

平成31年3月27日判決言渡 同日原本領収 裁判所書記官

平成29年(ワ)第31706号 損害賠償請求事件

口頭弁論終結日 平成31年1月28日

判 決

5 当事者の表示 別紙1当事者目録記載のとおり

主 文

- 1 原告の請求を棄却する。
- 2 訴訟費用は原告の負担とする。

事 実 及 び 理 由

10 第1 請求

被告は、原告に対し、3億4915万5000円及びこれに対する平成29年10月18日から支払済みまで年5分の割合による金員を支払え。

第2 事案の概要

1 本件は、発明の名称を「情報管理方法、情報管理プログラム、及び情報管理
15 装置」とする特許第3754438号の特許権（以下「本件特許権」といい、この
特許を「本件特許」という。また、本件特許の願書に添付した明細書及び図面を
「本件明細書等」という。）を有する原告が、被告に対し、被告においてウェブサ
イト上で提供している「Choregraphe」（コレグラフ）という名称のプ
ログラム（以下「被告プログラム」という。）は、本件特許の特許請求の範囲の請
20 求項14記載の発明（以下「本件発明」という。）の技術的範囲に属し、被告によ
る被告プログラムの提供は本件特許権を侵害する旨を主張して、民法709条の不
法行為による損害賠償請求権（対象期間は、平成27年6月1日から平成29年3
月31日まで）に基づき、3億4915万5000円及びこれに対する不法行為後
25 の日である平成29年10月18日（訴状送達の日翌日）から支払済みまでの民
法所定年5分の割合による遅延損害金の支払を求める事案である。

2 前提事実（当事者間に争いが無い又は後掲の証拠及び弁論の全趣旨により容

易に認められる事実)

(1) 当事者

原告は、コンピューターソフトウェアの研究、開発に関する事業等を目的とする株式会社である（甲1）。

5 被告は、コンピューターソフトウェアの研究、開発、販売及び賃貸等を目的とする株式会社である（甲2）。

(2) 本件特許権

原告は、次の内容の本件特許権を有している（甲3，18）。

出願日	平成16年8月13日
10 優先日	平成16年7月28日
登録日	平成17年12月22日
特許番号	特許第3754438号
発明の名称	情報管理方法、情報管理プログラム、及び情報管理装置

15 (3) 本件発明の特許請求の範囲

本件発明の特許請求の範囲の記載は、別紙2の本件特許に係る特許公報（甲3）の該当欄記載のとおりである。

(4) 本件発明の構成要件の分説

20 本件発明は、次のとおり、構成要件に分説される（以下、頭書の記号に従って、「構成要件A」などという。）。

- A コンピュータが情報を管理する情報管理方法であって、
- B 前記コンピュータに複数のノードそれぞれに対応付けて入力された管理すべき情報を、前記ノードを識別するノード識別情報に対応付けられた複数のノードデータを含む文書ファイルとして前記コンピュータが記憶する情報記憶ステップと、
- 25 C 前記情報記憶ステップで記憶された前記文書ファイルの情報を前記コンピュータが表示する情報表示ステップと、

D 前記ノードデータに含まれるスクリプトを前記コンピュータが実行する情報評価ステップとを備え、

E 前記ノードデータは、ルートノードを除いて、当該ノードの親ノードを特定する親ノード識別情報を含んでおり、

5 F 前記スクリプトは、当該ノードデータに含まれる変数データである自ノード変数データと、当該ノードの直系上位ノードのノードデータに含まれる変数データである上位ノード変数データを利用した演算を行って、前記自ノード変数データの値を求める代入用スクリプトを含んでおり、

G 前記情報表示ステップは、前記親ノード識別情報を利用して、前記ノードの木構造を表示する木構造表示ステップと、前記表示された木構造のノードのうちの
10 選択されたノードの前記自ノード変数データ、前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップを含み、

H 前記情報評価ステップは、前記代入用スクリプトの実行により、前記自ノード変数データの値を更新するステップを含む情報管理方法

15 I における各ステップを、コンピュータに実行させるための情報管理プログラム。

(5) 被告プログラム

ア 概要

被告プログラムは、被告が販売する「P e p p e r」及び「N a o」という名称
20 のロボット（以下、併せて「本件ロボット」という。）上で作動するアプリケーション（以下「ロボアプリ」という。）を開発するためのソフトウェアである。ロボアプリにおいては、本件ロボットの動作などの振る舞い（フローともいう。）を規定することになるが、振る舞いを定義する単位がビヘイビアであり、一つのビヘイビアはボックスという名称の機能単位で構成される。例えば「こんにちは」と話す
25 という振る舞いについては、言語を設定するボックス、話すという動作を規定するボックスなどで構成される。被告プログラムでは、このようなボックスを接続して

いくことによってロボアプリを作成することができるようになっており、被告プログラムを起動した際に表示される画面上には、振る舞いを図で表すフローダイアグラムを表示するパネルが設けられ、同パネル内にボックスを配置し、ボックス同士を接続してフローダイアグラムを作成していくことによって振る舞いを構築する。

5 このように構築された振る舞いをプログラミング言語及びその他の情報で記述したものが、「behavior. xar」という名称の文書ファイル（以下、単に「behavior. xar」という。）であり、振る舞いを構築すると、上記文書ファイルが作成される（甲6，7）。

イ 被告プログラムを起動させて作成されたbehavior. xar

10 別紙3-2 behavior. xarの内容は、原告において、被告プログラムを起動し、本件ロボットに「こんにちは」を意味する単語をしゃべらせる振る舞いを構築した際に作成されたbehavior. xarである（甲17）（以下、このbehavior. xarを「本件behavior. xar」という。）。

(6) 被告の行為

15 被告は、遅くとも平成27年9月から、被告プログラムをウェブサイト上で提供している（甲4～6）。

3 争点

(1) 被告プログラムは、本件発明の技術的範囲に属するか（争点1）

20 ア 被告プログラムは、「情報管理方法」（構成要件A）、「ノード」、「管理すべき情報」、「ノードデータ」、「文書ファイル」、「情報記憶ステップ」（構成要件B）、「構成要件C及びDを充足するか（争点1-1）

イ 被告プログラムは、「ノード識別情報」（構成要件B）、「親ノード識別情報」（構成要件E，G）、「ルートノード」（構成要件E）、「木構造」、「木構造表示ステップ」（構成要件G）を充足するか（争点1-2）

25 ウ 被告プログラムは、「（直系）上位ノード」（構成要件F）、「上位ノード変数データ」（構成要件F，G）、「代入用スクリプト」（構成要件F，H）を充

足するか（争点1-3）

エ 被告プログラムは、「自ノード変数データ，前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップ」（構成要件G）を充足するか（争点1-4）

5 オ 被告プログラムは、「更新するステップ」（構成要件H）を充足するか（争点1-5）

(2) 本件特許は特許無効審判により無効とされるべきものか（争点2）

ア 本件発明は本件特許出願前に頒布された刊行物である乙9（特開平6-175852号公報）に記載された発明（以下「乙9発明」という。）により新規性又は進歩性を欠くか（争点2-1）

イ 本件発明は本件特許出願前に頒布された刊行物である乙16（特開平10-69379号公報）に記載された発明（以下「乙16発明」という。）により新規性又は進歩性を欠くか（争点2-2）

ウ 本件特許は特許法36条6項1号及び同条4項1号に違反しているか（争点15 2-3）

エ 本件特許は特許法36条6項2号に違反しているか（争点2-4）

(3) 損害の発生の有無及びその額（争点3）

4 争点に対する当事者の主張

(1) 争点1（被告プログラムは，本件発明の技術的範囲に属するか）について

20 ア 争点1-1（被告プログラムは，「情報管理方法」（構成要件A），「ノード」，「管理すべき情報」，「ノードデータ」，「文書ファイル」，「情報記憶ステップ」（構成要件B），構成要件C及びDを充足するか）について

【原告の主張】

25 (ア) 構成要件B，DないしFの規定によれば，「管理すべき情報」は，ノードを識別するノード識別情報に対応付けられた複数の「ノードデータ」を含む文書ファイルとしてコンピュータに記憶されるものであり，この「ノードデータ」は，

