

IFT 2255 - Genie Logiciel

Équipe de développement

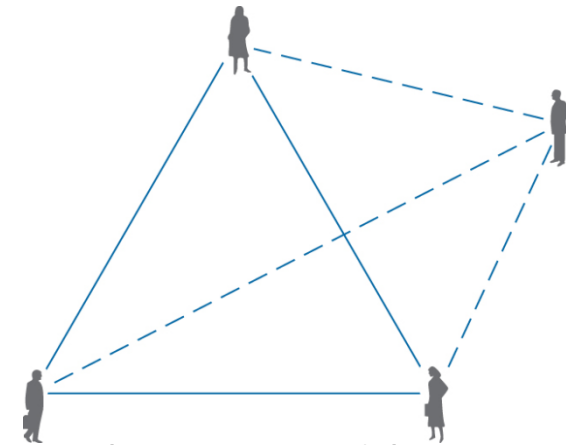
QUESTION

Un logiciel doit être complété en 3 mois, mais requiert 1 personne-année de programmation. Que faire ?

- Si 1 programmeur peut faire le travail en 1 an, 4 programmeurs peuvent le faire en 3 mois
- C'est absurde!
- 4 programmeurs vont probablement prendre presque un an
- La qualité du produit sera souvent moindre

Ajouter plus de programmeurs

- 3 programmeurs, échéance approche, code encore incomplet
- 3 canaux de communication entre 3 programmeurs
- Solution: ajouter un 4^e!
- Mais les 3 autres doivent expliquer en détail
 - Ce qui a été fait
 - Ce qu'il reste à faire
- Loi de Brooks



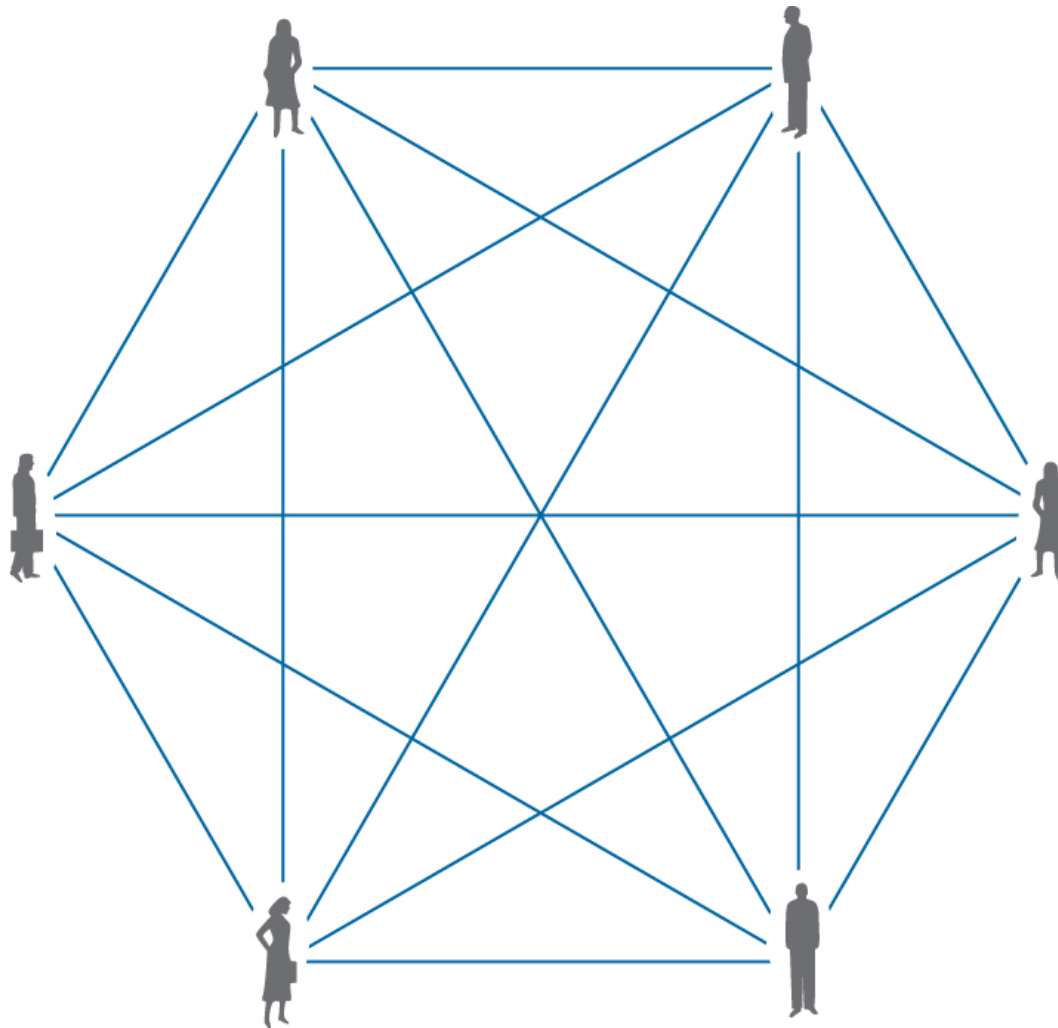
« Ajouter de la main-d'œuvre à un projet logiciel en retard le retardera » [Fred Brooks 1975. The Mythical Man-Month]

