

厚生労働行政推進調査事業費補助金
(地域医療基盤開発推進研究事業)
総括研究報告書

医療文書と診療情報共有のための FHIR ドキュメント実装ガイド
生成エコシステムに関する研究

研究代表者 大江 和彦
(順天堂大学大学院健康データサイエンス研究科 特任教授)

研究要旨

【目的】日本の医療現場で利用される多様な医療文書を FHIR で標準化するには、文書ごとの個別設計に起因する構造の不統一や保守性低下が課題となる。本研究は、日本の医療文書様式群を対象として、FHIR R4 document Bundle および FHIR Shorthand (FSH) を用いた統一的な文書構造化手法と実装ガイド生成エコシステムを検討することを目的とした。【方法】診療報酬関連文書を中心とする 80 種類の医療文書を収集し、記載項目と文書構造を分析した。文書の類型化、FHIR リソースへの対応付け、Composition.section を用いた階層構造設計、Observation 中心の記述方針を検討した。また、生成 AI を用いて PDF 様式から FHIR 実装ガイド (IG) を生成するプロセスを試行した。【結果】対象 80 文書から 165 項目を抽出し、22 カテゴリおよび 9 類型に整理した。FHIR document Bundle を基本とし、共通 Bundle Profile、共通 Composition Profile、類型別 Composition Profile、個別文書 Profile からなる階層的設計を提案した。さらに、主要リソース以外を Observation に集約する記述方針、文書構造を保持する section 階層化ルール、生成 AI を活用した FSH および IG 生成手順を整理した。【結論】提案手法により、日本の多様な医療文書に共通する情報構造を抽出し、FHIR 構造化における保守性、再利用性、文書横断検索性および将来的な AI 活用可能性の向上が期待される。医療文書 FHIR 構造化の基盤設計として有用であることが示唆された。

<研究分担者>

横田慎一郎：国立大学法人千葉大学
大学院看護学研究院・教授
土井俊祐：国立大学法人千葉大学
医学部附属病院・特任講師
三谷知広：国立大学法人東京大学
医学部附属病院・助教

A. 研究目的

近年、電子カルテ情報の標準化および医療機関間連携の高度化を目的として、HL7 FHIR (Fast Healthcare Interoperability Resources) の活用が急速に進展している。特に FHIR R4 は、国際的な医療情報交換標準として広く採用されつつあり、日本国内においても JP-Core 実装ガイドを中心として標準化が推進されている。

一方、日本の医療現場では依然として紙様式由来の医療文書が多数存在し、これらは診療報酬制度、地域連携、訪問看護、栄養管理、退院支援、介護連携など、多様な医療ワークフローにおいて利用されている。

しかし、多くの医療文書は人間可読性を主目的として設計されており、情報構造の統一性や意味論の一貫性が十分に整理されていない。そのため、FHIR による機械可読な情報構造へ変換する際には、文書ごとに個別設計が必要となり、Profile 数の増大、section 構造の不統一、FHIR Resource 責務の混乱などの問題が生じやすい。

特に、看護関連文書や在宅医療関連文書では、患者状態、ADL、家族支援、医療処置、栄養管理、看護目標、指示事項などが複雑に混在して記載されており、単純な FHIR Resource への対応付けだけでは十分な情報構造の整合性を確保できない。

さらに、FHIR Implementation Guide (IG) を長期的に保守運用する観点では、

文書ごとに独立した Profile 設計を行う方法では、将来的な拡張や保守性に重大な課題が生じる。

そこで本研究では、日本の医療文書様式群を対象として、FHIR R4 document Bundle を用いた統一的な医療文書構造化手法を検討した。特に、FHIR Shorthand (FSH) を用いて文書横断的な共通構造を抽出し、共通 Bundle Profile、共通 Composition Profile、カテゴリー別 Composition Profile、個別文書 Profile という階層的 Profile 設計手法を提案することを目的とする。

