Ocelotl: Time Aggregation Visualization for Trace Overview

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

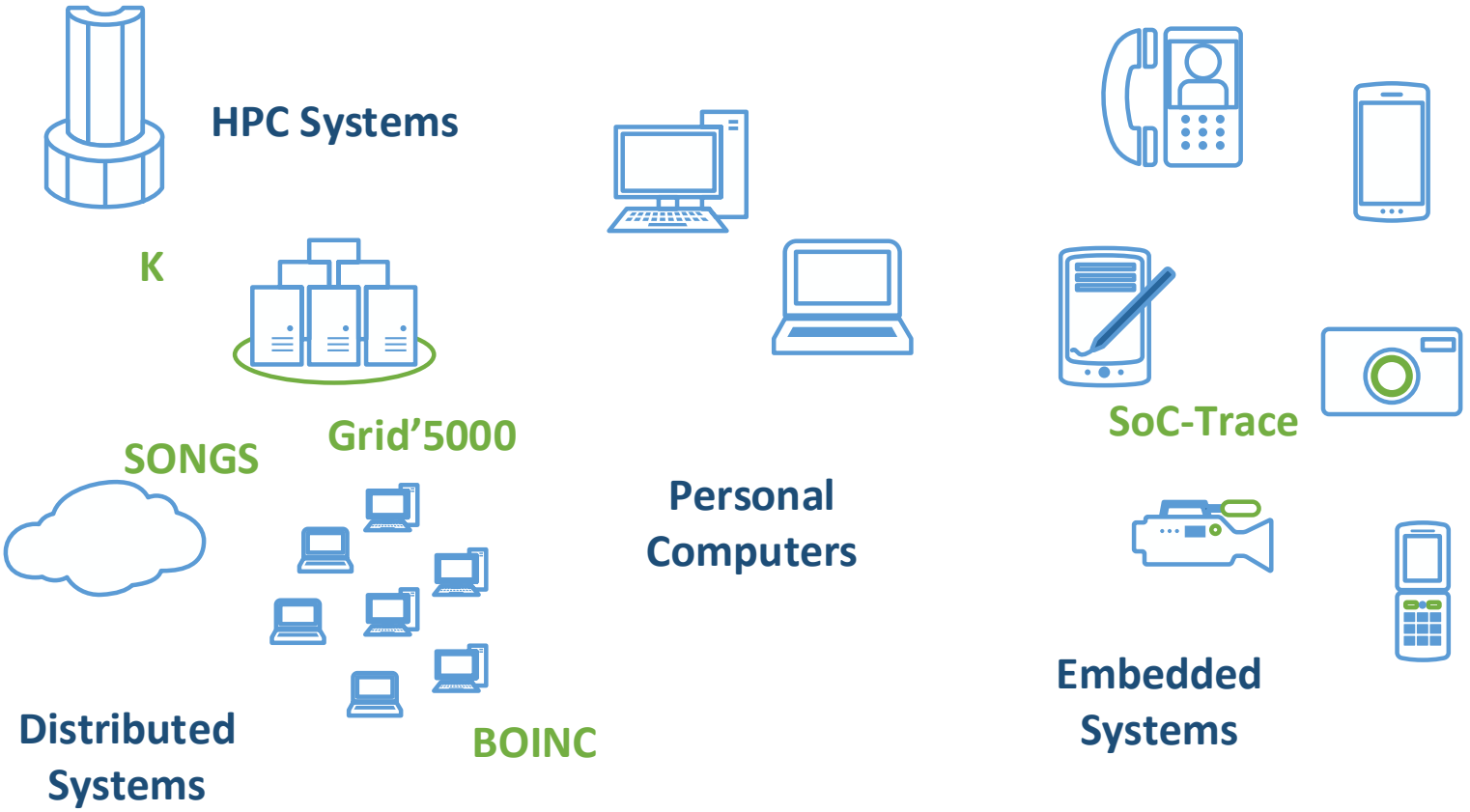
[1] INRIA

[2] Grenoble University

firstname.lastname@imag.fr

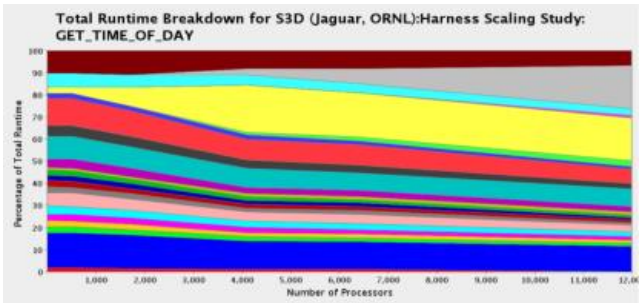
Riken Seminary (September 24th 2013)

Trace analysis problematics



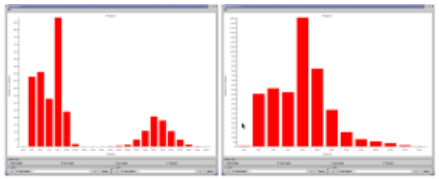
A visualization shows a trace particular aspect

Global Analysis

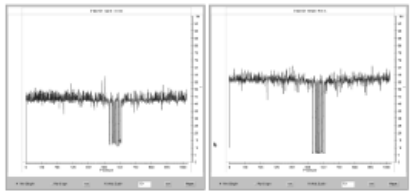


Execution Comparison

Correlations

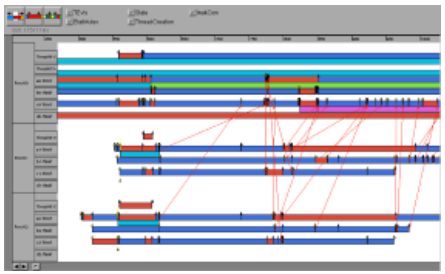


Outliers

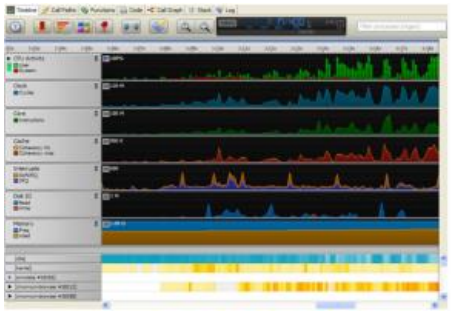


Workload

Run behavior

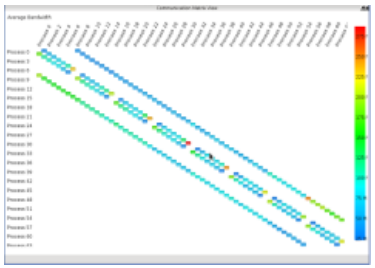


Causality relations

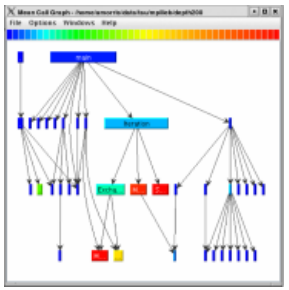


Resource usage

Structure



Communications



Call graphs

Trace overview and scalability issues

Schneiderman's analysis methodology

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

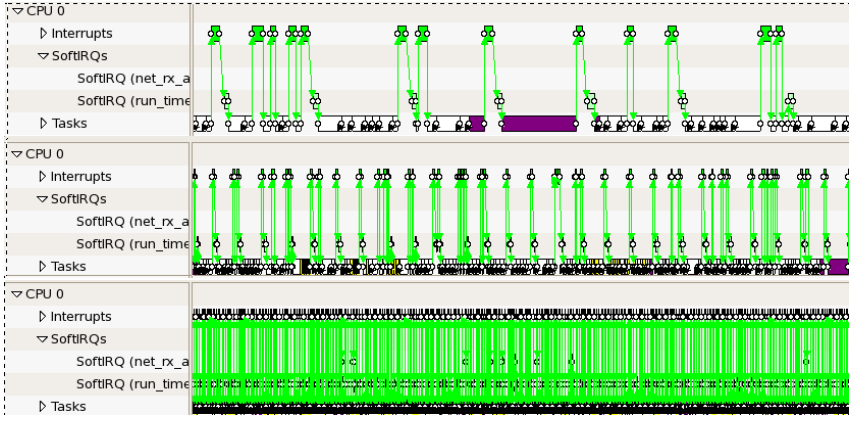


Figure 1 : *KPTrace dezoom : example of time axis scalability issues*

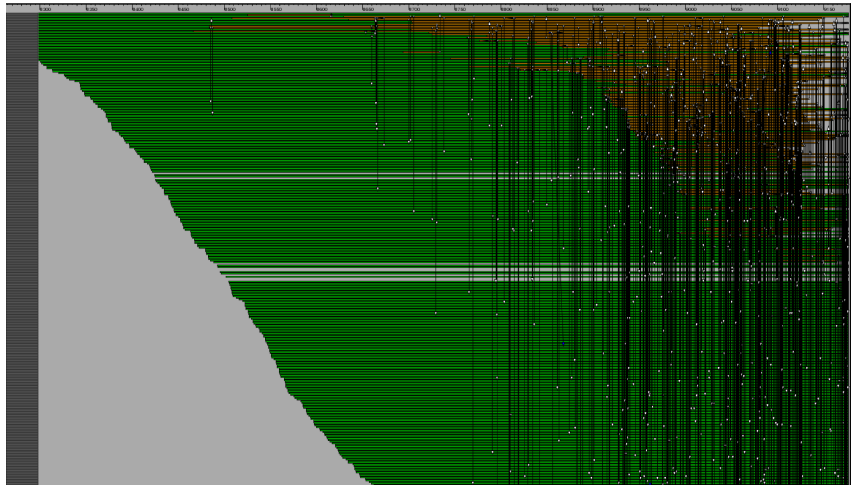


Figure 2 : *Example of space limitations : Pajé trace with 700 producers*

Structural representations

Lucas Schnorr's works: Triva (Viva)

