

**MIJS データ標準規格
説明資料
(Ver1.0)**

作成：2007年11月29日（木）



メイド・イン・ジャパン・ソフトウェア・コンソーシアム

■概要

1. 目的

カスタムメイド、パッケージソフトを問わず、複数のアプリケーションを同時利用する形態が増加し、アプリケーション間の連携が重要課題となっている。これまで、企業間取引という外部連携のための標準化の取り組みが多かったが、企業内のアプリケーション連携という内部連携は立ち遅れていた。そこで、企業内のアプリケーションが連携できるためのデータ標準規格を制定し、規格に準ずることでアプリケーション間のデータ連携ができるようにする。

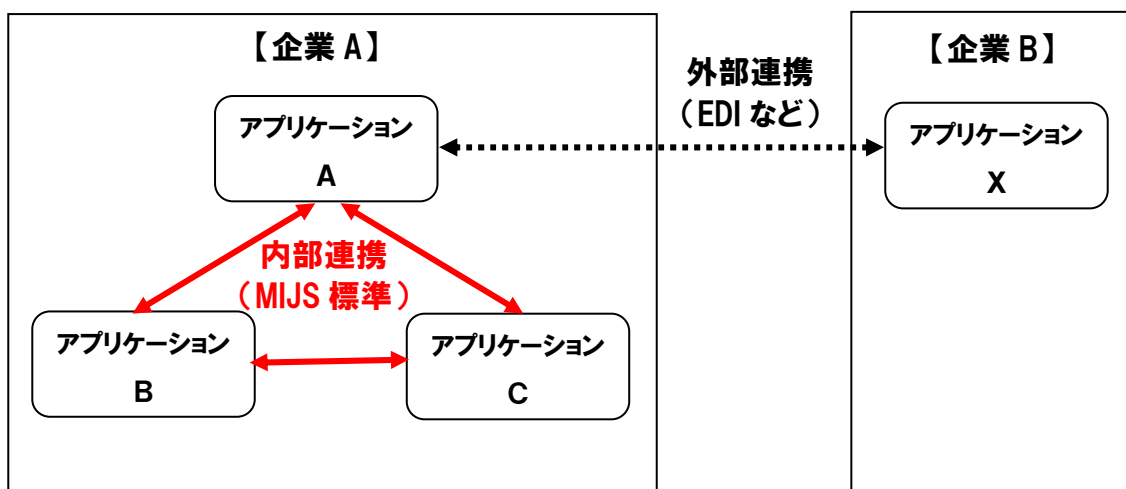


図 1：外部連携と内部連携の関係

2. 対象データ

アプリケーション間で連携されるデータには、日々の業務で増えてゆく「トランザクション」とあらかじめ登録しておく「マスタ」の2種類がある。

