

非機能要求を共通評価基準とし た並列要求獲得法

2002年1月17日 伊豆にて

信州大学 工学部

海谷 治彦

目次

- 非機能要求について
- Stakeholderについて
- 方法論の概要
- 例題
- 議論・今後の課題

非機能要求とは

- 詳細は予稿集参照
- 最終製品の特性を指定している
- Stakeholderにより(当然)重視する特性は違う.
- 場合によっては, 重視する特性が対立するかも
 - …… この点に本研究は着目.

Stakeholderとは 1/2

- 1980年頃にMitroffという人が言い出したらしい。
- 検討中の変化に対して関心を示す人, 利益・損失を受ける人である利害関係者のこと。

Pericles Loucopoulos, Vassilios Karakostas著, 富野 壽監訳.
要求定義工学入門.
共立出版, 初版, Oct. 1997.
ISBN 4-320-09719-X, pp. 85

Linda A. Macaulay.
Requirements Engineering.
Applied Computing. Springer, 1996.
ISBN 3-540-76006-7, pp.32

- めんどいので以下 SH と略記する。

Stakeholder 2/2

- コンピュータシステムでは以下の種類
 - 開発すべきシステムに経済的関心を持つ人々
 - 開発すべき設計・実装に責任がある人々
 - 開発後の導入に責任を持つ人々
 - システムを使用することに関心を持つ人々
- 負のStakeholder
 - 損失を極大にしなければいけないSHの存在.
 - 例: セキュリティシステムの侵入者, 成りすまし者, 競合する商売敵等.

方法論の概要

- 複数のアナリスト, 複数のSH.
- 同時並行獲得.
- SHの不満駆動型要求獲得
- 仕様記述は(現状では)シーケンス図に固定
- 評価表による不満・満足記録
- 逐次獲得法と併合法

評価表の例

List of Stakeholders

		List of Stakeholders			
		Contributer	Chair	Committee	Reviewer
List of Requirements types	Conf.				
	Avail.				
	Acc.			B +	Get accurate list.
	Time	I -	Can not know whether my abstract was received or not.	B -	Not
	Ea	A +	Reference of a refined specification		
			A +	Free from the task of gathering abstract	
			A -	Tiresome to make abst. list	
		B +			
	+	4	1	2	1
	-	1	0	0	1

