

Kansas eCitation – Inquiry Service Design

Detailed Design Document

Version 1.0.1

Table of Contents

1	Intent.....	3
2	Overview	3
3	Project Vision/Scope	3
3.1	Vision.....	3
3.2	Scope of eCitation Inquiry Service	3
4	Inquiry Service Specification	4
4.1	Inquiry Service Business Components	5
4.1.1	PerformStructuredSearch	6
4.1.2	PerformTextSearch	6
4.2	Search Results NIEM XML	7
4.3	Indexed Citation NIEM XML fields	7
4.3.1	Subject Person Indexing.....	7
4.3.2	Citation Batch Indexing.....	7
4.3.3	Citation Indexing	8
4.3.4	Officer Person & Agency Indexing	8
4.3.5	Vehicle Indexing.....	8
4.3.6	Offense Indexing.....	9
5	Appendix A – Search Web Service Definition	9
5.1	WSDL.....	9
5.2	Additional WSDL	11
5.3	XSD	13
6	Appendix B – References	13
7	Appendix C – Definitions.....	14

1 Intent

This document will define in detail the design of the eCitation Inquiry Service that will be used in the eCitation system.

This document is intended to serve as the technical documentation of the Kansas eCitation Inquiry Service.

2 Overview

The Traffic Records Coordinating Council (TRCC) commissioned a Strategic Plan for the development and implementation of a statewide electronic traffic citation (eCitation) system, with a central traffic citation information repository (central repository) accessible by state, local, and federal agencies, and the public. This eCitation system is an integral part of the statewide Traffic Records System (TRS) program initiated in 2005 and will integrate with the Kansas Criminal Justice Information System (KCJIS). The TRS will be a virtual data warehouse that will provide state and local agencies with the ability to efficiently access traffic data to increase the safety of the motoring public. It will bring together information that is currently housed in separate, isolated repositories at KDOT, KHP, KDOR, KBI, KDHE, KBEMS and other agencies.

As a vital component of the TRS system, the eCitation project has been initiated with the goal of implementing a statewide eCitation system through which traffic citation data can be collected, analyzed, and distributed accurately, quickly, and cost effectively for the benefit of the public and state, local, and federal agencies.

The eCitation Inquiry Service will facilitate the searching and retrieval of Citation records from the eCitation Repository using a compatible [Logical Entity Exchange Specification \(LEXS\) 3.1](#) solution. The result of the search will provide a NIEM conformant XML. The format of the search request and results are outlined in the Inquiry Service Specification section of this document. This interface will adhere to all security constraints outlined in the KS E-Cite Security Design documentation.

The eCitation Inquiry Service will be a collection of .NET classes implementing the eCitation search functionality through web services. Once valid search metadata has been submitted to the web service, the service will perform the specified index search and return the results in formatted NIEM XML.

The system will maintain an audit log in conformance with the documented KCJIS security policies.

3 Project Vision/Scope

3.1 Vision

The goal of the eCitation Inquiry Service is to provide an interface to external systems for querying and retrieving Citation records from the eCitation Repository in the form of a NIEM conformant XML.

3.2 Scope of eCitation Inquiry Service

The scope of this document is limited to the details of the eCitation Inquiry Service and does not include any details pertaining to other components utilized within the eCitation System. Thus, if some general

components of eCitation (e.g. Core, Utilities) are mentioned in this document, only the details significant to the search service are discussed.

4 Inquiry Service Specification

The eCitation Inquiry Service will provide a standard interface for retrieving Citation information from the eCitation Repository in the format of a NIEM conformant XML. The InquiryService will be based on the NIEM LEXS version 3.1 standard XML interface because this is an industry standard, general purpose interface for searching upon system data of many forms. The LEXS interface allows for both simple text search and detailed field-by-field search. The eCitation Inquiry Service Design will support both search mechanisms, but it is expected that field-by-field search mechanism will be the most practical for integrating with existing search screens which accept search parameters this way.

The LEXS interface provides for searching information for the following entity types that are relevant to eCitation:

- Activity
- Document
- Location *
- Organization
- Person
- Vehicle

* Location based searching will not be included in the eCitation Inquiry service at this time. It is intended to be provided as an enhanced search capability in the future.

