Research on Awareness Language for Creating Group-oriented Large-scale Cooperation Environment

Yushun LI
School of Educational Technology
Beijing Normal University
Beijing, China
lyshun@bnu.edu.cn

Haiguang FANG
Department of Education Technology
Capital Normal University
Beijing, China
hfang@mail.cnu.edu.cn

Ronghui HUANG
School of Educational Technology
Beijing Normal University
Beijing, China
huangrh@bnu.edu.cn

ABSTRACT—With the development of ICT technologies, such as ubiquitous computing, and semantic Web etc., Group-Oriented Large-scale Cooperation Environment used for special domains is needed. In e-Learning, the environment can be used as infrastructure for large-scale collaborative e-learning facility. In this infrastructure, awareness service is the key. The paper focuses on the research of innovative awareness service for this environment. According to the characteristics of groups in large scale environment and intrinsic requirements of cooperation, an awareness language is proposed, which is called Group-oriented Cooperative Awareness Language. The language is a declarative Language. The runtime for this awareness service is developing.

Keywords: GLCE (Group-Oriented Large-scale Cooperation Environment); GOCAL (Group-oriented Cooperative Awareness Language); Cooperative awareness; CSCL; CSCW

1 THE BACKGROUND OF RESEARCH

Cooperative awareness is a key concept in CSCL and CSCW, and there have many research efforts on awareness in CSCW. In 1992, Gwtin carried a systematic research on workspace awareness, which cover whole procedure of awareness expressing, managing and presenting [3]. In 2002, Schmidt, a key researcher in CSCW, carried out a comprehensive analysis on “awareness”, and pointed that awareness service should be the most important factor for any successful cooperative system [2]. Among existing research efforts, there are different research viewpoints, some research focus on synchronizing awareness [1], while other research efforts focus on asynchronous awareness [12]. Some research efforts pay attention to special awareness tool and user interface, while other efforts focus on awareness facilities [9]. With these research activities, more advanced topics have coming into being, for example, effect of cooperative awareness for cooperation has been the emphasis in these two years [10][11]. In e-Learning, the research on CSCL has also been a hot research area, for example, there are researches on providing augmented group awareness tools to promote cooperative efficiency of a group[8], some other researches aim to provide awareness of Learning Object Evolutions in e-Learning for more high-scale awareness [4]. Some research efforts pay attention to special domains is needed. In e-Learning, the environment can be used as infrastructure for large-scale collaborative e-learning facility. In this infrastructure, awareness service is the key. The paper focuses on the research of innovative awareness service for this environment. According to the characteristics of groups in large scale environment and intrinsic requirements of cooperation, an awareness language is proposed, which is called Group-oriented Cooperative Awareness Language. The language is a declarative Language. The runtime for this awareness service is developing.

Keywords: GLCE (Group-Oriented Large-scale Cooperation Environment); GOCAL (Group-oriented Cooperative Awareness Language); Cooperative awareness; CSCL; CSCW

II THE CORE CONCEPT IN GOCAL AND THE ROLE OF GOCAL FOR GLCE

A. Definition and understand the cooperation in intra-groups and inter-groups in GLCE

According to our research, in a large-scale cooperation domain, cooperation is happened inter-groups or intra-groups with varied cooperative intensity, and accordingly, four kinds of group are defined (refer to figure 1):

1. OPENING-GROUP: in this kind of group, cooperation is happened inside group, and accordingly, the cooperation relationship is hold tightly inside members of the group, all awareness information is consumed in intra-group.

2. CLOSED-GROUP: in this kind of group, cooperation is happened between members in the group, or between members in inter-groups. Therefore, there maybe are new groups created for promoting new requirements of cooperation, which support hierarchy of groups in large-scale cooperation domain. In this situation, awareness from the group can be sent into large-scale cooperation environment, and also, awareness from other groups in the large-scale cooperation domain can also be consumed by the group.

