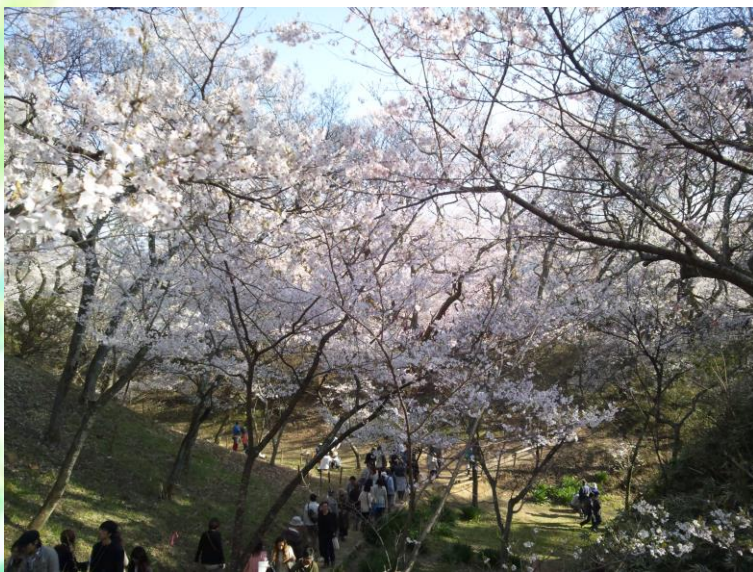


# WACATE WACATE マガジン Magazine 最終号

## ✿ご挨拶✿

WACATE-Magazine をご覧の皆様、こんにちは。  
WACATE-Magazine 編集部です。



今年は晴れ空が多く、桜+青空のコラボレーション  
を堪能できました♪

桜は旅立ちを連想しますね。  
新しい実行委員と  
新しいWACATE。

たくさんの新しい出会いが  
これからも皆さんにありますように！

それでは WACATE-Magazine 最終号、  
はじまるよー！

## ✿お品書き✿

### 【特集1】

WACATE2013 夏 申し込み締め切り間近！！  
最新情報

### 【特集2】

JaSST<sup>®</sup> 13 Tokyo Report!

### 【コラム】

WACATE2012 冬 BPP 受賞者寄稿！  
「ソフトウェアの品質ってなんだろう」

### 【リレーコラム】

せんぱいにきく

### 【雑談すべしやる】

てすバナ語るっcha!

### 【連載】

今月のSTAR☆

### 【連載】

源太郎のソフトウェアテスト占い

### 【Magazine】

WACATE-Magazine 振り返ってみた！

### 【あとがき】

## ✿WACATE Topics✿

WACATE2013 夏クロージング決定！

**WACATE2013 夏の日程決定！**  
**2013/06/22(土)~23(日)**

現在鋭意検討中！



WACATE 2013 Summer

# そのテスト、再考

みなさんは、普段どんな目的でテストをしていますか？ どのようにテスト分析を進めていますか？  
同値分割や境界値分析を意識的に使えていますか？

こうしたソフトウェアテストの基礎的な話題について、ワーク中心で、つまり皆さんが中心となって2日間考えて頂きます。

主体的に学びたい方、他の参加者の皆さんと疑問や意見をやりとりしてみたい方、  
ワークを通して自分の知識や経験を整理したい方、大歓迎です。



## テストクエスト ぼうけんのしょ 1

上條 飛鳥 (WACATE実行委員会)



## テスト分析発表 1日目の成果発表

藤原 洋平 (WACATE実行委員会)



## そのテスト、再考

中野さやか (WACATE実行委員会)



## みんなのテスト分析

上田 卓由 (WACATE実行委員会)



## 分けてみよう 悩んでみよう 同値分割・境界値分析

近江 久美子 (WACATE実行委員会)

*Closing Session*



## 細川 宣啓氏 (日本IBM)

欠陥とは何か？  
欠陥研究の発想、洞察力とその思考

### 開催概要

主催: WACATE実行委員会

日時: 2013年6月22日(土)~23日(一泊二日 4食付き)

場所: マホロバ・マインズ三浦(神奈川・京浜急行三浦海岸駅徒歩5分)

<http://www.maholova-minds.com/>

### 合宿費:

35歳以下(若手): ¥22,000

36歳以上(ベテラン): ¥25,000

※宿泊費、宴会費、食費、会場費、印刷費、その他運営にかかる事務費含む(キャンセル不可)合宿費は開催当日受付にて、現金でお支払いください。引き換えに領収書を発行いたします。

**WACATE**  
Workshop for Accelerating Capable Testing Engineers



# WACATE 2013夏

## 申し込み締め切り間近！！最新情報

開催まで1ヶ月をきったWACATE2013夏！！  
毎度ながら、メインどこのセッション担当者(実行委員)からのメッセージをお届けしようと思います。

ぶらすっ☆クロージングセッションにお招きしている細川さんと実行委員とで急遽**すバナちゃっとる一む**を開きました♪(細川さんにわがまま言いまくりました m(\_ \_)m)  
今回のWACATEに対する意気込み、想いといったところが読者のみなさまに伝わると嬉しく思います。

### ■テストクエスト ぼうけんのしょ1 (担当：上條 飛鳥)

若手枠の実行委員、あすかです。  
今回は皆さんに聴いてみたい事があるんですよ。

「あなたは、何のためにテストやってるんですか？」

私はようやく、小さな会社のひとつの部門で自立を始めたばかりの若輩者です。

会社、立場、役職、場面。  
色んな要素が絡まって、様々な理由が産まれるんだと思います。

「あなたは、何のためにテストやってるんですか？」

よろしければ、一緒にその答えの片鱗を見にいきませんか。  
若いと言われるみなさんと、共に探しにいけたら嬉しいです。  
もちろんベテランや中堅と言われるみなさんのご意見も伺いたいです。

「あなたは、何のためにテストやってるんですか？」

是非悩んで、自問して、考えてみてください。  
私も全力で悩んで、自問して、考えた結果をお話しします。

当日お会い出来ることを、心の底から祈っています。

### ■みんなのテスト分析 (担当：上田 卓由)

もう若手ではない枠の実行委員、上田です。  
あるシステムの仕様書をお渡しして、そのテスト分析をグループで行っていただくというセッション構成になっております。

「こんなテストが必要なんじゃない？」「これって仕様書に書かれてないけど飛鳥じゃない？」「えーそこまで考え

るのはちょっと細か過ぎ、、、」などなどワークショップを通じてディスカッションしながら、**どういったテストが必要であるかを分析**していけたらと考えております。

頭をフル回転させて、想像力を巡らして分析していきましょう！！

宜しく願いいたしますっ♪♪

### ■分けてみよう 悩んでみよう 同値分割・境界値分析 (担当：近江 久美子)

前回に引き続き、今回もテスト技法のセッションがあります。

今回のテーマは、「同値分割・境界値分析」。  
皆さんはどんなイメージを持っていますか？

「いつも使ってる」  
「テスト技法の書籍やサイトでよく見る」  
「よく目にするし、割と簡単そう？」  
「昨日も使った」

いろいろなイメージが出てくるかと思いますが、使い所が多い技法ですので、

「身近な技法」という回答が多いのではと思います。

そんな身近な「同値分割・境界値分析」について、今一度、考えてみませんか。

普段は何気なく使っている…という方は、知識の整理に。  
自信があるという方、逆に余りよく知らないという方は、腕試しに。

更に「同値分割・境界値分析」を身近に感じて頂けるセッションにしたいと思います！

### ■この他にも・・・

BPP セッションには、今回記事投稿して下さったしんすくさんが登場します！相当面白い話をしてくださる予感が・・・♪

2日目の朝には1日目の**成果発表会**、午後にはテストについて再考する**振り返りセッション**があります。

クロージングセッションについては、このあとのチャットと「せんぱいにキク」をご覧ください^^

おなじみのポジベセッション、ディナーセッション、分科会・・・いつもながら盛りだくさんです☆  
皆様の笑顔に会えることを楽しみにしています。



# WACATE2013夏を語る☆

ゲスト：クロージングセッション講師：ほそかわ氏  
実行委員参加者：  
うえだ、かみじょう、いせり、おうみ、こやま、ばん

## ■本題突入時刻 [2013/05/18 21:22:24]

ほそかわ：はい。じゃ、はじめましょう。  
今回のWACATEはテーマというか、中心となるべき何かトピックはあるのかな？  
うえだ：今回、タイトルを「**そのテスト、再考**」としております。  
ほそかわ：「サイコー！」でも、「最古」でもなく、「再考」なのね？  
うえだ：はい、そーですー  
ほそかわ：レビューだったら「**再鑑**」っていうよねwww  
うえだ：再考といったものをテーマとして「**テストの目的ってなんだろう**」っていうところからセッションを構成しております。  
ほそかわ：お。それ面白いね。「なんでテストすんのよ？」ってことだよな。  
うえだ：そうですね  
ほそかわ：「なぜテストするの？」って質問されたら、みなさんなんて答えるかな。。。  
うえだ：その部分は**上條さんのセッション**ですよ  
かみじょう：はい  
ばん：テストの目的というのと、テストという活動をする目的と、テストケースに対する目的とがあると思うのですが、どちら？  
かみじょう：前者ですわー  
うえだ：「**テストという活動をする目的ってなに？**」ということですよな。  
かみじょう：そうです。  
うえだ：いきなり、けっこう悩むセッションから始まる？  
かみじょう：**ガッツリ悩んでいただきたいですね^^**  
ほそかわ：いいね。頭抱えてしまうセッションwww「ノックから衝撃的」って大事です。  
かみじょう：でもそんなにこちらから衝撃を与える気はなかったりします  
ばん：わりと思いつく目的をあげて満足しちゃいそうなんですよな。もう少し俯瞰していくといろんな目的がみえてくる。  
ほそかわ：「目的を考えよう！それ大事だから！」って印象を与えるのと「これが目的でしょ！テストの！」って答えを与えるのだと、前者の方がよりインパクトが大きく初心者向き。後者がより扇動的なイメージでツウ向け。  
かみじょう：『**皆さんの“理由”を教えてください**』というのが私からの**唯一のメッセージ**です  
ほそかわ：そこで他人の意見聞いて「え？まじか？」ってなってくれるといいよねw  
かみじょう：ですわw

うえだ：人の意見が聴けるっていうのがいいですよな。「なんでそう思うの？」とか  
かみじょう：今回はセッション時間の大半を割いてまで、その時間を確保しています。  
ほそかわ：じゃ、そこひとつ面白い話。「テストの目的」って訪ねたら、多分みんなバラバラの答えを持つと思うのね。  
うえだ：そうですね  
ほそかわ：で、バラバラの答えになった上で、「なんでみんなテストの目的ってバラバラになってると思う？」って。聞いてみると面白いと思うよ。  
「**テストの”ほんとうの”目的ってなんなんだろうね**」って問いかけるといいと思うよ。  
テストのほんとうの目的って、時々私も見失う時があるもん。  
かみじょう：未だ見つけられているのかがわからないです  
ほそかわ：よくさ。。。好きなことを職業に出来るといいな。。。っていうけど、好きなだけじゃテスト屋とか品質屋とかは続けられないもんな。。。  
ばん：ちゃんと確認せずに作業に入っている感じはありますわ。  
ほそかわ：目的。。。って毎回確認してから活動しないと流されるwww（そういうのってみんなない？）  
うえだ：わからなくなる時はありますね。1つのプロジェクトが何年も続く場合、初期メンバーはいろいろ議論して目的とか作ってきてきたので、共有などは出来ていますが、途中から参画したメンバーは、そのへんがわからないでテストをやってる感があります  
ほそかわ：おお。時間に流されるパターンね。  
うえだ：そうですね。  
ほそかわ：「とにかく打鍵」とかっておおいもんなーwww  
ケース消化として実施するような  
かみじょう：カ業ですわ。。。  
うえだ：ありますわ。  
ばん：ちゃんと全員が目的を共有していないことはよくあると思います。メンバーが多いとか、委託とか。  
うえだ：「そういうルールだから」っていうのが目的みたいな  
ほそかわ：日本人の一番弱いやつね。>「そういうシステムになってますから」的な  
  
ほそかわ：さて。そろそろ今回のWACATE(のクロージングセッション)に向けた話をしようか？  
かみじょう：こくこく。  
ほそかわ：テーマでも質問でもなんでも突っ込んでくださいな。今回、ひさびさに初日の夕方から顔を出したいなって思ってます。ゆっくりお話する時間を作りたいなと。そして**今の若い人の悩みをききたいな**と。  
ばん：私自身はプログラムに関わっていないのもあって、実行委員側が細川さんにお話して欲しいなと思った経緯を簡単に説明していただけると嬉しいなー



ほそかわ: あ。それ聞きたい。いや、へんな話なんだけど、初めて来る人も含めてあんまり俺みたいな若造がしゃべってもそんなに参考になることないんじゃないかな。。って最近思っています。(「おらおら！いけ！」っていうパワープレイとかカ技以外、あんまり参考にならないんじゃないかな。。って) そういう意味では立ち位置見失ってますよ。<悩める43歳

かみじょう: .o0(原点回帰のようなイメージで、参加者一同を再加速させたいなーと。)

ほそかわ: 原点回帰。。。なるほど。

かみじょう: .o0(個人的な思いでございました。)

ほそかわ: 原点回帰ってこのだったら、実は、テスト屋さんも含めた品質屋さん全員に持って欲しい「思い」っていうのはあるなあ。。

かみじょう: 「思い」ですか

ほそかわ: テスト屋さん、品質屋さん全員、**「技に溺れるな」って思い**。なんていうんだろ。。。よく、品質屋やテスト屋さんって、勉強たくさんしてる人いるのよ。正確にいうと「勉強だけ」してる人。実践を伴っていない場合や、経験無いけど技法いっぱい知ってる人がいるの。

かみじょう: ”えりーと大学生(笑)”が沢山育ちそうな環境ですね

いせり: 耳が痛いですが自分も同感です

ばん: (技法の)使い方を知らずに過ぎてしまうことはありますね。使い方というか、適用のさせかた。

ほそかわ: 「腕ふるってナンボの世界」の世界で、いっぱいモノ知ってることは「前提」であって実践で苦しみモガくことは恐れてはいかんなど。

いせり: **開発は個人で実践できる技法多いですが、品質やテストは複雑な現場を相手にしたり、複数人でコミュニケーションを取りながら推進したりしないと実力に結びつかない技法が結構多いと感じています。**

