Echange INRIA-STMicroelectronics

Analyse de l'outil KPTrace (T-Charts, Outline View), et réalisation d'un prototype au sein de T-Charts Lite

Damien DOSIMONT

23 mars 2012

INVENTEURS DU MONDE NUMÉRIQUE

Sommaire

Objectif de l'échange

- Prise en main des outils de STMicroelectronics KPTrace (+ BTrace) -> T-Charts et OutlineView
- Confrontation à des traces issues de systèmes embarqués
- Analyse de la pertinence des outils
- Suggestions d'améliorations pour faciliter l'analyse
- Réalisation d'une preuve de concept : implémenter une vue agrégée de la trace au sein de T-Charts Lite

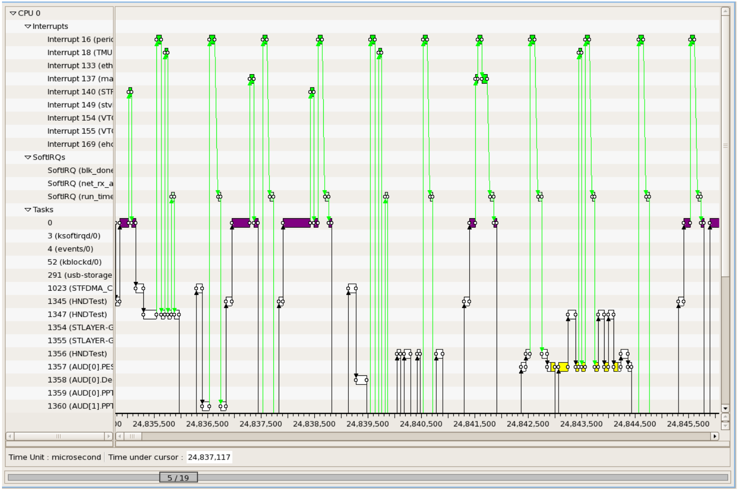
KPTrace

- Framework analyse de trace
- Plug-in Eclipse (Java)
- Différents modules :
    - Traitement de la trace : parsing, intégration des données au sein d'une base de donnée + méthodes d'accès aux données
    - Visualisation : T-Charts (et bientôt T-Charts Lite), Outline View, Statistiques, etc.

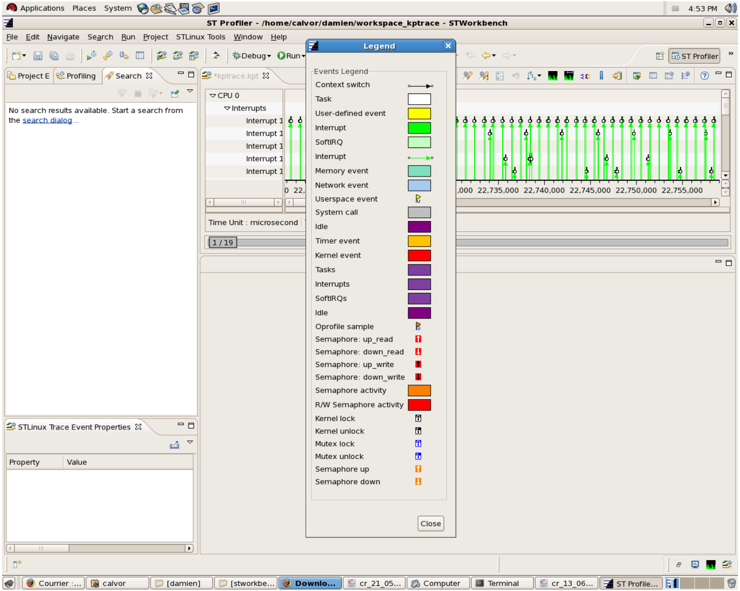
T-Charts : principe

- Représentation de type “Gantt Chart” : axe x = temps, axe y = hiérarchie
- Hiérarchie = conteneurs : Fonctions $\subset$ Tâches, Tâches $\subset$ CPU, Interruptions $\subset$ CPU, Soft IRQ $\subset$ CPU
- Evenements
  - Etats : Rectangles couleur selon leur type
  - Evenements ponctuels (sémaphores, mutex, etc.) = symboles
  - Commutation : Flèches
  - Fonctions et évenements kernel + possibilité de tracer user-space

