HPXML Implementation Guide Documentation

Release 2.0

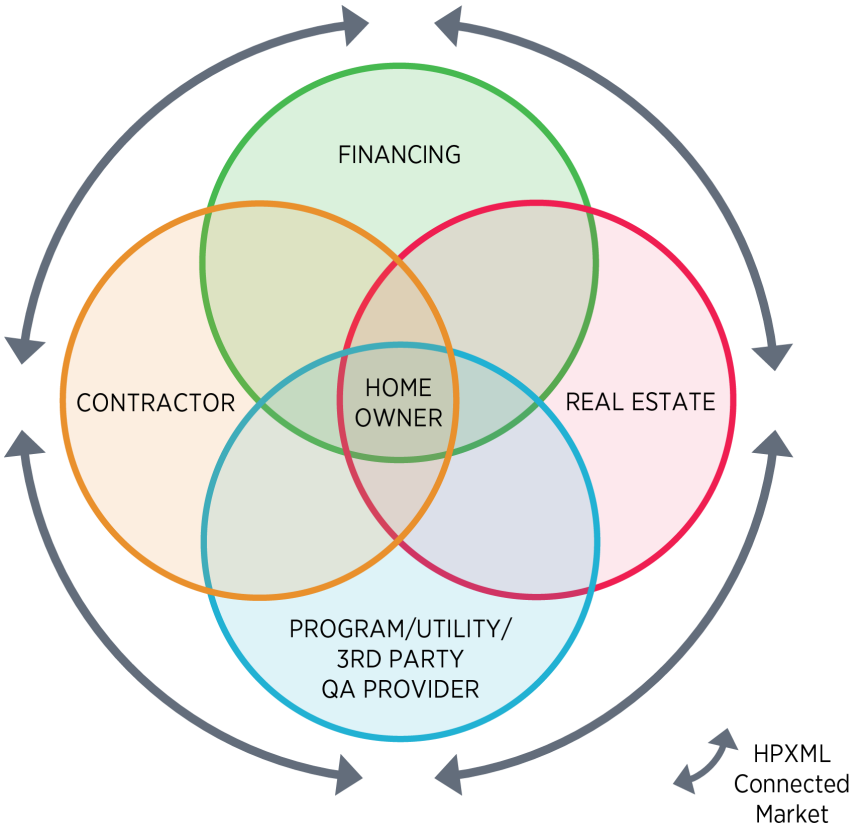
Noel Merket (NREL)

November 30, 2015

Contents

The Home Performance with ENERGY STAR Team developed the HPXML Implementation Guide to help program administrators and software developers integrate HPXML into their operations and products.

HPXML is a set of common definitions, based on Building Performance Institute's BPI-2100 and BPI-2200 data standards, for the attributes of the systems in a home and the computing language that facilitates the quick and easy transfer of home-related data between different market actors.



Today there is a fragmented, siloed marketplace where exchange of data occurs but with the non-uniform definitions for metrics and lack of two-way feedback systems. With the use of HPXML, we will see the easy exchange of data between and among different market actors.

CHAPTER 1

Contents

1.1 Introduction

An organized group of members of the home performance community with the support of the private, utility, and public sectors, conceived of and developed the Building Performance Institute's (BPI) BPI-2100 and BPI-2200 data standards commonly referred to as ("HPXML"). To expedite its deployment into the market, Home Performance with ENERGY STAR® developed this HPXML Implementation Guide to help program administrators and software developers integrate HPXML into their operations and products. Simply put, HPXML is a set of common definitions for the attributes of the systems in a home and the computing language that facilitates the quick and easy transfer of home-related data between different market actors. Without HPXML, home improvement contractors cannot easily exchange data with partnering businesses, energy efficiency programs, the real estate market, or the financial sector.

As conveyed in Figure 1 below, today there is a fragmented, siloed marketplace where exchange of data occurs but with non-uniform definitions for metrics and a lack of for two-way feedback systems. Figure 2 represents a market where HPXML has been successfully integrated which facilitates the easy exchange of data between and among different market actors. While each market actor would have specific uses for some data, other data could be shared creating more value for residential energy efficiency as a whole ultimately translating into greater energy savings from a more efficient marketplace.

It is the expectation of the U.S. Department of Energy that expanded use of HPXML will achieve the following:

- Reduce time and cost of collecting and transferring home and energy-related data;
- Foster new and strengthen existing organizational relationships within the residential supply chain;
- Increase the transparency of energy efficiency work tto facilitate deeper market penetration of energy efficiency products and services;
- Enhance ability to quantify energy savings through standardized, data-rich EM&V methods;
- Improve the quality assurance systems and practices needed to efficiently support, measure and verify energy performance.

Significant effort has gone into the development of HPXML and this Implementation Guide, but the work is not complete. HPXML is constantly being improved and this Implementation Guide will benefit from the lessons learned by different organizations integrating HPXML into their operations. If you have questions about HPXML or believe you can contribute to the overall success of its deployment, please email us at homeperformance@energystar.gov. HPXML was developed and deployed through a collaborative process, and we intend to continue in that spirit to expand its execution and improvement.

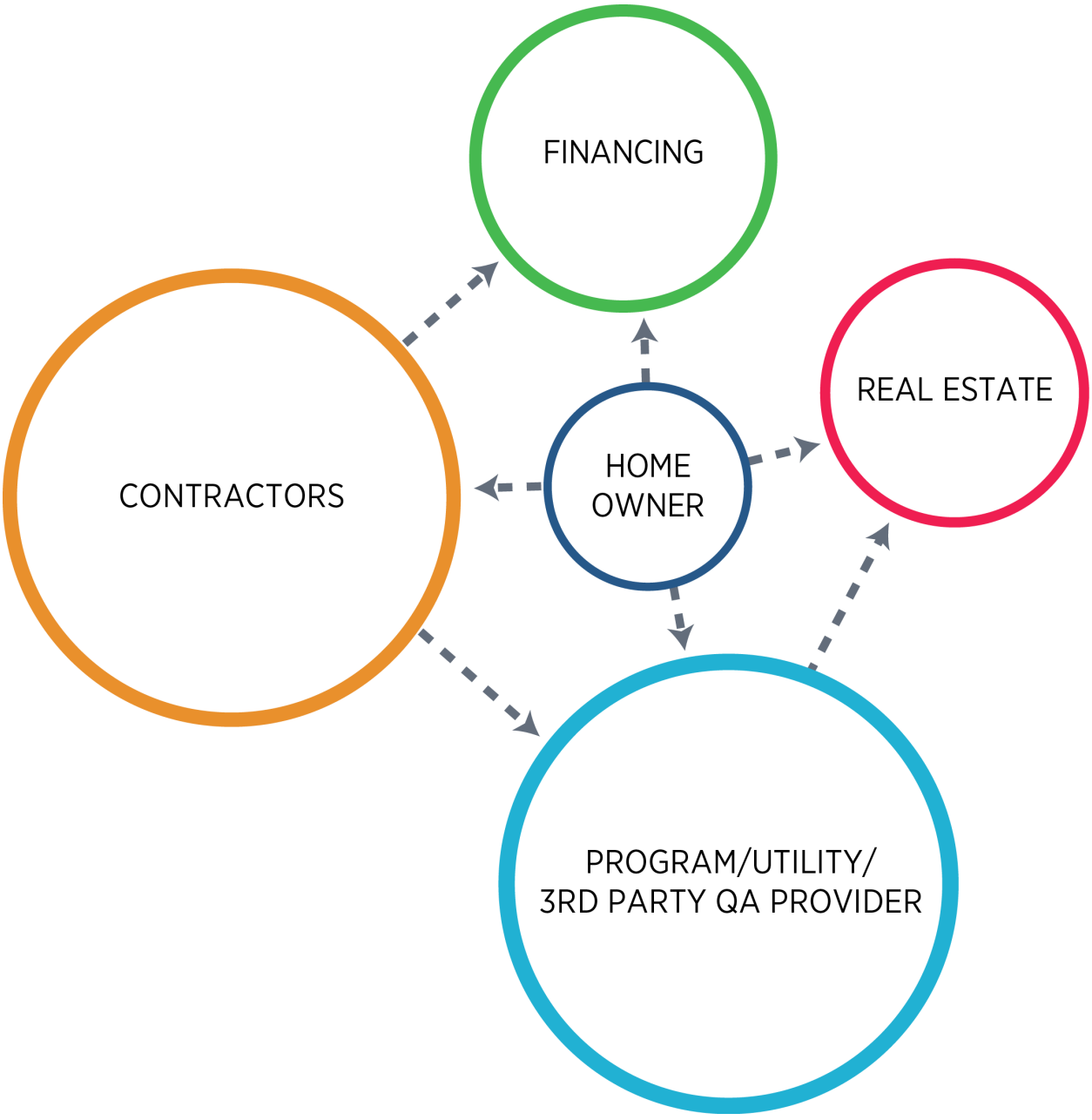


Fig. 1.1: Figure 1: Existing Market Conditions

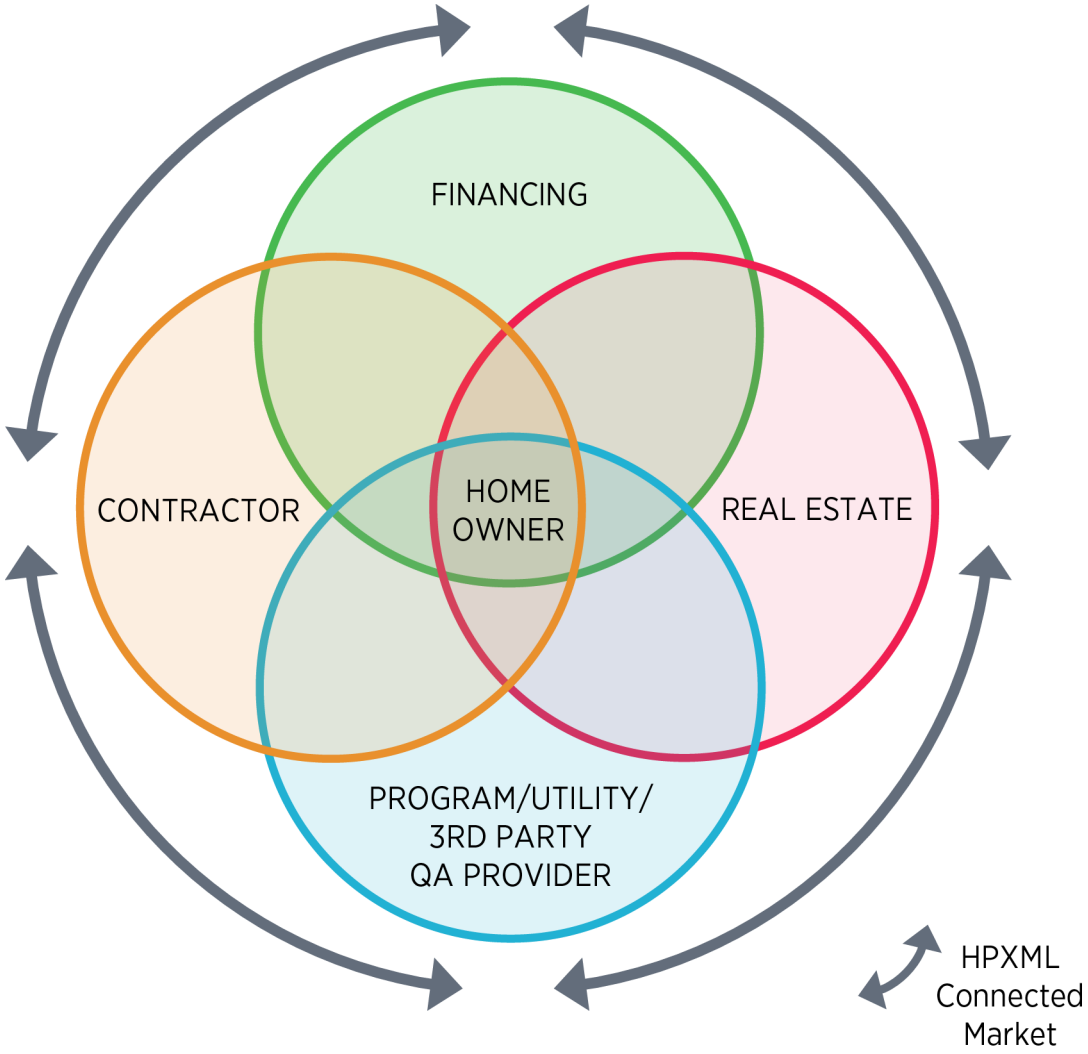


Fig. 1.2: Figure 2: Connected Market

1.2 Data Standards Ecosystem

1.2.1 What are the BPI Data Standards?

The BPI data standards are designed to help collect and transfer information about the characteristics of residential buildings and upgrade projects. This data can be shared with any party involved in a home performance or energy-efficiency program, including contractors, program administrators, utilities, federal agencies, and others. In addition, HPXML allows home performance data to be seamlessly shared with the financial, real estate, insurance, and other appropriate industries. This creates a standardized way to collect project information and to share that information to facilitate market growth.

