Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Direct Load Control System Design Using Active Database

Sang Yule Choi* and Kwang Y. Lee**

**Induk University, Seoul, Korea*
(Tel: 017-227-8606; e-mail: ppk99@induk.ac.kr).
***Baylor University, Waco, TX 76706 USA (e-mail: Kwang_Y_Lee@baylor.edu).*

Abstract: The electric utility has the responsibility of reducing the impact of peaks on electricity demand and related costs. Therefore, they have introduced Direct Load Control System (DLCS) to automate the external control of shedding customer load that it controls. Since the number of customer load participating in the DLC program are keep increasing, DLCS operators are facing difficulty in monitoring and controlling customer load. The existing DLCS requires constant operator intervention, e.g., whenever the load is about to exceed a predefined amount, it needs operator's intervention to control the on/off status of the load. In the case of increasing number of customer load, it may happen that the operator fails to control the on/off status of the load due to operator's miss judgment. Therefore, a state-of-the-art DLCS is needed, which can control automatically the on/off status of the customer load without intervention as much as possible. This paper presents a DLCS design method using the active database. The proposed DLCS consists of two parts in applying the active database to DLCS. One is to define the production rules which can avoid operator's intervention as much as possible. Another is to design a system architecture which can be easily applied to DLCS. To demonstrate the validity of the proposed system, variable production rules and DLCS design architecture are presented.

Keywords: Direct load control, Production rule, Active database, Design architecture.

1. INTRODUCTION

In Direct Load Control (DLC) programs, a utility as a programs' sponsor can remotely interrupt customer load at the time of peak load by DLCS operators. In return for customer's inconvenience, utility gives an incentive payment or credit. The existing DLCS has some system architectural weaknesses with regard to DLC system management, e.g., its system architecture is designed using a *passive database* system. A passive database has been widely used as a reporting tool and a monitoring tool because of its easiness for handling massive amounts of data. But its commands are executed by the database query (or delete or update) which is only requested by the user or program. Whenever the conventional DLCS forecasts that the amount of customer load is going to exceed a predefined amount, it remotely controls the on/off status of the load by a DLCS operator's request or DLC program. Likewise, the conventional DLCS always needs constant operator's attention and intervention. Therefore, it may happen that even an experienced operator fails to control the on/off status of the load due to his/her miss judgment.

On the other hand, *active database* supports automatic mechanisms that enable active database by itself to monitor and react to specific circumstances of relevance to an application (Widom and Ceri, 1996). A various applications of active database have been presented for software engineering, especially, testing, container packing and loading, order processing, stock market, financial activities and air traffic control system (Chandra and Segev,1994; Lin et al., 1997). An active database application for power system is introduced to energy management system by Baralis et al. (1994), where the authors presented various production rules for power distribution network topology to assist the operator's completing the network topology after redistributing power. Recently, a web based active database application to distribution automation system is introduced (Choi, 2006), where the author have presented production rules for loss minimization and web based system architecture. Active database system has a common approach for the knowledge model using production rule which consists of three components: an event, a condition and an action. By using the production rule, the DLCS database itself can be updated without operator or program's intervention. Therefore, the DLCS based on active database can control automatically the on/off status of the customer load without operator's intervention.

This paper, presents a DLCS design method using the active database to manage massive customer data in an active manor and control automatically the on/off status of the customer load at the time of peak load without operator's intervention as much as possible. Main steps to design the new DLCS have been implemented in two stages. First stage is to define the production rules which can avoid operator's intervention as much as possible and manage massive customer load data actively. Second stage is to design system architecture using active database, which can be easily applied to the conventional DLCS. A load metering device (LMD) and a man-machine interface for the main system are implemented to demonstrate the validity of the proposed production rules and DLCS design architecture.

2. A PRODUCTION RULES FOR DLCS

2.1 The Production Rule

The production rule paradigm originated in the field of Artificial Intelligence (AI) with expert systems languages such as OPS5 (Kang and Kim,2004). The production rule consisted of three components: an event, a condition and an action. When customer load data in the database is updated on-line, there are three kinds of component procedure for monitoring the updated data. An *event* component detects online update of the customer data. And a *condition* examines update's validation. If the condition has evaluated the update to be true, then an *action* is carried out for the described task. By using the E-C-A (Event-Condition-Action) rules, a DLCS using active database can monitor and control the status customer load without user's intervention. Generally, an E-C-A rule has the following features:

• Event: An event describes a happening to which the rule may be able to respond and it occurs when data in database is changed by the database query (update, insert, delete) or application program. One event can trigger more than one rule.

