

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/228539357>

Proyecto ARGOS: Un SCADA en Software Libre

Conference Paper · March 2010

CITATIONS

0

READS

3,736

2 authors:



[Alejandro Pina Ortega](#)

Dyson Ltd

28 PUBLICATIONS 272 CITATIONS

[SEE PROFILE](#)



[Rafael Aiquel](#)

CVG Venalum

2 PUBLICATIONS 4 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



ARGOS Scada [View project](#)



EMANFES: Electric Machines Analysis with Finite Element Solvers [View project](#)

Proyecto ARGOS: Un SCADA en Software Libre

Alejandro Piña, Rafael Aiquel

Resumen - Un software de SCADA es un conjunto de herramientas informáticas de uso extendido en la automatización de procesos industriales. El propósito del proyecto Argos es ofrecer un SCADA desarrollado bajo la filosofía de proyectos de software libre. Con esto se busca promover el estudio del núcleo de un sistema SCADA y permitir el desarrollo del mismo hasta convertirse en una alternativa libre para la automatización industrial. El SCADA está implementado en plataformas con GNU/Linux y los componentes de software han sido desarrollados usando el lenguaje de programación C/C++, se utiliza XML como formato para los archivos de configuración, además de varias utilidades y librerías de software libre.

Palabras claves – Linux, Scada, Software Libre

I. INTRODUCCIÓN

Entre los elementos fundamentales de un SCADA (Control Supervisor y Adquisición de Datos, por sus siglas en inglés) se encuentran los componentes de software que realizan las tareas necesarias para monitoreo y control de procesos desde un centro de computo [1]. Dichos procesos pueden ser de distinta índole, pero comparten características comunes en estado transitorio, como lo son: la recolección de información desde diversos dispositivos, la toma de decisión de acuerdo a algún algoritmo de control y la asignación de órdenes a elementos con capacidad de ejecutar una acción.

El software de un SCADA le proporciona a los usuarios un conjunto de herramientas informáticas con las cuales se pueda diseñar, desarrollar, implementar y mantener sistemas para la supervisión, control y adquisición de datos, permitiendo de esta manera automatizar procesos industriales, integrar los distintos niveles de información, además de brindar la posibilidad de crear interfaces gráficas entre los operadores y las máquinas [2].

A.P. y R.A. están con la Industria Venezolana de Aluminio VENALUM, Gerencia de Investigación y Desarrollo, Av. Fuerzas Armadas, Zona Industrial Matanzas, Puerto Ordaz 8050, Venezuela, E-mail: alejandro.pina@venalum.com.ve, rafael.aiquel@venalum.com.ve

A.P. y R.A. están con la Universidad Católica Andrés Bello, Nucleo Guayana, Escuela de Ingeniería Informática, Puerto Ordaz 8050, Venezuela, E-mail: apina@ucab.edu.ve, raiquelg@ucab.edu.ve

y modificar el software sin restricciones. La filosofía de desarrollo del software libre permite que, gracias al trabajo cooperativo, un proyecto evolucione. Las libertades del software libre garantizan que un proyecto no pueda ser cancelado unilateralmente por las razones que fuesen, es decir, siempre y cuando existan interesados en continuar con el proyecto, éste seguirá desarrollándose [3]. En contraste, un proyecto de software propietario, si los promotores del mismo decidieran abandonarlo, su desarrollo no continuará.

II. SCADA EN CÓDIGO ABIERTO

A. Arquitectura

El proyecto Argos está siendo desarrollado para ser un SCADA con código abierto, asimismo, las herramientas que Argos proporciona actualmente sientan una base (con funcionalidades básicas) para implementar sistemas de supervisión en procesos automatizados.

La arquitectura clásica mostrada en la Figura 1, muestra como interactúan los componentes del software para formar un SCADA. Estos están distribuidos en la totalidad de la red de supervisión, aunque pueden existir aplicaciones en las que todos los componentes de software se ejecuten dentro de la misma estación de trabajo.

