Tabular dataset structuring

Concepts and principles

Contents

Introduction

Presentation

Tools

Example

Dataset structure

Record oriented

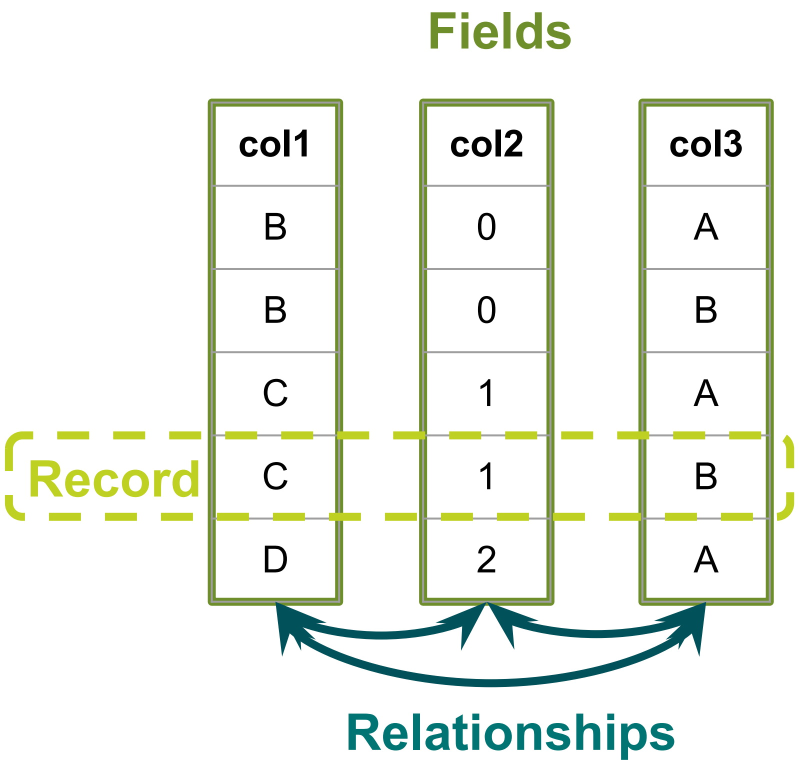
- Dataset is a list of records
- A semantic entity is a record
- Length is variable

Field oriented

- Fields have semantics
- Fields are dependent
- A semantic entity is a set of record or the entire Dataset

Dataset structure

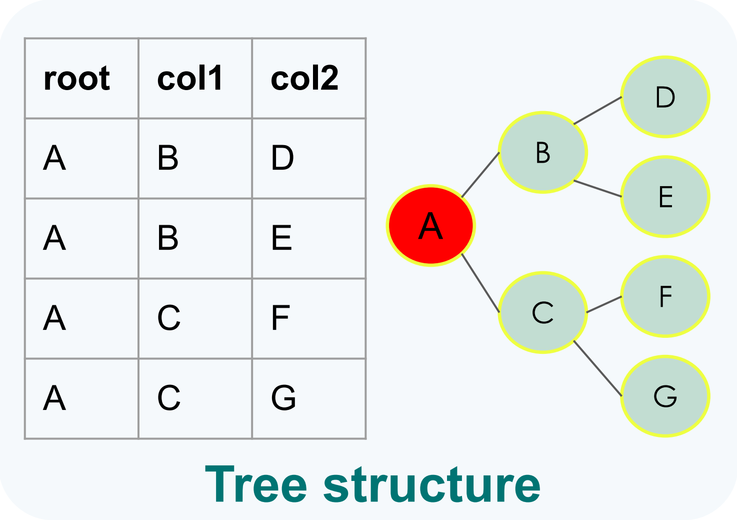
- Tree structure
- Matrix structure
- Mixed structure



Fields

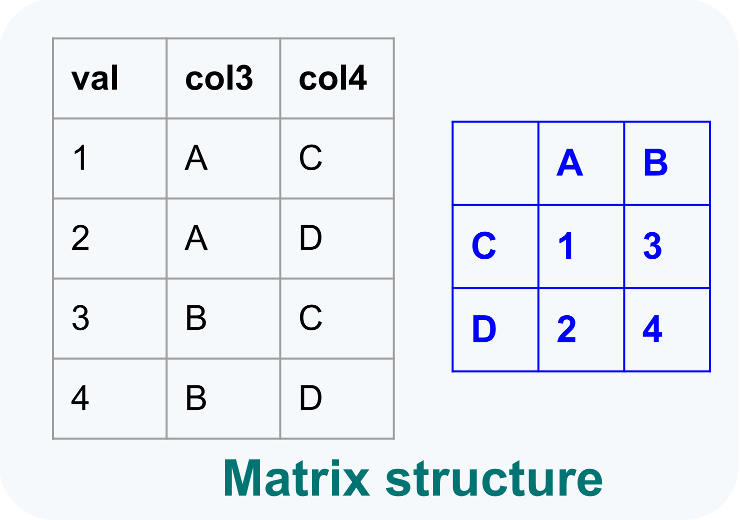
col1	col2	col3
B	0	A
B	0	B
C	1	A
C	1	B
D	2	A

Relationships



root	col1	col2
A	B	D
A	B	E
A	C	F
A	C	G

Tree structure



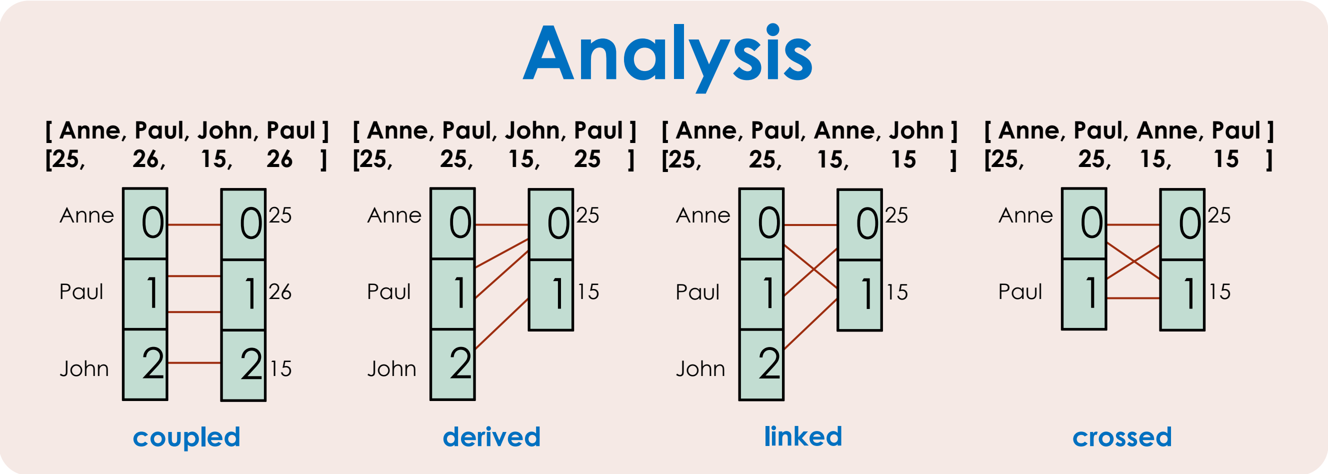
val	col3	col4
1	A	C
2	A	D
3	B	C
4	B	D

	A	B
C	1	3
D	2	4

Matrix structure

Dataset Structuring

- Structure analysis
  - Field qualification
  - Relationship
- Data structuration
  - Canonical format
  - Convergence