• Condition: When rules are triggered by an event, the condition part of the rule checks the event's context. If the condition is satisfied, then the action of the rule is executed.

• Action: An action specifies what to be carried out by the rule when the rule is triggered and its condition is satisfied.

It can be sequence of retrieval and modification commands over any data in the database and it can cause triggering another rules and application programs. An example of applying for DLCS is as follows:

Rule) If a customer load demand is going to exceed its projected amount of load then DLCS disconnect the customer load:

- Event: Update to customer load demand data

- Condition: customer load demand data > projected customer load data

- Action: Disconnect the customer load for the brief period of time.

2.2 The Requirement Gathering and Conceptual Design For DLCS Database

The main objective of requirement gathering is collecting the data used by an organization, identifying relationships between data and future data need. This process is initialized by interviewing the user of the organization. The required data tables and attributes for DLCS database is as follows:

- Load Metering Device (LMD): LMD ID, Customer ID, Load ID, On/off status (1: on, 0: off), Present active power, Present accumulated active power, Projected accumulated active power, Forecasted accumulated active power, Control failure (0: normal, 1: failure), Communication failure (0: normal, 1: failure).

- Customer: Customer ID, Name, Address, Phone number, The number of load, On/off status (1: on, 0: off), customer type (residential: 1, commercial: 2), Present Active Power. Present accumulated active power, Projected accumulated active power, Forecasted accumulated active power, Control failure (0: normal, 1: failure), Communication failure (0: normal, 1: failure).

- Load: Load ID, Customer ID, LMD ID, On/off status (1: on, 0: off), Present active power, Control failure (0: normal, 1: failure), Communication failure (0: normal, 1: failure)

- Report: Report ID, Customer ID, Controlled load ID, Time of control ID, Control failure load ID, Control failure LMD ID

- Time of Control: Time of control ID, Month, Day, Year, Time

Where, *Projected accumulated active power*: Accumulated active power set by DCLS operator to avoid the customer load demand at the time of peak.

Forecasted accumulated active power: Accumulated active power set by the customer load forecasting application program.

Each tables presented above have the relationships between tables. To illustrate the relationships between the presented tables in a DLCS database, Entity-Relationship(ER) diagram is used in Fig.1.

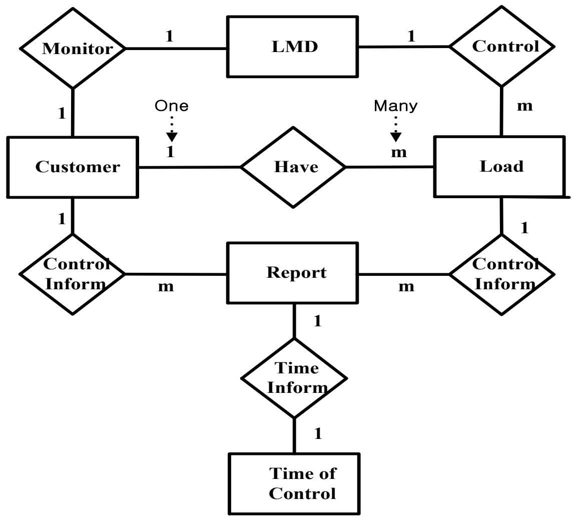

Fig. 1. Entity-Relationship(ER) diagram for DLCS database. (1:1 implies one to one mapping and 1: m implies one to many mapping.)

In Fig. 1, e.g., one LMD device controls many loads. Therefore, the relationship type is "control" and 1: m mapping.

2.3 Production Rule Definition

An LMD installed in a customer collects the different kinds of state data of electrical equipment such as air conditioners, lighting, motors and pumps. Every customer has one LMD device which collects data and sends them to the DLCS database on-line. A production rule checks the telemetered data and fires rules to ensure accuracy and consistency of the data in DLCS database, and it also actives DLC application program to control the on/off status of the load.

