



# TECHNIQUES DE COOPÉRATION (ISI 3)

**Manuele Kirsch Pinheiro**  
Maître de conférences en Informatique  
Centre de Recherche en Informatique  
Université Paris 1 – Panthéon Sorbonne  
[Manuele.Kirsch-Pinheiro@univ-paris1.fr](mailto:Manuele.Kirsch-Pinheiro@univ-paris1.fr)  
<http://mkirschp.free.fr>

---



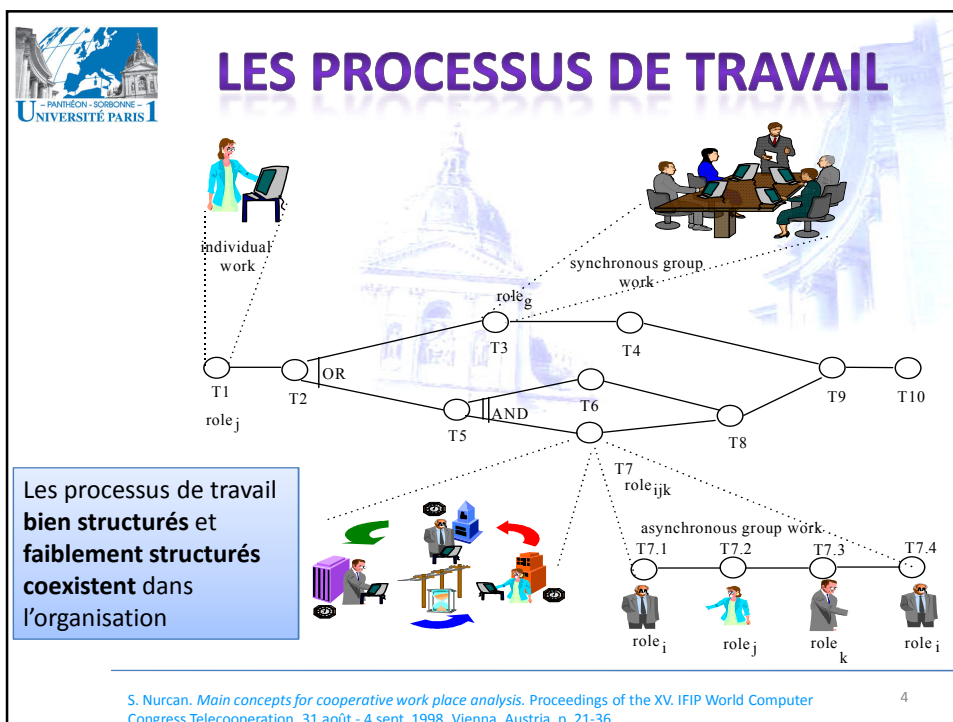
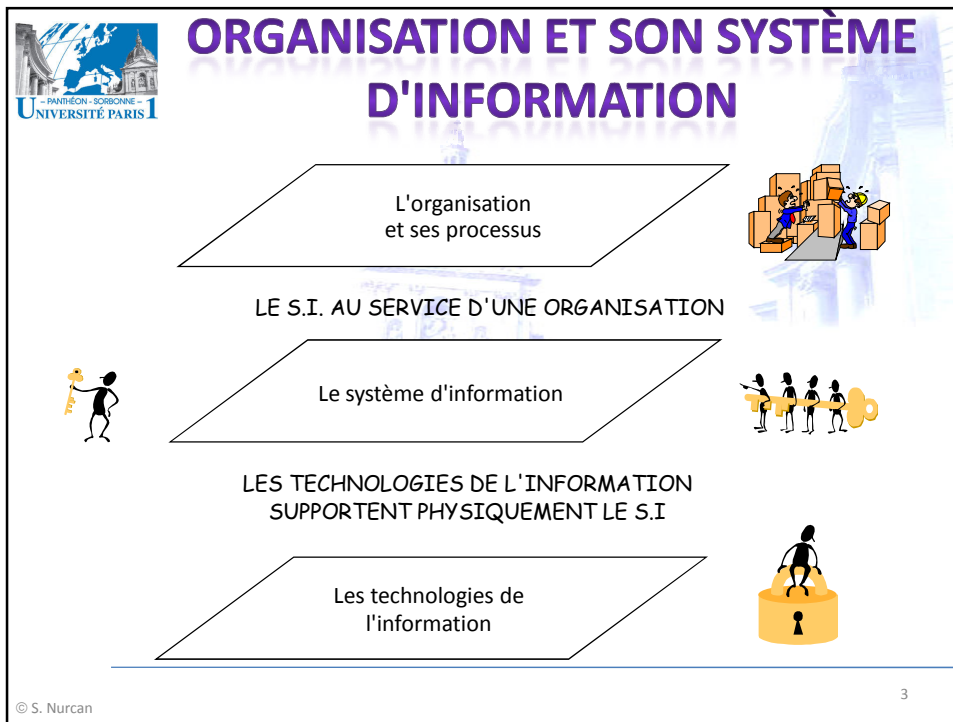
# PRÉSENTATION


## Contenu prévisionnel

- ✓ Introduction
  - ✓ Définitions
  - ✓ Historique
  - ✓ Taxonomies
- ✓ Les Systèmes de groupware
  - ✓ Introduction
  - ✓ Fonctionnalités
  - ✓ Exemples
- ✓ Collaboration sur le Web
  - ✓ Édition collaborative
  - ✓ Partage des ressources
  - ✓ Web 2.0
  - ✓ Gestions des sites
- Workflow
  - Définition
  - Caractéristiques
  - Fonctionnalités

---

05/04/2009 2






## SYSTÈME DE GESTION DE WORKFLOW (WFMS)

- Système informatique qui gère un **enchaînement de tâches**
  - Suivant une **procédure prédéfinie**
  - Permettant ainsi de **coordonner** les activités de **différentes personnes**
  - dans le but d'atteindre un **objectif** (résultat) *global*
- **Workflow** : Logiciel permettant d'assister, d'automatiser et/ou contrôler
  - Travail coopératif qui implique un nombre limité de personnes devant accomplir, en un temps limité, des **tâches articulées** autour d'une **procédure** définie et ayant un objectif global

---

S. Nurcan 5




## LES ORIGINES DU WORKFLOW

- Le workflow est né avec la GED lorsque les fournisseurs ont doté leurs produits de GED de fonctionnalités de workflow
- L'objectif principal des systèmes GED :
  - Convertir les documents papiers existants
  - Les documents convertis apparaissent sous plusieurs formes : images, formulaires révisables ou texte libre
  - Numérisation → Indexation → Stockage (archives, version) → Recherche

---

© S. Nurcan 6




## WORKFLOW : QUELQUES DÉFINITIONS

- Une **infrastructure** pour l'**automatisation** du flux de travail
  - "Le workflow désigne l'**automatisation** des **processus** que nous utilisons chaque jour pour faire notre **métier**. Une application de workflow automatise la **séquence** des actions, activités ou **tâches** que nécessite un processus de travail. Elle suit aussi l'**état** de chacune des **instances** du processus et **gère** le processus lui-même" (Marshak, 1994)
- La coopération et la **coordination** sont au cœur des applications workflows
  - "L'automatisation des flux est la structure qui est appliquée au mouvement de l'information afin d'améliorer les résultats d'un processus métier. Un logiciel d'automatisation de flux **gère activement** la **coordination** des activités entre les individus impliquées dans les processus métiers généraux" (Burns, 1994)

