



# **CCMS Court Policy File User Guide**

---

APRIL 2012



ADMINISTRATIVE OFFICE  
OF THE COURTS

---

INFORMATION SERVICES DIVISION

Judicial Council of California  
Administrative Office of the Courts  
Information Services Division  
455 Golden Gate Avenue  
San Francisco, California 94102-3688  
415-865-4200  
*www.courts.ca.gov*

Copyright © 2012 by Judicial Council of California/Administrative Office of the Courts.  
All rights reserved.

Except as permitted under the Copyright Act of 1976 and as otherwise expressly provided herein, no part of this publication may be reproduced in any form or by any means, electronic or mechanical, including the use of information storage and retrieval systems, without permission in writing from the copyright holder. Permission is hereby granted to nonprofit institutions to reproduce and distribute this publication for educational purposes if the copies credit the copyright holder.

## Revision History

Version	Date	Description of Changes	Author
1.0.0	January 2012	First Release	Saravanan Mani
1.0.1	April 2012	Updated with additional information based on Intersys feedback	Saravanan Mani

## Reference Documents

Version	Date	Document Title	Author
1.0	July 2010	<a href="#">V3 E-Filing Court Policy File Overview</a>	AOC
1.0.0	November 2011	<a href="#">CCMS Court Policy File Schema</a>	Deloitte Consulting



# Table of Contents

Revision History .....	iii
Reference Documents .....	iii
List of Figures .....	vi
List of Tables .....	vi
<b>1.0 Executive Summary .....</b>	<b>1</b>
1.1 Purpose.....	1
1.2 Audience .....	1
1.3 Scope.....	1
1.4 Out of Scope.....	1
1.5 Assumptions.....	1
<b>2.0 Court Policy File Overview .....</b>	<b>2</b>
2.1 Benefits of Court Policy to Statewide Business Services .....	2
2.1.1 Quality Assurance.....	2
2.1.2 Standardized Publication of Court Information.....	2
2.1.3 Reduction or Elimination of Court Software Customization.....	3
2.1.4 Statewide Scalability .....	3
2.1.5 Satisfaction of California’s CRC 2.254(b) Publication of Electronic Filing Requirements.....	3
2.2 Court Policy File Scope .....	3
2.3 Court Policy File Namespaces .....	3
<b>3.0 Content of a Court Policy File .....</b>	<b>5</b>
3.1 Court Keys .....	5
3.2 Key .....	7
3.3 Dx.....	7
3.4 Version .....	8
3.5 Publication Date and Time.....	8
3.6 Effective Date and Time .....	8
3.7 Court Details .....	8
3.8 Court Divisions .....	8
3.9 Code Tables.....	9
3.10 Minute Tables.....	10
3.11 Fees .....	11
<b>4.0 Data Exchange Mapping Sheets .....</b>	<b>13</b>
<b>5.0 Court Policy File XML Mapping .....</b>	<b>14</b>
5.1 XML Mapping File Structure.....	14
5.2 CPF Mapping File Example.....	15
5.3 CPF XML Mapping File Usage .....	15
<b>6.0 Accessing Court Policy Files .....</b>	<b>17</b>
<b>Appendix A. Acronyms .....</b>	<b>19</b>
<b>Appendix B. Deprecated Elements and Attributes.....</b>	<b>21</b>
<b>Appendix C. Court Policy File Class Model .....</b>	<b>23</b>

## List of Figures

Figure 1. Relationship of CourtKeyValue, CaseType, CaseCategory and CourtLocation ..... 6  
Figure 2. Snippet from a Mapping Sheet Showing "Configurable Indicator" Column ..... 13  
Figure 3. Sample Snippet from a CPF XML Mapping File..... 15  
Figure 4. Court Policy File Class Model..... 23

## List of Tables

Table 1. Court Policy File Namespaces ..... 4  
Table 2. List of Acronyms ..... 19

## **1.0 Executive Summary**

### **1.1 Purpose**

This document provides an overview of Court Policy Files (CPFs) and artifacts that support implementation of Court Policy Files. The goal of this document is to provide adequate information so the audience is able to comprehend and use Court Policy Files in conjunction with data exchange specifications.

### **1.2 Audience**

This document is targeted toward integration partners who will be implementing data exchanges with the California Court Case Management System (CCMS) that require usage of Court Policy Files.

### **1.3 Scope**

This document focuses on the format and content of Court Policy Files and related artifacts, including Court Policy XML mapping files and data exchange mapping spreadsheets.

### **1.4 Out of Scope**

This document is not concerned with the design of the CCMS system or the steps required for CCMS to produce Court Policy Files; nor does the document specify how partners must use technology to interpret court policy details.

