Therac-25 放射線事故

Therac-25 は電子線と X 線を使って，それぞれ身体の表面と深部にある腫瘍および癌を治療するための放射線治療器である．カナダに本拠を置く AECL 社が製造・販売したこのマシンは，1983 年から北米の病院に 11 台導入され，1985 年から 1987 年までの間に 6 件もの放射線の過剰照射事故を引き起した．

(1) 事故の経緯

Therac-25 の開発以前に，AECL 社は 6MeV の X 線を照射できる Therac-6 と，その後継機で X 線と電子線の二種類の放射線を 20MeV のレベルまで照射可能な Therac-20 を開発・販売していた．これら両機種共に放射線治療器として高い評価を得ており，もちろん事故を起こすこともなかった．Therac-20 の制御用ソフトウェアは Therac-6 のいくつかのモジュールを再利用して開発されたものであり，Therac-25 の制御システムは Therac-20 のソフトウェア・モジュールを再利用して作成された．さらに Therac-25 の開発においては，その動作を完全にソフトウェア制御するという「技術革新」が断行された．

機器本体に対する詳細な危険分析が行われた後， Therac-25 は 1983 年に出荷され，それが導入されたそれぞれの医療機関で 1985 年半ばまでは事故なく利用された．ただしこの危険分析では，ソフトウェアは分析の対象から除外されていた．これは，それまで無事故であった 2 つの先行機種のソフトウェア・モジュールを再利用することによって高いソフトウェア品質が維持されるという前提に立つものであった．
しかし，1985 年 6 月のジョージア州での過剰照射事故を皮切りに，続く 7 月にはカナダ・オンタリオ州で，12 月にはワシントン州で，さらに 1986 年 3，4 月にテキサス州において事故が起こり，最後に 1987 年 1 月に再びワシントン州において過剰照射事故が発生した．一連の事故の状況は 表 1 にまとめられている．そこに示したように，患者への被害は甚大なものであった．

表1：事故の結果と原因

時期	場所	患者への被害	原因	照射された放射線量	処方で予定された線量
1985年 6月	ケネストーン地域腫瘍学センター（ジョージア州）	放射線治療によってやけどが残ったので、胸部を切除する。完全に肩・腕の機能を失う	不明（訴訟が示談になり、調査が可能ではない）	15000～20000rad	鎖骨部位に10MeV（正確なradは不明）
1985年 7月	オンタリオ癌財団（オンタリオ州）	過剰照射が原因で、腰（治療箇所）にやけどと痛みを受け、重度の癌にかかり11月に死亡。	ターンテーブルの位置を決めるマイクロスイッチが壊れたが、ソフトウェアの可能性が濃厚で、最終的に、ターンテーブル位置を確認する独立したシステムの導入がユーザの技師・監督官庁から求められた。	13000～17000rad	不明
1985年12月	ヤキマ渓谷記念病院（ワシントン州）	慢性の皮膚癌（皮下組織の壊死）にかかり、手術の末、かろうじて生存。	ヤキマでの2度目の事故が調査されるまで、「原因不明」	不詳	不明
1986年 3月	東テキサス癌センター（テキサス州）	事故の翌週に、左脚の機能を失い、声帯麻痺、膀・膀胱の神経性の障害、頭部の骨髄炎、左横隔膜の麻痺で入院する。過剰照射の合併症が原因で8月に死去。	オペレーターのキー操作速度（モード入力の訂正）・コーディングエラー・あまりにも不明瞭なエラーメッセージ	約1 cmの範囲に対して1秒間に16500～25000rad	180rad
1986年 4月	東テキサス癌センター（テキサス州）	過剰照射が原因で5月に死去。（検視解剖では、脳幹と右側頭葉に高濃度の放射線障害があると発覚。）	オペレーターのキー操作速度（モード入力の訂正）・コーディングエラー・あまりにも不明瞭なエラーメッセージ	正確には不明（25000radとみられる）	不明
1987年 1月	ヤキマ渓谷記念病院（ワシントン州）	事故以前に末期癌にかかっていたが、過剰照射の影響もあって、4月に死亡。	エラー状態でも機能してしまうソフトウェアの動作設定・技師の入力ミス	8000～10000rad	86rad

※100rads=1Gy（グレイ）=1Sv（シーベルト）=100rem（レム）600～700radsが致死量といわれている。
一回の治療での最大の線量は約200radまでだった。通例、複数回の治療で合計で数千radが照射される。

(2) ソフトウェア開発の落とし穴

Therac-25 のソフトウェア開発は，Therac-20 の制御用ソフトウェアに対する全幅の信頼の上に成り立っていた．Therac-20 は医療現場で事故なく使用されており，医療現場も AECL 社もその安全性に高い信頼を置いていたのである．

