July 31, 2010

Re: SEC Proposal on Asset Backed Securities (File No. S70810)

In the proposal, the SEC presents the fundamental idea that ABS issuers must take more responsibility for their data. Today investors are burdened with the task of reading through prospectus documents, and translating them into useful models for data analysis. The SEC proposes that these models should instead be prepared by the ABS issuers and offered publically to investors. This shift in responsibility is key to improving the quality of reported data and increasing the accuracy of investor decision-making.

In order to make this shift effective, the right reporting format must be chosen, along with the right supporting framework. With the proper infrastructure in place, the SEC can offer these key advantages to both issuers and investors:

- Data accuracy – assurance that all data reported by ABS issuers are 100% accurate
- Data comparability – immediate comparability of data from one ABS issuer to another. For example, when one ABS issuer states a "loan-to-value" ratio, it should mean the same thing as another issuer's "loan-to-value" ratio.
- Data transparency – insight into where the data originated from and how it was computed. For example, if a tranche has a promised yield of 10% and receives an actual yield of 8%, it must be possible to determine exactly how the 8% yield was calculated
- Extensibility – the ability to consume and extend any SEC-filed computation models, such that models may be shared and customized among issuers, and such that investors may incorporate additional criteria to enhance their decision-making process
- Sufficiency – the ability to offer a single solution that adequately meets issuer and investor needs, without the need to build or acquire from third party vendors.

XBRL is an XML-based open standard particularly designed to meet these goals. XBRL offers a standardized way to articulate semantic meaning such that business concepts, relationships, and rules may be expressed in a generic, machine readable manner. Like Python, XBRL is both machine and human readable, and may be consumed by any application that supports the XBRL standard. Unlike Python, however, XBRL is designed to enable business-level users as well as developers to understand and extend upon complex data models. XBRL has been in production use by regulators across the globe including banking supervisors in Europe, the FDIC (beginning in 2005) and the SEC's own Interactive Data Initiative.

The attached document seeks to evaluate the current SEC proposal against the criteria discussed above, and propose XBRL as an alternative to meeting the SEC's requirements in this area.

Sincerely,

Sunir Kapoor
President & Chief Executive Officer
UBmatrix, Inc.

UBmatrix, Inc.
333 Twin Dolphin Drive
Redwood City, CA 94065
(t) +1 650-264-4510
(f) +1 650-264-4515

www.ubmatrix.com

Response to SEC Proposal on Asset-Backed Securities (File No. S7-08-10)

Summary

The SEC proposal offers some key insights into how to help investors make sense of ABS filings. However ultimately the proposed approach is undermined by a simple implementation detail – the intent to use a Python-based computer program as the issuer's vehicle of delivery.

This document seeks to evaluate the SEC's proposal from the perspective of both the issuer and investor, using five criteria that we believe to be key success metrics: data accuracy, comparability, transparency, extensibility, and sufficiency. The document offers XBRL (eXtensible Business Reporting Language) as an alternative approach, one that both satisfies the criteria above and meets the SEC's stated requirements.

Introduction

In the proposal, the SEC presents the fundamental idea that ABS issuers must take more responsibility for their data. Today investors are burdened with the task of reading through prospectus documents, and translating them into useful models for data analysis. The SEC proposes that these models should instead be prepared by the ABS issuers and offered publically to investors. This shift in responsibility is key to improving the quality of reported data and increasing the accuracy of investor decision-making.

In order to make this shift effective, the right reporting format must be chosen, along with the right supporting framework. With the proper infrastructure in place, the SEC can offer these key advantages to both issuers and investors:

- Data accuracy – assurance that all data reported by ABS issuers are 100% accurate
- Data comparability – immediate comparability of data from one ABS issuer to another. For example, when one ABS issuer states a "loan-to-value" ratio, it should mean the same thing as another issuer's "loan-to-value" ratio.
- Data transparency – insight into where the data originated from and how it was computed. For example, if a tranche has a promised yield of 10% and receives an actual yield of 8%, it must be possible to determine exactly how the 8% yield was calculated
- Extensibility – the ability to consume and extend any SEC-filed computation models, such that models may be shared and customized among issuers, and such that investors may incorporate additional criteria to enhance their decision-making process
- Sufficiency – the ability to offer a single solution that adequately meets issuer and investor needs, without the need to build or acquire from third party vendors.

