

# グローバル製造管理のためのデータ統合：XMLを用いたデータ収集とデータ翻訳・統合

Data Integration for Global Manufacturing Management: Data Collection, Translation, and Integration with XML

浪岡 保男\*

Yasuo NAMIOKA

グローバル生産においては、世界中に分散する製造拠点での製造実績情報を生産統括部門と全拠点で共有し、マーケットの変化や生産の異常発生に対して素早い管理アクションを起こすことが重要である。製造拠点の製造情報システムは、必ずしも標準化されておらず、こうした異種システム間のデータを有効に利用するためには、共通のデータ構造に共通のデータ・コードで一元管理する必要がある。そこで、XMLを用いて汎用に製造実績データを自動収集し、共通データに翻訳・統合する一手法を考案した。また、この手法に基づいてグローバル生産体制での製造マネージメントを支援するための製造情報管理システム(Global REALMICS†)を開発した。本システムを電子機器のグローバル製造品質管理に適用し、管理サイクルを従来の週次から日次に短縮した。

A data translation and integration method using XML for global manufacturing management is described in this paper. To manage worldwide manufacturing, one of the most important things is the efficiency of watching manufacturing bases' information about production and quality control, however, there are different information systems in the manufacturing bases generally. Therefore, the management information and control system for globally distributed manufacturing bases (Global REALMICS†) is developed with the data integration method. Global REALMICS has been applied to an electronic equipment manufacturing, and reduced the usual cycle time of the global quality control from weekly to daily.

## 1. まえがき

市場とモノづくりのグローバル化にともない、製造業各社では、複数の拠点での生産進捗や製造品質などの実績情報をグローバルに共有して、生産全体の効率をいっそう高める努力を進めている。筆者は、グローバルな生産体制での生産管理[1]および品質管理[2]などの生産マネージメントを支援するため、複数拠点の製造情報をXML(eXtensible Markup Language)[3][4]を用いて統合するシステム(Global

\*株式会社 東芝 生産技術センター

yasuo.namioka@toshiba.co.jp

† Global REALMICS は、株式会社 東芝の登録商標です。

REALMICS1)を開発した。本報告では、Global REALMICSのデータ統合に関するコンセプトについてデータ翻訳・統合手法を中心に述べる。さらに、電子機器のグローバル製造品質管理に適用した事例を簡単に紹介する。

## 2. グローバル製造品質管理と異種システム間の情報共有の課題

グローバル生産においては、各製造拠点の間で製造品質情報を共有し、この情報に基づいて的確な品質管理アクションを行うことが重要である。これは、不具合を検出した工程とこの不具合の原因を作った工程が、異なる拠点に属する場合が多いためである。

しかし、一般には、各製造拠点にある情報システムには差異があり、製造拠点間の製造情報を共有する際の障害となる。情報システムに差異が生じる背景としては、企業戦略的背景、時空間的背景、生産内容的背景などが考えられる。また、情報システムの差異は、大別すると次の3種類に分類することができる。

- システム構造：機能分担、ハードウェア・ミドルウェア・OS、システム開発に用いた計算機言語等の差異。
- 情報構造：データベース上の各種実績データの定義や、多次元データキューブの有無、定義、集計方法等の差異。
- 意味的構造：データ・コードの分類及びコード体系の差異や、データ・レコードにあるタイムスタンプと製造現場のワークフローとの対応関係等の差異。

こうした差異を吸収して、各拠点のデータを共通データ構造・データ定義に変換しなければ、製造情報を統括して分析可能な形で共有することは難しい。

## 3. 製造実績情報の共有方法

製造実績情報を共有するアプローチと Global REALMICSにおける共有方法について述べる。

### 3.1. 製造実績情報を共有するアプローチ

製造実績情報を共有するアプローチには、システム間のデータの差異の吸収と共有されるデータおよび情報の保持の2つの観点がある。

システム間のデータの差異を吸収するためのアプローチを大別すると次の3つがあげられる。

- リプレース：製造拠点の情報システムを標準システムに置き換える。
- 改良：製造拠点の情報システムに、標準的なデータ形式、データ・コード体系でデータを出力するインターフェースを加える。
- 翻訳・統合：製造拠点の情報システムに蓄積されている詳細データを、そのままのデータ形式及びデータ・コード体系で収集し、標準データ形式に翻訳して統合する。

