La coordination des équipes distribuées autour d'un espace de production partagé

Jacques Lonchamp
LORIA – équipe ECOO
jloncham@loria.fr

La coordination des équipes distribuées autour d'un espace de production partagé

1. Quelques points importants suggérées par les approches théoriques (TA, TC, CD).

2. Approfondissement de ces points à partir d'un cas concret (la production de logiciel libre sur Internet).

3. Vers une grille d'analyse générale. Des perspectives.

1.1. Théorie de l'activité

3 niveaux d'analyse des activités humaines

Opérations routinières	Déterminées par conditions objectives
Actions conscientes (individuelles ou collectives)	Guidées par buts
Activité (cadre général)	Créée et stimulée par un objet (motivation), évolutive, s'inscrit dans une histoire

TA et coordination :

opérations	Coordination 'mécanique'	Ex: workflow
actions	Planification Coordination mutuelle acteurs : partage des buts & connaissances ('cognitif')	Ex: réunions coordination
activité	Émergence, évolution de la coordination ('méta')	Ex: négociation des règles, formation de la communauté …

Point 1 Dans une situation de coopération distribuée inter-organisations, *le niveau 'méta' doit jouer un rôle essentiel* pour définir et adapter dynamiquement les règles de coordination.

'Espace méta' pour discuter les règles et documenter les *protocoles sociaux*

+ moyens d'adaptation dynamique des *protocoles systèmes* des outils (du paramétrage aux systèmes réflexifs).

1.2. Théorie de la coordination

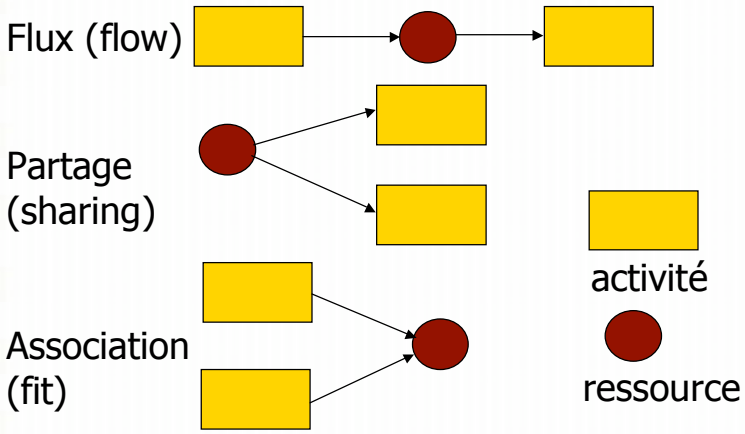
T ravail de groupe : {acteurs} qui effectuent des activités interdépendantes pour atteindre des buts donnés. Ces activités utilisent et créent divers types de ressources.

Coordination = gestion des dépendances entre activités.

Analyse 'macroscopique'.

Approche de la TC :

1. Catégoriser les types de dépendances



2. Rechercher {mécanismes coordination} pour chaque type de dépendance dans diverses disciplines (économie, éthologie, informatique, organisation, psychologie …)

Nombreuses alternatives.

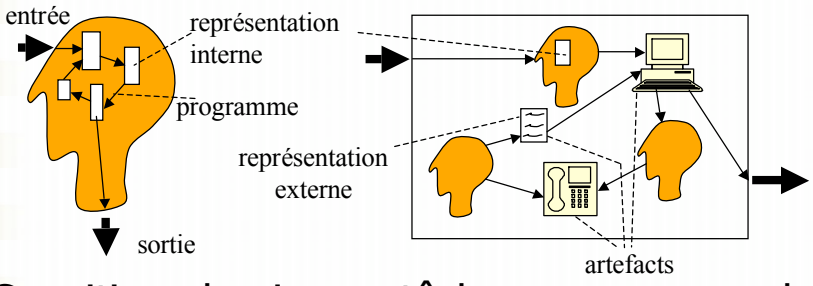
Ex : affectation/partage de ressource

- marché, enchères (économie),
- décisions managériales, budgétaires (gestion),
- allocation de ressources (informatique) FIFO, LIFO, priorités, etc.

