Ocelotl: Time Aggregation Visualization for Trace Overview

Damien Dosimont [1] Guillaume Huard [2] Jean-Marc Vincent [2]

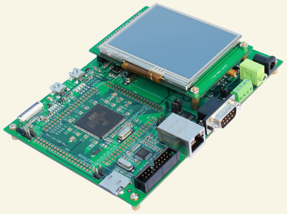
[1] INRIA

[2] Grenoble University

firstname.lastname@imag.fr

LICIA 3rd Workshop (October 22nd 2013)

Context: SoC-Trace project



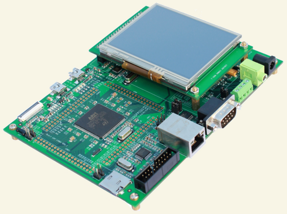
Embedded system trace analysis problematic

- **Hardware** and **software** complexity
- Trace **size** and **format** management
- Analysis technique **scalability**

Propositions

- **FrameSoC** infrastructure : storage, data-model, trace/tool/result management
- **Analysis flow** : statistics, data-mining, **visualization**...

Context: SoC-Trace project



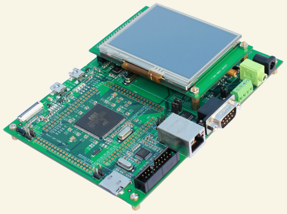
Embedded system trace analysis problematic

- **Hardware** and **software** complexity
- Trace **size** and **format** management
- Analysis technique **scalability**

Propositions

- **FrameSoC** infrastructure : storage, data-model, trace/tool/result management
- **Analysis flow** : statistics, data-mining, **visualization**...

Context: SoC-Trace project



Embedded system trace analysis problematic

- **Hardware** and **software** complexity
- Trace **size** and **format** management
- Analysis technique **scalability**

Propositions

- **FrameSoC** infrastructure : storage, data-model, trace/tool/result management
- **Analysis flow** : statistics, data-mining, **visualization**...

Fil rouge: typical embedded system use case

Fil rouge: application tracing result

Only 20 second duration but...

- Almost **1500** different functions, **4** threads
- **One million** of events
- **100 Mo** trace (Pajé format)

For a 10 minute-long video

- Same number of functions, but...
- More than **30 millions** of events!
- **3 GB** trace!

We can easily obtain well bigger traces!!

Fil rouge: application tracing result

Only 20 second duration but...

- Almost **1500** different functions, **4** threads
- **One million** of events
- **100 Mo** trace (Pajé format)

For a 10 minute-long video

- Same number of functions, but...
- More than **30 millions** of events!
- **3 GB** trace!

We can easily obtain well bigger traces!!

Fil rouge: application tracing result

Only 20 second duration but...

- Almost **1500** different functions, **4** threads
- **One million** of events
- **100 Mo** trace (Pajé format)

For a 10 minute-long video

- Same number of functions, but...
- More than **30 millions** of events!
- **3 GB** trace!

We can easily obtain well bigger traces!!

How to represent this trace behavior over time?

Gantt Chart is the most common technique employed by analysts...

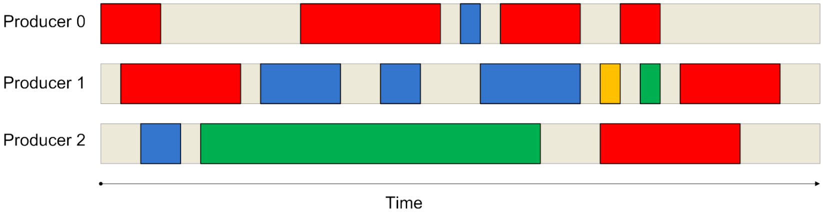


Figure 1: *Synthetic example of Gantt Chart*

How to represent this trace behavior over time?

Gantt Chart is the most common technique employed by analysts...

