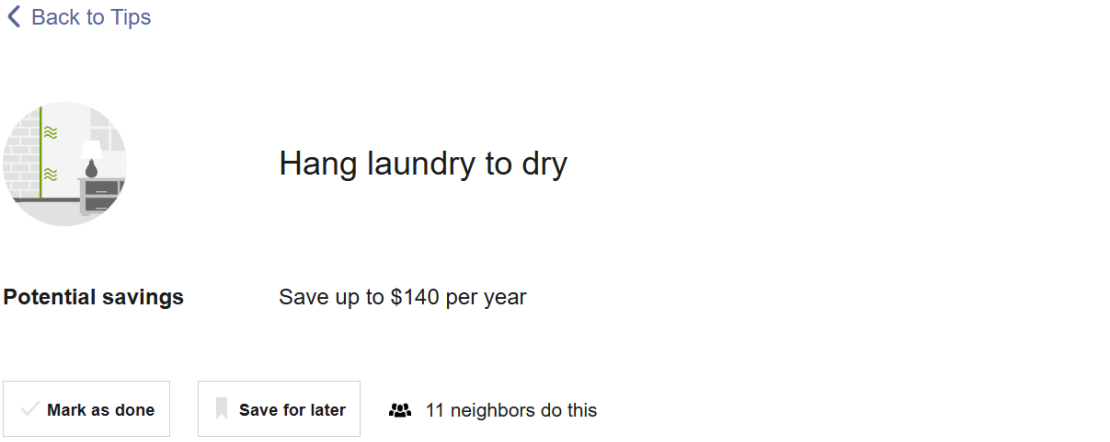
Customers can select a tip guide to view the list of tips included in that guide. From this list of tips, customers can review high-level information about each tip, and select a tip to view additional details.



Tip Details

Customers can select a tip to view reasons why they should complete a tip. An option to **Read More** about the tip displays all available information for the tip.

A breadcrumb link is included to allow customers to return to the list of tips.



Home Energy Analysis Call-to-Action

At the top of the Ways to Save page, you can embed a call-to-action widget that encourages customers to complete the Home Energy Analysis in order to receive better tips. See Home Energy Analysis Call-to-Action for more information.

Embeddable Guidelines

Widget Name: `widget-ways-to-save`

Full Embedding Code: `<opower-widget-ways-to-save opower-instance="widget-ways-to-save"></opower-widget-ways-to-save>`

- This widget is well-suited for embedding in sections that use the full width of the page. Consider the use of tabs, accordions, or other elements that can show or hide content when embedding this widget on a webpage along with other widgets and content.
- Ways to Save can support authenticated and pre-authenticated experiences. To support both, the widget must be embedded twice, once on an authenticated webpage, and once on a pre-authenticated webpage.

- Ways to Save is intended for customers who have analyzed their energy use and are ready to take action. Other products and widgets (such as High Bill Alerts AMI, Weekly Energy Updates, and the Home Energy Analysis v2) can send customers directly to Ways to Save to learn about practical next steps. Since many customers are redirected to this widget, prominent utility site location is not as important a factor to drive user interaction with this widget.

For more information about Ways to Save, including details about its data requirements and the overall user experience, see the resources below.

- Ways to Save for residential customers
- Ways to Save for business customers

5

Determining a Widget's Applicability

Not all widgets are applicable to all customers, and so you may need to determine whether or not a widget should be displayed on a webpage under certain conditions.

For example, some widgets are available for residential customers but not non-residential customers. Other widgets can only show data for electricity use and so should not be displayed for gas-only customers.

Use the steps below to help you determine what widgets are applicable to different types of customers. With this information, you can better understand what widgets to display for which customers, and consider if and how to implement conditional logic in your webpages to control the display of the widgets.

Determine a Widget's Supported Customer Types

Determine whether the widget you want to display is available for residential or non-residential customers. To do this, open the appropriate cloud service overview guide:

- For residential customers: Digital Self Service - Energy Management *Overview Guide*
- For non-residential customers: Business Customer Engagement Digital Self Service - Energy Management *Overview Guide*

Once you are in either of these guides, go to the **Customer Experience** section and see if your widget is listed there. If it is not, then it is not supported for that customer type.

Determine a Widget's Supported Fuels

Determine whether your widget can display electricity and gas data. Most widgets can display both gas and electricity data, but some can only show one or the other.

1. Open the appropriate cloud service overview guide which contains a description of the widget you are interested in.
    - For residential customers: Digital Self Service - Energy Management *Overview Guide*
    - For non-residential customers: Business Customer Engagement Digital Self Service - Energy Management *Overview Guide*
2. Go to the **Customer Experience** section of the guide and navigate to the appropriate widget description.
3. Once you are in the relevant description, go to **Customer Requirements** and look for the **Supported Fuels** row. This will explain what fuels the widget supports.

Determine a Widget's Required Data

Determine the type of data your widgets require and whether they will be relevant to customers who have that type of data. Most widgets work using billing data, but some widgets require smart meter (AMI) data to work.

1. Follow the steps in Determine a Widget's Supported Fuels above to navigate to a description of the widget in the relevant cloud service overview guide.
2. In the **Customer Requirements** subsection of the description, review the rows related to data requirements. This information will explain whether the widget requires smart meter data (AMI data), billing data, or some other type of customer data.

