

## 受領書

令和 3年 1月18日  
特許庁長官

識別番号 110000752  
氏名(名称) 特許業務法人朝日特許事務所 様

以下の書類を受領しました。

項番	書類名	整理番号	受付番号	提出日	出願番号通知(事件の表示)	アクセスコード
1	上申書	—	52100119233	令 3. 1. 18	特願2019-192265	

以 上

【書類名】 上申書  
【提出日】 令和 3年 1月18日  
【あて先】 特許庁長官 殿  
【事件の表示】  
【出願番号】 特願2019-192265  
【上申をする者】  
【識別番号】 598037422  
【氏名又は名称】 根来 文生  
【代理人】  
【識別番号】 110000752  
【氏名又は名称】 特許業務法人朝日特許事務所  
【代表者】 川▲崎▼ 研二  
【電話番号】 03-5280-5001

【上申の内容】

本願の親出願となる特願2015-200064に関し、令和2年11月10日付で発送された審決により、請求不成立（拒絶）との認定が行われている。

以下に、出願人（本願発明者）は上記審決の内容が不当であることを主張すると共に、本願請求項1に係る発明の前提となる重要な技術である同期型プログラム（SF）について補足説明を行う。

SFではその請求項目は異なっても、その明細書は同じになる本願の特徴をはじめに告げる。そして、以下の添付資料は、3件目の出願明細書に対する日本国特許庁の拒絶理由の誤りを本願発明者が指摘するために、本願発明者が別のところでSFについて解説したものである。

本願の審査にも参考になる筈なのでそれをここに添付する。因みに、明細書が同じで請求項だけが異なるSFに掛る出願では既に日本国で2件、米国1件が特許査定を受けている。

同期という形而上学的世界をプログラムで捉える本願発明のSFの世界を、本願参考文献を精読することもなく、認識することは容易でないことを告げておく。特に、非同期世界の動性アルゴリズムが実行時に発症させるプログラム課題（バグ事象、ウイルスの無防備）を全て自律的に解法するSFの動性アルゴリズムの仕組の理解を得ることは不可能であることを告げておく。

SFの静的構造とその動的アルゴリズムの世界と非同期の静的構造（CP）とその動的アルゴリズムの世界は天と地ほどの違いがあるからである。

[1] SFとCPについて

(1) 科学的と称されるひとの思考法は、時にはひらめきの様に同期型が混じる可能性があるが、本能的に非同期型が核になっている。

(2) そして、非同期型思考法はいくら山済みしても、その全体は部分である。故に非同期型の全体は唯一つとは限らない。

(3) 他方、同期型思考法は全体である。そして、その山済みもその全体は唯一つとなる。

(4) SFとはひとが同期世界を捉えるプログラムの静的構造である。

(5) SFはCPの未解決問題の解法論を同期世界で捉えることができる。

(6) CPは自己的な非同期世界を自己的に主張する存在である。

(7) 同期型思考法は人が真理を求めるために本能的な非同期型に抗する人類の反省の表れである。

(8) 本願発明者は別の明細書で貴庁に同期型SFの動性アルゴリズムの効果の審決を求めたのに対し、

無謀にも S F の動性アルゴリズムを貴庁は非同期型アルゴリズムに置き換えて論決し、S F の動性アルゴリズムを論題から退けて、S F の動性アルゴリズムを拒絶している。

(9) これがいかにどの恥ずべき誤りであるか貴庁はいつの日か気付くことになる。

(10) 名詞数 N の C P が実行時に創出する動性アルゴリズムの処理経路数を M とすれば、そして、S F が実行時に創出する動性アルゴリズムの処理経路数を E とすれば、 $E = 2$  の N 乗となる。

(11) E は M よりも巨大数となる。

(12) 巨大数となる S F の処理経路数はそれ故にその動性アルゴリズム上の仕組で発症するいかなるプログラム問題 (たとえば阻害事象) をも解法するアルゴリズムを自律的に成立させることができるのである。

他方、C P の動性の処理経路数では C P の動性アルゴリズム上に於いて発症するプログラム問題 (たとえば阻害事象) を解法するアルゴリズムを自律的に成立させることができない。これが S F と C P の根本的な相違いである。

[2] 以下に述べる内容は、貴庁の審決に対して、本願発明者が貴庁に提供する S F の講義的解説である。

(1) 改めて言う。本願は C P の動性アルゴリズムの効果に関する審決を貴庁に求めているのではなく、S F の動性アルゴリズムの効果に関する審決を貴庁に求めている。

