

# Aspekte der verteilten Bauwerksmodellierung in kooperativen Entwurfsumgebungen auf Basis dynamischer Objektstrukturen<sup>1</sup>

Thomas Hauschild, Reinhard Hübler  
Bauhaus–Universität Weimar  
Informations– und Wissensverarbeitung<sup>2</sup>

## 0 Einleitung

Die Untersuchung rechnernetzwerk-basierter Methoden zur besseren Unterstützung der kooperativen Aspekte beim Neuentwurf und der Revitalisierung von Bauwerken ist ein aktueller Forschungsgegenstand. Entwurfsprozesse im Bauwesen sind hochgradig arbeitsteilig und umfassen alternierende Phasen synchroner und asynchroner Kooperation. Die notwendige Kooperation der Teamarbeit sowie die Lösung von Entwurfskonflikten aufgrund unterschiedlicher Entwurfsintentionen erfolgt durch synchrone Kooperation. Arbeitet das Entwurfsteam geographisch verteilt, kommt der Unterstützung der gemeinsamen Tätigkeit durch eine Groupware–basierte Entwurfsumgebung eine entscheidende Rolle zu. Die Beachtung der asynchronen Phasen durch eine derartige Entwurfsumgebung ist ebenso wichtig, da hier im allgemeinen die feingranularen Entwurfstätigkeiten am gemeinsamen Arbeitsgegenstand ausgeführt werden. Die Beteiligten sollten über auftretende Entwurfskonflikte und andere wichtige Ereignisse im Team informiert werden und über geeignete synchrone und asynchrone Kommunikationskanäle verfügen. Daher liegt die Anwendung von CSCW–Techniken für temporal und geographisch verteilte Entwurfsprozesse im Fokus aktueller Forschungsprojekte.

## 1 Modellierung von Bauwerksinformationen

Als Basis für die Modellierung der Bauwerksinformationen sind laufzeitdynamische Objektstrukturen nutzbar, die in einem entsprechenden Modellverwaltungssystem gespeichert werden. Entsprechend der Rolle im Entwurfsprozess wird den Bearbeitern eine dömanenspezifische Taxonomie bereitgestellt, welche formalisiertes Wissen über ihren spezifischen Entwurfsbereich enthält. Diese kann während der Arbeit am Bauwerksentwurf erweitert werden, d.h. neue Konzepte können durch Spezialisierung vorhandener Informationen eingebracht und in begrenztem Rahmen Modifikationen an bestehenden Klassen vorgenommen werden. Verschiedene Typen schwer formalisierbarer Informationen können als spezielle Attribute mit Klassen und Exemplaren verknüpft werden, um möglichst sämtliche im Entwurfsprozess anfallenden Informationen für den Bearbeiter direkt zugreifbar zu halten.

Als Basis praktischer Umsetzungen sind hybride CSCW–Systemarchitekturen mit verteilbaren Modellierkernen, die sowohl Taxonomien als auch Projekte verwalten können,

---

<sup>1</sup> Diese Arbeit wurde teilweise von der Deutschen Forschungsgemeinschaft (DFG) im Rahmen des Teilprojektes „Digitales Bauwerksmodell als Grundlage der Datenintegrations-ebene für Bestandsinformationen“ des SFB 524 „Werkstoffe und Konstruktionen für die Revitalisierung von Bauwerken“ gefördert.

<sup>2</sup> Bauhaus–Universität Weimar,  
Informations– und Wissensverarbeitung,  
Coudraystraße 13, 99421 Weimar  
Telefon: 03643 / 584260, Fax: 03643 / 584292  
E-Mail: {Thomas.Hauschild|Reinhard.Huebler}@Informatik.Uni–Weimar.de

das heißt sowohl Klassen als auch deren Instanzen, geeignet. Die Kommunikation zwischen den einzelnen für die jeweiligen Partialmodelle verantwortlichen Modellierkernen erfolgt aus Taxonomiesicht auf Metaniveau durch statische CORBA–Methodenaufrufe. Dasselbe gilt auch für die Kommunikation mit dem zentralen Server für projektglobale Informationen. Die Bauwerksmodelle der einzelnen Entwerfenden können als Partialmodelle eines logischen globalen Bauwerksmodells angesehen werden. Das logische Globalmodell ist kohärent, da die Semantik der Partialmodelle nicht disjunkt ist. Dies wird durch den gemeinsamen Kontext der Gruppenarbeit am Entwurfsgegenstand ‚Bauwerk‘ gesichert.

