

# *Guerrilla IT* with Pharo

[piotr@palacz.net](mailto:piotr@palacz.net)

## First Things First: Clarifications

***Guerrilla in IT*** (be it with Pharo or without it) ***is just a metaphor.***

**It is not**, among others:

- An adaptation of Mao Zedong ideas on guerrilla tactics in practice  
*The enemy advances, we retreat; the enemy camps, we harass;  
the enemy tires, we attack; the enemy retreats, we pursue.*
- An endorsement or an encouragement to incite or actually carry out any form of physical violence in the workplace.
- And so on.

*No animals were hurt when preparing this talk.*

The ***Guerrilla in IT*** metaphor *brings out some facets* of the situation you are likely to find yourself in when *using Pharo in an Enterprise setting*.

This setting has inherent challenges:

- In some cases, mitigation could be attempted using Pharo, but it should not: there are better means for the mitigation.
- In some other cases, it could be attempted fruitfully.

I am going to talk of both. Hence: ***Guerrilla IT with Pharo.***

**01**

## **The Guerrilla Metaphor and IT**

... As a description and as a strategy

**02**

## ***Key Challenges and their Mitigations***

... *Featuring Guest Appearances by Requirements and Traceability*

**03**

## **Enter Pharo into the Guerrilla**

... Some examples and suggestions

**04**

## **Conclusions**

## More about Guerrilla

**Guerrilla warfare** is a type of asymmetric engagement: competition between opponents of *unequal strength*.

**Asymmetric engagement** takes place between belligerents

- Whose relative military power *differs significantly*, or
- Whose *strategy and/or tactics differ significantly*.

*These concepts have some elements of interest for us.*

**Guerrilla** as an indication of the **status quo** at hand, including:

- Existing asymmetry in strength (personnel, resources, etc.)
- Contrast in the impacts of failure on respective sides
- Actual differences in strategy and tactics (as in e.g. in corporate/PM culture).

**Guerrilla** as an indication of a **Strategy** aiming to:

- Magnify the impact of a small, mobile force on a larger, more cumbersome one
- Not simply to defeat an enemy, but (also) to win popular support

As a ***Strategy*** prescription:

- Driving the solution rather than waiting for it to happen
- Assuming broad SDLC responsibilities (from requirements to maintenance)
- Being open to non-conventional solutions
- Avoiding failure as a potentially lethal event
- ... etc.

As a ***Status Quo*** description:

- Small or tiny teams
- Relatively small budgets
- Short timeframes
- Dealing with defective requirements
- Dealing with urgency, nervousness,...
- ... etc.

## Key Challenges in Enterprise Setting

The key challenges in Enterprise-type projects center on:

- **Requirements**
- **Traceability**

[IMO – YMMV]

On ***Requirements***, because:

- Reqs-related problems are common and wide-spread
- Reqs are the first building block
  - Getting them wrong fundamentally dooms the effort

On ***Traceability***, because:

- It is either absent, misunderstood or mishandled
- It ties the building blocks together, *starting with Reqs*
  - Getting it wrong sky-rockets the risk of failure

## Meanings of *Traceability*

**Narrow meaning (most common):** *Tracing testing artefacts* (incl. tests and their results) back to specific requirements, in order to prove that all requirements have been (a) implemented, and (b) positively tested.

**Broad meaning** (IEEE): *the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another.* [IEEE-610]

**Abstract/academic** idea: Linking together subsequent models using well-defined *model transformations*  
(e.g., from Requirements Model to Execution and Testing Models)



## Traceability: Various Manifestations

### Manifestations of Traceability *in-the-large*:

1. Absent (when contract language is deficient)
2. Lip service (when things go OK)
3. **Antagonistic** (when things start going bad)
4. Nuclear Fusion (when litigation is looming)
5. **Backward-looking** (almost always)

### Manifestations of Traceability *in-the-small*:

1. Non-existent (the default)
2. Antagonistic and backward-looking (when things go sour)
3. **Collaborative and forward-looking** (when doing well)

## Uses of Pharo in the Guerrilla

(1) Can Pharo be used *outside* its usual place as a construction and testing toolkit?

**Yes!**

(2) Can it be used to mitigate the Requirements Predicament?

Yes ...

(3) Can it be used to help establishing a collaborative, forward-looking traceability?

Yes ...

How all the **Yes** are even *possible*?

In the *mainstream*, the specification/design/construction/testing overhead makes it *unthinkable* or at least *impractical* to use its own tools to support the project outside of the construction proper.