しかし Therac-20 には，入力したデータ項目をある特定の方法で変更するとヒューズとブレーカーが作動して，機器動作が強制終了させられるという異常があったのである．制御プログラムに潜むバグが引き起こしていたこの異常は，オペレータにとっては Therac-20 に付き物の不具合（nuisance）であって，深刻な欠陥であるとは考えられず，ましてや制御用ソフトウェアにバグが存在するとは認識されなかった．しかも，この不具合の存在そのものもテキサス州での事故後にシカゴ大学のエンジニアが指摘するまで問題視されることもなかった．つまり，こうした不具合を Therac-20 が抱えているにもかかわらず，制御ソフトウェアを「安全な」ものとし，そのモジュールを再利用して Therac-25 の制御ソフトウェアは開発されたのである．このことは，長年にわたって安全に運用されてきたソフトウェアのモジュールを再利用することによって，ソフトウェアの安全性とソフトウェア開発における生産性の向上を両立させ，さらにはメンテナンス費用を抑えることができるというソフトウェア開発における常識が，ソフトウェア品質の劣化を招いてしまったと見ることができるであろう．

(3) バグはなぜ隠れ続けていたのか

Therac-25 の使用によって引き起こされた事故は，オペレータが誤って「x」キー（X 線照射モードの設定）を押し，「8 秒以内」に「↑」キーを押すことで電子線照射モードに変更した場合にしばしば生じるものであった．

「8 秒以内」が問題になるのは，次項で詳述するように，制御用ソフトウェアの処理時間と機器本体の動作時間との関係で機器の異常動作が発生するためであり，その一方で「8 秒以内」が現実に発生するにはオペレータが機器操作に習熟し，キータイピング速度が速くならなければならない．この皮肉な条件のために，通常の使用・テストでは，問題となる機器の異常動作は発生しなかった．このため，ソフトウェア開発段階のテストではバグの発見は困難であり，一連の事故の後で事故を再現することも難しかった．Therac-25 で再利用された Therac-20 の制御ソフトウェア・モジュールに潜んでいたバグによって発生させられていたこの異常動作は，Therac-20 ではその設計が過剰照射をハード的に強制終了させるようになっていたために未然に防ぐことができたのである．他方，完全ソフトウェア制御の Therac-25 においては，人命を失う事故となって現れたのであった．

(4) 事故状況の詳細

テキサス州での事故では，最初，オペレータは電子モードで治療するところを「x」と入力してしまった．これ自体は，単なる入力ミスで，モードを切り換える操作をすることで解決できるとオペレータは期待するはずである．

他方，機器本体の動作を見ると，「x」が入力されたとき，Therac-25 内部の複数の磁石の位置を設定する（制御ソフトウェアの）サブモジュールが実行される．磁石位置の設定動作の完了には 8 秒間かかり，一方，この間に照射モードの変更が指示されると，磁石位置を示すフラグが「X 線照射用」を示す値に変わっていないことがあるため，モード変更のための別のサブモジュールが正しく実行されず，過剰照射が発生する．

ユーザインターフェースの不備も存在していた．制御用ソフトウェアのサブモジュール間でデータの受け渡しをしていて，オペレータの入力を受けつけない状態にある時に，適切なメッセージを表示して入力を待たせるといった機能はなかった．くわえて，モードや照射量の変更を行った際に，「変更完了」といったメッセージが操作用モニタ画面に表示されることもなかった．このため，オペレータは入力ミスを手早く直し，無事に変更されたと思い込んでしまった．過剰照射時にモニタ画面には「誤動作 54」という

表示がなされたのであるが, 取扱説明書等のこのエラーコードに関する記述や開発企業側の担当者の説明はまったく不足していた．さらに不幸なことに，被曝を防ぐため，オペレータは Therac-25 のある治療室とは別の部屋にいて，患者をビデオモニタ越しに見ているので, 患者の急変に気づいてとっさに医療機器を強制遮断するという対応は難しいものとなっていた．

(5) 事故原因の究明と報告における問題点

これだけ悲惨な事態を引き起こしながら, 事故原因を解明する AECL 社の取り組みは杜撰なものであった．特に，最初の事故によって 1985 年 11 月に遺族から提訴を受けているにもかかわらず，事故に対して調査を開始しなかった点や FDA（アメリカ食品衛生局）への報告を怠っていた点が事故発生当初の問題として指摘できる．

