



ETロボコン2010 ソースコード品質評価活動のご案内

2010年5月21日

ETロボコン実行委員会

東京地区実行委員会

関西地区実行委員会

(協力)

株式会社オージス総研 組み込みソリューション部

活動のねらい

これまでETロボコンは「走行性能」だけでなく「設計品質」を評価対象とすることで、組み込み開発現場で活躍できるエンジニアを教育する場として機能してきました。

しかし、実開発では「設計品質」を高くするだけでは不十分です。品質の高い「設計」を「実装」に反映して初めて製品としての品質を高めることにつながります。結局、システムに搭載されるモノはソースコードであり、その品質を意識することが求められるのです。

そこで、私達は貴重なETロボコンという教育の場を「ソースコード品質に対する意識を高めることができる場」にするために、希望者を募りソースコードの品質評価を実施したいと考えました。

ソースコード品質に対する評価が得られる機会をつくることで、ソースコード品質に対する意識を高め、エンジニア教育の場としてETロボコンへの参加をいっそう有意義にすることが本活動のねらいです。

活動の位置づけ

私達は、ETロボコンを「ソースコード品質に対する意識を高めることができる場」にするために、将来的にはETロボコンの全ての大会においてソースコードの評価あるいは審査が実施される必要があると考えています。

本活動は、この目標に向けたフィージビリティスタディと位置づけており、2010年度は東京地区と関西地区の独自取り組みとして実施します。

東京地区の取り組みと関西地区の取り組みは、それぞれ独立した内容であり、例えば、各地区実行委員会の方針に基づき、独自の方法でソースコード評価結果の分析やフィードバックを実施することがあります。