The primary interface for the eCitation Inquiry Service will be a Global Reference Architecture (GRA) based Web service that accepts NIEM LEXS version 3.1 conformant XML. The Search/Retrieve Service is named the LEXSSearchService and has the following method names:

DoTextSearch – This method accepts a `lexssr:doTextSearchRequest` message, parses the request, submits query to the eCitation Repository and returns a `lexssr:doSearchResponse` message with the results of the search.

DoStructuredSearch – This method accepts a `lexssr:doStructuredSearchRequest` message, parses the request, submits query to the eCitation Repository and returns a `lexssr:doSearchResponse` message with the results of the search.

GetCapabilities – This method accepts a `lexssr:getCapabilitiesRequest` message, and returns a `lexssr:getCapabilitiesResponse` with the capabilities available from this service in order to properly support the LEXS Search/Retrieve interface.

This Web service interface is depicted in Figure 7, LEXS Search Service Components.

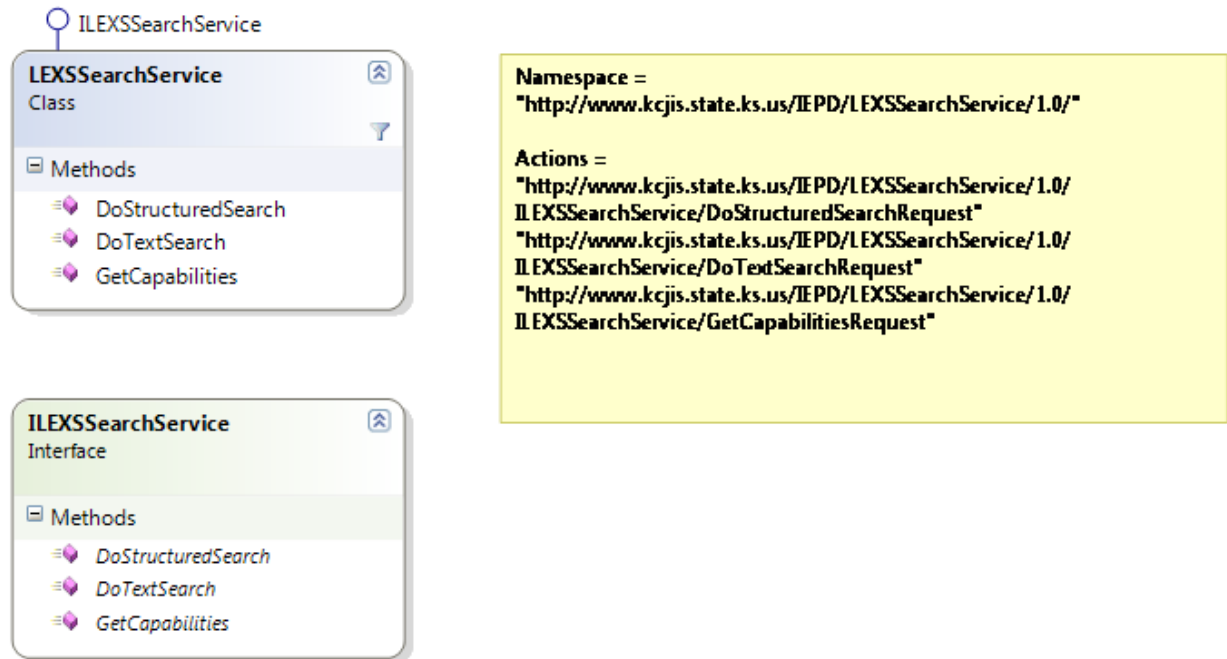


Figure 1. LEXS Search Service Components

The LEXS Search Service Web Service Description technical documentation can be found in Appendix A – Search Web Service Definition. The LEXS Search Service performs the search function by parsing the LEXS XML into common entity objects that act as Data Transfer Objects (DTO) and makes use of the SearchProvider interface and the SearchProviderFactory to locate a business component that can perform the appropriate business logic for the publish function. These business components will be described in the next section.