The two primary data standards, BPI-2100 and BPI-2200 are closely related and are used together to implement HPXML:

BPI-2100-S-2013: Standard for Home Performance-Related Data Transfer provides requirements for an extensible mark-up language (XML) standard data transfer protocol that can be used to transfer home performance-related data.

BPI-2200-S-2013: Standard for Home Performance-Related Data Collection provides a standard vocabulary for describing terms related to buildings, energy consumption, and energy conservation measures. Each of the data elements defined in BPI-2200 can be transferred using HPXML via the methodology outlined in BPI-2100.

BPI will be developing additional data collection standards, of which BPI-2101-S-2013 has been completed. Each of these standards specifies a set of data elements to be collected at specific points during the implementation of an energy efficiency program (e.g. the audit, project completion, etc.). While HPXML is capable of transferring a wide variety of data elements, any one program or jurisdiction will only need a small sub-set in most cases. To facilitate adoption of the data standards and to reduce variation between jurisdictions, which can be costly for software providers and contractors, BPI WG-5 in coordination with stakeholders from across the industry have started developing the following data collection standards:

BPI-2101-S-2013: Standard Requirements for a Certificate of Completion for Residential Energy Efficiency Upgrades This standard identifies a set of data elements for certificates that document the completion of a whole-house energy upgrade (HEU) and individual energy conservation measures (ECMs). The set of data elements required for inclusion in the certificate will provide a clear, easy-to-understand description of the HEU or ECMs, including information about major energy-related improvements implemented and, if relevant, predicted energy savings or other performance indicators. The certificate is designed to be used as a reference document by real estate agents, appraisers, and other professionals during the home sale process, and can be uploaded into Multiple Listing Service (MLS) databases. Home Performance with ENERGY STAR's Certificate of Completion template, available to Program Sponsors, complies with BPI-2101 to assure consistency of data elements.

Draft: Standard for Home Energy Auditing Data Collection with Energy Modeling Tools This draft standard represents the minimum data necessary for energy modeling tools to capture during an energy audit and transfer by HPXML to home performance programs. The standard is designed to promote consistency between programs and facilitate cost efficiency in the implementation of HPXML through alignment to a standardized specification.

Draft: Standard for Home Energy Job Completion Data Collection with Energy Modeling Tools This draft standard represents the minimum required data necessary to be captured by energy modeling or other data collection tools during a home energy upgrade and transfer by HPXML to home performance programs. The standard is designed to promote consistency between programs and facilitate cost efficiency in the implementation of HPXML through alignment to a standardized specification.

1.2.2 The Role of BPI Working Group 5

BPI Working Group 5 (BPI WG-5) is a volunteer effort with representation from program administrators, implementers, software developers, and government agencies. The group is an official Building Performance Institute committee tasked with developing and maintaining standards related to the collection and transfer of energy efficiency and home performance-related data. BPI WG-5 is responsible for maintaining the standards, and it works to ensure that the standards can meet the needs of various market actors. As a program administrator, you may want to familiarize yourself with this group, as they can serve as an invaluable resource to assist in the implementation of the BPI data standards, facilitate changes in the standard, and answer questions about the standards. To contact the BPI WG-5 or to get involved, email them at hpxml@homeperformance.org.

1.2.3 Data Standards and the U.S. DOE's Building Energy Data Exchange Specification (BEDES)

The BPI data standards were developed to serve the single family residential sector (i.e. 1-4 unit buildings). Since the creation of HPXML, and with shared objectives, the US Department of Energy has begun the development of the Building Energy Data Exchange Specification, referred to as BEDES (pronounced "beads" or /bi:ds/). BEDES is a dictionary of terms, definitions, and field formats which was created to help facilitate the exchange of information on building characteristics and energy use. It is intended to be used in tools and activities that help stakeholders make energy investment decisions, track building performance, and implement energy-efficient policies and programs.

Since the initiation of the BEDES effort, BPI WG-5 and representatives from the DOE have coordinated efforts to maintain interoperability between the two standards. The primary difference between the two efforts is one of scope, as the BPI data standards are designed for single-family residences, while BEDES is designed to include all building types — both commercial and residential.

For more information on visit the BEDES website.

1.3 Program Administrator Guide

1.3.1 Opportunities and Benefits for HPXML Adoption

HPXML was created to achieve some basic goals that are critical to a growing industry:

1. Standardize terminology and facilitate the collection of higher quality data as a means to track and quantify the work being completed across the residential energy-efficiency industry.
2. Create interoperability between software systems to allow the transfer of home performance data across a diverse set of market actors.
3. Improve industry efficiency by reducing the costs of data collection and exchange between market actors.

Once adopted, HPXML has the potential to dramatically lower costs in a wide range of areas as software vendors can use off-the-shelf solutions that they have already developed, rather than building new systems for each client. If this is done in coordination with other programs, utilities, and governments implementing HPXML, it can not only lower operational costs, but provide access to a wide range of new opportunities to help grow the local energy efficiency marketplace.

To illustrate the potential opportunities created by HPXML, Figure 3 offers a visualization of data flow for most program environments today, pre-HPXML. This is compared to a post-HPXML marketplace (Figure 4) in which home energy upgrades are captured and transferred in a more standardize way and available to a larger number of market actors. This new data flow will facilitate emerging opportunities in the valuation of energy efficiency and the availability of financing for home energy upgrades.