(2) S F とは本願発明者の 40 年余の論考により、もたらされた I T 世界では最初の同期型プログラムの静的構造である。

(3) S F を決定する普遍的なプロトタイプを特に「S F の方程式」と呼ぶ。

(4) S F の方程式を用いて開発要件を充足させる S F を求めれば、プログラム言語、開発要件が同じなら求められる S F は一義的になる。

(5) S F が一義的になるとはそのソースには誤りがないということの意味する。結果、S F の方程式は S F を決定する普遍的な方法論として位置づけられる。

(6) 貴庁の審決では S F は本願が初めてではないという主張だが、貴庁が知るそのプログラムは以下の条件を満たしていなければならない。即ち、

(i) そのプログラムは S F の唯一つの方程式で統治されている条件と同じ条件を満たしているか。

(ii) プログラム言語と開発条件が同じならそのプログラムのソースは一義的になるか。

(7) 上記 (i) (ii) が NO なら貴庁のいうプログラムは、同期型プログラムではない。

(8) 実行時に S F から導出される動性 (または動的) アルゴリズムには、S F が同期型プログラムの条件を満たしているので、同期性が成立する。同期性については、後述の「同期性について」を参照のこと。

(9) 貴庁が審決で指摘する阻害事象は S F の動性アルゴリズム上で発生してもこの動性アルゴリズムの同期性効果で自律的に正常な事象に再生されるかまたは排除される。これは阻害事象が S F の動性アルゴリズム上では実質的に存在できないことを意味する。この問題は動性アルゴリズムのある箇所だけを修正すれば解法されるという性質の問題ではない。つまり、動性アルゴリズム全体に掛る修正を必要とする問題となるので、C P の動性アルゴリズムではこのような再生作用を成立させることは不可能なのである。貴庁はそのことを拒絶理由書で述べている。

(10) 本願発明者は S F が決まれば全てのプログラム問題を解法する S F の動性アルゴリズムが決まるその動性アルゴリズムの効果について審決を貴庁に求めているのであり、C P の動性アルゴリズムの効果の審決を求めているわけではない。即ち、S F の動性アルゴリズムの世界では貴庁が指摘する拒絶理由は S F の動性アルゴリズムで自律的に解法されるので、特許法 36 条を動員しても存在しない存在なのである。

存在しない事象を存在とする貴庁の主張は貴庁の問題に由来する問題であり、貴庁の主張は本願発明とは何の関係もない主張なのである。

(1 1) S Fの動性アルゴリズムによりS Fの解(主語系譜)が成立する。解が成立しなければS Fは同期型プログラムではない。

(1 2) 貴庁が指摘する障害事象の発生はS Fの動性アルゴリズムで自律的に正統な事象に再生されることに於いて、S Fの動性アルゴリズム上では誤りではないのである。もし、存在するとすればついには消滅する動性アルゴリズムの途中経緯に過ぎない。

(1 3) S Fの動性アルゴリズムを学べば貴庁に於いてもこのことへの理解に至ることができるかもしれない。しかし、C Pの世界に住んでいるだけでは何十年その世界の経験を積んでも、その経験が本質的に軟弱なるがゆえにその誰もがC P世界の課題、S Fの静的構造の意味、同動性アルゴリズムの意味の理解に至るとは限らない。本願発明者は非同期型の技術者を指揮指導し様々なシステムを開発している。故に上述のことが解るのである。ひとを月世界に到着させようが何を果たしたところでそれらプログラムの由来がC Pである限り、それらプログラムがプログラムの正鵠を捉えているわけではない。これが今日の砂上の楼閣的なIT技術の限界なのである。

(i) S Fの動性アルゴリズムはその本質性で障害事象を自律的に捉える。

(ii)そして、S Fの動性アルゴリズムは障害事象を捉える限りにおいて、この障害事象を自律的に正統な事象に再生させる。付言するとこれがS Fの動性アルゴリズムの特性である。

(iii)結果、貴庁が指摘する障害事象は正当な事象に転移(解法)される。故に、S Fの動性アルゴリズムに於いては貴庁が指摘する障害事象は存在しない。付言するとこれがS Fの動性アルゴリズムの特性である。

(iv) S Fの個々の動的アルゴリズムは、障害事象の正常化処理の終了を自律的に捉え、さらなるステージの処理を自律的に選び、自律的にその処理に移る。

(1 6) 総括的に言えば、S Fの動性アルゴリズムとは上記(i)~(vi)を自律的に果たすプログラムである。そして、これがS Fの肝である。

(1 7) 貴庁の審決を見る限り、本願発明者は貴庁の誰もととは思いたくないが、貴庁の大勢はS Fの動的アルゴリズムのこの仕組みには気付くことができていないことは明らかなのである。

(1 8) 付言すれば、本願発明者は同期型S Fの動的アルゴリズムの効果の審決を求めているのに貴庁では非同期型C Pのアルゴリズムで審決を行っていることも明らかなのである。これは貴庁の多量の無意味な拒絶理由が自らこのことをよりによって同期型プログラムの専門家である本願発明者に示したのである。更に、貴庁にはC PとS Fの違いを理解できる者はゼロだとまでは言わないがこの審決にはこの理解者が参画していないことは明らかである。そして、S Fの理解度ゼロの担当者にS Fの審決を学ぶこともさせずに、させる貴庁のセンスにもただただ呆れるほかはない。