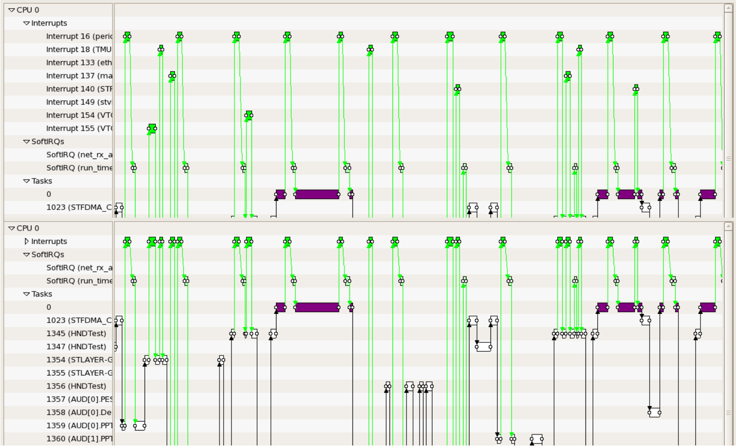
T-Charts


CPU 0
Interrupts
Interrupt 16 (peri
Interrupt 18 (TMU
Interrupt 133 (eth
Interrupt 137 (ma
Interrupt 140 (STF
Interrupt 149 (stv
Interrupt 154 (VT
Interrupt 155 (VT
Interrupt 169 (ehc
SoftIRQs
SoftIRQ (blk_done
SoftIRQ (net_rx_a
SoftIRQ (run_time
Tasks
0
3 (ksoftirqd/0)
4 (events/0)
52 (kblockd/0)
291 (usb-storage
1023 (STFDMA_C
1345 (HNDTest)
1347 (HNDTest)
1354 (STLAYER-G
1355 (STLAYER-G
1356 (HNDTest)
1357 (AUD[0].PES
1358 (AUD[0].De
1359 (AUD[0].PPT
1360 (AUD[1].PPT
24,835,500
24,836,500
24,837,500
24,838,500
24,839,500
24,840,500
24,841,500
24,842,500
24,843,500
24,844,500
24,845,500
Time Unit : microsecond
Time under cursor : 24,837,117
5 / 19

T-Charts : Légendes

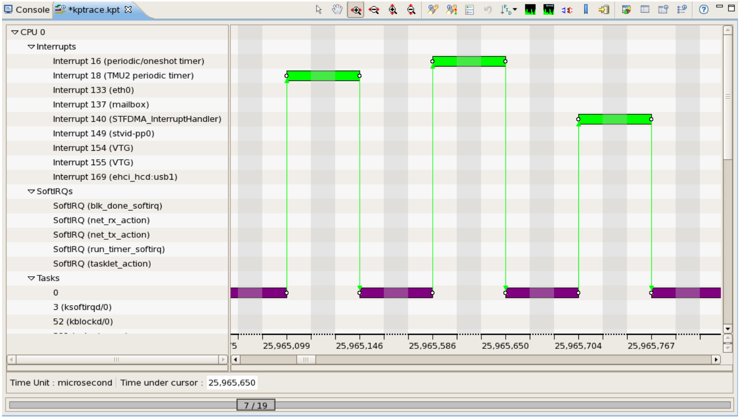


T-Charts : Agrégation hiérarchique



T-Charts : Gestion de l'axe du temps

- Compression du temps

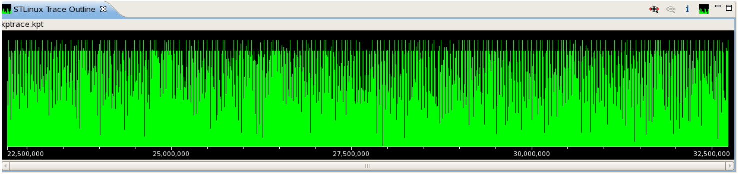


- Découpage de la trace : 20000 événements par page

OutlineView

- Représentation globale de la trace au niveau temporel (pas de découpage par pages)
- Métriques :
  - Charge CPU (inverse de idle)
  - Activity Time :% de temps passé dans un état par unité de temps
  - RateFrequency : nombre d'occurrence d'évenements par unité de temps
  - Délai entre deux évenements : $\sum$ des délais sur une tranche de temps
  - Lock held time : temps passé à l'état bloqué
  - Advanced queries : évolution valeur d'un $ (CPU, start time, end time), valeur d'un $ (idem), taux d'activité, d'inactivité, nombre d'évenements...
- Plot en 2D.
- Agrégation : tranche de temps = largeur de 1 pixel en abscisse.

OutlineView

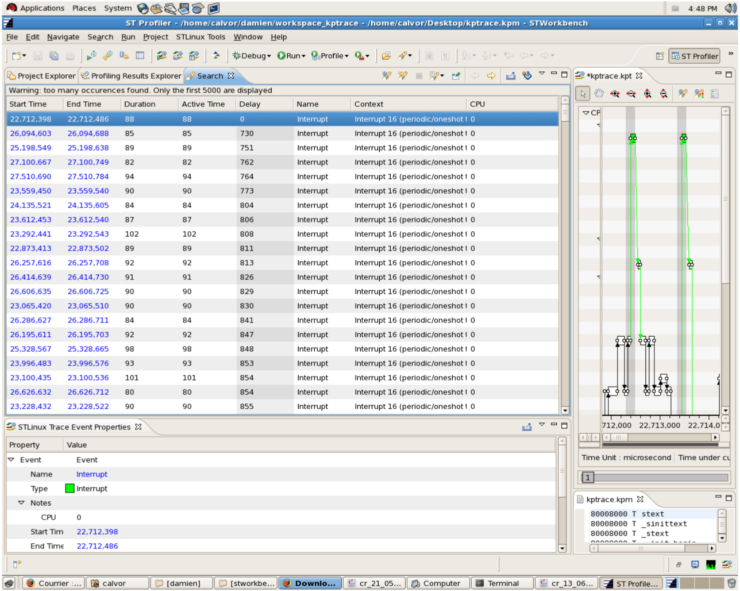


Statistiques

Time unit is us

Context	Total run time	% Total run tim	Fired/Time switched to	Min run time	Max run time	Avg run time	Min interval	Max interval	Avg interval
▽ Summary	9,473,219	100%	70,037	36	1,654	135	109	5,007,183	6,247
▷ Idle	895,779	9.5%	1,767	100	1,654	506	333	43,157	5,644
▷ Interrupts	1,404,533	14.8%	17,195	43	279	81	118	1,204,092	5,078
▽ SoftIRQs	634,341	6.7%	10,280	36	506	61	168	1,574,815	3,054
SoftIR	3,874	0%	16	129	363	242	7,986	1,206,969	533,376
SoftIR	36,528	0.4%	207	36	506	176	924	285,598	47,770
SoftIR	439	0%	9	45	54	48	11,134	1,574,815	222,034
SoftIR	590,952	6.2%	10,006	41	309	59	322	1,564	1,000
SoftIR	2,548	0%	42	37	260	60	168	1,574,743	43,334
SoftIR	0	0%	0	0	0	0	0	0	0
▷ Tasks	6,538,566	69%	40,795	48	1,604	160	109	5,007,183	7,570