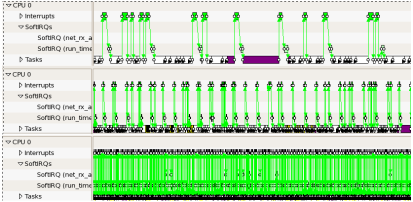


Figure 1: *KPTrace dezoom : example of time axis scalability issues*

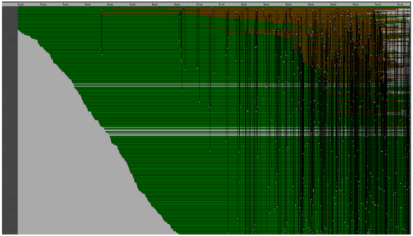


Figure 2: *Example of space limitations : Pajé trace with 700 producers*

... but it does not scale to voluminous traces

Our proposal: Ocelotl

Fit to Schneiderman's methodology...

- **Overview** first, zoom and filter, then details on demand

... build upon an algorithm proposed by Lamarche-Perrin

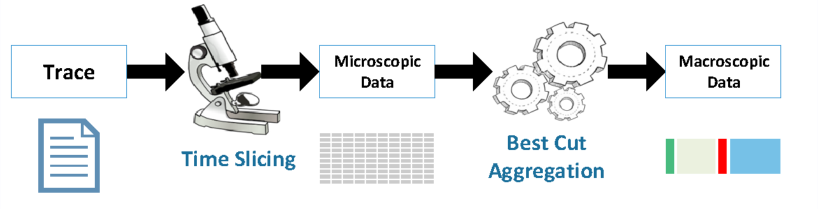
- Adapted to timestamped events using **time slicing**
- Extended to **multiple event sources**

Our proposal: Ocelotl

Fit to Schneiderman's methodology...

- **Overview** first, zoom and filter, then details on demand

... by providing a macroscopic description of the trace...



... build upon an algorithm proposed by Lamarche-Perrin

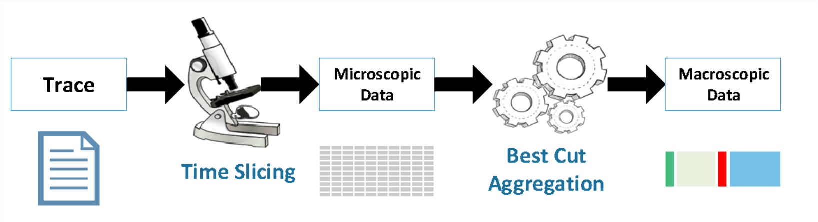
- Adapted to timestamped events using **time slicing**
- Extended to **multiple event sources**

Our proposal: Ocelotl

Fit to Schneiderman's methodology...

- **Overview** first, zoom and filter, then details on demand

... by providing a macroscopic description of the trace...

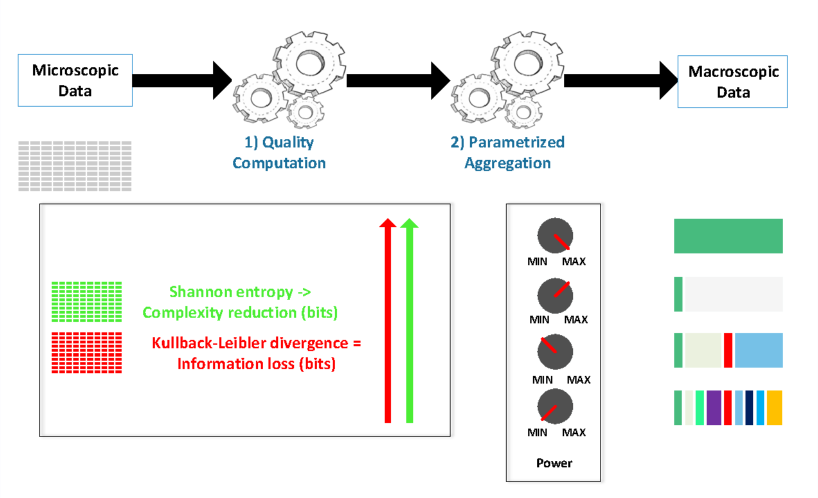


... build upon an algorithm proposed by Lamarche-Perrin

- Adapted to timestamped events using **time slicing**
- Extended to **multiple event sources**