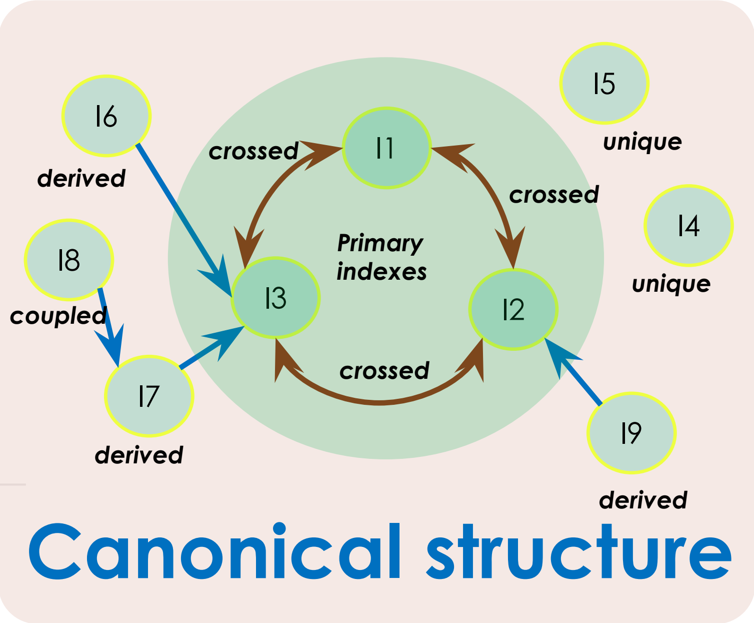
Analysis

[Anne, Paul, John, Paul]
[25, 26, 15, 26]

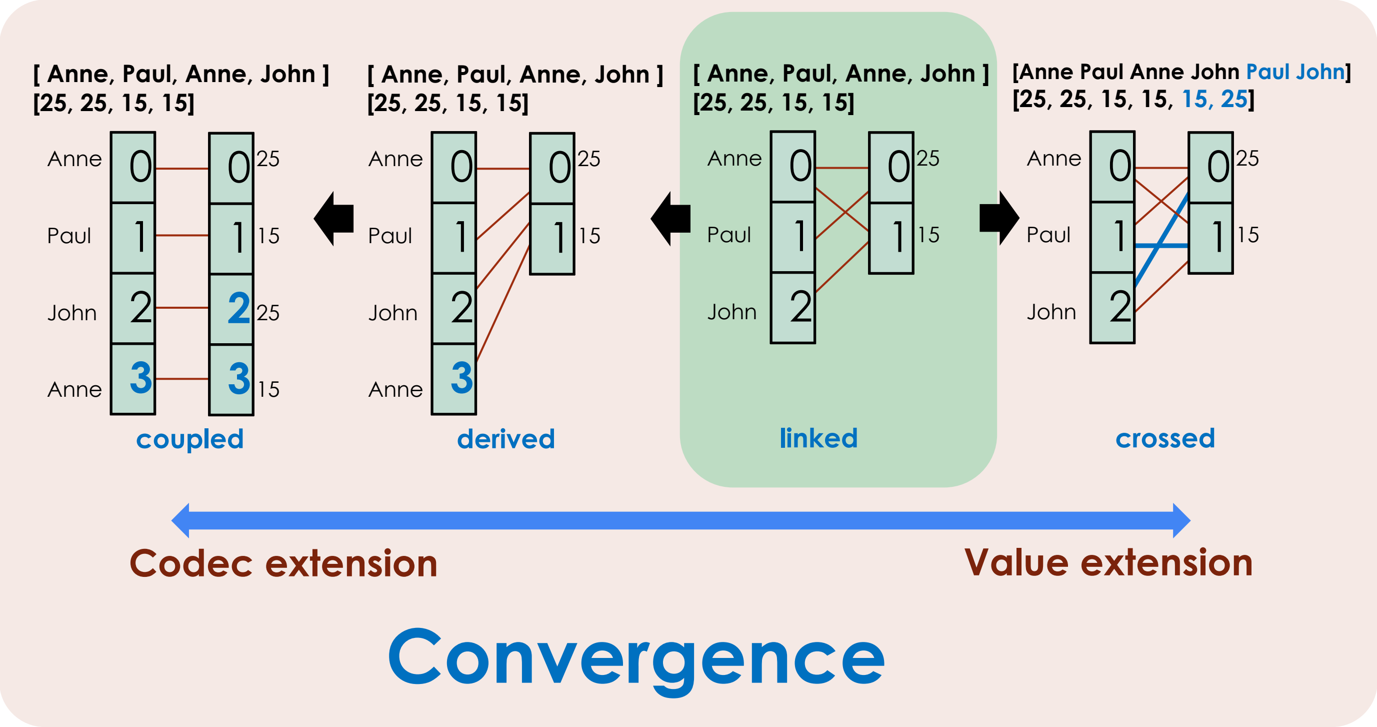
[Anne, Paul, John, Paul]
[25, 25, 15, 25]

[Anne, Paul, Anne, John]
[25, 25, 15, 15]

[Anne, Paul, Anne, Paul]
[25, 25, 15, 15]



Canonical structure



[Anne, Paul, Anne, John]
[25, 25, 15, 15]

[Anne, Paul, Anne, John]
[25, 25, 15, 15]

[Anne, Paul, Anne, John]
[25, 25, 15, 15]

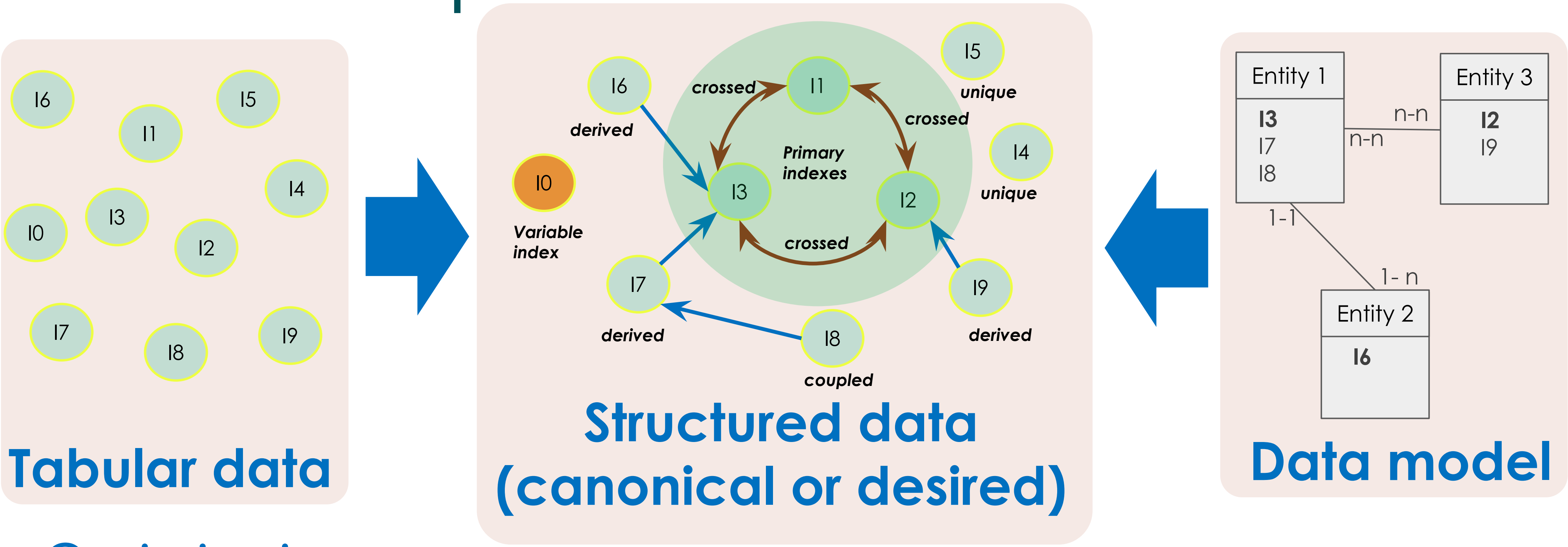
[Anne Paul Anne John Paul John]
[25, 25, 15, 15, 15, 25]

