

# **Knowledge Continuous Integration Process (K-CIP)**

**Hala Skaf-Molli, Emmanuel Desmontils,  
Pascal Molli, et al.**

Hala.Skaf@univ-nantes.fr

Nantes University, Lina

GDD Team

France

# Context

- In Social Semantic Web, **Information, Ontology** and **Queries** are mixed in the same space.
  - Semantic Wikis
  - **Wikidata ??**
- How can I modify the ontology without breaking the queries ?
- When I modify the ontology:
  - I want to know the impact of modifications on the queries
  - I want to ensure the non regression of the system

# Issues

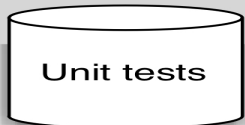
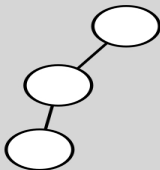
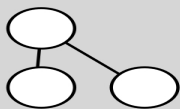
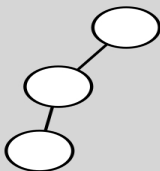
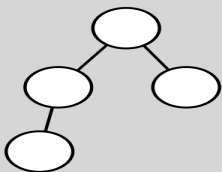
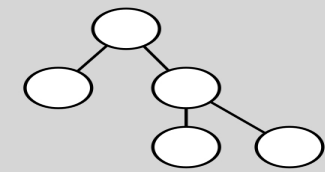
- Does the lightweight editing model of Social Web such as Wikipedia fit for ontologies editing ?
  - A small change in type definitions can break all semantic requests...
- How to manage ontology evolution in the context of Social Semantic Web ?

# Approach

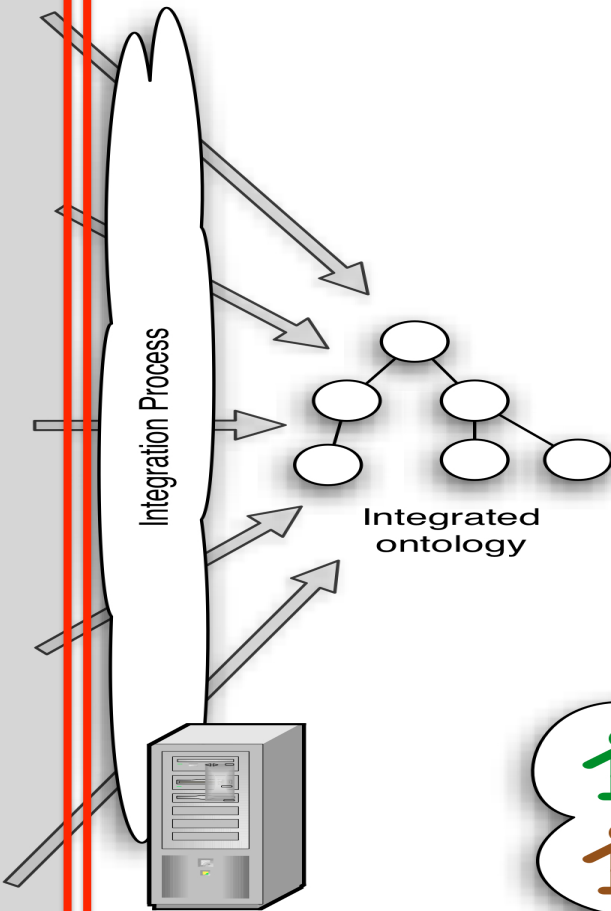
- Apply agile software development techniques to ontology engineering
  - Continuous integration (CI) processes
- **K-CIP** for ontology engineering in man-machine collaboration



Local ontologies

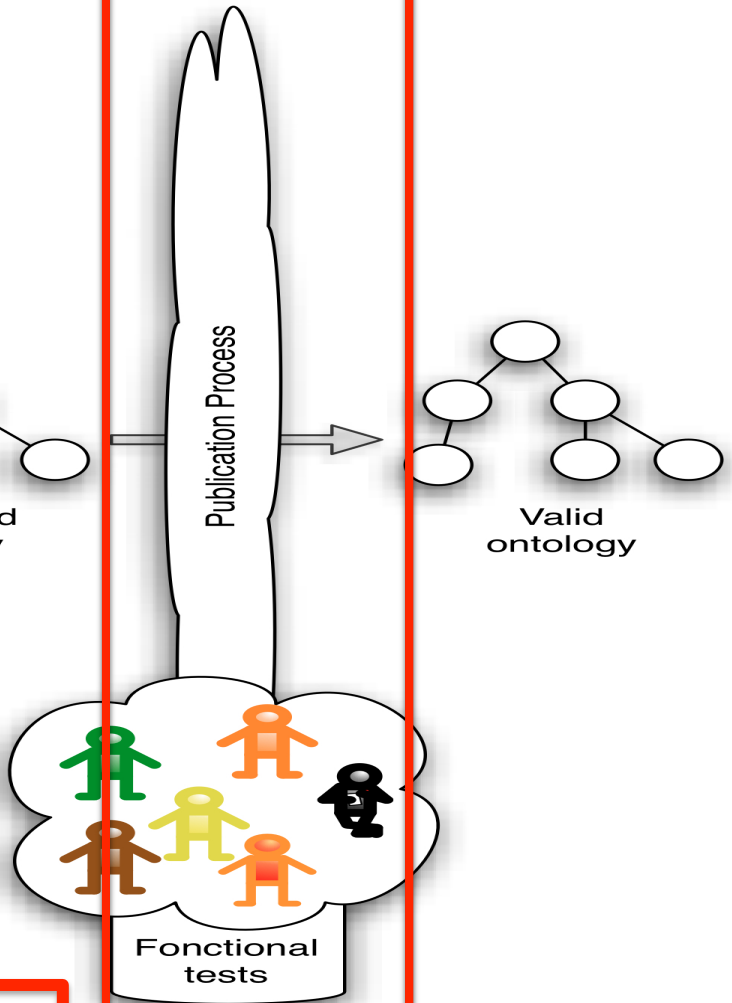


Integration Process



Integrated ontology

Publication Process



Valid ontology

Fonctional tests

**Tests ???**

# Tests

- Ensure that a modification of the ontology does not alter the system behavior
  - Which language to define tests ?
  - How to collect test data ?
  - How to write tests ?
  - When and where to execute tests ?
  - When and where to change the ontology ?

# Defining Tests

- A test  $T$  is defined as a set of assertions  $\{A_i\}$  on the results set  $R_{Q_T}$  for a given query  $Q_T$ 
  - $(Q_T, \{A_i\})$
- Assertions are defined as logical expressions using set operations on  $R_{Q_T}, R^+, R^-, R^?$ 
  - $R^+$ : a set of relevant answers
  - $R^-$ : a set of irrelevant answers
  - $R^?$ : a set of unknown answers

# Collecting Test Data

- **$R^+$ ,  $R^-$  and  $R^?$  are collected from the social feedback**
- When a user queries the system, according to the answer, she can decide to:
  - Agree : added to  $R^+$
  - Not agree: added to  $R^-$
  - Do not know : added to  $R^?$



# Writing Tests

- A modification does not reduce the positive answers of the query:

$$\text{Assert}( R^+ \subseteq R_Q )$$

- A modification does not change the positive answers of the query:

$$\text{Assert}( R^+ = R_Q )$$

- A modification does not introduce unwanted results for a query

$$\text{Assert}( R_Q \cap R^- = \emptyset )$$

- More positive answers than unwanted ones negative

$$\text{Assert}( | R^+ \cap R_Q | > | R^- \cap R_Q | )$$

# Example: Tests in WikiTaaable

- WikiTaaable is a semantic wiki that solves cooking problems using case-based reasoning engine
  - “I want a dessert with rice and figs”
  - Existing recipe with rice and figs as ingredients
  - Or an adaption of an existing recipe



I want a dessert with rice and fig

dessert\_dish rice fig

Find recipes!