Search Tool


Applications Places System
4:48 PM
ST Profiler - /home/calvor/damien/workspace_kptrace - /home/calvor/Desktop/kptrace.kpm - STWorkbench
File Edit Navigate Search Run Project STLinux Tools Window Help
Debug Run Profile
ST Profiler
Project Explorer
Profiling Results Explorer
Search
Warning: too many occurences found. Only the first 5000 are displayed
Start Time End Time Duration Active Time Delay Name Context CPU
22,712,398 22,712,486 88 88 0 Interrupt Interrupt 16 (periodic/oneshot t 0
26,094,603 26,094,688 85 85 730 Interrupt Interrupt 16 (periodic/oneshot t 0
25,198,549 25,198,638 89 89 751 Interrupt Interrupt 16 (periodic/oneshot t 0
27,100,667 27,100,749 82 82 762 Interrupt Interrupt 16 (periodic/oneshot t 0
27,510,690 27,510,784 94 94 764 Interrupt Interrupt 16 (periodic/oneshot t 0
23,559,450 23,559,540 90 90 773 Interrupt Interrupt 16 (periodic/oneshot t 0
24,135,521 24,135,605 84 84 804 Interrupt Interrupt 16 (periodic/oneshot t 0
23,612,453 23,612,540 87 87 806 Interrupt Interrupt 16 (periodic/oneshot t 0
23,292,441 23,292,543 102 102 808 Interrupt Interrupt 16 (periodic/oneshot t 0
22,873,413 22,873,502 89 89 811 Interrupt Interrupt 16 (periodic/oneshot t 0
26,257,616 26,257,708 92 92 813 Interrupt Interrupt 16 (periodic/oneshot t 0
26,414,639 26,414,730 91 91 826 Interrupt Interrupt 16 (periodic/oneshot t 0
26,606,635 26,606,725 90 90 829 Interrupt Interrupt 16 (periodic/oneshot t 0
23,065,420 23,065,510 90 90 830 Interrupt Interrupt 16 (periodic/oneshot t 0
26,286,627 26,286,711 84 84 841 Interrupt Interrupt 16 (periodic/oneshot t 0
26,195,611 26,195,703 92 92 847 Interrupt Interrupt 16 (periodic/oneshot t 0
25,328,567 25,328,665 98 98 848 Interrupt Interrupt 16 (periodic/oneshot t 0
23,996,483 23,996,576 93 93 853 Interrupt Interrupt 16 (periodic/oneshot t 0
23,100,435 23,100,536 101 101 854 Interrupt Interrupt 16 (periodic/oneshot t 0
26,626,632 26,626,712 80 80 854 Interrupt Interrupt 16 (periodic/oneshot t 0
23,228,432 23,228,522 90 90 855 Interrupt Interrupt 16 (periodic/oneshot t 0
STLinux Trace Event Properties
Property Value
Event Event
Name Interrupt
Type Interrupt
Notes
CPU 0
Start Tim 22,712,398
End Time 22,712,486
*kptrace.kpt
712,000 22,713,000 22,714,0
Time Unit : microsecond Time under cu
kptrace.kpm
80008000 T stext
80008000 T _sinittext
80008000 T _stext
Courrier :... calvor [damien] [stworkbe... Downlo... cr_21_05... Computer Terminal cr_13_06... ST Profile...

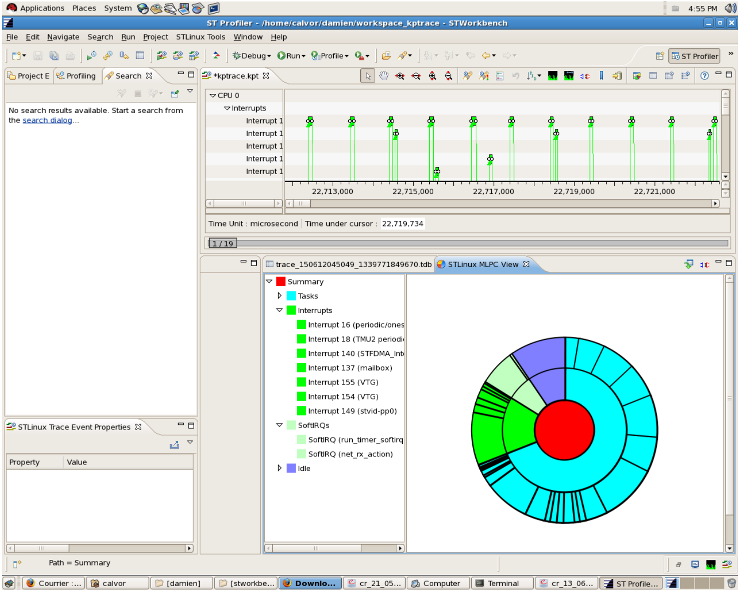
- Permet de rechercher des patterns dans la trace (succession de deux événements)
  - Différentes stats ou graphes : liste des patterns, événements ignorés, graphes (occurences, délai, durée)

