

Desarrollo de aplicación para la gestión del conocimiento de modelos matemáticos

Viridiana Rodríguez Cardiel
Centro de Investigación en Matemáticas (CIMAT) Unidad
Zacatecas
viridiana.rodriguez@cimat.mx

Edrisi Muñoz Mata
Centro de Investigación en Matemáticas (CIMAT) Unidad
Zacatecas
emunoz@cimat.mx

Resumen: En la actualidad diferentes áreas científicas hacen uso de modelos matemáticos, como una herramienta básica para diferentes funcionalidades. Generalmente, los modelos matemáticos son muy complejos debido a la naturaleza de los elementos representados. Distintos software, aplicaciones y herramientas, permiten representar y procesar resultados de estos modelos complejos, sin embargo estas herramientas no son diseñadas para entender y crear conocimiento de los modelos. El siguiente trabajo presenta el desarrollo de una aplicación de gestión de conocimiento matemático mediante la utilización de tecnologías de gestión de conocimiento, cuyo objetivo administra y entiende modelos matemáticos. Asimismo se dedica una parte esencial en el proceso y resultados de la herramienta utilizada en el algoritmo propuesto para la minimización del makespan de batch plants multipropósito en el área de procesos del área de ingeniería en su representación informal. Esta representación informal comprende dos códigos distintos: i) la expresión en

lenguaje matemático (por símbolos) y; ii) la expresión en lenguaje natural (oración entendible al usuario).

Palabras clave: Ingeniería de software, ontologías y modelos matemáticos.

Development of a knowledge management tool of mathematical model

Abstract: Nowadays most of the scientific areas use mathematical models as a basic tool to support different functionalities. In general, most of the mathematical models involves too much complexity, due to the nature of mathematical language and the elements that they intent to represent. A wide range of different software, applications and tools can handle those models, however they have not been designed to understand and create knowledge from those models. This work presents the development of an application based on knowledge management technologies, which aim to manage and understand mathematical models. Thus, this paper shows the process and result of the developed tool used in an algorithm for the minimization of the makespan of multipurpose batch plants in the process system engineering area. Finally, two main features for informal representation (for human understanding) are presented: i) each mathematical expression presented in a symbolic manner and; ii) each mathematical expression expressed as natural language sentences.

Keywords: Software engineering, ontology, mathematical models.

1. Introducción

En la actualidad, los modelos matemáticos son utilizados ampliamente para describir fenómenos físicos, químicos, biológicos, económicos o de cualquier otro tipo. Estas tareas contemplan desde usos simples tales como las

operaciones básicas para contabilizar objetos, hasta aplicaciones más complejas, como el uso de sistemas de ecuaciones que describen procesos, sistemas, etc. El objetivo principal de las aplicaciones matemáticas complejas es realizar una mejor toma de decisión sobre un problema encontrado en una realidad. Sin embargo, el desarrollo de estos sistemas de ecuaciones es sumamente complejo cuando la realidad modelada requiere un alto nivel de detalle. Además, el modelado matemático es realizado por expertos con un alto conocimiento matemático, por lo que aunque se tenga la nomenclatura, la complejidad está sujeta a la formación matemática del experto añadiendo más complejidad para su comprensión.

Por otra parte, la adopción de tecnologías semánticas es cada vez más usada por diferentes áreas científicas con el objetivo de capturar y gestionar conocimiento. En específico, el área de tecnologías de la información el desarrollo de sistemas basados en conocimiento para la representación de modelos matemáticos ha recibido una atención creciente en los últimos 20 años. Uno de los ejemplos más representativos para representación de modelos matemáticos para ingeniería es el desarrollado por Gruber & Olsen (1994) en su trabajo llamado “EngMath ontology”. Recientemente, Muñoz et al. (2013) presentaron un modelo llamado representación matemática ontológica (OMR, por sus siglas en inglés Ontological Mathematical Representation), el cual presenta como principal característica la relación entre los elementos matemáticos de una ecuación, con su representación real proveniente de modelos semánticos de otros dominios.