ばん: 同値分割だって、そんなの普段しているからといって実はきちんとおこなっていないんですよ。実力に結びつかない技法が結構多いと感じています

ほそかわ: 例えばさ。。。

**問1) 「同値分割で見つけ出せる／出しやすい欠陥ってなに？**

って質問されたらどう答える？

いせり: 単機能バグ、組み合わせバグ、とかでしょうか？

ほそかわ: うん。ということはその「同値分割という技法」やらは、いろんなバグに効くってことね？

だとするとどんな目的のテストの場面で使うべきなんだろ。特殊用途？それとも一般的な考え方としてどんな場面でも使うべき？

ばん: えーと、そう、他の技法を導入するときに同値分割をうまく行なって欠陥を見つけやすくするということはあると思います。

いせり: はい。バグを探す技法の基本となる技法だと個人的に思っています

ほそかわ: うん。そしたらさ。。

**問2) その技法は儲かる？**

かみじょう: .o0(それは、「同値分割」で欠陥を出すことにはならないのでは・・・手段に過ぎないならば、矛先は内

部に向きます。そうなる、コストを削る側の道具になる。。)

いせり: 難しい問いで間違っているかもしれませんが、

ほそかわ: (いえいえ。自由に思ったことを答えて。<直感が聞きたい)

いせり: 自分は、テスト対象やドメインごとに特化すれば儲かると思います。例えば、同値クラスの観点リストとか、あるテスト対象について、定番となる同値クラスとかなら、ビジネスになるかもしれません。

かみじょう: .o0(井芹さんと真逆だ)

ほそかわ: ほうほう。>二人面白いよね。答えが真逆

いせり: ただ、純粹に技法の進め方をおうだけだと、お金にするのは難しいかもしれないと感じます。かみじょうさんの考えも気になります

ほそかわ: うん。そこ二人で話し合ってみて。「同値分割はなぜ必要か？そしてその技法は儲かるか？」

かみじょう: 同値分割をした上で他の技法を使うことは必須ですか？

いせり: いえ、そのまま単機能テストに展開することも可能です

かみじょう: 同値分割”のみ”で出来る事は、単機能以上に広がる可能性がありますか？

いせり: 他の技法を使わずに、ということでしょうか？

かみじょう: はい

いせり: あと「広がる」とはどういう意味でしょうか？

かみじょう: 同値分割を使って他の技法による、テスト活動の支援をするのなら、同値分割はラクするための手段にすぎないと考えます。

ということは、多少のコストを払って実施する内部に向けた活動なので、直接的利益はない≠儲からない

ほそかわ: 今の二人の会話を交通整理すると、上條君は「同値分割は技法にあらず。考え方の整理方法であって、同値分割単体ではどんなことができるか不明」で、

いせり君は「技法の副産物(観点リストとか)や経験／ノウハウならお金になるかも。。技法単体では難しいかも」という理解だよ。

かみじょう: はい。おっしゃるとおりです

いせり: はい。そうです

ほそかわ: えっとね。。。 **「正解はない」と思っていいと思うよ**。誤解が多いのよ。例えば同値分割とか「技法」とか。

同値分割って、有効範囲／無効範囲等、テストしている対象範囲に「偏り」があるか無いかを確認する時に使うよね。

で、そういう「技」というか「考え方」があると、(多分みんなもそうだと思うけど)初めてその技を聞いた瞬間って、「つかってみよう！そうか！こういう考え方が必要なんだ！」って思うでしょ。下手すると「私の追い求めていたものはこれだ！」って思う人すらいるかもしれない。

で、その技法の正確な定義や、その技法の魅力／使ってみた経験なんかをまた「おれ知ってるぜー！」って使ってしまう。

でもね。 **技術が生まれた時にあった「この問題を解決したかった」って問題部分が、技と一緒に伝承されないのよ。「技」だけが一人歩きするの。**

かみじょう: .o0(デシジョンテーブルを使って、これは顧客に伝えるには細かすぎると指摘されたことがあったな



あ・・・)

ほそかわ: (そうそう。そういうのも典型的な例だね。) 顧客は「バグを見つけさえすればいい」または「こういうバグを残留させないでリリースして」っていうところまではいうと思うの。でも、「どの技法でそれを取り除いて！」とは指定しない場合が多いんじゃないかな。どう？お客は技法を望んでいる？

かみじょう: 私の周りに限ると、そういった場合はゼロですわ。手段不問。いいものください。という感じです。

いせり: 自分もかみじょうさんと同じです

ほそかわ: おけ。じゃ、いせりくんに質問。開発がおくれて、なんかあやしそーな製品やシステムをテストしなきゃいけなくて、自分が「テストの専門家を誘って来てください。お金はいくらかかってもいいから」って言われたとするよね。その時にいせりくんなら「技法のプロ」を呼んでくる？

いせり: いえ、技法とドメイン両方を精通するプロを呼ぶと思います。

ほそかわ: おおおお。すごい冷静。もしそうなら、細川呼んで来てても役に立たないかな。(少なくとも、私はそんなにテストの技法を知らないよ？)

いせり: あ、すみません。理想論的すぎたかもしれません。

ほそかわ: (笑) <そうだよわ

えっとね。。。。「技法」をたくさん知っていると、いろんなことを出来る「可能性」が広がる。若いうちはたくさんそれこそいろんな技法を試してみ取り入れてたくさん頭抱えるのがいいの。でも、さっきいせり君が言ったみたいに「ドメイン」とか「大規模のシステム開発」とかを経験した人なら「さらっと」テストやれそうじゃない？そこそこの品質も確保しながら。

いせり: はい。感や経験でリスクベースな考え方ができるので、さらっといける場合が多いと思います

ほそかわ: うん。つまり勉強してたらそれにこしたことないけど、知識なくても経験である程度のところまで行けるのかな。。とも思う。

それと同時に「技」をいくら知ってても、対象を知らないと適用難しいよね。

この場合の「対象」というのは、

- 1) 対象業務や対象システム
- 2) テストで検出対象とする欠陥

の両方ね。

いせり: なるほど

ほそかわ: 前に銀の弾丸の話をごっかのシンポジウムで話したんだけど、若い技術者は「銀の弾丸」、つまり技に心を奪われ過ぎて、倒す対象のモンスター(=この場合、抽出すべき欠陥)のことを意外と研究しないんだよね。

で、さっき質問した意図なんだけど、もし、技を知っているのみだと、「その技は儲かるか？」っていう質問にはなかなか(一般的には)答えにくいんだけど、その技で検出できる欠陥を知っていると、「取らないといくらの被害か？」と「いつまでに取れるといくら儲かるか？」が正確に見積もれるよ。(あくまで見積もりレベルだけだね。)

いせり: なるほど。細川さんがおっしゃられた「技に溺れるな」というメッセージの背景が、ようやく分かって来ました

かみじょう: 損益勘定はまだまだ苦手・・・

ほそかわ: うん。もう少し話しを進めると、顧客は「起きて欲しくないシステム障害」までは説明できるよね。「発生して欲しくない事故」や「起きて欲しくない不具合」これはテスト屋や品質屋が顧客と対話して決めていけるし、合意ができる。(むしろそれが仕事だと思ってるよ。)でもテスト屋が提示する「技法」は、残念ながらどんなにレアでどんなに高等なテスト技法でも興味ないと思う。

それよりも顧客がお金を払うのは「信頼できる雰囲気」や「期待できる人格」や「成果」だったりします。

こやま: よくわかりますね。「金を払うのは雰囲気や人格や成果」。身に沁みます

ばん: 現場レベルだと人格とかなんですけど、そこにたどり着くまでの間がそうじゃない気がしています。

ほそかわ: もうひとつのポイントとして、先輩から教えてもらった一番嬉しいものってなんだろう。みんなが教わったら嬉しい/伝承してくれたら嬉しいものってどんなもの？>ALL

うえだ: 「仕事の仕方や態度」みたいなところになりますわ

かみじょう: 「楽しく仕事をこなす方法」です。心底楽しそうに仕事している先輩方を見ると、ああなりたいなあ・・・と思う事がよくあります。そんな方法や雰囲気を伝えたいです。

ほそかわ: ほうほう。他には？

こやま: 伝承してくれたら嬉しいもの、は「魂」とか「矜持」かなあ。

いせり: うれしいのは、プロとしての価値を高め続ける姿勢や考え方、でしょうか。先輩が離れても、残った人たちを成長させていく考え方だったり、アプローチだったり

ほそかわ: まず今の意見整理してみるね。「魂」「矜持」「姿勢」あたりは、それぞれ個人差あるから、企業の中でも、業界中でも伝承しにくいよね。。。。

うえだ: あ、ごめんなさい、私が書いたのは「伝承しないと困ること」な気がする

こやま: 引き継ぎのときとかは、経緯と意味、理由などがうれしいですが、今回のとは少し違うかなあ。。

ほそかわ: あんまり根性や矜持っぽいのは伝えきれないな。

こやま: ほそかわさんの魂は、ほそかわさんの背中から透けて見えるかなあと思いますよ。魂がないことはないと思います。

ばん: 先輩は教えてくれるじゃなくて学ぶ環境を整えてくれることが多かった気がします。どうやったら学びやすくなるんだろう？と考えて入口まで導いてくれましたね。だから何かを教わったというより支えてもらったという感が強いです。先輩ってどっちもあるんじゃないですかね？示す人とサポートする人

ほそかわ: ティーチングとコーチング、どっちがいい？って話だよな。

ばん: 前者は教わったってイメージがあるけど、後者は教わったっけ？でも先輩がいてくれたから今の自分がある、みたいな。はい。そうだと思います。ティーチングとコーチング



ほそかわ: コーチングのありがたみは、後々まで気がつかないのがおおいと思います。

いせり: ちょっと前にコーチングの講習受けて来たんですが、ありがたみは同感です。受け手が問題意識を持っていないと効果を出すのも難しいですし

かみじょう: .o0(自分がやってきたことはコーチングなのかティーチングなのか・・・)

ほそかわ: そう。それ大事。<問題意識

いせりくんが「プロとしての価値」ってさっき言ったのすごく大切なんだけど、「儲かる技術」「普遍的に活躍できる技術」みたいなものって、実は「銀の弾丸」だと思ってる。

先輩から教わって、実は一番「価値がある」のは、汎用的な「コツ」とか「ノウハウ」なんじゃないかな。。。って思う。単なる業務知識だけでなく、単なる技法の適用例でもなく、「その先輩が考えた経緯も含めたコツ/ノウハウ」がもらえるとうれしいよね。<すぐ実践できるという意味でも。

うえだ: コツってありますね

かみじょう: 即物的なコツは若手にとって助かります

こやま: 同意です。>コツと経緯

ほそかわ: だから、私はいつも人前に立ってお話をする時は、(くどいしゃべりと言われても)コツやノウハウを伝えるようにしていますよ。「xx 技法」なんて得意げに説明するのって、多分プレゼンの技法くらいしかないと思う。

いせり: なるほど。確かに、WACATEに参加されたテスト初心者の方から、技法の使い方は分かった。じゃあ目の前の問題をどうすればいいの?ってという悩みを、分科会で頂いた事がありました

ばん: 目の前の問題は明確だったんでしょうか?目の前の問題が明確なときと、目の前の問題がそもそも整理されていないときがあると思うのです。目の前の問題が明確であって、それが技法と結びつかないという問題であれば、技法を適用する際に考えることなどのノウハウがあるといいね、という話になりそうですね。

ほそかわ: 俺の部署でよく言ってた話なんですが、**技術者とは「知識を仕入れて使ってみて技術に昇華させる人」のことを技術者と呼ぶのです。**って話があってな。

かみじょう: (:・(株)・)・・・!

ほそかわ: 先輩から「技術を盗む」っていうけど、伝達できるものの多くは「知識」でしかなくて、それを**技術**だと勘違いする若手も多い。

こやま: 知識や情報を「技術」だと勘違いするのは、よくわかります。

ほそかわ: だから教える側は、知識を技術に「昇華させた経緯」を教えなきゃいけないよね。。。ってよく話してた。

うえだ: なるほどです

ほそかわ: その技術に昇華させた経緯を「コツ」と呼ぶのです。だから、知識伝達の間として勉強会を開催したらいいかな。。。って思います。

こやま: おおー。これはぜひみんなに聞いてほしい話かも!

ほそかわ: え?WACATEで?

こやま: あ、はい。ってもちっと経験積んだ人のほうがいいのかなあ。。

ほそかわ: 初心者でもベテランでも、多分普遍的だと思う。いせり: **経験積んでいるのと、知識を技術に消化したいって問題意識も必要そうですね**

ほそかわ: うん。

ほそかわ: 勉強会の場で覚えて持ち帰るべきは、「目の前にこんな問題があった時にこの技法は有効」っていうような、技法の中身/知識的なものではなく**技法の選択方法/思考のほう**が価値あるんじゃないかな。。。って思うよ。「この場面ではこうやって選ぶのよ」。。。って。

**皆さんは、ほんとうに技術者ですか?たくさん技法を知っている「単なる物知り」ではありませんか?**ってというのが多分究極の質問だと思う。

ばん: そうそう、何を選ぶか、というのを経験させるのはよいです。

かみじょう: 昨日エンジニアから役所勤務に変わった友人が叫んでいた「やっぱモノ作ってナンボだよ」という言葉がリフレインしています。

ほそかわ: そういうのも面白いよね。モノ作ってる時の方が、達成感ある。。。って人もいるね。逆に「物知り」でなくては勤まらない職種もあるよ。弁護士、医者知識のうち、実際に活用したことのある知識は極めて限定的だからね。

うえだ: 「自分は技術者なのか?」っていうのはけっこう悩んでるところです。。。。

ほそかわ: おお。段々と実行委員が「技術者」について思考ハマりだした www

こやま: 自分が技術屋か?というのが実は割と今痛いところです@@

ばん: 技術を直接使わないから技術者じゃないかというところではないと思っています。

ほそかわ: (そこまで行くにはかなり時間がかかるものなのよ。年食ってくとその域に達するんだが)

ばん: あーそうですね^^; WACATEのレベルよりもう少し上になるかもしれませんね。

ほそかわ: 勉強してつけた知識はすぐに使ってみたくするのが**技術者の魂**だと思う www

うえだ: 「すぐ使ってみよう」っていうのはありますね。

いせり: 分野によりけりな感じですが。ただ品質やテストの技術は物知りとなっている所あると感じています

ほそかわ: うん。いせりくんいいね。<バグ以外の技法/技術ね。今、バグだけの専門家になりたいけどな。

こやま: 何か知ったらとりあえず使ってみる、っていうのは癖にはなっていますね^^; あんまり残らないけど^^;

かみじょう: 最近知ったものをガシガシ使っては首をかき上げています..

ほそかわ: うん。それ「知的好奇心」の充足っていう人間の基本欲求と関係ありそう。

うえだ: 「バグ」のことを考えると「なぜ人はミスするのか」っていうところについてしまうのですが、..

