一般社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
KBSE2013-2 (2013-05)

シーケンス図に基づくディペンダビリティケース作成法の研究

丁 峰† 山本 修一郎‡

†名古屋大学 大学院情報科学研究科 山本研究室〒464-8601 名古屋市千種区不老町
‡名古屋大学 情報連携統括本部 情報戦略室〒464-8601 名古屋市 464-8601 名古屋市千種区不老町
E-mail: † ding.feng@i.mbox.nagoya-u.ac.jp, ‡ yamamotosui@icts.nagoya-u.ac.jp

あらまし ディペンダビリティケースはシステムの安全性を向上する技術として注目されています.この技術をシステムの開発プロセスに導入することが重要な課題になっている.本報告では、開発プロセスで利用されるシーケンス図の各方面を分析して、その結果に基づくディペンダビリティケース作成法を提案する.

キーワード ディペンダビリティケース, UML, シーケンス図, システム開発

A method for developing a D-case based on Sequence diagram

Feng Ding† Shuichiro Yamamoto ‡

† Nagoya University, Gruduate School of Information Science, Yamamoto Lab.

Furo-cho. Chikusa-ku, Nagoya 464-8601 Japan

‡ Nagoya University, Strategy Office, Information and Communications Headquarters

Furo-cho. Chikusa-ku, Nagoya 464-8601 Japan
E-mail: † ding.feng@i.mbox.nagoya-u.ac.jp, ‡ yamamotosui@icts.nagoya-u.ac.jp

Abstract Dependability case is one of the methods used to assure system safety and availability. In this paper ,a method to develop dependability case based on sequence diagram will be proposed by analysing the sequence diagram used in the system development phase.

Keyword Dependability Case, UML, Sequence Diagram, System Development

1. はじめに

ディペンダビリティケース（Dependability Case）はシステムのディペンダビリティ（可用性，信頼性，安全性，一貫性，保守性）を保証しているか確認する手法として最近注目されている[1][2][3][4][6][7].

ディペンダビリティケースは GSN(Goal Structuring Notation)をベースに、DEOS で考えられてきた拡張を行った記述法を用いている[8][9].DEOS では、システムのディペンダビリティを検討するために、ディペンダビリティケースを用いて確認することは必要がある[9].したがって、システム開発プロセスでディペンダビリティケースを導入することの可能性は検討する必要がある. システム開発で利用される UML の各モデルについてディペンダビリティケースの作成法は重要な課題になる[2][10][11][12][13].しかし、現在 UML についてディペンダビリティケースの作成法は明確ではないという問題があった.本稿では、UML の重要なモデルとしてのシーケンス図に基づくディペンダビリティケースの作成法を提案する.

本稿の第 2 章で本研究の背景を示す.第 3 章でシーケンス図に基づいてディペンダビリティケースを作成するためにシーケンス図の各課題を検討して、それに対しての対策を提案する.そして、第 4 章はシーケンス図に基づくディペンダビリティケースの作成手法を提案する. 第 5 章は物品購入シーケンス図を用いて提案手法の有効性と適用性を検討する.第 6 章では本稿についてのまとめと今後の課題について説明する.

2. 研究背景

システムの高信頼化を目指し、システムの品質保証活動にかかわる予防活動および検知活動での各種手法や技法を中心に、高信頼化システムのための開発手法にかかわる研究は重要な課題になっている[2].ディペンダビリティケースはシステムの開発プロセスに導入してシステムのディペンダビリティを向上することができる.したがって、将来のシステムの開発プロセスには、ディペンダビリティケースが

システムの品質を保証する不可欠な手法として利用されるかもしれません.このように、ディペンダビリティケースに関わる研究を行っている.

ディペンダビリティケースを用いたディペンダブルな開発プロセスと、現在の開発プロセスを比較すると、図2-1のようになる.現状でも開発文書を用いてシステム開発を効率化している.しかしこれらの開発運用文書では、ディペンダビリティについての主張や、主張が成立することを示す明示的な証拠はありません.これに対してディペンダビリティケースを用いた開発プロセスでは、システムのディペンダビリティに対する主張、前提、証拠が明示的に記録されているので、主張が成立することを客観的に論証することができる.