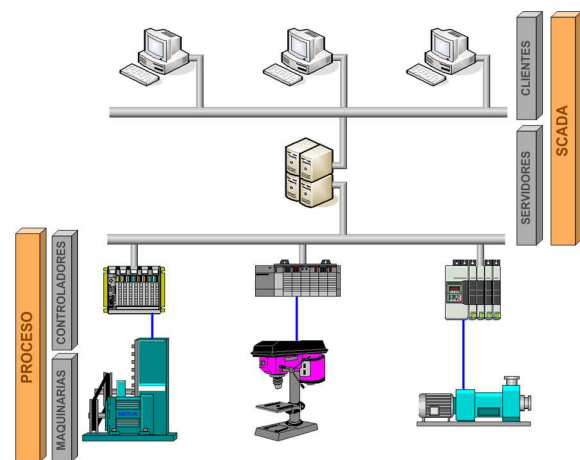


FIG 1. Arquitectura de un proceso automatizado

El software libre permite que el usuario pueda usar, probar

Argos se ha diseñado tomando en cuenta experiencias obtenidas durante el desarrollo e implementación del sistema supervisor de celdas de VENALUM [4], creando así una arquitectura (Figura 2) que permite adaptarse a los distintos esquemas de automatización moderna, en donde cada componente de software cuenta con estructuras de datos de alto rendimiento que operan de manera distribuida ya sea en una plataforma de red o en un mismo PC.

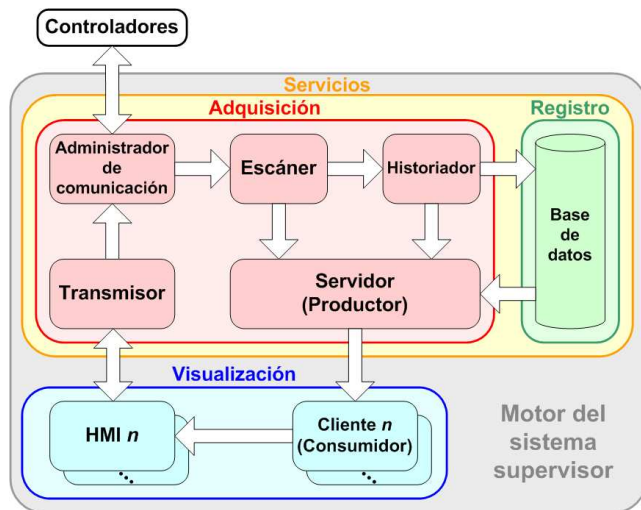


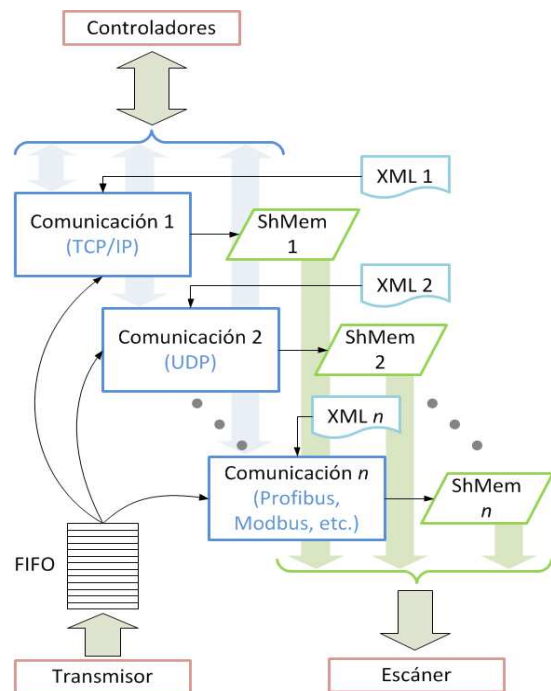
FIG 2. Organización de Argos

Entre las principales herramientas que se proporcionan en el SCADA de código abierto Argos, se encuentran los siguientes:

- **Comunicación**, estos procesos se encargan de establecer la comunicación con los equipos controladores de campo y pueden ejecutarse en uno o varios nodos. La importancia de estos servicios radica en la capacidad de manejar una comunicación efectiva con la gran diversidad de instrumentos presentes en un proceso automatizado[1], la realidad nos demuestra que estos equipos en su mayoría poseen distintos protocolos de comunicación que en algunos casos pueden ser cerrados o propietarios, mientras que otros pueden ser abiertos y estandarizados. En Argos se vienen desarrollando servicios con la capacidad de entender diversos protocolos y de esta manera organizar y centralizar las comunicaciones industriales de una planta. En la Figura 3 se observa la interacción de múltiples administradores de comunicación donde cada uno puede entender un protocolo distinto y manejar uno o más dispositivos de campo para luego a través de mecanismos de POSIX IPC

(Interprocess Communications) poder entregarle los datos a las demás herramientas del SCADA.

- **Escáner**, convierte los registros de todos los controladores en unidades de ingeniería, que posteriormente serán mostrados a los usuarios finales. Este servicio es uno de los que requiere mayor procesamiento ya que se encarga de recolectar los datos de cada una de las memorias de intercambio que están siendo actualizadas por los administradores de comunicaciones, para luego crear y modificar constantemente la base de datos de tiempo real en la cual se encontrarán las variables del proceso y las alarmas del sistema (Figura 4). Al igual que los demás servicios del SCADA Argos, el escáner se configura a través del procesamiento de ficheros en formato XML (Extensible Markup Language), los cuales tienen la información necesaria que permite el flujo de datos entre los múltiples servicios usando IPC.



TCP = Transmission Control Protocol

IP = Internet Protocol

UDP = User Datagram Protocol

FIFO = First In, First Out

Profibus, Modbus = Protocolos de Comunicación Industrial

FIG 3. Administradores de Comunicación

- *Historiador*, es un proceso configurado para almacenar información de manera permanente, principalmente usado para contar con gráficos de tendencias e históricos de alarmas y eventos (Figura 5). Este servicio toma la información a una frecuencia configurable desde la base de datos de tiempo real, con esto se encarga de mantener un buffer (Figura 6) para cada una de las variables que se están registrando, posteriormente hace el volcado de los datos en un medio de almacenamiento permanente (Figura 7), esto a través de un servicio dedicado a guardar la información utilizando para ello el motor de base de datos PostgreSQL.

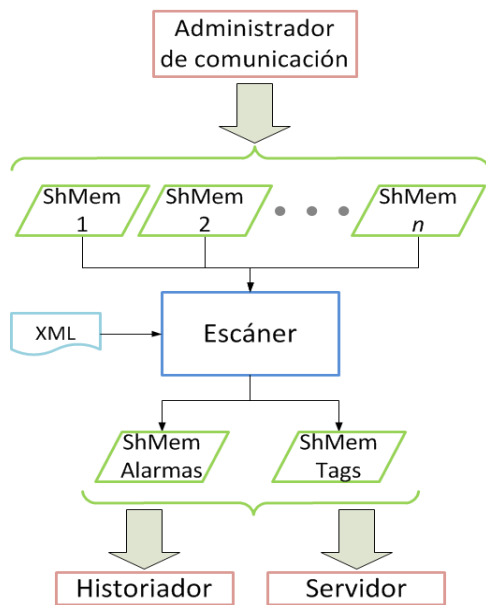


FIG 4. Escáner del SCADA

- Los procesos *Servidores* y *Clientes* se encargan de la transferencia de información, ya sea entre nodos localizados remotamente, o que se estén ejecutando en el mismo servidor (Figura 8), para que pueda ser presentada al operador. Actualmente Argos se basa en la filosofía de productor-consumidor para disminuir el tráfico en la red debido a los grandes volúmenes de información que se manejan en este tipo de sistemas, haciendo uso de grupos multicast los procesos productores ponen a disposición los datos en la red, para que así todos los consumidores que estén ligados al grupo puedan obtener la información que necesiten y que haya sido configurada a través de los ficheros en XML.