Event pattern search result

Summary | Event patterns | Ignored events | Event patterns + Ignored events | Graph: Occurence of patterns | Graph: Delay between 2 patterns | Graph: Pattern duration

Element	Value
Session	/home/calvor/damien/KP_Traces/0026/kptrace.kpt
Events name	1412 (HostRec40800003) 0
Duration	min = 156 avg = 1539 max = 17882
Delay	min = 0 avg = 23934 max = 49035
number of event patterns	416
number of 1st event ignored	1231
number of 2nd event ignored	1351

Pie Chart



ST Profiler - /home/calvor/damien/workspace_kptrace - STWorkbench
File Edit Navigate Search Run Project STLinux Tools Window Help
No search results available. Start a search from the search dialog...
*kptrace.kpt
CPU 0
Interrupts
22,713,000
22,715,000
22,717,000
22,719,000
22,721,000
Time Unit : microsecond
Time under cursor : 22,719,734
1 / 19
STLinux Trace Event Properties
Property
Value
trace_150612045049_1339771849670.tdb
STLinux MLPC View
Summary
Tasks
Interrupts
Interrupt 16 (periodic/ones
Interrupt 18 (TMU2 periodi
Interrupt 140 (STFDMA_Int
Interrupt 137 (mailbox)
Interrupt 155 (VTG)
Interrupt 154 (VTG)
Interrupt 149 (stvid-pp0)
SoftIRQs
SoftIRQ (run_timer_softirq
SoftIRQ (net_rx_action)
Idle
Path = Summary

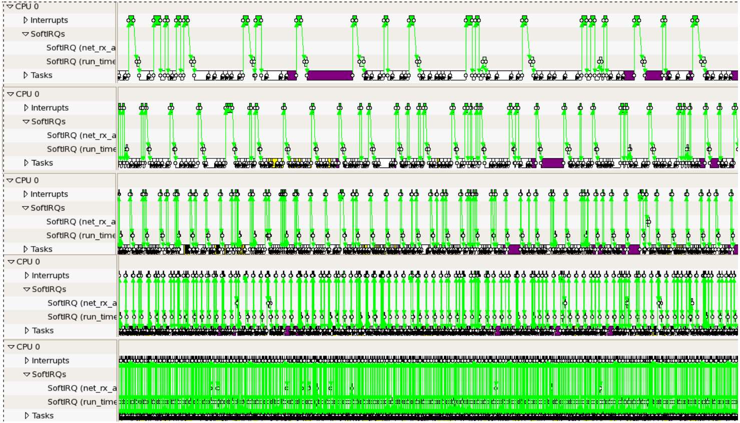
Cas d’études

- Certain nombre de traces fournies
  - Traces non reliées à des cas d’études précis
    - Prendre en main KPTrace
    - Analyser le rendu graphique
    - Dégager une “physionomie des traces” (structure hiérarchique, types d’événements, etc.)
  - Quelques traces avec problèmes de performances déterminés (applications multimédia)
    - Ré-effectuer l’analyse
    - Tenter différentes approches et déterminer la méthodologie la plus efficace

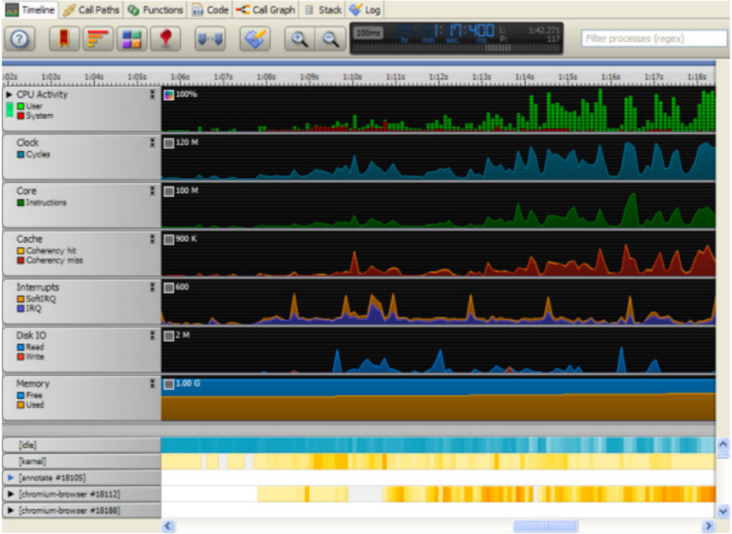
Représentation lors d'un fort dézoom

- Lors d'un dézoom : pas d'agrégation temporelle
- Informations représentées deviennent illisibles
- Fidélité de l'information retransmise ?
  - Chaque objet (ex : flèches, event) est toujours représenté même à zoom out max (1 pxl minimum)
  - Espace entre ces objets se réduit
  - Proportions ne sont plus exactes
- Solutions ? Agrégation
  - Barre oblique pour indiquer un état agrégé
  - Agrégation mathématique : vue synthétique avec agrégation des données (cf. Streamline d'ARM DS5)

