

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 763 935**

51 Int. Cl.:

G05B 23/02 (2006.01)

G06F 16/22 (2009.01)

G06F 16/25 (2009.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **03.08.2016 E 16182491 (7)**

97 Fecha y número de publicación de la concesión europea: **25.09.2019 EP 3279757**

54 Título: **Sistema y procedimiento para detectar fallas y/o para proporcionar medidas de optimización para mejorar el rendimiento de uno o varios edificios**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
01.06.2020

73 Titular/es:

**SIEMENS SCHWEIZ AG (100.0%)
Freilagerstrasse 40
8047 Zürich, CH**

72 Inventor/es:

**GÄRTNER, ROBERT y
COHEN, HENRIK**

74 Agente/Representante:

CARVAJAL Y URQUIJO, Isabel

ES 2 763 935 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Sistema y procedimiento para detectar fallas y/o para proporcionar medidas de optimización para mejorar el rendimiento de uno o varios edificios

Campo de la invención

5 La presente invención se refiere en general al campo técnico de sistemas de gestión de edificios o sistemas de automatización de edificios. En particular, la presente invención se refiere a sistemas y procedimientos para detectar fallas y/o para proporcionar medidas de optimización para mejorar el rendimiento de uno o varios edificios, especialmente los equipos de automatización de edificios de uno o varios edificios.

Antecedentes

10 La gestión de energía mediante un sistema de automatización de edificios (BAS) es la combinación de medidas adoptadas para maximizar la eficiencia en el uso de energía y optimizar el suministro de energía para reducir los costes de energía sin sacrificar las necesidades y la comodidad de los ocupantes. Para conseguir estos objetivos de rendimiento de los edificios, las medidas pueden incluir reducir el uso final de los equipos que consumen energía, aumentar la eficiencia de los equipos, eliminar el desperdicio energético, encontrar fuentes de energía alternativas menos costosas, minimizar los
15 costos de adquisición o la detección de fallas en los equipos del edificio.

La solicitud internacional WO2015/181338A1 revela un método asistido por ordenador y un sistema informatizado para proporcionar medidas de optimización o mejora para uno o más edificios, en el que, sobre la base de los datos de recursos relacionados con el edificio y los datos de rendimiento correspondientes, se determinan las medidas de mejora relacionadas con los medios consumibles en uno o más edificios mediante el uso de un sistema informático adaptado
20 para analizar los datos de recursos y los respectivos datos de rendimiento correspondientes basados en indicadores clave de rendimiento (KPIs) internos y/o externos proporcionados por una base de datos.

La solicitud de patente americana US2011/0178977A1 describe un controlador para un sistema de gestión de edificios configurado para analizar fallas en el sistema de gestión de edificios. El controlador detecta una falla en el sistema de gestión de edificios mediante la evaluación de los datos del sistema de gestión de edificios utilizando un sistema de reglas.
25 El controlador determina una probabilidad condicional para cada una de una pluralidad de posibles causas de falla dada la falla detectada. El controlador determina la causa de la falla más probable al comparar las probabilidades determinadas e informa electrónicamente de la causa más probable de la falla.

Las fallas en los equipos de los edificios causan efectos indeseables, como por ejemplo el aumento del consumo de energía o la disminución de la vida útil de los equipos. Se conocen varios procedimientos de detección de fallas, incluidos
30 los procedimientos para análisis estadístico y los procedimientos para el control estadístico de procesos. Estos métodos son a menudo complicados de manejar o usar en o con un BAS, o requieren experiencia en programación.

Por lo tanto, es necesario que los propietarios de edificios o los operadores de edificios dispongan de procedimientos o sistemas fáciles de usar para detectar fallas y/o para proporcionar medidas de optimización para mejorar el rendimiento de los edificios.

35 Resumen de la invención

El objeto de la invención es proporcionar procedimientos y sistemas para la detección y el diagnóstico de fallas que sean fáciles de utilizar por los operadores del edificio.

Este objeto será alcanzado por las reivindicaciones independientes.

Breve descripción de los dibujos

Los conceptos mencionados anteriormente y otros conceptos de la presente invención se abordarán ahora con referencia a los dibujos de las realizaciones preferente de la presente invención. Las realizaciones mostradas son para ilustrar, pero no limitan la invención. Los dibujos contienen las siguientes figuras, en las que los números similares se refieren a partes similares a lo largo de la descripción y de los dibujos y en los que:

5

FIGURA 1 ilustra un primer diagrama esquemático ejemplar de un sistema para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios que representan varios aspectos y realizaciones de la presente invención;

10

FIGURA 2 ilustra un segundo diagrama esquemático ejemplar de un sistema para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios que representa varios aspectos y realizaciones de la presente invención, incluyendo la integración del sistema a un sistema de gestión de edificios (BMS);

15

FIGURA 3 ilustra un tercer diagrama esquemático ejemplar de un sistema para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios que representa varios aspectos y realizaciones de la presente invención, incluida la integración de la unidad de procesamiento y la base de datos en la nube;

20

FIGURA 4 ilustra una primera base de datos ejemplar que almacena un modelo de datos que contiene datos de recursos y datos de rendimiento de uno o varios edificios, en la que el modelo de datos se representa en una o más estructuras de árbol jerárquicas según las realizaciones reveladas de la presente invención;

25

FIGURA 5 ilustra una segunda base de datos ejemplar que almacena un modelo de datos que contiene datos de recursos y datos de rendimiento de uno o varios edificios, en el que el modelo de datos se representa en una o más estructuras de árbol jerárquicas según las realizaciones reveladas de la presente invención;

30

FIGURA 6 ilustra una tercera base de datos ejemplar que almacena un modelo de datos que contiene datos de recursos y datos de rendimiento de uno o varios edificios, en el que el modelo de datos se representa en una o más estructuras de árbol jerárquicas según las realizaciones reveladas de la presente invención;

FIGURA 7 ilustra un diagrama general esquemático ejemplar de una herramienta de generación de scripts (guiones) para generar códigos ejecutables y/o módulos para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios;

35

FIGURA 8 muestra un diagrama de flujo ejemplar que representa un proceso para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios según las realizaciones reveladas de la presente invención;

FIGURA 9 ilustra una interfaz de usuario ejemplar para una herramienta de generación de scripts ejemplar;

FIGURA 10 muestra una lista ejemplar de configuraciones de tiempo de ejecución para configurar una herramienta de generación de scripts;

- FIGURA 11 ilustra una interfaz de usuario ejemplar de una herramienta de generación de scripts para definir campos calculados en puntos de datos;
- FIGURA 12 ilustra otra interfaz de usuario ejemplar para una herramienta de generación de scripts ejemplar;
- FIGURA 13 ilustra una interfaz de usuario ejemplar de una herramienta de generación de scripts para etiquetar scripts o reglas;
- FIGURA 14 ilustra una interfaz de usuario ejemplar de una herramienta de generación de scripts para la configuración de programación;
- FIGURA 15 ilustra una interfaz de usuario ejemplar de una herramienta de generación de scripts para el diagnóstico de fallas;
- FIGURA 16 ilustra una interfaz de usuario ejemplar para un editor de expresiones para definir reglas o parte de reglas; y
- FIGURA 17 muestra una interfaz de usuario ejemplar de un sistema para presentar los resultados según las realizaciones descritas.

Descripción detallada

Los "edificios inteligentes" generan mucho tráfico y una cantidad cada vez mayor de datos. Por lo tanto, hay que enfocarse en el rendimiento y la optimización de los sistemas de construcción, respectivamente subsistemas (por ejemplo, la distribución de aire, la refrigeración, la calefacción, la iluminación) y la infraestructura técnica individual o los equipos dentro de estos sistemas, como las unidades de tratamiento de aire, los enfriadores, los calentadores, etc. Esto requiere el análisis de grandes cantidades de datos de series temporales (por ejemplo, datos de rendimiento) y su correlación oportuna mediante reglas sofisticadas para encontrar, por ejemplo, ineficiencias en el funcionamiento, desviación de la estrategia de control o impactos negativos en la comodidad del edificio. Los ingenieros de sistemas o ingenieros de energía necesitan analizar los datos con frecuencia para medir, encontrar fallas o dar consejos para mejorar el rendimiento de los equipos, subsistemas, sistemas o edificios.

Por lo tanto, se espera que el usuario (por ejemplo, el propietario del edificio, el operador del edificio) conozca la configuración individual de los equipos en función de su tipo específico y los parámetros asociados, los puntos de datos disponibles, las programaciones operativas, etc. Además, el usuario debe alinear o normalizar la serie temporal de puntos de datos con respecto a la granularidad de análisis requerido, ya que la granularidad de datos sin procesar puede variar de un punto de datos a otro, por ejemplo, los datos de series temporales pueden estar disponibles como muestras de 1-, 5- o 15 minutos, y el análisis debe realizarse ventajosamente en intervalos de 15 minutos, lo que requeriría la compresión de series temporales granulares superiores a valores de 15 minutos basadas en una determinada función de compresión, como Suma/ Promedio/ Min/ Max. El usuario también debe asegurarse de que las unidades de medida de las series de datos se ajusten entre sí en el escenario de análisis para realizar cálculos correctos.

Una vez que se han realizado todas las tareas preparatorias y se ha definido completamente el contexto analítico y se han recopilado los datos para el período a analizar, el análisis actual puede comenzar definiendo la lógica que el usuario desea aplicar a los datos para detectar las anomalías o patrones que él está buscando. Esta lógica se puede llamar una regla y una colección de varias reglas, por ejemplo, para un cierto tipo de sistema o equipo se pueden reunir en un script. La ejecución de las reglas puede limitarse a ciertos tipos de recursos (edificios, sistemas, equipos, etc.), o la ejecución de las reglas puede diseñarse para un determinado recurso. Para apoyar el proceso de mitigación de fallas detectadas por los técnicos o gerentes de instalaciones, la herramienta de generación de scripts permite ventajosamente la asignación

de causas probables (esto es defectos o fallas) y acciones potenciales (medidas de optimización o mejora) que podrían tomarse en base a cada regla.

Ventajosamente, los scripts o reglas pueden ejecutarse incluso por usuarios ocasionales para toda la cartera de edificios, sistemas, subsistemas o equipos de un cliente a la vez, y se pueden aplicar a diferentes clientes sin tener que recrearlos o volver a generarlos. Los resultados se pueden presentar en varias formas, como un árbol de navegación que muestra los resultados agregados, una visión general de la rejilla y/o gráficamente. Para respaldar este objetivo, la herramienta de generación de scripts proporciona funcionalidad para configurar los datos publicados en función de cada script o base de reglas que comprende la definición de información agregada, como el número de fallas ocurridas y las duraciones de las mismas, y la capacidad de asignar las series de datos de tendencias que lo acompañan lo que puede ayudar al usuario a comprender mejor las condiciones reales en las que ocurrió la falla.

Hoy en día, la realización de tales tareas se lleva a cabo una a una por un operador humano aprovechando herramientas de hoja de cálculo como MS Excel o soluciones de scripting con lenguajes de scripting patentados e inspeccionando visualmente los gráficos de datos de tendencias para detectar anomalías y patrones. Alternativamente, un experto puede escribir y probar los scripts o reglas, ya que conoce los comandos necesarios y comprende el contexto de los datos consultados para elegir los operandos necesarios con el fin de obtener resultados útiles o las tareas a ejecutar. Si un operador humano (por ejemplo, un Facility Manager) quiere realizar consultas específicas o ejecutar trabajos que son relevantes para él, por ejemplo, para mejorar el rendimiento de su edificio o comparar edificios entre sí, tiene que convertirse en un experto o pedir a alguien que haga el trabajo de programación para él.