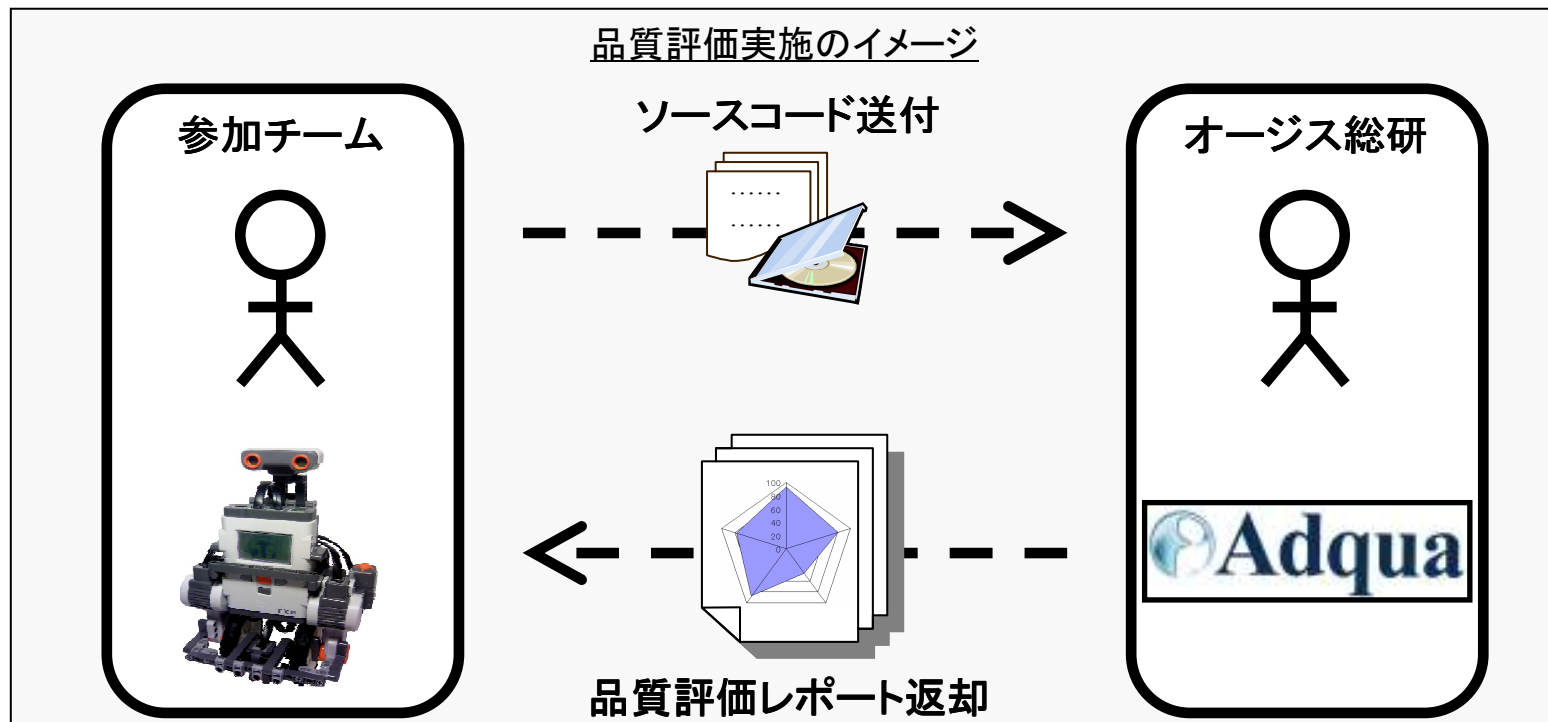
本活動の実施結果を踏まえ、次年度の活動内容や展開範囲を検討します。

参加のメリット

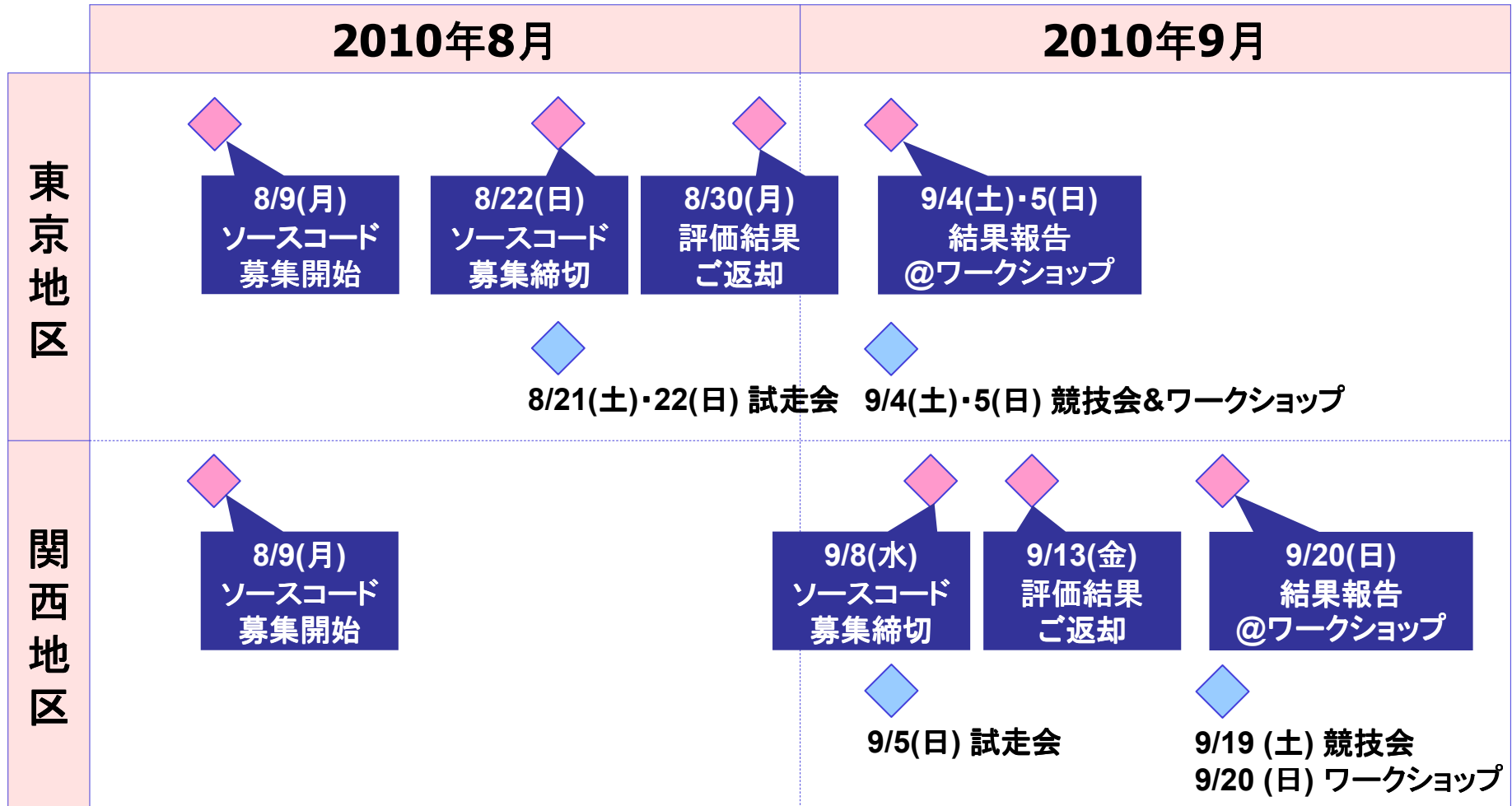
- ソースコード品質に対する意識が向上する
 - ソースコードを誰も評価しないという状況では、設計品質の向上に傾倒し、実装が疎かになりかねません。しかし、ソースコードを評価するとなれば、実装を疎かにすることはできませんし、ソースコードそのものの品質を意識するようになります。
 - 実際の開発現場では、設計品質とソースコード品質の両方を高めることが求められます。ETロボコンへの参加にあたり、ソースコード品質への意識付けを行うことができれば、いっそう有意義な教育の機会となるでしょう。
- ソースコード品質の良し悪しがわかる
 - ソースコードの評価結果では、信頼性や保守性といった“品質特性”から、詳細な“メトリクス”までの得点を知ることができます。これを見れば、ソースコードの“どこ”の“何”が良いのか、あるいは悪いのかがわかります。