Clear

**Dietary practices:**  Vegetarian  Nut-free  No alcohol  Low cholesterol  Gout Diet

[Customize your dietary practices...](#)

[Adapt a specific recipe...](#)

**Example.** If you want an apple pie without cinnamon, enter "apple pie\_dish -cinnamon". [Learn more about advanced queries...](#)



dessert\_dish rice fig

Find recipes!

Clear

[Customize your dietary practices...](#)

**Dietary practices:**  Vegetarian  Nut-free  No  
alcohol  Low cholesterol  Gout Diet

[Adapt a specific recipe...](#)

*Example.* If you want an apple pie without cinnamon, enter "apple pie\_dish -cinnamon".

[Learn more about advanced queries...](#)

Your request is: **dessert\_dish fig rice**

The request used for adaptation is: **dessert\_dish fig rice**

# **Original recipe name (click to open recipe)**

**Adaptation overview (click to see the details)**

1 [Glutinous rice with mangoes](#)

[Replace: Mango by Fig](#)

# Social Feedback for Collecting Test Data

## Glutinous rice with mangoes

---

**The ingredient substitutions**

OK

not OK

1. Mango → Fig

# Test Data Collecting in WikiTaaable

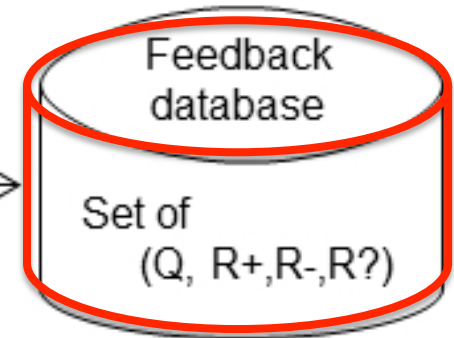
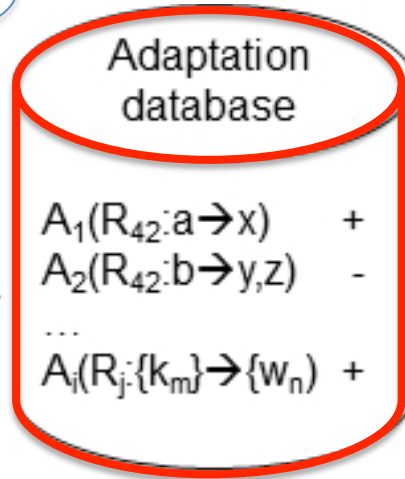
I want a dessert with rice and fig

Taaable

dessert\_dish rice fig

Dietary practices:  Vegetarian  Nut-free  No alcohol  Low cholesterol  Gout Diet [Adapt a specific recipe...](#) [Customize your dietary practices...](#)

Example. If you want an apple pie without cinnamon, enter 'apple pie\_dish -cinnamon'.

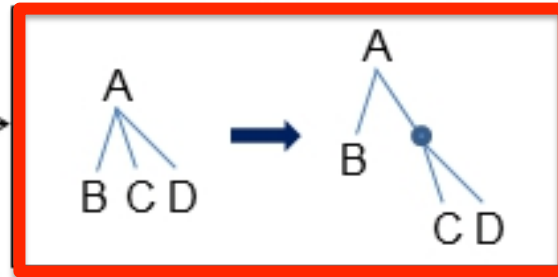


(1)



(4)

Ontology modifications



(5)

(6)