- Hierarchical aggregation using sum operator
- Time aggregation : time slice duration chosen by user

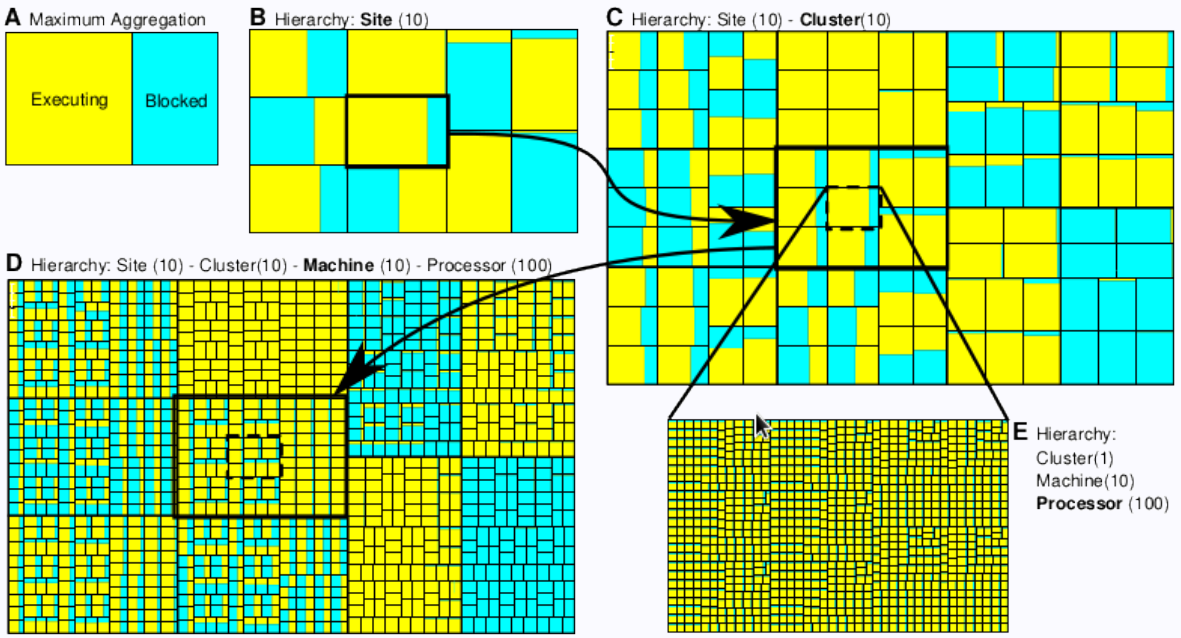


Figure 3 : *Triva treemap view example, showing different aggregation steps*

Temporal representations

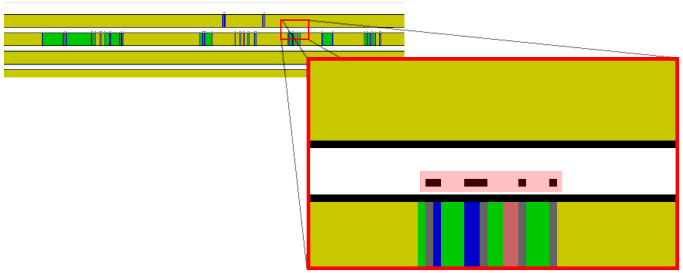


Figure 4 : *LLTng Eclipse Viewer visual aggregation uses dots*

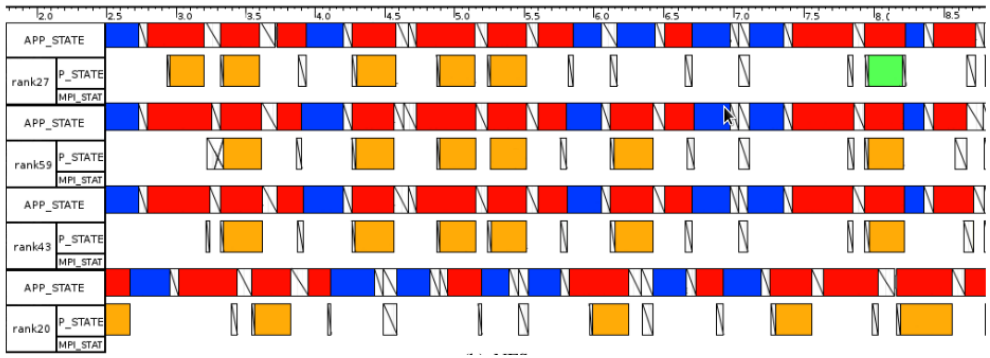


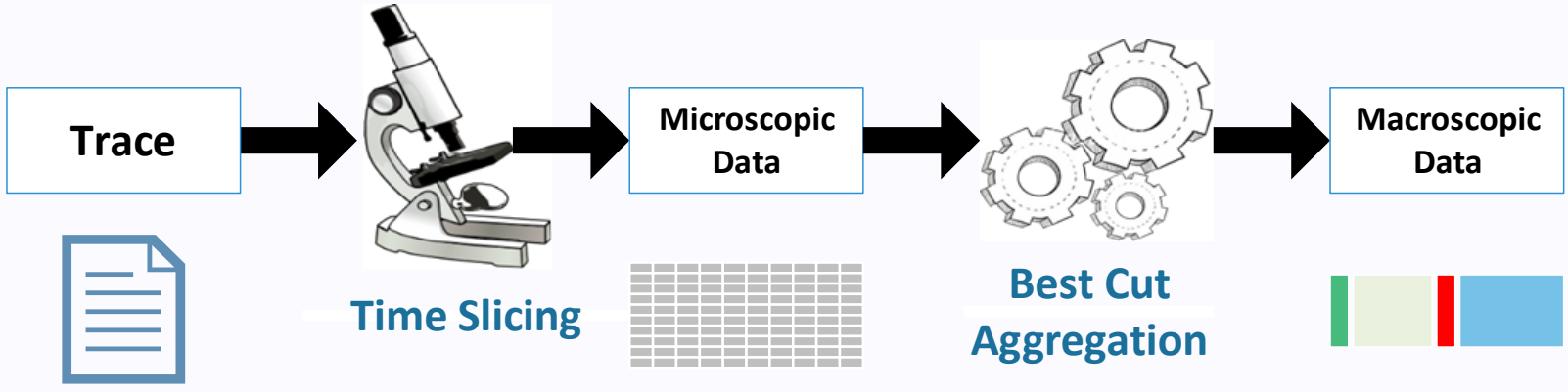
Figure 5 : *Pajé aggregates small states together*

Ex: Visual aggregation on Gantt Charts

- Lack of semantics
- Aggregation behavior difficult to predict

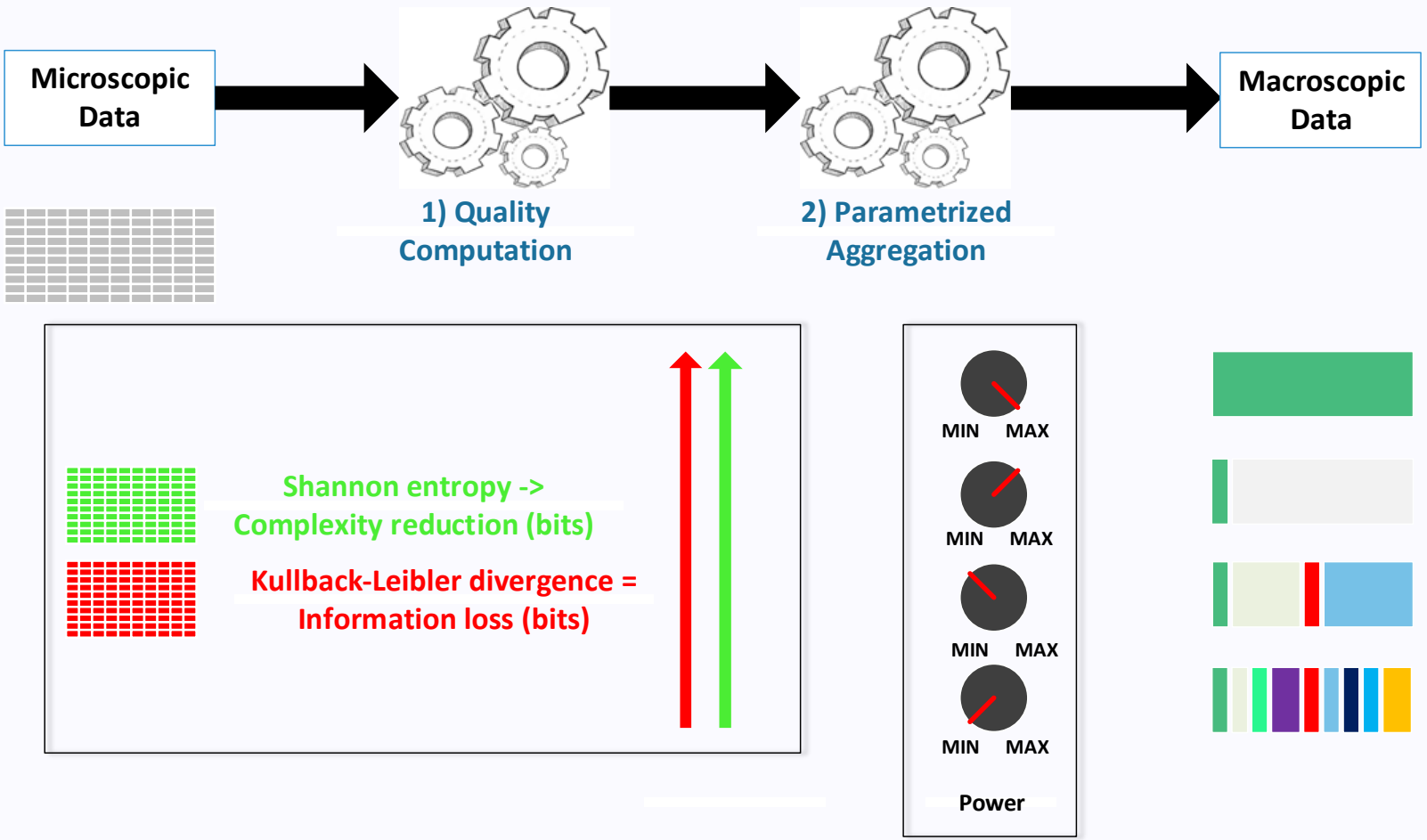
Our proposition: Ocelotl (SoC-Trace Project)