XBRL is an XML-based open standard particularly designed to meet these goals. XBRL offers a standardized way to articulate semantic meaning such that business concepts, relationships, and rules may be expressed in a generic, machine readable manner. Like Python, XBRL is both machine and human readable, and may be consumed by any application that supports the XBRL standard. Unlike Python, however, XBRL is designed to enable business-level users as well as developers to understand and extend upon complex data models.

The following sections seek to evaluate the current SEC proposal against the criteria discussed above, and propose XBRL as an alternative to meeting the SEC's requirements in this area.

The Waterfall Computer Program

The SEC proposes that ABS issuers file a Python-based waterfall computer program, allowing investors to download the source code and run the program on their own systems. The goal is to help investors conduct a more thorough and expedited analysis of the ABS offering, without requiring their own computational models or third party tools.

While this approach is viable in certain circumstances, its success relies strongly on several assumptions.

Accuracy of the Program

Assumption 1: The issuer's computer program provides an accurate representation of the desired cash flow model.

Without the guarantee of an accurate model, investors may be relying on faulty data and making erroneous decisions. Therefore the accuracy of an issuer's computer program is integral to the success of the SEC's proposal. Unfortunately, with the Python-based approach, neither the SEC nor investors have a mechanism to verify either accuracy of the data or the model.

The SEC does propose a testing methodology, stating "By using the sample inputs to run the program, the investor will be able to confirm that the program is working correctly by matching the actual outputs produced against the sample expected output provided by the issuer". Not only does this statement suggest a manual and tedious process, but is extremely misleading. By using sample inputs and outputs, the investor can only confirm that those specific inputs will ensure a correct output. There is no guarantee that other inputs will produce an equally correct response. Also, because investors cannot visualize how the data flows through the system, there is no way of knowing that a correct output response was actually computed in the appropriate manner.

As a result, neither the SEC nor investors have the capability to thoroughly test the computer program to ensure its validity. Instead it is entirely up to the issuer's discretion whether or not they deem the program to be "accurate".

Comparability across Programs

Assumption 2: By making computer programs available from every ABS issuer, the investor's task of making investment decisions becomes easier.

Since minimal requirements have been placed on how each program must be written, issuers have complete freedom in how their application is designed. It is the issuer's responsibility to determine what the application's user interface should look like, what questions a user should be allowed to ask, and how data should be presented back to the user.

Consequently investors are burdened with the task of familiarizing themselves with one computer program per issuer. Furthermore, they are constrained by the questions offered by each application. Since programs are not required to use common questions, investors have no ability to compare questions or answers across applications. Even if different applications did pose the same questions, there is no guarantee that the intent of the question is the same, or that the terminology used refers to the same concepts.

Thus the proposed ABS-filed computer programs offer no data comparability across issuers. In order for an investor to ask their own custom questions to each application, and more importantly, ask the same question to each application, they must still go through the process of generating their own computational models or relying on third parties, no different then what they do today.

Transparency of the Program

Assumption 3: Investors may take advantage of the source code to gain deeper insights into an ABS issuer's waterfall model.

The SEC proposes that the computer program be filed "in the form of downloadable source code", allowing investors to not only run the computer program but also examine its inner workings. Presumably this choice allows an investor the advantages of open source code to gain insights into how the waterfall model actually works.

While this possibility is certainly available to investors, we must consider the skills needed to review and comprehend the programming code. Not only must the investor be an expert in reading Python code, but must have the time and experience to break down complex cash flow models into individual units of code. Only once this process is done will the investor be able to fully comprehend the computational model.

Assuming that the investor has the time and skills to perform such an analysis, one must question how feasible this approach is. One might argue that reading a prospectus in text is easier and more efficient than deciphering code, particularly if no standards exist around how that code has been developed, or how self-documented that code must be.