B. 研究方法

初年度の計画

① 医療文書の FHIR 規格化の観点からの類型化の検討：類型として、患者情報提供（共有）、指示依頼文書、自機関での医療計画書、他機関への報告書、サマリー文書、法令等届出文書、文書の形態を伴わない情報共有データ、などが想定される。各種ユースケースや文書様式を通じて、FHIR ドキュメント化した場合の構造の共通性や共通情報の特性などの観点から分析分類し、上記のような類型化を行う。

1) 対象文書の収集

診療報酬関連様式、在宅医療関連文書、訪問看護関連文書、退院支援関連文書、栄養管理関連文書、リハビリ関連文書など、日本国内で実運用されている多数の医療文書様式を収集した。

2) 文書の類型化と FHIR 構造の検討

各文書の記載項目の構成を調査し、文書の用途や伝えたい情報に応じた類型化を行

った。そしてその類型ごとに、記載が必要となる FHIR リソースを整理した。

3) FHIR 記述における基本的な記述構造の検討

前項の検討にもとづいて、すべての文書をカバーできると考えられる、FHIR 記述における基本的な記述構造を検討した。

FHIR R4 document Bundle を基本構造として採用し、すべての文書について以下を基本構成要素とする前提で、各記載項目を共通的に FHIR で記述できる方法を検討した。

- Bundle.type = document
- Composition resource
- Patient resource

4) 医療文書から生成 AI により自動的に FHIR 実装ガイドを生成するプロセスの検討

今後、多様な医療文書を効率的に FHIR 仕様に落とし込み、その実装ガイドを作成するには、生成 AI を活用した効率的な FHIR 実装ガイド (IG) 作成手法の確立が必要である。本研究では、まず代表的なコーディング支援生成 AI である Anthropic 社の Claude Code を使用して、PDF 形式の医療文書様式ファイルから記載項目一覧エクセル (CSV 形式ファイル) を生成し、さらにそれをもとに IG のソースファイルとなる FSH (FHIR ShortHand) 言語で記述された構造定義ファイル (Structure Definition) を生成して、それを既存の専用ツールで IG に変換するという一連のプロセスを試行した。

C. 結果

1) 対象文書の収集

対象文書は医療保険の診療報酬点数対象となる医療文書を診療報酬点数規則中からリストアップし、その電子版である PDF 形式を収集した。

次に、これらを個別文書単位に分割した上で、文書タイトル、記載項目、見出し構造を抽出した。

対象文書数は 80 文書、記載項目は 165 項目であった。別紙表 1 に今回の対象文書一覧、表 2 に 165 項目とそれが収載されている様式の数などの整理表を示す。

また、表 3 に 165 項目をカテゴリに整理し、22 カテゴリに分類したクロスリファレンス表を示す。この表には各カテゴリに対応する FHIR リソースタイプを併記した。

2) 文書の類型化と FHIR 構造の検討

80 の文書の 165 項目の収載情報の整理結果をもとに、これらの医療文書を用途や記載されている項目の種類などの観点、および FHIR 文書を作成するときの構造の違いなどの観点から、10 種類以下の類型に整理することを目的として分析し、その特徴の要点を整理した。

その結果、以下の表 4 のように 9 の類型に整理できると判断した。

類型	主な用途	主な記載項目
1. 証明・診療情報提供型	退院証明、診療情報提供、学校・市町村・介護施設向け情報提供	患者情報、医療機関、傷病名、経過、転帰、提供先
2. 入院・退院・療養計画型	入院診療計画、退院支援計画、在宅療養移行計画	入院退院日、課題、支援計画、目標、見直し予定
3. 指示書・在宅医療型	訪問看護指示、特別訪問看護指示、喀痰吸引指示	指示期間、処置、薬剤、医療機器、緊急連絡先
4. 評価・アセスメント型	神経学的検査、せん妄リスク、DIEPSS、褥瘡評価	評価日、スコア、身体所見、リスク、症状
5. リハビリ・ADL・生活機能型	リハ計画、リハ総合実施計画、目標設定支援	ADL、FIM/Barthel、運動機能、活動・参加
6. 栄養・口腔・看護サマリー型	栄養治療計画、看護及び栄養管理情報	栄養状態、食事形態、嚥下、口腔ケア、看護問題、ADL
7. 精神科・認知症・包括支援型	精神科計画、認知症療養計画、総合支援計画	精神症状、認知機能、服薬、家族支援、社会資源
8. 説明・同意型	手術、輸血、血漿製剤、地域包括診療等の説明・同意	説明内容、同意、署名、代諾者、対象処置
9. 報告・届出・実績管理型	実績報告、業務日誌、届出添付、薬剤報告	実施状況、報告年月日、職員体制、算定要件、日誌