3. OUTWORK-LOOKING-GROUP: This kind of group is preferential in the large-scale cooperation domain, which is dominant in shaping cooperation in the large-scale cooperation domain. Awareness

4. INSIDEWORK-LOOKING-GROUP: in this kind of group, cooperation is happened between groups, but not in the same group, therefore, the cooperation relationship is loose tightly outside members of the group, awareness information is consumed in inter-group.
information from this group is sent into the cooperation domain, while awareness launched from other groups is neglected by the group.

INSIDEWORK-LOOKING-GROUP: this kind group is lawless in the large-scale cooperation domain, and usually is in bottom level of group structure, and in fact, they are groups for performing actual business. In this case, they don’t send awareness information into cooperation environment, and the other way round, they should consume awareness from the large-scale environment.

B. The origin of awareness information

Origin of awareness information is the foundation for constructing awareness service. With the aforementioned research, awareness information has wide provider, and unreasonable design will cause lots of unnecessary information, which block natural cooperation. Therefore, a programmable awareness model is needed, with which the designers and users of cooperation system can work more effectively, and at the same time, the model, the runtime of awareness service can filter, process and parse awareness information more conveniently. For these purpose, a three dimensional awareness mode is proposed, which is illuminated with activity theory [13], in which, cooperator, artifacts and cooperative tasks/process are classified as awareness source.

C. The role of GOCAL in Group-Oriented Large-scale Cooperation Environment

The paper focus on the research of awareness service in GLCE, and the service should consider the following requirements, first, every group in GLCE is not traditional online group, which usually is isolated and closed, and by contraries, it exists in more wide cooperation domain, which usually represent by Virtual Organization (VO). Second, the awareness service is the core for creating human-centered cooperation environment, and also is the key for executing cooperative process between human and computer, therefore, the service should provide functions for defining new type awareness information, for adjusting and refining awareness environment, for modifying mechanisms of parsing a special awareness to realize flexible more effective and lower-hindering cooperation. Third, because of abundant awareness information in GLCE, there need comprehended awareness process facility, which can be used for awareness event aggregating, filtrating and proceeding.

Based on aforementioned considerations, the role of the researched awareness service in GLCE is described in figure2. The GLCE is for integrated, group-oriented open cooperative environment, which can be used for large-scale cooperation. Based on the architecture in figure 2, the awareness service is realized based on Message Oriented Middleware (MOM), and it resides between MOM service (usually exist in distributed environment) and group-oriented cooperation workspace (therefore, in the paper, awareness information may be declared as awareness event).

III FORMATIVE DESCRIPTION FOR AWARENESS SERVICE IN GLCE

Define 3-1: AWARENESS INFORMATION. Information of a user/group filled in a cooperative environment, which usually is composed of behavior and status of other cooperators, and it include information in intra-groups and inter-groups. The total awareness information in GLCE (a cooperative domain) is a awareness space, called as ΣCA-GLCE.

Define 3-2: AWARENESS INTRA-GROUP. Awareness information launched inside a group and consumed by the group, which is used for creating human-centered cooperative workspace. The total awareness information in intra-group is earmarked as ΣCA-INTRA.

Define 3-3: AWARENESS INTER-GROUPS. This is awareness information launched in inter-groups, which is used by a group to understand cooperative environment with eyes on other groups. With this kind awareness, potential cooperation can be brought into existence, at the same time, this awareness information can be interpreted for inter-groups cooperation. The total awareness information in inter-groups is earmarked as ΣCA-INTER.

Define 3-4: AWARENESS FILTER. There are many kinds of awareness information in GLCE, and in order to creating high available and dynamic cooperation environment, there need awareness filters, named FILTERca, FILTERca is formed combining filter on group level (called FILTERca-group) and person level (FILTERca-personal), which is represented as FILTERca = FILTERca-group × FILTERca-personal. In a cooperation environment, the composing of FILTERca-group and FILTERca-personal is changed due to dynamic cooperation. According to three dimensional awareness model aforementioned, the filter of FILTERca-group and FILTERca-personal can be used on different level and different dimension.