Codec extension

Value extension

Convergence

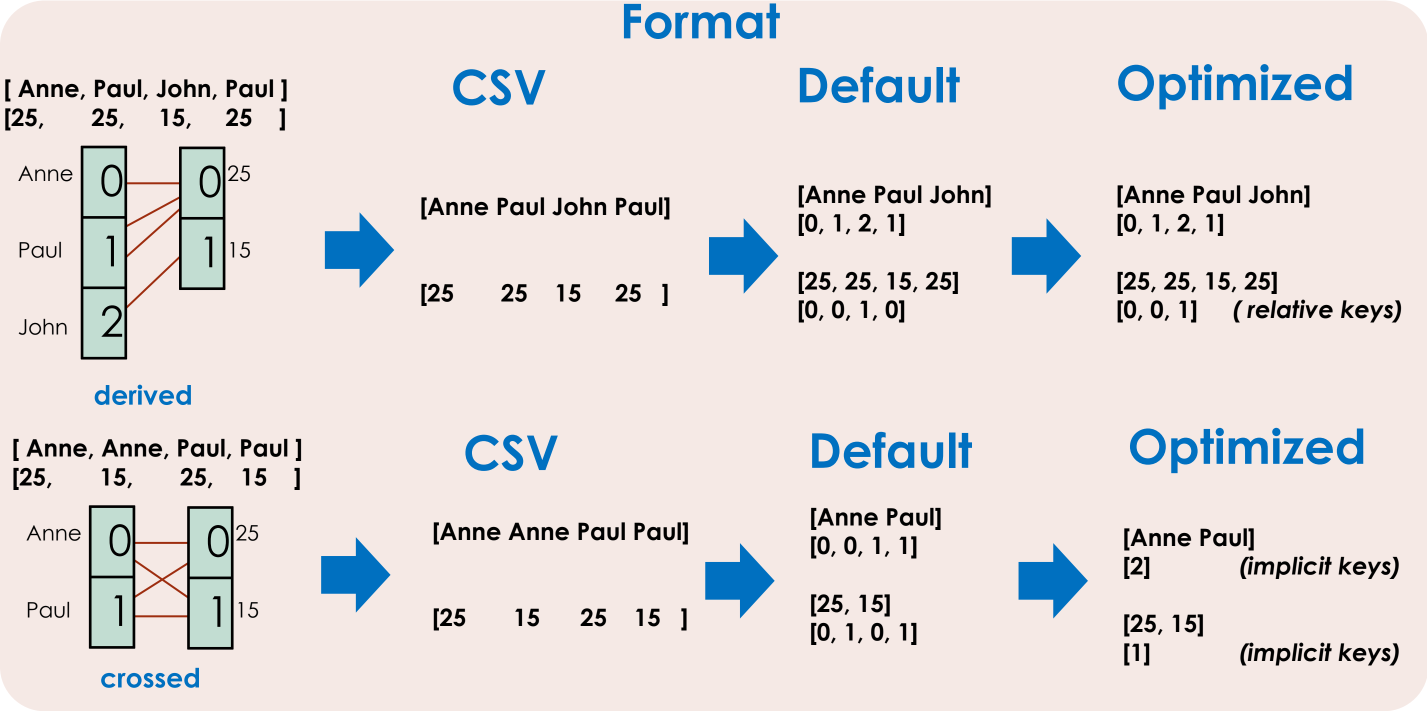
Structure optimization



- Optimization
  - minimization of additional data to achieve canonical structure
- Consistency
  - enforce compliance with the conceptual data model (e.g. cardinality)
  - identification of additional data to achieve the desired structure

Size optimization

- Canonical structure
  - Minimal structure
- Minimal size
  - No multiple value
  - Keys optimization
- Exchange format
  - Text : JSON format
  - Binary : CBOR (RFC 8949)



Format

[Anne, Paul, John, Paul]
[25, 25, 15, 25]

derived

CSV

[Anne Paul John Paul]

[25 25 15 25]

Default

[Anne Paul John]
[0, 1, 2, 1]

[25, 25, 15, 25]
[0, 0, 1, 0]

Optimized

[Anne Paul John]
[0, 1, 2, 1]

[25, 25, 15, 25]
[0, 0, 1] *(relative keys)*

[Anne, Anne, Paul, Paul]
[25, 15, 25, 15]

crossed

CSV

[Anne Anne Paul Paul]

[25 15 25 15]

Default

[Anne Paul]
[0, 0, 1, 1]

[25, 15]
[0, 1, 0, 1]

Optimized

[Anne Paul]
[2] *(implicit keys)*

[25, 15]
[1] *(implicit keys)*

Example : Open-data - french charging point (EVSE)

7.5 Mo – 11 000 rows – 49 columns

Analysis :

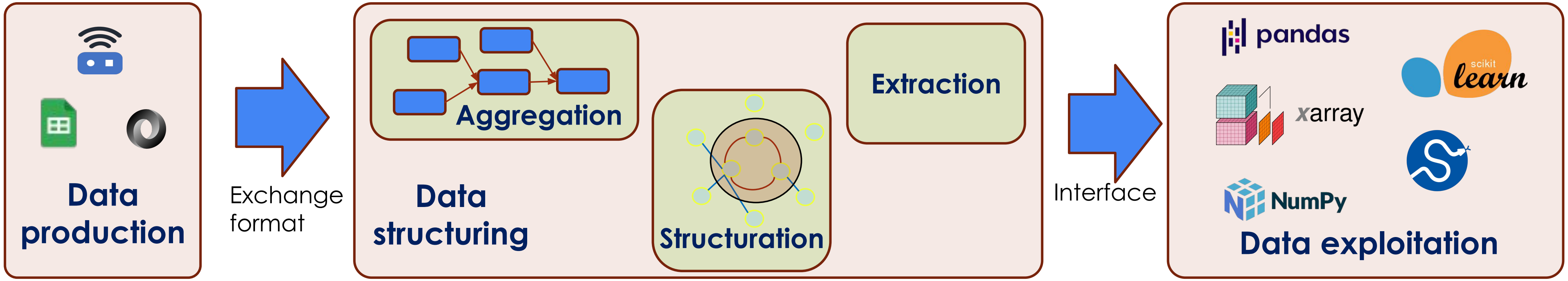
Indexes : 1 coupled, 6 derived, 1 crossed, 41 linked

