

Implementation Guide

**NSI XML Retail Accounting Reports
(Vendors & Lotteries)**

Version 1.0, March 2007

NASPL

THE *Open* GROUP
Making standards work™

Copyright © 2007, The Open Group

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owner.

The Open Group® is a registered trademark of The Open Group in the United States and other countries.

Implementation Guide

NSI XML Retail Accounting Reports (Vendors & Lotteries)

Version 1.0, March 2007

Document Number: IG0701

Published by The Open Group, March 2007.

Comments relating to the material contained in this document may be submitted to:

nsi-specifications@opengroup.org

Contents

1	Introduction.....	1
1.1	Purpose and Scope.....	1
1.2	About This Document.....	1
2	Why Implement the Technical Standard?.....	3
2.1	Consolidated Accounting Reports for Chains.....	3
2.2	Multi-Jurisdictional Consistency.....	3
2.3	Free Up POS Device.....	3
3	How to Implement the Technical Standard.....	4
3.1	Read the Technical Standard.....	4
3.2	Implement the Technical Standard.....	4
3.2.1	Familiarization and Commitment.....	4
3.2.2	Gap Analysis.....	5
3.2.3	Implementation of the Technical Standard.....	6
3.2.4	Validation-Ready Steps.....	12
3.3	Corrigenda, Interpretations, & Frequently Asked Questions.....	13
4	Certification/Verification Process.....	14
5	Contact Information.....	15
A	Decomposing the NSI-PCATS XML Files for Lottery Retail Accounting Reports ...	16
A.1	Introduction.....	16
A.2	Base Components of the PCATS Schema.....	16
A.3	Chain or Multiple Location XML Reports.....	34
A.4	Advanced Concepts.....	36

Preface

North American Association of State and Provincial Lotteries (NASPL)

The NASPL Standards Initiative (NSI) was approved and funded by NASPL and the vendor community as a collaborative development effort with participation from the lotteries, gaming vendors, and retail associations. Project management and facilitation services for standards development and certification are provided by The Open Group in conjunction with NASPL.

The NSI vision is to provide an interoperable lottery environment that is based on a set of open Technical Standards, approved Best Practices, Certification and Verification programs that, when implemented, will improve the quality and integrity of the lottery environment, and will provide increased efficiencies, resulting in reduced costs and increased profit margins for lotteries, vendors, and lottery retailers.

The NSI mission is to establish a resilient organizational structure, set of processes, and procedures that will engage all constituents (lotteries, vendors, and retail representatives) in an environment of open discussion and cooperative development.

Further information about NASPL is available at www.naspl.org.

The Open Group

The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information within and between enterprises based on open standards and global interoperability. The Open Group works with customers, suppliers, consortia, and other standards bodies. Its role is to capture, understand, and address current and emerging requirements, establish policies, and share best practices; to facilitate interoperability, develop consensus, and evolve and integrate specifications and Open Source technologies; to offer a comprehensive set of services to enhance the operational efficiency of consortia; and to operate the industry's premier certification service, including UNIX certification. Further information on The Open Group is available at www.opengroup.org.

The Open Group publishes a wide range of technical documentation, the main part of which is focused on development of Technical and Product Standards, Best Practices, and Guides. Full details and a catalog are available at www.opengroup.org/bookstore.

Readers should note that all published NSI Technical Standards and Best Practices, and any updates, in the form of Corrigenda, are available at www.opengroup.org/naspl/published.

1 Introduction

1.1 Purpose and Scope

This document is the Implementation Guide for NSI XML Retail Accounting Reports. It has been developed by The Open Group. It is designed to help lottery vendors and lotteries implement the Technical Standard for XML Retail Accounting Reports in the Lottery Industry. This Guide will help relate the components involved in implementation to one another and a typical lottery environment, consistent with the requirements established by the Technical Standard. The components are:

- PCATS DRAFT EB2B Lottery Standard Schemas
- Technical Standard for XML Retail Accounting Reports in the Lottery Industry
- Lottery Gaming and Back-office Support Systems

Taken together, these components provide the means to produce the common accounting reports needed by lottery retailers in a standardized format.

The PCATS DRAFT EB2B Lottery Standard Schemas – hereafter referred to as the “PCATS Schemas” – provide the final form of the XML accounting reports.

The Technical Standard for XML Retail Accounting Reports in the Lottery Industry – hereafter referred to as the “Technical Standard” – is a bridge between the lottery or vendor systems and the PCATS Schema. The Technical Standard provides definitions and establishes the required and optional requirements for the PCATS Schema.

Finally, the lottery or vendor systems are the source of the accounting data needed to populate the XML accounting reports.

1.2 About This Document

The structure of this document is as follows:

- Chapter 1: Introduction
This section introduces the document and describes the purpose and scope of the Implementation Guide.
- Chapter 2: Why Implement the Technical Standard?
This section addresses the business rationale and operational issues that are driving the implementation of the Technical Standard.

- Chapter 3: How to Implement the Technical Standard

This section provides a guide to implementing the Technical Standard.

- Chapter 4: Certification/Verification Process

This section looks at what comes after the implementation of the Technical Standard, with a focus on moving toward formal certification/verification; that is, NSI Verification for lotteries and NSI Certification for vendors.

- Chapter 5: Contact Information

- Appendix A: Decomposing the NSI-PCATS XML Files for Lottery Retail Accounting Reports

2 Why Implement the Technical Standard?

This section sets the operational context and describes the business drivers and objectives for implementing the Technical Standard. The Technical Standard has three main benefits for lotteries, as follows:

- Consolidated Accounting Reports for Chains
- Multi-jurisdictional Consistency
- Free Up POS Device

2.1 Consolidated Accounting Reports for Chains

Most lotteries have retailers that have multiple locations throughout the jurisdiction. These are referred to as “chain stores”. Most lottery systems provide accounting reports based on location. The chain needs to get these accounting reports at their accounting office. This often means that a store manager runs the report and sends it back to corporate, or worse the chain does not sell lottery products.

By having a consistent, standard means to provide accounting data electronically, lotteries can reduce store labor, improve retailer accounting accuracy, and potentially expand the retailer base.

2.2 Multi-Jurisdictional Consistency

This benefit is longer term. Chains that span jurisdictions now have to build customized manual systems to deal with lotteries. These chains are the largest and potentially highest traffic chains in a jurisdiction. Due to the complexity of handling and accounting for lottery products, most chains limit lottery products or treat them as an evil necessity. As more lotteries comply with the Technical Standard, the accounting portion of that equation can be resolved. Chains will receive their accounting reports in a manner consistent with the information provided by their other vendors. This will allow more lottery product types in the stores and increase retailer support of the products.

2.3 Free Up POS Device

Currently, accounting reports are run from the Point of Sale (POS) terminal. These reports can take a considerable amount of time to produce, time during which the terminal is not selling. By providing this information to the retailer via an electronic file, the POS terminal can stay focused on sales. This can be significant, especially during large jackpot timeframes.

3 How to Implement the Technical Standard

3.1 Read the Technical Standard

The Technical Standard was developed within the NSI Technical Standard Working Group and was subjected to a wide review open to all NASPL lotteries and NSI vendors. Following the review, it was approved by the NSI Steering Committee and ratified by the NASPL Executive Committee.

In order to implement the Technical Standard, you must read the Technical Standard. In particular, you should become very familiar with Chapter 5 in the current version of the Technical Standard for XML Retail Accounting Reports in the Lottery Industry, which can be found at www.opengroup.org/published. The requirements specified in Chapter 5 of the Technical Standard must be adhered to as part of conformance to the Technical Standard. It is important to note that all of the prescriptive terms found in that chapter must be interpreted according to the definitions in Section 1.3 (Terminology) of the Technical Standard.

3.2 Implement the Technical Standard

The following is a roadmap on how to implement the Technical Standard. It is a guide and not necessarily mandatory, but will help with correct implementation of the Technical Standard within your organization. Practitioners should refer to the Technical Standard to understand what the mandatory requirements are. Practitioners may choose to explicitly follow the steps as outlined in this Guide, or they may choose to combine them or do them in a different order, depending on their particular circumstances. For example, some practitioners will already have in place procedures, templates, working methods, and technology (where appropriate) that will merely need to be updated to reflect the Technical Standard; others may need to create these from scratch. The approach to implementing the Technical Standard may also be influenced by where an organization currently is in the lifecycle of activities defined by the Technical Standard. Regardless of a practitioner's current state of readiness, following all the steps as written in their entirety in the order stated will provide a deterministic roadmap to successful implementation of the Technical Standard.

3.2.1 Familiarization and Commitment

This is the starting point to implementation. It is very difficult to implement requirements that are not understood or to which staff may object on the basis of "that's not how we do things here".

All staff that will be responsible for operating under the Best Practice or working with technology that incorporates the Technical Standard should familiarize themselves with the content of the Technical Standard. It is unlikely that each individual will understand every requirement initially. There are recourses that can help with this. Team meetings will help to

ensure common understanding and it is possible that a requirement which may appear obscure to one individual is clear to another. A group discussion at this stage can help to establish common ground for the changes that will need to be made to implement the Technical Standard and can feed into the next stage in the process – the Gap Analysis.

Most importantly, the familiarization exercise should be used to identify any requirements that need explanation or clarification. The first resource to be consulted should be the Technical Standard FAQ (in Section 3.3). If an issue remains, then the next resource is to contact the Technical Standard support contact. It is far more efficient for all concerned – both the practitioner and the Certification/Verification Authority – when a requirement can be implemented correctly the first time, rather than need corrective action after formal assessment. Spending the time to fully understand the Technical Standard before starting to implement it is likely to save time overall by avoiding the need for rework.