Define 3-5: GROUP-LEVEL AWARENESS FILTER Group-level awareness filter, called FILTERca-group, is formed based on awareness space filtered according to group role in GLCE. In GOCAL, the function is implemented with awareness rules and awareness rule controls.

Define 3-6: PERSONAL-LEVEL AWARENESS FILTER Personal-level awareness filter, called FILTERca-personal, is formed based on awareness space filtered according to personal profile, preference and cooperative requirements in his workspace. In GOCAL, the function is realized with subscribe function.
Define 3-8: \textbf{CAccessoperator} represent awareness information related to cooperators, and it can be classified further into three aspects, \textbf{CAccessoperator} = \{ \textbf{CAccessoperator}-personal, \textbf{CAccessoperator}-group, \textbf{CAccessoperator}-intergroup \};

In the expression, \textbf{CAccessoperator}-personal represent awareness information related to a cooperator, \textbf{CAccessoperator}-group represent awareness information related to group that the cooperater belonged, and \textbf{CAccessoperator}-intergroup represent awareness information from other groups.

Define 3-9: \textbf{CAtask} represent awareness information related to cooperation task, and it can be classified further into three phases, \textbf{CAtask} = \{ \textbf{CAtask}-begin, \textbf{CAtask}-going, \textbf{CAtask}-end \}. Among them, \textbf{CAtask}-begin represent awareness information related to task creation, \textbf{CAtask}-going represent for awareness information related to task process, and \textbf{CAtask}-end represent awareness information related to task finish. Among them, awareness information of \textbf{CAtask}-begin and \textbf{CAtask} -end is very important for cooperation in inter-groups.

Define 3-10: \textbf{CArtifact} represent awareness information related to artifacts used in cooperation process, and it can be classified into two categories according to its sharing situation. \textbf{CArtifact} = \{ \textbf{CArtifact}-syn, \textbf{CArtifact}-asy \}. Among them, \textbf{CArtifact}-syn represents awareness information related to artifacts sharing in synchro-model, while \textbf{CArtifact}-asy represents awareness information related to artifacts sharing in asynchronous-model.

IV KEY SYNTAXES IN GOCAL

Based on the core concepts of GOCAL and formative definition of awareness service in GLCE, awareness language can be defined. And some key syntaxes in GOCAL are introduced as following:

A. Definition of awareness information type

AwarenessClassDeclation ::= AWTYPE \& GlobalQualifiedName \& SourceGroupNm \& Domain \& Dimension \& Body \& TimeExpression \& Domain = INTRAGROUP | INTERGROUP \& GroupNames = VO \& Dimension ::= \{ Cooperator | Taskprocedure | Artifacts \} = COOPTOR_GROUPINFO | COOPTOR_PERSONALINFO \& Taskprocedure ::= TASK_DECLAR | TASK_GOINGON \& TASK END \& Artifacts ::= ARTIFACT_SYN | ARTIFACTASYN \& GlobalQualifiedName \& SourceGroupNm \& Domain \& GroupNames \& Body ::= \{ \textbf{CArtifact}-syn, \textbf{CArtifact}-asy \}. Among them, \textbf{CArtifact}-syn represents awareness information related to artifacts sharing in synchro-model, while \textbf{CArtifact}-asy represents awareness information related to artifacts sharing in asynchronous-model.

B. The definition of awareness rule

RuleDeclation ::= RULE \& RuleName \& \{ RuleBody \} \& DO \& ActionBody \[ WITH \& ParametersXML \]

<RuleBody> ::= \{ AttributeRuleBody \| PatternNm \| NULL \}

For \textbf{RuleBody}, it can be defined according to awareness information attributes or composited event pattern. In the first situation, \textbf{AttrRuleBody} is composed logically of awareness information from cooperator, task and artifact with logical expression. In the second situation, \textbf{PatternNm} is composed logically of awareness event pattern, and special awareness event pattern is detected as condition for the rule. Awareness event type pattern in GOCAL will introduce in the following paragraph.