Score:
+ Satisfaction
- Dissatisfaction

Total score of Satisfaction

Subtotal score of Satisfaction

Content of the evaluation

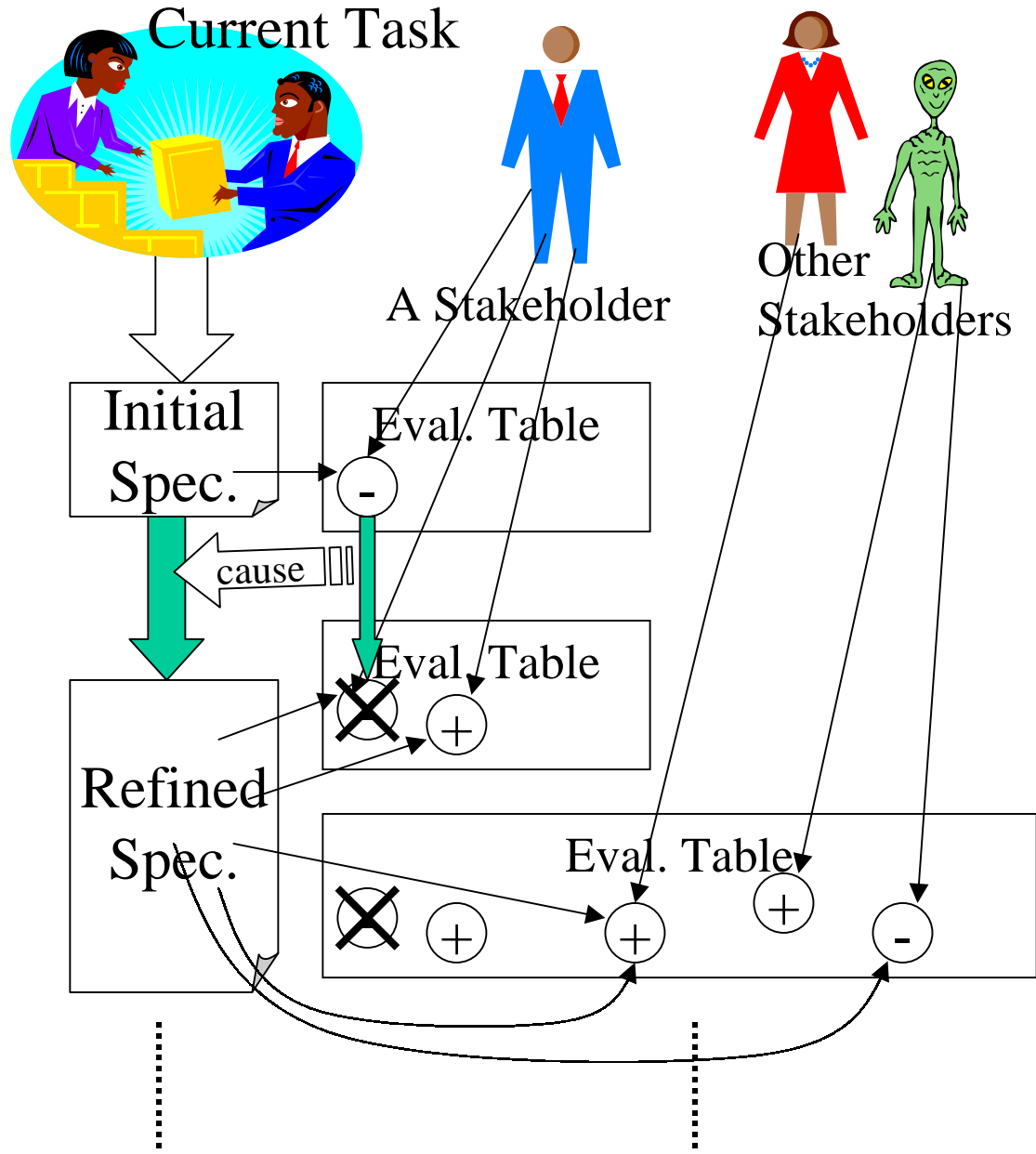
Subtotal score of Dissatisfaction

Total score of Dissatisfaction

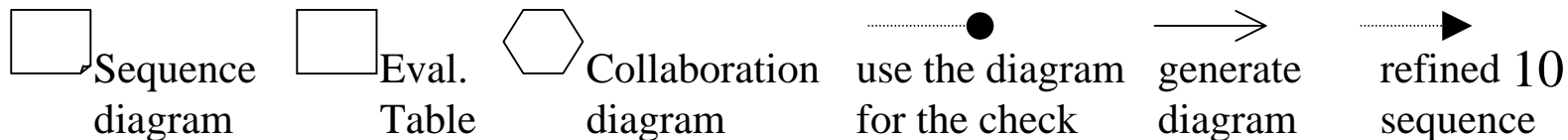
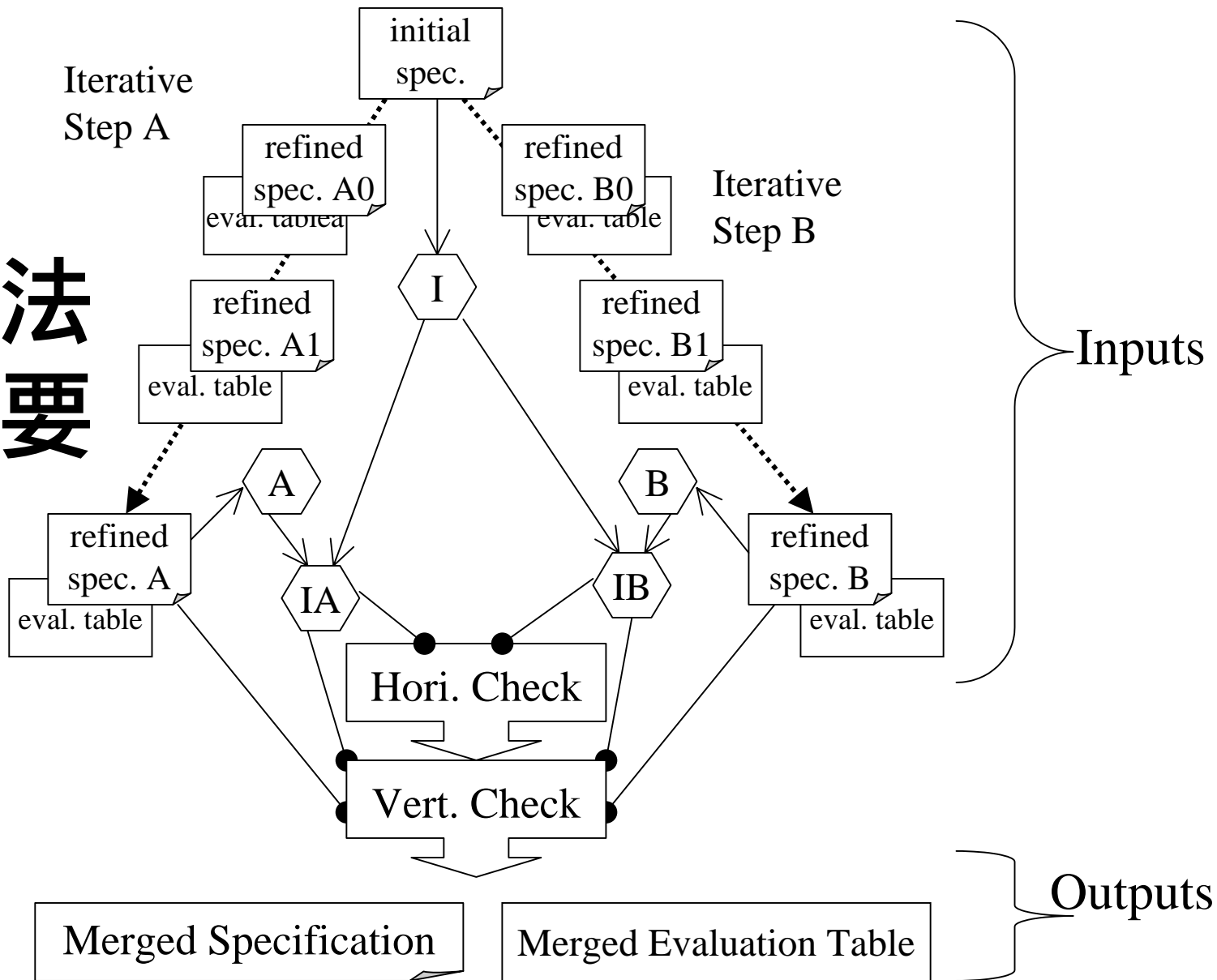
評価表の意義

- 要求変更過程での、ある断片でのSH毎の、その仕様に関する評価を記録。