## **2 Unterstützung von kooperativer Arbeit im Bauwerksentwurf**

Konkrete Umsetzungen von Groupware–Systemen für Workgroup–Computing basieren naturgemäß auf Verteilten Systemen. Der Informationsaustausch zwischen den Groupware–Applikationen auf den Arbeitsplatzrechnern der Beteiligten erfolgt unter Nutzung von verschiedenen Schichten der Infrastruktur von Rechnernetzwerken.

Die Nutzung von Rechnernetzwerken, das Auftreten von parallel ablaufenden Prozessen im System sowie die Kooperation einer Anzahl von Beteiligten impliziert die Notwendigkeit der Realisierung bestimmter Mechanismen, um eine korrekte Funktionsweise des Systems zu gewährleisten. Zu diesen gehören unter anderem Techniken zur Überwachung des potentiell parallelen Zugriffs auf gemeinsame Ressourcen und Mechanismen zur Authentisierung der beteiligten Personen sowie zur Autorisierung von deren Zugriffen auf Teile des gemeinsamen Informationsbestandes.

Für viele dieser Aufgaben existieren Lösungen, die für Standardprobleme von Datenbanken, Verteilten Systemen, Mehrbenutzersystemen oder Parallelen Prozessen anwendbar sind. Diese Techniken können jedoch nicht uneingeschränkt auf Groupware–Systeme übertragen werden, da diese teilweise die Spezifik von Groupware–Lösungen nicht hinreichend beachten. Eine Ursache dafür ist, dass Techniken, die für die Interaktion von Systemen geeignet sind, nicht grundsätzlich auf Probleme der Mensch–Maschine–Interaktion übertragbar sind. Die Konsequenz einer Anwendung herkömmlicher Techniken könnte eine unbefriedigende Responsivität der Nutzerschnittstelle oder ein für die Anwender und den Arbeitsprozess inakzeptables Systemverhalten sein.

Weiterhin ist es wichtig, dass die verwendeten Systeme und Techniken in einer CSCW–Umgebung „cooperation aware“ arbeiten. Darunter werden Systeme verstanden, die durch verschiedene Mittel der Schnittstellengestaltung dem Anwender die Präsenz und Aktivität anderer Teammitglieder verdeutlicht. Die Awareness–Eigenschaften der Groupware–Umgebung sind essentiell für den Gruppenprozess und haben damit Auswirkungen auf die Resultate der Teamarbeit, da letztendlich den Beteiligten eine bewusstere Entscheidung über die eigenen, nächsten Aktivitäten im Sinne der Erreichung des Teamzieles ermöglicht wird.

Klassische Mehrbenutzer–Systeme weisen dagegen im allgemeinen die entgegengesetzte Eigenschaft „Kooperations–Transparenz“ auf. Diese besagt, das durch das System versucht wird, beim Anwender die Illusion der alleinigen Nutzung des Systems zu erzeugen. Eine Teamarbeit bei solchen Systemen ist meist nur durch implizite Kooperation über das System in Verbindung mit expliziter Kommunikation über externe Kanäle möglich.

## **3 Zugriffskontrolle für Modellinformationen**

Bei einer kooperativen Arbeit in Netzwerken ist eine Überwachung des berechtigten Zugriffs auf Informationsbestände grundsätzlich erforderlich. Da im Ergebnis von Bauplanungsprozessen Dokumente mit rechtsverbindlichem Charakter entstehen, kommt dieser Anforderung eine besondere Bedeutung zu.

Damit sind einerseits Authentisierungsmechanismen zu integrieren, um den Zugang zu den Informationsbeständen und Ressourcen auf die Menge der am Planungsprozess beteiligten Personen und Systeme zu beschränken. Im Ergebnis des Authentisierungsverfahren ist der Entwurfsumgebung bekannt, durch wen der Zugriff erfolgt. Damit es ebenso möglich, die Rollen eines auf gemeinsame Ressourcen zugreifenden Beteiligten zu ermitteln. Dabei ist zu beachten, dass einzelne Bearbeiter mehrere Rollen, aber auch umgekehrt mehrere Personen dieselbe Rolle haben können.