Différent niveaux de dézoom



Agrégation dans la Streamline de ARM DS5



Inria
INVENTEURS DU MONDE NUMÉRIQUE

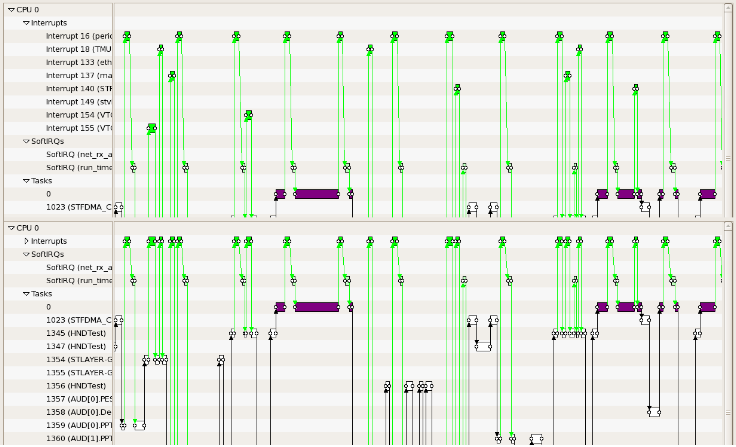
Inconvénients dûs à la pagination

- Pagination : affiche 20000 événements par page
- Avantages :
  - Permet de minimiser intéractions avec la base de donnée
  - Permet d'économiser de la mémoire et d'augmenter la fluidité lors du déplacement dans le Gantt
  - Permet à l'utilisateur de scroller plus facilement
- Inconvénients :
  - Pas de possibilité d'avoir un rendu intégral de la trace
  - -> Difficulté à analyser un comportement macroscopique dont la durée serait supérieure à une page

Agrégation hiérarchique

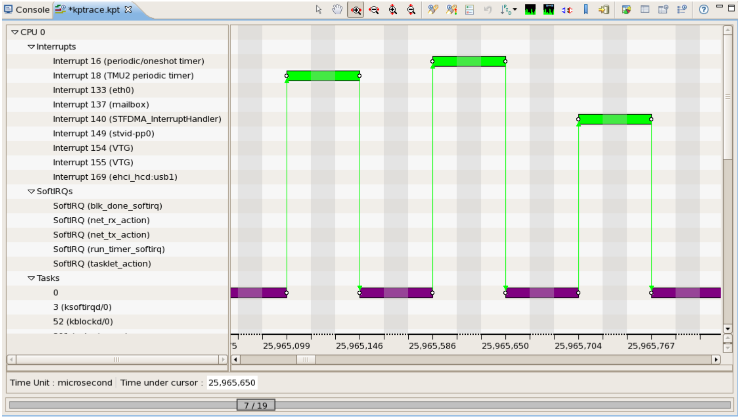
- Avantages :
  - Permet de représenter sur la ligne du conteneur père l'ensemble des conteneurs fils : vue synthétique
- Inconvénients :
  - Si les fils possèdent des évenements du même type (ex : interruptions), alors pas de possibilité de les discriminer selon la couleur
- Solutions :
  - Utilisation d'un discriminant : hybridation des couleurs (hachures), textures, personnalisation des couleurs, bordure de couleur différente...

Agrégation hiérarchique



Compression temporelle

- Avantages :
  - Permet de ne pas afficher l'intégralité d'un event qui aurait une durée très longue
  - Rend la trace plus lisible
- Inconvénients :
  - Ce procédé peut être trompeur : les durées compressées ne sont pas forcément de la même taille
- Solutions :
  - Ajouter la possibilité de désactiver cette fonction
  - Utiliser un gradient de gris pour différencier la durée des périodes (plus foncé = +compressé)


Console
*kptrace.kpt
CPU 0
Interrupts
Interrupt 16 (periodic/oneshot timer)
Interrupt 18 (TMU2 periodic timer)
Interrupt 133 (eth0)
Interrupt 137 (mailbox)
Interrupt 140 (STFDMA_InterruptHandler)
Interrupt 149 (stvid-pp0)
Interrupt 154 (VTG)
Interrupt 155 (VTG)
Interrupt 169 (ehci_hcd:usb1)
SoftIRQs
SoftIRQ (blk_done_softirq)
SoftIRQ (net_rx_action)
SoftIRQ (net_tx_action)
SoftIRQ (run_timer_softirq)
SoftIRQ (tasklet_action)
Tasks
0
3 (ksoftirqd/0)
52 (kblockd/0)
25,965,099
25,965,146
25,965,586
25,965,650
25,965,704
25,965,767
Time Unit : microsecond
Time under cursor : 25,965,650
7 / 19

OutlineView

- Avantages :
  - Pas de limitation de pages (timeline entière)
  - Renseignements statistiques (CPU load, % activité, rate, délai)
  - Possibilité de “recherches avancées”
- Inconvénients :
  - Focalisé sur un aspect statistique précis
  - S’intègre difficilement dans la méthodologie “Overview, zoom, details on demand”
- Solutions :
  - Meilleure intégration au sein de KPTrace
  - Meilleures intéractions avec T-Charts
  - Accentuer les capacités d’overview de l’OutlineView

Résultats sur les cas d'études

- Trouver bugs application multimédia (problème de fps)
- Trois méthodolgies possibles
  - Commencer par l'analyse de la vue T-Charts
    - Trop d'informations
  - Utiliser l'OutlineView
    - Trop focalisée sur un aspect précis de la trace, impossible de savoir quels types d'événements analyser
  - Outil Statistiques
    - Le plus pertinent : permet de voir facilement les % activité, les valeurs min et max et average pour durée et délai.