Build a macroscopic description of the trace (Lamarche-Perrin)

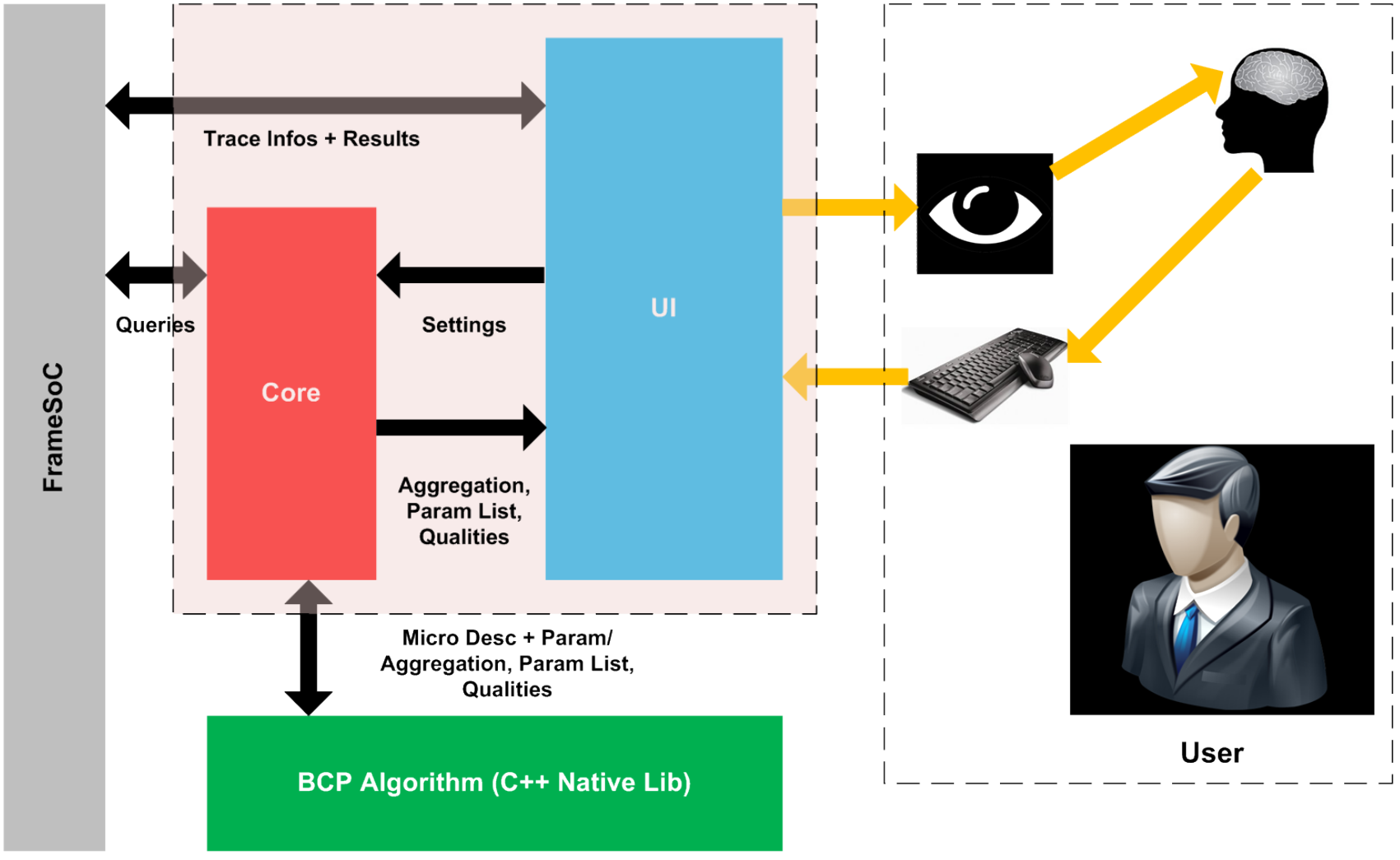


Our proposition: Ocelotl (SoC-Trace Project)

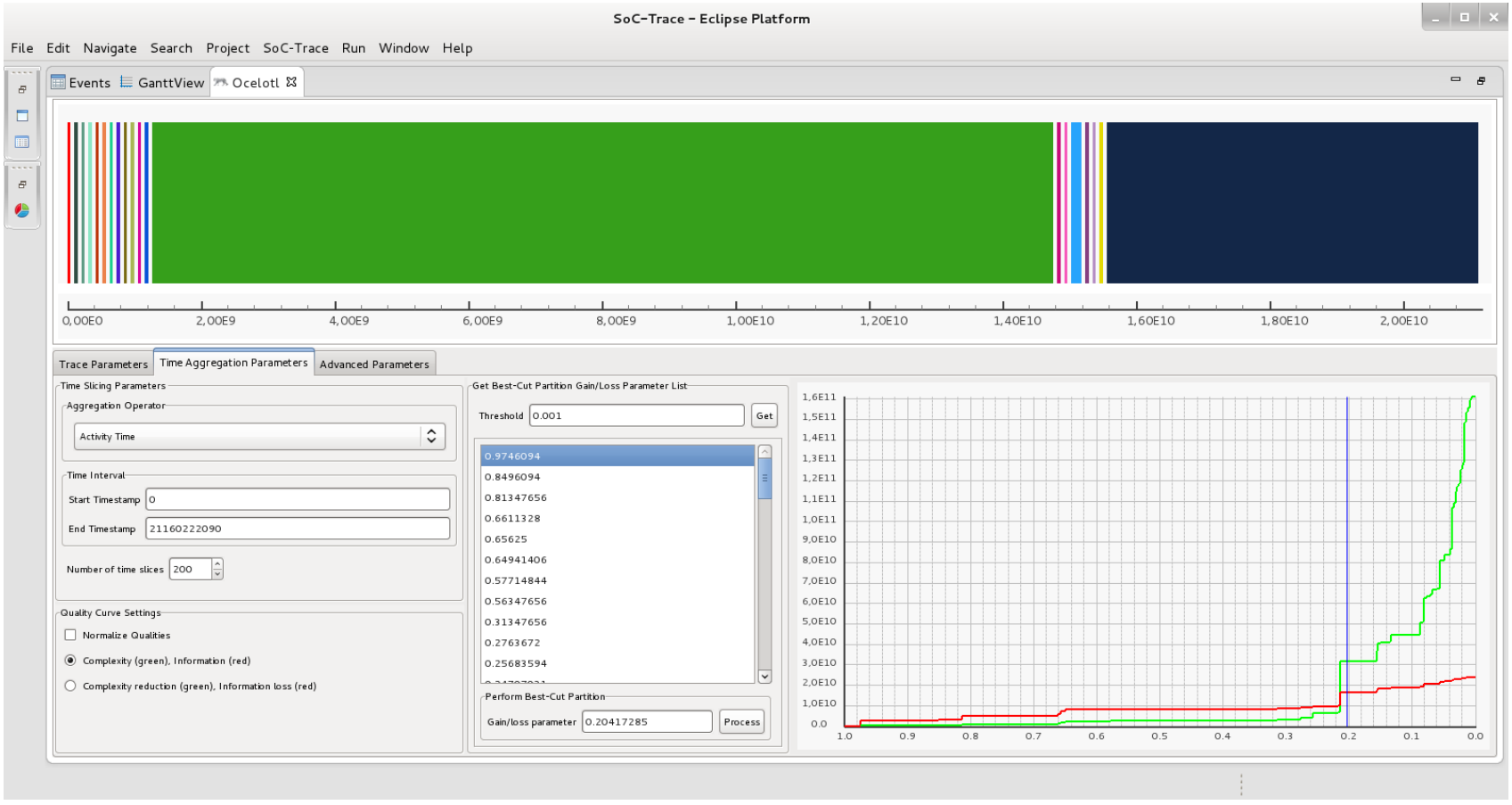
Enable the user to control the aggregation