La Figura 1 ilustra un primer diagrama esquemático ejemplar de un sistema S1, especialmente un sistema FDD (Sistema de detección y diagnóstico de fallas) para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios B. El sistema ejemplar S1 contiene una base de datos DB que almacena un modelo de datos que incluye datos de recursos AD y datos de rendimiento PD de uno o varios edificios B, en el que el modelo de datos está representado en una o más estructuras de árbol jerárquicas. El sistema ejemplar S1 comprende además una interfaz de usuario UI (ventajosamente una interfaz de usuario guiada por menús UI) para configurar reglas para analizar los datos de recursos AD y los datos de rendimiento PD con respecto al rendimiento del edificio de uno o varios edificios B. Las reglas hacen referencia a mapeos de los puntos de datos, parámetros, mapeos de los datos de recursos, configuraciones de tiempo de ejecución o planificaciones que se configuran mediante una entrada de usuario utilizando operadores lógicos y funciones definidas. Las reglas se crean y configuran usando una unidad de entrada IU que está compuesta por teclado, ratón, etc., y la interfaz de usuario UI. Ventajosamente, la interfaz de usuario UI proporciona campos de entrada IPF1 - IPF3 para crear y configurar las reglas utilizando operadores lógicos (por ejemplo, "AND", "OR", "EQUAL", "GREATER", "SMALLER") y funciones definidas, por ejemplo, macros o subrutinas predefinidas.

El sistema ejemplar S1 consta además de una unidad de procesamiento PU para generar módulos de código ejecutables (por ejemplo, biblioteca de enlace dinámico DLL, "Dynamic Link Libraries") en tiempo de ejecución en base a las reglas configuradas por el usuario. Las reglas son ejecutadas por un sistema de tiempo de ejecución que atraviesa las estructuras de árbol del modelo de datos. En la ilustración según la figura 1, el generador de código de módulo CG comprende un motor de reglas para comprobar la sintaxis de las reglas y para ensamblar las reglas en un script ejecutable. El generador de código de módulo CG comprende además un módulo de generación de código para generar automáticamente módulos de código ejecutables (por ejemplo, archivos dll o archivos exe) para ejecutar el script en el respectivo sistema en tiempo de ejecución RS. El sistema en tiempo de ejecución RS está configurado para ejecutar los módulos de código generados (por ejemplo, archivos dll o archivos exe) en la unidad de procesamiento PU. La unidad de procesamiento PU puede ser, por ejemplo, un ordenador personal (PC) o una estación de trabajo. La unidad de procesamiento PU está equipada y

configurada con componentes de hardware y software apropiados. La PU también se puede implementar en un entorno de computación en la nube.

5 La base de datos DB que almacena el modelo de datos que comprende los datos de recursos AD y los datos de rendimiento PD de uno o varios edificios B puede integrarse en una memoria (por ejemplo, disco o memoria flash) de la unidad de procesamiento PU. La base de datos DB también podría almacenarse en una base de datos en memoria de la unidad de procesamiento PU. El almacenamiento de la base de datos DB en una base de datos en memoria de la unidad de procesamiento PU es especialmente ventajoso para aplicaciones que requieren tiempos de respuesta muy altos para proporcionar fallas al usuario.

10 En la realización ejemplar según la figura 1, la unidad de procesamiento PU tiene acceso a la base de datos DB a través de una interfaz de comunicación adecuada (por ejemplo, una conexión inalámbrica o por radio). La base de datos DB también podría implementarse en una nube. Esto permite compartir la base de datos de recursos DB con más de una unidad de procesamiento PU. Un proveedor de la base de datos de recursos DB (incluyendo el modelo de datos respectivo que comprende datos de recursos AD y datos de rendimiento PD de uno o varios edificios B) podría proporcionar el contenido de la base de datos DB para el cliente como software como un servicio (software-as-a-service).

15 Ventajosamente, el contenido de la base de datos DB significa que el modelo de datos que comprende los datos de recursos AD y los datos de rendimiento PD de uno o varios edificios B esta dispuesto en una o más estructuras de árbol cada una con una pluralidad de nodos y una pluralidad de hojas. Los nodos de las estructuras de árbol representan datos de recursos AD y las hojas de las estructuras de árbol representan datos de rendimiento PD de uno o varios edificios B. En la una o más estructuras de árbol, los datos de rendimiento PD están representados, en cada caso, por puntos de datos, los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación del equipo técnico (por ejemplo, equipo HVAC) de uno o varios edificios B.

20 Los datos de recursos AD de uno o varios edificios B comprenden información sobre la infraestructura técnica o el equipo técnico para el control de climatización (HVAC), y/o el control de acceso, y/o suministro y control de energía, y/o seguridad y/o protección contra incendios del uno o varios edificios B. Los datos de recursos AD de uno o varios edificios B incluyen, entre otros, uno o más de los siguientes: datos pasivos (por ejemplo, tipos y cantidades de sistemas, subsistemas, componentes empleados equipados en uno o varios edificios B), datos de configuración, datos meteorológicos, tarifas de energía (por ejemplo, precios de la energía) o factores de emisión (por ejemplo, CO₂). Dado que los datos de recursos AD de un edificio B no cambian con mucha frecuencia, los datos de recursos AD se pueden enviar a la base de datos DB, por ejemplo, mediante un lote de actualización (por ejemplo, diario, semanal, mensual) o si el edificio B está equipado con una pieza de dispositivo de infraestructura instalado o modificado recientemente.

25 Los datos de rendimiento PD de uno o varios edificios B comprenden: datos del sistema de control del edificio y/o datos de medición de electricidad y/o datos de medición de calor y/o datos de medición de agua y/o datos de temperatura y/o datos de humedad, y/o datos de presión, y/o eventos, y/o alarmas. Este listado no está limitado. Los datos de rendimiento PD con respecto al equipo técnico deben ser monitoreados permanentemente para obtener conclusiones fiables con respecto a la detección de fallas o defectos del equipo técnico en un edificio B. Ventajosamente, los datos de rendimiento PD se envían a la base de datos DB en una resolución entre 1 y 60 minutos, en particular en una resolución entre 1 y 15 minutos. Esto permite una respuesta rápida a las solicitudes de tiempo crítico o a las aplicaciones de tiempo crítico para proporcionar fallas detectadas F y/o medidas de optimización OM para un cliente.

35 Los datos de recursos AD y los datos de rendimiento PD de uno o varios edificios B se pueden enviar a la base de datos de forma ventajosa mediante una conexión de comunicación inalámbrica (por ejemplo, una conexión de radio adecuada). También se pueden usar conexiones de comunicación por cable adecuadas.

El sistema ejemplar S1 comprende además una unidad de salida OU para emitir fallas detectadas F y/o para proporcionar medidas de optimización OM en relación a uno o más edificios B. La unidad de salida OU podría ser una pantalla de un monitor de ordenador, una pantalla de una tablet, una pantalla de un dispositivo portátil (p. ej., un smartphone) o la pantalla de una estación de monitorización de un sistema de automatización de edificios o un sistema de gestión de edificios (p. ej., Desigo CC).

La Figura 2 ilustra un segundo diagrama esquemático ejemplar de un sistema S2 para detectar fallas F y/o para proporcionar medidas de optimización OM con el fin de mejorar el rendimiento de uno o varios edificios B. Comparado con el sistema S1 ilustrado en la figura 1, el sistema S2 ilustrado en la figura 2 está adicionalmente conectada o integrada a un sistema de gestión de edificios BMS. Un sistema de gestión de edificios BMS controla y/o monitorea el equipo técnico (por ejemplo, dispositivos mecánicos o eléctricos) y subsistemas (por ejemplo, sistemas de energía, sistemas de seguridad, sistemas de calefacción, sistemas de ventilación o sistemas de evacuación) de uno o varios edificios B. Ventajosamente, las fallas detectadas F y/o las medidas de optimización OM se ingresan directamente (en línea) en el sistema de gestión de edificios BMS para el funcionamiento de uno o varios edificios B. Las fallas detectadas F y/o las medidas de optimización OM también se pueden comunicar fuera de línea (por ejemplo, como un lote de actualización diaria) al sistema de gestión de edificios BMS. En función de las fallas detectadas F y/o de las medidas de optimización OM, el sistema de gestión de edificios BMS puede derivar o crear datos de mejoras ID en una granularidad adecuada para la infraestructura técnica respectiva afectada por las fallas detectadas F o las medidas de optimización OM.

En otra realización ventajosa, las medidas de optimización OM se introducen directamente en el respectivo equipo de automatización técnica afectado de uno o varios edificios B. Esto puede realizarse a través de medios de comunicación adecuados (por ejemplo, comunicación por Internet o radio), en modo en línea (en tiempo real) o en modo fuera de línea (modo por lotes). Mantener contacto directo en línea o en tiempo real con equipos de automatización técnica afectados o defectuosos reconocidos de uno o varios edificios B es especialmente ventajoso si las medidas de optimización derivadas OM son fáciles de implementar, por ejemplo, el encendido o apagado de un dispositivo o subsistema en uno o más edificios B.

La figura 3 ilustra un tercer diagrama esquemático ejemplar de un sistema S3 para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios B, que representa varios aspectos y realizaciones de la presente invención, especialmente la integración de la unidad de procesamiento PU y la base de datos DB en una nube C. Ventajosamente, la infraestructura de nube C se implementa en Internet y la base de datos DB se realiza como servidor de base de datos o servidor de archivos de acceso remoto. Ventajosamente, la unidad de procesamiento PU se realiza como un servidor informático que ofrece servicios de diagnóstico a los respectivos clientes (por ejemplo, operadores de edificios, propietarios de edificios, administradores de instalaciones).

La unidad de procesamiento PU y la base de datos DB del sistema FDD S3 pueden integrarse o formar parte de un sistema de plataforma de servicios que se realiza como una plataforma basada en la nube, en la que la infraestructura y/o la plataforma y/o los programas de software (o partes de la infraestructura o de la plataforma o del software) son proporcionados por un proveedor de servicios en la nube.

Ventajosamente, la base de datos DB comprende no sólo datos de recursos AD y datos de rendimiento PD de uno o varios edificios B, sino también precios de la energía EP proporcionados por una empresa de servicios públicos o por un proveedor de energía ES. Ventajosamente, la base de datos DB incluye también datos meteorológicos, datos de pronósticos de tiempo o datos relativos a los recursos consumibles (por ejemplo, agua, combustible) utilizados en uno o más edificios B. Los datos meteorológicos pueden ser proporcionados por un proveedor de datos meteorológicos o de pronóstico del tiempo. Esto puede ser un proveedor privado y/o público.

La figura 4 ilustra una primera base de datos ejemplar DB que almacena un primer modelo de datos ejemplar DM1 que comprende datos de recursos y datos de rendimiento de uno o varios edificios. El modelo de datos DM1 está representado en una o más estructuras de árbol jerárquicas, en donde los nodos AHU, AD1, AD2 de las estructuras de árbol representan datos de recursos y las hojas de las estructuras de árbol representan datos de rendimiento de uno o varios edificios. Los datos de rendimiento PD1 - PD6 del equipo técnico de uno o varios edificios están representados, en cada caso, por puntos de datos, en los que los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación del equipo técnico. El modelo de datos ejemplar DM1 muestra una estructura de árbol para una unidad de tratamiento de aire esquemática ejemplar AHU equipada en un edificio B. La unidad de tratamiento de aire esquemática ilustrativa AHU comprende dos subsistemas ejemplares. Por ejemplo, un subsistema de calefacción (por ejemplo, una caldera) y un subsistema de refrigeración (por ejemplo, un enfriador). El subsistema de calefacción y el subsistema de refrigeración están representados en el modelo de datos DM1 como datos de recursos AD1 respectivamente AD2. Por ejemplo, los datos de recursos AD1 representan el subsistema de calefacción, y los datos de recursos AD2 el subsistema de enfriamiento de la unidad de tratamiento de aire AHU. A los datos de recursos AD1 que representan el subsistema de calefacción se les asignan los datos de rendimiento PD1 - PD3. Los datos de rendimiento PD1 - PD3 representan puntos de datos ejemplares del subsistema de calefacción de la unidad de tratamiento de aire AHU. Los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación del subsistema de calentamiento ejemplar de la unidad de tratamiento de aire AHU. Los datos de rendimiento PD1 - PD3 del subsistema de calefacción AD1 pueden ser, por ejemplo, la temperatura del aire, la velocidad del aire y la humedad del aire. Los datos de rendimiento PD4 - PD6 asignados a los datos de recursos AD2 pueden ser, por ejemplo, las medidas correspondientes con respecto a la temperatura del aire, la velocidad del aire y la humedad del aire del subsistema de enfriamiento AD2 de la unidad de tratamiento de aire AHU.

Según la infraestructura equipada en uno o más edificios B, el modelo de datos ejemplar DM1 puede comprender más de una estructura de árbol. La herramienta de generación de scripts de la invención atraviesa o rastrea a través del modelo de datos DM1 para recopilar los datos de recursos AD1, AD2 y los datos de rendimiento PD1 - PD6 requeridos para las respectivas reglas creadas por un usuario utilizando una interfaz de usuario adecuada.