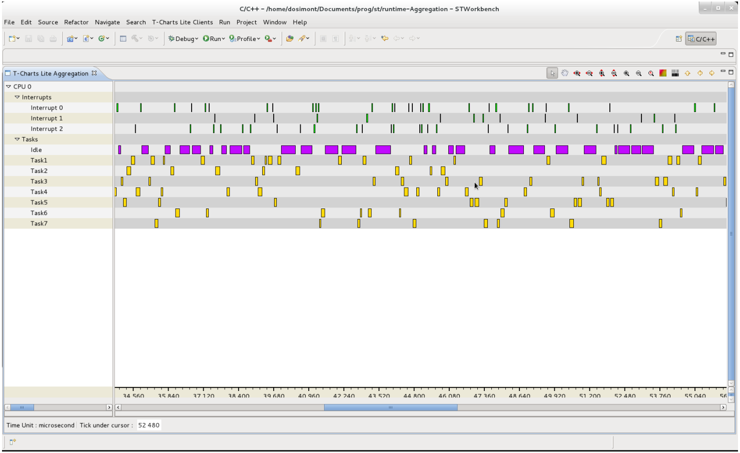
Prototypes : présentation générale

- Travail au sein de l'environnement T-Charts Lite
- Représentation de la trace entière sous forme agrégée
  - Découper la trace en "tranches de temps"
  - Effectuer l'agrégation sur chaque tranche de temps
  - Agrégation hiérarchique : gestion des collapse et expand
  - Possibilité de modifier la tranche de temps de manière dynamique
  - Gestion du zoom et dézoom graphique
  - Possibilité d'"ajuster" la représentation pour amplifier certains comportements
- Plusieurs types d'agrégations : taux d'activité, densité (nombre d'occurrences), charge CPU avec détail des tâches
- Preuve de concept : modèle limité -> utilisation d'une trace générée de manière aléatoire

Génération d'une trace aléatoire

- Permet de simplifier la conception du prototype
- Utilisation d'une hiérarchie simple mais représentative d'un système embarqué :
  - 1 CPU :
    - 3 interruptions
    - 6 tâches
    - tâche idle
- Génération des événements suivant une probabilité d'apparaître (différente selon les types) et une probabilité de durée (durée comprise dans un intervalle de temps)
  - Conséquence : distribution assez régulière des événements
- Environ 1000 événements < 20000 événements par page (soit pour une trace KPTrace complète entre 100000 et 1000000)
  - Limité par la mémoire...

Trace aléatoire



Représentation agrégée : taux d'activité

- Principe : % activité du conteneur par tranche de temps
- Tranche de temps : modifiable dynamiquement -> recalcule les valeur des tranches de temps
- Agrégation hiérarchique : lorsqu'on collapse/expand
- Zoom in / zoom out
- Mécanisme d'"ajustement"
- Trois implémentations :
  - Couleur
  - Couleur + amplitude selon y
  - Couleur + amplitude selon y + lissage

Taux d'activité : algorithme

- Trace principale :
  - SampleMaster : intégralité de la structure hiérarchique
  - SampleItem : conteneurs
  - SampleEvent : événements (marqueurs + figure)
- Récupération du SampleMaster
- Reconstruction de la hiérarchie
- Lecture de chaque conteneur
- Pour chaque évenement du conteneur, on met à jour la valeur de la (ou des) tranche(s) de temps correspondantes

$$value_{Ti->Tf} = \frac{\sum_{j=0}^{n}(min(Tf, tf_{event_j}) - max(Ti, ti_{event_j}))}{Tf - Ti}$$

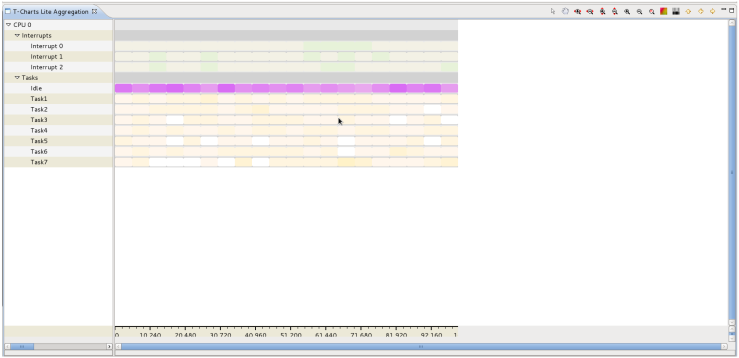
- Pour chaque tranche de temps, on créé un évenement (marqueurs temporels + figure associée, dépendant de la valeur de la tranche de temps et de l'ajustement)

Taux d’activité : ajustement

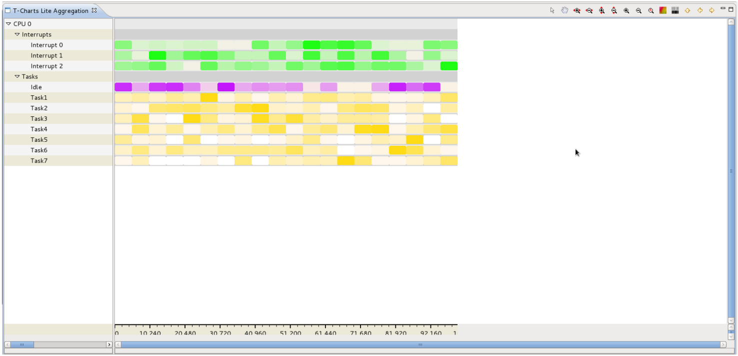
- Si les valeurs des tranches de temps sont proches en amplitude, difficulté pour les discriminer
- Ajustement : représenter les tranches de temps en introduisant un ratio dépendant de la tranche min et de la tranche max
- Homothétie, de $[0; Tf - Ti]$ vers $[value_{min}; value_{max}]$
  Si $value_{Ti->Tf} > 0$
  -> $value_{adj_{Ti->Tf}} = (value_{Ti->Tf} - Tmin) * \frac{Tf-Ti}{value_{max}-value_{min}}$
  Sinon
  -> $value_{adj_{Ti->Tf}} = 0$