### **1.5 Assumptions**

This document assumes the following:

- The reader has a basic understanding of court terminology such as “court key”, “court division”, “court configurable codes”, “minute codes”, etc.
- The reader is familiar the CCMS Data Exchange environment
- The reader is familiar with data exchange packages and the artifacts contained in those packages
- The reader has a basic understanding of XML Schema<sup>1</sup>
- The reader has a basic understanding of XPath<sup>2</sup>

---

<sup>1</sup> <http://www.w3.org/TR/xmlschema-0/>

<sup>2</sup> <http://www.w3.org/TR/xpath/>

## 2.0 Court Policy File Overview

CCMS allows courts to configure certain data elements with values applicable to their individual court and their divisions. Courts may also have their own “policies,” such as fee structures. In the context of this document, a Court Policy File is a technical configuration file representing this information in a way that computer programs can understand without human intervention. Electronic filing service providers (EFSPs) and justice partners (collectively called integration partners) programmatically<sup>3</sup> consume information in the CPF and use the information to configure their systems.

### 2.1 Benefits of Court Policy to Statewide Business Services

CPF provides a number of benefits, including the following:

- Quality Assurance
- Standardized Publication of Court Information
- Reduction or elimination of Court Software Customization
- Scalability
- Satisfaction of California’s CRC 2.254(b) Publication of Electronic Filing Requirements

The following sections explain these benefits.

#### 2.1.1 Quality Assurance

CPF data is included in the integration partner’s electronic submission to the court. This feature ensures that electronic transactions submitted from integration partners to the court have accurate information. For example, electronic filing applications use document types in a CPF as a pick list for filers. Only those document types in the CPF are available from the pick list.

Furthermore, different lists of document types can be associated with different case categories and case types. Because the document type list originates from the court and is not changed by the integration partner—unless there is a defect in the filing application—there is no opportunity for the document type value to be incorrect.

Increasing the accuracy of the electronic submission reduces the work of the court clerk to process the filing.

#### 2.1.2 Standardized Publication of Court Information

A CPF technical specification provides a standardized way for a court to publish information and for integration partners to consume information.

---

<sup>3</sup> The programmatic consumption of Court Policy information is what distinguishes a Court Policy from, for example, a website. The court’s website may include certain information that is also contained in a court policy. However, the website format does not allow an integration partner to easily consume the data and import it into their system.

### **2.1.3 Reduction or Elimination of Court Software Customization**

In technical implementations, court-specific customizations are a spoiler for Courts as well as integration partners. System integration work is complex and expensive; it delays projects; and economies of scale are elusive. Standard CPF formats and publication procedures allow integration partners to write software one time and then configure it for use in multiple courts, reducing or eliminating the need for software customization.

### **2.1.4 Statewide Scalability**

Electronic filing and other types of data submissions are possible without CPFs<sup>4</sup> and, indeed, occur regularly in projects around the country. Such implementations, however, are less scalable because they are more difficult to repeat. A high-quality, well-defined, and robust CPF specification that increases quality, provides a standard, and reduces court customization also allows implementations to multiply and scale more easily and more quickly.

Imagine, for example, writing code for, and consuming, different formats of the same type of information for each implementation of each court or court subdivision. This redundant effort would make each project unique and would therefore increase the time and cost to implement multiple projects that are otherwise the same or similar.

### **2.1.5 Satisfaction of California's CRC 2.254(b) Publication of Electronic Filing Requirements**

CPF also provides a means for a court to partially satisfy the publishing requirement of California Rules of Court (CRC) 2.254(b). CRC 2.254(b), which states:

“Each court that permits electronic filing must publish, in both electronic and print formats, the court’s electronic filing requirements.”

## **2.2 Court Policy File Scope**

Courts have the ability to produce CPFs of two different scopes: (1) a CPF that contains policy information for all data exchanges implemented by the court; and (2) a CPF that contains policy information only for a specific data exchange.

Consumers of a CPF are able to determine the scope of a given CPF based on information contained in the actual CPF. If a CPF is specific to a single data exchange, the CPF will contain the name of the data exchange in the file. If a CPF does not contain the name of a data exchange, then the CPF contains policy information for all data exchanges.

## **2.3 Court Policy File Namespaces**

For the purpose of this document, this section assigns prefixes to namespaces used in the CPF. This document contains several references to XML elements and many of those references use

---

<sup>4</sup> Projects that do not have a standard Policy XML format often use spreadsheets to convey similar information. Spreadsheets are better than nothing, simply because the data is most often structured. However, spreadsheets do not provide precise content validation and have technical limitations. Data in spreadsheets must usually be “massaged” prior to import, which is time consuming and error prone and hinders full automation.

namespace prefixes. Use this table for additional clarification on namespace prefix usage throughout this document. Table 1 lists the CPF namespaces and their prefixes.