「スクリプト」，「親ノード識別情報」，「自ノード変数データ」，「上位ノード変数データ」及び木構造を表示するための情報を含むものを意味する。

また，「ノード」は，木構造を前提とするものであるところ，「ノード」及び「木」は，数学の一分野であるグラフ理論における概念であって，「木」は，閉路を含まないグラフを意味し，グラフ理論におけるグラフは，ノード（頂点）と2つのノードを結ぶエッジ（辺）により構成される。よって，「ノード」は，ノード（頂点）と2つのノードを結ぶエッジ（辺）により構成される閉路を含まないグラフにおける頂点を意味する。

さらに，本件明細書等（以下，【】は，本件明細書等における発明の詳細な説明の段落番号を指す。【0009】）の記載に鑑みれば，「文書ファイル」とは，テキストエディタ等で読むことのできるテキスト形式のファイルであり，1つの文書データを意味する。

(イ) 被告プログラムは，ボックスを接続する形で本件ロボットのソフトウェアを実装することができる開発環境であり，別紙3-1被告プログラム説明書及び別紙4被告プログラムの構成（原告）記載のとおり，コンピュータによって情報を管理するものであるから，コンピュータが情報を管理する「情報管理方法」を充足する。

また，被告プログラムにおいて表示されるフローダイアグラムは，閉路を含まず，結合線により結合されたボックスで構成されるから，これらのボックスは，「ノード」に該当し，被告プログラムにおいてロボットの動作に関する情報がまとめられた文書ファイルであるbehavior.xar内で，ボックスに対応付けて入力されたスクリプトは，「管理すべき情報」及び「ノードデータ」に該当するから，被告プログラムは，「ノードデータを含む文書ファイル」としてコンピュータが記憶する「情報記憶ステップ」を有する。

さらに，被告プログラムは，文書ファイルであるbehavior.xarを読み込んで表示し，behavior.xarに含まれるスクリプトを実行するステ

ップを備えているから、構成要件C及び構成要件Dを充足する。

【被告の主張】

(ア) 本件発明の目的及び課題は、情報の更新を簡単かつ効率的に行う手段により、多人数で情報を共有し、再利用することができる情報管理方法を提供すること
5 であって（【0007】，【0009】），構成要件Hが「自ノード変数データの
値を更新するステップ」と規定していることにも照らせば、「管理すべき情報」は、
変数データの値を意味する。

よって、「情報管理方法」における情報及び当該情報を含む「文書ファイル」は、
それ自体、多人数で共有し、再利用する価値のある、変数データの値を意味する。

10 また、「複数のノードそれぞれに対応付けて入力された管理すべき情報」（構成
要件B）という文言によれば、「ノード」は、管理すべき情報を含むものであり、
木構造を前提とした概念である。

(イ) 被告プログラム

被告プログラムは、別紙5被告プログラムの構成（被告）記載のとおりであり、
15 被告プログラムが管理するbehavior.xarに含まれるのはロボットを動
作させるためのプログラムコードであって、人間がこの意味を読み取り、多人数で
共有して再利用するものではないから、被告プログラムには「情報」に該当するも
のが存在せず、「情報管理方法」，「管理すべき情報」，「ノード」及び「文書フ
ァイル」を充足しない。

20 また、下記イのとおり、木構造は、階層構造を備えるものであるところ、被告プ
ログラムのフローダイアグラムはプロセスを規定するものであって階層構造を備え
ていないから、被告プログラムにおけるボックスは「ノード」，「ノードデータ」
及び「情報記憶ステップ」を充足しない。

以上によれば、被告プログラムは、「情報記憶ステップ」，「文書ファイル」及
25 び「ノードデータ」を前提とする構成要件C及び構成要件Dも充足しない。

イ 争点1－2（被告プログラムは、「ノード識別情報」（構成要件B），“親

ノード識別情報」(構成要件E, G), 「ルートノード」(構成要件E), 「木構造」, 「木構造表示ステップ」(構成要件G)を充足するか)について

【原告の主張】

(ア) 木構造, 木構造表示ステップ

5 本件発明の背景技術である特許文献の記載, 「ノードの木構造の表示は, ラベルとリードによって行」(【0044】)われること及び「木」はグラフ理論における概念であることに照らせば, 「木構造」は, 閉路を持たないグラフであり, ノード(点)とエッジ(線)から構成される図として表示されるものを意味し, 表示に関する概念である。

10 (イ) ノード識別情報, 親ノード識別情報

本件明細書等の記載(【0042】, 【0049】, 【0054】)によれば, 親子のノード間を接続するリードは階層リードと呼ばれ, 階層リードは親子関係に基づいて表示されるから, 「親」は, 専ら木構造表示におけるノード間のリードの接続に関するものを意味する。

15 また, 構成要件G及び本件明細書等の記載(【0038】, 【0060】)によれば, 木構造は階層ごとに表示されるから, 「親ノード識別情報」は, 階層ごとの木構造表示のために親ノードを識別する機能を有する情報を意味する。

(ウ) 被告プログラム

20 被告プログラムは, 別紙3-1被告プログラム説明書及び別紙4被告プログラムの構成(原告)記載のとおり, 各ボックスに割り振られたidがフローダイアグラムにおいて木構造を表示するために親ボックスを識別する機能を有するから, 「ノード識別情報」及び「親ノード識別情報」を有する。

また, 被告プログラムにおいて, idが-1であるrootボックスは, 「ルートノード」に該当する。

25 さらに, 被告プログラムは, Linkタグ内のinputowner(入力側・子)とoutputowner(出力側・親)により結合情報を認識し, これに従

ってフローダイアグラム上でボックスとボックス間の結合線で表現することによって木構造を表示しているから、親ノード識別情報を利用して木構造を表示しているといえ、「木構造」及び「木構造表示ステップ」を有する。

【被告の主張】

5 (ア) ノード識別情報

本件明細書等の記載（【0029】）によれば、「ノード識別情報」は、ノードを識別する情報であって、ノード生成時に自動的に一意の番号が付与される情報を意味する。

また、特許請求の範囲の請求項2及び9の記載並びに本件明細書等の記載（【0042】，【0049】）によれば、ノードの結合関係を表す参照リードは、親ノード識別情報に基づく階層リードと区別されているから、「ノード識別情報」は、ノードの結合関係を表す情報を含まないものを意味する。

(イ) 親ノード識別情報

構成要件Eは、「親ノード識別情報」は、「ノードデータ」に含まれるものであると規定しているところ、本件明細書等の記載（【0036】，【0038】～【0042】，【図2】～【図4】，【図9】，【図14】）によれば、「ノードデータ」は、ノードを示す特定の開始タグと、そのエンドタグとの間に挟まれた領域内のデータを意味する。よって、「親ノード識別情報」は、ノードを示す特定の開始タグと、そのエンドタグとの間に挟まれた領域内のデータに含まれるものを意味する。

また、本件明細書等の記載（【0010】，【0013】，【0049】，【図6】，【図8】）によれば、「親ノード」は、木構造を前提とした概念であるところ、「木構造」は、基本となるルートノードから複数の要素に枝分かれをした階層構造を意味するから、「親」は、1つの基本となる要素から複数の要素に枝分かれをした階層構造によって規定される親子関係における親を意味する。

(ウ) 被告プログラム

被告プログラムは、別紙5被告プログラムの構成（被告）記載のとおりであり、被告プログラムには、同一のidを有する複数のボックスデータが存在するから、idによってボックスを識別することは不可能である上、idは編集可能であって、ボックス生成時に自動的に一意の番号が付与されるものではないから、被告プログラムは、「ノード識別情報」を充足しない。

また、原告が親ノード識別情報であると主張するLinkタグ内の情報は、ボックス間の結合関係を表すものである上、各ボックスのBoxタグの開始タグとエンドタグとに挟まれた領域の外にあるから「ノードデータ」に当たらない。よって、被告プログラムは、「ノード識別情報」及び「親ノード識別情報」を充足しない。

加えて、被告プログラムのフローダイアグラムは、ボックスの実行順序を定めるものであるところ、Rootボックス、Set Languageボックス、Sayボックス、Rootボックスという順序の処理の流れを定めており、Rootボックスから出てRootボックスに戻る閉路を構成しているから、階層構造は存在しないのであって、親子及び上下という概念は存在しない。よって、被告プログラムは、「親ノード」、「親ノード識別情報」、「木構造」及び「木構造表示ステップ」を充足せず、これを前提とした「ルートノード」も充足しない。