Furthermore the advantages of source code are restricted to those with development resources. To a business-level user, a computer program is simple a black box that may accept certain inputs. To these users, a computer program provides zero transparency and zero insight into the inner workings of the simulation.

Extending the Programs

While the ability to extend a computer program is not directly mentioned by the SEC proposal, the nature of an open source program infers that it may be reused and customized by other parties. Thus inherently the following assumption exists:

Assumption 4: Investors and issuers may take advantage of the source code to reuse and extend a waterfall computer program.

While this capability is certainly achievable, the audience must first have the tools to gain a full understanding of the computational model as described in Assumption 3. Without the ability to delve into Python code and learn the program's implementation, it would be extremely difficult to enhance the model for one's own purposes.

Assuming that the investor has the time and resources to modify the waterfall programs, the same customization must be developed for every filing they wish to analyze. Still even in such a case, true data comparability is not guaranteed, as discussed in Assumption 2.

A more efficient and effective approach would be for the investor to write their own simulation once, which they may run for every filing that comes in. Writing custom simulations becomes even more appealing with the introduction of XML-based ABS filings as also proposed by the SEC. Filings may simply be fed into an investor's own computational model with minimal pre-processing.

Sufficiency of the Programs

Assumption 5: A waterfall computer program is not a sufficient tool for investors. Investors will still need to acquire or build to gain a complete analysis of asset-backed information.

The SEC proposal states in footnote 339 that the "waterfall computer program is a necessary but not a sufficient tool for carrying out quantitative analysis of an ABS". Thus, the SEC's expectation is that an investor would still rely on their own tools or a third-party to fully carry out their analysis.

While this limitation may be true, it is important to consider other options that may not be so restrictive. An ideal system should be self-sufficient for meeting investor needs, enabling customizations when features are unavailable.

ABS filing with XBRL

Now that we've explored some of the consequences of a Python-based computer program, this document would like to present XBRL as a viable alternative, both as a mechanism for submitting ABS data and as a platform for modeling cash flows for investor analysis.

Data Accuracy

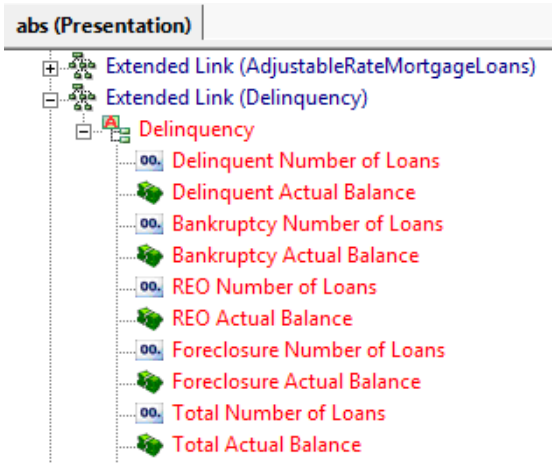
The SEC proposal clearly states a desire for more accuracy in filing prospectus data. In Section B.1, the proposal reads "We believe… the waterfall computer program would convey information to investors in a form that is both more accurate and more useful".

The proposal takes a first step to meeting these needs by requiring submissions in XML format. By requiring data tagging in XML, the SEC is enforcing a level of syntactic validation – meaning that the data can be compared to a set of guidelines. This approach ensures that the data is formatted in the appropriate structure and declared using the proper data types.

XBRL, as an XML-based standard, provides these same benefits and offers some additional ones as well. XBRL's fundamental goal is to express semantics, also known as business meaning. An XBRL taxonomy

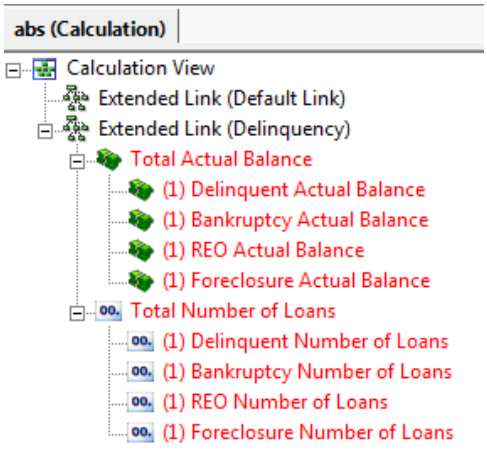
defines a dictionary of concepts and relationships between concepts in a manner that allows entire sets of standards and the specific domain knowledge around these standards to be captured. For example, IFRS (International Financial Reporting Standards), US-GAAP (U.S. generally accepted accounting principles), and the BASEL II Accords are all standards from which XBRL taxonomies have been created.