(1 9) 故に貴庁の50頁に及ぶ拒絶理由の内容は誤記と悪意に満ちた主張なので、貴庁の拒絶理由は本願発明者により上記(1 8)の理由だけで相手にされないだけでなく法的に拒絶されることは否めない。貴庁はS Fの動的アルゴリズムの効果プログラミングの技術的視点、その論考法の科学的視点、その哲学的視点で捉えなければならない立場に居ながら、拒絶理由を見る限り、50頁に及ぶ意味のない内容はアルバイト者にでもできるような拒絶理由に関する情報を集める作業と、そこに特許法36条をちりばめる作業で作成している感はぬぐい切れない。

(2 0) 本願発明者が日米に対して同じ内容の明細書を提出してS Fに係る4個目の出願に関する審査・審決を求めたのは拒絶理由を知りたかったからである。米日ともに特許を受けた3個目の出願の途中経緯に於ける質問、拒絶理由からは当方も学ぶべきものがあつたが、この度の貴庁の審決の内容は既に特許査定を受けたものの途中経緯のものが含まれていることも含めて貴庁のおぞましさを感じさせるだけの意味のない記載しかない。プログラミング技術に掛る不誠実さが浮かび上がっている。

(2 1) 貴庁には不要な助言だが、本願で論じられるアルゴリズムとは無限の精神性である。故に、その学びは無限性の由来を論考することに始まる。なぜなら、S Fの方程式は不可避的に課題を発症させる有限的なアルゴリズムを発生させるのに対して、S Fの動性

アルゴリズムはその課題を解法する無限性のアルゴリズムをS Fの動的アルゴリズムとして成立させているからである。

(22) もし、本発明者が要請する座学を貴庁が受けていれば、貴庁は誤った拒絶理由書を書かずに済んだことは確かである。貴庁は審決しなければならないS Fの動的アルゴリズムを理解できないにも関わらず、アルゴリズムの完全体と評されるS Fの動的アルゴリズムを伝統的に欠陥品であるC Pが創出する動的アルゴリズムを用いて、且つS Fの明細書を適当に流用して適当な拒絶理由をでっちあげて審決をすませていることは明らかである。

(23) 本願発明者により導かれた同期型プログラムの普遍的なソースの構造を「S Fの方程式」と呼称する。

(24) 本件に係る座学をこの度の貴庁は受けていないことに於いて、貴庁はS Fの知見を正確に得ているとは思えない。

(25) S Fの方程式に登場する全コードの全景は明細書に記載されている。これをコード単位で、トレースすれば電算機が実行時に創出されるS Fの動的アルゴリズムとなる。これを用いれば、貴庁が拒絶理由でいう阻害事象を正確に判定することができる。

(26) S Fの動的アルゴリズムは、1個の同期関数、3個の座標関数、そして、システムに属す全名詞の名詞ごとに定義される全ベクトルを用いてS Fの方程式で決まるソースを起源としている。そして、S F全体の再起が継続的に繰り返される構造になっている。論理結合型のC Pの動的アルゴリズムとはその成立由来からして異なっている。

(27) 貴庁はC Pの動的アルゴリズムを用いてS Fの動的アルゴリズムを拒絶し、結果として論拠のない50頁に及ぶ虚偽の拒絶理由を作成していることになるのだが、貴庁の誰かがこのアプローチが誤りであることに気付かなければならない。貴庁はC Pの動性アルゴリズムではなく、S Fの動性アルゴリズムを用いて(i)拒絶事象の発症経路を捉え、(ii)それにより貴庁が指摘する阻害事象がその動的アルゴリズムに生じ、(iii)実施要件が満たすことができないことを、このS Fの動性アルゴリズム上で明らかにしなければならぬ。貴庁はC Pの動性アルゴリズムで上記(i)(ii)に言及しているのだが、これは明らかに埒外である。故にこれ以上、貴庁の偽作の拒絶理由に言及しても意味はないが貴庁は存在しない阻害事象を用いてそれを拒絶理由の論拠にしているということである。貴庁がこの誤りに気付くにはS Fの動的アルゴリズムを学びその性質を十分に理解しない限り、貴庁がS Fの動的アルゴリズムの効果を論決する立場に立つことは出来ないということである。

(28) 本願に関し、S Fの方程式で示されるソースの構造、令和2年3月9日付で提出した審尋に対する回答書の添付資料(以下、単に添付資料という)の1つである「シナリオ関数の全景」は貴庁には初めてのものである。そして、言語特性を除けば、これは人類にとり唯一つとなる情報である。そして、その動性アルゴリズムが統治する経路数はC Pのそれとは比較にならない大きさとなる。貴庁はこのようなことについても何も理解できていないわけではない。審決する立場にはいても審決することができないということは明らかである。