User controlled level of details

Parametrized aggregation enables interaction



Microscopic Data

1) Quality Computation

2) Parametrized Aggregation

Macroscopic Data

Shannon entropy -> Complexity reduction (bits)

Kullback-Leibler divergence = Information loss (bits)

Analysis with Ocelotl (Settings)

Analysis with Ocelotl (Overview, Qualities)

Analysis with Ocelotl (Zoom and details)

Results

Information based time aggregation

- ... describes behavior by **highlighting phases** and **perturbations**
- **Interaction** helps to focus on these points
- **Performance**:
  - 20s to visualize 1M event trace
  - 2h to visualize 30M event trace (bounded by database query time)

But...

- Lack of **space dimension** representation

Results

Information based time aggregation

- ... describes behavior by **highlighting phases** and **perturbations**
- **Interaction** helps to focus on these points
- **Performance**:
  - 20s to visualize 1M event trace
  - 2h to visualize 30M event trace (bounded by database query time)

But...

- Lack of **space dimension** representation

Current Focus

New features

- **Spatio-temporal** aggregation

Use-cases

- **HPC/Distributed** system relevant **use-cases**

Current Focus

New features

- **Spatio-temporal** aggregation

Use-cases

- **HPC/Distributed** system relevant **use-cases**

Links

My website

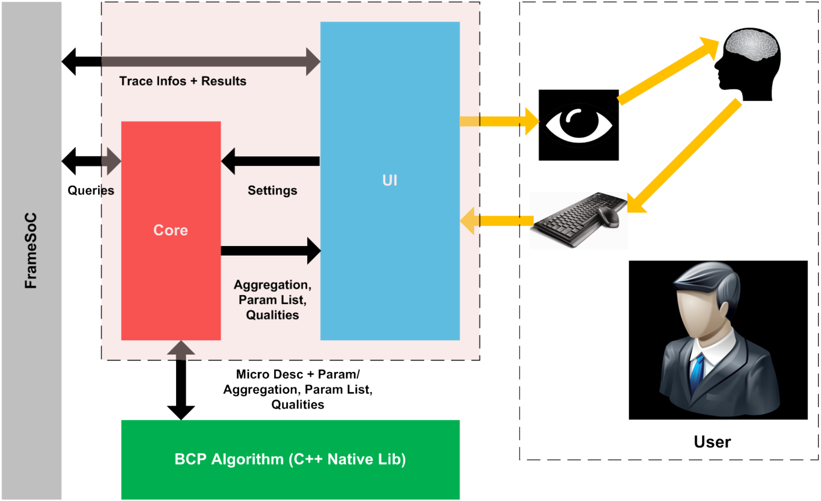
http://moais.imag.fr/membres/damien.dosimont/

Tools and libraries are available on my github

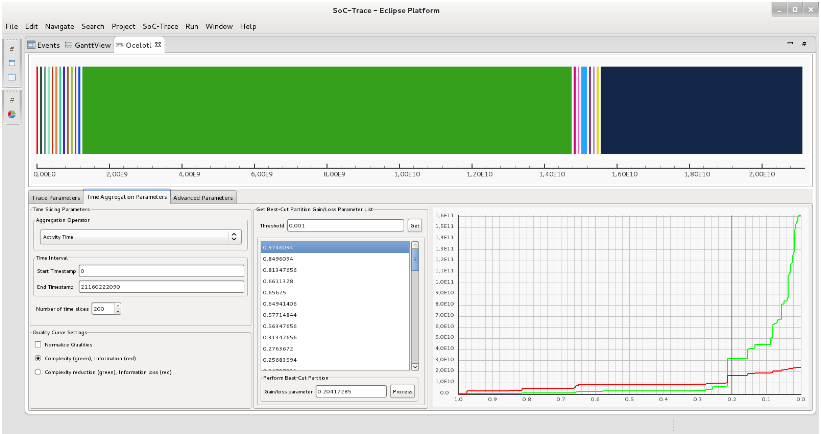
http://github.com/dosimont

Merci pour votre attention!