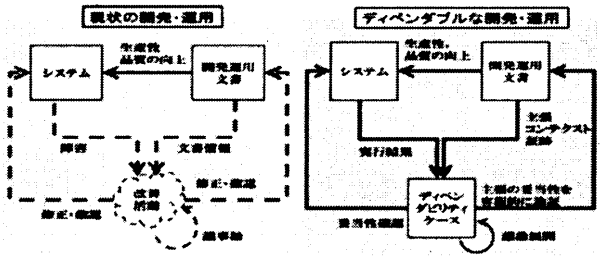


図 2-1

ディペンダビリティケースを作成するために、GSNを用いて記述されている.ここでは、トップダウンによる作成手順について述べる[5][6][7].

ステップ1：支持すべきゴールを同定する

ステップ2:ゴールが述べられている文脈(前提)について定義をする

ステップ3：ゴールを支持するために利用される戦略（説明）を同定する

ステップ4：戦略が述べられている基礎について定義する

ステップ5：戦略を洗練化するか（そして、ステップ1に戻り新しいグールを同定する）ステップ6に移る

ステップ6：根拠資料を同定する

UMLはシステムの設計や構造（アーキテクチャ）によく使われている仕様記述言語である.したがって、UMLのディペンダビリティを保証することは重要な課題になる.シーケンス図はオブジェクト指向を用いた開発において用いられ、オブジェクト間のメッセージの流れを時系列的に表現することが可能である.シーケンス図のディペンダビリティを確認するために、シーケンス図の記述項目のディペンダビリティを保証することは解決する必要がある．そして、システムの障害を出来るだけ除去するために、システムの開発中に潜在的なリスクについての検討は重要な課題になる.シーケンス図はディペンダビリティケースを利用して、シーケンス図の記述項目のリスクを確認する.

現状のディペンダビリティケースのトップダウン作成手順はシーケンス図に基づくディペンダビリティケースを作成することに適用するかが問題点になっているので、本稿はトップダウン作成手順に基づくシーケンス図のディペンダビリティケース作成法を提案する.

3. シーケンス図に基づくディペンダビリティケースを作成法についての検討

先節はディペンダビリティケースのトップダウンの作成手順について述べた.本節には、この作成手順を根拠として、シーケンス図に基づく、システムのディペンダビリティを保証するために、シーケンス図の各要素についてディペンダビリティケースの作成法を検討する.この作成法を検討するために、次の課題を研究する.

1．オブジェクトについての検討
2．メッセージについての検討
3．実行順序についての検討
4．実行仕様についての検討
5．結合フラグメントについての検討

3.1. オブジェクトについての検討

シーケンス図の中で幾つのオブジェクトがある.これらのオブジェクトのディペンダビリティを保証するために、ディペンダビリティケースを利用して、主張、前提，説明と証拠この四つの方面で検討する.

オブジェクトに関するディペンダビリティケースはどのように作成することを説明する.ディペンダビリティケースの主張では、オブジェクトは対象システムに対して、議論すべき命題である.次の説明ノードは対象システムのオブジェクトがどのように根拠つけるための戦略を決める必要がある.前提はオブジェクトに対する定義を参照する.各オブジェクトのリスクに検討するとき、それに対しての対策と根拠も必要である.このように、前提はリスク分析を提出する.オブジェクトに対する根拠はオブジェクトの操作に関する資料を参照する.

オブジェクトのディペンダビリティケースのノードの内容を整理する.

主張：オブジェクトはディペンダブルである
前提：オブジェクトリスク一覧
説明：オブジェクトについて説明する
証拠：＜オブジェクト＞動作の報告

3.2. メッセージについての検討

オブジェクトはディペンダブルである.シーケンス図には、クラス、コンポーネント、サブシステム、

またはアクターのインスタンス間でやり取りされるメッセージのシーケンスを表す相互作用が示される.オブジェクトの間の通信はメッセージを用いて実現する.送信側から受信側までメッセージを発信して、受信側はこのメッセージを処理する後別のメッセージを生成できる.基本的なメッセージの通信形式は

「通信形式」送信側->メッセージ①->受信側(処理)->メッセージ②

メッセージは対象オブジェクトの操作を呼び出すたり、情報を投げることを表す.これに基づく、メッセージの通信形式は次の形式を記述する.

「形式」<オブジェクト>が<他のオブジェクト>を<操作を呼び出す>、<情報を投げる>

メッセージがディペンダビリティを保証するために、ディペンダビリティケースを利用して説明する.以上の記述形式のディペンダビリティを確認する.