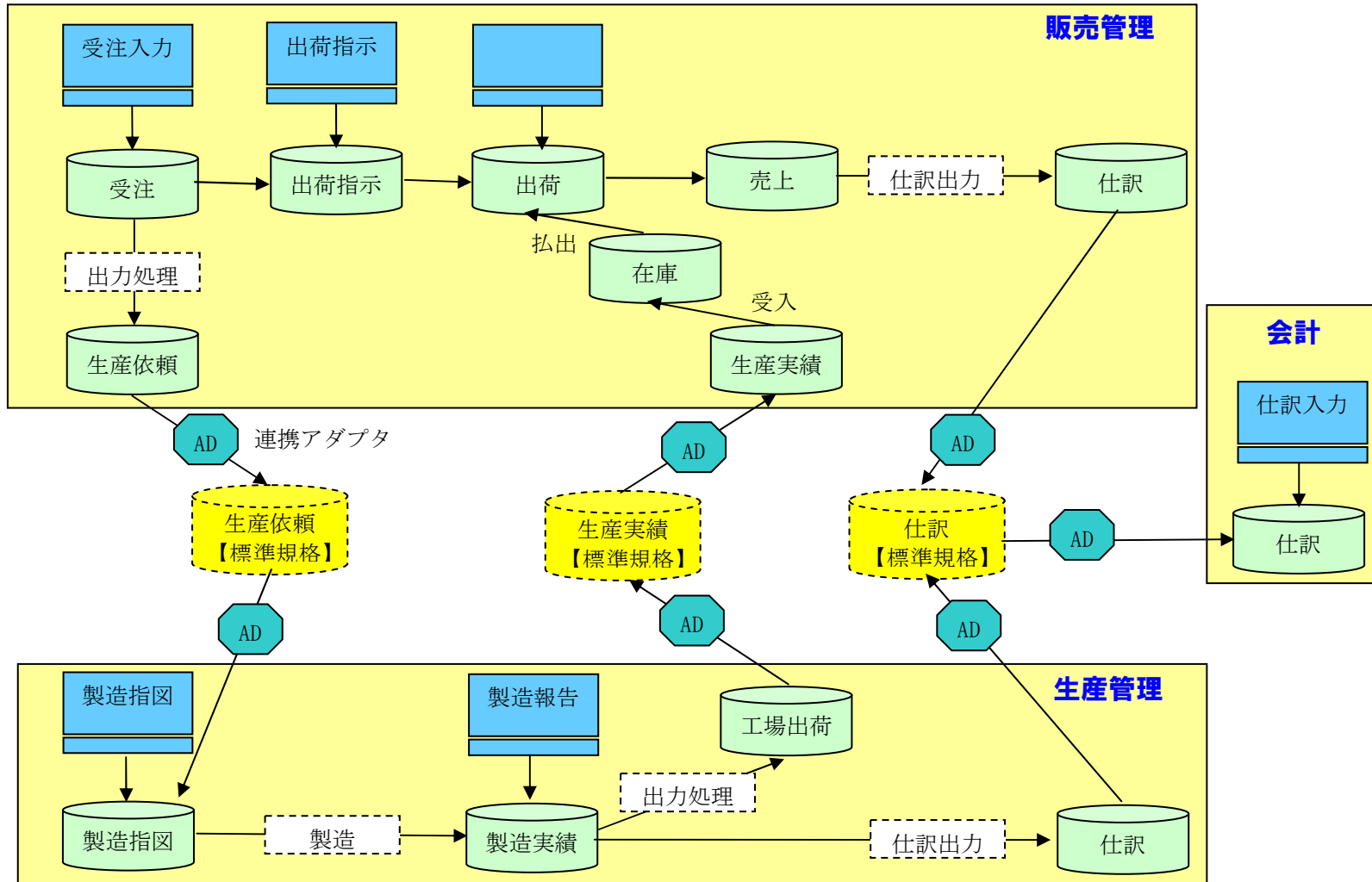
本規格はその両方を対象とし、多くのアプリケーションで共通に利用するデータから順番に規格化する。第一弾として次の5つのデータ標準を Ver1.0 とし、以降、順に対象範囲を広げて行く。

第一弾 (Ver1.0)	第二弾 (予定)
【トランザクション】 <ul style="list-style-type: none">・生産依頼データ・生産実績データ・仕訳データ 【マスタ】 <ul style="list-style-type: none">・社員マスタ・部門マスタ	【トランザクション】 <ul style="list-style-type: none">・見込データ・受注データ・売上データ 【マスタ】 <ul style="list-style-type: none">・取引先マスタ・商品マスタ