El presente trabajo tiene como objetivo desarrollar un sistema informático que explota el conocimiento capturado en el modelo de conocimiento OMR. Es importante destacar que mediante el uso del modelo OMR se puede obtener el conocimiento de diferentes funciones y ecuaciones (contenido matemático) resultando en un modelo potencialmente explotable para generar y compartir conocimiento sobre un sistema modelado matemáticamente. Tomando como base el modelo OMR, se identifican dos características que el sistema

contempla. La primera consiste en la representación informal (entendible para el humano) basado en la gestión del conocimiento matemático y la más significativa la representación formal (entendible para la máquina). Como parte de la primer característica, en este trabajo se presenta el proceso de validación o conocida dentro del ciclo tradicional del desarrollo del software “pruebas”. Este proceso de pruebas esta enfocado en el algoritmo propuesto por Maravelias & Grossmann (2003) para la minimización de makespan de plantas de producción por lotes multipropósito usando la formulación state-task network (STN).

A continuación se detalla cada una de estas dos características. La representación informal tiene como objetivo formar sentencias entendibles basadas en dos aspectos principales. El primer aspecto se enfoca en generar la sentencia por medio del lenguaje matemático (legible por medio de símbolos en forma de ecuación), y el segundo aspecto se enfoca en mostrar la sentencia como una oración en lenguaje natural. El resultado obtenido será la representación de la información suficiente y necesaria para que el usuario entienda la expresión matemática, además de entender el sistema que matemáticamente se modelo. La representación formal tiene como objetivo explotar el modelo OMR dentro del sistema informático y modelarlo a un dominio/sistemas reales. Con la finalidad de manipular conocimiento del sistema modelado.

Este trabajo se organiza de la siguiente manera: la sección 2 presenta las tecnologías, metodologías y modelos para el desarrollo de la herramienta, sección 3 describe cada una de las fases del ciclo de desarrollo, sección 4 expone el modelo matemático a representar y resultados y finalmente en la sección 5 muestra conclusiones y trabajo futuro.

2. Metodología

El sistema informático tiene como lenguaje base de programación Java (Nirosh, 2013) el cual está diseñado hacia la concurrencia basada y orientada a objetos. Para la segunda versión sobre web, se hace uso de JavaScript (Mcnavage, 2010), sin embargo se esta diseñando para utilizar el framework spring (Risberg et al.), cuyo patrón de diseño es modelo-vista-controlador (MVC, por sus siglas en ingles model-view -controller), para la manipulación/gestión de información requerida en el sistema informático para cumplir con su objetivo. El sistema se está desarrollando en base a metodología ágil adaptándose a prácticas de Scrum (Keith, 2010).

Asimismo, se hace uso del software Mathematica (Wolfram Research, 2014), el cual es un sistema de álgebra computacional, que hace uso de comandos y expresiones matemáticas para generar código en lenguaje MathML (Mathematical Markup Language) (Ausbrooks et al., 2010). MathML es una aplicación XML (Extensible Markup Language) (The World Wide Web Consortium, 2014) para describir la notación matemática. MathML tiene dos principales códigos: presentation markup y content markup. Por un lado, presentation markup define la estructura sintáctica a un nivel alto de la expresión matemática de la misma forma que se especifica el título, secciones y párrafos de un documento textual. Por otro lado, content markup se enfoca en el significado matemático, la lógica de estructura y etiquetas de la expresión matemática. La característica que explota el sistema informático está basado en el código content markup, pues su uso permite una mejor interpretación de las expresiones matemáticas.

Por otra parte, el sistema potencialmente permitirá la instanciación en el modelo del conocimiento OMR con la finalidad de definir: i) el comportamiento matemático, ii) la realidad a la que está asociada el modelo, iii) el tipo de operaciones matemáticas, y finalmente iv) los símbolos que representan los elementos de la ecuación (término u operador). Estas propiedades son

identificados en OMR por la clase “Término” (Term) (Muñoz, Capón-García, Laínez-Aguirre, Espuña, & Puigjaner, 2014) y están asociadas a cada uno de los elementos que forman la expresión matemática (ver Figura 1).