This is *not the case* with Pharo:

- Low cost/high speed of building assistive tooling makes using Pharo to support the Guerrilla strategy
  - (a) *conceivable* and (b) *feasible*
- Some types of that tooling (and associated simple practices introduced by it) produce disproportionately good returns ...

Real-Life Examples?

## Examples: Some Rules of Thumb

Go for simple/low-cost improvements that have high impact potential

- Transforming representation of relevant data/information
- Making accessing and/or modifying this information easier
- Making generation of derived information painless, simple and quick
- ... and so on

Note: *making this or that* here is not in the context of the *target app!*

Avoid:

- Things that cannot be fixed (e.g., an SDLC broken beyond repair)
- Replicating functions that can be had at even lower cost (e.g., an OSS issue tracker)
- Functions with low impact potential (which may still happen to be cool)
- ... and so on

Real-Life Examples?

## Examples: Dealing with Requirements

... Real-Life Case

- Transform the Reqs away from the typical inedible forms
- Republish them (or their relevant parts) in a digestible form
- (e.g. visualization) in an easy-to-access form (e.g., web site/portal)
- Provide simple query/reporting capabilities
- Show people that this is easier for them (and sometimes fun)



- Better traction with non-tech people
- Better external visibility of the issues
- More eyeballs -> better coverage -> better quality (eventually)
- Opportunities:
  - To introduce structural corrections easily or painlessly
  - To divide-and-conquer -> faster turnover
  - To eliminate stop-and-go -> less friction

Excel, Word, and SharePoint  
don't cut it

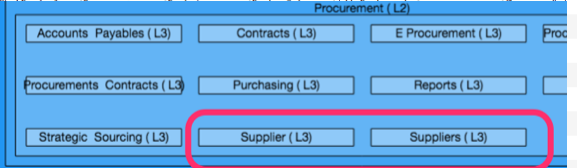
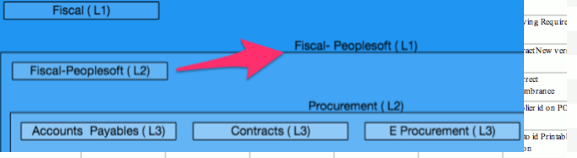
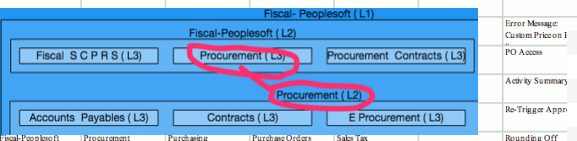
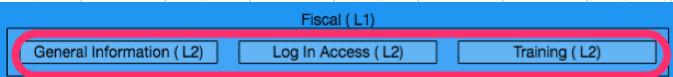
# Example: Dealing with Requirements