---

© S. Nurcan 7



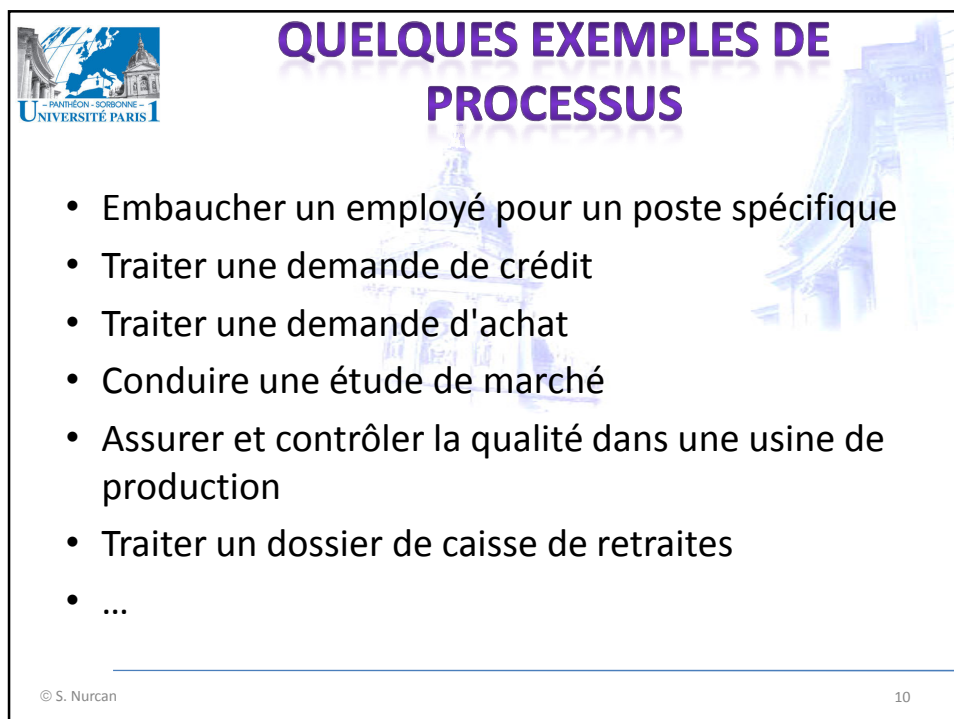
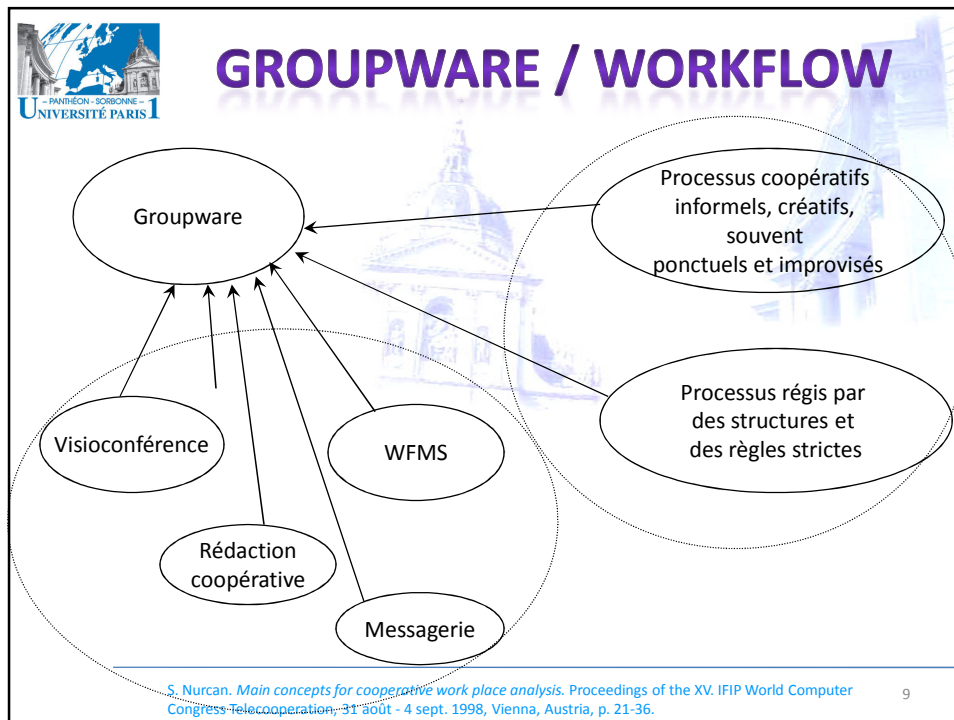
## GROUPWARE / WORKFLOW

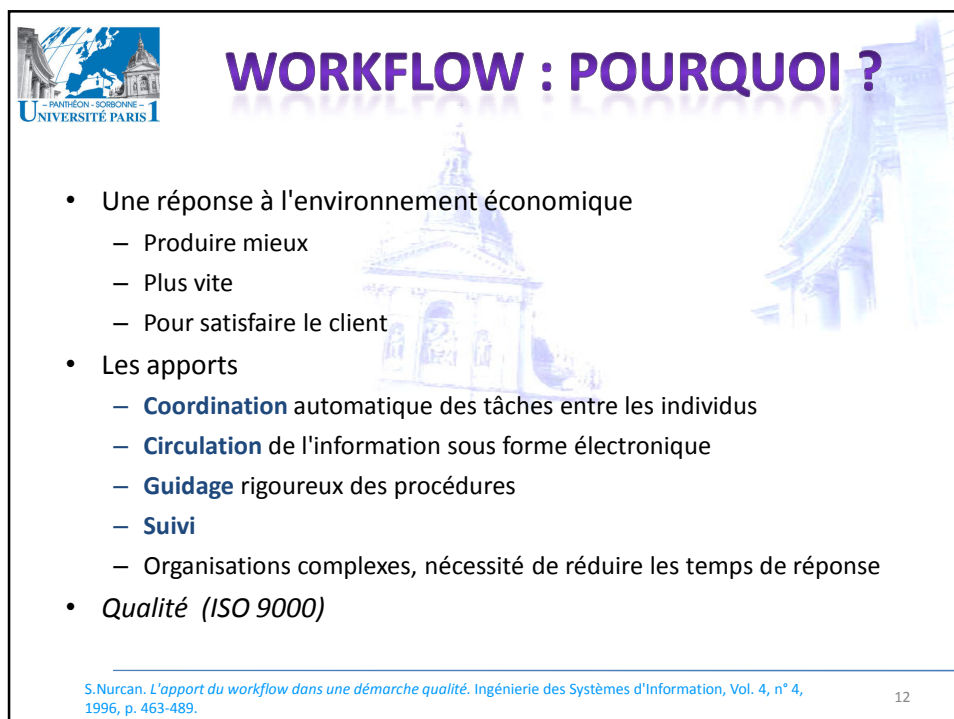
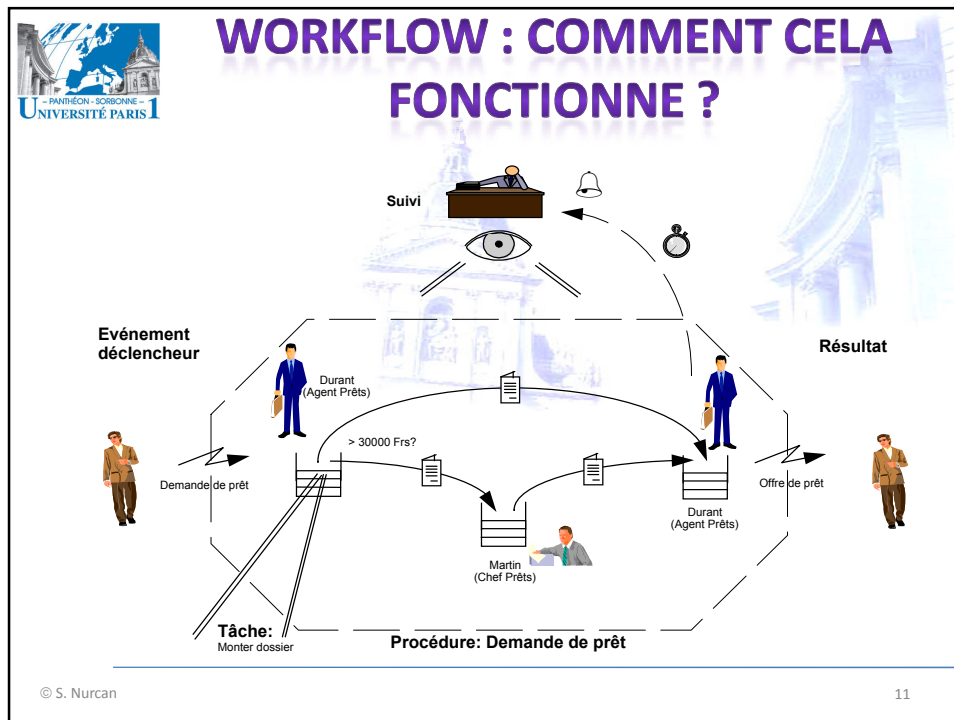
- "Le **workflow** cherche à automatiser les **règles formelles** en vue de restructurer les procédures métiers de l'entreprise; le **groupware** essaie de faciliter les **interactions informelles** entre les groupes en renforçant les aspects communication, coordination et coopération du travail en équipe" (Bock, 1992)
- "Le workflow est une technologie connexe au groupware... Les confondre serait de réduire le concept à sa dimension d'automate et ignorer la dimension humaine et organisationnelle du groupe" (Levan, 1994)


Dans sa définition la plus générale, le groupware désigne tout système qui aide le travail coopératif :  
si l'on accepte cette définition, le workflow est un groupware

---

© S. Nurcan 8








# WORKFLOW

- Éléments invariants
  - Un objectif :
    - Le workflow est mis en service pour accomplir un objectif
  - Des participants et des tâches :
    - Un **participant** est un individu, un groupe d'individus ou un périphérique (tout ce qui peut exécuter une tâche/activité)
    - Une **tâche** peut être exécutée par un ou plusieurs intervenants qui sont associés à un **rôle** (responsabilité) particulier
  - Exemples de tâches : approuver un document, traiter un formulaire, imprimer un document, numériser des formulaires

