

SS2009-HCI ワーキンググループ まとめ

議題1

コミュニケーションと協調

- ソフトウェア開発には対話によるコミュニケーションがたくさん存在する
 - ユーザと開発者、開発者と開発者、開発者とコンピュータ、上司と部下、同僚間
- ソフトウェアはCo-Constructionにより、最初に誰も思いつかないシステムが作れる (undreamed requirements)
 - ユーザと開発のCo-Construction
 - 開発者と開発者のCo-Construction
 - 開発者とマネージャのCo-Construction
- 規模は問題: 100人のプロジェクトをどうコントロールするか
 - ソフトウェア開発には不確実が確実である
 - 行く川の流は絶えずしてしかも元の水にあらず。よどみに浮かぶうたかたは、かつ消えかつ結びて久しくとどまることなし。
 - ソフトウェア開発は戦争に近い
 - 戦争では、指揮を執る将軍は敵も味方も不完全な情報で判断を下す
 - 不確実を承知したうえで、状況に応じて判断し、随時調整する
 - 遅れてちょっと我慢して、開発者の人間力をだせる
 - ダイナミックなチームの編成
 - できる開発者のチームだからといってメンバー同士が旨くコミュニケーションできるとは限らない
 - チームには個性がある
 - プロジェクトリーダーの資格
 - 知識、経験、人間力 (persistent, passion, patience; leadership, 洞察)、コンセプチャルスキル
 - プロジェクトリーダーがチームをコントロールできる限界は、現状の20%以内のメンバー増加まで。
- 開発者の人間力
 - 気合
 - やる気を育てる、引っ張り出すのは大事
 - 技術者のこだわりが必要

議題2

人間中心的なマネジメントの方式 管理、リード、支援

- プロジェクト成功の基準とは
 - プロジェクトによって違う
 - 納期を守る
 - 開発者が成長していく
 - 組織コンセプトを共有するのは大事
 - 成功の決まった標準がなく、satisficingしかできない
- プロジェクトの目的
 - 個人の目的とグループの目的と会社の目的とのバランス
 - プロジェクトリーダーのbenign dictatorship
 - しかし、「会社が。。。」という考え方はリーダーやマネージャの「私が。。。」ということになりがち
 - 適切な裁量権を与えることが大事
 - responsibility without authorityまたはauthority without responsibilityになってはいけない
- フリーランスの開発方式がありうるか：映画の生産方式がソフトウェアに適するか？
 - ソフトウェアは継続的に生きている
 - 作ってからのさきも考えないといけない、誰が保証してくれる？
 - 大きいから保証できる
 - フリーランスの開発には、ソフトウェアの保険会社が必要？
- マネージャとリーダーの関係
 - マネージャはチームのマネジメントをし、リーダーが技術を考えるチーム構成が可能か
 - 意図的に「攪乱」し、チームの力を出せること
 - チームに異分子を入れ、異分子の存在を生かす
 - 自動車工場生産ライン(たいてい男性の数が多)に女性の労働者が入ると、全体の生産性が上がる
- プロジェクトリーダーの育て方
 - 現状ではOJTに頼るが、OJTにはなっていない
 - Situation案件設計
 - SEAのメンバーを集め、リーダーを育てる案件を設計することをやったら
 - Noといえる技術者が大事、しかしNoを言わせるリーダーが少ない
 - マネジメント != 管理
 - 過剰管理が要警戒

議題3

ソフトウェア開発における知識交換と 獲得（考えるための道具）

- 知識・技術・思想の伝承
 - ツールの使い方をどう受けつながいしていくか
 - ソースコードとコメントとドキュメントが大事
- ハードウェアに振り回されるソフトウェア開発
 - 専用、汎用の間にいる準汎用を活用
 - (参考:HCDハンドブック(福田収一先生、丸善出版))
- 考えるための実用的な道具(既存ツールへの不満)
 - 既知の知識を手がかりとして未知の知識を得る
 - ツールが大きすぎる
 - テキスト至上(XML,JSON)
 - つぎはぎのGUI/CLI
 - 他人の考えを分析できる道具が必要
- タスクに応じる知識を収集する支援方法
 - ソフトウェアプロジェクトが一つのソシオテクニカル情報空間をなす
 - 様々な関連性を利用して、必要な知識を取り出す方法

結論と将来の課題

ソフトウェアの四つの特性:

不確実性、不収束性、永続性、公共性

ソフトウェア開発は常に不確実であり、決して収束もしない。開発されデプロイされたら、永続的に存在し、公共的なものとなる。

ソフトウェア開発の鍵を握るのは、開発者とリーダーの気合と人間力、そして関わっているすべての人間の間の対話であり、その対話によるco-constructionである。

人間力を引っ張り出す、育てる職場作りと道具作りがこれからのチャレンジである。