ウ 争点1-3（被告プログラムは、「（直系）上位ノード」（構成要件F）、「上位ノード変数データ」（構成要件F、G）、「代入用スクリプト」（構成要件F、H）を充足するか）について

【原告の主張】

(ア) （直系）上位ノード、上位ノード変数データ

「（直系）上位ノード」とは、属する順序系列内において、当該ノードに先行するノードを意味し、構成要件F及び本件明細書等の記載（【0031】、【0032】）によれば、「上位」は、専ら下位ノードによるデータの承継に関するものを意味する。

また、本件明細書等の記載（【0056】、【0066】）によれば、「変数デ

ータ」は、変数名及び変数の値の2つの要素からなるものであり、変数名のみによって特定することもできるものである。

さらに、本件明細書等の記載（【0007】）によれば、上位ノード変数データが表示されることによって、選択したノードで用いられている上位ノード変数データを容易に把握することができ、管理すべき情報の更新を、簡単かつ効率的に行うことができるのであるから、「上位ノード変数データ」は、当該ノードの直系上位ノードのノードデータに含まれる変数データを意味する。

(イ) 代入用スクリプト

「代入用スクリプト」には、単純な代入処理だけでなく、数式を含んだ代入処理を行うスクリプトも含まれる。

(ウ) 被告プログラム

a 被告プログラムにおいて、別紙3-1被告プログラム説明書（図11, 22）のとおり、Localized Textボックス、Sayボックス及びSet Languageボックスは、属する順序系列内において、Say Textボックスに先行するボックスであるから、「(直系)上位ノード」を有し、その変数名又は変数の値は、「上位ノード変数データ」を充足する。

b 被告プログラムが別紙4被告プログラムの構成(原告)記載の構成f(以下「被告プログラムの構成f」という。)を有するとすれば、被告プログラムのLocalized Textボックスは、直系上位ノードであるSet Languageボックスにおいて入力された、上位ノード変数データである変数langの値である「English」や「Japanese」を利用して演算を行い、Localized Textボックスの自ノード変数データである「Hello」や「こんにちは」を決定しているから、被告プログラムは「代入用スクリプト」を充足する。

c また、被告プログラムが、別紙4被告プログラムの構成(原告)記載の構成f'(以下「被告プログラムの構成f'」という。)を有する場合においても、被

告プログラムの S a y T e x t ボックスは、親からの変数の取得機能を使用する場合、直系上位ノードである S a y ボックスにおいて入力された上位ノード変数データである「S p e e d (%)」や「V o i c e S h a p i n g (%)」の値を利用して演算を行うことにより、S a y T e x t ボックスの自ノード変数である s e n t e n c e に代入し、S a y T e x t ボックスの自ノード変数データである s e n t e n c e の値を更新しているから、被告プログラムは「代入用スクリプト」を充足する。

(エ) 小括

以上のとおり、被告プログラムは、「(直系)上位ノード」、「上位ノード変数データ」及び「代入用スクリプト」を充足する。

【被告の主張】

(ア) ノードデータ，上位ノード変数データ

構成要件Fの文言によれば、上位ノード変数データは、「直系上位ノードのノードデータに含まれる」ものであるところ、上記イのとおり、「ノードデータ」は、ノードを示す特定の開始タグと、そのエンドタグとの間に挟まれた領域内のデータであるから、「上位ノード変数データ」は、直系上位ノードのノードを示す特定の開始タグと、そのエンドタグとの間に挟まれた領域内のデータに含まれる変数データを意味する。

また、上記アのとおり、本件発明は木構造を前提とするものであるから、「直系上位ノード」も木構造を前提とした概念であるところ、「直系」とは、一般に、人と人との間の血統が親子の関係で続いている系統を意味するから、「直系上位ノード」は、木構造を前提として、自ノードと親子関係にあり、自ノードよりもルートノードに近いノードを意味する。

さらに、構成要件Fは、「上位ノード変数データを利用した演算」を行って「自ノード変数データの値を求める」ことを規定しているところ、「上位ノード変数データ」の変数名だけでは、変数名を利用した演算を行って、自ノード変数データの

値を求めることは不可能であり、また、本件明細書等の記載（【0031】，【0032】）にも照らせば、「上位ノード変数データ」は、変数名のみならず、変数の値を意味するか、少なくとも変数の値を含むものと解釈される。

(イ) 代入用スクリプト

5 構成要件Dは、「前記ノードデータに含まれるスクリプト」と規定し、構成要件Fは、この記載を受けて「前記スクリプトは…自ノード変数データの値を求める代入用スクリプトを含んでおり」と規定しているから、「代入用スクリプト」は、自ノードのノードデータに含まれるものでなければならない。

10 また、本件明細書等の実施例（【0072】，【0073】）に照らせば、「代入用スクリプト」は、親ノード変数データの値を自ノード変数データの値として代入するスクリプトを意味する。

(ウ) 被告プログラム

15 a 上記イで主張したとおり、被告プログラムのフローダイアグラムにおける処理の流れは閉路を構成し、親子、上下という概念は存在しないから、被告プログラムは、「(直系)上位ノード」及び「上位ノード変数データ」を利用した演算を行って自ノード変数データの値を求める「代入用スクリプト」を充足しない。

20 b 被告プログラムの構成fについてみると、変数langは、Set Languageボックスに含まれる一時変数であるが、自ノードであるSayボックス内に記載されたスクリプトには利用されておらず、また、langの値はSet Languageボックスに記憶されていない。さらに、被告プログラムでは、Localized Textボックスに予め記載された値である「Hello」や「こんにちは」という値を、Set Languageボックスの変数データを用いて選択しているにすぎず、自ボックスの変数に上位ボックスの変数の値を代入していないから、「代入用スクリプト」を充足しない。

25 c 被告プログラムの構成f'についてみると、被告プログラムのSay Textボックスにおける「Speed (%)」及び「Voice Shaping

(%)」は、いずれも上位ノード変数ではなく、Say Textボックスの自ボックス変数である。また、Say Textボックスが「親から継承する」機能によって「Speed (%)」及び「Voice Shaping (%)」の値を継承する際、Sayボックスはその継承元のボックスであるところ、SayボックスとSay Textボックスは結合情報で結合されていないから、SayボックスはSay Textボックスの上位ノードに該当しない。さらに、「Speed (%)」及び「Voice Shaping (%)」は、直系上位ノードのノードデータに含まれるものではなく、また、上位ノードのノードデータ中に変数の値を含むものではない。

これに加えて、behavior.xarには、「親から継承する」機能を実行するスクリプトは存在しないから、被告プログラムは、自ノードのノードデータに含まれる代入用スクリプトに利用され、代入用スクリプトが親ノード変数データの値を自ノード変数データの値として代入するスクリプトという構成が存在しない。

(エ) 以上によれば、被告プログラムは、「(直系)上位ノード」、「上位ノード変数データ」及び「代入用スクリプト」を充足しない。

エ 争点1-4 (被告プログラムは、「自ノード変数データ、前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップ」(構成要件G)を充足するか)について

【原告の主張】

(ア) 本件明細書等の記載(【0046】)によれば、「デザインテーブル20は、ツリービューア10に表示されたノードのうちの選択されたノードが有する情報を表示する領域であるから、「テーブル」(構成要件G)は情報を表示する領域を意味する。

(イ) 被告プログラム

a スクリプト

被告プログラムのフローダイアグラム画面上でボックスを選択してダブルクリッ

クすると、別紙3-1 被告プログラム説明書（図14）のとおり、当該ボックスに対応して管理されているスクリプトをスクリプトエディタによって表示することができるから、被告プログラムは「スクリプトを表示するノードデータテーブル表示ステップ」を充足する。

5 b 自ノード変数データ

被告プログラムのフローダイアグラム画面上でボックスを選択してクリックすると、別紙3-1 被告プログラム説明書（図15, 16）のとおり、右下に区分けされたインスペクタという名称のウィンドウ（以下「インスペクタ」という。）に、自ノード変数データである当該ボックスの変数名が表示されるから、被告プログラ
10 ムは、「自ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足する。

c 別紙4 被告プログラムの構成（原告）記載の構成g（以下「被告プログラムの構成g」という。）における上位ノード変数データ

被告プログラムのインスペクタには、当該ボックスが有している入力コネクタの変
15 数名も表示されるところ、入力コネクタには、アプリケーションを実行した際に直近の親ボックスから引き渡される値が書き込まれる。また、インスペクタ上のアイコンをクリックすると、当該ボックスのコネクタや変数の追加、名称やタイプ等の編集、削除をすることができる。例えば、被告プログラムでは、S a y T e x t
20 ボックスの変数であるpは、S a y T e x tボックスの入力コネクタ” o n I n p u t _ o n S t a r t () ” という関数の構成要素であり、直系上位ノードであるL o c a l i z e d T e x tボックスの出力コネクタo n S t o p p e d (S e l f , s e n t e n c e s [s D e f a u l t L a n g]) から出力された値を受け取る、コネクタの内部領域である。この入力コネクタの内部領域pには、アプリケーションを実行した際に、L o c a l i z e d T e x tボックスから引き
25 渡される値が書き込まれる。