Let's take a quick look at what an XBRL taxonomy designed around asset-backed securities might include:



We have defined a set of concepts to represent monetary and integer values that an ABS issuer may report. Fundamentally these are XML elements with the same constraints as XML data, grouped together in a logical representation for users.

Now let's look at these same concepts in a different way:



Notice that the same concepts are represented here, simply in a different hierarchical structure. XBRL, unlike XML, allows you to define relationships between concepts in various pre-defined and custom ways. This particular structure represents a calculation hierarchy that describes how numbers should roll up to their totals. For example, let's say the issuer reports "Delinquent Actual Balance", "Bankruptcy Actual Balance", "REO Actual Balance", and "Foreclosure Actual Balance" are all equal to "$200,000", and also reports that "Total Actual Balance" equals "$600,000". Using the above hierarchy as a guide,

XBRL can automatically validate the data and notify the issuer of a rollup error. Through this approach, the SEC can enforce that only 100% validated data may be accepted as a submission.

The goal of XBRL taxonomies is to encapsulate the principles and regulations set forth by standards such as US-GAAP and IFRS, such that may be processed in a machine-readable way. In this manner, the submission process is designed to automatically assess the validity and quality of incoming data according to the pre-defined rules of the standards. This same approach allows the SEC to model information about assets-based securities as a way to standardize terminology and ensure 100% valid submitted data.

Data Comparability

Using an XBRL taxonomy offers the SEC data comparability among ABS filings. By requiring all issuers to report against the ABS taxonomy, the SEC may enforce a convergence on a common dictionary of terms. As a result, when two issuers report a value for the concept "Loan to Value Ratio", the two reported values are guaranteed to be directly comparable.

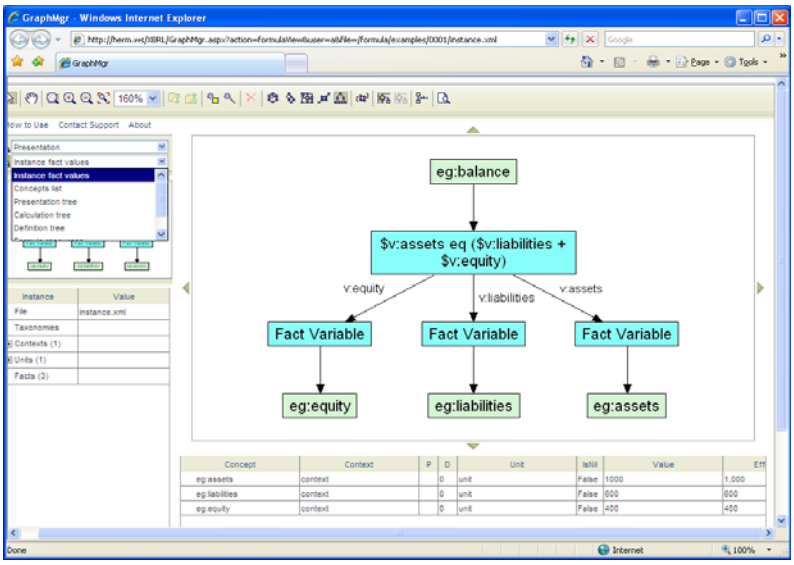
Along with cross-issuer comparisons, XBRL also make it easy to compare data across periods of time. Using XBRL, investors can easily write queries like "Show me all ABS prospectuses where the Loan to Value Ratio is greater than 80%, for data reported in June 2009 or later". Notice that the investor no longer has to focus on a specific ABS issuer or computer program to gain information about one prospectus for one period of time. Rather XBRL provides a standard platform for investors to run a single generic query across any or all prospectuses across any timeline.

