ソフトウェアシステムの「約束」をどう書く？組み立てる？守らせる？

「約束」を用いた賢く頼れるソフトウェアのつくりかた

石川 冬樹， 石川研究室， 本位田研究室

どんな研究？

ソフトウェアが社会の至るところに埋め込まれ、人の生活に深く関わるようになっています。ますます複雑に、そして重要になるソフトウェアを、「賢く頼れる」ようにする研究です。

どんな技術？

ソフトウェアが満たすべき機能や品質に関する「約束」をモデル化し、開発者が、あるいは実行時にソフトウェア自身が、「約束」の分析や組み立てを行う研究に取り組んでいます。

プログラムを作る前に満たすべき「約束」（仕様）を整理し分析，検証を行う技術

実行時に「約束」（仕様・SLAなど）をソフトウェアが把握し、それを満たすよう振る舞う技術

「約束」（仕様・SLAなど）を基にして複数ソフトウェア・サービスを組み合わせる技術

例１．なぜうまくいくかの証明を筋を通したまま小さな塊にばらす

B06で詳しく！

例２．具体例を見せて記述の意味を実感・確認させる

背景： 一般論で書かれた要件は実感しづらいが，特定の具体例だけでは足りない

→ **「うまく」様々な具体例を生成する！**

三辺の長さa, b, cに対して
a == b かつ b == c なら正三角形である

学生は講義のTAを担当してその講義のすべてのレポートを採点する

自分なりに書き出したつもりだけど
漏れや例外、書きすぎはないかなぁ？

一般のエンジニアがとっつきやすいツールを提供
仕様・テスト設計・具体例をすべて書ける言語
による記述を基に，仕様とテスト（具体例）を行き来

```
pre { a >= 0, b >= 0, c >= 0 }
partition "正" { a == b && b == c }
partition "二等辺"
  { a == b && b != c || b == c && …}
partition "不等辺" { a != b && b != c }
```

a	b	c	
3	3	3	正
4	2	1	不等辺
6	4	6	二等辺 不等辺

意図とのずれにすぐ気づく！

怪しいところをつっつくような具体例や反例を生成するための手法を構築

従来のツールより早く・多く誤りを顕在化させるような具体例を定時可能に

エンジニア60名による試用後アンケート
- 様々な使い方ができる
- 視点を広げる教育にいい

IPA RISE 「形式仕様とテスト生成の部分的・段階的な活用」

例３．人と話し合いアプリ間競合を避ける

背景： 「モノのネットワーク（IoT）」「スマート家電」など人の活動に影響する自動制御が増えている

背景： 複数のアプリケーション・ユーザー・機器の制御では，競合する動作によりおかしな状況に陥りがち

→ **起きうる様々な可能性を探索し、ユーザーに説明して競合を避ける方法を決める！**

アプリAの意図
居住者が**いない or 寝ている**とき
窓を閉める

アプリBの意図
息苦しいとき
窓を開ける

様々な可能性の中に，特定状況でのみ現れる競合が隠れている

こういう状況ではどうしたら良い？

居住者がいない時は窓を閉めたまま。寝ている時は窓を開けて良い。

「意味が違う」競合の起き方を分類しそれぞれを例で説明

ユーザの要望も踏まえて実現可能な動作に調整

「モノのネットワーク」サービス開発環境と連動

スマートホームに様々なアプリを随時入れる状況で高速に検査可能であることを確認（35アプリ）

Yagita et al., An Application Conflict Detection and Resolution System for Smart Homes, SEsCPS'15

NII
連絡先： 石川 冬樹 （国立情報学研究所 コンテンツ科学研究系 准教授）
Email： f-ishikawa@nii.ac.jp Web： http://research.nii.ac.jp/~f-ishikawa/