Las hojas PD1 - PD6 en una estructura de árbol del modelo de datos DM1 también pueden componerse de puntos de datos virtuales. Los puntos de datos virtuales pueden ser, por ejemplo, una serie temporal derivada. Un ejemplo de punto de datos virtual es, por ejemplo, la diferencia entre los valores de dos puntos de datos, por ejemplo, la diferencia de la temperatura de entrada y la temperatura de retorno. Mediante el uso de puntos de datos virtuales se puede aumentar la claridad de las reglas.

La Figura 5 ilustra una segunda base de datos ejemplar DB que almacena un segundo modelo de datos ejemplar DM2 que incluye datos de recursos y datos de rendimiento de un sitio o campus ejemplar del "Stephen Test". El sitio "Stephen Test" representa la raíz del modelo de datos ejemplar DM2. El modelo de datos ejemplar DM2 muestra en trazo grueso una estructura jerárquica de árbol con datos de recursos del sitio o campus "Stephen Test". El sitio o campus "Stephen Test" comprende un "Grupo de Edificio 1" que tiene un "Sitio 1". El "Sitio 1" está equipado con el "Equipo 1". Los datos de rendimiento asignados al "Equipo 1" no se muestran en la figura 5.

La segunda base de datos ejemplar DB puede comprender más de una estructura de árbol, dependiendo de la cantidad y tamaño de la infraestructura técnica del modelo de datos mapeado DM2 de uno o varios edificios.

En el modelo de datos, los datos meteorológicos DM2 (actuales, pronosticados) se pueden asignar a los nodos como datos de recursos o a las hojas de las estructuras de árbol como datos de rendimiento, según el enfoque o estrategia de modelado.

Ventajosamente, los datos de energía (precio de la energía EP, tarifa, escasez) se asignan en el modelo de datos DM2 a los nodos como datos de recursos. Los datos asignados a los nodos principales se heredan a los nodos secundarios y así sucesivamente hasta las hojas de las estructuras de árboles.

5 Ventajosamente, la estructura de árbol del modelo de datos DM2 comprende los siguientes niveles: Cliente, Grupo de Edificios, Sitio, Sistema y Equipo. Al implementar esto como un escenario estándar, hay una gran posibilidad de éxito para la implementación y ejecución, ya que los resultados de las configuraciones del cliente serán más predecibles.

Ventajosamente, el modelo de datos DM2 permite al usuario la capacidad de configurar diferentes niveles ilimitados de una estructura jerárquica (Ejemplo: Cliente > Región > Edificio > Piso > Zona > Empresa ACME, Fabricación ACME, Región ACME, Estado ACME, Campus ACME, Edificio ACME, Sistema, Piso, Zona, Punto).

10 Ventajosamente, la jerarquía de un árbol puede mostrar datos numéricos junto al nombre del nodo del árbol y el usuario podrá seleccionar qué parámetros de datos les gustaría ver (Ejemplo: Duración o Costo o Número de Fallas o Tendencias).

De forma ventajosa, el usuario puede seleccionar un campo de datos personalizado para agregar y mostrar en el árbol a la derecha de los nombres de nodo.

15 La figura 6 ilustra una tercera base de datos ejemplar DB que almacena un modelo de datos ejemplar DM3 que comprende datos de recursos y datos de rendimiento del uno o varios edificios, en el que el modelo de datos se representa en una o más estructuras de árbol jerárquicas según las realizaciones descritas de la presente invención. El modelo de datos DM3 representa un "Edificio Arnold" ejemplar que tiene el nodo raíz "Arnold - Bldg M". Al nodo raíz "Arnold - Bldg M" se asignan datos de rendimiento ejemplares "Humedad relativa del aire exterior Arnold" y "Temperatura del aire exterior Arnold". El "Edificio Arnold" comprende una infraestructura ejemplar para el tratamiento del aire que está representada en el modelo de datos DM3 con el nodo "tratamiento del aire". La infraestructura ejemplar para el tratamiento del aire comprende unidades de tratamiento del aire representadas en el modelo de datos DM3 con los nodos "AHU-M-1-1" a "AHU-M-4-1".
20 En la figura 6, el nodo "AHU-M-3-2" ilustra otras subestructuras ejemplares.

Las reglas solo se ejecutan en nodos en los que el tipo de nodo coincide con los tipos de nodo para los que la regla está habilitada, por ejemplo, una regla para la evaluación de AHU solo se ejecuta en los nodos "AHU-M -..." en el ejemplo según la figura 6 y no en los nodos VAV "Cajas VAV...". Los nodos respectivos en la estructura de árbol se identifican mediante metadatos, en este ejemplo por el tipo de nodo. Cualquier propiedad de metadatos de los recursos se puede utilizar para controlar el alcance de la ejecución de las reglas. Mientras se ejecuta una regla en un nodo relevante, los datos de rendimiento directamente relacionados con este nodo generalmente se utilizan para la evaluación de la expresión de la regla.
30

Los puntos de datos de rendimiento comunes como la "Humedad relativa del aire exterior" y la "Temperatura del aire exterior", que, por ejemplo, solo se definen a nivel del edificio, pueden tomarse como referencia para las reglas ejecutadas en los nodos de nivel inferior de la jerarquía de árbol. Esto se apoya en los principios de herencia/funcionalidad proporcionados por el sistema de tiempo de ejecución. Otra funcionalidad importante del sistema de tiempo de ejecución es la agregación. La agregación permite que una regla haga referencia a los datos de rendimiento de nodos de nivel inferior, por ejemplo, VAV para la ejecución en un nodo de nivel superior, por ejemplo, AHU. Se admiten varias funciones de agregación, por ejemplo, Promedio/Suma/Min/Máx.
35

A continuación, se muestra un ejemplo de una regla en relación con una unidad de tratamiento de aire ejemplar y el posible impacto de las fallas detectadas.

Ejemplo de la regla AHU:

Temperatura promedio de la zona es más baja que RAT, el promedio de la temperatura del aire de la zona para las zonas atendidas por la AHU es inferior a la temperatura del aire de retorno AHU

Expresión de la regla:

5 And(
 GT (SF_ STATUS, 0),
 LT (Average (Children (SPACE_TEMP)), Subtract (RAT,
 EPSILON AIR TEMP))
);

10 Impacto de las fallas detectadas:

Cuando la temperatura promedio del aire de la zona es más baja que la temperatura del aire de retorno AHU, podría indicar una fuga de aire caliente en el conducto de aire de retorno o un error de medición del sensor. Lo que podría dar como resultado un mayor uso de energía, especialmente durante la temporada de refrigeración (verano).

15 La Figura 7 ilustra un diagrama esquemático ejemplar de una herramienta de generación de scripts SB para generar un código ejecutable y/o módulos ejecutables para detectar fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios. La herramienta de generación de scripts SB contiene una base de datos DB para almacenar un modelo de datos que incluye datos de recursos y datos de rendimiento de uno o varios edificios o de un campus. En la base de datos DB, el modelo de datos se representa ventajosamente en una o más estructuras de árbol jerárquicas, en donde los nodos de las estructuras de árbol representan datos de recursos y las hojas de las estructuras de árbol representan datos de rendimiento de uno o varios edificios. En el modelo de datos, los datos de rendimiento se representan en cada caso por puntos de datos que comprenden series temporales de valores de campo medidos o derivados y metainformación (metadatos). Los metadatos de un punto de datos de un dispositivo instalado en un edificio pueden ser, por ejemplo, una identificación (ID) para el punto de datos, unidad de medida, objetivo (por ejemplo, la temperatura de entrada o la temperatura de retorno).

25 Ventajosamente, el modelo de datos incluye también datos meteorológicos (por ejemplo, las condiciones climáticas actuales, (por ejemplo, la temperatura, la humedad en el área local de uno o varios edificios), la información de las previsiones meteorológicas para el área local de uno o varios edificios). En el modelo de datos, los datos meteorológicos pueden asignarse a los nodos o a las hojas de las estructuras de árbol de acuerdo con la política de modelado subyacente. Si la política de modelado es principalmente un enfoque de arriba hacia abajo, los datos meteorológicos se asignan ventajosamente a los nodos. Si la política de modelado es principalmente un enfoque ascendente, los datos meteorológicos se asignan ventajosamente a las hojas. Por ejemplo, los datos meteorológicos pueden ser proporcionados por las oficinas meteorológicas o estaciones meteorológicas.

35 Ventajosamente, el modelo de datos contiene también datos de energía (por ejemplo, el precio de la energía, tarifas, escasez) proporcionados por una empresa de servicios públicos o por un proveedor de energía. Ventajosamente, los precios de la energía se asignan a los nodos, especialmente a los nodos de alto nivel, y el precio de la energía se heredará a los nodos y se deja debajo de este nodo. Un nodo de alto nivel puede ser, por ejemplo, un cliente que tiene un modelo de precio de energía específico acordado con una empresa de servicios públicos o un proveedor de energía. El modelo de precio de la energía específico es válido para todas las infraestructuras o dispositivos equipados en los edificios del cliente.

Además, la herramienta de generación de scripts SB consta de una interfaz de usuario UI (que comprende medios de entrada y visualización) para configurar reglas para analizar los datos de recursos y los datos de rendimiento con respecto al rendimiento del edificio de uno o varios edificios. Las reglas que se refieren a los mapeos de puntos de datos, parámetros, mapeos de datos de recursos, configuraciones de tiempo de ejecución o programaciones se configuran mediante la entrada del usuario utilizando operadores lógicos y funciones definidas. Las reglas son procesadas y realizadas por un motor de reglas adecuado RE. Un generador de código genera módulos de código ejecutables (por ejemplo, archivos dll o exe) en tiempo de ejecución según las reglas configuradas por el usuario. Las reglas son ejecutadas por un sistema de tiempo de ejecución RS atravesando las estructuras de árbol del modelo de datos y recogiendo y procesando los datos afectados por las reglas respectivas. Ventajosamente, la interfaz de usuario UI es una interfaz de usuario basada en menús o, en parte, una interfaz de usuario basada en menús.

Un script consta de 1 a n reglas, para cada script se genera un módulo de código ejecutable (por ejemplo, dll). La generación y ejecución de un módulo de código ejecutable se puede realizar en un solo paso. Ventajosamente, el módulo de código ejecutable (por ejemplo, dll) se genera y ejecuta en tiempo de ejecución mediante la compilación en tiempo de ejecución (JIT). El código generado es ejecutado directamente por el sistema de tiempo de ejecución RS. El sistema de tiempo de ejecución RS actúa como un host. La compilación en tiempo de ejecución (JIT) ofrece una gran flexibilidad para el usuario y un tiempo de ejecución rápido para el objeto generado o el código de máquina. Si no hay cambios en los puntos de datos de los respectivos datos de rendimiento generados, el objeto o el código de la máquina se puede reutilizar cuando se ejecuta un script una vez más.

Los scripts se pueden configurar mediante parámetros a los que se accederá durante el tiempo de ejecución y mediante la configuración del tiempo de ejecución. Los parámetros representan propiedades que son válidas para un nodo (es decir, un recurso, por ejemplo, sistema, subsistema, equipo técnico) y los nodos y hojas subyacentes. Los parámetros asignados a un nodo (padre) se heredarán a los nodos y hojas subyacentes. La configuración del tiempo de ejecución puede ser, por ejemplo, rangos de tolerancia que describen, por ejemplo, la receptibilidad del sistema FDD.

Ventajosamente, el motor de reglas RE, el generador de código CG y el sistema de tiempo de ejecución RS se implementan en módulos de software adecuados. El motor de reglas RE, el generador de código CG, el sistema de tiempo de ejecución RS y la base de datos DB se pueden cargar y almacenar en una memoria de la unidad de procesamiento PU y la unidad de procesamiento PU está configurada para ejecutar los módulos de software para el motor de reglas RE, el generador de código CG, y el sistema de tiempo de ejecución RS. Por ejemplo, la unidad de procesamiento PU puede ser un ordenador personal o una estación de trabajo. El motor de reglas RE, el generador de código CG y el sistema de tiempo de ejecución RS se pueden implementar en un módulo. También es posible implementar el motor de reglas RE, el generador de código CG y el sistema de tiempo de ejecución RS en cada caso en un módulo específico. Las funciones del motor de reglas RE y las funciones del generador de código CG también se pueden implementar en un módulo.