4.1 Inquiry Service Business Components

The primary software code component that provides access to the business logic for the eCitation Inquiry Service is through the eCitation Search Provider. This component exposes the ISearchProvider interface for any system that would need this functionality. The service obtains a reference to the required Search Provider using the SearchProviderFactory. The eCitation Search Provider makes use of data access, entity and other common components to perform the publish function. These components perform the following steps in the Search/Retrieve Sub-system:

- Parse request from LEXS XML into code components
- Translate request into database queries
- Perform database bulk keyword query and accept database response
- Perform additional filtering within the database and/or within the application server
- Consolidate and format response
- Translate response into LEXS XML and return to requestor

The eCitation Search Provider provides the implementation for following ISearchProvider interface methods:

- PerformStructuredSearch
- PerformTextSearch

The details of these provided web service interfaces and entity types are respectively described in detail below.

4.1.1 PerformStructuredSearch

The PerformStructuredSearch Search Provider method provides a search interface which accepts the requested search operation and the entity type with which to base the search operation on.

SearchResults PerformStructuredSearch(SearchOperation operation, SearchEntityType entityType)

The response to this web service method is the structured Citation search results which are provided to clients in a NIEM conformant XML format.

SearchOperation

The SearchOperation parameter provides the list of specific fields to search and the query parameter with which the field is to be searched.

SearchEntityType

The SearchEntityType parameter provides the relation of the search to the target fields in the Indexed NIEM XML fields available in the eCitation Repository and described in detail in Section 4.3 below.

4.1.2 PerformTextSearch

The PerformTextSearch Search Provider method provides a search interface which accepts a query for the search and the entity type with which to base the provided query search on.

SearchResults PerformTextSearch(string searchQuery, SearchEntityType type)

The response to this web service method is the structured eCitation search results which are provided to clients in a NIEM conformant XML format.

searchQuery

The searchQuery provides a text search from the provided query based on the given entity type.

SearchEntityType

See Section 4.1

4.2 Search Results NIEM XML

The search service will return NIEM LEXS version 3.1 conformant XML for all citation records in the repository that matches the submitted search request. Specifically, a lexssr:doSearchResponse message will be returned. This message format will include a summary of the matching entities as well as the complete NIEM eCitation XML. Due to the fact that the search results will contain the full details of a citation, in XML form, performing a targeted search will yield the same result as fetching a single record detail.

4.3 Indexed Citation NIEM XML fields

The following fields are provided indexes in the eCitation Repository to facilitate search requests on any of the following fields or combination of fields:

4.3.1 Subject Person Indexing

Index field	NIEM XML XPath
	./nc: PersonName/[starts-with(@ns1:id, 'CitationSubjectID')
Person First Name	./nc:PersonName/nc:PersonGivenName
Person Last Name	./nc:PersonName/nc:PersonSurName
Person Middle Name	./nc:PersonName/nc:PersonMiddleName
Person Name Suffix	./nc:PersonName/nc:PersonNameSuffixText
Person Driver License Number	./j:PersonAugmentation/nc:DriverLicense/nc:DriverLicenseIdentification/nc:IdentificationID
Person Driver License State	./j:PersonAugmentation/nc:DriverLicense/nc:DriverLicenseIdentification/j:DrivingJurisdictionAuthorityNCICLSTACode
Person Social Security Number	./nc:PersonSSNIdentification/nc:IdentificationID
Person Date of Birth	./nc:PersonBirthDate/nc:Date
Person Ethnicity	./nc:PersonEthnicityCode
Person Race	./nc:PersonRaceCode
Person Hair Color	./nc:PersonHairColorCode
Person Eye Color	./nc:PersonEyeColorCode
Person Gender	./nc:PersonSexCode
Person Height	./nc:PersonHeightDescriptionText
Person Weight	./nc:PersonWeightDescriptionText

4.3.2 Citation Batch Indexing

Index field	NIEM XML XPath
Batch Receipt Date	[Repository Index field]
Batch Quantity (calculated)	[Repository Index field]

4.3.3 Citation Indexing

Index field	NIEM XML XPath
Citation Number	./nc:ActivityIdentification[nc:IdentificationCategoryText='CitationNumber']/nc:IdentificationID
Issuance Date/Time	./nc:ActivityDate/nc:DateTime
Citation Status	./nc:ActivityStatus/nc:StatusText
Citation Version Number	./kstrs-ns:CitationVersionNumber
Court Name	./kstrs-ns:CitationCourtAppearance/j:CourtAppearanceCourt/j:CourtName