ParametersXML is a parameter structure, which is passed to \textbf{ActionBody} as parameters for latter cooperation execution when the rule is activated, and it usually include control information and cooperation data information, which are described in XML syntax. In the design of GOCAL runtime, parameters in the structure will be defined as \textbf{RULECONTEXT} for cooperative process. The structure of ParametersXML is as following:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<cooperationontology version="1.0" type="" parameter name="" parameter data="" filedata="" cooperationontology=""
```

In the structure, \textbf{type}" is represented awareness information type carried in \textbf{RULECONTEXT}; \textbf{parameter}" represents consistent parameter carried in \textbf{RULECONTEXT}, \textbf{data}" is represented the data file carried in \textbf{RULECONTEXT}.

Awareness rules in GOCAL are very important mechanisms for realizing flexible cooperation, and these rules are defined with group-oriented approach.

C. Definition of control rule for awareness rule

RuleControlDeclation ::= RULECTRL \& \{ RuleBody \} \& DO \& ControlBody

<RuleBody> ::= \{ \textbf{AttrRuleBody} \| PatternNm \| NULL \}

In GOCAL, awareness rules can be programmable, and therefore, all awareness rules can be programmed to set into active or inactive status, at the same, all these rules can be composed for variational cooperation environment.

In element of \textbf{RuleBody}, the operator of \textbf{AND}","\textbf{OR}" and \textbf{XOR}" will be provided in \textbf{GOCAL} runtime, and there will be three control mode, which are \textbf{AttrRuleBody}, \textbf{PatternNm} and \textbf{NULL}, and the syntax of \textbf{AttrRuleBody} and \textbf{PatternNm} is same as the syntax in awareness rule. At the same time, the value of the element may be NULL, in this situation, the control rule is just used as switch, with which a special rule can be set in active status or inactive status.
D. Definition of awareness event pattern

TypePatternDeclaration ::= PATTERNNM <ptnm> '{' 
<TypePatternBody> '}'


In cooperation system, awareness event pattern may be interested by cooperative pattern, for example, a cooperator may be care of a awareness event pattern which relate to multi online cooperators, with the event pattern, a cooperative mechanism may be created. Sometimes, awareness information should be compressed with awareness event pattern so that succinct awareness information can be displayed to cooperators, and in this way, usability of the cooperation system can be enhanced.

SEQUENCE used for defining event pattern based on sequence events, for example, awareness information of a cooperator “going online event” and “attending online audio/video event” can be defined as SEQUENCE event pattern, in that way, other cooperators can be invited to attending the online conference;

ALTEANATION used for defining replaceable awareness event;

ITERATION used for defining iterative awareness event pattern, with this syntax, the amount of awareness events can be reduced.

TIME used for defining event pattern with time limit, for example, for a special cooperative group, if a cooperative activity is happened in a special time limit, maybe some urgent awareness actions will be activated.

PARALL used for defining parallel awareness events, and it is suitable for awareness information executed in multi sites.

TW used for defining usable time limit for checking a special event pattern (in secs.). This mechanism is used for more efficiency processing because some awareness event pattern is time urgent.

E. Definition of group and role