Approche démocratique

- Programmeurs sont très attachés à leur code
 - Nomment leurs modules d'après eux
 - Ils voient leurs modules comme une extension d'eux-mêmes
- Concept de base: **programmation non-égocentrique**
 - Encourage les membres de l'équipe à trouver des fautes dans le code
 - Une faute doit être considérée comme un événement normal et accepté
- Équipes démocratiques sont extrêmement productives
- Meilleures quand le problème est difficile
- Fonctionnent bien dans un environnement de recherche
- Difficultés pour le gestionnaire
 - Équipe démocratique difficile à introduire dans un environnement qui ne l'est pas

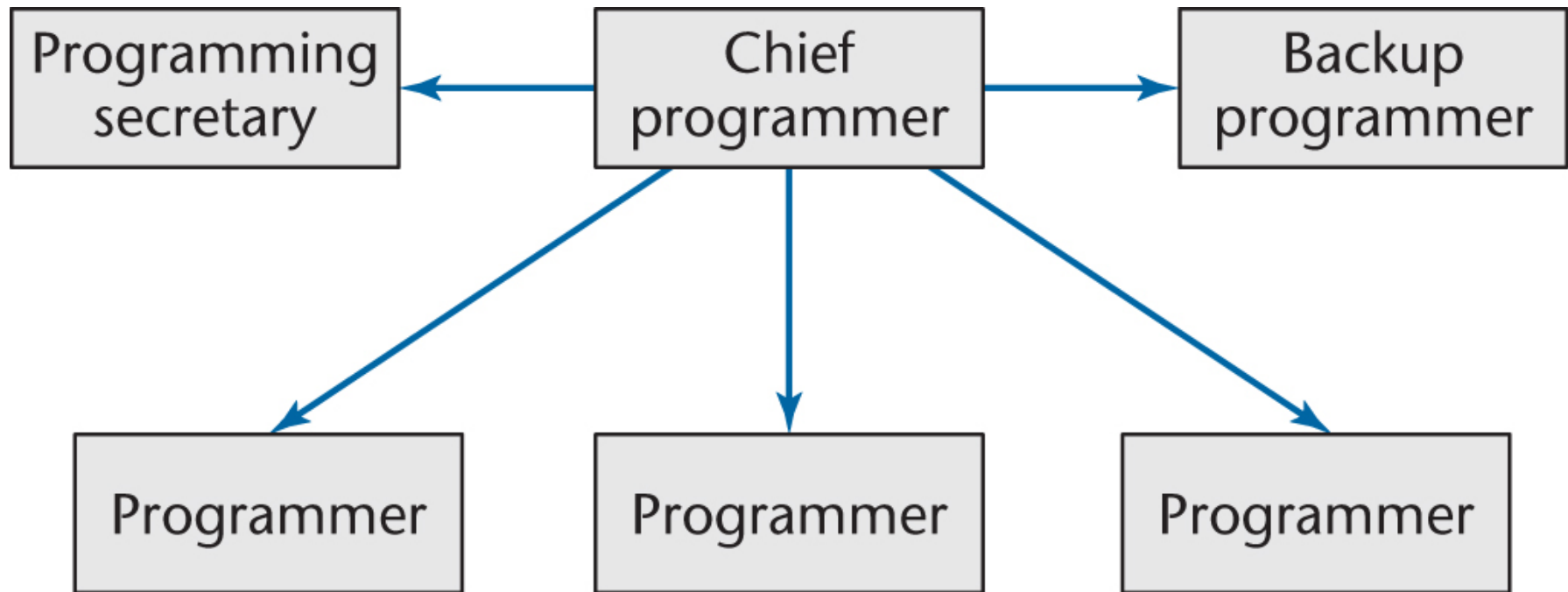
Équipe démocratique

Comment réduire l'excès de canaux de communication ?



Équipe de programmeur chef classique

