品質評価規格文書の構造化と それに基づくプロセス評価支援システムの試作

松下 誠[†]　世利 至彦[‡]　飯田 元[‡]　井上 克郎[†]

[†]大阪大学
〒560 大阪府豊中市待兼山町 1-3

[‡]奈良先端科学技術大学院大学
〒630-01 奈良県生駒市高山町 8916-5

プロセスの品質評価を行なう際には，品質評価規格文書を参照しながら行なう場合が多い．しかし，品質評価時に必要となる情報は，品質評価規格文書の各所に点在しており，文書中から捜し出すのに非常に手間がかかるといった問題がある．本研究では，品質評価規格文書として代表的なものの 1 つである SPICE (Software Process Improvement Capability dEtermination) を対象とし，品質評価時に必要な情報の効率の良い抽出，参照方法について検討した．まず，品質規格文書中から，評価時に必要となる情報を実際に抽出した．抽出した情報を分析した結果に基づいて文書の構造化を行なうために文書構造定義言語である SGML(Standard Generalized Markup Language) で用いられている方式を採用した．また，このようにして構造化された規格文書を入力として，必要な情報を検索，表示を行なうツールと，プロセス評価支援を行なうツールを試作した．これらのツールを使用することで，品質評価規格文書中に記述されている情報を容易に抽出することが可能となり，品質評価を効率よく作業が行なえることが期待できる．

Process Assessment Supporting System Based on Structuring of Assessment Documents

Makoto Matsushita[†]　Yoshihiko Seri[‡]　Hajimu Iida[‡]　Katsuro Inoue[†]

Osaka University
1-3 Machikaneyama, Toyonaka,
Osaka 560, Japan

Nara Institute of Science and Technology
8916-5 Takayama, Ikoma,
Nara 630-01, Japan

Process is usually assessed by comparing with the assessment documents, though the information for the assessment are scattered, it is a time-cosuming job to gather them. In this paper, we investigated the method to obtain information for process assessment by means of SPICE (Software Process Improvement Capability dEtermination). First, we extracted the information related to processes, products, levels. Then, to construct the SPICE documents based on that extracted information, SGML (Standard Generalized Markup Language) was adopted. Based on the constructed SPICE documents, we have made prototypes of two tools, a tool which shows the information about process assessment on the display, and a tool which investigates relations between processes and products. These tools enable us easily to get information for process assessment, and to improve process assessments greatly.

1 はじめに

ソフトウェア開発プロセスを改善することは，生産効率の向上，コストの削減などに直接影響を与えるため，非常に大きな意味を持つ．ソフトウェア開発プロセスの改善を行なうためには，まず現在行なわれているソフトウェア開発がどのように行なわれているか評価することが必要となろう．

このような背景から，近年ソフトウェアプロセスの品質保証に関する研究が盛んに行なわれて，利用されている [1, 2]．現在までに，ソフトウェアプロセスに対してさまざまな評価方法や参照モデルが提案されてきた．たとえば，SEI(ソフトウェア工学研究所) の CMM (Capability Maturity Model)[3, 4] や，ISO(国際標準化機構) の ISO 9000 シリーズ [5]，SPICE(Software Process Improvement Capability dEtermination)[6] などがあげられる．

しかし，これらの品質評価規格自体は自然言語で記述されているために，実際に評価を行なう際，必要な情報が文書の各所に散乱している，評価する対象が広範囲である，といった点が問題と考えられる．本研究では，品質評価を効率よく行なうために品質評価規格文書の構造化を行ない，それに基づいた評価支援システムを試作した．品質評価規格として，今回は SPICE を取りあげた．

2 プロセス品質評価規格 SPICE

SPICE (Software Process Improvement Capability dEtermination) とはプロセス品質評価規格の 1 つであり，ISO/IEC JTC 1/SC7 の WG10 にて国際標準化作業が行なわれているものである．SPICE 文書全体は 9 章で構成されており，記述は約 400 ページに渡っている．