Example of Widget Applicability

The table below gives an example of how you might create a list based on your research in the steps above. Your list can identify which embeddable widgets to show or hide based on different customer attributes. Work with your Delivery Team to create a more comprehensive list tailored to your situation.

Widget	Customer Type	Fuel Type	Data Type	Show / Hide
Bill or Usage Forecast	Residential	Electricity	AMI	Show
	Residential	Electricity	Billing	Hide Widget requires AMI data and will not work for customers with billing data only.
	Non-Residential	Electricity	AMI	Show
Data Browser	Residential	Electricity	AMI	Show
	Residential	Electricity	Billing	Show
	Residential	Gas	AMI	Show
	Non-Residential	Gas	Billing	Show
Highest Energy Use Days	Residential	Electricity	AMI	Show
	Residential	Electricity	Billing	Hide Widget requires AMI data and will not work for customers with billing data only.
	Residential	Gas	Billing	Hide Widget requires AMI data and will not work for customers with billing data only.
	Non-Residential	Electricity	Billing	Hide Widget is currently only available for residential customers.

6
Embedding a Widget Using Web Components

Embedding widgets using web components requires basic integration that includes the widget in the HTML head and body of a webpage, as well as additional configuration for performance, styling, custom events, and so on.

Prior to performing the steps below, review the Guidelines for Embedding Widgets for the widgets you intend to embed. Repeat these steps for each webpage that includes embedded widgets.

Widget Names

When embedding widgets, use the following widget names. The full embedding code is also provided which includes any required instance names:

> **Note**
>
> The full embedding code provided for widgets below includes the default widget and instance names. Contact Your Delivery Team to determine and receive any additional widget instance names if you embed multiple versions of widgets in your environment.

Feature	Widget Names and Full Embedding Code

Account Center	`widget-communication-preferences`
	Full embedding code:
	
	`<opower-widget-communication-preferences opower-instance="widget-communication-preferences"></opower-widget-communication-preferences>`

Affordability Savings Hub	For the authenticated workflow, you must use `widget-lmi-survey-programs`, and define the `opower-instance` to embed the survey questions on an authenticated webpage. The webpage must exclude footers and side navigation panels. The full embedded tag is:
	
	`<opower-widget-lmi-survey-programs opower-instance="widget-lmi-survey-programs"></opower-widget-lmi-survey-programs>`
	
	For the pre-authenticated workflow, follow the same steps but embed the survey questions on a pre-authenticated webpage. The full embedded tag is:
	
	`<opower-widget-lmi-survey-programs opower-instance="widget-lmi-survey-programs-easyopen-full"></opower-widget-lmi-survey-programs>`

Feature	Widget Names and Full Embedding Code

Bill Comparison	`widget-bill-compare-enhanced`
	Full embedding code:
	
	`<opower-widget-bill-compare-enhanced opower-instance="widget-bill-compare-enhanced"></opower-widget-bill-compare-enhanced>`

Bill or Usage Forecast	`widget-bill-forecast`
	Full embedding code:
	
	`<opower-widget-bill-forecast opower-instance="widget-bill-forecast"></opower-widget-bill-forecast>`

Bill Guide	`widget-bill-guide`
	Full embedding code:
	
	`<opower-widget-business-profile opower-instance="widget-business-profile"></opower-widget-business-profile>`

Business Profile	`widget-business-profile`
	The same `opower-instance` can be used for authenticated or pre-authenticated workflows. The full embedded tag is:
	
	`<opower-widget-business-profile opower-instance="widget-business-profile"></opower-widget-business-profile>`

Confirmation Message	`widget-bite-sized`
	Full embedding code:
	
	`<opower-widget-bite-sized opower-instance="widget-bite-sized"></opower-widget-bite-sized>`

Data Browser	`widget-data-browser`
	Full embedding code:
	
	`<opower-widget-data-browser opower-instance="widget-data-browser"></opower-widget-data-browser>`

Demand Heatmap	`widget-demand-heatmap`
	Full embedding code:
	
	`<opower-widget-demand-heatmap opower-instance="widget-demand-heatmap"></opower-widget-demand-heatmap>`

Feature	Widget Names and Full Embedding Code

Energy Use Overview	`widget-usage-overview`
	Full embedding code:
	
	`<opower-widget-usage-overview opower-instance="widget-usage-overview"></opower-widget-usage-overview>`

Green Button	`widget-usage-export`
	Full embedding code:
	
	`<opower-widget-usage-export opower-instance="widget-usage-export"></opower-widget-usage-export>`

Guest User Access	`widget-guest-user-access`
	Full embedding code:
	
	`<opower-widget-guest-user-access opower-instance="widget-guest-user-access"></opower-widget-guest-user-access>`

Highest Energy Use Days	`widget-highest-day-of-use`
	Full embedding code:
	
	`<opower-widget-highest-day-of-use opower-instance="widget-highest-day-of-use"></opower-widget-highest-day-of-use>`

Home Energy Analysis v1 - Authenticated Workflow	For the survey questions, you must use `widget-survey`, and define the `opower-instance` to embed the survey questions on a separate webpage. This webpage must exclude footers and side navigation panels. The full embedded tag is:
	
	`<opower-widget-survey opower-instance="widget-survey-full"></opower-widget-survey>`
	