Non regression test

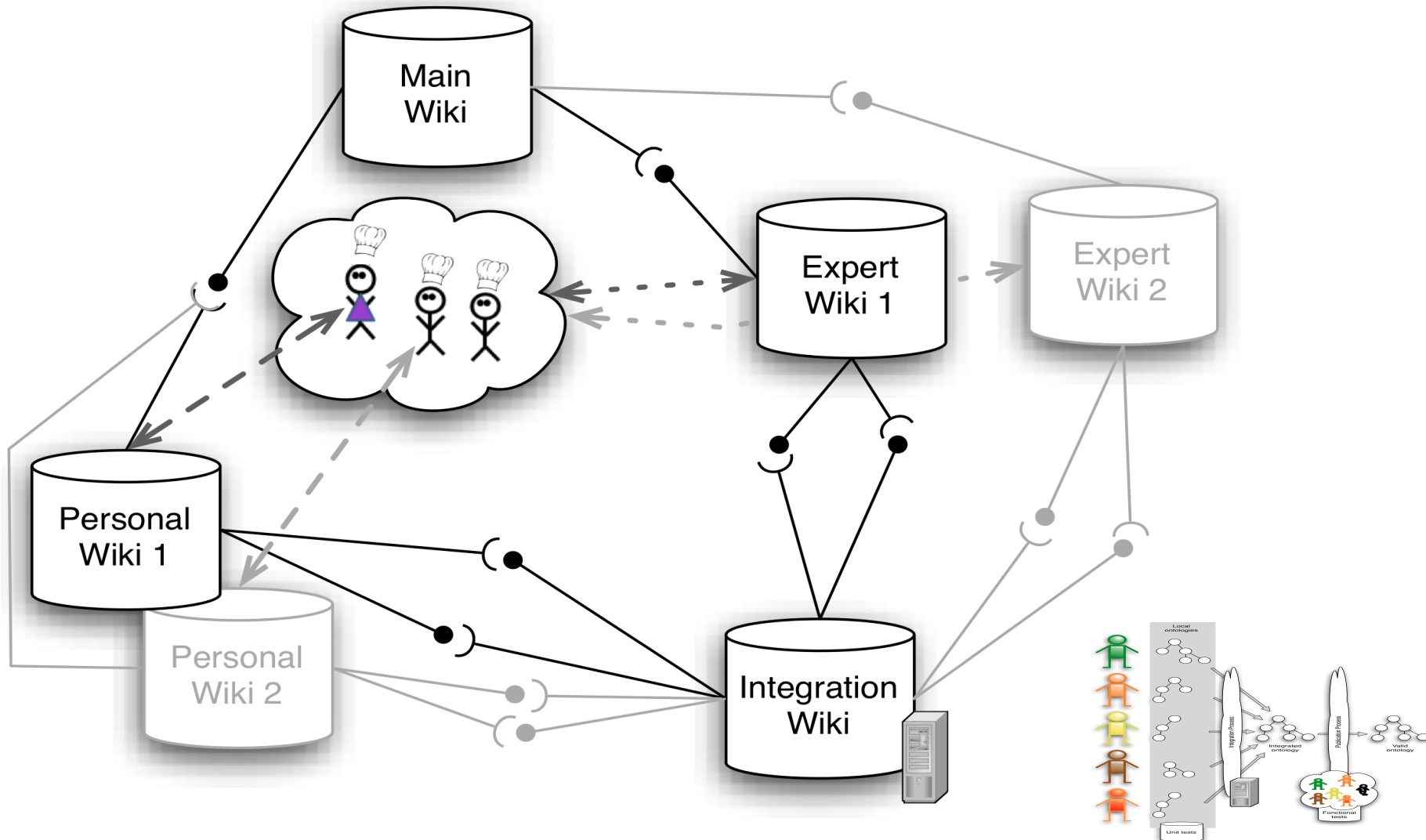
Notification ← FAILURE

SUCCESS

# Tests

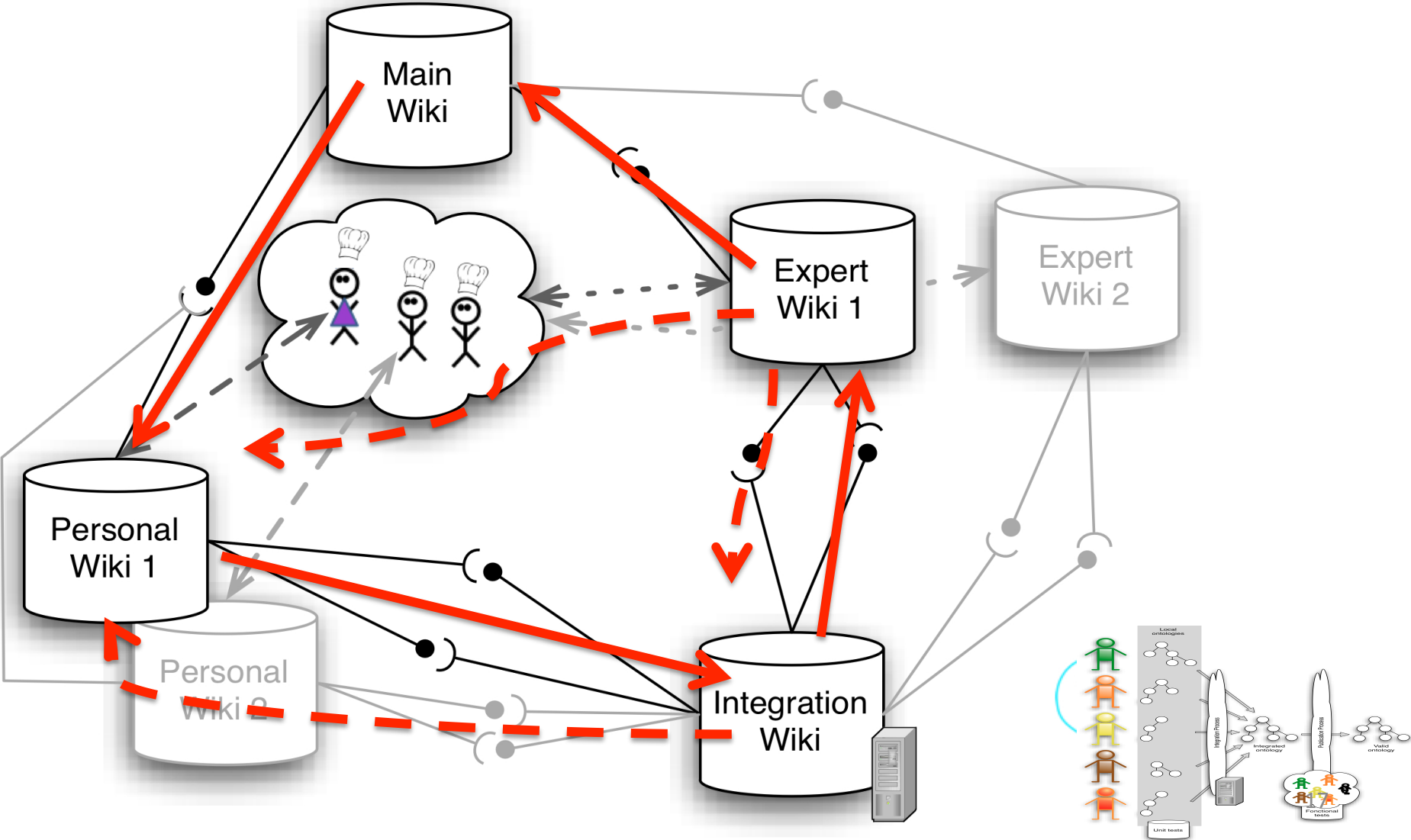
- Ensure that a modification of the ontology does not alter the system behavior
  - Which language to define tests ?
  - How to collect test data ?
  - How to write tests ?
  - When and where to execute tests ?
  - When and where to change the ontology ?

