



(12) 发明专利申请

(10) 申请公布号 CN 101710275 A

(43) 申请公布日 2010.05.19

(21) 申请号 200910168268.5

(51) Int. Cl.

(22) 申请日 2009.08.20

G06F 9/44 (2006.01)

G06Q 10/00 (2006.01)

(30) 优先权数据

2,639,000 2008.08.20 CA

12/196,069 2008.08.21 US

(71) 申请人 皆能科研有限公司

地址 加拿大不列颠哥伦比亚省

(72) 发明人 雷伟俊 劳沛荣 胡焯业 邬重富

帕特里克-阿兰·约瑟夫·纽曼菲尔

李淑仪

(74) 专利代理机构 北京集佳知识产权代理有限公司

11227

代理人 朱胜 陈炜

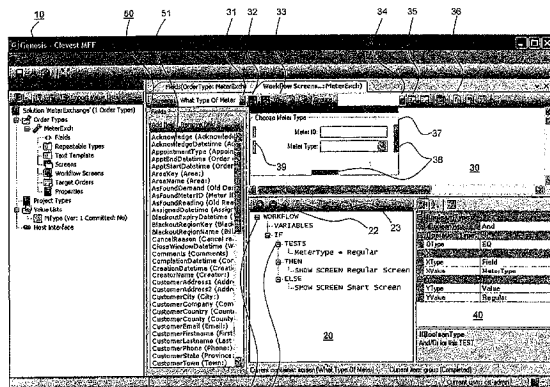
权利要求书 2 页 说明书 8 页 附图 21 页

(54) 发明名称

利用 GUI 工具来编辑 workflow 逻辑和屏幕的方法和系统

(57) 摘要

本发明公开了一种利用图形用户界面 (GUI) 工具来编辑 workflow 逻辑和屏幕的方法和系统。Workflow 管理系统提供 GUI 工具,其允许用户创建和/或编辑 workflow 逻辑而无需重新编写和重新编译应用软件,也不会任何时间中断应用软件的操作性。在用户创建和/或编辑 workflow 逻辑之时,GUI 工具允许用户同时查看和创建和/或编辑相应的应用程序屏幕,其中 workflow 逻辑应用于应用软件中。用户通过有选择地组合上下文有关菜单用口语语法提供的可用参数来创建和/或编辑 workflow 逻辑。Workflow 逻辑语句变得可由应用软件用来在适当时间调用。GUI 工具与动态可配置的数据库协同工作。



1. 一种用于编辑 workflow 逻辑的图形用户界面程序,包括:
图形用户界面工具,其同时显示:
第一屏幕区域,其中应用程序屏幕是可编辑的;
第二屏幕区域,其中所述 workflow 逻辑是可编辑的;
第三屏幕区域,其提供用于在编辑所述 workflow 逻辑时使用的用口语语法呈现的多个 workflow 部件的上下文有关菜单;以及
第四屏幕区域,其提供可用来为所述 workflow 逻辑创建 workflow 逻辑语句的数据字段和可重复字段的菜单。
2. 根据权利要求 1 所述的图形用户界面,其中,通过包括向所述图形用户界面工具可访问的数据库添加的数据字段,所述 workflow 逻辑是可编辑的。
3. 根据权利要求 2 所述的图形用户界面,其中,通过用新 workflow 逻辑语句更换所述 workflow 逻辑语句,所述 workflow 逻辑进一步可编辑。
4. 根据权利要求 1 所述的图形用户界面,其中,所述 workflow 逻辑进一步可编辑以包括多个所述数据字段和可重复项以及所述数据字段和可重复项的排列。
5. 根据权利要求 1 所述的图形用户界面,其中,通过生成新 workflow 逻辑语句并且包括数据库的先前添加或者新添加的字段,所述 workflow 逻辑进一步可编辑。
6. 根据权利要求 5 所述的图形用户界面,其中,所述第四屏幕区域被动态地更新以对应于对所述数据库中的所述字段的改变。
7. 根据权利要求 1 所述的图形用户界面,其中,通过生成包括在上下文有关菜单内提供的运算符、函数和字段的排列的所述 workflow 逻辑,所述 workflow 逻辑进一步可编辑。
8. 根据权利要求 7 所述的图形用户界面,其中,通过提供用于施加 while 循环的构造,所述 workflow 逻辑进一步可编辑。
9. 根据权利要求 7 所述的图形用户界面,其中,构造被提供用于将逻辑测试应用于可重复项中的所有字段,以及所述可重复项内的在所述可重复项内一级嵌套的所有字段。
10. 根据权利要求 1 所述的图形用户界面,其中,通过修改所述屏幕以包括数据库中的新添加或者修改的字段,所述应用程序屏幕进一步可编辑。
11. 根据权利要求 1 所述的图形用户界面,其中,通过用新应用程序屏幕更换所述应用程序屏幕,所述应用程序屏幕进一步可编辑。
12. 根据权利要求 1 所述的图形用户界面,其中,所述应用程序屏幕可以是可编辑的,以包括任何数目的字段和可重复字段以及所述字段和可重复字段的排列。
13. 根据权利要求 1 所述的图形用户界面,其中,通过生成新应用程序屏幕并且添加数据库内的先前添加的字段,所述应用程序屏幕进一步可编辑。
14. 根据权利要求 1 所述的图形用户界面,其中,通过创建子屏幕以对所述字段和可重复字段进行分组,所述应用程序屏幕进一步可编辑。
15. 一种用于将 workflow 逻辑和与所述 workflow 逻辑关联的应用程序屏幕与应用程序动态地集成的系统,所述系统包括:
应用程序,用于执行所述 workflow 逻辑和所述应用程序屏幕;
数据库,其存储多个字段和多个可重复字段以及与业务 workflow 对应的可执行构造;以及

图形用户界面工具,其向用户提供屏幕区域,用于使用多个已定义函数、多个逻辑运算符、多个关键字和所述多个字段来为所述业务 workflow 创建所述 workflow 逻辑,以产生与所述业务 workflow 的所述 workflow 逻辑协同工作的包括字段和可重复字段的应用程序屏幕。

16. 根据权利要求 15 所述的系统,其中,所述图形用户界面工具使新的或者修改的字段或者新的或者修改的可重复字段在所述新的或者修改的字段或者所述新的或者修改的可重复字段被添加到所述数据库时变得可用。

17. 根据权利要求 16 所述的系统,其中,所述多个字段和多个可重复字段在所述数据库中存储为可扩展标记语言 (XML) 代码。

18. 根据权利要求 15 所述的系统,其中,所述图形用户界面工具用口语语法提供上下文有关参数的菜单以创建所述 workflow 逻辑。

19. 根据权利要求 15 所述的系统,其中,所述图形用户界面工具提供用于创建与所述 workflow 逻辑协同的应用程序屏幕的手段。

20. 根据权利要求 15 所述的系统,其中,所述 workflow 逻辑的表示用口语语法来呈现,所述口语语法可转换成存储于所述数据库上并且变得可为所述应用程序所用的 XML 代码。

21. 根据权利要求 15 所述的系统,其中,所述应用程序屏幕的表示被转换成存储于所述数据库上并且变得可为所述应用程序所用的 XML 代码。

利用 GUI 工具来编辑 workflow 逻辑和屏幕的方法和系统

技术领域

[0001] 本发明涉及 workflow 管理系统,并且具体地涉及一种用于使用图形用户界面来创建和 / 或编辑业务 workflow 逻辑的系统和方法。

背景技术

[0002] 许多业务使用应用软件来管理和分析 workflow。通过对包括工作定单、后勤、库存管理、制造、交易批准和验证的业务功能和过程的控制进行辅助和 / 或自动化, workflow 软件应用程序潜在地提高了生产率和效率。

[0003] 每个业务实体实施它自身的随着它的产业、组织结构和独特业务计划而变的独特业务过程。结果,公司通常开发定制的软件,以满足它们的业务过程的特定要求。

[0004] 许多软件应用程序不能容易地适应对公司的业务计划和过程的改变,该改变要求修改应用程序的 workflow 逻辑,或者要求添加和 / 或重新定义系统中的数据字段。修改应用程序的 workflow 逻辑通常还要求修改应用程序的用户屏幕或者添加新用户屏幕。

[0005] 更新定制的软件应用程序以适应对公司的业务计划和过程的改变通常要求程序员重新编写、重新编译并测试应用程序。如果在测试中发现错误,则应用程序可能需要进一步修订,这将延迟实施对系统的必要更新。因此,对客户软件的更新在财务资源和时间延迟方面成本高。

[0006] workflow 软件应用程序需要能够适应业务改变并且允许用户编辑 workflow 逻辑。然而,这样的可配置软件应用程序必须克服几个设计问题。使用户能够在编辑 workflow 逻辑时自由地编写逻辑语句的应用程序使得有机会发生可能对用户不明显并且能够破坏系统的编程错误。允许用户创建或者编辑 workflow 逻辑语句的应用程序通常要求用户具备使用程序编辑工具进行逻辑编程的高级知识。旨在提供简单用户友好编辑工具的软件可能限制了提供用在 workflow 逻辑中的逻辑函数。

[0007] 在用户可配置 workflow 应用程序的设计中的进一步的问题是仅显示 workflow 逻辑语句以便进行编辑。用户不能在应用程序屏幕的环境中查看 workflow 语句,所述应用程序屏幕被呈现给执行 workflow 过程的工作人员。创建和编辑与新的和 / 或修订的 workflow 逻辑协同工作的应用程序屏幕通常通过单独的系统工具作为单独的任务来进行。通过使用多个工具和图形用户界面来使必要的应用程序屏幕与 workflow 逻辑的改变相协调的任务耗费时间、不方便并且使得易于发生可能对用户不明显的问题和不一致性。

[0008] 将新的变化集成到典型的应用软件中的 workflow 逻辑要求重新编译软件并且重启系统。系统的可操作性暂停,直至它被重启。这可能是对业务的高成本中断,特别是如果新的变化要求进一步的测试和调整,这会引入额外的重新编译和停工时间。

[0009] 允许将新的和修订的 workflow 逻辑集成到系统而不重新编写和重新编译系统的现有软件要求使用解析树,这降低了系统的处理速度和效率。

[0010] 允许用户为 workflow 创建应用程序屏幕的现有软件必须在一开始被设计成包括所有必要的屏幕和字段,以满足每个业务过程的需要。这样的屏幕可能过多载入对于多数标

准业务过程而言通常并不需要的并且可能让工作人员感到困惑和不便的字段和信息。

[0011] 需要的是一种利用用户友好图形用户界面工具的系统,该图形用户界面工具允许用户创建和 / 或编辑应用软件的工作流逻辑而错误机会最少,并且高效地创建和 / 或编辑相应的工作流应用程序屏幕而无需本来会中断应用程序可操作性的应用软件重新编写和重新编译。

发明内容

[0012] 本发明是一种在工作流管理系统中用来允许用户创建和 / 或编辑工作流并且同时查看、创建和 / 或编辑与工作流逻辑协同工作的相应应用程序屏幕的图形用户界面 (GUI)。

[0013] GUI 工具的用户友好设计允许用户实施该工具而无需高级编程语言。GUI 工具用用户友好的口语语法提供让用户用来创建和 / 或编辑工作流逻辑的函数、运算符、字段和变量的上下文有关菜单。GUI 工具的上下文有关菜单有效地防止用户在创建或者编辑工作流逻辑的过程中产生语法错误。因此,GUI 工具能够向用户赋予创建和编辑工作流的灵活性而又防止语法错误破坏系统。

[0014] GUI 工具也向用户提供用以查看、创建和编辑与工作流逻辑协同工作的应用程序屏幕的能力。在业务工作流期间向工作人员显示应用程序屏幕以辅助他们进行各种业务过程。这种创新解决方案允许用户同时创建和配置根据用户定义的工作流逻辑来工作的应用程序屏幕。

[0015] 系统也被设计成集成对工作流和应用程序屏幕的任何改变而无需重新编写和重新编译应用程序。用户因此可以对工作流逻辑和应用程序屏幕进行改变而不在任何时间中断工作流管理系统的可操作性、也不造成任何停工时间。

[0016] 提供了一种用于编辑工作流逻辑的图形用户界面程序,该程序包括同时显示以下屏幕区域的图形用户界面工具:第一屏幕区域,其中应用程序屏幕是可编辑的;第二屏幕区域,其中工作流逻辑是可编辑的;第三屏幕区域,其提供用于在编辑工作流逻辑时使用的用口语语法呈现的多个工作流部件的上下文有关菜单;以及第四屏幕区域,其提供可用来为所述工作流逻辑创建工作流逻辑语句的数据字段和可重复字段的菜单。

[0017] 通过包括向图形用户界面工具可访问的数据库添加的数据字段,工作流逻辑可以是可编辑的。通过用新工作流逻辑语句更换工作流逻辑语句,工作流逻辑可以进一步可编辑。工作流逻辑可以进一步可编辑以包括多个数据字段和可重复项及其排列。通过生成新工作流逻辑语句并且包括数据库的先前添加或者新添加的字段,工作流逻辑可以更进一步可编辑。

[0018] 可以动态地更新第四屏幕区域以对应于对数据库中的字段的改变。

[0019] 工作流逻辑可以通过以下进一步可编辑:通过包括在上下文有关菜单内提供的运算符、函数和字段的排列,生成所述工作流逻辑。通过提供用于施加 while 循环的构造,工作流逻辑可以进一步可编辑。用于将逻辑测试应用于所有字段的构造可以是在可重复项中,以及在可重复项内的在可重复项内一级嵌套的所有字段中。

[0020] 通过修改屏幕以包括数据库中的新添加或者修改的字段,应用程序屏幕可以是可编辑的。通过用新应用程序屏幕更换应用程序屏幕,应用程序屏幕可以进一步可编辑。应

用程序屏幕可以进一步可编辑以包括任何数目的字段和可重复字段及其排列。通过生成新应用程序屏幕并且添加数据库内的先前添加的字段,应用程序屏幕也可以可编辑。通过创建子屏幕以对字段和可重复字段进行分组,应用程序屏幕可以进一步可编辑。