実施の流れ

- ソースコードの送付
 - 評価したいソースコードを下記メールアドレス宛てにお送りください。
- Etrobo_Adqua@ogis-ri.co.jp**
株式会社オーグス総研 ETロボコン2010 ソースコード品質評価担当
- ソースコードのご提出は1チームにつき1回とさせていただきます。
- 品質評価レポート返却
 - ソースコードの状態によっては測定ができず、止むを得ず評価結果を返却できない場合がございます。予めご了承ください。



スケジュール



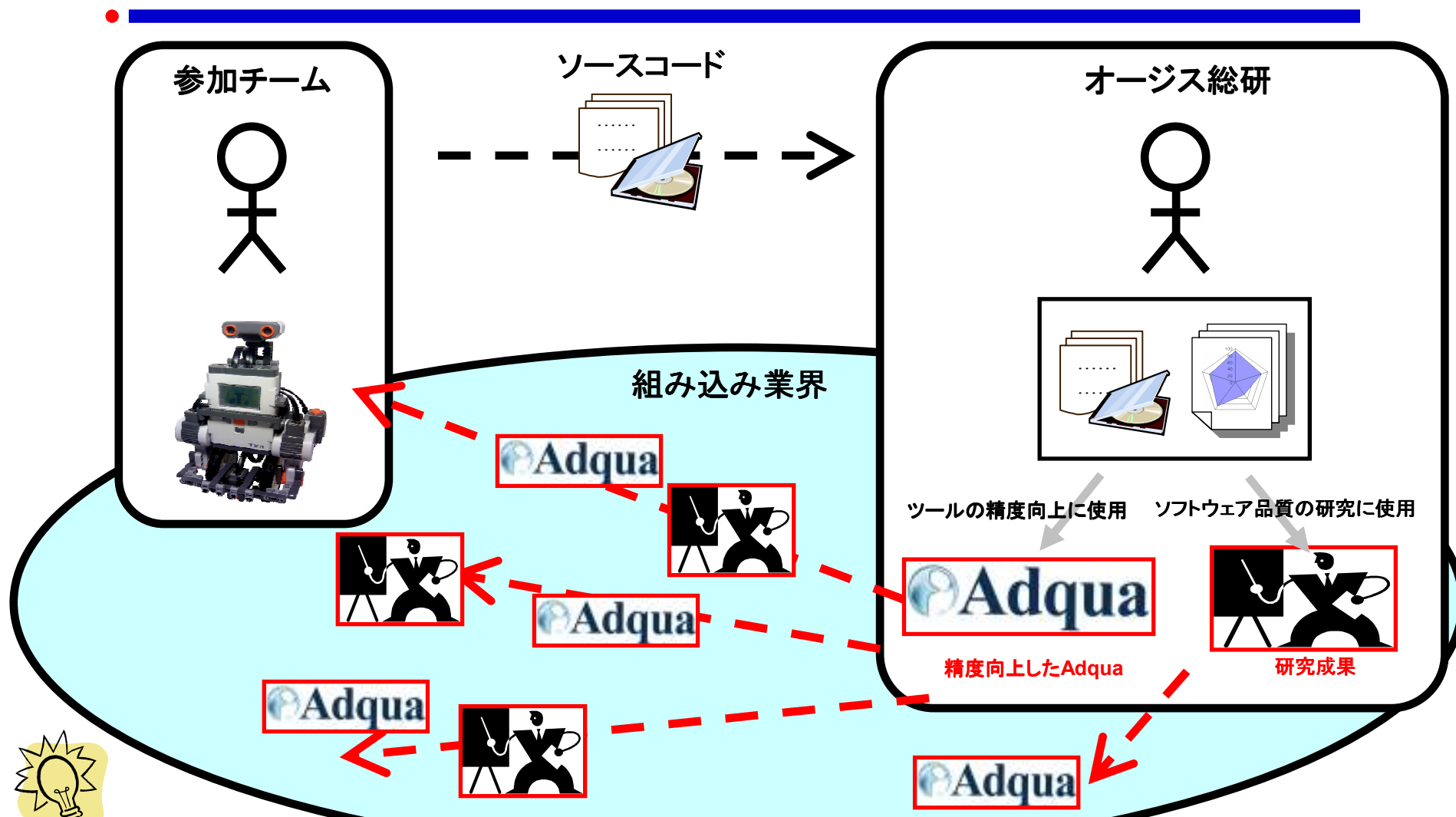
参加要件(1/2)