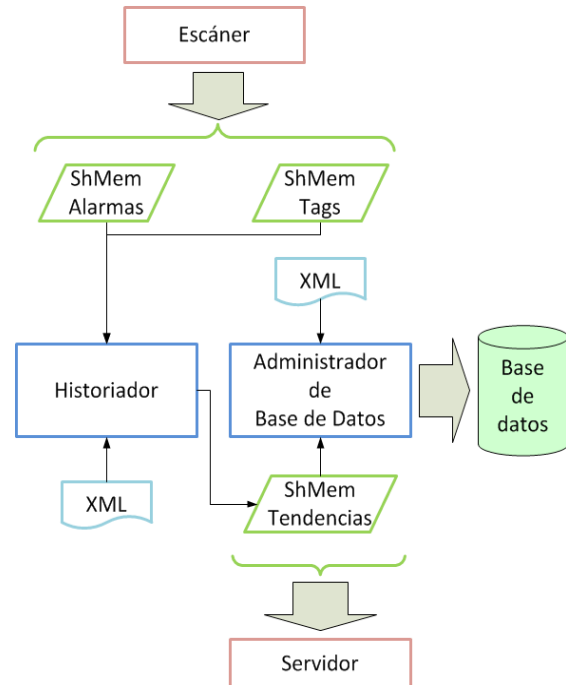


FIG 5. Registro de Alarmas y Tendencias

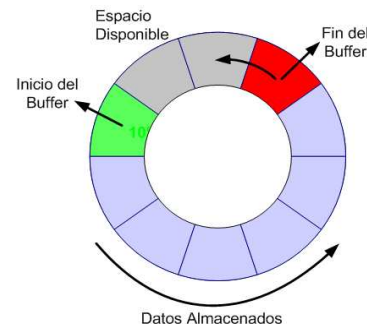


FIG 6. Buffer Circular de Variables del Proceso

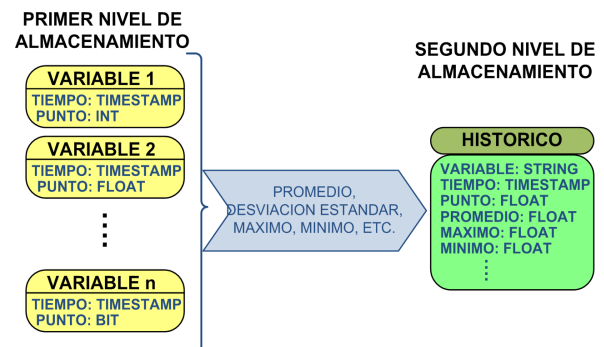


FIG 7. Modelo de Almacenamiento

- *Transmisor*, su única función será escribir registros en el controlador. Este proceso escucha las peticiones a través de conexiones TCP con los servicios HMI, los cuales canalizan los requerimientos de escritura de los usuarios del sistema SCADA. Cuando el transmisor recibe la petición coloca un mensaje en la estructura de tipo FIFO (POSIX) correspondiente a cada administrador de comunicación, que a su vez será el encargado de la operación de escritura en el instrumento de campo.

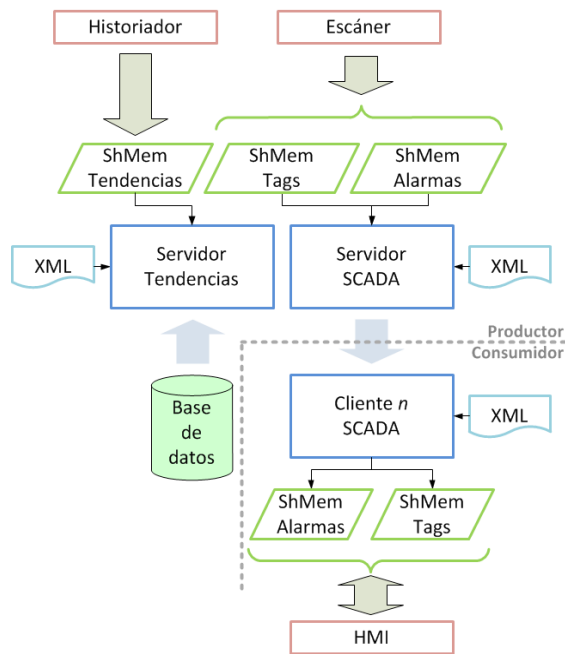


FIG 8. Relación Productor-Consumidor

- Por último, los procesos *HMI* serán los encargados de desplegar la información adquirida a los usuarios finales, mediante distintos recursos gráficos, tal y como se muestra en la Figura 9.