(29) S Fの動的アルゴリズム数はC Pのそれと較べれば巨大数となるがその成立の由来は普遍的である。そして、本業のための処理の傍らいかなる方法で侵入したウイルスでも、C Pでいういかなるタイプのウイルスでも自律的に捉え、自律的に無力化する。だからといってS Fのソースにはウイルス対応のための特別なコードが設けられているわけではないのである。それはS Fの方程式から導出されるソースから創出されるS Fの動的アルゴリズムが完全体の条件を満たしているからである、この完全体条件とは正統性が保証される情報以外は処理されないということである。情報に掛るこの正統性条件とは「主語の脈絡の終点がR 2ベクトルの主語並びに、L 2ベクトルの主語となることである。主語の正統性を保証するアルゴリズムはS Fの動的アルゴリズムの基底を成すものである。付言すれば、これを基にS Fの動的アルゴリズムが成立する関係になっているからS Fのソースにはウイルス対応のための特別なコードを設ける必然性が生じないのである。この仕組がウイルス問題だけに限らず、バグ構文問題、拒絶事象問題を解法する仕組にもなって

いるのである。CPの動性アルゴリズムでは及びもつかないSFの動性アルゴリズムの効果は以上のような仕組によりもたらされるのである。わずかな時間で済む座学を受けることもなく同期型プログラムの動的アルゴリズムの効果はだれが議論できるというのだろうか。CPの動的アルゴリズムと異なり、SFの動的アルゴリズムはウイルス問題を解法こそすれ発生させることはない。そして、ウイルスと同じ由来に立つサイバ攻撃はウイルス問題が解法されるのと同様に、飛来する攻撃型ミサイルを無力化させるのと同様にサイバ攻撃を無力化するのである。以上の記載を記憶にとどめ以下の記述内容に挑んで頂きたい。

(30) 本願発明者は貴庁の拒絶理由の話を聞きながら、普遍的で唯一組のSFの動性アルゴリズムが、貴庁の指摘する障害事象を解法する様子を座学程度のレベルで解説して、同期アルゴリズムの本質性が貴庁に伝えられればよいと考えていた。SFの動性アルゴリズムに掛る本願発明者の半世紀に及ぶ論考は既に誰からも拒絶されるようなレベルの事象ではない。そして、貴庁のSFの技術総量ではまだ同期アルゴリズムの頂上には立てないことは明らかではあったが、本願発明者はそのことで貴庁を否定するつもりはなかったからである。しかし、貴庁の審決の内容を読み進めると、貴庁はSFが同期性ゆえに普遍的なSFの動性アルゴリズム上で生じる障害事象をSFの動性アルゴリズムがどのように捉え、捉えた障害事象をSFの動性アルゴリズムがどのようにして正常化させるかについて、貴庁はSFの動性アルゴリズムの効果を学ぶべきところを、また、貴庁はCP上の動性アルゴリズムが発症させる障害事象をCP上の動性アルゴリズムでは対応処理が何もできないとするべきところを、SFの動性アルゴリズムでは対応処理が何もできないと書いている。これではSFの動性アルゴリズムの効果を論じていることにはならない。SFの動性アルゴリズムでは障害事象が生じればこれを自律的に捉え且つこれを自律的に解法して正常化させる。これがSFの特性である。故に、貴庁の審決は本願発明者が求めているSFの動性アルゴリズムの効果の審決にはなっていない。付言すれば、貴庁の主張はSFの動性アルゴリズムの効果とは関係のないCPの動性アルゴリズム上での様相である。本願発明者がSFの動性アルゴリズムで解法可能とするCPの未解決問題である。本願発明者が貴庁に求めている様相はSFの動性アルゴリズムの効果についての審決である。故に、貴庁の審決に記載の拒絶理由はそれに答えたものにはなり得ていないことは明らかである。

(31) 本願の肝である同期型プログラムSFの動的アルゴリズムが可能とする様相をCPの動性アルゴリズムを動員して否定している貴庁の拒絶理由は、同期型プログラムSFの動的アルゴリズムの特徴が理解できていないことが原因なのであるが、他方、これが非同期型CPの動性アルゴリズムの未解決問題であるとの常識にもいたっていないことである。そして、両者のアルゴリズムの特化されるべき区別すら理解できていないことである。そして、このありさまは貴庁に限られる問題ではなく、現状のIT業界の数多のCPの未解決問題を解法することができない専門家と称する人々の常識なのである。

(32) 貴庁はこのような状態で、本願発明者が要請している座学も受けずに、SFの動性アルゴリズムを拒絶する審決を作成していることは否めない。

### [3] SFの方程式

(1) SFの方程式とは本願発明者により導出されたSFの全ソースを決定する定義式(方法論)のことである。

(2) SFの全ソースがSFの方程式から求められるのでなければ、SFの動的アルゴリズムに同期性を成立させることは保証できない。

(3) プログラム言語と開発条件が同じであれば、SFの方程式で創出されるソースには一義性が成立する。これはSFのソースにはバグ構文が存在しないことを意味する。

(4) SFの方程式に至らない例えば貴庁の認識レベルとCP世界の所謂論理結合型の経験で同期型プログラムを求めることは不可能である。

(5) CPのソースをSFのソースに機械変換することは可能である。しかし、この逆変換は不可能である。

(6) 上記(3)を知った上で、貴庁の拒絶理由を見れば、この世界がよき指導者に恵まれる場に至れば、この世界はほっておいても、テストレス、機械的生産の世界に進化する。そして、自然とSFの世界に移る。これが同期型プログラムSFの道理である。