Andererseits kommt der Realisierung geeigneter Autorisierungsmechanismen eine große Bedeutung zu, um bestimmte Rechte des Zugriffs auf gemeinsame Informationen oder auf Informationen aus Verantwortungsbereichen anderer Beteiligter personenkonkret oder nach Rollenzugehörigkeit vergeben zu können.

Für die Realisierung von Autorisierungsmechanismen sind die meisten traditionellen Verfahren nur bedingt geeignet. Verfahren mit der Vergabe von Schreib-, Lese- und Ausführungsrechten für Individuen und Gruppen besitzen weder die notwendige Ausdruckskraft noch die notwendige Flexibilität. Verfahren, die auf Access Control Lists (ACL's) basieren, erfüllen ohne Anpassungen ebenso nicht die Anforderungen für die Realisierung von Groupware-Systemen. So ist es beispielsweise schwierig, Modifikationen eines gemeinsamen Datenbestands zu verweigern, aber Hinzufüge-Operationen zu gestatten. Rechte zur Verwaltung gemeinsamer WYSIWIS-Präsentationen oder zur Erstellung einer Bezugnahme auf Informationsbestände eines anderen Beteiligten sind ebenso schwer abbildbar.

Wichtige Anforderungen an eine effektive Zugriffskontrolle in CSCW-Applikationen sind die Möglichkeit einer dynamischen Zuordnung von Rechten infolge der hohen Änderungsfrequenz sowie eine feingranulare Vergabe von Rechten. Wünschenswert ist ebenso eine Ableitung der aktuellen Rechte aus dem momentan ausgeführten Arbeitsschritt oder der gegenwärtig ausgeübten Rolle. Günstig ist ferner ein einfaches Management der Rechte durch definierte Teammitglieder anstelle externer Systemadministratoren.

Aufgrund der genannten Anforderungen ist es oft erforderlich, dass Groupware-Systeme eine eigene Zugriffskontrolle realisieren. Je nach den konkreten Anforderungen der umzusetzenden Applikation können hierfür adaptierte ACL-Verfahren oder auf Zugriffsmatrizen basierende Mechanismen eingesetzt werden.

Das Matrixverfahren nach Lampson sowie Graham und Denning basiert im wesentlichen auf den Zugriffsrechtszustand beschreibenden Tripeln aus Subjektmenge, Objektmenge und einer Zugriffsrechtmatrix und Operationen zur Anpassung dieser Tripel. Dieses Modell wurde von Shen und Dewan um Kollaborationsrechte, negative Rechte und um eine Übertragung von Rechten der zugehörigen Subjekt- bzw. Objektgruppe für nichtspezifizierte Matrixelemente erweitert.

Eine noch allgemeinere und umfassendere Erweiterung des Matrixverfahrens schlagen Smith und Rodden vor. Deren Zugriffsmodell beinhaltet Nutzer, Rollen, Objekte und Rechte. Die Menge der Rechte ist nicht vordefiniert, sondern kann an die Bedürfnisse der Applikation bzw. des zu überwachenden Objekts angepasst werden. Für jede Zuordnung von Objekten zu Nutzern bzw. Rollen existiert eine zweispaltige erweiterbare Zugriffsrechte-Matrix. Die erste Spalte repräsentiert die spezifizierten Rechte des Zugriffs des Nutzers. Mit der zweiten Spalte wird verwaltet, ob die Ergebnisse des Zugriffs propagiert werden dürfen. Für die Elemente der Matrix schlagen Smith und Rodden bitweise Einträge vor, welche die aktuellen Zugriffsrechte sowie Defaultbelegungen bzw. generelle Verbote oder Erlaubnisse ausdrücken. Das Verfahren von Smith und Rodden weist die notwendige Flexibilität und Ausdruckskraft auf, um für in Modellverwaltungssystemen gespeicherte Bauwerksinformationen eine Zugriffskontrolle zu realisieren.

Einerseits lassen sich die nötigen Basisrechte wie Lesen, Modifizieren, Anfügen, Löschen ausdrücken, dasselbe gilt ebenso für durch die Groupware-Umgebung erforderlichen Rechte zur Verwaltung synchroner und asynchroner Kooperation.

Ein skaliertes Einsatz des Zugriffskontrollmechanismus ist ebenso möglich. Dieser kann sich zwischen den Extremen der globalen Verwaltung der Informationen eines Domänenmodells und der Verwaltung einzelner Objekte der Bauwerksmodellebene bewegen. Der erste Fall ist für Anwendungen als zu grob und unflexibel einzuschätzen, der zweite Extremfall würde einen enormen Verwaltungsaufwand mit hohem Ressourcenbedarf und negativen Auswirkungen auf das Laufzeitverhalten nach sich ziehen.