- SHの総和としての仕様に対する評価を**仕様の価値**とする。
 - 選択肢の選別に利用。
 - 縮退(後述)候補の選別に利用。
- 負の評価は全て潰すことを前提とする。

逐次法 の方針



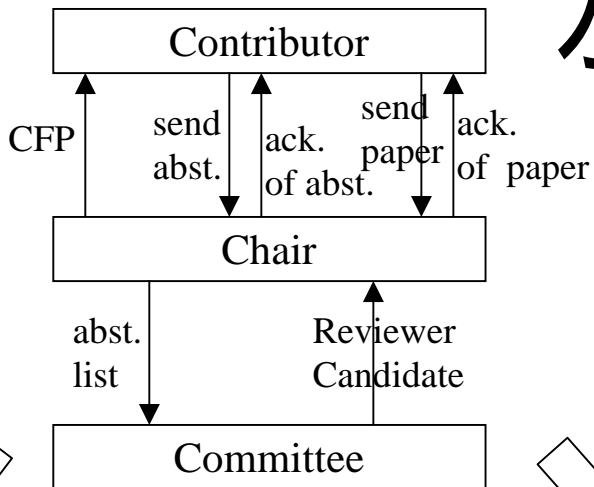
併合法 の概要



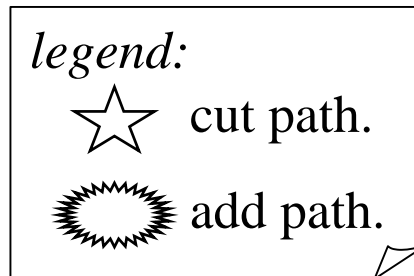
併合法の2つのチェック

- 水平チェック: シーケンス図をコレボレーション図に重ねて変更を重ねることがないことをチェック.
- 垂直チェック: シーケンスの順番の変更が矛盾がないかをチェック.
- シーケンス図に限定した部分.
- 水平の方が簡単にチェックできるため先に行う.