(7) 貴庁作成の50頁に及ぶ阻害事象はSFの世界の話ではなく、CPの世界の話である。方法論がないに等しい現世を占めるCP世界の専門家住民にこの違いを理解させるには最低SFの動性アルゴリズムの座学を受けさせる必要がある。

(8) 貴庁が、本願発明者が要請している座学を受講していれば、あるいは貴庁が真摯に明細書の記載について学び、質問をしていれば、そして、CPの世界の経験的専門家がいないければ、貴庁は同期型プログラムとその動性アルゴリズムについて多くを学ぶことができたのである。そして、本願発明者から見れば無責任な審決の作成を行わずに踏みとどまれたかもしれない。

(9) 貴庁が座学を受けていれば、SFの動的アルゴリズムの基でウイルスがどの様に捉えられどのように無力化されるかについても本願発明者から直接解説を受けることができたのである。貴庁は本願発明者に上記のような課題の質問を一切してもいない。唯不埒な拒絶をしているに過ぎない。貴庁が座学でSFの解説を受けていれば、SFの動的アルゴリズムに関する理解が進み、貴庁は言いがかりの拒絶理由を作成するようなことはなかったことだけは確かである。ここで気付かれたかもしれないが、SFの動的アルゴリズムが拒絶事象を自律的に捉える仕組とウイルスを自律的に捉える仕組はSFの動的アルゴリズムの中では同じアルゴリズムである。そして、拒絶事象が自律的に正常化されるアルゴリズムとウイルスが無力化されるアルゴリズムも同じアルゴリズムで処理されている。

(10) 同期型のソースを決定するSFの方程式は本願発明者の50年に及ぶ論考期間を経て2008年にSFの背景理論であるLYEE理論(ソフトウェア構築の摂理に関する統一理論)と共に完成した。LYEE理論は日本をはじめに世界的に特許査定を受けている。

(11) 故に、SFのアルゴリズムに拒絶事象が存在しそれが放置されるようなことは万に一つもない。その拒絶事象はSFの方程式で区分化されるどの区分のアルゴリズムかを明らかにしなければ、その主張は認められない。SFを学んでいる者には貴庁が掲げる拒絶事象には同意することも理解もすることも出来ないだろう。

(12) SFのソースコードを用いれば請求項の成否の判定ができる。

(13) SFとCPの哲学性はほぼ真逆である。

(14) 貴庁は本願発明者が明細書の中で要請している60時間の座学の重要性を斟酌していないことが審判官たちに否定的な理解を成立させた原因である。

(15) 貴庁が拒絶のよりどころのひとつにしているSFの当業者の件はSFの場合、その開発方法論が成立しているので、その座学を受ければ当業者になることができる。他方、CPでは方法論とは経験論に留まるので基本的にはCPの世界には当業者という概念は希薄なのが実情である。

(16) SFの世界では特許法36条は存在の意味がない。

(17) 本願発明者が主導してSFの基盤理論であるLYEE理論(ほぼ世界的に2001年に特許査定済み)を用いて2005年までに36個のシステム開発を済ませている。そして、発明者のこれらの総括は進歩性のないCP技術が世界的に横行している限り、そして、プログラム課題を解法することができない専門家が存在するかぎり、SFの普及は難しいと考え、特許化を優先して、CP世界の自然崩壊を待ち20年後(2025年)に備える考えに転じたのである。

#### [4] 同期型プログラムSFを決定する方法論(SFの方程式)

(1) SFの方程式はSFの全ソースを決定するための開発方法論を意味する。

(2) SFの方程式:  $SG = \Phi 0 [\Phi 4 [\{L 4, \} \{W 4, \} E 4 1, E 4 2, T 4] + [\Phi 2 [\{L 2, \} \{R 2, \} T 2, ] + [\Phi 3 [\{L 3, \} T 3 1, T 3 2, T 3 3]$