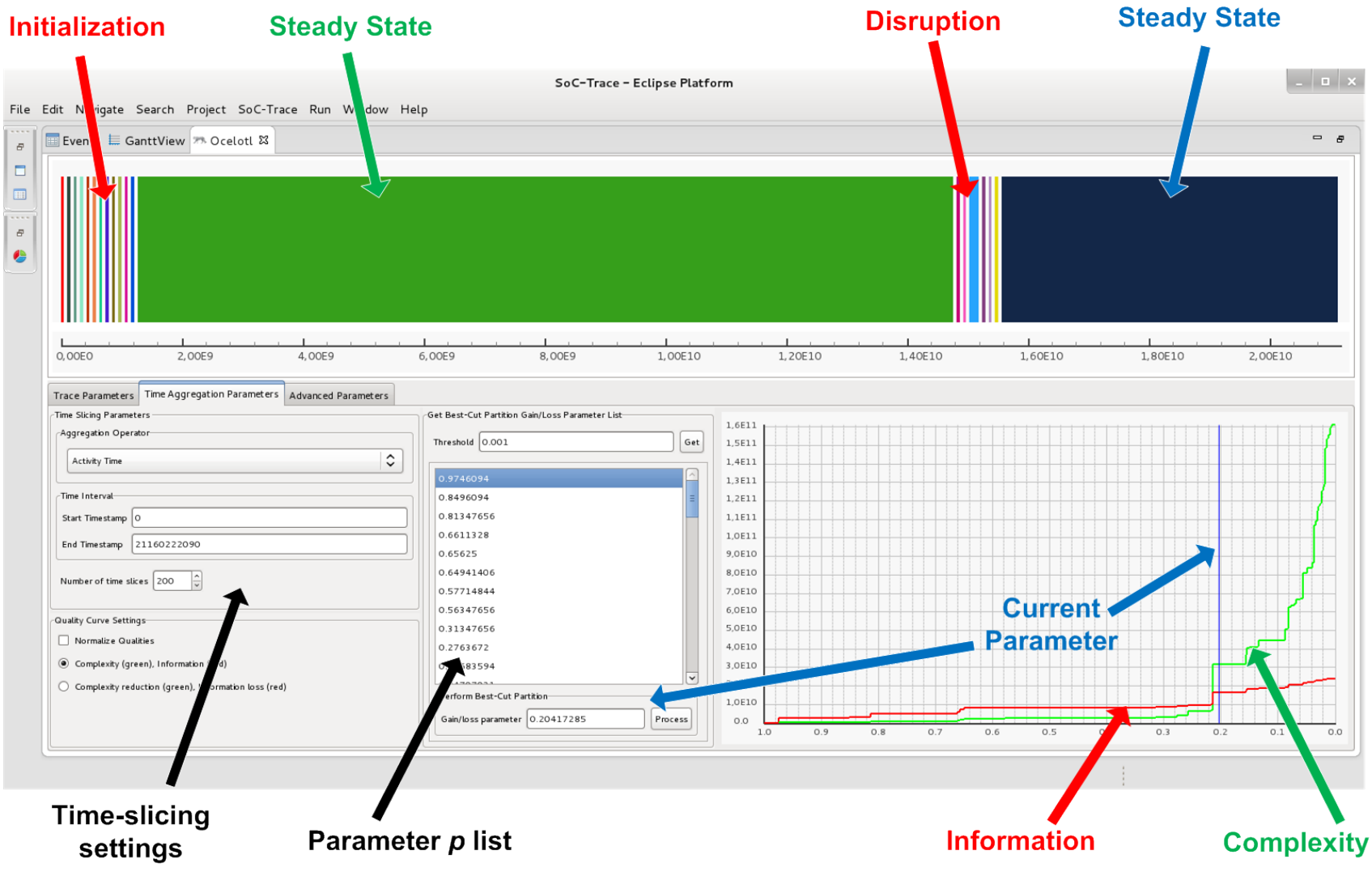
Implementation



Interface Overview



Interface Overview



Results

Time Aggregation Visualization

- Able to represent application **behavior over time**
- Solves some **time scalability**/overview issues
- Performances?

But...

- Lack of **space dimension** representation
- Lack of **interaction** (details-on-demand)

Results

Time Aggregation Visualization

- Able to represent application **behavior over time**
- Solves some **time scalability**/overview issues
- Performances?

But...

- Lack of **space dimension** representation
- Lack of **interaction** (details-on-demand)

Future Works

New features

- **Spatio-temporal** aggregation
- Aggregation operators
- User **interaction**

Outside embedded system domain

- What about HPC/Distributed Systems?

Future Works

New features

- **Spatio-temporal** aggregation
- Aggregation operators
- User **interaction**

Outside embedded system domain

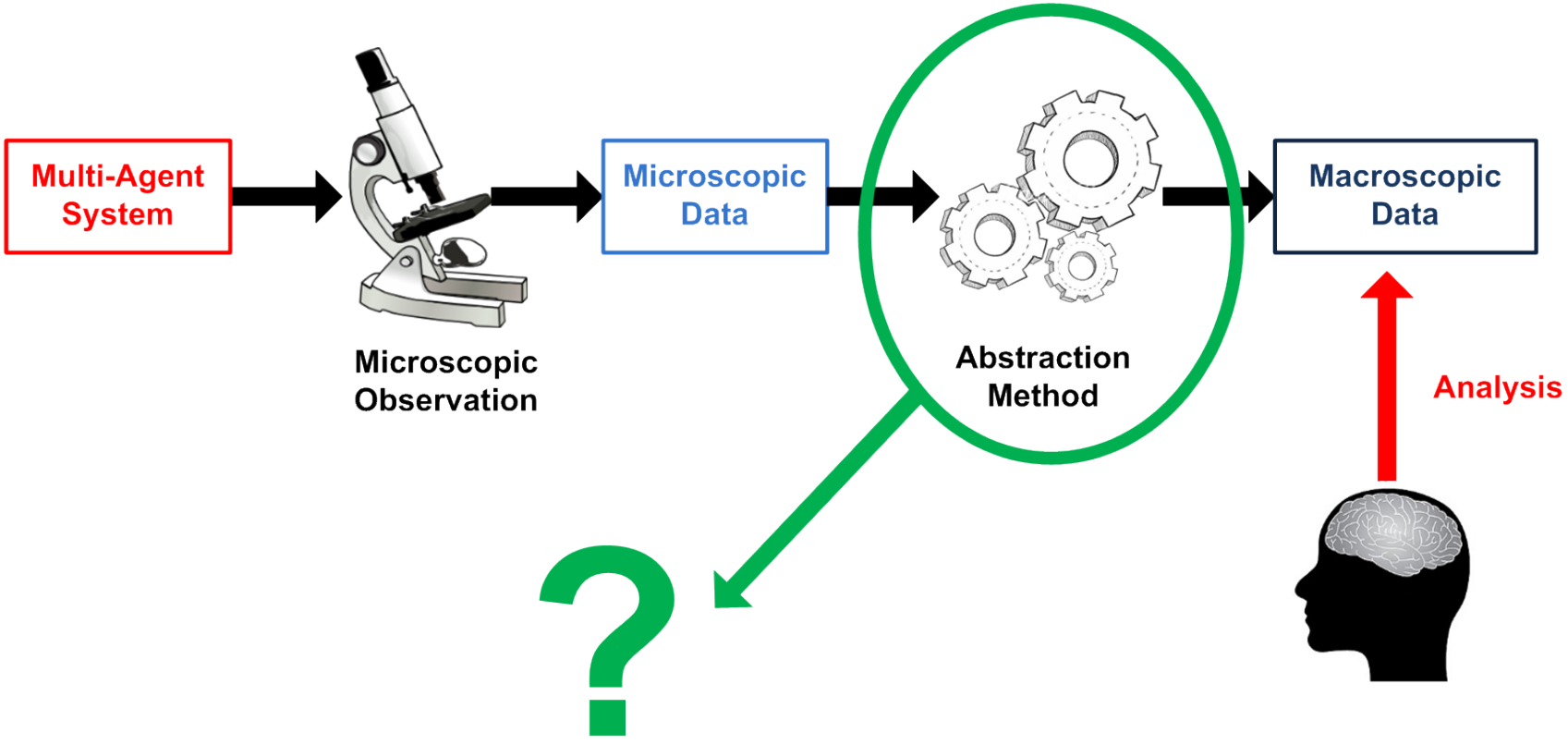
- What about HPC/Distributed Systems?

Merci de votre attention!

