



Michael & Susan Dell
FOUNDATION



Scholar Snapp™

Technical Implementation Guide

v1

Contents

About This Guide.....	4
The Problem to Solve	5
The Scholar Snapp™ Solution.....	6
How the Scholar Snapp™ solution was created	7
Student information carried in SDS files.....	7
The Scholar Snapp™ Schema	8
Schema Contents.....	8
Minimum required information in an SDS file	9
Considerations for various organization types	9
Scholarship providers with a single scholarship application.....	9
Scholarship application sourcing.....	9
Considerations for application internal processing.....	9
Considerations for application external sharing	10
Scholarship providers with multiple scholarship applications.....	10
Considerations.....	10
Scholarship management organizations with a single application	11
Scholarship listing, matching, or referral services.....	11
Considerations.....	12
How to adopt Scholar Snapp™.....	13
Prerequisites	13
Process Overview	13
Mapping your database to Scholar Snapp™	14
Software architecture for supporting the Scholar Snapp™ solution.....	15
Controlling access to private student data	16
Methods to receive applications automatically	17
Filtering for Eligibility.....	17
Student Custom Identifiers.....	18
Acknowledging Received Scholarship Applications.....	18

Frequently Asked Questions	19
Glossary	23
Appendix A: the Scholar SnappSchema	24

About This Guide

This guide helps people in a technical role to understand the key concepts, planning, and resources necessary to use the Scholar Snapp™ solution to help students access more scholarship funds for education. The Scholar Snapp solution was previously known as the Scholarship Data Standard. To use this guide, it is assumed that you have read the Scholar Snapp introduction available on the Scholar Snapp website at www.scholarsnapp.org. There are downloadable documents and discussion forums on the NSPA Web site where the Scholar Snapp solution adopters can get peer help and advice about implementing Scholar Snapp.

The Problem to Solve

Students spend many hours filling out scholarship applications in an effort to get money for college. Not only does one application require significant time, but also each application requires manual data entry regardless of how many questions it shares with a previous application. For example, the student must redundantly enter their name, address, and contact information, then repeat the process for nearly every application. Some students will employ awkward cut and paste mechanisms to try to re-use their information in multiple places. While the requirements of one application are not too much to bear, consider the labor when a student attempts to apply for forty scholarships. Students individually decide when “enough is enough” and they stop applying, mainly based on their threshold for the time investment they are willing to make.

Rightfully so, scholarship providers are selective about which students to whom they award scholarships, but collectively we have created a significant burden upon those we wish to help. When asking for the best candidates for our scholarships, we should employ the best processes and technology where possible. Our goal should be to maximize the ratio of seized student opportunities to student labor expenditure. The Scholar Snapp™ solution makes it possible to do this with technology.

The Scholar Snapp™ Solution

The Scholar Snapp™ solution lets students apply for a scholarship online and then automatically re-use that information for other scholarship applications. As a result, the student has access to considerably more scholarships with less time and labor investment. The Scholar Snapp solution achieves this without requiring scholarship applications to be aligned with each other or structured in a similar way. Since the Scholar Snapp solution is a common standard, it gives scholarship providers a way to package students' answers to an application and send it to another provider, who can then make use of the data without any knowledge of the application from which it was created. The Scholar Snapp solution is an XML schema definition (XSD) file that defines a standard structure for a student's answers to a scholarship application. This opens many useful, automatic document-processing scenarios as a result. Additionally, The Scholar Snapp solution is:

- A way to package student information for sharing automatically
- A way to let students take charge of their information and share it at their discretion
- A way to get more from your investment in your online scholarship application

Figure 1 shows how a student could benefit from Scholar Snapp.

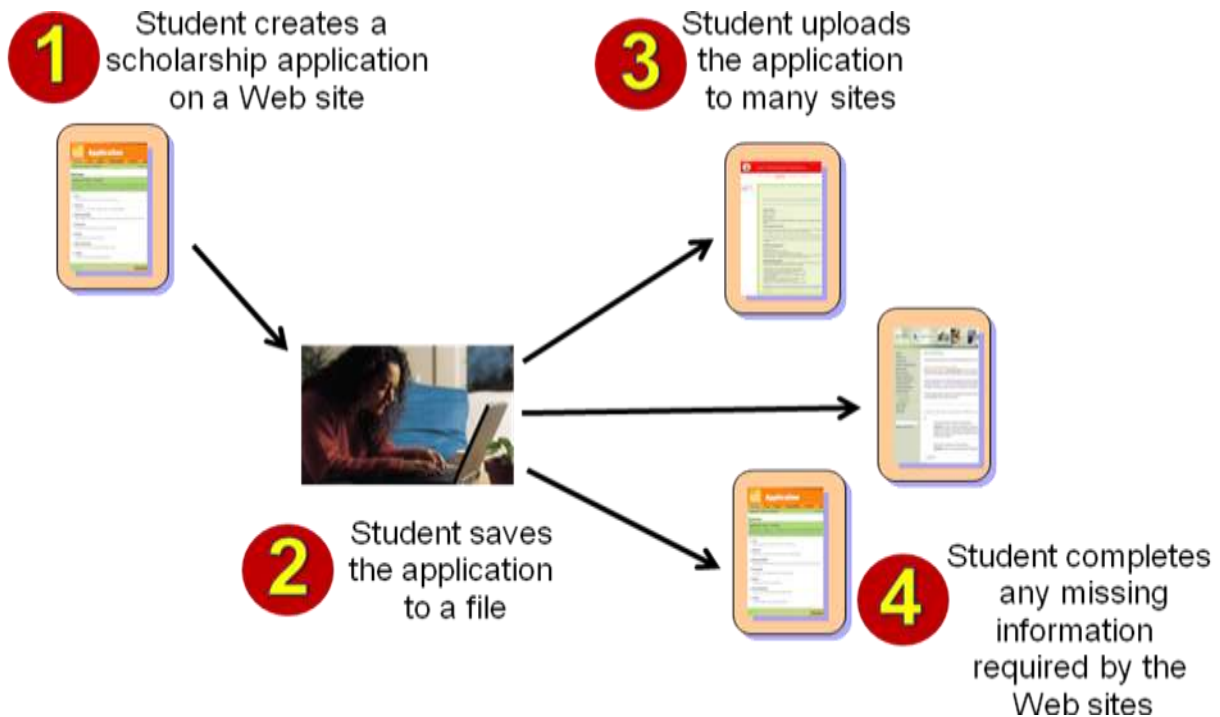


Figure 1

How the Scholar Snapp™ solution was created

The Scholar Snapp™ solution was developed with a grant from the Michael & Susan Dell Foundation. A workgroup was formed under the National Scholarship Providers Association (NSPA) with members carefully selected to represent many perspectives from the scholarship industry. Workgroup members at the time the Scholar Snapp solution was released were:

- Boettcher Foundation
- Canada Millennium Scholarship
- Coca-Cola Scholars Foundation
- FastWeb
- George Snow Scholarship Fund
- Hispanic Scholarship Fund
- Michael & Susan Dell Foundation
- Minnesota Community Foundation/The St. Paul Foundation
- National Scholarship Providers Association
- Oregon Student Assistance Commission
- Scholarship America
- Senator George J. Mitchell Scholarship
- University of Texas at Austin, Student Financial Services

Student information carried in SDS files

The Scholar Snapp solution requires that student information be packaged into a computer data file with an .SDS extension. Each SDS file represents one student. There is a specific process by which a file becomes a valid SDS file:

1. The student's answers are saved to an XML file that conforms to the Scholar Snapp schema
2. The file is shrunk with [gzip](#) compression
3. The file is re-named with the file extension 'SDS'

Gzip is an acronym for GNU Zip, which was chosen as the required compression algorithm because it is a broadly used compression format that is available in many programming platforms such as Microsoft .NET, Java, and ColdFusion. This SDS file compression is important because the Scholar Snapp schema supports a wide array of student file attachments that could become large. It is important to note that the Scholar Snapp solution does not define the business processes or communication mechanisms for transmitting and receiving SDS files, only the structure and contents of SDS files.

The Scholar Snapp™ Schema

The Scholar Snapp™ solution was released as a standard in the form of an XML schema (ScholarshipDataStandardv1.xsd) so that any organization can automatically create or read scholarship applications using software that conforms to the Scholar Snapp solution file format. The Scholar Snapp schema defines all the possible data elements that can exist in an SDS file. The schema differs from SDS files themselves. While the schema defines all possible data elements, an SDS file is a specific instance of a student's information that contains the minimum required information and possibly additional, optional information as dictated by the schema.

Schema Contents

The Scholar Snapp™ solution is comprehensive and provides maximum flexibility with regard to how much or how little information can be contained within an SDS file, governed by the Scholar Snapp schema. The Scholar Snapp schema is named ScholarshipDataStandardv1.xsd and is available at the Scholar Snapp website at www.scholarsnapp.org. The schema defines the following high top-level fields with corresponding detailed fields. For more detailed information, it is recommended that you review the schema itself available in Appendix A.

Activity	ContactInfo	GovernmentPrograms	Reference
ActivityTypeList	ContactMethodsList	GPA	RelationshipList
Address	Context	GPAInfo	Relative
AddressTypesList	CultureCodeList	GradingScaleList	ScholarshipApplication
ApplicationStatusList	CurrencyList	GraduationInfo	ScholarshipPurpose
AttachedFile	DependencyStatusList	HomeEnvironment	ScholarshipPurposeList
AttachedFileTypeList	EducationLevel	IncomeInfo	School
BirthPlace	EducationLevelsList	IntendedMajor	SchoolProgramTypesList
Citizenship	Enrollment	LivingWithList	SchoolTypesList
CitizenshipStatusList	Essay	LongString	SchoolYearsList
Class	EssayTypes	MaritalStatusList	ScoreTypes
ClassList	EthnicitiesList	MilitaryBranchList	ShortString
ClassRank	Ethnicity	Name	Spouse
ClassStatusList	FAFSAStatusList	OrientationList	Student
ClassTypeList	FamilyTitlesList	Race	TestScore
CollegeChoice	Gender	RacesList	TestsList
CollegeReadinessProgram	GenderList	RankMethodsList	
CollegeReadinessProgramList	GovernmentAid	RankSystemsList	

Minimum required information in an SDS file

While most of the data elements in an SDS file are optional, the Scholar Snapp™ schema defines a minimum amount of information that must be present in an SDS file to be considered as valid. The smallest valid SDS file must contain the following as part of the ScholarshipApplication complex type:

- A Context element that contains at least a Created, CreatedBy, and ApplicationLanguage element.
- A Student element that contains at least a Name and ContactInfo element. The Name element must contain at least a FirstName and LastName element. The ContactInfo element must contain at least an Email element.

These minimum requirements make sense when you consider the smallest SDS file that you could receive from someone, yet be useful. This also helps tell the difference between two students' SDS files with the same first and last name when you have a collection of small or incomplete SDS files. Additionally, the required Email address ensures that you have at least one communication method with any student, however many more contact methods are supported in the Scholar Snapp schema that are optional.

Considerations for various organization types

The Scholar Snapp™ solution is applicable to many organizations that server various scholarship industry functions.

Scholarship providers with a single scholarship application

Let us take for example a scholarship provider with the following qualities. The organization:

- Has a single scholarship application
- Sources scholarship applications without external help
- Does not plan to forward applications to other organizations
- May be a public or private non-profit, a university, or association

Scholarship application sourcing

For this example organization, the Scholar Snapp™ solution provides new options for receiving scholarship applications:

- From a scholar through the file upload process
- From a related organization by the application forwarding process
- From a clearinghouse by polling and searching the database

Considerations for application internal processing

There are several considerations for the example organization when streamlining their internal application-processing pipeline.

- Processing application data for database storage or deletion
- Detecting and filtering out duplicate applications or allowing scholar to view, modify, or choose which application is authoritative
- Automatically or manually completing missing information or update existing information either immediately or over a period of research time
- Filtering eligibility and verifying data provided when applicable
- If a scholar matches certain criteria, based on agreements with other organizations, ask the scholar for permission to share the application with those organizations from a blind or targeted filter list. It is assumed that the example organization would inform the scholar that further communication regarding forwarded applications will come from the target organization.
- Automated responses to the scholar based on certain events, e.g. application received, missing information, accepted to semifinalist stage, etc.

Considerations for application external sharing

There are several considerations for the example organization when sharing student information.

- Define agreements with other organizations to share applications that dictate data sharing terms, privacy policies, and any other security policies/practices for data use. This includes email opt-in processes. Opt-out lists to let a student revoke sharing permissions. After being revoked, from that point forward the organization would no longer send copies of the application anywhere.
- Build a mechanism to determine with whom the student gave permission to share their information and to track the source of the application which affects its criteria for sharing.
- Determine any security policies such as file locations, data transfer processes, usernames/passwords/ encryption mechanisms, frequency of data upload/download, etc.
- Use the CreatedBy and UpdatedBy elements provided in the Scholar Snapp™ solution to determine whether the application has already been forwarded to avoid endless forwarding chains.