- ソースコードの条件
 - 言語
 - C言語 / C++
 - 規模
 - ソースファイルの物理行数10000行程度まで
 - 対象ソースコード
 - NXTに搭載するプログラムのソースコード
 - 開発環境
 - ETロボコン2010実行委員会指定の環境

参加要件(2/2)

- **ご同意いただきたいこと**
 - お送りいただくソースコード(以下、受領物)および品質評価結果(以下、成果物)は、下記の用途で使用するためETロボコン実行委員会とオージス総研が機密情報として保持させていただきます。
 - 本活動で使用するソースコード品質評価ツール「Adqua」の精度向上
 - ソフトウェア品質に関する研究
 - 次年度の活動内容の検討材料
 - 受領物および成果物は上記以外の用途には一切使用いたしません。
 - 受領物および成果物の保持期間を下記のように定めます。
 - 受領時から2012年度末まで保持
 - 2012年度末時点で、期間の見直しを実施
 - 期間の見直し時に受領物提供者からの破棄要請があれば個別に受領物を破棄
 - 受領物および成果物は、ソフトウェア品質に関する研究を共同で実施する下記関係者間で共有します。
 - ETロボコン実行委員会
 - Adqua提供者(オージス総研、Adqua共同研究者)
 - 研究成果を一般に公表する可能性があります、個別のチーム名や個人が特定される状態で公表することはいたしません。
- **その他**
 - ソースコード品質評価の結果は、モデル審査には一切影響しません。

オージス総研における受領物や成果物の用途(1/2)



皆様からいただいたソースコードや評価結果を、品質評価ツール(原則無償提供)の精度向上やソフトウェア品質に関する研究に使わせていただき、成果を広く組み込み業界に還元することで、業界の発展に寄与します

オージス総研における受領物や成果物の用途(2/2)

- ---

 - 品質評価ツール「Adqua」の精度向上に使用
 - Adquaでは、多数のプロジェクトの測定データから得点化のためのパラメータを統計的に求めています。測定データの件数が多いほど精度の高いパラメータが得られると考えられますので、皆様からいただいたソースコードの測定データも使用させていただきます。
 - Adqua評価結果とソースコードの定性評価結果を比較し、そのギャップに対処することで更なる精度向上を努めたいと考えています。この際の評価対象として皆様からいただいたソースコードを使用させていただきます。
 - 「Adqua」は製造メーカーの組み込みソフトウェア開発現場に無料で提供しております。広く業界の皆様にお使いいただけるツールの精度向上を通じて、業界の発展に寄与したいと考えています。
 - ソフトウェア品質の研究に使用
 - オージス総研では、早稲田大学の鷲崎准教授と共同でソフトウェア品質に関する研究を行っています。「Adqua」も研究成果の一つです。
 - この研究において、皆様からいただいたソースコードや品質評価結果を使用させていただくことで、「設計品質とソースコード品質の相関」など組み込み業界にとって有意義な研究を進めていきたいと考えております。
 - また、モデル品質との相関を分析する上で、ETロボコン実行委員会と共同研究を進める可能性があります。
 - 研究成果を一般に公開することで、業界の発展に寄与したいと考えています。