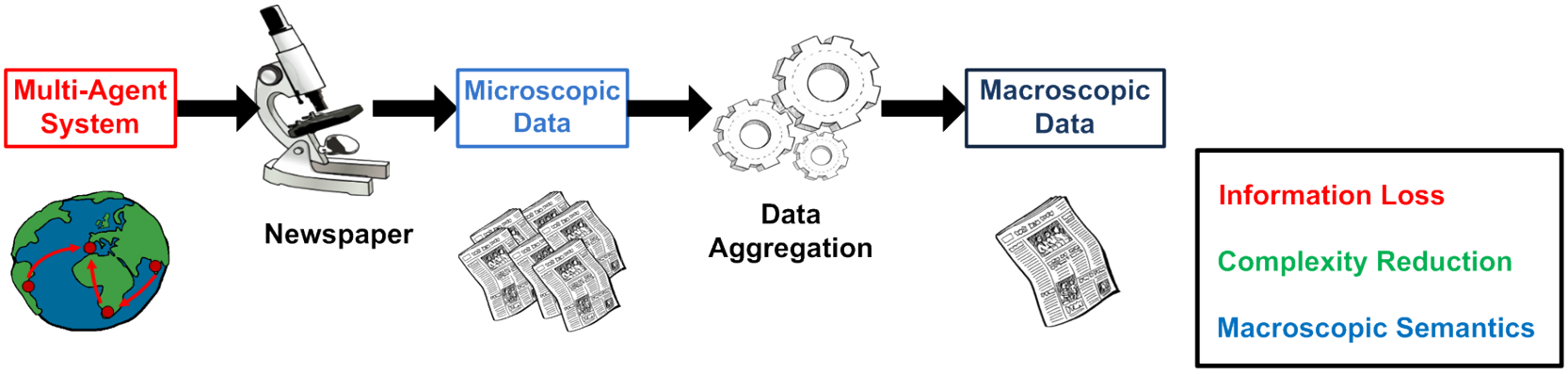
http://moais.imag.fr/membres/damien.dosimont/

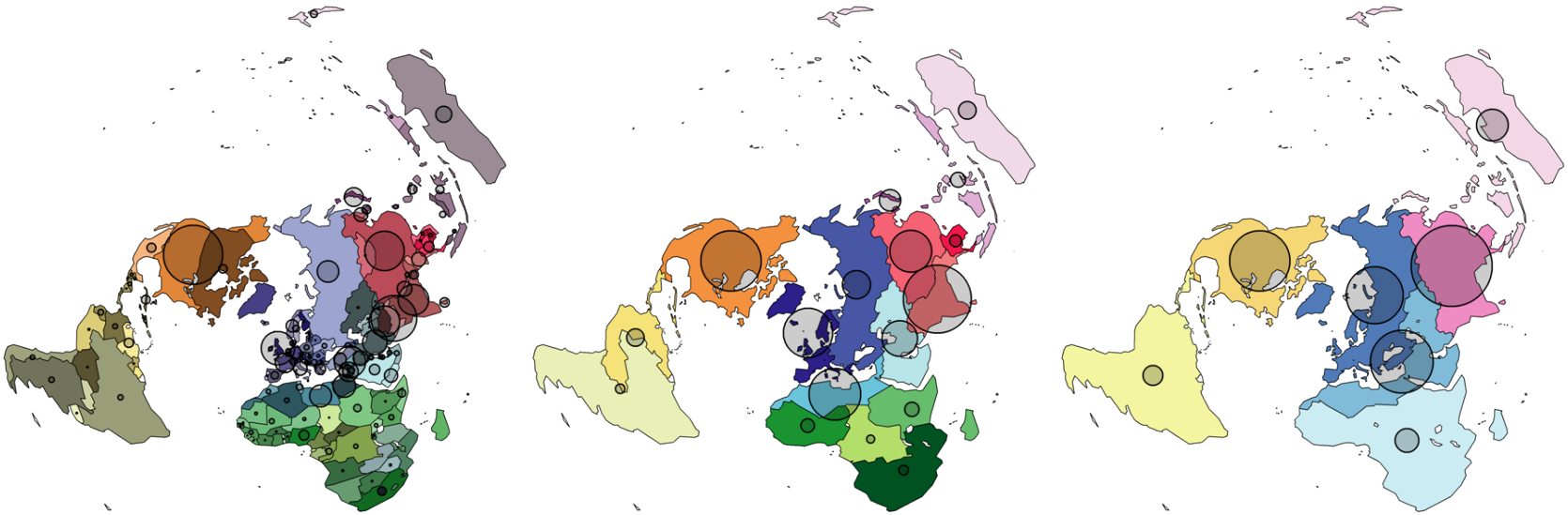
Lamarche-Perrin Works: Multi-Agent Systems

How to Build a Meaningful Macroscopic Description?