2.3.1 Production Rules for Control Failure Detection

When the on/off control between main system and LMD is failed, control failure device detects the failure and activates alarm notification to let the system operator knows about the control failure situation. Conventionally, one fault can be detected by several monitoring devices and activates different kinds of notification. But it needs different kinds of devices to complete alarm processing. Therefore, control failure detection production rules are presented to manage alarm processing easily by using database itself.

Rule 1) When the on/off status of LMD in DLCS database is updated to *on* by an operator, it triggers the updating of the on/off status of Customer.

Rule 2) When Active power of LMD in DLCS database is updated, it triggers the updating of Active power of Customer.

Rule 3) When Present accumulated active power of LMD in DLCS database is updated, it triggers the updating of Present accumulated active power of Customer.

Rule 4) When Forecasted accumulated active power of LMD in DLCS database is update, it triggers the updating of

Forecasted accumulated active power of Customer.

Rule 5) When the on/off status of Load is updated to *on* from *off* by an operator or DLC application program, the present active power is supposed to be changed from zero to metered value. But if it doesn't, it means a control failure has occurred between the main system and Load:

- Event: update the on/off status

- Condition: updated (Load), (New on/off status = On) && (Present active power = 0) && (Communication failure = normal)

- Action: update (Control failure = failure)

where, &&: the AND operator

Rule 6) When the on/off status of Load is updated to *off* from *on* by the operator or DLC application program, the present active power is supposed to be changed from metered value to zero. But if it doesn't, it means a control failure has occurred between the main system and Load.

Rule 7) When the control failure of Load is updated to failure, it triggers updating for the control failure of LMD as the same that of Load.

Rule 8) When the control failure of Load is updated to failure, it triggers updating for Control failure load ID and customer ID of Report:

- Event: update control failure

- Condition: update (Load), New control failure = failure

- Action: update {(control failure load ID of Report == Load ID) && (Customer ID of Report= Customer ID of Load)

Rule 9) When the control failure of Load is updated to failure, it activates alarm notification to let the system operator knows about the control failure situation:

- Event: update control failure

- Condition: update (Load), New control failure = failure

- Action: Activate alarm Notification

2.3.2 Production Rules for Direct Load Control

When an LMD device sends the total accumulated active power of customer to DLCS and updates the LMD in database on-line, the main system of DLCS calculates the rate of increase for the accumulated active power and forecasts the total accumulated power of load during 900 seconds. Production rules monitor the present state of customer by detecting updated data. If the total forecasted accumulated active power is about to exceed the projected accumulated active power during 900 seconds, then the production rules activate the on/off load control by using the DLC application program and the production rules for direct load control. If a customer has many electrical equipment and its total forecasted accumulated active power is about to exceed its projected accumulated active power, then DLCS have to choose which electrical equipment has to be turned off or cycled to meet the projected amount of active power. The DLC application program decides which electrical equipment among loads to be controlled when total forecasted accumulated active power of a customer is going to exceed its projected amounts.

Rule 9) When LMD sends the present accumulated active power to the DLCS database on-line, the sent data have to be validated by checking the on/off status and communication failure condition:

- Event: send the present accumulated active power to database

- Condition: (the on/off status = on) && (communication failure = normal)

- Action: update the present accumulated active power

Rule 10) When the present accumulated active power is updated, the DLCS activates Forecasting program which calculates the rate of increase for active power and forecasts the total accumulated power of load during 900 seconds.

Rule 11) When Forecasting program presents the forecasted accumulated active power value, that value is updated to LMD in the DLCS database

Rule 12) When the forecasted accumulated active power is about to exceed the projected accumulated active power during 900 seconds, DLC application program is executed to meet the projected accumulated active power by choosing an electrical equipment to be turned off or cycled.

Rule 13) When DLCS operator updates the projected accumulated active power of specific customer for special reasons (e.g., customer demand or reducing the impact of peaks on electricity demand), DLC application program is executed if new updated projected accumulated active power of specific customer is less than the forecasted accumulated active power of that customer:

- Event: update projected accumulated active power

- Condition: update (LMD), New projected accumulated active power < forecasted accumulated active power

- Event: execute DLC application program.