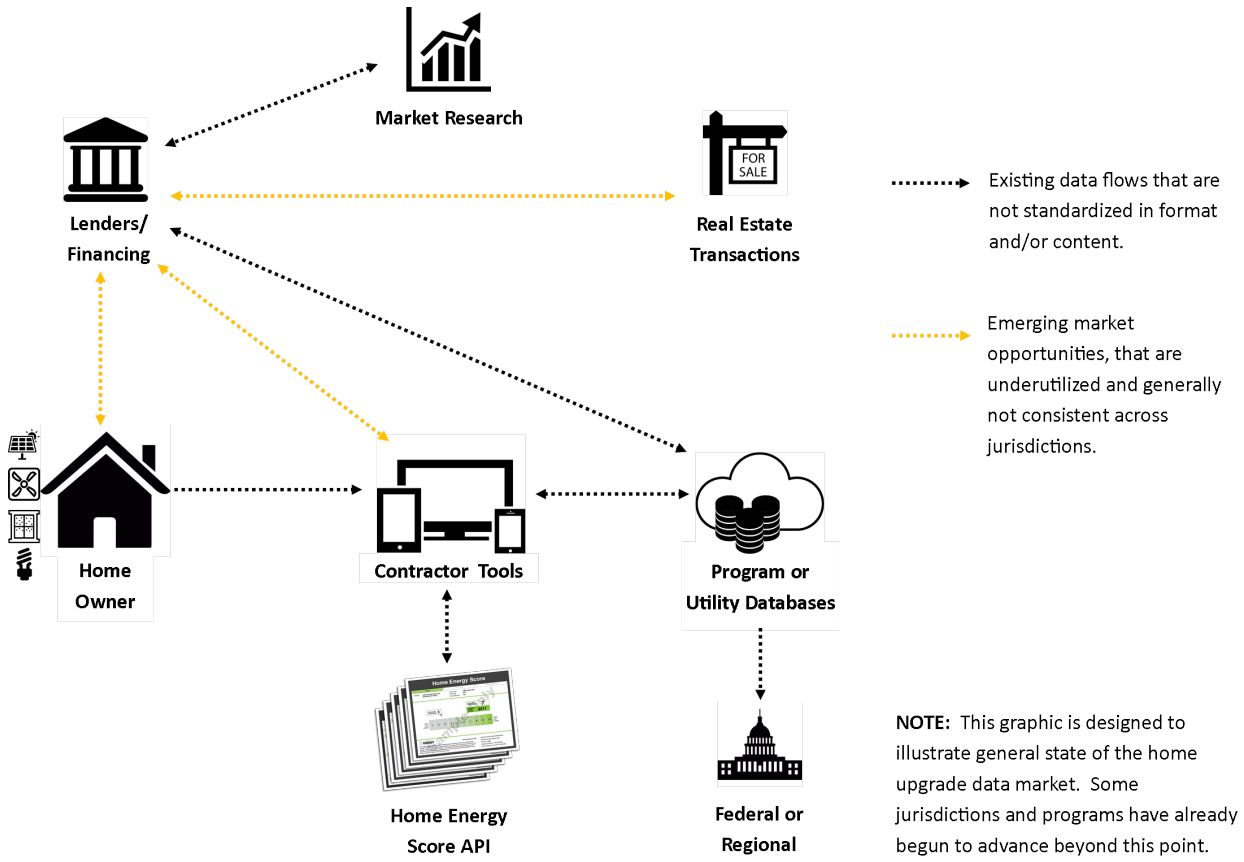


Fig. 1.3: Figure 3: Data flow of existing markets

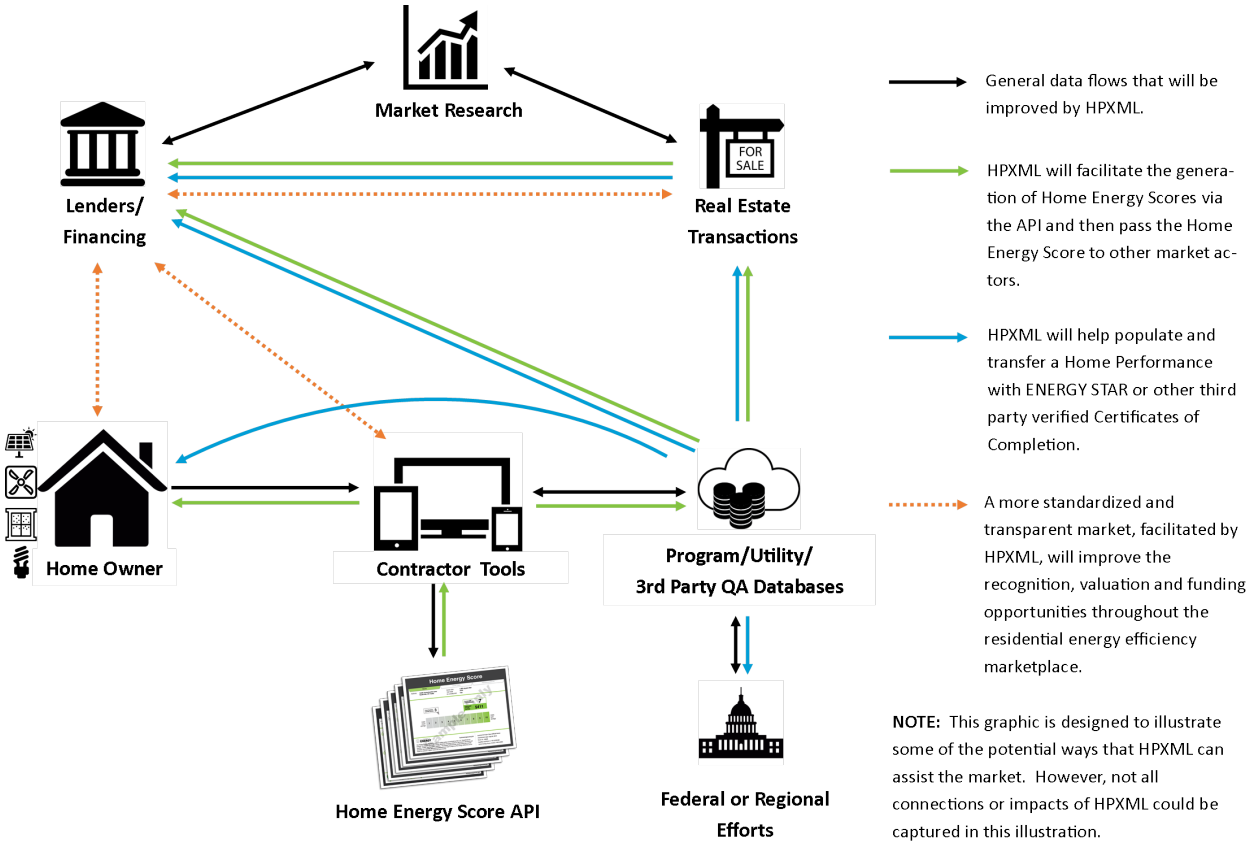


Fig. 1.4: Figure 4: HPXML connected market data flow

While these opportunities are expansive, it is important to identify priorities that are right for your program or jurisdiction and to stage adoption of these initiatives over time.

To help better clarify some of the opportunities that are available and encourage getting the most out of HPXML, several ideas have been identified at the trade ally, program and national levels for your consideration. These opportunities are as follows:

Trade Ally and Contractor Opportunities

Participation in home performance or other energy-efficiency programs can help contractors increase access to financing or incentives and align with marketing and education efforts to promote their business. HPXML can let contracting companies develop or procure systems that ease the burden of reporting to programs, while improving the process for sales and customer engagement. Some of the opportunities for contractors include:

- **Reduce Data Collection Burden for Program Participation.** Many of the HPXML-compliant software tools offer contractors a more streamlined modeling or data collection experience. Contractors can choose tools that better align with their business models, improving primary data collection and significantly reducing the need to fill out rebate forms, retype data collected in the field and streamline job processing. For example, based on a survey of contractors before and after the implementation of their HPXML based program, contractors participating in Arizona Public Service's Home Performance with ENERGY STAR program, reported an average 31% reduction in administrative time after switching to the new program system.
- **Improved Sales Tools.** HPXML-compliant software tools offer a wide range of sales tools that can enable contractors to offer a more customizable experience aligned with their customer's needs. This can help increase conversion rates and assist in the sales process to acquire larger projects and more savings.
- **Improve Business Analytics and Quality Management.** Standardization of performance data through HPXML enhances the opportunities to integrate this information into a contractor's business systems. Using HPXML data, contractors are evolving the way they can track performance drivers within projects to improve the delivery of internal quality management systems. Additionally, enhanced tracking and trend analysis can help contracting companies better identify key drivers that impact profitability, customer satisfaction, and the performance of their products. These efforts can help grow the business and deliver more consistent and reliable performance.
- **Acquiring New Investment Resources.** As better data on energy efficiency improvements is transparent and easily shared within the market, the opportunity to recruit new financing resources increases. This can help grow the market while reducing the dependency on program funding to facilitate that growth.

Program, Utility and Local Governmental Jurisdictional Opportunities

Faster and more flexible tools can lower program costs, increase trade ally satisfaction, improve data quality, and drive innovation into programs. Working together with local contractors, an HPXML implementation will not only yield benefits to the program, but can assist in making program efforts more transparent to the larger market of homeowners. The following are some of the opportunities for HPXML that are being used by programs today:

- **Create an Open Modelling Software Market.** HPXML allows participating contractors to choose which soft- ware tool they want to use for energy modeling. Implementing HPXML can streamline the process for contractors to submit consistent information to the program using a variety of tools. This can improve contractor satisfaction and reduce their labor per project by allowing them to choose the products that work best with their business model. Adding this flexibility in the program can be an important step toward accelerating the energy efficiency marketplace, by encouraging the growth of innovative tools and approaches.
- **Adoption of Mobile Data Collection Apps.** The ability to collect data on a mobile device can save time and reduce reporting errors. By coding these apps to HPXML, the data can be exported and stored in a standardized format that aligns with national reporting requirements and can be easily integrated with other software products. For example, if you are administering a low-income weatherization program, field staff can us HPXML based

mobile applications that sync to a central program database. Once in a central database the tools can be used for reporting, quality assurance, and data analysis across jurisdictions. This can reduce cost, but more importantly make it easier to track performance drivers against other HPXML jurisdictions.

- **Streamlined Quality Assurance (QA).** As data becomes more standardized and consistent across multiple jurisdictions, software developers are creating more advanced QA systems to evaluate incoming data and guide quality assurance activities. As desk reviews of projects for compliance with incentives and program rules become more automated, the more quality can be maintained while reducing labor costs to manually review files. For one of the pilot programs, Arizona Public Service, time spent completing desk QA reviews decreased by as much as 50% per project. In many implementations, programs are pursuing automated approvals that can allow contractors to seek incentives or financing approvals while they are still on-site with the customer and further reduce the program administration costs per project while improving assessment-to-upgrade conversion ratios.
- **Improved Program Analytics.** Improving both the quality and consistency of the data collected allows programs to complete comprehensive, ongoing analytics. HPXML data extracts can speed up or enhance measurement and evaluation efforts, while making it easier to compare program performance characteristics with other programs. For example, if multiple programs use HPXML and one of the program is showing better results than the others, it will be easier with HPXML to identify trends in your program that are attributed to the performance differences. This can then be compared to other programs who are tracking the same data, to better identify and share best practices.
- **Targeted or Trigger Based Marketing.** HPXML can help programs capture details about a home's characteristics using online or in-field audit tools. Integrating HPXML data into marketing analytics can help identify customers with a higher likelihood for savings or participation. This data can then be used to guide ongoing marketing investments by providing recommendations to customers that are most meaningful for their home and facilitate staging upgrade activities over time.
- **Export for Local Real Estate.** Coordinating with local real estate and appraisal networks can identify data points and export strategies that work for those markets. This could include publishing completion certificates, providing the data to populate the Appraisal Institute's Green and Energy Efficient Addendum or directly export data to the local Multiple Listing Service. Doing this can help increase the value of energy efficiency in the real estate transaction and provide greater incentive and investment opportunity for homeowners to fund energy efficiency upgrades.

Regional and National Opportunities

Collecting standardized data sets across multiple programs and jurisdictions is a critical step in helping guide national policy, financial investments, and research. Adoption of HPXML commands higher quality data and transfer of that data to a variety of actors in a cost-efficient manner:

- **Reporting to DOE or EPA Administered Programs.** Whether participating in Home Performance with ENERGY STAR, Low Income Weatherization, or as an EPA ENERGY STAR Partner, HPXML can lower the cost and burden of data collection and reporting.
- **Regional or National Data Repositories.** Managed by federal agencies, trade ally associations, or other national organizations committed to the advancement of energy-efficiency investments, these repositories can organize data to track and quantify impacts of energy efficiency. Efforts like the Building Performance Database Website or the Standard Energy Efficiency Data Platform will not only make it easier for programs to collect and share data, but can allow the industry to compare results, improve transparency, and better identify performance drivers for program success.
- **Coordination with Market Research.** National research efforts, whether lead by the Department of Energy, universities, industry organizations, or private companies, have struggled to get consistent and reliable data from multiple jurisdictions. By adopting HPXML, users can more easily participate in these efforts and gain valuable insights into a program's performance and the opportunities for emerging technologies in markets.

Beyond data collection and reporting, there are a number of national efforts that the interoperability of HPXML can facilitate. While new efforts are continually emerging, the following are some examples of current or emerging opportunities:

- **DOE Home Energy Score.** This tool allows homeowners to compare the energy performance of their homes to other homes nationwide. By collecting data in the field with an HPXML-compliant tool and transferring it to DOE's application program interface (API) a Home Energy Score can be generated in real time, and minimizes integration costs.

  The seamless integration of Home Energy Score through HPXML can make it easier for contractors and auditors to generate this simple 1 to 10 scale for your customers. This easy to understand score, combined with the standardized collection of data across Home Energy Score partners and jurisdictions can assist in raising the transparency of energy efficiency upgrades within your local market. As more homes are scored, this information should be shared with the real estate community and included in general consumer education campaigns to improve awareness, drive the value of energy efficiency, and increase project volume.

  For more information visit Home Energy Score.

- **EPA ENERGY STAR Home Advisor.** Create a profile of home energy efficiency features to receive a prioritized, customized list of energy-saving recommendations. Programs that complete on-line or in-person energy audits with HPXML-compliant tools could allow customers to upload their recommendations, track their progress over time, and receive other energy-saving information from the EPA. For more information go to the Home Energy Advisor.

- **Multiple Listing Services.** Being able to accurately track energy efficiency improvements and make them known to potential home buyers is a critical step in facilitating recognition of energy efficiency in the valuation of homes. The HPXML effort has been coordinated with Real Estate Transaction Standard (RETS), which is a national data standard from the Real Estate Standards Organization. The multiple listing services, which represent the backbone of that real estate information industry, use RETS to maintain consistent data. For more information, see BPI-2101-S-2013 Standard Requirements for a Certificate of Completion for Residential Energy Upgrades.

These are several of the recognized opportunities that have emerged to date. While others are being created all the time, it is recommended to start with those that are most relevant to your jurisdiction and incorporate them into your implementation plan as described in the next section of this guide.

1.3.2 Implementation Guide

To realize the benefits outlined above, a well-designed implementation plan is necessary. The following sections will provide program administrators with a walkthrough of best practices in implementing HPXML and provide guidance that can serve as a starting point for an implementation plan in your jurisdiction.

Each jurisdiction will have different goals, market needs, and regulatory requirements that will influence the scope of the HPXML projects. However, leveraging the efforts other market actors who have already implemented HPXML can lower individual implementation costs and will help to drive alignment between jurisdictions. The following "steps to HPXML implementation" are designed to help leverage existing efforts, while also making sure that individual jurisdictional needs are met.