Example: Geomedia Project

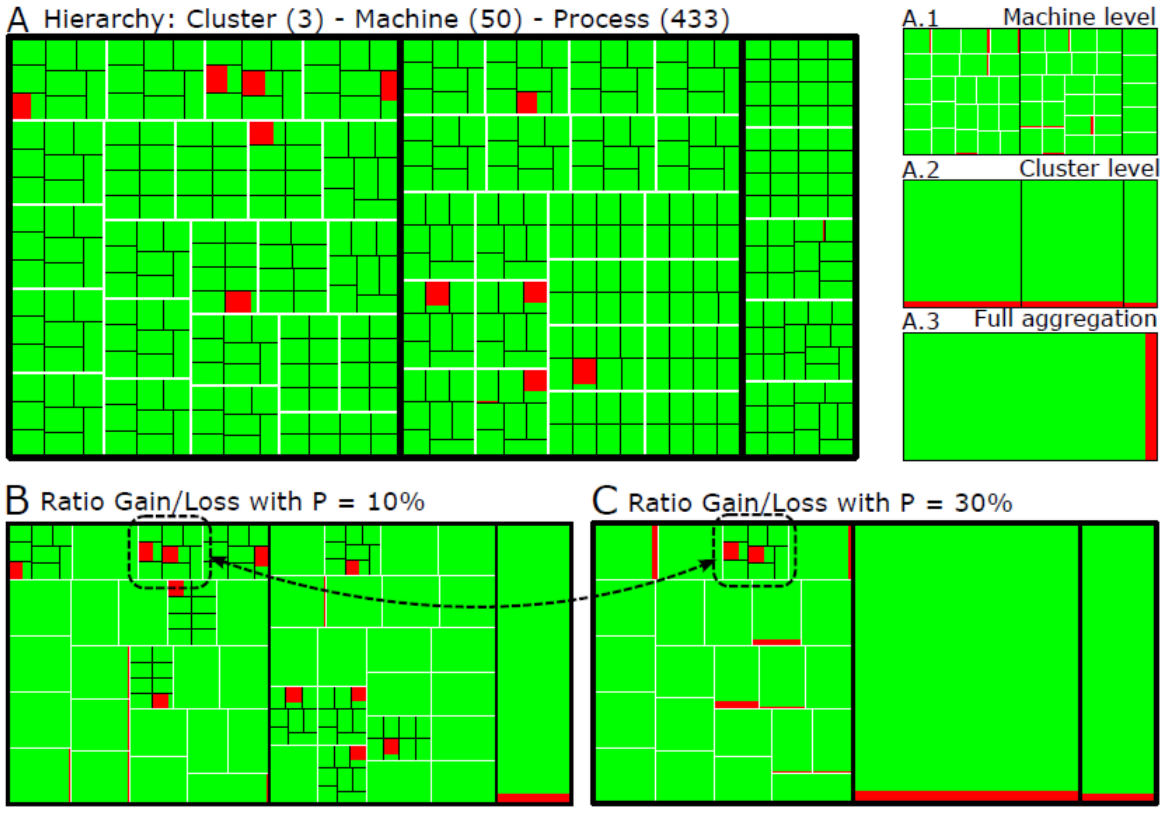




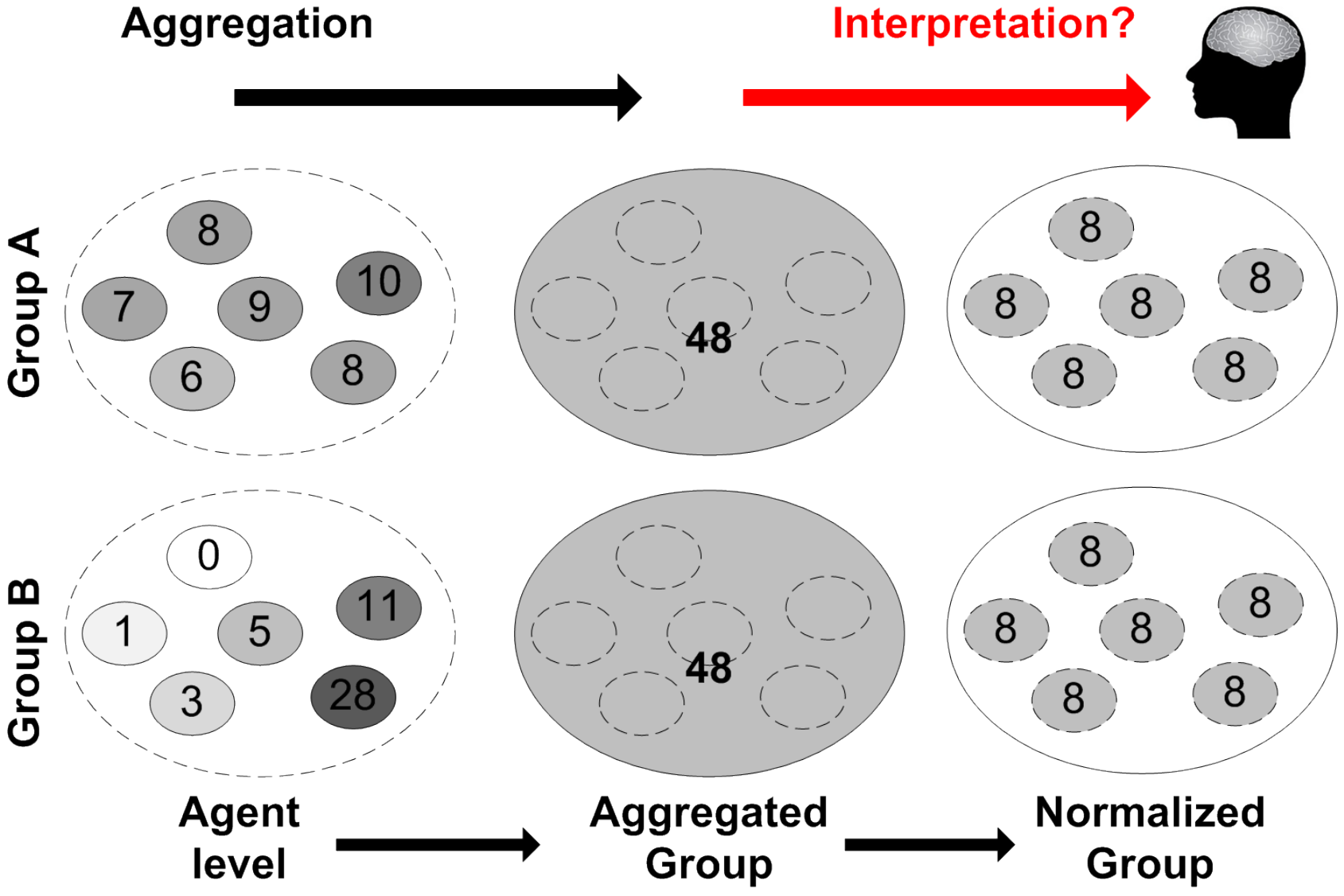
Resolution: Max | Resolution: Intermediary | Resolution: Min

Example: Viva

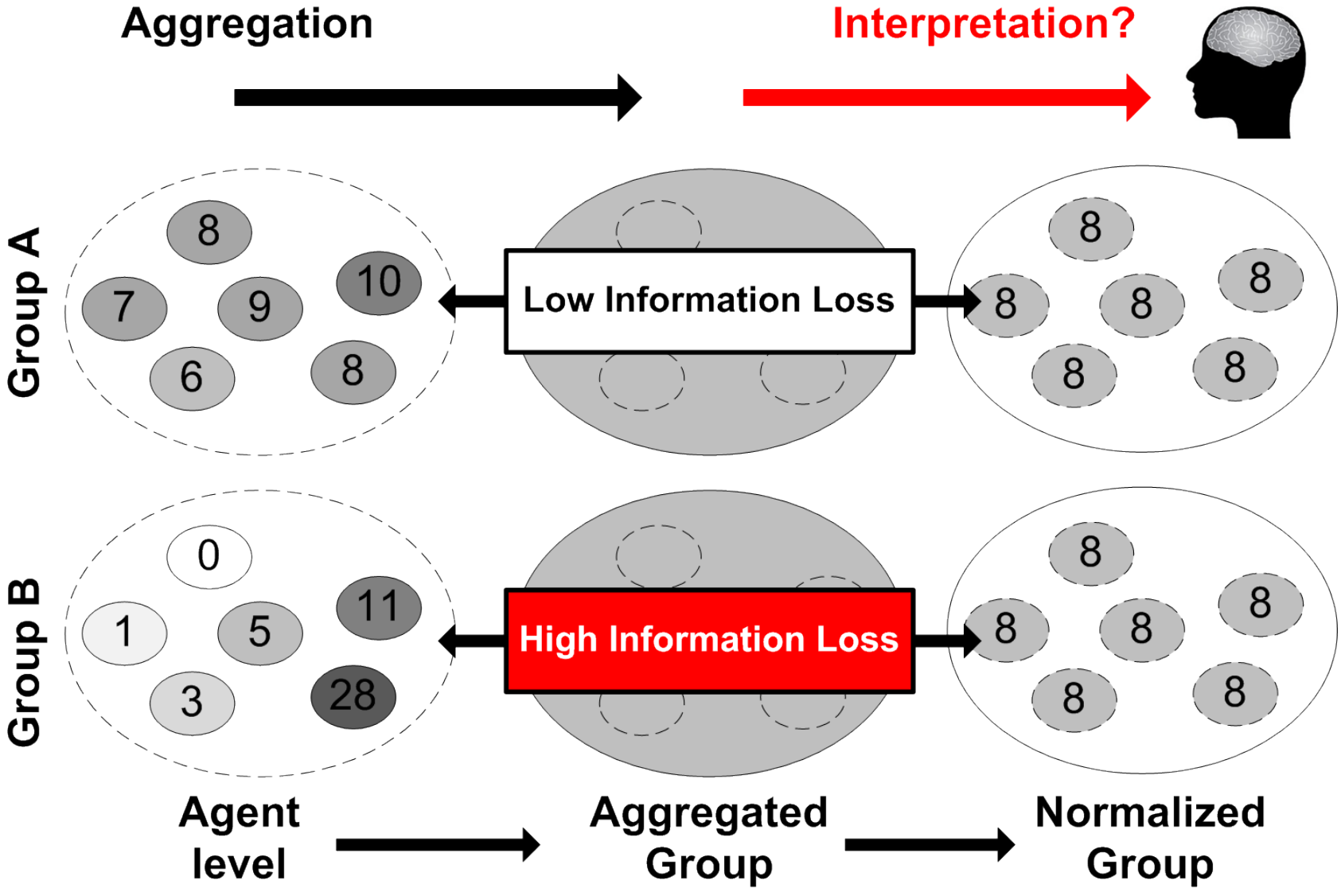
Represent Hierarchical Structure according to Value Heterogeneity