Introduire hiérarchie et spécialisation

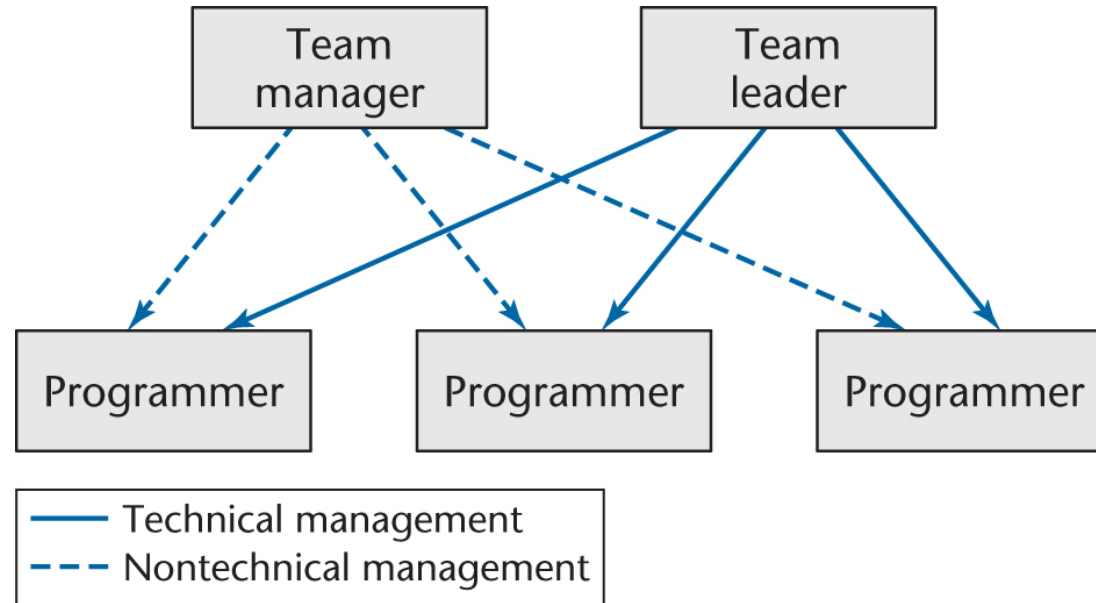


- 6 programmeurs, mais seulement 5 canaux de communication

Programmeur chef classique : irréaliste

- **Programmeur chef** doit être un programmeur hors norme ET un excellent gestionnaire
 - Très rare de trouver l'un ou l'autre
 - Qualités d'un excellent programmeur rarement présente chez un excellent gestionnaire, et vice-versa
- **Programmeur en réserve** doit être aussi bon que le chef et en connaitre autant sur le projet
 - Mais doit attendre dans l'ombre que quelque chose arrive au chef
 - Jamais un programmeur ou gestionnaire top ne ferait ça
- **Programmeur secrétaire** doit s'occuper de toute la documentation du projet, les **programmeurs** ne font que programmer
 - Tout ingénieur logiciel déteste la paperasse!