- Utilisation d’un gradient de couleur pour représenter l’amplitude
- Programmeur fourni un tableau contenant un panel de couleurs (couleur pour min, max, et éventuellement des couleurs intermédiaires) pour différents types d’event
- Le programme génère un gradient (par défaut 40 couleurs)
- Lors de la génération d’un “événement agrégé”, on génère un rectangle de largeur $Tf - Ti$ et de hauteur fixe, de couleur
  si $value_{Ti-Tf} > 0$,
  -> $color = colorgradient[(colornumber - 1) * value_{Ti->Tf}]$,
  $color = blanc$ sinon
- Dans le cas d’une représentation ajustée, on utilise $value_{adj_{Ti->Tf}}$

Taux d’activité sans ajustement



Taux d'activité avec ajustement



taux d'activité : implémentation 2 - couleurs + y

- Même principe, mais en plus d'afficher la couleur, on module la hauteur du rectangle en fonction de $value_{Ti->Tf}$ ou $value_{adj_{Ti->Tf}}$
- On rajoute un offset sur la hauteur pour afficher explicitement la valeur zéro (rectangle blanc) et mieux représenter les valeurs faibles
- Bénéfices : plus de facilité pour discerner les amplitudes qu'avec les couleurs
- Inconvénients :
  - les couleurs ont un rôle presque uniquement esthétique car redondance d'information
  - en raison de la taille réduite de certains rectangle, ne laisse pas la possibilité de coder deux informations différentes

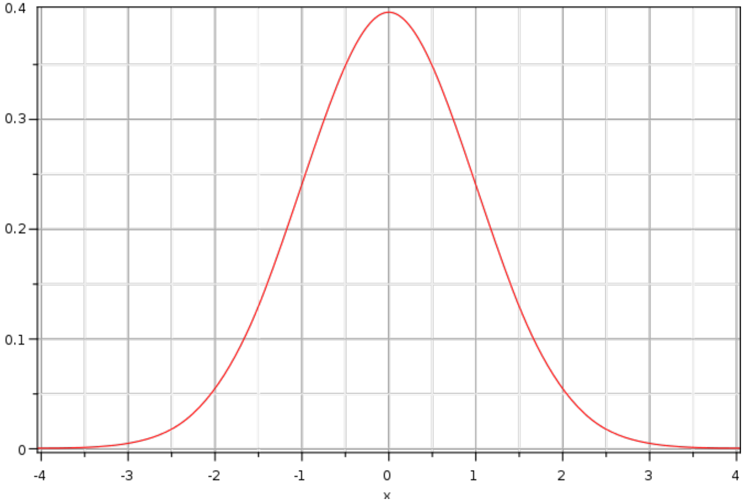
Taux d’activité sans ajustement

Taux d'activité avec ajustement

taux d'activité : implémentation 3 - couleurs + y + lissage

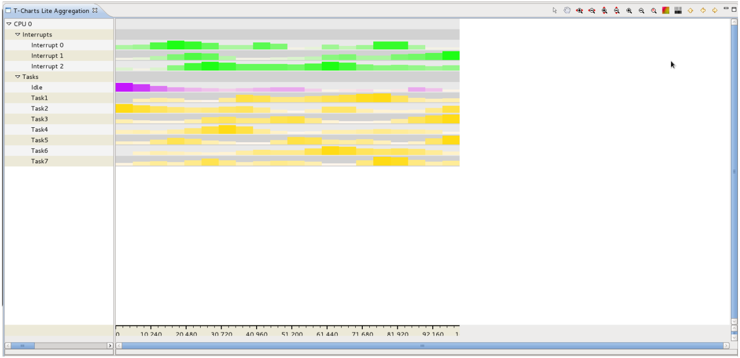
- Même principe, mais agrégation subit un lissage pour éviter les changements brutaux d'amplitude lors d'un “resize” dus à une nouvelle configuration des events dans la tranche de temps
- Utilisation des valeurs des tranches contigües qui participent au calcul de l'agrégation de la tranche en cours
- Moyenne pondérée entre les différentes tranches selon des coefficients issus d'une distribution de Gauss
- Exemple : Pour $coefficients = \{0.05, 0.25, 0.4, 0.25, 0.05\}$ (approximation d'une courbe de Gauss pour 5 points) : -> $value_{gauss_{Tn}} = \sum_{i=n-2}^{n+2} (value_{Ti} * coefficients[i-n+2])$
- Si les tranches contiguës sont hors de l'index, on ne les prend pas en compte et -> $value_{gauss_{Tn}} = value_{gauss_{Tn}} * (1 - \sum_{horsindex}(coefficients[horsindex]))$