【参考資料】

ソースコード品質評価ツール「Adqua」のご紹介



ソースコード品質評価ツール「Adqua」の特徴

1.品質の得点化

品質を客観的尺度で得点化します。
品質のレベルを一目で判断できます。

2.問題箇所の特定

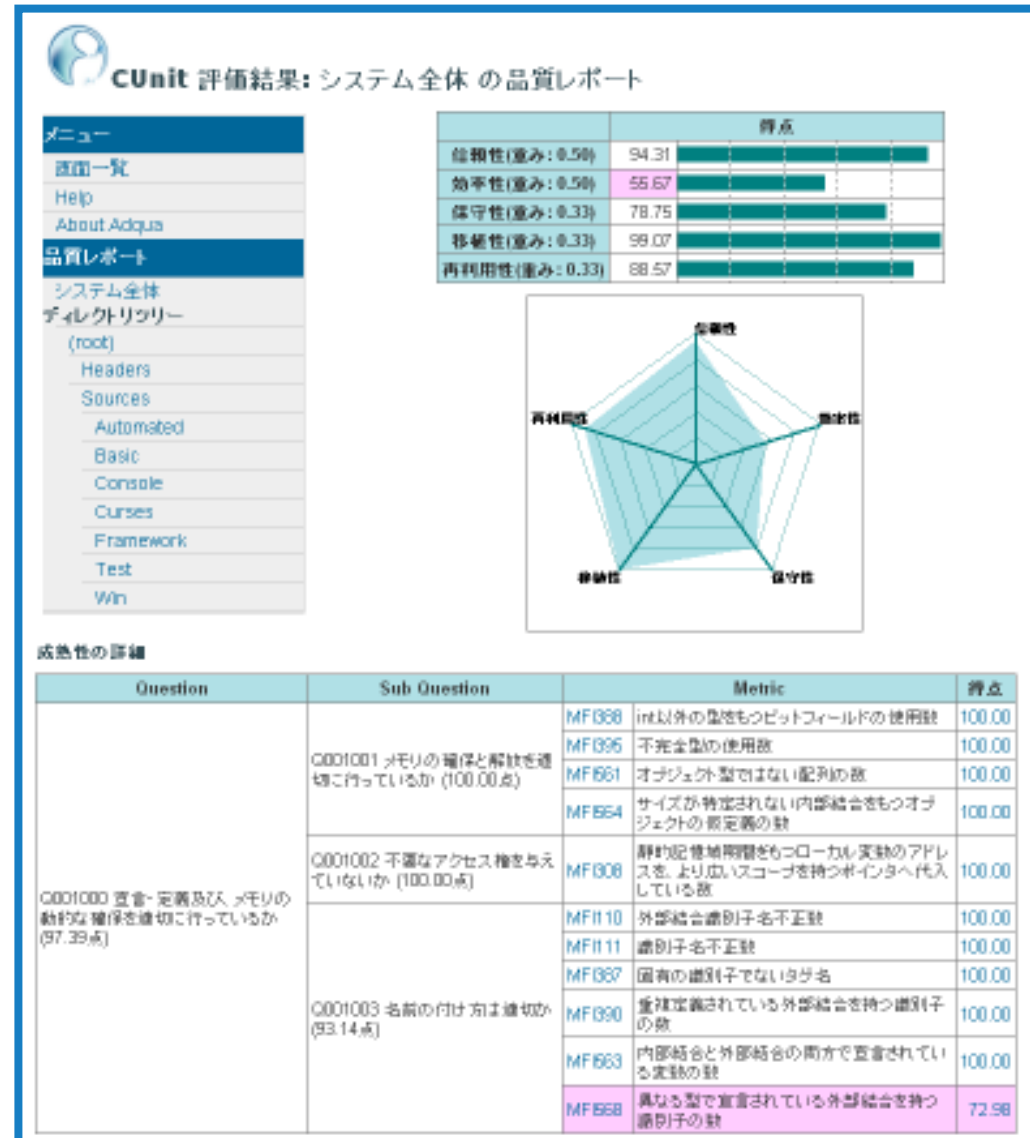
ソフトウェア全体の品質から、ソフトウェア要素の品質へトレースできます。
問題のある要素を容易に特定できます。

