



SDA と CDA の相互運用

Version 2023.1
2024-01-02

SDA と CDA の相互運用

InterSystems Version 2023.1 2024-01-02

Copyright © 2024 InterSystems Corporation

All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble®, InterSystems HealthShare®, InterSystems IRIS®, および TrakCare は、InterSystems Corporation の登録商標です。HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, および InterSystems TotalView™ For Asset Management は、InterSystems Corporation の商標です。TrakCare は、オーストラリアおよび EU における登録商標です。

ここで使われている他の全てのブランドまたは製品名は、各社および各組織の商標または登録商標です。

このドキュメントは、インターシステムズ社(住所: One Memorial Drive, Cambridge, MA 02142)あるいはその子会社が所有する企業秘密および秘密情報を含んでおり、インターシステムズ社の製品を稼働および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

かかるプログラムと関連ドキュメントについて書かれているインターシステムズ社の標準ライセンス契約に記載されている範囲を除き、ここに記載された本ドキュメントとソフトウェアプログラムの複製、使用、廃棄は禁じられている。インターシステムズ社は、ソフトウェアライセンス契約に記載されている事項以外にかかるソフトウェアプログラムに関する説明と保証をするものではない。さらに、かかるソフトウェアに関する、あるいはかかるソフトウェアの使用から起こるいかなる損失、損害に対するインターシステムズ社の責任は、ソフトウェアライセンス契約にある事項に制限される。

前述は、そのコンピュータソフトウェアの使用およびそれによって起こるインターシステムズ社の責任の範囲、制限に関する一般的な概略である。完全な参照情報は、インターシステムズ社の標準ライセンス契約に記載され、そのコピーは要望によって入手することができる。

インターシステムズ社は、本ドキュメントにある誤りに対する責任を放棄する。また、インターシステムズ社は、独自の裁量にて事前通知なしに、本ドキュメントに記載された製品および実行に対する代替と変更を行う権利を有する。

インターシステムズ社の製品に関するサポートやご質問は、以下にお問い合わせください:

InterSystems Worldwide Response Center (WRC)

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: support@InterSystems.com

目次

このドキュメントについて	1
1 CDA ドキュメントと XSL 変換	3
1.1 CDA ドキュメントの構造	3
1.2 CDA ドキュメントの XSLT ディレクトリ構造	7
1.2.1 System ディレクトリ	7
1.2.2 Site ディレクトリ	7
1.2.3 Import ディレクトリと Export ディレクトリ	13
2 CDA XSL 変換のカスタマイズ	15
2.1 変換のカスタマイズの準備 : Docker コンテナ	15
2.2 カスタム変換の作成	16
2.3 変換の拡張	17
2.4 カスタム変換のデバッグ	17
2.4.1 コードのデバッグ	18
2.4.2 相互運用性ツール	19
3 C-CDA 2.1 ドキュメントの前処理	21
4 CDA および SDA アノテーション	23
4.1 アノテーションの検索	23
4.2 レベル	24

テーブル一覧

テーブル 1-1: インポート・プロファイルの設定	9
テーブル 1-2: エクスポート・プロファイルの設定	12

このドキュメントについて

このドキュメントでは、インターシステムズの以下の医療製品で CDA ドキュメント、変換、およびアノテーションを使用する方法について説明します。

- ・ [InterSystems IRIS for Health™](#)
- ・ [HealthShare Health Connect™ \(Health Connect\)](#)
- ・ [HealthShare Unified Care Record™ \(Unified Care Record\)](#)

このドキュメントは、以下の章で構成されます。

- ・ [CDA ドキュメントと XSL 変換](#)
- ・ [CDA XSL 変換のカスタマイズ](#)
- ・ [C-CDA 2.1 ドキュメントの前処理](#)
- ・ [CDA および SDA アノテーション](#)

医療用途の相互運用性の詳細は、以下のドキュメントを参照してください。

- ・ [“IHE のユース・ケース”](#) では、IHE の通信の設定方法について説明します。
- ・ [“SDA：インターシステムズの臨床データ形式”](#) では、SDA の臨床データ・モデルとそのカスタマイズ方法を紹介します。
- ・ [“InterSystems IRIS for Health および Health Connect のレジストリ・ガイド”](#) では、必要なサービス・レジストリを設定および管理する方法について説明します。
- ・ [“Unified Care Record レジストリ”](#) では、HealthShare Unified Care Record システムを管理する方法について説明します。
- ・ [“インターシステムズ製品における FHIR のサポート”](#) では、インターシステムズ製品における FHIR のサポートについて説明します。

1

CDA ドキュメントと XSL 変換

この章では、インターシステムズ製品での CDA ドキュメントの処理方法について説明します。まず、CDA ドキュメントの構造の概要を示し、次に CDA ドキュメントと SDA を相互に変換する XSLT (XSL 変換) のライブラリについて説明します。

1.1 CDA ドキュメントの構造

すべての CDA ドキュメントのルート・ノードは <ClinicalDocument> です。その下に 3 つの論理ディビジョンがあります。ヘッダ、1 つ以上の Section、および各 Section 内の 1 つ以上の Entry です。

- ・ ヘッダには、メタデータ、患者基本情報、およびプロバイダ情報が含まれます。
- ・ Section では、Allergies や Medication などの広範な概念を設定します。Section には、Entry 形式の構造化データのほかに、構造化されていない“叙述”データも記述できます。
- ・ Entry は Section 内に埋め込まれており、より大きい概念の中でインスタンスを表します。例えば、Allergies セクションのある Entry でピーナッツ・アレルギーを表し、別の Entry で花粉アレルギーを表すことができます。

CDA ドキュメントを変換する際、Section と Entry はサポート・ファイルから変換命令を受け取ります。これについては、“[CDA ドキュメントの XSLT ディレクトリ構造](#)”のセクションで説明します。

CDA ドキュメントのすべての部分は、IHE または他の組織によって定義されたテンプレートに準拠します。これらのテンプレートは、OID (オブジェクト識別子) によって識別されます。テンプレートにより、CDA ドキュメントの臨床データを構造化された再利用可能な形式にすることができます。またテンプレートは、CDA ドキュメントが準拠すべき仕様も示します。複数のテンプレートを相互に継承し合い、さらに制約を課すことができます。OID はピリオドで区切られた整数の文字列で構成されます (例: 2.16.840.1.113883.3.88.11.83.102)。特定のテンプレートに関する詳細は、インターネットの検索エンジンに OID を入力してください。OID の仕様を詳しく説明したオンライン・リソースが多数あります。