4.3.4 Officer Person & Agency Indexing

Index field	NIEM XML XPath
	./nc:PersonName/[starts-with(@ns1:id, 'CitationIssuingOfficialID')]
Person First Name	./nc:PersonName/nc:PersonGivenName
Person Last Name	./nc:PersonName/nc:PersonSurName
Person Middle Name	./nc:PersonName/nc:PersonMiddleName
Person Name Suffix	./nc:PersonName/nc:PersonNameSuffixText
Badge Number	./j:CitationIssuingOfficial/j:EnforcementOfficialBadgeIdentification/nc:IdentificationID
Agency ORI	./kstrs-ns:CitationAgency/j:OrganizationAugmentation/j:OrganizationORIdentification/nc:IdentificationID
Officer Notes	TBD

4.3.5 Vehicle Indexing

Index field	NIEM XML XPath
Plate Number	./nc:ConveyanceRegistrationPlateIdentification/nc:IdentificationID
Plate State	./nc:ConveyanceRegistrationPlateIdentification/j:IdentificationJurisdictionNCICLISCode
Plate County	./nc:ConveyanceRegistrationPlateIdentification/nc:IdentificationJurisdictionText
VIN	./nc:VehicleIdentification/nc:IdentificationID
Vehicle Make	./nc:VehicleMakeCode
Vehicle Model	./nc:VehicleModelCode
Vehicle Year	./nc:ItemFirstSoldYearDate
Vehicle Type (Body Style)	./kstrs-ns:CitationVehicle/j:VehicleBodyCategoryCode
Vehicle Color	./nc:VehicleColorPrimaryCode
Vehicle DOT Number	./kstrs-ns:CitationVehicle/j:CommercialCarrierUSDOTNumber

4.3.6 Offense Indexing

Index field	NIEM XML XPath
Offense Date/Time	./kstrs-ns:CitationViolation/nc:ActivityDate/nc:DateTime

5 Appendix A – Search Web Service Definition

5.1 WSDL

```

<wsdl:definitions name="LEXSSearchService"
targetNamespace="http://tempuri.org/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-utility-1.0.xsd"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:tns="http://tempuri.org/"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"

xmlns:i0="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/"

xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
  xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
  xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
  xmlns:wsa10="http://www.w3.org/2005/08/addressing"
  xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
  xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata">
  <wsdl:import
namespace="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/"
  location="./LEXSSearchService2.wsdl"/>
  <wsdl:types/>
  <wsdl:binding name="BasicHttpBinding_ILEXSSearchService"
type="i0:ILEXSSearchService">
    <soap:binding
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="DoStructuredSearch">
      <soap:operation
soapAction="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/IL
EXSSearchService/DoStructuredSearchRequest"

```

```

        style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DoTextSearch">
    <soap:operation

soapAction="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/IL
EXSSearchService/DoTextSearchRequest"
        style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetCapabilities">
    <soap:operation

soapAction="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/IL
EXSSearchService/GetCapabilitiesRequest"
        style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="LEXSSearchService">
    <wsdl:port name="BasicHttpBinding_ILEXSSearchService"
        binding="tns:BasicHttpBinding_ILEXSSearchService">
        <soap:address
location="http://servername/LEXSSearchService.svc"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

5.2 Additional WSDL

```

<wsdl:definitions
targetNamespace="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1
.0/"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-utility-1.0.xsd"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"

xmlns:tns="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"

xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
    xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
    xmlns:wsa10="http://www.w3.org/2005/08/addressing"
    xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
    xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata">
    <wsdl:types>
        <xsd:schema

targetNamespace="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1
.0/Imports">
            <xsd:import schemaLocation="./LEXSSearchService.xsd"
                namespace="http://schemas.microsoft.com/Message"/>
        </xsd:schema>
    </wsdl:types>
    <wsdl:message
name="ILEXSSearchService_DoStructuredSearch_InputMessage">
        <wsdl:part name="lexs" type="q1:MessageBody"
xmlns:q1="http://schemas.microsoft.com/Message"
        />
    </wsdl:message>
    <wsdl:message
name="ILEXSSearchService_DoStructuredSearch_OutputMessage">
        <wsdl:part name="DoStructuredSearchResult"
type="q2:MessageBody"
            xmlns:q2="http://schemas.microsoft.com/Message"/>
    </wsdl:message>

```