SPICE ではソフトウエア開発における作業をプロセスとして表現し，各プロセスはいくつかの基本作業 (Base Practice) から構成されるとしている．また，プロセスはプロセスカテゴリー (Process Category) として 5 つにグループ化されている [7]．プロセスカテゴリーとしては以下の 5 つがある．

- Customer-Supplier
  顧客に直接影響のあるプロセス．

- Engineering
  システムやソフトウェアの仕様策定，実装，保守に直接影響のあるプロセス．

- Project
  開発プロジェクトの立案，およびプロジェクトにおける調整，資源の管理を行なうプロセス．

- Support
  プロジェクト中における他のプロセスの運用を支援するプロセス．

- Organization
  組織目標の立案や，組織目標を達成するために必要となるプロセス，プロダクト，その他資源の開発を行なうプロセス．

SPICE では習熟度 (Capability Level)，共通特性 (Common feature)，汎用作業 (General Practice) の 3 段階で構成される，能力判断基準となる作業が，上記のプロセスとは別に定義される．汎用作業においては，この作業を遂行するために必要となるプロセス/基本作業や，必要となる作業プロダクト (Work Product Type) が定められている [8]．これらの作業は全てのプロセスに対して適用される．この基準とプロセスを組みあわせることにより，各プロセスが目的に則してどの程度効率的に運用されているかを判断する．

SPICE では開発作業の評価を，1) 基本作業がどの程度実行されているか，2) 各プロセスを各汎用作業に照らしあわせ，どの程度効率的に実行されているか，という 2 種類について行なう．評価は 1) を 2 段階 (実行している (Y)，実行していない (N)) または 4 段階 (完全に適合する (F)，大部分適合する (L)，部分的に適合する (P)，全く適合しない (N)) で，2) を 4 段階で行ない，その結果は各共通特性毎，さらに各習熟度毎に集計される [9]．

SPICE ではさらに，集計された結果を用いたプロセス改善やプロセス能力判定のためのガイドや，評価作業の実行方法についても規定されており，開発プロセスを改善するための指針として十分な内容となっている．

3　品質評価規格の構造化

SPICE の文書中，SPICE におけるプロセスや習熟度等の定義や評価指針等は 2 つの章にそれぞれ書かれており，これら全部の記述は SPICE 全体のおよそ半分となる 200 ページ程度となる．このような大量の文書をフラットな構造のままで評価の際に参照するのは容易ではない．

本研究ではまず，SPICE において評価の際に利用される情報を抽出した．2節で述べた通り，SPICE においては，評価の際に必要な情報は「プロセスカテゴリーによって分類される各作業」「作業成果物」「習熟度」の 3 つに大きく分類することができた．さらに SPICE における評価作業はこれらの要素が評価対象中に存在するか，要素間に存在するいくつかの関連を参照することによって行なうことができると考えた．そこで，SPICE で述べられている評価を目的とした開発作業モデルを次のように定めた．

3.1　構造化のためのモデル

SPICE による品質評価対象となる開発作業全体は以下の 3 つの要素から構成する．

- タスク
  開発時における作業のうち，開発工程を構成するような作業．SPICE における基本作業，プロセスがこれに相当する．
- プロダクト
  開発時におけるあらゆる生成物，もしくは事前に準備されている成果物．SPICE における作業成果物がこれに相当する．
- レベル
  開発時における作業のうち，何らかの作業を遂行，改善するために必要となっている作業．SPICE における汎用作業，共通特性，習熟度がこれに相当する．

これら 3 つの要素間には，次の 4 種類の関係を導入する．

- タスク - プロダクト間関係
  あるタスクが実行時に必要とするプロダクトを表現
- タスク - タスク間関係
  あるタスクがその実行，評価時に参照されるタスクを表現
- レベル - タスク間関係
  あるレベルの遂行にあたって必要とされるタスクを表現
- レベル - プロダクト間関係
  あるレベルの遂行にあたって必要とされるプロダクトを表現

このような構造を導入すると，SPICE における品質評価の判断に必要となる情報は次のような形で明確に表現される．つまり，必要な情報はこの 3 要素 4 関係にて表現できている．