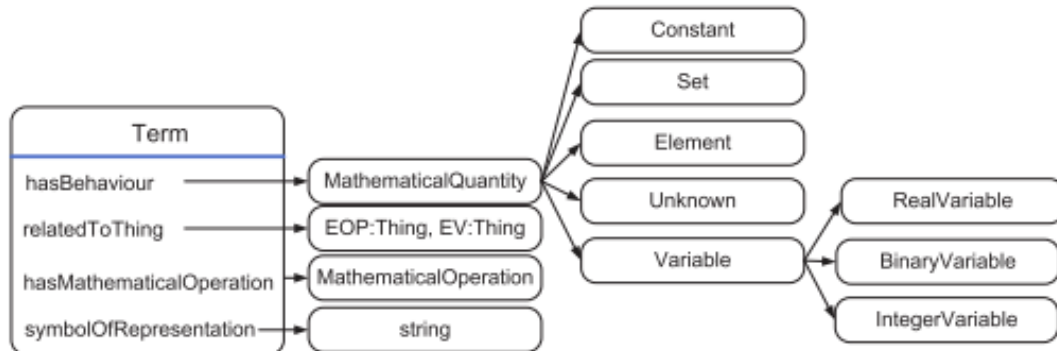


Figura 1. Principales relaciones entre la clase “Término” (Term) y el dominio matemático.

3. Desarrollo

El desarrollo de la aplicación, parte del resultado de la explotación del código content markup, donde el sistema obtiene una oración entendible para el usuario, permitiendo distinguir entre operaciones matemáticas, elementos de la expresión y la interacción entre los mismos. Dando seguimiento a la gestión de la información dentro sistema (base de datos por modelo, con respecto a descripciones, especificaciones de cada uno de los términos) y la instanciación del modelo OMR para generar la representación formal. Actualmente se esta trabajando en la fase de gestión.

El desarrollo de software sigue un modelado incremental para las etapas de requerimientos, diseño, codificación y pruebas que se han definido como sprint backlog dentro del método ágil. A continuación se detallan cada una de las etapas para el desarrollo del sistema informático. Tal detalle se especifica hasta la representación informal.

Definición de requerimientos

En esta etapa, se especifica cada uno de los requisitos que cumple cada una de las etapas del desarrollo. A continuación, se enlistan los diferentes requisitos planteados para el desarrollo de este sistema.

- **Lectura de archivo de texto basado en “content markup”:** mediante el uso del software Mathematica se genera los archivos de texto utilizados para exportar su contenido al sistema en lenguaje MathML.
- **Explotación detallada de expresión matemática:** identificación de cada uno de los elementos que conforman la expresión matemática, identificando operadores y términos. Cada operador puede representar operaciones lógicas o algorítmicas. Asimismo cada término representa un elemento real del sistema real descrito.
- **Definición simbólica:** el usuario define o carga la nomenclatura de cada uno de los términos previamente identificados.
- **Presentación informal:** presentar la expresión matemática en una oración de acuerdo a la nomenclatura dada por el usuario. Se define las clases de Java Apply y Operator, siguiendo la misma lógica de estructuración en content markup. Por cada apply existe un operador que corresponde uno a uno y dentro de un apply pueden existir sub applies, es decir una clase apply puede tener como atributo una variable de tipo apply. Siguiendo esta lógica, es como se desarrollaron los métodos necesarios para representar la expresión en una oración (ver Figura 5). Adicionalmente para la parte de operadores se cuenta con un archivo de texto donde se describe a cada uno de los diferentes operadores que pueden intervenir en un sistema de ecuaciones, en base a las etiquetas dadas por MathML.
- **Diseño**

Se hace uso de diagramas de flujo y de clase.

Pruebas

Para esta etapa se tomo el modelo/algoritmo mencionado en la introducción y el detallado en el caso de estudio, el cual esta formado por 24 ecuaciones. La información requerida fue tomada de la base de datos con el siguiente contenido: interpretación de cada ecuación/función, así como la nomenclatura/significado de cada uno de los términos/símbolos, tipo de variable, referencia/enlace/aparición de cada término en n ecuaciones, descripción del modelo y por cada grupo de ecuaciones, tipo de decisión y descripción de la decisión. En el siguiente esquema se describe los pasos realizados hasta generar la expresión lo mas acertada a la interpretación dado por el experto.