Canonical format : 1 crossed, 48 derived

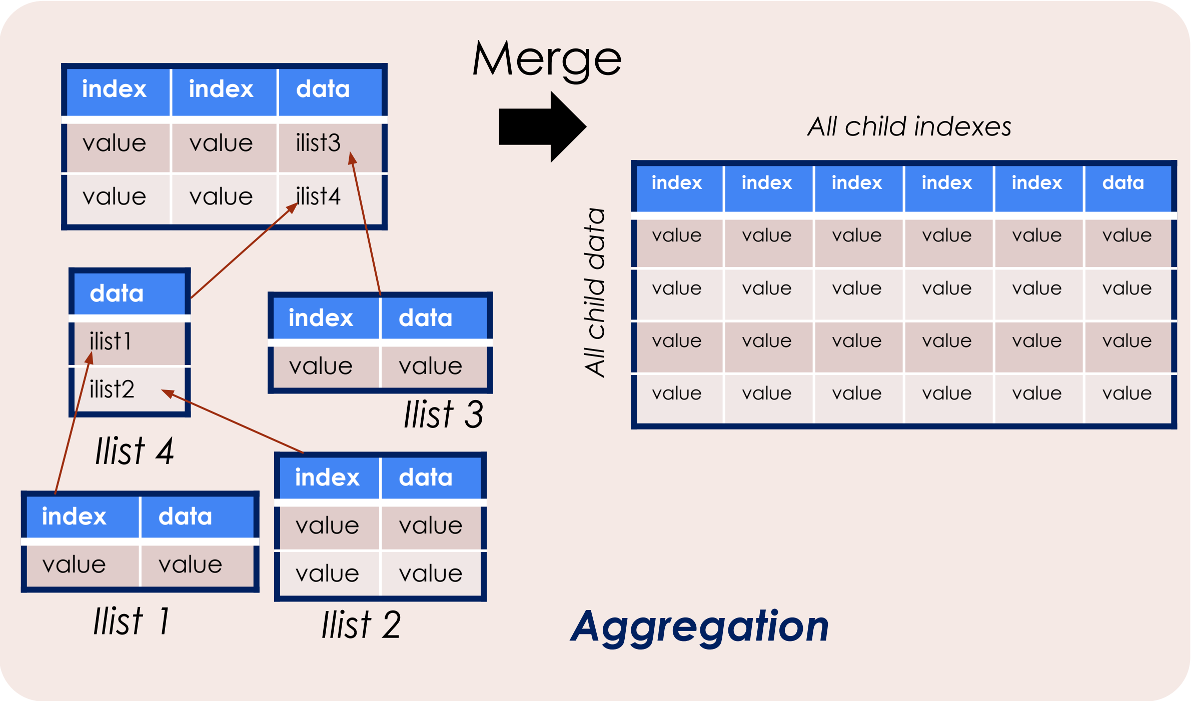
File size :

Default :	3.7 Mo
Optimized :	2.5 Mo
Binary optimized :	**1.7 Mo (gain : 77% !)**

Integrate process



- Data production interface
  - Exchange format (Json, Bluetooth, CSV)
- Aggregation / merge functions
  - Adapted to projects / organizations
  - Add information without altering
- Export to analysis tools
  - Canonical structure compatibility



index	index	data
value	value	ilist3
value	value	ilist4

data
ilist1
ilist2

index	data
value	value

index	data
value	value

index	data
value	value
value	value

index	index	index	index	index	data
value	value	value	value	value	value
value	value	value	value	value	value
value	value	value	value	value	value
value	value	value	value	value	value

loco

1 - Presentation

Semantic data - NTV format

- **Origin**
  - JSON-ND format defined in 2018 *(JSON with Named Data types)*
- **Structure**
  - NTV entity
    - Value : Data exchanged
    - Name : Interpretation or useful complement for understanding
    - Type : Nature of the data in a standard, catalog or software
  - JSON-NTV format (augmented JSON)
    - Primitive : Unique data (Value is a "JSON-value")
    - Structure : Composite data (Value is a list of NTV entities)

Unique data

43

Value : 43
Type : 'json' (default)
Name : None (default)

{ ':point' : [2.3, 48.9] }

Value : [2.3, 48.9]
Type : 'point'
Name : None (default)

{ 'Paris:point' : [2.3, 48.9] }

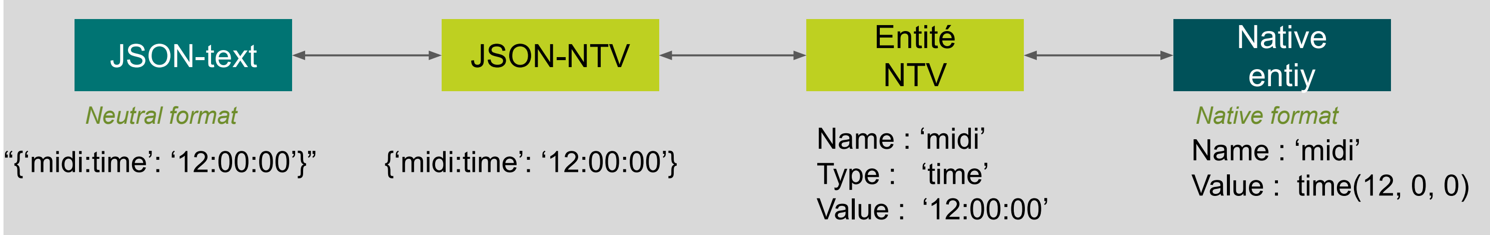
Value : [2.3, 48.9]
Type : 'point'
Name : Paris

Composite data

{ '::point' : [[2.3, 48.9], [4.8, 45.8]]

Value : [2.3, 48.9] et [4.8, 45.8]
Type : list of 'point'
Name : None (default)

{ 'city::point' : { 'paris: [2.3, 48.9], 'lyon': [4.8, 45.8] } }
or
{ 'city' : { 'paris:point': [2.3, 48.9], 'lyon:point': [4.8, 45.8] } }



"{'midi:time': '12:00:00'}"

{'midi:time': '12:00:00'}

Name : 'midi'
Type : 'time'
Value : '12:00:00'

Name : 'midi'
Value : time(12, 0, 0)