---

© S. Nurcan 13




# WORKFLOW

- Éléments invariants :
  - Des **flux** entre les **participants**:
    - Il peut s'agir d'un flux d'information, de documents ou d'instructions
    - Il existe de nombreux types de **contrôle de flux**, le plus simple étant le routage d'un document
      - **Branchements**
      - **Points de décision** (le circuit dépend alors du statut, de la disponibilité des utilisateurs ou d'autres conditions)
      - **Parallélismes**
    - Le processus peut aussi utiliser des flux des tâches peut impliquer des **files d'attente** dans lesquelles les intervenants participent aux traitements des tâches stockées
  - Des **documents** à traiter :
    - Différents types de données, formulaires fichiers texte, feuilles de calcul, images et autres données multimédia

---

© S. Nurcan 14




## WFMC

- Le **Workflow Management Coalition (WfMC)** a vu le jour en 1993 avec pour objectif de s'attaquer aux problèmes d'interopérabilité entre les divers plateformes de workflow
- Définition **terminologie** commune et des **interfaces de programmation** pour un workflow standard
- Mise en œuvre de **standards d'interopérabilité** et de connectivité entre les différents produits de workflow
- L'adoption de normes communes pour le déploiement du workflow dans les industries

---

© S. Nurcan 15



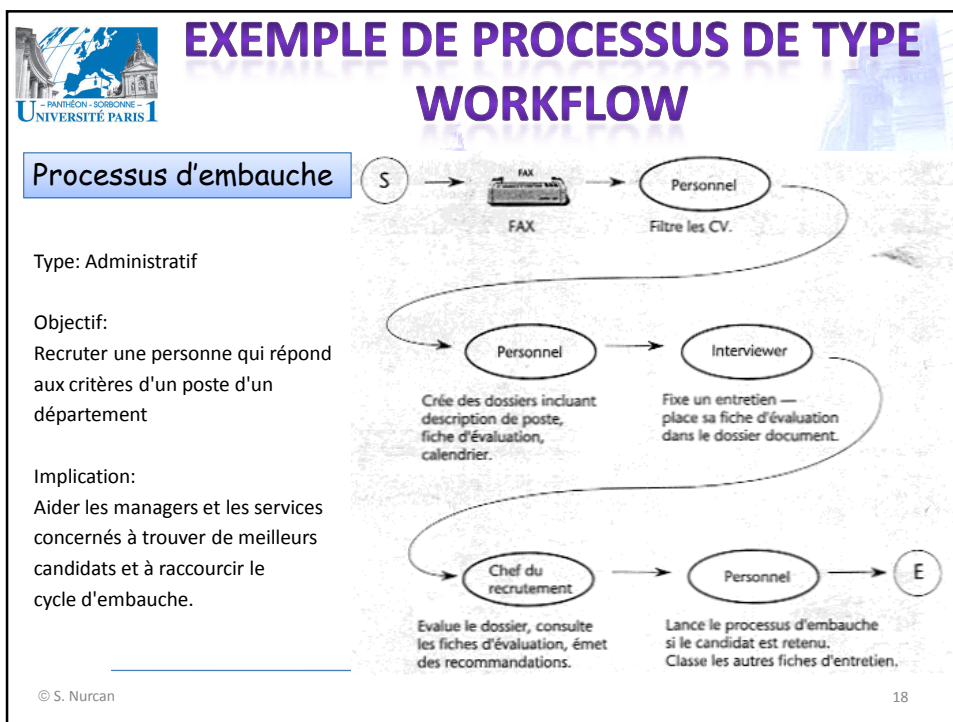
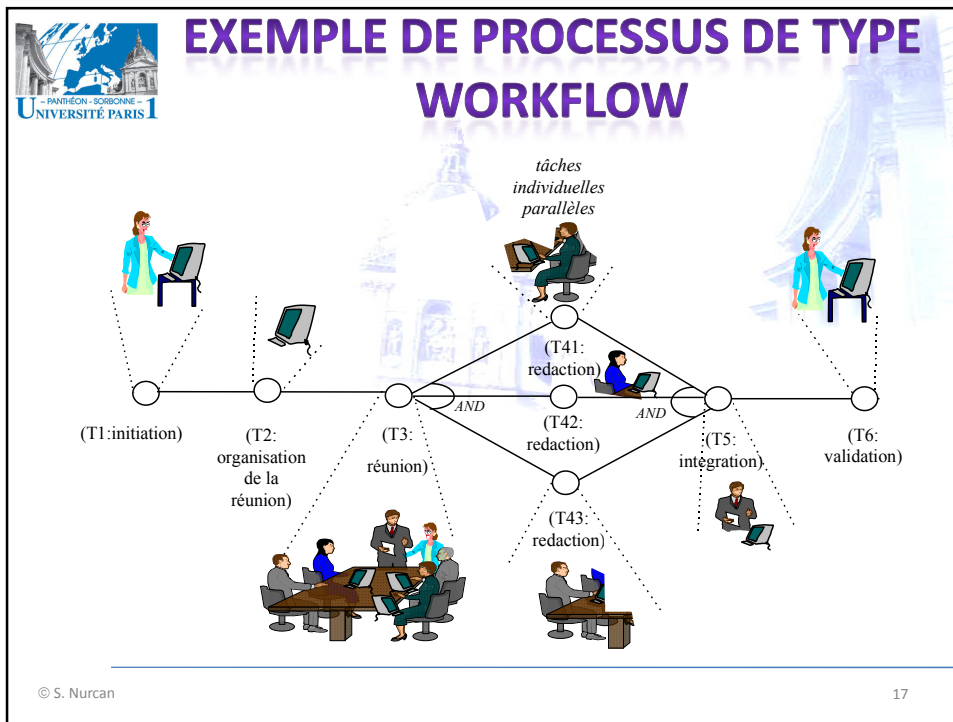
## NORMES ET STANDARDS

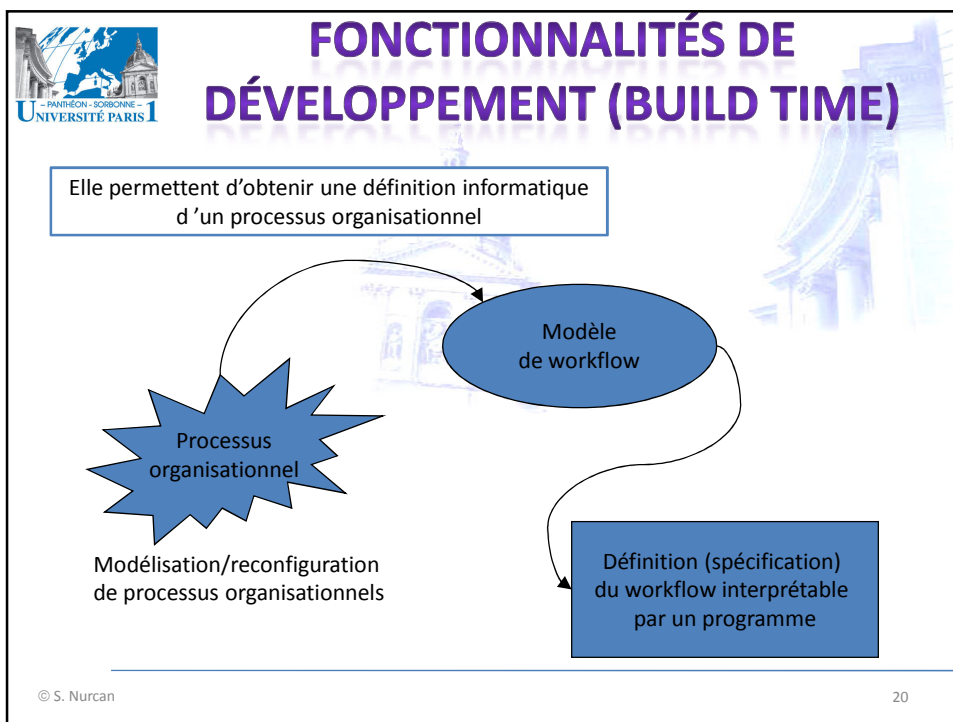
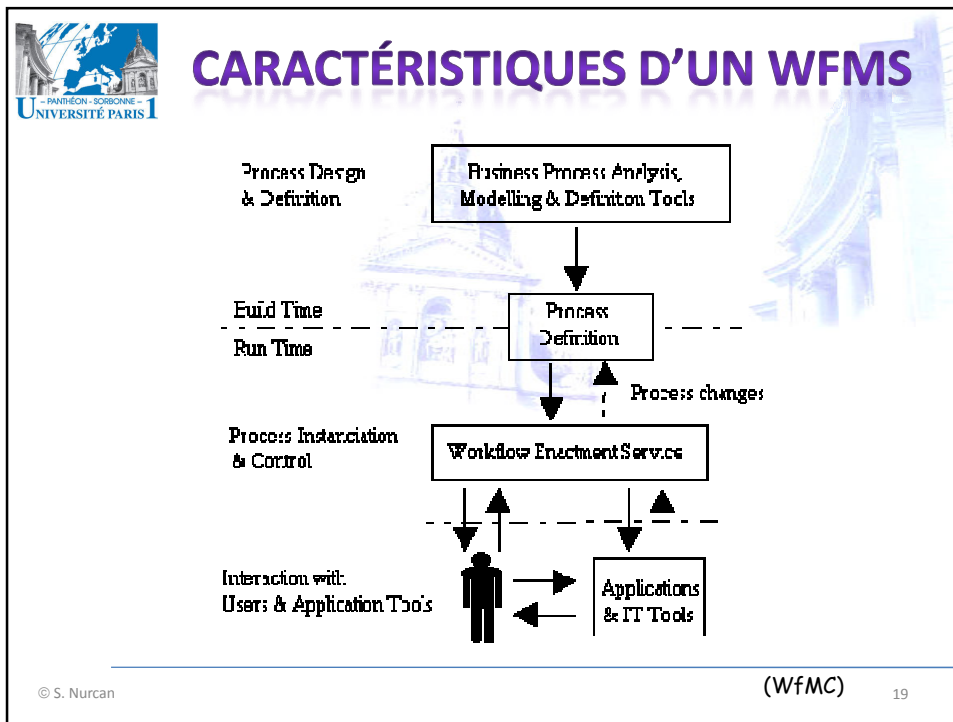
- **Modèle de workflow (*process definition*) :**
  - Effort de standardisation sur la capacité que peuvent avoir les produits de workflow à importer ou exporter les modèles créés par les outils de différents fournisseurs
- **Instance du processus (*process instance*) :**
  - L'objectif est que différents moteurs de workflow puissent participer au traitement d'une instance de workflow, qu'un processus défini par un outil puisse être exporté puis activé et interprété par d'autres moteurs de workflow
- **Échange de meta-données :**
  - Des meta-informations relatives à un workflow (associées généralement à des outils graphiques, des tableurs etc.) pourront être interprétées par différents produits si elles sont présentées dans un format standard
- **Corbeille (*worklist ou to do list*) :**
  - Normalisation de la liste des tâches affectées à un utilisateur