3.依存関係の解析

要素間の依存関係を解析し、DSMなどの設計情報として出力します。
設計レベルの問題の特定や改善に活用できます。

4.コードクローンの解析

コードクローンを解析し、クローンペアをリスト化して出力します。
コードクローンの特定や改善に活用できます。



1.品質の得点化 得点化のしくみ

GQM手法で作成した品質メトリクススイートを用いて得点化します。

品質メトリクススイートのイメージ

Goal

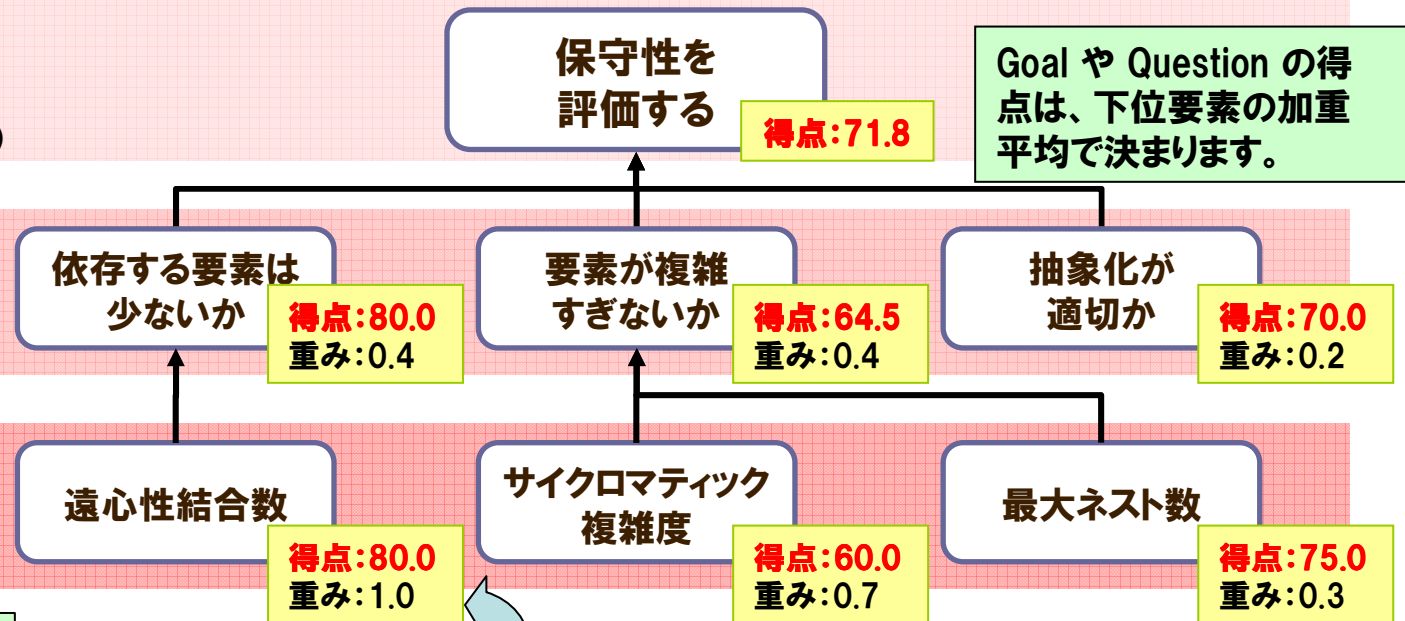
→ 達成したい目標
(品質レベルを知ること)