Steps to HPXML Implementation

1.3.3 Step 1: Setting Implementation Goals

HPXML can drive a lot of value into a program. The key to delivering this value is to determine the goals and scope of a HPXML project as early in the process as possible. To date, many programs have used HPXML as a means to open the modeling software market in their jurisdiction to multiple vendors. Others have used HPXML as a way to increase their ability to share information with other market actors, such as stakeholders in the real estate market. Regardless


Step 1
• Setting Implementation Goals
Step 2
• Incorporate Stakeholder Feedback
Step 3
• Identify Data Needs
Step 4
• Procuring or Modifying Program Management Systems
Step 5
• Designing a Data Validation Process
Step 6
• Implementing Testing Protocols and User Training
Step 7
• Develop a Quality Management Plan

of your plans, it is important to clearly define your objectives early, as these decisions will shape your implementation tactics.

To start, review each of the bullet points in the Opportunities and Benefits for HPXML Adoption section. Highlight which opportunities you would like to take advantage of in your program implementation. For each opportunity, identify the following information:

- **What is the baseline now and where do you want to be?** For example, if you want to "reduce the data collection burden for program participation," you will need to know how much it currently costs contractors in time or money to participate in the program, and work with them to identify a goal.
- **What are some of the current pain points?** For example, are there specific elements of your current program implementation that drive customer or contractor dissatisfaction? Likewise, are there any procedures in your process that create opportunities for error or mistakes?
- **How will this help achieve overall program goals?** For example, what is your current number of projects per year and what would you like to increase it to? If you are focused on volume, your priorities might lie in HPXML efforts designed to increase volume, like integration with Home Energy Score or coordination with your local real estate market.
- **Are there local rules or regulations to consider?** For example, utility program regulators may require specific project details to be reported. You should coordinate with your local experts in these fields to help identify potential issues that will need to be addressed.
- **Which aspects of our overall goal should be prioritized?** You can achieve a lot with the data standards. However, as with many IT projects, it may make sense to implement in phases. Prioritizing the different components of your efforts and addressing the most important ones first will improve budget and resource management, as well as change management. Coordination with trade allies and contractors is key, and identifying how much change to institute at once can help reduce the pressure on the industry.
- **What is a realistic timing and rollout schedule?** The balance of the implementation document will help to identify the tasks you must complete to rollout HPXML. Give yourself plenty of time to complete these tasks and, if possible, plan your testing and launch schedule in the low season, in order to reduce contractor burden.

By thinking strategically through the questions outlined above and identifying of which opportunities you want to take advantage, you should be prepared to design the balance of your implementation plan.

1.3.4 Step 2: Incorporate Stakeholder Feedback

HPXML can be a powerful tool to facilitate trade ally/contractor's choice in software systems, while delivering consistent and reliable data. It provides a great opportunity to improve contractor satisfaction and operational efficiency. To maximize your potential for success, it is important to establish a consistent means to collect feedback from both trade allies and the software community throughout the HPXML process. To start you may want to collect feedback and cultivate support for the HPXML goals you have established for your program in *Step 1: Setting Implementation Goals*.

Trade Ally Feedback

Consider working closely with your local trade ally network, trade associations, or energy-efficiency contracting network to collect feedback important to your planning process. For example, some pilots have hosted lunch and learning meetings before their design process began. This allowed the program administration team to identify the highest priorities for trade allies and cultivate support for the new program design. Ultimately, this helped focus the project scope to deliver successful results.

When engaging in this process, consider a few key topic areas:

- Pain points for data collection and reporting

- How program goals will impact contractors' businesses
- Priorities for program design and subsequent rollout
- Early adopters who can assist in testing and give feedback on initial designs
- Functionality and user experience desired from the new program software environment

Identifying a representative group of trade allies that can assist throughout the process will be key in later steps of the implementation plan. Many successfully implemented programs have coordinated with contractors by setting up regular meetings, hosting dinners to collect feedback, or working closely with representatives from local trade associations.

Software Vendor Coordination

As the program administrator, you need to decide on the minimum qualifications for software to participate in your program. It is recommended to begin communication with software vendors as soon as your project plan is developed. BPI WG-5 can help connect you to all of the software vendors that are currently coding to HPXML. Since many of these vendors have implemented in other jurisdictions, they can offer valuable support for a program's planning process.

Programs that are considering opening the market to multiple modeling software tools, may want to review the varying features of the different HPXML-compliant energy modeling software tools. Taking a demo from each of the vendors is a good place to start to help gain a general understanding of how their tools work and differ from each other. Each vendor takes a unique approach, which promotes innovation and helps contractors deliver high-quality proposals and energy savings feedback to customers. Reviewing how each of the software tools work will give your program in understanding of what the market is transferring now via HPXML. This will enable your program to be well prepared for the next step of determining data needs.

1.3.5 Step 3: Identify Data Needs

Contents

BPI-2200 defines a long list of data elements, many of which may not be needed for your project. To determine which data points are needed, start with the responses you developed in Step 1: Setting Implementation Goals. The goal of this exercise is to identify the minimum data collection requirements needed to meet project goals. This means identifying what must be collected in the field, what must be transferred to your program management system, and

what you are transferring/reporting to others, when applicable. The good news is that BPI WG-5 has coordinated with multiple programs across the country to develop standard use cases that can serve as a great starting point for most programs.

Standard Data Sets Developed by BPI WG-5

When exporting or importing data with third-party data systems, it is important to develop a concise set of data requirements. As stated above, the BPI data standards are capable of describing and transferring a large universe of data, only a portion of which is needed for any one use case. To facilitate efficiencies between programs, BPI is in the process of developing standard data sets, each of which is associated with a specific use case. Use cases (defined here as a set of interactions designed to achieve a specific function within an energy efficiency program) that BPI WG-5 has identified as particularly important for energy-efficiency programs include:

1. Reporting on the baseline conditions of a home and proposed improvements
2. Reporting on the improvements to the home compared to baseline conditions
3. Reporting on health and safety testing
4. Reviewing a contractor's work as part of a QA process
5. Reporting a home's energy efficiency assets to the parties in real estate transactions
6. Reporting data on program activity to DOE and other agencies

The standard audit and upgrade data sets are data collection and transfer requirements that were created by existing HPXML adopters to facilitate the transfer of project data from third-party energy modeling tools to a program management database. As a collaboration between Home Performance with ENERGY STAR programs in Arizona, California, New York, Vermont, and Virginia, the standard use cases are well suited for a wide range of program types, climate zones, and implementation models.

Although these standards are being developed for Home Performance with ENERGY STAR or whole house programs, they can be used by other types of programs as well, such as weatherization of single measure programs.

Even if implementation is not for a whole home program, these use cases are still a great place to start, as several software vendors are already able to transfer the data required in these use cases. Any subset of data within these use cases will be easy to implement for existing HPXML-compliant tools.

Standard use cases also represent the required fields that implementation partners have agreed to transfer to date. These data points have been identified by the BPI WG-5 as sufficient and achievable as data collection requirements for third-party energy modeling tools. The BPI data standards support the collection and transfer of additional data points beyond these use cases. However, additional data may require software vendors to make substantive changes to their software. Program administrators should recognize the financial impacts of custom data collection requests and consider providing financial assistance to software vendors to meet any customization requests, as appropriate.

Before reviewing the standard use cases, there are several best practices to consider.

Data Selection Best Practices

Be Transparent with Stakeholders

Transparency with your contractors and software vendors throughout the process will help guide programmatic decisions and prevent challenges down the road. This is particularly true for the data selection process as stakeholders can bring attention to potential challenges that need resolution. For example, a program may request a specific data point that is not typically collected by most software products. If vendors can identify this need early on, a resolution can be reached in the planning phase and not delay the project later on.

Be Sensitive to Data Collection Burden

When selecting data requirements, programs should collect just enough data for program compliance and measurement and verification of results. As a guiding principle, Home Performance with ENERGY STAR recommends that programs employ administrative procedures that minimize the burden of participation for contractors and homeowners. When choosing what data points are mandatory for your programs, it is important to recognize that every data point collected has a cost to contractors to collect. Collaborate with evaluators, contractors and software vendors to explore solutions that meet the need of the program as cost efficiently as possible.

Leverage data choices made by other Program Administrators

HPXML is easily extensible and therefore can be customized to the specific needs of a program. That said, program administrator should look at the data sets being implemented by other programs around the country as many software company are already exporting and importing these standard data sets. When a program is using unique data requirements, the associated customization can push significant development costs for that specific program onto other parties, like software tool vendors and service providers. To streamline implementation and best leverage efforts from other programs it is encouraged to coordinate with other program administrators to minimize the number of program-specific data fields required in your jurisdiction. This guide provides a HPXML data selection tool below to easily facilitate this coordination.

Schedule Updates to Data Requirements

Software developers tend to work in phases to control releases of their software. To avoid additional costs or confusion, try scheduling regular updates once or twice a year and communicate future changes as early as possible. This will help with version control and create more manageable process for software vendors and your implementation team.

Adopting these best practices will assist in establishing a streamlined data selection process.

HPXML Data Selection Tool

To assist program administrators in reviewing the standard use cases that are being implemented by other programs, a data selection tool is available to help identify what data points are currently required in these use cases and helps programs select and communicate the requirements for the program. The HPXML Data Selection Tool is a "living document", so if a programs identifies data points that are not in the use cases, WG-5 can assist in added them to the tool and thus meet the program's needs.

- Download: `HPXML Data Selection Tool`

The attached instructional video will provide a walkthrough of the HPXML data selection tool. Using the tool programs can quickly select the data required for HPXML program implementation. Programs can then forward the tool to the implementation team, trade allies and software providers, giving them clear guidance on the requirements for HPXML Implementation with your program.

Todo

YouTube video here.

Additional details about the data structure and standard data sets are provided in the sections below.

Understanding the HPXML Data Structure

In the reviewing tool, notice the following descriptors for each data point:

Data Category A general description of the information at the building characteristic or contact information level. Note: Insulation has several data categories, depending on the insulating plane. For example, referencing insulation installed on the attic floor "Attic Floor Insulation" or on the bottom of the roof deck "Attic Roof Insulation."

Data Element A specific data point or descriptor within that data category. For example, insulation material type or R-value.

Data Type How the data should be provided. For example, as a number, text, enumeration, etc.

Definition A written description of the data point and what it means. As the name of the data point is not always clear, this provides a narrative explanation of what each data point describes.

In most cases, there are several data points needed to describe any one building characteristic. For example, if you require blower door testing in your program, you will require "Air Infiltration" information. In this case, there are three data points that are required to describe an air leakage measurement, such as 2000 CFM_{50}:

"Building Air Leakage" = 2000

"Building Air Leakage Unit" = CFM

"House Pressure" = 50

This also provides flexibility to receive the same data in multiple formats. For example, air leakage could be represented in CFM50, ACH or ACH50.

Setting the Program's Data Requirement Level

The program administrator's main task in this step is to determine the data element "requirement level." This sets the minimum requirements for software tools to participate in your program. In each of the use cases, there are two requirement levels:

Required All software must collect this data point and transfer it any time it exists in a home. This usually is driven by rebate qualifications or quality assurance requirements.

Optional Not required.