Implementation



Interface Overview



Time Slicing Parameters

Aggregation Operator: Activity Time

Time Interval

Start Timestamp: 0

End Timestamp: 21160222090

Number of time slices: 200

Quality Curve Settings

□ Normalize Qualities

● Complexity (green), Information (red)

○ Complexity reduction (green), Information loss (red)

Get Best-Cut Partition Gain/Loss Parameter List

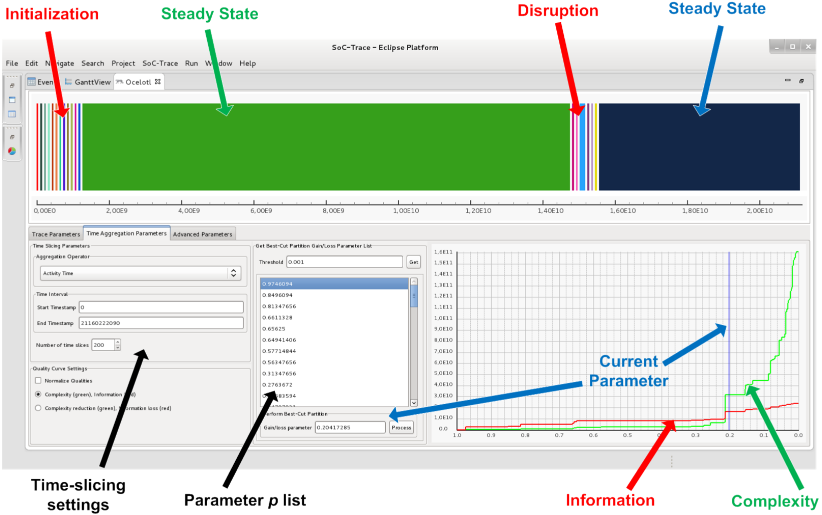
Threshold: 0.001

0.9746094
0.8496094
0.81347656
0.6611328
0.65625
0.64941406
0.57714844
0.56347656
0.31347656
0.2763672
0.25683594

Perform Best-Cut Partition

Gain/loss parameter: 0.20417285

Interface Overview



Initialization

Steady State

Disruption

Steady State

Current Parameter

Time-slicing settings

Parameter *p* list

Information

Complexity

SoC–Trace – Eclipse Platform

Trace Parameters | Time Aggregation Parameters | Advanced Parameters

Time Slicing Parameters

Aggregation Operator: Activity Time

Time Interval

Start Timestamp: 0

End Timestamp: 21160222090

Number of time slices: 200

Quality Curve Settings

□ Normalize Qualities

● Complexity (green), Information (red)

○ Complexity reduction (green), Information loss (red)

Get Best-Cut Partition Gain/Loss Parameter List

Threshold: 0.001 [Get]

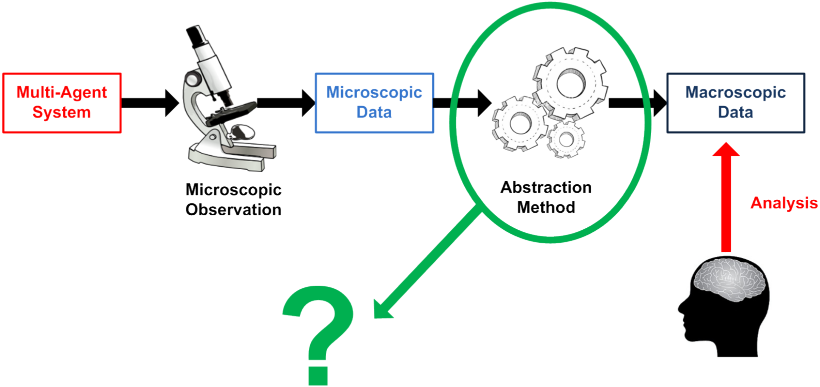
0.9746094
0.8496094
0.81347656
0.6611328
0.65625
0.64941406
0.57714844
0.56347656
0.31347656
0.2763672
0.[illegible]83594

Perform Best-Cut Partition

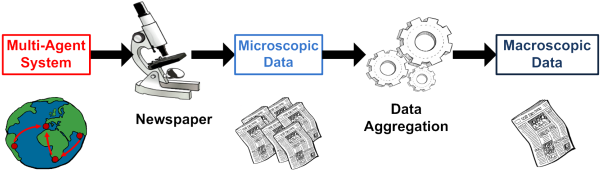
Gain/loss parameter: 0.20417285 [Process]

Lamarche-Perrin Works: Multi-Agent Systems

How to Build a Meaningful Macroscopic Description?