Table 1. Court Policy File Namespaces

Namespace Name	Namespace Prefix
http://policy.srv.courts-tc.ca.gov/Court/Policy/AssociatedPolicy/xsd	AssociatedPolicy
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Address/xsd	Address
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/CourtDetails/xsd	CourtDetails
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Fee/xsd	Fee
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Filter/xsd	Filter
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Key/xsd	Key
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Dx/xsd	Dx
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Organization/xsd	Organization
http://policy.srv.courts-tc.ca.gov/Court/Policy/CaseCategory/xsd	CaseCategoryKeys
http://policy.srv.courts-tc.ca.gov/Court/Policy/AllLocations/xsd	AllLocationKeys
http://policy.srv.courts-tc.ca.gov/Court/Policy/CaseType/xsd	CaseTypeKeys
http://policy.srv.courts-tc.ca.gov/Court/Policy/CodeTable/Code/xsd	Code
http://policy.srv.courts-tc.ca.gov/Court/Policy/CodeTable/xsd	CodeTable
http://policy.srv.courts-tc.ca.gov/Court/Policy/CourtDivision/xsd	CourtDivision
http://policy.srv.courts-tc.ca.gov/Court/Policy/Location/xsd	LocationKeys
http://policy.srv.courts-tc.ca.gov/Court/Policy/NonConfigTable/NonConfigCode/xsd	NonConfigCode
http://policy.srv.courts-tc.ca.gov/Court/Policy/NonConfigTable/xsd	NonConfigTable
http://policy.srv.courts-tc.ca.gov/Court/Policy/Parameter/xsd	Parameter
http://policy.srv.courts-tc.ca.gov/Court/Policy/BuildingBlocks/Primitives/Fee/xsd	n1
http://policy.srv.courts-tc.ca.gov/Court/Policy/DXCPFMapping/1.0.0/xsd	cpf

## 3.0 Content of a Court Policy File

This section provides an overview of the main sections of a CPF XML file. A CPF is an XML file divided into multiple sections where each section equates to a different piece of a court's data exchange policy. Appendix C provides an overview class diagram of CPF XML schema. The CPF XML is formally specified by an XML schema which is available at: [CCMS Court Policy File XML Schema](#).

Partners who are implementing exchanges that use CPFs should use this document as a guide but should consider the CPF XML schema as the authoritative representation of the content and structure of a CPF. The schema contains deprecated elements and attributes (elements and attributes of elements used in previous versions of CPF and are not supported in this version). Appendix B lists these deprecated elements and element attributes.

The root data element for all CPF XML files is `Policy`. This section also provides more details about each section of the CPF XML structure. The order of the sections below represents the order of the same sections in the actual CPF XML. Partners should refer to the CPF schema for an official understanding of the CPF XML structure beyond what is described in this section.

### 3.1 Court Keys

The Court Keys node is located in the CPF at `./Policy/CourtKeys`. CCMS uses Court Keys to assist with managing policies related to E-Filing Data Exchanges. Court Keys are a combination of court code, location, case category, and case type. This section is only applicable to E-Filing Data Exchanges so it may not appear in all CPFs.

**Note:** Court Keys contain `CaseCategoryKeys` and `Location Keys`. `CaseCategoryID` is included only when Court Keys contain `CaseCategoryKeys`.

Individual Court Keys in the CPF are represented as:

```
./Policy/CourtKeys/CaseCategoryKeys:CaseCategoryKeys/CaseTypeKeys:CaseTypeKeys/CaseTypeKeys:CaseTypeKeyID/CaseTypeKeys:CourtKeyValue.
```

Court Keys are grouped by Case Type, so several Court Keys (`CaseTypeKeys:CourtKeyValue`) can exist inside of `CaseTypeKeys:CaseTypeKeyID`.

The `CaseTypeKeys:CaseTypeKeyID` element contains an attribute called `ID` which uniquely identifies the group of Court Keys so they can be referenced through the CPF.

The Court Keys section of the CPF also contains a node called `CaseCategoryID` which is located at:

```
./Policy/CourtKeys/CaseCategoryKeys:CaseCategoryKeys/CaseCategoryKeys:CaseCategoryID.
```

This section of the XML groups Case Types together by Case Category.

CaseCategoryKeys:CaseCategoryID contains one or more CaseCategoryKeys:CaseTypeRef nodes which reference the ID attribute of a CaseTypeKeys:CaseTypeID to indicate that a given Case Type falls within a specific Case Category.

The CaseCategoryKeys:CaseCategoryID node also contains an attribute called ID which uniquely identifies the group of Case Types for reference throughout the CPF. Case Category Keys are then grouped as CKALL, which includes keys from all Case Categories, as shown in Figure 1 below:

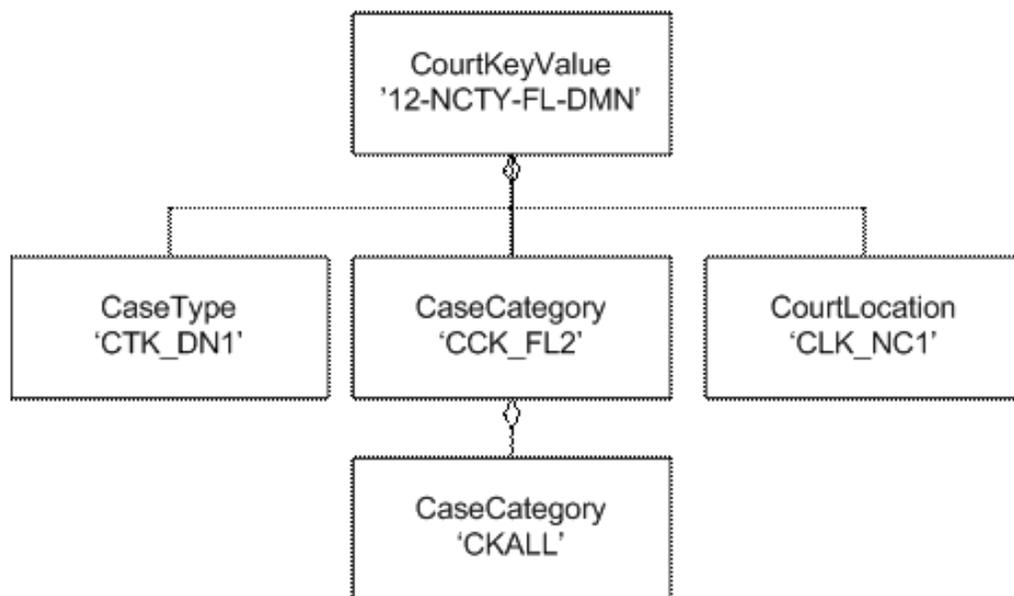


Figure 1. Relationship of CourtKeyValue, CaseType, CaseCategory and CourtLocation

**Sample CKALL snippet:**

```

<CaseCategoryKeys:CaseCategoryID ID="CKALL">
    <CaseCategoryKeys:CaseCategoryRef ref="CCK1"/>
    <CaseCategoryKeys:CaseCategoryRef ref="CCK2"/>
    <CaseCategoryKeys:CaseCategoryRef ref="CCK3"/>
</CaseCategoryKeys:CaseCategoryID>
  
```

**Note:** Users implementing CPF should understand that a CPF is designed to include all courts in a single file generated per DX, and court codes could belong to case types, case categories, and CKALL.

The Court Keys section of the CPF also contains a node called `AllLocationKeys:AllLocationKeys`.

This is a grouping of court keys based on court location:

`AllLocationKeys:AllLocationKeys` contains an element called `Location:LocationKeys` which contains one or multiple `Location:LocationKeyID`; each `Location:LocationKeyID` has an attribute called `ID` which is a unique identifier for the location.

**Note:** This ID is used as a reference by CPF in identifying `LocationKeys`.

Each `Location:LocationKeyID` contains one or multiple `Location:LocationKeyValue` elements, which refer to the court key for a court that exists in the given location .

### 3.2 Key

This node is located in the CPF at `./Policy/Key`. `Key` consists of several elements that collectively uniquely identify the CPF. The following list represents the elements that compose the `Key`, and the paths for these elements, relative to `./Policy/Key`:

- **Court Key** – `Key:CourtKey`, represents the key of the court that generated the CPF.
- **Year** – `Key:Year`, represents the year in which the CPF was generated.
- **Month** – `Key:Month`, represents the month in which the CPF was generated.
- **Day** – `Key:Day`, represents the day in which the CPF was generated.
- **Hour** – `Key:Hour`, represents the hour in which the CPF was generated.
- **Minute** – `Key:Minute`, represents the minute in which the CPF was generated.
- **Second** – `Key:Second`, represents the second in which the CPF was generated.
- **Organization Key** – `Key:OrganizationKey`
- **Application** – `Key:Application`
- **Version** – `Key:Version`
- **Suffix** – `Key:Suffix`

### 3.3 Dx

The `Dx` node is located in the CPF at `./Policy/Dx`. This section of the CPF XML only exists in the CPF and is generated for an individual data exchange. This section conveys the unique identifier for the data exchange as well as the official business name of the data exchange. The unique identifier of the data exchange (e.g., INI801) is represented as `./Policy/Dx/Dx:ID`. The business name of the data exchange (in this case, Receive Case Initiation Filing – Citations) is represented as `./Policy/Dx/Dx:Name`.

### 3.4 Version

The Version node is located in the CPF at `./Policy/Version`. This node contains two sub-nodes: `MajorVersion` and `MinorVersion`. Together these elements represent the specific version of the CPF.

### 3.5 Publication Date and Time

Publication date and time nodes are located in the CPF at `./Policy/PublicationDate` and `./Policy/PublicationHour`. Together these elements indicate the date and time that the court published the CPF.

### 3.6 Effective Date and Time

Effective date and time nodes are located in the CPF at `./Policy/EffectiveDate` and `./Policy/EffectiveHour`. Together these elements indicate the date and time that the information in the CPF is to become effective.

### 3.7 Court Details

The Court Details node is located in the CPF at `./Policy/CourtDetails`. This node contains details about the court which issues the specific CPF. The following list represents each element in this section and the path to the element within the section, relative to `CourtDetails`:

- **Organization Key** – `CourtDetails:OrganizationKey`
- **Court Name** --  
`CourtDetails:Court/Organization:Name/Organization:FullName`
- **A court can have multiple addresses, where each address is represented by:**  
`CourtDetails:Court/Organization:Addresses/Organization:Address`. Segments of each address are listed below, relative to `Organization:Address`:
  - *Building Name* – `Address:Line/Address:BuildingName`
  - *Street Number* – `Address:Line/Address:StreetNumber`
  - *Street Name* – `Address:Line/Address:StreetName`
  - *City* – `Address:City`
  - *State* – `Address:State`
  - *County* – `Address:County`
  - *Postal Code* – `Address:PostalCode`

### 3.8 Court Divisions

The Court Divisions node is located in the CPF at `./Policy/CourtDivisions`. This section represents information about one or more court divisions, where each division is represented as `./Policy/CourtDivisions/CourtDivision`. The following list represents each element within `CourtDivision` and the path to the element, relative to `CourtDivision`:

- **Organization Key** –  
CourtDivision:CourtDetails/CourtDetails:OrganizationKey
- **Court Name** –  
CourtDivision:CourtDetails/CourtDetails:Court/Organization:  
Name/Organization:FullName
- **State** – CourtDivision:CourtDetails/CourtDetails:State
- **County** – CourtDivision:CourtDetails/CourtDetails:County
- **Division** – CourtDivision:CourtDetails/CourtDetails:Division
- **Location** – CourtDivision:CourtDetails/CourtDetails:District,  
CourtDetails:District has an attribute called ref which points back to a  
Location:LocationKeyID from the AllLocations:AllLocationKeys  
section.
- **Court Code** – CourtDivision:CourtDetails/CourtDetails:CourtCode
- **District** – CourtDivision:CourtDetails/CourtDetails:District

### 3.9 Code Tables

The Code Tables node is located in the CPF at ./Policy/CodeTables. This section contains court code values for data exchange data elements that are court configurable. Each code table is represented as ./Policy/CourtDivisions/CodeTables/CodeTable. Each CodeTable element contains two attributes: CodifiedName which is the unique name of the code table and ID which is an identifier for the code table.

The following list represents each element within CodeTable and the path to the element, relative to CodeTable:

- **Code Table Name** – CodeTable:Name
- **Court Keys** – CodeTable:CourtKeys contains one or more elements called CodeTable:CaseCategoryRef or CodeTable:CaseTypeRef. Each CodeTable:CaseCategoryRef contains a ref attribute which points to a CaseCategoryKeys:CaseCategoryID node in the CourtKeys section of the XML.
- **Code Values** – CodeTable:Codes contains the actual code values for the code table, where each code value is represented as CodeTable:Codes/CodeTable:Code. Each CodeTable:Code contains two elements: Code:Name, the name of the code value, and Code:Value, the actual code literal or the actual value that would appear for an element in the data exchange.
- **Filter Name** –  
CodeTable:Codes/Code:Filters/Code:Filter/Filter:Name
- **Filter Value** –  
CodeTable:Codes/Code:Filters/Code:Filter/Filter:Value

This section of the CPF defines allowable values for court-configurable data elements in the data exchanges. Refer to section 4.0 and 5.0 of this document to learn more about the correlation between CPF code tables and data elements in the data exchanges.

### 3.10 Minute Tables

The Minute Tables node is located in the CPF at `./Policy/MinuteTables` and represents a collection of minute tables, where each minute table is represented as `./Policy/MinuteTables/MinuteTable`. This section is only applicable to data exchanges that use minute codes. This section of the CPF XML contains code values and parameter associations based on the case types related to the Minute Codes.

`MinuteTable` contains a structure called `MinuteTable:Codes`, which represents specific minute codes, where each code is represented as `MinuteTable:Codes/MinuteTable:Code`. Each `MinuteTable:Codes:Code` contains `MinuteTable:Codes:Code:CourtKeys`. `CourtKeys` contains one or more `MinuteCode:CaseTypeRef` elements. Each `MinuteCode:CaseTypeRef` contains an attribute called `ref` which points to a `CaseCategoryKeys:CaseCategoryID` in the CPF XML. This element indicates the Case Category related to the minute code.

Each `MinuteTable:Code` contains an element called `MinuteCode:Name`, which is the name of the minute code, and `MinuteCode:Alias`, which is an alias for the minute code. Each `MinuteTable` also contains one or more `MinuteCode:Parameter` elements which are located at

`MinuteTable:Codes/MinuteTable:Code/MinuteCode:Parameters/MinuteCode:Parameter`.

`MinuteCode:Parameter` represents details associated to each minute code. A parameter can be transactional or configurable. The following list represents the elements that compose the `MinuteCode:Parameter`, and the paths for these elements, relative to

`./Policy/Key/MinuteCode:Parameter` :

- **Sequence** – `n1:Sequence`, represents the sequence in which the parameter exists in the minute order code
- **Name** – `n1:Name`, represents the name of the parameter
- **ID** – `n1:ID`, represents the unique identifier of the parameter
- **Type** – `n1:Type`, represents the type of the parameter, this can be transactional or configurable

The `MinuteTables` section can also contain a node called `NonConfigTable` which represents the non-configurable code table associated to the minute codes. The name of each `NonConfigTable` is represented as `NonConfigTable/NonConfigTable:Name`. A `NonConfigTable` can be associated to one or more case categories using the `ref` attribute on the `NonConfigTable:CourtKeys/NonConfigTable:CaseCategoryRef` element. Each `NonConfigTable` contains a node called `NonConfigTable:Codes`, which in turn contains one or more nodes called `NonConfigTable:Code`. Each

`NonConfigTable:Code` contains the following, with path location relative to `NonConfigTable:Code`:

- **Name** – `NonConfigCode:Name`
- **Value** – `NonConfigCode:Value`
- **Filter Name** –  
`NonConfigCode:Filters/NonConfigCode:Filter/Filter:Name`
- **Filter Value** –  
`NonConfigCode:Filters/NonConfigCode:Filter/Filter:Value`

A minute code parameter (`MinuteCode:Parameter`) can be associated to a configurable or non-configurable code table. This association is made clear in the CPF.