---

© S. Nurcan 16










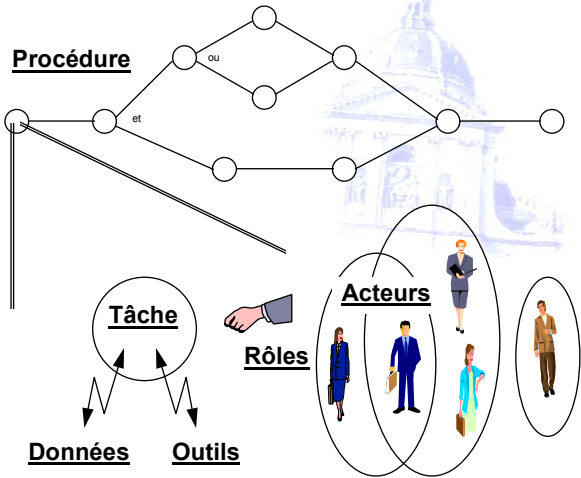
## FONCTIONNALITÉS DE DÉVELOPPEMENT (BUILD TIME)

- Définition du modèle de workflow
  - Le **modèle du workflow** capture les **règles** et les **étapes** qui doivent suivre les réalisations du processus concerné

© S. Nurcan 21



## MODÈLE DE WORKFLOW



The diagram illustrates the components of a workflow model:

- Procédure**: A flowchart showing a sequence of steps connected by lines, with 'ou' (or) and 'et' (and) indicating logical relationships between steps.
- Acteurs**: Represented by icons of people, they are the individuals who perform the tasks.
- Tâche**: A central concept that involves **Données** (Data) and **Outils** (Tools).
- Rôles**: Represented by a hand icon, they are the specific functions or positions associated with the actors.

On the right, a vertical flow shows the relationship between **Rôles**, **Coordination**, and **Tâches**, with arrows indicating the flow from roles to coordination and then to tasks.

S.Nurcan. *Analyse et conception de systèmes d'information coopératifs*, Numéro thématique "Multimédia et collectif" de TSI (Techniques et Science Informatiques), Vol. 15, n° 9, 1996, p. 1287-1315.




## MODÈLE DE WORKFLOW

- **Groupes & rôles**
  - Les **tâches** peuvent être associés à des **participants** de natures diverses: personnes, périphériques, processus (d'autres workflows), applications (on les appelle aussi acteurs)
  - Les **individus** accomplissent le véritable travail au sein du workflow
  - Un **rôle** désigne un titre, une fonction ou une dénomination affecté(e) à un individu ou un groupe
  - Un **rôle** est la définition d'une **intention organisationnelle** partagée par un ensemble de participants (ou groupes de participants), chacun ayant les **mêmes privilèges** et **obligations** vis à vis de l'ensemble des processus de l'organisation
  - Un **participant (acteur)** peut jouer **plusieurs rôles** et un **rôle** peut être affecté à **plusieurs participants** (acteurs)

---

© S. Nurcan 23




## MODÈLE DE WORKFLOW

- **Types de tâche/activité**
  - Activités nécessitant l'intervention humaine pour leur exécution
  - Activités automatiques
- **Types d'entités exécutantes**
  - Humains
  - Interpréteurs de scripts, compilateurs
  - « legacy » systèmes
  - Serveurs
  - SGBDs

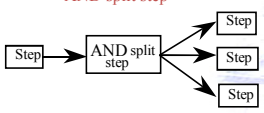

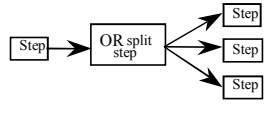
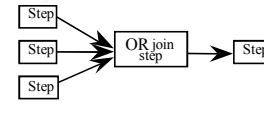
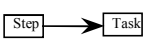
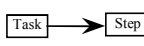
---

© S. Nurcan 24




# MODÈLE DE WORKFLOW

Relations de précédence entre étapes/nœuds

<p style="text-align: center; color: red; font-weight: bold;">AND split step</p>  <p style="font-size: small;">A single thread of control splits into parallel process steps</p>	<p style="text-align: center; color: red; font-weight: bold;">AND join step</p>  <p style="font-size: small;">Parallel process steps converge into a single thread of control (called rendezvous or synchronization point)</p>
<p style="text-align: center; color: red; font-weight: bold;">OR split step</p>  <p style="font-size: small;">A single thread of control allows to decide which branch to take when multiple alternatives exist</p>	<p style="text-align: center; color: red; font-weight: bold;">OR join step</p>  <p style="font-size: small;">Alternative process steps converge into a single process step activated when only one alternative process step is terminated</p>
 <p style="font-size: small;">A task can only be preceded by one step except the first one</p>	 <p style="font-size: small;">A task can only be followed by one step except the last one</p>

© S. Nurcan
25

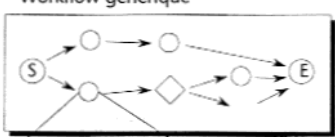


# MODÈLE DE WORKFLOW

Une tâche peut être tout un processus

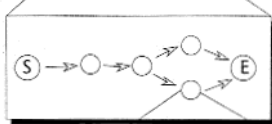
Imbrication de workflows

**Workflow générique**

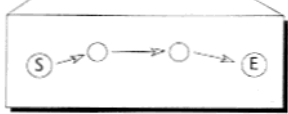


**Niveau 0**

**Workflow imbriqué**




**Niveau 1**



**Niveau 2**


© S. Nurcan
26



## FONCTIONNALITÉS DE CONTRÔLE (RUN TIME)

- Moteur d'exécution de workflow (*workflow engine*)
  - Création, suppression des instances de workflow
  - Contrôle de l'ordonnancement des activités
  - Interaction avec les outils d'application et ressources humaines
- Les moteurs d'exécution peuvent être distribués sur plusieurs plates-formes afin de gérer les processus opérationnels géographiquement distribués