Question

→ 何ができていれば
Goalを達成できるか

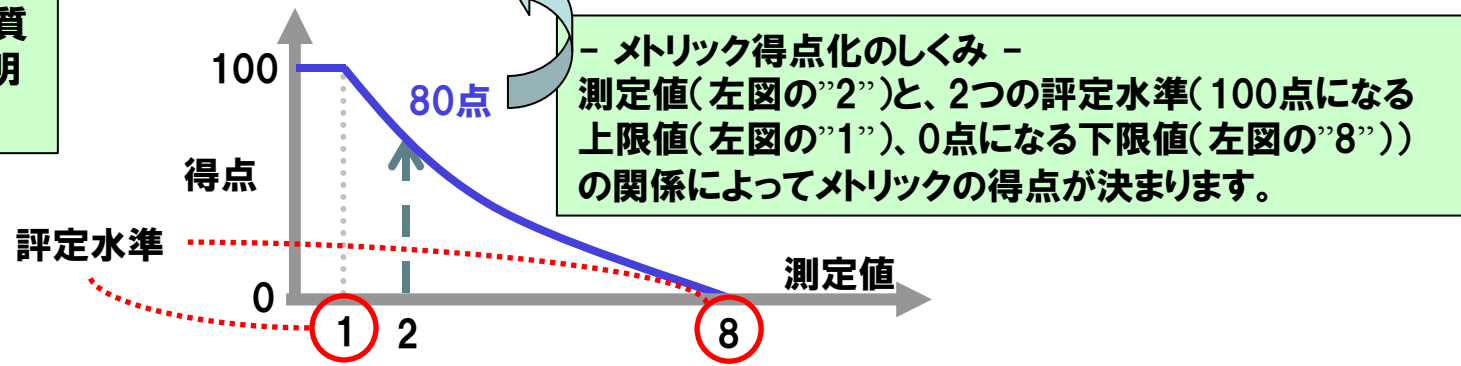
Metric

→ 測定可能な指標



Goal や Question の得点は、下位要素の加重平均で決まります。

GQM手法の採用により品質特性とメトリクスの関係が明確になります



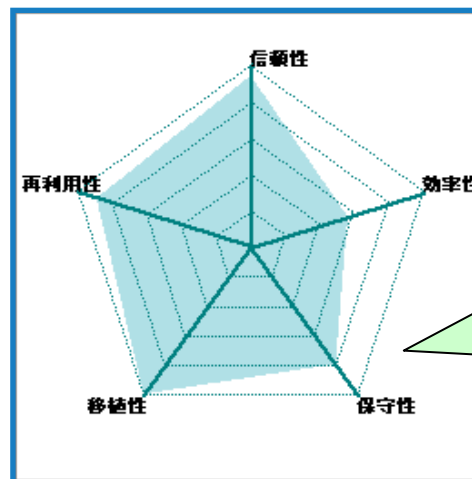
1.品質の得点化 品質評価レポートのイメージ

得点を見やすい形でレポートします。

品質レポート

⇒HTMLファイル形式で提供します。

	得点		偏差値(参考値)	
信頼性(重み: 0.50)	94.31	<div style="width: 94.31%;"></div>	58.20	<div style="width: 58.20%;"></div>
効率性(重み: 0.50)	55.67	<div style="width: 55.67%;"></div>	34.07	<div style="width: 34.07%;"></div>
保守性(重み: 0.33)	78.75	<div style="width: 78.75%;"></div>	62.40	<div style="width: 62.40%;"></div>
移植性(重み: 0.33)	99.07	<div style="width: 99.07%;"></div>	58.04	<div style="width: 58.04%;"></div>
再利用性(重み: 0.33)	88.57	<div style="width: 88.57%;"></div>	64.07	<div style="width: 64.07%;"></div>



一目で品質の良し悪しがわかります。

偏差値も表示されるので、他社プロジェクトとの相対的な品質水準がわかります。

成熟性の詳細

Question	Sub Question	Metric	得点
Q001000 宣言・定義及び、メモリの動的な確保を適切に行っているか (97.39点)	Q001001 メモリの確保と解放を適切に行っているか (100.00点)	MF1388 int以外の型をもつビットフィールドの使用数	100.00
		MF1395 不完全型の使用数	100.00
		MF1561 オブジェクト型ではない配列の数	100.00
		MF1564 サイズが特定されない内部結合をもつオブジェクトの仮定義の数	100.00
	Q001002 不要なアクセス権を与えていないか (100.00点)	MF1308 静的記憶域期間をもつローカル変数のアドレスを、より広いスコープを持つポインタへ代入している数	100.00
		Q001003 名前の付け方は適切か (93.14点)	MF1110 外部結合識別子名不正数
	MF1111 識別子名不正数		100.00
	MF1387 固有の識別子でないタグ名		100.00
	MF1390 重複定義されている外部結合を持つ識別子の数		100.00
		MF1563 内部結合と外部結合の両方で宣言されている変数の数	100.00
	MF1568 異なる型で宣言されている外部結合を持つ識別子の数	72.98	

品質メトリクススイートの構造にしたがって、詳細をレポートします。

2.問題箇所の特定制

成熟性の詳細

Question	Sub Question	Metric	得点
Q001000 宣言・定義及び、メモリの動的な確保を適切に行っているか (97.39点)	Q001001 メモリの確保と解放を適切に行っているか (100.00点)	MF1388 int以外の型をもつビットフィールドの使用数	100.00
		MF1395 不完全型の使用数	100.00
		MF1561 オブジェクト型ではないオブジェクトの仮定義の数	100.00
		MF1564 サイズが特定されないオブジェクトの仮定義の数	100.00
	Q001002 不要なアクセス権を与えていないか (100.00点)	MF1308 静的記憶域期間をもたない変数を、より広いスコープで宣言している数	100.00
		MF1110 外部結合識別子名不正数	100.00
	Q001003 名前の付け方は適切か (93.14点)	MF1111 識別子名不正数	100.00
		MF1387 固有の識別子でないタグ名	100.00
		MF1390 重複定義されている外部結合を持つ識別子の数	100.00
		MF1563 内部結合と外部結合の両方で宣言されている変数の数	100.00
		MF1568 異なる型で宣言されている外部結合を持つ識別子の数	72.98