Ventajosamente, la unidad de procesamiento PU se realiza mediante una infraestructura de computación en la nube que contiene mecanismos para el almacenamiento de datos, el procesamiento de datos y la conexión de datos, especialmente para conexiones desde fuentes remotas y/o objetivos remotos.

Ventajosamente, la herramienta de generación de scripts SB incluye además una unidad de salida OU (por ejemplo, un monitor o una pantalla de una tablet) para emitir las fallas detectadas y/o para proporcionar medidas de optimización para uno o más edificios.

Ventajosamente, la herramienta de generación de scripts SB incluye además una interfaz para entregar fallas detectadas y/o medidas de optimización a un sistema de automatización de edificios que funciona en uno o más edificios.

Ventajosamente, la herramienta de generación de scripts SB incluye además una interfaz para entregar fallas detectadas y/o medidas de optimización directamente a la infraestructura técnica afectada de uno o varios edificios.

La entrega de las fallas detectadas y/o de las medidas de optimización se pueden realizar en tiempo real o en un lote de actualización (por ejemplo, por hora o por día).

5 Existen diferentes modos de funcionamiento de la herramienta de generación de scripts SB. La herramienta de generación de scripts SB puede proporcionar fallas y/o medidas de optimización a petición, por ejemplo, de un propietario de un edificio que se da cuenta que la factura de energía de su edificio parece ser demasiado alta. La herramienta de generación de scripts SB también puede funcionar en modo por lotes, por ejemplo, con una base diaria, semanal o mensual. En el modo por lotes, la herramienta de generación de scripts SB realiza un lote con un conjunto de reglas regularmente en un punto de tiempo definido. La herramienta de generación de scripts SB también se puede utilizar en modo en línea. El modo en línea tiene sentido para la monitorización o verificación de los equipos técnicos que deben proporcionar una alta fiabilidad.

15 Ventajosamente, la herramienta de generación de scripts SB puede agregar la métrica de fallas, la cantidad de fallas, la duración y el coste a cualquier objeto nodo de la jerarquía de árbol del sistema (por ejemplo, un sitio sumaría la cantidad de fallas, la duración de las fallas y las penalizaciones de costes en y debajo del nodo del sitio).

Ventajosamente, la herramienta de generación de scripts SB permite a un administrador definir un campo de datos personalizado que se puede agregar a cualquier objeto nodo de la jerarquía de árbol del sistema (por ejemplo, un campo de datos sería la duración o el coste).

20 Ventajosamente, la herramienta de generación de scripts proporciona la capacidad de apilar fallas repetidas para mostrar solo una línea de esas fallas y proporcionar una indicación de fallas repetidas (por ejemplo, si la misma falla ocurre 1000 veces, el sistema apilará las fallas ocurridas y mostrará una línea para esta falla).

25 La herramienta de generación de script SB permite a los usuarios (por ejemplo, al administrador) crear sus propias reglas y activarlas para equipos específicos. Por ejemplo, un administrador es la persona responsable de configurar el ajuste del generador de scripts SB para un cliente. El administrador también puede ser un cliente usuario avanzado que desee implementar un sistema FDD (detección y diagnóstico de fallas) por su cuenta.

Ventajosamente, la interfaz de usuario UI de la herramienta de generación de guiones SB proporciona una página de configuración y una página de detalles. La página de configuración permite al usuario definir las condiciones que deben ser verdaderas para que se produzca la excepción de la regla. La página de detalles permite al usuario definir toda la información asociada a la regla.

30 Ventajosamente, la herramienta de generación de scripts SB proporciona una interfaz de usuario UI para cualquier usuario sin conocimientos de programación para crear nuevas reglas utilizando funciones matemáticas y estadísticas comunes. Por ejemplo, esto se puede activar por medio de un entorno de programación de arrastrar y soltar o desplegable que permite ensamblar expresiones o reglas de simples a complejas simplemente arrastrando los datos disponibles y las funciones matemáticas o lógicas deseadas juntas para producir cualquier combinación permitida.

35 Características (entre otras) compatibles con la herramienta de generación de scripts SB son:

- La herramienta de generación de scripts SB ofrece opciones generales de configuración a nivel de script que pueden ser utilizadas por todas las reglas, incluidas las funciones administrativas, para administrar scripts.

- La herramienta de generación de scripts SB proporciona la configuración de reglas para el script, incluyendo su expresión de evaluación, seleccionar los puntos de datos que se mostrarán junto con el resultado de la falla y la configuración de tiempo de ejecución relacionada.
- La herramienta de generación de scripts SB permite al usuario definir las configuraciones de tiempo de ejecución (valores de entrada) para controlar la ejecución de la regla. Los valores pueden ser establecidos por el usuario final antes de ejecutar un script.
- La herramienta de generación de scripts SB permite definir campos calculados accesibles en puntos de datos y etiquetar y establecer opciones de publicación para la cuadrícula de resultados y/o el árbol de resultados
- La herramienta de generación de scripts SB permite seleccionar o asignar tipos de puntos de datos y propiedades de nodo/recursos relevantes para el script y definir los puntos de datos virtuales necesarios (calculados/derivados).
- La herramienta de generación de scripts SB permite seleccionar los tipos de nodo aplicables para el script/reglas y definir además filtros basados en las propiedades del nodo/recursos para el script completo o las reglas individuales.
- La herramienta de generación de scripts SB permite establecer las planificaciones de los scripts/reglas para ciertos nodos/recursos. Solo las fallas detectadas dentro de estas planificaciones se reconocerán como fallas y se mostrarán al usuario. El principio de herencia se aplica a las planificaciones, lo que significa que también se toman para los nodos inferiores cuando se definen en un nodo superior en el árbol de recursos.
- La herramienta de generación de scripts SB proporciona las causas probables y sugiere acciones (de mitigación) para las reglas. Esto ayuda al usuario a saber qué hacer si se produce una determinada falla.
- La herramienta de generación de scripts SB permite definir la expresión de evaluación (lógica) de la regla apoyada por las listas de selección para funciones personalizadas disponibles, los puntos de datos definidos (virtuales) y las propiedades de los nodos/recursos desde el nivel de script, así como las configuraciones de tiempo de ejecución, los campos calculados y los parámetros globales.
- La herramienta de generación de scripts SB proporciona un concepto simple de función-en-función-en-función que ayuda al usuario a unir la lógica.
- Para diferentes recursos se pueden usar las mismas reglas si los recursos tienen los mismos puntos de datos. Por ejemplo, dos unidades de tratamiento de aire que tienen los mismos puntos de datos pueden ser analizadas por el mismo conjunto de reglas. Esto proporciona una alta reutilización de reglas o scripts.

La herramienta de generación de scripts SB es una herramienta para apoyar al operador humano (ingeniero en energía, de sistemas, de aplicaciones) en su trabajo diario, sin conocimientos avanzados de un lenguaje de programación específico o consultando a un programador de software. Esto ahorra tiempo, permite al usuario enfocarse en la lógica de evaluación y conduce a scripts individuales que pueden aprovechar todos los usuarios autorizados y que se pueden aplicar a muchos clientes diferentes y a su cartera de edificios.

La cantidad de tiempo para definir nuevas reglas y scripts se reduce drásticamente, ya que los ingenieros pueden trabajar en un entorno en el que encuentran terminología relevante para el dominio técnico y pueden usar expresiones simplificadas soportadas por un diseño "función-en-función-en-función ..." para describir la lógica, en lugar de tener que desarrollar programas complejos. Los usuarios con un buen conocimiento de su dominio técnico pueden usar el generador de scripts SB, no es necesario que ningún programador se involucre en la codificación de las reglas. En ese sentido, los ingenieros son autosuficientes en su trabajo diario. Al crear nuevas reglas o realizar modificaciones a las reglas existentes, los cambios están directamente disponibles para la próxima ejecución en el entorno de tiempo de ejecución, una vez que se ha guardado el script. No se requieren pasos adicionales para la compilación, etc. (Cámbielo, guárdelo, ejecútelo).

Los scripts/reglas personalizados para clientes individuales se pueden crear y asignar fácilmente a los clientes, ya que todo está basado en datos/expresiones y almacenado en una base de datos global. La funcionalidad del generador de scripts en sí puede incluso habilitarse para el propio cliente. El coste que supone la creación de nuevas reglas para los sistemas existentes o nuevos tipos de sistemas se reduce al mínimo. Con la integración en un sistema de plataforma de servicios (por ejemplo, la plataforma Siemens Navigator), el tiempo de entrega al cliente es muy corto, las tareas se pueden distribuir a muchas personas. Esto reduce el coste y aumenta la eficiencia y el rendimiento al cliente para proporcionar scripts y reglas personalizadas.

La Figura 8 muestra un diagrama de flujo ejemplar que representa un proceso para un método facilitado por ordenador para detectar fallas, proporcionando medidas de optimización, o las dos anteriores para mejorar el rendimiento de uno o varios edificios, incluido el equipo de automatización de edificios dispuesto en relación con uno o más edificios, el método comprende:

(S1) proporcionar una base de datos que almacena un modelo de datos que incluye datos de recursos y datos de rendimiento de uno o varios edificios, en el que el modelo de datos está representado en una o más estructuras de árbol jerárquicas, cada una de las cuales tiene una pluralidad de nodos y una pluralidad de hojas, en el que los nodos de las estructuras de árbol representan datos de recursos y las hojas de las estructuras de árbol representan datos de rendimiento de uno o varios edificios, en el que los datos de rendimiento están representados en cada caso por puntos de datos, en el que los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación;

(S2) proporciona una interfaz de usuario para configurar reglas para analizar los datos de recursos y los datos de rendimiento en relación con el rendimiento del edificio de uno o varios edificios, en el que las reglas se refieren a mapeos de puntos de datos, parámetros, mapeos de datos de recursos, configuraciones de tiempo de ejecución, o planificaciones y se configuran por una entrada de usuario utilizando operadores lógicos y funciones definidas;

(S3) generar módulos de código ejecutables en tiempo de ejecución basados en las reglas configuradas por el usuario por medio de una unidad de procesamiento, en el que las reglas son ejecutadas por un sistema de tiempo de ejecución que atraviesa las estructuras de árbol del modelo de datos; y

(S4) emitiendo fallas detectadas, medidas de optimización o las dos mencionadas anteriormente con respecto a uno o más edificios en un dispositivo de salida.

Ventajosamente, la ejecución del script se realiza de acuerdo con la configuración de tiempo de ejecución. Las configuraciones de tiempo de ejecución, por ejemplo, pueden ser umbrales definidos por el usuario, límites superiores o límites inferiores con respecto a las series temporales en los puntos de datos. La configuración del tiempo de ejecución ofrece una parametrización definida por el usuario para ejecutar el método. Por ejemplo, solo los datos entre un límite superior y uno inferior podrían ser utilizados por el generador de scripts. Esto permite, por ejemplo, una presentación rápida de resultados.

El recorrido ventajoso de las estructuras de árbol del modelo de datos se realiza de acuerdo con la metainformación de los puntos de datos. Los metadatos para un punto de datos de un dispositivo instalado en un edificio pueden ser, por ejemplo, una identificación (identificador) para el punto de datos, unidad de medida, finalidad (por ejemplo, temperatura de entrada o temperatura de retorno). Al conocer los metadatos relevantes o la información semántica en la serie temporal de los puntos de datos que atraviesan la estructura de árbol, se puede realizar de manera muy efectiva ya que el generador de scripts solo tiene que recopilar los datos necesarios o requeridos para realizar una regla.

Ventajosamente, las medidas de optimización generadas comprenden períodos de falla, puntos de datos seleccionados, columnas de salida o diagnóstico de fallas. Esto le da al cliente flexibilidad y posibilidad de elegir los datos de salida y/o el formato de los datos de salida que necesita para sus necesidades.

5 En una realización ventajosa, las medidas de optimización se introducen en un sistema de gestión de edificios para su funcionamiento en uno o más edificios. Un operador puede, utilizando una estación de gestión (por ejemplo, Desigo CC) de un sistema de gestión de edificios, emplear las medidas de optimización recomendadas para los edificios respectivos. Ventajosamente, el sistema de gestión de edificios está directamente acoplado a un sistema FDD que proporciona fallas, medidas de optimización o las dos anteriores para los respectivos edificios.