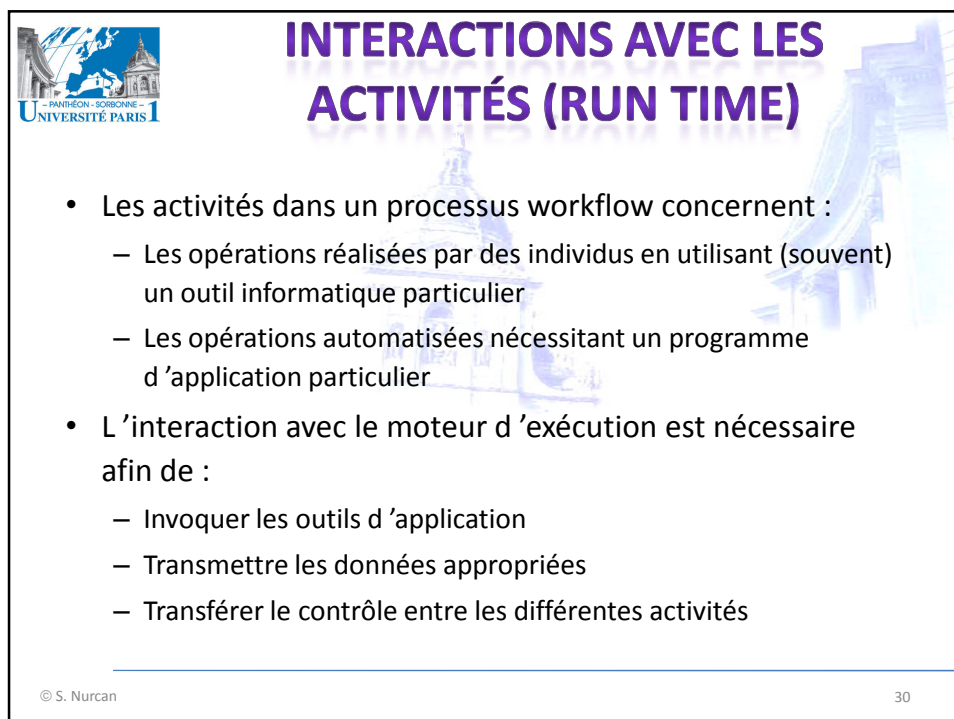
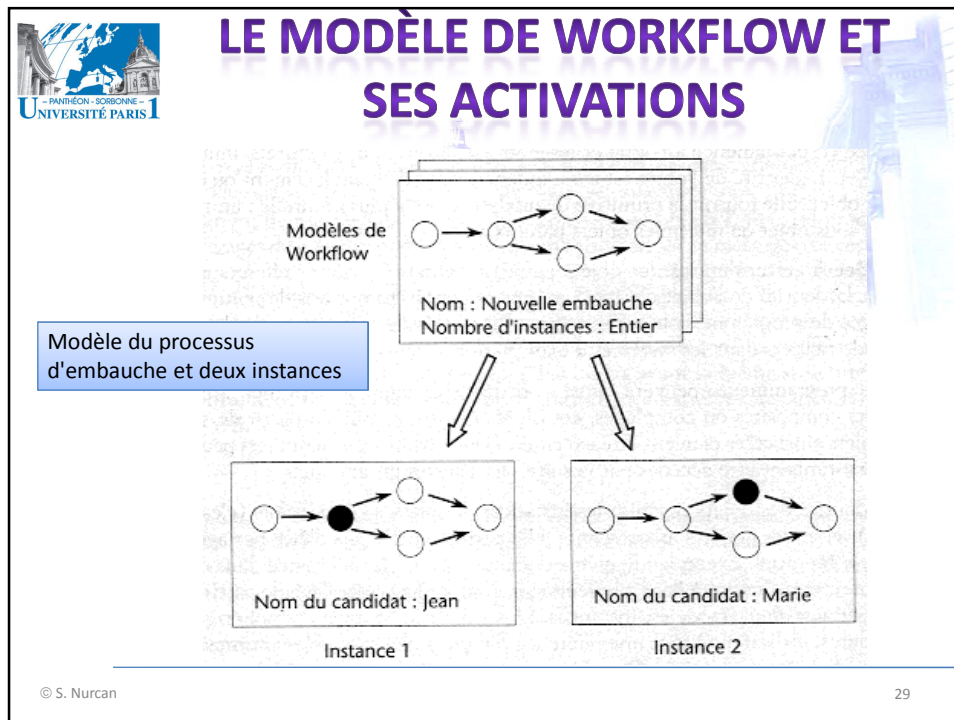
© S. Nurcan 27




## FONCTIONNALITÉS DE CONTRÔLE (RUN TIME)

- L'activation du modèle de workflow (processus)
  - Les modèles de workflow (*process definition*) sont similaires aux **classes** qui décrivent la **structure** et le **comportement** de leurs **instances** (les activations du modèle)
  - Il existe deux types de **variables** et **d'opérations** dans la définition d'un modèle de workflow :
    - celles qui appartiennent aux **instances** ou activations du modèle
    - celles qui appartiennent au **modèle**
      - la description du modèle, son créateur, la personne qui l'a modifié pour la dernière fois, la personne qui dispose des autorisations pour le modifier...

© S. Nurcan 28






## FONCTIONNALITÉS DE CONTRÔLE (RUN TIME)

- **Suivi statut et statistiques**
  - Il existe des outils de suivi du statut et des outils de statistiques qui permettent à l'information d'être gérée au fur et à mesure que le travail progresse
    - Exemples: *Nom du workflow actif, activé à (heure), activé par (opérateur/processus), durée estimée, tâche en cours, durée du traitement, personne à qui elle a été affectée, etc.*
  - Ces informations sont généralement transmises aux responsables et disponibles sous forme de rapports
  - A posteriori, un historique du déroulement des opérations permet d'analyser les goulets d'étranglement potentiels et d'améliorer la qualité du processus

---

© S. Nurcan 31




## FONCTIONNALITÉS DE CONTRÔLE (RUN TIME)

- **Règles et conditions**
  - Les règles permettent aux messages d'être expédiés en fonction de certains critères et conditions
  - Exemple: *Si le workflow "autorisation d'embauche" est stoppé depuis plus de 3 jours au nœud Directeur des RH et si l'embauche est urgente, alors envoyer un mémo au manager du département*
- **Notification**
  - Règles intégrant des conditions et des actions activées par des changements d'état des objets ou des contraintes de temps
  - *Envoyer reçu à <acteur notifié> si le workflow a atteint une certaine étape ou Envoyer reçu à <acteur notifié> si un certain nœud n'a pas été activé*