- 基本作業に対する評価
  該当するタスクがどのように行なわれているか，ということについて，タスクの定義内容や，このタスクとタスク - プロダクト関係を持つプロダクト等を参照することによって判断を行なうことができる．
- プロセスの習熟度に対する評価
  タスクとレベルがどのように行なわれているか，ということについて，該当タスクとタスク - タスク間関係を持つ他のタスクの実行状況や，該当レベルとレベル - タスク間関係を持つタスク及びレベル- プロダクト間関係を持つプロダクトを参照することによって判断を行なうことができる．

3.2　SGML 方式による構造化

本研究ではこの構造を表現するにあたり SGML (Standard Generalized Markup Language)[10] 方式を用いることにした．SGML とは，ISO によって標準化作業が行なわれている共通符号化を基本とした文書記述言語である．文書は SGML 方式を用いて構造化することができ，文書処理，文書データベース，文書交換等を行なうことができる [11].

SGML を用いることにより，次のような利点があると考えられる．

- 比較的簡単な構造をしているために，利用しやすいこと．

表 1: ELEMENT タグの属性

TYPE	要素の種類を表わす．TASK, PRODUCT, LEVEL のいずれかの値を取り，それぞれタスク，プロダクト，レベルを表わす．
ID	各種類中で一意に決定される文字列．SPICE における基本作業等に付けられた識別子を値として持つ．

- SGML 方式によって表現される構造は，元の文書に記述を加えることによって行われる．SGML 方式によって記述されたデータは SPICE の文書自身を包含することができるため，データを用いた文書の閲覧ツールなどを作成するのが容易である．

3.1節で定義した構造を，要素については ELEMENT というタグ (マーク) で，関係については RELATION というタグ (文章中に存在する構造を示すマーク) で定義した．この他，文書それ自身の構造を表現できるようないくつかのタグを定義した．

3.2.1 要素に関するタグ

要素は全て ELEMENT という名前のタグで表現する．ELEMENT タグには表 1のような属性が存在する．この他，属性 ID の持つ情報を補うために SUBID, SUBSUBID といった属性が必要に応じて定義される．

3.2.2 関係に関するタグ

関係は全て RELATION という名前のタグで表現する．RELATION タグには表 2のような属性が存在する．

3.2.3 SGML タグの記述例

このようにして定義した SGML タグを，SPICE 文書中に付加する作業を行なった．この作業は文章を読んで付加する必要があったため，手作業で行なった．文書全体で約 1500 のタグを付加した．SGML によって構造化された文書の例を図 1に示す．

この例では，作成するソフトウエアの要求仕様を定義する基本作業について定義されている文章に SGML のタグが振られている．先に挙げた ELEMENT や RELATION の他，文書自体の構造を表現するタグが付けられている．

表 2: RELATION タグの属性

TYPE	関係の種類を表わす．TKPD, TKTK, LTP のいずれかの値を取り，それぞれ「タスク-プロダクト間関係」「タスク-タスク間関係」「レベル-タスク間関係もしくはレベル-プロダクト間関係」を表わす．
SRC	関係上における参照元を表わし，参照元となる要素の識別子を値として持つ．
DST	関係上における参照先を表わし．参照先となる要素の識別子を値として持つ．
DIRECT	タスク-プロダクト間関係にのみ存在する属性で，プロダクトがタスクへの入力となるか出力となるかを表わす．IN もしくは OUT のいずれかの値を取る．

4 支援システムの試作

SGML 方式によって構造化された文書を利用する，品質評価支援システムを試作した．システムは図 2のような構成であり，2 つのツールと付随するデータベースが存在している．