The *HPXML Data Selection Tool* will allow you to see the minimum required fields that have been agreed upon by BPI WG-5 for audit and retrofit use cases. In addition, you can use the Home Energy Score and BPI – 2101 requirement toggles. By activating these toggles, you can see which fields would be required if you wanted to complete a Home Energy Score or to fill out a BPI-2101 Home Performance Certificate of Completion.

The grayed out fields are optional and represent fields that are relevant in many programs, but not required in the standard use cases. You can choose to make optional fields required in your program. However, not all software products on the market collect every possible data point. Making some of these data points "required" may restrict which products are eligible to participate, or may require you to pay software vendors to code their software for this requirement. Communication with potential software vendors is key and consider their feedback on which fields you require for your program before determining your final data requirements. BPI WG-5 can help facilitate that conversation in a constructive environment.

Some programs with a large number of measure-specific rebates are choosing to identify data points as "optional" to allow flexibility in implementation. Your program can use a minimum data collection standard that is required for every home. However, if the contractor or software vendor wants to participate in the full spectrum of rebates, they can choose to send "optional" fields that trigger a rebate payment. This allows a diverse set of software products and contractor business models to participate, without mandating that every software and contractor support the full spectrum of rebates your program portfolio may offer. If you want to choose this path, it will be important to provide clear specifications on which "optional" fields will trigger which rebate payments.

Defining Use Cases

The standard use cases provided as a part of this guide are the primary use cases that have been developed to date. Each use case only uses a fraction of the data points that the HPXML standard can support. Over time, more use cases will be developed to meet market needs.

Audit Use Case

The audit use case is designed for Home Performance with ENERGY STAR or whole house programs that require energy assessments. This use case allows auditors to submit their audit results and proposed scope of work for an eligibility review from the program. Required fields are established to help identify the home's existing characteristics, health and safety needs, recommended improvements, and associated savings predictions.

An example of an audit use case HPXML file can be found on the HPXML GitHub repository and more technical description of the audit and upgrade use case can be found in the software developer guide at Audit-Upgrade.

Upgrade Use Case

The upgrade use case is designed to facilitate the transfer of completed whole house upgrade projects, such as Home Performance with ENERGY STAR or Weatherization programs. This includes the pre-upgrade condition of the home and a description of the installed measures, as well as associated predicted savings. Required fields are established to complete a full quality assurance review of all installed measures and determine rebate or financing eligibility. The minimum requirements reflect those most common between all of the HPXML-compliant programs so far. Programs that offer more diverse rebates may need to consider changing "optional" fields to "required" in order to meet program needs.

The upgrade use case HPXML file is very similar to the audit use case. The differences are detailed in Audit-Upgrade.

Home Energy Score Use Case

The Home Energy Score use case defines the minimum data set required by the DOE's Home Energy Score tool, in order to properly generate the 1 to 10 score. These data point are clearly identified in the data selection tool. Programs interested in generating a Home Energy Score, will need to make sure that their HPXML software tool are collecting this minimum dataset.

In order to generate the score, your program software team will also need to integrate with the DOE's Home Energy Score API. HPXML can be transferred through the API and generate a Home Energy Score in real time. For more information on integration with the Home Energy Score API, see Home Energy Score in the software developer guide.

A full list of HPXML data elements that can currently be incorporated into use cases is available in the online schema documentation.

BPI – 2101 Certificate of Completion Use Case

The Certificate of Completion use case is designed to ensure that the value of energy efficiency improvements is visible in the real estate transaction. Studies suggest that buyers will pay higher prices for efficient homes – but only if they know that the homes are efficient. Programs have traditionally had difficulty in getting information about energy efficiency improvements into the real estate transaction. The Certificate of Completion use case provides a standardized framework for programs to collect and assemble data about a home that features energy efficiency improvements ranging from a single installation to a whole-house upgrade. The standard data set in this use case is aligned with both the Appraisal Institute's Green and Energy Efficient Addendum and the Real Estate Transaction

Standard, allowing data from a trusted third party (a Home Performance with ENERGY STAR or other efficiency program) to flow seamlessly to appraisers and real estate (MLS) databases.

Note: The BPI – 2101 Certificate of Completion Use Case is designed to be highly flexible and inclusive of a wide range of technologies. When implementing this use case, most contractors and software vendors will not support the full spectrum of technology to import to your program. However being able to capture the full spectrum of data point and export to other third parties, will give your program the widest range of options for interacting with the real estate industry.

How to Add Data Elements

When reviewing the data sets that are required for your implementation, it is possible to identify a data point you require that is not in one of the pre-defined use cases or the HPXML standards. If this is the case, BPI WG-5 can assist in adding the new data element and in identifying how to incorporate it into the standard. In some cases this might include adding new elements to the standard to account for data points that could be applicable across many programs. However, if the data point is truly unique to your program, WG-5 has also introduced "measure codes" that allow a code to be assigned for a specific measure in a specific program. This creates added flexibility without needing to modify the standards in all cases.

To submit a new data element for consideration, you can use the WG-5 GitHub account. This way all members can see your recommendations and address them immediately. Follow the steps below to submit additional requests if needed:

1. Sign up for a user account on GitHub.
2. Go to the HPXML GitHub issues page.
3. Click "New Issue"
4. Fill out the form to ask a question or make a request. No need to assign a person, milestone, or label.
5. Click "Submit New Issue".

Once you have defined the use case needed for your program and have identified all required fields, you are ready to proceed to the next step. Remember, this can be an iterative process. It is good to do due diligence in the planning process. However, even the best implementation plans may need to be modified as the program goes to market and a large number of homes start running through it.

Note: Schedule opportunities later in your implementation to check in on data requirements and adjust as needed.

1.3.6 Step 4: Procuring or Modifying Program Management Systems

Procurement of an HPXML-compliant program management system or modification of an existing system to be HPXML compliant is a critical juncture for every HPXML implementation. The cost and scope of this endeavor will be largely dictated by the business objective(s) established in *Step 1: Setting Implementation Goals*, *Step 2: Incorporate Stakeholder Feedback* and the data use defined in *Step 3: Identify Data Needs*. The following section will provide some best practice guidance for acquiring HPXML-compliant program management software and how to use the tools provided in this guide to assist in that process.

Build a Clear Scope

Clearly outlining your business objectives and defining the dataset you want to use are two large steps toward building an accurate scope of work to inform your procurement process. Within your business objectives, you will want to define how you want the data to work for your program. Here are some examples of items to discuss:

- Are you using the system to qualify projects for incentives or financing?
- What kind of quality assurance/quality management do you want to do?
  - How much QA automation do you want to include in the system?
  - What data does your implementation team need to review on the project and at what stages?
  - Programs may want to reference guidance from the Home Performance with ENERGY STAR Sponsor Guide (v1.5). Section 6 and Appendix F provide a description of quality management system approach, how it uses data to determine compliance and measure performance, and directions for developing a QMS plan.
- What is the workflow of your program?
- What feedback systems do you want for both contractors and participating customers?
- What type of program reporting are you doing?
- Are you exporting data to third parties? If so, what are the specifications of their systems?

If your program needs to issue a request for proposals, there are some additional best practices to consider:

- Include all of your program's incentive or financing rules. This will help the developers understand what pieces of information are most important to you and what you will be reviewing.
- Consider including narrative examples of what you are trying to accomplish with this system's functionality. Technical specifications are important, but adding commentary on how you want to use it is equally helpful.
- Ask other programs to share their RFP scopes of work. With each new program adoption of HPXML, RFPs are getting more consistent and comprehensive. BPI WG-5 can help your procurement team identify recent RFPs and connect you with a program contact.

Leverage HPXML Experience

In procuring a software provider, it is recommended that you consider their HPXML experience — including prior implementations of HPXML and participation in the BPI W-5. Many software vendors have already built in HPXML compliance or have been an active part of the HPXML process. This experience should help speed up implementation and can reduce costs.

There are also several program administrators that participate in BPI WG-5. If you have questions or would like to share their experiences, please reach out by contacting the group at hpxml@homeperformance.org.

Plan for Testing and HPXML Coordination Support

In your scope and timeline, leave time for testing and provide resources for technical support in setup and integration. The amount of coordination and testing will vary depending on your data selection and use cases. Most HPXML programs implementations to date, have not left enough time for testing, which can be a couple of months depending on the scope of the project. If not accounted for in planning, this can put the project behind schedule or place a lot of pressure on the implementation team. See the testing section below for further guidance.

1.3.7 Step 5: Designing a Data Validation Process

Designing a good system for data validation that automatically checks all submitted data is critical to ensuring high-quality data, maintaining contractor satisfaction, and streamlining quality assurance activities. The good news is that most of this work will be completed by your software vendor. However, you will want to undergo a thorough scoping exercise with your vendor to identify what types of validation checks you want your program software to complete. For example, a basic validation check ensures that all data is present and in the right format. More advanced validation

checks can ensure that data falls within an acceptable range for program compliance purposes or to guide quality assurance. For example, you may be able to automate quality assurance review on health and safety results to ensure all standards are being meet. If not met, a user can be warned and intervene on the project.

If done correctly, a good validation system can speed up your process and significantly reduce cost. Badly done, it can increase frustration within your contractor base or yield lower-quality data.

Your data validation systems should align with your business objectives. For example, if you are using HPXML for a rebate program, you will want to validate that:

- All files are in the proper HPXML format
- Data points required for rebate eligibility screening are collected
- All rebate eligibility rules are met
- Health and safety standards have been followed

Your program will probably have specific requirements that you are trying to validate against. As you develop your validation system share the requirements with prospective third-party software vendors. These software vendors will likely include these same requirements in their software validation protocols to warn contractors when a requirement has not been met.

Once validation rules have been set, it is equally important to ensure that the user's experience is optimized to minimize frustration and clearly communicate validation errors. If a contractor is not receiving clear validation error messaging and cannot resolve the issue during the upload process, this can lead to a large number of phone calls and a higher technical assistance requirement resulting in greater cost to the program. If the process is not managed properly, you can burn out your users and create contractor dissatisfaction. This, of course, is not an observation exclusive to HPXML, but a best practice in any software implementation in general.

Phasing of Validation

As you prepare your rollout schedule, consider implementing a phased validation system roll out. If you clearly define the scope of each phase and roll them out following a regimented schedule, you can greatly assist the market in adapting to the new systems, while allowing you to improve the data quality and functionality of your system over time. For example, if you are running a whole home program with incentives or rebates, here are some phases to consider:

Phase 1 – File validation and minimum data requirements

In this phase, you will want to verify that the uploaded HPXML file is in the correct structure and minimum data points required are present and in the correct format. This will allow you to get to market quickly and begin to test your systems. However, you will still need a person to review the files to ensure that the data provided meets the technical requirements of your rebate program. For example, you will validate that all required insulation data points are present, but review the installed R-value to make sure that it meets your program requirements. You will also want to add a check to ensure that no health and safety problems exist.

Phase 2 – Advanced Data Validation and Automated QA

Over time, you can begin to layer in engineering assumptions that provide automated QA or guided QA. For example, you can add a validation check to see if installed measures are consistent with standard building practices. This way, if you receive a file that claims to have an attic with R-100 a QA advisor is triggered to review the project. Using this type of validation system, you can significantly reduce your labor requirements for reviewing submitted files as you can focus your labor on probable errors or problem areas. In some cases, you may be able to get to auto-approval on a select number of projects.

Note: Over time you can add even more sophisticated systems. If you launch with a very complex validation system, there is a high likelihood that many of the initial project submissions will fail as the market is still adapting to the new program environment. If you take a phased approach, you can ease this tension and coordinate with the market to facilitate high-quality data transactions while reducing admin costs.