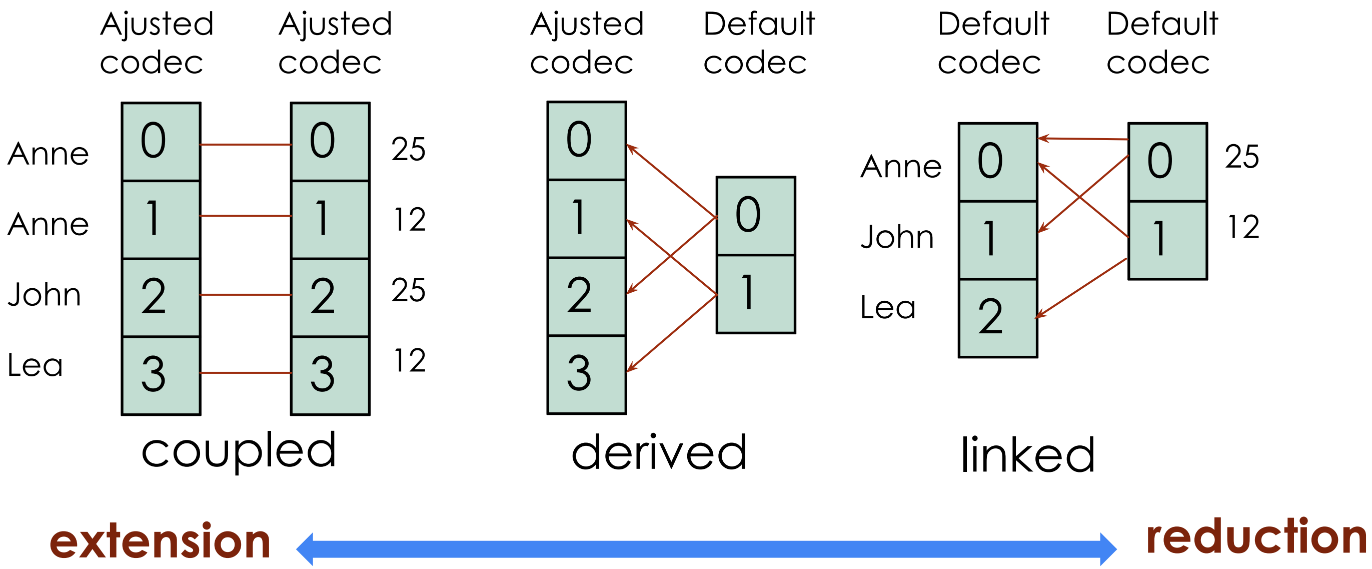
Relationship adjustment

- **Codec reduction / extension**
  - Codec changed
  - Values unchanged

Reduction is useful to minimize codec size

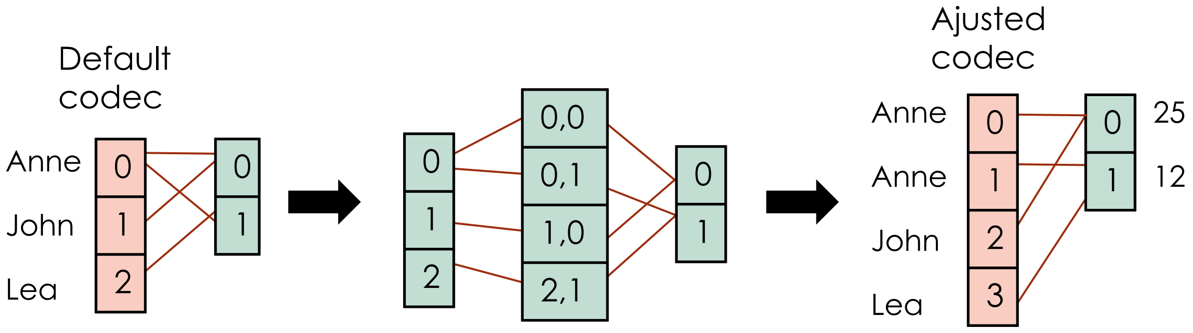
Extension is useful to identify incorrect data / relationship

[Anne, Anne, John, Lea]
[25, 12, 25, 12]



- **Codec adjustment**
  - Codec is adjusted to the other codec
  - Other index is derived or coupled to the adjusted index
  - If A is derived from B and if B is adjusted to C, A is still derived from B

Keys can be deduced from keys parent



loco

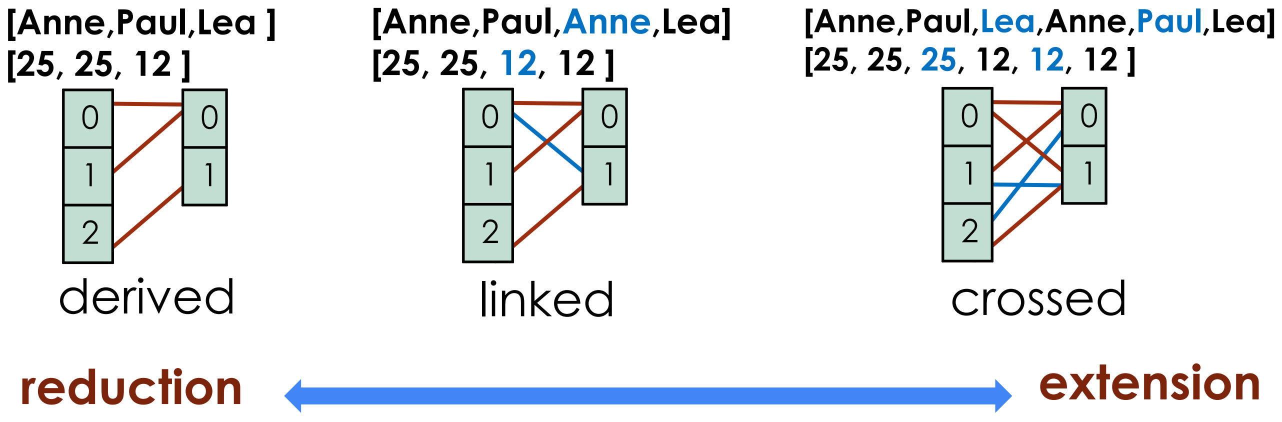
2 - Tools

Relationship adjustment

- **Values reduction / extension**
  - Codec unchanged
  - Values changed

Extension is useful to generate matrix

Reduction is useful to increase codec readability



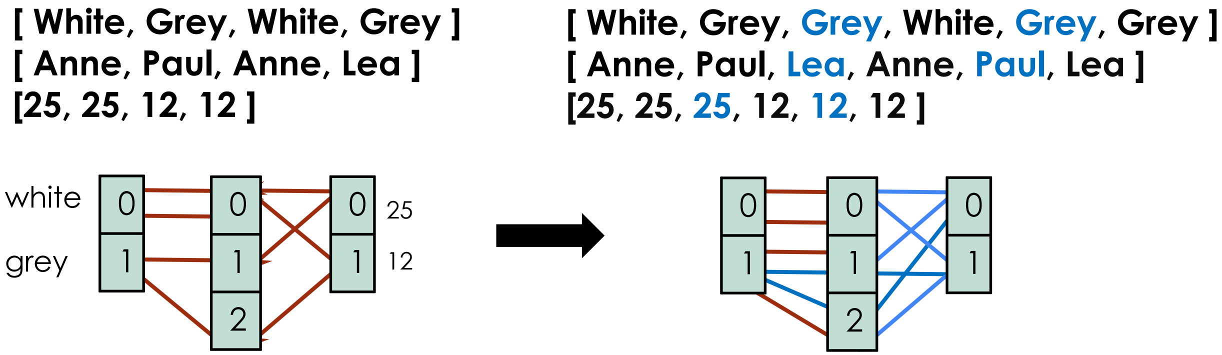
[Anne,Paul,Lea]
[25, 25, 12]

[Anne,Paul,Anne,Lea]
[25, 25, 12, 12]

[Anne,Paul,Lea,Anne,Paul,Lea]
[25, 25, 25, 12, 12, 12]

- **Propagation**
  - Values reduction / extension can be propagated to derived or coupled indexes

Extension can't be propagated to crossed or linked Indexes.