Information Loss



Information Loss



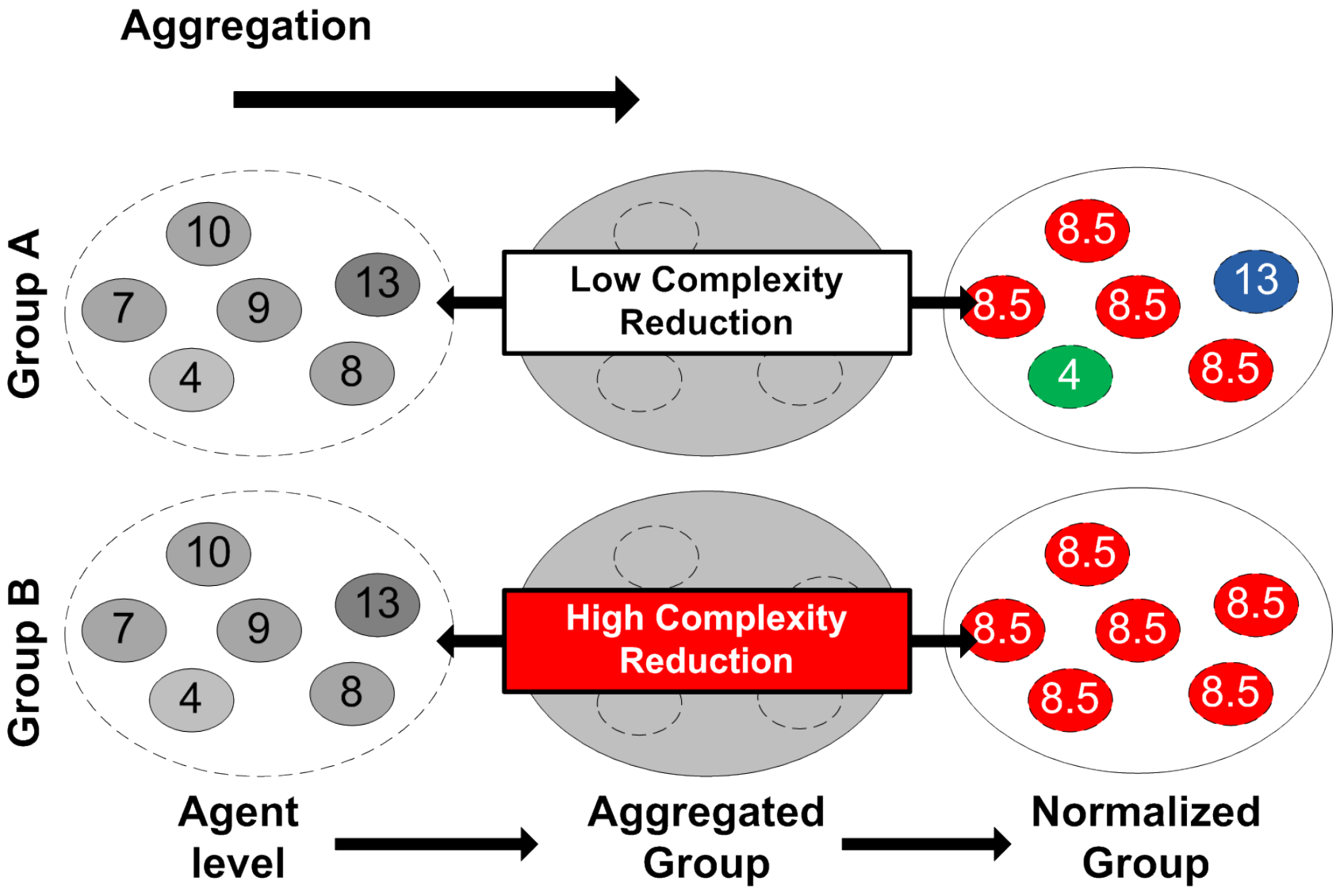
Information Loss Measure

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

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

Entropy Reduction

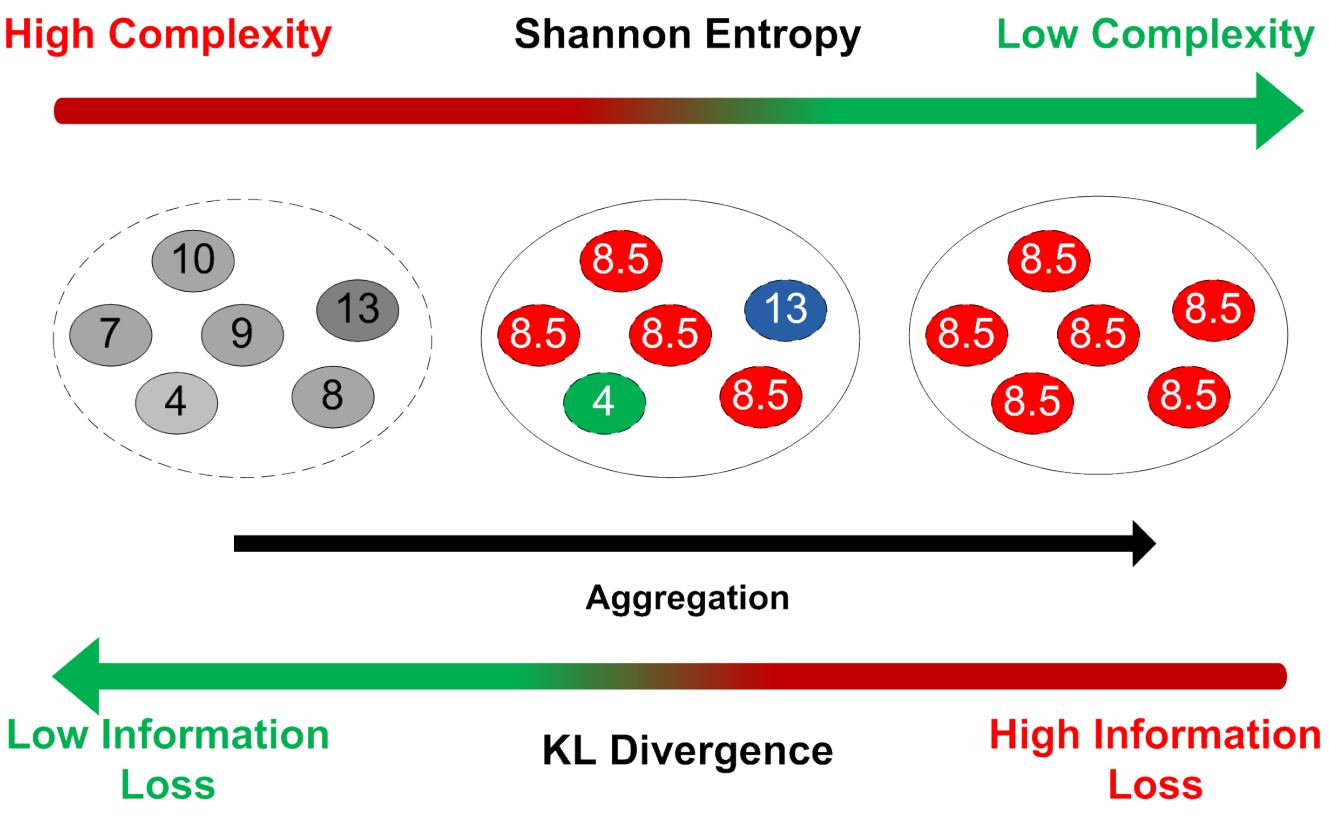
$gain(A||e) = H(A) - H(e)$ in bits/x

- 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

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

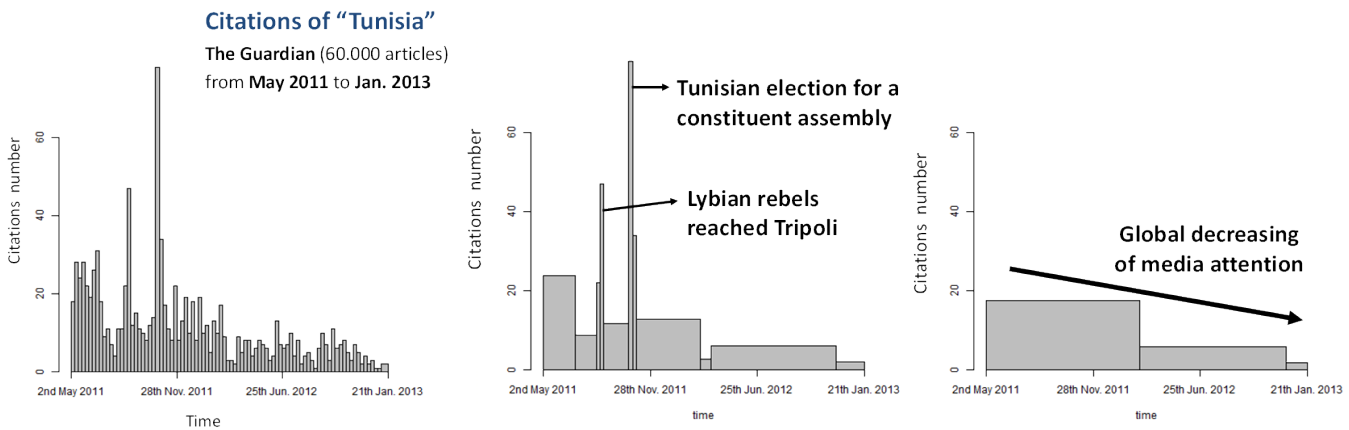


Temporal Aggregation

Temporal Aggregation principle

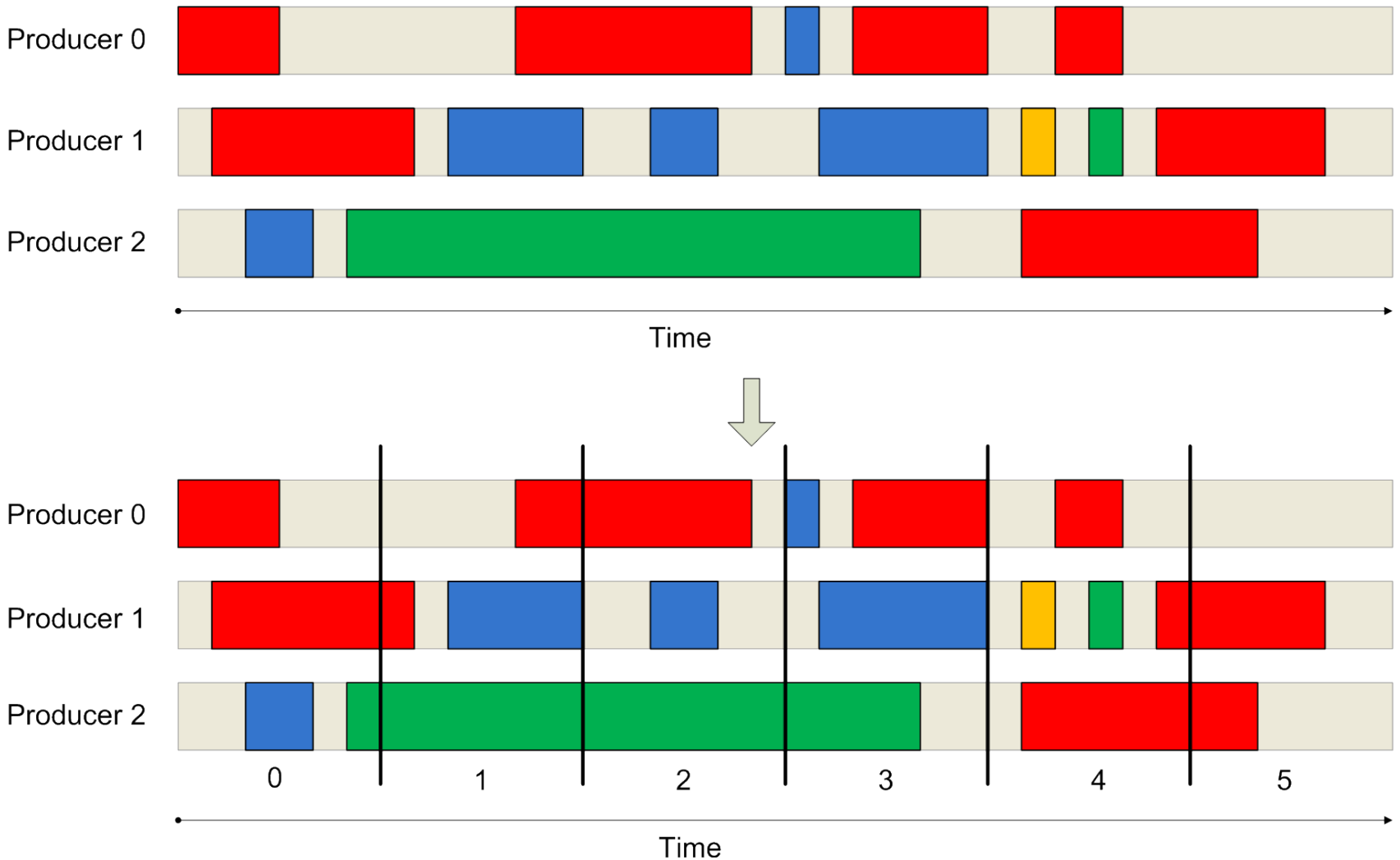
- Same principle but only consecutive data can be aggregated