Scholarship providers with multiple scholarship applications

Let us take for example a scholarship provider with the following qualities. The organization:

- Has multiple scholarship applications
- Sources scholarship applications without external help
- Does not plan to forward applications to other organizations
- May be a public or private non-profit, a university, or association

Considerations

Given the considerations for the previous example organization that has a single application, the same considerations apply, but additionally:

- Provide screens to allow the student to select one or more scholarship programs for which they are applying. This information is needed to determine the eligibility or other programs that the organization might recommend to them.
- Segment data for matching scholar applications to the program(s) for which they are intended.
- Provide opportunity to “opt in” to other programs (meaning those not already suggested or offered to the scholar) for which they might be eligible.
- Track data about the origin of applications segmented by donor, program, etc. to understand the impact of the Scholar Snapp™ solution.
- As a scholar progresses through the pipelines of multiple programs, maintain status and communication threads with the scholar for each program independently. Drive the communication mechanism based on the scholar’s indicated “preferred communication method” field in the Scholar Snapp solution.
- When dealing with multiple scholarship programs and applications, use the most detailed application as the basis for creating a file that the student saves to their local computer for use elsewhere.

Scholarship management organizations with a single application

Let us take for example a scholarship provider with the following qualities. The organization:

- Has a one or more scholarship application
- Has many donors or clients with differing needs
- Provides scholarship administration services from end to end
- May be a community foundation or service provider

Considerations

For this example organization, the only differences between it and the previous example of an organization with multiple applications would be driven by the differing needs of the donors/clients.

- Segment data reporting and application tracking/communication for the different donor/clients
- Differing the process steps based on the varying depth of the services provided
- Accommodate processes that have external phases completed by other organizations, for example when scholar selection is conducted by an external board, but afterwards the scholarship administrative services continue through the remaining process steps.

Scholarship listing, matching, or referral services

Let us take for example a scholarship provider with the following qualities. The organization:

- Hosts many scholarship applications or simply refers applicants to other locations
- Has many clients with differing needs

Considerations

For this example organization, the only differences between it and the previous example of an organization with multiple applications would be driven by the differing nature of the engagement with partners.

- Seize the opportunity to host the students application file and automatically forward to target organizations with scholar simply providing permission
- Provide the ability to “one click apply” to many different organizations and manage the collection of data for target organizations
- Provide statistics to clients about the demographics of their applicants

How to adopt Scholar Snapp™

This guide assumes that you meet the prerequisites for adopting Scholar Snapp™.

Prerequisites

To adopt Scholar Snapp™, it is assumed that you have:

- At least one online scholarship application
- A copy of the Scholar Snapp schema and the accompanying documentation
- Access to technical staff skilled in Web, XML, and database programming (either on staff or vendor/contractor)
- A desire to help scholars access more funds for college beyond your own scholarship

Process Overview

For most organizations, adopting the Scholar Snapp solution involves the following high-level tasks.

1. Creating a list of all of your databases and database fields that contains all the data that you store from your scholarship applications.
2. Create a mapping of your database fields list to the related Scholar Snapp fields for those values.
3. Create a software module that will extract students' answers to a scholarship application and write those in the schema compliant Scholar Snapp solution file. This module would also permit the student to save the file to their local computer.
4. Create a software module that will allow a student to upload an existing scholarship application file and to populate your application form with the values that are available in the file.
5. Implement the processes necessary to share the scholarship applications with other organizations if applicable, such as file transfer mechanisms, authentication, encryption, etc.

Mapping your database to Scholar Snapp™

Once you create a list of all your database fields for scholarship applications, the next step is to identify the corresponding Scholar Snapp™ fields and to understand what data transformations may be necessary to fill those fields. While there is a minimum amount of information that you must put into an SDS file, it is a very small percentage of all the available fields. You should focus on just populating the minimum fields plus those for which you have the information. It is possible that you will add some information to an SDS file that does not apply to other scholarship applications. In that case, the recipient of your SDS file can choose to store the information that they want without the obligation to store it all. The lack of heavy restrictions, combined with choice and flexibility are a few reasons that the Scholar Snapp™ solution is so powerful.

Figure 2 shows the type of data transformations that you may need to perform to write your scholar data into the Scholar Snapp file.

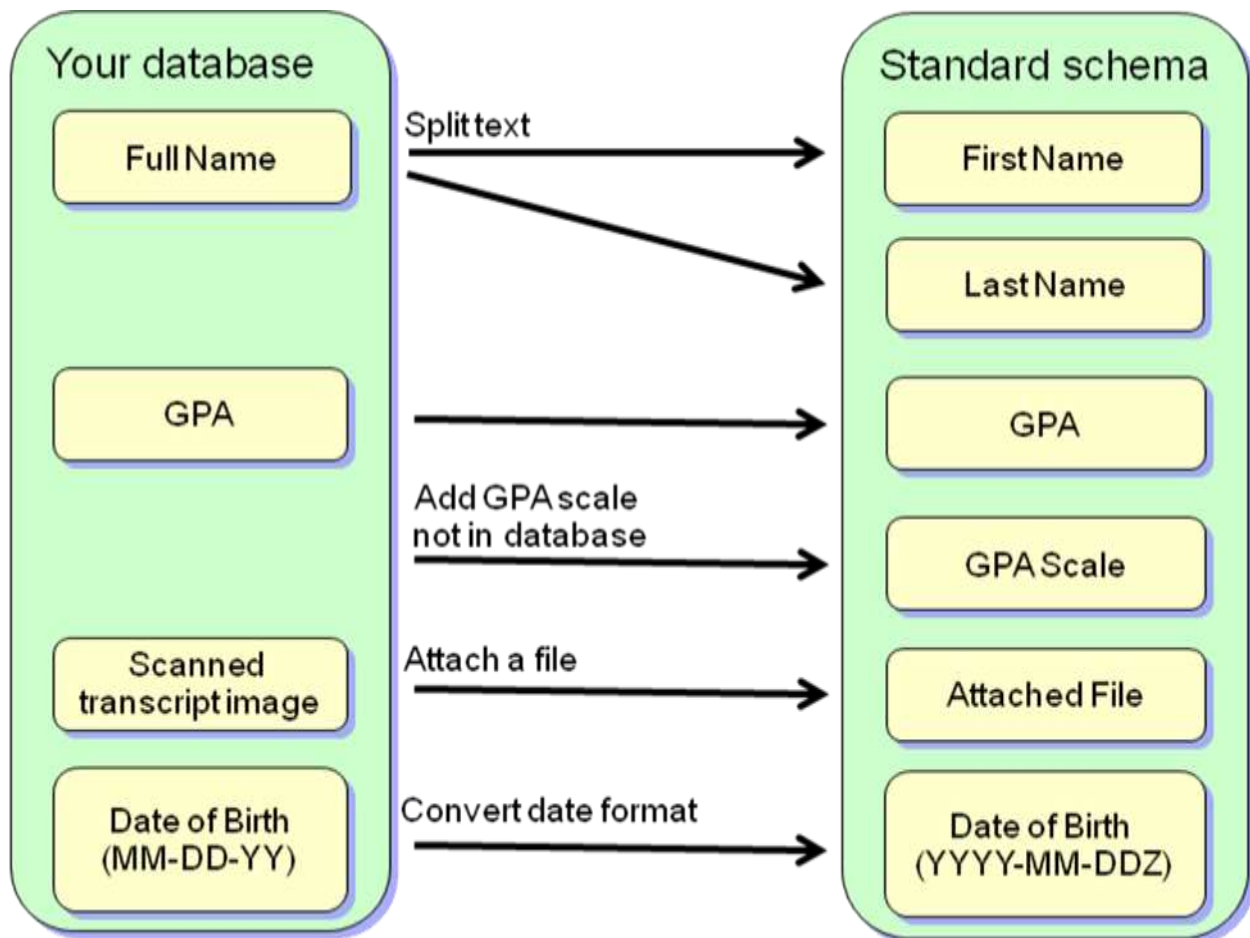


Figure 2

Each arrow in the diagram represents a data transformation of some type, even in the case where the data is simply copied as-is. These arrows also represent potential code that you need to write to transform data. Some transformations are simple, such as splitting a full name into a first and last name. Others are more sophisticated, such as retrieving data from multiple databases, performing some calculation, then converting the information to another data type. Additionally, your database may contain ambiguous information, such as a GPA with no corresponding information about what scale on which the GPA score is based (4.0 scale, 5.0 scale, etc.). In those cases, you must determine what data you have and then provide additional information to disambiguate it for the Scholar Snapp™ file, which accepts very little ambiguity for good reason. Any ambiguity represents problems for your communication partner to qualify the information and to infer what your data means. Those who are creating a new database for a new online application would do well to mirror the Scholar Snapp structure and types for easier adoption. However, many organizations have existing databases with pre-defined schemas, thus the need for data mapping and transformation exists.

Software architecture for supporting the Scholar Snapp™ solution

There are many ways to implement the necessary code to transform your data. A primary focus should be to re-use as much of your existing software as possible to reduce labor and cost. Some database vendors provide excellent tools to do some automatic data conversion. Aside from available tools, you may need custom code to do the transformation. There are two major architectural pieces to any custom code you may write:

1. Converting data from the database to the Scholar Snapp™ file.
2. Reading a standard file and populating a Web form.

Changes are common for online scholarship applications as organizations respond to feedback and attempt to improve the Web site. This may lead to changes to the database structure and thus any custom code written for specific data transformations may be affected. As a result, it is recommended to use a data transformation engine to which you can provide a data-mapping file. This simply means that whenever changes are needed to the online application and database, that the mapping file can change and not the core code. The mapping file would typically contain information such as:

- Data source name
- Table
- Column
- Destination XML element
- Destination data type
- Reference to any necessary data retrieval or transformation code

An additional aspect of adopting the Scholar Snapp solution is to pre-populate your application with information provided in a file from the student. There is a need for a second mapping engine and mapping file to convert the XML elements to corresponding Web controls. The mapping file would typically contain information such as:

- Source XML element
- Destination Web page
- Destination Web control name
- Destination data type
- Reference to any necessary data retrieval or transformation code

An approach such as this can help you keep your current Web site data input validation code and enter student data into your form in the same way as a student would manually type it. With such an investment in architecture, your solution can accommodate changes to the application and database with minimal effort.

Controlling access to private student data

Your scope of private student data control is limited by your own system and ends whenever you send it to anyone else. Scholar Snapp™ doesn't create any new privacy requirements and you should still use the same due diligence that you use with your current online or paper applications. As you currently do, you must get student permission to share their information with others, ensure that your own data storage is fully secure, and that you can trust any downstream partners and vendors with the data.

Successful implementation of the Scholar Snapp solution requires careful data security practices. You are responsible for all aspects of your data security, including any private student data that you handle. In general, there are six major security attacks to be concerned with and that you should have plans to mitigate (STRIDE):

- Spoofing
- Tampering
- Repudiation
- Information disclosure
- Denial of service
- Elevation of privilege

Based on this list, it is not only important to control access to data, but to ensure that someone is who they say they are, that they have access to only the data they need, that their access is logged, that repeated failed login attempts are recorded and alerts sent, etc. Security is not just about implementing password protection, but encompasses a wide array of activities and concerns. Data security is an industry in itself and a heavily detailed discussion is beyond the scope of this document.

- For more information about these attacks and useful links to information that may help you mitigate them, see [http://en.wikipedia.org/wiki/STRIDE_\(security\)](http://en.wikipedia.org/wiki/STRIDE_(security))
- While it is outside the scope of this standard, there is a standard named the Extensible Access Control Markup Language (XACML) that expresses authorization policies in XML that may be useful to implementers. For more information, see <http://xml.coverpages.org/xacml.html>

Methods to receive applications automatically

Since the Scholar Snapp™ solution does not dictate the communication mechanism, only the format of the messages, you may choose a manual or automatic mechanism to share scholar data. When receiving an application that you want to automatically route to another organization, you must have set up the communication channel in advance. Several options include:

- A common file drop-point under control of the target organization (it is important that visitors can only write files at this location, not read or view any files for security/privacy purposes)
- A secure Web service connection point-to-point between two computers (data encryption is important here)
- Sent via email where the attached file is the student's scholarship application in the scholar data standard format (beware the security implications of how easily emails can be forwarded, read, etc.)
- A location on your Web site where the scholar can upload the file themselves (ZAP my app)

All options carry pros, cons, and security implications. It is important to balance complexity, ease of use, and security to be most effective. Always consider the possibility that someone wants to steal the student's information and you must make plans to safeguard the data exchange and storage points where this could occur.

Filtering for Eligibility

With the Scholar Snapp™ solution makes it possible to receive more scholarship applications than before. The Scholar Snapp solution simply makes it easier to share data and organizations may share the data more readily as a result. With the increased load, you may wish to implement a way to filter out applications that do not meet your program's basic requirements. This would be done in an effort to increase the quality of applications that you process and store, and to avoid heavily processing applications that don't match your program.

Some organizations implement an eligibility quiz as part of their application, requiring that students answer a small set of basic questions up front and only allow those that meet eligibility to move along to a full application. This technique also works for electronic applications, but can occur automatically. Upon receiving an electronic application, you implement a script or program that reads only a handful of basic fields to determine eligibility. If the application passes the check, the script moves the application

to permanent storage location to be further processed. Otherwise, the script deletes the application. This saves the time and space necessary to process applications.

- XML document validation and processing is within the domain of entry-level programming skills and expensive IT consultants may not be necessary. For techniques to do this, see the following links.