各 Section には、構造化データと非構造化データが含まれます。非構造化データにはテキスト、数値、段落全体などのアイテムが含まれ、Section の叙述部分に配置されます。非構造化データの役目は 2 つあります。第 1 に、CDA 仕様で要求される、人間が読める CDA のセクションを提供します。第 2 に、後続の構造化データから参照できる参照ポイントを提供します。

構造化データは、Section 内に Entry として記述します。その名前のとおり、構造化データはより多くの仕様に準拠する必要があります。各 Entry にはテンプレートが 1 つ以上関連付けられていて、その Entry が準拠する標準が指定されています。Entry には、日付、間隔、文字列、OID などさまざまなデータが格納されます。また、コード化データを格納し、特定のフィールドで特定の入力パターンを要求することもできます。例えば、codeSystem 属性は有効な OID でなければならない、空白にすることはできません。

次の表に、インターシステムズでサポートされる CCDA v2.1 のSection を示します。左の列に XSL を示し、右の列に対応するセクション名を示します。

XSL	CCDA セクション	メモ
AdvanceDirectives.xsl	Advance Directives	
AllergiesAndOtherAdverseReactions.xsl	Allergies and Intolerances	
AssessmentAndPlan.xsl	Assessment and Plan	
Assessments.xsl	Assessment	
ChiefComplaint.xsl	Chief Complaint	エクスポートでのみサポートされます。
ChiefComplaintAndReasonForVisit.xsl	Chief Complaint and Reason for Visit	エクスポートでのみサポートされます。
CarePlan.xsl		<p>この XSL は以下の SDA 要素を定義します。</p> <ul style="list-style-type: none"> • Care Plan Type • Care Plan Provider • Care Plan Support Contacts • Care Plan SetId • Care Plan Version • Care Plan Authors • Care Plan Organizations • Care Plan Health Concern IDs • Care Plan Goal IDs
DiagnosticResults.xsl	Results	
DischargeDiagnosis.xsl	Discharge Diagnosis	
EncounterDiagnoses.xsl		既定の Encounter の仕様を補完するために使用します。
FamilyHistory.xsl	Family History	
FunctionalStatus.xsl	Functional Status	
Goals.xsl	Goals	
HealthConcerns.xsl	Health Concerns	
HistoryOfPastIllness.xsl	History of Past Illness	
HistoryOfPresentIllness.xsl	History of Present Illness	
HospitalAdmissionDiagnosis.xsl	Admission Diagnosis	
HospitalCourse.xsl	Hospital Course	エクスポートでのみサポートされます。
HospitalDischargeInstructions.xsl	Hospital Discharge Instructions	
HospitalDischargeMedications.xsl	Discharge Medications	
Immunizations.xsl	Immunizations	

XSL	CCDA セクション	メモ
Instructions.xsl	Instructions	
Interventions.xsl	Interventions	
Medications.xsl	Medications	
MedicationsAdministered.xsl	Medications Administered	
Non-RatifiedSections.xsl		<p>この XSL は、Care Considerations の SDA 要素を定義し、以下のサブフィールドが含まれます。</p> <ul style="list-style-type: none"> Entered By Entered At Entered On External ID Document Time Document Number Document Name Document Type File Type Document Stream Document Status Clinician Custom SDA Data
Outcomes.xsl	Health Status Evaluations and Outcomes	
Payers.xsl	Payers	
PhysicalExams.xsl	Physical Exam	エクスポートでのみサポートされます。
PlanOfTreatment.xsl	Plan of Treatment	
ProblemList.xsl	Problem	
ProceduresAndInterventions.xsl	Procedures	
ReasonForReferral.xsl	Reason for Referral	エクスポートでのみサポートされます。
ReasonForVisit.xsl	Reason for Visit	
SocialHistory.xsl	Social History	
VitalSigns.xsl	Vital Signs	

1.2 CDA ドキュメントの XSLT ディレクトリ構造

インターシステムズの医療製品には、CDA ドキュメントと SDA を相互に変換する XSLT のライブラリが付属しています。

使用可能なルートレベルの XSLT を表示するには、インストール・ディレクトリに移動して、パス `install-dir\CSP\xslt\SDA3` に従います。XSLT はこのディレクトリに含まれており、そのほとんどに XSLT が実行するアクションの名前が付いています。

例えば、**CCDA-to-SDA** は Consolidated CDA 1.1 の CCD を SDA に変換し、**CCDAv21-to-SDA** は Consolidated CDA 2.1 の CCD を SDA に変換し、**SDA-to-C32v25** は SDA を C32 に変換します。その後、これらのファイルは、`install-dir\CSP\xslt\SDA3\CDA-Support-Files` の **CDA-Support-Files** ディレクトリにある他のファイルを呼び出します。

ルートレベルの XSLT (**CDA-to-SDA.xsl** など) はそれぞれ、**CDA-Support-Files** ディレクトリにある適切なファイルをインクルードする `xsl:include` 宣言で始まります。

ここでは、**CDA-Support-Files** 内のさまざまなディレクトリとその機能について説明します。イメージからわかるように、**CDA-Support-Files** には以下のディレクトリが含まれます。

- ・ **System** – OID やテンプレートなどの幅広く使用されるアイテムを定義する、ソース・コントロールされた構成不可能なファイル。
- ・ **Site** – さまざまな XSLT で使用される構成可能なファイル。
- ・ **Import と Export** – CDA と SDA を相互に変換する際に呼び出されるファイル。
- ・ **Reports** – 主に、CDA ドキュメントを HTML に変換して Web ブラウザで表示できるようにする変換。このディレクトリについては、以下では説明しません。