ほそかわ: うん。それでいい。それが普通だよ。全ての欠陥は人工物だからね。("All defects are also artifacts"です)



うえだ：なので、未然に防ぐことを頑張るより、どうやって発見できるかっていうほうが現実的だと思っています。

ほそかわ：おおおお。なるほどな。よく、「どうやってバグの混入を防ぐか」まではみんな考えるけど、バグを「生み出さない」っていうそもそも発生を止めるって考え方はあまりしないよね。

かみじょう：こくこく

うえだ：そうですね

ほそかわ：生まれる瞬間を捕まえられる／把握できたら、かなり止められるんだがな。。。 (ニヤ)

欠陥のライフサイクルも、レビューやテストでバグをどう扱うか？っていう欠陥対処のライフサイクルも。ちょっと今考えているものを紹介しますね。>当日

こやま：あとは、ミスしたときにどうその影響を減らすか、ですかねえ。

ほそかわ：あー。影響軽減の技術は、バグそのものの扱いでも、検査技法の扱いでもなく、「事故の管理」っていう世界です。(インシデントマネジメントっていう分野)

こやま：なるほど。その分野も興味深いですね。

うえだ：バグって人工物だし、人じゃなきゃ、バグって認識できないんですよ。

ほそかわ：うん。日本ではソフトウェアの研究はまだまだだね。

うえだ：そのあたりがバグの面白いところだと思ってます

ほそかわ：不思議なことに静的解析ができれば、「自動修正」も出来そうなもんだけどね www<できない理由があるのよ。

うえだ：え、そうなんですか

かみじょう：自己修復するソフトウェアってあっても不思議じゃない気がするんですが・・・

#### ★この間、おふれこころー

ほそかわ：ちなみに、他の皆さんもリクエストってありますか？

うえだ：先ほどの欠陥研究をしない怖さはみなさまに伝えたいですね。(だいが考えさせられています)

ほそかわ：おお。欠陥の知識って、意識して持ってないとテスト屋さんは致命的だと思う。<個人の私見としてだけど、西さんも野中さんも同じ意見だ。

ばん：テストの欠陥についてって考えたことはありますか？

ほそかわ：面白い。テストの欠陥。よく、テストで「漏れた」っていうよね。

ばん：はい。テスト担当者って品質が？って言うわりに自分の成果物の品質は考えなかつたりしがちなので。

こやま：そこ、すっごく大事なんですよー。成果物の品質がしっかりしてない人間は信用されません。

ほそかわ：「漏れたテストはテスト屋の責任か？」って話も含まれるね。(ああ。。。ごめんなさい。ごめんなさい。俺技術者とか品質屋の前に、人間としての品質が低くて。。。)

かみじょう：.o0(誤解とかを承知で言うなら、冬にWACATEに来る気がなくなるような内容がいいなーと。来ている隙

がないほどやることと活かに満ち溢れたイメージ。なので「動き方」を伝えていただけたらとリクエストしてみます。)

ほそかわ：「勉強会でてる場合じゃない！」ってあせるほど、インパクトあるもの。。。ってことね。

かみじょう：はい

ほそかわ：(インパクトなのかな。。。命題の提示と技術者としての刺激の両方なんだろうなあ。。。<バランス。。。かな)

かみじょう：(両方になってしまいそうです。無茶なリクエストで申し訳ないです。)

おうみ：私の思いを書き出してみました。全部、という意味ではなく、例えばこの中のどれか、という程度でお考え下さいませ。

1：ファールの話…旬なので！

2：品質エンジニアとしての話…WACATEなので、テストの話も踏まえつつ視野を広げられるようなものということ

3：「デキる」「貢献する」エンジニアの話…テストって、板挟みにあったり、「新人さんの仕事」といった印象を持つ人もいるので、その辺りで刺激を受けられるような話。

4：(上記のいずれか1つ以上とあわせて副次的に)プロのプレゼン…テストをする上でも大事な「見せ方」を考えてもらうきっかけ

細川さんはいつも、ワーク中は優しく各班にアドバイスし、プレゼンでどーんとめめていらっしゃる印象なんです。今回も、両方みられると嬉しいな、と。(要望多くてすみません…)

また、いわゆる「テスト」ド直球以外の話しもあった方がいいのかな、と私は思っています。先ほどあげられていた、テストの品質も、なので共感します。

ほそかわ：おうみさんの「思い」は伝わる。。。もう思いとしてしっかと受け止めたw

品質エンジニアってのは、どっかで連呼したいなあ。。。<こういう職種／プロもいるぜ>って主張しときたい点

おうみ：それは、ぜひ聞きたいです。

ほそかわ：あー。ちょっとへんな質問になるけど、「**デキるエンジニア**」ってどんなんだろ。みんなの持つイメージ。

みんなのご意見を。>ALL

「誰か」ではなくて、もしイメージのある人がいたら、その人は「どんな人」なの？

かみじょう：ずっと「楽しそう」にしているエンジニアさんですかね

ほそかわ：「いえーい！」っていう勢いは大事だよ

かみじょう：そう思います

おうみ：私のイメージは、ざくっというと、プロジェクトに貢献し、かつ、チームに貢献する人。技術面、人間面両方という意味です。

ほそかわ：その場合の貢献というのは「ドライブ」することだね。組織やチームをまわす人。

おうみ：あー両方です。

うえだ：「この人に頼めば課題をクリアしてもらえる」という安定感



ほそかわ: 期待値が高い人?

うえだ: はい

私がイメージしてる人たちは「あまりしゃべらない」人が多いかも

ばん: あたしがイメージしている人たちは、ダメなひとが多いかも (え

いせり: 自分は、自己犠牲をせずに、QCD を顧客が期待する以上の水準で達成できる人、なんて感じています

ほそかわ: それ大事! <自己犠牲しない人。「おれこんなにがんばってるのに!」って言わない人だな www

ばん: 広い視野をもって物事を判断して1番よいものを選び実現する人、かなあ・・・

ほそかわ: 視野狭窄な人は、チームの勢いにブレーキをかけるからね。。

ばん: そうですね T-T

こやま: 何を言っても頼んでも、安心できるアウトプットが出てくる人、かなあ。

ほそかわ: 「安定」は、別の言葉でいうと「ぶれない」ともいえるね。

こやま: スタンスがぶれないイメージですね。

ほそかわ: (俺みたいにレビュー技法を説明して、数年したら「欠陥だ!」って過去否定してるようではだめなんです)

こやま: いやいやw レビューで直接取り除けるのが欠陥なので、一貫していると思います。

ほそかわ: おそらく、おうみさんの4つの質問の中で、この「できる」とか「貢献する」ってのが一番回答が難しいだろうなあ。。。。多様な価値観の参加者を画一的にリードしたりするのは難しいと思う。

こやま: 難しいですよえ。。

ばん: 貢献って、しよう、と思ってるのは難しいし、なんか自己満足で終わりそうな気がするのです。やってたらそれがなんか貢献につながっちゃったーあははー、みたいなのがよいのではないかなあ、と。

おうみ: そうか、そうですね。 >難しい

ほそかわ: (貢献は狙ってるものか? っていうのだけで1テーマになりそうだし www)

### ★しばしオフレコタイムー♪

ばん: あとひとつ質問というかインタビューになるのですが、

ほそかわ: はい。どうぞ。

ばん: 「俺の話聴く前に、こういうことを考えてみられる? こういうことを周りの人と話ってみてくれる?」といったことがありましたら教えて下さい。。クローズングセッションの予習として。

ほそかわ: すげー。。。。そんなにエラそうなこと言えない wwwwww なんだろ。。クローズングセッションの前か。。。

例えば、昼飯のカレー食ってる時でもいいし、夜の分科会でもいいけど、

「テストエンジニアはどこまで行ったら完成か?」っていう命題を夢も含めて話ってみてください。

これが質問かな。テストエンジニアに限定しないで「品質に携わるものは、どこまで行ったら完成か?」っていうのはどうでしょう?

いせり: おお、高度な命題ですね

ばん: 完成ですか?! 完成っ?!

ほそかわ: うん。

ばん: 完成させたいんですかねえ?・・・ってところも含みそうな。

かみじょう: .o0(早速頭から煙が出始めまし)

ほそかわ: (考えたことなかった?)

かみじょう: (完成形を考えたことはなかったです。もとより周りを巻き込んで楽しく進みたいだけの人なので。)

ほそかわ: ちなみに、私自身は「完成した姿を考えたこともない」っていう立場です。完成できるはずがない。。。ってのと、そもそも完成系がイメージできん。でも、なれるかどうかは別にして) 憧れる姿=完成系だとしたら、「どんなバグでも見つけてみせる」って言い切れる人になりたいな。

自信と根拠を持って「かかってこい」って言い切りたい。そういえば「完成」じゃないかな。でも完成じゃないだろうなあ。。。学び続けなきゃだし

ばん: 「かかってこいや!」というのは、多くの人が憧れていると思います。良く聴きますね。(あ、どんなバグでも見つけたら、かかってこいや!という話です。)

ほそかわ: あー。それ言ってみたい。言ってみたい。すっごい言ってみたい。

そうなりたいから欠陥研究したいんだろうな。きっと。武器の使い方じゃなくて、武器の性質じゃなくて、攻撃する相手を知りたいんだと思うわ。

ばん: あと、完成というより「今の1番」なのかもしれないね。

ほそかわ: あー。医学でもそうなんだけど「この人が無理と行ったら諦めましょう」の世界ってあるんだよ。そこまでは少なくとも行きたくない。

ほそかわ: あ。で質問なんだっけ?

「考えといてください」っていう質問は、「どこまで行ったら完成ですか?」っていう点を考えてみるといいと思うわー。

人にモノを教えたり、指導したりする時、「どこまで行きたい?」って聞いてから教えると、すっごく目的がはっきりして教わった満足度が高くなるよ? <教育学の分野にそういうのがある。

かみじょう: (最近はどうせ楽しく進むならかっこよく、です。ゴールをしめして~のくだりはしっかりと学んだ記憶があります笑)

ほそかわ: おお。それ健全だな。バグをたくさん集めて来て標本箱に入れて、「うひひひ。にへにへ」ってしてる俺より全然健全>周り巻き込んで楽しんで)

かみじょう: (何でかっこ良さを出すか考え中で、よく煙が出ています)

ほそかわ: 「かっこよさ」ってどうやったら伝えられるかな。

かみじょう: 「かっこよさ」は、「後で気付く」ような伝わり方をするのかなと



.o0(一緒に動いた人が、今度自分が関わる人にも同じように振舞ってあげようと思ったなら、「かっこよさ」は伝わったといえると思うですね)

ほそかわ: 「かっこよさ」ってのは、だれもまねできねええ！って思わせたらかっこよさじゃないのよ。一部分でも「ここ真似てみろよ」って伝えるものでな。

かみじょう: .o0(よかったイメージぴったり)

ばん: 確かにそうかもしれませんね。真似られるっていうのは大事かも。

ほそかわ: 多分、加減してその人が真似できる程度に「弱めて」伝えるのも大事。<加減が難しいけどな。

後、「こういうかっこよさもあるのか」っていうのを、視野の広がり伝えるべきだよな。

こやま: ですね。

かみじょう: 視野狭窄は恐怖を呼びますね・・・

ほそかわ: 今回、予告編として、チャットでネタをバラすとしたら、

今回のクロージングセッションで使うプレゼンの技法はありません。プレゼンの技法ではなく、プレゼンの最初に、**「どうやってこのクロージングセッションの内容を構成したか」その誕生秘話を説明します。**

かみじょう: 誕生秘話！

いせり: おお

ほそかわ: 普段、「細川さんってどうしてあんなに得意げに自信もってプレゼンするんだろ」ってスゴく言われるのよ。なので、その秘密を説明します。題して、「得意げにプレゼンをするにはどう組み立てるか？」です。プレゼンの技法ではなく、構成方法。

★内容は当日のお楽しみ・・・♪

ほそかわ: さて。気がついたらこんな時間なんだけど、最後。

**この夏、WACATE にてほんとうの技術者になる方法を教えます。全ては三浦海岸にて！乞うご期待です！**

おうみ: (かっこいい！)

ばん: 本当の、ってところを考えさせられそうですね。よろしく願いします！！

ほそかわ: うん。2日間技法たっぷり学んだ後の若い技術者に対して、**価値観崩壊させて終わらせる www**

かみじょう: .o0(楽しそう・・・ww)

ほそかわ: はい。というわけで締めましょう。

かみじょう: (インパクト負けしないように仕込まないですね)

おうみ: かみじょうさん、まかせた！

かみじょう: お疲れ様でした！ ありがとうございます  
m (、m

ほそかわ: うん。みんなが運営する WACATE だからね。

おうみ: お疲れ様でしたー

ほそかわ: おつかれさまでした！

うえだ: お疲れさまでしたー本日はありがとうございます

かみじょう: (まかされた・・・！)

ばん: いろいろありがとうございます。おやすみなさいですー^^

いせり: おつかれさまでしたー！！

こやま: zzz (←すでに寝落ち)

[1:50:34]おひらき

・・・なげーよー(T◇T)(←引っ越し前夜にここまでひっぱられた管理人のこころの声)

いやいやいやwそれでも長いだけのことはあってですね、ものすごく濃い話をたくさん聴くことができました。実はもっともっと面白い話があったんですよ♪そこは実行委員の特典ということで^^ふっふっふー

当日もみんなでアツく語りましょう☆三浦にGo!!!