[0021] 提供了一种用于将 workflows 逻辑和与 workflows 逻辑关联的应用程序屏幕与应用程序动态地集成的系统,该系统包括:应用程序,用于执行 workflows 逻辑和应用程序屏幕;数据库,其存储多个字段和多个可重复字段以及与业务 workflows 对应的可执行构造;图形用户界面工具,其向用户提供屏幕区域,用于使用多个已定义函数、多个逻辑运算符、多个关键字和所述多个字段来为业务 workflows 创建工作流的逻辑,以产生与业务 workflows 的工作流的逻辑协同工作的包括字段和可重复字段的应用程序屏幕。

[0022] 图形用户接口工具可以使新的或者修改的字段或者新的或者修改的可重复字段在新的或者修改的字段或者新的或者修改的可重复字段被添加到数据库时变得可用。所述多个字段和多个可重复字段可以在数据库中存储为可扩展标记语言 (XML) 代码。图形用户界面工具可以用口语语法提供上下文有关参数的菜单以创建工作流的逻辑,并且可以提供用于创建与 workflows 逻辑协同的应用程序屏幕的手段。

[0023] workflows 逻辑的表示可以用口语语法来呈现,该口语语法可转换成存储于数据库上并且变得可为应用程序所用的 XML 代码。同样地,应用程序屏幕的表示可以被转换成存储于数据库上并且变得可为所述应用程序所用的 XML 代码。本发明的其它优点将从这里提供的对本发明的详细描述和示例中变得明显。

附图说明

[0024] 图 1 图示了根据本发明的示例图形用户界面工具及其关联部件区域;

[0025] 图 2 图示了根据一个优选实施例的其中可以实施本发明的计算环境的框图;

[0026] 图 3 是业务 workflows 的流程图;

[0027] 图 4 是示出了每个过程如何通过应用程序屏幕来表示的工作流的流程图,其中该应用程序屏幕有效地提供系统的工作流的逻辑相对于图 3 中所示的业务 workflows 的一对一映射;

[0028] 图 5 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的“Verify Info(验证信息)”屏幕,如何使用工具部件来配置应用程序屏幕和相应的工作流的逻辑;

[0029] 图 6 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的“What Type Of Meter(什么类型的计量器)”屏幕,如何使用工具部件来配置应用程序屏幕和相应的工作流的逻辑;

[0030] 图 7 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的“Regular Screen(普通屏幕)”,如何使用工具部件来配置应用程序屏幕和相应的工作流的逻辑;

[0031] 图 8 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的“Smart Meter Info(智能计量器信息)”,如何使用工具部件来配置应用程序屏幕和相应的工作流的逻辑;

[0032] 图 9 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的“New Meter(新计量器)”,如何使用工具部件来配置应用程序屏幕和相应的工作流的逻辑;

[0033] 图 10 是 GUI 工具的屏幕截图,其图示了针对图 4 中所示的系统工作流的

“Complete Screen(完成屏幕)”,如何使用工具部件来配置应用程序屏幕和相应的工作流逻辑;

[0034] 图 11 是 GUI 工具的屏幕截图,其图示了当用户选择“Workflow(工作流)”语句时提供的允许构造的上下文有关菜单;

[0035] 图 12 是 GUI 工具的屏幕截图,其图示了当用户选择“Variable(变量)”语句时提供的允许构造的上下文有关菜单;

[0036] 图 13 是 GUI 工具的屏幕截图,其图示了当用户选择“Test(测试)”函数时提供的允许构造的上下文有关菜单;

[0037] 图 14 是 GUI 工具的屏幕截图,其图示了当用户选择“Then(则)”函数时提供的允许构造的上下文有关菜单;

[0038] 图 15 是 GUI 工具的屏幕截图,其图示了当用户选择“Else(否则)”函数时提供的允许构造的上下文有关菜单;

[0039] 图 16 是 GUI 工具的屏幕截图,其图示了当用户选择“While(当……时)”函数时提供的允许构造的上下文有关菜单;

[0040] 图 17 是 GUI 工具的屏幕截图,其图示了在用户选择配置“Set(置值)”构造内的“Z”变量时由上下文有关编辑器提供的允许参数;

[0041] 图 18 是 GUI 工具的屏幕截图,其图示了在用户选择配置“Set(置值)”构造内的“Operation(运算)”元素时由上下文有关编辑器提供的允许参数;

[0042] 图 19 是工作流编程语言的例子;

[0043] 图 20 是用于在工作流逻辑中利用 while 循环时的独特构造的例子;以及

[0044] 图 21 图示了用于在工作流逻辑中并入可重复字段的独特构造的例子。

图 22 图示了使用如图 21 中所示的“Tests(多项测试)”中的可重复项的用于构造的编程语言。

具体实施方式

[0045] 可以用许多不同的配置、形式和设计元素来实现本发明及其原理。以如下理解来描述这里本发明的附图、示例和描述:本公开内容被视为举例说明本发明的原理而并非意欲将本发明限制于所示实施例。本领域技术人员可以预想在本发明的范围内的许多可能变化。

[0046] workflows 系统包括图形用户界面 (GUI) 工具 10,其允许用户创建和 / 或编辑工作流逻辑并且同时查看、创建和 / 或编辑与工作流逻辑协同工作的应用程序工作流屏幕。

[0047] 图 1 图示了称为起源 (Genesis) 的 GUI 工作流屏幕工具 10 中的部件。GUI 工作流屏幕工具 10 提供:

[0048] 工作流编辑器 20 的屏幕区域,用于让用户创建和 / 或编辑工作流逻辑;

[0049] 屏幕编辑器 30 的屏幕区域,用于让用户查看、创建和 / 或编辑相应的工作流屏幕,其中工作流逻辑应用于应用程序中;

[0050] 上下文有关编辑器 40 的屏幕区域,其提供用以在工作流逻辑内创建和 / 或编辑逻辑语句的允许参数的上下文有关菜单;以及

[0051] 字段菜单 50 的屏幕区域,其提供数据字段和可重复数据字段。这里也称为“可重复项 (repeatable)”的可重复数据字段是如下字段,这些字段是往往重复许多行并且在

HTML 范式中也称为表的字段组合。

[0052] 图 2 示出了其中可以实施本发明的示例计算环境 70。本领域技术人员将会意识到可以用其它计算机系统配置来实施 GUI 工具 10。

[0053] GUI 工作流屏幕工具 10 可以实施于工作流管理应用程序 71 中。通过内部网或者网络 74 如因特网,远程桌面客户机 72 和 / 或无线移动客户机 73 可以访问服务器 76,并且数据库 77 可以访问工作流管理应用程序 71。数据库 77 用来存储与数据字段、可重复字段、工作流逻辑和应用程序屏幕有关的信息。也可以提供其它部件、函数和数据库。

[0054] 工作流逻辑和应用程序屏幕一旦被定义就可以存储于数据库 77 中,以便由一个或者多个客户机 72、73 在执行工作流时使用。

[0055] 工作流可以用于任一类业务或者其它活动,而图 3 图示了用于计量器更换工作的示例业务工作流程 80。工作人员根据由业务工作流程 80 定义的必要过程来进行计量器更换工作。在工作流 80 的步骤 81 中,工作人员验证他们已经到达正确地址并且验证在工作定单中声明的正确计量器 ID。在步骤 82 中,工作人员标识计量器是否为“Regular(普通)”计量器或者计量器是否为“Smart(智能)”计量器。如果计量器为“Regular(普通)”计量器,则工作人员进行步骤 83a,该步骤要求录入“Old reading(旧读数)”。如果计量器为“Smart(智能)”计量器,则工作人员进行步骤 83b,该步骤要求录入“Old reading(旧读数)”并且录入“Demand read(需求读数)”。工作人员然后进行步骤 84,该步骤要求拆卸旧计量器、安装新更换的计量器并且收集新计量器 ID。步骤 85 表明工作流完成。

[0056] 参照图 4,应用程序 71 允许用户创建和配置应用程序屏幕 93 到 98,这些应用程序屏幕在工作流 90 中用来辅助工作人员准确和高效地执行图 3 中所示的业务工作流程 80 的必要过程。

[0057] 各屏幕 93 到 98 与用户定义的工作流逻辑(比如 91、92)协同工作,并且在本质上为工作流逻辑的表示层。用户可以创建多个屏幕,这些屏幕一起工作以形成对业务工作流程 80 的必要阶段进行建模的完整工作流 90。屏幕 93 到 98 有效地提供系统工作流逻辑 90 相对于业务工作流程 80 的一对一映射和便利检阅。

[0058] 图 5 到图 10 图示了形成如图 4 中所示的完整工作流 90 的一系列屏幕的配置。各屏幕代表如图 3 中所示的业务工作流程 80 内的单独过程。

[0059] 图 5 图示了使用 GUI 工具来配置在工作人员的移动设备上呈现的工作流屏幕 37,其有效地表示了业务工作流程 80 的步骤 1 和工作流 90 内的屏幕 93。屏幕 37 要求工作人员录入地址和计量器 ID 以验证他们在正确的工作定单位置。GUI 工具允许用户同时查看和配置与工作流屏幕 37 协同工作的工作流逻辑 21。工作流逻辑 21 被配置成在工作人员已经录入和验证地址和计量器 ID 之后示出表示业务工作流程 80 内的步骤 82 的“WhatType OfMeter(什么类型的计量器)”屏幕 94。

[0060] 图 6 图示了配置屏幕 37 以显示有效地表示业务工作流程 80 的步骤 82 的“What Type OfMeter(什么类型的计量器)”屏幕 94。屏幕 94 要求工作人员录入计量器 ID 和计量器类型。工作流逻辑 21 被配置成如果工作人员标识计量器为“Regular(普通)”计量器则示出“Regular Screen(普通屏幕)”。当将计量器标识为不同于“Regular(普通)”计量器时,工作流逻辑 21 被配置成示出“Smart Meter Info(智能计量器信息)”屏幕。

[0061] 图 7 图示了将屏幕 37 配置成表示业务工作流程 80 的步骤 83a 的“Regular

Screen(普通屏幕)”95。屏幕 95 要求工作人员将旧计量器的计量器读数录入到“Old Read(旧读数)”字段中并且提供用以允许录入适用注释的“Comments(注释)”字段。 workflows 21 被配置成在工作人员已经完成录入必要数据之后示出表示业务 workflow 80 的步骤 84 的“NewMeter(新计量器)”屏幕 97。

[0062] 图 8 图示了配置屏幕 37 以显示表示业务 workflow 80 的步骤 83b 的“Smart Meter Info(智能计量器信息)”屏幕 96。屏幕 96 要求工作人员录入用于“Old Read(旧读数)”字段和“Old Demand(旧需求)”字段的数据并且还提供用以允许录入适用注释的“Comments(注释)”字段。 workflows 21 被配置成在工作人员已经录入所有必要数据之后示出表示业务 workflow 80 的步骤 84 的“New Meter(新计量器)”屏幕 97。

[0063] 图 9 图示了配置屏幕 37 以显示表示业务 workflow 80 的步骤 84 的“NewMeter(新计量器)”屏幕 97。在工作人员已经物理上用新计量器更换旧计量器之后,要求工作人员录入新计量器 ID 并在“New Read(新读数)”字段中录入新计量器读数。 workflows 21 被配置成在工作人员已经录入新计量器 ID 和新计量器读数之后示出“Complete Screen(完成屏幕)”屏幕 98。

[0064] 图 10 图示了配置屏幕 37 以显示表示业务 workflow 80 的步骤 85 的“Completed(完成)”屏幕 98。屏幕 98 被配置成显示用以向工作人员通知已经成功地完成计量器更换工作的消息。 workflows 21 示出无需后续屏幕,这有效地创建了 workflow 中的结束点。

[0065] GUI 工具 10 允许创建和配置屏幕以从工作人员收集用以执行 workflows 的数据。也可以创建和配置屏幕以显示来自 workflows 处理的信息,和 / 或向工作人员请求附加数据以执行附加 workflows,或者向工作人员提供与手头上的立即任务相关的信息。

[0066] 参照图 1, 用户可以利用屏幕编辑器 30 来创建和配置屏幕。用户可以选择使用新屏幕按钮 32 来创建新屏幕并且为系统定义名称,或者使用屏幕下拉菜单 31 来选择 Screen(屏幕)菜单中列举的屏幕以便进行编辑。

[0067] 用户可以使用开始屏幕按钮 33 为特定 workflow 91 设置作为开始屏幕的屏幕。用户可以通过从 Fields(字段)菜单 50 添加字段并且使用工具栏上的文本框按钮 36 插入文本框来配置屏幕。字段和文本框用来由工作人员收集数据以处理如用户定义的 workflows。通过将 Fields 菜单 50 可得到的字段和可重复项拖放到屏幕区域 30 中,或者通过利用指示设备如鼠标进行指示和点击,向适当屏幕添加字段和文本框。

[0068] Fields 菜单 50 提供来自动态可配置数据库 77 的字段和可重复项。数据库 77 中的新的和 / 或修订的字段立即可为 GUI workflow 屏幕工具 10 所用。字段具有可以由用户配置的多个属性。字段属性的例子包括标注、宽度、高度、类型定向、必备、只读和标签顺序。系统防止用户从数据库 77 删除已经并入到逻辑流语句中或者已经被引用的字段 51。所有字段和有关属性在数据库 77 中存储为 XML 数据。

[0069] 用户也可以为屏幕配置用以将字段和文本框一起分组的称为“Groups(组)”的一个或者多个子屏幕 37。对字段的分组提供一种在屏幕中对字段和文本框进行呈现和定位的便利手段。用户可以通过从工具栏选择 Group 选项按钮 35 以添加子屏幕 37 来为屏幕配置子屏幕。字段可以直接添加到子屏幕 37 的区域。通过拖动在子屏幕 37 的侧边上提供的尺寸调整句柄 38,可以对子屏幕 37 进行定位和尺寸调整以容纳多个字段。通过拖动在字段的一侧(在图 1 中为左侧)上提供的位置句柄 39,可以将字段定位于屏幕内。所配置的屏幕