<http://www.codetoad.com/xml/>

http://www.hotscripts.com/XML/Scripts_and_Programs/Parsers/index.html

<http://www.developertutorials.com/scripts/65/105/>

Student Custom Identifiers

This identifier option is provided in the schema for organizations to implement a different student identifier system than that provided by social security numbers, taxpayer IDs, etc. In fact, the Scholar Snapp solution does not provide a place for the storing of social security numbers for security. It is useful to do this when you choose not to collect social security or taxpayer ID numbers, yet you need a mechanism to determine the difference between two potential John Smiths for example. To take advantage of this, your system needs to generate a unique ID for students and the solution can accommodate this ID through the StudentCustomId element. It is the responsibility of the host to generate these numbers and to ensure they are unique within their database because the solution simply allows a place for them to reside when transmitting and receiving the student data. It is common practice to use the Universal Unique Identifier (UUID) standard (known as GUIDs in the Microsoft implementation) to generate identifiers that are 128bits long. While each key cannot be guaranteed to be unique universally, the number of keys available using that standard are 3.4×10^{38} . Thus, the odds of accidentally generating a duplicate key are so incredibly small that the Scholar Snapp™ solution is generally considered good enough to use for this purpose.

Acknowledging Received Scholarship Applications

It is important to send at minimum an email notification in response to any scholarship application received automatically via the scholar data standard. There are many reasons for doing this, but primarily it involves:

- Let the student know what organizations have received their application so far
- Ensuring that the student meant for you to receive the information and giving them the option to delete the application if not
- Prevent redundant work on the student's part and redundant data processing on yours in the event that the student accidentally applies to your program more than once

Frequently Asked Questions

This section lists common questions posed by the community and provides the answers to the questions.

For IT professionals, please explain the Scholar Snapp™ solution in technical terms.

The Scholar Snapp solution is an XML Schema definition (XSD) that defines the contents of scholarship application documents that are stored as XML files that have been shrunk with gzip compression and re-named with the SDS file extension.

Are there minimum requirements needed to implement the Scholar Snapp™?

There are two basic requirements.

1. You must be able to receive a scholarship application file and fill out your application with the information in the file.
2. You must be able to save a student's application to a file that conforms to Scholar Snapp.

To provide this application upload capability to students that use my online application, do I have to change the look and feel of my site?

No, your existing online application does not change in appearance, with the exception of a screen that asks the scholar where to save or retrieve their application file. Most changes are invisible to the scholar and represent work done “under the hood” to make it happen.

Do you improve the Scholar Snapp™ solution as new needs arise? How do I keep up with any changes?

The Scholar Snapp™ solution may be updated to address new needs, but any changes will be released under a new version number to identify it clearly versus the original version. Thanks to the way XML works, a computer can receive an application based on the original or new version and tell them apart. As a result, a new standard coming into existence does not automatically invalidate your investment to implement the original standard on your Web site. You will always have a choice whether to make any changes on your Web site to accommodate applications based on the new standard or to continue to accept applications that are based on the original version of The Scholar Snapp solution only.

Does the Scholar Snapp™ solution require me to have the same application questions or the same question text as other organizations? Is this a common application?

No, the Scholar Snapp™ solution defines answers to application questions, but does not define the questions or the order in which they appear. For example, regardless of how, where, and when you ask the question, “What community service work have you performed?” there is a “community service essay” field in the Scholar Snapp solution to accommodate it. Therefore, organizations can use custom question text presented in any order they wish on their own applications. Then, another organization

can look in the scholar file for the “community service essay” field to extract the information and use it to automatically fill out that part of their application.

How did you choose the contents of the data package for Scholar Snapp™? I would guess that it cannot hold every answer ever given on any application.

The Scholar Snapp™ solution was designed using a similar workgroup process as used to define many Internet standards. In our case, we designed it with the input of many scholarship organizations and those that represent various aspects of the scholarship world, such as community foundations, corporate scholarships, universities, private foundations, scholarship service providers, and scholarship directories. The Scholar Snapp solution was designed to accommodate the most common data fields in use today and should work well for the majority of organizations. It is expected that your system will prompt the student to provide any missing information for fields that are not represented in Scholar Snapp.

The Scholar Snapp™ solution accommodates the answers to hundreds of common scholarship questions. Do I have to add all of those questions to my online application?

No, the Scholar Snapp™ solution does not dictate the contents of any one application. Instead, it accommodates the wide array of questions that many scholarship providers ask students and provides a carrier/transmission format for them.

How do I keep a scholars information from falling into the wrong hands?

Even though the Scholar Snapp™ solution gives you the means to share information much more easily than ever before, you are still in complete control about where you send the data. As with any sensitive information, do not share the information with any organization that you do not trust. It is expected that most information sharing will occur between scholarship providers and/or scholarship matching services.

Can I employ a mix of paper-based and online processes?

Yes. Many aspects of the scholarship application process can be automated while using paper in various stages. For example, you may collect electronic applications, but manually verify faxed/physical transcripts. You may use an automatic quantitative algorithm to select semifinalists, yet print out your scholarship applications for qualitative reading and scoring and then conduct data entry back into a system. While many paper-based processes are not very efficient, most organizations, even those with advanced online/electronic workflows continue to use paper for some parts of their process. Additionally, upgrading to new electronic processes is best done in a layered/phased fashion in an effort to manage change over time, which implies a mixed solution at many points along the way.

Where can I as a scholarship provider get information or assistance to build the necessary software module for my Web site?

There are many ways to implement the Scholar Snapp™ solution depending upon your needs and the technical expertise available to you. Generally, an important early decision is whether you plan to write the software modules yourself or hire a vendor to do this for you. Implementing the Scholar Snapp solution requires a person with programming skills in XML, Web technologies, such as JavaScript, PHP,

and/or ASP.NET, and database skills such as SQL/Oracle, depending upon those technologies used in your Web site. For more information about Scholar Snapp, including a list of organizations that can assist others with planning and implementing Scholar Snapp, please refer to the NSPA Web site <http://scholarshipproviders.org>

How do I send scholarship applications to other organizations? Do I use email attachments, HTTP, FTP, etc?

All of those transmission methods are available to you. The Scholar Snapp™ solution dictates only the structure of the data files, not the way that you send and receive it. You should ensure that the mechanism that you choose meets your data security requirements.

To share scholarship applications with other organizations, how would I know what questions they ask to make sure they get the correct information?

You would not have to determine that at all. After getting student permission, you simply put whatever student data you currently have into the data package and send it to another organization. At the destination, they would automatically read the file and determine what data is available and what is not, depending on the needs of their organization. The target organization would gather the data from the student for anything that is missing.

Can the Scholar Snapp™ solution accommodate highly complex application questions?

The Scholar Snapp™ solution can accommodate complex questions. For example, your application may ask a very specific question such as “Are you a disabled female student who plays both varsity hockey and the cello, and whose father is a full time employee of the X Corporation?” While there is no simple yes/no field available in the Scholar Snapp solution for such a complex question, you can infer the answer by reading a few fields as a group, such as the disability, gender, extracurricular, and parent employment fields.

What type of questions can't the Scholar Snapp™ solution accommodate?

The Scholar Snapp™ solution doesn't accommodate highly situational or specific questions that fall outside the traditional scholarship application questions. For example, some scholarship applications serve double-duty as applications for other things (such as a college application). As a result, such applications contain questions that are not expected on scholarship applications. A few examples of types of questions we encountered in the Scholar Snapp workgroup and were determined to be out of scope for the Scholar Snapp solution are:

“Why have you chosen to apply to our university?” – This question is specific to the application of a school, not applying for a scholarship.

“Did your parents live in X county anytime between 1980 and 2004?” – This question requires significant data collection about a student's parents beyond a typical scholarship application.

How can I tell if the data entered into an application is correct?

The Scholar Snapp™ solution allows you to detect automatically whether answers are in the proper format. For example, the Scholar Snapp solution allows you to detect if there is incorrect text data in a numeric field, like if a student enters "xyzyz" into the field asking how many years they have lived in their state. However, the Scholar Snapp™ solution does not protect against dishonest answers or anything that requires manual verification such as whether or not the student actually lives at the address they claim.

What does the Scholar Snapp™ solution do for people that use paper applications?

Organizations that use paper today, but have plans to upgrade to electronic applications will benefit in several ways by having a data standard available. For example, the Scholar Snapp™ solution provides a clear migration target, reduces ambiguity about data formats when planning the upgrade process, and is especially helpful when working with vendors to narrow the scope of work. However, you need to process most of your scholarship applications electronically to realize all the Scholar Snapp solution benefits.

Can the Scholar Snapp™ solution handle scholarship application file attachments, such as scanned High School transcript images, Student Aid Reports, etc.?

Yes, the Scholar Snapp™ solution allows for unlimited file attachments in a scholarship application file in the same way that you attach files to an email (in MIME/Base64 encoded format).

Glossary

This section defines common terms used when discussing the Scholar Snapp™ solution.

Clearinghouse - A central repository for scholarship applications from which scholarship providers can locate eligible scholars and from which statistics can be gathered

Online application - A Web site that lets students apply for a scholarship electronically

Schema - A defined structure for a set of information

Paper-based process – A scholarship application process where the student fills out a paper for and then mails it to a scholarship provider

Appendix A: the Scholar SnappSchema

The following text represents the contents of the ScholarshipDataStandardv1.xsd file. For your convenience, the actual XSD file is available for download on the NSPA Web site at

<http://scholarshipproviders.org>

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns="http://scholarshipproviders.org/scholarshipapplication1.0"
xmlns:sds="http://scholarshipproviders.org/scholarshipapplication"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://scholarshipproviders.org/scholarshipapplication"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="ScholarshipApplication"
type="sds:ScholarshipApplication">
    <xs:annotation>
      <xs:documentation>A scholarship application from one
student</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="ScholarshipApplication">
    <xs:annotation>
      <xs:documentation>The root element of a scholarship
application</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="Context" type="sds:Context">
        <xs:annotation>
          <xs:documentation>The scholarship application</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="AttachedFiles" type="sds:AttachedFile" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>MIME encoded file attachments such as
transcripts, SARs, student pictures, etc.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Student" type="sds:Student">
        <xs:annotation>
          <xs:documentation>The student and represents the majority of
information in the scholarship application</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="Context">
    <xs:annotation>
      <xs:documentation>Information about the scholarship application itself,
rather than about the student</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="Created" type="xs:dateTime">
        <xs:annotation>
```

```

    <xs:documentation>Date that the scholarship application file was
originally created. Once provided, this field should be considered read-only
and further updates made through the 'LastUpdated' field.</xs:documentation>
  </xs:annotation>
</xs:element>
  <xs:element name="CreatedBy" type="sds:ShortString">
    <xs:annotation>
      <xs:documentation>Organization or Web site that originally created
the application. Once provided, this field should be considered read-only and
further updates made through the 'LastUpdatedBy' field.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="LastUpdated" type="xs:dateTime" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Date that the scholarship application file was
last updated</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="LastUpdatedBy" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Organization or Web site that last updated the
application</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="ApplicationLanguage" type="sds:CultureCodeList">
    <xs:annotation>
      <xs:documentation>Language in which the scholarship application
file was created</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="PermissionGivenDate" type="xs:dateTime"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Date that student gave permission to forward
their application to other organizations.</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="AttachedFile">
  <xs:annotation>
    <xs:documentation>Type that represents a file attached to a scholarship
application, such as a SAR, transcript, student's photo,
etc.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="FileName" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>Name of the attached file</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="FileNameExtension" type="sds:ShortString">
      <xs:annotation>

```

```

    <xs:documentation>Filename extension that is used to determine the
program needed to read the file, for example TXT for raw text, DOC for Word
document, BMP for a bitmap image, etc.</xs:documentation>
  </xs:annotation>
</xs:element>
  <xs:element name="FileDescription" type="sds:AttachedFileTypeList"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Content of the attached file, whether a
transcript, a picture, etc.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="FileOtherDesc" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Free-form description of the attached file, for
example 'Photo of Joe Miller'</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="DateAttached" type="xs:dateTime" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Date that the file was attached to the
application</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="File" type="xs:base64Binary">
    <xs:annotation>
      <xs:documentation>The attached file itself in MIME (base64) encoded
format</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="AttachedFileTypeList">
  <xs:annotation>
    <xs:documentation>Content of the attached file, whether a transcript, a
picture, etc.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Student Aid Report (SAR)"/>
    <xs:enumeration value="Transcript"/>
    <xs:enumeration value="FAFSA"/>
    <xs:enumeration value="Picture"/>
    <xs:enumeration value="Recommendation/Reference Letter"/>
    <xs:enumeration value="Portfolio Document"/>
    <xs:enumeration value="Diploma Certificate"/>
    <xs:enumeration value="Degree Certificate"/>
    <xs:enumeration value="Proof of Achievement"/>
    <xs:enumeration value="School Acceptance Letter"/>
    <xs:enumeration value="Medical Document"/>
    <xs:enumeration value="Proof of Employment"/>
    <xs:enumeration value="Video"/>
    <xs:enumeration value="Audio"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>

```