(by WACATE 実行委員会)



# JaSST'13 Tokyo Report

## —Japan Symposium on Software Testing—

さる 1/30~31、目黒雅叙園にて JaSST' 13 Tokyo が開催されました。



言わずとしれた日本最大級のソフトウェアテスト・品質をテーマにしたシンポジウムも 11 回目を迎え、のべ 1700 人前後の参加者が雅叙園に集結しました。

恒例となった JaSST Tokyo レポートですが、今回は基調講演、招待講演、クロージングパネルに絞ってお届けします！

詳しくはぜひ、各所のレポートをご覧ください！

### ■基調講演

「Challenges in Software Testing ソフトウェアテストのチャレンジ」

Dorothy Graham 氏 (Software Testing Consultant)

### ■各年代におけるテスト

1970 年代から現在までのテストについてまとめる。

#### ・70 年代

開発、テストの区別がない。

#### ・80 年代

テストツールが出だしてくる。

また、IEEE829 などが策定される。

#### ・90 年代

テスターのキャリアが出来上がってくる。

また、開発者とテスターの比率が変わってくる。

#### ・00 年代

テストの自動化が台頭する。

ソーシャルメディアの出現など新しいものがでてきたため、 TESTING に関しても従来の方法では対応できなくなってくる。

### ■ISTQB

ISTQB (International Software Testing Qualifications Board (2002 年)) の出現で、重要な部分に集中できるようになる (共通認識を持つ)

TESTING についての認知があがってきたことにより、キャリアが出来た。

テストはだれても出来るという認識から専門知識が必要という認識にかわった。

### ■まだ変わっていないこと

テストに関して、マネージャーの認識が変わっていない。(誰でも出来るという認識)

テストで何が出来て、何が出来ないのかが明確になっていない。ツールを買えば OK という認識。

過去から学ぶことがかけている。(学ぶことが出来ない)

### ■DDP (Defect Detection Percentage)

テストに関する成果報告とは、リスクに関する情報 (価値=Value) を提示することになる。

開発したものの品質や傾向については、DDP (Defect Detection Percentage) を用いることにより明示できる。DDP を継続して使うことにより、開発に何が起きているかを見ることが出来る。

時間の経過とともに、そのように変化していくかを観察することが題字。

DDP はテスト工程でどの程度不具合を検出できたかを示す値となる。

DDP の計算は「テスト工程で検出した不具合数」を「不具合総数」で割ることでパーセンテージを出すことが出来る。※不具合総数は、開発全体に対しての DDP を出したい場合は、テスト工程で検出した不具合数 + 市場で発見した不具合数となる。

テスト工程で検出した不具合数が 150 として、不具合総数が 50 である場合、

$$150 / 150 + 50 = 75$$

となり、DDP は 75% となる。

テスト工程での不具合検出率は 75% であると言える。

不具合総数が必要であることから、DDP はテスト終了後に有用なものになる。

DDP は時間経過による値の推移に着目する指標であることから、一部分を切り出して「75%だからよい、わるい」といった判断をしてはいけない。(数値自体が重要なものではない)

### ■DDP の傾向分析

アジャイル開発に置いては、スプリントごとに DDP を算出することで、テストがうまくいっているか、うまくいっていないかを測ることが出来る。

スプリントを重ねていくごとに DDP が下がって言っている場合、「テストがうまくいっていない」逆に DDP が上がっていく場合、「テストがうまくいっている」と言える。



DDP が低下した場合、その理由や原因を見つけることが大切。

「DDP の数値が 95%以上の高い値の場合」  
ユーザーがまだ使っていないため、DDP は高い数値になっている可能性がある。

「DDP の数値が 60%以下の場合」  
あまりよくないテストを行っているかもしれない。  
製造しているものの品質が悪いのかもしれない。  
テストの実施に関して、外部要因の影響であまりテストがうまく出来ていないかもしれない。  
上記のように仮説を立てて、テストの改善点をさぐる事が出来る。

DDP は数値や制度よりも**一貫性が重要**。  
導入するにあたっては、まずはシンプルに行うことがよい。

#### ■テスト自動化

テストを自動化する目的について考える。  
テストを自動化することで、**不具合の検出率が上がるという認識は間違っている**。  
自動化が不具合を見つけるのではなく、**テストが不具合を見つける**。  
(自動化することにより、単位時間あたりにこなせるテスト実施が多くなるため、不具合の検出も上がる)  
要は、自動化したテストを人の手で行って、不具合の検出は同じになる。自動化はなんのために行うのか？それは「**よいテストを効率的に行うためである**」。  
質の悪いテストを自動化しても、質の悪いテストしか出来ない。

テスト自動化は、テスト実行時は、手動によるテストよりも効率的に実施することが出来、単位時間あたりに消化できるテストケース数が向上するが、準備工程（自動化の適用個所の分析、テストスクリプトの作成、ツールのメンテナンス等）に時間があかるため、自動化の導入は安易に行くと失敗する可能性が高い。  
回帰テストにおいて、自動化で発見できる不具合の割合は 9%程度である。

**不具合を見つけるには、「探索的テスト」が有効である。**

#### ■テスト動向

テストの方向性が変わってきている。  
より消費者指向になってきている。(Customer Driven Testing)  
よって、テスターはより**コミュニケーション能力が重要**になっていく。  
また、MBT(Model Based Testing)によりテスト自動化が進み、人間はより設計行為に集中することが出来るようになる。

#### ■質問

会場からの質問と回答。

Q:探索的テストを自動化する際の仕方は？

A:探索的テストを自動化するのではなく、**探索的な性質を持つ自動化を行う**。

例えば、性能測定を自動化する。性能測定自体は探索的テストではないが、測定結果から探索的テストの材料にする。

Q:DDP をアジャイル開発に組み込むとするとどうやって使っていくべきか？

A:テストの総数として DDP を使う。(DDP はトレンドを見極めるために有効な指標となる)

やってはいけない使い方として、個々のテスターを評価するために使うこと。

人の評価に使った場合、正しい数値が報告としてあがってこなくなる可能性が高いため、DDP の数値が正しくなくなる。

#### ■招待講演

「ソフトウェアテストについて

—無形労働の視点から—

岸田 孝一氏 (SRA)

SRA の創業者のひとりであり、日本で有名なプログラマーでありながら、絵描きを目指しているという岸田氏。今回の招待講演では、芸術的な独特の視点からソフトウェアの世界を覗きこんでみた、そんな感覚の一風変わった講演でした。

#### ◆抽象絵画とソフトウェア

美術の目的は「目に見えるものを、再現して写すことではなく、目に見えないものを目に見えるように実体化する」ことだそうです。プログラミングも共通する所があり、PC の中で行われる目に見えないプログラムの実行プロセスを見えるようにします。これは抽象絵画に似ているのだと岸田氏は言います。

抽象絵画とは、現実を抽象化していくことではなく、ただ絵を描くだけ。色、形、肌触りを表現するのみで、何かを現しているのではないのです。この抽象絵画の要素をソフトウェアに当てはめるとどうなるか？3要素のうちの一つ、形はプログラム構造ではないかという切り口で講演が進みました。

#### ◆プログラミング構造化

1つの(入口から出口に通じる)プログラムの構造は全て、連節・分岐・反復の組み合わせでできているというのが、プログラム構造化の定理です。



この構造定理に触発されて、自分のプログラム設計方法を書いたのが、構造化プログラムだと言います。

しかしこの構造化プログラムのGOTO文は有害であるという説があります。GOTO文は人間の理解の妨げになっているとEdgar Dijkstra氏は言いました。1968年のことです。何故GOTO文が有害であると言ったのか？それは、そのプログラムが人間の目で正しいかどうか判断しないといけなのに対し、プログラムは機械が読むには簡単であっても、人間がすぐ理解するには難しいからだという事です。

1960年頃の関心事はプロセスと構造で、ハードウェア内部で起こるプログラム実行プロセスを、紙上のプログラム構造と一致させたいというものでした。

Edgar Dijkstra氏の名言に、『プログラムテストはバグの存在を示すのには使えるが、バグが存在しない証拠（アリバイ）にはならない』というのがあります。ではテスターとは何でしょう？岸田氏は、プログラムの何処がおかしいのかを証明しようとする、まるでハードボイルドな探偵のようだと言います。何かを創造する華やかな仕事ではないけれど、けして妥協することなく真実を見つけて示すために、強靱な精神と肉体で立ち向かうイメージということでしょうか。なかなか面白い表現ですね。

#### ◆ソフトウェア工学のはじまり

ソフトウェア工学が誕生したのは同じく1968年頃のNATOワークショップでのことです。その頃には機械内部のプログラム実行プロセスから、機械外部でのソフトウェア開発プロセスへと遷移していました。

そのワークショップでFriedrich Baur氏は、その頃には既にあった電子工学や建築工学等を参考にソフトウェア工学の概念と提唱しましたが、当時はハードウェア製造の工学しかなく、ソフトウェアも1つのプロダクト（もの）であると認識されていたそうです。ハードウェア工学のようにソフトウェア工学が扱われてしまったのは悲劇であると岸田氏は言いました。

#### ◆大規模開発のマネジメント

ハードウェアとソフトウェアの大規模開発をマネジメントでどうするかについて、Winston Royce氏はWaterfall Modelの論文を書きました。ハードウェア製造プロセスとソフトウェア製造プロセスの大きな違いを指摘していたのですが、

論文の意図とが誤解され、Waterfall Modelを推奨したかのように捉えられてしまったそうです。Waterfallは要件定義が全ての始まりですが、完璧な要件定義は難しく、要件は時間とともに変化するため、大規模で複雑な開発であるほどに要件を把握するのは難しいということでした。

その問題に対して、マネジメントサイドからは、プロセスモデルの改良や、プロセスマネジメントの改良、テストマネジメントの工夫等を行い、様々な開発手法や、テスト技法（ツール含む）を工夫してきました。

岸田氏自身の経験としては、テストカバレッジについて、Edward Miller氏に講演して頂き、大規模アプリ開発のシステムテストで応用し、モジュール機能分類とテスト進捗の関係について知見を得る等されて来られました。

またModel Based Testing on Unixという、実機でのテストではなく、Unix上でのテスト計画を実施し、カバレッジやアサーションの機能確認をされたそうです。



#### ◆ソフトウェアの進化論

1978年頃、M. M. Lehman氏をお招きしてセミナーをしていただいた頃、ソフトウェアには2種類のタイプが存在すると言われたそうです。

仕様が確定され変化しないS-Type、現実世界でアプリに組み込まれており時間とともに変化するE-Type。E-Typeは時間とともにアプリも変化するため、仕様が古いものは既に使えなくなることもあるとのことでした。

時間の流れとともに私たちも流れており、何処かで立ち止まって定点観測することはできない。常にソフトウェアも立ち止まることなく、進化します。

ソフトウェアは大きく進化する。また、ソフトウェアの部品は半壊か繰り返ししても摩耗することはなく、メンテナンスすることで劣化する場合があるだけであるとM. M. Lehman氏は自身のHPでも語っています。



常に進化し続けるソフトウェア。その要求仕様は進化とともに変化し、永久に確定できないものだと覚悟する必要があるということです。世の中のほとんどはE-Typeであり、S-Typeと言われるものはごく一部なのかもしれないですね。35年も前の言葉ではありますが、現在のソフトウェア開発のほとんどはE-Typeですから、共感できる場所ではないでしょうか。

#### ◆進化に伴うテストの限界

要求仕様とプログラムを突き合わせるテストには限界があると言います。VモデルやWモデルベースのテストは、ソフトウェア進化プロセスのマルチループの1回目の一部でしかなく、その突き合わせだけでは足りないのです。では何回ループさせるといいのでしょうか？常に進化し続けるソフトウェアに対して、何処までテストできるのでしょう？

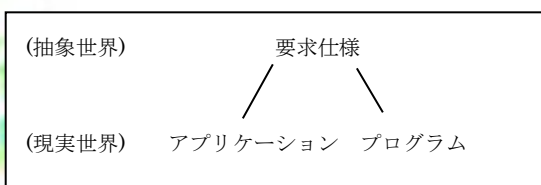
矛盾のない仕様づくりを目指すのは難しい。また、開発したシステムが運用される組織や環境と切り離して仕様のモデル化や分析はできません。ゆえにシステム機能仕様を確定させることよりも、運用する環境でシステムの動作特性を注目すべきではないかと岸田氏は言いました。

#### ◆Immaterial Labor とは？

この言葉はイタリア人社会学者のM. Lazzarato氏が論文で発表したものがインターネット上で公開されたのがはじまり。非物質的労働、あるいは無形労働と呼ばれるものです。プロダクト（もの）を作るのではなく、情報などの形のないプロダクトを作り出す仕事をさします。ソフトウェア開発は無形労働であると言います。

また、ソフトウェア開発は、単に有形なソフトウェアを製造することではなく、開発者とユーザー間の密なコミュニケーションと相互学習プロセスから構成され、それらにより絶えず変化し続けるアプリケーション環境への適応できるということです。こうした思想により、ソフトウェア工学ができた当初のプロダクト指向からプロセス指向へと1970年頃から緩やかにパラダイムシフトが始まっていたそうです。

#### ◆現実と抽象の関係



曖昧な抽象世界の「要求仕様」は現実世界では「アプリケーション」であり「プログラム」です。アプリケーションを抽象化すると仕様要求に、要求

仕様を具体化したものがプログラム。この3つは上記のような逆のV字を描いています。

「仕様は現実世界をうまく反映し、要求を捉えているか。プログラムは現実世界のニーズを十分に満足させられるものなのか。」検証するのがテストの重要な課題のひとつですが、要求に対して満足いく正しいプログラムであるかをテストすることが最重要課題なのではなく、システムを運用する環境での振る舞いを考慮して検証すべきだと言います。要求に対する答え合わせのようなテストではなく、そのシステムを使う環境にとって嬉しいものは何かを意識することが重要だということではないでしょうか。

#### ◆要求仕様の開発

この抽象化されている仕様の開発とは何でしょうか。

ソフトウェアは対象アプリケーションを記述したひとつの仮想世界であり、エンジニアは仮想世界の創造者であるという言葉があります。この世界を創造する仕様開発とは、ゼロから仕様を作るのではなく、ユーザーにヒアリングして既にある仕様から次のVersionを開発することです。この世界創造は、既存世界を分解して構成し、重みや優先度を付け、削除・補完・変形のプロセスを辿って形成されていきます。

#### ◆仮説マネジメント

世界とはものの集合を指すのではなく、成立している事柄の総称だとLudwing Wittgenstein氏と言いますが、私たちが創造する世界は現実に成立しているものではなく、これから成立させるものです。そこで仮説を立てていく必要がでてきます。

現実世界をパラメータ化した場合、その数は無限大ですが、仕様書を作成しようとした場合には、ページ数は有限です。そのため、仕様を起こす際には、不要と思われるいくつかの要素を切り捨てることとなります。

しかしそういった仮説は西暦2000年問題のようにいつか破綻することがあります。

大規模かつ複雑な社会システムが増加するとともに、破綻しないよう、プロセスモデル化を改善する等の仮説の管理と検証をするのがこれからのテスターの役割になってくるのだと岸田氏は言いました。

#### ◆プロセスモデルの改善

プロセスモデルの基本的概念は同じようなステップを繰り返し、反復することです。その反復の中で、経験から次はもっと良くしようと改善することがプロセスモデルの改善です。



プロセスモデルを改善する際、ベストプラクティスから学ぶことが、成功への近道のように見えますが、ベストプラクティスは常に同じではなく、環境が違えば適用は難しいのです。ベストプラクティスを再現するより、ワーストプラクティスから学ぶ方がわかりやすい。失敗した経験から学ぶことで、ワーストプラクティス環境を避けることはできます。

経験から学ぶとはすなわち、失敗の繰り返しの中から、次への改善を繋げていくことにあります。何度も繰り返す反復は常に同じではなく、ワーストプラクティスから学ぶことはとても重要です。何事も自分が経験した苦い思い出は忘れることはありませんね。意外と覚えていて、次は同じ失敗をしたくないという意識は働きますね。そこから改善への一歩を踏み出します。

苦い思い出とその改善活動は、自分の頭の中だけにもっているのは勿体ないです。自らの経験を発信することで他者と共通認識が持てたり、他者から聞いたりすることで、自分の経験だけでなく、他者の経験も疑似体験して意識することができ、失敗談は面白く、共感することも多いですが、そこから学ぶことも多いですね。発信と受信により、多くの経験から学ぶことができます。私たちが経験したことは、誰かの改善に役立つ可能性を秘めています。会社でも社外でも、ネットでもいいですから、発信と受信をうまく使って、経験を学び次に繋げていきたいと思いました。

## ■クロージングパネル

### 「エンジニアリングの本質

～人月、自動化、無形労働～」

モデレータ：

板倉 稔氏(イネーブル・ツリー)