Courbe de Gauss



Taux d’activité sans ajustement

Taux d’activité avec ajustement



Représentation agrégée : densité d'événements

- Mêmes fonctionnalités que taux d'activité mais agrégation différente : nombre d'événements par tranche de temps
  $value_{Ti->Tf} = \frac{\sum_{i=0}^{n}(event_i)}{\sum_{max}}$
- Version ajustée
  $value_{adj_{Ti->Tf}} = (value_{Ti->Tf} - \sum_{min}) * (\frac{\sum_{max}}{\sum_{max} - \sum_{min}})$
- Version couleur et couleur + y, mais pas d'implémentation de version lissée

Densité sans ajustement (couleur)

Densité avec ajustement (couleur)

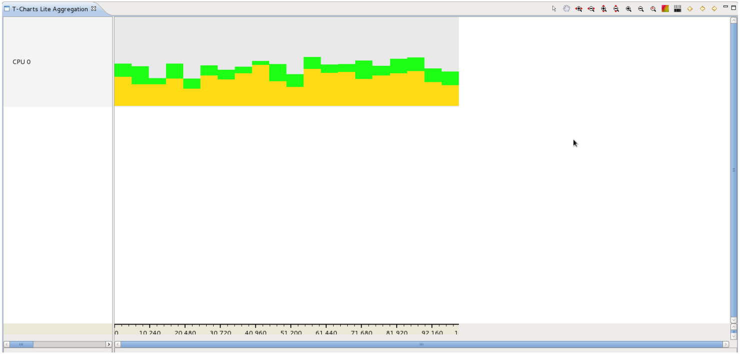
Densité sans ajustement (y)

Densité avec ajustement (y)

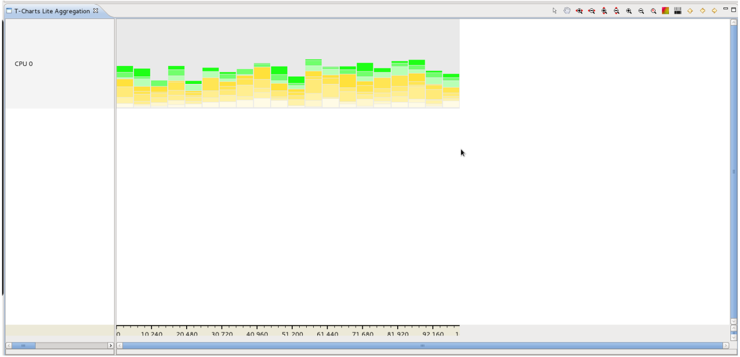
Graphe d'utilisation du CPU

- Fontionne de la même manière que l'agrégation “taux d'activité”, mais 1 seul niveau hiérarchique : CPU
- Tous les événements des conteneurs fils sont empilés dans la même représentation
- Séparation par couleur :
  - Mode non ajusté : uniquement les tâches, les interruptions (jaune, vert)
  - Mode ajusté : détail des tâches et des interruptions (gradient de jaunes, de verts)
- Implémentations : couleur + y, couleur + y + lissage

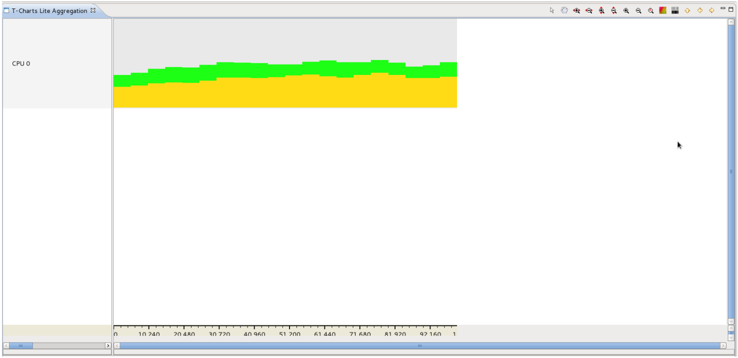
CPU sans ajustement (y)



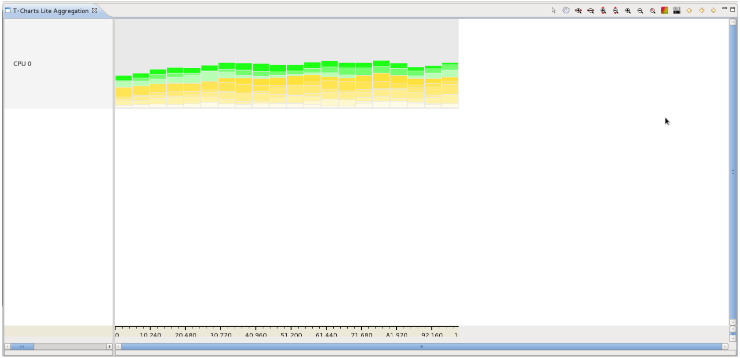
CPU avec ajustement (y)



CPU sans ajustement (lissage)



CPU avec ajustement (lissage)



Conclusion

- Preuve de concept de plusieurs mécanismes d'agrégation (algo mathématiques, visualisation)
- Mais pas de vraie trace pour confirmer l'intérêt de ce prototype
- Trace générée (1000 events) < vraie trace (100000 à 1000000)
- Algo : coûteux car recalculent tout à chaque changement de taille de tranche de temps
- Nécessité d'introduire intéractivité + transition entre trace et état agrégé
- Etats agrégés : pourraient contenir des informations accessibles par double-clique par exemple
- Attention aux artefacts dus au redimensionnement des tranches de temps