実際の情報共有では、これら3つのアプローチを各製造拠点の状況に応じて適用するのが現実的である。Global REALMICSでは、既存システムの資産活用やシステム環境のダイナミックな変化への対応の柔軟性などの利点から、翻訳・統合アプローチを中心に、適宜、改良やリプレース・アプローチを取り入れて情報を共有する。

また、収集・統合されるデータおよび情報の保持の仕方には、分散型と集中型の2つのアプローチがある。

- 分散型：各拠点のシステムに適宜データを問い合わせデータ分析を行うアプローチ。常に最新のデータを用いた分析が可能である反面、各拠点の生産情報システムへの

不定期的な問い合わせが発生し、その負荷の予測は難しい。

- 集中型：各拠点のシステムから定期的にデータを収集・統合するアプローチ。統合されたデータを基にデータ分析を行う。データの同期方法の検討が必要であり、必ずしも最新のデータを用いた分析とはならない代わりに、各拠点の製造情報システムへの負荷は定期的であり、統合データへのアクセス負荷が大きく発生しても、各拠点のシステムへの影響は無い。

このうち、製造情報をエンドユーザが参照する際のサービスに要するレスポンス速度と拠点システムに予期しない負荷をかけることを防ぐなどの観点から、Global REALMICSでは、集中型のデータ保持アプローチを採用している。

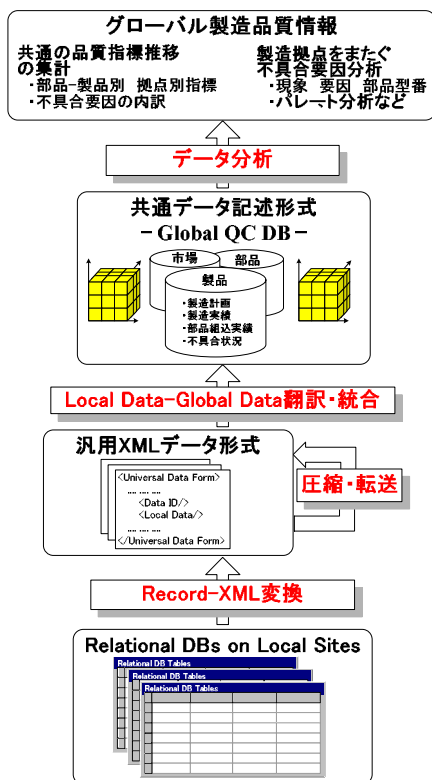


図 1 製造実績情報の共有方法

### 3.2. Global REALMICS の製造実績情報の共有方法

Global REALMICS における製造実績情報の共有方法を図 1 に示す。各製造拠点の製造の製造実績データベース(図では、Relational DBs on Local Sites)に格納されているデータを、Record-XML 変換、圧縮・転送、Local Data-Global Data 翻訳・統合、データ分析の 4 つの機能を経て、グローバル製造品質情報として活用できる形に変換する。

- Record-XML 変換：システム構造及び情報構造の差異を吸収しながら各種実績データを取得する。製造拠点のMES(Manufacturing Execution System)[5]などの製造実績データベースに、予め与えられる権限で接続し、更新されたデータを検索し、汎用 XML データ形式(Universal Data Format)のファイルに変換する。汎用データフォーマットは、任意のリレーションを持つデータを転送するための XML ベースのフォーマットである。

- 圧縮・転送：上記のデータファイルを圧縮して転送する。
- Local Data-Global Data 翻訳・統合：情報構造および意味的構造の差異を吸収し、共通のデータ形式に共通データ・コードを用いて統合する。汎用 XML データ形式のファイルを入力し、データの内容に応じてデータ翻訳を行う。グローバル製造品質データベース(Global QC DB)には、実績データと分析を効率化するための多次元データキューブが適宜蓄積される。