存储于数据库 77 中并且变得可为应用程序 71 所用。

[0070] 在从工具栏选择下拉菜单选项 34 时变得可用的下拉菜单中列举已经向屏幕添加的子屏幕 37。用户可以选择下拉菜单 34 中列举的单独子屏幕以便进行编辑或者删除。也可以通过在单独子屏幕区域上点击指示设备来单独选择子屏幕 37。

[0071] GUI 工具 10 为用户提供用以创建和配置屏幕的能力,这些屏幕通过向工作人员显示和 / 或通过从工作人员取回用以执行业务工作流 80 的各阶段中的特定过程的仅仅必要和充足信息来辅助对业务工作流 80 内的业务过程的准确和高效执行。GUI 工具 10 的用户具有创建和配置屏幕的灵活性,这些屏幕提供及时信息并且避免显示在业务过程的执行期间可能让工作人员感到困惑和不便的过多和无关信息。

[0072] 参照图 11, 工作流程编辑器 20 允许用户创建与应用程序屏幕协同工作的新工作流程逻辑和 / 或编辑与应用程序屏幕协同工作的现有工作流程逻辑。用户可以通过以下来创建和 / 或编辑工作流程逻辑 : 选择工作流程逻辑内的元素 21, 然后从工具栏选择添加选项 22, 或者通过利用指示设备右击元素 21。工作流程编辑器 20 然后提供用户可以用来创建和 / 或编辑工作流程逻辑的允许构造 41 的上下文有关菜单。用户可以仅选择在允许构造 41 的菜单内提供的选项。因此防止了用户录入将会破坏工作流程逻辑的语法错误。例如参照图 11, 当用户选择工作流程编辑器 20 内的“Workflow(工作流)”元素 21 时, GUI 工具 10 提供可以用来配置工作流程逻辑的允许构造 41 的菜单。

[0073] 参照图 12, 当用户选择工作流程编辑器 20 内的“Variable(变量)”元素 24 时, 允许构造 41 的上下文有关菜单仅向用户提供用以定义新变量的允许“Variable(变量)”选项。

[0074] 参照图 13, 当用户选择“Tests(多项测试)”元素 25 时, 工作流程编辑器 20 提供“Tests(多项测试)”元素 25 特有的允许构造 41 的菜单, 包括“Test(一项测试)”、“Tests(多项测试)”和“For Each Repeatable Item(针对各可重复项)”。

[0075] 参照图 14, 当用户选择“Then(则)”元素 26 时, 工作流程编辑器 20 提供用于“Then(则)”元素 26 的允许构造 41 的上下文有关菜单, 包括“If(如果)”、“While(当)”、“Set2(置值 2)”、“Show(示出)”、“ForEach Repeatable Item2(对于各可重复项 2)”和“Run(运行)”。

[0076] 参照图 15, 当用户选择“Else(否则)”元素 27 时, 工作流程编辑器 20 提供用于“Else(否则)”元素 27 的允许构造 41 的上下文有关菜单, 包括“If(如果)”、“While(当)”、“Set2(置值 2)”、“Show(示出)”、“ForEach Repeatable Item2(对于各可重复项 2)”和“Run(运行)”。

[0077] 参照图 16, 当用户选择“While(当)”元素 28 时, 工作流程编辑器 20 提供用于“While(当)”元素 28 的允许构造 41 的上下文有关菜单。

[0078] 当用户从工作流程编辑器 20 选择工作流程逻辑内的构造时, 上下文有关编辑器 40 向用户提供用于所选构造的上下文有关允许参数。用户可以有选择地组合来自上下文有关编辑器 40 的可用参数以配置工作流程逻辑语句。上下文有关编辑器 40 仅向用户提供允许的语法选项, 用以防止在创建或者编辑逻辑语句时出现语法错误。上下文有关编辑器 40 用口语语法提供相关参数以求用户友好性。

[0079] 图 17 图示了在用户选择工作流程编辑器 20 内的“Set(置值)”命令 29 时上下文有关编辑器 40 可以提供的参数。“Set(置值)”命令 29 表示以下构造 :

[0080] Set Z = X{Operation}Y

[0081] 其中上下文有关编辑器 40 提供用以配置“Set(置值)”构造的“X”、“Y”和“Z”参数的允许选项。图 17 图示了为了配置用于“Z”参数的参数类型 42 而向用户提供的下拉菜单 43。该图示作为例子示出了用户已经从下拉菜单 43 选择“Field(字段)”选项 44 以定义用于“Z”参数的参数类型 42。图 18 图示了为了配置构造内的“Operation(运算)”元素 45 而向用户提供的允许选项 46。该图示作为例子示出了用户已经选择“Fill(填入)”选项 47 以允许工作人员录入用于“X”参数的值。图 17 和图 18 中的例子图示了上下文有关编辑器 40 如何仅向用户提供用以创建和 / 或编辑 workflow 逻辑的相关和语法正确的选项。

[0082] 用户也可以通过使用工具栏 23 中提供的定位箭头对 workflow 语句的序列重新排序来修订 workflow。

[0083] 参照图 11, GUI 工具 10 允许用户在创建 workflow 逻辑构造时利用包括“if(如果)”、“Tests(多项测试)”、“Then(则)”、“Else(否则)”、“For each(对于各项)”、“While(当)”、“Show(示出)”和“Run(运行)”的函数作为构造 41 以实现复杂 workflow 定义。参照图 20, 系统利用构造 101 来提供用于在处理 workflow 逻辑时使用的 while 循环。

[0084] 参照图 21, 系统利用构造 102 来允许逻辑测试应用于可重复项内的所有字段。可重复字段也可以嵌套于可重复字段内, 并且在这样的情形中可以描述为“one-level-nested(一级嵌套)”。系统构造 102 也可以用来将逻辑测试应用于可重复项内的在可重复字段内一级嵌套的所有字段。构造 102 允许在处理 workflow 逻辑时使用对可重复字段和一级嵌套可重复字段的综合分析。图 22 图示了使用如图 21 中所示的“Tests(多项测试)”中的可重复项的用于构造 102 的编程语言 103。

[0085] GUI workflow 屏幕工具 10 将 workflow 逻辑集成为可为应用软件所用的可执行构造。应用软件解释新的或者编辑的 workflow 逻辑, 而无需重新编写或者重新编译应用软件, 也不在任何时间中断应用软件的操作性。

[0086] 参照图 19, 系统使用基于可扩展标记语言 (XML) 的编程语言来集成 workflow 逻辑, 该编程语言提供编程构造如变量、条件、运算、循环和定制动作并且允许用户利用变量和全局变量来辅助逻辑。该语言在用户将变化上传到数据库时作为 XML 文档来加载。

[0087] 系统可以使用文档对象模型 (DOM) 方法来遍历 workflow 逻辑。系统使用 DOM 来执行由逻辑定义的 XML 指令。

[0088] 针对桌上型计算机、服务器或者 PDA 消费类设备一次性编写并且编译共同项目。共同项目然后读取用于文档的 XML 定义并且允许客户机 72、73 利用文档和 workflow。客户机应用程序将 XML 代码解序列化能够成能够对 XML 命令节点进行逻辑求值的对象。应用程序在启动时加载相关文档文件, 并且在被通知时重新加载更新的文件。