```

<xs:simpleType name="ShortString">
  <xs:annotation>
    <xs:documentation>String used for short text fields</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:maxLength value="100"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="LongString">
  <xs:annotation>
    <xs:documentation>String used for long text fields, such as
essays</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:maxLength value="4000"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="CultureCodeList">
  <xs:annotation>
    <xs:documentation>The list of culture codes per RFC
2119</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:language">
    <xs:enumeration value="en-US"/>
    <xs:enumeration value="en-CA"/>
    <xs:enumeration value="af-ZA"/>
    <xs:enumeration value="sq-AL"/>
    <xs:enumeration value="ar-DZ"/>
    <xs:enumeration value="ar-BH"/>
    <xs:enumeration value="ar-EG"/>
    <xs:enumeration value="ar-IQ"/>
    <xs:enumeration value="ar-JO"/>
    <xs:enumeration value="ar-KW"/>
    <xs:enumeration value="ar-LB"/>
    <xs:enumeration value="ar-LY"/>
    <xs:enumeration value="ar-MA"/>
    <xs:enumeration value="ar-OM"/>
    <xs:enumeration value="ar-QA"/>
    <xs:enumeration value="ar-SA"/>
    <xs:enumeration value="ar-SY"/>
    <xs:enumeration value="ar-TN"/>
    <xs:enumeration value="ar-AE"/>
    <xs:enumeration value="ar-YE"/>
    <xs:enumeration value="hy-AM"/>
    <xs:enumeration value="Cy-az-AZ"/>
    <xs:enumeration value="Lt-az-AZ"/>
    <xs:enumeration value="eu-ES"/>
    <xs:enumeration value="be-BY"/>
    <xs:enumeration value="bg-BG"/>
    <xs:enumeration value="ca-ES"/>
    <xs:enumeration value="zh-CN"/>
    <xs:enumeration value="zh-HK"/>
    <xs:enumeration value="zh-MO"/>
    <xs:enumeration value="zh-SG"/>
    <xs:enumeration value="zh-TW"/>
  </xs:restriction>
</xs:simpleType>

```

```
<xs:enumeration value="zh-CHS"/>
<xs:enumeration value="zh-CHT"/>
<xs:enumeration value="hr-HR"/>
<xs:enumeration value="cs-CZ"/>
<xs:enumeration value="da-DK"/>
<xs:enumeration value="nl-BE"/>
<xs:enumeration value="nl-NL"/>
<xs:enumeration value="en-AU"/>
<xs:enumeration value="en-BZ"/>
<xs:enumeration value="en-CB"/>
<xs:enumeration value="en-IE"/>
<xs:enumeration value="en-JM"/>
<xs:enumeration value="en-NZ"/>
<xs:enumeration value="en-PH"/>
<xs:enumeration value="en-ZA"/>
<xs:enumeration value="en-TT"/>
<xs:enumeration value="en-GB"/>
<xs:enumeration value="en-ZW"/>
<xs:enumeration value="et-EE"/>
<xs:enumeration value="fo-FO"/>
<xs:enumeration value="fa-IR"/>
<xs:enumeration value="fi-FI"/>
<xs:enumeration value="fr-BE"/>
<xs:enumeration value="fr-CA"/>
<xs:enumeration value="fr-FR"/>
<xs:enumeration value="fr-LU"/>
<xs:enumeration value="fr-MC"/>
<xs:enumeration value="fr-CH"/>
<xs:enumeration value="gl-ES"/>
<xs:enumeration value="ka-GE"/>
<xs:enumeration value="de-AT"/>
<xs:enumeration value="de-DE"/>
<xs:enumeration value="de-LI"/>
<xs:enumeration value="de-LU"/>
<xs:enumeration value="de-CH"/>
<xs:enumeration value="el-GR"/>
<xs:enumeration value="gu-IN"/>
<xs:enumeration value="he-IL"/>
<xs:enumeration value="hi-IN"/>
<xs:enumeration value="hu-HU"/>
<xs:enumeration value="is-IS"/>
<xs:enumeration value="id-ID"/>
<xs:enumeration value="it-IT"/>
<xs:enumeration value="it-CH"/>
<xs:enumeration value="ja-JP"/>
<xs:enumeration value="kn-IN"/>
<xs:enumeration value="kk-KZ"/>
<xs:enumeration value="ko-KR"/>
<xs:enumeration value="ky-KZ"/>
<xs:enumeration value="lv-LV"/>
<xs:enumeration value="lt-LT"/>
<xs:enumeration value="mk-MK"/>
<xs:enumeration value="ms-BN"/>
<xs:enumeration value="ms-MY"/>
<xs:enumeration value="mr-IN"/>
```

```

<xs:enumeration value="mn-MN"/>
<xs:enumeration value="nb-NO"/>
<xs:enumeration value="nn-NO"/>
<xs:enumeration value="pl-PL"/>
<xs:enumeration value="pt-BR"/>
<xs:enumeration value="pt-PT"/>
<xs:enumeration value="pa-IN"/>
<xs:enumeration value="ro-RO"/>
<xs:enumeration value="ru-RU"/>
<xs:enumeration value="sa-IN"/>
<xs:enumeration value="Cy-sr-SP"/>
<xs:enumeration value="Lt-sr-SP"/>
<xs:enumeration value="sk-SK"/>
<xs:enumeration value="sl-SI"/>
<xs:enumeration value="es-AR"/>
<xs:enumeration value="es-BO"/>
<xs:enumeration value="es-CL"/>
<xs:enumeration value="es-CO"/>
<xs:enumeration value="es-CR"/>
<xs:enumeration value="es-DO"/>
<xs:enumeration value="es-EC"/>
<xs:enumeration value="es-SV"/>
<xs:enumeration value="es-GT"/>
<xs:enumeration value="es-HN"/>
<xs:enumeration value="es-MX"/>
<xs:enumeration value="es-NI"/>
<xs:enumeration value="es-PA"/>
<xs:enumeration value="es-PY"/>
<xs:enumeration value="es-PE"/>
<xs:enumeration value="es-PR"/>
<xs:enumeration value="es-ES"/>
<xs:enumeration value="es-UY"/>
<xs:enumeration value="es-VE"/>
<xs:enumeration value="sw-KE"/>
<xs:enumeration value="sv-FI"/>
<xs:enumeration value="sv-SE"/>
<xs:enumeration value="ta-IN"/>
<xs:enumeration value="tt-RU"/>
<xs:enumeration value="te-IN"/>
<xs:enumeration value="th-TH"/>
<xs:enumeration value="tr-TR"/>
<xs:enumeration value="uk-UA"/>
<xs:enumeration value="ur-PK"/>
<xs:enumeration value="Cy-uz-UZ"/>
<xs:enumeration value="Lt-uz-UZ"/>
<xs:enumeration value="vi-VN"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="Student">
  <xs:annotation>
    <xs:documentation>Type that represents all the information about a
student in a scholarship application</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Name" type="sds:Name">

```

```

    <xs:annotation>
      <xs:documentation>Name of the student</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="ContactInfo" type="sds:ContactInfo">
    <xs:annotation>
      <xs:documentation>Methods by which to reach the
student</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="StudentCustomId" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Alternative program-generated student identifier
that is intended for students who may not have or do not provide the typical
identification numbers.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="Addresses" type="sds:Address" minOccurs="0"
maxOccurs="unbounded">
    <xs:annotation>
      <xs:documentation>The student's various
addresses</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="TimeAtCurrentAddress" type="xs:duration"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Length of time the student has lived at their
current address</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="BirthDate" type="xs:date" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Student's date of birth</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="Gender" type="sds:GenderList" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Student's gender</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="GenderDetail" type="sds:GenderDetailList"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Student's detailed gender
information</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="Orientation" type="sds:OrientationList"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>Student's indicated sexual
orientation</xs:documentation>
    </xs:annotation>
  </xs:element>

```

```

</xs:element>
<xs:element name="Race" type="sds:Race" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Student's race</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Ethnicity" type="sds:Ethnicity" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Student's ethnicity</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="BirthPlace" type="sds:BirthPlace" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The location where the person was
born</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Citizenship" type="sds:Citizenship" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The student's citizenship</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Disabled" type="xs:boolean" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Whether the student indicates a
disability</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="DisabilityDesc" type="sds:ShortString" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The description of a student's
disability</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Veteran" type="xs:boolean" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Whether the student is a military
veteran</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="VeteranDesc" type="sds:MilitaryBranchList"
minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>The branches of the military in which the student
served</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="POW" type="xs:boolean" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Whether the student was ever a prisoner of
war</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="POWDesc" type="sds:ShortString" minOccurs="0">
  <xs:annotation>

```

```

        <xs:documentation>War or conflict in which the student was a
prisoner of war</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="ChildofPOWMIA" type="xs:boolean" minOccurs="0">
        <xs:annotation>
            <xs:documentation>Whether the student is a child of a POW or MIA
military service member</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="EducationLevel" type="sds:EducationLevel"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>Student's highest completed level of
school</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="TestScores" type="sds:TestScore" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>The student's test scores</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="GraduationInfo" type="sds:GraduationInfo"
minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Student's graduation and/or GED
information</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="GPA" type="sds:GPA" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Student's grade point averages for various years
of school</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ClassRank" type="sds:ClassRank" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Student's current class rank
indicator</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="CurrentEnrollments" type="sds:Enrollment"
minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Schools in which the student is
enrolled</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="CollegeChoices" type="sds:CollegeChoice"
minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Student's college choices</xs:documentation>

```

```

        </xs:annotation>
    </xs:element>
    <xs:element name="FinalCollegeChoice" type="sds:CollegeChoice"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>Student's indicated top choice assuming that they
receive multiple acceptances</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ScholarshipPurpose" type="sds:ScholarshipPurpose"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The type of school for which the student seeks a
scholarship</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ApplyingForOtherScholarships" type="xs:boolean"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>Whether the student is applying for other
scholarships</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="IntendedMajors" type="sds:Major" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>The student's intended areas of study when in
school</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="CurrentMajors" type="sds:Major" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>The student's current areas of study assuming
they are currently in school</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="IntendedDiscipline" type="sds:ShortString"
minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>The student's intended area of employment after
graduation</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Essays" type="sds:Essay" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>The student's narrative application
essays</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Activities" type="sds:Activity" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>

```

```

        <xs:documentation>The amount of time the student has spent on
certain activities</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="Employed" type="xs:boolean" minOccurs="0">
    <xs:annotation>
        <xs:documentation>Whether the student is
employed</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="Income" type="sds:Money" minOccurs="0">
    <xs:annotation>
        <xs:documentation>The student's financial
information</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="FAFSAParentsCash" type="sds:Money" minOccurs="0">
    <xs:annotation>
        <xs:documentation>
            The student's parents total current balance in cash, savings, and
            checking accounts
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="FAFSAParentsNetWorth" type="sds:Money" minOccurs="0">
    <xs:annotation>
        <xs:documentation>
            The student's parents net worth of their investments, including
            real estate (not their home)
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="FAFSAParentsBusinessInvestments" type="sds:Money"
minOccurs="0">
    <xs:annotation>
        <xs:documentation>
            The student's parents net worth of their current business and/or
            investment
            farms.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="FAFSAStatus" type="sds:FAFSAStatusList"
minOccurs="0">
    <xs:annotation>
        <xs:documentation>Student's current status regarding the filing of
a Free Application for Federal Student Aid (FAFSA) form</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="FAFSAStatusDesc" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
        <xs:documentation>Free form description when the student chose the
'other' option</xs:documentation>
    </xs:annotation>

```

```

</xs:element>
<xs:element name="FAFSAEFC" type="xs:decimal" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The Expected Family Contribution (EFC) from the
student's FAFSA form</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="FAFSAEFCurrency" type="sds:CurrencyList"
minOccurs="0">
  <xs:annotation>
    <xs:documentation>The currency used to report the FAFSA EFC
amount</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="FAFSAFilingYear" type="sds:ShortString"
minOccurs="0">
  <xs:annotation>
    <xs:documentation>The year of the student's FAFSA
filing</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="DisciplineOrLegal" type="xs:boolean" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Whether the student has been subjected to any
school or legal disciplinary action. Use disciplinary essay type for
description.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="MaritalStatus" type="sds:MaritalStatusList"
minOccurs="0">
  <xs:annotation>
    <xs:documentation>The student's marital status</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Spouse" type="sds:Spouse" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The student's spouse, significant other, or
partner</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="DependencyStatus" type="sds:DependencyStatusList"
minOccurs="0">
  <xs:annotation>
    <xs:documentation>Whether the student is a dependent of someone
else or independent</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="DependentsCount" type="xs:int" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The number of dependents under the student's
care, such as the count of the student's children</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="HomeEnvironment" type="sds:HomeEnvironment"
minOccurs="0">

```

```

        <xs:annotation>
          <xs:documentation>The student's home environment and living
arrangements</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="FamilyMembersInCollegeCount" type="xs:int"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The number of student family members that are
attending college, not counting the student</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="SiblingsCount" type="xs:int" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The student's number of
siblings</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Parents" type="sds:Relative" minOccurs="0"
maxOccurs="4">
        <xs:annotation>
          <xs:documentation>The person's parents</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GovernmentAidParticipation" type="sds:GovernmentAid"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>Any government aid that the student
received</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="CollegeReadinessPrograms"
type="sds:CollegeReadinessProgram" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>The student's college readiness program
participation</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="StudentReferences" type="sds:Reference" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>The student's references, such as school
counselors, registrars, and recommenders</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="SimpleTranscript" type="sds:Class" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>The student's transcript in an abbreviated
form</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:complexType name="Major">

```

```

    <xs:annotation>
      <xs:documentation>The student's intended course of
study</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="MajorName" type="sds:ShortString">
        <xs:annotation>
          <xs:documentation>The student's intended area of study when in
school</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="MajorRank" type="xs:int" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The rank of the college major choice per the
student's order of preference, for example first, second, third,
etc.</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="OrientationList">
    <xs:annotation>
      <xs:documentation>Sexual orientation of a student</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="Straight"/>
      <xs:enumeration value="Gay"/>
      <xs:enumeration value="Lesbian"/>
      <xs:enumeration value="Bisexual"/>
      <xs:enumeration value="Other"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="FAFSASStatusList">
    <xs:annotation>
      <xs:documentation>Sexual orientation of a student</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="Filed"/>
      <xs:enumeration value="Planning to file"/>
      <xs:enumeration value="Not planning to file"/>
      <xs:enumeration value="Unable to file"/>
      <xs:enumeration value="Other"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="MilitaryBranchList">
    <xs:annotation>
      <xs:documentation>Military branches in which a student may have
served</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="Army"/>
      <xs:enumeration value="Navy"/>
      <xs:enumeration value="Air Force"/>
      <xs:enumeration value="Marines"/>
      <xs:enumeration value="Coast Guard"/>
    </xs:restriction>
  </xs:simpleType>

```