GroupDeclaration ::= <creategroup> | <deletegroup> | <adduser> | <deleteuser> | <roleoperator>

<Kcreategroup> ::= [<GroupModifier>] GROUP <groupname>

Kdeletegroup ::= GROUP <groupname>

Kadduser ::= ADD <groupname> <usmn> AS <rulemn>

Kdeleteuser ::= ASK <assign> <takeback>

The syntax is used for defining group and rule in GOCAL runtime, in the syntax, group name is same as the name defined in hierarchy structure in a VO, and also, they are in same name space. In <GroupModifier>, the type of a group is remarked with value of OPEN, CLOSE, OUT_DIRECTION and IN_DIRECTION, which are aforementioned.

With the syntax, more roles can be defined into GLCE besides existing ones, and therefore, more elegant awareness display style can be created.

F. Definition of syntax for awareness subscription.

SubscribeDeclaration ::= <subscribe> | <unsubscribe>

<subscribe> ::= [<SourceGroupNm>]:[<TypePatternBody>]:[<ActionRegionExpression>]:[<DimensionExpression>]:[<DetailExpression>]

Awareness rules and awareness control rules are used for customizing awareness environment for a special group. On the other hand, for creating personalized cooperative environment, awareness facility should be provided for awareness tuning. This syntax is used for this purpose. In GOCAL, awareness subscription is ground on awareness policies on a group, that is, a cooperator can only subscribe/unsubscribe awareness information filtered by group awareness policies.

Above all, GOCAL is defined according to the requirements described in section 3, and it can be used for cooperative groups and cooperators with advanced awareness processing capability, and with the facility, awareness information can be processed based on awareness group, awareness type, awareness function area, awareness dimension, and awareness subscription etc. Correspondingly, functions of the awareness facility cover the origin of awareness, the process of awareness, the use and individuation of awareness. All these functions are important for the realization of GOCAL. The detail syntax of GOCAL is described in appendix.

V. COMPARISON WITH RELATED RESEARCH EFFORTS

The language of GOCAL combine fruits of event and rule technologies, and it is a declarative language, therefore, it is extended with Event-Condition-Conditions approach [5]. There have some research efforts in this direction. In literature [4][6], a system called NESSIE is created for group awareness environment, and in the system, awareness information is carried in event, and is displayed with multi display capabilities. In literature [Chyng Yang, Jung 2000], a system called TeamSCOPE is developed with event supported awareness, and in the system, awareness type include other cooperators activities, availability of other cooperators, awareness on task process, awareness on emotion of cooperator. In literature [16][17], a system called CASSIUS is developed, which aim to provide awareness infrastructure for cooperation environment, especially, it care interoperability capability of awareness service between different cooperation tools.

The awareness service created from GOCAL is different to the aforementioned research in the following area, first, GOCAL aims to create advanced awareness service based on event, and it can be used as an uniform facility for awareness process, which can be used for many features for cooperation in GLCE. Second, the service created by GOCAL is for group-oriented large-scale cooperation environment, and it can be used for both cooperation happened in intra-groups and inter-groups. Based on MOM, these awareness events can be routed, distributed and stored in GLCE for different groups, and this is different to aforementioned research efforts. Third, the interpretation and use of awareness information is different, in GOCAL, awareness information is used not only for displaying awareness information to cooperators, and also used for automatic cooperation control executing.