重要 XSLT はキャッシュされるため、変換の編集後に変更を反映させるには、該当するネームスペースでプロダクションを再起動する必要があります。

1.2.1 System ディレクトリ

System ディレクトリには、広範なアイテムを定義する静的ファイルが含まれます。このフォルダのアイテムは構成不可能です。

System 内には、次のディレクトリがあります。

- ・ **Common** – あまり使用されないユーティリティ・テンプレートです。
- ・ **OIDs** – OID に関連付けられている変数です。
- ・ **Site-Defaults** – **Site** ディレクトリのソース・コントロール・バージョンです。このディレクトリは、実行時には呼び出されません。詳細は、“[Site ディレクトリ](#)” のセクションを参照してください。
- ・ **Templates** – テンプレート ID に関連付けられている変数です。

1.2.2 Site ディレクトリ

Site ディレクトリには、一意に構成可能なファイルが含まれます。

重要 このディレクトリ内のファイルはアップグレード時に変更されないため、カスタマイズ内容が保持されます。アップグレード後、これらのファイルを install-dir¥CSP¥xslt¥SDA3¥CDA-Support-Files¥System¥Site-Defaults にある新規バージョンのファイルで手動で調整する必要があります。この作業は、これらのファイルをカスタマイズしているかどうかに関係なく行う必要があります。Site ディレクトリ内にあるカスタマイズしていないファイルもすべてこのディレクトリに手動でコピーし、Site-Defaults にある新規バージョンで更新する必要があります。

Site 内には、次のファイルが含まれます。

- ・ **ImportProfile** および **ExportProfile** – CDA ドキュメントをインスタンスにインポートするプロセス、およびインスタンスから CDA ドキュメントをエクスポートするプロセスで使用する構成設定です。以下の“[インポート・プロファイル](#)”および“[エクスポート・プロファイル](#)”のセクションを参照してください。
- ・ **OutputEncoding** – 出力される CDA ドキュメントのエンコードを制御するために使用される XSLT です。既定値は UTF-8 です。
- ・ **Variables** – インポートまたはエクスポートの際に使用される構成可能な変数です。これらの変数は組織を表し、“ホーム”情報を設定します。

1.2.2.1 インポート・プロファイル

インポート・プロファイルは、CDA ドキュメントのインポート時に構成設定を制御します。インポート・プロファイルのほぼすべての変数には、<sectionTemplateId>、<entryTemplateId>、またはその両方があります。例外は resultsImportConfiguration で、代わりに <sectionC32TemplateId> と <sectionC37TemplateId> に別個のテンプレート ID が付けられています。<sectionTemplateId> または <entryTemplateId> のどちらが存在するかは、問題の変数が Section モジュールと Entry モジュールのどちらにあるかによって決まります。

さらに、変数によっては他の設定が含まれることもあります。次の表に、このような設定、および設定が属する変数とその値を示します。

テーブル 1-1: インポート・プロファイルの設定

変数名	設定	値
<code>generalImportConfiguration/blockImportCTDCodeFromText</code>	disabled	<p>CDA の <code>@code</code> 属性が利用できない場合に、CDA 文字列、叙述テキスト、または <code>originalText</code> の SDA <code>CodeTableDetail</code> コード・プロパティへのインポートをブロックします。</p> <p>この設定が有効化されていて、コード化要素が <code>nullFlavor</code> の場合、テキストは SDA のコード・プロパティにロードされません。ただし、ターゲットの SDA 要素が <code>OrderItem</code> であり、構成パラメータ <code>orderItemDefaultCode</code> または <code>orderItemDefaultDescription</code> が有効になっている場合を除きます。</p> <p><code>blockImportCTDCodeFromText</code> が有効になっていない場合、インポート動作は前バージョンと同じです。</p>
<code>generalImportConfiguration/sdaActionCodes</code>	enabled	SDA アクション・コードが有効化されているかどうかを示します。SDA アクション・コードは、データの更新と削除を制御します。
<code>generalImportConfiguration/sdaActionCodes</code>	overrideExternalId	該当する場合、CDA <code><id></code> 要素を使用して SDA <code>ExternalId</code> プロパティ値をインポートする必要があるかどうかを示します。
<code>generalImportConfiguration</code>	enableOtherOrders	有効化されていて、CDA の <code>Result</code> を SDA の <code>LabOrder</code> または <code>RadOrder</code> として分類できない場合、 <code>Result</code> は SDA <code>OtherOrder</code> としてインポートされます。それ以外の場合、 <code>Result</code> はインポートされません。

変数名	設定	値
generalImportConfiguration/representedOrganizationId	concatRootAndNumericExtension	値が 1 の場合、hl7:representedOrganization/hl7:id @root が OID で、@extension が数値であれば、両者が 1 つの施設 OID に連結されます。
generalImportConfiguration	narrativeImportMode	値 1 では、叙述セクションをテキストとしてインポートし、 と叙述行フィードを両方とも行フィードとしてインポートします。値 2 では、 のみを行フィードとして使用して、テキストとしてインポートします。 Result Text、Hospital Discharge Instructions、および Reason for Visit のインポートにのみ適用されます。
dischargeMedicationsImportConfiguration medicationsImportConfiguration medicationsAdministeredImportConfiguration	pharmacyStatus	Medication のステータスを示します。値はどの変数に配置されているかによって異なり、Medication のインポート元の CDA セクションを示します。
planImportConfiguration	effectiveTimeCenter	0 に設定すると、effectiveTime/center 値は FromTime にインポートされます。 1 に設定すると、effectiveTime/center 値は FromTime と ToTime にインポートされます。 注意：effectiveTime/center を特定の care plan に対して生成する場合、その care plan に effectiveTime/high および effectiveTime/low を入力しないでください。
resultsImportConfiguration	resultOrganizerTemplateId	hl7:organizer が複数ある場合に、指定された結果エントリ内で正しい hl7:organizer を選択するのに役立ちます。指定される可能性がある代替値の 1 つとして、\$ihe-PCC-LabBattery-Organizer があります。