3. Générer de nouvelles formes organisationnelles en tentant de changer les mécanismes de coordination (réingénierie des processus). Analyse côut/vulnérabilité.

Point 2 Les mécanismes de coordination sont adaptés à un contexte et *souvent non transposables* d'un contexte à un autre (ex: localisé → distribué).
Dans notre contexte, les mécanismes visent à satisfaire les propriétés liées au caractère distribué de la production. Noyau commun?

1.3. Cognition distribuée



Cognition classique : tâche = processus de traitement de l'information au niveau du cerveau d'un individu.
Cognition distribuée : tâche = processus de traitement distribué dans un cadre technologique et social.

Analyse 'microscopique'. Qq concepts en CD:

- horizon d'observation (limité),
- trajectoires de l'information (imprévisibles)
- construction d'une connaissance partagée (coordination des buts et représentations)
- compréhension intersubjective (communauté de pratique), …

Point 3 A ce niveau d'analyse, le caractère souvent *opportuniste* et *non explicite* de la coordination dans un contexte distribué apparaît clairement. Nécessite des solutions adaptées.

2. Production de logiciel libre sur Internet

Communautés de développeurs [2-milliers] autour de sites portails Ex: Linux, Apache, Mozilla, NetBeans, SourceForge, ...

Constatations immédiates.

Pas de technologies évoluées : pas d'outils de workflow, d'awareness, hypermédia, représentation des connaissances. Que des outils classiques : mailing lists, (Majordomo), gestion configurations (CVS), gestion bugs/problèmes (Bugzilla).

Pas d'artefacts complexes : spécifs formelles ou semi-formelles, modèles d'analyse/conception, documents de planification, bases de connaissances, … Seulement codes, archives (messages, logs, bugs/problèmes), roadmap, définitions de rôles, fichiers d'état de la release (STATUS file), statistiques sur les bugs.

Peu de règles : affectation des rôles, gestion des releases (contenu, calendrier, choix manager, procédure), vote.

Point 1 (niveau 'méta')

- Listes de diffusion, spécialisées/rôles, archivées, indexées (Hypermail) = *notification* + source *d'awareness à la demande* + *mémoire de projet partagée semi-structurée.*

NetBeans: 26+, Apache: 17, FreeBSD: 68

- Beaucoup *discussions au niveau méta* (règles) dans la période de mise en place (NetBeans). Plus rares après (Apache).
- Adaptation dynamique : *protocoles sociaux dominants.* Protocoles systèmes liés aux droits d'accès dérivant des rôles.

Point 2 (adaptation au contexte)

• Constatation : différences énormes entre développement classique et open source.
- pas d'analyse préalable des besoins,
- pas de documentation de l'analyse et de la conception,
- pas d'allocation des tâches,
- pas de gestion de projet (planification),
- pas de gestion des équipes (méritocratie élective), …

• Analyse coordination des activités : mélange *très grande liberté* (modifier le code, créer une nouvelle branche expérimentale dans les versions, créer un nouveau sous-projet, etc.) + *contrôle très fort* (définition et production releases tous les 3 à 12 mois). 'Benevolent dictator'.

Adapté au contexte de la production fortement distribuée : peu de contrôle *a priori* (disponibilités, compétences, …) mais fort contrôle *a posteriori* (revues/pairs, déboguage // massif).

Point 3 (caractère opportuniste)

- Synchronisation cognitive des acteurs *grâce au caractère public des discussions et du code* (horizon d'observation large) + appartenance à une *communauté de pratique* (hackers ⇒ compréhension inter-subjective).
- Se concrétise par le suivi des *listes de diffusion spécialisées* et donc de ce que font les autres ('overhearing'). La diffusion à tous ('broadcasting') permet de toucher tous les experts censés suivre les listes.