AECL 社側は Therac-25 の安全性に対して事故の根本的な原因が機器本体にあると思い込んでおり，ソフトウェアに対する過信が存在していた．くわえて，ユーザへの対応や事故の情報を監督官庁やユーザと共有し解決していく仕組みを欠いている点が問題であったといえる．この事故原因については，ソフトウェアのエラー，不明瞭なエラーメッセージ，不完全なマイクロスイッチ，ハードによる安全制御の欠落，システムに対する不適切なテストと評価が指摘されている（Anderson and Goodman, 2002）．しかし，こうした組織的問題も無視することはできないのである．

(6) 事故の問題の整理

Leveson and Turner (1995)も指摘しているように，この事故は複数の要素が折り重なって生じたといえる．それは，「利用している現場の問題」，「ソフトウェア開発の問題」，「システム開発における問題」，「マネジメントの問題」，「政府・監督官庁の問題」に大別できる．

ソフトウェアシステムを利用する現場では，技術者（開発側）の説明を鵜呑みにする傾向が強い．基本的に，ユーザだけで事故や不具合の原因を追究するのは困難であり，致命的な問題があったとしてもその存在・重要性を知覚できない．くわえて，多少の問題があっても，通常の業務に支障がなければ，システムを利用し続けることが多い．
特に，Therac-25 の事例では，現場の人間は，事故の発生当初は，「今まで，事故は起きていない」，「過剰照射など起こるはずがない」と伝えられていた．過剰照射事故が発生した後も，引き続き Therac-25 を使い続ける医療機関もあった．

一般にソフトウェア開発では, many hands と文書化とバグの存在が問題となる. また, ソフトウェアの利用においては, 開発者の意図が狙い通りにユーザに受け入れられるとはかぎらないことが問題となる．場合によっては，ユーザ支援や安全性確保のための機能やツールを削除・無視して利用することもあり得る．

この事例ではさらに，モジュール再利用に関する問題が存在している．再利用したソフトウェア・モジュールを異なる設計思想の機械システムに組み込んで作動させる場合, 利用現場に近い条件での機械システムの安全性テストが必要になる．
マネジメントの課題としては，開発・運用・保守・事故対応といったそれぞれの段階でユーザとの情報共有（共有範囲の明確化）と責任の範囲の明確化，事故発生時に迅速に対応できるような体制をつくっておくことやシステムに対するリスクマネジメントなど多岐に渡る．

また，Therac-25 の事例において明白になったように，事故原因の究明が疎かであると，事故にかかわった当事者のみならず，医療機器のユーザや監督官庁の人間を不安にさせ混乱させる．悲惨な事故が出来るだけ発生しないように医療機器を設計・製造することが大事であると同時に, 事故が起きた際に迅速に対応できるように組織の体制を整えておくことが医療機器製造のマネジメントに求められる．

こうした事故の発生にそなえた組織作りがなされていたならば, ユーザからの情報もより有効に活用できたのではないであろうか．実際，本事例においては， Therac-20 の

バグの存在の発見やバグの発生条件の解明など事態の究明にユーザが大きく貢献した．ユーザとの情報共有は，情報システムの安全な運用において極めて重要であるといえる．

政府・監督官庁については，情報収集が不十分であり，適切な行政指導が行えなかったと指摘できる．Therac-25 の事件当時は，FDA は病院で起きた死亡・重症事故と設備エラーの１％すら知らなかったため（Bowsher, 1990），事態の収拾に対して全く無力であったといえる．本事例でも，FDA は開発側に行動修正計画書の提出を求めたが，要求内容が不明瞭であったため計画書の提出まで相当な日数を要した．

参考文献

Anderson, J. G and Goodman, K. W., Ethics and Information Technology: a Case-based Approach to a Health Care System in Transition, Springer, 2002.

Bowsher, C. A., “Medical Devices: The Public Health at Risk”, US Government Accounting Office Report GAO/T-PEMD-90-2, 046987/139922, 1990.

Leveson, N. G. and Turner, C. S., “An Investigation of the Therac-25 Accidents,” in Johnson, D. G. and Nissenbaum, H. (eds.), Computers, Ethics & Social Values, Prentice Hall, pp.474-514, 1995.

ケース・メソッドのための質問

１．この事故において，どのような行動主体が，どのような責任を負わなければなりませんか。

ヒント：事例に登場するすべての行動主体を書き出し，それぞれがどのように振舞えば事故を未然に防げたかを考えてみなさい。また，事例に登場しない行動主体についても検討してみなさい。

２．ソフトウェアエンジニアがソフトウェアを開発するとき，どのような葛藤状態に陥ると考えられるでしょうか。

ヒント：「専門家」であるソフトウェアエンジニアが，多くの場合，企業に雇用されていることに注意しなさい。