SKRT は，品質評価文章中に記述されている各項目や，項目間の関連を表示するためのものである．KHST は，品質評価結果の入力，保存を行ない，評価作業を支援するものである．両ツールとも，構造化された文書を入力とし，SGML タグの

```
⟨ELEMENT TYPE=TASK ID="CUS.1" SUBID="2"⟩
⟨PREAMBLE⟩
CUS.1.2
⟨/PREAMBLE⟩
⟨TITLE⟩
Define the requirements.
⟨/TITLE⟩
⟨BODY⟩
  Prepare the system and software requirements to satisfy
  the need for a new product and/or service.
Note:  This definition of the requirements may be done
       completely or partially by the supplier.
⟨RELATION TYPE=TKTK SRC="CUS.1.2" DST="ENG.1"
  DST="ENG.2"⟩
       See "Develop System Requirements and Design"
       ENG.1, and "Develop Software Requirements" ENG.2.
⟨/RELATION⟩
⟨RELATION TYPE=TKTK SRC="CUS.1.2" DST="CUS.3.1"⟩
       Also see CUS.3.1, " Obtain customer requirements
       and requests. " CUS.1.2 is focusing on defining
       requirements when the software organization is
       acting as a customer.  CUS.3.1 is focusing on
       obtaining requirements when the software organi-
       zation is acting as a supplier.
       The primary difference is one of perspective,
       the role being performed.
⟨/RELATION⟩
⟨/BODY⟩
⟨/ELEMENT⟩
```

図 1: SGML による構造化の例

解析を行ない，ユーザからの要求を処理する．解析結果は必要に応じてデータベースとして保存し，再度検索する際に利用される．過去の入力された品質評価結果も同様にデータベースの形で保存される．以下，2 つのツールについてそれぞれ説明する．

4.1 SKRT

SKRT は，SPICE 文書中に記述されている各項目に関する内容を表示したり，タスク間の参照関係を調べて簡単なグラフ表現にするといった，文書中の情報を参照しやすくするためのツールである．実行例を図 3に示す．

文章中の参照関係を自由に検索するために，3.1 節で定義されるモデル上における操作について定めた (表 3)．各操作はそれぞれ 1 対多の写像として定義した．これらの操作にはモデル上で定義される参照関係の他，定義から導出される参照関係が含まれる．

また，これらの操作を組み合せて利用できるように，2 つの集合に対して，その和集合と積集合を計算する演算を定義した．また，各演算は引数として 1 つの値しか取ることができないため，集合の要素に対して同一の演算を行ない，その結果の和集合を取るための演算 (マッピング演算) を定義した．

表 3: 基本操作

操作名	意味
PD_TKIN	プロダクトを入力とし，そのプロダクトを入力として利用するタスクの集合を返す
PD_TKOUT	プロダクトを入力とし，そのプロダクトを出力するプロセスの集合を返す
TK_PDIN	タスクを入力とし，そのタスクが入力として利用するプロダクトの集合を返す
TK_PDOUT	タスクを入力とし，そのタスクが出力するプロダクトの集合を返す
TASK_C	文字列を与えると，その文字列を識別子の接頭語として持つようなタスクの集合を返す
LEVEL_C	文字列を与えると，その文字列を識別子の接頭語として持つようなレベルの集合を返す
LEVEL_TK	レベルを入力とし，そのレベルの評価時に関連するプロセスの集合を返す
LEVEL_PD	レベルを入力とし，そのレベルの評価時に関連するプロダクトの集合を返す
TKPD_LEVEL	タスクもしくはプロダクトを入力とし，それが評価時に必要とされるレベルの集合を返す
R_TK_ABOVE	タスクを入力とし，そのタスクに対して参照関係が再帰的に定義できるタスクの集合を木構造として返す
R_TK_UNDER	タスクを入力とし，そのタスクから見て参照関係が再帰的に定義できるタスクの集合を木構造として返す