- データ分析：共通のデータ記述形式となった各種実績データや多次元データキューブを用いて、品質指標集計、パレート分析などのさまざまな側面からの分析を行う。

これらの機能により、製造拠点の情報システム間に差異が存在しても、共通のデータ記述形式の Global QC DB に製造データを統合して、製造実績情報の共有が可能となる。

以下では、データ翻訳・統合について述べる。

## 4. データ翻訳・統合

上述の集中型のアプローチで共有するデータを保持する際には、製造拠点の情報システム上の各種実績データ(以下、ローカルデータ)と Global QC DB 上のデータ(以下、グローバルデータ)との同期に着目する必要がある。ローカルデータは、不具合発生時に挿入し、後日、解析結果が後付されるなど、時々刻々と更新される生きたデータが多い。このため、同期の際には、各種実績データを過去にさかのぼって修正し、そのつど多次元データキューブを再計算するなどの複雑な処理が発生する。このため、データ翻訳・統合には次の機能が要求される。1.データ構造の変換、2.コピー、3.補填、4.数値演算や文字列操作、5.複数データの演算、6.データ・コードの変換、7.データ間の関連情報の付加、8.データ内容による手続きの選択、9.複数データに対する一括処理、など。

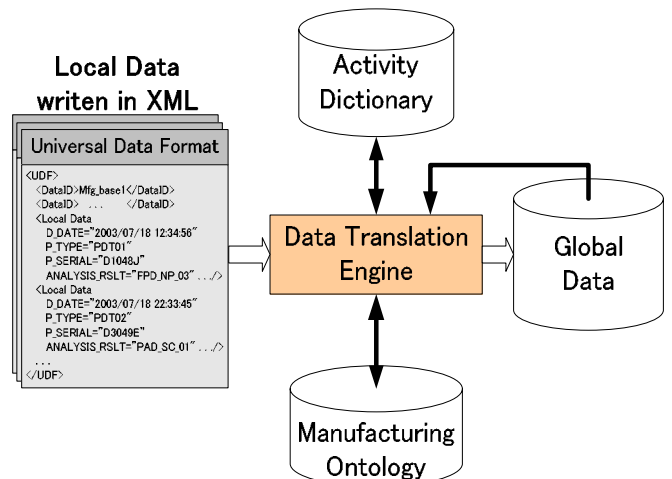


図 2 データ翻訳・統合のアーキテクチャ

### 4.1. データ翻訳・統合のアーキテクチャ

Global REALMICS のデータ翻訳・統合機能は、図 2 のアーキテクチャに基づいており、Data Translation Engine (DTE)を簡単化するために次の特徴を持つ。

- 多様なリレーションを汎用 XML データ形式(Universal Data Format)という単一の形式にして入力とする。
- XML の再帰性に着目したデータ操作のレンジを採用して

いる。  
DTEは、汎用XMLデータ形式のLocal Dataを入力とする。汎用XMLデータ形式は、簡単なメタデータ[6](図2では<DataID>)とデータの実態(図2では<LocalData>)からなる。DTEは、データ翻訳の手順が記述されたActivity Dictionary及び、Manufacturing Ontologyを参照して、Local DataをGlobal Dataに翻訳・統合する。Global Data、Activity Dictionary、Manufacturing Ontologyはともに、XMLドキュメントである。Activity DictionaryやManufacturing Ontology自身もDTEにより登録、更新などの操作が可能である。

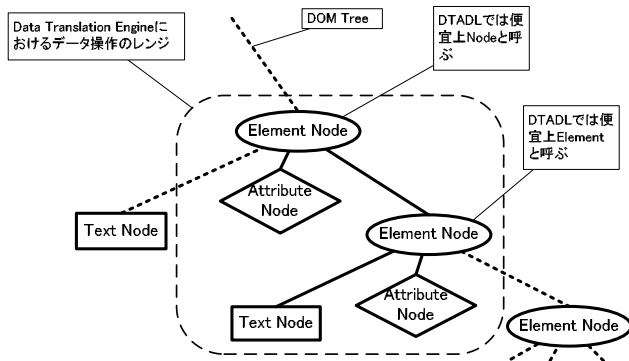


図3 DOM ツリーと Data Translation Engine におけるデータ操作のレンジの関係

4.1.1. Manufacturing Ontology

Manufacturing Ontologyには、グローバル製造品質管理に関わるオントロジー[7]が定義される。ローカルデータをこの概念定義とマッチングすることにより、データ・コードの変換やデータの補填に用いる。概念の構成には、製造に用いられる物、製品や部品の供給者、製造における不具合などの概念構成が定義される。