	For the pre- and post-survey experience, you must embed both `widget-survey` and `widget-disaggregation` together on the same webpage, and define the `opower-instance` attribute as part of the embedded tag. Only one experience is displayed to customers at a time. The full embedded tags are:
	`<opower-widget-survey opower-instance="widget-survey-splash"></opower-widget-survey>`
	
	`<opower-widget-disaggregation opower-instance="widget-disaggregation"></opower-widget-disaggregation>`

Feature	Widget Names and Full Embedding Code

Pre-Authenticated - EasyOpen Workflow - Pre-Authenticated Workflow	You can allow customers to access the survey without logging in to their utility account.
	For the survey questions, you must use `widget-survey` to embed the survey questions on a separate webpage. This webpage must exclude footers and side navigation panels. The full embedded tag is:
	
	`<opower-widget-survey opower-instance="widget-survey-easyopen-full"></opower-widget-survey>`
	
	For the pre and post survey experience you must embed both `widget-survey` and `widget-disaggregation` together on the same webpage. Only one experience is displayed to customers at a time. The full embedded tags are:
	
	`<opower-widget-survey opower-instance="widget-survey-easyopen-splash"></opower-widget-survey>`
	
	`<opower-widget-disaggregation opower-instance="widget-dissagregation-easyopen"></opower-widget-disaggregation>`

Home Energy Analysis v2 - Authenticated Workflow	For the survey questions, you must use `widget-survey`, and define the `opower-instance`, to embed the survey questions on a separate webpage. This webpage must exclude footers and side navigation panels. The full embedded tag is:
	
	`<opower-widget-survey opower-instance="widget-survey-full"></opower-widget-survey>`
	
	For the pre and post survey experience you must embed both `widget-survey` and `widget-usage-categories` together on the same webpage, and define the `opower-instance` attribute as part of the embedded tag. Only one experience is displayed to customers at a time. The full embedded tags are:
	
	`<opower-widget-survey opower-instance="widget-survey-splash"></opower-widget-survey>`
	
	`<opower-widget-usage-categories opower-instance="widget-usage-categories"></opower-widget-usage-categories>`

Feature	Widget Names and Full Embedding Code

Pre-Authenticated - EasyOpen Workflow - Pre-Authenticated Workflow	You can allow customers to access the survey without logging in to their utility account.
	For the survey questions, you must use `widget-survey` to embed the survey questions on a separate webpage. This webpage must exclude footers and side navigation panels. The full embedded tag is:
	
	`<opower-widget-survey opower-instance="widget-survey-easyopen-full"></opower-widget-survey>`
	
	For the pre and post survey experience you must embed both `widget-survey` and `widget-usage-categories` together on the same webpage. Only one experience is displayed to customers at a time. The full embedded tags are:
	
	`<opower-widget-survey opower-instance="widget-survey-easyopen-splash"></opower-widget-survey>`
	
	`<opower-widget-usage-categories opower-instance="widget-usage-categories-easyopen"></opower-widget-usage-categories>`

Home Energy Analysis Call-to-Action	`widget-hea-cta`
	Full embedding code:
	
	`<opower-widget-hea-cta opower-instance="widget-hea-cta"></opower-widget-hea-cta>`

Home Energy Analysis Light	`widget-usage-categories-overview`
	Full embedding code:
	
	`<opower-widget-usage-categories-overview opower-instance="widget-usage-categories-overview"></opower-widget-usage-categories>`

How Businesses Use Energy	`widget-business-disaggregation`
	Full Embedding Code:
	
	`<opower-widget-business-disaggregation opower-instance="widget-business-disaggregation"></opower-widget-business-disaggregation>`

Mini Bill Comparison	`widget-mini-bill-compare`
	Full embedding code:
	
	`<opower-widget-mini-bill-compare opower-instance="widget-mini-bill-compare"></opower-widget-mini-bill-compare>`

Feature	Widget Names and Full Embedding Code

Neighbor Comparison	`widget-neighbor-comparison`
	Full embedding code:
	
	`<opower-widget-neighbor-comparison opower-instance="widget-neighbor-comparison"></opower-widget-neighbor-comparison>`

Next Best Action	`widget-next-best-action`
	Full embedding code:
	
	`<opower-widget-next-best-action opower-instance="widget-next-best-action"></opower-widget-next-best-action>`

Peak Time Rebates	`widget-peak-time-rebate`
	Full embedding code:
	
	`<opower-widget-peak-time-rebate opower-instance="widget-peak-time-rebate"></opower-widget-peak-time-rebate>`

Portfolio View	`widget-portfolio-view`
	Full embedding code:
	
	`<opower-widget-portfolio-view opower-instance="widget-portfolio-view"></opower-widget-portfolio-view>`

Rate Comparison	`widget-rates`
	Full embedding code:
	
	`<opower-widget-rates opower-instance="widget-rates"></opower-widget-rates>`

Rates Light	`widget-rates-light`
	Full embedding code:
	
	`<opower-widget-rates-light opower-instance="widget-rates-light"></opower-widget-rates-light>`

Near Real Time Usage	`widget-real-time-ami`
	Full embedding code:
	
	`<opower-widget-real-time-ami opower-instance="widget-real-time-ami"></opower-widget-real-time-ami>`

Feature	Widget Names and Full Embedding Code

Solar Savings Insight	`widget-solar-savings`
	Full embedding code:
	
	`<opower-widget-solar-savings opower-instance="widget-solar-savings"></opower-widget-solar-savings>`