10 En otra realización ventajosa, las medidas de optimización se introducen directamente en los equipos de automatización de edificios (infraestructura, dispositivos) de uno o varios edificios. En principio, las medidas de optimización se pueden emplear en tiempo real en los equipos de automatización de edificios. En este enfoque se pueden emplear medidas de optimización especialmente simples, por ejemplo, el encendido o apagado recomendado de un dispositivo.

15 Otras realizaciones ventajosas de la invención son un medio legible por ordenador no transitorio (por ejemplo, CD, SSD o memoria USB) y un producto de programa de ordenador (por ejemplo, un módulo de programa escrito en un lenguaje de programación adecuado, por ejemplo, C ++, Java) que comprende instrucciones para realizar el método inventivo para proporcionar medidas de optimización o mejoras para uno o más edificios. El medio legible por ordenador y el producto de programa informático tienen segmentos de programa para, cuando se ejecutan en un dispositivo informático, hacer que el dispositivo informático (por ejemplo, estación de trabajo, computadora de escritorio, computadora portátil, tablet) implemente el procedimiento. El producto del programa informático se puede cargar directa o indirectamente en la

20 memoria de un ordenador digital.

Los medios legibles por ordenador pueden ser cualquier medio disponible al que se pueda acceder por un ordenador. A modo de ejemplo, y no limitativo, los medios legibles por ordenador pueden comprender, por ejemplo, los medios de almacenamiento informáticos y los medios de comunicación. Los medios de almacenamiento informáticos comprenden medios volátiles y no volátiles, así como medios extraíbles y no extraíbles implementados en cualquier método o

25 tecnología para el almacenamiento de información, como instrucciones legibles por ordenador, estructuras de datos, módulos de programa u otros datos. Los medios de almacenamiento por ordenador, por ejemplo, comprenden RAM, ROM, EEPROM, memoria flash u otra tecnología de memoria, CDROM, discos versátiles digitales (DVD) u otro almacenamiento en disco óptico, o cualquier otro medio que se pueda usar para almacenar la información deseada y que pueda accederse desde un ordenador.

30 La figura 9 ilustra una interfaz de usuario ejemplar para un generador de scripts ejemplar. La interfaz de usuario ejemplar comprende un nivel de scripts SL y un nivel de regla RL. El nivel de scripts SL proporciona opciones de configuración general que pueden ser utilizadas por todas las reglas y funciones administrativas para gestionar los scripts. En el nivel de regla RL, un usuario configura un conjunto de reglas para el script, incluyendo su expresión de evaluación, selecciona los puntos de datos que se mostrarán junto con el resultado de la falla, y la configuración de tiempo de ejecución

35 relacionada.

La Figura 10 muestra una lista de LRS ejemplar de configuración de tiempo de ejecución para configurar un generador de scripts. La configuración de tiempo de ejecución controla la ejecución de la regla. El usuario final puede establecer los valores (configuración) antes de ejecutar un script.

La figura 11 ilustra una interfaz de usuario UI ejemplar para definir campos calculados en puntos de datos. La interfaz de usuario (UI) del generador de scripts permite definir campos calculados accesibles en puntos de datos y etiquetar y establecer opciones de publicación para la cuadrícula de resultados y/o el árbol de resultados.

5 La Figura 12 ilustra otro ejemplo de interfaz de usuario IU para un generador de scripts ejemplar. La interfaz de usuario IU que se muestra en la figura 12 permite al usuario seleccionar/mapear tipos de puntos de datos y propiedades de nodos/recursos relevantes para el script y definir los puntos de datos virtuales (calculados/derivados) necesarios.

La Figura 13 ilustra una interfaz de usuario UI ejemplar para etiquetar scripts o reglas. El usuario puede seleccionar los tipos de nodo aplicables para la script/regla y definir además filtros basados en las propiedades de nodo/recurso para el script completo o las reglas individuales.

10 La Figura 14 ilustra una interfaz de usuario UI ejemplar para la configuración de la planificación. El usuario puede establecer la planificación para el script o las reglas para ciertos nodos de los recursos respectivos. Sólo las fallas detectadas dentro de esta planificación se reconocerán como fallas y se mostrarán al usuario. El principio de herencia se aplica a las planificaciones, lo que significa que también se toman para los nodos inferiores cuando se definen en un nodo superior en el árbol de recursos.

15 La Figura 15 ilustra una interfaz de usuario UI ejemplar para el diagnóstico de fallas. El sistema para la detección de fallas y/o para proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios puede definir las causas probables y las acciones **sugeridas** (de mitigación) para las respectivas normas. Esto ayuda al usuario a entender qué hacer en caso de que se produzca una determinada falla.

20 La Figura 16 ilustra una interfaz de usuario UI ejemplar para un editor de expresiones para definir reglas o parte de reglas. Un usuario puede definir la expresión de evaluación (lógica) de la regla respaldada por listas de selección para funciones personalizadas disponibles, puntos de datos (virtuales) definidos y propiedades de nodos/recursos desde el nivel de script, así como configuraciones de tiempo de ejecución, campos calculados y parámetros globales. Un concepto simple "función-en-función-en-función ..." ayuda al usuario a unir la lógica.

25 La Figura 17 muestra una interfaz de usuario UI ejemplar para presentar los resultados de la ejecución del script. La interfaz de usuario UI de acuerdo con la figura 17 muestra un panel de control ejemplar para presentar los resultados, por ejemplo, fallas detectadas, defectos y/o medidas de optimización sugeridas. La ventana en el lado izquierdo del panel de control ejemplar muestra una jerarquía de un árbol de resultados. La jerarquía de árbol puede mostrar datos numéricos junto al nombre del nodo de árbol y los usuarios pueden seleccionar qué parámetros de datos les gustaría ver (Ejemplo: Duración o Costo o Número de Fallas o Tendencias). Ventajosamente, el usuario puede seleccionar un campo de datos personalizado para agregar y mostrar en el árbol a la derecha de los nombres de nodo.

30 Procedimiento y sistema para detectar fallas y/o proporcionar medidas de optimización con el fin de mejorar el rendimiento de uno o varios edificios, especialmente equipos de automatización de edificios de uno o varios edificios, en el que se proporciona una base de datos para almacenar un modelo de datos que comprende datos de recursos y datos de rendimiento de uno o varios edificios, en el que el modelo de datos se representa en una o más estructuras de árbol jerárquicas, en las que los nodos de las estructuras de árbol representan datos de recursos y las hojas de las estructuras de árbol representan datos de rendimiento de uno o varios edificios, en el que los datos de rendimiento están representados en cada caso por puntos de datos, en los que los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación. Además, se proporciona una interfaz de usuario (por ejemplo, una interfaz de usuario basada en menús) para configurar las reglas para analizar los datos de recursos y los datos de rendimiento con respecto al rendimiento del edificio de uno o varios edificios, en el que las reglas que se refieren a los

5 mapeos de puntos de datos, parámetros , mapeo de datos de recursos, configuración de tiempo de ejecución o las programaciones se configuran mediante la entrada del usuario utilizando operadores lógicos y funciones definidas. Los módulos de código ejecutables (por ejemplo, archivos dll) son generados en tiempo de ejecución por una unidad de procesamiento en base a las reglas configuradas por el usuario, en donde las reglas son ejecutadas por un sistema de tiempo de ejecución que atraviesa las estructuras de árbol del modelo de datos. Se proporciona una unidad de salida adecuada para emitir fallas detectadas y/o para proporcionar medidas de optimización para uno o más edificios.

10 Si bien se han ilustrado y descrito realizaciones y aplicaciones particulares de la presente invención, se debe entender que la invención no se limita a la construcción y a las composiciones precisas descritas aquí y que pueden ser evidentes diversas modificaciones, cambios y variaciones a partir de las descripciones anteriores sin apartarse del alcance de la invención como se define en las reivindicaciones adjuntas.

Signos de referencia

	S1 - S3	Sistema FDD
	C	Nube
	SB	Generador de scripts SB
15	IU	Unidad de entrada
	UI	Interfaz de usuario
	IPF1 - IPF3:	Campo de entrada
	OU	Unidad de salida
	PU	Unidad de procesamiento
20	CG	Generador de código
	RS	Sistema de tiempo de ejecución
	RE	Motor de reglas
	B	Edificio
	AD, AD1, AD2	Datos de recursos
25	PD, PD1 - PD6	Datos de rendimiento
	ES	Proveedor de energía
	EP	Precio de la energía
	DB	Base de datos
	DMI - DM3	Modelo de datos
30	BMS	Sistema de gestión de edificios
	F	Fallas
	OM	Medidas de optimización
	ID	Datos de mejora
	AHU	Unidad de tratamiento de aire
35	SL	Nivel de scrips
	RL	Nivel de regla
	LRS	Lista de configuración de tiempo de ejecución

REIVINDICACIONES

1. Procedimiento asistido por ordenador para detectar fallas (F), para proporcionar medidas de optimización (OM) o las dos mencionadas anteriormente, con el fin de mejorar el rendimiento de un edificio de uno o varios edificios (B), incluidos los equipos de automatización de edificios dispuestos en relación con uno o más edificios (B), el procedimiento comprende:
 - 5 (S1) proporcionar una base de datos (DB) que almacena un modelo de datos (DM1 – DM3) que incluye datos de recursos (AD, AD1 – AD2) y datos de rendimiento de uno o varios edificios (B), en el que el modelo de datos (DM1 – DM3) está representado en una o más estructuras de árbol jerárquicas, cada una de las cuales tiene una pluralidad de nodos y una pluralidad de hojas,
 - 10 en el que los nodos de las estructuras de árbol representan datos de recursos (AD, AD1 – AD2) y las hojas de las estructuras de árbol representan datos de rendimiento (PD, PD1 – PD6) de uno o varios edificios (B),
 - en el que los datos de rendimiento (PD, PD1 – PD6) están representados en cada caso por puntos de datos, en donde los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación;
 - 15 (S2) proporcionar una interfaz de usuario (UI) para configurar reglas para analizar los datos de recursos (AD, AD1 – AD2) y los datos de rendimiento (PD, PD1 – PD6) en relación con el rendimiento del edificio de uno o varios edificios (B), en el que las reglas se refieren a mapeos de los puntos de datos, parámetros, mapeos de los datos de recursos, configuraciones de tiempo de ejecución, o planificaciones y se configuran por una entrada de usuario utilizando operadores lógicos y funciones definidas;
 - 20 (S3) generar módulos de código ejecutable en tiempo de ejecución basados en las reglas configuradas por el usuario por medio de una unidad de procesamiento (PU), en el que las reglas son ejecutadas por un sistema de tiempo de ejecución (RS) que atraviesa las estructuras de árbol del modelo de datos (DM1 - DM3), en el que el recorrido de las estructuras de árbol del modelo de datos (DM1 - DM3) se realiza de acuerdo con la metainformación de los puntos de datos; y
 - 25 (S4) emitiendo fallas (F) detectadas, medidas de optimización (OM) o las dos mencionadas con respecto a uno o más edificios (B) en un dispositivo de salida (OU).
 2. Procedimiento según la reivindicación 1, en el que la ejecución de la regla se realiza según la configuración del tiempo de ejecución.
 - 30 3. Procedimiento según una de las reivindicaciones anteriores, en el que las medidas de optimización (OM) comprenden períodos de falla, puntos de datos seleccionados, columnas de salida o diagnóstico de fallas.
 4. Procedimiento según una de las reivindicaciones anteriores, en el que las medidas de optimización (OM) se introducen en un sistema de gestión de edificios (BMS) para su funcionamiento en uno o más edificios (B).
 - 35 5. Método según una de las reivindicaciones anteriores, en el que las medidas de optimización (OM) se introducen en los equipos de automatización de edificios de uno o varios edificios (B).
 6. Medio legible por ordenador que contiene un conjunto de instrucciones que hacen que un ordenador realice un procedimiento según una de las reivindicaciones 1 a 5.
 - 40