ディペンダビリティケースの主張ノードでは、メッセージがディペンダブルである.る下位のディペンダビリティケースの主張では、メッセージの形式の各要素に対して検討する.そして、メッセージの種類にも検討する必要がある.から、オブジェクトと対象オブジェクトと操作を呼び出すことと情報を投げることメッセージの種類はディペンダブルであることを主張ノードになる.説明ノードはメッセージの要素について検討する.メッセージの各要素のリスク分析も必要である.このために前提ノードはメッセージの各要素のリスク一覧になる.メッセージのディペンダビリティケースのノードの内容を整理する.

主張：メッセージがディペンダブルである
説明：メッセージの要素について説明
下位主張：
(1) オブジェクトがディペンダブルである
(2) 対象オブジェクトがディペンダブルである
(3) 操作がディペンダブルである
前提：メッセージの各要素のリスク一覧

3.3. 実行順序についての検討

シーケンス図はシステムの実行順序を表される.実行順序もユースケースのシナリオの経路を表すことができる.だから、シーケンス図の順序のディペンダビリティを保証するために、実行順序とシステムのシナリオの経路は一致することを保証する.根拠ノードは実行順序とシステムのシナリオの経路は一致することになる.この主張ノードでは、実行順序はディペンダブルがある.ディペンダビリティケースの説明ノードはシーケンス図のかく実行順序について説明する.要求分析の前段階で完成したユースケースのシナリオと実行順序のリスクの分析内容は前提になる.実行順序のディペンダビリティケースのノードの内容を整理する.

主張：実行順序はディペンダブルである
前提： ユースケースシナリオの各経路、実行順序リスク一覧
説明：シーケンス図のかく実行順序について説明する
証拠：実行順序はシナリオと一致する

3.4. 実行仕様についての検討

メッセージは対象オブジェクトの操作を呼び出すたり、情報を投げることを示す.実行仕様のディペンダビリティを保証するために、説明ノードはオブジェクトの操作を検討することを置いている.下位主張ノードはオブジェクトの操作はディペンダブルである.

3.5. 結合フラグメントについての検討

一般的なシステムの開発プロセスで利用されたシーケンス図は結合フラグメントを使用する、したがって、さまざまな状況で発生する可能性があるバリエーションを記述できる.こんな複合シーケンス図に対してのディペンダビリティケースでは、かく結合フラグメントを検討する必要がある.結合フラグメントはユースケースのシナリオを分析して作った.だから、ディペンダビリティケースを作成する前に、主シーケンス図の各結合フラグメントの正しさを確認する.

4. シーケンス図に基づくディペンダビリティケースの作成手順

結合フラグメントがある場合は各状況を発生する可能性がある.このシーケンス図に基づくディペンダビリティケースを作成する時は結合フラグメント種類に基づくシーケンス図を分解して、各シーケンス図に基づくディペンダビリティケースを作成する.この作成手順は次のようになる.

「作成手順」<A>結合フラグメント検討

(1) 結合フラグメントがあれば、結合フラグメントの種類によって分析して、複合シーケンス図は単純なシーケンス図に分解する.

(2) 各シーケンス図についてディペンダビリティケースを作成する.

そして、GSNのトップダウンによる作成手順に参考して、作成手順<B>を提案する.

「作成手順」<B>シーケンス図検討

(1) シーケンス図に対する主張を作成する(シーケンス図がディペンダビリティである)

(2) シーケンス図の各要素について説明する

(3) 主張に対しての前提を提出する

（4）シーケンス図の各要素をサブゴールにさせる
（5）オブジェクトをサブゴールになる
（6）メッセージをサブゴールになる
（7）実行順序をサブゴールになる
（8）実行仕様をサブゴールになる
「作成手順」＜C＞トップダウンのステップ５
（1）第三章でシーケンス図の各要素についての検討について各サブゴールのディペンダビリティケースを作成
（2）シーケンス図の各要素をサブゴールにさせる
（3）オブジェクトについてディペンダビリティケースを作成
（4）メッセージについてディペンダビリティケースを作成
（5）実行順序についてディペンダビリティケースを作成
（6）実行仕様についての検討

5. 具体的な例に作成法を導入する

名古屋大学の事務案内の物品購入の手続きを参考して物品購入のシーケンス図を作った.この物品購入シーケンス図を利用して、シーケンス図に基づくディペンダビリティケースの作成法を説明する.この物品購入シーケンス図は付録図 A に表する.