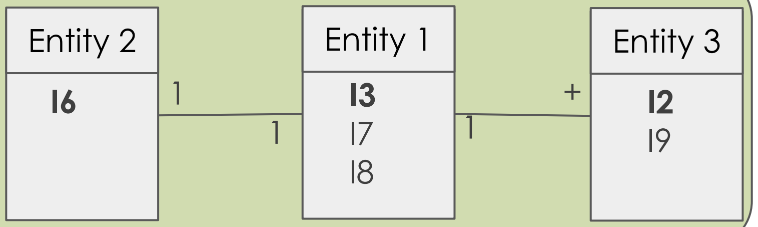


[White, Grey, White, Grey]
[Anne, Paul, Anne, Lea]
[25, 25, 12, 12]

[White, Grey, Grey, White, Grey, Grey]
[Anne, Paul, Lea, Anne, Paul, Lea]
[25, 25, 25, 12, 12, 12]

Relationship control

- **Data model**
  - Sets entities, attributes, relationships



Entity 2

I6

Entity 1

I3
I7
I8

Entity 3

I2
I9

- **Dataset**
  - Relationships between fields
  - Cardinality 1 - 1 (coupled), 1 - n (derived)

Entities :
I6 - I3 : coupled
I3 - I2 : derived

Attributes:
I7 - I3 : derived
I8 - I3 : derived
I9 - I2 : derived

- **Analysis**
  - Check relationships

Coupled
Coupling measure : distance
Distance = 0

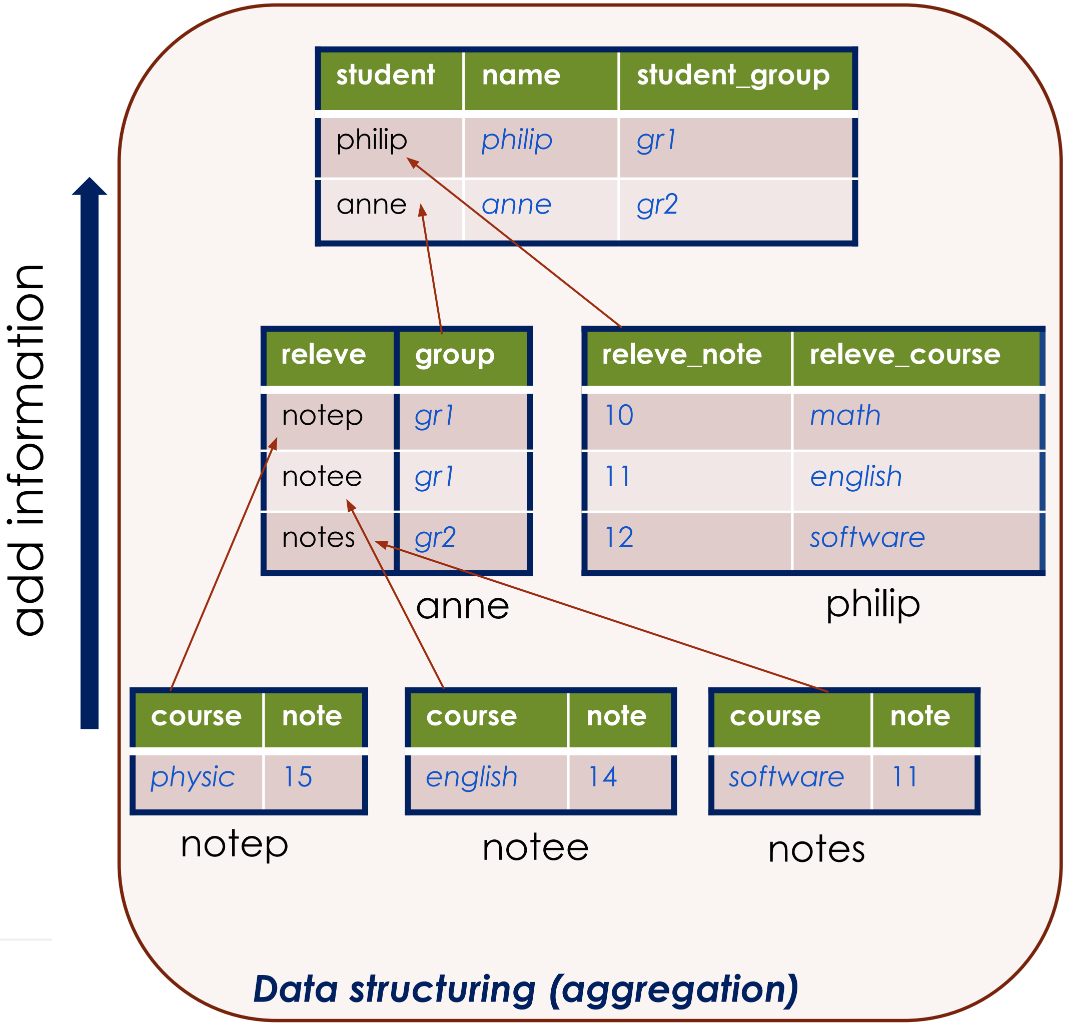
Derived
Deriving measure : distomin
Distomin = 0

- **Inconsistent data**
  - Identification of inconsistent values

Codec extension tools

Aggregation

Build



student	name	student_group
philip	*philip*	*gr1*
anne	*anne*	*gr2*

releve	group
notep	*gr1*
notee	*gr1*
notes	*gr2*

anne

releve_note	releve_course
10	*math*
11	*english*
12	*software*

philip

course	note
physic	15

notep

course	note
english	14

notee

course	note
software	11

notes

Data structuring (aggregation)

Merge

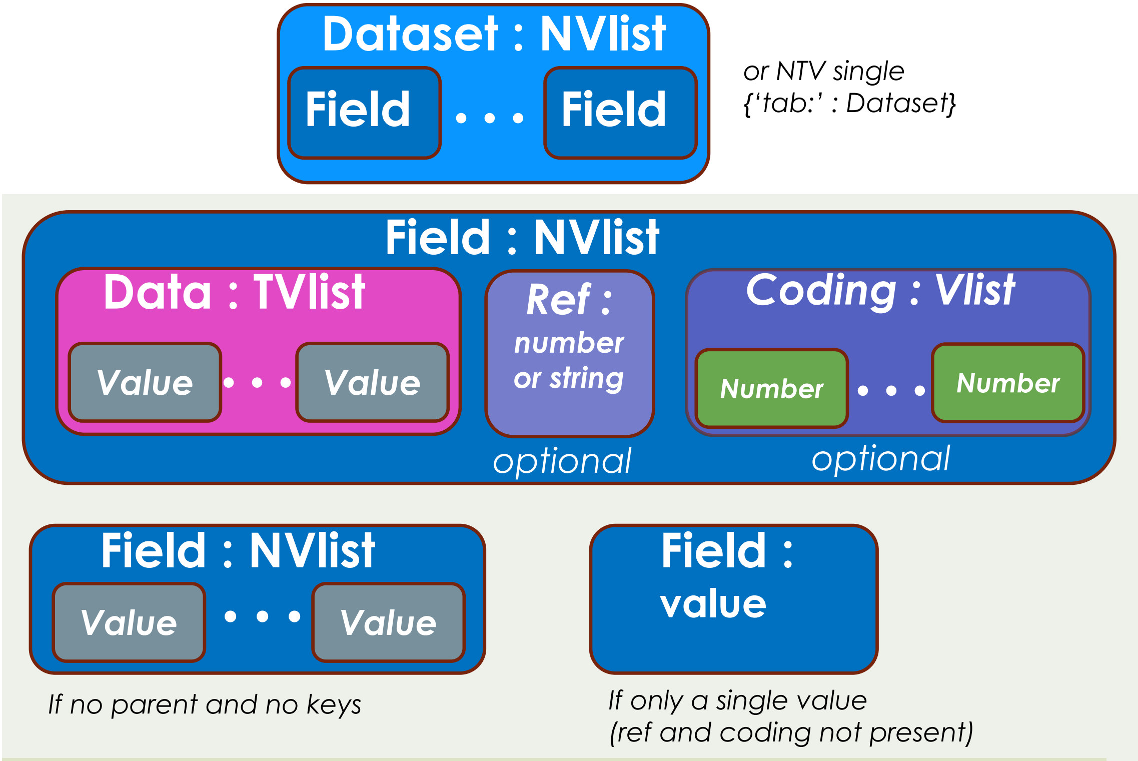
recursive

Use

student_releve_note	student_releve_course	name	student_group
10	*math*	*philip*	*gr1*
11	*english*	*philip*	*gr1*
12	*software*	*philip*	*gr1*
15	*physic*	*anne*	*gr1*
14	*english*	*anne*	*gr1*
11	*software*	*anne*	*gr2*