変数名	設定	値
resultsImportConfiguration	orderItemDefaultCode	OrderItem Code または Description を導出するための情報が CDA の Result に含まれない場合に、SDA の OrderItem Code に使用するコードです。
resultsImportConfiguration	orderItemDefaultDescription	SDA の OrderItem Code または Description を導出するための情報が CDA の Result に含まれない場合に、SDA の OrderItem Code に使用する説明です。
healthFundImportMode	enabled	値が 1 であれば、この設定は有効です。これが既定の設定です。得られた SDA への Encounter ごとに Health Fund ストリームレットが追加されます。
healthFundImportMode	disabled	値が 0 であれば、この設定は無効です。Encounter ごとに Health Fund ストリームレットが作成されることはありません。
memberEnrollmentImportMode	enabled	値が 1 であれば、この設定は有効です。CCD/CCDA に Payor を受け取るたびに、得られた SDA に Member Enrollment ストリームレットが作成されます。
memberEnrollmentImportMode	disabled	値が 0 であれば、この設定は無効です。これが既定の設定です。得られた SDA に Member Enrollment ストリームレットは作成されません。

1.2.2.2 エクスポート・プロファイル

エクスポート・プロファイルのほとんどの要素には、<emptySection> 要素、<narrativeLinkPrefixes> 要素、またはその両方があります。<emptySection> 要素は、情報を含まないセクションをエクスポートするかどうかを制御します。<narrativeLinkPrefixes> には、CDA ドキュメントの叙述部分、すなわち人間が読める部分に含めるテキストを入力できます。

さらに、セクションによっては他の設定が含まれることもあります。このような設定は、エクスポートするデータに応じて編集および拡張できます。次の表に、このような一意の設定、および設定が存在するセクションとその値を示します。

テーブル 1-2: エクスポート・プロファイルの設定

セクション	設定	値
admissionDiagnoses/diagnosisType	codes	Admission の追加の診断タイプとして使用できるコード。
advanceDirectives/advanceDirectiveType	codes	Resuscitation や intubation など、さまざまな事前指示タイプに対応するコード。
assessment/diagnosisType	codes	Assessment の追加の診断タイプとして使用できるコード。
dischargeDiagnoses/diagnosisType	codes	Discharge の追加の診断タイプとして使用できるコード。
encounterDiagnoses/exportToC32	disabled	有効な場合 (値 = 1)、encounter diagnosis を CDA C32 にエクスポートできます。
medications/currentMedication	includeHistoricalMedications	現在の Medications リストに、Medications 履歴を含めるかどうかを制御します。
medications/currentMedication	windowInDays	Medication の有効期間と、Medication を引き続き現在の Medications リストに含めることができる期間を制限します (日数)。
medications/currentMedication	hideNarrativeColumn	叙述列を非表示にします。
planOfCare	effectiveTimeCenter	<p>0 に設定すると、FromTime が effectiveTime/low にエクスポートされ、ToTime が effectiveTime/high にエクスポートされます。</p> <p>1 に設定すると、FromTime と ToTime は同じ NULL 以外の値になり、その値が effectiveTime/center にエクスポートされます。それ以外の場合は、FromTime に値があれば、その値が effectiveTime/low にエクスポートされます。ToTime に値があれば、その値が effectiveTime/high にエクスポートされます。</p> <p>例外 : ProcedureTime に値がある場合、その値が FromTime の代わりに effectiveTime/Center にエクスポートされます。</p>
problems/currentCondition	codes	ACTIVE や CHRONIC など、さまざまな問題タイプに対応するコード。
problems/currentCondition	windowInDays	Medication の有効期間と、Medication を引き続き現在の問題リストに含めることができる期間を制限します (日数)。

セクション	設定	値
socialHistory/emptySmokingStatus	exportData	データが含まれていない場合に smokingStatus をエクスポートするかどうかを指定します。

1.2.3 Import ディレクトリと Export ディレクトリ

Import ディレクトリと Export ディレクトリには、XSLT がデータを変換する際に呼び出されるファイルが含まれます。

- ・ Import ディレクトリには、CDA を SDA に変換する際に呼び出されるファイルが含まれます。
- ・ Export ディレクトリには、SDA を CDA に変換する際に呼び出されるファイルが含まれます。

Import ディレクトリと Export ディレクトリの両方に、Common、Entry-Modules、Section-Modules というラベルが付いたサブディレクトリが含まれます。

- ・ Common ディレクトリには、さまざまな変換でよく使用されるグローバル変数を設定する XSLT が含まれます。また、これらの XSLT は、よく使用されるロジック (address-Home など) が含まれるテンプレートを提供します。
- ・ Section-Modules ディレクトリには、指定された CDA セクションとの間でデータを変換する XSLT が含まれます。各 XSLT の名前は、その XSLT で変換する CDA セクションの名前に厳密に一致します。
- ・ Entry-Modules ディレクトリには、変換をドキュメントの正しい Entry にマップしてからコード化データを解析する XSLT が含まれます。このため、このディレクトリ内の XSLT は、Section-Modules ディレクトリ内の XSLT と比べて大幅に長くなる傾向があります。

ルートレベルの XSLT (CDA-to-SDA.xsl など) はそれぞれ、適切な Section-Modules のファイル、Entry-Modules のファイル、Common のファイル、必要な他のファイルをインクルードする `xsl:include` 宣言で始まります。

2

CDA XSL 変換のカスタマイズ

CDA XSL 変換の標準ライブラリを拡張する必要が生じることがあります。例えば、特定の CDA セクションを変換する場合に、ロジックの一部を変更する必要が生じることがあります。あるいは、カスタム SDA 拡張、カスタム・ペア、またはカスタム・オブジェクトのサポートを追加したい場合もあります。カスタム SDA 拡張、ペア、およびオブジェクトの詳細は、“SDA：インターシステムズの臨床データ形式”の“[SDA のカスタマイズ](#)”のセクションを参照してください。

XSL の標準ライブラリを直接編集する代わりに、カスタム XSL を作成することができます。次の 2 つの理由により、直接編集よりもカスタム XSL を作成することを推奨します。第 1 に、標準のルート XSL をこの方法で上書きすることで、その変換への既存のアプリケーション呼び出しを変更する必要がなくなります。第 2 に、アップグレードの結果、標準の XSL ライブラリが置き換えられてカスタマイズが消去される事態を防止することができます。カスタム・ディレクトリ内の XSL はアップグレード時に置き換えられません。