To leverage XBRL, an investor simply needs to download the ABS taxonomy and the associated ABS filings. Since XBRL is an open standard, any XBRL compliant system may be used to read and analyze XBRL data.

Data Transparency

Using XBRL Formulas

Through XBRL, computation models may be developed and reviewed by business-level users rather than developers. Using XBRL-enabled tools like the one shown below, ABS issuers can easily generate cash flow models through the creation of business rules known as "XBRL formulas".



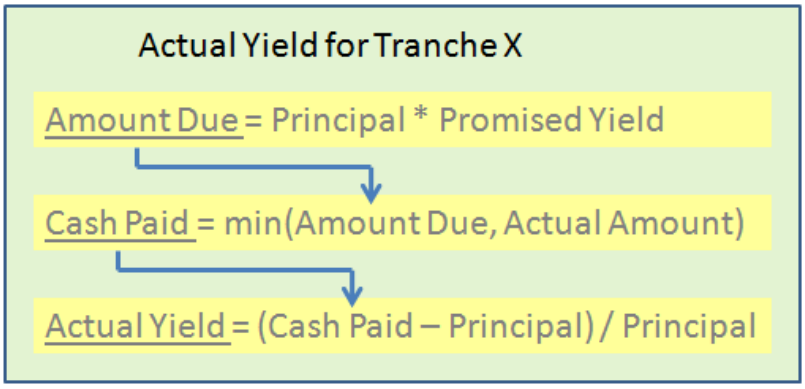
The above diagram demonstrates the use of an XBRL Formula editing tool to generate a "Balance" value from several contributing factors. The resulting model is saved in XBRL format and may be run automatically against an ABS filing to produce the desired "Balance" value.

XBRL formulas may be built to simulate ABS cash flow models and identify areas of potential risk. For example, let's consider a very basic formula that might be used in a waterfall.

Amount Due = Principal * Promised Yield

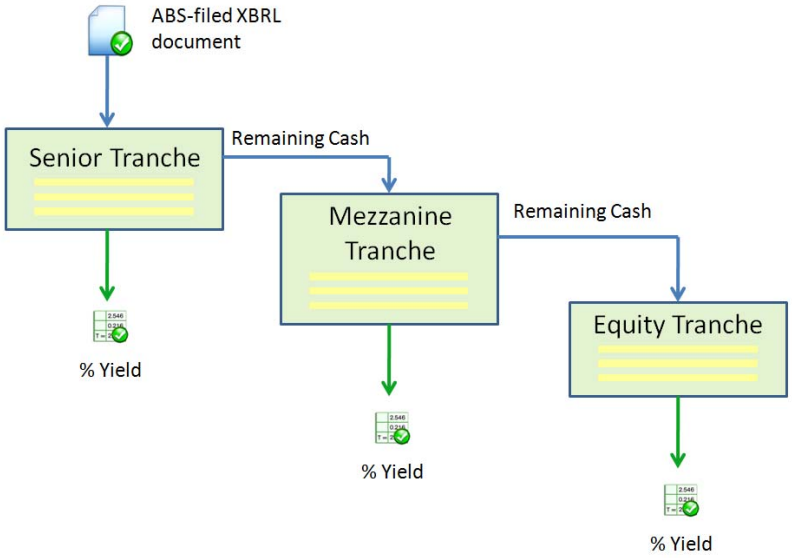
Each of these concepts "Amount Due", "Principal", "Promised Yield", etc. represent XBRL taxonomy concepts. An ABS issuer could write such a formula against the taxonomy by selecting the appropriate concepts and modeling their relationships using any formula editing tool.