よって、入力コネクタは、親ボックスから引き渡される値を記憶する内部領域を

保持する変数であるといえるから、入力コネクタの名称は、「上位ノード変数データ」に該当する。したがって、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足する。

5 d 別紙4被告プログラムの構成(原告)記載の構成g'(以下「被告プログラムの構成g'」という。)における上位ノード変数データ

被告プログラムのSay Textボックスのスク립トエディタにおいて親からの変数の取得機能を使う場合、上位ノードであるSayボックスの変数から利用可能なものを一覧表示する機能があるから、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足する。

10 e なお、本件発明において、インスペクタとスク립トエディタを同時に表示する必要はないが、被告プログラムにおいては、スク립トエディタをインスペクタと同時に表示することも可能であるから、被告プログラムは「ノードデータテーブル表示ステップ」を充足する。

(ウ) 小括

15 以上によれば、被告プログラムは、「自ノード変数データ、前記上位ノード変数データ及び前記スク립トを表示するノードデータテーブル表示ステップ」を充足する。

【被告の主張】

20 (ア) 「テーブル」は表を意味することや、本件明細書等の記載(【0065】、【0066】、【0057】、【図6】、【図9】、【図10】、【図13】)によれば、「ノードデータテーブル表示ステップ」は、変数名と変数の値とを表形式で表示するステップを意味し、また、自ノード変数データ、上位ノード変数データ、並びに当該自ノード変数データ及び上位ノード変数データを用いた代入スクリプトを、全て同時に表示するものを意味する。

25 (イ) 被告プログラムの構成g

原告は、インスペクタに表示される入力コネクタの名称が「上位ノード変数デー

タ」に該当すると主張するが、入力コネクタの名称は、コネクタ名であって変数ではない上、変数の値を含まないから、「上位ノード変数データ」に当たらない。また、S a y T e x t ボックスの p は、自ボックス関数の引数であって上位ノード変数ではなく、一時変数であり、その値はS a y T e x t ボックスに記憶されておらず、被告プログラムにおいて p の値を表示することはできないから、「上位ノード変数データ」に当たらない。よって、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足しない。

(ウ) 被告プログラムの構成 g'

被告プログラムにおいて、「親からの変数を取得」機能によって表示されるのは、変数名にとどまり、変数の値は表示されないから、「親からの変数を取得」機能による表示は「上位ノード変数データ」に当たらない。よって、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足しない。

(エ) 小括

以上によれば、被告プログラムは、「自ノード変数データ、前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップ」を充足しない。

オ 争点1-5（被告プログラムは、「更新するステップ」（構成要件H）を充足するか）について

【原告の主張】

「更新」とは、文書ファイルを変更するか否かに関わらず、自ノード変数データの値を更新することを意味し、「更新するステップ」は、このような意味での「更新」を行うステップを意味する。

被告プログラムでは、別紙3-1被告プログラム説明書記載のとおり、S a y T e x t ボックスの自ノード変数である s e n t e n c e の値が、代入用スクリプトの実行によって導かれるところ、s e n t e n c e の値は、直系上位ノードであ

る Say ボックスの「Speed (%)」及び「Voice Shaping (%)」の値の設定や、Set Language で設定した言語、Localized Text で言語に対応して入力されるあいさつ文に応じて更新されるから、被告プログラムは、「更新するステップ」を充足する。

5 以上のとおり、被告プログラムは、構成要件 A ないし H を充足するから、これを前提とする構成要件 I を充足し、本件発明の技術的範囲に属する。

【被告の主張】

構成要件 B, C 及び G 並びに本件明細書等の記載（【0043】、【0064】、
10 【0067】、【0073】、【図10】）に照らせば、「更新ステップ」においては、文書ファイルに含まれる自ノード変数データが変更される必要がある。

しかしながら、被告プログラムにおいては、スクリプトの実行により behavior.xar の値が変更されることはないから、被告プログラムは、「更新するステップ」を充足しない。

15 以上のとおり、被告プログラムは、構成要件 A ないし H を充足しないから、これを前提とする構成要件 I を充足せず、本件発明の技術的範囲に属しない。

(2) 争点 2 (本件特許は特許無効審判により無効とされるべきものか) について
ア 争点 2-1 (本件発明は乙 9 発明により新規性又は進歩性を欠くか) について

【被告の主張】

20 本件発明は、以下のとおり、乙 9 発明と同一であるか、何らかの相違点があるとしても、当業者が出願時の技術常識を用いて容易に想到できたから、新規性又は進歩性を欠き、本件発明に係る本件特許には、特許法 29 条 1 項 3 号又は同条 2 項に違反する無効理由 (同法 123 条 1 項 2 号) がある。

(ア) 新規性

25 a 相違点

本件発明と乙 9 発明には、以下の相違点が一応存在し、その余の点において一致

する。

(a) 本件発明では、情報が文書ファイルとしてコンピュータに記憶されるのに対して、乙9発明では、知識が文書ファイルとしてコンピュータに記憶されるものであるかが明示されていない点（相違点1-1）。

5 (b) 本件発明では、スクリプトがノードデータに含まれているのに対して、乙9発明では、スクリプトがノードデータに含まれているかが明示されていない点（相違点1-2）。

(c) 本件発明では、ノードデータテーブル表示ステップがスクリプトを表示するものであるのに対して、乙9発明では、属性値表示ステップがスクリプトを表示するものであるかが明示されていない点（相違点1-3）。

b 検討

当業者の技術常識を参酌すれば、乙9発明の公報には相違点1-1ないし1-3に係る構成が記載されているに等しく、相違点1ないし3は実質的相違点ではないから、新規性を欠く。

15 (イ) 進歩性

仮に相違点1-1ないし1-3が実質的相違点であるとしても、これらに係る構成を採用することは当業者が周知技術（乙10から15）を適用して適宜なし得る設計事項にすぎないから、進歩性を欠く。

【原告の主張】

20 乙9発明は、本件発明における「情報」と乙9発明における「知識」が異なるなどの理由から、構成要件AないしHに相当する構成を備えていないから、本件発明と同一の発明ということはできず、また、本件発明と乙9発明は、利用場面が異なる上、課題が共通しておらず、その構成も全く異なるものであるから、相違点について容易想到ということもできない。よって、本件発明は乙9により新規性及び進歩性を欠くということもできない。

25 イ 争点2-2（本件発明は乙16発明により新規性又は進歩性を欠くか）につ

いて

【被告の主張】

本件発明は、充足論における原告の主張を前提とすれば、乙16発明と同一であるか、何らかの相違点があるとしても、当業者が出願時の技術常識を用いて容易に
5 想到できたから、新規性又は進歩性を欠き、本件発明に係る本件特許には、特許法
29条1項3号又は同条2項に違反する無効理由（同法123条1項2号）がある。

【原告の主張】

乙16発明には、機能ブロックに含まれるプログラムコードに機能ブロックの
物理名称が含まれる構成が開示されていないなど、本件発明との間に数多くの相違
10 点が存在し、その相違点が既存の技術常識又は設計事項であると認めることはでき
ないから、本件発明は乙16により新規性及び進歩性を欠くということとはできない。

ウ 争点2-3（本件特許は特許法36条6項1号及び同条4項1号に違反して
いるか）について

【被告の主張】

15 本件発明が、文書ファイルの変更を伴わない態様を含むものであるという原告
の主張を前提にすると、多人数で情報を共有し再利用することができる情報管理
方法を提供するという本件発明の課題を解決することができない態様を発明の範
囲に含むことになる。