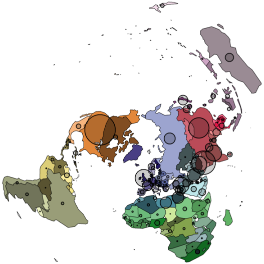
Example: Geomedia Project



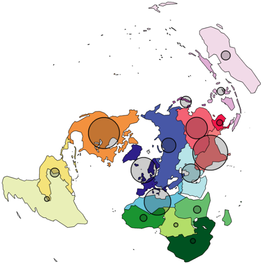
Information Loss

Complexity Reduction

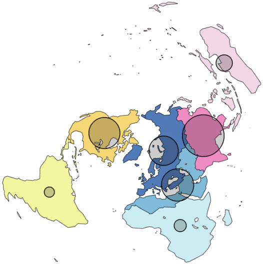
Macroscopic Semantics



Resolution: Max



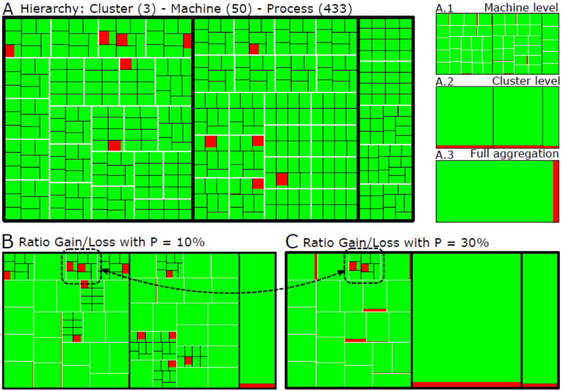
Resolution: Intermediary



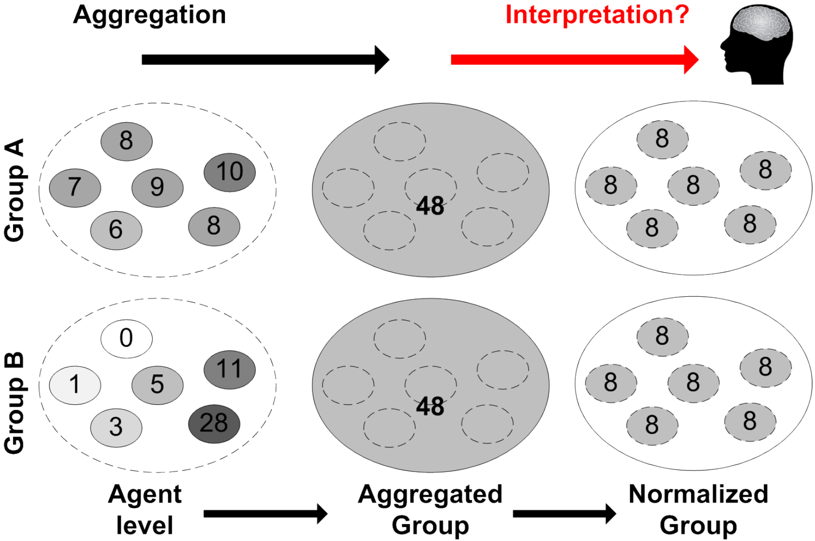
Resolution: Min

Example: Viva

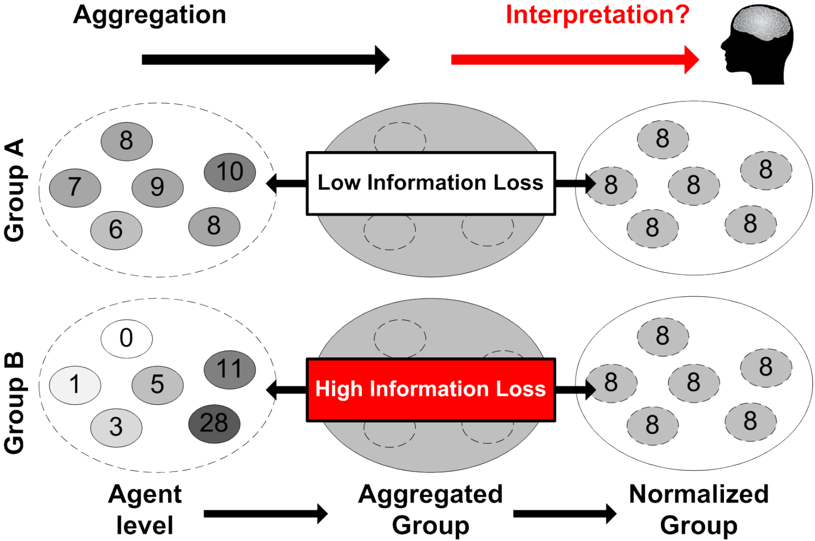
Represent Hierarchical Structure according to Value Heterogeneity


A Hierarchy: Cluster (3) - Machine (50) - Process (433)
A.1 Machine level
A.2 Cluster level
A.3 Full aggregation
B Ratio Gain/Loss with P = 10%