[0089] 虽然出于说明的目的已经详细地公开了本发明的特定优选实施例, 但是应当认识到, 对所公开的装置的改变或者修改落在本发明的范围内。

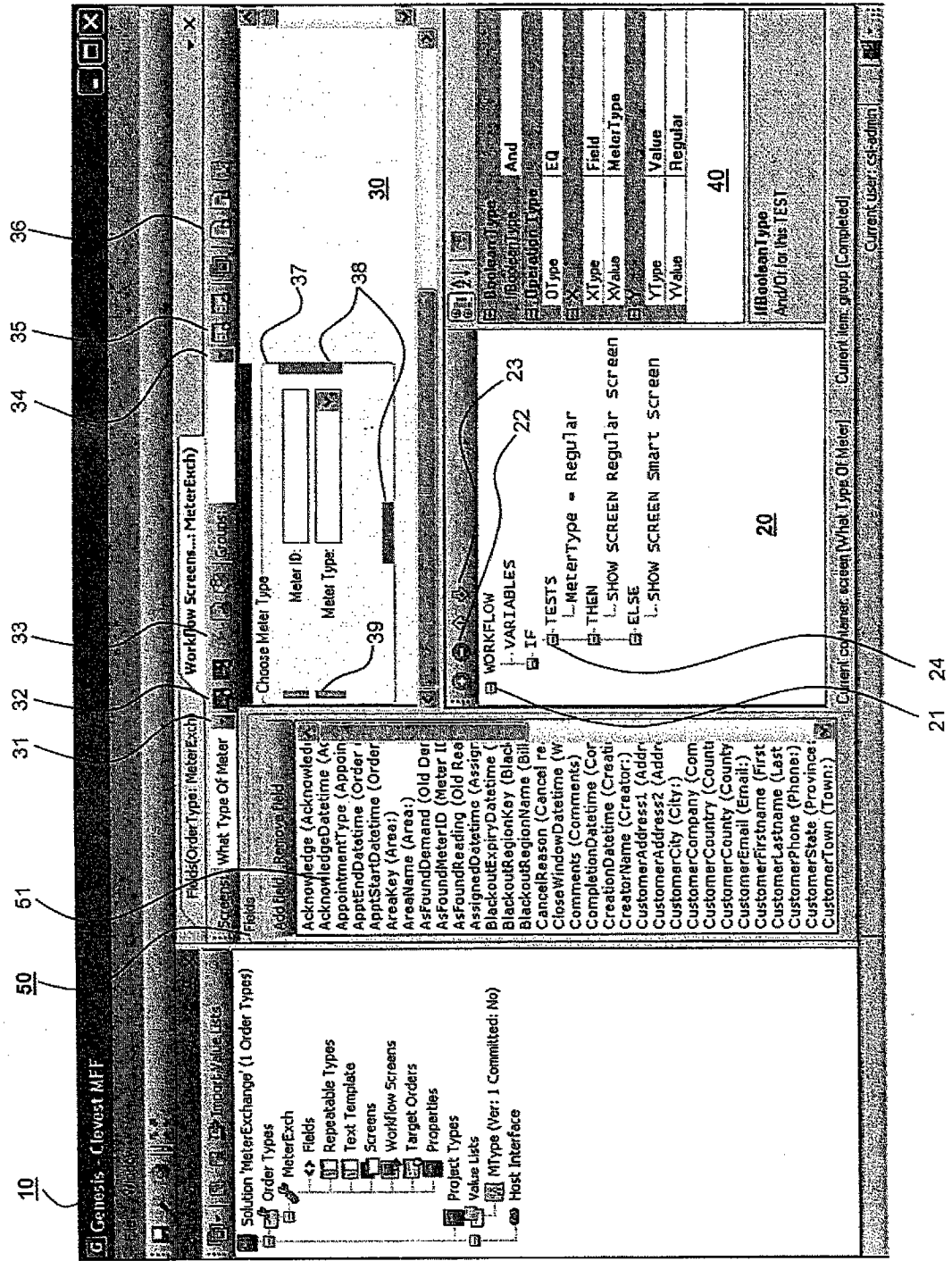


图 1

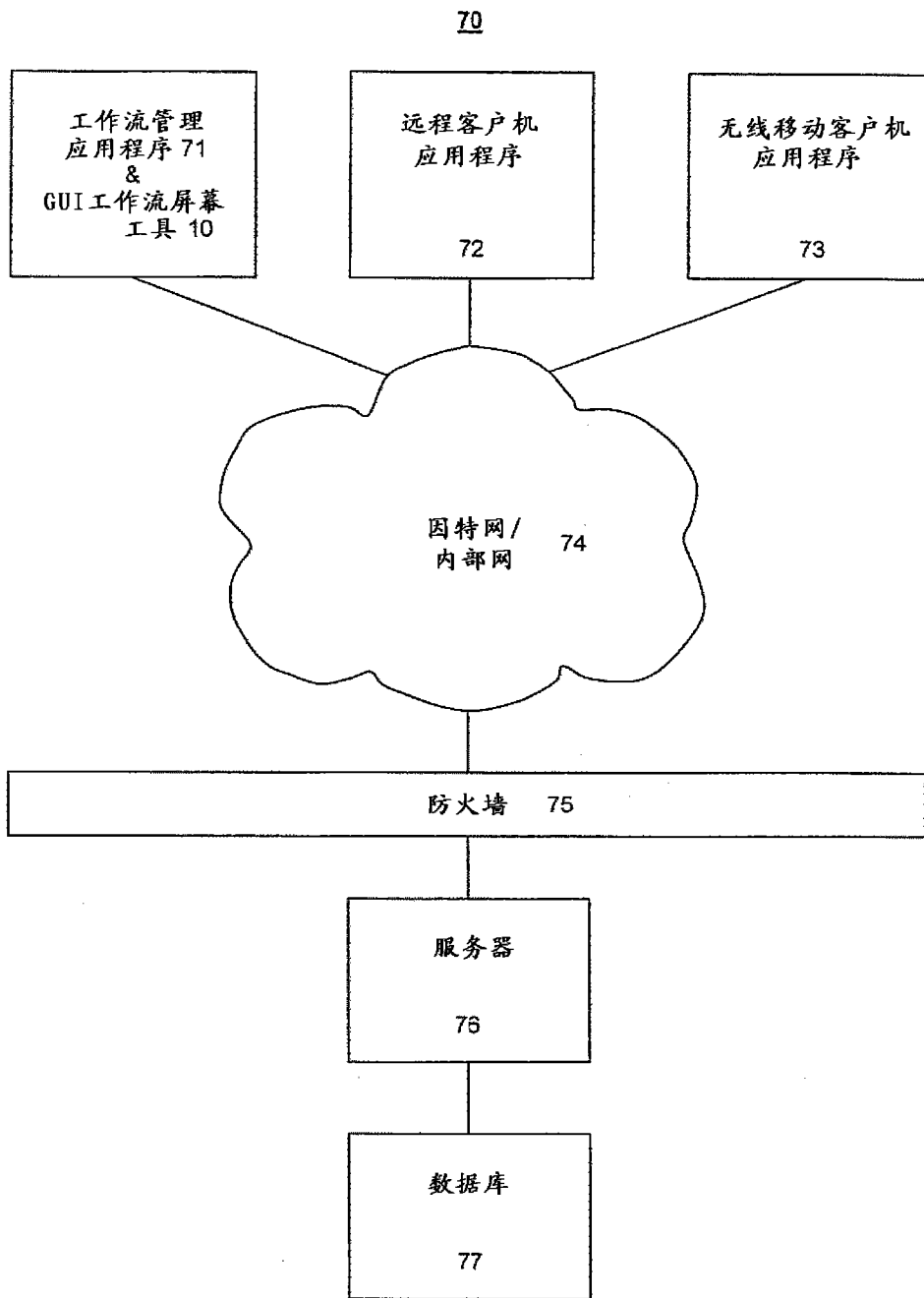


图 2

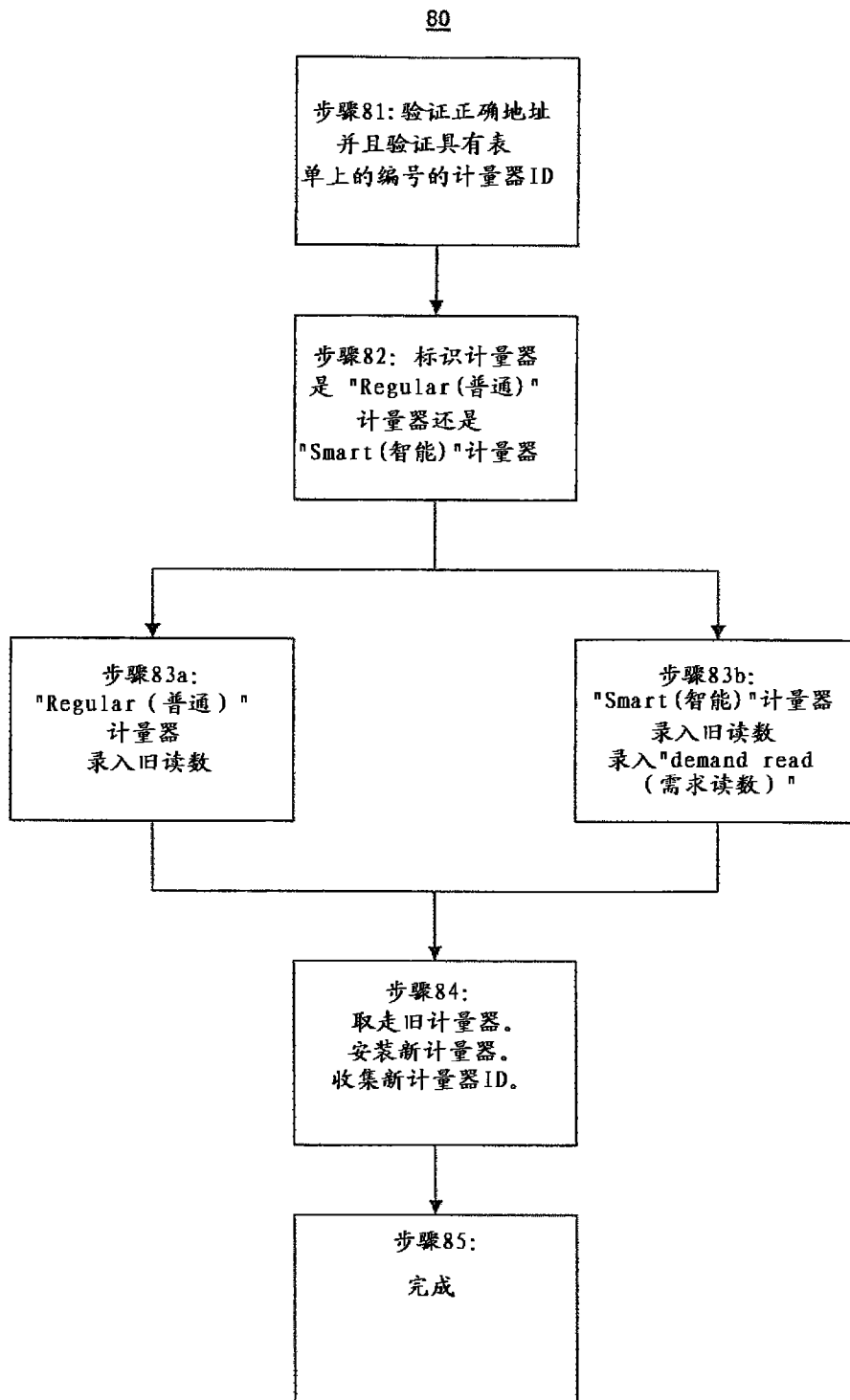


图 3

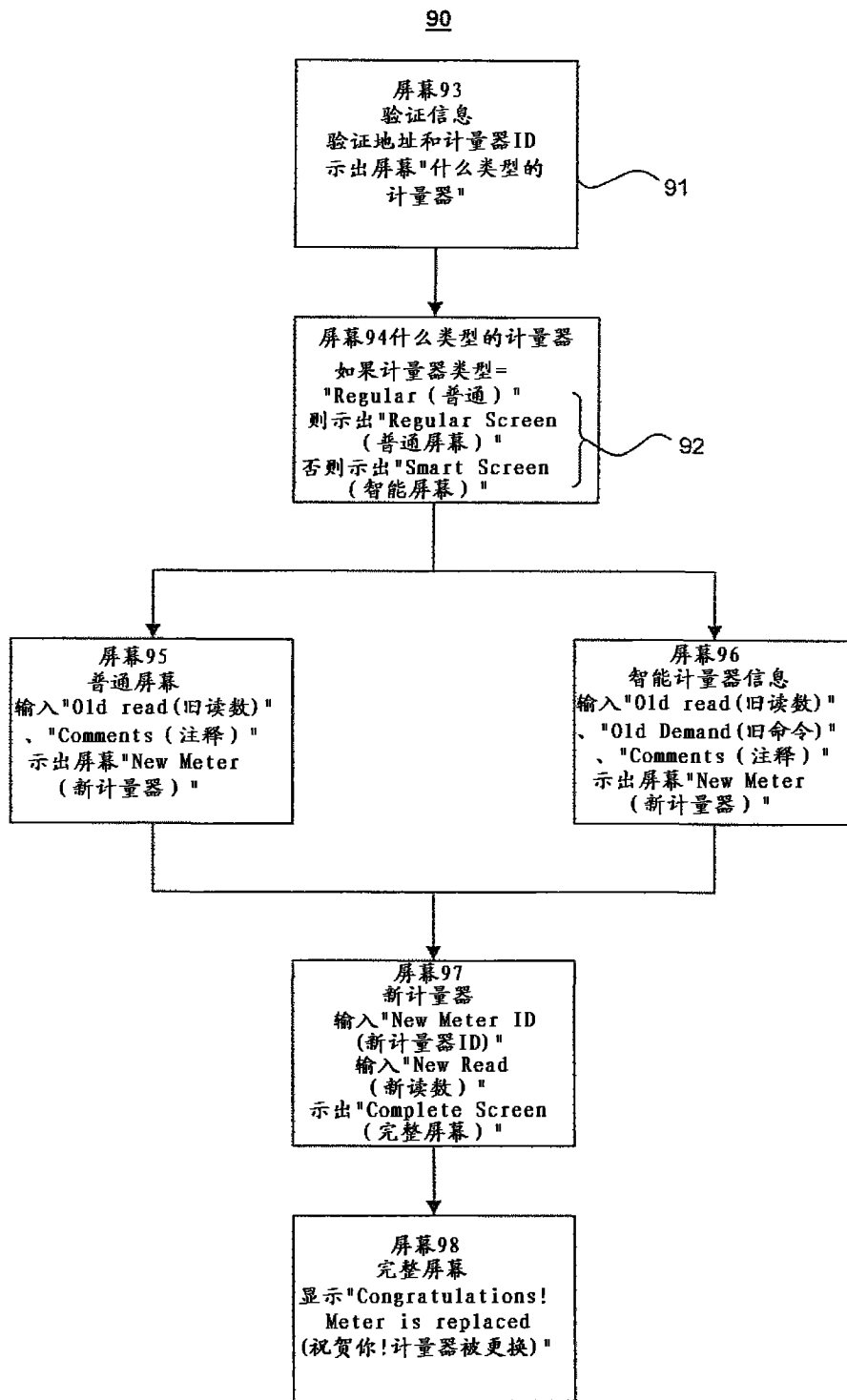


图 4

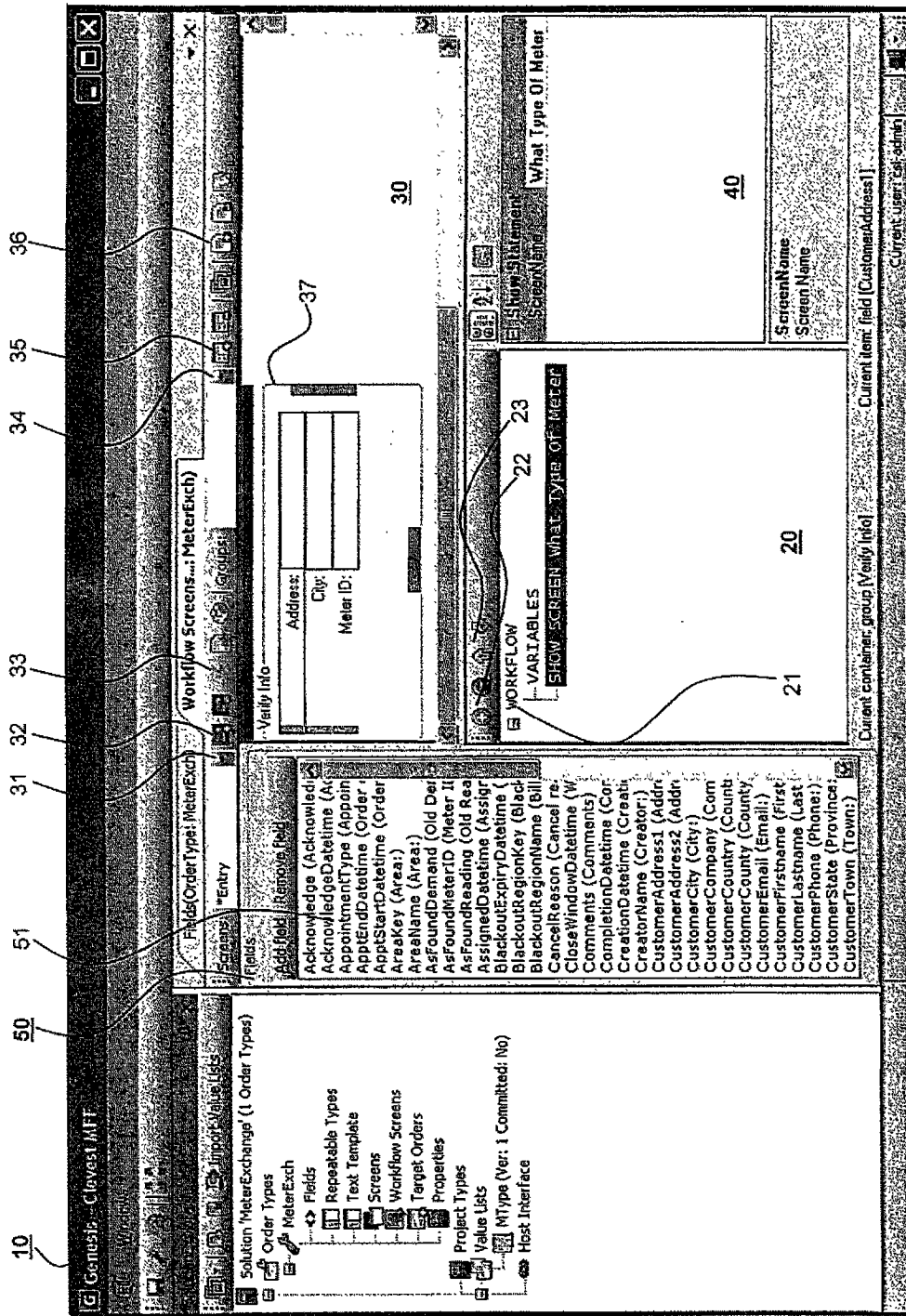


图 5

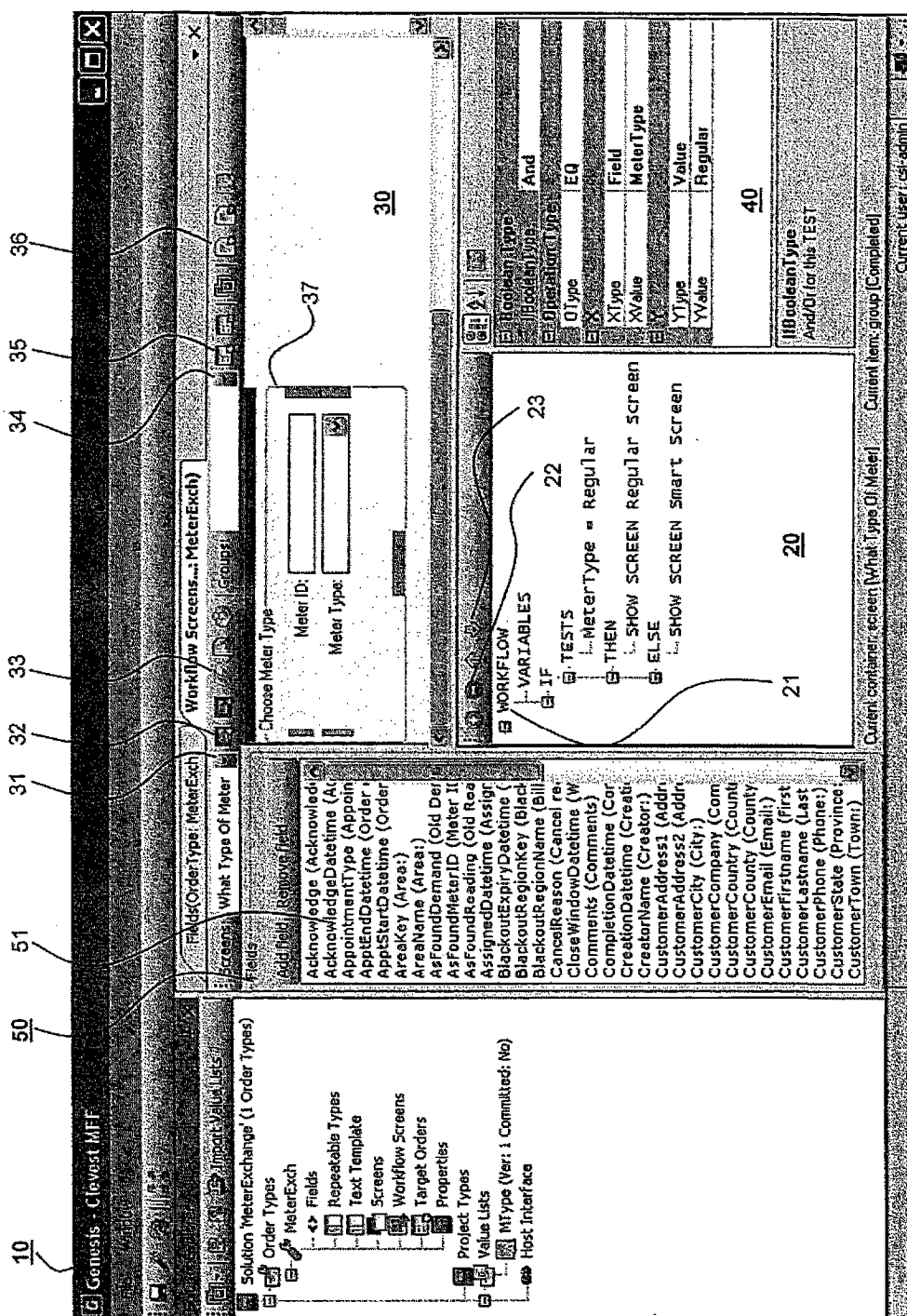


图 6

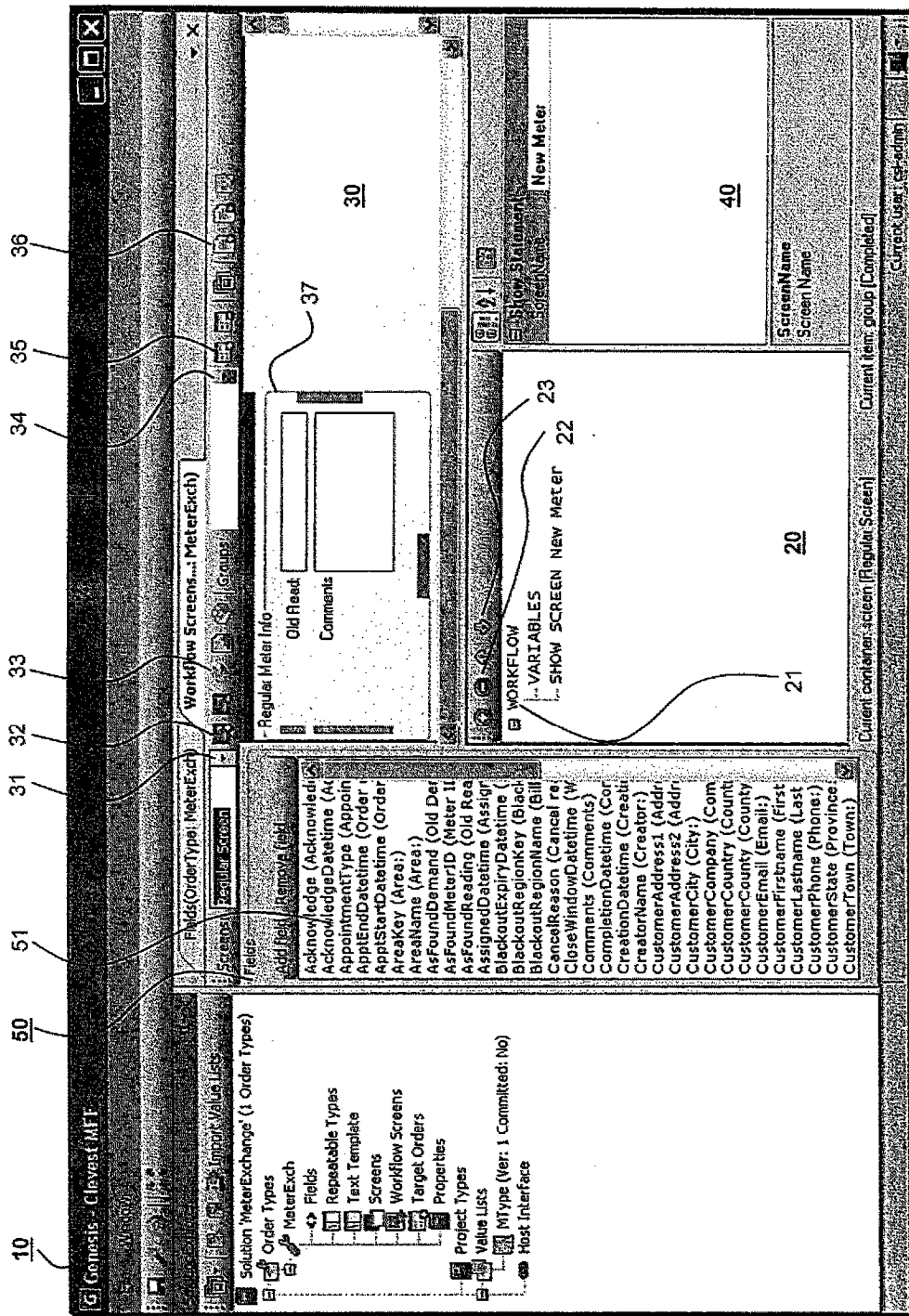


图 7

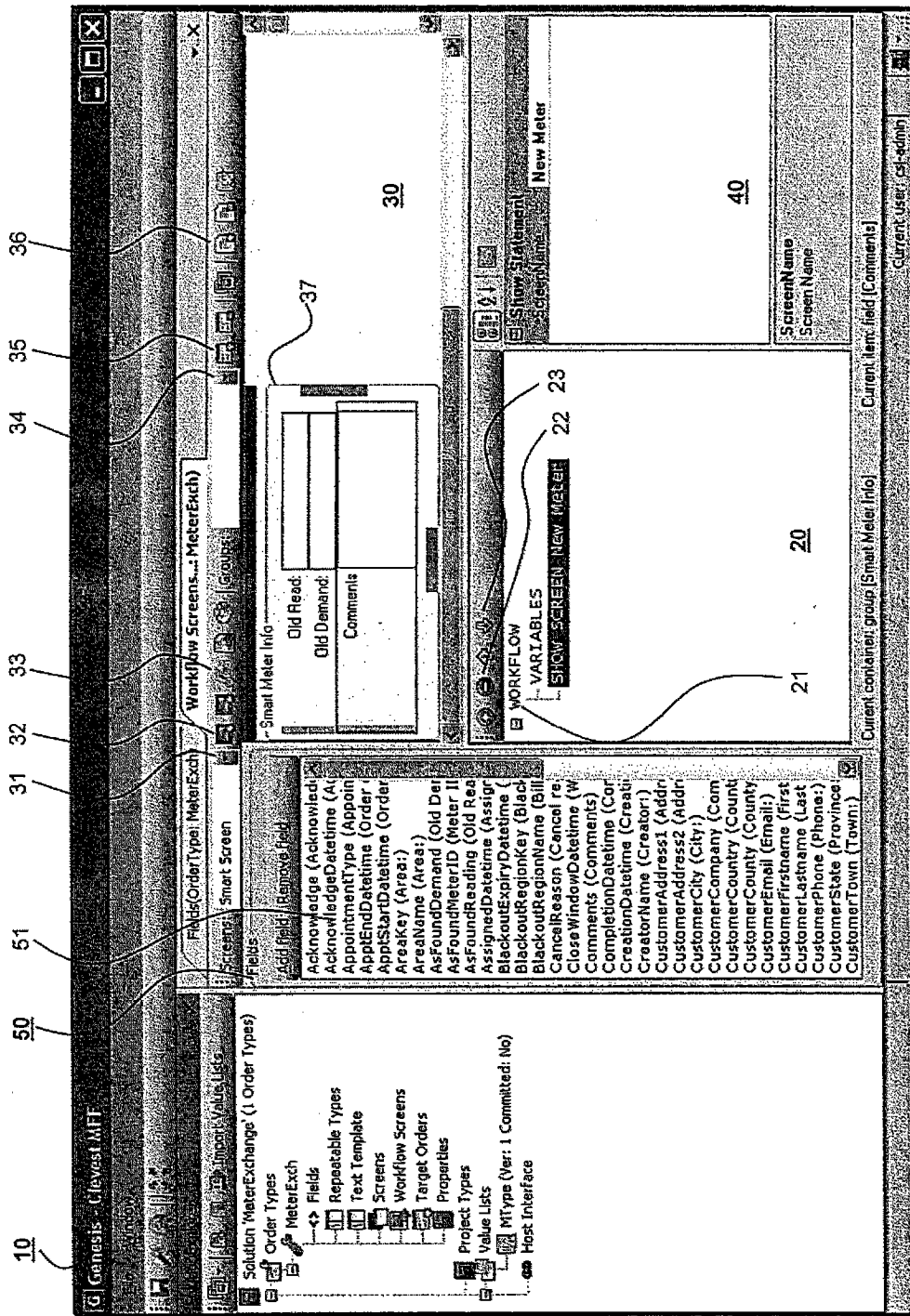


图 8

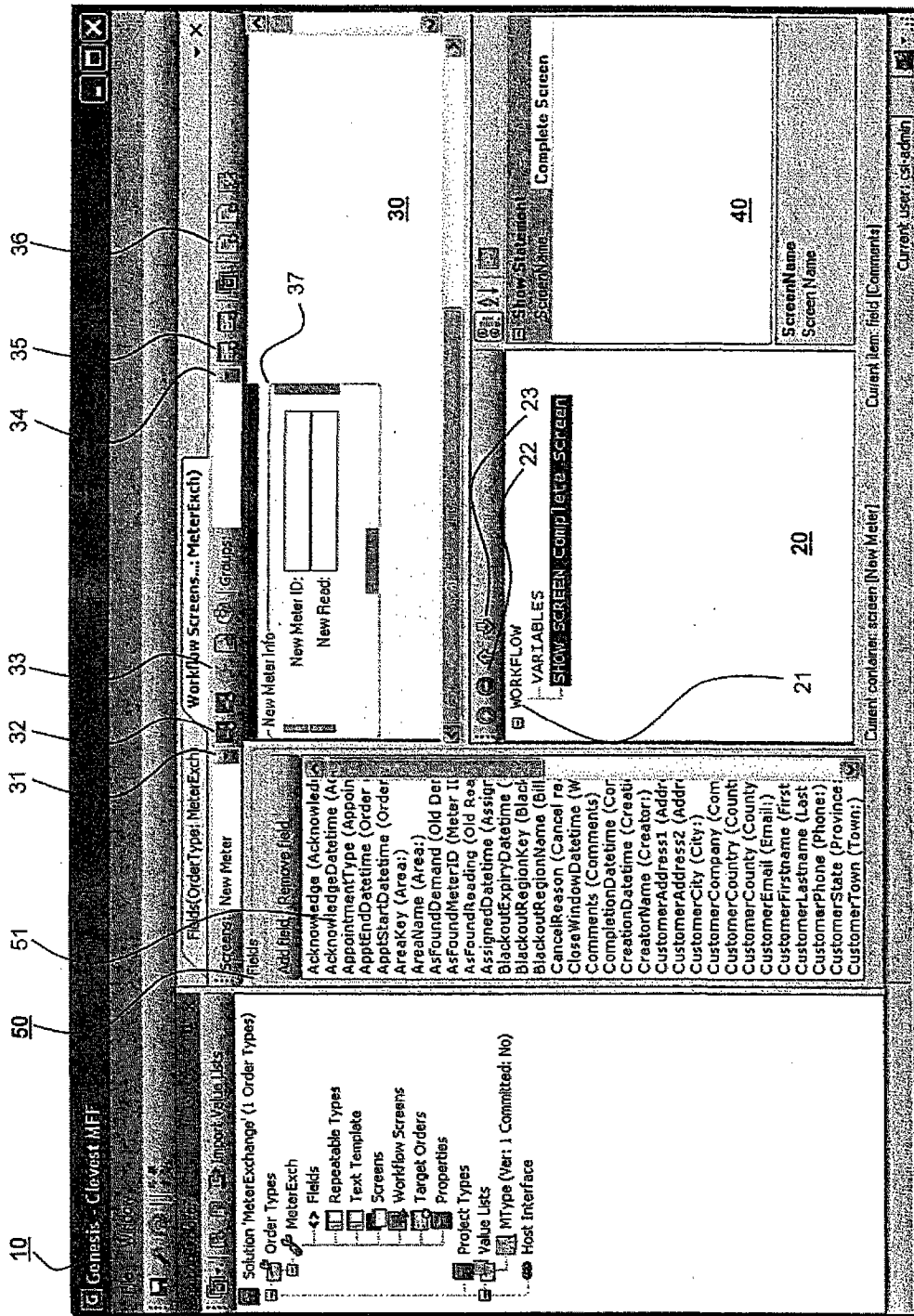


图 9

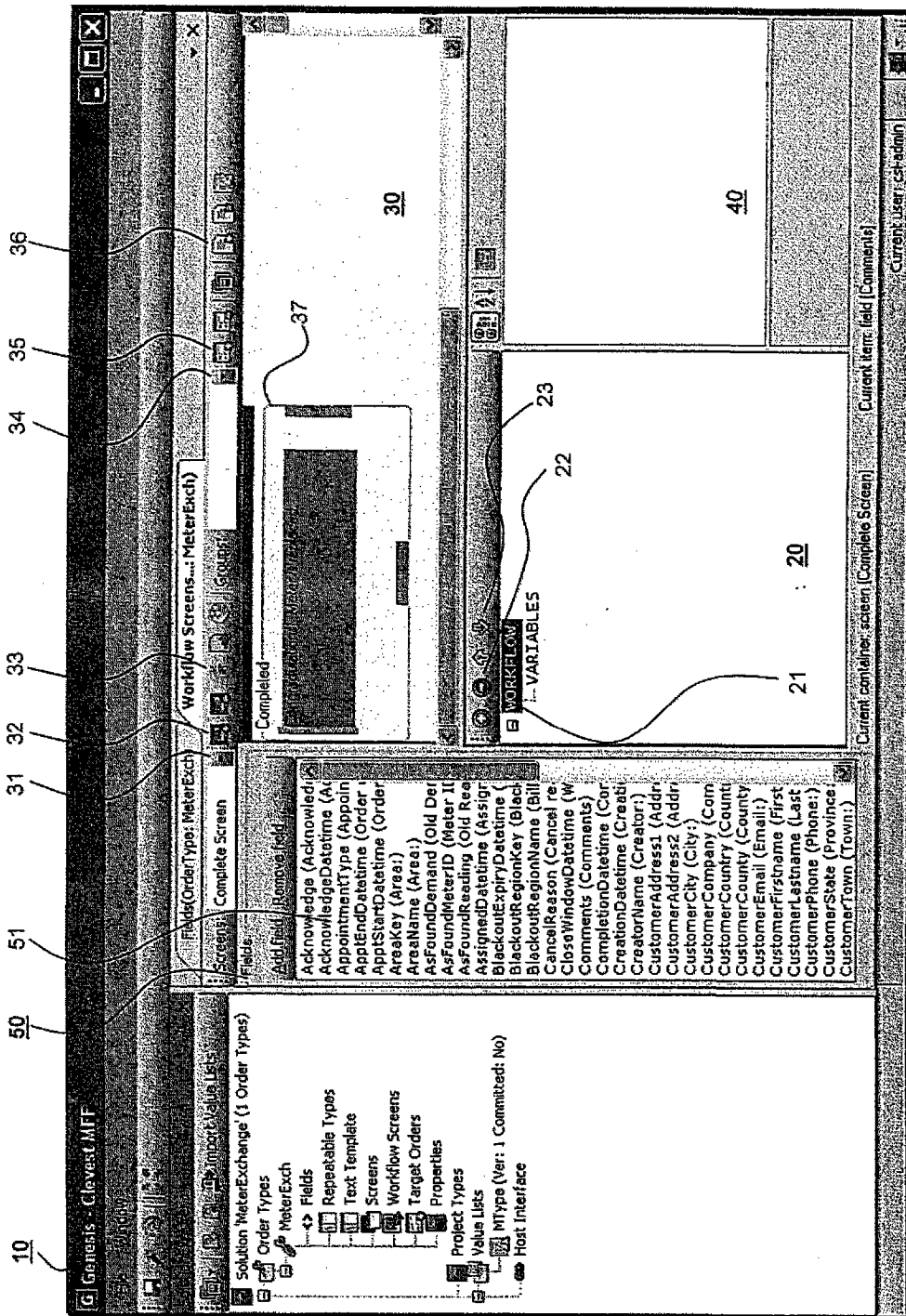


图 10

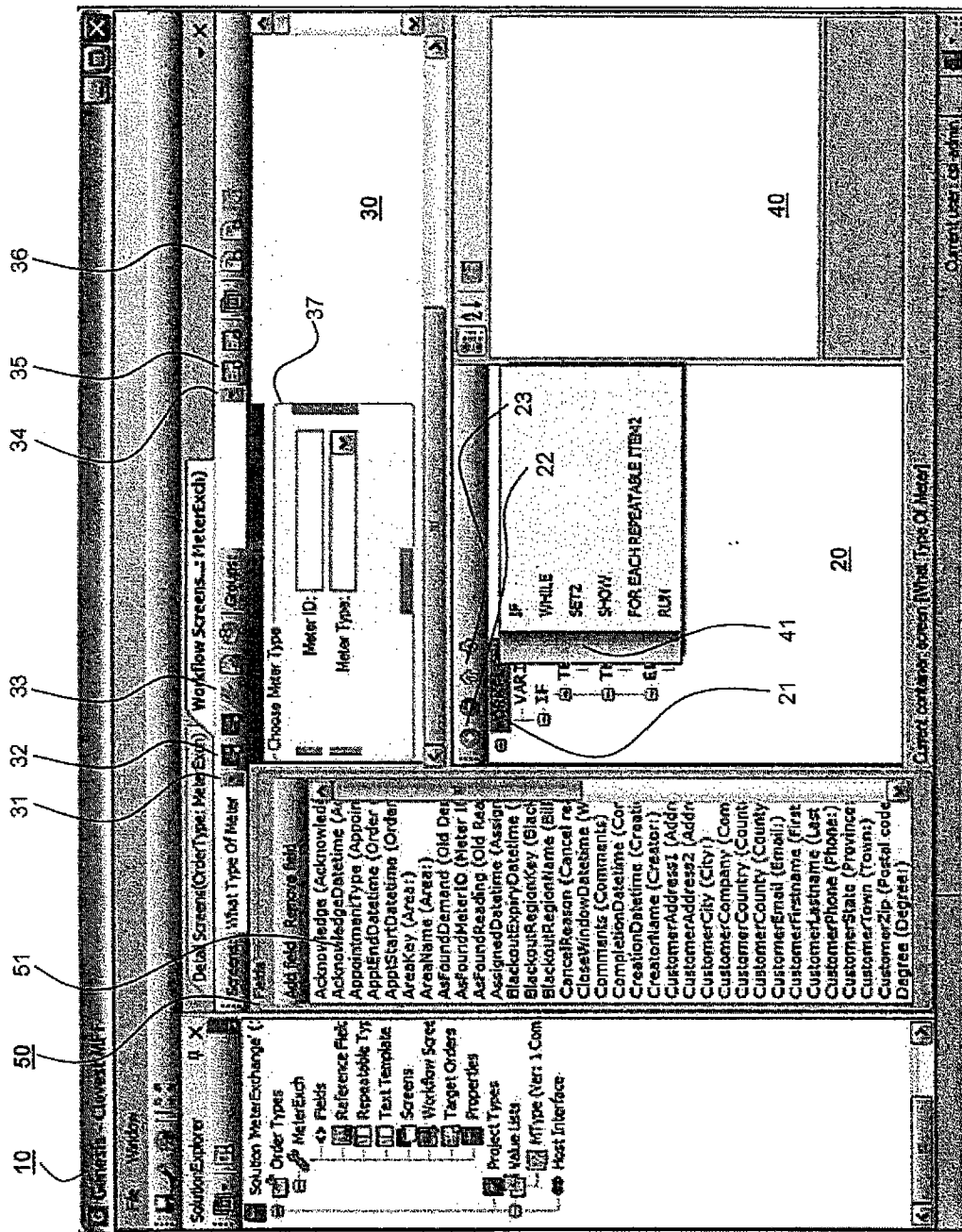


图 11

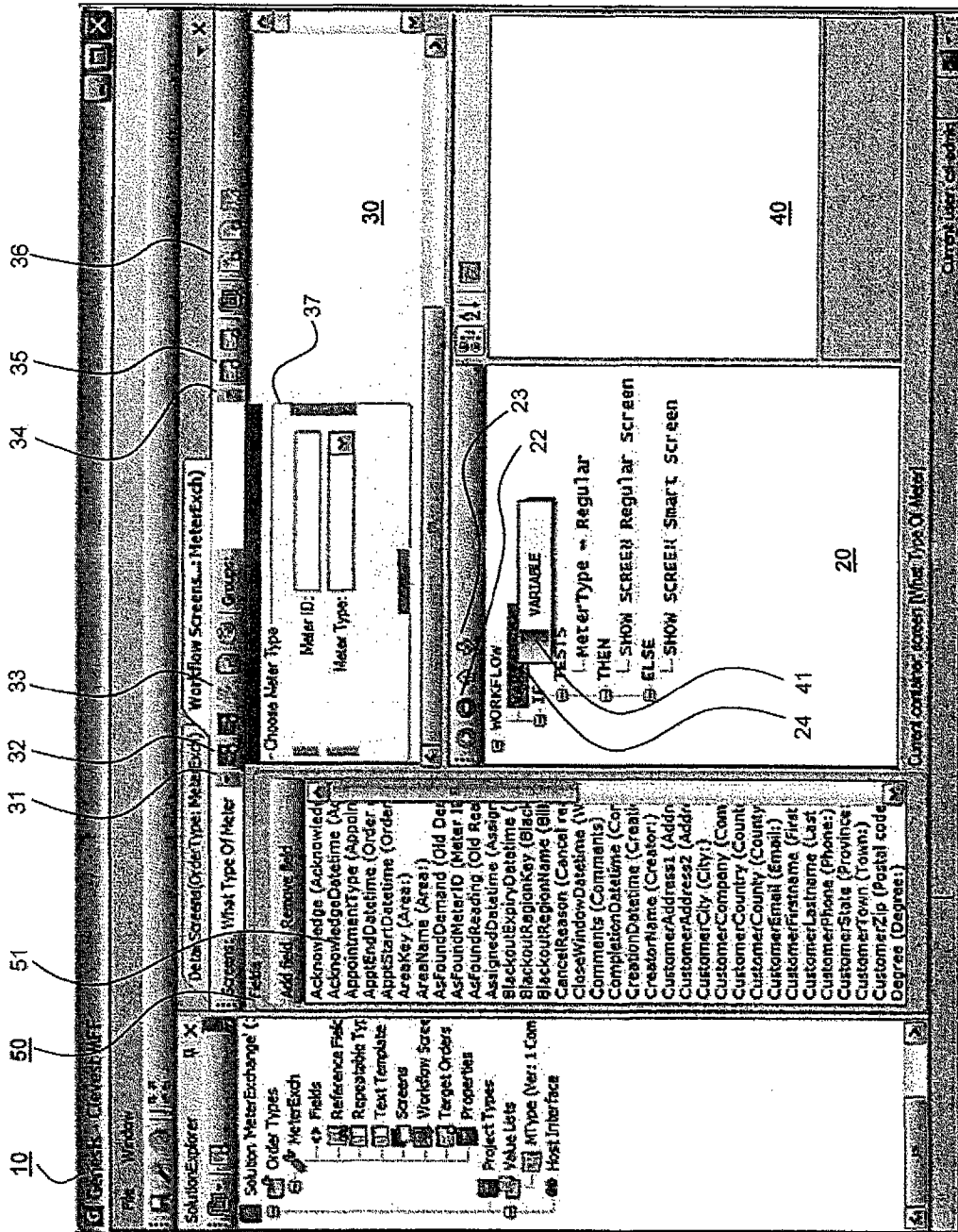


图 12

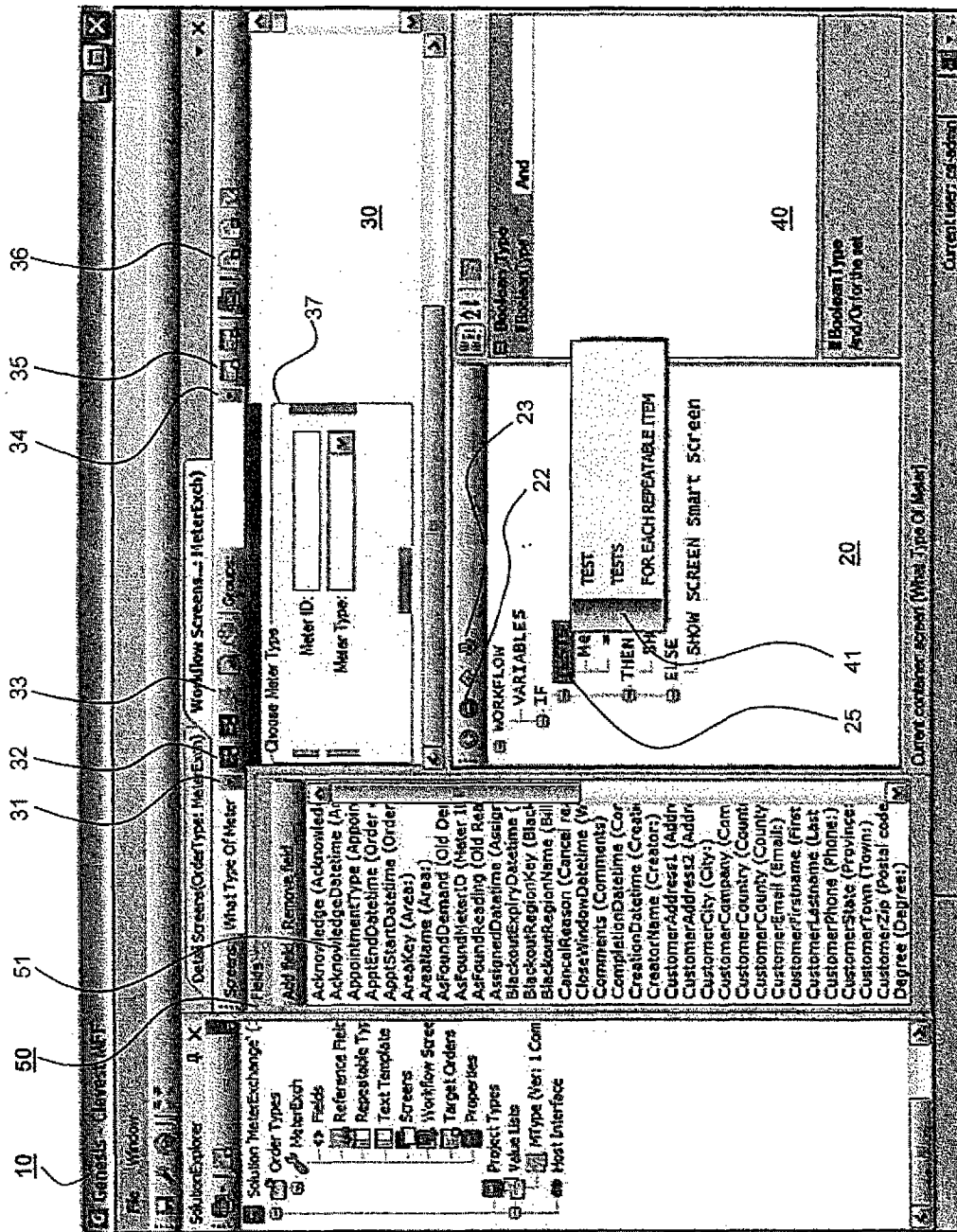


图 13

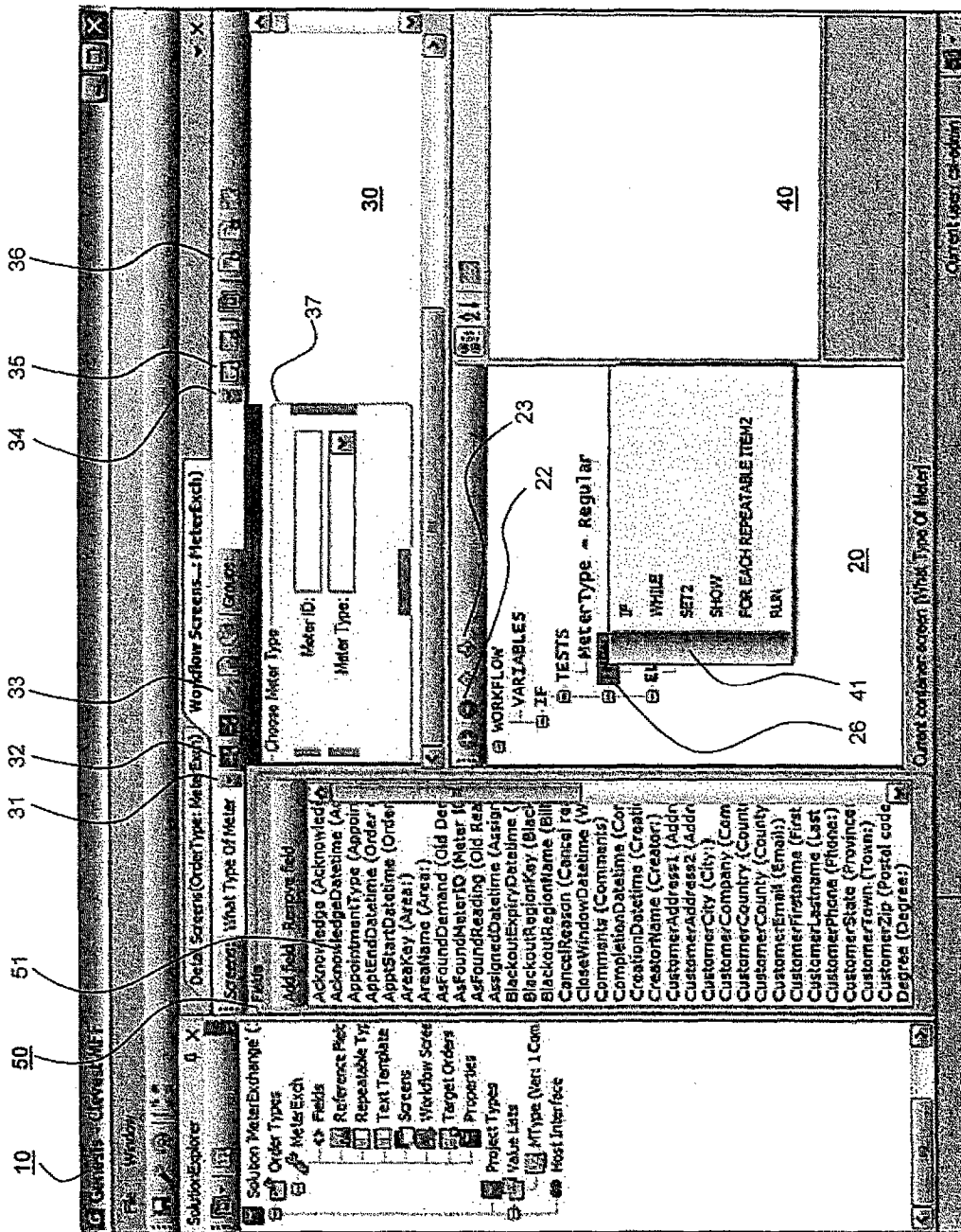


图 14

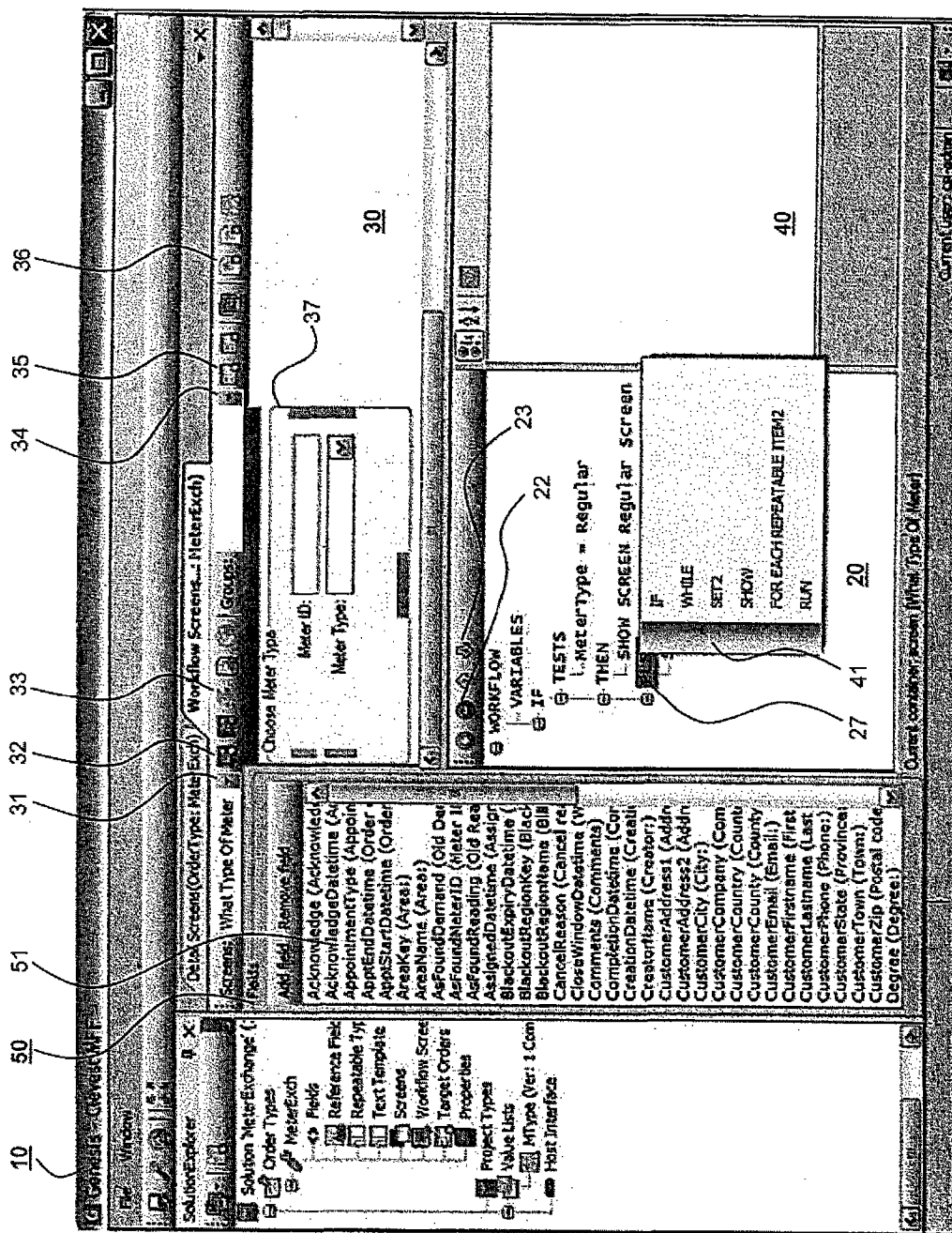


图 15

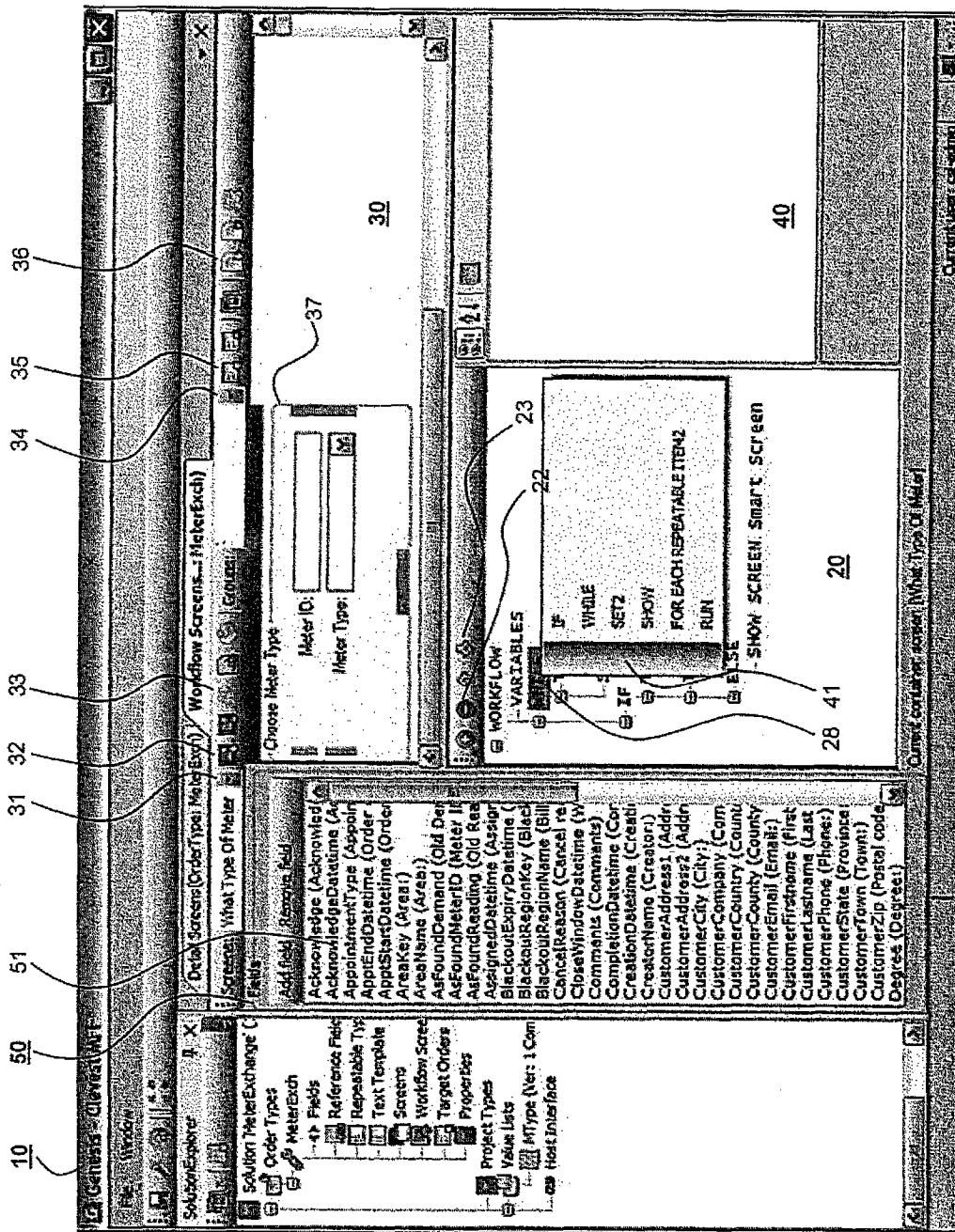


图 16

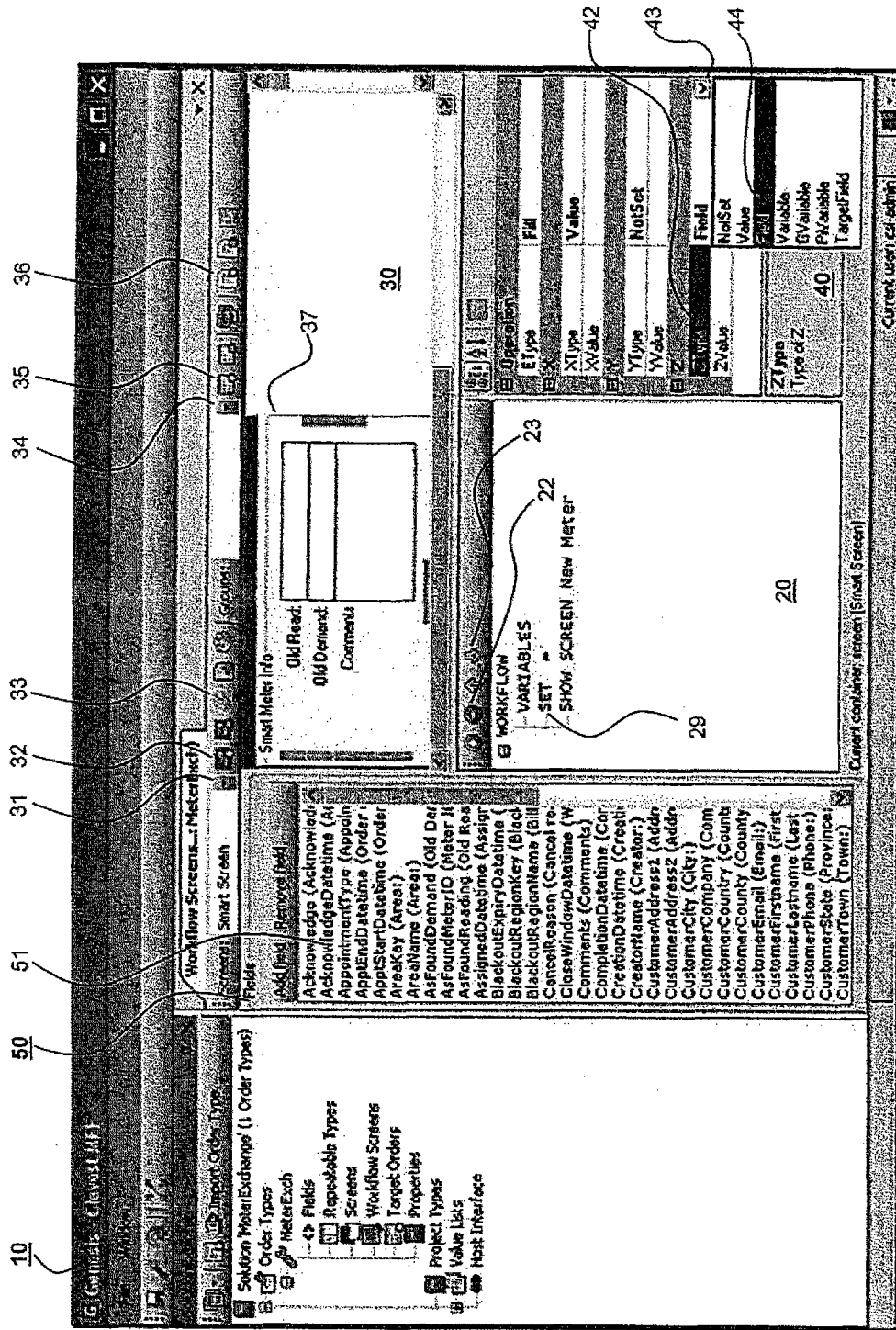