20 また、本件発明は、代入用スクリプトを用いて情報を更新するものであり、代
入用スクリプト自体を更新することはできないものであるところ、管理すべき情
報にはスクリプトが含まれるという原告の主張を前提にすると、管理すべき情報
の更新を簡単かつ効率的に行うという本件発明の課題を解決することができない
態様を含むことになる。

25 よって、本件特許は、サポート要件（特許法36条6項1号）及び実施可能要
件（特許法36条4項1号）に違反する。

【原告の主張】

争う。

本件発明は、スクリプトを用いて情報を更新することを目的としているが、文書ファイルを書き換えることを目的としていない。また、本件発明においては代入用スクリプトを含む管理すべき情報が更新されるという被告の主張は、その前提自体が誤っている。

エ 争点 2 - 4 (本件特許は特許法 3 6 条 6 項 2 号に違反しているか) について

【被告の主張】

本件発明に規定される「文書ファイル」及び「情報」は、その意味が極めて多義的である上、本件明細書等に定義規定が置かれておらず、発明の外縁が著しく不明確であるから、本件特許は、明確性要件(特許法 3 6 条 6 項 2 号)に違反する。

【原告の主張】

争う。本件明細書等の記載や当業者の技術常識を基礎とすれば、「文書ファイル」及び「情報」は、不明確であるということとはできない。

(3) 争点 3 (損害の発生の有無及びその額) について

【原告の主張】

ア 被告は、被告プログラム及び被告プログラムによって作成したアプリケーションを使用することができる本件ロボットを販売している。

イ 本件ロボットの売上は、被告の売上(平成 2 7 年 4 月 1 日から平成 2 9 年 3 月 3 1 日までの売上の合計は、7 7 億 5 9 0 0 万円)のうちの 6 0 %に相当する金額である。

また、被告プログラムは、ロボット購入の強い動機となっているから、本件ロボットの売上に対する本件特許権の寄与度は 5 0 %である。

さらに、本件におけるライセンス料率は、被告プログラムの態様が本件発明の実施例と極めて似ていることや、同分野のプログラムに関するロイヤルティ料率の相場を考慮すれば、1 5 %を下回らない。

ウ 以上によれば、被告の不法行為によって平成27年6月1日から平成29年3月31日までの間に原告に生じた本件特許権のライセンス料（特許法102条3項）に相当する損害額は、77億5900万円×60%×50%×15%=3億4915万5000円である。

5 エ よって、原告は、被告に対して、民法709条の不法行為による損害賠償請求権に基づいて、3億4915万5000円及びこれに対する不法行為後の日である平成29年10月18日（訴状送達の日翌日）から支払済みまでの民法所定年5分の割合による遅延損害金の支払を求める。

【被告の主張】

10 争う。なお、被告プログラムは、公開以来、誰でも無償でダウンロード可能である。

第3 当裁判所の判断

1 本件発明の意義について

(1) 本件明細書等の発明の詳細な説明の記載

15 本件明細書等の発明の詳細な説明は、別紙2特許公報（甲3）の該当欄記載のとおりであるが、概要、次のとおりである。

ア 技術分野

「【0001】

20 本発明は、コンピュータを用いて情報を管理する情報管理方法、情報管理プログラム、及び情報管理装置に関する。」

イ 背景技術

「【0002】

25 コンピュータを用いて各種情報の管理を行う場合、それぞれの情報を記憶したファイル（文書ファイル、画像ファイル等）を、所定のフォルダに保管することによって行うのが一般的である。しかし、作成したフォルダの構造及びそれぞれのフォルダに保管するファイルの種類等は、任意であってフォルダの作成者に依存するた

め、作成者以外の者が必要な情報に適確にアクセスすることは、必ずしも簡単ではない。すなわち、多数の者が情報を共有化し、再利用できるように、情報管理を行うことは容易ではない。

【0003】

5 特許文献1には、情報の共有化、再利用を効率よく実現することができる文書情報管理システムが記載されている。この文書情報管理システムは、案件（プロジェクト）毎にツリーを作成して表示し、作成した文書ファイルを、表示されたツリーの任意のノードに付随させて、サーバコンピュータに保管するものである。

【0004】

10 また、異なる計算機やアプリケーションで共通に取扱うことができるデータ形式として、XML（Extensible Markup Language）等の構造化文書規格があるが、特許文献2には、このような構造化文書を木構造として捉えて処理する構造化文書処理システムが記載されている。」

ウ 発明が解決しようとする課題

15 「【0005】

しかし、管理される各種情報の更新については、上記した管理システムにおいても、効率化が十分図られているとはいえない。すなわち、木構造のノードに含まれる情報は、相互に関連するものが多いが、上記した管理システムにおいては、それぞれの文書の該当する部分を個別に更新する必要がある、十分効率的とはいえない。」

20 「【0007】

本発明は、上記事情に鑑みなされたもので、管理すべき情報の更新を、簡単かつ効率的に行うことができる情報管理情報を提供することを目的とする。」

25 なお、弁論の全趣旨によれば、「情報管理情報」は、「情報管理方法」の誤記と認められる。

エ 課題を解決するための手段

「【0009】

本発明によれば、利用者が入力したデータに含まれるスクリプトを利用して、ノードデータを更新することができるので、管理すべき情報の更新を、簡単かつ効率的に行うことができる。また、複数のノードデータを含む1つの文書データを用いて、個々の業務や案件に関する情報を管理しているので、多数の利用者による情報の共有化、再利用を、簡単かつ効率的に行うことができるとともに、文書データに基づいて、簡単にノードの木構造を表示させることができ、業務や案件全体の把握を簡単に行うことができる。さらに、表示された木構造の個々のノードに対応付けられた詳細情報を簡単に表示することができる。」

オ 発明の効果

「【0024】

以上の説明から明らかなように、本発明によれば、管理すべき情報の更新を、簡単かつ効率的に行うことができる。」

カ 発明を実施するための最良の形態

「【0028】

図1に、ノードデータとして記憶される情報の一例を示す。記憶される情報は、ノード番号、ページ番号、親ノード番号、ノードラベル、ノード表示属性情報、変数情報、代入用スクリプト、生成用スクリプト、リンク情報を含む。

【0029】

ノード番号は、ノードを識別する情報であり、ノード生成時に自動的に一意の番号が付与される。ページ番号は、文書に含まれるノードを複数の木構造として表示するためのもので、そのノードが所属するページを識別する所属ページ番号に、そのノードが別のページを形成する場合にそのページを識別する自己ページ番号を含む。したがって、両方のページ番号が記憶されているノードは、2つのページに属することになる。親ノード番号は、そのノードの親ノードを識別する番号であり、ノード生成時に親ノードを指定することにより、その指定された親ノードのノード

番号が自動的に記憶される。」

「【0031】

変数情報は、各ノードが保持するデータであって、変数名に対応させて記憶される。記憶される変数は、下位ノードから参照される公開変数と、自ノード内でのみ使用する限定変数を含む。また、変数の値（「変数データ」と記述する場合もある。）は、固定値が設定されても、スクリプトの実行によって演算された値が設定されてもよい。また、URLが設定されてもよい。どのような値が設定されるかは任意である。

【0032】

代入用スクリプトは、自ノードの変数の値を演算するためのものである。代入用スクリプトは、自ノードの変数の値である自ノード変数データと、そのノードの直系上位ノードの公開変数の値である上位ノード変数データを利用して記述することが可能である。」

「【0034】

なお、代入用スクリプト及び生成用スクリプトに使用する言語としては、スクリプト言語として使用されている任意の言語を使用することができる。

【0035】

リンク情報は、各ノードにリンクするファイルに関する情報である。スタンドアロン型のコンピュータで実施する場合、この情報はリンクファイルのインデックス情報である。また、クライアント-サーバ型のコンピュータで実施する場合、リンクファイルをサーバに転送後、インデックス情報を作成し、記憶する。リンク情報を記憶することにより、各ノードをフォルダとして利用することが可能となる。

【0036】

ノードデータは、例えばタグ付き文書情報として記憶される。図2に、ノードデータの一例を示す。図2のデータは、ルートノードのノードデータの例であり、ノード番号 (nodeNo) が「3450」、自己ページ番号 (ownPageNo) が「10」、ノード

ラベル(label)が「パッセル操作マニュアル」である。所属ページ番号を示す (belongPageNo) が「0」, 親ノード番号を示す (parentNodeNo) が「0」であることで, ルートノードであることを示している。図 2 の「x=" 100" 」から「color=" 0" 」までは, ノードの表示位置等のノード表示属性情報である。