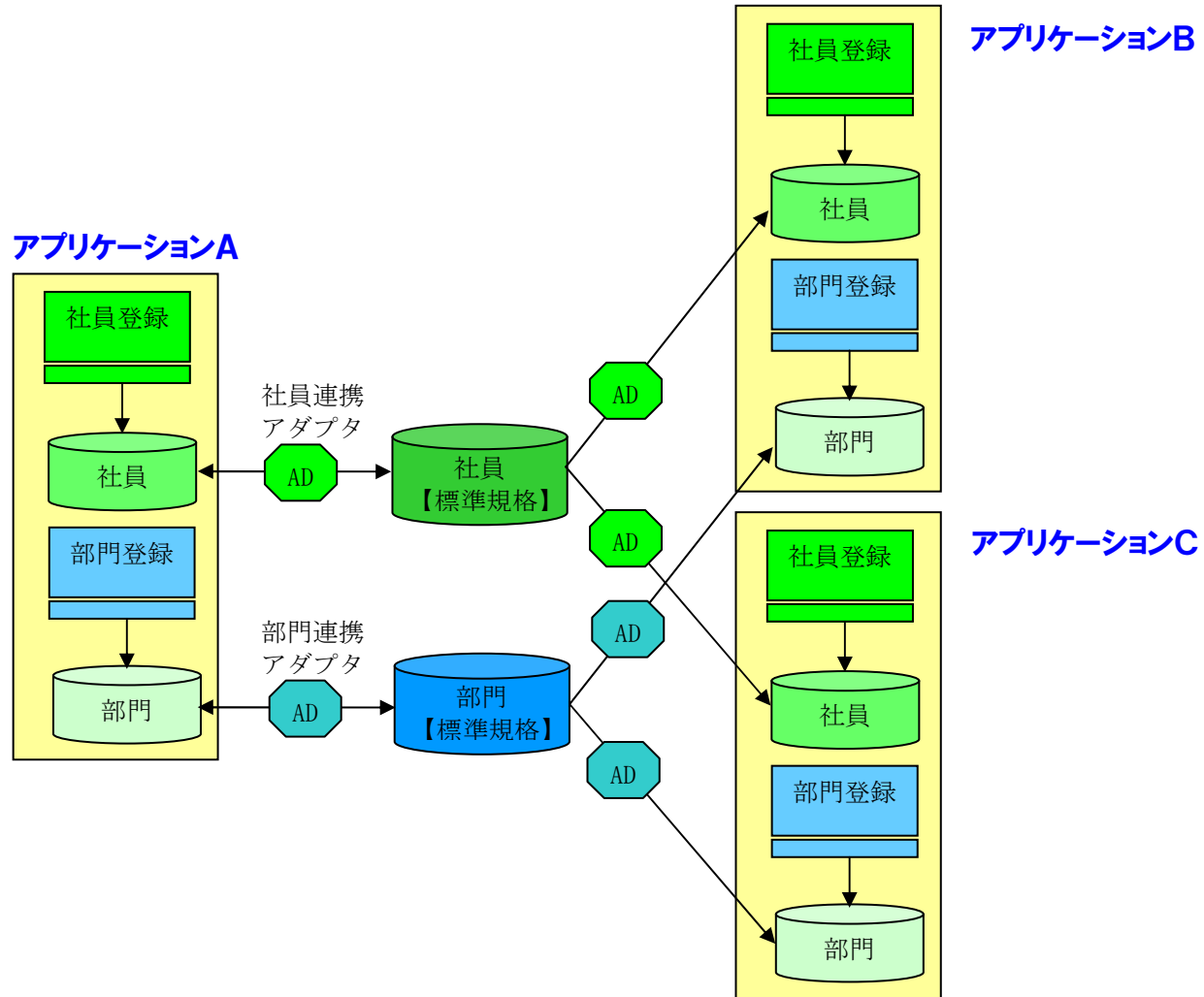
表 1：標準化対象データ

図2：MIJSデータ標準規格（第一弾）の連携イメージ図

(1) トランザクションデータ：「生産依頼」「生産実績」「仕訳」



(2) マスタデータ : 「社員」「部門」



3. 規格化の対象

アプリケーション間でデータを連携するために取り決める仕様には、①データ構造、②データインターフェース、③連携プロセスの3種類がある。

データ構造とはデータ項目の定義情報である。たとえば社員データの場合はそこに含まれる社員コード、姓、名、部門コードなどの項目をデータ型、サイズなども含めて定義する。データインターフェースとは、データにアクセスするためのインターフェース仕様に関するもので XML 定義などが含まれる。連携プロセスは、データ送信のタイミング（自動/手動）や送信方向（単方向/双方向）などを定義するものである。

