

AIDE A LA CONCEPTION DE NOYAU APPLICATIF POUR LE LOGICIEL SYNDEX

QUENTIN QUADRAT

Stage de fin de tronc commun
1er Septembre au 31 Décembre 2005

—
Supervisé par Yves Sorel, Directeur de Recherche à l'INRIA



1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEx
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

Page 1 sur 20

Précéd.

Suivant



1. SOMMAIRE

- Structure d'accueil : INRIA, le projet AOSTE.
- Problématiques des systèmes temps réel embarqués.
- Méthodologie Adéquation Algorithme Architecture.
- Les exécutifs et les noyaux d'exécutifs.
- Le logiciel SynDEx.
- Présentation du travail : l'Éditeur de Code, création d'une application Automatique.

1.	<i>Sommaire</i>
2.	<i>INRIA, AOSTE</i>
3.	<i>Probl. STR</i>
4.	<i>Méthodo. AAA</i>
5.	<i>SynDEx</i>
6.	<i>Graphe...</i>
7.	<i>Graphe...</i>
8.	<i>Diagramme...</i>
9.	<i>Exécutif</i>
10.	<i>Noyau...</i>
11.	<i>Génération...</i>
12.	<i>Premier travail</i>
13.	<i>Editeur de code</i>
14.	<i>Exemple</i>
15.	<i>Deuxième...</i>
16.	<i>Diag. avant...</i>
17.	<i>Diag. après...</i>
18.	<i>Conclusion</i>
19.	<i>FIN</i>

2. INRIA, AOSTE

2.1. INRIA.

- Institut National de Recherche en Informatique et en Automatique (créé en 1967)
- Il est placé sous la double tutelle du ministre chargé de la Recherche et de l'Industrie.

2.2. Le projet AOSTE.

- Analysis Optimisation of Systems with real-Time and Embedded constraints,
- Méthodologie AAA (Adéquation Algorithme Architecture),
- *SynDEx* : logiciel de CAO pour l'aide à l'implantation de systèmes distribués temps réel embarqués.

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEx
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

3. PROBL. STR

Le temps est la problématique des systèmes temps réel embarqués :

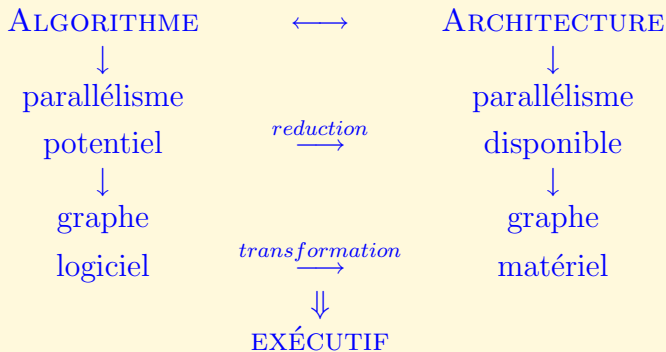
- Une opération doit pouvoir s'exécuter dans un temps prévu à l'avance.
- Réagir trop tard peut conduire à des conséquences catastrophiques pour le système lui-même ou son environnement.
- Il faut donc recourir à des calculateurs à architecture parallèle ou à des circuits intégrés spécialisés pour augmenter la puissance de calcul.

DONC, Il faut pouvoir affecter les tâches sur les différents processeurs et assurer la synchronisation de leur exécution.

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEX
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

4. MÉTHODO. AAA

ADÉQUATION



- Algorithme : graphe d'ordonnancement des tâches.
- Architecture : graphe des ressources et des moyens de communication.
- Exécutifs : affectation temporelle des tâches aux ressources.
- Adéquation : générer l'exécutif.

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEX
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

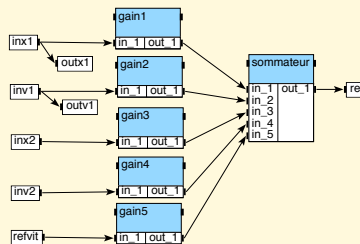
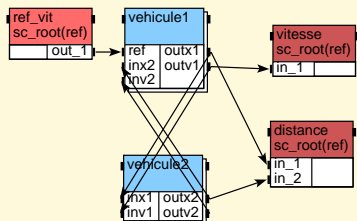
5. SYNDEX

- Met en oeuvre la méthodologie AAA.
- Entrée : graphe algorithme et graphe architecture.
- Sortie : les exécutifs distribués temps réel, sans inter-blocage et statiques écrits en macro-code m4.
- Visualise la prédiction des performances temps réel pour le dimensionnement de l'architecture.
- Utilise des heuristiques pour la distribution et l'ordonancement de l'algorithme d'application sur l'architecture.
- Des noyaux d'exécutifs génériques sont fournis pour : SHARC, i80C196, TMS320C40, i80386, MC68332 et stations de travail Unix ou Linux.