各概念には、製造拠点の各々のシステムで定義されているデータ・コードとの対応関係を必要に応じて定義する。例えば、「電源が入らない不具合(no power)および傷(scratch)といった不具合が、製造拠点 Mfg\_base1 のシステム上では、FPD\_NP\_03 および PAD\_SC\_01 と定義されている。」といった対応関係が定義される。

4.1.2. Data Translation Engine のデータ操作と Activity Dictionary

データ操作の記述の簡単化と処理系の簡単化のためにDTEでは、DOM ツリー[4]に対して図3のようなレンジを適宜あてはめて、基本的には、このレンジの範囲でデータ操作を行う。DOMツリーでは、Element Nodeの下には、Text、Element Nodeの属性を表すAttribute Node、子要素のElement Nodeなどが接続される。また、子要素のElement Node、さらに、その子の孫要素も同様の構造になりうる。データ操作のレンジは、特定のElement Node(1)を選択すると、そのElement Nodeと属性、子要素のElement Node(2)とそのTextと属性を含む。Activity Dictionaryでは、便宜上、Element Node(1)、(2)をそれぞれ、Node、Elementと呼ぶ。

Activity Dictionaryは、DTADL(Data Translation Activity Description Language)を用いて記述する。DTADL

の言語仕様概要は図4のようになる。ここで、「:=」を挟んで左辺にタグ(<...>)が記されている場合は、右辺の属性(<...>でないもの)、タグ、Textを含む。左辺に属性が記されている場合は、右辺の文字列(String)を値とする。(<a>|<b>)は、<a>と<b>のうちどちらかが選択されることを表す。また、\*はその直前の属性、タグあるいは括弧(...)が0回以上発生することを示す。

このDTADLに従い記述されるActivity Dictionaryの簡単な例は、図5のようになる。このActivity Dictionaryを用いて、図2の1つ目のLocal Dataを翻訳・統合すると、図6のdefectタグのようなグローバルデータが得られる。また、より複雑な手続きを記述することにより、多次元データキューブの作成や同期を行うこともできる。

```
<ActivityDictionary>:= <TranslationActivityDef>*
<TranslationActivityDef>:=dataID*, <ActivitySet> //手続きの使用条件,手続き
dataID:=Attribute to show String to identify local data
//データ操作の手続き, タグの記述順に評価
<ActivitySet>:=(<CreateOrUpdateNode>|<ForEach>|<Modify>|<Invoke>|<Activity_if>)*
//データ(Node)の挿入・更新
<CreateOrUpdateNode>:=<SetToCommonVariable>*|<TargetNode>
<ExistenceCheckCondition>(<Setatt>|<Setatt_if>)*...
<ActivitiesForElements>
<SetToCommonVariable>:=SetFrom SetTo
SetFrom:=Attribute to show XPath like description to get value to be bound
SetTo:=Attribute to show common variable name
//データ操作のレンジ設定
<TargetNode>:=Text of XPath like description of target element
//挿入が更新の判定のための条件
<ExistenceCheckCondition>:=Text of XPath filter like description
//属性データの挿入・更新
<Setatt>:=action_mode, <AttName>|<SetattFrom>
action_mode:=Attribute to control the Setatt function
<AttName>:=Text of attribute name
<SetattFrom>:=Text of XPath like description
//データ(Element)の挿入・更新手続き
<ActivitiesForElements>:=(<CreateORUpdateElement>|<CreateElement>|
<UpdateElement>|<ActivityForElement_if>)*
<CreateORUpdateElement>:=<Element_name>
(<SetToCommonVariable>|<Invoke>|
<Setatt>|<Setatt_if>|<CreateLink>| //属性データ(Element)の操作
<Copy>|<Translate>|...)* //Text(Element)の操作
<Element_name>:=Text of immediate sub-Element name
<Copy>:=<CopyFrom> //データの演算とコピー
<CopyFrom>:=Text of XPath like description of value which will copy
<Translate>:=<GCodeDefSearchRoot>|<ToBeTranslate>... //コード変換
<GCodeDefSearchRoot>:=Text of XPath like description to specify root node to search a
corresponded concept
<ToBeTranslate>:=Text of XPath like description of local data or common variable name
```