Ex: Tunisia citation

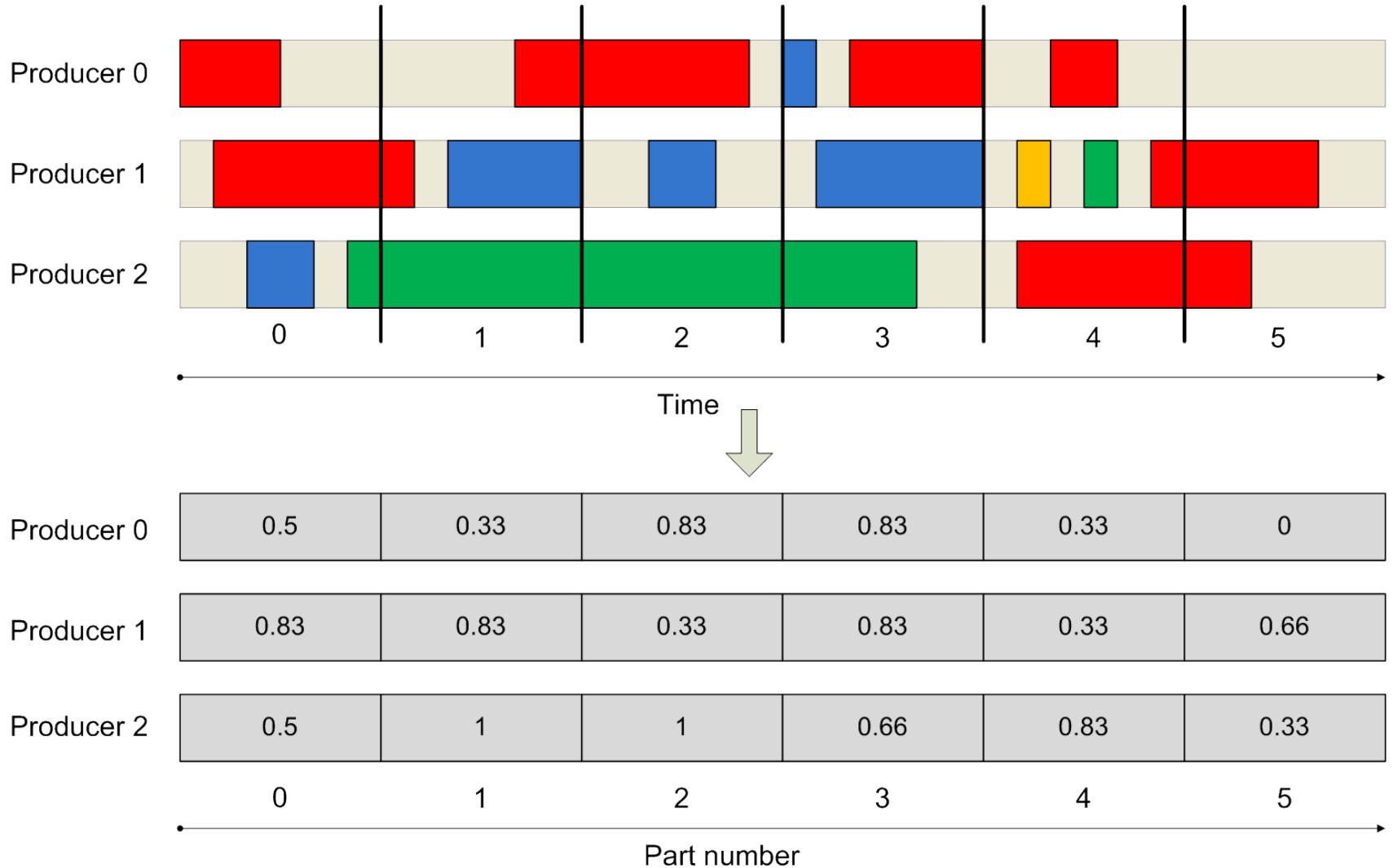


Need of a microscopic level description

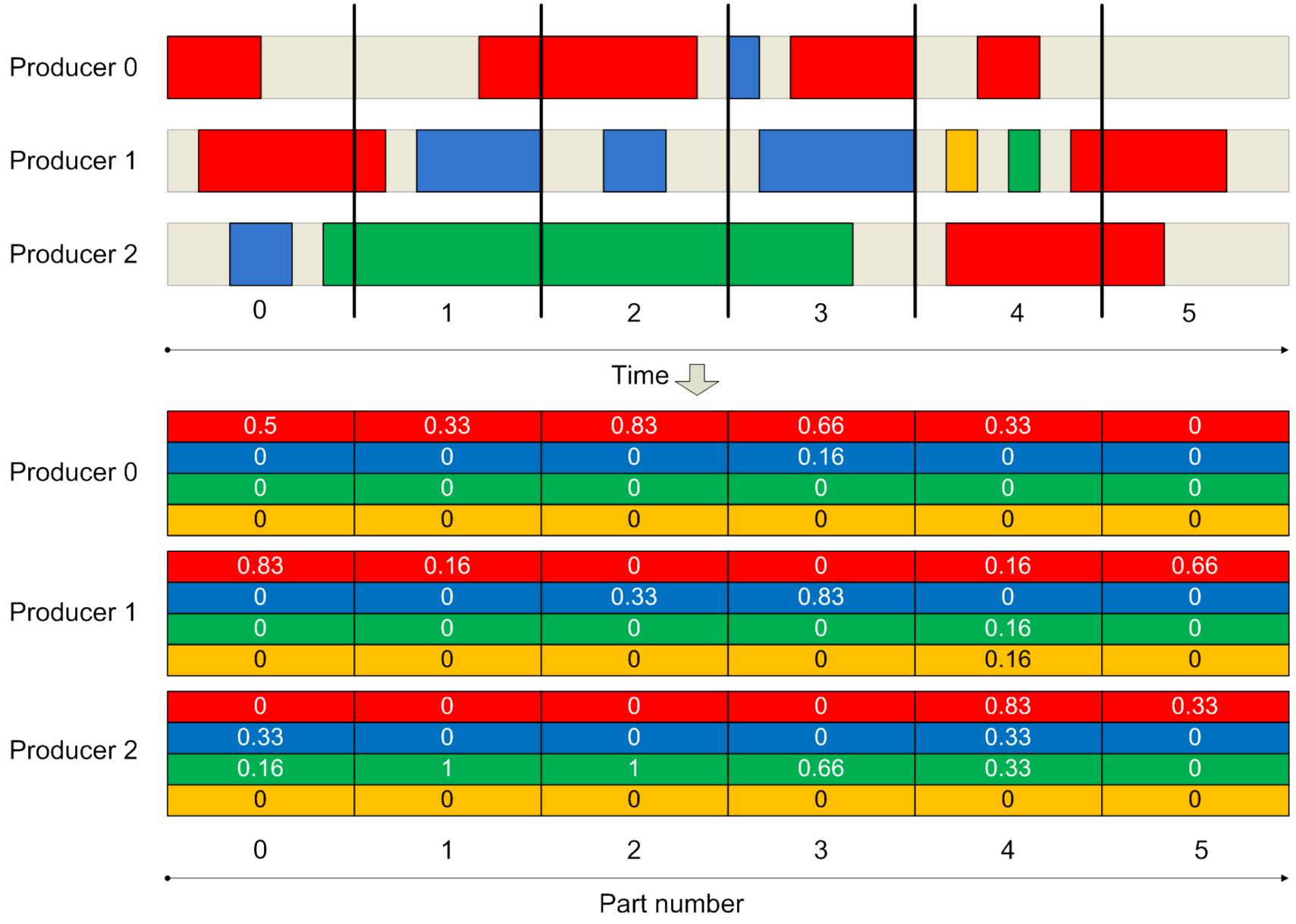
Microscopic Level: Time-Slicing



Microscopic Level: Producer Activity Time Matrix



Microscopic Level: State Activity Time Cubic Matrix



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*