(3) SFの方程式は唯ひとつである。これを単元とすれば増殖させることができる。添

付資料を参照。

- (4) SFの方程式にシステム上の違いがあるとすれば、ベクトルの数である。
- (5) SGはSFの解である。添付資料を参照。SGは全ベクトルからグラフ化ツールで求められる。
- (6)  $\Phi 0$  (同期関数) コード例は唯1組である。添付資料を参照。
- (7)  $\Phi 4$ 、 $\Phi 2$ 、 $\Phi 3$ は3種の座標関数、コード例は添付資料を参照。
- (9) ベクトルの作成法は添付資料を参照。
- (8) ベクトルの種別数は以下の12種である。L4、W4、E41、E42、T4、L2、R2、T2、L3、T31、T32、T33
- (10) SFには解(主語系譜: SG)が成立する。
- (11) SFに内包されるベクトルはシステムに属す名詞数ごとに定義される。
- (12) ベクトルには解(主語)が成立する。
- (13) 同期型プログラムSFの構造、ベクトルの定義法の理解には約60時間の座学が必要である。
- (14) 座学のための学習資料は準備されている。添付資料はこの学習要領の資料である。
- (15) SFの方程式で決まるソースには誤記の生じる余地がない。
- (16) SFのソースコードは一義的である。一義的の意味を理解するためには60時間の座学を受ける必要がある。
- (17) 座学を抜きにSFの方程式、ベクトルの定義法、SFの動的アルゴリズムを正確に理解することは出来ない。貴庁の拒絶理由の不備は座学を抜きに作成された結果である。

[5] SFの方程式で決まる全ソースSFが実行時創出するSFの動性アルゴリズムの特性

- (1) 同期型プログラムSFとはSFの方程式で決まる全ソースのことである。そして、SFの動性アルゴリズムとはSFの方程式から創出される全ソースが実行時に創出する動性アルゴリズムの全体である。同期型プログラムSFの動性アルゴリズムはその部分ではなくその全体で成立する。部分では何も成立しないのが同期性の本質である。部分で捉えられる事象、例えば貴庁が言う本願発明とは無関係である拒絶事象は同期型プログラムSFの動性アルゴリズムの世界では虚像である。他方、非同期型CPの動性アルゴリズムの世界ではこの虚像は実在となる。貴庁はこの虚像を用いて本願を否定している。これは動性アルゴリズムの効果の審決を貴庁に求めている本願発明者の意図からすれば誤りである。もし、貴方が上記の全てを理解できないのであれば、貴方は同期型プログラムSFの動性アルゴリズムについて学ばない限り語れる資格はない。同期型の動性アルゴリズムの効果は絶大だがその由来の起点の論考は奥が深い。されど、貴方はそこまで理解しなくても方法論化されているSFの方程式を用いれば、そしてこれをあなたが理解出来れば、貴方は当業者になることができ、且つあなたは当業者が不在でもSFを実用化する指導者になることができる。これはSFの方程式が方法論化されていることの成果である。
- (2) SFの動性アルゴリズムではベクトルがベクトルごとに生成統治する主語と総称される処理情報の正統性の成否が全ベクトルを用いて自律的に判定される。
- (3) SFの動性アルゴリズムでは例えばベクトルAの主語の正統性はこの主語を起点とする他のベクトルの主語との木構造が自律的に求められる仕組みになっている。この木構造を成立させるベクトルは全ベクトルを場として「ベクトル演算」を用いて求められる。そして、ベクトルAの主語を起点とする主語の木構造の端点の主語がベクトルL2の主語、R2の主語で成立すれば、この端点の条件を満たせばこの木構造の起点となる主語は正統であると判定される。もし、端点の主語がこの条件を満たさなければ、起点となる主語は不成立となる。
- (4) SFの動性アルゴリズムで処理される情報処理は全ての主語が正当である場合に実施される。木構造に於ける主語ひとつでも正統でなければSFの動性アルゴリズムは自律



的にそれを捉える。

(5) そして、木構造が正当に成立しなければその木構造はS Fの動性アルゴリズムにより自律的に廃棄される。

(6) S Fの方程式に於ける全ベクトル、3種の座標関数、ひとつの同期関数は上記の自律的処理を果たすための仕組として存在する。

(7) 同期型プログラムを理解しようとするのなら本記述を読み切り理解しなければならない。そうでなければ、同期プログラムについて語ることは出来ない。S Fを利用しようとするのなら本願発明者が座学を要請するのは必然なのである。ウイルスを無力化する仕組、誤り情報を排除する仕組はこの木構造により成立する。

[6] S Fの動性アルゴリズムがウイルスを無力化する仕組

(1) S Fの動性アルゴリズムでは不正情報の処理は実施されない仕組になっている。

(2) S Fの動性アルゴリズムが理解出来れば、S Fの動性アルゴリズムに侵入したウイルスはそこで処理されることがないので、自律的に無力化される。

[7] 全主語に同期性を成立させるために主語間のデータ結合を成立させるベクトル演算添付資料のベクトルに関する項目を参照。

[8] S Fのソースは方法論で導かれるので当業者不在でも作成可能

S Fに多くの当業者はいない。本願発明者は本願発明者によるS Fの方程式の基礎理論(俗称L Y E E理論)の特許査定を2001年に世界的に受けている。本願発明者はL Y E E理論を用いて、他の当業者が不在の中で、受注総額約800億円、36システム、12年間、の開発指導をしている。