物品購入シーケンス図は結合フラグメントがあるので、「作成手順」＜A＞を利用して,複合シーケンス図を単純なシーケンス図に分解する.分解した四つのシーケンス図を付録に描いている.

付録の図 A を選んでディペンダビリティケースの作成法を用いてディペンダビリティケースを作成する.作成手順＜B＞を用いて、主なディペンダビリティケースを図 5-1 に示す.作成手順の B が終わった.

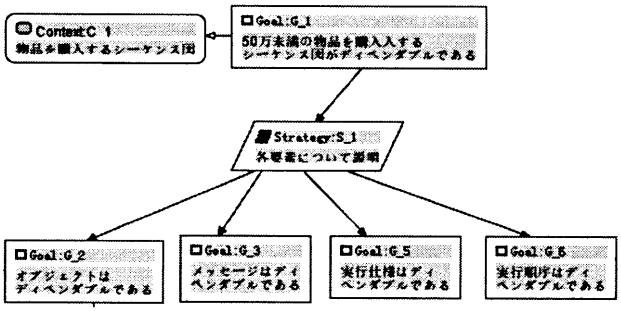


図 5-1 シーケンス図に基づくディペンダビリティケースを作成する

そして、作成手順＜C＞を用いて,各サブオールについてのディペンダビリティケースを作成する.

最初では、オブジェクトに対すてディペンダビリティケースを作る.物品購入シーケンス図のオブジェクトは依頼者、契約課、承認者、購入担当者、商品販売者、受取人、会計掛がある.各オブジェクトは主張になってディペンダビリティケースを作成する.ここは依頼者を分析して分解するが、他のオブジェクトのディペンダビリティケースの作成プロセスは省略する.オブジェクトに対してのディペンダビリティケースは次の図 5-2 を示す.

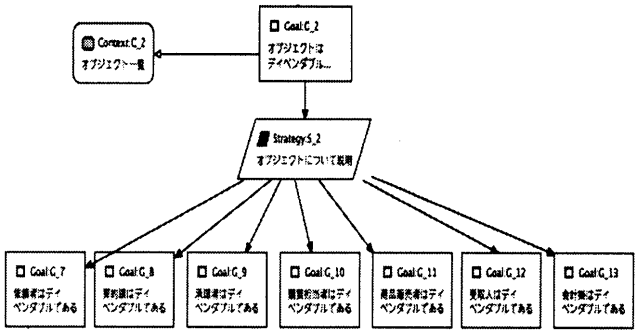


図 5-2　オブジェクトを分解する

オブジェクトを検討した結果について、オブジェクトのディペンダビリティを保証するために、オブジェクトのリクスについて説明する.依頼者のリスク一覧は前提ノードに置いている.依頼者のディペンダブルがあるために、依頼者は依頼を提出する権限の確認が必要である.根拠としては、依頼を提出前に依頼者の発注権限の証明書を出する.依頼者のリスク分析に対してのディペンダビリティケースは次の図 5-3 を示す.

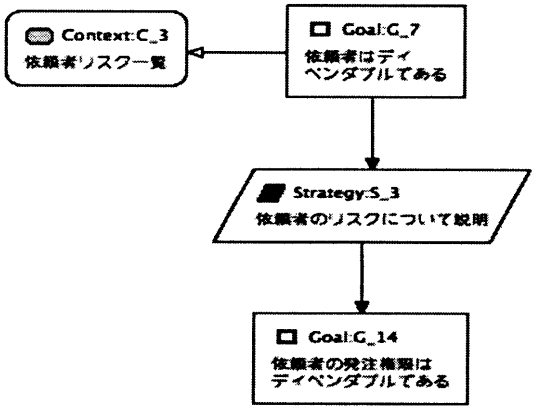


図 5-3　依頼者のリスク分析

＜承認者は購買担当者に見積書を出す＞というメッセージはメッセージを分解される各例の一つである.この例についてディペンダビリティケースを作成するが、他の例のディペンダビリティケースの作成プロセスは省略する.三つの要素＜承認者＞、＜購買担当者＞、＜見積書を出す＞はディペンダブルがあることを検討して、このメッセージのディペンダ

ビリティを確認する.そして、この三つの要素についてのリスク分析結果を前提になる.メッセージに対してディペンダビリティケースは次の図 5-4 を示す.