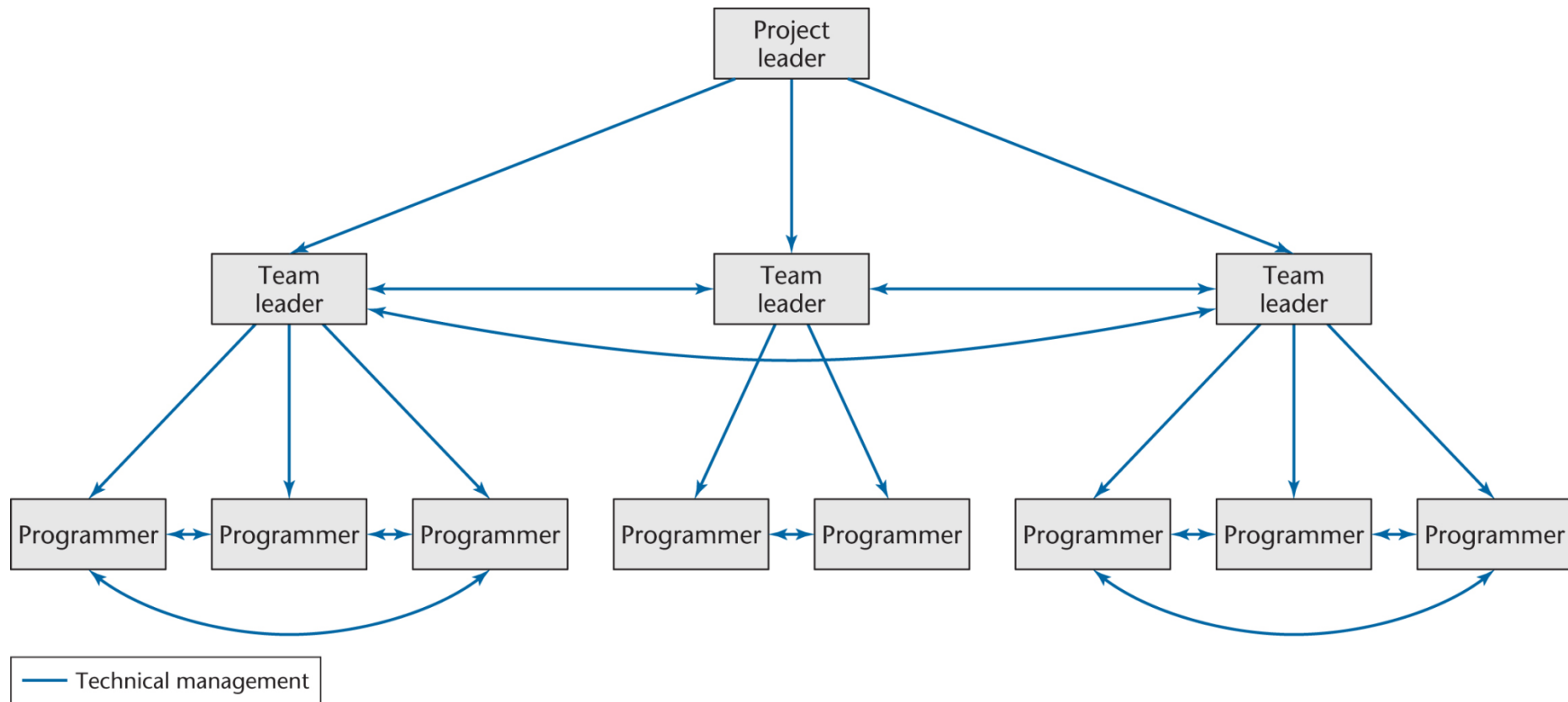
Séparation des pouvoirs



- Organisation qui combine les modèles démocratique et programmeur chef
 - Attitude positive
 - Réduit le rôle de gestion du programmeur chef

Pour les projets plus grand

Décentralisation des prises de décision



- La partie non-technique reste identique
- Ajouter plus de couche pour des projets encore plus grands

Équipe agile Scrum

- **Propriétaire du produit**

- Représentant des client et utilisateurs
- Seul qui dirige l'équipe de développement
- Responsable de création et maintien du carnet du produit

- **Scrum master**

- Leader au service de l'équipe
 - Entraîneur, facilitateur, très expérimenté
- S'assure du respect du processus

- **Équipe de développement**

- 7 ± 2 personnes
- Pluridisciplinaire pour avoir toutes les compétences nécessaires
- Une fois l'engagement d'un sprint pris, elle a pleine autorité
 - Travail à faire, heures d'ouvrage, responsabilités, propriété du code