# WikiTaaable in Distributed Semantic Wiki (DSMW)

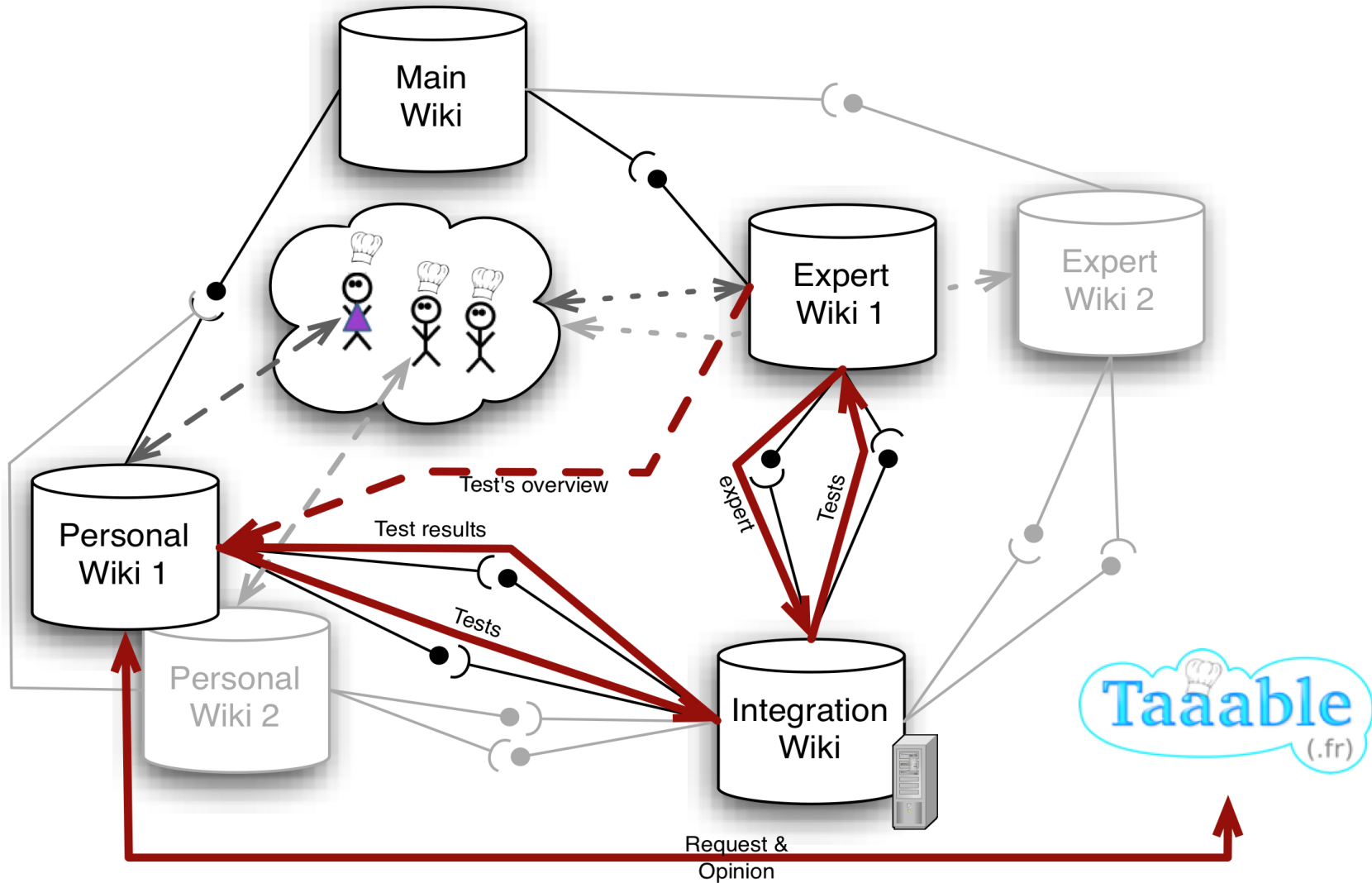




# Changes Propagation in K-CIP in Distributed WikiTaaable



# Tests Propagation in K-CIP

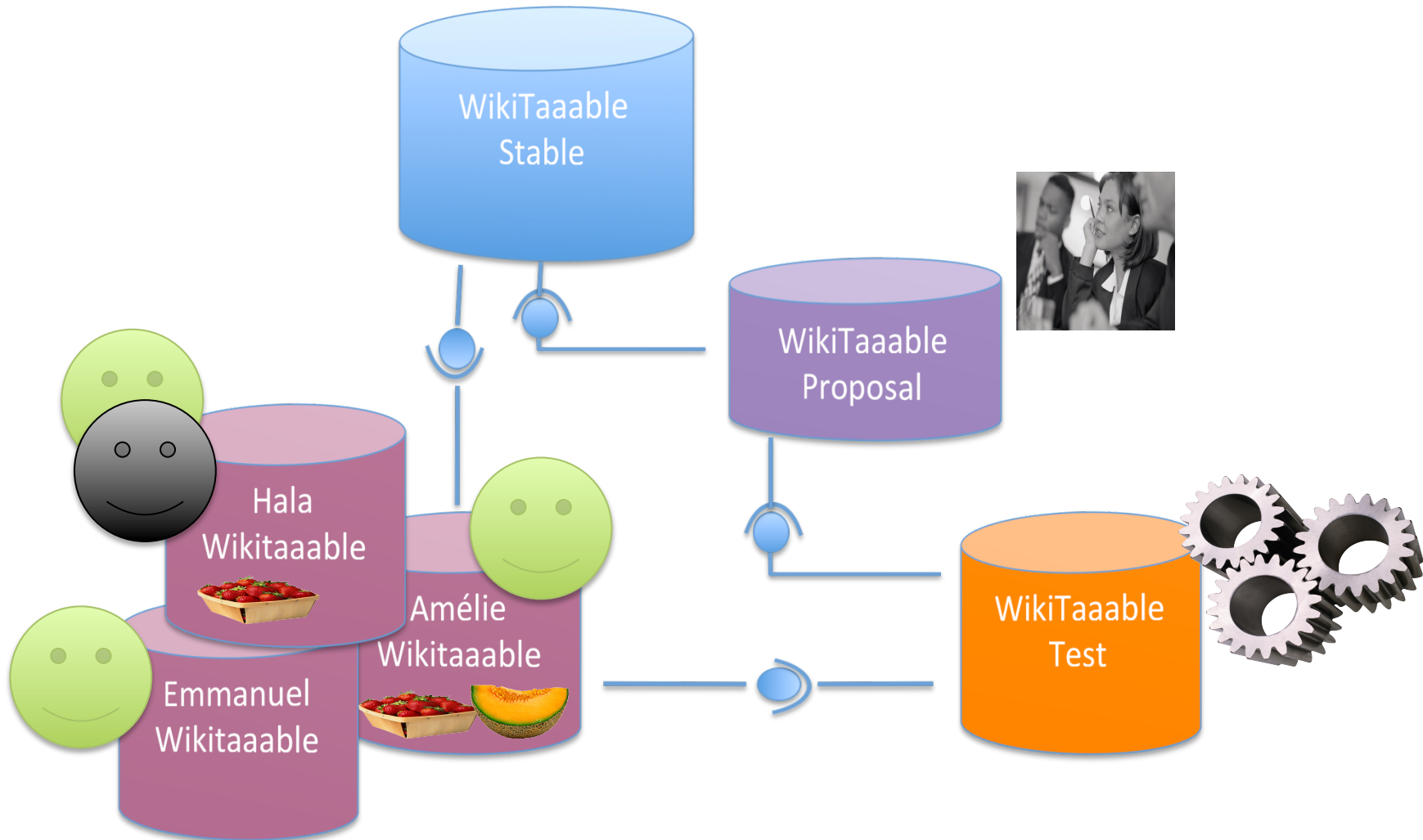


# Discussion

- K-CIP is a continuous integration process for ontology evolution in the Social Semantic Web
  - Takes advantage of social feedbacks to collect test data
  - Example with distributed WikiTaaable in DSMW
- The same process can be applied to Semantic Wikipedia
  - Replacing Wikipedia+DBPedia by a Semantic Wikipedia
- **Wikidata ??**
- Large scale experimentation
- Management of negative feedback:
  - Example: in case of disagreement, we might want to keep all opinions -> parallel versions of resources (reflect diversity), while ensuring the overall system will continue to work

Questions ?

# K-CIP in Distributed WikiTaable using DSMW

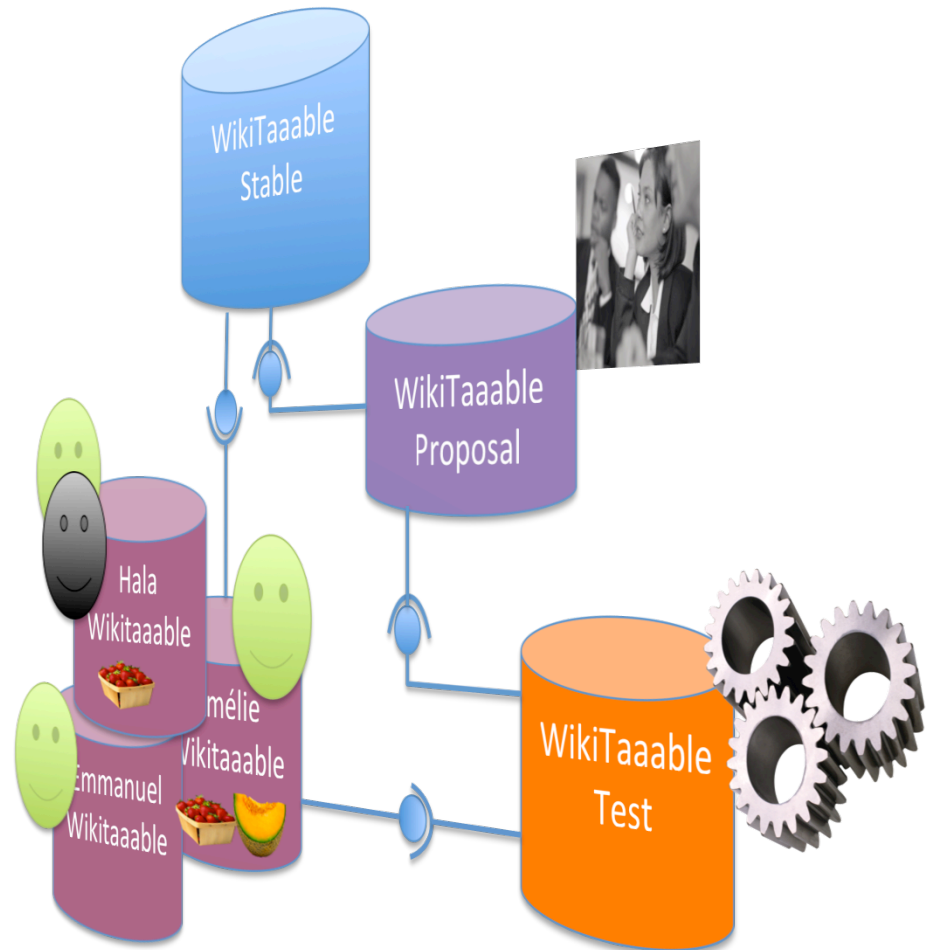


# Issues

- How to define tests ?
- How to collect ?