C Ratio Gain/Loss with P = 30%

Information Loss



Information Loss



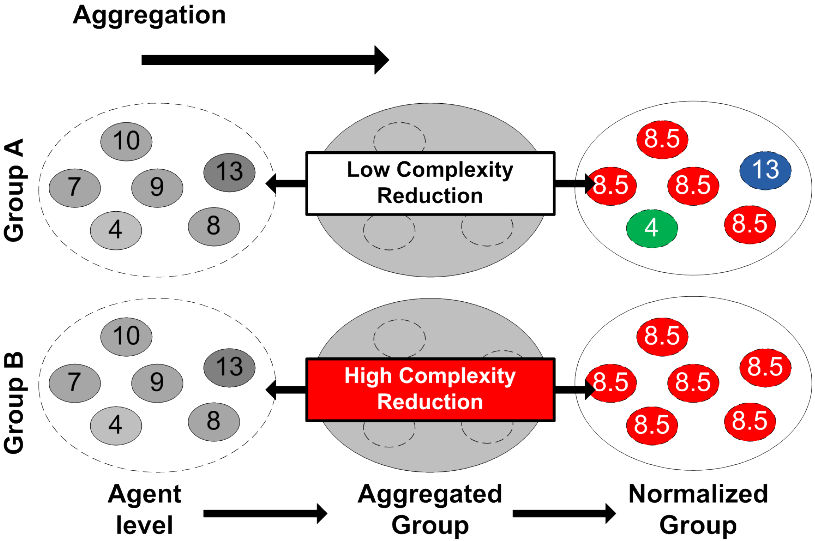
Information Loss Measure

Kullback-Leibler Divergence

$$\text{loss}(A||e) = \sum_{e \in A} v(e) \times \log_2 \left(\frac{v(e)}{v(A)} \right) \text{ in bits/x}$$

- Quantity of information than one **loses** by using an **aggregated description** instead of the **microscopic description**

Complexity Reduction



Complexity Reduction Measure

Shannon Entropy

$\mathrm{H}(v) = \sum (v(i) \times \log_2 v(i))$ in bits/x

Entropy Reduction

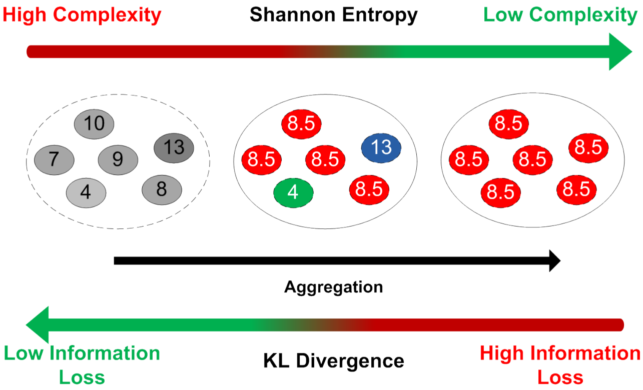
$\mathrm{gain}(A||e) = \mathrm{H}(A) - \mathrm{H}(e)$ in bits/x