VI. FURTHER PLAN OF THE RESEARCH

In the paper, the requirement on awareness service for large-scale cooperation environment is researched, and accordingly, a awareness language is defined. This is the foundation for realizing such kind awareness service. Now, runtime for the language should be realized. The runtime will be composed with two parts. In the first part, UI functions for the awareness service, which is in charge of the interface between user/application and awareness runtime, and the interface include awareness visualize module, GOCAL language editing/submitting module, awareness type defining/querying interface, awareness rule/composite awareness event pattern interface, group/role defining interface, subscription interface etc. in the second part, core service of the runtime should be created, which include services and interfaces for integrating with other tools or services in workspace in GLCE. This part includes the following modules, the module of exception process for awareness information, the module of parsing/executing of GOCAL, the module of verifying of awareness information, the module for event pattern matching, the module for active awareness rules managing, the module for timer managing, and the module for awareness event receiving and sending etc.

Based on the runtime, a system supporting for large-scale cooperation will be developed, and will further validate the awareness service created from GOCAL.

ACKNOWLEDGMENTS

The study was made possible by a research grant (BCA070052) for Study of Ubiquitous Learning Resources Support System and Key Technologies from China National Office for Educational Sciences Planning and by the grant (NSFC: 60705023) from National Natural Science Foundation of China.

REFERENCES

Appendix GOCAL Syntax

The lists describe the syntax of GOCAL in BNF:

A.1 definition for awareness type

AwarenessClassDeclation::= AWTYPE <GlobalQualifiedName> <SourceGroupNm> <Domain> <Dimension> <Body> <TimeExpression>

<GlobalQualifiedName> ::= StringLiteral

<SourceGroupNm> ::= StringLiteral

<Domain> ::= INTRAGROUP | INTERGROUP <GroupNames> | VO

<GroupNames> ::= <GroupName> [';' <GroupNames>]

<GroupName> ::= <NameValue> <GroupNmType>

{NameValue} ::= StringLiteral

<GroupNmType> ::= STRING

<Dimension> ::= <Cooperator> | <Taskprocedure> | <Artifacts>

<Cooperator> ::= COOPTOR_GROUPINFO | COOPTOR_PERSONALINFO

<Taskprocedure> ::= TASK_DECLAR | TASK_GOINGON | TASK_END

<Artifacts> ::= ARTIFACT_SYN | ARTIFACTASYN

<Body> ::= '{' [';' <TypeAttributes>']' '}'

<TypeAttributes> ::= <TypeAttribute> [';' <TypeAttributes>]

<TypeAttribute> ::= <AttrName> ':' <type>

<AttrName> ::= StringLiteral


A.2 Definition for awareness rule (1)
RuleDeclaration::= RULE <RuleName> <RuleBody> DO <ActionBody> [WITH <ParametersXML>]
<RuleName> ::= StringLiteral
<RuleBody> = '{' <AttrRuleBody> | <PatternNm> '}'
<AttrRuleBody>::= <RuleSentence> [Logic_Operator <AttrRuleBody>]
<RuleSentence>::= <AwtypeAttr> <Relative_String_Operator> <ComparedValue>
<AwtypeAttr>::= StringLiteral
<Relative_String_Operator>::= <RelativeOperator> | <StringOperator>
<RelativeOperator>::= > | >= | < | <= | ==
<StringOperator>::= BEGINWITH | ENDWITH | WITH
<ComparedValue>::= StringLiteral
<Logic_Operator>::= AND | OR | XOR
>ActionBody>::= '{' AWARENESSNOTIFY '(' NOTIFYBody ')' | AUTOMATION '(' AUTOExecBody ')' | BLENDACTION '(' BlendActionBody ')' '