本規格 Ver1.0 では、この中から最も基本となるデータ構造に関する標準仕様を定める。データインターフェースや連携プロセスは規定せず、どのような手段を用いても良いものとする。ただし、連携の容易性を考慮してデータ構造の標準規格が定まった時点で規格化を検討する予定。

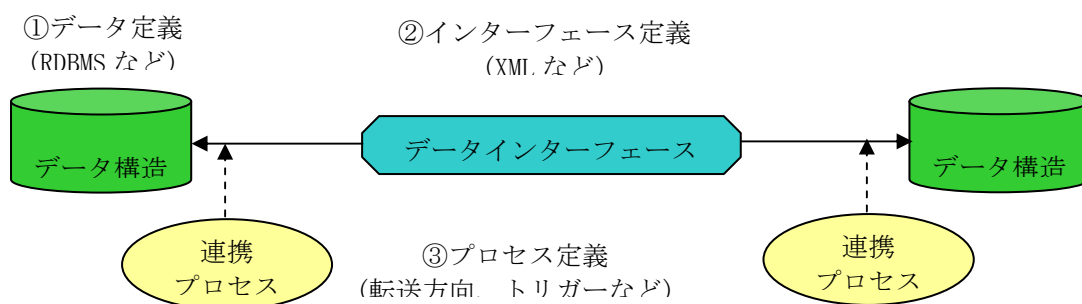


図 3 : 3 種類の規格化対象

4. 規格の利用形態

(1) 新規アプリケーション作成時

現状はデータの標準規格がないため、新規にアプリケーションを作成する際にデータ構造がばらばらになっている。今後作成するアプリケーションは、MIJS 標準規格に準拠することにより、他のアプリケーションと自在にデータ連携できることになる。

(2) 既存アプリケーションの連携

既存アプリケーションは、それぞれ独自のデータ構造を持っている。そのため、アプリケーションごとにピア・トゥ・ピアでデータ連携を行うと、その組合せ数だけ連携の仕組みを作りこむ必要が生ずる。

その課題を解決する手段が MIJS 標準規格を介した連携である。各アプリケーションが MIJS 標準規格に対応する連携アダプタを1つ用意することにより、MIJS 標準に対応するだけで複数アプリケーションとの連携を実現することができる。