5 **【0037】**

この形式では, 変数情報が, 「<DataDivision>」と「</DataDivision>」の間に挿入され, 代入用スクリプトが, 「<ProcesureDivision>」と「</ProcesureDivision>」の間に挿入され, 生成用スクリプトが, 「<GenerateDivision>」と「</GenerateDivision>」の間に, リンク情報が, 10 「<LinkageDivision>」と「</LinkageDivision>」の間に挿入される。ただし, 図 2 の例では, 変数情報, 代入用スクリプト, 生成用スクリプトは, 記憶されていない。

【0038】

図 3 に, ノードデータの別の例を示す。図 3 のデータは, ルートノード以外のノードデータの例である。所属ページ番号が「3484」, 親ノード番号が「3488」となっており, ルートノード以外のノードのノードデータであることが把握できる。また, 自己ページ番号が「3526」となっていることから, 別ページの木構造の先頭ノードであることも把握できる。

【0039】

20 図 4 に, 管理すべきデータを複数のノードデータを含む文書情報として記憶させたものの一例を示す。図 4 の文書は, ヘッダ部 4 0, ノードデータ部 4 1 a ~ 4 1 n, ライン部 4 2, レポート部 4 3 を備える。」

【0041】

 ノードデータ部 4 1 a は, ルートノードのノードデータを示し, ノードデータ部 25 4 1 b ~ 4 1 n は, ルートノード以外のノードのノードデータである。

【0042】

ライン部 4 2 は、ノード間を接続するリードを定義する情報が記憶される領域である。ノード間を接続するリードは、親子のノード間を接続する階層リードと、階層関係とは無関係に一時的に変数を参照する参照元ノードと参照先ノード間を接続する参照リードがあるが、ライン部 4 2 は、参照リードの存在及び、位置、表示属性等を規定する。

【 0 0 4 3 】

次に、記憶された文書情報の表示について説明する。図 5 に、文書情報の表示を行う場合の概略動作フローを示し、図 4 に示す文書の表示を行った場合の表示画面の例を図 6 に示す。

【 0 0 4 4 】

図 6 の表示画面は、ツリービューア 1 0 とデザインテーブル 2 0 を有する。ツリービューア 1 0 は、ノードの木構造を表示する領域であり、情報管理時の各種操作を行うためのプルダウンメニュー、及びポップアップメニューを表示する領域も兼ねる。ノードの木構造の表示は、ラベルとリードの表示によって行い、図 6 の例では、ルートノードのラベル表示 1 1 a とルートノード以外のノードのラベル表示 1 1 b, 1 1 c, 1 1 d と、それらの間を接続する階層リード 1 2 b, 1 2 c, 1 2 d が表示されている。」

「【 0 0 4 6 】

デザインテーブル 2 0 は、ツリービューア 1 0 に表示されたノードのうちの選択されたノードが有する情報を表示する領域であり、公開変数表示領域 2 1, 限定変数表示領域 2 2, 代入用スクリプト表示領域 2 3, 生成用スクリプト表示領域 2 4, 操作ボタン表示領域 2 1 a, 2 2 a, 2 0 a を有する。操作ボタン表示領域 2 1 a の各操作ボタンは、公開変数に対する各種操作を行うためのものであり、操作ボタン表示領域 2 2 a の各操作ボタンは、限定変数に対する各種操作を行うためのものである。また、操作ボタン表示領域 2 0 a の各操作ボタンは、デザインテーブル 2 0 に関する各種操作を行うためのものである。」

「【0049】

次いで、ステップS103で認識したノードのラベル表示を行う（ステップS104）。ラベル表示は、そのノードのノードラベル及びノード表示属性情報に基づいて表示する。そして、表示したノードの親子関係に基づいて階層リードを表示し、
5 さらに、文書のライン部42の情報を参考にして、参照リードを表示する（ステップS105）。

【0050】

この状態では、ツリービューア10に木構造が表示された状態である。ステップS106では、ツリービューア10に表示されたノードが選択されたかどうかを判定し、
10 選択されている場合、デザインテーブル20の表示を行い（ステップS107）、そのノードの変数、スクリプト等を表示する（ステップS108）。図6は、ルートノードを選択した場合の例であり、ルートノードには、変数情報等が記憶されていないので、デザインテーブル20に、データの表示はない。この状態で、別のノードを選択すると、そのノードの変数等が表示される。」

「【0054】

図6に示した状態で、表示された木構造及びノードデータの編集が可能であり、編集操作に対応した表示を行う（ステップS109）。ツリービューア10上では、ノードの追加、削除、表示位置移動、表示属性変更、ノードラベル変更等を、プル
20 ダウンメニュー、ポップアップメニューの設定により行う。例えば、表示位置の変更は、ラベル表示をドラッグすることによって行い、ノードラベル及び表示属性の変更は、変更用のウィンドウを表示させて行う。また、ノードの削除は、削除したいノードを選択した状態で、メニューを表示させて削除する。ノードの追加は、メニューで追加モードに設定後、追加したいノードの親となるノードを選択し、そのままドラッグすることにより、新規ノードを生成する。また、ノードの繋ぎ換えは、
25 繋ぎ換えたいノードを選択し、メニューを表示させて「ノード繋ぎ換え」を選択し、変更したい繋ぎ先のノード（親ノードとしたいノード）を選択することによって行

う。生成されたノードのノードラベル，表示属性情報は，修正用のウィンドウを表示させて設定する。

【0055】

それぞれのノードに関するデータは，ノードデータとして1まとまりになっているので，これらの編集を行った場合でも，編集を行ったノードのノードデータに反映させるだけでよく，軽い処理負担で編集作業を行うことができる。」

「【0057】

代入用スクリプト及び生成用スクリプトは，公開変数領域2 1および限定変数領域2 2に表示された変数を利用して作成する。公開変数領域2 1には，自ノードの公開変数だけでなく，直系上位のすべてのノードの公開変数が表示される。直系上位のノード以外のノードの変数を参照したい場合は，参照リードを生成して，参照先のノードと関連付けておく。代入用スクリプト及び生成用スクリプトは代入用スクリプト領域2 3及び生成用スクリプト領域2 4に直接入力してもよいし，別のウィンドウを開いて入力するようにしてもよい。

【0058】

なお，デザインテーブル2 0の編集内容は，領域2 0 aの更新ボタンを押すことによって，文書情報に反映される。」

「【0063】

続いて，ノードデータに含まれる変数情報，代入用スクリプト，生成用スクリプトについて，部品管理を例に説明する。図8は，部品管理に適用した場合の木構造の一例を示す図である。図8の木構造は，製品番号「990070」，品名「MW-70」の製品の部品管理に適用したもので，製品は，「fore」，「center」，「bone」，「back」の部分からなり，さらにそれぞれの部分が，複数の部品から構成されることを示している。ただし，図8では，部分「fore」の部品のみを記載し，他の部分については，省略してある。

【0064】

図8から明らかなように、部分「fore」は、3つの「MW70巾木(表)」と3つの「MW70パネル(表)」から構成される。図9に、1つの部品「MW70巾木(表)」に対応するノード(図8では、便宜的に右肩に「*」を付してある。)のノードデータの一部を示し、図10に、そのノードが選択された場合の公開変数表示領域21の表示例を…示す。

【0065】

公開変数表示領域に表示される公開変数は、自ノードの公開変数51と、直系上位ノードの公開変数52を含み、直系上位のノードの公開変数52は、自ノードの公開変数51と異なる色で表示される(図10では、フォントを変えて示してある。)。また、公開変数には、固定値が入力される公開変数と、代入用スクリプトの実行によって計算される公開変数があり、修飾領域に「なし」あるいは「要計算」を表示することによりに区別される。

【0066】

要計算の公開変数の値は、後述するように代入用スクリプトが実行されるまでは空欄であり、図9及び図10は、代入用スクリプト実行前の状態を示している。なお、直系上位ノードに要計算の公開変数が含まれ、その公開変数の値が代入スクリプトの実行前で定まっていない場合、その公開変数は、下位ノードの公開変数領域21に表示されない。すなわち、他のノードからの参照が一時停止される。