Solar Welcome Experience - How Solar Works	`widget-how-solar-billing-works`
	Full embedding code:
	
	`<opower-widget-how-solar-billing-works opower-instance="widget-how-solar-billing-works"></opower-widget-how-solar-billing-works>`

Tips Light	`widget-tips-light`
	Full embedding code:
	
	`<opower-widget-tips-light opower-instance="widget-tips-light"></opower-widget-tips-light>`

Tips List	`widget-ways-to-save`
	This widget is part of the Ways to Save widget. This widget can be configured to display the Tips List user experience by default rather than the Ways to Save experience.
	Full embedding code:
	
	`<opower-widget-ways-to-save opower-instance="widget-ways-to-save"></opower-widget-ways-to-save>`

Ways to Save	`widget-ways-to-save`
	Full embedding code:
	
	`<opower-widget-ways-to-save opower-instance="widget-ways-to-save"></opower-widget-ways-to-save>`

Embedding a Widget

Widgets are embedded by adding scripts and tags directly in the source code of your webpages. These scripts and tags need to be set in accordance with the type of authentication required. Some widgets are embedded to require authentication, while others are embedded to not require authentication.

> **⚠ Caution**
>
> Do not mix authentication modes within the same page or application context. Doing so can cause several issues.
>
> - **Browser Blocking:** Safari and other browsers may block content when authenticated widgets are embedded in non-authenticated pages.
> - **Authentication Conflicts:** Mixing widgets can trigger unexpected authentication workflows.
>
> The best practice is to use the same authentication approach for all widgets on a page. Either all widgets should require OpenID Connect authentication, or all widgets should require pre-authenticated / non-authenticated contexts.

Embedding Content Requiring Authentication

The following code is an example of the basic integration of widgets on a webpage for OpenID Connect-based SSO implementations, with emphasis applied to the main integration points.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<title>Page Title</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="https://ei-util-stage.opower.com/ei/x/embedded-api/core.js?auth-
mode=oauth"></script>
</head>
<body>
 <div>
  <opower-widget-neighbor-comparison opower-instance="widget-neighbor-comparison"></
opower-widget-neighbor-comparison>
 </div>
 <div>
  <opower-widget-data-browser opower-instance="widget-data-browser"></opower-widget-data-
browser>
 </div>
</body>
</html>
```

The instructions below guide you through each step of the process and provide more background information about each the scripts and tags that are required.

> **⚠ Caution**
>
> If you are using Oracle Infinity to track engagement analytics for your website, then you must load your Oracle Analytics Infinity CX Tag on your webpage before completing the steps below. Otherwise, analytics tracking for your widgets will not work. See Tracking User Interaction Analytics for instructions. You can ignore this step if you are not using Oracle Analytics Infinity.

To prepare a webpage for embedding content requiring authentication:

1. Include the main script tag for the core library within the HTML head of the webpage. A single script tag is required regardless of the number of widgets embedded in the body of the page. The script tag uses the following syntax:

   ```
   <script src="https://[HostName]/ei/x/embedded-api/core.js"></script>
   ```

   Where `HostName` is the applicable host information. When embedding on the stage environment, the format is `ei-[ClientCode]-stage.opower.com`, where `ClientCode` is the client code assigned for the utility. Contact Your Delivery Team to confirm your client code. When embedding on the production environment, the format is `[ClientCode].opower.com`.

2. Append `auth-mode=oauth` as a query parameter and value to the `src` attribute in the script tag.

3. Append `locale` as a query parameter to the `src` attribute in the `script` tag. This is an optional parameter that allows embedded widgets to reflect a locale preference change by the customer that occurs after the customer has signed in to the Energy Management Web Portal. This parameter does not affect the outbound communication locale preference for a customer. The value for the `locale` parameter must be defined as the locale for the page where the widget is embedded. The example below for the production tier uses a static definition of US English:

   ```
   <script src="https://util.opower.com/ei/x/embedded-api/core.js?auth-
   mode=oauth&locale=en_US"></script>
   ```

4. Include the required HTML in the location on the page where the widget is to appear. This HTML includes a call to the relevant widget, which uses the following format:

   ```
   <opower-[WidgetName] opower-instance="[InstanceName]"></opower-
   [WidgetName]>
   ```

   Where `WidgetName` is the name of the applicable widget. You must also define the `opower-instance` attribute by supplying the applicable `InstanceName`. The full list of widget names and instance names is provided above. An example of embedding the Bill Comparison widget is shown below:

   ```
   <div>
   <opower-widget-bill-compare-enhanced opower-instance="widget-bill-compare-
   enhanced"></opower-widget-bill-compare-enhanced>
   </div>
   ```

5. Repeat the previous step for each widget you are including on the webpage. If you embed multiple widgets on a single webpage, consider a strategy to improve the load performance of the widgets that link to other widgets on the same page, which is described in Avoiding Webpage Refreshes.

6. Include listeners on the webpages, where a widget displays the authenticated experience, to provide required entity IDs and access tokens. These techniques are described at Providing Access Tokens for OpenID Connect.

7. Send your Delivery Team a listing of all your webpage URLs that contain the embedded widgets, along with the widget names included on each webpage. Any updates to the locations where the widget is embedded must be communicated to Delivery Team in advance of deploying the widget to the new location. This information can be provided along with other configuration inputs, as instructed in the Oracle Utilities Opower Digital Self Service - Energy Management Configuration Guide.

8. If your website uses a Content Security Policy, configure it to allow Oracle Utilities Opower resources. Contact your Delivery Team to obtain the domains and resources that should be allowed.

This process completes the minimum required steps to embed widgets on a webpage. You can employ additional techniques to improve the performance, user interaction, and overall look-and-feel of the widgets within the website. See Improving the Embedded Widget User Experience with Custom Events for more information.

Embedding Content Not Requiring Authentication

Pre-authenticated or unauthenticated content requires a similar setup with a few important differences. The following code is an example of the basic integration of widgets on a webpage for pre-authenticated implementations, with emphasis applied to the main integration points.