图 17

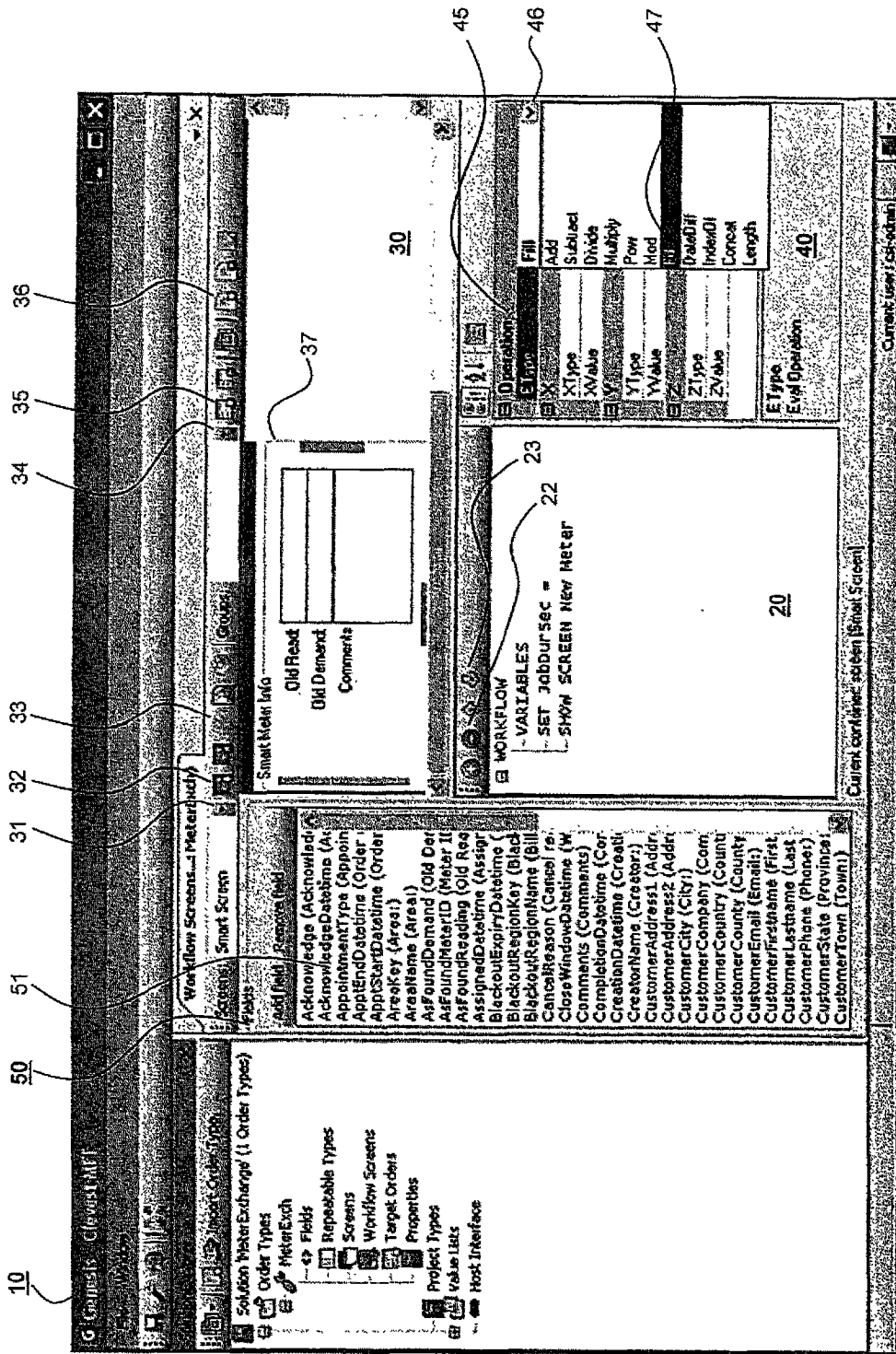


图 18

100

```

<wfscreen name="What Type Of Meter" type="Workflow" isstart="False" isJIT="False">
  <group name="Choose Meter Type" x="0" y="0" height="120" width="270">
    <screenitem type="Field" name="MeterType" mandatory="False" readonly="False" to="1"
    lbl="Meter Type:" lw="116" lh="20" lo="" sl="True" h="20" w="120" x="0" y="30" vls="s" ir=""
    lstst="DropDownCombo"/>
    <screenitem type="Field" name="AsFoundMeterID" mandatory="False" readonly="False"
    to="1" lbl="Meter ID:" lw="116" lh="20" lo="" sl="True" h="20" w="120" x="0" y="5" vls="s" ir=""
    lstst="DropDownCombo"/>
  </group>
  <variables></variables>
  <instructions>
    <if>
      <tests boolean="And">
        <test boolean="And">Field::MeterType[EQ]Value::Regular</test>
      </tests>
      <true>