それ以前の20年間では従来法のプログラムで、記憶では80システムの開発指導をしている。私の開発指導法は技術のトップランカとして開発実務の先頭に立ち、作業員一人一人の作業実務を指導する方法である。伝統的な開発リーダーではない。このような日常の中で同期型プログラムの方法論の研究が行われてきたのである。故に、貴庁が言う当業者については意味がないことはよく知っている。

海外の特許事情、関係者も私の場合はプログラムの開発方法論に限られるがある程度は知っている。方法論の革新性の議論はどこでも極めて深い議論になるが、当業者の成否、有無、開発実施要件の問題が議論されたことは一度もない。これは開発方法論の利用者が自ら解決すべき問題で、特許が保証する問題ではないからである。

オランダでは20名の審査官と私一人で、私の同行者4人が同席する場で、6時間に及び方法論の革新性について議論したことがある。プログラムの開発方法論の研究者である本願発明者が貴庁に審決を求めているのは同期型プログラムの動性アルゴリズムの効果であるが、その動性アルゴリズムを創出させるソースはS Fの方程式である。

故に、普遍的なS Fの方程式こそが動性アルゴリズムを成立させる方法論として位置づけられるのである。

付言すれば、貴庁が拒絶理由で解決できないという愚問を開発方法論の研究者でもある本願発明者がS Fの方程式の中で放置する筈はないのである。因みに、既に日本で特許査定を受けている「論理結合型プログラムが実行時に発生する非同期型アルゴリズムの正統性の検証を実施可能にするためのプログラム」(特許第6719798号)では同期型プログラムS Fの動性アルゴリズムが用いられている。更に言えば、本願発明者による特許明細書の内容は全て実施検証が行われていることも貴庁に伝えておく。請求項数はそれぞれ異なるが明細書の内容が同等なS Fは日本で2個(特許第5992079号、特許第6086977号)、米国では1個(米国特許第10235522号)、特許されている。

[9] S Fの請求項目の成立証明の方法

(1) S Fは同期型プログラムであるので、S Fを構成する全ソースを用いてS Fの動性の全景図を求めることができる。

- (2) S F のソースの全数は S F の方程式から求められる。
- (3) 上記 (1) の全コードの全景図からコードの実行順序列を求めることができる。
- (4) 本願の請求項の成否は上記 (3) の全実行順序列を用いて判定することができる。

#### [10] 同期性について

「同期性」とは存在の無謬性 (真理) を稼働時に保証するためのシステムのことである。

S F とは名詞の主語の同期性を成立させるための普遍的な構造 (ベクトル) を用いて、稼働時に全主語の非同期性を自律的に監視し、自律的に同期性を維持することを可能にするプログラムのことである。付言すれば、S F は稼働時に S F に内在するバグ構文問題、S F に侵入するウイルス問題を自律的に捉え、且つ自律的に問題を解法しながら同期的に稼働することを可能にするプログラムである。

注1: 本願の親出願 (特願 2015-200064 に) において回答書と共に提出した「シナリオ関数の全景」と E 42 の存在を理解するとよい。

注2: (i) 不敗のアルゴリズム、(ii) 従来プログラムのバグ構文を捉える方法は S F が成立させる同期性の人智を超える応用成果例の一部である。

例えば、生命体、宇宙の同期性をプログラムで捉えれば、その同期性を成立させる要件がそれら存在の存在証明に他ならない。プログラムの完全体を成す S F のベクトルから、プログラムは機能的要件で成立するのではなく、プログラムが対象とする存在の成立証明を要件として成立することが明らかにされたのである。

人類が高速のコンピュータを創作しても、それにより、同期性が捉えられるわけではない。そのわけは工学的産物では同期性は捉えることができないからである。

S F の完全性が示す様に、人類は同期性を成立させるプログラムとコンピュータとで、そのシステムのコンピュータ効果が獲得できるのであり、非同期型のプログラムではコンピュータ効果を獲得することは出来る話ではないのである。即ち、同期を成立させる S F と電算機が合体されるのでなければ、人類はコンピュータの効果を獲得することが出来ないことに気付かなければならない。

非同期型プログラムの自動プログラミングはその機能的要件が不定となることから、不可能であるが、S F では自動プログラミングは可能である。これについて付言すれば、非同期型プログラムではそれを捉える要件が S F の場合の様に確立されないからである。即ち、非同期型プログラムの要件は恒常的に今日においても不定のままなのである。そして、このことが、非同期型プログラムがバグ構文を恒常的に内在させる原因であり、今日においてはウイルスを発症させる理由なのである。このことは、本発明者が膨大な量の非同期型プログラムを精査することにより、明らかにされたのである。そして、このことが、本発明者が 1973 年から同期型プログラムの研究に着手する動機となったのである。