図4 DTADL の言語仕様(抜粋)

```
<ActivityDictionary>
<TranslationActivityDef dataID1="Mfg_base1" dataID2="MES-a" ... >
<ActivitySet><!--以下のタグを記述順に評価実行-->
<CreateOrUpdateNode><!--データの挿入・更新-->
<SetToCommonVariable SetFrom="@D_DATE" SetTo="str01"/>
<SetToCommonVariable SetFrom="substring("$str01$", 1, 10)" SetTo="str01"/>
<SetToCommonVariable SetFrom="@P_TYPE" SetTo="str02"/>
<SetToCommonVariable SetFrom="@P_SERIAL" SetTo="str03"/>
<!--データ操作のレンジ設定-->
<TargetNode>.../product/product_type="Sstr02$" and serial_number ="Sstr03$"/
defects/defect</TargetNode>
<ExistenceCheckCondition>defect_date="$str01$"</ExistenceCheckCondition>
<Setatt action_mode="only create">
<AttName>id</AttName><!--id属性の挿入-->
<SetattFrom>concat("defect_Mfg_base1_", "...")</SetattFrom>
</Setatt>
<ActivitiesForElements>
<CreateElement>
<Element_name>defect_date</Element_name>
<Copy><!--文字列操作をして挿入-->
<CopyFrom>substring("$strLVH01$", 1, 10)</CopyFrom>
</Copy>
</CreateElement>
<CreateORUpdateElement>
<Element_name>cause_code</Element_name>
<Translate><!--不具合の解析結果を変換して挿入・更新-->
<GCodeDefSearchRoot>.../DefectCauseThing</GCodeDefSearchRoot>
```

```

<ToBeTranslate>@ANALYSIS_RESULT</ToBeTranslate>
</Translate>
<Setatt>...</Setatt>
</CreateORUpdateElement>
...
</ActivitiesForElements>
</CreateOrUpdateNode>
...
</ActivitySet>
</TranslationActivityDef>
...
</ActivityDictionary>
    
```

図 5 Activity Dictionary の例

```

<global_production_data>
...
<product><!--製品データ-->
<product_type>PDT01</product_type>
<serial_number>D1048J</serial_number>
...
<defects><!--不具合データ-->
<defect id="defect_Mfg_base1...">
<defect_date>2003/07/18</defect_date>
<cause_code>no_power</cause_code>
...
</defect>
</defects>
...
</product>
...
</global_production_data>
    
```

図 6 Global Data の不具合状況データの例(抜粋)

### 5. 電子機器のグローバル製造への適用

Global REALMICS を、ある電子機器のグローバル品質管理に適用(図 7)し有効性を確認した。製造拠点は、プリント基板製造、製品組み立て、仕向け先別カスタマイズなどを担当しており、北米、欧州、アジアに分散している。10 システム程度の製造情報 DB システムから、製品情報、部品情報、製造計画、製造実績、部品組込実績、不具合状況などの 5 千 ~ 7 千レコード程度のデータをデイリー収集している。Global QC DB は、各種実績データとともに 2 次元から 8 次元までの十数種類の多次元データキューブにより構成した一種のデータウェアハウス[8]である。