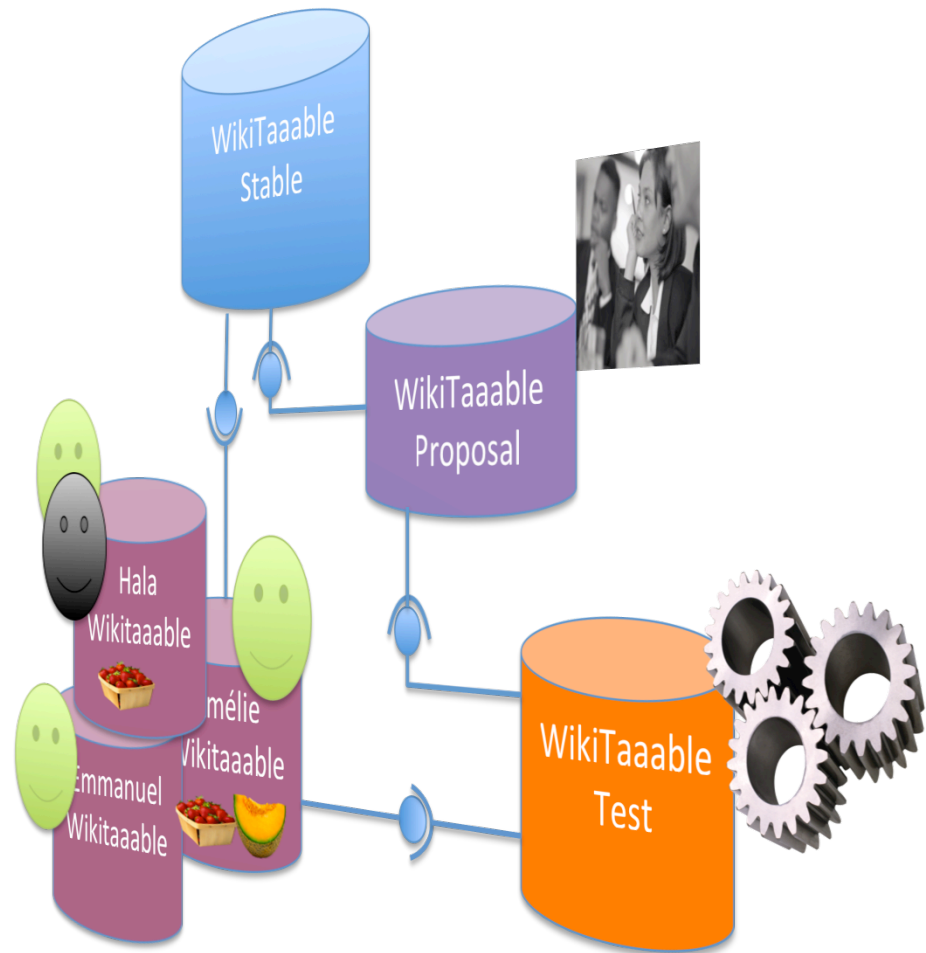
# WikiTaaable Stable

- Public “consistent” version of WikiTaaable
- Developers start by make a local copy (partial or total) of the WikiTaaable Stable
- Wikitaaable stable plays the role of the blessed repo in software engineering.



# In My local WikiTaaable

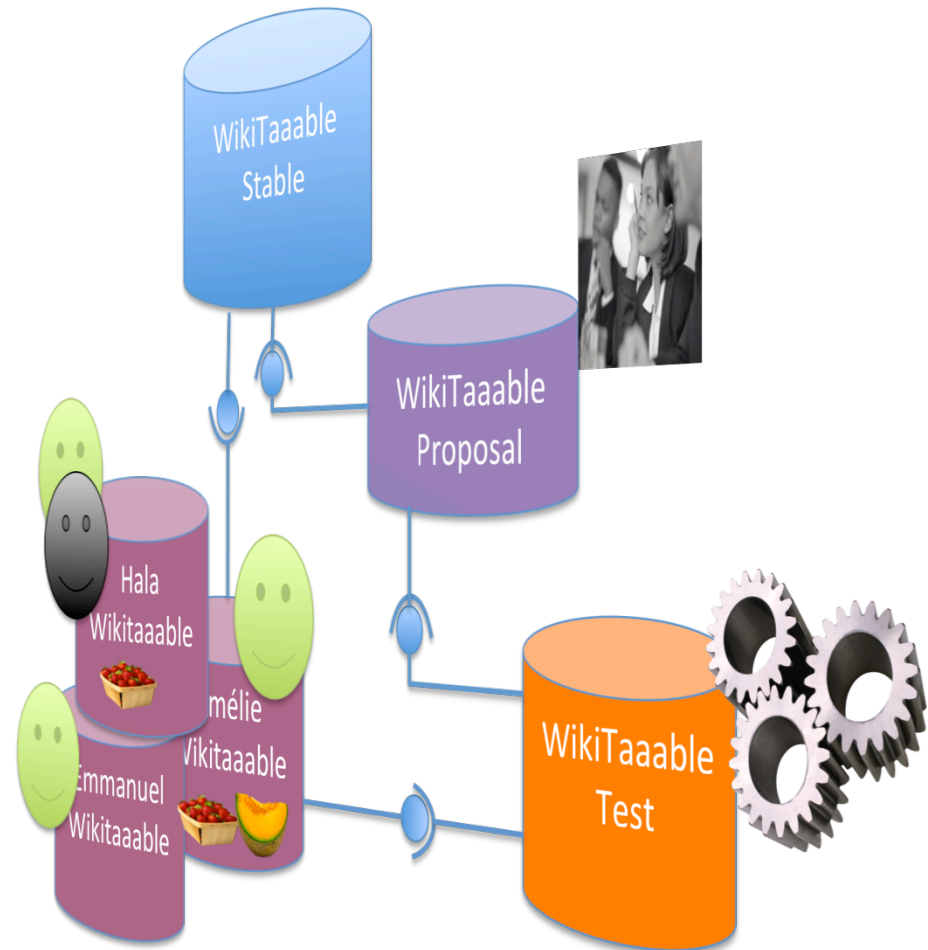
- Extend and modify the ontology
  - Construction and *modification SILEX scenario*
- Test locally the ontology
  - Question: which tests ? Which language to write the tests ?
  - How to add new tests ?
- I can pull updates from Stable and from others
  - *Fusion Silex scenario*
  - In DSMW the merging is done automatically by the Logoot algorithm (without the user interaction)
- I can make my modifications available for others “push modifications” but mainly to WikiTaaable Test