---

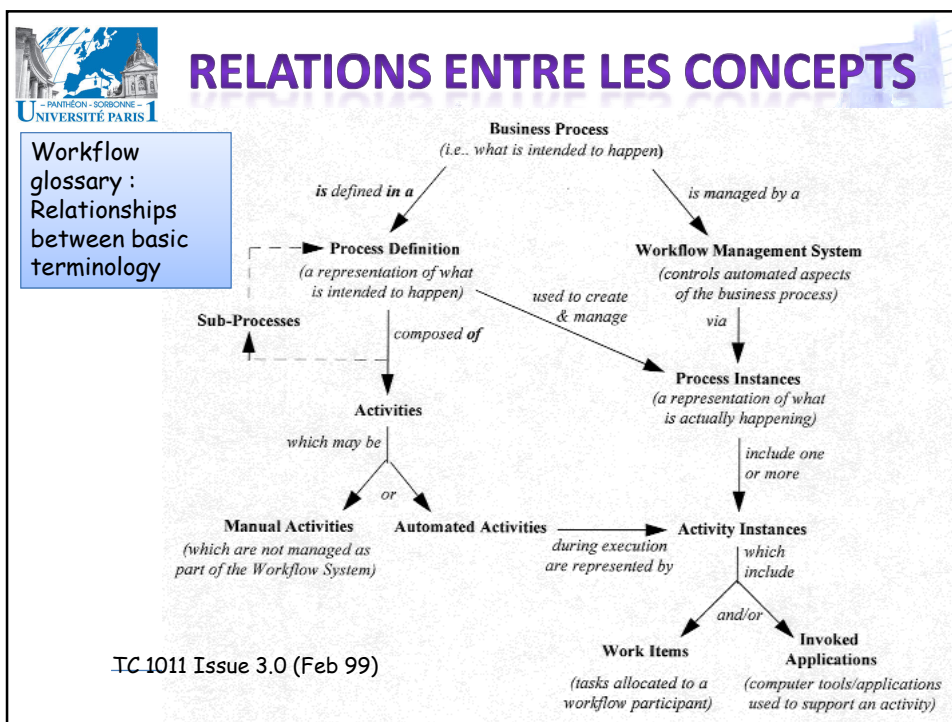
© S. Nurcan 32

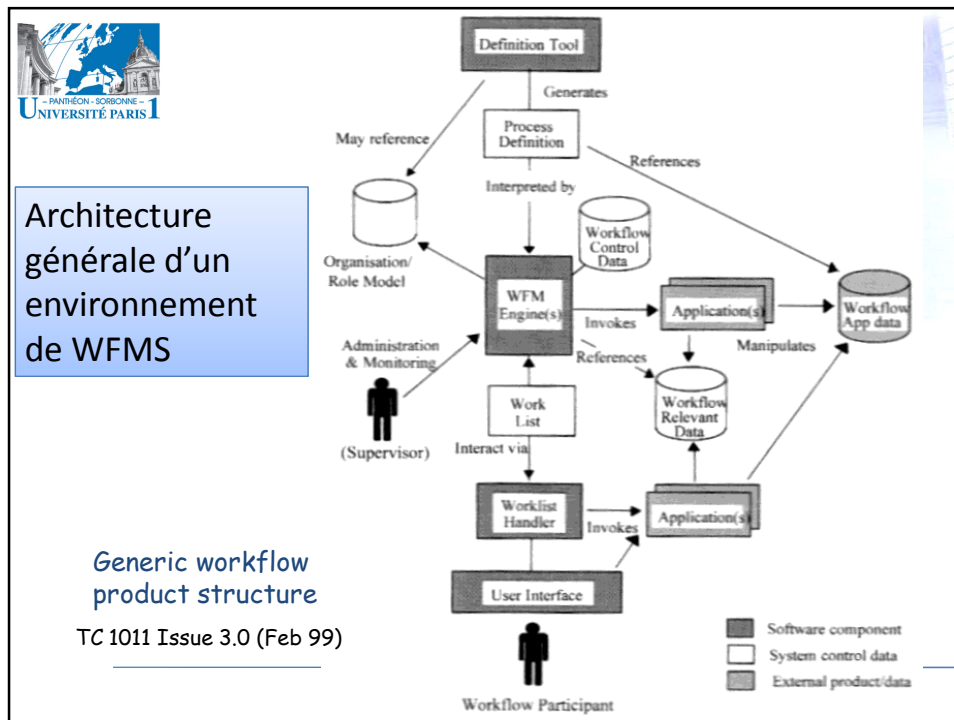


 **FONCTIONNALITÉS DE CONTRÔLE (RUN TIME)**

- **Modification dynamique des affectations (rétraction)**
  - La **réaffectation** d'une tâche à un nouveau participant que celui auquel le moteur de workflow l'avait affecté
  - Plusieurs types de rétraction: **modification** de l'affectation des rôles, retrait de la tâche et l'affectation à un autre participant
  - Le logiciel peut suivre à la **progression** et le **statut** des différentes affectations, les rétractions et les réaffectations, la durée des tâches affectées, l'état d'une instance de workflow à un instant t, etc.
  - Le **suivi** de ces informations par le management est primordial pour l'amélioration de la performance des processus et la qualité du service fourni.

© S. Nurcan 33






## CORBEILLE ET GESTIONNAIRE DE CORBEILLE (WORKLIST)

- Lorsque des interactions avec des utilisateurs sont nécessaires pendant l'exécution du workflow, le **moteur d'exécution** place les **tâches** dans une **corbeille** (worklist) à l'attention du gestionnaire de corbeille
- La corbeille peut être visible ou invisible à l'utilisateur
- Le **gestionnaire de corbeille** gère les interactions entre les participants du workflow et le service d'exécution.
  - Il peut être plus ou moins sophistiqué
  - Exemple : contrôler l'allocation du travail à un ensemble d'utilisateurs en utilisant des fonctions d'équilibrage de la charge de travail et de réassignation des tâches

© S. Nurcan 36




## OUTIL DE DÉFINITION DU PROCESSUS

- Il est utilisé pour créer une **description du processus** qui soit exécutable par un programme
- Il peut être basé sur un langage formel, le modèle objet, un langage de script, ou un ensemble de commandes de routage entre les participants
- Il peut être fourni comme partie d'un logiciel WfMS ou d'un produit d'analyse de processus organisationnel.
- Dans ce dernier cas, il est nécessaire d'avoir un **format d'échange** compatible entre (vers/de) les définitions de processus et le logiciel de workflow

---

© S. Nurcan 37

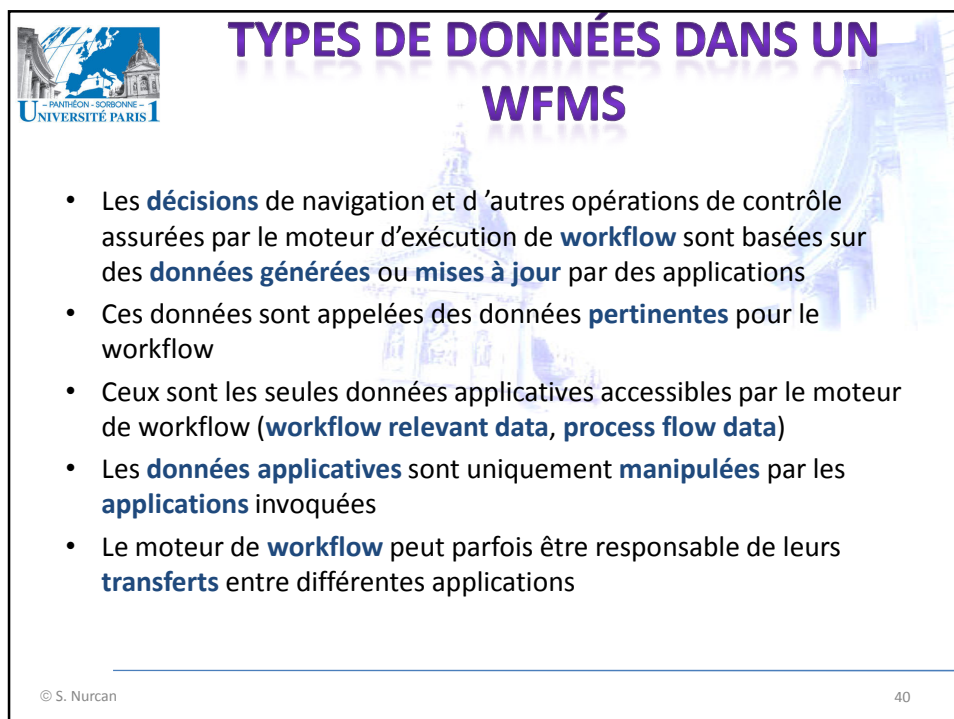
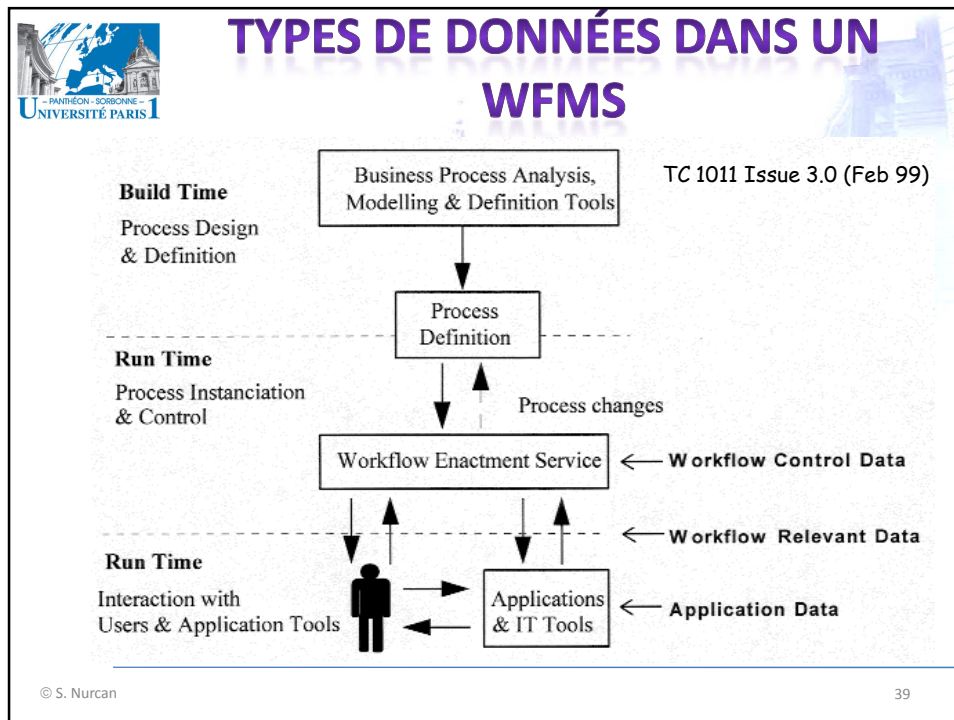