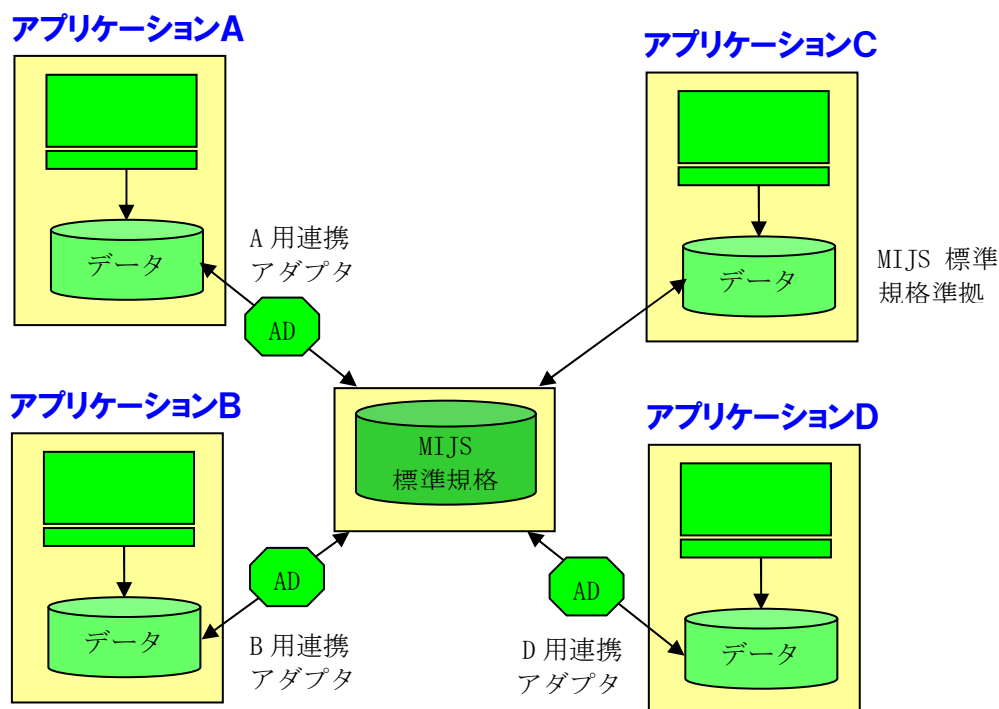


図 4：標準規格を利用した n 対 n 連携

■MIJS データ標準規格

1. 定義範囲

本規格で定義するデータ構造は、一般的なデータのエンティティ構造のうち次のような事項である。

- ・エンティティ (テーブル)・・・(例) 社員基本
- ・アトリビュート (項目)・・・(例) 姓
- ・データ型・・・・・・・・(例) NVARCHAR
- ・データサイズ・・・・・・・・(例) 10

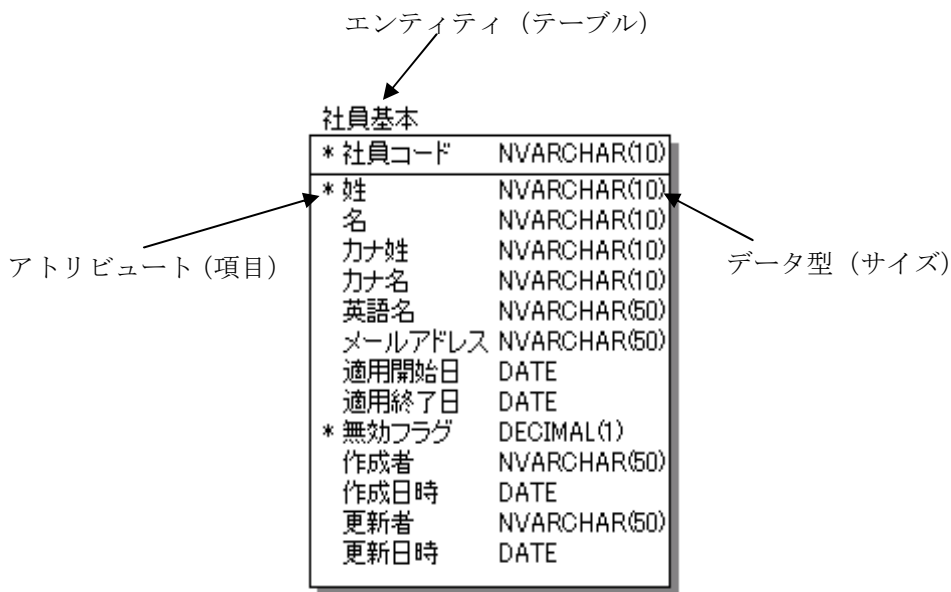


図 5：標準規格で定義するデータ項目

2. 標準 SQL 規格準拠

データ型は ISO の SQL 標準規格 (SQL99) に準拠して定義している。RDBMS により次のような関連となっているので、適用する RDBMS に対応したデータ型を使用する。

型	SQL 標準	データタイプ	Oracle	SQL Server	DB2
文字列	NVARCHAR	可変長文字列 (各国語文字)	NVARCHAR2	NVARCHAR	VARCHAR
数値	DECIMAL	真数 (有効数字と位取り)	NUMBER	DECIMAL	DECIMAL
日付	DATE	日付	DATE	DATETIME	DATE