        <show name="Regular Screen"/>
      </true>
      <false>
        <show name="Smart Screen"/>
      </false>
    </if>
  </instructions>
</wfscreen>

```

图 19

```

<while>
  <tests boolean="And">
    <test boolean="And">Variable::i[LT]Value::5</test>
  </tests>
  <set2>Variable::i = Variable::i(Add)Value::1</set2>
</while>

```

} 101

The above represents:
 While(i < 5)
 {
 i = i + 1;
 }

图 20

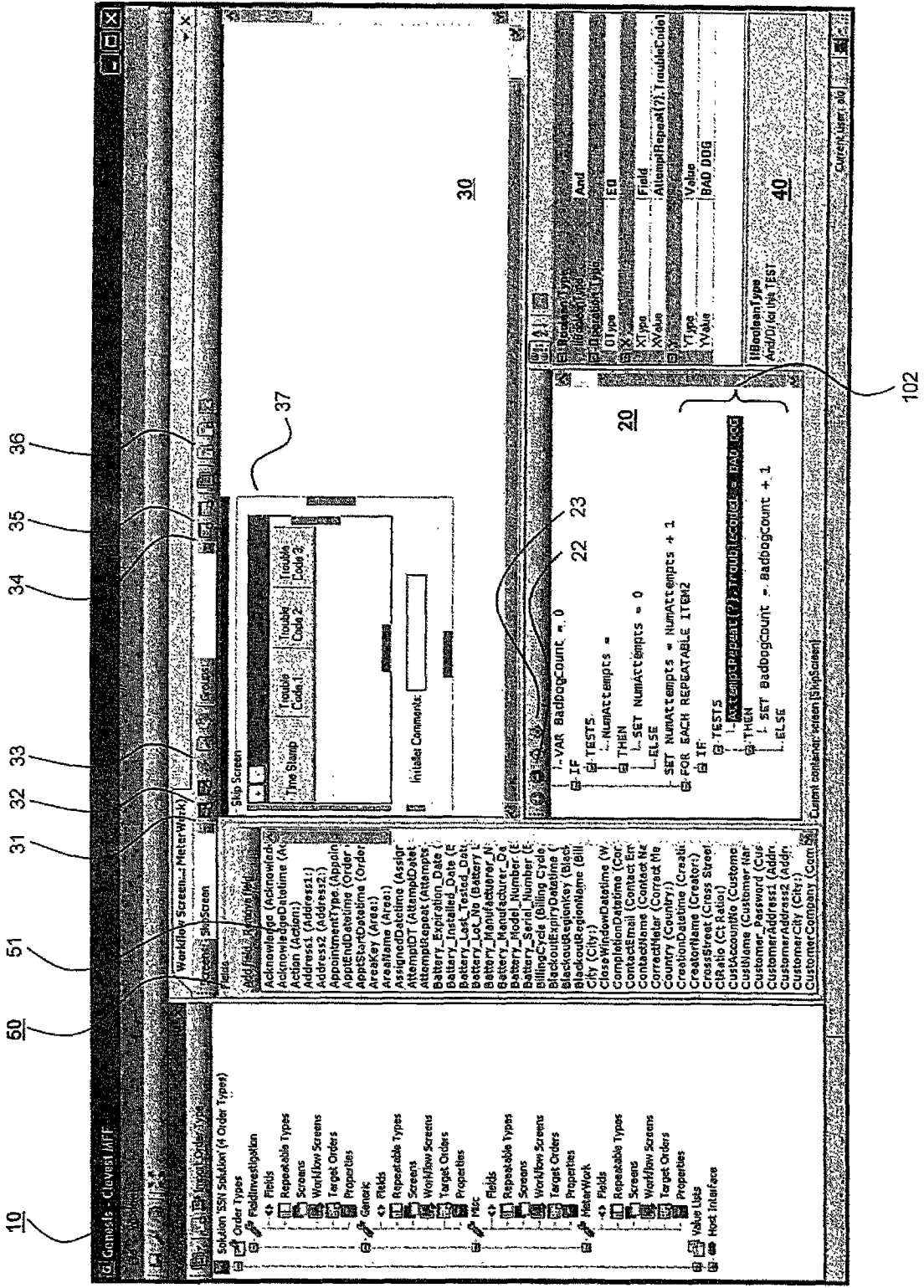


图 21

```

<wfscreen name="SkipScreen" type="Workflow" isstart="False" isJIT="False">
  <group name="Skip Screen" x="0" y="0" height="280" width="365">
    <screenitem type="Repeatable" name="AttemptRepeat" mandatory="False"
readonly="False" h="190" w="310" x="0" y="0" max="15" ct="RecordView">
      <screenitem type="Field" name="AttemptDT" mandatory="False" readonly="False"
to="1" lbl="Time Stamp" lw="116" lh="20" lo="1" sl="True" h="20" w="120" x="0" y="5" vls="s" ir=""