パネリスト：

Dorothy Graham 氏 (Software Testing Consultant)

岸田 孝一氏 (SRA)

榊原 彰氏 (日本 IBM)

西 康晴氏 (電気通信大学)

クロージングパネルでは世界的に活躍しているエキスパート達に、技術者としての矜持など「**エンジニアリングの本質**」について語り合って頂き、(テスト)エンジニアとしての生き方を考えるヒントを貰いました。

## ◆日々の工夫について

目の前に簡単に大きな問題があったとしても、昨日よりも今日、少しずつでも日々改善・スキルアップを継続していくことが重要である。新たなスキルは一朝一夕では身につかないものだが、練習を続けていくことで突然出来るようになる。

またスキルアップの際は、物事の形をなぞるだけではなく、本質を理解するよう努めることが重要である。改善やスキルアップが当初の目的を達成しているか俯瞰してみることも重要である。

## ◆不具合に関して

不具合を検出する際、不具合の混入過程すなわち人の「**間違え方**」を予測する事で検出効率が上がる。間違え方は各々のドメインも影響する。組織の構成や国による文化の違いなども不具合の混入に影響する。

また実際に不具合を検出した場合、混入した原因を分析し、類似する不具合の検出および混入防止と言った活動の水平展開も重要になってくる。

## ◆テストに期待されるものとは？

間違った期待だが、開発スケジュールに影響するので不具合を検出して欲しくない人もいる。不具合の検出から修正を含めて、テストをスケジュール通り間に合わせる事への期待は高いだろう。

プロジェクトの**各工程によってもテストに期待されるものが変わってくる**。相手がどのタイミングで何が欲しいか把握することが重要である。

## ◆開発現場の将来

ネットの出現で社会は大きく変わった。今後主流となっていく技術があるだろうが、大事なのはどれか一つの技術に絞るのではなく、対話と議論を続けていくことである。

開発現場の将来を考えるには、**社会工学**が重要になってくるのではないかと。

## ◆エンジニアリングとプラクティスの違い

工程を細分化して詳細化すれば、誰でも同じ事が出来るようになるか？ 出来ていない。人材のスキルレベルや適性を考慮することは重要である。

## ◆感想

話の中で出てきた「日々の改善とスキルアップ」というキーワードを聞いて、自分は実践できているか身が引き締まる思いでした。今後の社会で何が必要で何を学んでいくのか、改めて見つめ直していきたいです。

いかがでしたでしょうか。

JaSST<sup>13</sup> Tokyo。これからも日本最大級のシンポジウムであり続けるであろう JaSST Tokyo には WACATE の仲間や元 WACATE 実行委員など、さまざまなエンジニアが活躍しています。

仲間に会いに、是非皆さんもお越しください！



# WACATE2012冬 BPP受賞者寄稿 「ソフトウェアの品質ってなんだろう」

こんにちは。しんすく (@snsk) と申します。前回の WACATE、2012 冬で BPP 賞と共に WACATE MAGAZINE への寄稿権というありがたい機会を頂きました。WACATE に参加される方々のうち、テストを主たる技術として活躍されている方のほとんどは「品質保証部」やそれに類する組織に属しておられることと思います。「品質保証部」です。「品質」を「保証」する「部門」です。「品質」を… クドいですね。しかし、みなさんご自身や、所属している企業の部門長、その部門を設置している経営陣の方々は「品質」、特にソフトウェアにおけるそれを適切に説明することができるでしょうか。「バグの多寡」では 30 点です。

そこで今回は、僕自身もソフトウェアテストという職業にはじめて触れた時に抱いたギモンである「ソフトウェアの品質って何？」について現時点での解を示してみたいと思います。

このような大きな概念を知るときは歴史にあたるのが王道です。きっと僕が疑問に思うよりはるか昔から、比べ物にならないぐらいの膨大な時間を掛けて、賢人たちが議論して辿り着いた解が既に存在すると考えるべきです。ただし、広くそれを伝えるために抽象化されているので、明日から現場で使えるほどの具体性を期待するのは筋違いでしょう。ラストワンマイル、現場への適用は常に自らの手で試行錯誤する必要があります。

閑話休題、それでは賢人たちにおける「ソフトウェア品質の定義」について歴史を振り返ってみましょう。ほぼ時系列に並べましたが、いくつか年代不明であったため掲載していないものがあります。

- ソフトウェアを完全に停止させたり、容認できないような結果を出す欠陥が全くないこと (Capers Jones: 恐らく 70 年代)
- 要件に対する適合 (Philip B Crosby: 1980)
- 「狭義の品質」と「広義の品質」(石川馨: 1981)
  - 狭義の品質: 製品の品質
  - 広義の品質: 仕事、サービス、情報、工程、部門、人、システム、会社の全てを含めた質
- 魅力的品質と当たり前品質/狩野モデル (狩野紀昭: 1984)
- 製品またはサービスが、明示してあるか、あるいは暗黙の要望を満たす能力として持っている特性の総称 (ISO 8402: 1986)
- システムが本稼働するとき、どこまで真のビジネス(ユーザ)ニーズにあっているかということ (James Martin: 1994)
- 品質は誰かにとっての価値である (G. M. Weinberg: 1994)

- プロダクトの特性が顧客のニーズに応えることで満足を提供する & 不備(障害や誤り)から免れる (Joseph M. Juran: 1998)
- 指定された特定の条件で利用する場合の、明示的または暗黙的なニーズを満たすソフトウェア製品の能力 (ISO 25000: 2005)
- 機能及び性能に関する明示的な要求事項、明確に文書化された開発標準、および職業的に開発が行われた全てのソフトウェアに期待される暗黙の特性に対する適合 (Roger S. Pressman: 2005)

実際、1970 年代では「バグが無いこと」と定義されていました。海外での変遷は大雑把には「バグが無いこと→要求を満たしていること→暗黙の要求もあるよ→ビジネスニーズもあるよ→もう、何だかわからない(ワインバーグ)→悟りの世界(プレスマン)」という流れです。ワインバーグの言はわかったような、わからないような印象を受けますが、実は**品質の相対性という重要なポイントを初めて示しています**。筆者の俗な想像ですが、ワインバーグはキレてぶっきらぼうに言い放ち、プレスマンはムキになって説明した、のではないのでしょうか。特にプレスマンの言は「**それでわかるなら定義なんか調べないよ!**」といたくなるようなヒステリックさです。

これだけ品質の定義について幅自体がブレてきたのは、ソフトウェアシステムの社会における存在感、価値の占めるその割合の増加と密接な関係があるのではないのでしょうか。掲載した限り最も古い言及である Capers Jones の時代には B2C などというドメインすら無かったはずで、社会におけるソフトウェアの役割も小さく「最悪動かなくても手でやるよ!」という牧歌的な頃であったかと思います。時代を経るにつれ、ソフトウェアの利用は効果的、もはや無くては立ち行かないもの、水道や電気と同じぐらい、あるいはそれ以上に重要かつクリティカルなインフラ、とその重要さを増すたびに、人間がシステムに持つ期待がどんどん膨らんでいき、今現在においてもその膨張はとどまることを知らないように見えます。

## Column:

80 年代における日本とアメリカの品質認識の違いはとても興味深いものがあります。「品質」の定義を異様に広くとっている日本に対し、アメリカのそれは「先立つ何かがあり、十全にそれに従えていること」とされています。これは、アメリカのソフトウェア産業が主に軍からの厳密な発注と契約に基づいて発展してきたのに対し、日本はお客様の望む以上のものを提供する、という工業製品における思想に範を置いてきたためと考えられます。「プロ



セス保証」と「テスト」という軸を並列に置いた時日本では前者が、海外では後者に重きが置かれているように感じるのもこの歴史的経緯とは無関係ではないはずです。

日本の定義はさすがお客様第一主義らしく、最初から幅が広く、著名な狩野モデルでは製品自体の品質とユーザの満足度を相対的に表すモデルが84年の時点で既に提唱されています。魅力的品質なんて今でも量化が難しいのに30年以上前からその端緒をつけるあたり、品質大国ニッポンは伊達ではないです。日本の定義はシステムに限った話ではないのですが、伝統的に工業製品の製造過程を見習ってきたため、ソフトウェア産業においても同様の定義が使われてきたと考えて良さそうです。海外からすると「ニホンすげえよ、奥が深いよ！（でも具体的に何したらいいのかわからないよ）」で、日本からすると「海外の方法論はスゴいけど、お客様が語らないんだよねえ」という印象をお互いに持っていたのではないのでしょうか。

一方、品質を一意に定義する、ことはどうやら難しいと悟った別の賢人達はソフトウェアに限ってその抽象度を一段下げ「ソフトウェア品質モデル」を提唱しはじめます。こちらも時系列に沿っていくつかの例を挙げます。

- McCall's Quality Model (McCall: 1977)
- Boehm's Quality Model (Boehm: 1978)
- FURPS (Hewlett-Packard: 70年代後半) \*のちに+
- ISO/IEC 9126 (1991)
- ISO/IEC 25000 シリーズ SQuaRE (2005)

このどれもが今でも使えるものばかりなのは驚愕です。余談ですが、最初に触るものとしてはFURPS+が特性の数が少ないのでオススメです。特徴的なのはFURPS(+)を除きそのどれもがツリー構造を採用している点です。BoehmとMcCallのそれは親特性と子特性が交差しあうN:Nの関係を持っていたのに対し、ISO9126以降では1:Nの構造が遵守されているのは、恐らく普及のためにある程度の単純化が図られたからではないかと予想されます。

特に有名なISO/IEC 9126は、前述のお互いへのリスペクトが、海外の具体的な計測法と日本の品質の幅を広くとる思想の両方をソフトウェアという分野に限って標準化できた幸福な例と言えます。

さきほど、人間が持つシステムへの期待は今も膨張していると述べましたが、それはISO/IEC9126から25010への改訂時に主特性が6個から8個に。利用時の品質も主特性が1つ増加し、副特性が正式に定められたことから伺えます。9126だけでも大変だったのに、更に増えてこの先どうしたらいいのだろうとこの原稿を書いている今も頭を抱えています。人間がある程度諦めてくれれば特性も減るのでしょうか、それを言ったら負けな気もするので頑張りたいと思います。

気を取り直して、ここまで「品質」自体の定義の例とソフトウェアに限って「品質」を一段具体化した「品質モデル」の例を見てきました。冒頭の質問に立ち返ってみましょう。「ソフトウェア品質って何ですか」。これらの考察を踏まえて解答するなら、現時点では

「定義自体は以上のような変遷を経て未だに定まり切れていないが、ソフトウェアに限定してモデル化されたものがいくつかある。その中でも網羅性を求めるならISO/IEC9126(→25010)を、使用性を求めるならFURPS+が良さそう。9126は品質モデルの定義と内部/外部品質と利用時品質に分かれており、モデルは6つの主特性と27の副特性から構成され…」

といった形になります。このあと対象の製品にまでスコープが及べば数時間は話せますが、相手が寝てしまうと思うので、少なくともソフトウェア企業の品質保証部という「品質の実戦部隊」としての短い答えは、

「既存の品質モデルを製品にテーラリングして良し悪しを語れるようにした概念」

となります。そこから、その良し悪しを良い方向に導いたり計測したりするための、今現在最も効果的な、パズルのような、“知る”技術として、ソフトウェアテストがあるのだと理解しています。少なくとも僕の面接ではこういう解答を得られれば満点です。もちろんみなさんがいまおかれているコンテキストにおいて異論は多々あるでしょうし、僕自身、もっと良い解答を得られる事もこの先あると思いますが、先ず議論自体が行えるように、自分なりの解を用意しておくことも大事ななと思います。

僕はこの手の「とりあえずの解答」をその場で打ち上げたり、予め考えてストックしておくのがとっても得意なので、ぜひぜひみなさん議論を吹っかけて頂いて「浅いな」と感じたら、より良い解答とその導き方について、ご指導をお願い致します。これからも一緒に技術を楽しんでいきましょう！

(しんすく)



# せんぱいにきく。も

細川 宣啓せんぱい

3 special season



WACATE の皆さんこんにちわ。細川と申します。私は現在43才です。不思議なもので30代は全ての仕事に「俺が！オレが！」って自己主張していたのですが40を過ぎたあたりから別の欲求が生まれて来たのです。それは

「足跡を残したい」

という欲求です。これは二つに分解できます。

- 1) 世界を変えること
- 2) 歴史に名を残すこと

例えば1)は、世間の技術者が「あっ！」と驚くような社会影響のある仕事をするプロであり続けたいですね。2)は自分が書いた論文、書いたコード、作った設計が私が死んだ後に誰かの目にとまり、「これ考えたやつすげーな。この時代にかああ」と唸ってくれたら技術者として本望です。

では、1)や2)を実現するためにはどうしたらいいでしょうか？私が心がけているは以下の2つです。

- 3) オリジナルであり続けること。  
(誰にも似ていない自分であり続けること)
- 4) 誇りを持つこと

です。前者は「オリジナルであろう」と常に意識することで、1)と関係があります。常に自分を肯定しないこと＝問題意識を持ち続けることで実現できそうです。後者は自らプロとしてのプライドを維持し続けること。これは2)と密接に関連があると思っています。

4)の誇りは、きっと他人と同じことをしていたら前述2)の「歴史に名を残す」なんてできないんじゃないかと思っています。まとめると以下2つです。

- 5) 問題意識を常に持つこと
- 6) 人と違うことをすること

ここに記載した6個のポリシーを用いて、この夏のWACATEクロージングセッションを構成しようと思います。お話のトピック(予定)は次のとおりです。

1) なぜそんなにドヤ顔でプレゼンできるのか  
(DOYA 曲線と COBI 曲線の交点に関する一考察)

2) 新しい技術が生まれる時+場所+瞬間。  
(天使が二度降ってくる話)

3) 欠陥とは何か？：欠陥研究の発想、洞察力とその思考

4) 未来を担う WACATE 技術者にむけて  
(Does/Don'ts)

WACATE で登壇させて頂くのは約4年ぶりです。皆さんとお会いできるのを楽しみにしております。どうぞ宜しくお願いいたします！

追伸) この「センパイニキク」コーナーの真意が確かめられないまま WACATE Magazine が終わってしまうのが非常に残念です。正解は次のうちいずれだったのでしょうか？

1. センパイに聞く  
(まあいろいろ聞いてみようの意味で)
2. センパイに菊  
(死んだあの人へ献花の意味で)
3. センパイに効く  
(あの先輩ならこの話題だと効果的に落ちますよ！？の意味で)
4. センパイに訊く  
(あの先輩はどうも信用ならんつるし上げろ！の意味で)
5. センパイに起句  
(あの先輩腰が重いから最初の文章書いてあげ (ry という代筆の意味で)