7. Producto de programa informático que se puede cargar en la memoria de un ordenador digital que comprende secciones de código de software para realizar un procedimiento según una de las reivindicaciones 1 a 5 cuando dicho producto se ejecuta en un ordenador.
- 5 8. Sistema (S1 - S3) para detectar fallas (F), proporcionar medidas de optimización (OM), o las dos mencionadas anteriormente, con el fin de mejorar el rendimiento de un edificio de uno o varios edificios (B), incluidos los equipos de automatización de edificios dispuestos en relación con uno o más edificios (B), el sistema (S1 - S3) comprende:
- 10 una base de datos (DB) que almacena un modelo de datos (DM1 – DM3) que incluye datos de recursos (AD, AD1 – AD2) y datos de rendimiento (PD, PD1 - PD6) de uno o varios edificios (B), en el que el modelo de datos (DM1 – DM3) está representado en una o más estructuras de árbol jerárquicas, cada una de las cuales tiene una pluralidad de nodos y una pluralidad de hojas,
- 15 en el que los nodos de las estructuras de árbol representan datos de recursos (AD, AD1 – AD2) y las hojas de las estructuras de árbol representan datos de rendimiento (PD, PD1 – PD6) de uno o varios edificios (B),
- en el que los datos de rendimiento (PD, PD1 – PD6) están representados en cada caso por puntos de datos, en donde los puntos de datos comprenden series temporales de valores de campo medidos o derivados y metainformación;
- 20 una interfaz de usuario (UI) para configurar reglas para analizar los datos de recursos (AD, AD1 – AD2) y los datos de rendimiento (PD, PD1 – PD6) en relación con el rendimiento del edificio de uno o varios edificios (B),
- en el que las reglas se refieren a mapeos de los puntos de datos, parámetros, mapeos de los datos de recursos, configuraciones de tiempo de ejecución, o planificaciones y se configuran por una entrada del usuario utilizando operadores lógicos y funciones definidas;
- 25 una unidad de procesamiento (PU) para generar módulos de código ejecutable (dll) en tiempo de ejecución basados en las reglas configuradas por el usuario, en el que las reglas son ejecutadas por un sistema de tiempo de ejecución (RS) que atraviesa las estructuras de árbol del modelo de datos (DM1 - DM3), en el que el recorrido de las estructuras de árbol del modelo de datos (DM1 - DM3) se realiza de acuerdo con la metainformación de los puntos de datos; y
- 30 una unidad de salida (OU) para emitir fallas detectadas (F) y/o para proporcionar medidas de optimización (OM) para uno o más edificios (B).
9. El sistema (S1 - S3) según la reivindicación 8, en el que la unidad de salida (OU) es una pantalla de un monitor de ordenador, una pantalla de un dispositivo portátil o la pantalla de una estación de monitorización de un sistema de automatización de edificios (BMS).
- 35 10. Sistema (S1 - S3) según una de las reivindicaciones 8 a 9, que comprende además una interfaz con un sistema de automatización de edificios (BMS) para su funcionamiento en uno o más edificios (B).
11. Sistema (S1 - S3) según una de las reivindicaciones 8 a 10, que comprende además una interfaz en los equipos de automatización de edificios de uno o varios edificios (B)