- *Le fondement effectif de la coordination = culture du partage de l'information ('making it public' culture) et d'écoute de la vie de la communauté ('overhearing').*

  Correspond à un effort demandé à chaque participant. Pas toujours fourni ⇒ besoin assistance.

  Ex: commit logs Apache = 9,7 mots de description en moyenne !

3. Vers une grille d'analyse générale

L'espace de production partagé *ne suffit pas* pour assurer une coordination efficace (activités, buts, représentations individuelles). Il faut :

1. Une communauté et un système *structurés autour de rôles* clairement définis.

2. *Des canaux de communication ouverts* et *une mémoire de projet partagée*, structurés /rôles, faciles à alimenter et à exploiter.

Décisif pour :

- l'auto-organisation de la coordination (discussion des règles + documentation des protocoles sociaux),
- la synchronisation cognitive des acteurs (buts et représentations),

3. Des moyens *d'adaptation dynamique* pour les protocoles systèmes.

4. Une vraie *culture* du partage de l'information et de participation à la vie de la communauté ('make it public' et 'overhearing'). Une assistance adaptée.

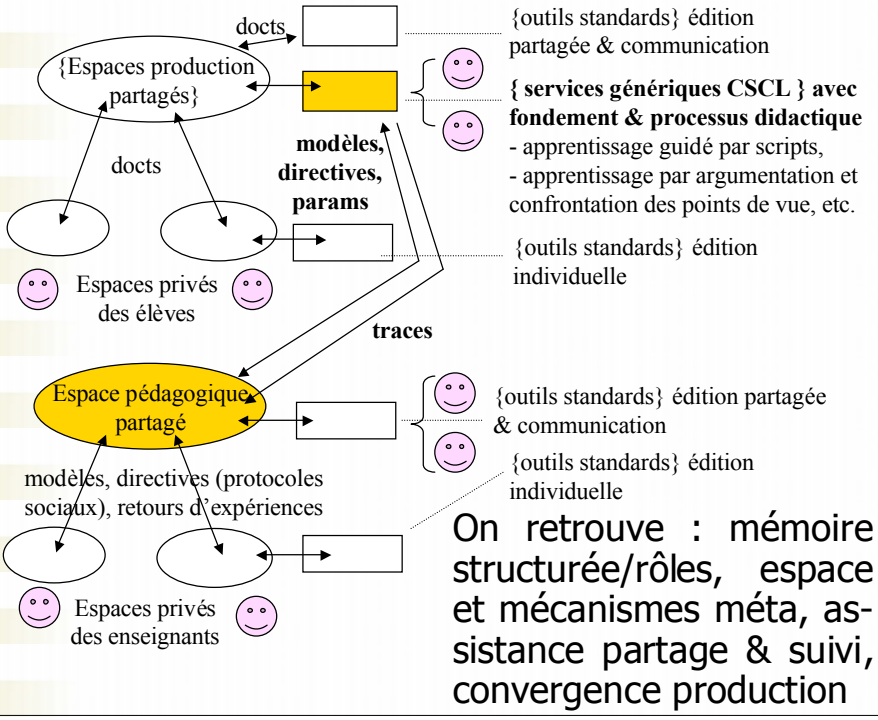
5. Un mélange de liberté, pour permettre aux *initiatives individuelles* de s'exprimer, et de règles de contrôle très strictes (protocoles sociaux et/ou protocoles systèmes) pour assurer une *convergence périodique de la production*.

Perspectives

• Confronter cette grille analyse à d'autres situations et systèmes de coordination d' équipes distribuées autour d'espaces de production (architecture & construction ?). Réalité de ce noyau commun ?

• Mieux comprendre, pour mieux assister tous les aspects de la coordination (méta, contextuels, cognitifs).

Exemple de réflexion en cours : support à des communautés pédagogiques autour de l'apprentissage coopératif sur Internet (réalisation et partage d'expériences).



On retrouve : mémoire structurée/rôles, espace et mécanismes méta, assistance partage & suivi, convergence production

Merci de votre attention !