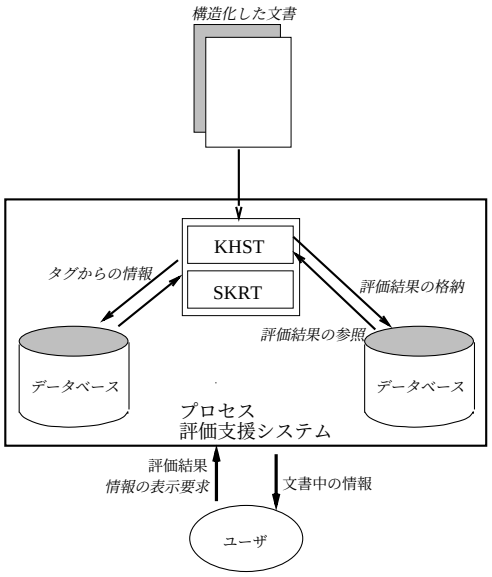

図 2: システム概要

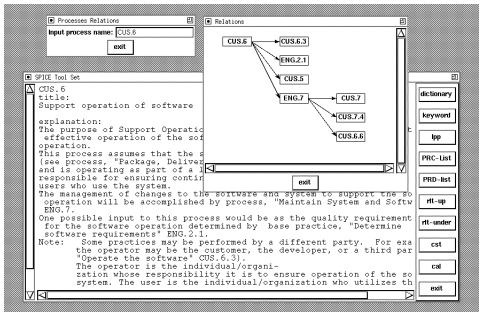

図 3: SKRT 実行例

例えば「ある 2 つのタスク (仮に A, B とする) に共通して入力となるプロダクト」を調べる場合には，TK_PRDIN(A) * TK_PRDIN(B) を実行させることによって求めたいプロダクトの集合の一覧が表示される．

SKRT では，このように定められた操作，演算を組み合せ，その結果を視覚的に表示することができる．結果が集合となる場合にはそれをリストの形で表示し，木構造はそのまま表示する．図 3では，木構造の表示例となっている．この他，SKRT では以下のような機能が提供される．

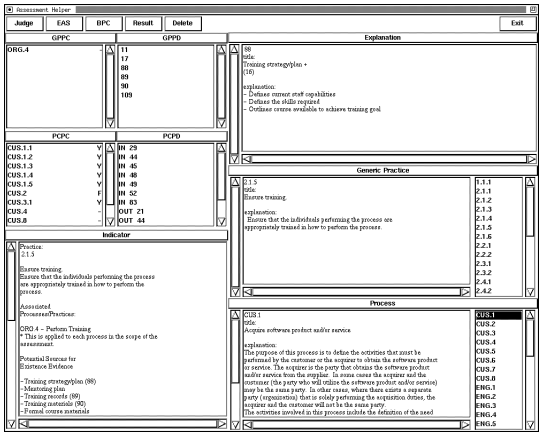

図 4: KHST 実行例

辞書機能

識別子からその識別子の定義が記述されている文章を表示する．識別子はユーザが明示的にキーボードから与えることができる他，ある操作を行なった結果表示される画面中にある識別子をマウスで指定することによっても与えることができる．

簡易検索機能

単語を与えると，文書を検索しその単語が含まれているタスク/プロダクト/ レベルの識別子の一覧を表示する．一覧表示された識別子をマウスで指定することにより，その識別子に対して辞書機能を呼びだして識別子の説明を表示させることができる．

4.2 KHST

KHST は品質評価時に利用し，評価を行なった結果の入力，保存，表示を行なえる他，SKRT 同様，規格文章の閲覧等も行なえる．動作例を図 4 に示す．

KHST では SPICE における品質評価作業のうち，プロセスの習熟度評価の実行を支援する．まず，評価対象となるプロセスを選択すると，該当プロセスの定義，このプロセス定義中に記述されている他のタスク，プロダクトとの関連が表示される．汎用作業を指定すると，該当する汎用作業の定義が表示され，判断基準を記述した文書や，そ

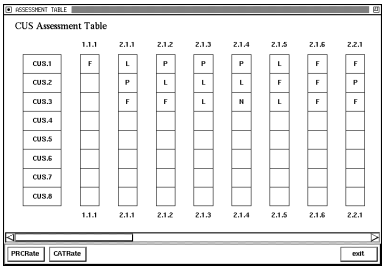

図 5: 評価結果表示

の文書中に記載されているタスクやプロダクト等の一覧が表示される．表示された一覧等から該当する項目の内容を参照することもできる．

プロセスと汎用作業の指定を行なった後に，評価結果の入力を行なうことができる．入力された評価結果はファイルに保存される．また，結果の表示を行なう際には，プロセスカテゴリー毎に表の形にまとめて表示する (図 5)．また，プロセスカテゴリーや習熟度単位で集計し，結果を 4 段階評価における各段階ごとの百分率で表示する機能を持つ．