表 4. 医療文書の 9 類型

これにもとづき 80 の医療文書を類型に割り当て、FHIR 構造化の方針をまとめた

結果を別紙表 4 に示す。

3) FHIR 記述における基本的な記述構造の検討

3-1) Bundle 構造

FHIR document Bundle を用いる。

```
FSH>
Profile: JP_Bundle_xxx_Document
Parent: Bundle
* type = #document
```

Bundle.entry には最低限以下を含める。

Composition
Patient

必要に応じて：

Observation
Condition
AllergyIntolerance
Procedure
Device
DeviceUseStatement
MedicationStatement
MedicationRequest
NutritionOrder
CarePlan
Goal
RelatedPerson
Organization
Practitioner
PractitionerRole
Encounter
などを含める。

3-2) Composition 設計原則

section = 文書構造

Composition.section は、医療文書の見出し・記載欄・章立て構造を表す。

- 紙様式のグループ化
- 枠囲み

- サブ項目
- 表構造

を FHIR section 階層に対応づける。

3-3) section 階層化ルール

紙様式上でグループ化されている欄は、必ず階層 section として表現する。

例：



3-4) 文書横断的な名称統一

3-4-1) section 名・slice 名・title

同じ意味を持つ項目は、文書が異なっても必ず同じ section 名・slice 名・title を使用する。

共通 section 名

以下を優先使用する。

意味	section 名
患者基本情報	patientInfo
入退院情報	hospitalization
診断・既往歴	diagnoses
看護サマリー	nursingSummary
家族・支援体制	familySupport
生活状況・ADL	adlSection
精神状態	mentalStatus
運動・感覚機能障害	functionalDisability
医療処置・医療機器	medicalProcedure
服薬管理	medicationManagement

意味	section 名
栄養管理	nutritionManagement
看護計画	nursingPlan
看護目標	nursingGoal
指示事項	instructionSection
緊急連絡先	emergencyContact

3-4-2) section.code

可能な限り LOINC を使用する。

LOINC が適切でない場合：

- SNOMED CT
- ローカル CodeSystem

を用いる。

コードが不明な場合でも：

```
FSH>
* section[x].code = $loinc#xxxx-x
```

形式で必ず記述するように試みる。

3-4-3) section.text

各 section には最低限：

```
FSH>
* section[x].text.status = #generated
```

を付加し、text を設定する。

3-4-4) section.entry 型制約

entry には必ず FHIR リソース型制約を付与する。

例：

```
FSH>
* section[diagnoses].entry only
Reference(Condition or AllergyIntolerance)

* section[adlSection].entry only
Reference(Observation)

* section[medicalProcedure].entry only
Reference(
  Procedure or
  Device or
  DeviceUseStatement
```

```
)

* section[nursingPlan].entry only
Reference(CarePlan)

* section[nursingGoal].entry only
Reference(Goal)
```

3-4-5) section slicing

section は FHIR slicing で定義する。

```
FSH>
* section ^slicing.discriminator.type = #value
* section ^slicing.discriminator.path = "code"
* section ^slicing.rules = #open
```

3-5) Composition.section.section を優先
文書構造表現には：

Composition.section.section.section...