Rule 14) When a DLC application program is ended it presents numbers of load ID to be turned off or cycled and triggers updating of the on/off status of load:

- Event: DLC application program

- Condition: True

- Action: update (the on/off status of load = *off,* where the load is to be turned off by DLC application program) && (the on/off status of load = *on,* where the load is to be turned *on* by DLC application program)

Rule 15) When the on/off status of load is updated by the DLC application program, it triggers updating of the Customer ID, Controlled load ID and Time of control ID of Report:

- Event: update the on/off status of load.

- Condition: update (Load), (New on/off status of load = result of DLC application program.)

- Action: update (the customer ID of report) && (Controlled load ID of Report) && (Time of control ID of Report)

Rule 16) When the on/off status of load is updated to *on* by the DLC application program, it triggers to turn on the load in customer by sending *on* signal to the LMD device:

Rule 17) When the on/off status of load is updated to *on* by the DLC application program, it triggers to turn on the load in customer by sending *on* signal to the LMD device.

3. PRODUCTION RULE MANAGER FOR DLCS

When an action of rule is executed, it may triggers several other rules because it can be an event of other several rules. And newly triggered rules can trigger others in the same manor. If several rules are triggered simultaneously, it requires some techniques to manage and evaluate when specified the multiple events. An active rule manager has a responsibility to monitor and optimize every event including database events and application program events by coordinating interaction between the rules. It also coordinates the execution of active rules during transaction by interfacing with the constraint manager designed for on line data constraint. To coordinate interaction between the rules, Triggering Graph (Baralis et al., 1998) is used to represent interaction between rules.

DEFINITION: Let R be an arbitrary active rule set. The triggering Graph is a directed graph {V,E}, where each node $v_i \in$ V corresponds to a rule $r_i \in$ R, A directed arc $<_j r_j , r_k >$ $\in$ E means that the action of rule r_j generates events which trigger rule r_k .

3.1 Triggering Graph for Control Failure Detecting

When a DLC operator or DLC program updates the on/off status of load to *on* from *off*, active power is supposed be changed to the metered value from zero. If it is not changed under normal communication, it triggers Rule 5 which updates failure for the control failure of load. The control failure update triggers Rule 7, Rule8 and Rule 9 alternately, and it update Report, LDM control failure and activate alarm to let the operator knows the control failure situation. A triggering graph for control failure detection is shown in Fig. 2.

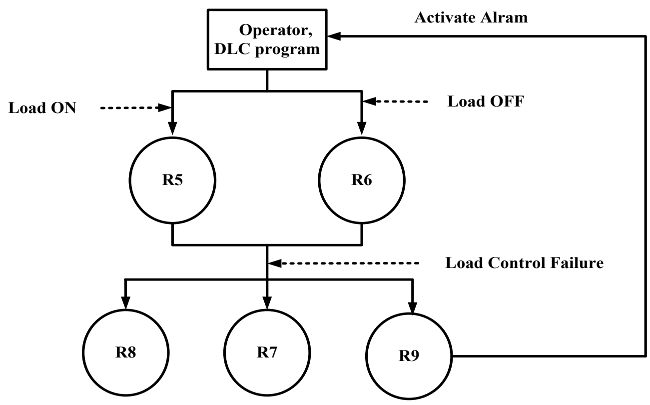

Fig. 2. Triggering graph for control failure detection. (R implies the Rule Number.)

3.2 Triggering Graph for Direct Load Control

When an LMD device sends an accumulated active power, it triggers Rule 9 to validate date and updates the present active power of LMD in the database. The main system calculates the forecasted accumulated active power (Rule 10) and it triggers Rule 11 to update the forecasted accumulated active power of LMD. If updated one is about to exceed the projected one, Rule 12 is triggered DLC program to find loads to be turned off or cycled. After loads to be turned off or cycled are found, Rule 15, Rule 16 and Rule 17 are triggered to send the switch on/off signal to LMD Device. If a load chosen to be turned off has metered value after a DLCS sent the turned off signal to the LMD, then it triggers

Rule 6 to activate alarm signal to let the operator knows a control failure situation. A triggering graph for DLC is shown in Fig. 3.