We can add more business rules to calculate the actual yield for a tranche:

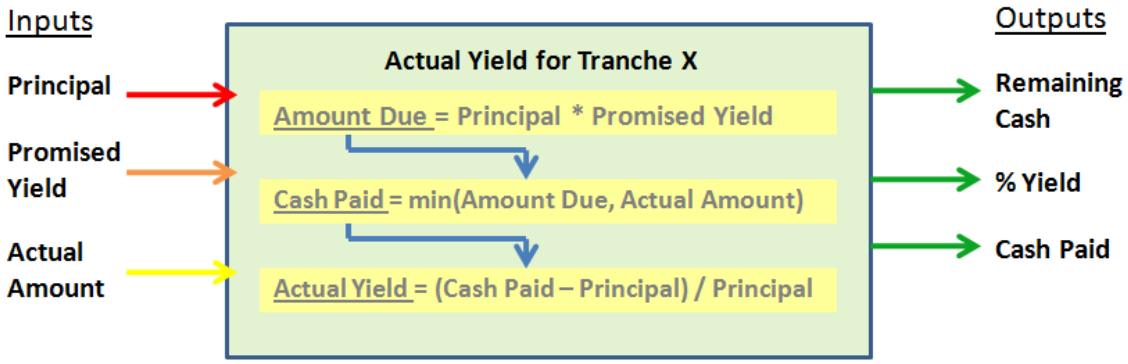


Notice that the outputs from one business rule may become an input into another rule… this is known as "formula chaining". Each business rule may use a variety of concepts and other input parameters to generate the desired outputs.

Using the same three formulas above, we can quickly create a simple waterfall model that allows the result from one tranche to flow into another.



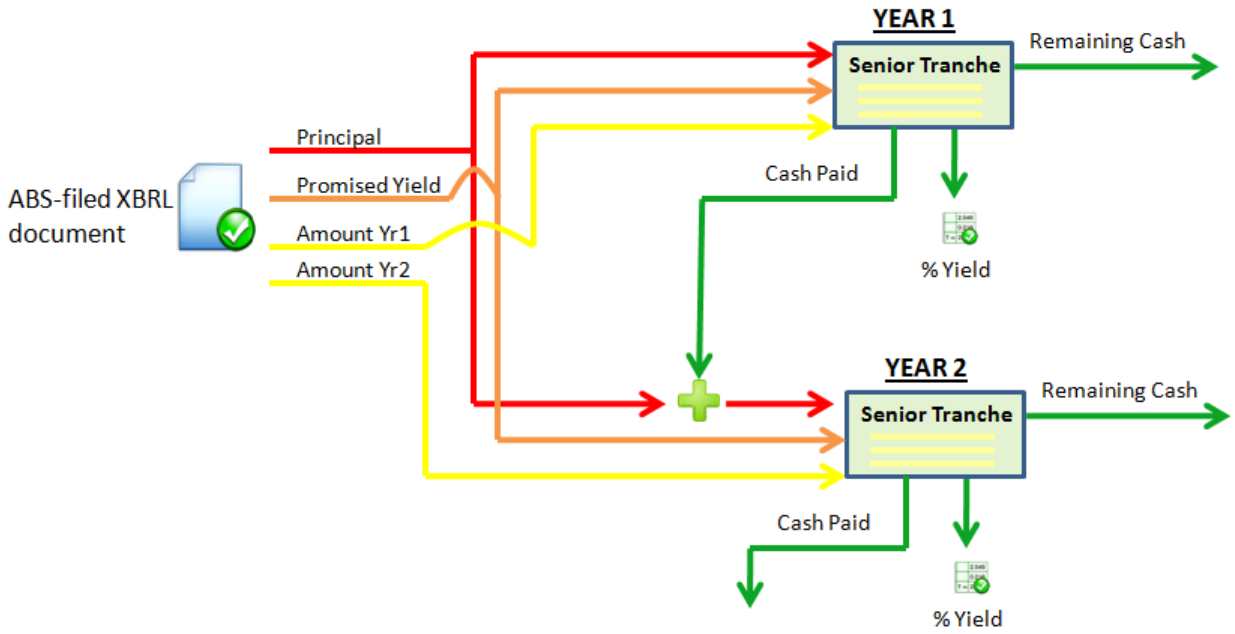
We can further elaborate on this simple model to develop a multi-year cash flow scenario. To do so, let's go back to our original set of 3 formulas and focus on the input and output parameters we need.