を優先使用する。

Composition を再帰的に entry 参照する設計は原則使用しない。

3-6) リソース化の原則

各記載項目をリソース化する方法として2種類の原則案が考えられた。1つ目の案は可能な限り記載項目に意味に整合のとれる FHIR リソースタイプを使用する。2つ目の案は、可能な限り少ない FHIR リソースタイプに集約することで、リソース選択の困難さ、恣意性を排除する方法である。

案1：記載項目に意味に整合のとれる

FHIR リソースタイプを使用

まず1つ目の案では、以下のように原則を整理することが提案された。

i) Observation

ADL

認知症自立度

日常生活自立度

バイタル

栄養評価

食事摂取量

- 睡眠
- 精神状態
- 疼痛
- 排泄状況
- ii) Procedure
 - 点滴
 - 吸引
 - 創傷処置
 - ストーマ処置
 - 褥瘡処置
 - 呼吸管理
 - カテーテル管理
- iii) Device/DeviceUseStatement
 - 酸素
 - 人工呼吸器
 - PEG
 - PICC
 - CVC
 - 膀胱留置カテーテル
 - ストーマ
 - 補聴器
 - 車椅子
- iv) 栄養関連
 - Observation
 - NutritionOrder
 - を組み合わせる。
- v) ケア計画・目標
 - 目標 → Goal
 - 計画 → CarePlan

案2：可能な限り少ない FHIR リソースタイプに集約

案1ではあまりにも多様な記載項目が各文書に存在するため、すべての記載項目を適切な FHIR リソースの要素に対応させるのは労力が多く記述方式が統一されない上に、データを利用する側からみればどの項目がどのリソースのどの要素なのか分から

ないためデータ取得方式を定められない。
このような理由から、主要なリソースに明確に対応づけられる記載項

Observation.code は同一にし、
Observation.value[x] の型も統一する目以外は、すべて Observation にするという案が考えられた。

i) 原則：FHIR として意味が明確に定義されている主要リソースだけはそのまま使い、それ以外は Observation に集約する。

主要リソース：

文書項目	推奨リソース
患者情報	Patient
医療者	Practitioner
医療機関	Organization
傷病名	Condition
アレルギー	AllergyIntolerance
投薬	MedicationStatement / MedicationRequest
看護計画	CarePlan
看護目標	Goal
処置実施記録	Procedure
医療機器	Device / DeviceUseStatement
上記以外	Observation

ii) Observation により記述する記載項目では Observation.code として記載項目コード表を作成する（例：

JPDocumentItemCodeSystem）。

例：

患者希望：patient-hope、家族の希望：family-hope、介護状況：caregiver-status
家族構成：family-structure、口腔ケア：oral-care、移動方法：adl-transfer、など。

3-7) Observation リソースへの値記述方法
背景：同じ意味を持つ記載項目であっても、文書によって、選択肢タイプ（コード型）であったり、自由記述テキストタイプであったり、あるいは選択肢タイプではあ

るが「その他」という選択肢があって「その他」の場合には追加でテキストを書くタイプであったり、異なるデータタイプとなる。これらは、同じ意味をもつ記載項目なので、ひとつの同じ Observation.code にしたい。

そこで、同じ意味を持つ記載項目はその記載データ形式にかかわらず、

Observation.code は同一にし

Observation.value[x] の型も統一する
とする。

一方、patient-hope → valueString、body-weight → valueQuantity、のように決めていても、実際の記載項目では、「体重:変わりなし」のような変則的な記載がありえる。そのため、すべての

Observation.value[x]は最も柔軟な記載が可能な codeableConcept にしておき、数値ができる場合には

```
component[originalData].code =  
JPDocumentComponentCS#type-quantity  
* component[originalData].valueQuantity  
のように追加記述できるよう、
```

Observation.code : 文書項目の意味を一意に表す
Observation.valueCodeableConcept : 回答・状態・
記載内容を最も柔軟に正規化して保持
component[originalText] : 原文保持
component[typedValue] : 数値・日付・真偽値など、
機械処理しやすい型付き値
component[otherText] : その他欄の自由記述