```

        <xs:enumeration value="National Guard"/>
        <xs:enumeration value="Army Reserve"/>
        <xs:enumeration value="Navy Reserve"/>
        <xs:enumeration value="Air Force Reserve"/>
        <xs:enumeration value="Marines Reserve"/>
        <xs:enumeration value="Coast Guard Reserve"/>
        <xs:enumeration value="National Guard Reserve"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="DependencyStatusList">
    <xs:annotation>
        <xs:documentation>List of possible legal dependency status
names</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Independent adult"/>
        <xs:enumeration value="Dependent minor"/>
        <xs:enumeration value="Independent or emancipated minor"/>
        <xs:enumeration value="Dependent adult"/>
        <xs:enumeration value="Ward of the court"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Essay">
    <xs:annotation>
        <xs:documentation>A free-form essay written by the student on a given
topic</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="EssayType" type="sds:EssayTypes">
            <xs:annotation>
                <xs:documentation>The type of essay provided</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="EssayTypeOtherDesc" type="sds:LongString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>Explanation of the essay or the question that was
asked in a custom essay question</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="EssayContent" type="sds:LongString">
            <xs:annotation>
                <xs:documentation>The text contents of the essay
itself</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="EssayTypes">
    <xs:annotation>
        <xs:documentation>The essay categories that are
available</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">

```

```

<xs:enumeration value="Post graduation and career goals"/>
<xs:enumeration value="Education plans beyond graduation"/>
<xs:enumeration value="Leadership experiences"/>
<xs:enumeration value="Organizational leadership roles and outcomes"/>
<xs:enumeration value="Lifetime goals and objectives"/>
<xs:enumeration value="Personal background or history"/>
<xs:enumeration value="Family background or history"/>
<xs:enumeration value="Affecting the lives of others"/>
<xs:enumeration value="Community contribution"/>
<xs:enumeration value="Community leadership"/>
<xs:enumeration value="The community impact upon the student"/>
<xs:enumeration value="Favorite activity"/>
<xs:enumeration value="Home responsibilities"/>
<xs:enumeration value="Overcoming adversity"/>
<xs:enumeration value="Work responsibilities"/>
<xs:enumeration value="Personal achievements, awards, and honors"/>
<xs:enumeration value="Special circumstances affecting college
finance"/>
<xs:enumeration value="Why student deserves the scholarship"/>
<xs:enumeration value="How ethnicity shaped the student"/>
<xs:enumeration value="How college readiness programs helped the
student"/>
<xs:enumeration value="Athletic involvement, achievements, and impact
on the student"/>
<xs:enumeration value="Special programs participation and outcomes"/>
<xs:enumeration value="Explanation of student's college choices"/>
<xs:enumeration value="Explanation of student's major choices"/>
<xs:enumeration value="Failures and the resulting lessons learned"/>
<xs:enumeration value="How the student challenged herself"/>
<xs:enumeration value="Disciplinary or legal problems"/>
<xs:enumeration value="Other"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="CollegeReadinessProgram">
  <xs:annotation>
    <xs:documentation>The student's participation in a college readiness
program</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="ProgramName" type="sds:CollegeReadinessProgramList"/>
    <xs:element name="ProgramOtherDesc" type="sds:ShortString"
minOccurs="0"/>
    <xs:element name="CounselorFirstName" type="sds:ShortString"
minOccurs="0"/>
    <xs:element name="CounselorLastName" type="sds:ShortString"
minOccurs="0"/>
    <xs:element name="CounselorEmail" type="sds:ShortString"
minOccurs="0"/>
    <xs:element name="LengthOfParticipation" type="xs:duration"
minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="CollegeReadinessProgramList">
  <xs:annotation>
    <xs:documentation>List of college readiness programs</xs:documentation>

```

```

</xs:annotation>
<xs:restriction base="sds:ShortString">
  <xs:enumeration value="AP Strategies"/>
  <xs:enumeration value="Aspire"/>
  <xs:enumeration value="AVID"/>
  <xs:enumeration value="GEAR UP"/>
  <xs:enumeration value="Green Dot Public Schools"/>
  <xs:enumeration value="KIPP Academy"/>
  <xs:enumeration value="Noble Charter"/>
  <xs:enumeration value="Philadelphia Futures"/>
  <xs:enumeration value="PUC"/>
  <xs:enumeration value="Upward Bound"/>
  <xs:enumeration value="Upward Bound - Math and Science"/>
  <xs:enumeration value="YES Charter"/>
  <xs:enumeration value="Breakthrough Austin"/>
  <xs:enumeration value="IDEA Academy"/>
  <xs:enumeration value="Other"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="MaritalStatusList">
  <xs:annotation>
    <xs:documentation>The list of marital status choices</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Single"/>
    <xs:enumeration value="Married"/>
    <xs:enumeration value="Re-Married"/>
    <xs:enumeration value="Divorced"/>
    <xs:enumeration value="Legally Separated"/>
    <xs:enumeration value="Informally Separated"/>
    <xs:enumeration value="Widowed"/>
    <xs:enumeration value="Deceased"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="Reference">
  <xs:annotation>
    <xs:documentation>Type that represents a person who is not a student
nor a relative, for example counselors, recommenders, etc.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="ReferenceName" type="sds:Name" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's salutation, such as Mr., Mrs., Dr.
and so on</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ReferenceContactInfo" type="sds:ContactInfo"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>Methods by which to reach the
person</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Relationship" type="sds:RelationshipList">

```

```

    <xs:annotation>
      <xs:documentation>The nature of the relationship between the person
and the student when not family</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="RelationshipOtherDesc" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>The full text description when 'other' was
chosen</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="YearsKnown" type="xs:int" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The number of years that the person has known the
student</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="TitleOrOccupation" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>The person's title or
occupation</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="CompanyName" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The name of the organization</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="ReferenceAddress" type="sds:Address" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The person's full address</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Spouse">
  <xs:annotation>
    <xs:documentation>Type that represents a student's
spouse</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="SpouseName" type="sds:Name">
      <xs:annotation>
        <xs:documentation>The person's salutation, such as Mr., Mrs., Dr.
and so on</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Employed" type="xs:boolean" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the person is currently
employed</xs:documentation>
      </xs:annotation>
    </xs:element>

```

```

    <xs:element name="TitleOrOccupation" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
    <xs:documentation>The person's title or
occupation</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="CompanyName" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The name of the organization</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="SpouseIncome" type="sds:Money" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The person's income</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="Disabled" type="xs:boolean" minOccurs="0">
    <xs:annotation>
    <xs:documentation>Whether the person indicates a
disability</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="DisabilityDesc" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The description of a person's
disability</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="WorkingHoursPerWeek" type="xs:int" minOccurs="0">
    <xs:annotation>
    <xs:documentation>the number of hours per week that the person
works</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="SpouseEducationLevel" type="sds:EducationLevel"
minOccurs="0">
    <xs:annotation>
    <xs:documentation>The person's education level</xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Relative">
    <xs:annotation>
    <xs:documentation>Type that represents a person who is a relative of
the student, for example father, mother, uncle, etc.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
    <xs:element name="RelativeName" type="sds:Name" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The person's salutation, such as Mr., Mrs., Dr.
and so on</xs:documentation>
    </xs:annotation>
</xs:element>

```

```

    <xs:element name="Relation" type="sds:FamilyTitlesList">
      <xs:annotation>
        <xs:documentation>The type of relative defined by the relationships
in a family tree</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="RelationOtherDesc" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The type of relative described when the 'other'
choice is selected</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Living" type="xs:boolean" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the relative is alive or
deceased</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="BirthDate" type="xs:date" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Person's date of birth</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="MaritalStatus" type="sds:MaritalStatusList"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's marital status</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Employed" type="xs:boolean" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the person is currently
employed</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="TitleOrOccupation" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's title or
occupation</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="CompanyName" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The name of the organization</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="RelativeIncome" type="sds:Money" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's income</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Disabled" type="xs:boolean" minOccurs="0">
      <xs:annotation>

```

```

        <xs:documentation>Whether the person indicates a
disability</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="DisabilityDesc" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The description of a person's
disability</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="WorkingHoursPerWeek" type="xs:int" minOccurs="0">
        <xs:annotation>
            <xs:documentation>the number of hours per week that the person
works</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="RelativeEducationLevel" type="sds:EducationLevel"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>Person's highest completed level of
school</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="RelationshipList">
    <xs:annotation>
        <xs:documentation>The different relationships that a person can
have</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Counselor"/>
        <xs:enumeration value="Registrar"/>
        <xs:enumeration value="Supervisor"/>
        <xs:enumeration value="Recommender"/>
        <xs:enumeration value="Teacher/Professor"/>
        <xs:enumeration value="Relative"/>
        <xs:enumeration value="Mentor"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ScholarshipPurpose">
    <xs:annotation>
        <xs:documentation>The student's reasons for pursuing a
scholarship</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Purpose" type="sds:ScholarshipPurposeList">
            <xs:annotation>
                <xs:documentation>The intended purpose of the student's scholarship
application</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="PurposeOtherDesc" type="sds:ShortString"
minOccurs="0">

```

```

        <xs:annotation>
          <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="ScholarshipPurposeList">
    <xs:annotation>
      <xs:documentation>The types of schools for which a student may use a
scholarship</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="High school"/>
      <xs:enumeration value="Certificate program"/>
      <xs:enumeration value="2-year program"/>
      <xs:enumeration value="4-year program"/>
      <xs:enumeration value="Graduate program"/>
      <xs:enumeration value="Other"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="School">
    <xs:annotation>
      <xs:documentation>A type to store typical school
information</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="SchoolName" type="sds:ShortString">
        <xs:annotation>
          <xs:documentation>The official name of the
school</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="NCESCode" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The National Center for Education Statistics
(NCES) code for the school</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="CEEBCode" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The College Board registry number for the
school</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="SchoolType" type="sds:SchoolTypesList" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The type of the school</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="OtherSchoolDesc" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

        </xs:annotation>
    </xs:element>
    <xs:element name="SchoolAddress" type="sds:Address" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The full address of the school</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="SchoolTypesList">
    <xs:annotation>
        <xs:documentation>The types of schools that a student could
attend</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Elementary school"/>
        <xs:enumeration value="Middle school"/>
        <xs:enumeration value="High school"/>
        <xs:enumeration value="Technical school"/>
        <xs:enumeration value="College"/>
        <xs:enumeration value="University"/>
        <xs:enumeration value="Home school"/>
        <xs:enumeration value="Graduate school"/>
        <xs:enumeration value="Other school"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Enrollment">
    <xs:annotation>
        <xs:documentation>The types schools in which a student is
enrolled</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="School" type="sds:School">
            <xs:annotation>
                <xs:documentation>The student's school</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="SchoolProgram" type="sds:SchoolProgramTypesList"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The school program in which the student
enrolled</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="EnrollmentOtherDesc" type="sds:ShortString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="ExpectedGraduationDate" type="xs:date" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The expected graduation date from the
program</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>

```

```

        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="SchoolProgramTypesList">
    <xs:annotation>
        <xs:documentation>The types of school enrollments available to a
student</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Not in school"/>
        <xs:enumeration value="High school"/>
        <xs:enumeration value="Certificate program"/>
        <xs:enumeration value="2-year program"/>
        <xs:enumeration value="4-year program"/>
        <xs:enumeration value="Graduate program"/>
        <xs:enumeration value="Home school program"/>
        <xs:enumeration value="High school equivalence (GED) program"/>
        <xs:enumeration value="Other program"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="GraduationInfo">
    <xs:annotation>
        <xs:documentation>The student's graduation and/or GED
information</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Graduated" type="xs:boolean">
            <xs:annotation>
                <xs:documentation>Whether the student has graduated from any
school</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="GraduationType" type="sds:SchoolProgramTypesList"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The type of school from which the student
graduated</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="GraduationTypeOtherDesc" type="sds:ShortString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The free text explanation when the student chose
'other'</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="GraduationSchoolName" type="sds:ShortString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The school where the student
graduated</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="GraduationDate" type="xs:date" minOccurs="0">

```

```

        <xs:annotation>
          <xs:documentation>The date that the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GraduationCity" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The city where the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GraduationCounty" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The county where the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GraduationStateorProvince" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The state where the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GraduationZIPOrPostalCode" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The school's zip or postal code where the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="GraduationCountry" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The country where the student
graduated</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="CollegeChoice">
    <xs:annotation>
      <xs:documentation>A college choice that the student has
made</xs:documentation>
    </xs:annotation>
  </xs:sequence>
    <xs:element name="CollegeName" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>The college choice name</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="NCESCode" type="sds:ShortString" minOccurs="0">
      <xs:annotation>

```