FIG 1

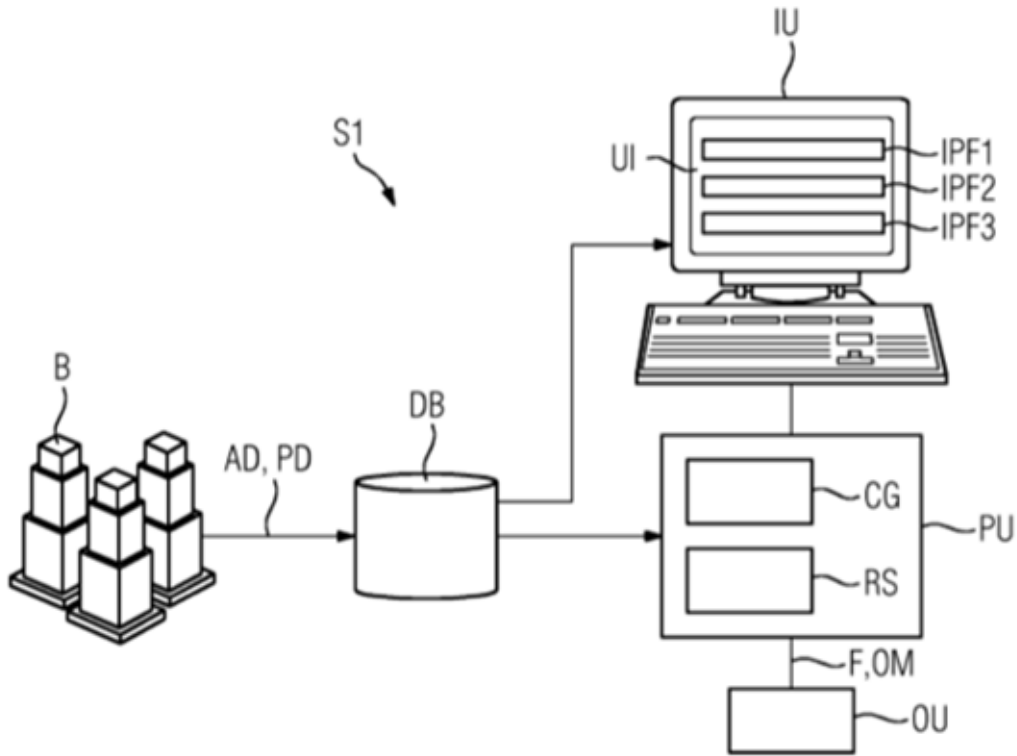


FIG 2

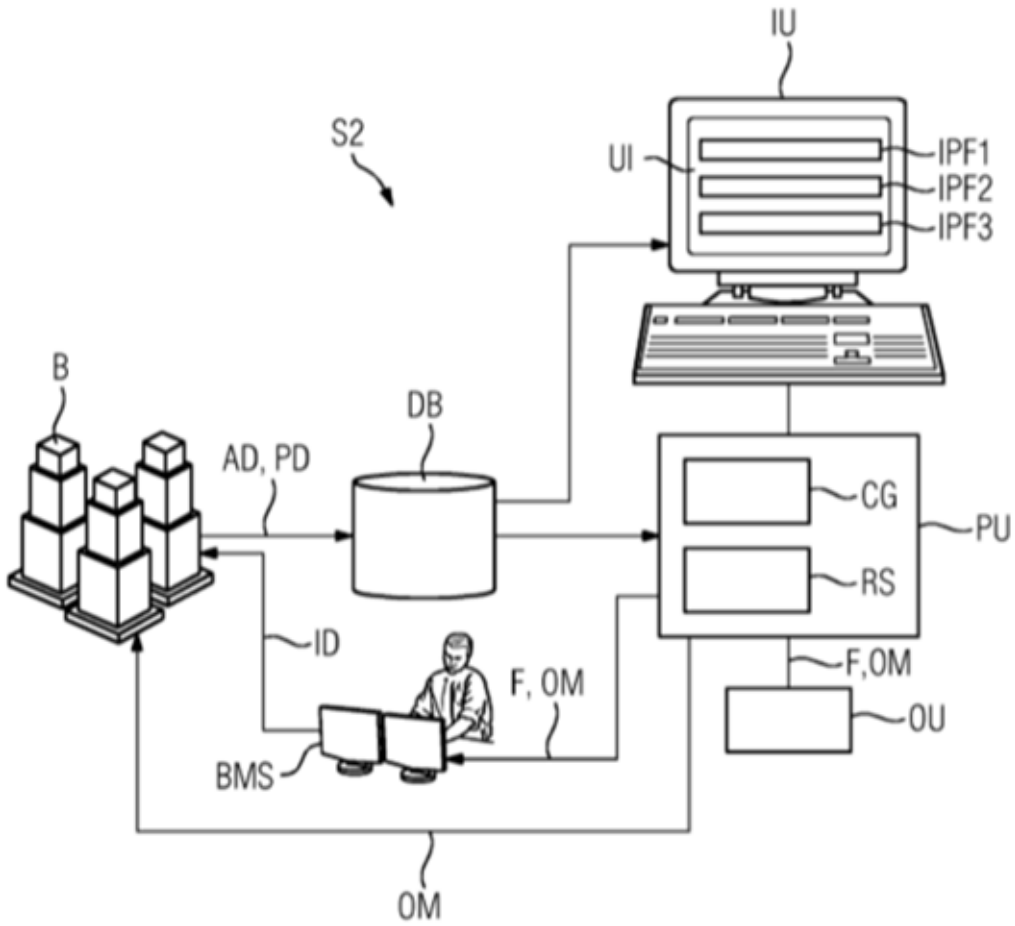


FIG 3

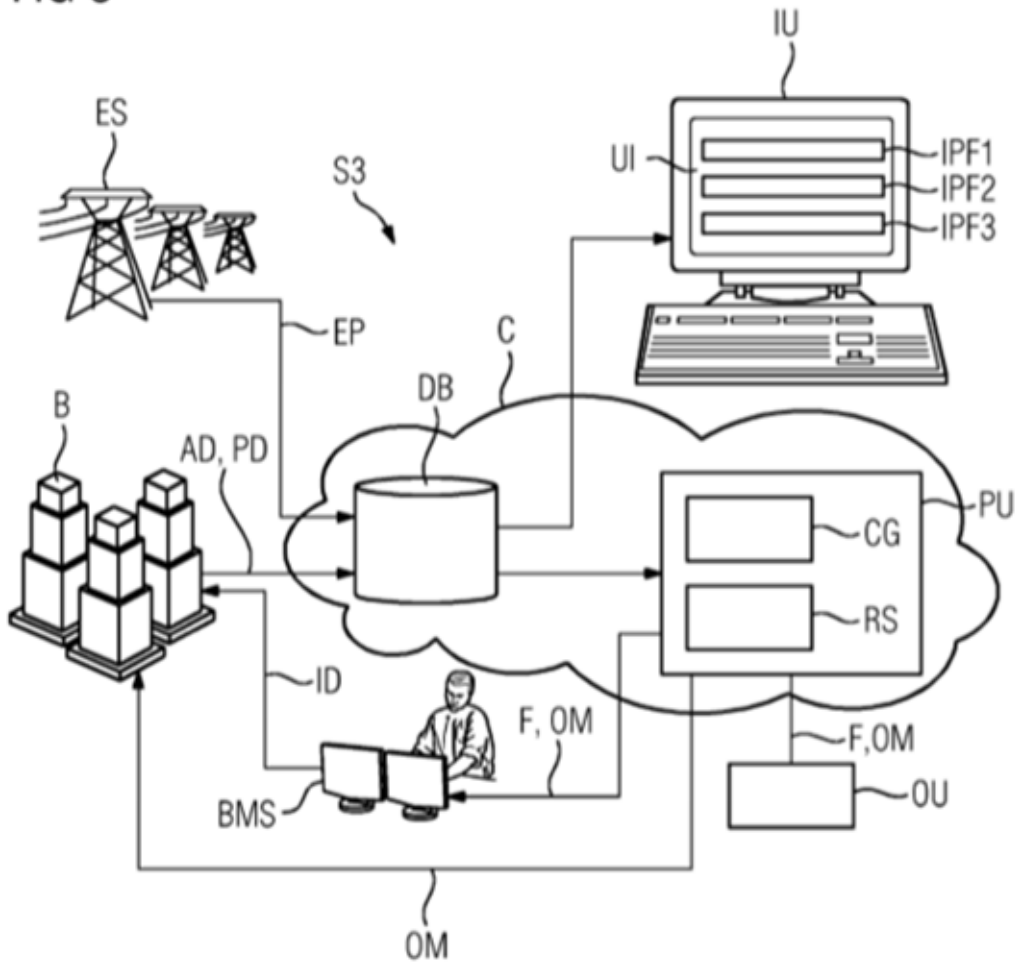


FIG 4

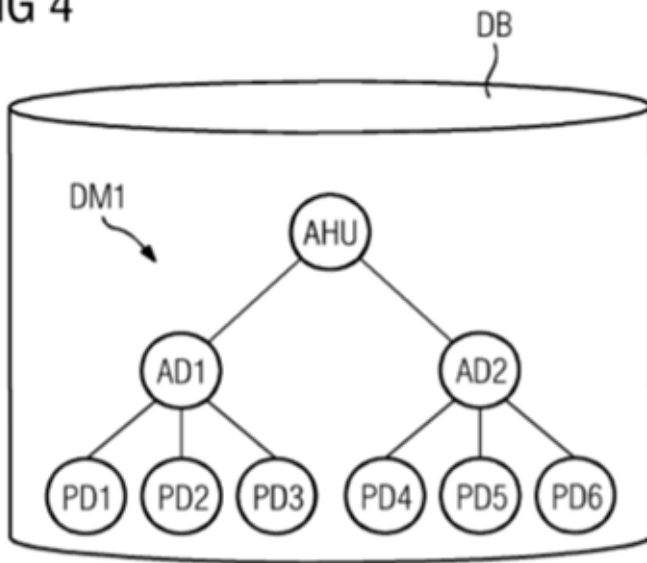


FIG 5

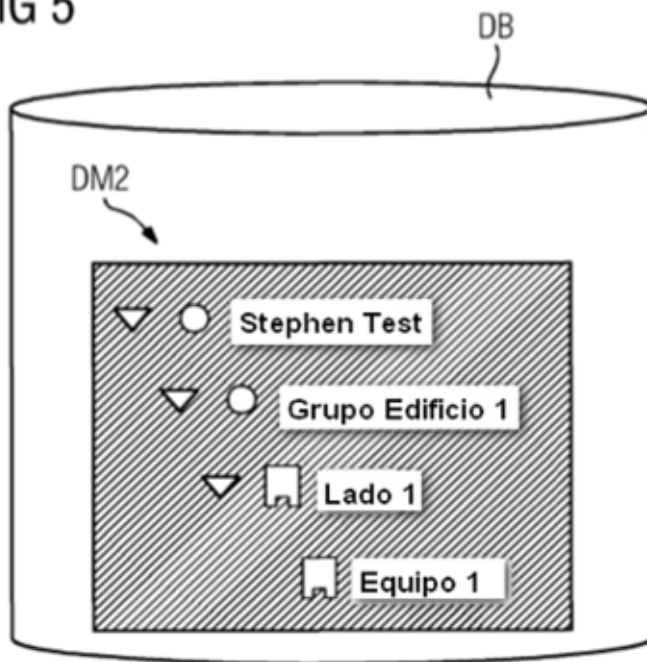


FIG 6

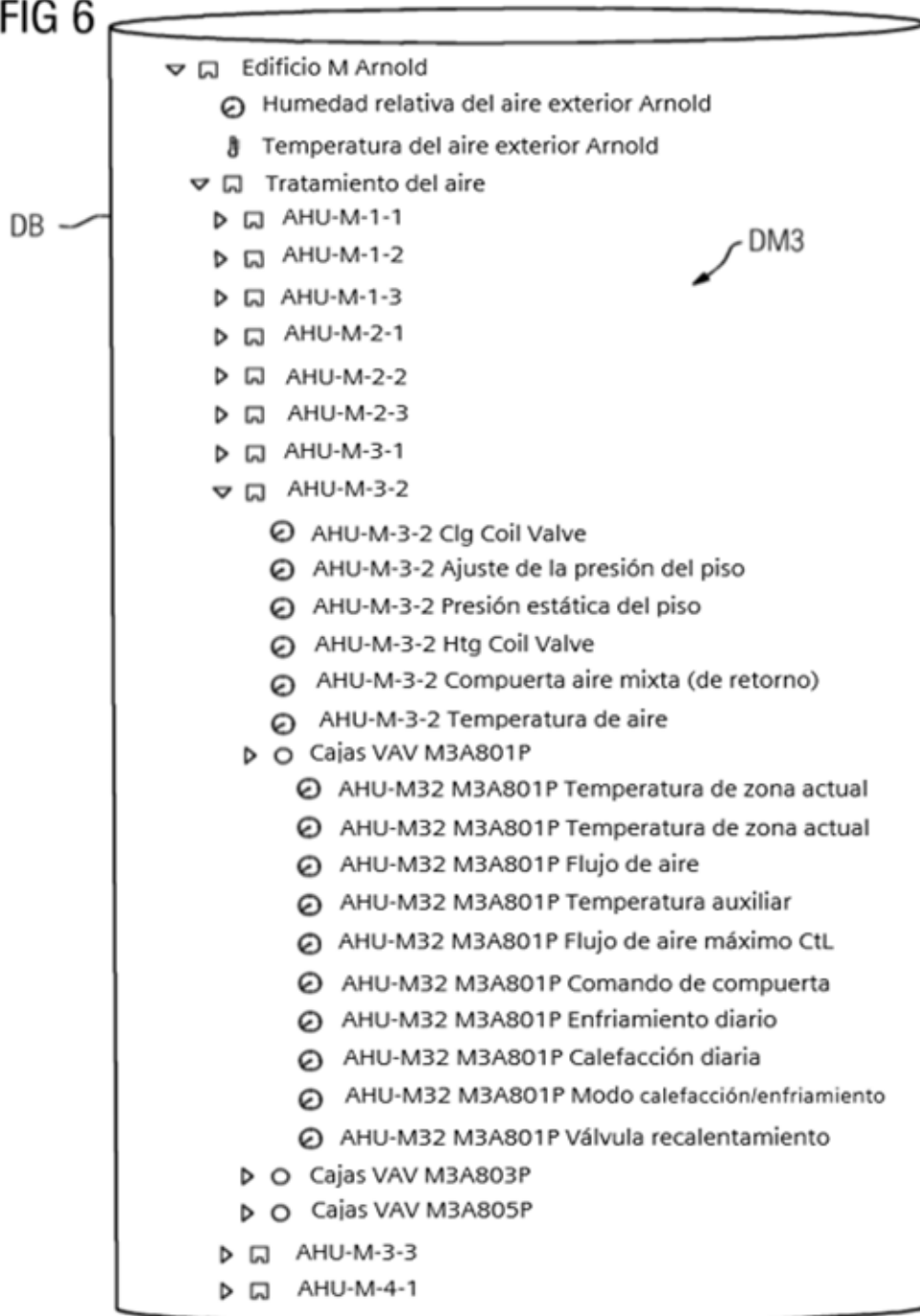


FIG 7

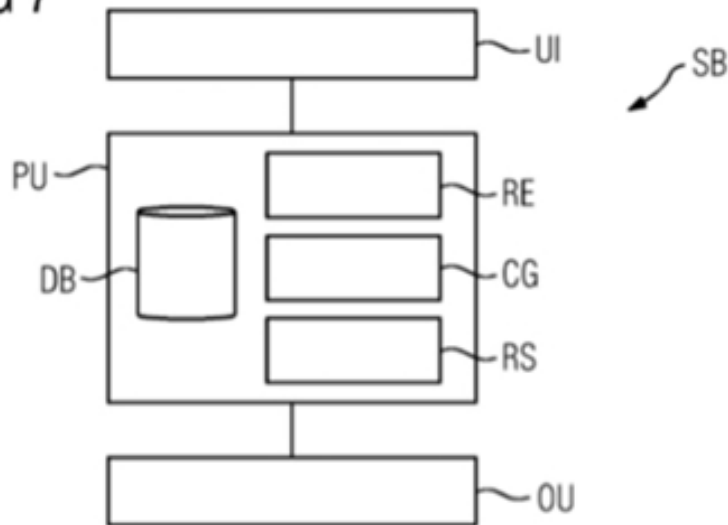


FIG 8

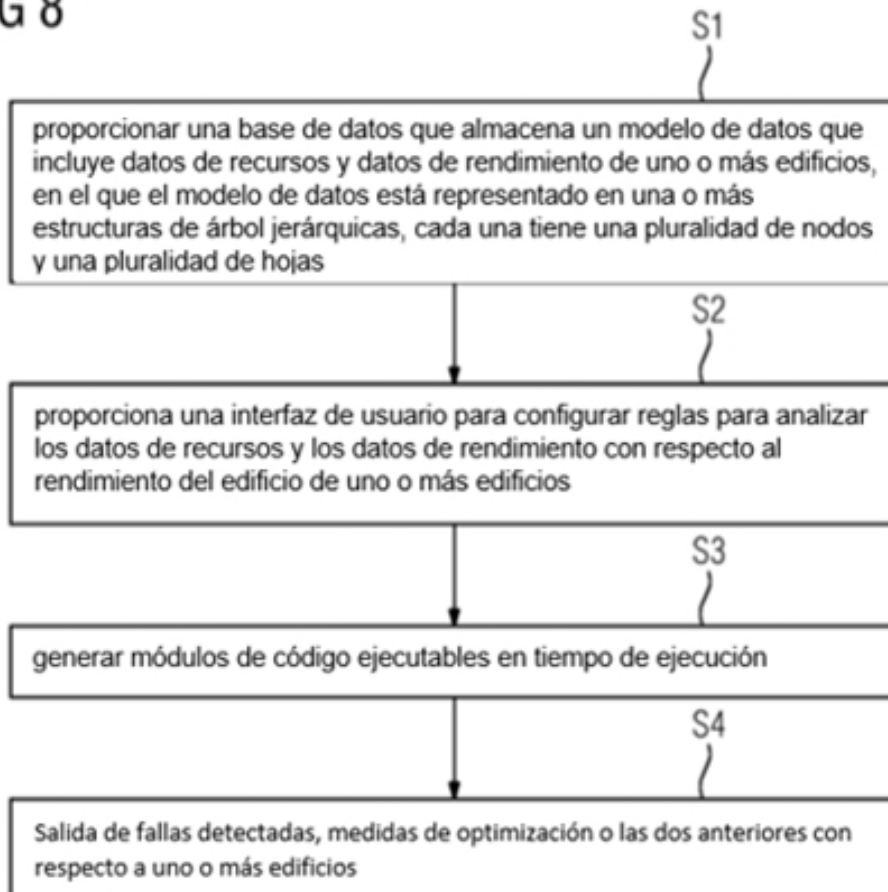
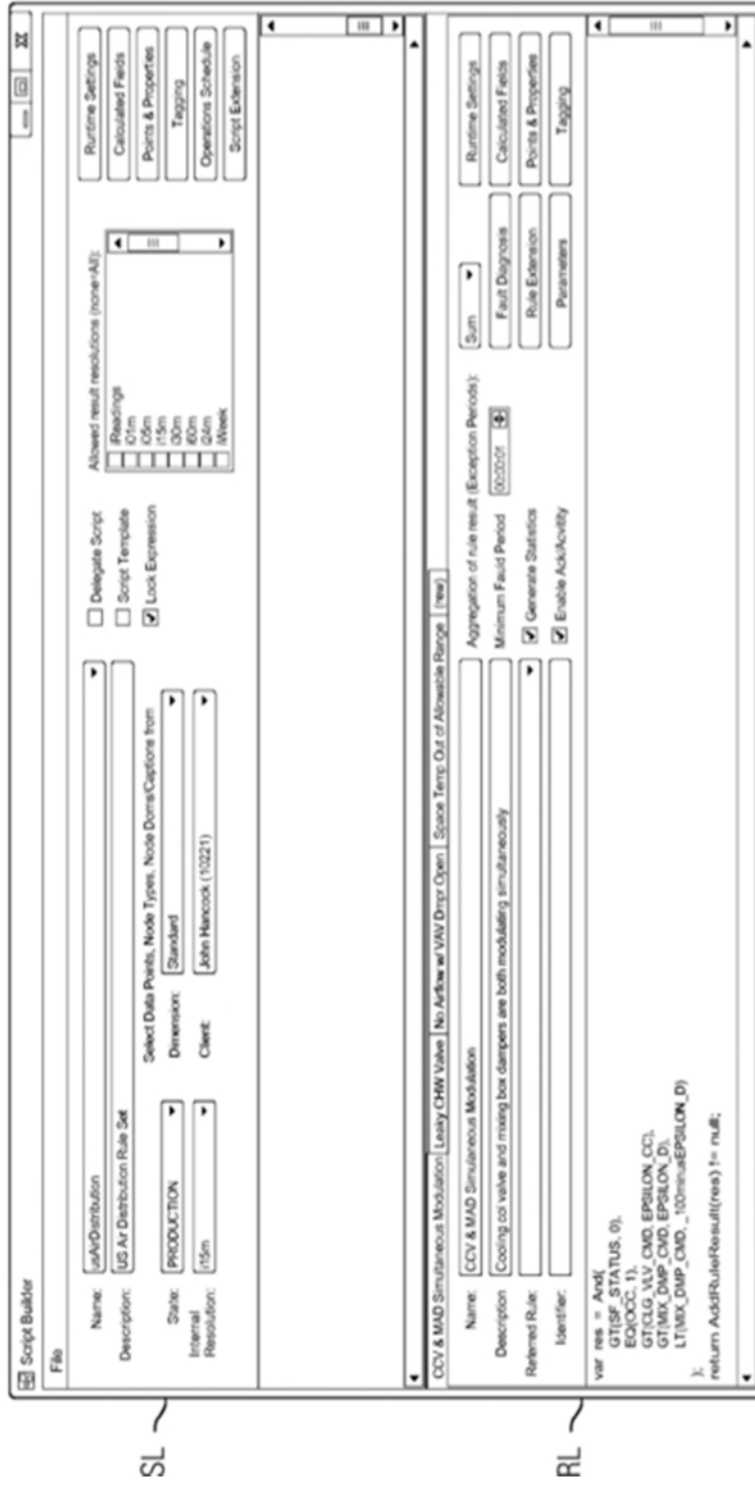