```
<!DOCTYPE html>
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<title>Page Title</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="https://ei-util-stage.opower.com/ei/x/embedded-api/core.js?auth-
mode=none"></script>
</head>
<body>
<div>
<opower-widget-survey opower-instance="widget-survey-easyopen-splash"></opower-widget-
survey>
</div>
</body>
</html>
```

The instructions below guide you through each step of the process and provide more background information about each of the scripts and tags that are required.

> **⚠ Caution**
>
> If you are using Oracle Infinity to track engagement analytics for your website, then you must load your Oracle Analytics Infinity CX Tag on your webpage before completing the steps below. Otherwise, analytics tracking for your widgets will not work. See Tracking User Interaction Analytics for instructions. You can ignore this step if you are not using Oracle Analytics Infinity.

To prepare a webpage for embedding content not requiring authentication:

1. Include the main script tag for the core library within the HTML head of the webpage. A single script tag is required regardless of the number of widgets embedded in the body of the page. The script tag uses the following syntax:

   ```
   <script src="https://[HostName]/ei/x/embedded-api/core.js"></script>
   ```

   Where `HostName` is the applicable host information. When embedding on the stage environment, the format is `ei-[ClientCode]-stage.opower.com`, where `ClientCode` is the

client code assigned for the utility. Contact Your Delivery Team to confirm your client code. When embedding on the production environment, the format is `[ClientCode].opower.com`.

2. Append `auth-mode=none` as a query parameter and value to the `src` attribute in the script tag.
3. Append `locale` as a query parameter to the `src` attribute in the `<script>` tag. This is an optional parameter that allows embedded widgets to reflect a locale preference change by the customer that occurs after the customer has signed in to the Energy Management Web Portal. This parameter does not affect the outbound communication locale preference for a customer. The value for the `locale` parameter must be defined as the locale for the page where the widget is embedded. The example below for the production tier uses a static definition of US English:

   ```
   <script src="https://util.opower.com/ei/x/embedded-api/core.js?auth-
   mode=none&locale=en_US"></script>
   ```

4. Include the required HTML in the location on the page where the widget is to appear. This HTML includes a call to the relevant widget, which uses the following format:

   ```
   <opower-[WidgetName] opower-instance="[InstanceName]"></opower-
   [WidgetName]>
   ```

   Where `WidgetName` is the name of the applicable widget. You must also define the `opower-instance` attribute by supplying the applicable `InstanceName`. The full list of widget names and instance names is provided above. An example of embedding the Bill Comparison widget is shown below:

   ```
   <div>
   <opower-widget-survey opower-instance="widget-survey-easyopen-splash"></
   opower-widget-survey>
   </div>
   ```

5. Repeat the previous step for each widget you are including on the webpage. If you embed multiple widgets on a single webpage, consider a strategy to improve the load performance of the widgets that link to other widgets on the same page, which is described in Avoiding Webpage Refreshes.
6. Include listeners on the webpages, where a widget displays the authenticated experience, to provide required entity IDs and access tokens. These techniques are described at Providing Access Tokens for OpenID Connect.
7. Send your Delivery Team a listing of all your webpage URLs that contain the embedded widgets, along with the widget names included on each webpage. Any updates to the locations where the widget is embedded must be communicated to Delivery Team in advance of deploying the widget to the new location. This information can be provided along with other configuration inputs, as instructed in the Oracle Utilities Opower Digital Self Service - Energy Management Configuration Guide.
8. If your website uses a Content Security Policy, configure it to allow Oracle Utilities Opower resources. Contact your Delivery Team to obtain the domains and resources that should be allowed.

This process completes the minimum required steps to embed widgets on a webpage. You can employ additional techniques to improve the performance, user interaction, and overall look-and-feel of the widgets within the website. See Improving the Embedded Widget User Experience with Custom Events for more information.

Tracking User Interaction Analytics

With widgets placed directly into a webpage, utilities can self-serve analytics tracking such as whether or not a user views a widget. This can be accomplished using any analytics platform you prefer.

Oracle Utilities Opower also uses the Oracle Infinity analytics platform to collect user interaction metrics. If you are already using Oracle Infinity to track engagement analytics for your website, then you must load your Oracle Infinity CX Tag in your webpage before adding any Opower-specific widget code snippets. The Oracle Infinity CX Tag is a snippet of JavaScript code that collects data from the webpages that users visit.

> **Note**
>
> The steps below assume you have already generated an Oracle Infinity CX Tag based on the instructions in CX Tag Quick Start > Standard.

To load your Oracle Infinity CX Tag:

1. Open the source file of the webpage where you intend to embed an Oracle Utilities Opower widget.
2. Add your Oracle Infinity CX Tag to the section of your webpage that is optimal for your website implementation. For example, in a static webpage design, you could add it to the `<head>` tag. For a more dynamic implementation (such as a single-page application), you may add it elsewhere. In the case of a static webpage, the simplest way to ensure the CX Tag is loaded and executed in the proper order is to use the `defer` script tag attribute. See the HTML5 specification for more information about the `defer` attribute. An example of how the attribute could look in your CX Tag is as follows:

   ```
   <script src="https://c.oracleinfinity.io/acs/account/{Account GUID}/js/
   {Tag ID}/odc.js" defer></script>
   ```

   In this example, `{Account GUID}` and `{Tag ID}` are variables which are generated during the process of creating your Oracle Infinity CX Tag.
3. Ensure that the script for loading the core widget library comes after the CX Tag. Otherwise, the script for the core widget library will initialize its own Infinity Analytics handler, and will not allow another Infinity Analytics handler to be used. The example below shows that the script for the CX Tag comes first and is followed by the script for the core widget library, which also has its own `defer` attribute.

   ```
   <script src="https://c.oracleinfinity.io/acs/account/{Account GUID}/js/
   {Tag ID}/odc.js" defer></script>