Die hier vorgeschlagene Lösung basiert auf einer Default-Zugriffskontrollmatrix pro Domänenmodell. Diese Defaultwerte sind für Modellklassen oder deren Objekte überschreibbar. Werden Referenzen auf diese überschriebenen Zugriffsmatrizen entlang von Vererbungs- und Aggregationsbeziehungen und auf die Instanzen dieser Klassen propagiert, ergibt sich eine Kontrolle der Berechtigungen in einer auf die Modellsemantik anpassbaren Form. Um eine flexible, noch feingranularere Verwaltung der Berechtigungen zu erzielen, können den Attributen der Klassen bei deren Definition jeweils ein Grad auf einer abgestuften Permissivitätsskala zugewiesen werden. Dadurch wird es in Verbindung mit dem adaptierten Matrixverfahren möglich, den Zugriff anderer Beteiligter auf kritische Attribute restriktiv zu gestalten, während die Interaktionsmöglichkeiten mit anderen Attributen offener gehalten werden können. Ebenso ist die Flexibilität für eine häufige Änderung der Rechte vorhanden. Neben einer effektiven Zugriffskontrolle ist es wünschenswert, eine flexible Meta-Zugriffskontrolle zu integrieren, welche die Rechte der Anwender bei der Zuweisung und Weitergabe von Rechten überwacht. Durch die ‚Meta Access Control‘-Komponente werden daher die Berechtigungen zum Einräumen und Entziehen von Rechten überwacht und eine komfortable Oberfläche zur Verwaltung der Zugriffskontrolle bereitgestellt. Eine technische Umsetzung ist wiederum durch adaptierte ACL's oder Zugriffsmatrizen möglich.

#### **4 Mechanismen zur Kontrolle von Nebenläufigkeiten**

Synchrone kooperative Arbeit erfordert effektive Concurrency-Control Mechanismen, die einerseits die Integrität des gemeinsamen Informationsbestandes sichert, andererseits die Tätigkeiten am Bauwerksmodell nicht unnötig beeinträchtigt. Traditionelle Techniken sind meist für die Nebenläufigkeitskontrolle für parallele Prozesse in Rechnersystemen konzipiert und beabsichtigen häufig, nur einen aktiven Prozess mit Modifikationsabsichten am gemeinsamen Datenbestand zuzulassen. An Techniken, die in CSCW-Umgebungen zum Einsatz kommen, sind spezielle Anforderungen zu stellen, die nicht grundsätzlich durch sämtliche traditionelle Verfahren erfüllt werden. Die hier zum Einsatz kommenden Verfahren sollen unter anderem eine echte synchrone Arbeit zulassen, eine gute Responsivität des Systems garantieren, eine hohe Robustheit besitzen, auch für Wide Area Networks geeignet sein und auf die Bedürfnisse der ablaufenden Teamprozesse ausgerichtet sein. Beispielsweise wäre eine zeitweise Blockierung sämtlicher Aktivitäten von Benutzern nicht wünschenswert. Längere Wartezeiten auf Systemreaktionen beispielsweise infolge komplizierter Mechanismen zur Anforderung von Sperren tragen nicht zu einer guten Systemakzeptanz beim Anwender bei. Das Zurücksetzen der Resultate einer größeren Anzahl von Arbeitsschritten infolge einer abgebrochenen langen Transaktion ist ebenfalls inakzeptabel. Grundsätzlich kann zwischen optimistischen und pessimistischen Verfahren zur Nebenläufigkeitskontrolle unterschieden werden. Optimistische Verfahren lassen zunächst alle Zugriffe auf gemeinsame Ressourcen zu. Tritt dabei ein Konflikt auf, muss dieser erkannt und anschließend die entstandenen Auswirkungen behoben werden. Optimistische Verfahren sind besonders für CSCW-Anwendungen geeignet, in denen Konflikte relativ selten auftreten und sich mit herkömmlichen Verfahren kein befriedigendes Interaktionsverhalten erzielen lässt. Nachteilig sind bei diesen Verfahren allerdings der höhere Realisierungsaufwand und der geringere garantierbare Grad der Datenkonsistenz. In CSCW-Systemen für „Workgroup Computing“ kommen häufig optimistische Transaktionsverfahren zum Einsatz, da

pessimistische Verfahren nicht für lange Transaktionen wie beispielsweise kooperative Dokumentenbearbeitung geeignet sind. Diese Verfahren prüfen erst vor Abschluss der Transaktion, ob Anzeichen für Konflikte vorliegen. Dies wird den betroffenen Teammitgliedern mitgeteilt, die dann über Abbruch der Transaktion oder manuelle Behebung der Auswirkungen entscheiden können.