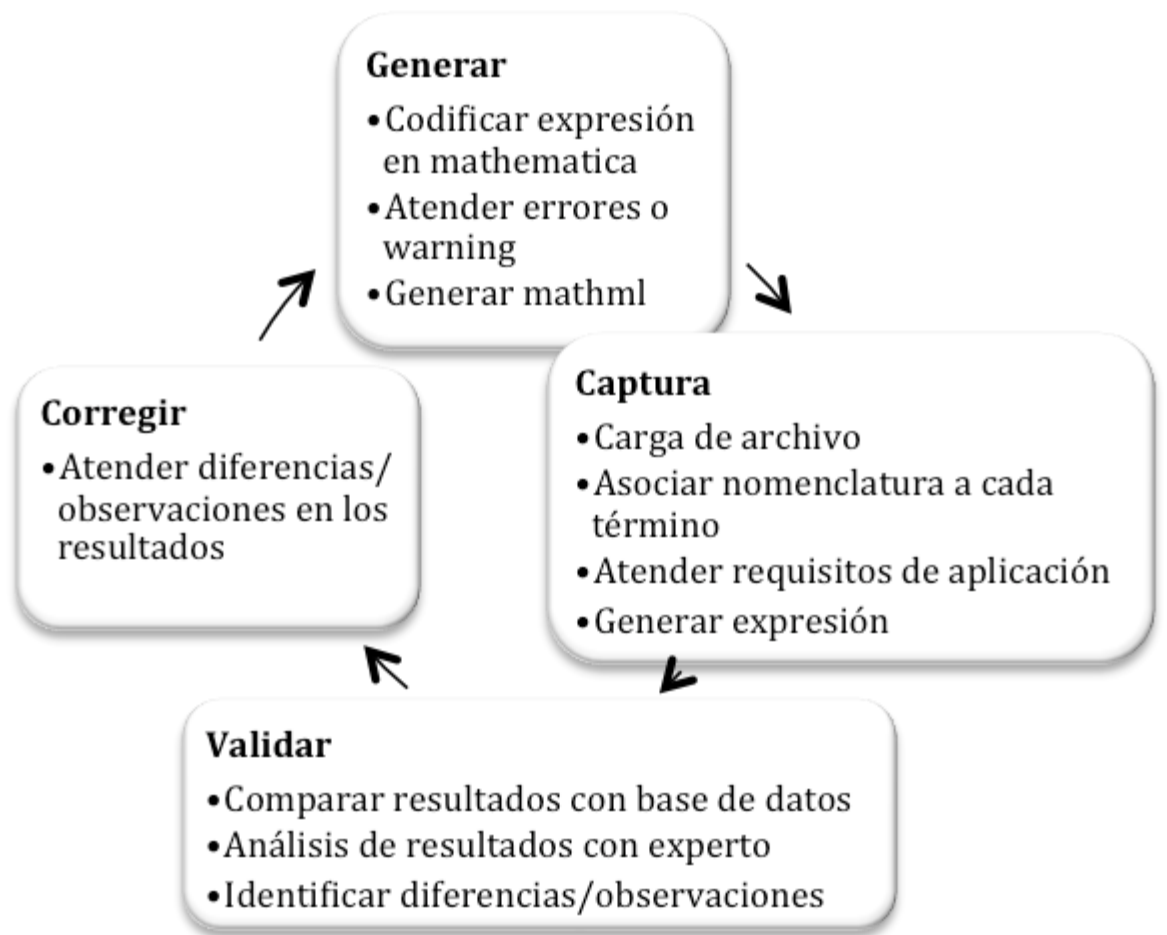


Figura 2. Ciclo de prueba utilizado para validar motor de representación informal al modelar algoritmo.

4. Caso de estudio

Como motor fundamental del desarrollo del sistema/aplicación para el modelado matemático, se ha considerado hacer énfasis en la parte de pruebas para soportar un modelo complejo de una realidad. El modelo a probar se basa en el procesamiento de las tareas que se asumen de forma constante y el horizonte temporal fijo se divide en intervalos de tiempo de duración conocida igual al máximo común divisor de los tiempos de procesamiento de todas las tareas. La asunción de los tiempos de procesamiento constantes no siempre es realista, mientras que la longitud de los intervalos puede ser tan pequeño que sea conduce a un número prohibitivo de intervalos de representación del modelo resultante insoluble, o bien requiere aproximaciones que pueden comprometer la viabilidad y la forma más óptima de la solución. El algoritmo propuesto se basa en la formulación STN discreta de Shah et al. (1993), Donde el binario W_{ijt} es igual a 1 si la tarea $i \in I$ comienza en la unidad $j \in J$ en el momento $t \in T$, B_{ijt} es el tamaño de lote de la tarea i que comienza en la unidad j en el tiempo t , S_{st} es el inventario nivel de $s \in S$ estatales en el tiempo t , R_{st} / D_{ist} son las compras / ventas de estado s en el tiempo t y U_{st} es el nivel de consumo de utilidad $u \in U$ en el tiempo t .