Consider how each input parameter evolves from year to year:

- Principal – As each year goes by, the new Principal = original Principal + Cash Paid in the previous year
- Promised Yield – Assume in this case that the promised yield is constant
- Actual Amount – The actual gains or losses for a given year will vary depending on how well the ABS performs

Let's focus on just the "Senior Tranche" to see how cash flows from Year 1 to Year2:

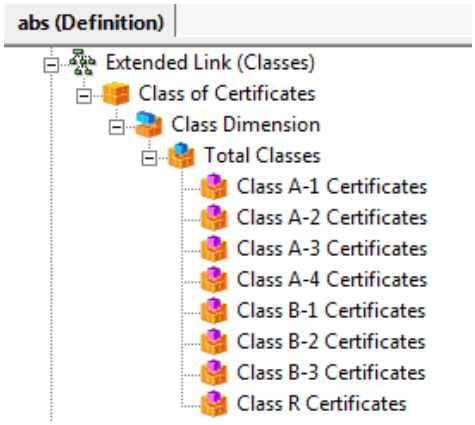


Notice that the formulas themselves *never change* -- only the inputs and outputs. As a result the formula chaining principle may be used to recurse through any number of years and any number of tranches.

Using XBRL Dimensions

Recall that XBRL taxonomies offer the ability to model concepts in a variety of custom and pre-defined ways. Earlier we looked at a possible ABS taxonomy representation illustrating both presentation and calculation hierarchies. XBRL also provides its own multi-dimensional modeling capabilities using these same techniques.

Consider the following dimensional hierarchy in an ABS taxonomy:



Now let's apply this hierarchy to a simple waterfall that calculates the scheduled distribution of interest.

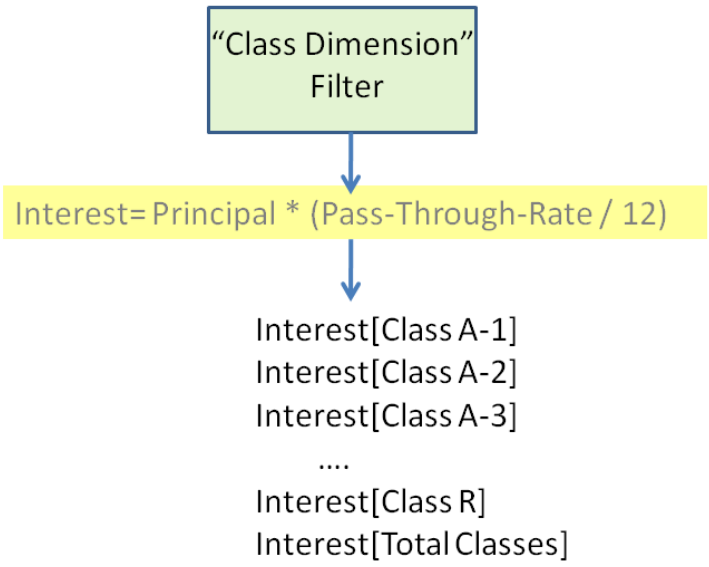
First we need to identify the taxonomy concepts used in this calculation:

- Principal
- Weighted Average Pass-Through Rate
- Interest

We write an XBRL formula that represents this type of calculation:

```
Interest= Principal * (Pass-Through-Rate / 12)
```

Since we need to apply this formula for every class of certificate, we apply a *dimension filter* to this formula. The filter notifies the formula that every member in the "Class Dimension" type must run through this model.



Notice also that we can calculate the Total interest along with class distributions. Since the taxonomy includes the class dimension model, the following validation checks can be automated:

- Interest[Total Classes] = Interest[Class A-1] + Interest[Class A-2] + Interest[Class A-3] + …
- Principal[Total Classes] = Principal[Class A-1] + Principal[Class A-2] + Principal[Class A-3] + ….

Depending on modeling requirements, filters may be customized to be more specific. For example, Class A certificates may be filtered through one formula, and Class B certificates may be filtered through another. In addition, we have modeled this particular waterfall using only a single dimension. Using the same techniques, multi-dimension flows may also be constructed.