システムレベルで
あるメトリックの得点が悪い場合、

得点をクリックすると、



原因となるディレクトリやファイルをトレースすることができます。

MF1568 異なる型で宣言されている外部結合を持つ識別子の数

評価水準(基準値)(ノース) 0.00

ファイル名	測定値	得点
Curses.c	6	0.00

MF1568 異なる型で宣言されている外部結合を持つ識別子の数

評価水準(基準値)(ノース) 0.0

サブディレクトリ名	対象ファイル総数	平均得点
Automated	1	100.00
Basic	1	100.00
Console	1	100.00
Curses	1	0.00
Framework	5	100.00
Test	1	100.00
Win	1	100.00

ボトルネックとなっている要素を特定できます。

3.依存関係の解析

要素(ディレクトリ、ファイル)間の依存関係に関する情報を提供します。

DSM(Dependency Structure Matrix)

⇒CSVファイル形式で提供します。

from→

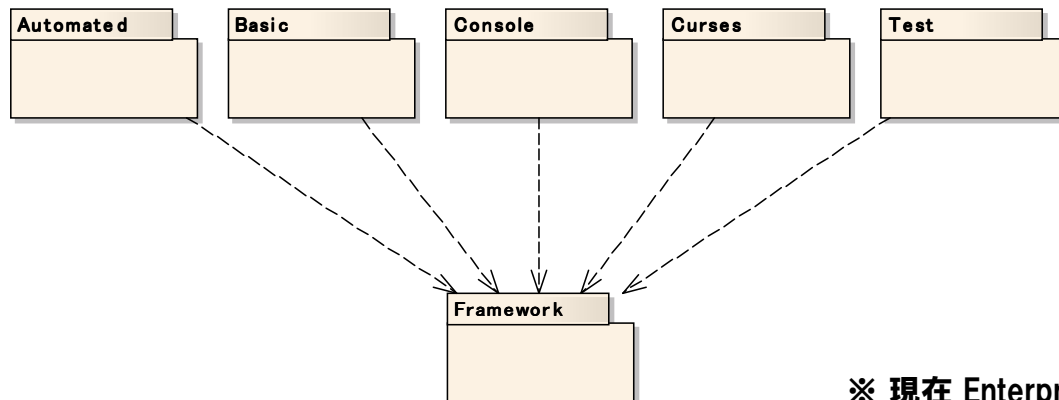
	/Automated	/Basic	/Console	/Curses	/Framework	/Test
/Automated	2	0	0	0	0	0
/Basic	0	0	0	0	0	0
/Console	0	0	0	0	0	0
/Curses	0	0	0	0	0	0
/Framework	18	14	18	19	34	1
/Test	0	0	0	0	0	2

Curses から Frameworkへの依存が 19 個あることを示しています。

DSMにより依存が多いモジュールを特定できます。

UMLモデル定義

⇒XMIファイル形式で提供します。(モデル表示にはモデリングツール [※] が必要です)



UMLモデル(パッケージ図)により依存関係の複雑さを概観できます。

※ 現在 Enterprise Architect と JUDE/Professional に対応しています。

4.コードクローンの解析

クローンが存在するファイルと行番号を示したクローンペアリストを出力します。

クローンペアリスト

⇒CSVファイル形式で提供します。

lhs_name	lhs_from	lhs_to	rhs_name	rhs_from	rhs_to
D:/src/Output.c	30	37	D:/src/Output.c	38	45
D:/src/Output.c	32	45	D:/src/Output.c	50	63
D:/src/Output.c	50	55	D:/src/Output.c	56	61
D:/src/Output.c	142	150	D:/src/Output.c	211	219
D:/src/Output.c	194	200	D:/src/Output.c	442	448
D:/src/Output.c	59	63	D:/src/Input.c	20	24
D:/src/Output.c	112	121	D:/src/Input.c	68	77

Output.cの112行目から121行目とInput.cの68行目から77行目がクローンであることを示しています。

どのファイルのどの行にクローンが存在しているかが明らかになります。