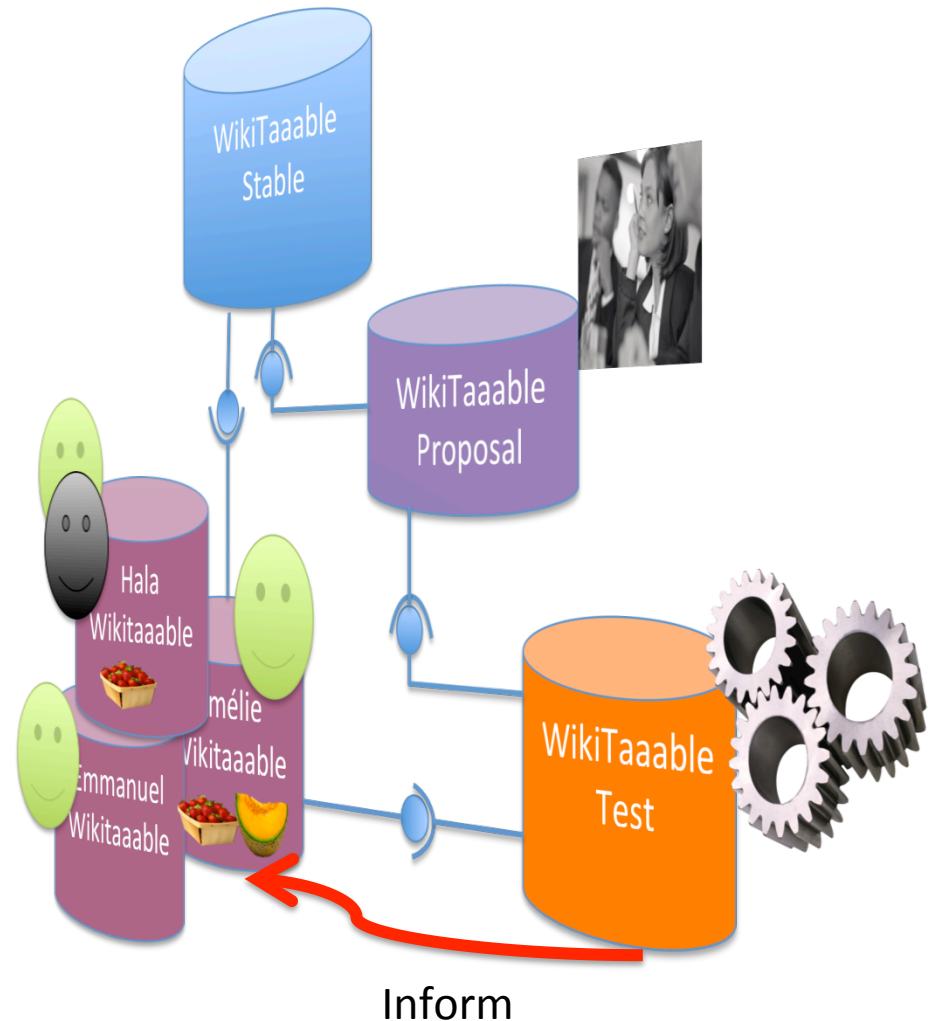
# WikiTaaable Test

- Automatic agent will pull modifications and run tests
  - Test can be failed du to concurrent activities
- Which kind of tests?
  - Basic tests:
    - Consistency ? (*Task3: Interactions and inferences*)
  - Functional tests
    - The queries produce the expected results
    - The initial tests sets: can be the set of queries proposed by the CCC
    - How to run and assert the results automatically ?
      - SMWUnit ?



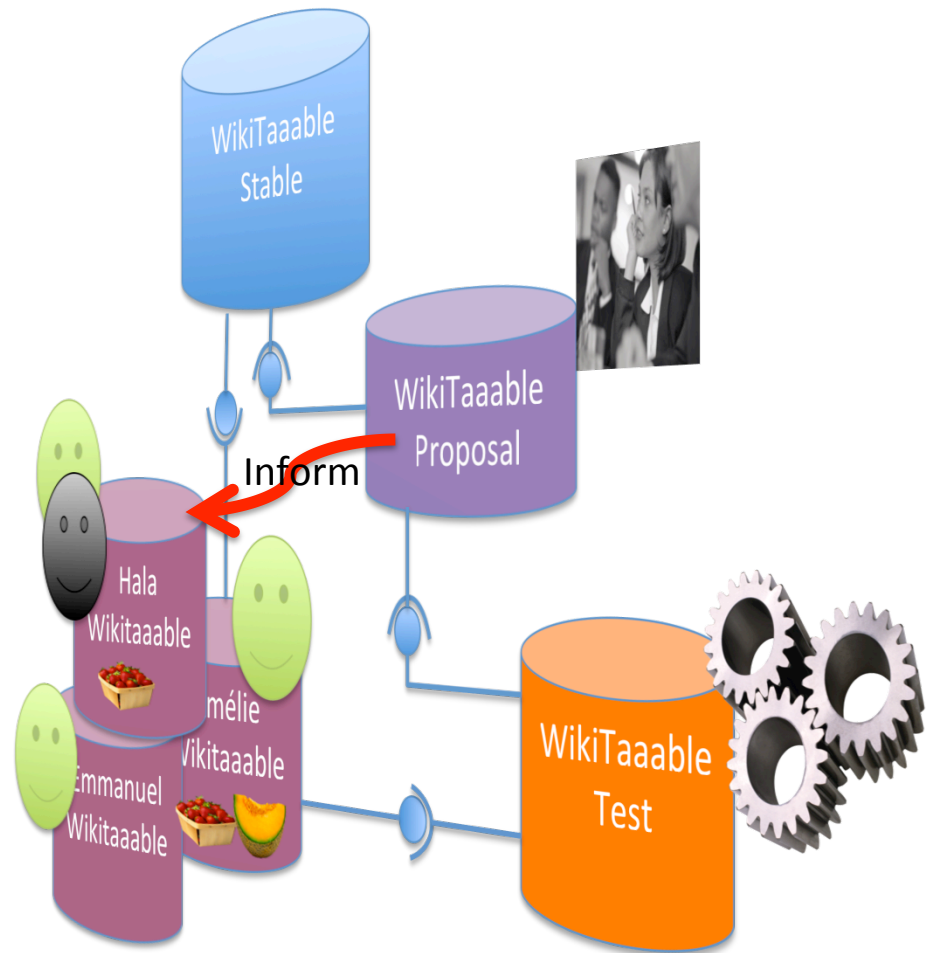
# WikiTaaable Test

- Inform the developers about the issue of their modifications
- If there is a problem, we have to explain (for human and machine) why, where and which parts of the knowledge are involved in the problem (*Task 4: Trace and Explication*)
- How to do that ?
  - Real-time semantic notifications
  - Notification can be seen as inputs in discussion pages for en user...