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEX
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

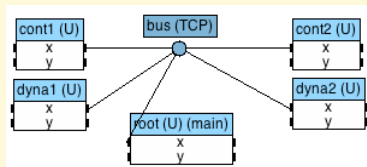
6. GRAPHE ALGORITHME

- Un graphe hiérarchique définit l'algorithme.
- Les noeuds sont les tâches (opérations) à réaliser et les arcs définissent les relations de précedence.



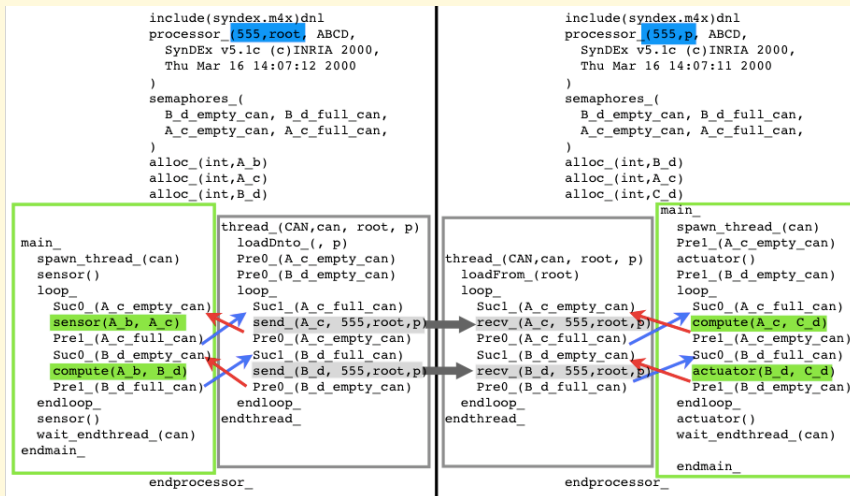
7. GRAPHE ARCHITECTURE

- Un graphe non hiérarchique définit l'architecture.
- Les noeuds sont les ressources (soit des opérateurs soit des médias de communication).
- Les arcs les communications entre les ressources.



1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEX
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

9. EXÉCUTIF



Les opérations et les synchronisations sont spécifiées en macro-code m4.

1. [Sommaire](#)
2. [INRIA, AOSTE](#)
3. [Probl. STR](#)
4. [Méthodo. AAA](#)
5. [SynDex](#)
6. [Graphe...](#)
7. [Graphe...](#)
8. [Diagramme...](#)
9. **Exécutif**
10. [Noyau...](#)
11. [Génération...](#)
12. [Premier travail](#)
13. [Editeur de code](#)
14. [Exemple](#)
15. [Deuxième...](#)
16. [Diag. avant...](#)
17. [Diag. après...](#)
18. [Conclusion](#)
19. [FIN](#)

Page 10 sur 20

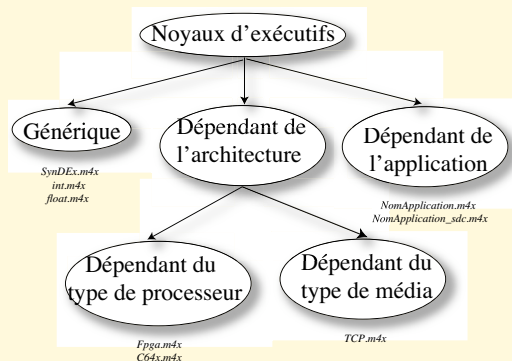
Précéd.

Suivant



10. NOYAU D'EXÉCUTIF

Noyau d'exécutif : ensemble de définitions des différentes macros utilisées par les exécutifs, dans un langage haut niveau (C, ...) ou assembleur.



Travail du stage a concerné la génération de noyau applicatif.