とする案を作成した。

実際の出力例は以下となる。

```
Observation.code  
= JPDocumentItemCS#body-weight  
Observation.valueCodeableConcept  
= JPDocumentAnswerCS#unchanged "変わりなし"  
Observation.component[originalText]  
.code = JPDocumentComponentCS#original-text  
valueString = "体重: 変わりなし"  
Observation.component[typedValue]  
.code = JPDocumentComponentCS#typed-value  
valueQuantity = 50 kg // 数値化できる場合のみ
```

体重は以下を同じ Observation.code =
body-weight で扱える。

体重 50 kg
体重 変わりなし
体重 不明
体重 未測定
体重 減少傾向

データ取得側はまず Observation.code =
body-weight で拾い、標準的な状態は
valueCodeableConcept を見て、数値解析
が必要なら
component[typedValue].valueQuantity を
参照すればよい。

一方、この案では本来明確に数値である項目の数値検索が難しい。そこで、さらに普通は明確な数値型であるような項目で
Observation?code=body-weight&value-
quantity=50|kg といった検索もできるように
するため、そのような項目では
Observation?code=body-weight-quantity
をもう一つ Observation として並列記述する
案が考えられた。この案ではひとつの記載
項目にこの場合は2つの Observation が
出現しうることになってしまう欠点がある
ため、両者に関連付けが必要である。
そこで次の案を作成した。

・基本 Observation

```
Observation.code = body-weight  
Observation.valueCodeableConcept = measured  
component[originalText] = "体重 50kg"  
component[typedValue].valueQuantity = 50 kg
```

・検索用 Observation

```
Observation.code = body-weight-quantity  
Observation.valueQuantity = 50 kg
```

・2つの Observation の連結

```
* Observation[bodyWeightQuantity]  
.derivedFrom  
= Reference(Observation/bodyWeight)
```

以上をルール化すると

- ✓ 文書項目 Observation は必ず 1 つ作る
code = body-weight
- ✓ 型付き検索用 Observation は、機械処理可能な値がある場合だけ追加で作る
code = body-weight-quantity
- ✓ 検索用 Observation は必ず元 Observation を参照する
derivedFrom = 文書項目 Observation
- ✓ Composition.section.entry には原則として文書項目 Observation を入れる
検索用 Observation は Bundle には含めるが、section.entry に入れるかどうかは IG で明示する。

となる。

また、ひとつの記載項目の記入値が複数繰り返される場合の記述ルールを以下のとおりとした。

- ✓ 1 つの症状・1 つの値につき 1 Observation
- ✓ 同じ記載欄から生成された Observation 群には、共通の identifier を付ける
- ✓ 必要なら Observation.partOf または derivedFrom で元の文書項目 Observation に紐づける
- ✓ 表示順が重要な場合は、extension[itemOrder] のような順序拡張を用意する

3-8) 叙述記載の格納場所

ひとつのセクションにひとつの observation しかない記載欄の場合であって、その記載欄に自由記載欄しかない場合には、この自由記載データを section.text に格納するのか、section.entry の参照する observation の valueString に格納するのかを決めておく必要がある。

Observation の valueString に格納することとし、section.text は表示用・原文再現用の narrative と位置づけ、データ取得対象にはしない方針とした。

例：

```
Composition.section.title =  
  "看護上の問題"  
Composition.section.text =  
  "看護上の問題：夜間せん妄あり"  
Composition.section.entry =  
  Observation/nursing-problem-1  
Observation.code =  
  JPDocumentItemCS#nursing-problem  
Observation.valueCodeableConcept.text =  
  "夜間せん妄あり"  
component[originalText].valueString =  
  "看護上の問題：夜間せん妄あり"
```

また、ここまでのルールだと、多くのセクションで、section.text と

Observation.valueString の両方に、似ているが異なるテキストが格納されるリスクがあり、利用する側はどちらをどのように使い分けるのかが判断しにくい。そこで、