Para que el sistema supervisor a implementar pueda contar con todas las variables del proceso, se ha diseñado una base de datos con un mínimo tiempo de acceso. Esta estructura de almacenamiento debe permitir búsquedas de datos de forma rápida y precisa, por tal razón se decidió utilizar una estructura de datos de tipo Tabla Hash, la cual posee las características idóneas para satisfacer los requerimientos del sistema.

Una tabla *hash* (Figura 10) es un conjunto arbitrario de elementos agrupados sin ningún tipo de ordenamiento

físico aparente. Esto disminuye los tiempos de inserción de elementos (en nuestro caso variables del proceso) debido al ahorro en los procesos de ordenamiento, que normalmente son efectuados en otras estructuras de datos para optimizar los tiempos de búsqueda. El método de búsqueda de hashing no es basado en comparaciones, en lugar de navegar por las estructuras cotejando palabras claves con las claves en los elementos, se intentan ubicar elementos en una tabla directamente haciendo operaciones aritméticas para transformar claves en direcciones sobre la tabla a través de las funciones hash accediendo normalmente a las variables del proceso almacenadas en la tabla con sólo una operación. Como resultado se obtiene una alta disponibilidad de datos, lo que permite que se pueda mostrar lo que está sucediendo en el proceso en tiempo real, a través de las interfaces de visualización.

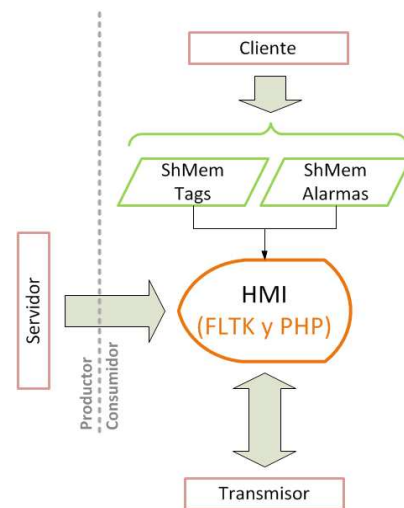


FIG 9. Interfaz para Visualización

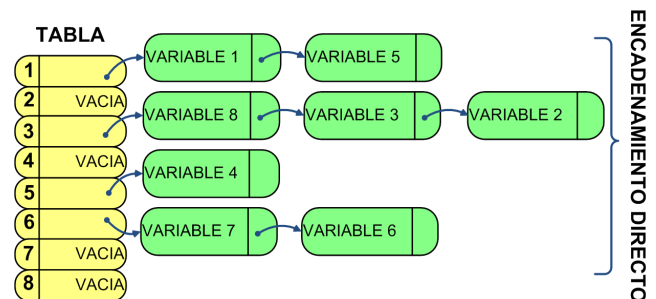


FIG 10. Estructura de Base de Datos de Tiempo Real

Cuando existen variables que deben ser almacenadas por largos períodos de tiempo, se hace uso del motor de base de datos. Sin embargo, debido a la alta frecuencia de adquisición de datos en algunos casos se hace imposible guardar directamente cada valor de variable de proceso, por ende, Argos provee algunas herramientas que usan un buffer circular (Figura 6) para realizar operaciones estadísticas con los datos, para luego ser almacenados siguiendo el modelo mostrado en la Figura 7.