表 2：ISO 標準 SQL と Oracle/SQL Server/DB2 のデータ型対比 (例)

■データ構造の説明

データ構造については、別紙「MIJS 標準規格データ定義書」に詳細を示す。
以下、データ構造について説明する

1. 基本的な考え方

(1) 細分化して値を持つ

標準データは、基本的に細分化してデータを持つ。たとえば社員の名前をデータとして管理する場合、「名前」という1つの値で保有するのではなく、「姓」と「名」で分けてデータを持つ。これは、細分化していた方が連携の際のデータ変換が容易になるためである。

たとえば連携先のアプリケーションが「姓」＋「名」に分割して値を保有していた場合、標準データが「名前」という統合した項目で管理しているとデータをどこで分割してよいか判断がつかない。逆に連携先が「名前」で標準データが「姓」と「名」であれば、「姓」と「名」を連結してデータ転送すればよい。

(2) ログ情報

各エンティティには、「作成者」「作成日時」「更新者」「更新日時」という4つのログ項目を持つ。これはデータの作成・更新におけるログを記録し、アクセスの証跡を残すものである。このようなデータ上にログを記録する方法のほかに、データアクセスに関するログデータを別エンティティに保存する方法もあるが、必要に応じて併用することを想定している（標準 Ver1.0 では規定していない）。

2. マスタデータ

(1) 社員データの基本と拡張、履歴の分離

社員データには、業務の担当者（指示者、実行者など）としての項目と社員の人事情報の項目がある。また、人事情報には、現在情報と履歴情報がある。さらに人事情報には口座情報およびログイン情報という機密情報がある。これらの役割別にエンティティを分割し、社員基本と社員拡張、社員履歴、社員口座、そしてログインの5つのエンティティに分割している。ほとんどのアプリケーションは、業務の担当者特定の目的で社員データを保有しているため、社員基本データの連携だけで対応できる。

(2) 社員の複数部門兼務対応

社員と部門が多対多の関係、つまり1人の社員が複数部門を兼務できる構造としている。そのため、「社員基本」と「部門」の間に連結用のエンティティ「社員_部門」をはさんでいる。

(3) 部門の組織変更対応

部門は、組織変更を考慮して履歴型でデータを管理している。そのためアイ

デンティファイアー（主キー）に適用開始日という項目が加わっている。

(4) 部門の階層化

部門は階層数の制限のない階層構造としている。そのため、上位部門と自部門の階層という項目が管理されている。

(5) 論理削除

社員基本データおよび部門データは、論理削除に対応している。そのため、それぞれのエンティティ内に「無効フラグ」というアトリビュートを用意している。

3. トランザクションデータ

(1) 仕訳データ

仕訳データは、伝票と明細行のほかに貸借区分もアイデンティファイアー（主キー）項目としている。つまり借方で1レコード、貸方で1レコードという形態でデータを持つ構造である。