Finally, at a team meeting it will be necessary to remove roadblocks to implementation. Many of us become fixed in the way we approach our work and can be resistant to change. For the implementation to be a success, everyone responsible for operating in accordance with the Best Practice or working with technology that incorporates the Technical Standard needs to be committed to it. This may mean certain customs and practices or technical approaches have to be abandoned or modified. It is the business practice or technical manager's duty to ensure all staff affected by the Technical Standard are committed to making it work within their organization and in their day-to-day work.

3.2.2 Gap Analysis

The gaps are the differences between the way things have been done, and are currently done, and the requirements of the Technical Standard. A gap may be a requirement of the Technical Standard which is handled some other way, is only partly met, or may not be addressed at all in the current practice.

It is recommended that, in the case of a Best Practice, a current project and/or a recently completed project, or, in the case of a Technical Standard, the current technology, be used as the basis for the gap analysis. The gap analysis is an internal informal method to establish to what extent the Technical Standard is currently applied, and to what extent existing custom and practice and/or technology must change to implement all the requirements of the Technical Standard.

3.2.2.1 *Requirements Checklist as a Tool for Gap Analysis*

Gap analysis is most readily approached by a compliance matrix between each Technical Standard requirement on the one hand, and the way things are currently done on the other.

In the case of a Best Practice, this would be a comparison of the requirements with the current project procedures, plans, specification, records, and general documentation.

In the case of a Technical Standard, this would be a matrix of the requirements and the current technology (hardware or software programs) that is currently being deployed.

Fortunately, the Requirements Checklist, found in Appendix A of the Technical Standard, can form the basis of this matrix and has already done the job of deconstructing the Technical Standard into a set of discrete stand-alone requirements.

For each requirement listed in the Requirements Checklist, the practitioners should determine which of the following categorizations apply. At this point in time it is not necessary to consider whether the requirement is categorized as “must”, “should”, or “may” in the Technical Standard; that will come later in the process.

1. **Compliant:** The practitioners believe that the processes or technical approach they normally use comply with the Technical Standard requirement and they have documents, records, or technology in which the requirement is instantiated.
2. **Partly-compliant:** The practitioners believe they meet the spirit of the requirement but they omit some of the detail or they do it in a slightly different way.
3. **Non-compliant:** They do not do it.

In addition, for every requirement the practitioner should determine if (as applicable):

1. It is realized in planning documentation and/or standard templates.
2. It is realized in project records.
3. It is realized in technology.

Finally, the practitioner or technical manager should note the status of each requirement marked partly-compliant or non-compliant.

- If its status is “must”, then this is a deficiency that has to be corrected for the organization’s implementation of the Technical Standard to be compliant.
- If the status is “should”, then the practitioner or technical manager should treat this as a strong recommendation to implement; however, if the practitioner or technical manager has a compelling reason to use an alternative method of meeting the requirement, this will not necessarily be a barrier to compliance in the future. It should be noted, though, that rationale such as “at the moment that requirement may just not be the way it is done in custom and practice” is not in itself a compelling reason to depart from the Technical Standard.
- If the status is “may”, then implementation is optional and the practitioner or technical manager might want to decide whether implementation is desirable or not.

By methodically going through each requirement in this way, it should be possible to identify the areas where the Technical Standard is not currently followed and whether there are documented processes, templates, or technology (if applicable) that need to be created or modified to ensure that the Technical Standard is followed in future projects.

3.2.3 Implementation of the Technical Standard

This section focuses on how to implement the Technical Standard requirements with an eye to being ready to apply to the formal validation program.

The bottom line of this Technical Standard is utilizing a defined XML format to deliver accounting information to retailers. Implementation requires:

- An understanding of XML
- A documented Data Source
- Understanding the NSI required XML format
- Data Mapping
- Supporting Applications
- Delivering the XML Output

3.2.3.1 XML

XML in the simplest terms is a structure for sharing data using a text file that contains tags, or markup. This structure is ideally suited to utilize prevalent web technologies for both distribution and use of the data within the structure. It is beyond the scope of this document to fully explain XML. However, the links below and the search parameters that follow those links will provide a starting point to further understand XML.

Links

- www.w3schools.com/xml/xml_what_is.asp: This is a tutorial on XML provided by the organization that established and maintains the XML standard, the W3C.
- www.w3.org/XML: This is the official location of the XML standard. It includes the technical details for existing versions as well as information about current activities.
- en.wikipedia.org/wiki/XML: This is the primary Wikipedia listing for XML. This site provides definitions and links to many of the commonly used terms and acronyms.

Search Parameters

Utilize these phrases in any search engine to get additional information on XML from various organizations including commercial sources:

- “XML Intro”
- “XML Data Mapping”
- “XML Development Tools”

3.2.3.2 Source Data Identification

The source data for this Technical Standard will reside in the lottery gaming or back-office system. A documented structure for the retailer accounting data is the best source for creating the XML files for retailers. Lacking the documented structure, reports can be a key place to begin identification of source data. Every lottery system has a means to provide accounting reports to retailers. This is typically a printout from the lottery sales terminal, such as daily activity reports,

scratch ticket inventory, and invoices. These accounting reports coincide with the information required by the Technical Standard.

The Technical Standard does not require a frequency or type of report; rather it provides a general structure to produce the reports. Additional work by NSI has established a frequency and type utilizing this Technical Standard. For details see the NSI Core Retailer Web-Based Applications Best Practice (available at www.opengroup.org/naspl/published). The summary of the frequency and type of reports included in the Best Practice are:

- Daily recap report of sales activity
- Daily scratch ticket inventory levels
- Invoices that summarize a period of activity established by each jurisdiction

This should be the starting point of source data identification. This will allow compliance with the Technical Standard and ensure consistency for future compliance with the Best Practice.

To effectively implement the Technical Standard, these source data documents need to be broken down into the individual elements or line items. In data conversion work this is called an “attribute”. For instance, total sales would be an attribute. Sales by a specific game would be an attribute, as would packs or books of an existing game. Using the reports as the source, attributes generally correspond to line items.

Some lottery or vendor systems also produce an electronic file for invoices to assist their chains and larger retailers. Other systems may produce an electronic version of daily activity and scratch inventories as well. If the system can already generate an electronic version of the accounting data, the attributes would correspond to the field names of the electronic data.

Using the report lines or the electronic field names, the existing or source data needs to be documented in a way that will allow further processing or mapping. A simple approach is to enter the attributes each report contains in a spreadsheet. Each report should be a tab in the spreadsheet and each attribute should be placed in a column with a row for each attribute. This spreadsheet will be filled out in more detail during the next step.

3.2.3.3 *NSI Required XML Format*

XML data formats are represented in two ways, either by DTD or schema. PCATS and NSI utilize a schema format.

Where to get the Files

To get the PCATS Schemas, follow the link below. At the time of this writing, the most current collection of lottery-specific information is PCATS Lottery Version 1.5 Release 4.

www.pcats.org/website/navdispatch.asp?id=717

Note: This schema is listed as “without documentation” which only refers to programmer notes referenced as comments within the schema about changes from Version 1 to Version 1.5. This information does not aid in implementation.

There are only seven schemas required by the Technical Standard. These are:

- ISO 4217:2001currency.xsd
- PCATS-NAXML-CountryCodes15.xsd
- PCATS-NAXML-LotDoc15.xsd
- PCATS-NAXML-Common15.xsd
- PCATS-NAXML-LotteryAllowanceOrChangereason15.xsd
- PCATS-NAXML-LotteryDictionary15.xsd
- PCATS-NAXML-StatesProvince15.xsd

Note: The 15 at the end of each schema name represents the version of the schema, in this case Version 1.5.

The PCATS download includes additional files you do not need. These are:

- NAXML-Acknowledgements15.xsd
- NAXML-AuthDeAuth15.xsd
- NAXML-LotDocItemSync15.xsd
- NAXML-Organization15.xsd
- NAXML-Payments15.xsd
- NAXML-Statement15.xsd

PCATS XML Schemas in Context

Six of the seven schemas required by the Technical Standard are supporting schemas; that is, they further extend and define the elements in the main schema. PCATS-NAXML-LotDoc15.xsd is the schema that will be used as the main point for creation of XML files that are compliant with the Technical Standard. This has three main elements:

- Invoice
- GameActivity
- MiscellaneousTransactions

For further details and an example of an XML file based on the schemas, see Appendix A.

History

NSI did not create the XML format for lottery accounting reports. The XML format was created by PCATS as part of a broader project designed to allow total electronic management of a retail

location and the supporting suppliers. This project supports orders, invoices, statements, payments, and the ability to update the organizational information about the businesses involved with PCATS members. NSI has worked closely with PCATS to focus in on the primary data exchange between lotteries and lottery retailers.