   <script src="https://util.opower.com/ei/x/embedded-api/core.js" defer></
   script>
   ```

4. Follow the rest of the standard steps to embed a widget as described in Embedding a Widget.

> **Note**
>
> The steps above are meant as an example only. Depending on your implementation, there may be other options to ensure the scripts are loaded and executed in the required order. Contact Your Delivery Team if you need additional guidance.

Improving the Embedded Widget User Experience with Custom Events

This page provides guidance for embedding Opower widgets for multi-premise, multi-operating company (multi-OpCo), and single-page application (SPA) scenarios. It is intended for implementers, web developers, and utility administrators who integrate the widgets and need to manage customer context, authentication, and navigation events.

Embedded Opower widgets emit events at key moments, including when a user selects a widget link, when authorization is needed, or when the widget is ready to start. You can configure event listeners in your portal code to handle these events, improving the embedded experience in the following situations:

- Single-page applications that must avoid full-page reloads
- Pages with multiple widgets that link together across tabs or accordions
- Sessions where customers switch between multiple premises or OpCos
- When you need to optimize performance by loading widgets only when necessary

Use this guidance to integrate customer account data, set up performance features, and maintain a consistent user experience across embedded widgets.

Handling Widget Navigation and Preventing Page Refreshes

Applies to: Single-page applications and multi-widget pages. Not required for implementations that allow full-page navigation for all widget links.

To keep the user on the same page and avoid full-page reloads, intercept and manage widget navigation events. This enables in-page routing and helps maintain context across widgets.

> **Note**
>
> Each `opower:link` event includes a specific `experienceName` value. The `experienceName` value is included in the `opower:link` event payload. You can use it to route the customer to the appropriate page or correct location within a page. The `experienceName` values configured for your embedded widgets will be provided to you by your Contact Your Delivery Team.

Handling Widget-to-Widget Navigation

When a user selects a link inside an embedded widget, the widget triggers an `opower:link` event. By default, this event causes the browser to navigate to the target location, resulting in a full-page reload. If the destination content already exists on the current page (for example, in another tab or accordion), you can prevent unnecessary full-page reloads by intercepting

`opower:link`, calling `event.preventDefault()`, and routing within the existing page (for example, switching tabs or opening an accordion).

Example JavaScript for Routing within a Page:

```
window.addEventListener('opower:link', function(event) {
 if (event.detail.experienceName === 'energyUseDetails') {
   event.preventDefault();
   $.hide(event.target);
   $.show('#energyUseDetails');
  }
});
```

> **Note**
>
> Add animation or smooth scrolling logic when redirecting to minimize user confusion. This best practice helps maintain the user's orientation during navigation.

Example Event Payload

When a user selects a link in a widget, the `opower:link` event is triggered. The event includes details about the intended navigation action, including the widget that initiated the link.

```
{
widgetName: 'widget-data-browser',
experienceName: 'energyUseDetails',
url: {
    protocol: 'https:',
    slashes: true,
    auth: null,
    host: 'util.opower.com',
    port: null,
    hostname: 'util.opower.com',
    hash: null,
    search: '?fixtures=true',
    pathname: '/ei/x/energy-use-details',
    path: '/ei/x/energy-use-details?fixtures=true',
    href: 'https://util.opower.com/ei/x/energy-use-details?fixtures=true',},
    target: '_self'
}
```

Setting or Switching Customer Context

Setting Customer Context at Startup

Applies to: Multi-premise, single-page applications, and multi-OpCo scenarios. May also apply to OpenID Connect and other authentication models that provide account context data.

To ensure embedded widgets load the correct customer context—such as for customers with multiple premises or customers switching accounts across multiple operating companies (OpCos)—listen for and handle the `opower:start` event. When `opower:start` fires, the event payload includes an Opower JavaScript API object (`opowerApi`) for managing customer context. Your portal must:

1. Capture the API object for future use (for example, account switching without page reload).
2. Call `opowerApi.setEntityIds(...)` to set the initial customer context.
3. Call `opowerApi.start()` to load widget content.

> **Note**
>
> The identifiers you pass to `opowerApi.setEntityIds(...)` must exist in the identity data returned by your authentication integration (for example, an OpenID Connect UserInfo response). See the Oracle Utilities Opower SSO Configuration Guide for guidance, which describes creating a UserInfo endpoint.

Providing a Single Customer or Account ID (Multi-Premise)

Applies to: Multi-premise and single-page application scenarios (recommended). Not required for single-account customers unless your portal still needs to set context explicitly.

You can provide the customer ID to further identify the customer interacting with an embedded widget. You can accomplish this by listening for and handling the `opower:start` event. Providing the customer ID ensures that customers with multiple premises are given the correct experience.

The `setEntityIds` method accepts an array of exactly one value:

```
window.addEventListener('opower:start', function(event) {
    window.opowerApi = event.detail;
    window.opowerApi.setEntityIds(['ABC-123']);
    window.opowerApi.start();
});
```

Providing a Customer or Account ID with a Utility Code (Multi-OpCo)

Applies to: Multi-OpCo and single-page application scenarios (recommended). Not required for single-operating company portals.

For multi-OpCo parent portals, you must set context using an object that includes both the account identifier and the utility or OpCo utility code. A utility code is set up as part of the program launch process. Contact Your Delivery Team if you are unsure of your code. The `setEntityIds` method still accepts an array of exactly one value, but the value is an object.

```
{ accountId: 'ABC-123', clientCode: 'UTIL' }

window.addEventListener('opower:start', function (event) {
  window.opowerApi = event.detail;
  window.opowerApi.setEntityIds([{
    accountId: 'ABC-123',
    clientCode: 'UTIL'
  }]);
  window.opowerApi.start();
});
```

Handling Asynchronous Startup for Customer Context Lookup

Applies to: Single-page applications, multi-premise scenarios, and multi-OpCo portals. Not required for implementations that can set context synchronously.

If additional logic is required to determine the customer context, use an asynchronous handler and call the `event.preventDefault()` function to prevent synchronous execution.

```
window.addEventListener('opower:start', function(event) {
  window.opowerApi = event.detail;
  event.preventDefault(); // Prevent synchronous start of widget.
  fetchOpowerConfiguration(function(error, configuration){ //
fetchOpowerConfiguration is an example utility defined function
      if (error) {
      //  Handle your own error. Widget content has not loaded.
      return;
      }
      window.opowerApi.setEntityIds(configuration.entityIds);
      window.opowerApi.start();
      }
    });
});
```

Specifying Script Loading Order

Applies to: Single-page applications, multi-widget pages, multi-premise scenarios, multi-OpCo portals, and OpenID Connect integrations. In practice, this applies to any implementation that uses the `opower:start` handler.

Define the `opower:start` event handler in the correct order relative to the `core.js` script tag, which is the main script for our core library.

It is recommended that you place the `core.js` script tag in the `<head>` and the `opower:start` event handler in the `<body>`. The `core.js` must contain the `defer` attribute to ensure that core widget library is loaded and executed in the proper order. See the HTM5 specification for more information about the `defer` attribute.

Alternatively, place the event handler and the `core.js` script tag in the `<head>`. However, in that case the event handler must be defined before the `core.js` script tag.

Switching Accounts after Page Load without Refresh

Applies to: Single-page applications and multi-premise scenarios. Not required for static sites that reload the page on account switch.

In multi-premise and multi-OpCo scenarios, customers may switch accounts during the same session. In single-page applications, account switching should happen without a page reload. After your portal completes its own account-switching logic (session updates, permission validation, navigation updates), update Opower widgets by calling `window.opowerApi.setEntityIds(...)`.

```
function switchToAccount(newCustomerId) {
    // Execute utility-specific account switching logic first
    // (for example, update session, validate permissions, etc.).
    if (window.opowerApi) {
```

```
        window.opowerApi.setEntityIds(entityIDsArray);
        // Note: opower:start will not fire again during this session.
    }
}
```

Implementation Note: The `setEntityIds` method accepts an array of exactly one value:

Single-OpCo: `['ABC-123']`

Switching Accounts for Multi-OpCo Portals

Applies to: Multi-OpCo portals and single-page applications (recommended). Not required for single-OpCo portals.

For multi-OpCo portals, the best method for switching accounts depends on the utility website implementation. Utilities may use the Billing Account Selector widget provided that your widgets are embedded in a single page and the area would not be affected by the site's billing account selector. The Billing Account Selector enables reliable account switching, supports browser redirects, and provides tight integration with other Opower widgets (including automatic customer data re-fetching upon account selection).

Utilities can also use their custom account selector and call `window.opowerApi.setEntityIds()` for multi-OpCo switching, similar to the Providing a Single Customer or Account ID (Multi-Premise). However, a page refresh may be required depending on the implementation.

Implementation Note: The `setEntityIds` method accepts an array of exactly one value:

Multi-OpCo: `[{ accountId: 'ABC-123', clientCode: 'UTIL' }]`

Supporting Cross-OpCo Authorization with OIDC UserInfo (Example Structure)

Applies to: OpenID Connect integrations and multi-OpCo portals. Not required for non-OpenID Connect authentication models or single-OpCo portals.

For utilities operating multiple subsidiaries or OpCos under a single parent site, the OpenID Connect UserInfo response should be structured to return all accounts that the customer is authorized to access, each tagged with its associated operating company.