Existen básicamente dos grupos de tablas, en el primero se almacenan las variables de acuerdo a una frecuencia de muestreo preestablecida (se pueden establecer diferentes frecuencias de muestreos por grupos de datos según lo requiera el proceso), mientras que el otro grupo contiene la información histórica de todas las variables almacenadas. El proceso de almacenamiento es llevado cabo de la siguiente manera: en cada instante de tiempo y de acuerdo a su frecuencia de muestreo se van almacenando los valores en las tablas del primer grupo. Después de un tiempo preestablecido, se realiza una transacción donde se transfieren los datos almacenados hacia las tablas del segundo grupo, que contienen la información procesada y permanente. Es importante acotar que los datos antes de ser transferidos son sometidos a un proceso de análisis estadístico para evitar almacenar grandes volúmenes de datos sin que estos tengan información relevante.

El diagrama de despliegue [5] de la Figura 11 muestra una de las múltiples disposiciones de los nodos propuesta por Argos, que incluyen elementos de hardware, así como también los componentes de software, evidenciando como interaccionan.

B. Bibliotecas

El Proyecto Argos hace uso pleno de otros proyectos de software libre [6], como lo son: las bibliotecas muParser, tinyXML, FLTK y su ambiente de desarrollo Fluid, así como el software de base de datos PostgreSQL con sus API (Application Programming Interface)).

Esto ha facilitado el desarrollo del proyecto, ya que, son componentes probados que reducen el trabajo requerido y apalancan las funcionalidades del sistema.

MuParser: es una biblioteca para el análisis de expresiones matemáticas. Es usado en Argos para calcular variables de procesos a partir de otras usando operaciones matemáticas.

TinyXML: se usa para el manejo de los archivos de configuración con formato XML.

FLTK: las interfaces gráficas están basadas en esta biblioteca. Se eligió por su ligereza para garantizar el rendimiento en computadores de recursos limitados, como sistemas embebidos. Se está implementando el uso de SVG para facilitar el desarrollo de las interfaces gráficas.

PostgreSQL: Es una de las bases de datos de software libre más utilizadas, es un proyecto probado y con una comunidad extensa. Se usa la API para integrar la base de datos al sistema.

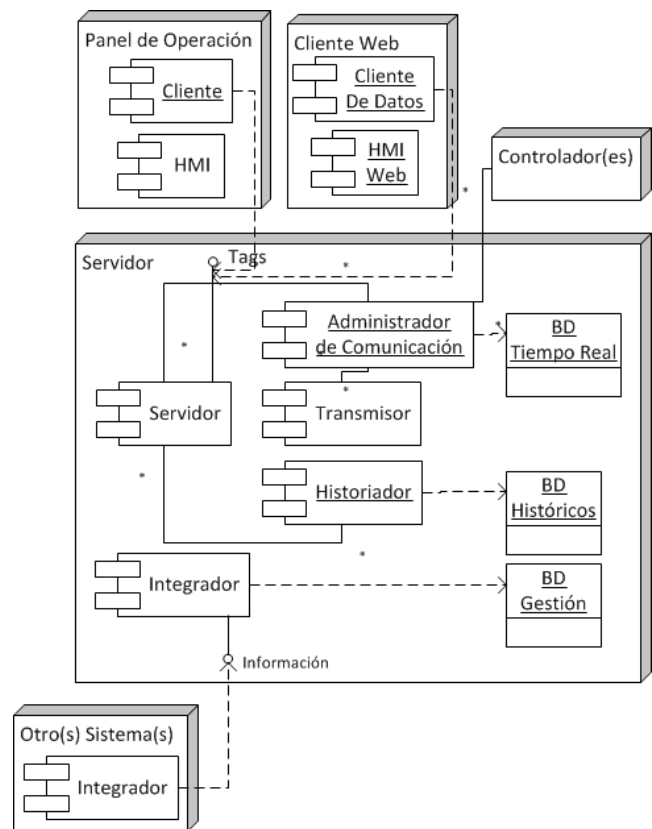


FIG 11. Diagrama de Despliegue de los Componentes

III. RESULTADOS

Todas las herramientas que proporciona Argos han sido desarrolladas y probadas en estaciones de trabajo de Laboratorio con las siguientes características: x86 y x86_64 con procesadores Pentium IV, AMD 64 y 1Gb de memoria RAM.