1.3.8 Step 6: Implementing Testing Protocols and User Training

With every new HPXML program launch, there are a number of testing activities that need to happen to maximize the opportunity for success. You should plan for at least a couple of months of testing. Work with the software community to schedule each stage of testing. Try to schedule time for your staff to focus on this activity. If you can set a clear plan for testing early on, and stick to your production schedule, this should allow you to get to market in a timely and organized manner.

There are four main testing roles, each of which requires a different group of testers:

HPXML File Testing

This is best completed between your program software provider and third-party software vendors that are exporting to your system. Ideally, your program software provider will supply third-party software vendors with a testing environment in which they can submit test files and receive direct validation error feedback. Activity will likely be between the IT experts from both your program software provider and the third-party software vendors.

Program Compliance Testing

Based on the validation protocols that you have set up, and in order to check program compliance, a group of testers that are intimately familiar with the program requirements will review files in a test environment. The goal is to see if files submitted by the third-party software vendor are being captured in your program management system and that the system is parsing the data out as expected. Usually, this is completed by someone from the implementation or QA team.

End-to-End Testing

Especially if you are using third-party modeling tools, it is important for testers from the program implementation team to run a series of test homes using each of the software systems as if they were a participating contractor. This is to check to see if data entered into each system is making it all the way to your program database as designed. In addition, you will be familiarizing yourself with the user experience and can identify potential problems before releasing to the contractor base.

User Testing

Once you have completed end-to-end testing, it is always good to identify a group of users (contractors or trade allies) who can try the system in a test environment or as a soft program launch. Using a limited number of contractors, you can identify any potential problems before full-scale release.

It is good to designate a program representative responsible for overseeing the testing process. This person should be equipped to make decisions regarding software requirements and functionality on behalf of the program. As things are discovered during the testing process, it will be important to determine what may need to be fixed before launch verses what can be addressed over time. This role will help ensure that the program requirements are being met, while keeping the project on track.

Home Energy Score Testing

The Department of Energy has developed a Home Energy Score integration testing protocol. For programs participating in Home Energy Score, the DOE will provide a series of sample homes to facilitate software testing. Your program implementation team will need to submit these sample homes as if they were a contractor or auditor in the field, much like the end-to-end testing process identified above. Then using the DOE's Home Energy Score API, submit the homes to the Home Energy Score system. Following the protocols, your program implementation team will be able to identify validation errors or unexpected outcomes in the data transfer process.

For more information on the Home Energy Scores integration testing protocols, contact the Home Energy Score support team at assessor@sra.com.

User Training and Contractor Role Out

After testing is complete, you can launch your new HPXML based systems. As in most implementation steps up to this point, it is extremely important to engage your contractor and trade ally networks to achieve success. When rolling out a new software environment, especially if multiple software products are being introduced into the market at the same time, programs should coordinate software training along with the software vendors. Each software vendor may have their own training practices, webinars, or other resources they prefer to use.

Fig. 1.5: HPXML Contractor Training for Arizona Public Service (Source: EnergySavvy)

The trainings should include a clear understanding of how to download and upload HPXML files. This includes making transparent to all users the data collection requirements that you have determined as a part of your HPXML implementation. If contractors are unclear on what to collect and how the data validation system works, this can create a lot of frustration if contractor files are rejected by your data validation system. Being very transparent about this in training can significantly reduce the tech support requirement for your program implementation team and software vendors, while also increasing contractor satisfaction.

Note: If you are implementing Home Energy Score for the first time, all participating contractors and auditors must

complete a mandatory Home Energy Score training. If you are rolling out HPXML and Home Energy Score together, you will want to provide ample opportunity for contractors to participate in training and complete all requirements before launch.

The Home Energy Score website has more information on training requirements.

1.3.9 Step 7: Develop a Quality Management Plan

Once your HPXML program is launched, you can take advantage of the standardized data you have been collecting. This could be enhanced reporting, integration with real estate databases, or any of the other activities identified in the benefits section of this document. However, to ensure success with these activities, the collected data must accurately reflect the characteristics of the homes participating in the program. It is important to develop a data quality management plan that allows for continuous evaluation of the program to maximize accuracy.

The following practices can be integrated into a program plan, in addition to standard quality assurance activities:

Complete a Regular Data Review

Consider scheduling a regular data review at least once a year, if not more frequently. This is a basic audit of your data to verify that you are capturing all of the required data, a screen for data anomalies, and a way to ensure your validation systems are working as expected. After these meetings, consider communicating findings to all trade allies and software vendors to facilitate continuous improvement.

QA Trend Analysis

Consider working with your program software vendor to add trend analysis capability into your reporting systems. Even with an automated validation system, it is possible to game the data and report only what will pass the validation protocol. Your project-level QA strategy should provide some verification activities to reduce the potential of this. However, it is often not cost-efficient to schedule third-party verification of every home. Looking for data trends can assist in identifying potential issues. For example, if a large portion of jobs are reporting an installed condition that is almost identical to manufacturer specifications, you could focus QA on these jobs to further verify data accuracy.

Integrated Measurement and Evaluation

With a rise in automated billing analysis software, automated metering, and connected thermostats, there is an emerging ability to verify performance data in a more dynamic fashion. This can help quickly identify potential issues with data quality and help focus on continuous process improvements.

Regardless of the strategy you choose for your program, it is important to complete a comprehensive evaluation of your data before you start exporting data to market. Especially if you intend to use the data to guide financial investments, continuous quality management will be a key to driving success.

1.3.10 Conclusion

This concludes the Program Administrator section of the Implementation Guide. The next section will provide similar guidance for software developers.

This Implementation Guide is a living document, but updated as HPXML and the market evolves. This Implementation Guide will benefit from the lessons learned by different organizations integrating HPXML into their operations. If you have questions about HPXML or believe you can contribute to the overall success of its deployment, please email us at homeperformance@energystar.gov.

Likewise, if you would like to become more involved in the BPI-WG 5 and help guide the future of HPXML, email hpxml@homeperformance.org, to be added to the distribution list and meeting notices.

1.4 Software Developer Guide

1.4.1 Introduction

This software developer guide is designed to outline the technical details of getting a Home Performance XML (HPXML) implementation up and running. HPXML is an expansive standard data format based on XML and maintained by the Building Performance Institute's Working Group 5 (BPI WG-5). For more information on the BPI data standards and their relation to HPXML see *What are the BPI Data Standards?*.

The HPXML format is defined by a set of XML Schema (XSD) documents that outline all the acceptable data elements, their structure, and relation to one another. The schema itself is very flexible. Almost none of the elements are required, which allows for any level of detail in home performance data to be transmitted in the format. This level of flexibility is very useful but it also can be dangerous for the software developer. Two developers could each create an implementation that would represent their data in valid HPXML, but would be very divergent. The purpose of this guide is to document the assumptions that are not codified in the schema that are necessary to write and interpret HPXML documents across platforms.

The majority of the assumptions and recommendations here come from the set of pilot implementations in the Audit-Upgrade Use Case. This guide primarily documents lessons learned from those pilots. As other use cases present themselves, this guide will be augmented appropriately.

BPI Working Group 5

One of the best resources you will have available to you as you develop to HPXML is BPI Working Group 5 (a.k.a. HPXML Working Group). Many of the current developers are there. They have a lot of experience in getting HPXML working and can help you avoid costly pitfalls.

To join the working group send an email to hpxml@homeperformance.org.

GitHub

The HPXML schemas, example files, and this guide are all maintained on GitHub. You will always be able to find the latest version of the schemas there. Additionally as the schemas evolve over time you can be connected to and influence that process.

1.4.2 Versioning

The HPXML schemas follow the Semantic Versioning v2.0 specification. The version numbers follow a pattern of *Major*. *Minor*. *Patch* (i.e. 2.0.0).

The first element of `HPXML.xsd` will indicate the version of the schema via the `version` attribute. Note that when referencing a version, the patch number is assumed to be zero if it is omitted.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://hpxmlonline.com/2014/6"
 targetNamespace="http://hpxmlonline.com/2014/6" elementFormDefault="qualified" version="2.0">
```

Additionally, starting with version 2.0, the version of the schema used is required in the `schemaVersion` attribute on the root element of every document.

```
<HPXML xmlns="http://hpxmlonline.com/2014/6" schemaVersion="2.0">
```

Major

The major version number is incremented when the schemas are changed in a manner which is incompatible with previous versions. Examples of changes which necessitate a major version change include:

- Renaming elements
- Removing elements
- Moving elements
- Removing enumerations

A different xml namespace is used for each major revision. Starting with version 2.0, the namespaces follow the pattern

```
http://hpxmlonline.com/[Year]/[Month]
```

where the year and month are when the major version number was changed.

Minor

The minor version number is incremented when the schemas are changed in a manner which is backwards compatible with previous versions that share the same *Major* version. Backwards compatible in the context of HPXML means that given the schema changes, a document created in a previous version of the schema will also validate against the new schema. Example of changes which necessitate a minor version change include:

- Adding elements
- Adding enumerations
- Changing the annotation in the schema for an element

Warning: Based on the definition of backwards compatibility above, adding enumerations is a non-breaking change. However, it can be breaking for receiving systems if they're not expecting the change. The working group will provide warning when new enumerations are added so that receivers have an opportunity to respond by updating support.

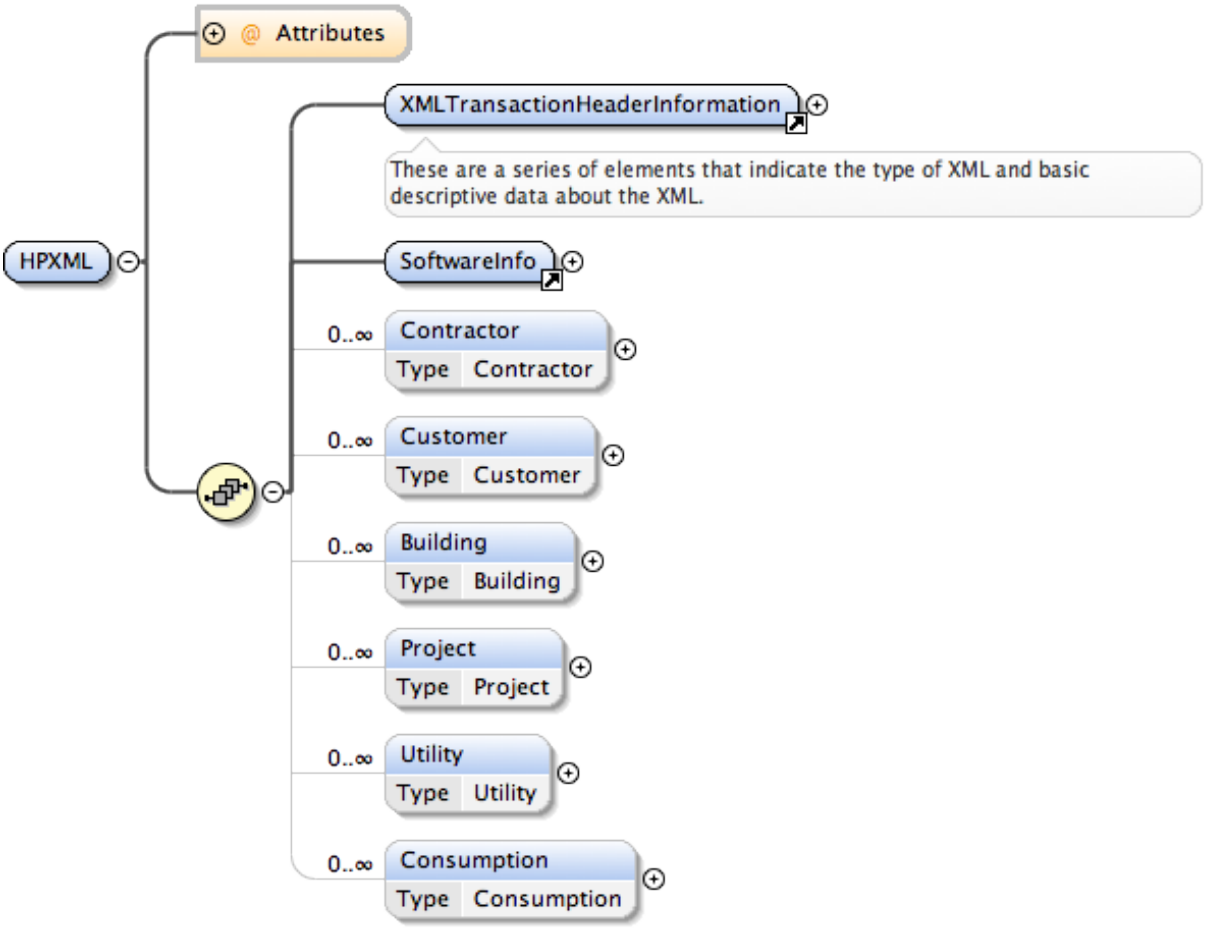
Patch

A patch version number is incremented when a backwards compatible change (as described in *Minor*) is made to address a bug.

1.4.3 Document Structure

Table of Contents

Top Level Nodes



Each of the top level nodes described below with the exception of *XMLTransactionHeaderInformation* and *SoftwareInfo* represent a high-level block of information about a building or project that can be related to other nodes to describe useful information about a building, the people and businesses who interact with the building, and actions taken on the building. The relationships between the top level nodes are defined with XML Element References that can be used, for example, to associate a *Building* with a *Project*, *Consumption* with a *Building*, or a *Contractor* with a *Project*.

The schema itself does not enforce the particular constraints for Use Cases, but rather provides a container for all the

relevant components and a referencing scheme to relate them.

XMLTransactionHeaderInformation

The `XMLTransactionHeaderInformation` element meta data about the HPXML file.