The Technical Standard for XML Retail Accounting Reports in the Lottery Industry focused on a subset of this functionality. The Technical Standard defined the use of attributes contained in the PCATS Schemas to further standardize the use of attributes in lottery implementation. The Technical Standard specifically adopts the following schemas in Chapter 4, NSI Adoption of the PCATS-NAXML Standard:

- ISO 4217:2001, Codes for the Representation of Currencies and Funds
- PCATS-NAXML-CountryCodes.xsd or latest version as posted on the PCATS website (www.pcats.org)
- PCATS-NAXML-LotDoc.xsd, Version 1.5.3 or latest version as posted on the PCATS website (www.pcats.org)
- PCATS-NAXML-Common.xsd, Version 1.5.1 or latest version as posted on the PCATS website (www.pcats.org)
- PCATS-NAXML-LotteryAllowanceOrChangereason.xsd, Version 1.5.2 or latest version as posted on the PCATS website (www.pcats.org)
- PCATS-NAXML-LotteryDictionary.xsd, Version 1.5.2 or latest version as posted on the PCATS website (www.pcats.org)
- PCATS-NAXMLStatesProvinces.xsd or latest version as posted on the PCATS website (www.pcats.org)

The Technical Standard effort utilized Version 1 of the PCATS Schemas which was the version available at the time of the work. Version 1.5 of the PCATS Schemas was in process and was based on inputs from NSI to bring Version 1 up to the requirements of mandatory and optional as required by the Technical Standard. This is explained in the Technical Standard, Section 5.1.2. Column 2: PCATS Mapping:

“Each row has the mapping in the PCATS XML dictionary and schema, which corresponds to the common name of the data item used in non-XML-based accounting. These items were mapped to Version 1 of the PCATS lottery document set (see “Referenced Documents”). It is a presumption of the NSI Technical Standards Working Group that there is backward compatibility between Version 1 and Version 1.5 of the document sets. Please note that an item that is optional in the PCATS mapping may be required in this Technical Standard. The section reproduced to show the mapping is exactly as found in the PCATS Schema and thus may appear to contradict the requirements of this Technical Standard.”

3.2.3.4 *Mapping to Standard Format*

These source attributes need to be cross-referenced to the data dictionary in Chapter 5 of the Technical Standard. The cross-reference is called a “mapping”. Using the spreadsheet started in

the previous step, a new column should be used to reference the number from the table in Chapter 5 of the Technical Standard.

Once every source attribute has been related to a definition in the Technical Standard and *vice versa*, you may have gaps. These gaps can be identified in two ways.

First, if there is a source attribute that does not have a corresponding reference number, then the attribute has not been related to anything in the Technical Standard. Resolve this gap by choosing an attribute in the Technical Standard that is the closest match to what the source data attribute represents.

Second, sort the spreadsheet by the column of numbers from the Technical Standard. Any resulting gaps in sequential numbering represent attributes defined in the Technical Standard that are not related to a source attribute. This can also help to identify the potential attributes that may be assigned for the previous type of gap.

To resolve these gaps, you need to go back to the Technical Standard and determine whether the attribute defined is mandatory or optional.

If the Technical Standard defines the attribute as optional, no further action is needed. If, however, the attribute is mandatory, some additional mapping needs to occur. It is possible to have a mandatory attribute that does not relate to a lottery or vendor's system. For example, if the system does not support scratch products, none of the defined mandatory attributes in the Technical Standard would have matching source data. This can still be resolved by mapping the mandatory attributes to a fixed value of zero.

The best part of this process is that the additional mapping of the definitions provided by the Technical Standard to the PCATS Schema is already done. It is part of the Technical Standard. The definitions defined in the Technical Standard form a bridge between the existing vendor or lottery accounting reports and XML schemas.

3.2.3.5 *Supporting Applications*

This is where the actual software implementation of the Technical Standard occurs. This is a specialized coding effort that takes the source data, utilizes the mapping, and produces a compliant XML file. There are various tools that can accomplish this conversion, and other approaches include customized programming. There are several approaches that can be taken, and this document will cover a few.

The first case is a best practice approach. In this case, the source data identified in the first step is used to create an XML format that directly correlates to the source data. For instance, assume that an existing system has an invoice. The lottery or vendor would create an XML schema which could be named "SourceInvoice" which is comprised of the actual line items or fields that exist in the source data for that invoice. This same process could be used for any other accounting reports that the system produces.

With the source schemas in place, the conversion of the source data to XML is now a direct process in that every attribute in the source data has a corresponding attribute in an XML schema. This keeps the customized conversion work to a minimum and resources familiar with the lottery or vendor system can focus their efforts on known formats without having to modify the actual data contents.

Now the process can be handled by XML mapping tools. These tools can be used to convert the source XML to the XML format required by the Technical Standard. Many of these tools provide a graphical interface that can accept two differing schemas and provide an on-screen display that allows a user to connect data attributes via a drag and click. Mapping tools are widely available and any search engine will provide multiple hits from a search on “XML mapping”.

The second case assumes that the lottery or vendor system is based on XML. This is an improved system and eliminates the need to create source schemas referred to in the previous example. This allows for the immediate use of the XML mapping tool as detailed in the second part of the first example.

The last style is a conversion from the source data directly to the XML formats defined by the Technical Standard. In this approach an expert familiar with the source system builds a process that converts source to XML directly. The advantage of this approach is that it is a one-step process that may be simpler to operate over time. It has the disadvantage of being expensive to modify if either the source system or the XML schemas defined in the Technical Standard change.

The Technical Standard is also silent on how these XML accounting reports are delivered. This is another area that is addressed in the NSI work on Core Retailer Web-Based Applications since the topic is better handled by a Best Practice. In short, the Best Practice utilizes a secure retailer website to allow retailers to login and access their accounting reports via the Internet.

3.2.3.6 *Delivering the XML Format*

The Technical Standard does not cover how the XML file is to be delivered to the retailer. XML can be delivered in many ways utilizing web-based technologies. These include:

- Email
- FTP sites
- Websites

In 2006, the NSI approved the Core Retailer Web-Based Applications Best Practice which does provide guidance on how to deliver the XML files defined in the Technical Standard. In short, it recommends a secure retailer-focused website that allows retailers to access their account reports directly in the form of XML files. For further details, see the Best Practice.

3.2.4 **Validation-Ready Steps**

In becoming validation-ready, it is helpful for the practitioner to have an understanding of what is required during the certification or verification process as it will help to prepare more effectively for validation. For the XML Retail Accounting Reports in the Lottery Industry, the current validation procedures call for the vendor or lottery to produce accounting data with the source gaming or back-office system and process the results as compliant XML accounting reports. At the highest level, validation will verify the following:

- Mandatory data attributes are present.

- Data is accurate.
- Format is valid, well-formed XML.
- Format complies with the PCATS Schemas.

This validation-ready period is, in a sense, preparation for these future assessments, and during that period, practitioners should be attempting to determine whether they have met the Technical Standard requirements and whether they are ready to apply for the formal validation process.

3.3 Corrigenda, Interpretations, & Frequently Asked Questions

Often, during implementation, practitioners will have questions that others have asked before them and for which there is already a response in the Frequently Asked Questions (FAQ) document, which can be found at www.opengroup.org/naspl/conformance/docs/faq.html.

If the questions and answers are not in the FAQ, the practitioner should submit their questions as follows:

- For questions about the Best Practice or Technical Standard:
nsi-specifications@opengroup.org
- For questions about the certification or verification process:
naspl-cv-auth@opengroup.org

In addition to the FAQ, it is worth noting that once an NSI Best Practice or Technical Standard has been published, changes may be needed from time-to-time. Change requests may occur when, for example:

- The relevant Working Group raises issues about the Best Practice or Technical Standard.
- An ambiguity or inconsistency is discovered when implementing the Best Practice or Technical Standard.
- The certification process results in approved interpretations against the Best Practice or Technical Standard.
- Changes in technology or operations at the lottery, vendor, or retail sites affect the Best Practice or Technical Standard as it was originally defined.

There is a documented process called the Corrigenda Process for dealing with change requests and updates to the Best Practices and Technical Standards. That process can be found at www.opengroup.org/naspl/published.

It is important that practitioners are aware that this process exists so they can check for any existing updates or interpretations they should know about while implementing the Best Practice or Technical Standard, and conversely if they have any questions during implementation, they know there is a process in place for resolution.

4 Certification/Verification Process

Once your organization has started implementing the Technical Standard, your IT manager should familiarize him/herself with the certification/verification processes, though of course you will not be able to register for certification/verification until you have completed the implementation and have determined that you are validation-ready.

The first step in the certification/verification process is for the IT manager to visit the NSI Certification/Verification website at www.opengroup.org/naspl/conformance/cert. All of the NSI Best Practices, Technical Standards, and Certification/Verification documents are available online and accessible from this website, including: Certification/Verification Policies, Conformance Requirements, Conformance Statement Questionnaires, Certification and Trademark License Agreements, Fee Schedule, Frequently Asked Questions, and User Guides.

The next step in the certification/verification process is to read the Guide to NSI Certification/Verification and the Supplement that applies to the Technical Standard you will be certifying against. These documents should be read thoroughly prior to attempting to certify a best practice or technology as they describe the program and the process in its entirety.

The following documents should be read and understood prior to certification/verification, since you will be required to agree to them during that process:

- The **NSI Certification/Verification Policy and Supplements** define the policies that govern the operation of the NSI Certification/Verification program. These policies define what can be certified, what it means to be certified, and the process for achieving and maintaining certification/verification.
- The **NSI Certification/Verification Agreement** covers the terms and conditions of the certification/verification service.

For more details on the certification/verification process, please refer to the Certification Guide, available at www.opengroup.org/naspl/conformance/cert.

If you have additional questions, please contact the NSI Certification Authority at naspl-cv-auth@opengroup.org.

It is a requirement of certification that at least two projects have been conducted in accordance with the Technical Standard and this will form a gating factor as to when the Technical Standard can be regarded as being fully implemented. It is strongly recommended that at least two complete project lifecycles be implemented in accordance with the Technical Standard prior to an application for external assessment.