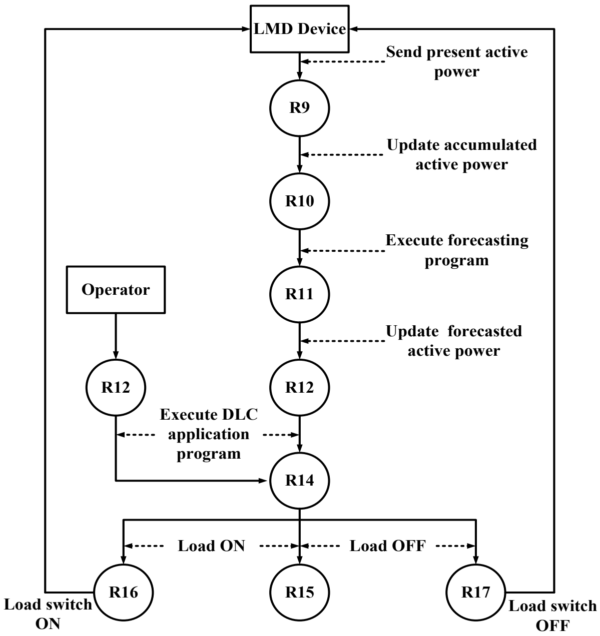

Fig. 3. Triggering graph for DLC. (R implies the Rule Number.)

3.3 DLCS Design Using Active Database

In this section, the authors present functional architecture design for a DLCS using active database. The architecture consists of three components, and each component has its own functionality. The functionality of components as follows

- LMD device

An LMD device installed in customer collects the on/off status, an accumulated active power, present active power from loads and sends them to the DLCS database on-line. And it controls the on/off status of load according to the operator or DLC program control signal.

- Production rule manager

A production rule manager validates the integrity constraint of sent data from LMD device and data modification between data tables in the DLCS database. It also decides which rule should be fired when various events are detected and manipulates output data from the DLCS application programs.

- Production rule

A production rule provides an automatic carried out mechanism if certain events including integrity constraint enforcement, control failure, load forecasting and DLC application program are detected, and the condition is evaluated to be true. A DLCS design using active database is presented in Fig. 4

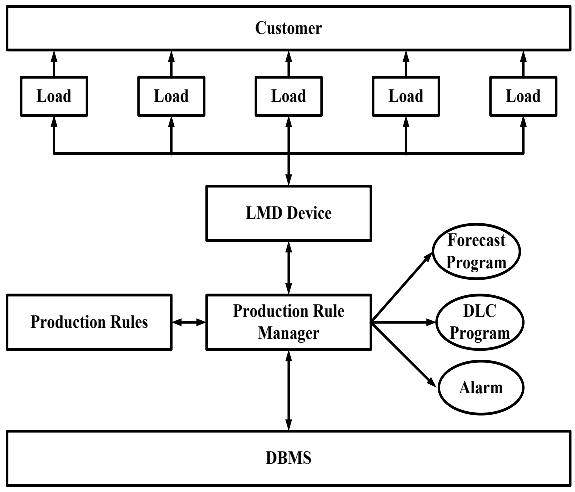

Fig. 4. DLCS design using active database.

4. THE DEVELOPMENT OF DLCS USING ACTIVE DATABASE

An LMD device and the man-machine interface of the main system are developed to demonstrate the validity and effectiveness of the proposed DLCS design.

4.1 LMD Device

The electric equipment to be controlled by a LMD device is classified by their characteristics such as air conditioner, ventilation air pumps, an office light, an underground parking lot, a stairway light, etc. And customer can put a priority on each equipment to be turned off when a forecasted active power is about to exceed to the projected active power. The device can collects loads data and sends them to the main system by using wireless network. The developed LMD device is shown in Fig.5

Fig. 5. The LMD device.

4.2 Man-Machine Interface for DLCS.

The menu of main system consists of 10 categories such as connect to LMD, disconnect to LMD, wireless setting, upgrade, finish, monitoring LMD, Report, environment settings, choosing customer, and communication log. The developed man-machine interface (MMI) for DLCS is presented in Fig. 6.

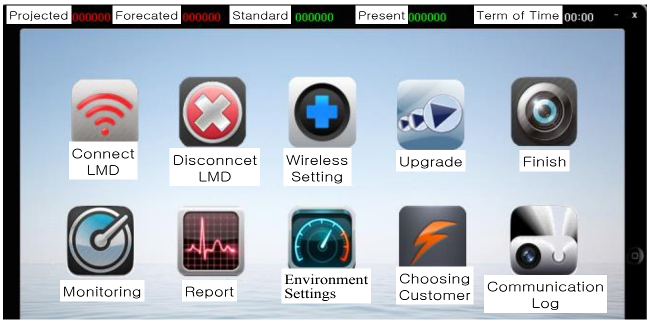

Fig. 6. MMI for DLCS.