```
<XMLTransactionHeaderInformation>
  <XMLType>audit</XMLType>
  <XMLGeneratedBy>Housesoft 1.0</XMLGeneratedBy>
  <CreatedDateAndTime>2014-09-02T17:32:12Z</CreatedDateAndTime>
  <Transaction>create</Transaction>
</XMLTransactionHeaderInformation>
```

`XMLType` is generally unused and may be deprecated in the future.

`XMLGeneratedBy` is often used to transmit the name of the software that generated the HPXML file. It may also be deprecated in the future due to its redundancy with *SoftwareInfo*.

`CreatedDateAndTime` is the date and time the file was generated in the ISO 8601 format.

`Transaction` describes whether this is a new document or an update to a previous one.

SoftwareInfo

`SoftwareInfo` provides a place to transmit information about the software used to generate the HPXML.

```
<SoftwareInfo>
  <SoftwareProgramUsed>WOPR</SoftwareProgramUsed>
  <SoftwareProgramVersion>1.0</SoftwareProgramVersion>
</SoftwareInfo>
```

Contractor

The `Contractor` node describes a business that the customer works with to do an audit or upgrade to their building.

Customer

A customer is the owner, tenant, or some other person who has a vested interest in the house being described and worked on. This node is a place to describe that person, their contact information, and their relation to the building.

Building

The `Building` node describes the physical characteristics of a building at a point in time past, present, or future.

Project

The `Project` node describes work that has been done or is to be done to a *Building*. The measures described can have references pointing to specific components on the building and what was changed between the pre- and post-upgrade states and associated costs.

Utility

The `Utility` node represents a utility company.

Consumption

The `Consumption` node stores and represents the energy and/or water use of a building. It can contain high resolution electric smart meter data, the fuel oil fill up that happens once or twice a year, or more typically, monthly gas or electric bills.

Extension Elements

Because it is impossible to foresee every possible data point that will ever need to be collected and transmitted about a house or upgrade, most elements in HPXML contain an `extension` element containing an `<xs:any>` designation. That allows any element from any namespace to be inserted there. This is to facilitate transfer of data elements not available in the standard.

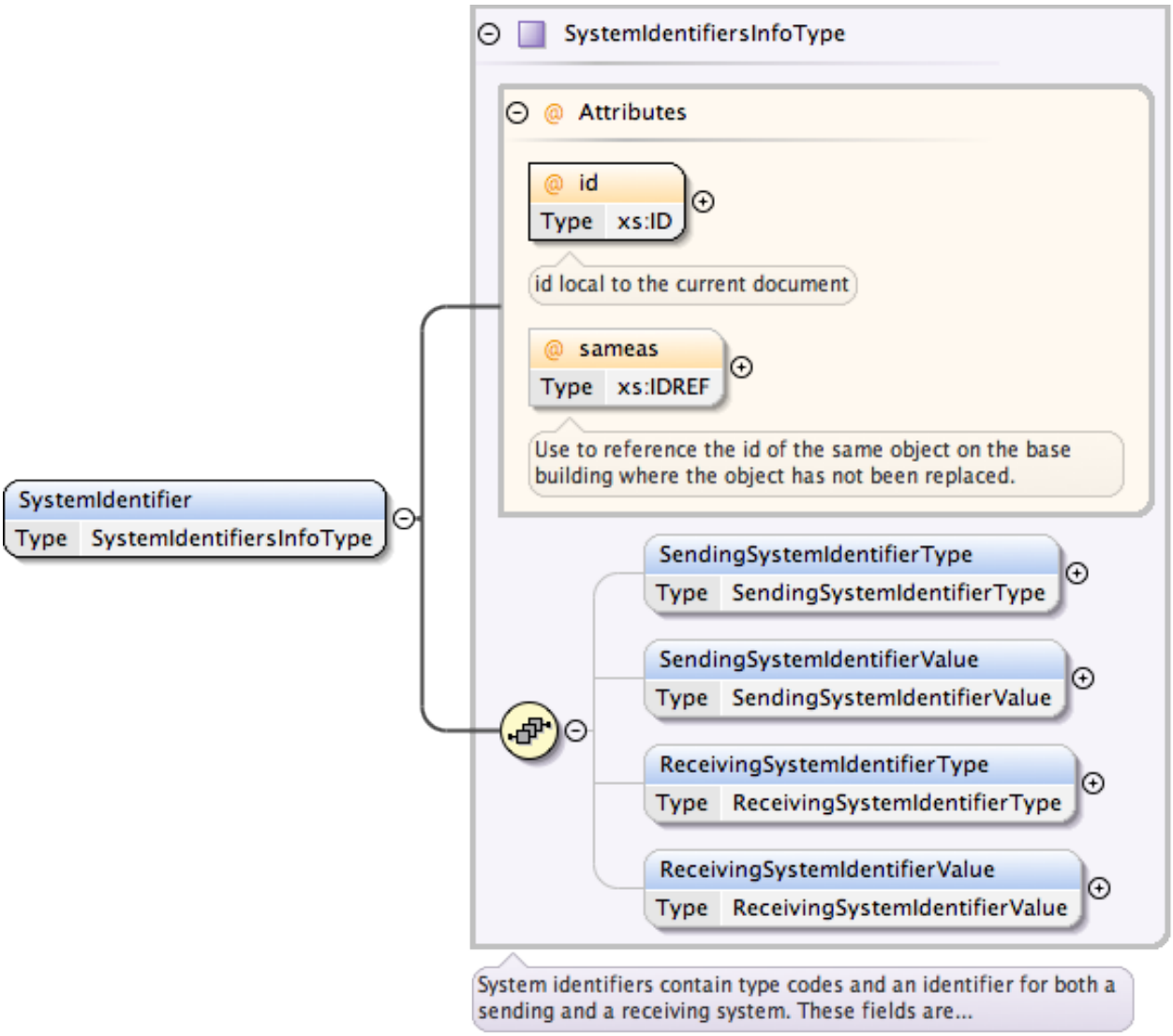
```
<extension>
   <QuantityWoodChucked>as much wood as a wood chuck could chuck</QuantityWoodChucked>
</extension>
```

Warning: Please exercise extreme caution and discretion when you consider implementing `extension` elements. Often times the temptation to use them happens when a difference arises between the way your software and/or home performance program represents a certain data field and the way HPXML represents it. It is crucial in these cases to either map your data into HPXML or change the way you represent it internally to conform to the HPXML standard. **If each software vendor and home performance program extends HPXML in non-standard ways, the value proposition of the standard is nullified.**
If there is no possible way to map your data into existing HPXML data fields, please contact BPI Working Group 5 (a.k.a. the HPXML working group) before implementing an extension. The working group would prefer to extend the standard for the benefit of everyone and avoid the use of extensions wherever possible. Often times you will not be the only one with the need for a particular element that was overlooked in the standard. By participating in the working group and lobbying for the elements you need you can enhance the value of HPXML for all parties.

1.4.4 XML Element References

XML documents are inherently hierarchical. This works out quite well for describing many things regarding houses and home performance in general. For instance, an AFUE is a property of a furnace which is part of an HVAC system on a building. Many other relationships are not as strictly hierarchical, however. An example of this would be the relationship between two furnaces in different snapshots of the building where one furnace replaced the other. In that case, there is no clear parent-child relationship.

In HPXML we have the ability and requirement to identify many elements with unique identifiers. This facilitates referential relationships between elements throughout the document. This is done with the `SystemIdentifier` element.



Intra-document references

The most common reference you will make is a reference that is internal to the document.

ID and IDREF

Each element in the HPXML document that could need to be referenced has a required `SystemIdentifier` sub-element which in turn has a required `id` attribute. That attribute has the xs:ID data type in XML Schema. This provides a unique identifier for that element within the document. This is similar to a primary key in a relational database.

When an element needs to reference another element it uses an xs:IDREF datatype, which ensures that the id referenced exists somewhere within the document. One simple example of this is how a window can reference which wall it is attached to.

```
<?xml version="1.0" encoding="UTF-8"?>
<HPXML xmlns="http://hpxmlonline.com/2014/6" schemaVersion="2.0">
    <XMLTransactionHeaderInformation>
```

```
        <XMLType></XMLType>
        <XMLGeneratedBy></XMLGeneratedBy>
        <CreatedDateAndTime>2014-09-03T16:06:24Z</CreatedDateAndTime>
        <Transaction>create</Transaction>
    </XMLTransactionHeaderInformation>
    <SoftwareInfo/>
    <Building>
        <BuildingID id="bldg1"/>
        <ProjectStatus>
            <EventType>audit</EventType>
        </ProjectStatus>
        <BuildingDetails>
            <Enclosure>
                <Walls>
                    <Wall>
                        <SystemIdentifier id="wall1"/>
                    </Wall>
                </Walls>
                <Windows>
                    <Window>
                        <SystemIdentifier id="window1"/>
                        <AttachedToWall idref="wall1"/>
                    </Window>
                </Windows>
            </Enclosure>
        </BuildingDetails>
    </Building>
</HPXML>
```

sameas

The `sameas` attribute is a special IDREF that is used most predominantly in the case of the Audit-Upgrade. It serves to link components of buildings between the pre- and post-upgrade `Building` nodes. Each `Building` node is a full description of the building and measures only affect some of the components. For components that do not change it is useful to have a way to indicate that they are the same item.