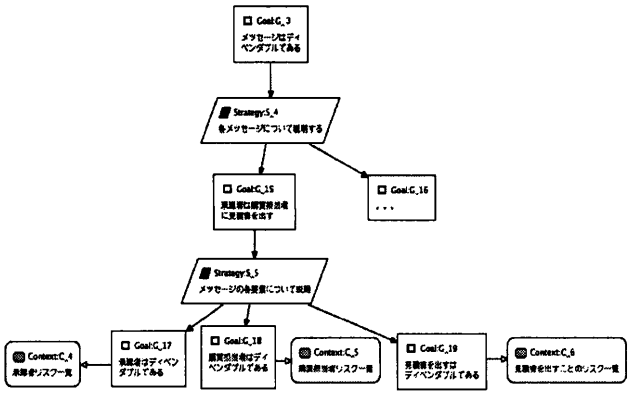


図 5-4　メッセージの分解例

商品の価格は 50 万円未満のシーケンス図の実行順序についてディペンダビリティケースを作成するとき、この実行順序についてのリスク分析結果とユースケースのシナリオは前提ノードになる.そして、実行順序とシナリオを対照する結果はエビデンスになる.実行順序に対してのディペンダビリティケースは次の図 5-5 を示す.作成手順<C>が終わった.

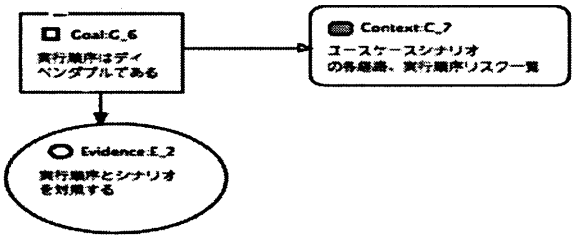


図 5-5 実行順序のディペンダビリティケース

6. まとめと今後の課題

本稿はシーケンス図に基づくディペンダビリティケースの作成手順を提案して、具体的な例に導入した.今後はこの作成法の有効性と適用性を確認することを継続するために、本稿の一例だけでは不十分であるから、他の具体的な例に導入する必要がある.そして、本稿のディペンダビリティケースのエビデンスはほんどんとないだから、今後はこの例に対して根拠を作って、完全なディペンダビリティケースを作成する.

システムのディペンダビリティを保証するために、システム開発で利用される UML の各モデルについてディペンダビリティケースの作成法の研究を進める必要がある.UML に基づくディペンダビリティケースの各作成法を作って、実際のシステムの開発に導入するように研究する予定である.

文献

[1] Yutaka Matsuno, Jin Nakazawa, Makoto Takeyama, Midori Sugaya, and Yutaka Ishikawa. Toward a language for communication among stakeholders. In Proc. of the 16th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC'10), pages 93–100, 2010.

[2] 松野裕, 高井利憲,山本修一郎, D-Case 入門～ディペンダビリティ・ケースをかいてもよう！～.株式会社ダイテックホールディング,2012.ISBN978-4-86293-079-8.

[3] Yutaka Matsuno and Shuichiro Yamamoto. A new method for writing assurance cases. Interna- tional Journal of Secure Software Engineering (IJSSE), Special Issue on Cybersecurity Scientific Validation, January 2013. Accepted for Publication.

[4] Yutaka Matsuno and Shuichiro Yamamoto. Consensus building and in-operation assurance for service dependability. In Proc. of CD-ARES, LNCS 7465, pages 639–653. Springer, 2012.

[5] Kelly, T. P, A Six-Step Method for the Development of Goal Structures, York Software Engineering, 1997.

[6] T. Kelly. "Arguing Safety, a Systematic Approach to Managing Safety Cases". PhD Thesis, Department of Computer Science, University of York, 1998

[7] Tim Kelly, Rob Weaver(2004). The Goal Structuring Notation-a safety argument notation. In Proc. Of DSN 2004, Workshop on assurance Cases, 2004

[8] DEOS プロジェクト, http://www.crest-os.jst.go.jp.

[9] DEOS プロジェクト, 2011 科学技術振興機構 White Paper DEOS-FY2011-WP-03J.