FIG 9



SL

RL

FIG 10

LRS

Runtime Settings

Define the Runtime Settings - these are displayed at runtime in the Runtime Settings Dialog

Abbreviation	Description	Type	Value	Unit
temperatureE...	Temperature Error Tolerance	DOUBLE	0	Delta °F
temperatureR...	Temperature Rise across Return Fan	DOUBLE	0	Delta °F
thresholdHiV	Threshold error heating coil valve	DOUBLE	0	
thresholdCCV	Threshold error cooling coil valve	DOUBLE	0	
electricEnergy...	Electric Energy Rate	DOUBLE	0	\$
utilityEnergyR...	Utility Energy Rate	DOUBLE	0	\$
thresholdMin...	Threshold error minimum outdoor air fraction	DOUBLE	0	
thresholdErro...	Threshold error ratio of flows	DOUBLE	0	
thresholdNor...	Threshold error normalized damper control signal	DOUBLE	0	
maxModeCh...	Maximum number of mode changes per hour	INT	0	
temperatureR...	Temperature Rise across Supply Fan	DOUBLE	0	Delta °F
lowerEconLo...	Lower Econ Lockout	DOUBLE	0	°F
upperEconLo...	Upper Econ Lockout	DOUBLE	0	°F

OK Cancel

FIG 11

UI

Calculated Fields for Script

Define the Calculated Fields - these are accessible from Data Points and Tagging

Calc.Fields | Node Specific Settings

Name	Type	Expression	Aggregation	Publish	..in Tree	..in Stat	Sequence	Display Unit	Formatting
▶ Node Type	STRING	return locl= null && lcl...	None	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2		
Exceptions	DOUBLE	return errorPeriods.C...	UseRule...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	8		
Duration	DOUBLE	return AnalyticsServe...	UseRule...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	11		timespan
Minimum	DOUBLE	return errorPeriods.M...	Min	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	9		timespan
Maximum	DOUBLE	return errorPeriods.M...	Max	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	10		timespan
TARGET	STRING	return locl= null? lcc...	None	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0		
RULE	STRING	return rule.TK_Name;	None	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1		
Priority	STRING	return rulePriority;	None	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0		
kpiCategory	STRING	return ruleKpCategory;	None	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0		
.				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0		

OK

FIG 12

UI

Points & Properties

Points | Node Properties | Virtual Points

Define the Valuetype Data Points of the Script. Hint: Double-Click the "Value Type" or "Name" cell to open the Da

Value Type	Name	Flow Natures	Unit	Resolution	Inherit
Terminal Unit Airflow Percent [100000	BOX_AIRFLOW	32767		i15m	<input type="checkbox"/>
Terminal Unit Damper Command [...	BOX_DMP_CMD	32767		i15m	<input type="checkbox"/>
Cooling Coil Valve Command [100...	CLG_VLV_CMD	32767	%	i15m	<input type="checkbox"/>
Mixed Air Temperature [100000003]	MAT	32767	°F	i15m	<input type="checkbox"/>
Mixing Box Dampers Command [1...	MIX_DMP_CMD	32767	%	i15m	<input type="checkbox"/>
Occupied Mode [100000002]	OCC	32767		i15m	<input type="checkbox"/>
Supply Air Temperature [100000520]	SAT	16383	°F	i15m	<input type="checkbox"/>
Supply Fan Status/Proof [10000001...	SF_STATUS	32767		i15m	<input checked="" type="checkbox"/>
Terminal Unit Occupancy Status [1...	SPACE_OCC	32767		i15m	<input type="checkbox"/>
Terminal Unit Occ Cooling Setpoint...	SPACE_OCC_CLG_T...	32767	°F	i15m	<input type="checkbox"/>
Terminal Unit Occ Heating Setpoint...	SPACE_OCC_HTG_T...	32767	°F	i15m	<input type="checkbox"/>
Terminal Unit Room Temperature [...	SPACE_TEMP	32767	°F	i15m	<input type="checkbox"/>
Terminal Unit Room Temperature ...	SPACE_TEMP_SP	32767	°F	i15m	<input type="checkbox"/>

OK Cancel

FIG 13

UI

Tagging of Script

Node Types Filter on Numeric Properties Filter on String Properties

Select the applicable Node Types - no selection means all nodes

- Account [160]
- Administrator [31]
- Air Handling Unit [155]
- airDistributionSystem [154]
- Customer [26]
- cvBox [157]
- Element [33]
- Object [23]
- Pool [4]
- Site [5]
- Sub Element [101]
- Sub Element 2 [134]
- Sub Element 3 [135]
- System Manager [111]
- VAV Box [156]
- Vendor [159]

OK Cancel

FIG 14

UI

Operations Schedule

Schedule inherited to lower Nodes (0 is top Node). No Operations: Leave out the day, Full operations: 00:00:00 - 00:00:00

Workdays Holydays

Day	NodeId	Time from	Time to
Monday	▼ 282458	06:00:00	18:00:00
Tuesday	▼ 282458	08:00:00	20:00:00
	▼		

FIG 15

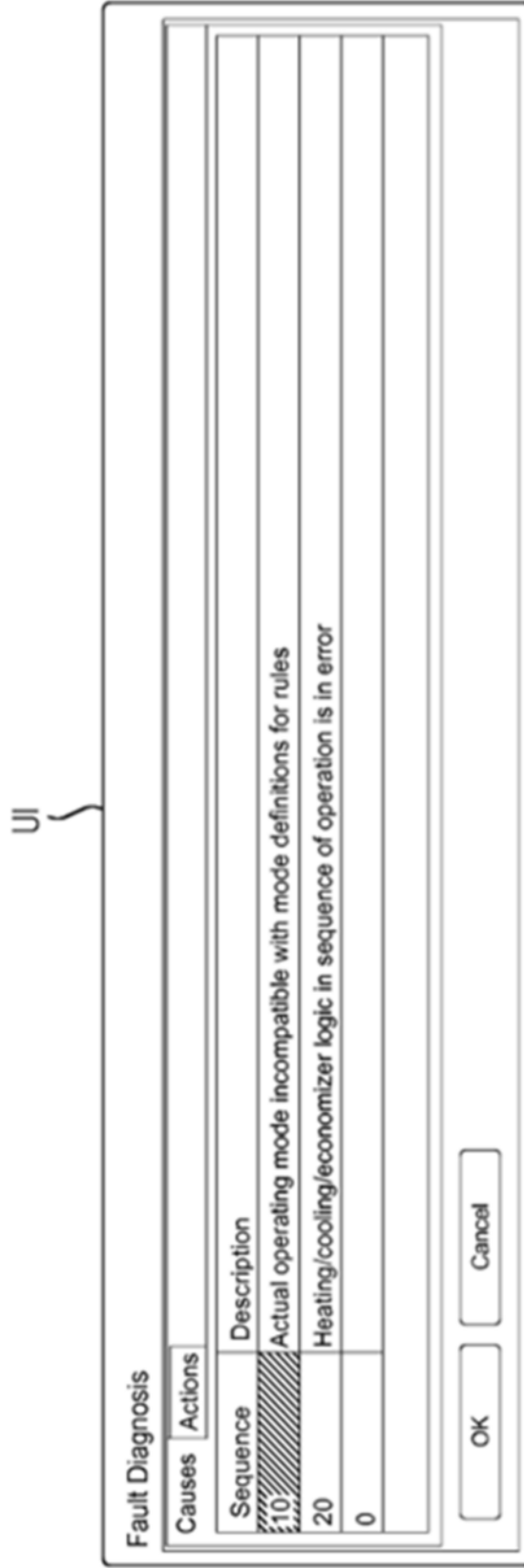


FIG 16

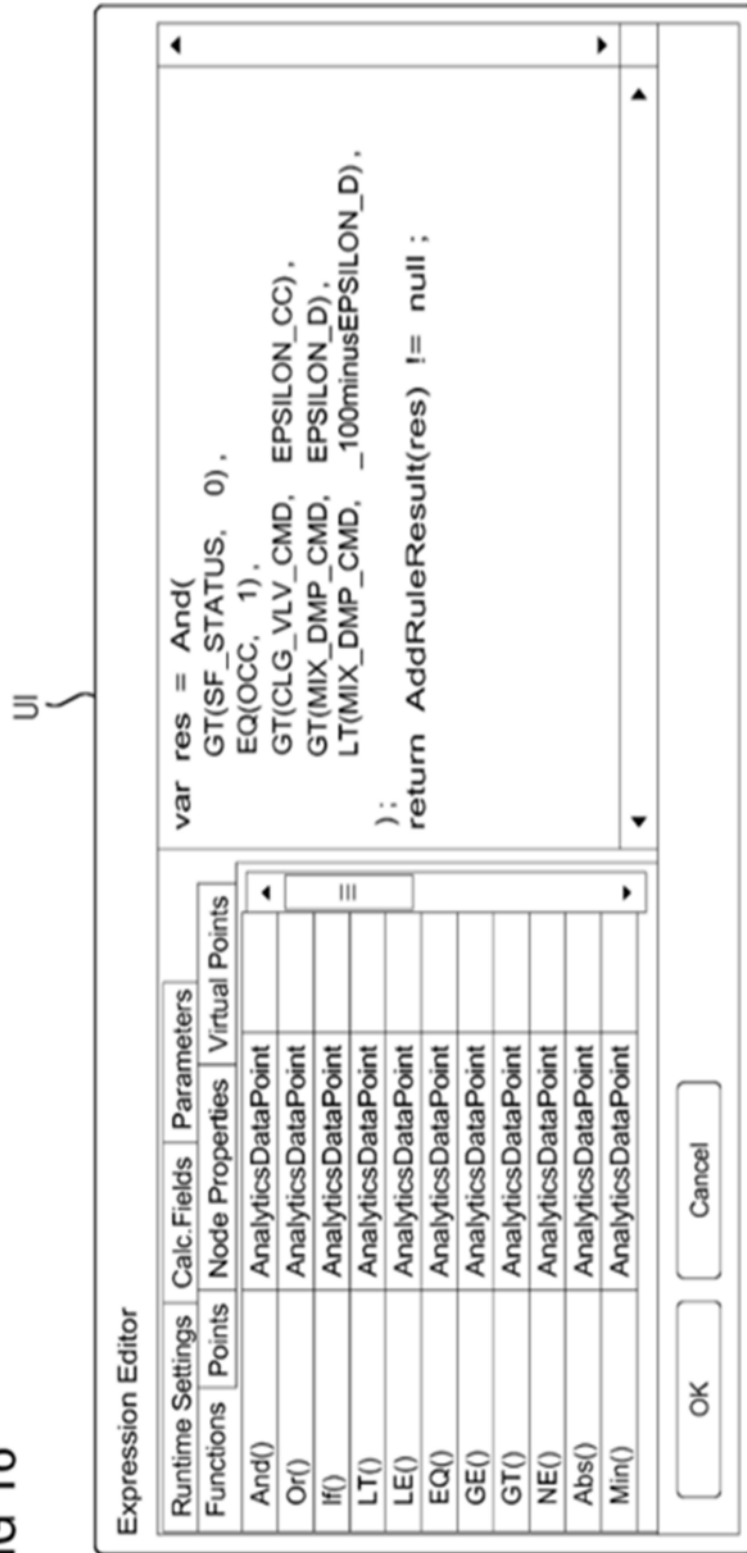


FIG 17

UI