Initial Spec.

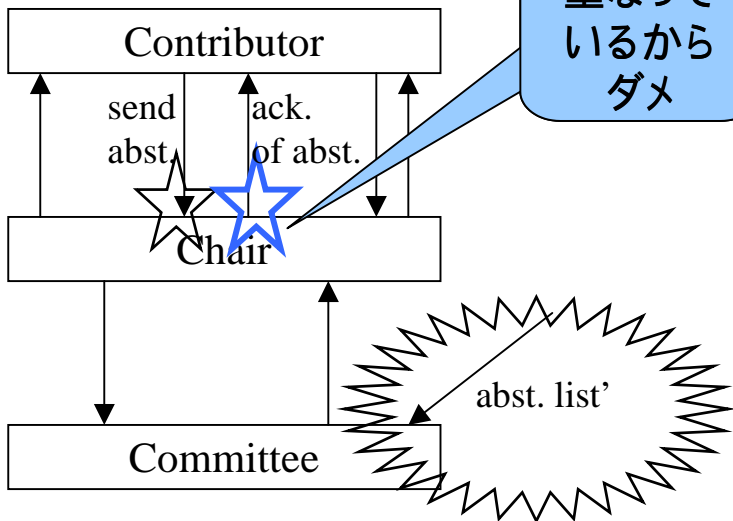


水平チェックの例



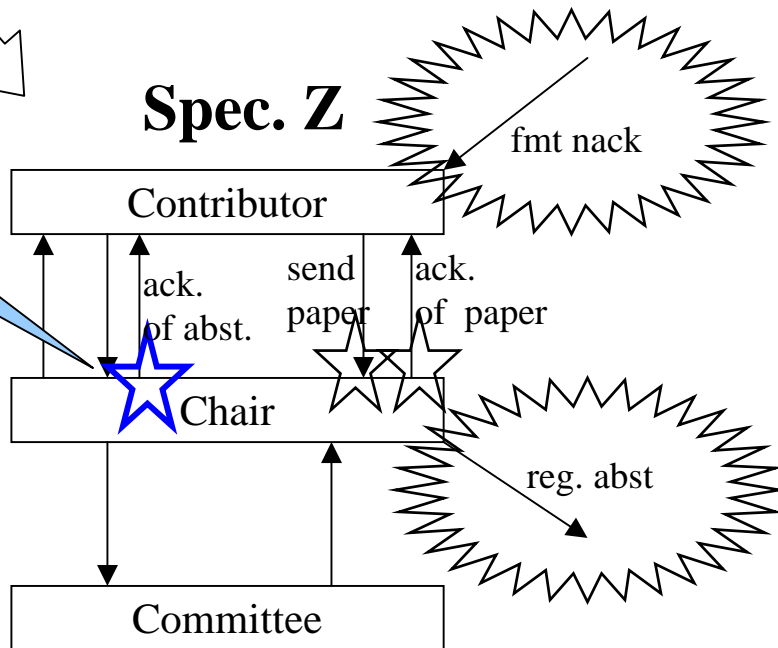
refinement
A, B, C, D

Spec. D



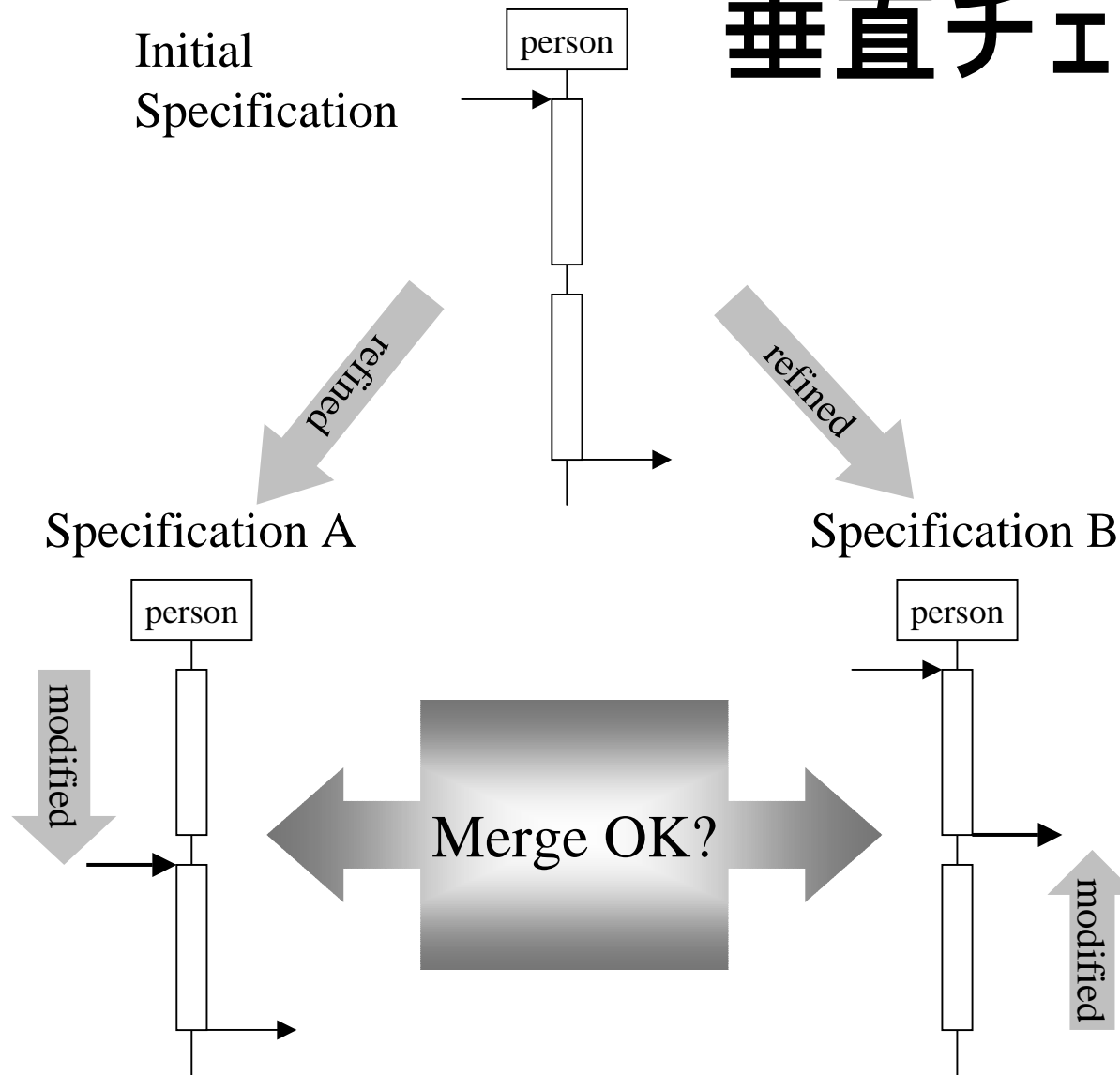
refinement
X, Y

Spec. Z



重なって
いるから
ダメ

垂直チェックの例



チェックに通らなかったら？

- 併合できない。
- 併合する一方の要求変更列を縮めて、再度併合をする。(縮退)
- 当然、要求仕様全体の価値は落ちる。
- 実は、多少、チェックがキツイので、弱くしても良いが、その分チェックの手間が増える。

併合が成功したら？

- 別個，獲得した仕様を混ぜてよい。
- 併合した仕様に付帯する評価表のスコアの合計価値を，その仕様の価値としてよい。
- 混ぜた結果発生する再評価は現在考慮していない。

議論・今後の課題

- 仕様の併合は仕様記述言語に依存するが、まあ、これは仕方がない。
- 併合成功後の仕様の再評価戦略。
 - 網羅的にやっていたのでは、並行作業した意味が無くなってしまう。
- 評価表における優先度付け
 - Stakeholder毎
 - 要求型毎
- 負のStakeholderの扱いの検討。
- 支援ツールを作ってもう一本論文を書こう！