```

        <xs:documentation>The National Center for Education Statistics
(NCES) code for the school</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CEEBCode" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
        <xs:documentation>The College Board registry number for the
school</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CollegeChoiceRank" type="xs:int" minOccurs="0">
    <xs:annotation>
        <xs:documentation>The rank of the college choice per the student's
order of preference, for example first, second, third,
etc.</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CollegeApplicationStatus"
type="sds:ApplicationStatusList" minOccurs="0">
    <xs:annotation>
        <xs:documentation>The status of a student's application to a
college</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CollegeApplicationStatusOtherDesc"
type="sds:ShortString" minOccurs="0">
    <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="ApplicationStatusList">
    <xs:annotation>
        <xs:documentation>The list of application status
choices</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Not yet applied"/>
        <xs:enumeration value="Applied"/>
        <xs:enumeration value="Accepted"/>
        <xs:enumeration value="Waiting List"/>
        <xs:enumeration value="Denied"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ClassRank">
    <xs:annotation>
        <xs:documentation>The student's class rank
information</xs:documentation>
    </xs:annotation>
</xs:sequence>
    <xs:element name="SchoolName" type="sds:ShortString" minOccurs="0">
        <xs:annotation>

```

```

        <xs:documentation>The name of the school for which class rank is
reported</xs:documentation>
    </xs:annotation>
</xs:element>
    <xs:element name="SchoolType" type="sds:SchoolTypesList" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The type of the school for which class rank is
reported</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="OtherSchoolDesc" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SchoolRanksStudents" type="xs:boolean">
        <xs:annotation>
            <xs:documentation>Whether the school ranks
students</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="System" type="sds:RankSystemsList" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The system used by the school to rank
students</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SystemOtherDesc" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ClassRank" type="xs:int" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The student's class rank</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ClassSize" type="xs:int" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The student's class size when used to rank the
student</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ClassRankPercentile" type="xs:decimal" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The student's class rank expressed as a
percentile</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ClassRankPercentileBest" type="xs:decimal"
minOccurs="0">

```

```

        <xs:annotation>
            <xs:documentation>Whether 100% or 1% is considered the best class
rank, when expressed as a percentile</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="OtherRank" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The student's class rank using an alternate
method of class ranking</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="OtherRankDesc" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="RankMethod" type="sds:RankMethodsList" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The method used to deterime the student's class
rank, whether calculated or estimated</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="RankMethodsList">
    <xs:annotation>
        <xs:documentation>The possible methods to determine a student's class
rank</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Calculated"/>
        <xs:enumeration value="Estimated"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="RankSystemsList">
    <xs:annotation>
        <xs:documentation>The possible student ranking measurements to
communicate a student's class rank</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="Integer"/>
        <xs:enumeration value="Decile"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="TestsList">
    <xs:annotation>
        <xs:documentation>The different tests that a person can
take</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="SAT Critical Reading"/>
        <xs:enumeration value="SAT Verbal (Obsolete)"/>
        <xs:enumeration value="SAT Math"/>
    </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="SAT Writing"/>
<xs:enumeration value="ACT Composite"/>
<xs:enumeration value="ACT English"/>
<xs:enumeration value="ACT Math"/>
<xs:enumeration value="ACT Reading"/>
<xs:enumeration value="ACT Science"/>
<xs:enumeration value="ACT Writing"/>
<xs:enumeration value="PSAT Critical Reading"/>
<xs:enumeration value="PSAT Verbal (Obsolete)"/>
<xs:enumeration value="PSAT Math"/>
<xs:enumeration value="PSAT Writing"/>
<xs:enumeration value="PSAT Selection Index"/>
<xs:enumeration value="GMAT Verbal"/>
<xs:enumeration value="GMAT Quantitative"/>
<xs:enumeration value="GMAT Analytical Writing Assessment"/>
<xs:enumeration value="GRE Verbal"/>
<xs:enumeration value="GRE Quantitative"/>
<xs:enumeration value="LSAT Verbal"/>
<xs:enumeration value="LSAT Quant"/>
<xs:enumeration value="LSAT Analytic"/>
<xs:enumeration value="TOEFL Listening"/>
<xs:enumeration value="TOEFL Structure/Writing"/>
<xs:enumeration value="TOEFL Reading"/>
<xs:enumeration value="TOEFL Total Score"/>
<xs:enumeration value="MCAT Verbal Reasoning"/>
<xs:enumeration value="MCAT Physical Sciences"/>
<xs:enumeration value="MCAT Writing Sample"/>
<xs:enumeration value="MCAT Biological Sciences"/>
<xs:enumeration value="AP Art History"/>
<xs:enumeration value="AP Biology"/>
<xs:enumeration value="AP Calculus AB"/>
<xs:enumeration value="AP Calculus BC"/>
<xs:enumeration value="AP Chemistry"/>
<xs:enumeration value="AP Chinese Language and Culture"/>
<xs:enumeration value="AP Computer Science A"/>
<xs:enumeration value="AP Computer Science AB"/>
<xs:enumeration value="AP Macroeconomics"/>
<xs:enumeration value="AP Microeconomics"/>
<xs:enumeration value="AP English Language"/>
<xs:enumeration value="AP English Literature"/>
<xs:enumeration value="AP Environmental Science"/>
<xs:enumeration value="AP European History"/>
<xs:enumeration value="AP French Language"/>
<xs:enumeration value="AP French Literature"/>
<xs:enumeration value="AP German Language"/>
<xs:enumeration value="AP Comp Government and Politics"/>
<xs:enumeration value="AP U.S. Government and Politics"/>
<xs:enumeration value="AP Human Geography"/>
<xs:enumeration value="AP Italian Language and Culture"/>
<xs:enumeration value="AP Japanese Language and Culture"/>
<xs:enumeration value="AP Latin Literature"/>
<xs:enumeration value="AP Latin: Vergil"/>
<xs:enumeration value="AP Music Theory"/>
<xs:enumeration value="AP Physics B"/>
<xs:enumeration value="AP Physics C"/>

```

```

    <xs:enumeration value="AP Psychology"/>
    <xs:enumeration value="AP Spanish Language"/>
    <xs:enumeration value="AP Spanish Literature"/>
    <xs:enumeration value="AP Statistics"/>
    <xs:enumeration value="AP Studio Art"/>
    <xs:enumeration value="AP U.S. History"/>
    <xs:enumeration value="AP World History "/>
    <xs:enumeration value="GED Score"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="TestScore">
  <xs:annotation>
    <xs:documentation>Type that represents a particular test
score</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Test" type="sds:TestsList">
      <xs:annotation>
        <xs:documentation>The student's test taken</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="TestOtherDesc" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description when they chose
'other'</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Score" type="xs:decimal">
      <xs:annotation>
        <xs:documentation>The student's test score</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ScoreType" type="sds:ScoreTypes" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the student's score was the most recent
score or the highest score ever achieved on the test</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ScoreNumericType" type="sds:ScoreNumericTypes"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The type of number used to report the test
score</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ScoreDate" type="xs:date" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The date the score was
achieved</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="ScoreTypes">

```

```

    <xs:annotation>
      <xs:documentation>Whether the student's score was the most recent score
or the highest score ever achieved on the test</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="Most recent score"/>
      <xs:enumeration value="Highest ever score"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="ScoreNumericTypes">
    <xs:annotation>
      <xs:documentation>The type of numbers used to report the test
score</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
      <xs:enumeration value="Integer"/>
      <xs:enumeration value="Percent"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="GPA">
    <xs:annotation>
      <xs:documentation>Type that represents the student's GPA information
over multiple years</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="SchoolName" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The name of the school for which GPA is
reported</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="SchoolType" type="sds:SchoolTypesList" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The type of the school for which GPA is
reported</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="OtherSchoolDesc" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="CurrentSchoolCalculatesGPA" type="xs:boolean">
        <xs:annotation>
          <xs:documentation>Whether or not the school calculates student
GPA</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="CumulativeGPA" type="sds:GPAInfo" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The student's current cumulative grade point
average for the current school year</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:element>
    <xs:element name="GPAsPerSchoolYear" type="sds:GPAInfo" minOccurs="0"
maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation>The student's list of grade point averages for
previous school years</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="GPAInfo">
  <xs:annotation>
    <xs:documentation>Type that represents the student's GPA for a
particular school year</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="SchoolYear" type="sds:SchoolYearsList" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The school year for which a grade point average
is reported</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="GPA" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>The student's grade point average for the given
school year</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Weighted" type="xs:boolean" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the student's grade point average is
weighted</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Scale" type="sds:GradingScaleList">
      <xs:annotation>
        <xs:documentation>The scale used to calculate the student's grade
point average</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ScaleOther" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="GradingScaleList">
  <xs:annotation>
    <xs:documentation>The scale used to calculate the student's grade point
average</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="4.0"/>
  </xs:restriction>

```

```

    <xs:enumeration value="4.3"/>
    <xs:enumeration value="5.0"/>
    <xs:enumeration value="100"/>
    <xs:enumeration value="Alphabetic"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="GovernmentAid">
  <xs:annotation>
    <xs:documentation>Type that represents the various government programs
that a student could be associated with</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Programs" type="sds:GovernmentPrograms"
maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation>Government aid programs of which the student has
been a recipient</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ProgramOtherDesc" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="GovernmentPrograms">
  <xs:annotation>
    <xs:documentation>The list of government aid
programs</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="FreeOrReducedSchoolLunch">
      <xs:annotation>
        <xs:documentation>The National School Lunch Program (NSLP) from the
U.S. Department of Agriculture</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="SpecialSupplementalNutritionWIC">
      <xs:annotation>
        <xs:documentation>The Special Supplemental Nutrition Program for
Women, Infants, and Children from the U.S. Department of
Agriculture</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="TemporaryAssistanceToNeedyFamilies">
      <xs:annotation>
        <xs:documentation>The Temporary Assistance for Needy Families
(TANF) program from the U.S. Department of Health and Human
Services</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>

```

```

    <xs:enumeration value="LIHEAPOrOtherUtilityAssistance">
      <xs:annotation>
        <xs:documentation>The Low Income Home Energy Assistance Program
(LIHEAP) from the U.S. Department of Health and Human
Services</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="SupplementalSocialSecurity">
      <xs:annotation>
        <xs:documentation>Supplemental Security Income (SSI) from the U.S.
Social Security Administration</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="OtherGovernmentProgram">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="Class">
  <xs:annotation>
    <xs:documentation>The peron's class information</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="ClassName" type="sds:ClassList">
      <xs:annotation>
        <xs:documentation>The name of the student's
course</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ClassNameOtherDesc" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The name of the student's course if 'other' was
chosen</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Year" type="sds:SchoolYearsList" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The school year during which the student
participated in the class</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="YearOtherDesc" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Status" type="sds:ClassStatusList" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The status of a student's participation in the
course</xs:documentation>

```

```

        </xs:annotation>
    </xs:element>
    <xs:element name="Grade" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The grade that the student earned in a given
course</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Scale" type="sds:GradingScaleList" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The scale used to calculate the student's grade
point average</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Type" type="sds:ClassTypeList" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
            <xs:documentation>Any special designation of the
course</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="TypeOtherDesc" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SchoolName" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The name of the school where the student attended
the class</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="ClassList">
    <xs:annotation>
        <xs:documentation>The different classes that a person can
attend</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="English grade 9"/>
        <xs:enumeration value="English grade 10"/>
        <xs:enumeration value="English grade 11"/>
        <xs:enumeration value="English grade 12"/>
        <xs:enumeration value="Algebra 1"/>
        <xs:enumeration value="Geometry"/>
        <xs:enumeration value="Algebra 2"/>
        <xs:enumeration value="Trigonometry"/>
        <xs:enumeration value="Pre-Calculus"/>
        <xs:enumeration value="Calculus"/>
        <xs:enumeration value="Other advanced math"/>
        <xs:enumeration value="Computer science"/>
        <xs:enumeration value="Earth or general science"/>
        <xs:enumeration value="Biology"/>
    </xs:restriction>

```

```

<xs:enumeration value="Chemistry"/>
<xs:enumeration value="Physics"/>
<xs:enumeration value="Anatomy"/>
<xs:enumeration value="Physiology"/>
<xs:enumeration value="Other science"/>
<xs:enumeration value="U.S. History"/>
<xs:enumeration value="World History"/>
<xs:enumeration value="Government or civics"/>
<xs:enumeration value="Economics"/>
<xs:enumeration value="World culture or global studies"/>
<xs:enumeration value="Geography"/>
<xs:enumeration value="Psychology"/>
<xs:enumeration value="Sociology"/>
<xs:enumeration value="Foreign language"/>
<xs:enumeration value="Philosophy"/>
<xs:enumeration value="Other social studies"/>
<xs:enumeration value="Other courses"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ClassStatusList">
  <xs:annotation>
    <xs:documentation>The status of a student's participation in a given
course</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Complete"/>
    <xs:enumeration value="In progress"/>
    <xs:enumeration value="Pending"/>
    <xs:enumeration value="Not taken"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ClassTypeList">
  <xs:annotation>
    <xs:documentation>The type of student courses</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Senior"/>
    <xs:enumeration value="Honors"/>
    <xs:enumeration value="AP"/>
    <xs:enumeration value="IB"/>
    <xs:enumeration value="Dual enrollment"/>
    <xs:enumeration value="Special education"/>
    <xs:enumeration value="Remedial"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActivityTypeList">
  <xs:annotation>
    <xs:documentation>The types of student activities</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Community service or volunteering"/>
    <xs:enumeration value="Employment"/>
    <xs:enumeration value="Athletics"/>
    <xs:enumeration value="Arts"/>
  </xs:restriction>

```