If a minute code parameter is related to a configurable code table, the `n1:ID` element inside of `MinuteCode:Parameter` is equal to `Filter:Value` element inside of `CodeTable:Code`.

If a minute code parameter is related to a non-configurable code table, the `n1:ID` element inside of `MinuteCode:Parameter` is equal to `Filter:Value` element inside of `NonConfigCode:Filter`.

### 3.11 Fees

The Fees node is located in the CPF at `./Policy/Fees` and represents valid Fee information related to data exchanges, where each fee is represented as `./Policy/Fees/Fee`. Each `Fee` can contain one or more occurrences of `n1:CourtKeys/n1:CaseTypeRef`. `n1:CaseTypeRef` contains an attribute called `ref` which points to `CaseTypeKeys:CaseTypeKeyID` or `CaseCategory:CaseCategoryID` and ultimately indicates to which court key(s) the Fee applies.

The following are characteristics of a Fee, with paths relative to `./Policy/Fees/Fee`:

- **Name** – `Fee:Name`
- **Value** – `Fee:Value`
- **Cost** – `Fee:Cost`
- **Document Types** – `Fee:DocumentTypes`

Some data exchanges contain an element called “Fee Type”. “Fee Type” is a court-configurable data element, and the allowable values for this element are defined in the CPF by each occurrence of `./Policy/Fees/Fee/Fee:Name`.

The CPF details fees applicable to the Filing Names and Case Types associated with each exchange whether or not the fee type element itself is contained in the data exchange payload. This allows the justice partners not only to consume data exchanges with the fee type configurable element (i.e., understand the values/fee types in exchanges) but also to understand the fee types applicable for exchange without the fee type elements (i.e., determine the fee types applicable for collection data exchanges).



## 4.0 Data Exchange Mapping Sheets

All data exchange packages contain multiple spreadsheets called data exchange mapping sheets. These mapping sheets define and map the logical content models of messages involved in a specific data exchange. Each of these mapping sheets contains a column called “Configurable Indicator” which indicates elements as being configurable or non-configurable. If the element is a configurable element, this column equals “Y”; if the element is non-configurable, then this column equals “N”. If an element is flagged as configurable, it means policy information specific to that element is included in the CPF.

Figure 2 is a snippet from a mapping sheet showing the “Configurable Indicator” column:

Field Type in XML	XML Cardinality	Screen Specifications	Comment	Configurable Indicator
	1,1	Branch	None	N
INTEGER	1,1	V4-CAS-INI07 Initiate Citation Batch Information Section	None	N
INTEGER	1,1	V4-CAS-INI07 Initiate Citation Batch Information Section	The Batch ID in the request message is partner provided and CCMS-V4 generates its own Batch ID. The partner will get its own Batch ID as part of response message	N
	1,*	Branch	None	N
STRING	1,1	N/A	None	N
STRING	1,1	V4-CAS-INI07 Initiate Citation Citation Information Section	1. Configurable by Court	Y
STRING	1,1	V4-CAS-INI07 Initiate Citation Citation Information Section	1. Configurable by Court	Y
			1. Configurable by Court	

Figure 2. Snippet from a Mapping Sheet Showing “Configurable Indicator” Column

## 5.0 Court Policy File XML Mapping

A data exchange specification package for a given data exchange contains a CPF XML mapping file if that data exchange contains one or more court-configurable data elements. Each CPF XML mapping file formalizes the correlation between the configurable elements in the data exchange XML message and the proper code table in the CPF that defines the allowable values for that data element.

It should be noted that some data exchanges may only use court policies, i.e., fee structures, and yet have no court-configurable data element. For these data exchanges, the data exchange specification package will not contain a CPF XML mapping file. However, since it uses policy information, a CPF will still be generated for those data exchanges.

The CPF XML mapping file is included in each applicable data exchange package inside the “SchemaDefinitions/CAJUD/[DX Name]/V\_x\_y\_z/CPF” directory. If both the request and response messages of a given data exchange contain court configurable elements, then the “CPF” directory will contain two XML mapping files: one for the request and one for the response. The “CPF” directory also contains the XML schema for the CPF XML mapping file.

### 5.1 XML Mapping File Structure

The CPF mapping file is represented using a simple XML structure. Mapping between CPF and data exchange concepts is handled using XPath. Simply put, the CPF XML mapping is performed by grouping together an XPath specific to an element in a data exchange and an XPath specific to a code table in the CPF.

The root element for all CPF XML mapping files is `cpf:DX_CPFMapping`. Each CPF XML mapping file contains an element, directly beneath the root element, called `cpf:DXName`. This element specifies the uniquely identifying name of the data exchange, for example, INI801.

Each CPF XML mapping file contains one or more `cpf:NamespaceDefinition` structures which convey specific details about namespaces used in the CPF and the data exchange and ultimately in the XPaths contained in the mapping file:

`cpf:NamespaceDefinition` contains two elements: `cpf:NamespacePrefix` and `cpf:NamespaceName`. `cpf:NamespacePrefix` represents a namespace prefix that is assigned to a specific namespace which is represented as `cpf:NamespaceName`.