2.1 変換のカスタマイズの準備：Docker コンテナ

重要 変換をカスタマイズする計画があり、InterSystems IRIS for Health が Docker コンテナで実行されている場合、コンテナは永続ディレクトリを指している必要があります。永続ディレクトリの詳細は、“コンテナ内でのインターシステムズ製品の実行”を参照してください。

インスタンスがコンテナ内で実行されている場合は、以下の準備ステップを実行してから、変換のカスタマイズを開始する必要があります。

1. 管理ポータルで、`/csp/xslt` Web アプリケーションを作成します。
 - a. 管理ポータルに `%HS_Administrator` ロールを持つユーザとしてログインします。
 - b. [システム管理]→[セキュリティ]→[アプリケーション]→[ウェブ・アプリケーション] に移動します。
 - c. [新しいウェブ・アプリケーションを作成] をクリックします。
 - d. フォームで、以下のようにフィールドに情報を入力します。
 - ・ 名前 – `/csp/xslt`
 - ・ ネームスペース – `HSLIB`
 - ・ 有効 – [アプリケーション有効] と [CSP/ZEN] を選択
 - ・ [CSP ファイルの設定] は以下のようにします。
 - 静的ファイルの提供 – 既定値を使用
 - 静的ファイルの提供タイムアウト – 既定値を使用

- **物理パス** - 永続ディレクトリ内のディレクトリ (/container/durable/csp/xslt/ など)。必ず末尾にスラッシュを付けてください。
- **ウェブ設定** - 既定値を使用

e. [保存] をクリックします。

- ターミナル・セッションを開き、**%HS_Administrator** ロールを持つユーザとしてログインします。
- カスタマイズする XSLT ファイルを上の手順で指定された物理パスにコピーするには、**HSLIB** ネームスペース内で以下を実行します。

```
do ##class(HS.HC.Util.Installer.Upgrade.XSLTDirectoryCopy).Update()
```

詳細な出力を表示するには、Update() メソッドの単一パラメータとして 1 を指定します。

2.2 カスタム変換の作成

カスタム CDA 変換を作成するには、以下の手順に従います。

- 変換が存在するディレクトリに **Custom** サブディレクトリを作成します。例えば、install-dir/CSP/xslt/SDA3/CDA-to-SDA.xsl をカスタマイズする場合、カスタム・サブディレクトリは install-dir/CSP/xslt/SDA3/Custom にする必要があります。カスタム・フォルダ内のファイルはアップグレード時に上書きされません。
- 好みの IDE で、カスタマイズする変換を開き、その一部をコピーできるようにします。
- カスタマイズする XSL と同じ名前の空のカスタム XSL を作成します。例えば、CDA-to-SDA.xsl をカスタマイズする場合は、install-dir/CSP/xslt/SDA3/Custom に CDA-to-SDA.xsl という名前の XSL を作成します。
- 以下の初期文をカスタム XSL に追加します。最初の 2 つ (XSL 宣言と XSL スタイルシート) は、元の CDA XSL ファイルからコピーできます。

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:isc="http://extension-functions.intersystems.com" xmlns:hl7="urn:hl7-org:v3"
  xmlns:sdtc="urn:hl7-org:sdtc" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:exslt="http://exslt.org/common" exclude-result-prefixes="isc hl7 sdtc xsi exslt">
```

- スタイルシートの終了タグを追加します。カスタマイズは、xsl:stylesheet の開始タグと終了タグの間に配置されます。

```
</xsl:stylesheet>
```

- カスタマイズする変換の import 文を追加します。

```
<xsl:import href="../../../<filename>"/>
```

<filename> はカスタマイズする変換の名前です (例: CDA-to-SDA.xsl)。カスタマイズするファイルがインポートされ、そのテンプレートを 1 つ以上上書きできるようになります。

- 上書きするテンプレートを元の変換からコピーして、新しいファイルに貼り付けます。
- 必要に応じて、テンプレート・ロジックをカスタマイズします。これにより、元の XSL のテンプレートが上書きされます。
- ベストプラクティスとして、テンプレートのコピー元を示すコメントを追加します。例えば、テンプレートが元々 install-dir/CSP/xslt/SDA3/CDA-Support-Files/Import/Entry-Modules/Medications.xsl にあった場合は、その旨を記載します。

10. 作業内容を保存します。
11. インストール環境がミラーにデプロイされている場合は、上記のプロセスをすべてのミラー・メンバで繰り返します。
12. プロダクションを再起動してカスタム XSL を有効にします。

2.3 変換の拡張

“SDA：インターシステムズの臨床データ形式”の“[SDA のカスタマイズ](#)”の章で説明されているように、SDA は複数の方法で拡張できます。以下の拡張をカスタム CDA 変換に組み込むことができます。

- ・ カスタム・コンテナまたはカスタム・オブジェクトにより、サポートされていない CDA セクションを取得できる
- ・ 拡張クラスにより、サポートされているセクションに属しているのに SDA で取得されないデータを追加できる

空のテンプレートを CDA のインポートおよびエクスポートの XSL に含めて、以下を容易に行うことができます。

- ・ サポートされていない CDA セクションをカスタム・コンテナまたは `<CustomObject>` にインポートする
- ・ CDA データを SDA 拡張クラスまたは既存の SDA セクション内の `<CustomPairs>` にインポートする
- ・ カスタム・コンテナまたは `<CustomObject>` をサポートされていない CDA セクションにエクスポートする

空のテンプレートには常に `Custom` という語が含まれるため、容易に識別できます。

以下で、上記の各カスタマイズを実装する方法を詳しく説明します。

- ・ サポートされていない CDA セクションをカスタム・コンテナまたは `<CustomObject>` にインポートするには、編集する XSL でカスタム・テンプレートを見つけます。例えば、**CDA-to-SDA.xsl** では、テンプレートは `<xsl:template match="*" mode="ImportCustom-Container">` という名前です。このテンプレートをカスタム **CDA-to-SDA.xsl** ファイル（または、カスタマイズするファイルのカスタム・バージョン）にコピーし、空のカスタム・テンプレートを編集して必要なロジックを含めます。
- ・ CDA データを SDA 拡張クラスまたは既存の SDA セクション内の `<CustomPairs>` にインポートするには、目的のセクションのインポート XSL でカスタム・テンプレートを見つけます。例えば、**BodySite** 要素を **Procedure** にインポートするには、**Entry-Modules/Procedure.xsl** 内でカスタム・テンプレートを見つけます。この例では、テンプレートは `<xsl:template match="*" mode="ImportCustom-Procedure">` という名前です。このテンプレートをカスタム **CDA-to-SDA.xsl** ファイル（または、カスタマイズするファイルのカスタム・バージョン）にコピーし、空のカスタム・テンプレートを編集して必要なロジックを含めます。
- ・ カスタム・コンテナまたは `<CustomObject>` をサポートされていない CDA セクションにエクスポートするには、XSL 内で編集するカスタム・テンプレートを見つけます。例えば、**SDA-to-C32v25.xsl** では、テンプレートは `xsl:template match="*" mode="ExportCustom-ClinicalDocument">` という名前です。このテンプレートをカスタム **SDA-to-C32v25.xsl** ファイル（または、カスタマイズするファイルのカスタム・バージョン）にコピーし、空のカスタム・テンプレートを編集して必要なロジックを含めます。

2.4 カスタム変換のデバッグ

カスタム変換のデバッグは、さまざまな状況に適用可能な一連のツールと手法で構成されます。

デバッグでは、以下を行うことができます。

- ・ `<DEBUGGING>` タグ内に含まれるコードを使用して、変換の実行時に値を取得できます。

- ・ プロダクション内で相互運用性ツールを使用してドキュメントのトレースを表示できます (HealthShare Unified Care Record のみ)。

2.4.1 コードのデバッグ

<DEBUGGING> タグを使用すると、ステップを追って変換を実行することができます。変換内で値を取得したい場所に <DEBUGGING> タグを配置します。タグ内のコードで変数、属性、または XPath の値を取得できます。

以下に、さまざまな状況におけるデバッグ・コードの例を示します。

変数の値の取得

この例では、FileName.xml という XSL 内で変数 \$variable の値を取得します。