- Quantity of information than one **saves** by encoding the **aggregated description** instead of the **microscopic description**

Compromise Finding between Information Loss and Complexity Reduction

Parametrized Information Criterion

$$\text{pIC}(\mathcal{A}) = p \times \text{gain}(\mathcal{A}) - (1 - p) \times \text{loss}(\mathcal{A})$$

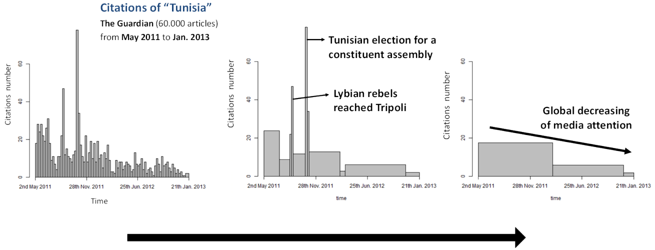


Temporal Aggregation

Temporal Aggregation principle

- Same principle but only consecutive data can be aggregated

Ex: Tunisia citation


Citations of "Tunisia"
The Guardian (60.000 articles)
from **May 2011** to **Jan. 2013**

Tunisian election for a constituent assembly

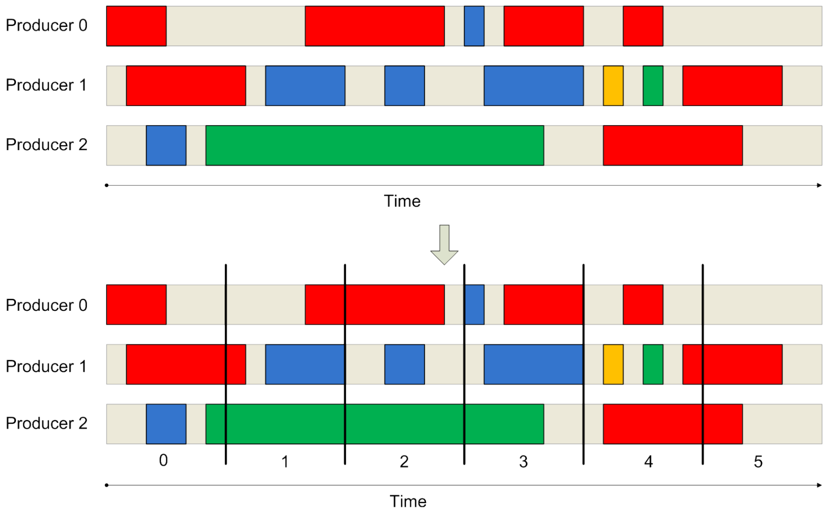
Lybian rebels reached Tripoli

Global decreasing of media attention

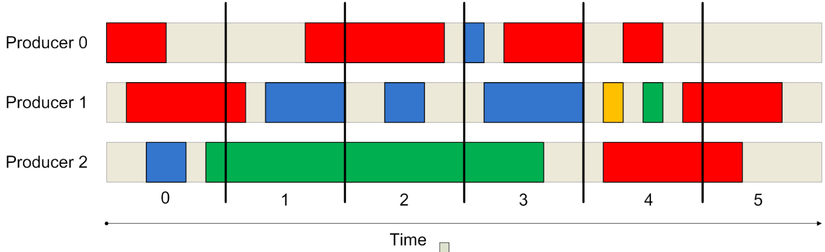
p is growing

Need of a microscopic level description

Microscopic Level: Time-Slicing



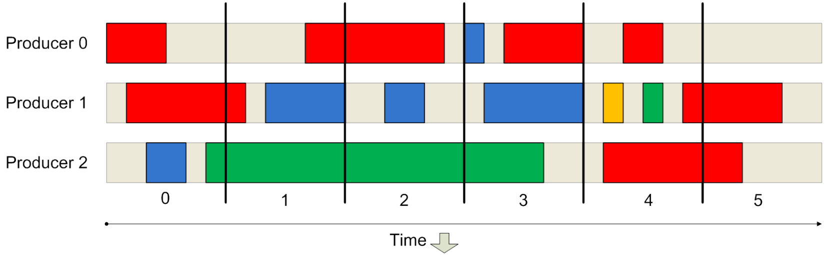
Microscopic Level: Producer Activity Time Matrix



	0	1	2	3	4	5
Producer 0	0.5	0.33	0.83	0.83	0.33	0
Producer 1	0.83	0.83	0.33	0.83	0.33	0.66