4.3 考察

本研究では，文書の構造化に SGML 方式を採用し，それに基づいたツールを試作した．SGML は電子文書の再利用，全文データベース等に利用されている例があり [12, 13]，広く使われている手法と言える．しかし，これらのデータベース等では同種類の大量の文書に対する検索を提供していると考えられるが，SKRT では一つの長い文書に対する検索等の機能を提供していると考えられる．

実際のプロジェクトに対して品質評価を行なう場合には，非常に長い期間が必要となると言われている [2, 14]．KHST では，評価結果を保存することができるため，長期間に渡る評価時にも利用することが可能であると考えられる．

5 まとめ

プロセスの品質評価時に利用できる評価支援システムの試作を行なった．本システムは品質規格文書を SGML 方式により構造化した文書を入力としており，文章をそのままデータとして持つ．本システムを用いることにより，品質評価規格文書の閲覧を容易に行なうことができ，また，入力した評価結果を自動的に集計し表示することができる．このようなシステムを利用することで，品質評価作業時に発生する文書閲覧作業を軽減されると考えられる．

今回は品質規格文書として SPICE を取りあげたが，今後は他の規格等へ本手法を応用することを考えている．また，本システムの有効性を検証するためには，実際に運用することが必要になると考えられる．さらに，品質評価を行なう際に規格文書中で定義されている開発プロセスの構成要素と，評価対象となる現実の作業との対応づけを行なえるようにシステムを拡充したい．

参考文献

[1] Saiedian, H. and Kuzara, R.: “SEI Capability Maturity Model’s Impact on Contractors”, IEEE Computer, Vol.28, No.1, pp.16–26 (1995).

[2] MacLennan, F. and Ostrolenk, G.: “The SPICE Trials: Validating the Framework”, in Proceedings of the 2nd International SPICE Symposium, pp.109–118 (1995).

[3] Paulk, M., Curtis, B., Chrissis, M. and Wever, C.: “Capability Maturity Model for Software, Version 1.1”, Software Engineering Institute, CMU/SEI-93-TR-24 (1993).

[4] Paulk, M., Curtis, B., Chrissis, M. and Wever, C.: “Key Practices of the Capability Maturity Model, Version 1.1” , Software Engineering Institute, CMU/SEI-93-TR-25 (1993).

[5] ISO 9000-3 Guidelines for the Application of ISO 9001 to the Development, Supply, and Maintenance of Software (1991).

[6] The SPICE Project: “Software Process Assessment – Part 1: Concepts and Introductory Guide”, Version 0.02 (1994)

[7] The SPICE Project: “Software Process Assessment – Part 2: A Model for Process Management”, Version 0.01 (1994)

[8] The SPICE Project: “Software Process Assessment – Part 5: Construction, Selection and Use of Assessment Instruments and Tools”, Version 0.02 (1994)

[9] The SPICE Project: “Software Process Assessment – Part 3: Rating Process”, Version 0.01 (1994)

[10] ISO 8879: “Information Processing - Text and Office System - Standard Generalized Markup Language (SGML)” (1986).

[11] 田中洋一: “文書記述言語 SGML とその動向”, 情報処理, Vol.32, No.10, pp.1118–1125 (1991).

[12] 高柳由美子, 坂田英俊, 田中洋一: “SGML による全文データベースシステム”, 情処研報, 93-CH-18-5, pp.35–42 (1993).

[13] 森田歌子, 鈴木政彦, 宮川謹至, 浜中寿: “SGML 方式による情報管理誌全文データベース化の可能性と HTML による電子情報管理誌の試作”, 情処研報, 95-FI-37-2, pp.7–14 (1995)

[14] Woodman, I. and Hunter, R.: “Analysis of Assessment Data from Phase 1 of the SPICE trials”, Software Process Newsletter, No.6, pp.1–6 (1996).