```
<DEBUGGING-VARIABLE-file-FileName.xml-TemplateName>
$variable = <xsl:value-of select="$variable"/>
</DEBUGGING-VARIABLE-file-FileName.xml-TemplateName>
```

属性の値の取得

この例では、/ClinicalDocument/id にある root 属性の XSL パス値を取得します。

```
<DEBUGGING-PATHVALUE>
CDA source OID = <xsl:value-of select="/hl7:ClinicalDocument/hl7:id/@root"/>
</DEBUGGING-PATHVALUE>
```

ノード位置付きの XPath の取得

XPath の取得では、テンプレートを利用します。テンプレート currentXPathWithPos を呼び出すと、現在の XPath がノード位置付きで返されます。例えば ClinicalDocument/component[1] と指定すると、ClinicalDocument 内の最初の component 要素が返されます。以下の例は、現在の位置の XPath をノード位置付きで取得する方法を示しています。

```
<DEBUGGING-PATHXMLWithPos>
<xsl:apply-templates select="." mode="currentXPathWithPos"/>
</DEBUGGING-PATHXMLWithPos>
```

この呼び出しの結果は、以下のようになります。

XML

```
<DEBUGGING-PATHXMLWithPos>/ClinicalDocument[1]/component[1]/structuredBody[1]/component[8]
/section[1]/entry[1]/procedure[1]</DEBUGGING-PATHXMLWithPos>
```

ノード位置なしでの XPath の取得

currentXPath テンプレートを呼び出すと、XPath がノード位置なしで返されます。以下の例は、現在の位置の XPath をノード位置なしで取得する方法を示しています。

```
<DEBUGGING-PATHXMLWithoutPos>
<xsl:apply-templates select="." mode="currentXPath"/>
</DEBUGGING-PATHXMLWithoutPos>
```

この呼び出しの結果は、以下のようになります。

XML

```
<DEBUGGING-PATHXMLWithoutPos>/ClinicalDocument/component/structuredBody/component
/section/entry/procedure</DEBUGGING-PATHXMLWithoutPos>
```

テンプレート currentXPathWithPos と currentXPath は、install-dir/csp/xslt/SDA3/CDA-Support-Files/System/Common/Functions.xsl にあります。

XPath ツリーの取得

copy テンプレートを呼び出すと、現在の XPath ツリーが返されます。以下の例では、DEBUGGING タグに現在の XPath ツリーを挿入して、現在の XPath ツリーを取得します。