```

<xs:enumeration value="School organization or club"/>
<xs:enumeration value="Major award or honor"/>
<xs:enumeration value="College readiness program"/>
<xs:enumeration value="Foreign travel or study"/>
<xs:enumeration value="Home and family responsibilities"/>
<xs:enumeration value="Hobby"/>
<xs:enumeration value="Government"/>
<xs:enumeration value="Internship"/>
<xs:enumeration value="Military"/>
<xs:enumeration value="Other activity"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="Activity">
  <xs:annotation>
    <xs:documentation>The person's activity information</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Type" type="sds:ActivityTypeList">
      <xs:annotation>
        <xs:documentation>The type of student activity</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Year" type="sds:SchoolYearsList" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The school year during which the student
participated in the activity</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="YearOtherDesc" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Description" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The description of the student's
activity</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="OrganizationName" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The name of the organization</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="OrganizationAddress" type="sds:Address"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The organization's full
address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="StartDate" type="xs:date" minOccurs="0">
      <xs:annotation>

```

```

        <xs:documentation>When the student started the
activity</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="EndDate" type="xs:date" minOccurs="0">
        <xs:annotation>
            <xs:documentation>When the student stopped the
activity</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="GraduationRequirementHours" type="xs:int"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The number of activity hours that were part of a
graduation requirement</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SummerHoursPerWeek" type="xs:int" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The summer hours per week that the student
participated in the activity</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SummerWeeksPerYear" type="xs:int" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The summer weeks per year that the student
participated in the activity</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SchoolSeasonHoursPerWeek" type="xs:int"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The school season hours per week that the student
participated in the activity</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SchoolSeasonWeeksPerYear" type="xs:int"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The school season weeks per year that the student
participated in the activity</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="SchoolYearsList">
    <xs:annotation>
        <xs:documentation>The years of school that a student can
attend</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="High School Grade 9"/>
        <xs:enumeration value="High School Grade 10"/>
        <xs:enumeration value="High School Grade 11"/>
        <xs:enumeration value="High School Grade 12"/>
    </xs:restriction>
</xs:simpleType>

```

```

<xs:enumeration value="College Year 1"/>
<xs:enumeration value="College Year 2"/>
<xs:enumeration value="College Year 3"/>
<xs:enumeration value="College Year 4"/>
<xs:enumeration value="College Year 5"/>
<xs:enumeration value="College Year 6"/>
<xs:enumeration value="College Year 7"/>
<xs:enumeration value="College Year 8"/>
<xs:enumeration value="Graduate School Year 1"/>
<xs:enumeration value="Graduate School Year 2"/>
<xs:enumeration value="Graduate School Year 3"/>
<xs:enumeration value="Graduate School Year 4"/>
<xs:enumeration value="Other"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="FamilyTitlesList">
  <xs:annotation>
    <xs:documentation>The different titles for a family
member</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Spouse"/>
    <xs:enumeration value="Guardian"/>
    <xs:enumeration value="Significant Other"/>
    <xs:enumeration value="Domestic Partner"/>
    <xs:enumeration value="Father"/>
    <xs:enumeration value="Mother"/>
    <xs:enumeration value="Grandfather"/>
    <xs:enumeration value="Grandmother"/>
    <xs:enumeration value="Son"/>
    <xs:enumeration value="Daughter"/>
    <xs:enumeration value="Fosterparent"/>
    <xs:enumeration value="Fosterchild"/>
    <xs:enumeration value="Grandson"/>
    <xs:enumeration value="Granddaughter"/>
    <xs:enumeration value="Brother"/>
    <xs:enumeration value="Sister"/>
    <xs:enumeration value="Uncle"/>
    <xs:enumeration value="Aunt"/>
    <xs:enumeration value="Cousin"/>
    <xs:enumeration value="Nephew"/>
    <xs:enumeration value="Niece"/>
    <xs:enumeration value="Stepfather"/>
    <xs:enumeration value="Stepmother"/>
    <xs:enumeration value="Stepson"/>
    <xs:enumeration value="Stepdaughter"/>
    <xs:enumeration value="Stepparent"/>
    <xs:enumeration value="Stepchild"/>
    <xs:enumeration value="Stepbrother"/>
    <xs:enumeration value="Stepsister"/>
    <xs:enumeration value="Father in law"/>
    <xs:enumeration value="Mother in law"/>
    <xs:enumeration value="Son in law"/>
    <xs:enumeration value="Daughter in law"/>
    <xs:enumeration value="Parent in law"/>
  </xs:restriction>
</xs:simpleType>

```

```

        <xs:enumeration value="Child in law"/>
        <xs:enumeration value="Brother in law"/>
        <xs:enumeration value="Sister in law"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="EducationLevel">
    <xs:annotation>
        <xs:documentation>The levels of education for a
person</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="EducationLevel" type="sds:EducationLevelsList">
            <xs:annotation>
                <xs:documentation>The highest level of education that the person
completed</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="EducationLevelOther" type="sds:ShortString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="EducationLevelsList">
    <xs:annotation>
        <xs:documentation>The levels of education that a person may
indicate</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="No school"/>
        <xs:enumeration value="Elementary school"/>
        <xs:enumeration value="Middle school"/>
        <xs:enumeration value="Some high school"/>
        <xs:enumeration value="High school graduate"/>
        <xs:enumeration value="Some college"/>
        <xs:enumeration value="Associate's degree"/>
        <xs:enumeration value="Bachelor's degree"/>
        <xs:enumeration value="Master's degree"/>
        <xs:enumeration value="Doctorate"/>
        <xs:enumeration value="Professional degree"/>
        <xs:enumeration value="Unknown"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="HomeEnvironment">
    <xs:annotation>
        <xs:documentation>The person's home environment
information</xs:documentation>
    </xs:annotation>
    <xs:sequence>

```

```

    <xs:element name="LivingWith" type="sds:LivingWithList"
maxOccurs="unbounded">
    <xs:annotation>
    <xs:documentation>The type of people with whom the student
lives</xs:documentation>
    </xs:annotation>
    </xs:element>
    <xs:element name="LivingWithOtherDesc" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
    <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
    </xs:annotation>
    </xs:element>
    <xs:element name="ResidentsInHomeCount" type="xs:int" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The number of people that live in the student's
home, not counting themselves</xs:documentation>
    </xs:annotation>
    </xs:element>
    <xs:element name="NativeLanguage" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
    <xs:documentation>The language that the student speaks at home or
what they indicate as their native language</xs:documentation>
    </xs:annotation>
    </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="LivingWithList">
    <xs:annotation>
    <xs:documentation>The list of people with whom the student might
live</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Alone"/>
    <xs:enumeration value="Mother"/>
    <xs:enumeration value="Father"/>
    <xs:enumeration value="Step-parents"/>
    <xs:enumeration value="Sibling"/>
    <xs:enumeration value="Grandparents"/>
    <xs:enumeration value="Guardian"/>
    <xs:enumeration value="Adopted family"/>
    <xs:enumeration value="Foster family"/>
    <xs:enumeration value="Other relatives"/>
    <xs:enumeration value="Spouse or significant other"/>
    <xs:enumeration value="Your children"/>
    <xs:enumeration value="Friends"/>
    <xs:enumeration value="Group home"/>
    <xs:enumeration value="Homeless"/>
    <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Money">
    <xs:annotation>
    <xs:documentation>An amount of money</xs:documentation>

```

```

</xs:annotation>
<xs:sequence>
  <xs:element name="Amount" type="xs:decimal">
    <xs:annotation>
      <xs:documentation>The money amount</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="Currency" type="sds:CurrencyList">
    <xs:annotation>
      <xs:documentation>The national currency of the
amount</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="CurrencyList">
  <xs:annotation>
    <xs:documentation>Currency abbreviations derived from the ISO 4217
standard</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="USD"/>
    <xs:enumeration value="CAD"/>
    <xs:enumeration value="AED"/>
    <xs:enumeration value="AFN"/>
    <xs:enumeration value="ALL"/>
    <xs:enumeration value="AMD"/>
    <xs:enumeration value="ANG"/>
    <xs:enumeration value="AOA"/>
    <xs:enumeration value="ARS"/>
    <xs:enumeration value="AUD"/>
    <xs:enumeration value="AWG"/>
    <xs:enumeration value="AZN"/>
    <xs:enumeration value="BAM"/>
    <xs:enumeration value="BBD"/>
    <xs:enumeration value="BDT"/>
    <xs:enumeration value="BGN"/>
    <xs:enumeration value="BHD"/>
    <xs:enumeration value="BIF"/>
    <xs:enumeration value="BMD"/>
    <xs:enumeration value="BND"/>
    <xs:enumeration value="BOB"/>
    <xs:enumeration value="BRL"/>
    <xs:enumeration value="BSD"/>
    <xs:enumeration value="BTN"/>
    <xs:enumeration value="BWP"/>
    <xs:enumeration value="BYR"/>
    <xs:enumeration value="BZD"/>
    <xs:enumeration value="CDF"/>
    <xs:enumeration value="CHF"/>
    <xs:enumeration value="CLP"/>
    <xs:enumeration value="CNY"/>
    <xs:enumeration value="COP"/>
    <xs:enumeration value="CRC"/>
    <xs:enumeration value="CUP"/>
  </xs:restriction>
</xs:simpleType>

```

```
<xs:enumeration value="CVE"/>
<xs:enumeration value="CYP"/>
<xs:enumeration value="CZK"/>
<xs:enumeration value="DJF"/>
<xs:enumeration value="DKK"/>
<xs:enumeration value="DOP"/>
<xs:enumeration value="DZD"/>
<xs:enumeration value="EEK"/>
<xs:enumeration value="EGP"/>
<xs:enumeration value="ERN"/>
<xs:enumeration value="ETB"/>
<xs:enumeration value="EUR"/>
<xs:enumeration value="FJD"/>
<xs:enumeration value="FKP"/>
<xs:enumeration value="GBP"/>
<xs:enumeration value="GEL"/>
<xs:enumeration value="GGP"/>
<xs:enumeration value="GHS"/>
<xs:enumeration value="GIP"/>
<xs:enumeration value="GMD"/>
<xs:enumeration value="GNF"/>
<xs:enumeration value="GTQ"/>
<xs:enumeration value="GYD"/>
<xs:enumeration value="HKD"/>
<xs:enumeration value="HNL"/>
<xs:enumeration value="HRK"/>
<xs:enumeration value="HTG"/>
<xs:enumeration value="HUF"/>
<xs:enumeration value="IDR"/>
<xs:enumeration value="ILS"/>
<xs:enumeration value="IMP"/>
<xs:enumeration value="INR"/>
<xs:enumeration value="IQD"/>
<xs:enumeration value="IRR"/>
<xs:enumeration value="ISK"/>
<xs:enumeration value="JEP"/>
<xs:enumeration value="JMD"/>
<xs:enumeration value="JOD"/>
<xs:enumeration value="JPY"/>
<xs:enumeration value="KES"/>
<xs:enumeration value="KGS"/>
<xs:enumeration value="KHR"/>
<xs:enumeration value="KMF"/>
<xs:enumeration value="KPW"/>
<xs:enumeration value="KWD"/>
<xs:enumeration value="KYD"/>
<xs:enumeration value="KZT"/>
<xs:enumeration value="LAK"/>
<xs:enumeration value="LBP"/>
<xs:enumeration value="LKR"/>
<xs:enumeration value="LRD"/>
<xs:enumeration value="LSL"/>
<xs:enumeration value="LTL"/>
<xs:enumeration value="LVL"/>
<xs:enumeration value="LYD"/>
```

```
<xs:enumeration value="MAD"/>
<xs:enumeration value="MDL"/>
<xs:enumeration value="MGA"/>
<xs:enumeration value="MKD"/>
<xs:enumeration value="MMK"/>
<xs:enumeration value="MNT"/>
<xs:enumeration value="MOP"/>
<xs:enumeration value="MRO"/>
<xs:enumeration value="MTL"/>
<xs:enumeration value="MUR"/>
<xs:enumeration value="MVR"/>
<xs:enumeration value="MWK"/>
<xs:enumeration value="MXN"/>
<xs:enumeration value="MYR"/>
<xs:enumeration value="MZN"/>
<xs:enumeration value="NAD"/>
<xs:enumeration value="NGN"/>
<xs:enumeration value="NIO"/>
<xs:enumeration value="NOK"/>
<xs:enumeration value="NPR"/>
<xs:enumeration value="NZD"/>
<xs:enumeration value="OMR"/>
<xs:enumeration value="PAB"/>
<xs:enumeration value="PEN"/>
<xs:enumeration value="PGK"/>
<xs:enumeration value="PHP"/>
<xs:enumeration value="PKR"/>
<xs:enumeration value="PLN"/>
<xs:enumeration value="PYG"/>
<xs:enumeration value="QAR"/>
<xs:enumeration value="RON"/>
<xs:enumeration value="RSD"/>
<xs:enumeration value="RUB"/>
<xs:enumeration value="RWF"/>
<xs:enumeration value="SAR"/>
<xs:enumeration value="SBD"/>
<xs:enumeration value="SCR"/>
<xs:enumeration value="SDG"/>
<xs:enumeration value="SEK"/>
<xs:enumeration value="SGD"/>
<xs:enumeration value="SHP"/>
<xs:enumeration value="SKK"/>
<xs:enumeration value="SLL"/>
<xs:enumeration value="SOS"/>
<xs:enumeration value="SPL"/>
<xs:enumeration value="SRD"/>
<xs:enumeration value="STD"/>
<xs:enumeration value="SVC"/>
<xs:enumeration value="SYP"/>
<xs:enumeration value="SZL"/>
<xs:enumeration value="THB"/>
<xs:enumeration value="TJS"/>
<xs:enumeration value="TMM"/>
<xs:enumeration value="TND"/>
<xs:enumeration value="TOP"/>
```