En la siguiente figura se presenta el sistema de ecuaciones para el modelo.

$$\begin{aligned}
& \forall_{\{j,t\}} \sum_{i \in I(j)} \sum_{t'=t}^{t-p_i+1} w_{i,j,t'} \leq 1 \\
& \forall_{\{j,t\}} Vmin_j W_{i,j,t} \leq B_{i,j,t} \leq Vmax_j W_{i,j,t} \\
& \forall_{\{s,t\}} S_{s,t} = S_{s,t-1} + \sum_{j \in (i|j) \cap SO_s} \rho O_{i,s} B_{i,j,t-p_i} - \sum_{j \in (i|j) \cap SO_s} \rho I_{i,s} B_{i,j,t} + R_{s,t} - D_{s,t} \\
& \forall_{\{s,t\}} 0 \leq S_{s,t} \leq C_s \\
& \forall_{\{s,t\}} R_{s,t} \leq r_{s,t} \\
& \forall_{\{s,t\}} D_{s,t} \geq d_{s,t} \\
& \forall_{\{u,t\}} U_{u,t} = \sum_j \sum_{i \in I(j)} \sum_{\theta=1}^{p_i-1} (\alpha_{u,j} W_{i,j(t-\theta)} + \beta_{u,j} B_{i,j(t-\theta)}) \\
& \forall_{\{u,t\}} 0 \leq U_{u,t} \leq Umax_u \\
& maxZ = \sum_s \sum_t \zeta_s D_{s,t} \\
& ((W_{i,j,t} \in \{0,1\}) B_{i,j,t} S_{s,t} D_{s,t} R_{s,t} U_{u,t}) \geq 0 \\
& \forall_{\{j,t \leq CARD_t\}} \sum_{i \in I(j)} \sum_{t'=t}^{t-p_i+1} w_{i,j,t'} \leq 1 \\
& \forall_{\{j,t \leq CARD_t\}} Vmin_j W_{i,j,t} \leq B_{i,j,t} \leq Vmax_j W_{i,j,t} \\
& \forall_{\{s,t \leq CARD_t\}} S_{s,t} = S_{s,t-1} + \sum_{j \in (i|j) \cap SO_s} \rho O_{i,s} B_{i,j,t-p_i} - \sum_{j \in (i|j) \cap SO_s} \rho I_{i,s} B_{i,j,t} + R_{s,t} - D_{s,t} \\
& \forall_{\{s,t \leq CARD_t\}} 0 \leq S_{s,t} \leq C_s \\
& \forall_{\{s,t \leq CARD_t\}} R_{s,t} \leq r_{s,t} \\
& \forall_{\{u,t \leq CARD_t\}} U_{u,t} = \sum_j \sum_{i \in I(j)} \sum_{\theta=1}^{p_i-1} (\alpha_{u,j} W_{i,j(t-\theta)} + \beta_{u,j} B_{i,j(t-\theta)}) \\
& \forall_{\{u,t \leq CARD_t\}} 0 \leq U_{u,t} \leq Umax_u \\
& \forall_{s,s \in FP} S_{s,t=CARD_t} \geq Dem_s \\
& \forall_{i,(j \in Ji) \wedge t \leq (CARD_t - p_i + 1)} W_{i,j,t} = 0 \\
& maxZ = \sum_s S_{s,t=CARD_t} \\
& ((W_{i,j,t} \in \{0,1\}) B_{i,j,t} S_{s,t} U_{u,t}) \geq 0 \\
& \forall_{\{s\}, s \in FP} ST_{s,t=CARD_t} \geq RDem_s \\
& \forall_{s,s \in FP} \sum_{i \in SO_s} \sum_j \sum_t \rho O_{i,s} B_{i,j,t} \geq Dem_s \\
& \forall_{\{i,t\}} (t - p_i) \leq MS
\end{aligned}$$

Figura 3. Modelo presentado para minimización del makespan usando la formulación formulación red estado-tarea.

La ecuación 1 a presentar como caso de estudio, corresponde a la seleccionada del sistema de ecuaciones de la figura anterior. Específicamente la ecuación describe el balance de masa para un estado s en un tiempo t, donde $\rho I_{i,s}$ y $\rho O_{i,s}$ son fracciones de masa para el consumo/producción de un estado s de la tarea i. Dentro de la cual marca un horizonte de tiempo donde se venderán los productos finales, ver ecuación 1.

$$\forall_{\{s,t \leq CARD_t\}} S_{s,t} = S_{s,t-1} + \sum_{j \in (i|j) \cap SO_s} \rho O_{i,s} B_{i,j,t-p_i} - \sum_{j \in (i|j) \cap SO_s} \rho I_{i,s} B_{i,j,t} + R_{s,t} - D_{s,t}$$

En base a esta ecuación se presenta el happy path del caso de uso correspondiente a la generación de la representación informal. Haciendo uso de pantallas tomas del sistema.