ぽじゃま★パーティー

# てすバナ語るっcha

## ■「そういえば、投稿戦士以外に

WACATE ファンの発信方法ってないよね・・・」

「ないなら創ってしまえばいいじゃない！」

投稿戦士以外のカタチで読者など WACATE ファンが関われるイベントとかできないかなあ？・・・という案から、チャットで座談会(分科会出張所的なもの)を開催したらいいんじゃないかなあ・・・ということで、だいたい月1回のペースで開催されてきました。

開始時には3人しかいない><ということもよくありましたが^^;最終的には深夜にどこか参加してきて朝まで盛り上がるということもありました。たいてい日付変更線を越えたあたりがいつも盛り上がってましたねー♪

今回掲載するのは、実は昨年10月に開催された回の内容です。それがなんでここまで放置されていたかといいますと・・・オーナーの体力不足ゆえ・・・orz お待たせいたしました。

今回のテーマは、「テストの開始基準」。「開始基準が守られない場合でも実施しなければならない場合の対処方法ってどうしてる？」という問いかけから始まり、「そもそも開始基準は必要なのだろうか？」という話で盛り上がりました。テストを潤滑に行うにはテストの開始基準を満たしていることが重要だと思いますが、実際の現場では理想となっているケースが多く見受けられます・・・そういう話はどうしても現場に近い生の声になってしまうため、ハンドルネームでも機密事項が無くても発言者と会話が結びつくのは危険だね、という声もあり、今回はチャットをそのまま載せるよりも、会話をピックアップしながら「こんな話がでたよ」という載せ方にさせていただきます。

## ■第12回♪

本題突入時刻[12/10/13 22:46:42]

まずは**開始基準が守られない状況でテストを実施しなければならないことはあるか?**については、「あるある」「...そっちのほうが多いんじゃないか」「最近毎回」など・・・開始基準があっても守られていない現場は多いようです。

### 開始基準にどんなものを置いている?

「あるあるなのは「開発チームができたって言ったら」

### なんで守れないの?に対する答えを把握している?

「把握はしているが対応できてませんw開始基準は知っているはずだけど知らんぷりが多いような@@」

「(以下の4つを把握)

1. 単に見積り失敗
2. 開始基準が周知徹底されていない
3. 決まっていない

4. 基準を満たしているか確認したところ、満たしていなかった」

「最近の案件ですが、4.だった・・・orz」

「この業界で二つ目に経験した現場は3でしたね^^;; あーあーあー、みたいな」

### どんな開始基準があるんだろう?

「ここ1~2年で設計側がレビューでの1KLあたりの欠陥検出数の基準を決めたようです。←過去PJで痛い目にあったので・・・」

「レビューでの欠陥検出数基準はイイネ」

「品質会計的なアプローチだと結合テストまでに想定バグ数のN%が検出されていること、になりますね。」

「うち10個くらいあったなー、確か。開始基準の基本的なセット。」

「10個のメトリクス??」

「んにゃwしょーもない奴が10個。機能が決定しているか? 単体テストは終了しているか? 的なウォーターフォール臭がするやつ」

「どでかいシステムが多いので、品質会計的な、各工程のレビュー密度/テスト密度/バグ検出密度+基準満たさない場合はその言い訳って感じですねー」

### 開始基準が「組織で決まっているから」というだけで、プロジェクトに則していない場合もあるのではないかな?

「あるあるあるある! 全社基準値がこれなので(きりっ

・・・あほか」

「即していないので、割りとは無視してたり、「条件付きで合意」とかいう風にしたりw便利です条件付きw」

「うちはプロジェクト毎にドメインが異なるのもうぐっちゃぐちゃ。なので、いちばん厳しいやつを例にしたほうがよさそうですね。理想的?なのが、CIシステムのグリーンのハードルを、テストエンジニアがどんどん上げていってグリーンじゃないと開始しませんーってやつかな。CIシステムってユニットテストばかり注目されるけど、システムテストを回すこともできるので。」

「チェックで妥当に確認できるやつ、というのはすなわち、最近みなさんがネタにしてる仕様書設計書のコピーあんどペーすとあんどもでいふあいほうで出せるテストは全部やっつけよ。という解釈でok?」

「んー。コピー法はあんまりイケてないかも。「妥当に」という言葉を取っておいたのは、粒度の細かいテストをCIに置くとレッドの信頼性が落ちることです。「もう、これだけは絶対に機能してないとダメ!」ってやつをチェックに落としこんで、要はスモークテストなわけだけど。せっかくテストエンジニアなので、コードではなくお客様の価値にそったテストを厳選してチェックまで落とし込んで、自動化して、CIに放り込む、と。それが毎日ぐるぐるまわって。グリーンならテストしてもよいですよ・・・あたりが理想的な開始基準かなと僕はおもいました。」



「なるほど。基本機能とでもいうべきテストをチェックにする、て感じか。」

### 開始基準って、果たして必要なのか？

「テストに開始基準が必要というのがイマイチわかってないです。ごめんなさい。ないと困るの？Doneの定義がないことが問題なのではなくて？」

「ああ。そうか。開始基準というのは、作る人とテストする人、が分かれている場合、テストする人が、「テストをはじめめるレベルに達している」ことを見定めるために必要なクライテリア、です。うん。作る人側のDONEの定義と言い換えてもいいです。」

「なぜ開始基準が必要なのか？か。。。」

「そうそう。なんというか、自分でいろんなテストを触る自分としてはそこらへんがよくわからんです。。。」

「端的に言うと、テストチームが最大のパフォーマンスを発揮できないからです。」

「意味が違うかもしれないけど、テストチームのパフォーマンス減っても開発者のパフォーマンスが出ればいいんじゃないの？」

「あまりにバカバカしいバグでも見つけた以上は報告しなければならず。逆にいうと、こんなバカバカしいバグを報告する手間をかけさせるな、と。」

「バグが出るっていう前提ならそうですね。」

「開始基準と終了基準で同じじゃね？が同じにならないんだよね。」

「はい。仮に開発者のパフォーマンスが200%だとしても、バグは出ます。それぐらいでテストチームにリリースしてほしいですね。出せるもんなら出して！ぐらいのほうがチームは燃えます。。。萌えます。バグをどこまで広い意味で捉えられるかの差もあり。」

「なぜ開始基準が必要かっていうのは、とある作業を開始するのにとっかかりがあるからじゃないかな。」

「逆に開始基準が不要な理由ってなんだろ？」

「必要だと思うからする。とかじゃないですかね。」

「必要だと思うからする、というのは、なぜ必要と思うか＝開始基準と同義ではないの？」

「や、テストチームが本来(期待される)仕事の邪魔になるから、だよ。だっさいバグが。なので、「せめてここまではテストしてからテストチームにリリースしてくれませんか？」を漢字4文字でいうと、開始基準、ですよ。」

「リクエストを受けるのと、リクエストが何であるかは別じゃない？」

「結果的にやるのであれば、やるとっかかりは同じでは？うーん。別にチームがわかれている前提は不要だと思うんだけどな。」

「あ。僕はチームがわかれていなければ、開始基準はなくていいと思う。なぜかという、いつもチームの中でテストしているので、基準に達していなければ、隣の席に「これまだ早くね？」って言えば済む。基準はルールなので、そういうのを明文化しておくべきなのはチームが違う、別のルールで仕事している場合のI/Fとして必要なとき、だと思うから。どっちがいい、って話はしてませんよ。」

「うーん。チームの中で他人に頼むときも不要？ルールきつすぎ？ってイメージなのかしらん」

「うん。俺のイメージでは、チームの中なら、話しかければいいかなーっておもた。たとえば、パフォーマンステストってけっこう遅めにやるっしょ。」

「うん」

「かろうじて結合できた瞬間にパフォーマンステスト走らす人がいれば、いあいあ、ちょっと待ってー。ごめんー。ってなる。けど、別のチームにリリースするなら、ある程度チューニングしてから、出す。その程度の度合いの明文化が開始基準の一例かなと。」

「開始基準って静的型付けにして、コンパイルエラーにするか動的型付けにして、実行時エラーにするかくらいの違いでしょう？動的型付けでも暗黙的型変換とかダックタイピングとかできるし、そこらへんはメンバーがどんなことやりたいか次第な気がする。」

「うん。その暗黙的型変換の「されてしまってよい度合い」が、作ってない人によって異なるのですよ。バインドにも強い、弱いがあるでしょ。実行時エラーはゼロだけど、起動するのに1年かかるソフト使う気になる？・・・例がわるいなーm(\_\_\_\_)m」

「まあ例はあれとして、一般的にも、そこらへんは型とかテストで縛る人が多いんじゃないですかね。xsdとか典型的だとおもう。あれは型じゃないけど。」

ざっくり言うと、xmlをWebAPIのデータとして使うときにxsdっていうのを使うと「こういう名前のタグがないとダメ」とかをチェックできるんですよ。だから、WebのAPIをXMLで提供するサービスはxsdとかを提供する事が多い訳です。」

「たぶんDSLの話をしてるんだと思うけど、実装レベルの合理性と利用時の合理性が全く異なるのはたぶんわかると思う。」

「えーと、僕の理解が正しければ、チームの中ではその度合いは共有されていて問題ないが、チームの外に出るとルールが違うから、基準の明文化が必要と主張している、でよい？」

「そう」

「多くの人が関わる程「実行可能」か「形式的」な「基準」がないと困るよねって言う。そんなかんじです。」

### どうしても満たせないときに、どんな工夫ができるんだろう？

「グレーボックスなテストをするというのが最も効果的だと思うんだけどどうですかね？」

「開始基準に満たされずにテストするハメになったときにはこういうテストをする！みたいなものがあるとよいのかな？という点で、グレーボックスなテストをする？」

「そうそう。」

「グレーボックステストってやったことがないのですが、開発者と結合移行のテスト担当が別れている場合は難しいのでしょうか？」

「そのドメインに詳しい人が担当だとできるよ。」

「開発者が済ませてくるテスト、のレビューの一環としてテストチームがグレーボックステスト的に参加するのはアリだと思う〜。」

「作りに着目しながらも価値に着目して重要ところを検討することで、開始基準を満たしていなくとも、意味のあ



るテストができる、というイメージかな。>グレーボックス」

「コードを読んじゃいけないっていう理由があるなら別だけど、テストエンジニアがコード読まない方が僕はどうかしていると思っているくらいです。」

「そう考えると、コードっていうのもひとつのドメインだねー。誰でも読めるけど。」

「コードについては、開発者以外は見れないような環境下にいたことがほとんどでした。会社の違う第三者的な立場だったからかもしれませんが。」

「そういう環境だと、難しいんですよね。開発の方がやる範囲にまで入り込むのが。」

「個人的に思っているのは、上記にあるような結合テストであまりにもバグが多いのは「一度も結合したことがない」か「コードから腐臭がするか」のどっちかです。プログラミング系ではよくいいますよ。「臭うコード」ってw」

「重複具合は簡単に測れる腐臭のひとつですね。」

「たぶんくだらないバグを出す人は静的解析ツールレベルのことを取舍選択してコード書けていないですね。」

「開始基準に満たされていない時に仕方がないからテストする、よりも、テストしない代わりに何かができるという考えたりもする？」

「テストの教育のためや、テストプロセスが回ることを確認するために始めることはあるよ。」

「ああ、そういう異なる目的にすり替えるという手もあるかもしれません。」

「テストの代わりに出来る事って「会話」「レビュー」「ドキュメント」以外にあるんでしたっけ？いまあげたのって、駆動するためであって、基準にはしにくいんですよね。。。」

「開始基準を満たしていない、ということは「やるべきこと」ができてこない状態だと思うのですが、その「できていないこと」はやる／やらない、どちらの方向なのでしょう??」

「やらないことが多いと思う。」

「できていないことを「誰もやらない」という選択肢はあり？誰もやらなくてよいものなら、「できていないこと」になっていることに違和感があり、それは、開始基準がフィットしていないことに含まれると思うのです。そうではなくて、本当にやらなければいけないことができていない場合、でもテストをはじめなきゃいけない場合、どうする？」

「「本当にやらなければいけないこと」って何要件ですか？ビジネス要件？「テストを楽にする要件」だったら「やらない」もありだと思うし。」

「開始基準として挙げられていることです。。で済ませちゃダメかしら？」

「やる、やらないは最終的に「ビジネス要件を満たせるか」に関わるので、ビジネス要件を満たせれば何をやってもやらなくてもいいと思います。「結合テスト開始基準」に「ビジネス要件」が絡んでくるパターンがあるのか僕はわかりませんが。納品物とかチームの形態によってかわりそう。」

「受託の場合はそだね。」

「仮に「ビジネス要件にかかわらないもの」であるなら「や

らない」という判断もありなんじゃないですかね。さきのコンテキストで重要なのは「ビジネスリスクが高い」「修正リスクが高い」などなどのリスク観点でフィードバックを開発者とPOに出すかだと思います。」

「PO=プロダクトオーナー=顧客？」

「はい。」

「直接ビジネス要件に絡むかどうかはあれだけど、あるビジネス要件を満たすための手段の一部として絡むことはあるかも？」

「スプリントの計画の話っぼい…」

「そうそう。そうなんですよ。」

あとと思ったんですけど、「開始基準が満たされているかどうか」って何を想像しているんでしょうか？自動テストじゃダメなの？ハッピーパステストつくるのに1dayもかかるようなパターンが思いつかないというか。環境構築とかは別として。

今話していたのは「自動テスト以外の開始基準検査を自分でやるかやらないかは、開始基準自体の影響範囲によるよね」というのが言いたかった。」

「ハッピーパステスト」と「ユースケーステスト」て何か違う？」

「ハッピーパスは「フィーチャを通す王道パターン」。ICONIXのユースケースの一番最初にでてくるやつかな。」

「実践アジャイルテストで書いてあった気がする」

「スモークテストとおなじ？似たような意味なら、自動テストにしちゃえばいいじゃん、は同意。」

「スモークテストは「Hello World ページを表示できる」とかくらいなのでちょっと意味合いが違うかも。疎通確認くらいのイメージで使う単語かな。」

「なるほど、僕は最上位の価値を確認するテストの意味でスモークテストを使ってました。見解の相違ですね。」

「あい。」

「テストチームが定義するスモークテストは価値の確認になると思うのです。分かれてる場合ね。で、それは大抵一意になるので、チェックにしやすい。チェックにできれば自動化できる→CIに取り込める→うまー。」

「そこらへんは継続的デリバリーでも「コミットステージ（開発者がやる自動テスト）には自動受け入れテストをいれましょう。あと、テストエンジニアがやる結合テストの一部を自動化しておきましょう」とあるよ。それが具体的にはハッピーパスになる感じ。」