4.3 Case Study

Eleven loads of a customer are used to validate the effectiveness of the proposed DLCS design. When an LMD device sends present active signal to the DLCS database, Rule 9, Rule 10, and Rule 11 are triggered. If the forecasted accumulated active power is exceed to the projected accumulated active power, it triggers Rule 12, Rule 14, and Rule 17 consecutively and send load off signal to LMS device. If forecasted accumulated active power is far below form the projected accumulated active power, it triggers Rule 12, Rule 14, and Rule 16 consecutively and send load on signal to LMS device. A load control following to the forecast is shown in Fig. 7.

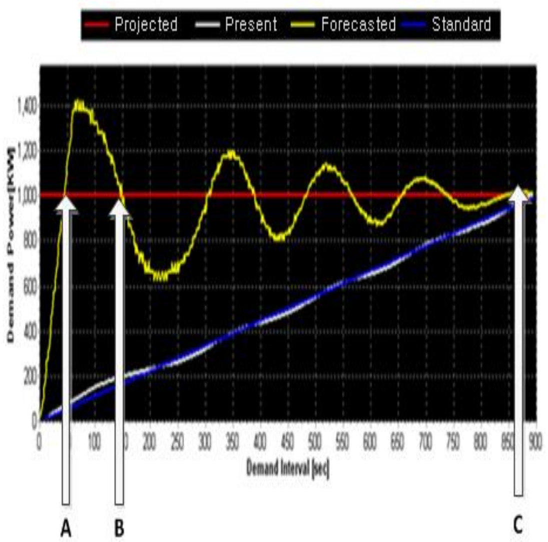

Fig. 7. Load control following the forecast.

In Fig. 7, from point A, the forecasted active power exceeds the projected one, therefore Rule 12 is triggered to execute the DLC program to find loads to be turned *off.* On the contrary, from point B, the forecasted power is far below from the projected one, so Rule 12 is triggered to find loads to be turn *on.* The point C is the end time of 900 seconds and all kinds of active power (the projected, the present, the forecasted and the standard) are converged to one another at point C, which means the accumulated active power is equal to the projected one.

5. CONCLUSIONS AND FUTURE WORK

The conventional DLCS always needs operator's intervention whenever customer load demand exceeds a predefined amount. This paper presents a DLCS design method using the active database to avoid operator's intervention as much as possible. A various production rules and a production rule manager are presented. And an LMD device and man-machine interface is also developed. Simulation results on eleven loads of a customer shows that the proposed DLCS design provides a good performance on the DLCS system without operator's intervention. To validate the proposed DLCS design more realistically, more loads and customers are needed. Since the developed LMD device is only a prototype device and it needs to be tested with more customers

REFERENCES

Baralis, E., Ceri, S., Monteleone, G., and Paraboschi, S. (1994). An intelligent database system application: The design of EMS. *In Proceedings of the First International Conference on Application of Database*.

Baralis, E., Ceri, S. and.Paraboshi, S. (1998). Compile-Time and Runtime Analysis of Active Behavior. IEEE Trans, on Knowledge and Data Engineering, Vol. 10, No.3, pp.353 – 370.

Chandra, R., Segev, A.. (1994). Active database for financial application. *In Proceedings of the Fourth International Workshop on Research Issues in Data Engineering,* pp.264-273.

Choi, S. (2006). Power Distribution Automation System Using Information Technology Based Web Active Database. *Lecture notes in Computer Science* 3984, pp. 354-364.

Kang, J. A., Kim, A. M. (2004). Shortening Matching Time in OPS5 Production Systems. *IEEE Trans. Software Engineering,* Vol. 30, No. 7, pp.448-457.

Lin, K. J., Peng, C. S., Ng, T. (1997). An active real-time database model for air traffic control systems. *In Proceedings of the Second International Workshop on Active, Real-Time, and Temporal Database Systems*, Como, Italy, pp.73-97.

Widom, J., Ceri, S. (1996). Active Database Systems: Triggers and rules for Advanced Database Processing. *Morgan Kaufmann Publishers.*