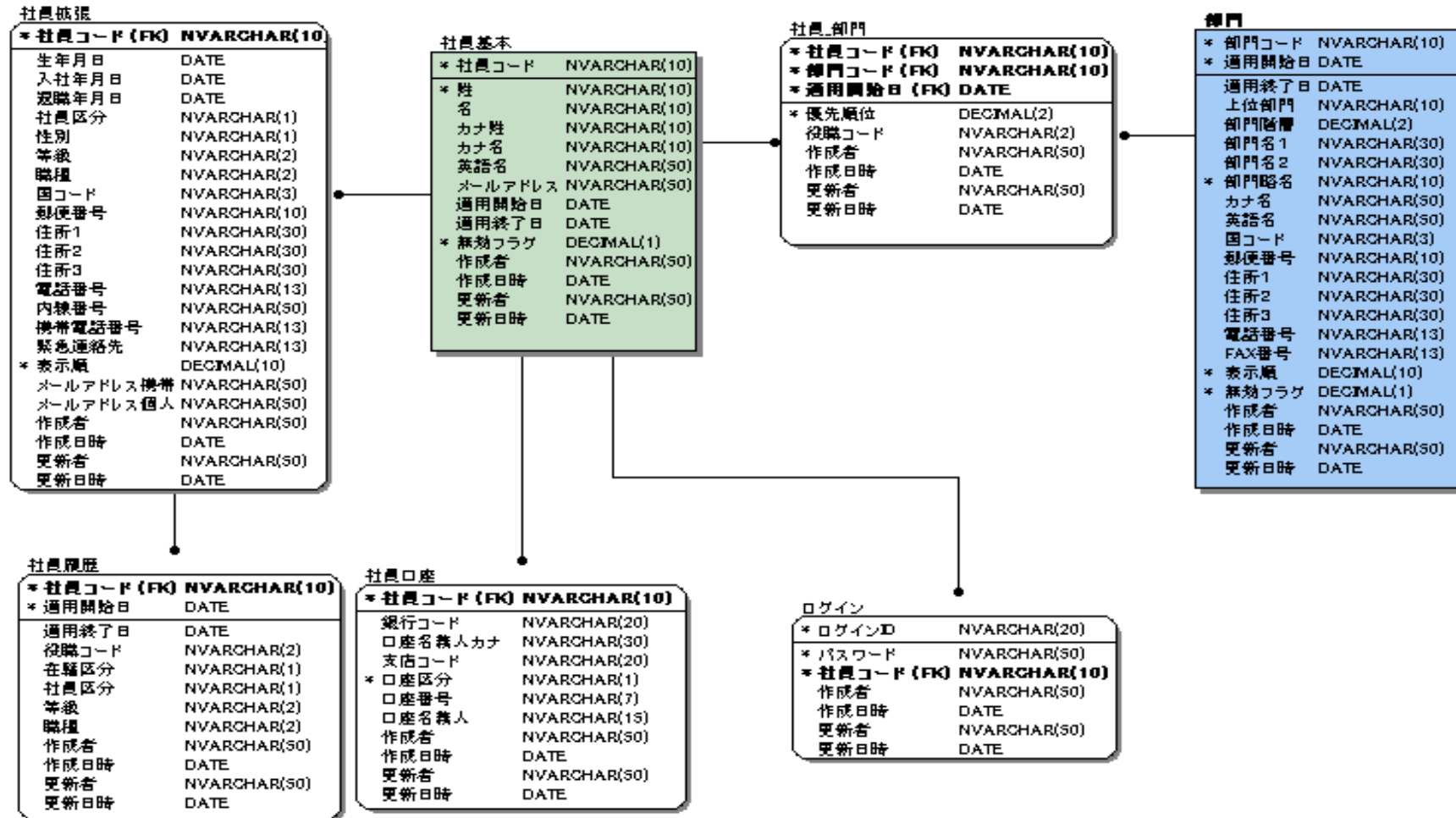
仕訳データの構造としては、明細行ごと（借方と貸方）で1レコードという形式も多い。そのようなアプリケーションから標準規格にデータ連携する際は、レコード数が1対2になるので、借方と貸方の2回に分けてデータ更新する必要がある。

(2) トランザクションデータで使用するマスターデータ

トランザクションデータで使用するマスターデータ（勘定科目コードや品目コードなど）は、基本的に各アプリケーションで共通化されているものと想定している。本バージョンで社員と部門を標準設定したが、それ以外のマスターデータについては以降のバージョンで順次標準化するものとしている。

図6：MIJS データ標準規格のエンティティ構造

(1) マスタデータ（社員関連および部門関連）



(2) トランザクションデータ (仕訳、生産依頼・生産報告)