「なるほど。ハッピーパスのイメージがつかめてきた。」

「チームがうまく稼働する条件みたいな感じだね。ハッピーパス。」

「ストーリーを INVEST にしておくとか簡単にさせるのがハッピーパスですよ！」

「何でグレーボックスがいいんだっけ？」

「くだらないバグが多いときに、バグの偏りをみつけやすいからですかね。」

「バグバグな中で通りやすいテストを選択しないといけなから？テストすこしでも進めるためにとか・・・」

「通りやすいテストをするため」ではなくって「無駄なテストをしないため」ってイメージですかね。A モジュールはバグだらけだから数件のテストで「つくりなせやああ！」で済ましておいて、B モジュールは綺麗なコードだから結



構テストやれそうだな。っていう当たりをつける的な。」  
「むしろ、すこしでもできてるところを探すのにグレーボックスが必要なのかと思ったよ。」  
「僕は人情的な意味で異常系のテストを作る人がやり切る、のは難しいのではないかと考えてます。」  
「テスト中にやらないといけないことができはじめたときは、グダグダなモジュールを懇切丁寧にテストしておいて、再テストを何度もやらないように歯止めをかけるとか。」  
「そういう状況下ではくだらないバグが労せずとも当たるだろうに。」

時間が時間なのでなんとかしたいなw

「単体テストやってないけど、結合テスト開始だぜひゃっはー！みたいな？」  
「ああーw」  
「そ、そゆことです！」  
「グレーボックスはそれをイメージしてたけど、要は前工程でやるべきことを間引いてやるのが多いと思うというか、自分はそれがいいと思っています。システムテストでそうきたなら、結合テストでやるべきことをいくつかピックアップするとか。」  
「ああ、システムテストの実施計画に合わせるってことね？」  
「きっとそうなるのがよいのでしょうね。で、それを仕方がないからシステムテストの一環として行うのではなく、あくまでわけて計画を立て直すのがよいのかと。で、現状はそういう手間を省いて仕方ねーな、テストしちゃえー、になっているんじゃないかなあと。。。」

最終的に一旦撃ち切ったのが

[12/10/14 1:16:43]

なんとなく仕方ないなって自分たちのテストをはじめると、先に基準が満たされるために必要な箇所を補うなり、あくまで基準が満たされるのを待つ間別の目的でテストをしたり・・・ともかくどう作戦変更したかを明確にすることが大事なんじゃないかなあという印象をもちました。

その後もいろいろワイガヤな状態で、最終が

[12/10/14 2:35:22]

まあ朝までじゃないからいいのかw  
(オーナーは力尽きて寝てましたw)  
みなさまありがとうございましたー♪  
そして、今まで参加して下さったみなさま、ありがとうございました。またどこかで^^語りましょう♪

今回のチャットルームオーナー：ぼんぼん



# 今月のSTAR★

◆おつかれスター☆  
とってもお疲れのスター☆。



スター・ゴーグルでリラックスしています。

元気になったところで  
ユニケルスパークリングを飲んでシャッキリ。



そのテスト、再考です。  
そのカメラ目線、スター☆最高です！

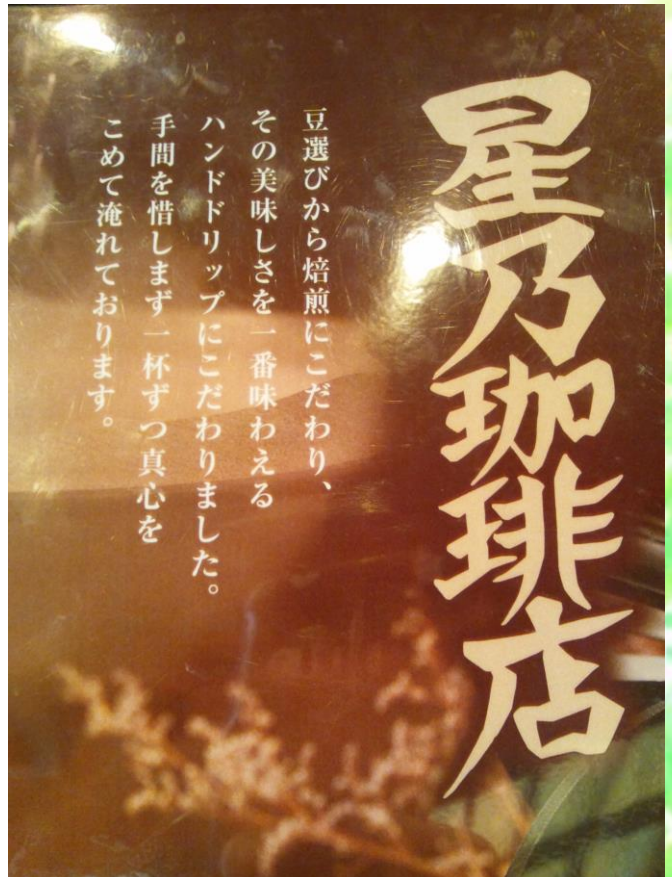
◆STAR☆は永遠に不滅です。



アイスのパッケージになったり



ときに薬局になったり



ときに喫茶店になったり・・・と、私が行くところ、至るところにスター☆がいました！

Magazine は終わりますが、これからもいろんなところで僕を見守っててください！（違  
書いた人：コヤマン



# 開運

## 源太郎のソフトウェアテスト占い

※「ソフトウェアテスト占い」とは…  
日本の伝統的な暦で知られる気学(九気性)をもとに、日本古来の統計学とソフトウェアテストで培った時代の統計学を組み合わせ考察された占いである！(笑)

本命星	生まれ年	2013年の運勢
一白水星	S38/S47/S56/H2	「水」の特徴をもち、柔軟性が高く、あらゆるものを育む力がある。 吉凶半々の年になるであろう。対人関係を見直し、人助けが吉。 「冷え」「酒」「男女関係」には注意すべし！ スキルアップには、良い年なり。 また部下・周囲へのサポートを心掛けよ！
二黒土星	S37/S46/S55/H1	「大地」の特徴をもち、温和で堅実(足元しっかり)、支える力がある。 今年は、忍耐・準備の年になるであろう。勇気と努力を忘れず、将来を考えることが吉。人脈を広げることも肝要なり。 ただし、物事を頼まれやすいので、八方美人もほどほどと心得よ。
三碧木星	S36/S45/S54/S63	「雷」「春」という特徴をもち、切り替えが早く陽気で行動的な面がある。 今年は、将来的な目標に臨み、気持ちを一定することが吉と心得よ。 効率&スピードアップが重要で、弱点克服の努力も寛容なり。 今年は苦勞が報われるであろう。
四緑木星	S35/S44/S53/S62	「風」の特徴をもつために、柔和で社交的だが優柔不断な面ある。 日頃の人間・信頼関係を活かし、すべてに根気強く対応することが吉なり。 周囲の知恵と力を借り、人脈がカギと心得よ。 今年は、去年まいた種が芽吹く年なり。
五黄土星	S34/S43/S52/S61	「帝王」「地震」の特徴を持ち、善悪・吉凶いずれも両極端にでる、自身が動くと他に影響を与えるので、注意すべし。動くときには、慎重に行動すべし！ 来年に備えて不要な枝をカットすることが吉なり。 力不足を思い知るであろうが、見栄と意地を捨てることが肝要と心得よ。
六白金星	S33/S42/S51/S60	「天」「主」の特徴を持ち、「弱きを助け強きを挫く」面をもつ。 今年は、まいた種を収穫する年。「奉仕」が運氣向上のカギなり、トラブルが起きやすいので、対人関係には注意すべし。 やさしさを忘れず、相手の立場で考えることが肝要なり。
七赤金星	S32/S41/S50/S59	「実」「愛嬌」の特徴を持ち、表面は陽気で社交的だが内面は陰気な面もあり。 チームワークを大事にして、自分に厳しく、粘りと根気が吉なり。 人脈を拡大して責任感が肝要と心得よ。 「実」りに「恵」まれているので、「自信」を持ってポジティブに行動すべし！
八白土星	S31/S40/S49/S58	「山」の特徴を持ち、温厚で意思が強く、好き嫌いがはっきりしている。 物事の流れを意識することを忘れず、準備や根回しが肝要なり。 強すぎる我は災いを招くため、人の和を大切にすべし。 疲れやすいので、休養をしっかり取るべし。
九紫火星	S30/S39/S48/S57	「火」「炎」の特徴を持ち、明るく華やかで、熱しやすく冷めやすい面をもつ。 外観・体裁にとらわれることなかれ、決断・行動は迅速が吉。 知性・感性が高まる時期なり、努力と誠意が肝要なり。 礼節と規則が運氣向上のカギなり。

### 年盤 五黄土星

南			
	4	9	2
東	3	5	7
	8	1	6
	北		

### ◆2013年のポイント

今年は「五黄土星」中宮・「癸」「巳」であるために、「土」と「水(癸)」と相性の良い【金星(六白・七赤)】が好調、次に【木星(三碧・四緑)】の運氣が良い一年になるであろう。また「五黄」中宮の年は、すべての星が本来の性質が強くなる年となり、吉凶半々で両極端な年になるので、個性を活かして慎重に進むが吉と心得よ。「五黄」には「天変地異」という特徴があり、「水(癸)」とあいまって「水難」の危険もあるので、注意すべし。ちなみに、十二支で「巳年」のため金・財産の年でもありますので、大きな利益を得るかたもいるであろう。ただし、収支の波が大きな一年になるので、悪銭・あぶく銭に、浮かれることなかれ。

### ◆2013年 ラッキーカラー

「五黄」ゆえに「黄」「(黄)金」「白金」で勝負！！



# WACATE-Magazine

## 振り返ってみた!

ども。コヤマンどえす。

創刊号から続けてきた WACATE-Magazine ですが、この度、  
終結となりました。

WACATE 参加者たちの発表の場やつながりの場の提供という  
その主旨として Facebook や Twitter、Blog などの SNS  
の普及により、より簡単につながり、より簡単に発表が  
できるようになったことが、その大きな理由です。

そこで、当時のできごとと、WACATE-Magazine がどう変  
遷していったかをご紹介します。

### ◆0号

実は最初にチラシを作っていました。

WACATE を発信する場としてあらかじめ参加者にチラシと  
いう形で告知していたのですね。

### ◆創刊号

そして発行された創刊号。

WACATE2008 Winter のレポートをメインに

「せんばいにきく」

「ワカテにきく」

「Software Test Topics」

「池田暁のミュージカルに恋して」

「源太郎の『開運ソフトウェアテスト占い』」

などのラインナップで始まりました。

この時には STT は 2 つに分かれていましたね。

### ◆2号

そして世の中で最も早い JaSST Report として

「JaSST'09 Tokyo Report」が作成されました。

なんと開催日の 3 日後にはリリースするという無茶なス  
ケジュールで当時の WACATE 実行委員会総動員で作成し  
たのはいい思い出です。

電子計算機温故知新の第一回もこの号でした。

### ◆3号

JaSST'09 Tokyo で WACATE 実行委員が「やりがいワー  
クショップ」をした参加してきたレポートが掲載されまし  
た。

また、湯本剛様による「ゆもつよの『はい、こちらテ  
スティング事業部』」の連載もこの号から開始されました。  
そして投稿戦士第一号として、現在実行委員の川西さん  
が初投稿されました。

そのほか「補給戦線異状なし」や「WACATE-Blog 出張所」  
の一回目もこの号ということで、大きな方向性が出てき  
た号となりました。

### ◆4号

軽快に続くゆもつよさんの「テストング事業部」のほ  
か、イベントのレポートをメインに据え、WACATE  
ShortShort の告知などがありました。

第一回の WACATE ShortShort は LT 大会でしたね。

この大会で現在実行委員の井芹さんに会ったことを今で  
も思い出します。

巻末にスター☆の写真が掲載されるようになったのも、  
この号からでした。

### ◆5号

WACATE2009 夏の告知とゆもつよさんの連載、そして  
ShortShort の開催レポートなど

WACATE の情報を告知することをメインに据え、アイデン  
ティティを確立していきます。

### ◆6号

ShortShort も第二回のレポートが掲載されました。

ShortShort の 2 回目は御岳山への登山でしたね。

写真を見るとみなさん若いですねw

楽しい交流会でしたね♪

### ◆7号

WACATE2009 夏のレポートがメインでした。

そして初の BPP 受賞者コラムが掲載されました。

このころ WACATE 女子部が結成されたのか、ShortShort  
の vol.6 として現在実行委員の中野さんが Girl's Talk  
の告知をしています。

加瀬さんによるコラム「勉強会のモデレータをやってみ  
よう！」の第 1 回も掲載されました。

### ◆8号

ゆもつよさんの連載が終わりを迎えました。

本連載を読まれた方は少ないかも知れませんがとても  
良い記事ですのでぜひ皆さんもお読みください。

そのほか、JaSST'09 Shikoku をはじめにワークショップ  
や S-Open、@IT などのイベントの報告に加え  
ShortShort vol.6 Girl's Talk のレポート、

そして WACATE 2009 夏の優秀班の投稿など、たくさんの  
WACATE 情報も発信されています。

### ◆9号

本号では Software Testing ManiaX の作成レポートが掲  
載されていました。

最初は WACATE-Magazine の活動の一部のような位置づ  
けでしたね。

そして ShortShort vol.7、国広さんの少林寺拳法イベ  
ントの告知がありました。

当時はそれほど twitter などの SNS が流行していなかつ  
たこともありとても交流がさかんに行われていることが  
わかりますね。

### ◆10号

JaSST'09 Hokaido レポートです。この年から JaSST  
sapporo でなく JaSST Hokkaido となったこともあり、TEF  
道と WACATE の交流が行われたレポートでした。

特集が 4 つも生まれ、とても活発ですがこのあたりから

「せんばいにきく」のコーナーがなりを潜めます。

そして ShortShort の vol.8、永田さんのワイン祭りテ  
ィスティングツアーの告知などがありました。

### ◆11号

JaSST'09 Tokai レポートをメインに据え、そのほか  
WACATE 2009 冬の告知や WACATE-Girls のコラムなど、  
WACATE の情報の発信も忘れません。

すでに ManiaX vol.2 の紹介が行われていますね。



#### ◆12号

WACATE 2009 冬の直前告知や前夜祭後夜祭の告知。  
そのほかにも JaSST'09 Kyushu のレポートのほか  
ShortShort vol. 8 のレポート、**加瀬さんによるコラム「勉強会のモデレータをやってみよう！」**の最終回でもありました。

#### ◆13号

WACATE 2009 冬のレポートです。  
2009 冬の BPP 受賞者の永田さんのコラムも掲載されました。  
この時から実行委員のレポートや参加者視点でのレポートが掲載されるようになりました。  
そして本人非公認・謎の人気企画「今月の STAR☆」が始まりましたw

#### ◆14号

恒例となった JaSST'10 Tokyo のレポートです。  
こう見ると、約4年間、JaSST tokyo のレポートを書き続けたんですね。  
そしてそれをしながらメディアスポンサーとして ManiaX を頒布するということが1月の恒例行事となっていました。

#### ◆15号

SQIP 研究会レポートや勉強会、読書会コミュニティの紹介など WACATE の開催が遠い間も、ほかのコミュニティや情報などでさまざまな情報を発信しています。  
こういったスタイルはこのあたりから完全に確立したような気がします。