1. El usuario carga el archivo correspondiente al código mathml, generado en mathematica para la ecuación 1.
2. El sistema muestra cada uno de los términos identificados a ser capturados.
3. El usuario captura/selecciona los siguientes campos, ver ejemplo en la Figura 4:
 - a. Nomenclature. Descripción del término.
 - b. Mathematical quantity. Haciendo referencia a la clase Term de la Figura 1.
 - c. Type of variable. Haciendo referencia a la clase Term de la Figura 1.

Term	Nomenclature	Mathematical Quantity	Type of Variable
s	state	Variable	RealVariable
t	time point	Variable	IntegerVariable
CARDt	time horizon of makesp	Constant	
S		Set	
j		UnKnown	
i		Variable	
\mathfrak{S}		Constant	
SO		Constant	
qO		Constant	
B		Constant	
-1		Constant	
p		Constant	
qI		Constant	
R		Constant	
D		Constant	

Show

Figura 4. Captura de nomenclatura para cada uno de los términos, en el sistema informático.

4. El usuario da clic en el boton Show.
5. El sistema muestra la oración generada en base a la nomenclatura, como se muestra en la Figura 5.

Informal Representation

Expression:

for all state and time point less than or equal time horizon of makespan where set of states in domain state and time point equal unit summation task in unit that multiplies set of tasks intersection set of tasks producing in domain state summation mass balance coefficient for the production in domain task and state that multiplies batch size in domain task , unit and time point plus least one unit that multiplies fixed duration in domain task plus least one unit that multiplies unit summation task in unit that multiplies set of tasks intersection set of tasks producing in domain state summation mass balance coefficient for the consumption in domain task and state that multiplies batch size in domain task , unit and time point plus set maximum amount in domain state and time point plus least one unit that multiplies set of minimum amount in domain state and time point plus set of states in domain state and time point plus least one unit

Figura 5. Resultado de armar estructura de expresión matemática en MathML con la notación dada por el usuario.

Ahora bien podemos comparar el resultado de la Figura 5 con el párrafo anterior donde el experto de manera sencilla describe la expresión. La finalidad de realizar esta comparación es con el objetivo de identificar todos los elementos y que se valide que se hable de lo mismo. Una vez realizada la verificación podemos pasar a la siguiente fase que es generar el conocimiento mediante OMR.

5. Conclusiones

Mediante la integración de la gestión del conocimiento aplicada a los diferentes modelos matemáticos se ha creado una herramienta web que permite la representación informal para un entendimiento más fácil para el usuario. Además esta herramienta es explícita en el contenido de las diferentes expresiones matemáticas ayudando al entendimiento de los sistemas previamente concebidos.

Las principales etapas que se desarrollaron hasta lograr la representación informal de una expresión matemática son: procesamiento de entrada de archivo

para clasificar los elementos en términos y operadores, asociación de cada uno de los elementos con su correspondiente nomenclatura el análisis y procesamiento de estructura de content markup para enunciar una oración entendible al usuario. En cuanto a mejoras se ha probado y corregido diferencias en cuanto a la interpretación de la representación informal con un modelo matemático/ algoritmo del mundo real o dominio. Actualmente se esta trabajando en la gestión de la representación informal, como son las siguientes tareas: alta, bajas, cambios de modelos y nomenclatura. También se esta preparando la información necesaria a ser explotada en conjunto con el modelo OMR y generar la representación formal.

Este trabajo forma parte de un proyecto general que tiene como objetivo explotar el conocimiento matemático formal (entendible por la máquina) y la integración con otros modelos semánticos de diferentes dominios.

Referencias

Ausbrooks, R., Buswell, S., Carlisle, D., Chavchanidze, G., Dalmas, S., Devitt, S., Díaz, A., Dooley, S., Hunter, R., Ion, P., Kohlhase, M., Lazrek, A., Libbrecht, P., Miller, B., Miner, R., Rowley, C., Sargent, M., Smith, B., Soiffer, N., Sutor, R., Watt, S. (2010). *Mathematical Markup Language (MathML) Version 3 . 0, (October)*.