【0067】

代入用スクリプト及び生成用スクリプトは、操作ボタン表示領域20aの「評価」ボタンを押すことにより実行される。」

「【0072】

次に、代入用スクリプト及び生成用スクリプトの具体例を、図8の「*」を付したノードをターゲットノードとして説明する。図9に示すように、ターゲットノードは、要計算の公開変数として、「スライス数」と「色」を有しており、代入用スクリプトとして、「スライス数=同一面数;」と「色=同一面数」を有している。

評価前は、図10に示すように、公開変数「スライス数」と「色」の値は空欄となっている。

【0073】

この状態で、このノードを選択し、「評価ボタン」を押し、評価条件として代入用スクリプトの実行を設定すると、記憶された代入用スクリプトを実行する。したがって、公開変数「スライス数」の値は、上位ノードの公開変数である「同一面数」の値「1」となり、公開変数「色」の値は、同様に上位ノードの公開変数である「巾木色」の値「F-205」となる。代入用スクリプト実行後のデザインテーブルの公開変数表示領域21の表示例を、図13に示す。

【0074】

ターゲットノードは、生成用スクリプトも有しているので、評価ボタンを押し、評価条件として生成用スクリプトの実行を設定すると、条件文によって選択された辞書ノードを参照し、新規ノードを生成する。この例では、辞書コード「C07000400000000001」の辞書ノードが参照され、ノードラベル「巾木(S T)」のノードが生成される。」

「【0076】

図14に示す辞書ノードのノードデータは、ノード番号及びノード表示属性情報位置を含んでいるが、ノード番号は、生成後のノードにおいては書き換えられる。また、ノード表示属性情報のうち、表示位置に関する情報は、予め生成元のノードの表示位置に対する相対位置を示すものとして利用してもよいし、生成時に生成元のノードの表示位置に応じて生成するようにしてもよい。」

(2) 本件発明の概要

以上の本件明細書等の発明の詳細な説明の記載並びに本件特許の請求項1及び14の記載によれば、本件発明の概要は、次のとおりであると認められる。

ア 本件発明は、コンピュータを用いて情報を管理する情報管理方法、情報管理プログラム、及び情報管理装置に関する（【0001】）。

イ 従来、コンピュータを用いて各種情報の管理を行う場合に、多数の者が情報を効率よく共有化及び再利用できるシステムとして、木構造を用いた情報管理システムが存在した（【0002】～【0004】）が、情報の更新を行う場合には、それぞれの文書の該当箇所を個別に更新する必要があったため、相互に関連する情報の更新を効率的に行うことが課題であった（【0005】）ところ、本件発明は、
5 代入用スクリプトを利用してデータを更新する構成を採用し（【0009】，【0032】），管理すべき情報の更新を，簡単かつ効率的に行うことができる（【0024】）ようにしたものである。

2 争点1（被告プログラムは，本件発明の技術的範囲に属するか）について

10 (1) 争点1－2（被告プログラムは，「ノード識別情報」（構成要件B），「親ノード識別情報」（構成要件E，G），「ルートノード」（構成要件E），「木構造」，「木構造表示ステップ」（構成要件G）を充足するか）について

ア 「木構造」及び「親ノード」の意義

15 「ノード識別情報」，「親ノード識別情報」，「ルートノード」，「木構造」及び「木構造表示ステップ」の各文言のうち，まず，「木構造」及び「親ノード識別情報」における「親ノード」の意義について検討する。

「木構造」，「親ノード」との文言は，文言のみから一義的にその意義を明らかにすることができないことから，本件明細書等の発明の詳細な説明の内容を考慮して検討する必要があるところ，まず，「木構造」についてみると，本件明細書等の
20 記載（【0042】，【0044】，【図6】，【図8】，【図16】），証拠（甲14）及び弁論の全趣旨によれば，「木構造」は，ノードを表示するラベルとラベル間を接続する結合線であるリードから構成される図として表現される表示に関する概念であって，基本となる要素，すなわちルートから複数の要素に枝分かれをした階層構造を意味し，閉路を含まないものと解するのが一般的かつ合理的である。また，「親ノード」についてみると，本件明細書等の「親子のノード間を接続
25 する階層リード」（【0042】），「ノードの親子関係に基づいて階層リードを

表示」(【0049】)及び「繋ぎ先のノード(親ノードとしたいノード)」
(【0054】)という記載を踏まえれば、「親」は、木構造の表示におけるノ
ド間のリードの接続に関する概念であり、「親ノード」は、あるノードに対して、
当該ノードが属する階層内において当該ノードに直近して先行するノードを意味す
ると解するのが自然かつ合理的である。「木構造」の意義に関して本件明細書等に
5 これに反する記載は見当たらず、その他、同文言の意義を示す証拠もない。

そして、本件明細書等の記載(【0042】)によれば、本件発明では、一時的
に変数を参照する参照元ノードと参照先ノードを含み得るものであり、この参照元
ノードと参照先ノードの関係は、参照元ノードから出発して参照先ノードを
10 参照元ノードに戻る閉路を示すものと推認されるが、これらは、階層関係とは無関
係であることが示されている。このことと、前記の「木構造」及び「親ノード」の
意義についての検討を踏まえれば、階層リードで接続され、ノードの親子関係が示
されている部分は木構造であることが前提とされていると認められる。

イ 被告プログラムについての事実認定

15 前記第2, 2(前提事実)(5), 証拠(甲7, 17, 22)及び弁論の全趣旨によ
れば、被告プログラムについて、次の事実が認められる。

(ア) 被告プログラムを起動して本件ロボットの振る舞いを構築する際に用いる
ボックスは、ボックスとして囲まれている範囲内に「S a y」などとしてボックス
の名前が表示され、また、左側に、シグナルやデータを受け取るための入力コネク
20 タが設けられ、右側に、ボックスからのデータや終了を示すシグナルを出力するた
めの出力コネクタが設けられており、これらの、ボックスの名前、入力コネクタ又
は出力コネクタは、ボックスの構成要素である。入力コネクタ又は出力コネクタに
は、複数の種類があるが、「o n S t a r t」と呼ばれる入力コネクタは、ここに
シグナルが送られると、ボックスが開始状態になることを意味するコネクタであり
25 (以下、この入力コネクタを「o n S t a r tコネクタ」という。), 「o n S t
o p p e d」と呼ばれる出力コネクタは、ここからシグナルが送られた場合、ボッ

クスが停止したことを意味するコネクタである（以下、この出力コネクタを「on Stoppedコネクタ」という。）。

(イ) 本件ロボットに「こんにちは」を意味する単語をしゃべらせる振る舞いを構築した際に作成されたbehavior.xarが本件behavior.xarであり、この内容をフローダイアグラムとして示したものが別紙6フローダイアグラム目録記載の図1ないし図3（以下、「別紙6の図1」などという。）である。別紙6の図1及び図2に示された行番号やコネクタの説明は、本件behavior.xarの行番号ないしボックスの名前を「Say」とするボックス（以下、個々のボックスについて、ボックスの名前を付して「Sayボックス」などという。）のonStartコネクタ及びonStoppedコネクタを示している。

本件behavior.xarで記述されるSayボックスは、別紙6の図1のとおり位置付けられているところ、SayボックスのonStartコネクタより後のフローダイアグラムは、別紙6の図2のとおりであり、リードで接続される順に、①SayボックスのonStartコネクタ、②Localized Textボックス、③Say Textボックス、④SayボックスのonStoppedコネクタとなっている。

ウ 検討

そこで、被告プログラムが「木構造」を有するか、すなわち、被告プログラムを使用して表示されるフローダイアグラムの親子関係が示されている部分が「木構造」であるかについて検討する。

原告は、前記イ(イ)の①から④までのSayボックスの接続関係について、木構造、すなわち階層リードで接続され、ノードの親子関係が示されている部分であると主張するのでこれをみると、前記イ(イ)のとおり、Sayボックスについて、SayボックスのonStartコネクタから出発して、SayボックスのonStoppedコネクタに接続されているのであり、SayボックスのonStartコネクタ及びonStoppedコネクタは、いずれも、Sayボックスの構成要

素である以上、S a yボックスのフローダイアグラムにおけるボックスの接続関係は、S a yボックスから出発してS a yボックスに戻る閉路として表示されていることになり、木構造であるとはいえない。