Summary

Through XBRL-enabled tools, issuers may construct the necessary computational models without development resources or any understanding of Python or other computer language. Using these same tools, investors may review these models in a visual manner to gain a complete understanding of cash

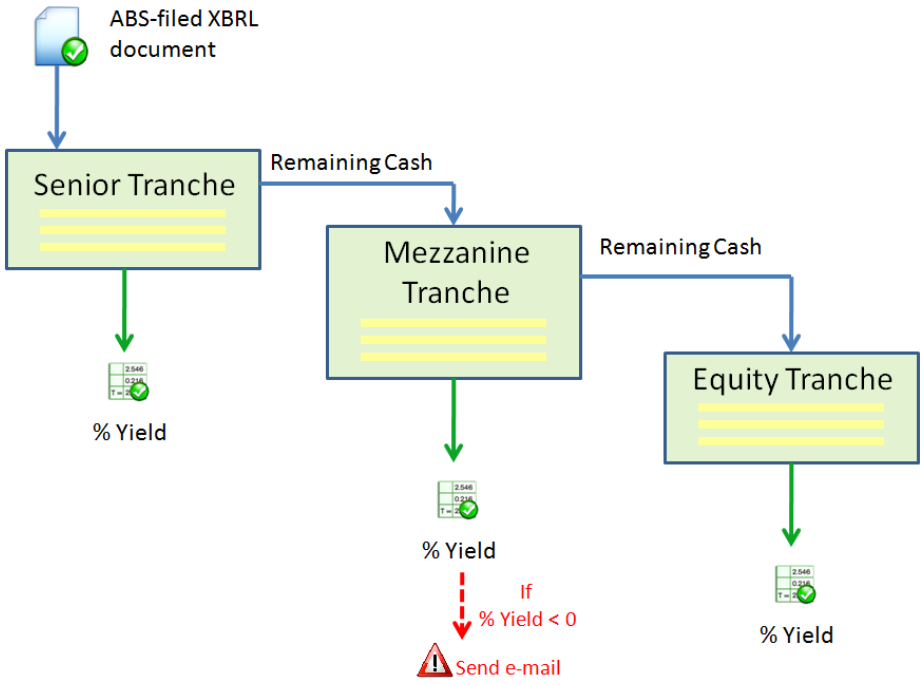
flow models. Investors may use input parameters and examine different outputs, but even more importantly may visualize how the input flows through the entire system and evolves into the final result.

Extensibility

With the ability to express semantics, XBRL enables business-level users to communicate and build complex data models with minimal technical skills. As domain knowledge evolves or new requirements are added, business users can make the necessary modeling changes by adjusting the XBRL taxonomy.

The SEC proposal acknowledges that "registrants may want to provide more program functionality… than would be required". XBRL allows both issuers and investors to enhance the computational models as desired. Like XML schema, XBRL provides mechanisms for extensibility. Extensibility allows users to define a cleaner separation of data, and encourages others to leverage existing taxonomies with their own customizations.

Through XBRL, an investor may enhance an issuer's computation models by adding their own XBRL formulas. For example, formulas may be added to identify potential risks and trigger alert notifications to the investor, as shown below.



Formulas could also perform additional analytics on the data, to generate custom ratios or results, or to incorporate additional input parameters from external sources.

Unlike modifying a computer program, using XBRL formulas allows the investor to apply the same formulas to any XBRL computational model. Since the formula language is defined strictly in terms of

XBRL concepts provided by the taxonomy, the same formulas may be applied repeatedly to different models from various ABS issuers.

Sufficiency

We believe XBRL to be a necessary and sufficient tool for carrying out ABS quantitative analysis, one which is beneficial to all parties. For the SEC, an XBRL taxonomy offers a core foundation for defining ABS concepts and enforcing ABS regulations to ensure 100% valid submitted data. For issuers, XBRL provides a platform for filing data accurately and generating computational models through tools rather than code. And for investors, XBRL presents a richer more efficient mechanism to gather and analyze prospectus data. Investors are guaranteed full data comparability and given access to a fully transparent and extensible framework that provides deep insights and customizability into ABS cash flow models.