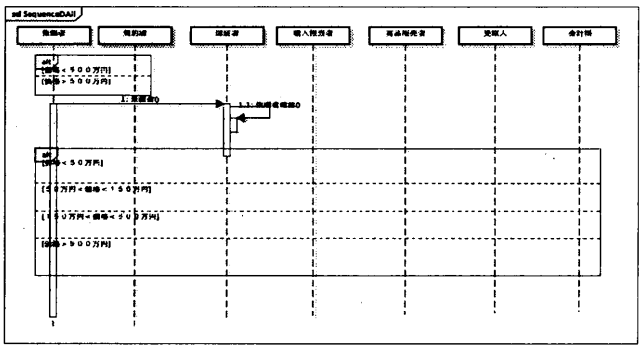
[10] 山本修一郎, 松野裕, ディペンダビリティケース作成法に関する一考察, KBSE 研究会, IEICE-112, vol. IEICE-SS-164, No. IEICE-KBSE-165, pp.61-66, 2012

[11] 松野 裕, 高井利憲, ヴァイセ パテゥー, 山本修一郎, アシュアランスケース構築法の提案, KBSE 研究会, 2012

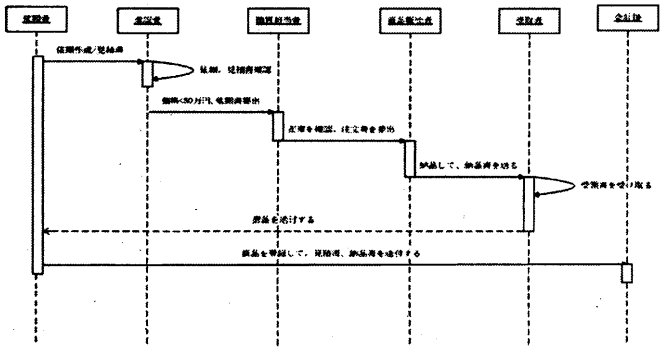
[12] Vaise Patu, Yutaka Matsuno, Shuichiro Yamamoto, Application of D-Case to the usage flow diagram scenario of the Distributed E-Learning System called KISSEL in Asian Pacific Universities, KBSE 研究会, 2012

[13] 山本修一郎, 松野裕, ユースケース分析に基づくディペンダビリティケース作成法の提案, KBSE 研究会,信学技報, vol. 112, no. 419, KBSE2012-61, pp. 19-24, 2013 年 1 月

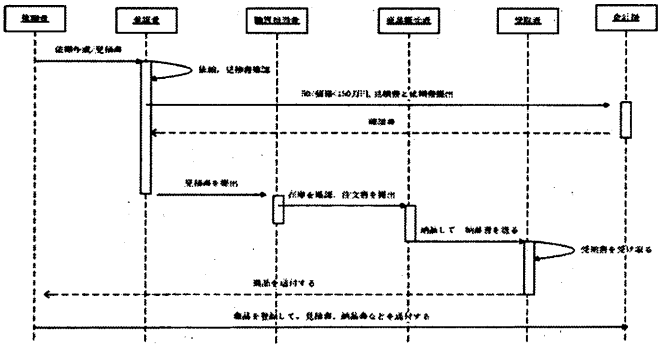
付録 シーケンス図一覧



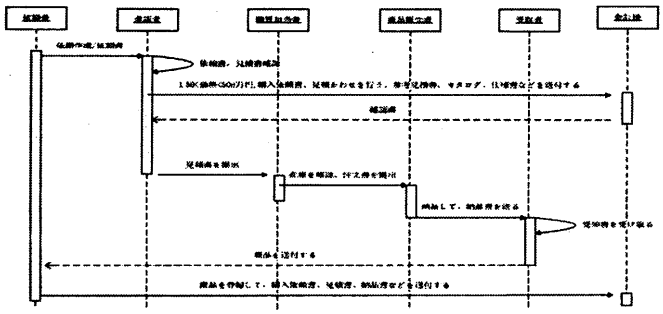
付図 A　主なシーケンス図



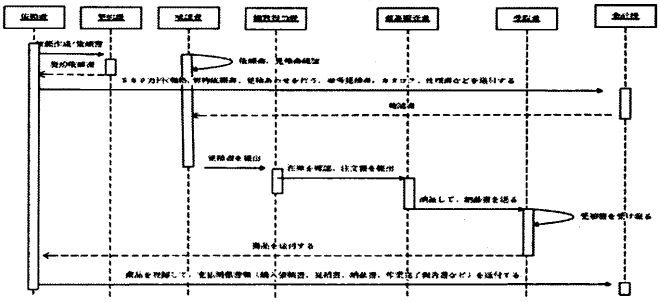
付図 B　商品価格＜５０万円



付図 C 50<商品価格<150 万円



付図 D 150<商品価格＜500 万円



付図 E 商品価格>500 万円