5 その他、階層リードで接続され、ノードの親子関係が示されている部分が全て木構造であることを認めるに足りる証拠もない。

そうであれば、被告プログラムは、「木構造」を有しているとはいえず、したがって、「木構造表示ステップ」（構成要件G）を充足しないというべきである。

エ(ア) この点、原告は、「木構造」の意義について、ノード（点）とエッジ（線）から構成される図として表示されるものであって、閉路を含まない概念であるとした上で、前記イ(イ)でみたS a yボックスの構成は、閉路ではないと主張する。すなわち、被告プログラムのS a yボックスのフローダイアグラムにおいて、
③S a y T e x tボックスの出力コネクタから①S a yボックスの入力コネクタに直接リードが接続されている場合には、S a yボックスからS a yボックスに戻る閉路であるといえるが、③S a y T e x tボックスの出力コネクタは、④S a
15 yボックスの出力コネクタに接続されており、①S a yボックスの入力コネクタと④S a yボックスの出力コネクタは異なるものとして表示されているのであるから閉路ではない旨主張する。

しかしながら、「木構造」はコネクタの接続関係ではなく、ノード間の接続関係を表示するものであり、被告プログラムにおいて、それはボックス間の接続関係を表示するものであるところ、別紙6の図2は、別紙6の図1に表示されたフローダイアグラムのうち、S a yボックスの構成要素を表示した図であって、前記認定のとおり、S a yボックスのo n S t a r tコネクタとS a yボックスのo n S t o
20 p p e dコネクタはいずれもS a yボックスの構成要素であるから、S a yボックスのo n S t a r tコネクタとS a yボックスのo n S t o p p e dコネクタの表示位置が離れているとしても、S a yボックスから出発してS a yボックスへ戻る
25 接続関係がないとみることはできない。よって、原告の上記主張はその前提を欠き、

採用することができない。

(イ) また、原告は、出力コネクタである onStopped は、ボックスの動作が終了したことを示すにすぎず、SayText ボックスの onStopped コネクタから出力されたデータは、Say ボックスを経由して流れることはないから、Say ボックスのフローダイアグラムは、データの流れの観点からみても閉路ではない旨主張する。しかしながら、証拠（乙30，31）及び弁論の全趣旨によれば、SayText ボックスの onStopped コネクタから出力されたデータは、Say ボックスを経由していることが認められるから、原告の主張はその前提を欠き、採用できない。

オ 小括

以上のとおり、被告プログラムは、「木構造表示ステップ」（構成要件G）を充足しない。

(2) 争点1－4（被告プログラムは、「自ノード変数データ、前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップ」（構成要件G）を充足するか）について

ア 「ノードデータテーブル表示ステップ」及び「ノード変数データ」の意義について

まず、「ノードデータテーブル表示ステップ」の意義について検討すると、「テーブル」は、表、一覧表を意味するところ、本件明細書等（【0046】，【0055】，【0057】，【0065】，【0066】，【図6】，【図10】）において、「ノードデータテーブル」に相当するデザインテーブルは、自ノード変数データ及び全ての直系上位ノード変数データを表示する領域（【図6】における公開変数表示領域）と、代入用スクリプトを表示する領域を含む一覧表になっており、「図6に示した状態で、表示された木構造及びノードデータの編集が可能であり」（【0054】），「ノードデータとして1まとまりになっている」（【0055】）と記載されていることにも照らせば、「ノードデータテーブル」とは、「ノ

ードデータ」の一覧表であり、上位ノード変数データ、自ノード変数データ及び代入用スクリプトを同時に表示するものと解するのが一般的かつ自然である。

次に、ノードデータテーブルが表示する「ノード変数データ」の意義について検討すると、本件明細書等には、「変数の値（「変数データ」と記述する場合もある。）」（【0031】）、「ノードの直系上位ノードの公開変数の値である上位ノード変数データ」（【0032】）と記載されており、これと異なる解釈を導くような説明がされていることは認められないから、「ノード変数データ」は、変数の値を意味すると解するのが自然かつ合理的である。

イ この点、原告は、「テーブル」の意義について、本件明細書等に「デザイン
10 テーブル20は、ツリービューア10に表示されたノードのうちの選択されたノードが有する情報を表示する領域であり」（【0046】）と記載されているから、「テーブル」（構成要件G）は、情報を表示する領域を意味すると主張する。しかしながら、この記載はデザインテーブルの性質を説明するものにすぎず、「テーブル」の意義を一般的意味より広く解釈すべきことを示唆する記載とみることはできないから、原告の同主張は採用することができない。
15

また、原告は、「ノード変数データ」の意義について、本件特許の請求項1及び請求項9並びに本件明細書等の記載（【0008】【0017】）には、「前記自ノード変数データの値」という文言があり、「変数データ」は、「変数データの値」と区別して用いられているから、「ノードデータテーブル表示ステップ」において、変数の値を表示することは必要ではなく、また、上位ノード変数データと自
20 ノード変数データとを同時に表示することも必要ではないと主張するが、同主張は、前記認定に照らして採用することができない。

ウ 被告プログラム

(ア) 被告プログラムの構成g

まず、原告は、被告プログラムのフローダイアグラム画面上のインスペクタに表示された入力コネクタの名称が「上位ノード変数データ」に当たると主張している
25

ところ、入力コネクタの名称は変数の値ではないから、「上位ノード変数データ」に当たると認めることはできない。よって、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足しない。

(4) 被告プログラムの構成 g'

5 また、原告は、被告プログラムの S a y T e x t ボックスのスク립トエディタにおいて親からの変数の取得機能を使う場合、上位ノードである S a y ボックスの変数のうち利用可能なものを一覧表示させることができる機能があるから、被告プログラムは、「上位ノード変数データ」「を表示するノードデータテーブル表示ステップ」を充足すると主張する。

10 この点、S a y T e x t ボックスにおいて親からの承継を選択した場合、別紙 3-1 被告プログラム説明書図 19 のとおり、インスペクタ上に、S a y T e x t ボックスの変数 S p e e d (%) の値が表示されるが、これは S a y T e x t ボックスにおいて表示されるものであるから自ノード変数を表示しているものと認められ、「上位ノード変数データ」を表示しているとみることはできない。よって、
15 被告プログラムは、一覧表として「自ノード変数データ」及び「上位ノード変数データ」を同時に表示しているということとはできない。

さらに、原告は、別紙 6 の図 3 のように、上位ノード変数と代入用スク립トを同時に表示することができる旨主張するが、同図の表示形態を一覧表とみることはできない上、同図では、上位ノードの名称が表示されているにとどまり、上位ノード
20 変数の値が表示されていると認めることはできないから、「ノード変数データ」を一覧表として表示しているということとはできず、原告の同主張は採用することができない。

加えて、本件全証拠によっても、b e h a v i o r . x a r 内に、親からの承継の機能に関して、自ノード変数データ及び上位ノード変数データを利用した演算を行って自ノード変数データの値を求める「代入用スク립ト」があると認めるに足
25 りる証拠はないから、被告プログラムは、「前記スク립トを表示するノードデー

テーブル表示ステップ」を充足すると認めることはできない。

エ 以上のとおり，被告プログラムは，「自ノード変数データ，前記上位ノード変数データ及び前記スクリプトを表示するノードデータテーブル表示ステップ」（構成要件G）を充足しない。

5 (3) 小括

したがって，その余の点について判断するまでもなく，被告プログラムは本件発明の技術的範囲に属すると認めることはできない。

3 結論

10 以上によれば，その余の争点につき検討するまでもなく，原告の請求には理由がないから，これを棄却することとして，主文のとおり判決する。

東京地方裁判所民事第29部

裁判長裁判官

15

山 田 真 紀

裁判官

20

伊 藤 清 隆

裁判官

25

棚 橋 知 子

別紙一覧

別紙1 当事者目録

別紙2 特許公報 省略（42～59頁）

5 別紙3-1 被告プログラム説明書

図1～26 別添（75～104頁）

別紙3-2 behavior. xarの内容 別添（105～109頁）

別紙4 被告プログラムの構成（原告）

別紙5 被告プログラムの構成（被告）

10 別紙6 フローダイアグラム目録 別添（114～116頁）

(別紙1)

当事者目録

	原	告	株式会社パッセルインテグレーション
5	同	訴訟代理人弁護士	中 村 隆 夫
	同		加 藤 伸 樹
	同		我 妻 崇 明
	被	告	ソフトバンクロボティクス株式会社
10	同	訴訟代理人弁護士	鮫 島 正 洋
	同		和 田 祐 造
	同		森 下 梓