- **Process adapted to organizations**
- **Add information without altering**
- **Separation of structuring and use**

JSON Representation



Data is :
- The list of values (full format)
- Codec list (complete, implicit, relative, primary and unique format)
- Sparse and fill values (sparse format)

Coding is :
- An absolute Keys list (complete format)
- A relative Keys list (relative format)
- A coefficient (primary format)
- An index list (sparse format)

Ref is :
- Index or Name of parent Field (implicit or relative format)
- -1 (unique sparse format)

If Data contains one value, Data and value are merged
If Ref and Coding are not present, Data and Field are merged.

Field example:
Name : ‘team1’
Values : [‘Anne’, ‘Anne’, ‘John’, ‘Paul’, ‘John’]

- **Full format (without name)**
  [‘Anne’, ‘Anne’ ‘John’, ‘Paul’, ‘John’]
  -> Full codec (e.g. csv format)
- **Sparse format (without name)**
  [[‘Anne’, ‘Anne’, ‘Paul’, ‘John], [0,1,3]]
  -> Sparse values, index list
- **Complete format (with name)**
  {‘team1’ : [[‘Anne’, ‘John’, ‘Paul’], [0,0,1,2,1]] }
  -> Default codec, absolute keys
- **Implicit format (with name)**
  {‘team1’: [[‘Anne’, ‘John’, ‘Paul’, ‘John’], 2] }
  -> Codec, parent id
- **Relative format (with name)**
  {‘team1’ : [[‘Anne’, ‘John’, ‘Paul’], 2, [0,1,2,1]] }
  -> Codec, parent id, relative keys
- **Unique format**
  {‘team1’ : ‘Anne’ }(with name) ‘Anne’ (without name)
  -> Value

loco

3 - Example

Build

IndexSet Data

aw

course	year	examen	score
math	2021	t1	11
math	2021	t2	13
math	2021	t3	15
english	2021	t2	10
english	2021	t3	12

pw

course	year	examen	score
math	2021	t1	15
english	2021	t2	8

cr

course	year	examen	score
software	2021	t3	17
software	2021	t2	18
english	2021	t1	2
english	2021	t2	4

pb

course	year	examen	score
software	2021	t3	18
english	2021	t1	6

aggregation

total

first name	last name	full name	surname	group	student
Anne	White	Anne White	skyler	gr1	**aw**
Philippe	White	Philippe White	heisenberg	gr2	**pw**
Camille	Red	Camille Red	saul	gr3	**cr**
Philippe	Black	Philippe Black	gus	gr3	**pb**

merge

first name	last name	full name	surname	group	course	year	examen	score
Anne	White	Anne White	skyler	gr1	math	2021	t1	11
Anne	White	Anne White	skyler	gr1	math	2021	t2	13
Anne	White	Anne White	skyler	gr1	math	2021	t3	15
Anne	White	Anne White	skyler	gr1	english	2021	t2	10
Anne	White	Anne White	skyler	gr1	english	2021	t3	12
Philippe	White	Philippe White	heisenberg	gr2	math	2021	t1	15
Philippe	White	Philippe White	heisenberg	gr2	english	2021	t2	8
Camille	Red	Camille Red	saul	gr3	software	2021	t3	17
Camille	Red	Camille Red	saul	gr3	software	2021	t2	18
Camille	Red	Camille Red	saul	gr3	english	2021	t1	2
Camille	Red	Camille Red	saul	gr3	english	2021	t2	4
Philippe	Black	Philippe Black	gus	gr3	software	2021	t3	18
Philippe	Black	Philippe Black	gus	gr3	english	2021	t1	6