Actualmente el software está disponible como un proyecto de código abierto [6], bajo licencia GPL, en el repositorio SourceForge.NET, su estado es ALPHA y en continuo desarrollo para generar los candidatos a lanzamiento (RC)

4TO CONGRESO IBEROAMERICANO DE ESTUDIANTES DE INGENIERÍA ELÉCTRICA (IV CIBELEC 2010)
5TAS JORNADAS DE INGENIERÍA ELÉCTRICA (V JIELECTRIC 2010)

y cuenta con los siguientes avances:

- Procesos de adquisición y administración de variables, eventos y alarmas.
- Procesos de envío y recepción de datos con los clientes HMI.
- Procesos para el almacenamiento en base de datos de las variables, eventos y alarmas.
- Capacidad para configuración a través de archivos en formato XML para todos los procesos anteriormente mencionados.
- Objetos gráficos diseñados para ser incorporados en aplicaciones de Ventanas usados para desplegar la información de las variables del proceso (como clientes HMI).

Para desarrollos futuros orientados a cubrir la mayoría de los requerimientos del sector industrial, en lo que a sistemas de control supervisor se refiere, se consideran los siguientes requerimientos:

- Desarrollo de manejadores para la adquisición de datos de la gran gama de dispositivos de instrumentación y control que actualmente se encuentran en el mercado e instalados en la mayoría de las plantas industriales.
- Simuladores para poder diseñar y desarrollar todo un sistema SCADA sin la necesidad de estar conectado a hardware industrial.
- Diseño y desarrollo de un entorno integrado para la configuración e implementación de todas las herramientas en un sistema SCADA.
- Diseño y desarrollo de un entorno que permita la creación de interfaces Web dinámicas compatibles con navegadores Web estándar.
- Procesos que administren la redundancia del sistema SCADA, asegurando de ésta manera la

robustez del control supervisor sobre un proceso productivo.

IV. CONCLUSIONES

El proyecto Argos ha sido desarrollado para ser una alternativa en el área de automatización industrial basada en herramientas de software libre. De esta manera pueda ser utilizado, estudiado y modificado por las distintas comunidades de conocimiento libre y desarrolladores de software, siguiendo la filosofía de los proyectos de código abierto.

Asimismo, el desarrollo comunitario permitirá proporcionar una arquitectura de software robusta y tecnológicamente avanzada, tal y como lo requiere el demandante sector industrial para lograr optimizar procesos productivos.

El modelo de desarrollo propuesto para el proyecto Argos permitirá la evolución del software de SCADA en un ambiente de colaboración, teniendo como punto de partida el desarrollado realizado por un equipo de investigación y desarrollo del centro de innovación tecnológica del aluminio, CINTAL, usando en su totalidad herramientas libres.

REFERENCIAS

- [1] A. Rodríguez. “**Comunicaciones Industriales**”. Marcombo, Primera Edición, 2008, pp. 256-270.
- [2] J. Hernández., L. León. “**Implementación de Objetos Gráficos para el Desarrollo de Despliegues Operacionales en el Sector Comercio y Suministro del SCADA Nacional de PDVSA**”. Tesis de Grado, Escuela de Ingeniería de Sistemas, Universidad de los Andes, Mérida, Venezuela, 2008.
- [3] J. M. Hernández. “**Software Libre: Técnicamente Viable, Económicamente Sostenible y Socialmente Justo**”. Infonomia, primera edición, 2005, pp. 43-63.
- [4] Abaffy, C., Lárez, J., Aiquel, R., González, J. “**CVG Venalum Potline Supervisory System**”. Artículo Técnico, TMS 2006, pp. 301-305, 2006.
- [5] I. Besembel, J. Montilva. “**Modelado de Sistemas usando UML 2.0**”. Centro de Excelencia en Ingeniería del Software (CEISOFT), 2006.
- [6] CINTAL, “**Argos el SCADA en Software Libre**”. <http://www.cintal.com.ve/argos/>, 2009.