```
<DEBUGGING-PATHXMLTree>
<xsl:apply-templates select="." mode="copy" />
</DEBUGGING-PATHXMLTree>
```

テンプレート copy は、install-dir/csp/xslt/SDA3/CDA-Support-Files/System/Common/Functions.xsl にあります。

これらの <DEBUGGING> タグと取得した値は、変換には表示されますが、Clinical Viewer や [要約の表示] ページには表示されません。

<DEBUGGING> タグを使用すると、CDA の処理が失敗する場合があります。このため、<DEBUGGING> タグはデバッグ・プロセスでのみ使用してください。

2.4.2 相互運用性ツール

インターシステムズ製品では、変換プロセスを検査できる相互運用性ツールが用意されています。メッセージ・トレースを使用して、以下のポイントでプロセスを検査できます。

- ・ HealthShare Unified Care Record :
 - ドキュメント取得時に、コンシューマの Edge Gateway プロダクションで (リポジトリ内の CDA が SDA に変換されるとき)
 - Clinical Viewer が受信 SDA を受け取ったとき
- ・ InterSystems IRIS for Health および Health Connect : 関連するビジネス・オペレーションのメッセージ・トレースを検証します。例えば、インバウンド SDA の場合は、インバウンド SDA 処理の変換を扱うビジネス・オペレーションのメッセージ・トレースを調べます。

さらに、[要約を表示] ページで送信 CDA の変換を確認することもできます。

2.4.2.1 メッセージ・トレース

メッセージ・トレースを表示するには、以下の手順に従います。

1. 管理ポータルに %EnsRole.Administrator ロールを持つユーザとしてログインします。
2. 適切なネームスペースを選択します。
 - ・ HealthShare Unified Care Record の場合 :
 - ドキュメント取得のトレースを表示するには、コンシューマ Edge Gateway のネームスペースを選択します。
 - 受信 SDA のトレースを表示するには、Access Gateway のネームスペースを選択します。
 - ・ InterSystems IRIS for Health または Health Connect の場合 : 変換を処理するビジネス・ホストの Foundation ネームスペースを選択します。
3. [相互運用性]→[構成]→[プロダクション] に移動します。
4. プロダクションを開きます。
5. 適切なコンポーネントを選択します。例えば、HS.IHE.XDSb.Consumer.Operations を使用してドキュメントを取得できます。
6. 右側のパネルで [メッセージ] タブをクリックします。

7. [メッセージ・ビューワに移動] リンクをクリックします。
8. 変換を表すメッセージを選択します。上の例に従うと、このメッセージのソースは **HS.IHE.XDSb.Consumer.Operations** です。
9. 右側のパネルで [トレース] タブをクリックします。
10. [完全トレースを表示] リンクをクリックします。
11. 右側のパネルで [コンテンツ] タブをクリックします。

[コンテンツ] タブにメッセージ・トレースが表示されます。

2.4.2.2 要約の表示

HealthShare Unified Care Record を実行している場合は、CDA ドキュメントと SDA の対応ドキュメントとを Clinical Viewer 内で比較することもできます。

Clinical Viewer を使用するには、以下の手順に従います。

1. 管理ポータルに %HS.Administrator ロールを持つユーザとしてログインします。
2. [HealthShare] → <Access_Gateway_Namespace> [患者検索] に移動します。
3. 表示する CDA ドキュメントの患者の検索情報を入力します。
4. 患者を選択します。

Clinical Viewer でページ上部の [要約の表示] リンクをクリックして、[要約の表示] ページを開きます。このページで、患者の要約を SDA ドキュメントとして、または Continuity of Care Document (CCD、HITSP C32) などの CDA ドキュメントとして表示できます。[要約の表示] ページの使用の詳細は、“Unified Care Record セキュリティ・ガイド”の“[患者レポートの表示および変更](#)”の章を参照してください。

3

C-CDA 2.1 ドキュメントの前処理

インターシステムズの医療製品では、XSLT 1.0 を使用した C-CDA 2.1 から SDA へのインポート変換をサポートしています。XSLT 2.0 を使用した C-CDA 2.1 から SDA への変換はサポートしていません。

C-CDA 2.1 のサポートに関連するインポート機能の強化の 1 つとして、インポート用の変換を実行する前に C-CDA 入力ファイルを前処理する機能が追加されました。前処理サポートの導入により、変換が大幅に簡素化され、変換に要する合計処理時間が削減されます。

考えられるユース・ケースは次の 2 つです。

- ・ 要素がない、またはその形式が正しくない
- ・ 日付スタンプの変換

C-CDA を前処理して句読点を取り除き、特定の要素に必要な子ノードを追加する方法、およびプリプロセッサ変換のサンプル・コード、トップレベル変換の変更、入力ファイルの詳細は、InterSystems Developer Community の記事 [“C-CDA 2.1 からインポートトランスフォーメーションを実行する際の前処理サポート”](#) を参照してください。

一般に、以下のような手順になります。

1. 組織のドメイン名の管理者の協力を得て、コード拡張に使用するサブドメイン名を取得します。
2. 前処理のカスタム・モードに使用するプレフィックスと名前を選択します。
3. XSL プリプロセッサを作成します。
 - ・ stylesheet 要素は以下の形式にします。

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:hl7="urn:hl7-org:v3" xmlns:custom_mode_prefix="<subdomain>"
exclude-result-prefixes="hl7 <custom_mode_prefix>"></xsl:stylesheet>
```

カスタム・モードのプレフィックスを含めることに注意してください。
 - ・ トップレベルの要素を複製してすべての属性、要素、テキスト・ノードをコピーするテンプレートを含めます。両方のテンプレートで、カスタム・モードを使用する必要があります。
 - ・ C-CDA の変更ごとにテンプレートを作成します。これらのテンプレートでもカスタム・モードを使用する必要があります。
4. XSL プリプロセッサを ../CDA-Support-Files/Import/ にコピーします。
5. **CCDAv21-to-SDA.xsl** または **CCDAv21-nonXML-to-SDA.xsl** を複製し、以下のように変更して、カスタムのトップレベル変換を作成します。
 - ・ stylesheet 要素にカスタム・ネームスペースを含めます。

- ・ XSL プリプロセッサのパス名を指定する include 文を追加します。
- ・ 次の行を見つけてます。

```
<xsl:variable name="input" select="/hl7:ClinicalDocument"/>
```

これを次のブロックに置き換えます。

```
<xsl:variable name="inputRTF">
```

```
<xsl:apply-templates select="/hl7:ClinicalDocument"
mode="<custom_mode_prefix>:<mode>"/>
```

```
</xsl:variable>
```

```
<xsl:variable name="input"
select="exsl:node-set($inputRTF)/hl7:ClinicalDocument"></xsl:variable>
```

6. プロダクションで、C-CDA > SDA 変換用のファイル入力を取るビジネス・サービスを見つけてます。
7. そのサービスの **InputXSL** という追加設定の値を、カスタムのトップレベル変換のファイル名に設定します。
8. C-CDA ファイルを通常どおりにインポートします。

4

CDA および SDA アノテーション

CDAドキュメントとSDAを相互に変換する際、ソース・ドキュメントのフィールドとターゲット・ドキュメントのフィールドの関係を判断するのが難しいことがよくあります。こうした関係を解明するために、ドキュメント・マッピング・ユーティリティを使用して、指定されたターゲット・フィールドのソース・フィールドを特定することができます。

4.1 アノテーションの検索

管理ポータルを開いて、以下に移動します。

- ・ [HealthShare]→<namespace>→[スキーマ・ドキュメント]→[SDA/CDA アノテーション]。<namespace> は、HealthShare Unified Care Record の Edge Gateway ネームスペースです。
- ・ [Health]→<namespace>→[スキーマ・ドキュメント]→[SDA/CDA アノテーション]。<namespace> は InterSystems IRIS for Health または HealthConnect の Foundation ネームスペースです。