#### ◆16号

裏ではさまざまな議論があった WACATE ShortShort も vol. 11 を迎え企画開催方法についての情報なども発信するようになりました。

#### ◆17号

そして WACATE 2010 夏の告知が始まります。  
コヤマンがなかなか更新しない「補給戦線」をまーくさんが代替するという乗り取りもされていますw

#### ◆18号

WACATE 2010 夏と前夜後夜祭の告知がされました。  
編集部員小田部さんによる日々撮影の開始もこの号でした。  
工場見学ツアーの写真にほれ惚れた方も多いのではないのでしょうか。

#### ◆19号

WACATE 2010 夏のレポートと、余興のバンドの裏話、そして BPP 賞のレポートが掲載されました。  
本号は編集部のおたべさんが BPP に前夜祭のレポートに、日々撮影にと大忙しでしたわ。

#### ◆20号

JaSST'10 Shikoku のレポートが掲載されました。  
すでに3号連続で「補給戦線」をまーくさんにされていますw  
本号のあとがきでは、スター☆の実行委員募集ではなく、余興バンドでご活躍された**ゆもつよさんと細川さんのかっこいい写真が掲載**されました。レアですね。

#### ◆21号

WACATE の情報がない時期は JaSST の情報が軒を連ねます。  
JaSST'10 kansai のレポートと JaSST'10 Hokkaido の告知、そして JaSST'11 Tokyo の論文募集などが掲載されました。  
そして久しぶりのコヤマンの「補給戦線」により、軍曹と新兵が帰ってきましたw

#### ◆22号

JaSST'10 Hokkaido レポートです。  
毎年濃密な JaSST'10 Hokkaido はレポートからも熱さが伝わってきますね。  
そして補給戦線はまたまーくさんのコーナーに逆戻り。  
なかなかこちらも熱いバトルが (ry

#### ◆23号

WACATE 2010 冬の告知です。JaSST の情報は Tokai のレポートが掲載されました。  
そして JSTQB の総会に合わせて行われた JSTQB カンファレンス、こちらのイベントは本号から2号に分けてレポートが掲載されました。  
そして、Blog 出張所では JaSST Hokkaido 恒例となった「脱がっかりツアー、ニセコツアー」のレポートが！これで JaSST Hokkaido に行きたい読者が増えたに違いありませんw

#### ◆24号

WACATE 2010 冬の直前告知と JaSST'10 Tokai, JaSST'10 Kyushu のレポートです。  
JaSST'10 Tokai でのポスターセッションの報告などもありました。  
WACATE だけでなく、他のところで何かをしたレポートが増えると、読者のみなさんの加速っぷりが伝わってきますね。

#### ◆25号

WACATE 2010 冬のレポート、BPP のレポートです。  
BPP ができた当初は女性が連続で取りましたがこの頃は永田さん、小田部さん、水野さんと男性陣がリードしています。  
そしてこの号から新企画「てすバナ」が開始しました。

#### ◆26号

毎年恒例となりました JaSST Tokyo のレポートです。  
本号では論文発表をした編集部コヤマンさんのレポートが掲載されました。  
JaSST Tokyo のレポートが入ると一気にボリュームが上がりますね。

#### ◆27号

WACATE 参加者でもあり、@IT 自分戦略のコラムニスト第3パイオリンさんが実行委員長を務めた新 JaSST、JaSST'11 Niigata のレポートです。  
にいがたのお酒を飲む飲んだくれツアーも企画され、にいがたも北海道のように名物ツアーを含めての JaSST スタイルが確立されました。



#### ◆28号

珍しくイベントレポートではなく、書籍の特集です。本号の日々撮影はなんとワイハ！ソフトウェアテストの情報だけではなく幅広い情報が手に入る情報誌です。えっへん。

#### ◆29号

千駄ヶ谷にある日本科学技術連盟にて実施された SQiP ミニシンポジウムのレポートです。そして人気コラム「ワカテにきく。」では先月新潟で JaSST の実行委員長を務めあげた第3バイオリンさんが世界の広がりやを主なテーマとして書かれています。てすバナでは深い深いテストタイプの話が繰り広げられています。世の議論に合わせて WACATE-Magazine の内容もどんどん変遷しています。

#### ◆30号

WACATE 2011 夏の告知です。新企画のてすバナは自動化の話に始まり、メンバーのモチベーションの話などに派生して、どんどん話が膨らんでいきます。悩みを抱えている方も多いかと思うので、ぜひ読んでみてください。何かヒントがあるかもしれません。

#### ◆31号

WACATE 2011 夏、前夜祭後夜祭のレポートです。てすバナでは夏の前に行われたこともあり、「WACATE 2011 夏で得たいこと」をメインに若者が悩みや考えていることなどの意見を交わす熱いチャットでした。

#### ◆32号

WACATE2011 夏の本格的レポートと、JaSST'11 Kansai のレポートです。WACATE 2011 夏の BPP 受賞者はなのっちゃんです。なのっちゃんの熱い熱いレポートは必見です。そして本号にて、創刊号から続いていた池田さんの連載「池田暁のミュージカルに恋して」が終幕となりました。

#### ◆33号

てすバナの内容が濃い33号。「WACATE 2011 夏で得たいこと」に対して「WACATE 2011 夏で得たもの」という内容で語られました。当初の目的を含めてたくさんものを持ち帰った参加者の皆さんの熱い言葉が身に沁みます。そして創刊当初から参加者のバトンがわたっていた「ワカテにきく」が、本号より記載がされなくなりました。

#### ◆34号

JaSST'11 Hokkaido レポートとツアーです。コードネーム「すねーく」の楽し気な脱がっかりツアーの後には「WACATE とは何か」というテーマの深いてすバナが繰り広げられます。

#### ◆35号

WACATE 2011 冬の直前レポートと JaSST'11 Tokai レポートの特集が組まれています。そしててすバナで大活躍をしている名古屋のきょんさんが投稿戦士としてご自身主催のイベント「SCMBootCamp」を紹介してくださっています。

#### ◆36号

WACATE 2011 冬のレポートと JaSST'11 Kyushu のレポートが特集として組まれました。本号のヒットは WACATE 2011 冬のスター☆の名言「自分自身の成長は加速しつつも、結果がでないことについて、焦らない。」ですわね。

#### ◆37号

JaSST'12 Tokyo のレポートです。なんと本号では初となる JaSST 基調講演者へのインタビュー記事が掲載されています。BJ のやさしい顔と力強い言葉「Don't Stop Learning.」は読者の心に刻みこまれたことと思います。その他にも JaSST にて WACATE セッションをやってきたレポートや大投稿として BPP 受賞者の記事やテスト設計コンテストやってきたレポートなど盛りだくさんです。

#### ◆38号

JaSST'12 Niigata のレポートです。基調講演は WACATE でもクロージングをご担当いただいた、清水吉男氏でした。また、JaSST Niigata 恒例となった「にいがた酒の陣」についてのレポートもある、新潟を感じることもできる特集でした。

#### ◆39号

恒例となりました WACATE 参加者の予備知識。初心者の方にはまず vol.39 を読んでいただく、という一つのパターンができました。てすバナでは「テストエンジニアの常識」という深いテーマで議論がされています。こちらも必見です。

#### ◆40号

WACATE 2012 夏の直前メッセージと第二回 Android テスト祭りの特集です。Android テスト祭りでは Android のセキュリティの話からデバイスの仮想化サービスの紹介など、さまざまな情報が発信されていました。参加できなかったモバイル関連の方は今読み返しても役に立つこと請け合いです！

#### ◆41号

WACATE 2012 夏のレポートです。熱いワークに熱い講演、熱いことづくしの WACATE が詰まっています。若手技術者がいきいきとしていると元氣になれますわね。後夜祭の様子もとても楽しそうです♪

#### ◆42号

WACATE 2012 夏の参加者視点でのレポートをはじめ、投稿戦士として3本の投稿があるなど、とても熱い号になりました。BPP 賞受賞者のレポート、TEF 東海による WACATE の振り返り、そして WACATE 参加者中心に集まった仕事術コンテスト。こうして技術者の交流が生まれていくのですわね。



#### ◆43号

WACATE 2012 冬のセッション紹介と予備知識のほか、JaSST Kansai レポートが特集されています。JaSST Kansai ではテスト設計コンテストセッションとして WACATE 参加者でもある名野さんと水野さんが講演されています。さすが BPP 受賞者。加速も半端じゃありません。

#### ◆44号

WACATE 2012 冬の直前情報と JaSST Kyushu、JaSST Hokkaido のレポートです。Kyushu ではあのソニックガーデン倉貫義人氏の講演があり、必見です。Hokkaido はここ数年、濃いプログラムに濃いエクスカージョンツアーと参加者に嬉しいプログラムで毎年行きたくなくなってしまいますね。地域 JaSST の魅力も、たくさんのテストエンジニアに読んでいただきたいですね。

#### ◆45号

WACATE 2012 冬のレポートと JaSST'12 Tokai のレポートです。新実行委員の上條さんの活躍ぶりや、冬なのに熱いワークをあますところなく伝えていきます。JaSST Tokai レポートでは、日本 HP の湯本さんが WACATE でもお話しくださった「目的」と「手段」の階層の話について、さらに洗練された説明をしてくださりました。WACATE 参加者でもあるきょんさんの事例発表など、若手技術者の活躍が見ることが出来るテスト業界を伝えました。

#### ◆そして、最終号一。

JaSST' 13 Tokyo のレポートです。そして、この Magazine 振り返り。かなり独善的というか、やってみようと思って始めたのですが、Magazine を読み返すのは結構大変でしたw あますっペーなーw と思いましたw

一さて。いかがでしたでしょうか。

最終号ということで、勝手に WACATE-Magazine の振り返りをさせていただきます。こう書いてみると、**時代の流れなどをたくさん感じながら、さまざまな活動をさせていただいたのだな**、と改めて感じています。

私コヤマンは、はじめて実行委員になった時から Magazine を作成し続けていたことになりすね。自分は大した人間ではないので、**テストや品質を考える技術者たちがこんなに熱いのだ、ということ**を伝えるのが自分の責務なのだ、とっていました。

そして、それは往々にして受け入れていただいたと思っていますし、それを楽しみに、WACATE-Magazine を手に取っていただいていた、と思っています。

延べ DL 数は万単位になり、参加者だけでなく、興味のある方にたくさん DL していただき、たくさんの WACATE を知っていただいたと思っています。

最近では同人誌発行団体wなどと JaSST で揶揄されることもあります。WACATE とは、テストをする若手技術者向けの熱いイベントなのだ、ということについては、一定の認識を得られており、テスト業界でも通じるようになってきました。

Magazine 発行時は google で WACATE を検索すると 3 万程であったヒット数が、すでに 2300 万にまでなりました。

これらは熱いディスカッション、熱いワークをしてくださった参加者の皆さんの姿がとても印象的であり、Magazine はそれを伝えるお手伝いできていたのだ、という証拠だと自負しております。

そして、WACATE という登竜門をくぐった熱い技術者たちがさまざまなところで活躍をし、情報の発信などを行うに伴い、Magazine はその役割をひっそりと終えることとしました。

今まで WACATE-Magazine を楽しみにしていただき、読んでくださった読者の皆様、本当にありがとうございました。

これからは以下などで、WACATE を知ることができると思います。

◆公式 twitter : [https://twitter.com/WACATE\\_PR](https://twitter.com/WACATE_PR)

◆公式サイト : <http://wacate.jp/>

◆公式ブログ : <http://wacate.jp/blog/>

◆TechVillage 様 (WACATE まとめ)

<http://www.kumikomi.net/archives/2013/05/fo02waca.php>

◆gihyo.jp 様 (ソフトウェアテストタグ) :

<http://gihyo.jp/tagList/%E3%82%BD%E3%83%95%E3%83%88%E3%82%A6%E3%82%A7%E3%82%A2%E3%83%86%E3%82%B9%E3%83%88>

◆@IT 自分戦略様 (第 3 バイオリンさんのコラム) :

[http://el.jibun.atmarkit.co.jp/obligato/all\\_entry\\_list.html](http://el.jibun.atmarkit.co.jp/obligato/all_entry_list.html)

今、とても感慨深い気持ちでこの文章を書いています。私はこの WACATE-Magazine により、たくさんの機会をいただきました。そして、たくさんの素晴らしい人たち、仲間と出会ってきました。

ありがとうございます。WACATE-Magazine。

そして、これからも素晴らしいイベント、WACATE をよろしくお願いたします。

これからはよりライトに繋がっていきましょう。

書いた人：コヤマン



## ❁お知らせ❁

WACATE-Magazine が終わっても、WACATE は続いていますよ！

次回のWACATEは6/22~6/23！

# WACATE実行委員募集

私たちWACATE実行委員会では、  
一緒にWACATEを創っていく同志を募集中です



### ★実行委員になるための条件

何回かWACATEに参加し、実行委員として更なる成長をしたい！というやる気にあふれている方、是非一緒にWACATEという場を作っていきましょう！ 参加したいという方はお近くの実行委員にお声掛けください。是非一緒に活動しませんか？

WACATE夏、およびWACATE冬に最低一回は参加している方  
受身ではなく、積極的に取り組んでいただける方  
WACATEを有る程度知っており、なおかつ、チャレンジ精神にあふれる方をお待ちしております！

### ★得られる価値

実行委員として得られる価値は以下のようなのがあります。

イベントを運営する側としてのノウハウが得られる

司会や発表の場を得られる

他の実行委員や外部講師との交流が深まる

実行委員になるということは、いわゆる

WACATEというプロジェクトのメンバーになることです。プロジェクトの運営を通じた生の経験がきつと職場でも役に立つことでしょう。



### ★注意点

実行委員になるにあたり、以下のようなことに注意していただく必要があります。

活動は基本的にはMLをメインに行いますが、定期的に実行委員会会議を行います（会議は平日の定時時間外や休日に行います）

実行委員会活動に関する諸経費や当日の参加費は自己負担です  
様々な情報を扱うため、その取り扱い等に十分気をつけていただきます

営業目的の参加は固くお断りします

### ★実行委員になるには

お近くの実行委員に直接意思表示を行ってください！



# WACATE

Workshop for Accelerating C(A)Pable Testing Engineers

<http://wacate.jp>

## 編集後記

いのみん★最終回まで読んでくださりありがとうございました。

♪さよならだけど、さよならじゃない〜♪

ヨミ子★いままでありがとう!!!

あたべ★ここまでお付き合い頂き、本当にありがとうございます。

コヤマ★またどこかでお会いしましょう♪

ぼんぼん★これからもすパナは開催したいと思ってまーす☆

★あくづい★

2012/05/28(Tue)はっころ  
WACATE-Magazineへんしゅうび

✉[Magazine@wacate.jp](mailto:Magazine@wacate.jp)

<http://wacate.jp/Magazine/>