Für die Unterstützung verschiedener Aspekte der asynchronen Kooperation im Bauwerksentwurf eignen sich Lock-Verfahren. Diese bewirken eine Sperrung von Ressourcen, die sich bereits in Benutzung befinden. Je nach konkreter Implementierung werden sämtliche Zugriffe auf die betroffene Informationseinheit oder Ressource verweigert oder nur nichtmodifizierende Zugriffe zugelassen. Lock-Verfahren besitzen die Eigenschaft der „Group-Awareness“, da die Verursacher von Sperren durch andere Teammitglieder ermittelbar sind und gesperrte Informationen je nach genutztem Präsentationstyp vergleichsweise einfach an der Nutzerschnittstelle kenntlich gemacht werden können. Ein Problem der Anwendung von Lock-Verfahren sind irrtümlich aufrecht erhaltene Sperren, die andere Teammitglieder über längere Zeiträume behindern können. Es existieren verschiedene Ansätze zu dessen Lösung, so ist eine Aufhebung nach einer verstrichenen Zeit der Inaktivität des Lock-Inhabers möglich. Ebenso können aber auch Sperren grundsätzlich nur für festgelegte Zeiträume vergeben werden, nach denen sich der ehemalige Inhaber neu um eine Lock-Zuweisung bemühen muß. Lock-Verfahren sind für verschiedene Grade der Granularität anwendbar, als Extrema können wiederum gesamte Domänenmodelle für eine äußerst grobe Sperrung bzw. einzelne Objekte oder Attribute in diesen für eine außerordentlich feingranulare Sperrung angesehen werden können. Für die Nebenläufigkeitskontrolle für in Modellverwaltungssystemen gespeicherte Bauwerksinformationen ist eine Propagation von Locks entlang von Aggregationsbeziehungen ein geeignetes Mittel, um bestimmte Modellbereiche zu sperren. Beinhaltet das Domänenmodell beispielsweise die konstruktive Bauwerksgliederung, lassen sich Gebäudeteile, Etagen aber auch kleinere Bauteile sperren.

Für Entwurfskonferenzen, also Phasen der verteilten synchronen kooperativen Arbeit am gemeinsamen Entwurfsgegenstand sind ebenfalls effektive Concurrency-Control-Techniken erforderlich. In diesen Phasen müssen derartige Mechanismen sowohl für gemeinsame Informationsbestände als auch für andere gemeinsam genutzte Ressourcen realisiert werden. Insofern die vorliegende Relaxierung der WYSIWIS-Präsentation der aktuellen gemeinsamen Entwurfsgegenstände das Kongruenz-Constraint nicht oder nicht signifikant verletzt, müssen auch Operationen, die diese betreffen, serialisiert werden.

Für derartige Aufgaben können Floor-Passing-Verfahren eingesetzt werden. Diese gehören zu den sogenannten ‚Turn-Taking-Protocols‘ und sorgen für eine temporal wechselnde Zuordnung der Rechte, indem das Vorbild von face-to-face-Konferenzen auf verteilte Konferenzen übertragen wird. Wie bei diesen hat nur ein Teilnehmer die Möglichkeit der aktiven Nutzung der Ressourcen, wobei ihm diese nach einer gewissen Zeit oder durch einen Chairman, der in diesem Kontext ‚Facilitator‘ genannt wird, entzogen werden kann. Die Berechtigung zur Benutzung der geschützten Ressourcen wird durch den Besitz eines ‚floor-tokens‘ ausgedrückt, der bei expliziten Verfahren durch die Teilnehmer und bei impliziten Verfahren durch das System weitergereicht wird. Floor-Passing-Verfahren sind relativ leicht implementierbar, wirken aber durch die Beschränkung synchroner Tätigkeit im Team einschränkend.