5 Contact Information

For further general information on the implementation process, please contact either of the following:

- Andy White (awhite@nasplhq.org)
- Norm Day (n.day@opengroup.org)

For questions about specific requirements of the Technical Standard or the certification/verification process, please contact:

- naspl-cv-auth@opengroup.org

A Decomposing the NSI-PCATS XML Files for Lottery Retail Accounting Reports

A.1 Introduction

The main part of this document has provided information about where the PCATS Schema can be obtained, which schemas are actually part of the Technical Standard, how those schema are technically related, and a general overview of what needs to be done to implement the Technical Standard.

This appendix is focused on the actual XML file. The approach will be to examine a sample XML file that is compliant with the Technical Standard for XML Retail Accounting Reports in the Lottery Industry. By utilizing extracted portions of the sample file, each component required by the Technical Standard will be examined. The process begins at the top level of the file and then works through the required parts. In places where there are multiple options, there will be an explanation of the choices and some relation to lottery business drivers that will affect the decision.

This is by no means the only way to comply with the Technical Standard, but rather an explanation of how various lottery policies can be implemented within the Technical Standard. It is anticipated that this appendix will be modified and further enhanced by input from lotteries that implement XML retail accounting reports.

A.2 Base Components of the PCATS Schema

The table below shows the root-level elements of the NAXML-LotDoc15.xsd XML-compliant file. Following the table is a further explanation of each element and how it may be implemented.

1	<code><?xml version="1.0" encoding="UTF-8"?></code>
2	<code><pcats:NAXML-LotDoc release="3" version="1.5" xsi:schemaLocation="http://www.naxml.org/Retail-EDI/Vocabulary/2003-10-16 NAXML-LotDoc15.xsd" xmlns:pcats="http://www.naxml.org/Retail-EDI/Vocabulary/2003-10-16" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"></code>
3	<code><pcats:TransmissionHeader>Detail below</pcats:TransmissionHeader></code>
4	<code><pcats:Parties>Detail below</pcats:Parties></code>
5	<code><pcats:GameActivity>Detail below</pcats:GameActivity></code>
6	<code><pcats:Invoice>Detail below</pcats:Invoice></code>
7	<code><pcats:MiscellaneousTransactions>Detail below</pcats:MiscellaneousTransactions></code>
8	<code></pcats:NAXML-LotDoc></code>

The XML Declaration (Row 1)

This is the required initial line to every XML document. It sites the version of XML and the character encoding.

Initialization of the NAXML-LotDoc Root Element (Row 2)

Two attributes establish the version and release number of the schema. It should be noted that this file is based on the PCATS Schema from Release 4, but the release information was not updated so the release attribute is still 3. Version and release information is included to reflect the information about which schemas are in use. The extract from line 2 below reflects these attributes:

```
release="3" version="1.5"
```

The next attribute is where the schemas are located. This is a reference to the PCATS website, but keeping the required schemas in the same directory as the XML file provides direct access without the need to utilize the Internet. The attribute as set is extracted below:

```
xsi:schemaLocation="http://www.naxml.org/Retail-EDI/Vocabulary/2003-10-16 NAXML-LotDoc15.xsd"
```

The name space definitions are a critical component of compliance with the Technical Standard. XML provides a means to make elements and attributes defined within a schema unique. This is called "name space". The PCATS Schemas define two name spaces: one for PCATS which is reflected in the name of all elements by the preceding "pcats:" before each element name, and the second name space for XSI which references common attribute types defined by the W3C for use in schemas.

Name space use is a required part of the PCATS Schema and the Technical Standard. The attributes as set in the sample are extracted below:

```
xmlns:pcats="http://www.naxml.org/Retail-EDI/Vocabulary/2003-10-16"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

TransmissionHeader Element (Row 3)

Row 3 is the first of at least three required elements defined by the schema. Row 3 in the table is actually a collapsed structure of an element, specifically the TransmissionHeader. The actual TransmissionHeader element from the sample file looks like this:

```
<pcats:TransmissionHeader>  
  <pcats:TransmissionId>17298</pcats:TransmissionId>  
  <pcats:TransmissionDate>2007-02-12</pcats:TransmissionDate>  
  <pcats:TransmissionTime>14:20:00.0Z</pcats:TransmissionTime>  
  <pcats:TransmissionStatus actionType="original"/>  
</pcats:TransmissionHeader>
```

The sole purpose of this element is to provide a reference for when this XML file was transmitted and what type of transmission occurred. This can be one of the types which are:

- Original – For a first transmission.

- Resend – If the data is regenerated and resent.
- Cancel – For data that is withdrawn.

Each XML file needs a TransmissionHeader. It is a required part of the schema and has a required position as reflected in the sample data.

Parties Element (Row 4)

The Parties element is also required and needs to be in a specific order. The Parties element is used much in the same way that the address portion of correspondence is used. There are several defined types which are:

- LotteryAuthority
- Retailer
- Supplier
- Buyer
- ShipTo
- Other

As with correspondence, this provides formatted information of who sent the data and who it is intended for. This element expanded with sample data for a single location retailer would appear as follows:

```
<pcats:Parties>
  <pcats:LotteryAuthority>
    <pcats:Name>Sample Lottery</pcats:Name>
    <pcats:Address>1234 Lottery Way</pcats:Address>
    <pcats:City>Sampleton</pcats:City>
    <pcats:State>QC</pcats:State>
    <pcats:PostalCode>55555</pcats:PostalCode>
    <pcats:Contact>
      <pcats:ContactType role="Retailer Billing Contact"/>
      <pcats:Name>
        <pcats:FirstName>John</pcats:FirstName>
        <pcats:LastName>Doe</pcats:LastName>
      </pcats:Name>
      <pcats:Email>john.doe@samplelottery.org</pcats:Email>
      <pcats:Phone>(000) 555-1234</pcats:Phone>
    </pcats:Contact>
  </pcats:LotteryAuthority>
  <pcats:Retailer>
    <pcats:BillToName identType="License Number" ident="888777">
      <pcats:Name>Single Location Retailer</pcats:Name>
      <pcats:Address>123 Main</pcats:Address>
      <pcats:City>Downtown</pcats:City>
      <pcats:State>QC</pcats:State>
      <pcats:PostalCode>55511</pcats:PostalCode>
    </pcats:BillToName>
  </pcats:Retailer>
</pcats:Parties>
```

```

        <pcats:Phone>(000) 555-5555</pcats:Phone>
        <pcats:Contact>
            <pcats:ContactType role="Owner" />
            <pcats:Name><pcats:FirstName>Ted </pcats:FirstName>
                <pcats:LastName>Simth</pcats:LastName></pcats:Name>
            <pcats:Email>ted@store.com</pcats:Email>
            <pcats:Phone>(000) 555-5551</pcats:Phone>
        </pcats:Contact>
    </pcats:BillToName>
</pcats:Retailer>
<pcats:ShipTo identType="Store Location" ident="888777">
    <pcats:Name>Single Location Retailer</pcats:Name>
    <pcats:Address>123 Main</pcats:Address>
    <pcats:City>Downtown</pcats:City>
    <pcats:State>QC</pcats:State>
    <pcats:PostalCode>55511</pcats:PostalCode>
    <pcats:Phone>(000) 555-5555</pcats:Phone>
    <pcats:Contact>
        <pcats:ContactType role="Owner" />
        <pcats:Name><pcats:FirstName>Ted</pcats:FirstName>
            <pcats:LastName>Simth</pcats:LastName></pcats:Name>
        <pcats:Email>ted@store.com</pcats:Email>
        <pcats:Phone>(000) 555-5551</pcats:Phone>
    </pcats:Contact>
</pcats:ShipTo>
</pcats:Parties>

```

Note the use of the `FirstName` and `LastName` attributes in the `Contact` element of each party. These attributes are required as part of the Technical Standard but have not been flagged as required in the PCATS Schema.

The `Parties` element is required, but there is no requirement within the schema on what to include as the parties. The Technical Standard requires the lottery and bill information to be present. `ShipTo` is optional but provides clear communication about which locations are included. The above example would reflect a typical set of data for a retailer. If this were a chain billing, then there could be more `ShipTo` elements corresponding to each location that was active for the chain.

GameActivity Element (Row 5)

The `GameActivity` element is one of three elements that are considered a choice within the schema. This means that any one or more of the elements can be included. In this sample data, we are examining a file that will include all three choices so there will be `GameActivity`, `Invoice`, and `MiscellaneousTransactions`. A daily XML file would normally include simply `GameActivity`. On an invoice day, the `Invoice` element would also be included. `MiscellaneousTransactions` would only be included when needed.

`GameActivity` is used to provide retailers with information about lottery-based activity for a defined timeframe. This is most commonly a recap of daily activity. This element allows transmission of both financial information and scratch inventory levels. The financial information is much the same as the information included on an invoice, only at the granularity

of the period involved – again most commonly a business day. Scratch inventory levels are a point in time reference reflecting the information about pack or book status that is on file at the lottery at that point in time, which would normally be at the end of the period.

The `GameActivity` element when viewed in total can be confusing, so we will break this down into the component parts and explain each element. At the first level, the `GameActivity` element has the form below:

```
<pcats:GameActivity>
  <pcats:ReportPeriod>day</pcats:ReportPeriod>
  <pcats:ActivityTotals>detail below</pcats:ActivityTotals>
  <pcats:ActivityDetail>detail below</pcats:ActivityDetail>
</pcats:GameActivity>
```

The PCATS Schema does not define the `ReportPeriod` element, but the Technical Standard does at Data Dictionary item number 47. The defined periods are day, week, and month. Typically, the `GameActivity` element is used to communicate daily information. The actual date will be reflected in the activity elements explored below. The example shows the full element populated with the attribute `day` and is repeated below:

```
<pcats:ReportPeriod>day</pcats:ReportPeriod>
```

The `ActivityTotals` element provides a summary of the `GameActivity` for the period. The schema again does not require any additional information, but there would be a summary line for each `ActivityStatus` totaling all packs or books that meet the status type. The example below shows this summary:

```
<pcats:ActivityTotals>
  <pcats:TotalActivityStatus identType="" ident="">
    <pcats:ActivityStatus status="QuantityOrdered">1
    </pcats:ActivityStatus>
    <pcats:ActivityStatus status="QuantityReceived">1
    </pcats:ActivityStatus>
    <pcats:ActivityStatus status="QuantityReturned">1
    </pcats:ActivityStatus>
    <pcats:ActivityStatus status="QuantityAvailableForSale">2
    </pcats:ActivityStatus>
  </pcats:TotalActivityStatus>
</pcats:ActivityTotals>
```

The `ActivityDetails` element contains the granular detail of all of the game activity for the period reported. At the first level, this appears as follows:

```
<pcats:ActivityDetail>
  <pcats:LineItem count="1">detail below</pcats:LineItem>
  <pcats:LineItem count="2">detail below</pcats:LineItem>
  <pcats:LineItem count="3">detail below</pcats:LineItem>
  <pcats:LineItem count="4">detail below</pcats:LineItem>
  <pcats:LineItem count="5">detail below</pcats:LineItem>
  <pcats:LineItem count="6">detail below</pcats:LineItem>
  <pcats:LineItem count="7">detail below</pcats:LineItem>
  <pcats:LineItem count="8">detail below</pcats:LineItem>
</pcats:ActivityDetail>
```

Each `LineItem` element contains a specific type of game activity. This sample set includes several examples that reflect the typical range of lottery activity at a retailer. Each `LineItem` will be examined below. `LineItems` require one of two possible elements, as follows:

- `Activity`
- `InvoiceUnit`

Most lotteries provide a means for retailers to stock scratch product on a consignment basis; that is, the retailer can order, receive, and maintain adequate stock without incurring financial responsibility. As with any consigned product, the retailer still has a stewardship responsibility which is most often defined in the agreement between the lottery and the retailer. In general, the retailer only gets charged for the inventory if a pack is activated, settled, or mishandled. This will be explored further as we step through the example.

The important distinction between an `Activity` and an `InvoiceUnit` is that an `Activity` reflects a non-monetary reporting of the status of a consigned scratch product, while the `InvoiceUnit` reports on the financial aspects of a retailer's lottery-based activity during the period. The following examples utilize the `Activity` element and therefore pertain to consigned inventory levels.

`LineItem 1` expanded appears as follows:

```
<pcats:LineItem count="1">
  <pcats:GameDescription gameType="Scratch">427 - Bingo
  </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="">Pack
  </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityOrdered" UOMBasis="Pack"
      valuePerPack="125.00" >1</pcats:ActivityStatus>
    <pcats:ActivityDate status="Ordered">2007-02-11
    </pcats:ActivityDate>
  </pcats:Activity>
</pcats:LineItem>
```

All of the activity choice is based on an XML sequence; that is, a set of elements that must appear in a certain order. The above example reflects that order. The first element required is `GameDescription`. There must be one and only one of these. This includes the required attribute of `gameType`, which in this case is `Scratch`. All options for this attribute are:

- `Scratch`
- `Online`
- `Pulltab`
- `Other`

The content of `GameDescription` is a required element of both the PCATS Schema and the Technical Standard. It is defined in the Technical Standard as “Textual description of a game” in line 14 of the data dictionary in Chapter 5.

This element extracted from the example is included below:

```
<pcats:GameDescription gameType="Scratch">427 - Bingo
  </pcats:GameDescription>
```

The next required element for this `LineItem` is `GameId`. This is both a requirement of the PCATS Schema and the Technical Standard which is defined on line 15 of the NSI data dictionary as "Numeric ID for a game". The required attributes include `identType` and `ident`. Options for `identType` include:

- `Pack-Book_Number`
- `Tickets`
- `ReferenceNumber`

This is set at `Pack-Book_Number` in this example. Since this is an order, the pack ID has not been set in the `ident` attribute. However, this attribute is required so it is included without any value. The extracted element from the sample is shown again below:

```
<pcats:GameId identType="Pack-Book_Number" ident="">Pack</pcats:GameId>
```

The next required `LineItem` 1 element for inventory levels is `Activity`. This element simply contains two additional elements that provide the actual data for the `Activity`. These are `ActivityStatus` and `ActivityDate`.

`ActivityStatus` has the following optional attributes:

- `status`: Conveys the type of activity. Possible options are:
 - `QuantityAvailableForSale`
 - `QuantityDamaged`
 - `QunatityCommitted`
 - `QuantityLost`
 - `QuantityOrdered`
 - `QuantityRecieved`
 - `QuantityReturned`
 - `QuantityShipped`
 - `QuantitySold`
 - `QuantityStolen`
 - `QuantityInTransit`
 - `SettledByStore`
 - `SettledByLottery`

- ForceSettlementPercent
- ForceSettlementByDay
- UOMBasis (Unit Of Measure Basis)
 - Pack
 - Each
- valuePerPack: The value in monetary terms of each pack.
- type: General type not defined further.

These attributes cover a wide range of options and may not all apply at each lottery. However, they do constitute a superset of the status used across lotteries. Additionally, each lottery may have a unique twist on how a status is used but this will still convey the needed information to the retailer about the status of consigned scratch product within the jurisdiction. The extracted section of the sample is shown below:

```
<pcats:ActivityStatus status="QuantityOrdered" UOMBasis="Pack"
  valuePerPack="125.00" >1</pcats:ActivityStatus>
```

Note that while ActivityStatus is required, there are no formal requirements for the attributes. In fact, this element could be empty and be valid. However, to provide the best information consider including these elements in your XML.

ActivityDate has the following required attributes:

- Status: The status at the date specified:
 - Ordered
 - Issued
 - Activated
 - Settlement
 - Shipped
 - Received
 - Returned

The value included in the ActivityDate should be the actual date of the activity. This date must fall within the period of activity. The extracted section of the sample is included below:

```
<pcats:ActivityDate status="Ordered">2007-02-11</pcats:ActivityDate>
```

It should be noted that not all lotteries will use all of these status types. From the standpoint of implementing the Technical Standard in XML, a lottery should use the types of activity for all scratch inventory that has not transferred to a billable status. There are also two optional attributes of the ActivityDate element. These are:

- method

- referenceNumber

Each attribute would hold slightly different information based on the status. For instance, in this example of Ordered, the method could be phone, online, or auto. The referenceNumber could be a confirmation number for the order. This could look like this:

```
<pcats:ActivityDate status="Ordered" method="phone"
  referenceNumber="F54-21">2007-02-11</pcats:ActivityDate>
```

LineItems 2 and 3 provide additional examples of GameActivity. They follow the same sequence as the first line item and simply convey additional examples of a Status type.

```
<pcats:LineItem count="2">
  <pcats:GameDescription gameType="Scratch">394 -Pogo
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="27">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityReceived" UOMBasis="Pack"
      ValuePerPack="450.00">1</pcats:ActivityStatus>
    <pcats:ActivityDate status="Received">2007-02-11
      </pcats:ActivityDate>
    </pcats:Activity>
  </pcats:LineItem>
<pcats:LineItem count="3">
  <pcats:GameDescription gameType="Scratch">401 Lucky Sevens
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="1263">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityReturned" UOMBasis="Pack"
      valuePerPack="300.00">1</pcats:ActivityStatus>
    <pcats:ActivityDate status="Returned">2007-02-11
      </pcats:ActivityDate>
    </pcats:Activity>
  </pcats:LineItem>
```

Finally, the Activity sequence also includes several other optional items, as follows:

- OnlineReplay
- InstantCoupons
- DamagedTicketValue
- ReturnAmt

These are valid items, but can run into complexity from both an implementation and lottery policy standpoint. These will be dealt with in Section A.4.

Once scratch tickets change from a lottery consignment to a retailer owned and sold lottery product, a billing event occurs. There are several possible cases, such as:

- Sales tracked at a ticket level

- Pack Activate/Settle immediately
- Activate start a delayed billing

If actual ticket sales are tracked, then this would result in an entry in the form of an InvoiceUnit for each day's activity by game. If there is a method where making a pack available for sale by activation also results in a billable amount the same day, then this type of event would also be handled as an InvoiceUnit discussed below.

If, however, the activation making a pack available for sale starts a timer for a delay billing amount, then there is a need to track the packs and their targeted billing date so that retailers can plan their cash flow. This last type of transaction would still be represented in a GameActivity entity. Samples of entries for activation and another for future settlement are reflected here:

```
<pcats:LineItem count="4">
  <pcats:GameDescription gameType="Scratch">401 - Lucky Sevens
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="1263">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityAvailableForSale"
      UOMBasis="Pack" valuePerPack="300.00">1
    </pcats:ActivityStatus>
    <pcats:ActivityDate status="Activated" method="Auto-Activated"
      referenceNumber="6517234">2007-02-11</pcats:ActivityDate>
  </pcats:Activity>
</pcats:LineItem>
<pcats:LineItem count="5">
  <pcats:GameDescription gameType="Scratch">337 - Slingo
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="2315">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityAvailableForSale"
      UOMBasis="Pack" valuePerPack="750.00">1
    </pcats:ActivityStatus>
    <pcats:ActivityDate status="Settlement">2007-02-25
    </pcats:ActivityDate>
  </pcats:Activity>
</pcats:LineItem>
```

LineItem 4 shows a pack that was activated via auto-activate where the lottery activates a pack based on the first redemption. This shows the pack as QuantityAvailableForSale. This sample lottery also defers billing and the entry for LineItem 4 would have an entry similar to LineItem 5 on the very next business day. In LineItem 5, the pack also has a status of QuantityAvailableForSale, but now reflects a future activity date for the settlement or billing event of February 25th 2007. On the 25th, LineItem 5 would become an InvoiceUnit for the day and bill out the pack value. LineItem 8 below is an example of this type of entry.

This element is used here and also used within the `Invoice` element. The only difference is timing or period. In the context of `GameActivity`, the reporting period can be day, week, or month. In the context of the `Invoice`, the reporting will be for the lottery-defined invoice period. In the sample provided here, which is also consistent with the requirements of the NSI Core Retailer Web-Based Applications Best Practice, the `GameActivity` is reported on a business day basis. The lottery still defines the business day, but the concept is to provide timely information on a daily basis to retailers about their lottery-based activity.

From an integrity standpoint, a retailer should be able to add up all of the `InvoiceUnit` information provided by a `GameActivity` element for each day of the invoice period and match the `InvoiceUnit` elements reported in an `Invoice` element. Thus, the daily XML file provides the retailer with all of the pertinent information to successfully account for both consigned scratch product and lottery activity.

The `InvoiceUnit` is an XML sequence. The required elements in order for this sequence are listed below:

- `GameDescription`
- `InvoiceUnitQuantity`
- `LineItemGrossAmt`
- `LineItemNetAmt`

Optional elements of the schema are:

- `InvoiceUnitCost`
- `RetailUnitQty`
- `TicketCount`
- `RetailPrice`
- `DateSold`
- `DateSettled`
- `DateRedeemed`
- `AllowanceOrCharge`
- `PackNumber`
- `TicketNumber`

LineItem 6 of the sample data shows the XML format of a typical `InvoiceUnit`:

```
<pcats:LineItem count="6">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Online">Lotto
  </pcats:GameDescription>
```

```

    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
      </pcats:InvoiceUnitQty>
    <pcats:DateSold>2007-02-11</pcats:DateSold>
    <pcats:LineItemGrossAmt currency="USD">100.00
      </pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt currency="USD">100.00
      </pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
  </pcats:LineItem>

```

Working through this example, we see GameDescription which is the exact same element discussed above in the description of GameActivity and has the same purpose. In this case, the game is an Online type with a name Lotto.

InvoiceUnitQty is required, but there is no requirement for included attributes. The optional attribute lotteryUOMBasis is actually a loosely defined UOMBasis (Unit Of Measure Basis) that can reflect the way in which a lottery handles quantity. This can be as shown above, simply a gross sales definition with a value of 1. It could also be a per-ticket unit of measure and show the actual count of tickets sold. For instance, if the ticket price is \$2 then the quantity would be 50 in this example.

DateSold is an optional element, but is included here to provide clarity of the actual activity date since the period is referenced as day.

LineItemGrossAmt is required and shows the gross sale amount of the game. If a lottery pays commissions based on individual games, there is a means to reflect the commission amount within the invoice line item. This involves the use of the AllowanceOrCharge element and will be covered in Section A.4.

LineItemNetAmt is also required and in this example shows the same values as the LineItemGrossAmt with the assumption that commissions or retailer payment will be handled on a separate line item.

There would be a similar line item for each of the lottery online games sold by the retailer during the period. Any other product that is simply a function of straight retailer sales would also be represented here, such as the example of sale of a pulltab game to a retailer per LineItem 5 below:

```

<pcats:LineItem count="7">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="PullTab">Pot of Gold
      </pcats:GameDescription>
    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
      </pcats:InvoiceUnitQty>
    <pcats:DateSold>2007-02-11</pcats:DateSold>
    <pcats:LineItemGrossAmt currency="USD">400.00
      </pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt currency="USD">400.00
      </pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
  </pcats:LineItem>

```

Again, this example assumes retailer compensation as a separate line item. Pulltabs can be sold at a discount to include an allowance for prizes and retailer payment. This will be handled in Section A.4.

For packs that are settled and now due, a LineItem such as the one below in the sample set would be generated:

```
<pcats:LineItem count="8">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Scratch">401 - Lucky Sevens
    </pcats:GameDescription>
    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
    </pcats:InvoiceUnitQty>
    <pcats:DateSettled>2007-02-11</pcats:DateSettled>
    <pcats:LineItemGrossAmt currency="USD">200.00
    </pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt currency="USD">200.00
    </pcats:LineItemNetAmt>
  </pcats:InvoiceUnit>
</pcats:LineItem>
```

In this example, the game 401 - Lucky Sevens was settled on this day and now due. On the previous day there would be an entry in the GameActivity section for the pack as discussed above.

The last examples in the GameActivity section focus on validations. The examples below show validations for low and mid-tier prizes redeemed by the retailer:

```
<pcats:LineItem count="9">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Scratch">Validations Summary
    </pcats:GameDescription>
    <pcats:InvoiceUnitQty/>
    <pcats:TicketCount>41</pcats:TicketCount>
    <pcats:DateRedeemed>2007-02-11</pcats:DateRedeemed>
    <pcats:AllowanceOrCharge>
      <pcats:AllowanceOrChargeReason identType="LowTier"/>
      <pcats:AllowanceAmt UOMBasis="Total">-76.00
      </pcats:AllowanceAmt>
    </pcats:AllowanceOrCharge>
    <pcats:LineItemGrossAmt>-76.00</pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt>-76.00</pcats:LineItemNetAmt>
  </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="10">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Scratch">Validations Summary
    </pcats:GameDescription>
    <pcats:InvoiceUnitQty/>
    <pcats:TicketCount>1</pcats:TicketCount>
    <pcats:DateRedeemed>2007-02-11</pcats:DateRedeemed>
    <pcats:AllowanceOrCharge>
      <pcats:AllowanceOrChargeReason identType="MidTier"/>
      <pcats:AllowanceAmt UOMBasis="Total">-25.00
```

```

        </pcats:AllowanceAmt>
    </pcats:AllowanceOrCharge>
    <pcats:LineItemGrossAmt>-25.00</pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt>-25.00</pcats:LineItemNetAmt>
</pcats:InvoiceUnit>
</pcats:LineItem>

```

In both of these examples, the AllowanceOrCharge element is utilized. The allowance portion is for amounts credited to the retailer, while the charge concept is used for additional charges to the retailer, such as line charges. The AllowanceOrCharge element is a sequence that consists of a required AllowanceOrChargeReason and then either the AllowanceAmt as shown or the ChargeAmt. The Technical Standard requires that both low and mid-tier redemptions are represented in the XML file.

Invoice Element (Row 6)

The Invoice element is designed to contain the information about an actual lottery invoice for a retailer. The lottery maintains the invoice period, the date the invoice is sent to the retailer, and the due date of the invoice. The Invoice element is an XML sequence with the following format as populated with sample data:

```

<pcats:Invoice>
    <pcats:InvoiceNumber>775432</pcats:InvoiceNumber>
    <pcats:InvoiceDate>2007-02-12</pcats:InvoiceDate>
    <pcats:SweepDate>2007-02-14</pcats:SweepDate>
<pcats:InvoiceDetail>More Detail Below</pcats:InvoiceDetail>
</pcats:Invoice>

```

InvoiceNumber and InvoiceDate are required elements, while the SweepDate is optional. In this example we are continuing with the theme of the sample XML file for a single location retailer. Chains and multiple locations are covered in detail below. This will cover the choice of the LocationInvoiceData element.

Since this sample is a single location, the next part of the Invoice element is the InvoiceDetail. InvoiceDetail has a sequence that consists of the now familiar LineItem covered in our example above and an optional element of the schema, InvoiceSummary. InvoiceSummary is the new element not included in GameActivity and while optional within the schema, becomes required via the Technical Standard since both game-level and product-level GrossSales totals are required, which would be reflected in the InvoiceSummary. First, however, the LineItem element will be explored.

As previously stated, the LineItem element is identical to the LineItem element covered in the ActivityDetail example. The only difference is the period covered. As part of the Invoice element, the period covered is defined by the lottery as an invoice period. So invoice data reflects the same type of information already covered in our example, reflecting the financial activity for the retailer during the invoice period via the InvoiceUnit and the status of consigned scratch product at the end of the invoice period via the Activity element. A populated example would look like this:

```

<pcats:LineItem count="11">

```

```

<pcats:GameDescription gameType="Scratch">394 - Pogo
  </pcats:GameDescription>
<pcats:GameId identType="Pack-Book_Number" ident="5231">Pack
  </pcats:GameId>
<pcats:Activity>
  <pcats:ActivityStatus status="QuantityReceived" UOMBasis="Pack"
    valuePerPack="450.00">1</pcats:ActivityStatus>
  <pcats:ActivityDate status="Received">2007-02-06
    </pcats:ActivityDate>
  </pcats:Activity>
</pcats:LineItem>
<pcats:LineItem count="12">
  <pcats:GameDescription gameType="Scratch">401 Lucky Sevens
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="3219">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityReceived" UOMBasis="Pack"
      valuePerPack="300.00">1</pcats:ActivityStatus>
    <pcats:ActivityDate status="Received">2007-02-07
      </pcats:ActivityDate>
    </pcats:Activity>
  </pcats:LineItem>
<pcats:LineItem count="13">
  <pcats:GameDescription gameType="Scratch">401 - Lucky Sevens
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="1263">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityAvailableForSale"
      UOMBasis="Pack" valuePerPack="300.00">1
      </pcats:ActivityStatus>
    <pcats:ActivityDate status="Activated" method="Via Terminal"
      referenceNumber="23167">2007-02-07</pcats:ActivityDate>
    </pcats:Activity>
  </pcats:LineItem>
<pcats:LineItem count="14">
  <pcats:GameDescription gameType="Scratch">337 - Slingo
    </pcats:GameDescription>
  <pcats:GameId identType="Pack-Book_Number" ident="2315">Pack
    </pcats:GameId>
  <pcats:Activity>
    <pcats:ActivityStatus status="QuantityAvailableForSale"
      UOMBasis="Pack" valuePerPack="750.00">1
      </pcats:ActivityStatus>
    <pcats:ActivityDate status="Settlement">2007-02-15
      </pcats:ActivityDate>
    </pcats:Activity>
  </pcats:LineItem>
<pcats:LineItem count="15">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Online">Lotto
      </pcats:GameDescription>
    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1

```

```

        </pcats:InvoiceUnitQty>
        <pcats:LineItemGrossAmt currency="USD">5100.00
        </pcats:LineItemGrossAmt>
        <pcats:LineItemNetAmt currency="USD">5100.00
        </pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="16">
    <pcats:InvoiceUnit>
        <pcats:GameDescription gameType="Scratch">401 - Lucky Sevens
        </pcats:GameDescription>
        <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
        </pcats:InvoiceUnitQty>
        <pcats:DateSettled>2007-02-09</pcats:DateSettled>
        <pcats:LineItemGrossAmt currency="USD">200.00
        </pcats:LineItemGrossAmt>
        <pcats:LineItemNetAmt currency="USD">200.00
        </pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="17">
    <pcats:InvoiceUnit>
        <pcats:GameDescription gameType="Scratch">401 - Lucky Sevens
        </pcats:GameDescription>
        <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
        </pcats:InvoiceUnitQty>
        <pcats:DateSettled>2007-02-10</pcats:DateSettled>
        <pcats:LineItemGrossAmt currency="USD">200.00
        </pcats:LineItemGrossAmt>
        <pcats:LineItemNetAmt currency="USD">200.00
        </pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="20">
    <pcats:InvoiceUnit>
        <pcats:GameDescription gameType="Scratch">Validations Summary
        </pcats:GameDescription>
        <pcats:InvoiceUnitQty/>
        <pcats:TicketCount>41</pcats:TicketCount>
        <pcats:AllowanceOrCharge>
            <pcats:AllowanceOrChargeReason identType="LowTier"/>
            <pcats:AllowanceAmt UOMBasis="Total">-15.00
            </pcats:AllowanceAmt>
        </pcats:AllowanceOrCharge>
        <pcats:LineItemGrossAmt>-15.00</pcats:LineItemGrossAmt>
        <pcats:LineItemNetAmt>-15.00</pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="21">
    <pcats:InvoiceUnit>
        <pcats:GameDescription gameType="Scratch">Validations Summary
        </pcats:GameDescription>
        <pcats:InvoiceUnitQty/>
        <pcats:TicketCount>1</pcats:TicketCount>
    </pcats:InvoiceUnit>
</pcats:LineItem>

```

```

    <pcats:AllowanceOrCharge>
      <pcats:AllowanceOrChargeReason identType="MidTier"/>
      <pcats:AllowanceAmt UOMBasis="Total">-50.00
    </pcats:AllowanceAmt>
    </pcats:AllowanceOrCharge>
    <pcats:LineItemGrossAmt>-50.00</pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt>-50.00</pcats:LineItemNetAmt>
  </pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="22">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Online"/>
    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
  </pcats:InvoiceUnitQty>
  <pcats:AllowanceOrCharge>
    <pcats:AllowanceOrChargeReason identType="SalesCommission"/>
    <pcats:AllowanceAmt UOMBasis="Total">-306.00
  </pcats:AllowanceAmt>
  </pcats:AllowanceOrCharge>
  <pcats:LineItemGrossAmt>-306.00</pcats:LineItemGrossAmt>
  <pcats:LineItemNetAmt>-306.00</pcats:LineItemNetAmt>
</pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="23">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Online"/>
    <pcats:InvoiceUnitQty lotteryUOMBasis="Redeemed">1
  </pcats:InvoiceUnitQty>
  <pcats:AllowanceOrCharge>
    <pcats:AllowanceOrChargeReason identType="Redemptions"/>
    <pcats:AllowanceAmt UOMBasis="Total">-240.00
  </pcats:AllowanceAmt>
  </pcats:AllowanceOrCharge>
  <pcats:LineItemGrossAmt>-240.00</pcats:LineItemGrossAmt>
  <pcats:LineItemNetAmt>-240.00</pcats:LineItemNetAmt>
</pcats:InvoiceUnit>
</pcats:LineItem>
<pcats:LineItem count="24">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Scratch"/>
    <pcats:InvoiceUnitQty lotteryUOMBasis="GrossSales">1
  </pcats:InvoiceUnitQty>
  <pcats:AllowanceOrCharge>
    <pcats:AllowanceOrChargeReason identType="SalesCommission"/>
    <pcats:AllowanceAmt UOMBasis="Total">-40.00
  </pcats:AllowanceAmt>
  </pcats:AllowanceOrCharge>
  <pcats:LineItemGrossAmt>-40.00</pcats:LineItemGrossAmt>
  <pcats:LineItemNetAmt>-40.00</pcats:LineItemNetAmt>
</pcats:InvoiceUnit>
</pcats:LineItem>

```

All of these LineItems are consistent with previously covered explanations. In terms of the invoice period though, there are some potential differences. First, neither the PCATS Schema

nor the Technical Standard dictates the frequency of data that is represented in the invoice. It would be proper to show sales for products on a daily basis, one entry summarizing each day, or to provide the total product activity for the invoice. Second, scratch product for the invoice would be identical to the daily activity for scratch on the day the invoice period closed. Finally, sales commission calculations may not permit a daily calculation, but the commission amount is required on the invoice.

The final element within the InvoiceDetail is the InvoiceSummary element. This is a recap of all of the financial data of the invoice. The extract from the sample XML appears below:

```
<pcats:InvoiceSummary>
  <pcats:InvoiceTotals>
    <pcats:LineItems count="13" />
    <pcats:TotalLineItemNetAmt identType="Debit">5120.00
      </pcats:TotalLineItemNetAmt>
    <pcats:AllowancesAndCharges>
      <pcats:OtherAllowanceOrCharge>
        <pcats:AllowanceOrCharge>
          <pcats:AllowanceOrChargeReason identType="LineCharge" />
          <pcats:ChargeAmt>35.00</pcats:ChargeAmt>
        </pcats:AllowanceOrCharge>
      </pcats:OtherAllowanceOrCharge>
    </pcats:AllowancesAndCharges>
    <pcats:ActivityAndAllowanceOrChargeTotals>
      <pcats:ActivityTransaction identType="Online Sales">5100.00
        </pcats:ActivityTransaction>
      <pcats:ActivityTransaction identType="Settled">400.00
        </pcats:ActivityTransaction>
      <pcats:ActivityTransaction identType="Cashed/Redemptions">
        -380.00</pcats:ActivityTransaction>
      <pcats:ActivityTransaction identType="Other">35.00
        </pcats:ActivityTransaction>
      <pcats:TotalActivityTransactionNetAmt>5500.00
        </pcats:TotalActivityTransactionNetAmt>
      <pcats:TotalAllowancesAndChargesNetAmt>-345.00
        </pcats:TotalAllowancesAndChargesNetAmt>
    </pcats:ActivityAndAllowanceOrChargeTotals>
    <pcats:TotalInvoiceDueAmt identType="Debit">5155.00
      </pcats:TotalInvoiceDueAmt>
  </pcats:InvoiceTotals>
</pcats:InvoiceSummary>
```

Stepping through the above example, it begins with an element LineItems which is the total count of line items in the invoice.

Note: The line count increments across both GameActivity and the invoice, but this total includes invoice LineItems only.

TotalLineItemNetAmt is simply the sum of all the LineItemNetAmt elements of each InvoiceUnit. The AllowancesAndCharges element allows for non-game-related charges or allowances (credits) to be issued to the retailer. In the example, a line charge is assessed. The ActivityAndAllowanceOrChargeTotals element is optional in the

PCATS Schema, but meets the required part of the Technical Standard to show gross sales by product type. The TotalActivityTransactionNetAmt and the TotalAllowancesAndChargesNetAmt elements sub-total the Invoice and the final element TotalInvoiceDueAmt reflects the amount the retailer owes the lottery.

MiscellaneousTransactions (Row 7)

MiscellaneousTransactions are not well defined in either the PCATS Schema or the Technical Standard. They are free-form transactions that can contain any data. The XML structure is:

```
<pcats:MiscellaneousTransactions>
  <pcats:MiscellaneousTransactionDescription>
    </pcats:MiscellaneousTransactionDescription>
</pcats:MiscellaneousTransactions>
```

This is an optional element so there is no need to include this in the file.

Close of NAXML-LotDoc Root Element (Row 8)

This is simply the closing tag of the NAXML-LotDoc element. This schema is defined for one and only one NAXML-LotDoc element per file. One possible future enhancement would be to define a simple schema that would allow multiple NAXML-LotDoc elements. This could allow consolidation of the daily or invoice period output from a gaming or back-office system into a single file for transfer to a web-based application that would provide the parsed NAXML-LotDoc files to the proper retailer, maintain a history, and otherwise comply with requirements of the NSI Core Retailer Web-Based Applications Best Practice.

A.3 Chain or Multiple Location XML Reports

Chain or multiple location retailers simply stated have multiple retail locations with a common owner or billing address. In this case, the NAXML-LotDoc element has the capability to report the information about all of the locations in a single XML file. The actual detail for each location is exactly the same in format and requirements as the XML example stated above. For chains, the only real difference is including optional information as part of the schema that differentiates the locations.

GameActivity has an optional element in the sequence that allows location information to be entered. In the XML it looks like this:

```
<pcats:ActivityDetail>
  <pcats:Location>Info About the Location</pcats:Location>
  <pcats:LocationActivityTotals>
    <pcats:TotalLocationActivityStatus identType="Scratch"
      ident="Summary">
      <pcats:ActivityStatus></pcats:ActivityStatus>
    </pcats:TotalLocationActivityStatus>
  </pcats:LocationActivityTotals>
<pcats:LineItem count="N">
```

```

        <pcats:InvoiceUnit>
            <pcats:GameDescription gameType=""></pcats:GameDescription>
            <pcats:InvoiceUnitQty></pcats:InvoiceUnitQty>
            <pcats:LineItemGrossAmt></pcats:LineItemGrossAmt>
            <pcats:LineItemNetAmt></pcats:LineItemNetAmt>
        </pcats:InvoiceUnit>
    </pcats:LineItem>
</pcats:ActivityDetail>

```

This sequence can be repeated for each retail location included in the file. The Location element has the following attributes that can be included:

- Name
- Address
- City
- State
- PostalCode
- PostalCodeExt
- CountryCode
- Fax
- Email
- Phone
- Contact
- OrganizationalID

LocationActivityTotals provides the location-specific summary similar in content to the ActivityTotals covered in the XML file example above.

The Invoice element has a similar but slightly different approach. The LocationInvoiceData element is a container that provides a sequence detailed in the XML below.

```

<pcats:Invoice>
    <pcats:InvoiceNumber>775432</pcats:InvoiceNumber>
    <pcats:InvoiceDate>2007-02-12</pcats:InvoiceDate>
    <pcats:SweepDate>2007-02-14</pcats:SweepDate>
    <pcats:LocationInvoiceData>
        <pcats:Location></pcats:Location>
        <pcats:LocationInvoiceDetail>
            <pcats:LineItem count="">
                <pcats:GameDescription gameType=""></pcats:GameDescription>
                <pcats:GameId identType="" ident=""></pcats:GameId>
            </pcats:LineItem>
        </pcats:LocationInvoiceDetail>
    </pcats:LocationInvoiceData>
</pcats:Invoice>

```

```

        <pcats:ActivityStatus></pcats:ActivityStatus>
        <pcats:ActivityDate status=""></pcats:ActivityDate>
    </pcats:Activity>
</pcats:LineItem>
<pcats:LineItem count="">
    <pcats:InvoiceUnit>
        <pcats:GameDescription gameType="">
            </pcats:GameDescription>
        <pcats:InvoiceUnitQty></pcats:InvoiceUnitQty>
        <pcats:LineItemGrossAmt></pcats:LineItemGrossAmt>
        <pcats:LineItemNetAmt></pcats:LineItemNetAmt>
    </pcats:InvoiceUnit>
</pcats:LineItem>
    <pcats:LocationInvoiceSummary>
        <pcats:LocationInvoiceTotals>
            <pcats:LineItems count=""></pcats:LineItems>
            <pcats:LocationTotalLineItemNetAmt>
                </pcats:LocationTotalLineItemNetAmt>
            <pcats:AllowancesAndCharges>
                </pcats:AllowancesAndCharges>
            <pcats:LocationTotalInvoiceDueAmt identType="">
                </pcats:LocationTotalInvoiceDueAmt>
            </pcats:LocationInvoiceTotals>
        </pcats:LocationInvoiceSummary>
    </pcats:LocationInvoiceDetail>
</pcats:LocationInvoiceData>
</pcats:Invoice>

```

The LocationInvoiceData element can be repeated within the Invoice structure for each location included in the chain file. The LocationInvoiceData element is a sequence which provides the Location element. The Location element is exactly the same as the Location element in the GameActivity for chains described above. Under that is the LocationInvoiceDetail element that is a container for the LineItem elements covered in the XML file example above. The LineItems are followed by a LocationInvoiceTotals which is a location-specific listing that holds the same information as the InvoiceTotals in the XML file example above.

A.4 Advanced Concepts

These concepts are included here because they vary widely by lottery based on statute, regulation, policy, or simply practice. Following the 80-20 rule, the 80% of things most lotteries have were included in the examples above while the 20% that vary widely are covered here.

Coupons/Promotions

The schema allows for reporting of activity on both instant coupons and promotions, such as online replay, etc. These non-monetary amounts can be included in the elements below:

```

<pcats:InstantCoupons></pcats:InstantCoupons>
<pcats:OnlineReplay></pcats:OnlineReplay>

```

The lottery would define the terms and details of each promotion or coupon offering. These details would drive the data that would be contained in these elements. This could be an equivalent dollar value, or simply a count of the coupon or replay.

Other Ticket Ownership Changes and/or Credits

Scratch products on consignment and product that is available for sale can be affected by many interesting and sometimes creative events. This section deals with events that cause the scratch product to be unsellable and how that affects the retailer. Keep in mind that lottery practice drives how these events are handled. Examples of these events include:

- Damaged Tickets
- Stolen Tickets
- Missing Tickets

If the pack becomes unsellable prior to activation, most lotteries have the means to flag the tickets in the pack to reflect the unsellable status. This becomes a `GameActivity LineItem` element that moves the pack out of the retailer location and back to the lottery. No monetary effect occurs. If the pack has been activated, then credit needs to be issued if lottery policy allows. This credit is handled just like a return detailed below.

Returns

Returns of scratch product can be interesting due to the possible states of the returned pack. In the XML file example, a case of a pack that had not been activated, but returned. This is the simplest case and has no monetary effect. In other instances, there is a financial impact. If the pack is settled, the process is again straightforward. A credit is issued for the amount paid and the pack is returned. If, however, the pack is activated but not yet settled, then several events need to occur to balance out the transaction. First, the pack needs to be settled. Then, the credit can be issued to offset the charge.

In all of these examples, the schema has the capacity to handle the activity. The XML file showed the consigned pack return. In the case of a returned pack that has already been settled, an `AllowanceOrCharge` element of an `InvoiceUnit` would be used to issue an `AllowanceAmt` for the returned pack. A populated example is shown below:

```
<pcats:LineItem count="10">
  <pcats:InvoiceUnit>
    <pcats:GameDescription gameType="Scratch">Returned Pack
    </pcats:GameDescription>
    <pcats:InvoiceUnitQty/>
    <pcats:DateSettled>2007-02-05</pcats:DateSettled>
    <pcats:AllowanceOrCharge>
      <pcats:AllowanceOrChargeReason identType="Adjustments"/>
      <pcats:AllowanceAmt UOMBasis="Total">-425.00
      </pcats:AllowanceAmt>
    </pcats:AllowanceOrCharge>
    <pcats:LineItemGrossAmt>-425.00</pcats:LineItemGrossAmt>
    <pcats:LineItemNetAmt>-425.00</pcats:LineItemNetAmt>
  </pcats:InvoiceUnit>
</pcats:LineItem>
```

```
</pcats:InvoiceUnit>
</pcats:LineItem>
```

Notice the DateSettled element which confirms the pack was settled and charged. The only difference for a pack that was not settled would be to force the settlement for the same invoice period as the return.

Commissions and Other Allowances by Product

The XML file example showed a simple case where sales were reported separately as a LineItem element and later validations, commissions, and other activity were reported on a separate LineItem. The PCATS Schema also allows for these allowances to be included in a single LineItem. The example below shows this structure:

```
<LineItem count="41">
  <InvoiceUnit>
    <GameDescription gameType="Online">Lotto</GameDescription>
    <InvoiceUnitQty lotteryUOMBasis="GrossSales">1</InvoiceUnitQty>
    </InvoiceUnitCost>
    <DateSold>2007-02-10</DateSold>
    <AllowanceOrCharge>
      <AllowanceOrChargeReason identType="SalesCommission"/>
      <AllowanceAmt UOMBasis="Total">-30.00</AllowanceAmt>
    </AllowanceOrCharge>
    <AllowanceOrCharge>
      <AllowanceOrChargeReason identType="Redemptions"/>
      <AllowanceAmt UOMBasis="Total">-67.00</AllowanceAmt>
    </AllowanceOrCharge>
    <AllowanceOrCharge>
      <AllowanceOrChargeReason identType="CancelledTickets"/>
      <AllowanceAmt UOMBasis="Total">-3.00</AllowanceAmt>
    </AllowanceOrCharge>
    <LineItemGrossAmt>500.00</LineItemGrossAmt>
    <LineItemNetAmt>400.00</LineItemNetAmt>
  </InvoiceUnit>
</LineItem>
```