Each CPF XML also contains one or more `cpf:Mapping` structures. Each `cpf:Mapping` structure defines the actual mapping between a data exchange element and a CPF code table:

`cpf:Mapping` contains two elements: `cpf:DX_Path` and `cpf:CPF_Path`. `cpf:DX_Path` contains an XPath to a court-configurable element in the data exchange XML message. `cpf:CPF_Path` contains an XPath to a code table in the CPF that defines allowable values for the court-configurable element.

## 5.2 CPF Mapping File Example

Figure 3 below shows a sample snippet from a CPF XML mapping file:

```
<cpf:DX_CPFMapping xmlns:cpf="http://policy.srv.courts-
tc.ca.gov/Court/Policy/DXCPFMapping/1.0.0/xsd"
  <cpf:DXName>INI801</cpf:DXName>
  <cpf:NamespaceDefinition>
    <cpf:NamespacePrefix>cajud</cpf:NamespacePrefix>
    <cpf:NamespaceName>https://isb.srv.courts-
      tc.ca.gov/portal/cajud/schema/ini801req/2.0.0/xsd</cpf:Names
      paceName>
  </cpf:NamespaceDefinition>
  <cpf:NamespaceDefinition>
    <cpf:NamespacePrefix>Policy</cpf:NamespacePrefix>
    <cpf:NamespaceName>http://policy.srv.courts-
      tc.ca.gov/Court/Policy/1.0.0/xsd</cpf:NamespaceName>
  </cpf:NamespaceDefinition>
  <cpf:NamespaceDefinition>
    <cpf:NamespacePrefix>CodeTable</cpf:NamespacePrefix>
    <cpf:NamespaceName>http://policy.srv.courts-
      tc.ca.gov/Court/Policy/CodeTable/1.0.0/xsd</cpf:NamespaceNa
      me>
  </cpf:NamespaceDefinition>
  <cpf:Mapping>
    <cpf:DX_XPath>//cajud:ProductExchangePackage/cajud:ProductExchange
      Batch/cajud:BatchRecord/cajud:ProductRelatedObjects/cajud:ProductRelat
      edItems/cajud:Vehicle/cajud:ConveyanceColorPrimaryCode</cpf:DX_XPat
      h>
    <cpf:CPF_XPath>//Policy:Policy/Policy:CodeTables/Policy:CodeTable[Code
      Table:Name='VEHICLE_COLOR']</cpf:CPF_XPath>
  </cpf:Mapping>
</cpf:DX_CPFMapping>
```

Figure 3. Sample Snippet from a CPF XML Mapping File

This XML file simply states that the allowable values for a configurable element called “ConveyanceColorPrimaryCode” are specified in a code table called “VEHICLE\_COLOR” in the CPF XML file.

## 5.3 CPF XML Mapping File Usage

Partners should rely on the CPF XML mapping file as the authoritative mapping between configurable data elements in a data exchange and the list of allowable values in a CPF. The CPF XML mapping file is constructed using industry standards such as XML and XPath, so partners have the option of developing tooling around the CPF XML mapping to automatically draw correlations between a data element and its allowable values.



## **6.0 Accessing Court Policy Files**

Please contact your Court representative or AOC representative to learn how to access Court Policy Files.



## Appendix A. Acronyms

*Table 1. List of Acronyms*

<b>Acronym</b>	<b>Name</b>
ACCMS	Appellate Court Case Management System
AOC	Administrative Office of the Courts
CCMS	California Case Management System
CCTC	California Courts Technology Center
CEB	Common Element Block
DI	data integration
FFA	Fees, Fines and Assessments
FTA	Failure To Appear
IP	integration partner
ISB	Integrated Services Backbone
LIAM	Local Integration Assessment Methodology
NIEM	National Information Exchange Model
OWSM	Oracle Web Services Management
RRA	request/reply-asynchronous
RRS	request/reply-synchronous
SDD	Service Description Document
SLA	service level agreement
SME	subject matter expert
URL	Universal Resource Locator
WSDL	Web Services Description Language
XML	eXtensible Markup Language



## Appendix B. Deprecated Elements and Attributes

Policy: Name  
Policy: MaximumFilingSize  
Policy: MaximumDocumentSize  
Policy: DelimitExhibits  
Policy: AcceptedMIMETypes  
Policy: AcceptedEncodings  
Policy: AcceptedReplyToProtocols  
Policy: UTCOffset  
Policy: UTCOffsetDaylightSavings  
Policy: Exchanges  
Policy: AssocaitePolicies  
Policy: HumanPolicies  
Policy: Extensions

CourtDivisions: CourtDetails: County  
CourtDivisions: CourtDetails: CourtType  
CourtDivisions: CourtDetails: Group  
CourtDivisions: CourtDetails: Department  
CourtDivisions: CourtDetails: Include

CodeTable: Alias  
CodeTable: Mappings  
CodeTable: Filters  
CodeTable: Codes: Alias  
CodeTable: Codes: GenericType  
CodeTable: Codes: Parameters  
CodeTable: Codes: Include

