

Niklaus Wirth の逝去を悼む

中所武司

【Niklaus Wirth の訃報】:

<https://ethz.ch/en/news-and-events/eth-news/news/2024/01/computer-pioneer-niklaus-wirth-has-died.html>

- ・コンピュータ分野のパイオニアのニクラウス・ヴィルト（89 歳）が 2024.1.1 に亡くなった。彼は、長期間、チューリッヒ工科大学の計算機科学の教授だった。1970 年代には、プログラミング言語 Pascal の開発で有名になり、1984 年には、計算機科学分野のノーベル賞といわれるチューリング賞を受賞した。

私は、初期の論文で彼の論文を何度も引用し、40 年前には研究室を訪問したこともあり、以下では、引用箇所のほか、研究室訪問時の報告書の抜粋を転記する。

【拙著・論文での引用】

<博士論文での引用：4 件>

- *N. Wirth, Program development by stepwise refinement, Communications of the ACM, 14, 4, 221-227 (April 1971)
- *N. Wirth, The programming language Pascal, Acta Informatica, 35-63, 1971.
- *N. Wirth, Systematic Programming: An Introduction, Prentice-Hall, (1973).
- *N. Wirth, Modula: a language for modular multiprogramming, Software-practice and Experience, 7, 1, 3-35 (1977)

<博士論文>

「段階的詳細化とデータ抽象化を支援する言語 SPL の処理系と環境に関する研究」
学位論文：工学博士（東京大学）（1984. 6）.

<http://www.1968start.com/M/paper/doc.html>

<最初の私の英語論文で、1 番目の N. Wirth の段階的詳細化の論文を引用>

Two-stage programming: interactive optimization after structured programming,
Proc. the 3rd USA-Japan Computer Conference (UJCC), 171-175 (Oct. 1978)

<http://www.1968start.com/M/paper/1978UJCCchusho.pdf>

<拙著での引用>

・引用した N. Wirth の著書:

N. Wirth: Algorithms + data structure = programs, Prentice-Hall, 1976.

(片山卓也 訳: アルゴリズム + データ構造 = プログラム、日本コンピュータ協会)

・拙著 (引用先)

*「ソフトウェア工学 (第2版)」朝倉書店 (Mar. 2004)

*「ソフトウェア工学 - オープンシステムとパラダイムシフト - 」, 朝倉書店 (1997).

- *「ソフトウェア危機とプログラミングパラダイムーわかりやすさの追求ー」啓学出版 (1992)
- *「プログラミングツール」昭晃堂 (1989)

【N. Wirth の研究室訪問】

1983.9の世界コンピュータ会議 (IFIP83、パリ)での論文発表のための海外出張のおり、その前の週に、エジンバラ大学 (D. L. Bowen)、ロンドン大学 (R. Kowalski)、チューリッヒ工科大学 (N. Wirth) を訪問した。

以下は出張報告書からの抜粋：

<http://www.1968start.com/M/essay/8309VisitWirth.pdf>

(このときの海外出張報告 [39 ページ] の電子化ファイルが残っていた)

・訪問先

チューリッヒ工科大学 N. Wirth 教授

(目的：言語設計技法、話題：Pascal Ada Modula-2)

・訪問日

1983.9.16 (金曜)

・特記事項

N. Wirth 教授は、システム記述言語として自作の Modula-2 を推奨している。

・入手資料リスト

N. Wirth (ETH) : Release of Modula-2 Compilers

・内容

(1) Pascal について

Q：言語使用拡張が不統一

A：ISOの標準化が遅すぎた

Q：スカラー型の定数名のスコープルールが変数と同じ

A：コンパイラの簡単化のため。

Q：ユーザ定義データ型の型チェック方式があいまい

A：ISO規格では、name equivalence

(2) Modula-2 について

*特徴：システム記述言語、分割コンパイル方式、モジュール間の型チェック

*コンパイラは、入手可能

(3) Ada について

*パッケージの parameterization は魅力的だが、コンパイラが複雑化してダメ。

(4) 最近の研究

*ソフトウェア技術者用パソコン Lilith のマシンの研究は終わり、

Sophisticated Interface を研究中。

・感想

*N. Wirth の「段階的詳細化」を実現する言語の開発報告をすると飲んでくれた。

*Pascal に関する質問事項を用意していったが、本人が興味を失っていた。

*彼の設計思想は「Simple is best」

*最近、学生が多くなって、研究時間が少なくなった。

以上