#### [11] 同期世界のプログラム (S F) を実現させるため媒介変数 (要件) を捉える方法

(1) S F の構成項目 (S F の解、同期関数、3種の座標関数、3種のパレット、12種のベクトル) と媒介変数と関係

(2) 現世の媒介変数を捉えるための思考法は既に本能化 (無疑問化) された非同期型思考法になっている。

(3) 非同期世界と同期世界の様相は真逆の関係を成す。

(5) 普遍性を有する定義は非同期世界の証明法を凌駕する。

(4) 非同期世界では普遍的定義が成立させられない (公理主義)。結果、非同期世界では普遍的定義に代わり普遍性が保証されない証明法が重視されている。

(6) 名詞とその主語 (実体化された名詞の様相) は平等に二つの思考世界に共存する。

(7) S F のベクトルは名詞ごと唯一的に定義される。

(8) 名詞に属す主語はベクトルの普遍構造で定義される。

(9) S F のベクトルは 12 種である。

(10) S F に同期を成立させる S F の同期関数は普遍的である。「シナリオ関数の全景

」参照

- (1 1) S Fの3種のパレットを制御する座標関数は「シナリオ関数の全景」参照
- (1 2) S Fの媒介変数は1 2種のベクトルの定義に集約される。
- (1 3) S Fで使用する画面、帳票、ファイル用レコード形式は非同期型の定義法でよい。
- (1 4) S Fが創出する動性アルゴリズムの解説を受けること。
- (1 5) 非同期型プログラムとその要件は一致していない。これが、非同期型プログラムが、自動生成ができない原因である。
- (1 6) S Fのプログラムの自動生成は可能になる。

[1 2] 不敗の根来資料からS Fのベクトルの媒介変数を抽出する方法

- (1) 作業員たちは全員で協議して不敗の根来資料から唯名詞を求めよ。そして、名詞にマーカをつけよ。
- (2) その結果を根来に見せて検収を受けよ。
- (3) 根来は完成した名詞リストの全名詞それぞれに対しベクトル定義する。
- (4) ベクトルを定義するとは名詞ごとに(i)ベクトル種別、(ii)主語の決定、(iii)ベクトルの7種の規約を定義することである。
- (5) L Y E E空間T B Lとは名詞ごとに定義される全名詞のベクトル定義の総攬表のことである。その定義内容は(i)ベクトル識別子、(ii)ベクトルの7種の規約の定義、(iii)名詞領域の定義である。
- (6) 実装されるS FのベクトルのソースコードはL Y E E空間T B Lから専用ツールで自動生成される。
- (7) 自動生成ツールのアルゴリズムはベクトルの7種の規約とS Fの動性アルゴリズムから求積される。
- (8) S Fでは既に記述している様に実行時に同期性が創出される。そして、同期性が生じる過程の中で処理すべきデータが処理され且つこの処理の中で、内在するバグ構文、侵入するウイルスは自律的に排除されて、正統な処理ロジックの完全体が自律的に創出されるのである。これはS Fを構成する(i)ベクトル、(ii)3種の座標関数、そして(iii)同期関数の成果である。非同期型プログラムでは人が所謂要件を経験側で事前に定義し処理ロジックを定義している。そして、このロジックには際限なくプログラムテストを実施しても、その正統性が担保されることはない。このことは、既述の根来特許で証明されている。

[1 3] ベクトルE 4 2について

S Fでは全名詞の部分の主語が成立する場合、ベクトルE 4 1は成立する。しかし、人智では科学がそうであるように全名詞を捉えることが出来ない。故に、全名詞の主語も捉えることは出来ない。S FではこのことをE 4 2で律している。E 4 2は機械物の障害告知にも使用される。

[1 4] 非同期型プログラム（従来プログラム）で構成されるシステムはその自己矛盾からおのずと崩壊に向かう

伝統性I T世界にはソフト技術の専門性を見出すことは出来ない。故に構築されるソフトのシステムはやがて1 0 0年を経ようとしているが非同期型のままである。そして、不完全なW E B世界、A I世界を普及させることに唯ま進し且つその影響を受けている人類は、(i)新新型コロナ、(ii)核戦争、(iii)流星衝突、(iv)銀河系軌道事情、(v)太陽の消滅以前に自律的に破壊されると思われる。

プログラムウイルスが深化すれば、やがて、電算機の機能破壊を発症させることになる。それは少なくとも電算機が人類から葬られてしまうことである。人類はプログラムを同期型に移行しなければ、これら惨禍から逃れることが出来ない。

[15] ベクトルの性質

(1) ベクトルとは識別子となる名詞ごとに定義されるその名詞の主語を捉えるための論理原子である。

(2) ベクトルの変数となる名詞は別のベクトルの識別子となる。

(3) ベクトルの変数となる名詞はその名詞を識別子とするベクトルと融合（連鎖）する

。

(4) 実行時のベクトルは融合して、ロジックを同期的に捉える。

(5) ロジックの真性は同期性以外には、例えば、機能的手段では捉えられない。

(6) ベクトルの識別子の定義規則

(7) ベクトル構造は7種の普遍的規則で成立する。

(8) 従来プログラムのバグ構文を捉えるアルゴリズム、囲碁の不敗のアルゴリズムはS Fのベクトルの応用である。

以上