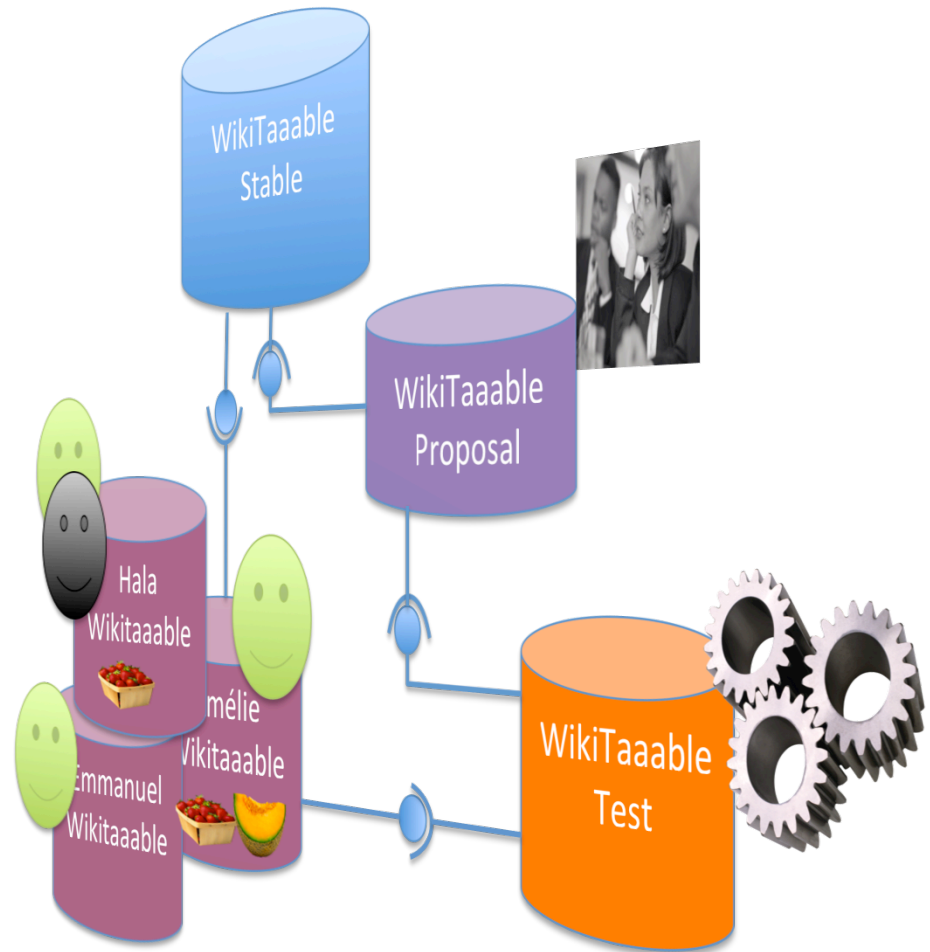
# WikiTaaable Proposal

- All automatic tests are OK
- We need the validation of "domain experts"
- If they agree, the modifications are integrated into "WikiTaaable Stable"
- Else, they call "*Undo function of DSMW*" to cancel modifications, they inform the concerned developer (again with explication)



# WikiTaaable Stable

- Developers synchronize their local copy with the “*WikiTaaable Stable*” server

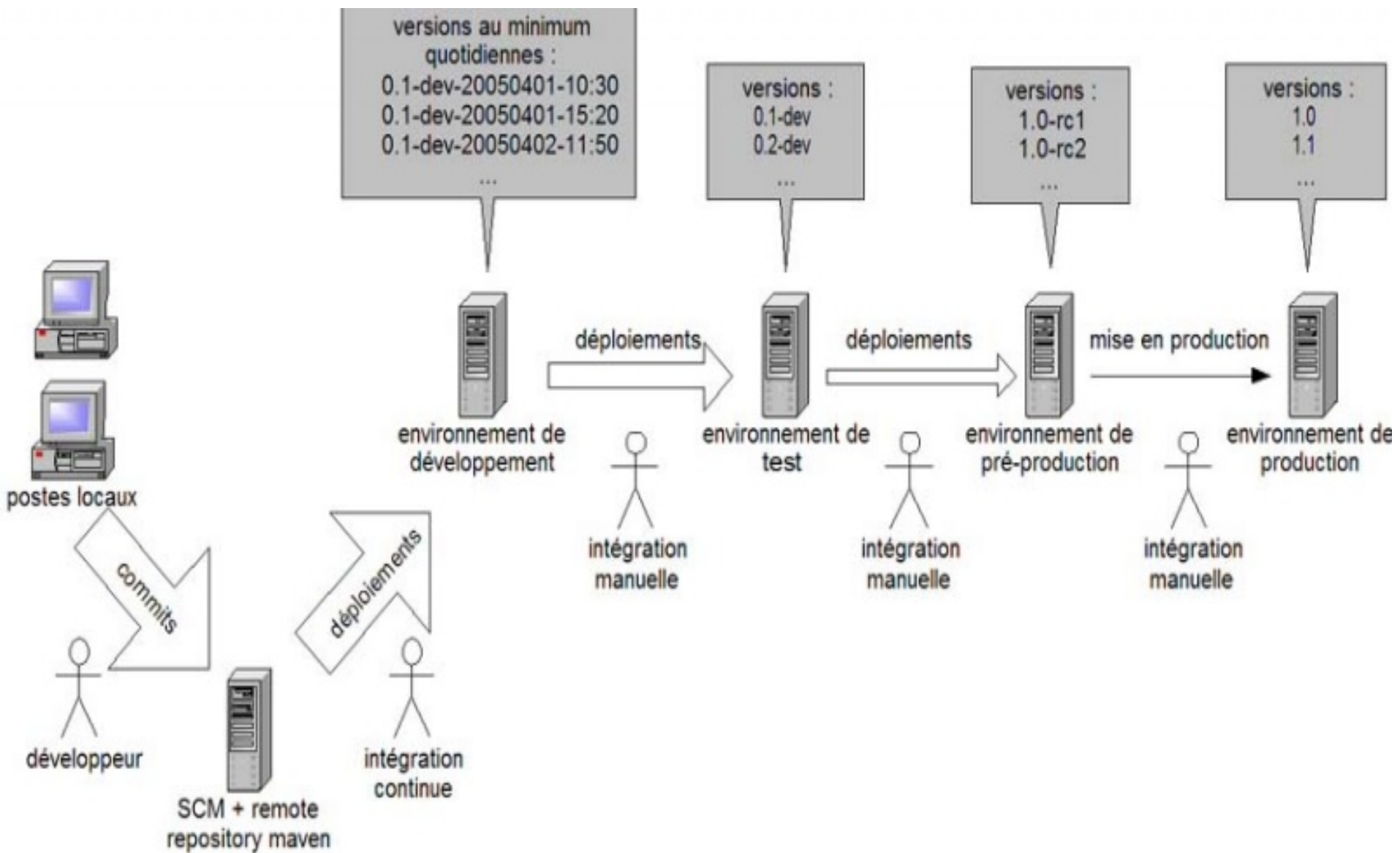


# Open Issues

- How to collect tests in Social Spaces ?
  - For Taaable, User feedback can transformed into a test suite. Ex : I like this adation, I dislike this adaptation can generate implicit test case...
  - In Wikipedia, Tables can be considered as validated results, changes on ontology can be evaluated on recomputing every tables and compare with originals
- And How to transform it into formal tests/ rules, with which language ?

# CI in Software Engineering

- CI implements *continuous* processes of applying [quality control](#) — small pieces of effort, applied frequently.
- CI aims to improve the [quality of software](#), and to reduce the time taken to deliver it, by replacing the traditional practice of applying quality control *after* completing all development. (source wikipedia)



## Calgary



Downtown Calgary.