Global QC DB 上のデータを用いるデータ分析機能は、グローバル品質センター及び全拠点から利用できる。本適用により、次のような効果が得られた。

- 広範囲の品質指標の自動集計が可能になり、全拠点の品質状況推移の把握、拠点間の比較、品質会議への報告などのサイクルを、従来の週次から日次に短縮できた。
- 不具合要因の責任を明確に集計して、拠点をまたぐ品質の向上のための源流フィードバックや下流拠点へのフィードフォワード、さらに、不具合の波及の防止のための定量的報告が迅速におこなえるようになった。

### 6. あとがき

グローバル製造管理を支援するための Global REALMICS において、製造拠点の様々な情報システムのデータを共通 DB に統合・同期するために、XML ベースのスク립ト言語 DTADL (Data Translation Activity Description Language) とこれを実行するデータ翻訳エンジンを開発した。本報告では、このデータ翻訳・統合手法について述べた。本手法の有効性は、電子機器のグローバル製

造品質管理への適用において、グローバル製造品質管理の管理サイクルを、週次から日次へ短縮するなどの効果により確認した。今後も、各製造拠点の品質レベルの均一化とグローバルな製造品質のさらなる向上に活用を進めたい。さらに、グローバルな生産管理などへも適用を展開しながら、複数の分散した拠点が一体となって生産活動を推進するための仕組みを確立したい。

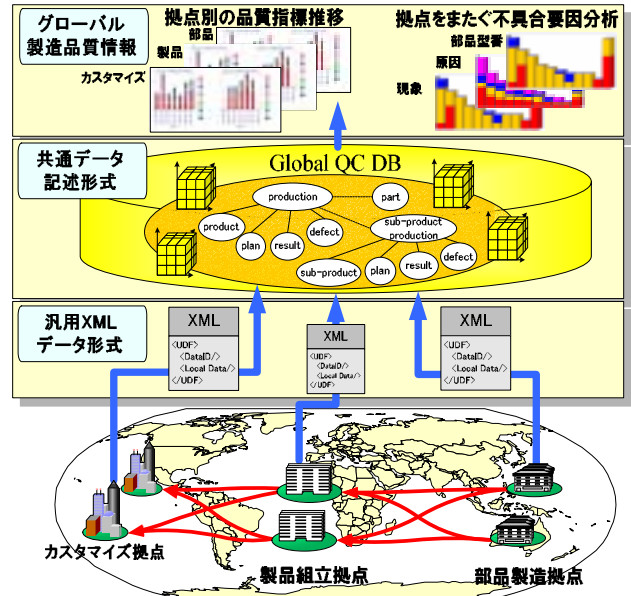


図 7 グローバル製造品質管理への適用

### 【文献】

- [1] 塹江清志, 沢田善次郎: 生産管理 理論と実践 1 生産管理総論, 日刊工業新聞社, 1995.
- [2] 沢田善次郎, 仁科健, 伊藤賢次: 生産管理 理論と実践 3 品質管理, 日刊工業新聞社, 1995.
- [3] 中山幹敏, 奥井康弘: 標準 XML 完全解説<上>, 技術評論社, 2001.
- [4] 中山幹敏, 奥井康弘, 日本ユニテック: 標準 XML 完全解説<下>, 技術評論社, 2001.
- [5] Michael McClellan: Applying Manufacturing Execution System, St. Lucie Press, 1997.
- [6] Schulte, R.: Real-World Lessons for Systematic Application Integration, Gartner Group Article Top View, AV-15-5519, 2002.
- [7] Gruber, T. R.: Ontolingua: A Mechanism to Support Portable Ontologies, Knowledge Systems Laboratory, Stanford Univ., Technical Report, KSL 91-66, 1991.
- [8] Kimball, R., et al.: The Data Warehouse Lifecycle Toolkit, John Wiley & Sons, Inc., 1998.

### 浪岡 保男 Yasuo NAMIOKA

株式会社 東芝 生産技術センター 研究主務。1990 中央大学大学院理工学研究科博士課程前期修了、2002 大阪大学大学院工学研究科博士課程後期修了、博士(工学)。製造情報システム、データ翻訳・統合、知識情報処理などに関する研究・開発に従事。電子情報通信学会、情報処理学会、人工知能学会、IEEE、日本データベース学会、各会員。