MinuteTable: Value  
MinuteTable: CourtKeys  
MinuteTable: Mappings  
MinuteTable: Filters  
MinuteTable: Codes: Alias  
MinuteTable: Codes: GenericType  
MinuteTable: Codes: Parameter: Table  
MinuteTable: Codes: Include

NonConfigTable: Alias  
NonConfigTable: CourtKeys  
NonConfigTable: Mappings  
NonConfigTable: Filters  
NonConfigTable: Codes: Alias

NonConfigTable: Codes: GenericType  
NonConfigTable: Codes: Parameters  
NonConfigTable: Codes: Include

Fee:Category  
Fee:CourtKeys  
Fee:Alias  
Fee:Sub category  
Fee:Currency  
Fee:SubsequentPages  
Fee:EffectiveDate  
Fee:ExpirationDate  
Fee>LastUpdate  
Fee:Note  
Fee:FirstPaper  
Fee:CaseTypes  
Fee:AmountInControversy  
Fee:JurisdictionalAmount  
Fee:DocumentGenericTypes  
Fee:DocumentKey  
Fee:Filters  
Fee:Include

## Appendix C. Court Policy File Class Model

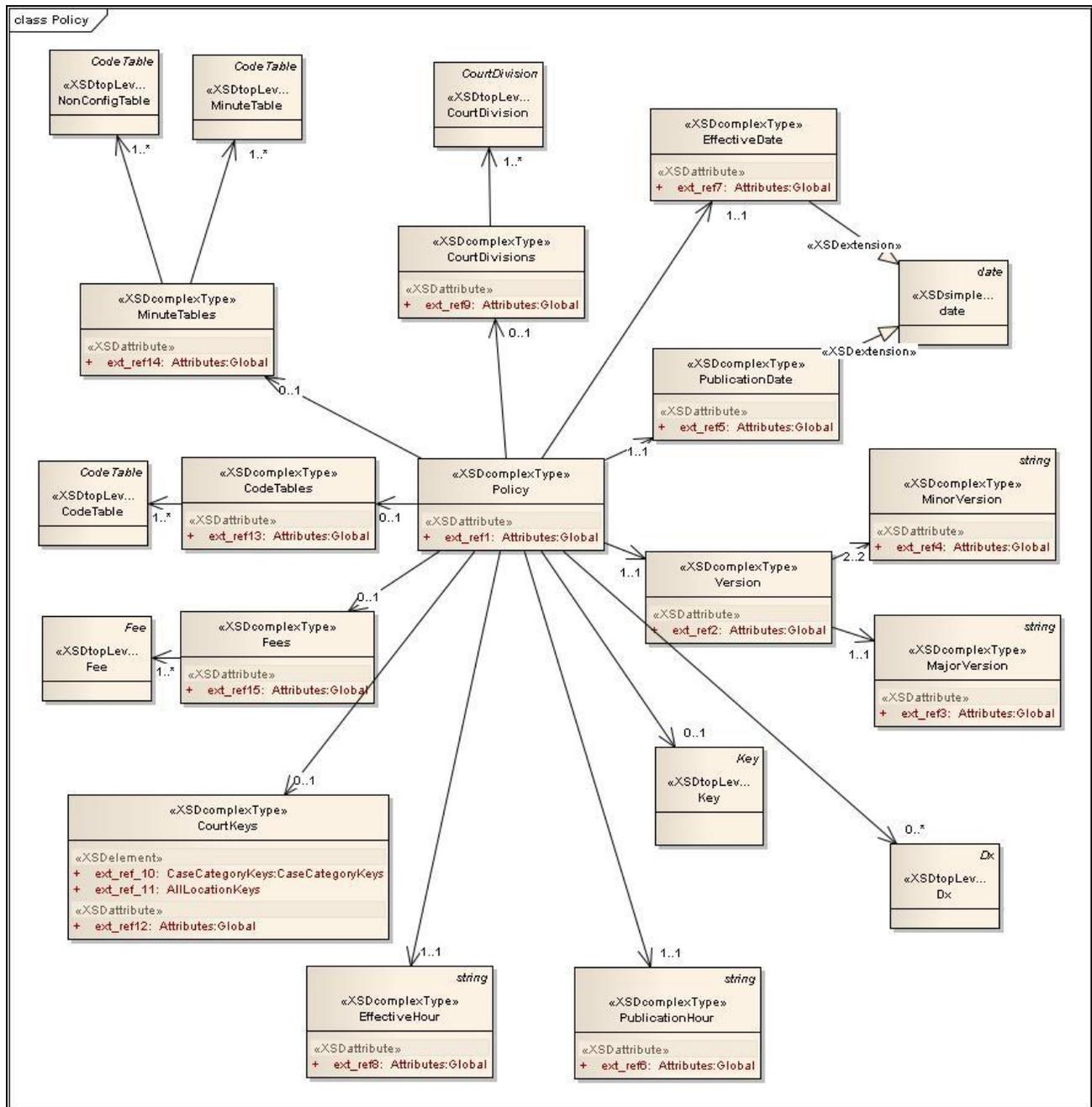


Figure 4. Court Policy File Class Model