loco

3 - Example

Analysis

3 fields are derived

- First name
- Last name
- Group

1 field is coupled

- Surname

1 field is unique

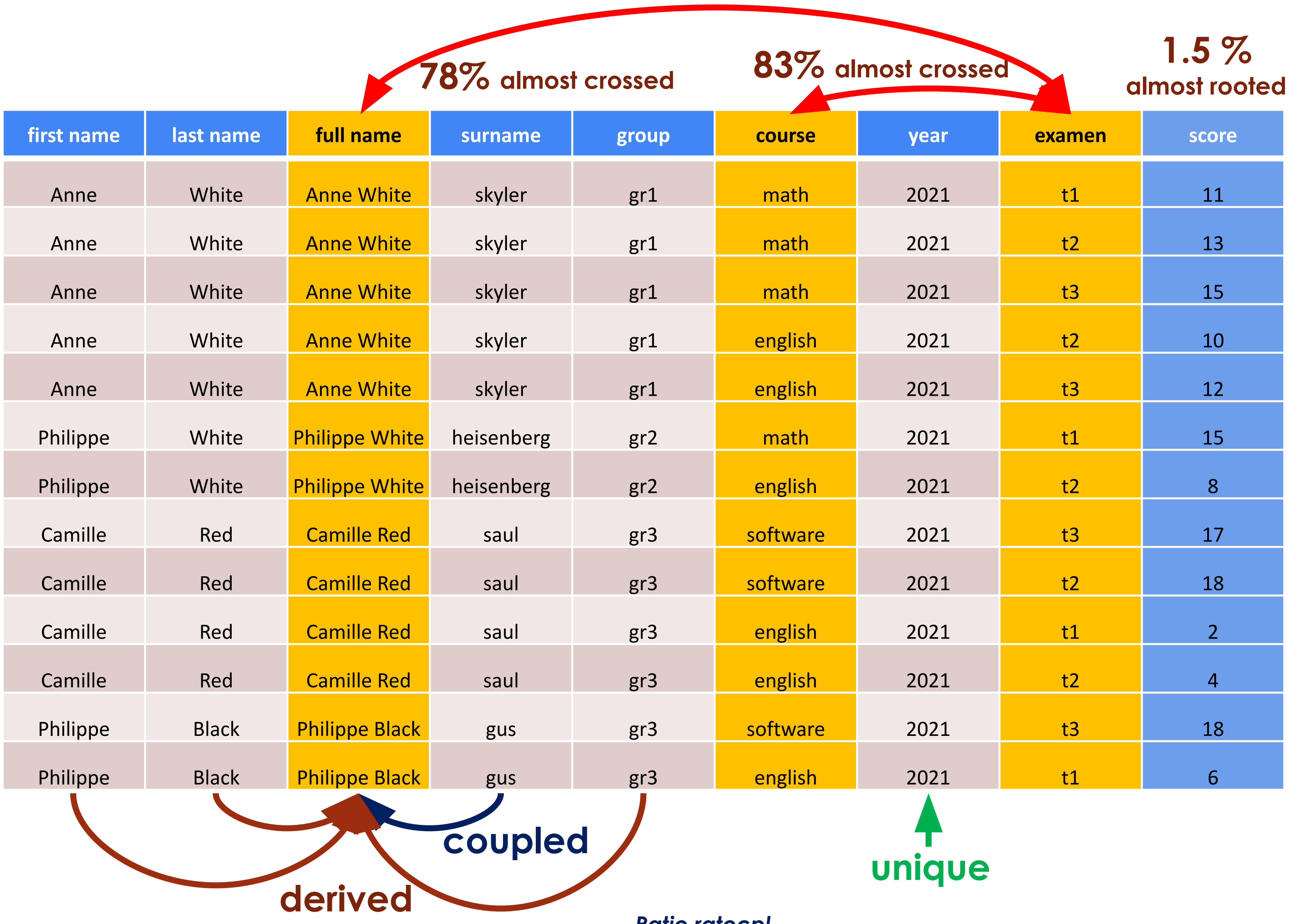
- Year

3 fields are almost crossed

- Full name
- Course
- Examen

1 field is almost rooted

- Score



first name	last name	full name	surname	group	course	year	examen	score
Anne	White	Anne White	skyler	gr1	math	2021	t1	11
Anne	White	Anne White	skyler	gr1	math	2021	t2	13
Anne	White	Anne White	skyler	gr1	math	2021	t3	15
Anne	White	Anne White	skyler	gr1	english	2021	t2	10
Anne	White	Anne White	skyler	gr1	english	2021	t3	12
Philippe	White	Philippe White	heisenberg	gr2	math	2021	t1	15
Philippe	White	Philippe White	heisenberg	gr2	english	2021	t2	8
Camille	Red	Camille Red	saul	gr3	software	2021	t3	17
Camille	Red	Camille Red	saul	gr3	software	2021	t2	18
Camille	Red	Camille Red	saul	gr3	english	2021	t1	2
Camille	Red	Camille Red	saul	gr3	english	2021	t2	4
Philippe	Black	Philippe Black	gus	gr3	software	2021	t3	18
Philippe	Black	Philippe Black	gus	gr3	english	2021	t1	6

Ratio ratecpl

- *Full name – Examen : 78 %*
- *Score – Root : 1,5 %*
- *Course – Examen : 83 %*

loco

3 - Example

Uses

Values extension

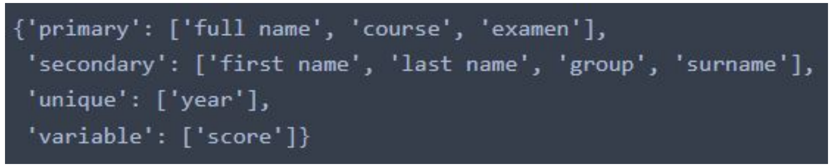
- Full name
- Course
- Examen

completed

first name	last name	full name	surname	group	course	year	examen	score
Anne	White	Anne White	skyler	gr1	english	2021	t2	10
Anne	White	Anne White	skyler	gr1	english	2021	t3	12
Anne	White	Anne White	skyler	gr1	math	2021	t1	11
Anne	White	Anne White	skyler	gr1	math	2021	t2	13
Anne	White	Anne White	skyler	gr1	math	2021	t3	15
Anne	***White***	***Anne White***	***skyler***	***gr1***	***software***	***2021***	***t1***	-
Anne	***White***	***Anne White***	***skyler***	***gr1***	***software***	***2021***	***t2***	-
Anne	***White***	***Anne White***	***skyler***	***gr1***	***software***	***2021***	***t3***	-
Anne	***White***	***Anne White***	***skyler***	***gr1***	***english***	***2021***	***t1***	-

Analysis

- Partition

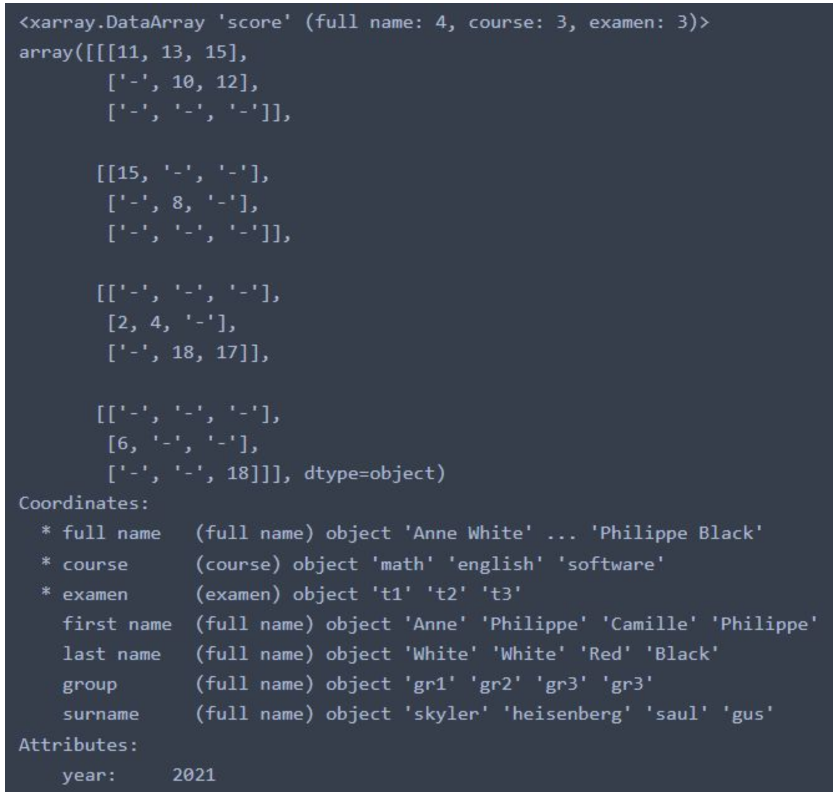


```
{'primary': ['full name', 'course', 'examen'],
 'secondary': ['first name', 'last name', 'group', 'surname'],
 'unique': ['year'],
 'variable': ['score']}
```

Interface

- Export Xarray

Multi dimensional tool



```
<xarray.DataArray 'score' (full name: 4, course: 3, examen: 3)>
array([[[11, 13, 15],
        ['-', 10, 12],
        ['-', '-', '-']],

       [[15, '-', '-'],
        ['-', 8, '-'],
        ['-', '-', '-']],

       [['-', '-', '-'],
        [2, 4, '-'],
        ['-', 18, 17]],

       [['-', '-', '-'],
        [6, '-', '-'],
        ['-', '-', 18]]], dtype=object)
Coordinates:
  * full name   (full name) object 'Anne White' ... 'Philippe Black'
  * course      (course) object 'math' 'english' 'software'
  * examen      (examen) object 't1' 't2' 't3'
    first name  (full name) object 'Anne' 'Philippe' 'Camille' 'Philippe'
    last name   (full name) object 'White' 'White' 'Red' 'Black'
    group       (full name) object 'gr1' 'gr2' 'gr3' 'gr3'
    surname     (full name) object 'skyler' 'heisenberg' 'saul' 'gus'
Attributes:
    year:     2021
```