```

    <wsdl:message name="ILEXSSearchService_DoTextSearch_InputMessage">
      <wsdl:part name="lexs" type="q3:MessageBody"
xmlns:q3="http://schemas.microsoft.com/Message"
      />
    </wsdl:message>
    <wsdl:message
name="ILEXSSearchService_DoTextSearch_OutputMessage">
      <wsdl:part name="DoTextSearchResult" type="q4:MessageBody"
xmlns:q4="http://schemas.microsoft.com/Message"/>
    </wsdl:message>
    <wsdl:message
name="ILEXSSearchService_GetCapabilities_InputMessage">
      <wsdl:part name="lexs" type="q5:MessageBody"
xmlns:q5="http://schemas.microsoft.com/Message"
      />
    </wsdl:message>
    <wsdl:message
name="ILEXSSearchService_GetCapabilities_OutputMessage">
      <wsdl:part name="GetCapabilitiesResult" type="q6:MessageBody"
xmlns:q6="http://schemas.microsoft.com/Message"/>
    </wsdl:message>
    <wsdl:portType name="ILEXSSearchService">
      <wsdl:operation name="DoStructuredSearch">
        <wsdl:input

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/DoStructuredSearchRequest"

message="tns:ILEXSSearchService_DoStructuredSearch_InputMessage"/>
        <wsdl:output

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/DoStructuredSearchResponse"

message="tns:ILEXSSearchService_DoStructuredSearch_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="DoTextSearch">
          <wsdl:input

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/DoTextSearchRequest"

message="tns:ILEXSSearchService_DoTextSearch_InputMessage"/>

```

```

<wsdl:output

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/DoTextSearchResponse"

message="tns:ILEXSSearchService_DoTextSearch_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="GetCapabilities">
    <wsdl:input

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/GetCapabilitiesRequest"

message="tns:ILEXSSearchService_GetCapabilities_InputMessage"/>
  <wsdl:output

wsaw:Action="http://www.kcjis.state.ks.us/IEPD/LEXSSearchService/1.0/I
LEXSSearchService/GetCapabilitiesResponse"

message="tns:ILEXSSearchService_GetCapabilities_OutputMessage"/>
  </wsdl:operation>
</wsdl:portType>
</wsdl:definitions>

```

5.3 XSD

```

<xs:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/Message"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://schemas.microsoft.com/Message">
  <xs:complexType name="MessageBody">
    <xs:sequence>
      <xs:any minOccurs="0" maxOccurs="unbounded"
namespace="##any"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

6 Appendix B – References

LEXS IEPD	http://www.lexs.gov/sites/all/lexs/docs/LEXS3.1.4_2009-02-06.zip
-----------	---

KS E-Cite Security Design Document	https://projects.analysts.com/kansasecitation/TechnicalDocuments/Security%20Model/KS%20E-Cite%20Security%20Detailed%20Design%20v1.0.0.docx
Kansas eCitation IEPD	http://www.kstrs.org/Traffic%20Reports/Forms/AllItems.aspx?RootFolder=%2FTraffic%20Reports%2FXML%20Schemas&FolderCTID=&View=%7BF8681317-7A7B-4FB4-B062-DDF68A88C345%7D
Kansas eCitation High-level Design Document	https://projects.analysts.com/kansasecitation/internal/Shared%20Documents/Phase%201B%20Deliverables/KS%20E-Cite%20High-level%20Design%20V1.2.docx
Kansas eCitation Data Requirements	http://www.kstrs.org/Traffic%20Reports/eCitation/eCitation%20Data%20Requirements-v1.1.docx

7 Appendix C – Definitions

Acronym/Term	Definition
NIEM	National Information Exchange Model
XML	eXtensible Markup Language
IEPD	Information Exchange Package Documentation
WSDL	Web Services Description Language
LEXS	Logical Entity Exchange Specification
DTO	Data Transfer Object – A code component for transferring data between other components.
XSD	XML Schema Definition