- ✓ データ本体は必ず Observation 側に置く。
 - ✓ section.text は原則として空、または自動生成 narrative に限定する。
- とした。

以上のモデルを「別添資料 1 基本共通情報モデル仕様書案」をして、生成 AI の支援によりとりまとめ文書として作成した。

4) 医療文書から生成 AI により自動的に FHIR 実装ガイドを生成するプロセスの検討

本検討は、研究代表者が主宰する学会の FHIR 関連研究会において生成 AI によるプログラムコーディングを試行する知見を有する（株）ファインデックスに外注する

ことで共同で実施した。3) の分析と時期的に並行して実施したため、3) の成果を導入していない。

検討・試行は以下の5つのステップで実施した。

- 1) 様式分析：PDF から項目抽出、TSV 整理・仮マッピング（生成 AI+人による確認）
- 2) FHIR マッピング：FHIR 要素マッピング確定、リソース選択・VS 検討（生成 AI+人による確認）
- 3) FSH 実装：プロファイル・CS・VS サンプル実装（生成 AI）
- 4) ビルド検証：SUSHI 0 エラー確認、エラー修正ループ（生成 AI）
- 5) IG 作成：IG Publisher HTML 生成、qa.html 確認（人による作業）

詳細の説明は別添資料2、3に示す。

D. 考察

本研究では、日本の多様な医療文書様式に対して、FHIR R4 document Bundle を用いた統一的構造化手法を提案するため、FHIR リソース構造と医療文書のユースケース、含まれる情報の共通性の観点から類型に分類して構造化を行うべく分析と整理を行った。

従来、医療文書 FHIR 構造化では、個別文書単位で Profile 設計が行われることが多く、Profile 数の増大や section 構造の不統一が問題となりやすかった。

本研究では、文書横断的に共通する意味構造を抽出し、共通 Bundle／共通 Composition／カテゴリーComposition／個別 Profile という階層構造を導入することで、これらの問題を軽減できる可能性があると考えられた。

特に重要であった点は、FHIR Resource を単純に文書項目へ対応付けるのではなく、「意味論的責務」に基づいて Resource 利用方針を統一した点である。

また、Composition.section 構造を積極的に利用した点も重要である。

多くの日本の医療文書では、紙様式由来の複雑な階層構造が存在するが、

Composition.section.section を用いることで、文書本来の意味構造を維持したまま FHIR 構造化できると考えられた。

さらに、section 名統一により、将来的な UI 自動生成や Questionnaire 自動生成への応用可能性も高い。

一方、本研究にはいくつかの課題も存在する。

第一に、日本独自の制度依存項目については、FHIR 標準 Resource のみでは十分表現できない場合がある。

第二に、LOINC や SNOMED CT 等の標準用語へのマッピングについては、今後さらなる詳細検討が必要である。

第三に、section.text と section.entry の責務分離についても、実運用に応じた設計指針整備が必要である。

今後は、

Questionnaire/QuestionnaireResponse との対応付けを行うか、FHIR Search 最適化、生成 AI による自動 FSH 生成、SS-MIX2 連携、HL7 CDA 連携などへの応用を検討する予定である。

E. 結論

本研究では、日本の多様な医療文書様式 80 件を対象として、FHIR R4 document Bundle および FHIR Shorthand (FSH) による統一的構造化手法を検討した。その結果、文書横断的に共通する情報構造を抽出

し、共通 Bundle Profile、共通 Composition Profile と、記載項目のリソースによる効率的記述方針を示した。さらに、section 構造や FHIR Resource 責務を統一することで、保守性、再利用性、文書横断検索性、UI 共通化、将来的な AI 活用可能性を向上できることを示した。本手法は、日本の医療文書 FHIR 構造化の基盤設計として有用であると考えられるが、さらなる分析と検討が必要である。

F. 健康危険情報

なし

G. 研究発表

1. 論文発表

なし

2. 学会発表

なし

H. 知的財産権の出願・登録状況

なし