ソース・フィールドを特定するには、まず、選択元のターゲット・フィールドのリストを生成する必要があります。

1. ターゲット・ドキュメント・タイプを 1 つ以上選択します。例えば、SDA ドキュメント内のフィールドのソースを指定する場合は、[SDA] チェック・ボックスにチェックを付けます。複数のドキュメント・タイプを選択するか、[] を選択できます。選択できるタイプは次のとおりです。
 - ・ SDA — 特定の SDA 要素のクラスおよびプロパティ・パス。例：HS.SDA3.Medication.Comments。
 - ・ XPathSDA — 特定の SDA 要素の XPath。コンテナの XML を調べる場合や、XML の XSLT を記述する場合は、このタイプの方が SDA よりも便利です。例えば、XPath /Container/Medications/Medication/Comments はクラスおよびプロパティ・パス HS.SDA3.Medication.Comments に対応します。
 - ・ CCDA1 — Consolidated CDA Release 1.1。
 - ・ CCDA2 — Consolidated CDA Release 2.1。
 - ・ C32 — HITSP/C32。
 - ・ XDLAB — XD-LAB。IHE Laboratory Technical Framework の CDA ドキュメントです。
2. テキスト・ボックスにフィールド名またはパスの全体または一部を入力します。わかっている情報を入力するだけでかまいません。XML フラグメントを貼り付けることもできます。フラグメントの開始タグと終了タグが同じである必要はありませんが、検索するフィールド名は <target-field> のように山カッコで閉じる必要があります。これにより、複雑なフィールド名を単純な検索条件で見つけることができます。

3. テキストの入力が終わったら、[オプションの表示]をクリックするか、カーソルがテキスト・ボックス内にある間に Enter キーを押して、ターゲット・フィールドの候補のリストを生成します。
4. ドロップダウン・メニューからターゲット・フィールドを選択します。このメニューでは結果が 250 件に制限されています。結果を絞り込むには、さらにテキストを入力するか、選択するドキュメント・タイプを減らします。

ドキュメント変換マッピング情報が検索領域の下に表示されます。[結果を一番上に移動] リンクをクリックして、ページに再度フォーカスできます。選択したターゲット・フィールドが [ターゲット] という見出しの下に以下の情報と共に表示されます。

- ・ ターゲットが対応する [ドキュメントタイプ]。
- ・ [ターゲット・フィールド名] (存在する場合)。
- ・ ターゲットの [パス]。

[ソース] という見出しの下には、ソース・フィールドのドキュメント・タイプ、フィールド名 (存在する場合)、パスも表示されます。また、マッピングに関する注記と、ソースをターゲットに変換するために実行されるマッピングのタイプも表示されます (例えば、HL7 ソースを SDA ターゲットに変換するための HL7 to SDA3 Mapping)。

4.2 レベル

ソース・フィールド自体が他のフィールドのターゲットである場合、マッピング・ユーティリティでは、それらのフィールドは新しいレベルに表示されます。例えば、下図では、HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code はレベル 1 の C32 フィールドのソースであると同時に、レベル 2 の C-CDA1 ドキュメントのフィールドのターゲットでもあります。

Level 1

Target

Document Type C32
Target Field Name Medication Information Source
Path /ClinicalDocument/component/structuredBody/component/section[templateId/@root='1.3.6.1.4.1.19376.1.5.3.1.3.19']/entry/sr

Source

Document Type SDA
Source Field Name Medication Information Source
Path HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code
Mapping SDA To CDA Mapping (Export)
Trace Down This source is a target in Level 2.
Trace to Top ►

Level 2

Target

Document Type SDA
Target Field Name Medication Information Source
Path HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code
Trace to Top ►

Sources

Document Type CCD1
Source Field Name Medication Information Source
Path /ClinicalDocument/component/structuredBody/component/section[templateId/@root='2.16.840.1.113883.10.20.22.2.1.1']/ent

レベル間を移動するには、[このソースはレベル [n] のターゲットです] というリンクをクリックするか、[このターゲットはレベル [n] のソースです] というリンクをクリックします。n は、正の整数です。パス値をクリックして、すべてのレベルで同じパス値を色付きで強調表示することもできます。再度クリックすると、強調表示は取り消されます。

[ターゲット] および [ソース] の各見出しには [一番上までトレース] オプションもあります。矢印をクリックして展開します。このオプションを選択すると、マッピング階層が表示されます。

- ・ トップレベルのターゲット・フィールド（検索対象）が一番上に表示されます。
- ・ 現在参照しているフィールドは一番下に表示されます。
- ・ 中間フィールド（存在する場合）はその間に表示されます。

Level 2

Target

Document Type SDA
Target Field Name Medication Information Source
Path HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code

Trace to Top ▶

Sources

Document Type CCDA1
Source Field Name Medication Information Source
Path /ClinicalDocument/component/structuredBody/component/section[templateId/@root='2.16.840.1.113883.10.20.22.2.1.1']/entry/substance
Mapping CDA To SDA Mapping (Import)
Notes If informant is not present then author is used.
 - Else if author is not present and current source node is informant then current source node is used.
 - Else if current source node is author then current source node is used.
 - Else if document-level informant is present then document-level informant is used.
 - Else if document-level author is present then document-level author is used.
 - Else if document authoring device is present then use document authoring device.

Trace Down This source is a target in Level 3.

Trace to Top ▼

(C32) /ClinicalDocument/component/structuredBody/component/section[templateId/@root='1.3.6.1.4.1.19376.1.5.3.1.3.19']/entry/substance
 ^ SDA To CDA Mapping (Export)
 (SDA) HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code
 ^ CDA To SDA Mapping (Import)
 (CCDA1) /ClinicalDocument/component/structuredBody/component/section[templateId/@root='2.16.840.1.113883.10.20.22.2.1.1']/entry/substance

これらの階層レベルはマッピングによって結び付けられ、階層の方向がキャレット ^ で示されます。

階層内の各フィールドには、そのフィールドがターゲットとして存在する場所のページをフォーカスするリンクが含まれます。例えば、上の図で (SDA) HS.SDA3.AbstractOrder.EnteredAt.Address.State.Code をクリックすると、そのすぐ上にある [ターゲット] セクションのページが再度フォーカスされます。このページが、このフィールドがターゲットとして機能している場所です。