```
{
   "sub": "456aae91-90bf-4ef4-9e1b-cc46e38a265a",
   "user_accounts": [
      { "id": "965203814700-0031002578", "opco": "UTIL" },
      { "id": "203184955371-4067812994", "opco": "epeo" },
      { "id": "710298357401-9840361257", "opco": "tanc" }
   ]
}
```

- The `sub` field represents the unique identifier for the user.
- The `user_accounts` array lists all accounts available to the user, with each entry including an account `id` and an `opco` code for the operating company.

Providing Access Tokens for OpenID Connect

Responding to Authorization Challenges

Applies to: OpenID Connect integrations. Not required for pre-authenticated experiences (do not include this listener).

When authenticating using OpenID Connect, provide an access token by listening for and handling the `opower:unauthorized` event. Oracle uses the access token to issue a `GET` request to the UserInfo endpoint and retrieve the user's account details.

Embedded widgets require access tokens when they are initially loaded on the page, as well as when the widget requires the access token again but the token has expired.

```
// Synchronous handler example
window.addEventListener('opower:unauthorized', function(event) {
  var authorize = event.detail.authorize;
  var authorization = {
   accessToken: 'ABC123XYZ' // The access token string as issued by the
authorization server
  }
  authorize(null, authorization); // Do not call if user is logged out
}
```

For asynchronous handlers:

```
// Asynchronous handler example
window.addEventListener('opower:unauthorized', function(event) {
  var authorize = event.detail.authorize;
  event.preventDefault(); // to instruct Opower authorization logic to wait
for async callback
  fetchOpowerAuthorization(function(error, authorization) { //
fetchOpowerAuthorization is an example utility defined function
    authorize(error, authorization);
  }
}
```

Specifying Script Loading Order (OpenID Connect)

Applies to: OpenID Connect integrations. Not required for pre-authenticated experiences (where the listener is excluded).

Place the `opower:start` in the correct order relative to the `core.js` script tag, which is the main script for our core library. For more information, see Specifying Script Loading Order above.

Optimizing Widget Load Performance

Delaying Widget Initialization

Applies to: Multi-widget pages and single-page applications. Not required for single-widget pages where immediate widget load is acceptable.

By default, embedded widget content begins downloading and executing immediately. If widgets are not the primary focus of the page, you can improve perceived page load performance by delaying widget initialization until the customer opens the tab or accordion containing the widget.

```
function addInitializeTabListener(tabId, tabContentId, html) {
  var tab = document.getElementById(tabId);
  var tabContent = document.getElementById(tabContentId);
  tab.addEventListener('click',
    function() {
      tabContent.innerHTML = html
    },
    { once: true }
  )
}

addInitializeTabListener(
  'widget-data-browser-tab',
  'widget-data-browser-content',
  '<h3>Data Browser</h3><opower-widget-data-browser></opower-widget-data-
browser>'
);

<div id="widget-data-browser-tab">
  <div id="widget-data-browser-content">
    <!-- On click of the tab, the Data Browser widget is inserted here -->
  </div>
</div>
```

To extend this pattern to a second widget (for example, Neighbor Comparison):

```
addInitializeTabListener(
  'widget-neighbor-comparison-tab',
  'widget-neighbor-comparison-content',
  '<opower-widget-neighbor-comparison></opower-widget-neighbor-comparison>'
);
```

7

Testing an Embedded Widget

The utility can view the embedded widgets with fixture data by appending `fixtures=true` as a query parameter definition in the `<script>` tag for any webpage that includes embedded widgets. Enabling fixtures removes the authentication process and displays the widget content with example data and configuration. This configuration applies to an entire page and thus applies to all widgets embedded on the page. This type of testing is helpful for viewing a widget in a web environment ahead of full widget configuration and authentication setup. Examples are shown below.

> **Warning**
>
> Fixture-based testing is not a replacement for more comprehensive testing procedures. It is recommended to test with a test user account available in the Oracle Utilities Opower data set, or once single sign-on setup is completed.

Stage Environment Example

```
<!DOCTYPE html><html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge">
 <title>Page Title</title>
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <script src="https://ei-util-stage.opower.com/ei/x/embedded-api/core.js?
fixtures=true"></script>
</head>
<body>
 <div>
 <opower-widget-neighbor-comparison opower-instance="widget-neighbor-
comparison"></opower-widget-neighbor-comparison>
 </div>
 <div>
 <opower-widget-data-browser opower-instance="widget-data-browser"></opower-
widget-data-browser>
 </div>
</body>
</html>
```

Production Environment Example

```
<!DOCTYPE html><html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge">
 <title>Page Title</title>
```

```
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <script src="https://util.opower.com/ei/x/embedded-api/core.js?
fixtures=true"></script>
</head>
<body>
 <div>
 <opower-widget-neighbor-comparison opower-instance="widget-neighbor-
comparison"></opower-widget-neighbor-comparison>
 </div>
 <div>
 <opower-widget-data-browser opower-instance="widget-data-browser"></opower-
widget-data-browser>
 </div>
</body>
</html>
```

8

Contact Your Delivery Team

Your Oracle Delivery Team is the group responsible for setting up, configuring, launching, or expanding your Oracle Utilities Opower program. Contact your Delivery Team if you have any questions about your program products and implementation.

To contact your Delivery Team:

1. Sign in to Inside Opower (https://inside.opower.com). This is your portal for questions and information related to your program.
2. Go to the Community tab to see who is on your Delivery Team.
3. Contact any of the team members using the information provided.

If you need to report an issue or get technical support, contact My Oracle Support.