11. GÉNÉRATION DE L'EXÉCUTABLE

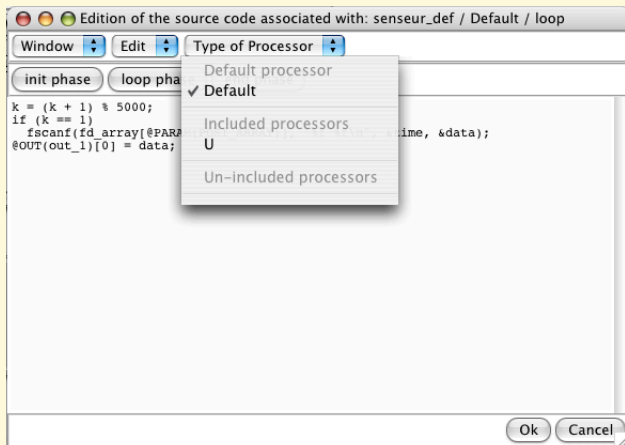
Macro-processeur M4 : Exécutifs \times Noyaux d'exécutifs \rightarrow
Codes sources (C)

Compilateur (GCC) : Codes sources (C) \rightarrow Exécutables

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEX
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

12. PREMIER TRAVAIL

Développer un éditeur de noyau applicatif *Éditeur de Code* dans l'IHM de SynDEx.



1. [Sommaire](#)
2. [INRIA, AOSTE](#)
3. [Probl. STR](#)
4. [Méthodo. AAA](#)
5. [SynDEx](#)
6. [Graphe...](#)
7. [Graphe...](#)
8. [Diagramme...](#)
9. [Exécutif](#)
10. [Noyau...](#)
11. [Génération...](#)
12. [Premier travail](#)
13. [Editeur de code](#)
14. [Exemple](#)
15. [Deuxième...](#)
16. [Diag. avant...](#)
17. [Diag. après...](#)
18. [Conclusion](#)
19. [FIN](#)

13. EDITEUR DE CODE

Avant le stage, un utilisateur devait spécifier manuellement le noyau applicatif en en terme de macros instructions m4.

Après le stage, un utilisateur édite le code de chaque opérations en langage naturel.

L'Éditeur de Code :

- Génère les macros m4 correspondantes.
- Automatise la traduction des noms de ports en arguments m4.
- Complète les noms de ports.
- Communique avec Emacs.

1.	<i>Sommaire</i>
2.	<i>INRIA, AOSTE</i>
3.	<i>Probl. STR</i>
4.	<i>Méthodo. AAA</i>
5.	<i>SynDEx</i>
6.	<i>Graphe...</i>
7.	<i>Graphe...</i>
8.	<i>Diagramme...</i>
9.	<i>Exécutif</i>
10.	<i>Noyau...</i>
11.	<i>Génération...</i>
12.	<i>Premier travail</i>
13.	<i>Editeur de code</i>
14.	<i>Exemple</i>
15.	<i>Deuxième...</i>
16.	<i>Diag. avant...</i>
17.	<i>Diag. après...</i>
18.	<i>Conclusion</i>
19.	<i>FIN</i>

14. EXEMPLE

- Le code que l'on devait écrire dans un fichier:

```
define('foo','ifelse(  
    processorType_,processorType_,'ifelse(  
        MGC,'INIT','''',  
        MGC,'LOOP',''$3[0] = $2[0] + $1;''',  
        MGC,'END','''')''')
```

- s'écrit dans l'Éditeur de Code :

```
@OUT(out)[0] = @IN(in)[0] + @PARAM(P);
```

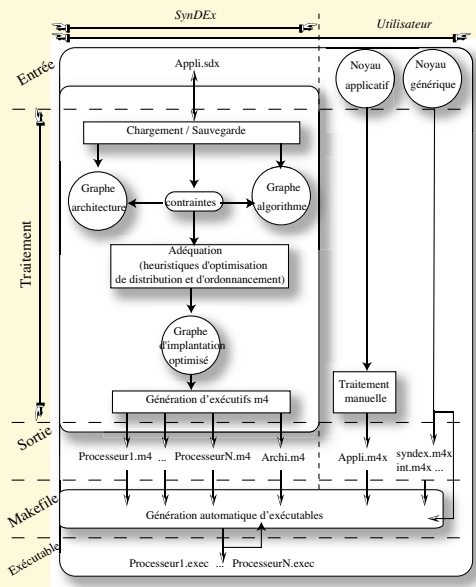
1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEx
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

15. DEUXIÈME TRAVAIL

- Documenter l'Éditeur de Code.
- Créer une application d'automatique où une voiture suit une autre et maintient une distance fixe alors que la première choisit une vitesse.
- Apprentissage : introduction à l'automatique, logiciels Scilab et Scicos
- Simulation : Scicos et SynDEx.