仕訳

* 仕訳伝票番号	NVARCHAR(10)
* 明細行番号	DECIMAL(3)
* 貸借区分	NVARCHAR(1)
会社コード	NVARCHAR(5)
* 伝票日付	DATE
入力日付	DATE
* 赤伝フラグ	DECIMAL(1)
取込元伝票番号	NVARCHAR(10)
* 決算仕訳区分	NVARCHAR(1)
* 勘定科目コード	NVARCHAR(7)
補助科目コード	NVARCHAR(10)
* 部門コード	NVARCHAR(10)
取引先区分	NVARCHAR(1)
取引先コード	NVARCHAR(10)
汎用セグメントコード1	NVARCHAR(10)
汎用セグメントコード2	NVARCHAR(10)
汎用セグメントコード3	NVARCHAR(10)
汎用セグメントコード4	NVARCHAR(10)
プロジェクトコード	NVARCHAR(20)
プロジェクト枝番	NVARCHAR(2)
* 仕訳金額	DECIMAL(12,2)
消費税区分コード	NVARCHAR(1)
* 消費税率コード	NVARCHAR(3)
* 消費税計算区分	NVARCHAR(1)
* 消費税金額	DECIMAL(12,2)
* 外貨フラグ	NVARCHAR(1)
通貨コード	NVARCHAR(3)
通貨レート	DECIMAL(4,2)
外貨換算仕訳金額	DECIMAL(12,2)
外貨換算消費税金額	DECIMAL(12,2)
摘要	NVARCHAR(50)
作成者	NVARCHAR(50)
作成日時	DATE
更新者	NVARCHAR(50)
更新日時	DATE

生産依頼

* 製造番号	NVARCHAR(13)
製造依頼日	DATE
* 依頼部門コード	NVARCHAR(10)
* 依頼社員コード	NVARCHAR(10)
品目コード	NVARCHAR(40)
製造指示数	DECIMAL(21,6)
完成指示日	DATE
納入場所コード	NVARCHAR(8)
プロジェクトコード	NVARCHAR(20)
プロジェクト枝番	NVARCHAR(2)
摘要	NVARCHAR(50)
作成者	NVARCHAR(50)
作成日時	DATE
更新者	NVARCHAR(50)
更新日時	DATE

生産報告

* 製造報告番号	NVARCHAR(13)
製造日	DATE
* 製造番号 (FK)	NVARCHAR(13)
* 製造部門コード	NVARCHAR(10)
* 製造報告社員コード	NVARCHAR(10)
納入場所コード	NVARCHAR(8)
プロジェクトコード	NVARCHAR(20)
プロジェクト枝番	NVARCHAR(2)
品目コード	NVARCHAR(40)
ロット番号	NVARCHAR(30)
製造数	DECIMAL(21,6)
原価金額	DECIMAL(12,2)
摘要	NVARCHAR(50)
* 赤伝フラグ	DECIMAL(1)
作成者	NVARCHAR(50)
作成日時	DATE
更新者	NVARCHAR(50)
更新日時	DATE

消費税率

* 消費税率コード	NVARCHAR(3)
* 消費税率名	NVARCHAR(10)
* 消費税率	DECIMAL(5,3)
* 地方消費税率	DECIMAL(5,3)
* 旧消費税率	DECIMAL(5,3)
* 旧地方消費税率	DECIMAL(5,3)
* 新消費税率	DECIMAL(5,3)
* 新地方消費税率	DECIMAL(5,3)
消費税率決定日	DATE
作成者	NVARCHAR(50)
作成日時	DATE
更新者	NVARCHAR(50)
更新日時	DATE

消費税区分税率関連

* 消費税区分コード (FK)	NVARCHAR(1)
* 消費税率コード (FK)	NVARCHAR(3)

消費税区分

* 消費税区分コード	NVARCHAR(1)
消費税区分名	NVARCHAR(15)
* 消費税科目区分	NVARCHAR(1)
* 無効フラグ	DECIMAL(1)
作成者	NVARCHAR(50)
作成日時	DATE
更新者	NVARCHAR(50)
更新日時	DATE