### Government

- Mayor	Dave Bronconnier <i>(Past mayors)</i>
- Governing body	Calgary City Council
- Manager	Owen A. Tobert

### Area <sup>[1]</sup>

- City	726.50 km <sup>2</sup> (280.5 sq mi)
- Metro	5,107.43 km <sup>2</sup> (1,972 sq mi)

Elevation 1,048 m (3,438.3 ft)

### Population (2006)<sup>[1]</sup>

- City	988,193
- Density	1,360.2/km <sup>2</sup> (3,522.9/sq mi)
- Metro	1,079,310
- Population rank	3rd
- Metro rank	5th

<http://en.wikipedia.org/wiki/Calgary>

```
<http://dbpedia.org/resource/Calgary>
  dbpedia:native_name "Calgary" ;
  dbpedia:elevation "1048" ;
  dbpedia:population_city "988193" ;
  dbpedia:population_metro "1079310" ;
  mayor_name
    dbpedia:Dave_Bronconnier ;
  governing_body
    dbpedia:Calgary_City_Council ;
  ...
```

DBPedia 2.0 in July 2007

DBPedia 3.5 in March 2010

DBPedia 3.6 in October 2010

Release every 3 to 5 months... Time needed by a trained team to update knowledge from contents... 32



# Context

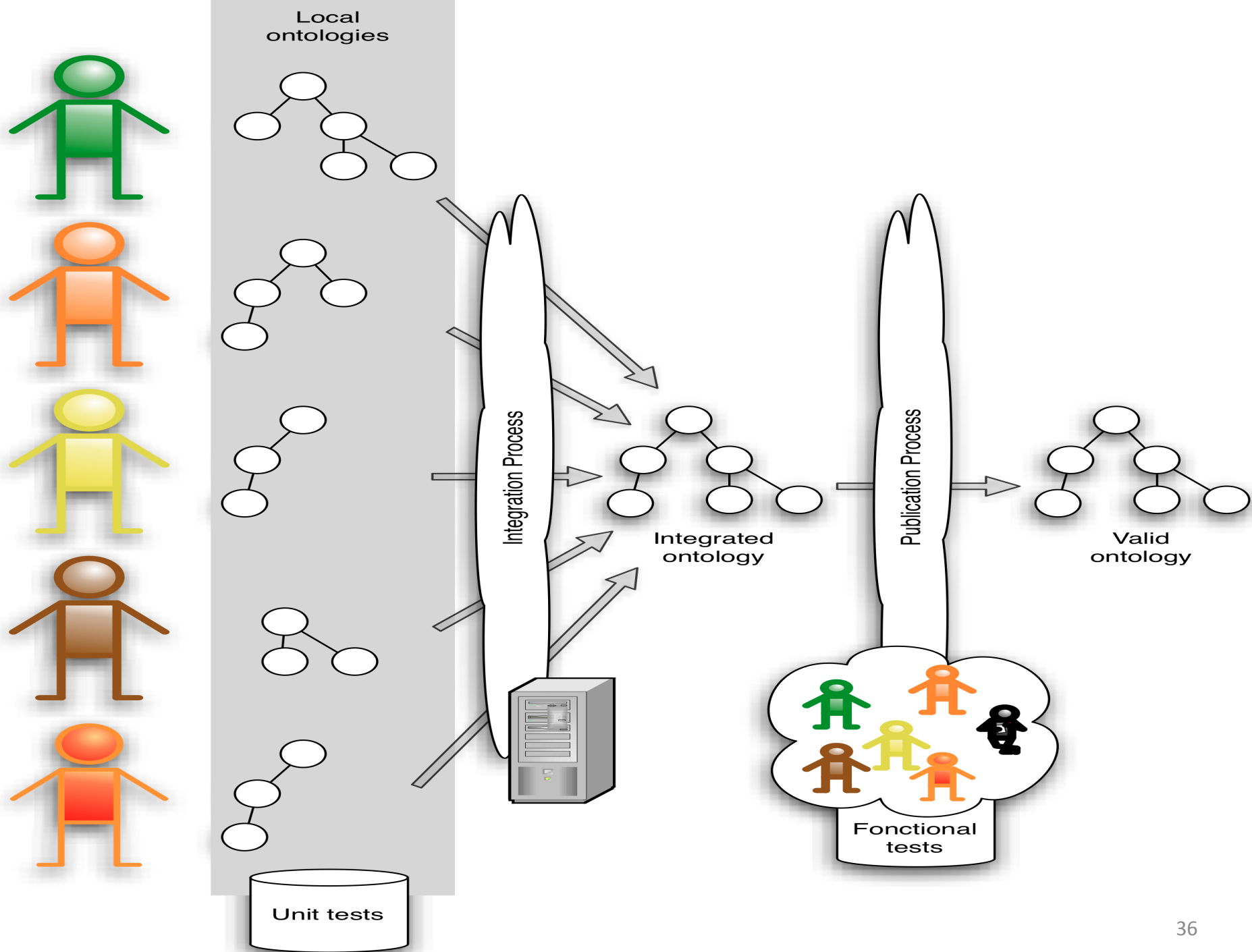
- Social web is producing and updating a huge amount of information... **continuously**.
  - Ex : Wikipedia, Blogs, Twitter,
- Information need to be transformed **continuously** into knowledge in order to provide better search and better navigability on the web : semantic web vision
  - This transformation is costly (KDD Process)
  - Ex: Wikipedia -> DBPedia,

# Context

- The transformation of Wikipedia to Dbpedia is not executed within Wikipedia as social tool or in a social semantic tool as Semantic Wikipedia.
  - The DBPedia team takes dumps from Wikipedia, filter noisy data, scan changes on templates, evaluate if DBPedia ontology needs to be updated, modify their mappings from template to ontology...
- DBPedia ontology is not editable has any Wikipedia pages... The evolution of ontology is done by DBPedia maintainers...
- Transformations are done **outside the social semantic web.**

# Context

- What happens if DBPedia Ontology become writable in the social space ?
  - Replacing Wikipedia+DBPedia by a Semantic Wikipedia ?
- The lightweight editing model of Wikipedia works well for Wikipedia documents
- It does not apply to ontologies, even to lightweight ontologies
  - A small change in type definitions can break all semantic requests...
  - We can try on the kolflow site ;) just change the type ??



# Distributed WikiTaaable using DSMW

- WikiTaaable is semantic wiki that solve cooking problems using case-based reasoning engine.
  - “I want a dessert with rice and figs”
  - The answer: existing recipe with rice and figs as ingredients or an adaption of an existing recipe
  - Replace mangoes by figs in existing recipe in the cookbook.
- Allow isolated work on semantic wikis
  - What is done in DSMW.. Allow to make change locally, but does not ensure that make changes public will not break the semantic wiki.
  - Need a validation process similar to Continuous Integration process in Software engineering...

Dietary practices:  Vegetarian  Nut-free  No alcohol  Low cholesterol  Gout Diet

[Adapt a specific recipe...](#)

[Customize your dietary practices...](#)

*Example.* If you want an apple pie without cinnamon, enter "apple pie\_dish -cinnamon".

[Learn more about advanced queries...](#)

Your request is: **dessert\_dish fig rice**

The request used for adaptation is: **dessert\_dish fig rice**

#	Original recipe name (click to open recipe)	Adaptation overview (click to see the details)
1	<a href="#">Glutinous rice with mangoes</a>	<a href="#">Replace: Mango by Fig</a>

Results 1 - 1 on 1 | Processing time: 0.6189 seconds



## [Glutinous rice with mangoes](#)

The ingredient substitutions

- Mango → Fig

### Preparation Adaptation

Cut figs into wedges. SEASONINGS SAUCE GARNISH Soak the rice in cold water for 2 hours. Drain. Line a steamer with cheesecloth, heat steamer and lay rice on the cheesecloth. Steam for 30 minutes or until cooked through. The rice will become glossy. Mix the SEASONINGS\_ingredients in a large bowl and gently mix in the hot steamed rice. Cover tightly and let soak for 30 minutes to absorb the coconut flavour. Blend the SAUCE\_ingredients in a pot and heat until it just reaches the boiling point. Let cool. ~~Peel the mangoes, slice lengthwise and remove the pits.~~ Divide the rice among 6 plates. Place ~~mango~~ fig slices on top and cover with the sauce. Sprinkle with the sesame seeds and serve.