```
<?xml version="1.0" encoding="UTF-8"?>
<HPXML xmlns="http://hpxmlonline.com/2014/6" schemaVersion="2.0">
    <XMLTransactionHeaderInformation>
        <XMLType></XMLType>
        <XMLGeneratedBy></XMLGeneratedBy>
        <CreatedDateAndTime>2014-09-03T16:06:24Z</CreatedDateAndTime>
        <Transaction>create</Transaction>
    </XMLTransactionHeaderInformation>
    <SoftwareInfo/>
    <Building>
        <BuildingID id="bldg1"/>
        <ProjectStatus>
            <EventType>audit</EventType>
        </ProjectStatus>
        <BuildingDetails>
            <Enclosure>
                <Walls>
                    <Wall>
                        <SystemIdentifier id="wall1"/>
                    </Wall>
                </Walls>
```

```
            </Enclosure>
        </BuildingDetails>
    </Building>
    <Building>
        <BuildingID id="bldg1post"/>
        <ProjectStatus>
            <EventType>proposed workscope</EventType>
        </ProjectStatus>
        <BuildingDetails>
            <Enclosure>
                <Walls>
                    <Wall>
                        <SystemIdentifier id="wall1post" sameas="wall1"/>
                    </Wall>
                </Walls>
            </Enclosure>
        </BuildingDetails>
    </Building>
</HPXML>
```

Inter-document references

The `SystemIdentifier` element also has sub-elements that facilitate specifying identifiers for both a sending and receiving system. This way a document could identify components based on where it is coming from and going to. This feature currently isn't used much in lieu of the much simplified *Intra-document references*.

1.4.5 Use Cases

Audit-Upgrade

Contents

The audit-upgrade use case covers two scenarios:

1. A baseline building with a proposed work scope
2. A baseline building with a completed work scope

Both scenarios describe a pre- and post-upgrade building and the actions (measures) that occur between the two states to make the difference. To achieve this the HPXML document needs to have the *top level nodes* described as below.

All xml examples are taken from the `examples/audit.xml` document. It contains all of the required fields for an audit use case data transfer.

Contractor

The `Contractor` elements should list the name of the contractor–the person who is proposing and or completing the work. The company being represented and an email address are also included.

```
<Contractor>
    <ContractorDetails>
        <SystemIdentifier id="contractor1"/>
        <BusinessInfo>
            <SystemIdentifier id="business1"/>
            <BusinessName>ACME Home Performance Company</BusinessName>
            <BusinessContact>
                <Person>
                    <SystemIdentifier id="contractorperson1"/>
                    <Name>
                        <FirstName>John</FirstName>
                        <LastName>Doe</LastName>
                    </Name>
                    <Email>
                        <EmailAddress>john.doe@hpxmlonline.com</EmailAddress>
                    </Email>
                </Person>
            </BusinessContact>
        </BusinessInfo>
    </ContractorDetails>
</Contractor>
```

Customer

The customer is the homeowner or resident. The `Customer` element should include their name and phone number as shown.

```
<Customer>
    <CustomerDetails>
        <Person>
            <SystemIdentifier id="customer1"/>
            <Name>
                <FirstName>Jane</FirstName>
                <LastName>Customer</LastName>
            </Name>
            <Telephone>
                <TelephoneNumber>555-555-5555</TelephoneNumber>
            </Telephone>
        </Person>
    </CustomerDetails>
</Customer>
```

Building

There are two `Building` nodes in an Audit-Upgrade use case document: pre- and post-upgrade. Each is a full description the house. For the audit use case, the pre-upgrade condition is the house as audited and the post-upgrade condition is the proposed work scope. For the upgrade use case, the pre-upgrade building is the state of the house before any work was done and the post-upgrade state is the final building state audit.

Pre-upgrade The pre-upgrade `Building` element comes first in the document. It describes the initial state of the building. It should have a `ProjectStatus/EventType` of `audit`.

```
<Building>
    <BuildingID id="bldg1"/>
    <ProjectStatus>
        <EventType>audit</EventType>
    </ProjectStatus>
</Building>
```

Many items within the building require a unique `SystemIdentifier` element. The `id` attribute is used to specify this id within the document (see *ID and IDREF*).

For example, the water heater in the pre-upgrade building has an id of `dhw1`.

```
<WaterHeatingSystem>
    <SystemIdentifier id="dhw1"/>
    <FuelType>natural gas</FuelType>
    <WaterHeaterType>storage water heater</WaterHeaterType>
    <Location>conditioned space</Location>
    <CombustionVentingSystem idref="combvent1"/>
</WaterHeatingSystem>
```

Post-upgrade The post-upgrade `Building` element appears second in the document. It describes the "after" state of the building. In the audit use case, that means the *proposed* state of the building after the upgrades. In the upgrade use case, that means the *actual* audited state of the building after the work is completed. The `ProjectStatus/EventType` element has a different value depending on the use case:

Table 1.1: Post-upgrade Event Types

Use Case	Event Type
Audit	proposed workscope
Upgrade	job completion testing/final inspection

The post-upgrade building is mostly a duplicate of the pre-upgrade building where components of the building that do not change remain the same. However, each component in the post-upgrade building needs a unique identifier that is different from the unique identifier in the pre-upgrade building. The `sameas` attribute of the `SystemIdentifier` element is used to link identical elements in the pre- and post-upgrade buildings (see *sameas*).

Going back to the water heater example, the water heater in the post-upgrade building has a different `id` than the identical water heater in the pre-upgrade building, but it has a `sameas` attribute to link it back to the pre-upgrade water heater and indicate it is indeed the same equipment.

```
<WaterHeatingSystem>
    <SystemIdentifier id="dhw1p" sameas="dhw1"/>
    <FuelType>natural gas</FuelType>
    <WaterHeaterType>storage water heater</WaterHeaterType>
    <Location>conditioned space</Location>
    <CombustionVentingSystem idref="combvent1"/>
</WaterHeatingSystem>
```

Note: When a measure changes a component between a pre- and post-upgrade building, the `SystemIdentifier/@sameas` attribute is omitted because the measure references the relationship between components.

Project

In this paradigm, the *Pre-upgrade* and *Post-upgrade* building elements describes the state of the building at points in time. The `Project` element describes what was done or is to be done to get from one state to another.

The `ProjectSystemIdentifiers` are used to reference the pre- and post- building ids. The redundant `BuildingID` element should reference the post- building.

```
<Project>
    <BuildingID id="bldg1p"/>
    <ProjectDetails>
        <ProjectSystemIdentifiers id="bldg1"/>
        <ProjectSystemIdentifiers id="bldg1p"/>
        <EnergySavingsInfo>
        </EnergySavingsInfo>
        <Measures>
        </Measures>
    </ProjectDetails>
</Project>
```

Energy Savings `EnergySavingsInfo` is used to transmit either or both the estimated or measured energy use and savings achieved in an upgrade.

```
<EnergySavingsInfo>
    <EnergySavingsType>estimated</EnergySavingsType>
    <FuelSavings>
        <Fuel>electricity</Fuel>
        <Units>kWh</Units>
        <TotalSavings>3000</TotalSavings>
        <TotalDollarSavings>55</TotalDollarSavings>
        <PctReduction>0.1</PctReduction>
    </FuelSavings>
    <FuelSavings>
        <Fuel>natural gas</Fuel>
        <Units>therms</Units>
        <TotalSavings>100</TotalSavings>
        <TotalDollarSavings>123</TotalDollarSavings>
        <PctReduction>0.3</PctReduction>
    </FuelSavings>
    <AnnualPercentReduction>0.25</AnnualPercentReduction><!-- 25% -->
</EnergySavingsInfo>
```

Note: All percentages are expressed in the form of fractions. i.e. 10% => 0.1

Measures The `Measure` element describes a piece of work done for a job. Each measure references one or more replaced components in the pre-upgrade building and one or more (usually one) installed components in the post-upgrade building. In cases where a component was installed without replacing an existing component the `ReplacedComponent` can be omitted. Similarly if something was removed and nothing was installed in its place `InstalledComponent` would be omitted. The measure cost is also included.

From the example file, this measure

```
<Measure>
    <MeasureSystemIdentifiers>
        <SystemIdentifiersInfo id="furnacereplacement"/>
    </MeasureSystemIdentifiers>
```

```
                <Cost>3000</Cost>
                <ReplacedComponents>
                    <ReplacedComponent id="htgsys1"/>
                </ReplacedComponents>
                <InstalledComponent id="htgsys1p"/>
            </Measure>
```

replaces this furnace in the pre-upgrade building

```
                <HeatingSystem>
                    <SystemIdentifier id="htgsys1"/>
                    <UnitLocation>basement - conditioned</UnitLocation>
                    <CombustionVentingSystem idref="combvent1"/>
                    <HeatingSystemType>
                        <Furnace/>
                    </HeatingSystemType>
                    <HeatingSystemFuel>natural gas</HeatingSystemFuel>
                    <AnnualHeatingEfficiency>
                        <Units>AFUE</Units>
                        <Value>0.80</Value>
                    </AnnualHeatingEfficiency>
                    <FractionHeatLoadServed>0.7</FractionHeatLoadServed>
                </HeatingSystem>
```

with this one.

```
                <HeatingSystem>
                    <SystemIdentifier id="htgsys1p"/>
                    <UnitLocation>basement - conditioned</UnitLocation>
                    <CombustionVentingSystem idref="combvent1p"/>
                    <HeatingSystemType>
                        <Furnace/>
                    </HeatingSystemType>
                    <HeatingSystemFuel>natural gas</HeatingSystemFuel>
                    <AnnualHeatingEfficiency>
                        <Units>AFUE</Units>
                        <Value>0.92</Value>
                    </AnnualHeatingEfficiency>
                    <FractionHeatLoadServed>0.7</FractionHeatLoadServed>
                </HeatingSystem>
```

Home Energy Score

DOE's Home Energy Score rates a home's energy performance on a scale of 1 (least efficient) to 10 (most efficient). The score is determined using the assessed characteristics of the building that are either entered into a web interface by a qualified assessor or submitted via software through a SOAP API. The API requires data inputs to be submitted in terms of the data model set forth by Home Energy Score. Therefore, any users of the Home Energy Score API must translate their data into the appropriate location and representation in the Home Energy Score input array.

The latest version of the Home Energy Score API provides the capability for an HPXML import. This receives an HPXML file as input and translates the user-specified *Building* element (whether pre- or post-upgrade) into corresponding Home Energy Score inputs and populates the Home Energy Score input array. By using this import capability software developers can leverage their investment in HPXML to provide Home Energy Score functionality at a minimum incremental development cost.

Home Energy Score API

For more information on how to use the Home Energy Score API, see the API documentation. The API method, `submit_hpxml_inputs` provides the HPXML import capability.

Data elements required and translation details

The *HPXML Data Selection Tool* includes Home Energy Score. It represents the minimum required data fields for successful import. It is a good starting point.

The HPXML import into Home Energy Score can accept a larger variety of data elements than the minimum. The details of the translation and required HPXML elements are documented separately.

Additionally, The translator has been released open source on GitHub. Example HPXML files are available in that repository.