Diese Einschränkungen können unter Umständen durch den Einsatz von Transformationsverfahren überwunden werden. Diese Verfahren sind anwendbar, wenn die Daten des aktuellen Arbeitsgegenstandes repliziert werden können und eine eng gekoppelte, synchrone Teamarbeit wie bei kooperativer Dokumentenbearbeitung im weitesten Sinne vorliegt. Der Grundgedanke dieser Verfahren ist, das alle Operationen zuerst lokal ausgeführt werden und dann zusammen mit einem Statusvektor an alle anderen Teilnehmer gesandt wird. Die Empfänger können durch einen Vergleich mit einem lokalen Vektor den eigenen

Abarbeitungszustand ermitteln. Sind die Statusvektoren äquivalent, wird die Operation sofort ausgeführt, handelt es sich um eine ‚zukünftige‘ Operation, wird sie vorerst in eine Warteschlange zurückgestellt. Wurde eine ‚zurückliegende‘ Operation empfangen, wird diese transformiert, um die Auswirkungen der veränderten Reihenfolge auszugleichen. Dies geschieht unter Nutzung einer quadratischen Transformationsmatrix, deren Elemente die anzuwendenden Transformationsfunktionen darstellen, nach deren Anwendung auf die empfangene Operation und die anschließende Ausführung der transformierten Operation dasselbe Ergebnis wie bei der Ausführung der Operationen in ihrer korrekten Reihenfolge entsteht. Der Nachteil dieser Verfahren liegt in der Tatsache, dass die Aufstellung dieser Transformationsmatrix nicht trivial ist, besonders wenn eine hohe Anzahl von Operationen vorliegt. Inwiefern dieses Problem mit vertretbarem Aufwand für CSCW–Applikationen im Bauwerksentwurf gelöst werden kann, sollte untersucht werden.

## 5 Ausblick

Die Notwendigkeit der Beachtung besonderer Aspekte bei der Realisierung kooperativer Entwurfsumgebungen erstreckt sich nicht ausschließlich auf die besonders wichtigen Bereiche Zugriffs– und Nebenläufigkeitskontrolle. So müssen spezifische Undo– und Redo–Mechanismen implementiert werden, da in kooperativen Umgebungen überlappende Arbeitsbereiche der Teammitglieder auftreten können. Die Anwendung traditioneller Verfahren würde hier zu Nebenwirkungen auf die Resultate anderer Teilnehmer führen können. Ebenso ist es beispielsweise notwendig, CSCW–fähige Mechanismen zum Zusammenführen unterschiedlicher Versionen von Modellinformationen zu realisieren, da im Ergebnis asynchroner Kooperation oder optimistischer Concurrency–Control–Verfahren Differenzen zwischen lokalen Informationsbeständen auftreten können.

## Literatur

1. Dewan, P., Shen, H.: Controlling Access in Multiuser Interfaces, ACM Transactions on Computer Human Interaction 1998, Vol. 5:1, pp34-62, ACM Press, 1998
2. Dewan, P., Shen, H.: Flexible Meta Access-Control for Collaborative Applications, Proceedings of the ACM 1998 Conference on Computer–Supported Cooperative Work CSCW’98, Seattle, USA, 1998
3. Dourish, P.; Belotti, V.: Awareness and Coordination in Shared Workspaces, Proceedings of the ACM Conference on Computer–Supported Cooperative Work CSCW’92, Toronto, Canada, 1992
4. Greenberg, S., Marwood, D.: Real Time Groupware as a Distributed System: Concurrency Control and its Effect on the Interface, Proceedings of the ACM Conference on Computer–Supported Cooperative Work CSCW’94, Chapel Hill, USA, 1994
5. Hauschild, T., Hübler, R., Schleinitz, M.: Unterstützung von Gruppenarbeit im Bauwerksentwurf auf Basis eines dynamischen objektorientierten Bauwerks–Modelliersystems, GI–Fachtagung CAD 2000: Proceedings, Gesellschaft für Informatik e.V., 2000
6. Rodden, T., Blair, G.: CSCW and Distributed Systems: The Problem of Control, Proceedings of the 2<sup>nd</sup> European Conference on Computer–Supported Cooperative Work ECSCW 91, Amsterdam 1991, Kluwer Academic Publishers, 1991
7. Smith, G., Rodden, T.: Access as a Means of Configuring Cooperative Interfaces, Proceedings of the ACM Conference on Organizational Computing Systems COCS 93, Milpitas, CA, USA, 1993