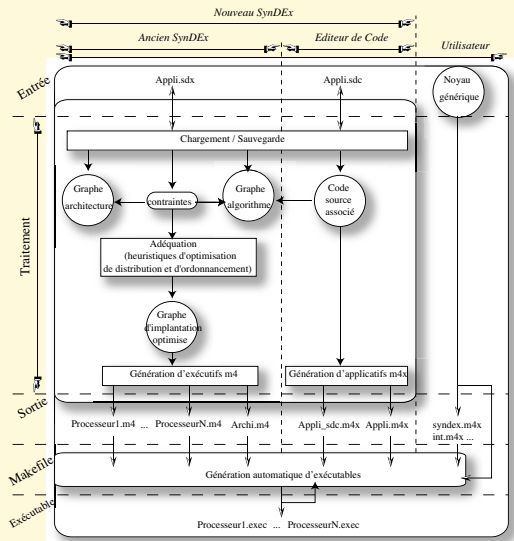
1.	<i>Sommaire</i>
2.	<i>INRIA, AOSTE</i>
3.	<i>Probl. STR</i>
4.	<i>Méthodo. AAA</i>
5.	<i>SynDEx</i>
6.	<i>Graphe...</i>
7.	<i>Graphe...</i>
8.	<i>Diagramme...</i>
9.	<i>Exécutif</i>
10.	<i>Noyau...</i>
11.	<i>Génération...</i>
12.	<i>Premier travail</i>
13.	<i>Editeur de code</i>
14.	<i>Exemple</i>
15.	<i>Deuxième...</i>
16.	<i>Diag. avant...</i>
17.	<i>Diag. après...</i>
18.	<i>Conclusion</i>
19.	<i>FIN</i>

16. DIAG. AVANT LE STAGE



1. [Sommaire](#)
2. [INRIA, AOSTE](#)
3. [Probl. STR](#)
4. [Méthodo. AAA](#)
5. [SynDEx](#)
6. [Graphe...](#)
7. [Graphe...](#)
8. [Diagramme...](#)
9. [Exécutif](#)
10. [Noyau...](#)
11. [Génération...](#)
12. [Premier travail](#)
13. [Editeur de code](#)
14. [Exemple](#)
15. [Deuxième...](#)
16. [Diag. avant...](#)
17. [Diag. après...](#)
18. [Conclusion](#)
19. [FIN](#)

17. DIAG. APRÈS LE STAGE



1. [Sommaire](#)
2. [INRIA, AOSTE](#)
3. [Probl. STR](#)
4. [Méthodo. AAA](#)
5. [SynDEx](#)
6. [Graphe ...](#)
7. [Graphe ...](#)
8. [Diagramme ...](#)
9. [Exécutif](#)
10. [Noyau ...](#)
11. [Génération ...](#)
12. [Premier travail](#)
13. [Editeur de code](#)
14. [Exemple](#)
15. [Deuxième ...](#)
16. [Diag. avant ...](#)
17. [Diag. après ...](#)
18. [Conclusion](#)
19. [FIN](#)

18. CONCLUSION

18.1. Appréciations des utilisateurs de l'Éditeur de Code.

Ce travail était une demande par l'un (MBDA) des principaux utilisateur industriel de SynDEx dont les commentaires après utilisation sont positifs.

18.2. Expérience acquise. Ce stage m'a permis de :

- Découvrir les problèmes d'affectation pour réaliser des systèmes temps réel.
- Découvrir des notions en automatique.
- De manipuler les logiciels SynDEx, Scilab, Scicos,
- De progresser dans la connaissance du langage OCaml.

1.	Sommaire
2.	INRIA, AOSTE
3.	Probl. STR
4.	Méthodo. AAA
5.	SynDEx
6.	Graphe...
7.	Graphe...
8.	Diagramme...
9.	Exécutif
10.	Noyau...
11.	Génération...
12.	Premier travail
13.	Editeur de code
14.	Exemple
15.	Deuxième...
16.	Diag. avant...
17.	Diag. après...
18.	Conclusion
19.	FIN

19. FIN

Questions ?

1. *Sommaire*
2. *INRIA, AOSTE*
3. *Probl. STR*
4. *Méthodo. AAA*
5. *SynDEX*
6. *Graphe...*
7. *Graphe...*
8. *Diagramme...*
9. *Exécutif*
10. *Noyau...*
11. *Génération...*
12. *Premier travail*
13. *Editeur de code*
14. *Exemple*
15. *Deuxième...*
16. *Diag. avant...*
17. *Diag. après...*
18. *Conclusion*
19. *FIN*

Page 20 sur 20

Précéd.

Suivant