lstst="DropDownCombo"/>
      <screenitem type="Field" name="TroubleCode1" mandatory="False" readonly="False"
to="1" lbl="Trouble Code 1:" lw="116" lh="20" lo="1" sl="True" h="20" w="120" x="0" y="35"
vls="s" ir="" lstst="DropDownCombo"/>
      <screenitem type="Field" name="TroubleCode2" mandatory="False" readonly="False"
to="1" lbl="Trouble Code 2:" lw="116" lh="20" lo="1" sl="True" h="20" w="120" x="0" y="65"
vls="s" ir="" lstst="DropDownCombo"/>
      <screenitem type="Field" name="TroubleCode3" mandatory="False" readonly="False"
to="1" lbl="Trouble Code 3:" lw="116" lh="20" lo="1" sl="True" h="20" w="120" x="0" y="95"
vls="s" ir="" lstst="DropDownCombo"/>
    </screenitem>
    <screenitem type="Field" name="InstallerComments" mandatory="False"
readonly="False" to="1" lbl="Installer Comments:" lw="116" lh="20" lo="1" sl="True" h="20"
w="120" x="0" y="200" vls="s" ir="" lstst="DropDownCombo"/>
  </group>
  <variables>
    <var name="BadDogCount" type="" value="0"/>
  </variables>
  <instructions>
    <if>
      <tests boolean="And">
        <test boolean="And">Field::NumAttempts[EQ]Value::</test>
      </tests>
      <true>
        <set2>Field::NumAttempts = Value::0</set2>
      </true>
      <>false></false>
    </if>
    <set2>Field::NumAttempts = Field::NumAttempts(Add)Value::1</set2>
    <foreachitem2 repeatableField="">
      <if>
        <tests boolean="And">
          <test boolean="And">Field::AttemptRepeat{?}.TroubleCode1[EQ]Value::BAD
DOG</test>
        </tests>
        <true>
          <set2>Variable::BadDogCount = Variable::(Add)Value::1</set2>
        </true>
        <>false></false>
      </if>
    </foreachitem2>
  </instructions>
</wfscreen>

```

103

图 22