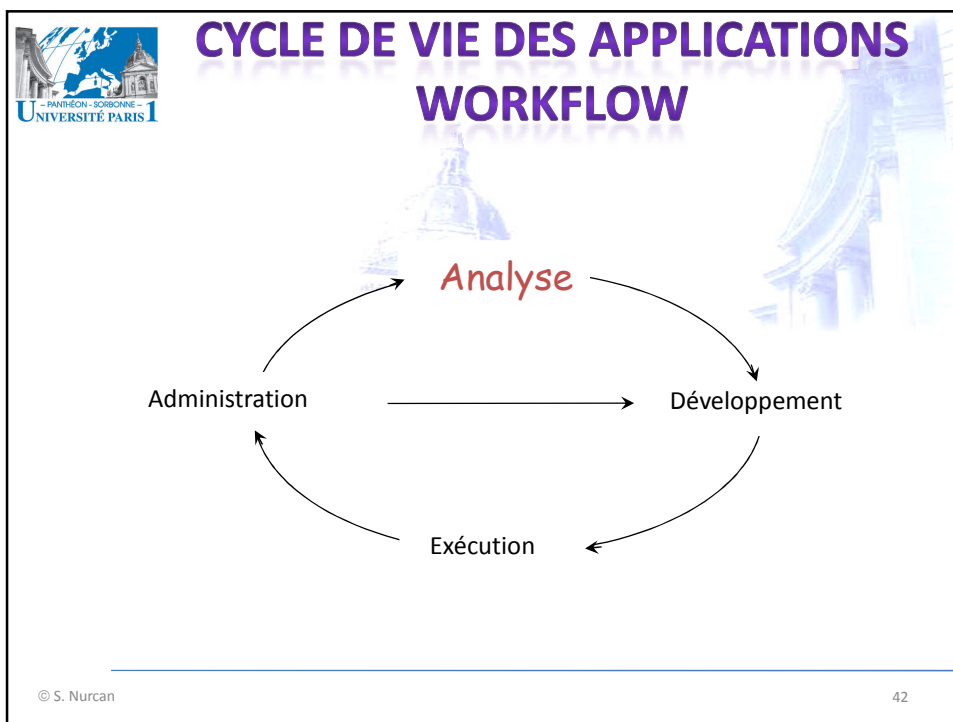
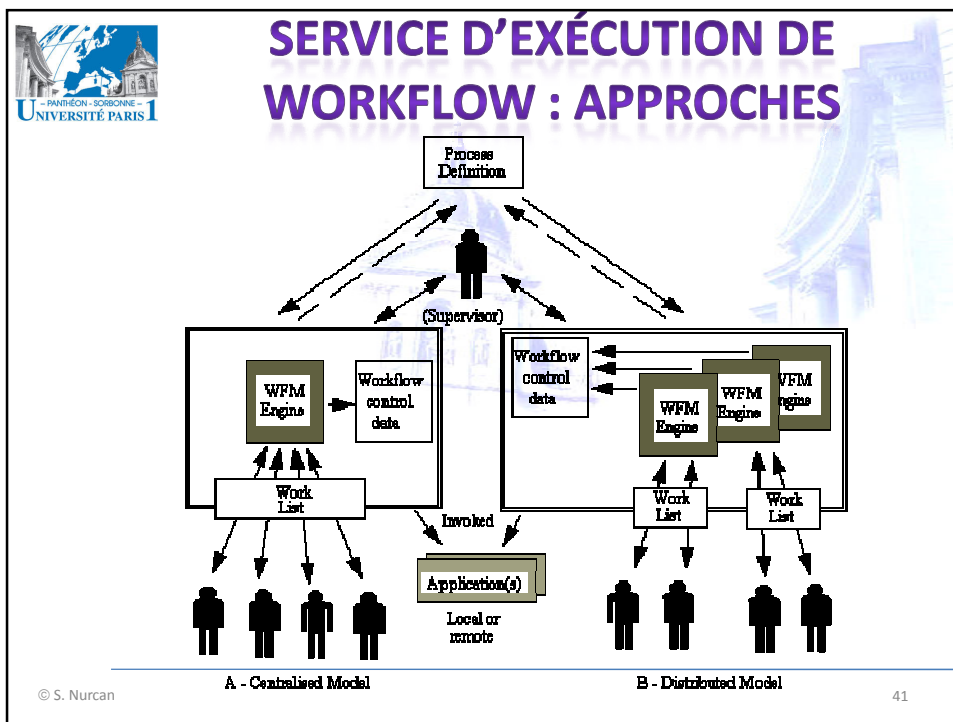
## DÉFINITION DU PROCESSUS


- La définition du processus contient toutes les informations nécessaires sur le processus afin qu'il soit exécutable par le moteur d'exécution du workflow :
  - les conditions de début et de terminaison,
  - les activités incluses et les règles de navigation entre elles
  - les participants
  - les références aux applications qui doivent être invoquées
  - la définition des données workflow pertinentes

---

© S. Nurcan 38








## ANALYSE

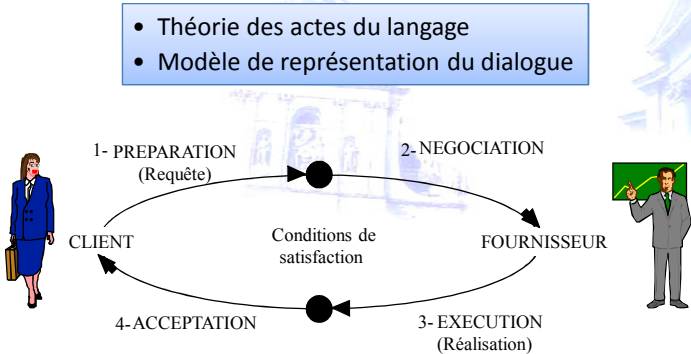
- Quels modèles de produit choisir pour concevoir des applications workflow ?
- **L'objectif de la modélisation :**
  - Produire un modèle de processus qui servira de base pour la spécification du Workflow
- **Catégories de modèles de processus :**
  - orienté communication (message)
  - orienté artéfact
  - orienté activité

© S. Nurcan 43



## MODÉLISATION ORIENTÉE COMMUNICATION


- Théorie des actes du langage
- Modèle de représentation du dialogue



Le modèle «Conversation pour Action»

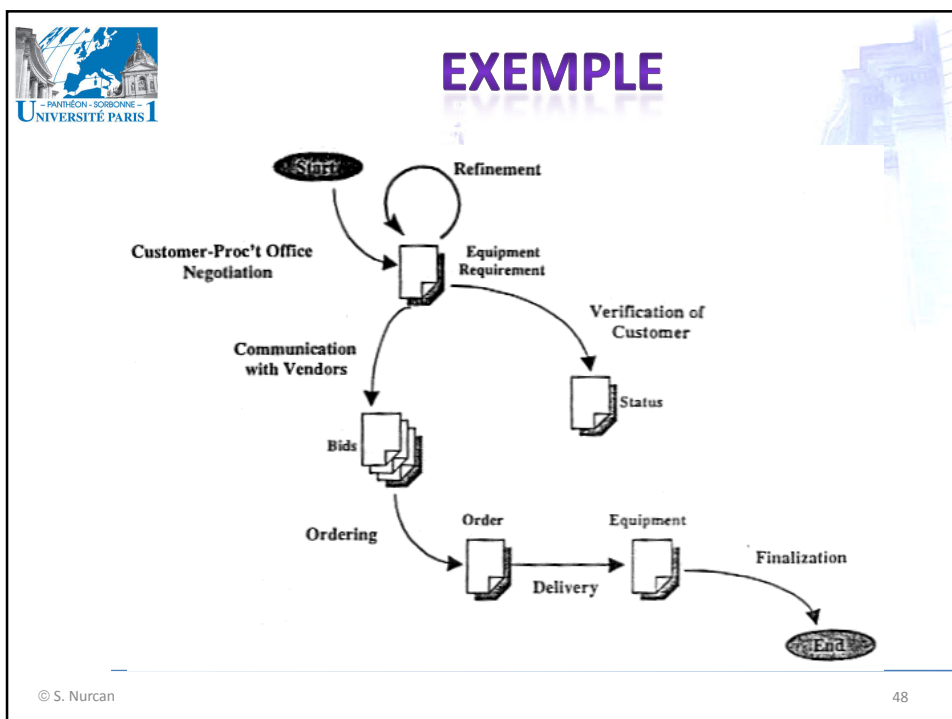
© S. Nurcan 44



 **MODÉLISATION ORIENTÉE ARTÉFACT**

- L'accent est mis sur les **objets (artéfacts)** qui sont créés, modifiés et utilisés pendant le processus
- La modélisation du processus est basée sur les **produits** et leur «**route**» à travers les activités du workflow

© S. Nurcan 47





**MODÉLISATION ORIENTÉE ACTIVITÉ**

- Décomposition du processus de travail en **tâches** (**activités**) ordonnancées suivant leurs dépendances
- L'avantage principal: peut rapidement aboutir en une spécification de workflow

```

graph TD
    PE[Procure Equipment]
    VS[Verify Status]
    GB[Get Bids]
    O[Order]
    VS --> GB
    GB --> O
    PE ---|consists of| VS
    PE ---|consists of| GB
    PE ---|consists of| O
  
```


© S. Nurcan 49

**CYCLE DE VIE DES APPLICATIONS WORKFLOW**

```

graph TD
    Analyse --> Développement
    Développement --> Exécution
    Exécution --> Administration
    Administration --> Analyse
  
```

© S. Nurcan 50



## LANGAGES DE DESCRIPTION

- **Puissance d'expression du langage de spécification**
  - Le modèle de workflow doit être spécifié en utilisant un langage qui peut être procédural, déclaratif, basé sur des flux de données, visuel, basé sur des réseaux de Petri, etc.
- **Exigences**
  - Avoir une sémantique précise et non ambiguë de manière à ce que le modèle de workflow puisse être exécuté par un programme
  - Être suffisamment expressif pour spécifier les activités du workflow et leurs interrelations (ordre partiel)
  - + expression des responsabilités associés aux activités, délais...
- La spécification du workflow doit contenir suffisamment d'information pour le rendre **portable** à travers des systèmes différents