Capón-García, E.(2011).*Integrated scheduling decision making in enterprise wide optimization. Ph.D. thesis, Universitat Politècnica de Catalunya. pp.215-217.*

Chero, R. I., León, R. V., & Matemática, D. A. D. E. (2005). *Breve manual de Mathematica 5.1 (eumed.). Castilla - Piura, Perú.*

Gruber, T. R., & Olsen, G. R. (1994). *An ontology for engineering mathematics, Proceedings of the fourth international conference on principles of knowledge representation and reasoning. pp. 258–269, Gustav Stresemann Institut, Bonn, Germany.*

Maravelias, CT; Grossmann, IE.(2003).*Minimization of the makespan with a discrete-time state-task network formulation. AMER Chemical Soc, 1155 16TH ST, NW, Washington, DC 20036 USA*

Mcnavage, T. (2010). *JavaScript for Absolute Beginners*. Apress. doi:10.1007/978-1-4302-7218-2

Muñoz, E., Capón-García, E., Laínez-Aguirre, J. M., Espuña, A., & Puigjaner, L. (2014). Using mathematical knowledge management to support integrated decision-making in the enterprise. *Computers & Chemical Engineering*, 66, 139–150. doi:10.1016/j.compchemeng.2014.02.026

Nirosh. (11 de Dec de 2013). *Introduction to Object Oriented Programming Concepts (OOP) and More*. CodeProject, <http://www.codeproject.com/Articles/22769/Introduction-to-Object-Oriented-Programming-Concep#OOP>

Risberg, T., Evans, R., & Tung, P. (n.d.). *Developing a Spring Framework MVC application step-by-step*. Retrieved December 01, 2014, from <http://docs.spring.io/docs/Spring-MVC-step-by-step/>

Shah, N.;E.; Pantelides, C.; Sargent, R. A General Algorithm for Short-Term Schedduling of Batch Operations-II.Computational Issues. *Comput. Chem Eng.* 1993, 17, 229-244

The World Wide Web Consortium. (17 de Julio de 2014). *Extensible Markup Language (XML)*.W3C 's Ubiquitous Web Domain,<http://www.w3.org/XML/>

Keith, C. (2010). *Praise for Agile Game Development with Scrum*. Addison-Wesley Professional; 1 edition.

Wolfram Research. (2014). *Working with MathML*.WOLFRAM, <http://reference.wolfram.com/language/XML/tutorial/MathML.html>

Notas biográficas:



Viridiana Rodríguez Cardiel Ingeniero en Computación, egresada de la Universidad Autónoma de Zacatecas, en la ciudad de Zacatecas. Actualmente estudia la maestría en Ingeniería de Software en el Centro de Investigación en Matemáticas (CIMAT).

Ha trabajado en ambiente de desarrollo de software, ejecutando los roles de tester, analista, desarrollador y líder de proyecto. En el 2014 formó parte del

grupo evaluador de SCAMPI A, para obtener el nivel 2 de madurez del modelo CMMI-DEV v1.3. Su interés se centra en el desarrollo de software con calidad y gestión de conocimiento.



Edrisi Muñoz Mata Ingeniero industrial con especialidad en manufactura y Maestro en ciencias en ingeniería industrial con especialidad en calidad por el Instituto tecnológico de Orizaba (ITO) de México. Doctor en filosofía en ingeniería de procesos químicos por la Universidad Politécnica (UPC) de Cataluña de España. Su área de investigación principal es la gestión del conocimiento mediante el desarrollo de modelos ontológicos, sistemas de soporte a las decisiones en distintas áreas de proceso y optimización de procesos mediante el uso de modelos analíticos rigurosos. Actualmente es investigador asociado del Centro de Investigación en Matemáticas A.C. (CIMAT) de México, así como investigador invitado en el Centro de procesos y medio ambiente en la UPC. Su participación compete diferentes proyectos de investigación mexicanos y europeos. Participa en la publicación de diferentes artículos en revistas internacionales indexadas, así como en distintos congresos internacionales de renombre. Última publicación: Edrisi Muñoz, Elisabet Capón-García, José Miguel Laínez, Antonio Espuña, Luis Puigjaner, Integration of enterprise levels based on an ontological framework, Chemical Engineering Research and Design, Volume 91, Issue 8, August 2013, Pages 1542-1556, ISSN 0263-8762.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.