 NOTIFYBody)::= [ONLINEMESSAGE | EMAIL | MOBILE] '(' RoleLists ')' <MessageContent>
<RoleLists>::= (Role) | RoleLists
<Role>::= StringLiteral
<MessageContent>::= StringLiteral
<AutoExecBody>::= <ProgramLists>
<ProgramLists>::= ProgramList | ProgramLists
<ProgramList>::= StringLiteral
<BlendActionBody>::= NOTIFY '(' NotifyBody ')' RUN '(' ProgramLists ')' 
<ParametersXML>::= StringLiteral

A.3 Definition for awareness control rule (2)
RuleControlDeclaration::= RULECTRL <RuleBody> DO <ControlBody>
<RuleBody> = '{' <AttrRuleBody> | <PatternNm> | NULL '}'
<AttrRuleBody>::= <RuleSentence> [Logic_Operator <AttrRuleBody>]
<RuleSentence>::= <AwtypeAttr> <Relative_String_Operator> <ComparedValue>
<AwtypeAttr>::= StringLiteral
<Relative_String_Operator>::= <RelativeOperator> | <StringOperator>
<RelativeOperator>::= > | >= | < | <= | ==
<StringOperator>::= BEGINWITH | ENDWITH | WITH
<ComparedValue>::= StringLiteral
<Logic_Operator>::= AND | OR | XOR
<ControlBody>::= '{' ENABLE <rulelist> | DISENABLE <rulelist> '}'
<rulelist>::= (ruleName) | rulelist
<ruleName>::= StringLiteral
A.4 Definition for awareness pattern

TypePatternDeclaration ::= \texttt{PATTERNNM} <ptnm='{'}<TypePatternBody>{'}'}

\texttt{<TypePatternBody> ::= \texttt{SEQUENCE} <seqpattern>[TW <TimeWindowValue>] \texttt{ALTEANATION} <alterpattern>[TW <TimeWindowValue>] \texttt{ITERATION} <iterpattern>[TW <TimeWindowValue>]\texttt{TIME} <timepattern> \texttt{PARALL} <parpattern>[TW <TimeWindowValue>]

\texttt{<seqpattern> ::= <seqlists>[‘THEN’<seqlists>]

\texttt{<AwtypeName> ::= StringLiteral

\texttt{<alterpattern> ::= <alterlists>[‘ALT’<alterlists>]

\texttt{<iterpattern> ::= <AwtypeName>\{‘REP’<AwtypeName>\}}

\texttt{<iternum> ::= \texttt{<Integerstring>}

\texttt{<timepattern> ::= <AwtypeName1>;<AwtypeName2> WITHIN \texttt{<timespace>}

\texttt{<AwtypeName1> ::= StringLiteral

\texttt{<AwtypeName2> ::= StringLiteral

\texttt{<parpattern> ::= <partenlists>

\texttt{<partenlists> ::= <AwtypeName>[‘PAL’<partenlists>]

\texttt{<TimeWindowValue> ::= \texttt{<Integerstring>}

A.5 Definition for group and role

GroupDeclaration ::= \texttt{<creategroup> | <deletegroup> | <adduser> | <deleteuser> | <roleoperator>}

\texttt{<creategroup> ::= [<GroupModifier>] \texttt{GROUP} <groupname> <memberlists>

\texttt{<GroupModifier> ::= \texttt{OPEN | CLOSE | OUT_DIRECTION | IN_DIRECTION}

\texttt{<groupname> ::= StringLiteral

\texttt{<memberlists> ::= \{‘[’<members>‘]’\}

\texttt{<members> ::= \{‘[’<usrmn>‘]’\}

\texttt{<adduser> ::= \texttt{ADD} <groupnm> <usrmn> AS <rulenm>

\texttt{<rulenm> ::= StringLiteral

\texttt{<groupnm> ::= StringLiteral

\texttt{<usrmn> ::= StringLiteral

\texttt{<deleteuser> ::= \texttt{DELETE} <groupnm> <usrmn>

\texttt{<deletegroup> ::= \texttt{DELETE} <groupnm>

\texttt{<roleoperator> ::= \texttt{ROLE} <rulenm> ASSIGN_TO <groupnm>

\texttt{<takeback> ::= \texttt{TAKEBACK} <role> FROM <groupnm>

A.6 Subscription for awareness

SubscribeDeclaration ::= \texttt{<subscrible> | <unsubscrible>}

\texttt{<subscrible> ::= \texttt{SUBSCRIBE} <subnm><subbody>

\texttt{<subbody> ::= \{‘<subexpression>’\}'}
<subexpression>::= [<SourceGroupNm>] [<typenmexpression>] [<actionregionexpression>] [<dimensionexpression>] [<detailexpression>]<typenmexpression>::= <GlobalQualifiedName><typenm>::= StringLiteral
<actionregionexpression>::= INTERGROUP | INTRAGROUP | VO
<dimensionexpression>::= <Dimension>;
<detailexpression>::= [<typenmexpression>] [<actionregionexpression>] [<dimensionexpression>] [<detailexpressions>]<detailexpressions>::= <AwtypeAttr><Relative_String_Operator> <ComparedValue>[';'<detailexpressions>]

<unsubscribe> ::= UNSUBSCRIBE <subnm>