---

© S. Nurcan 51

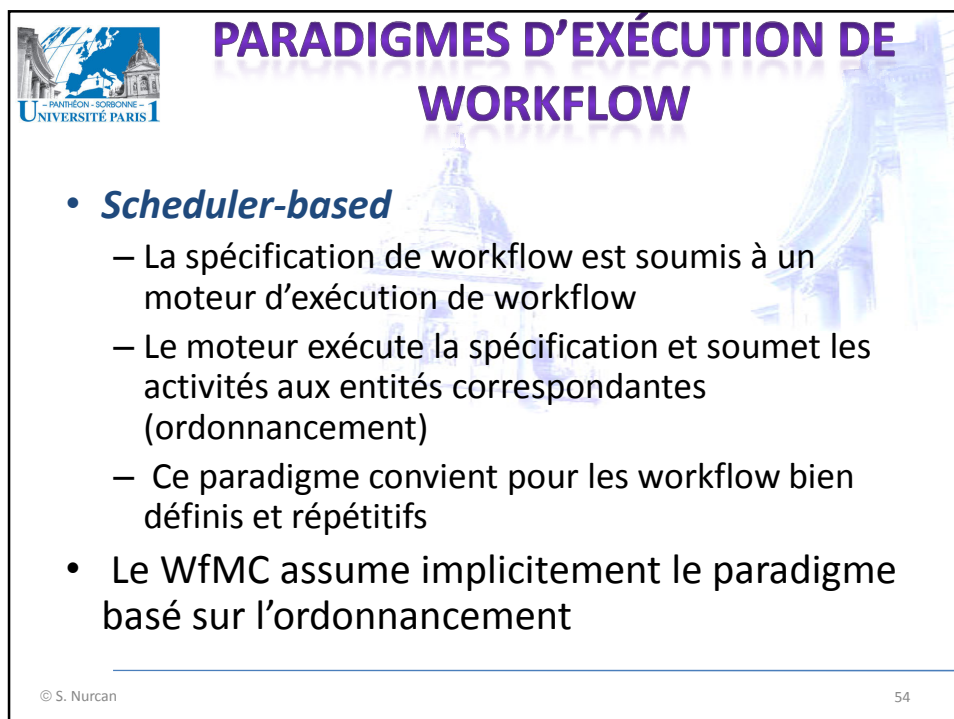
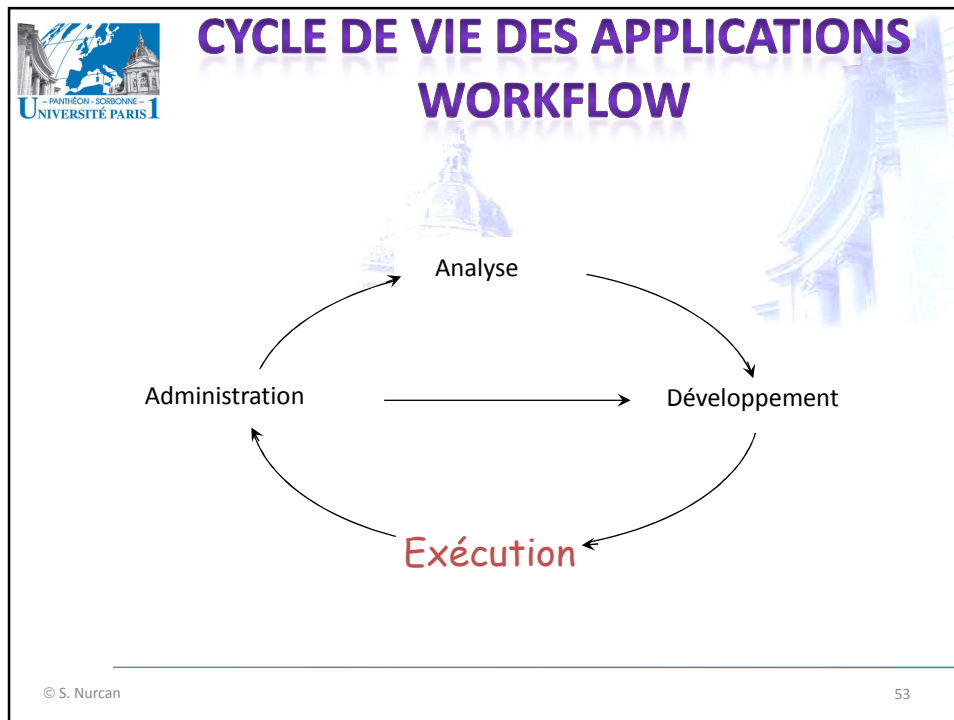



## LANGAGES DE DESCRIPTION

- **Modularité**
  - Un bon langage de spécification doit être modulaire de façon à ce que des bibliothèques de « sous-workflow » puissent être utilisées (réutilisabilité)
- **Traitement des exceptions:**
  - Chaque composant de workflow (le workflow entier, activités, rôles) doit avoir des comportements par défaut prédéfinis pour le traitement des exceptions.
- Le langage de spécification de workflow doit être suffisamment formel pour se prêter à une analyse formelle de « *correctness* » telle que absence de verrous mortels, terminaison

---

© S. Nurcan 52






## PARADIGMES D'EXÉCUTION DE WORKFLOW

- **Data-flow oriented**
  - Le workflow est traité comme un conteneur de données qui passe d'un poste à un autre
  - A chaque poste les données sont partiellement traitées selon le modèle de workflow et les règles d'exécution locales à la station
  - La route du workflow est déterminée localement, basée sur la spécification, la trace de l'exécution et les résultats du traitement sur le poste courant
- Ce paradigme est approprié pour les workflow partiellement définis et orienté but

© S. Nurcan 55



## PARADIGMES D'EXÉCUTION DE WORKFLOW

- **Information pull**
  - Un workflow complètement dynamique est créé comme réponse à un besoin d'information
  - Le système peut être basé sur des agents intelligents et autonomes qui coopèrent pour trouver l'information et la présenter à l'utilisateur
  - Ce paradigme ne peut pas être utilisé pour les workflow de production ni même les workflow administratifs
- Il s'agit d'un champ de recherche intensif, mais il existe peu de résultats concrets pour l'instant

© S. Nurcan 56




## GARANTIES OFFERTES PAR UN MOTEUR D'EXÉCUTION

- **Exécution correcte d'une instance de workflow**
  - Exécution correcte de toutes les activités
  - Définition d'un ensemble d'états de terminaison acceptables
  - Définition d'un ensemble de prédicats de consistance de données
- Fonction importante de WFMS relative à cette garantie :
  - **Vérification du modèle de workflow et génération d'un plan d'exécution**
  - Le système doit pouvoir rejeter le workflow (instance) sans essayer de l'exécuter s'il ne peut garantir la terminaison de l'exécution dans l'un des états acceptables prévus

---

© S. Nurcan 57




## GARANTIES OFFERTES PAR UN MOTEUR D'EXÉCUTION

- **Propriétés transactionnelles**
  - Nécessaire en particulier pour les workflow de production
  - La plupart des produits ne les fournissent pas toutes
    - Atomicité de l'échec
    - Consistance des données
    - Deadlines

---

© S. Nurcan 58




## PROPRIÉTÉS TRANSACTIONNELLES

- **Atomicité de l'échec**
  - Un workflow doit être exécuté entièrement ou pas du tout
  - aucun résultat partiel ne doit subsister dans le système
- Méthodes empruntées aux technologies de SGBD et systèmes distribués
  - *Forward recoverability*: l'état est récupéré à partir du fichier LOG et l'exécution continue
  - *Backward recoverability*: les effets des activités exécutées sont supprimés (**rollback**)
  - *Compensation* : les effets des activités terminées sont défaits (undo) en exécutant d'autres activités ayant les effets opposés.  
Ex: retrait pour compenser un dépôt
- Les problèmes de reprise ne sont pas purement techniques mais bien souvent administratifs.

---

© S. Nurcan 59




## PROPRIÉTÉS TRANSACTIONNELLES

- **Consistance des données :**
  - Une imbrication non contrôlée d'opérations concurrentes sur les données peut causer des inconsistances des données
- Différents degrés d'isolation entre les activités de workflow allant de « pas de contrôle » vers des restrictions de plus en plus importantes
  - Certaines activités nécessitent une consistance absolue des données même si cela signifie une perte de performance.  
D'autres peuvent accepter des réponses approximatives
- Caractéristique d'un WFMS réussi :
  - **Capacité à fournir le niveau de consistance de données requis pour des instances de workflow particulières**

---

© S. Nurcan 60




## PROPRIÉTÉS TRANSACTIONNELLES

- **Deadlines**
  - La plupart des WFMS offrent cette fonctionnalité
  - Deadlines pour les activités
    - temps absolu (avant 17 heures)
    - temps relatif (dans une heure)
- Dépendances temporelles entre les activités (l'activité doit être exécutée au plus tard une heure après la terminaison d'une autre tâche)
  - **Hard deadlines** : l'activité est exécutée à temps ou pas du tout
  - **Soft deadlines**: principe du meilleur effort pour minimiser le nombre de violations de deadlines

---

© S. Nurcan 61



## GARANTIES OFFERTES PAR UN MOTEUR D'EXÉCUTION

- Pour fournir ces garanties, les WFMS utilisent différentes techniques :
  - des moniteurs de transaction pour l'atomicité de l'échec
  - des algorithmes d'ordonnancement et des plannings pour les deadlines et l'exactitude de l'exécution

---

© S. Nurcan 62