```

<xs:enumeration value="TRY"/>
<xs:enumeration value="TTD"/>
<xs:enumeration value="TVD"/>
<xs:enumeration value="TWD"/>
<xs:enumeration value="TZS"/>
<xs:enumeration value="UAH"/>
<xs:enumeration value="UGX"/>
<xs:enumeration value="UYU"/>
<xs:enumeration value="UZS"/>
<xs:enumeration value="VEB"/>
<xs:enumeration value="VEF"/>
<xs:enumeration value="VND"/>
<xs:enumeration value="VUV"/>
<xs:enumeration value="WST"/>
<xs:enumeration value="XAF"/>
<xs:enumeration value="XAG"/>
<xs:enumeration value="XAU"/>
<xs:enumeration value="XCD"/>
<xs:enumeration value="XDR"/>
<xs:enumeration value="XOF"/>
<xs:enumeration value="XPD"/>
<xs:enumeration value="XPF"/>
<xs:enumeration value="XPT"/>
<xs:enumeration value="YER"/>
<xs:enumeration value="ZAR"/>
<xs:enumeration value="ZMK"/>
<xs:enumeration value="ZWD"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="ContactInfo">
  <xs:annotation>
    <xs:documentation>The ways to contact a person</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Email" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>The person's email address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="PrimaryPhoneNumber" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's primary telephone
number</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="IsPrimaryPhoneTextCapable" type="xs:boolean"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>Whether the telephone can receive text
messages</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="AlternatePhoneNumber" type="sds:ShortString"
minOccurs="0">

```

```

        <xs:annotation>
            <xs:documentation>The person's alternate telephone
number</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="IsAlternatePhoneTextCapable" type="xs:boolean"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>Whether the telephone can receive text
messages</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="FaxNumber" type="sds:ShortString" minOccurs="0">
        <xs:annotation>
            <xs:documentation>The person's fax phone number</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="OtherMessagingAddress" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The person's alternate messaging address or
nickname</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="OtherMessagingAddressType" type="sds:ShortString"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The name of the alternate messaging system, for
example MySpace, Facebook, Yahoo Messenger, MSN Messenger, AOL Instant
Messenger, and so on</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="PreferredContactMethod" type="sds:ContactMethodsList"
minOccurs="0">
        <xs:annotation>
            <xs:documentation>The contact method that the person indicates is
the preferred way</xs:documentation>
        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="ContactMethodsList">
    <xs:annotation>
        <xs:documentation>The list of possible contact
methods</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="PrimaryPhone"/>
        <xs:enumeration value="AlternatePhone"/>
        <xs:enumeration value="Fax"/>
        <xs:enumeration value="TextMessagePrimaryPhone"/>
        <xs:enumeration value="TextMessageAlternatePhone"/>
        <xs:enumeration value="OtherMessaging"/>
    </xs:restriction>
</xs:simpleType>

```

```

<xs:complexType name="Name">
  <xs:annotation>
    <xs:documentation>The person's full name information</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Salutation" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's salutation, such as Mr., Mrs., Dr.
and so on</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="FirstName" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>The person's first name, given name, or
forename</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="MiddleName" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's middle name, if
applicable</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="MiddleName2" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's second middle name, if
applicable</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="LastName" type="sds:ShortString">
      <xs:annotation>
        <xs:documentation>The person's last name, surname, or family
name</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="Suffix" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The person's name suffix, such as II, Jr., Sr.,
and so on</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="PreferredName" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The name that the person prefers to be
called</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="GenderList">
  <xs:annotation>
    <xs:documentation>The person's gender</xs:documentation>
  </xs:annotation>

```

```

<xs:restriction base="sds:ShortString">
  <xs:enumeration value="Male"/>
  <xs:enumeration value="Female"/>
</xs:restriction>
</xs:simpleType>

<xs:simpleType name="GenderDetailList">
  <xs:annotation>
    <xs:documentation>The person's detailed gender list</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Transgender"/>
    <xs:enumeration value="Intersex"/>
    <xs:enumeration value="Male to Female"/>
    <xs:enumeration value="Female to Male"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="Address">
  <xs:annotation>
    <xs:documentation>An address for a person or organization in a standard
format</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="AddressType" type="sds:AddressTypesList"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The type of address provided</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="AddressLine1" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The first line of the street
address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="AddressLine2" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The second line of the street
address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="AddressLine3" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The third line of the street
address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="AddressLine4" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The fourth line of the street
address</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="City" type="sds:ShortString" minOccurs="0">

```

```

    <xs:annotation>
      <xs:documentation>The city</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="County" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The county, if applicable</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="StateOrProvince" type="sds:ShortString">
    <xs:annotation>
      <xs:documentation>The state or province</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="ZipOrPostalCode" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>The zip code or postal code</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="Country" type="sds:ShortString" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The country</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="AddressTypesList">
  <xs:annotation>
    <xs:documentation>The different kinds of addresses for a
person</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="Current address"/>
    <xs:enumeration value="Permanent address"/>
    <xs:enumeration value="Mailing/shipping address"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="Citizenship">
  <xs:annotation>
    <xs:documentation>Details about a person's citizenship for a
country</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="CitizenshipStatus" type="sds:CitizenshipStatusList">
      <xs:annotation>
        <xs:documentation>The status of the person's
citizenship</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="CitizenshipStatusOtherDesc" type="sds:ShortString"
minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

```

    </xs:annotation>
  </xs:element>
  <xs:element name="CitizenshipDate" type="xs:date" minOccurs="0">
    <xs:annotation>
      <xs:documentation>The date of the person's citizenship, if
applicable</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="CitizenshipCountry" type="sds:ShortString"
minOccurs="0">
    <xs:annotation>
      <xs:documentation>The country of which the person is a
citizen</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="CitizenshipStatusList">
  <xs:annotation>
    <xs:documentation>The list of possible citizenship
status</xs:documentation>
  </xs:annotation>
  <xs:restriction base="sds:ShortString">
    <xs:enumeration value="U.S. Citizen"/>
    <xs:enumeration value="Canadian Citizen"/>
    <xs:enumeration value="Resident Alien or Permanent Resident"/>
    <xs:enumeration value="Refugee or Protected Person"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="BirthPlace">
  <xs:annotation>
    <xs:documentation>The location where a person was
born</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="City" type="sds:ShortString" minOccurs="0"/>
    <xs:element name="StateOrProvince" type="sds:ShortString"/>
    <xs:element name="Country" type="sds:ShortString" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="Race">
  <xs:annotation>
    <xs:documentation>Information about a person's race</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Race" type="sds:RacesList" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation>The person's indicated races</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="RaceOtherDesc" type="sds:ShortString" minOccurs="0">
      <xs:annotation>
        <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>

```

```

        </xs:annotation>
    </xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="RacesList">
    <xs:annotation>
        <xs:documentation>The official list of races according to the U.S.
Census Bureau</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="AmericanIndianAndAlaskaNative"/>
        <xs:enumeration value="AsianOrPacificIslander"/>
        <xs:enumeration value="Black"/>
        <xs:enumeration value="White"/>
        <xs:enumeration value="Other"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Ethnicity">
    <xs:annotation>
        <xs:documentation>The person's ethnicity information</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Ethnicity" type="sds:EthnicitiesList"
maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>The person's indicated
ethnicities</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="EthnicityOtherDesc" type="sds:ShortString"
minOccurs="0">
            <xs:annotation>
                <xs:documentation>The full text description provided when the
'other' choice was selected</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="EthnicitiesList">
    <xs:annotation>
        <xs:documentation>The list of ethnicity choices</xs:documentation>
    </xs:annotation>
    <xs:restriction base="sds:ShortString">
        <xs:enumeration value="African"/>
        <xs:enumeration value="AfricanAmerican"/>
        <xs:enumeration value="Albanian"/>
        <xs:enumeration value="American"/>
        <xs:enumeration value="AmericanIndian"/>
        <xs:enumeration value="Arab"/>
        <xs:enumeration value="Armenian"/>
        <xs:enumeration value="Asian"/>
        <xs:enumeration value="AsianIndian"/>
        <xs:enumeration value="Austrian"/>
        <xs:enumeration value="Belgian"/>
        <xs:enumeration value="Brazilian"/>

```

```
<xs:enumeration value="British"/>
<xs:enumeration value="Cambodian"/>
<xs:enumeration value="Canadian"/>
<xs:enumeration value="Chinese"/>
<xs:enumeration value="Colombian"/>
<xs:enumeration value="Croatian"/>
<xs:enumeration value="Cuban"/>
<xs:enumeration value="Czech"/>
<xs:enumeration value="Czechoslovakian"/>
<xs:enumeration value="Danish"/>
<xs:enumeration value="Dominican"/>
<xs:enumeration value="Dutch"/>
<xs:enumeration value="Ecuadorian"/>
<xs:enumeration value="Egyptian"/>
<xs:enumeration value="English"/>
<xs:enumeration value="European"/>
<xs:enumeration value="Filipino"/>
<xs:enumeration value="Finnish"/>
<xs:enumeration value="French"/>
<xs:enumeration value="FrenchCanadian"/>
<xs:enumeration value="German"/>
<xs:enumeration value="Greek"/>
<xs:enumeration value="Guatemalan"/>
<xs:enumeration value="Guyanese"/>
<xs:enumeration value="Haitian"/>
<xs:enumeration value="Hawaiian"/>
<xs:enumeration value="Hispanic"/>
<xs:enumeration value="Hmong"/>
<xs:enumeration value="Honduran"/>
<xs:enumeration value="Hungarian"/>
<xs:enumeration value="Iranian"/>
<xs:enumeration value="Irish"/>
<xs:enumeration value="Israeli"/>
<xs:enumeration value="Italian"/>
<xs:enumeration value="Jamaican"/>
<xs:enumeration value="Japanese"/>
<xs:enumeration value="Korean"/>
<xs:enumeration value="Laotian"/>
<xs:enumeration value="LatinAmerican"/>
<xs:enumeration value="Lebanese"/>
<xs:enumeration value="Lithuanian"/>
<xs:enumeration value="Mexican"/>
<xs:enumeration value="Nicaraguan"/>
<xs:enumeration value="Nigerian"/>
<xs:enumeration value="NorthernEuropean"/>
<xs:enumeration value="Norwegian"/>
<xs:enumeration value="Pakistani"/>
<xs:enumeration value="Panamanian"/>
<xs:enumeration value="Peruvian"/>
<xs:enumeration value="Polish"/>
<xs:enumeration value="Portuguese"/>
<xs:enumeration value="PuertoRican"/>
<xs:enumeration value="Romanian"/>
<xs:enumeration value="Russian"/>
<xs:enumeration value="Salvadoran"/>
```

```
<xs:enumeration value="Scandinavian"/>
<xs:enumeration value="Scotch-Irish"/>
<xs:enumeration value="Scottish"/>
<xs:enumeration value="Serbian"/>
<xs:enumeration value="Slavic"/>
<xs:enumeration value="Slovak"/>
<xs:enumeration value="Slovene"/>
<xs:enumeration value="Spaniard"/>
<xs:enumeration value="Spanish"/>
<xs:enumeration value="Swedish"/>
<xs:enumeration value="Swiss"/>
<xs:enumeration value="Syrian"/>
<xs:enumeration value="Taiwanese"/>
<xs:enumeration value="Thai"/>
<xs:enumeration value="TrinidadianAndTobagonian"/>
<xs:enumeration value="Turkish"/>
<xs:enumeration value="Ukrainian"/>
<xs:enumeration value="Vietnamese"/>
<xs:enumeration value="Welsh"/>
<xs:enumeration value="WestIndian"/>
<xs:enumeration value="WesternEuropean"/>
<xs:enumeration value="Yugoslavian"/>
<xs:enumeration value="Apache"/>
<xs:enumeration value="Blackfeet"/>
<xs:enumeration value="Cherokee"/>
<xs:enumeration value="Cheyenne"/>
<xs:enumeration value="Chickasaw"/>
<xs:enumeration value="Chippewa"/>
<xs:enumeration value="Choctaw"/>
<xs:enumeration value="Colville"/>
<xs:enumeration value="Comanche"/>
<xs:enumeration value="Cree"/>
<xs:enumeration value="Creek"/>
<xs:enumeration value="Crow"/>
<xs:enumeration value="Delaware"/>
<xs:enumeration value="Houma"/>
<xs:enumeration value="Iroquois"/>
<xs:enumeration value="Kiowa"/>
<xs:enumeration value="Latin American Indian"/>
<xs:enumeration value="Lumbee"/>
<xs:enumeration value="Menominee"/>
<xs:enumeration value="Navajo"/>
<xs:enumeration value="Osage"/>
<xs:enumeration value="Ottawa"/>
<xs:enumeration value="Paiute"/>
<xs:enumeration value="Pima"/>
<xs:enumeration value="Potawatomi"/>
<xs:enumeration value="Pueblo"/>
<xs:enumeration value="Puget Sound Salish"/>
<xs:enumeration value="Seminole"/>
<xs:enumeration value="Shoshone"/>
<xs:enumeration value="Sioux"/>
<xs:enumeration value="Tohono O' odham"/>
<xs:enumeration value="Ute"/>
<xs:enumeration value="Yakama"/>
```

```
<xs:enumeration value="Yaqui"/>
<xs:enumeration value="Yuman"/>
<xs:enumeration value="Alaska Athabascan"/>
<xs:enumeration value="Aleut"/>
<xs:enumeration value="Eskimo"/>
<xs:enumeration value="Tlingit-Haida"/>
<xs:enumeration value="Other"/>
</xs:restriction>
</xs:simpleType>
</xs:schema>
```