Producer 2	0.5	1	1	0.66	0.83	0.33

Part number

Microscopic Level: State Activity Time Cubic Matrix



Producer	State	0	1	2	3	4	5
Producer 0	🟥	0.5	0.33	0.83	0.66	0.33	0
	🟦	0	0	0	0.16	0	0
	🟩	0	0	0	0	0	0
	🟨	0	0	0	0	0	0
Producer 1	🟥	0.83	0.16	0	0	0.16	0.66
	🟦	0	0	0.33	0.83	0	0
	🟩	0	0	0	0	0.16	0
	🟨	0	0	0	0	0.16	0
Producer 2	🟥	0	0	0	0	0.83	0.33
	🟦	0.33	0	0	0	0.33	0
	🟩	0.16	1	1	0.66	0.33	0
	🟨	0	0	0	0	0	0

Part number

Quality Computation

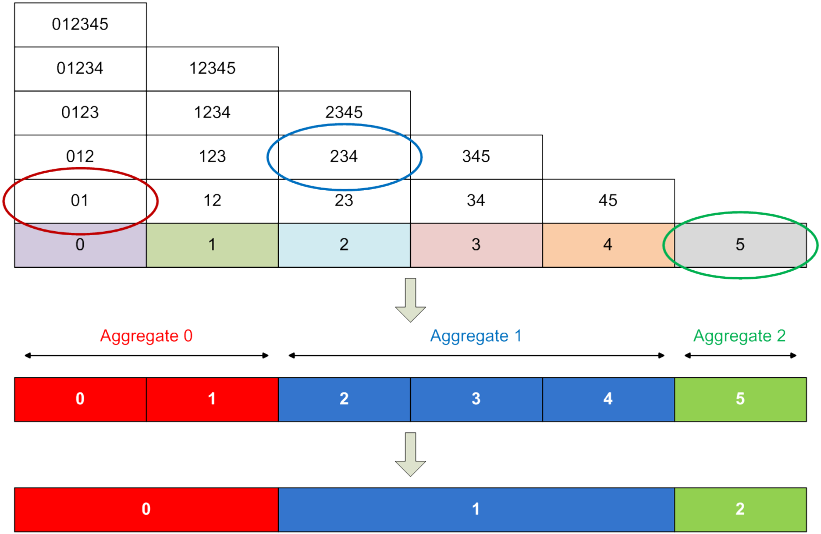
Gain and loss formulas: originally for scalars

012345					
01234	12345				
0123	1234	2345			
012	123	234	345		
01	12	23	34	45	
0	1	2	3	4	5

Adaptation for time-sliced description

- Vector (ex: activity time per process)
  $\text{quality}(A) = \sum_{i \in n} \text{quality}(A[i])$
- Matrix (ex: activity time per state type)
  $\text{quality}(A) = \sum_{i \in n} (\sum_{j \in m} \text{quality}(A[i][j]))$

Best-Cut Partition for a given *p*



012345					
01234	12345				
0123	1234	2345			
012	123	234	345		
01	12	23	34	45	
0	1	2	3	4	5

Aggregate 0 Aggregate 1 Aggregate 2

0	1	2	3	4	5

0	1	2