... one of many ways

Pharo 6 + Cytoscape + Object Model

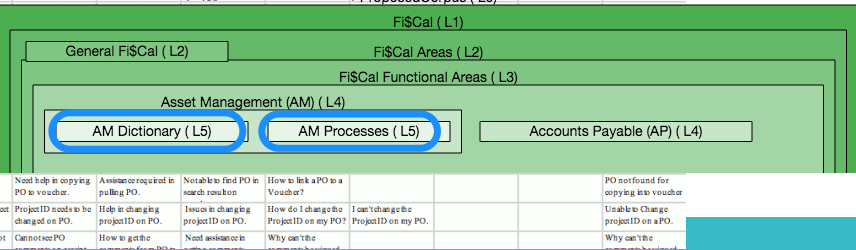
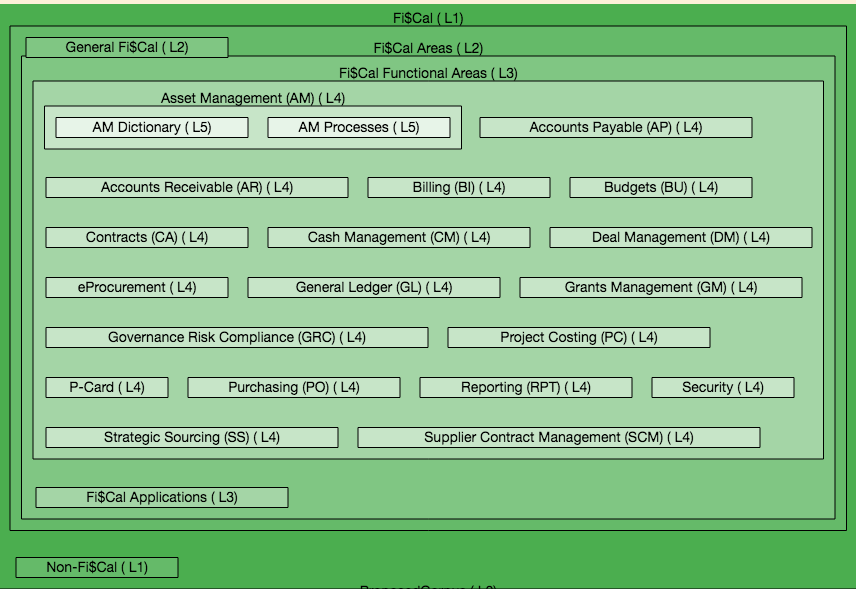
StrNo	Domain	SubDomain	Sub Category 1	Sub Category 2	Sub Category 3	Sub Category 4	Issue Type	Issue Description	Resolution	Question 1	Question 2	Question 3	Question 4	Question 5	Question 6	Question 7	Question 8	Question 9	Final Question 1
1	Fiscal-PeopleSoft	Procurement	Purchasing	Receipts	Chart fields		Chartfield not appearing	Error Message When Creating a Receipt	When they have a department FISCal will	When trying to save Receipt system gives	Chartfield not appearing on Receipt								
2	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		Change Conflict error on PO	Change conflict error on PO	The "Change Conflict error occurs when the	Facing change conflict error while saving PO	Notable to save PO getting error messa								
3	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		Accounting Date issue	Accounting Date Error and Budget Issue in doing Purchase Order	FISCal does not allow user to save any	Accounting date error message is appearing	Notable to save PO Accounting date is								
4	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		Discounts PO	Discounts provided by	If you need to create a PO	How can I add a discount Price on a PO	PO calculating sales on Discounted price								
5	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		Source	Issue in doing Purchase Order	Once a Purchase Order is done and	How to Close a PO?	How to Discontinued the PO								
6	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		if fields gray out	Notable to change chartfields of PO	You must create a Change Order on a	Notable to Change Chartfield of PO	How can I change chartfields, fund,								
7	Fiscal-PeopleSoft	Procurement	Purchasing	Purchase Orders	PO Updating		Discontinued PO	Discontinued provided by	If you need to create a PO	How can I add a discount Price on a PO	PO calculating sales on Discounted price								
8																			
9																			
10																			
11																			
12																			
13																			
14																			
15																			
16																			
17																			
18																			
19																			
20																			
21																			
22																			
23																			
24																			
25																			
26																			
27																			
28																			
29																			
30																			
31																			
32																			
33																			
34																			
35																			
36																			
37																			
38																			
39																			
40																			
41																			
42																			
43																			
44																			
45																			

## Initial Excel Corpus and its Issues



- Fs-Chatbot.package
  - .filetree
  - FsAbstractCorpusNode.class
  - FsCatCorpusNode.class
  - FsCorpusCloudDictionary.class
  - FsCorpusEntry.class
  - FsCorpusHierarchy.class
  - FsCorpusNameCloud.class
  - FsCorpusNameCloudBuilder.class
  - FsCorpusNode.class
  - FsCorpusSet.class
  - FsCorpusSetReader.class
  - FsCorpusSetVisualizer.class
  - FsCorpusTree.class
  - FsCorpusTreeBuilder.class
  - FsWebFilter.class
  - monticello.meta
  - NeoCSVReader.extension
  - Object.extension
  - properties.json
  - String.extension
  - TLDemos.extension
  - TLVisualization.extension
  - UndefinedObject.extension

## Proposed Corpus Structure – Interactive Exploration



## Examples: Towards Better Traceability

... Real-Life Case

- Identify requirement types that can be handled by a DSL or the Interpreter pattern (typically, Business Rules are good candidate)
- Reify the specs in a model
- Provide for human-friendly self-description of that model
- Provide self-descriptions for eyeballing (re-generation on change) and simple collection of feedback



**Modeling tools and Rule Engines  
don't quite cut it**

- Disambiguation of notoriously problematic areas (esp. Business Rules)
- Early, pre-construction and pre-testing validation of *essential* Reqs with *potential design impacts*
- Opportunities:
  - To introduce improved precision and cohesion of the language
  - To divide-and-conquer the space
  - To introduce ownership and accountability

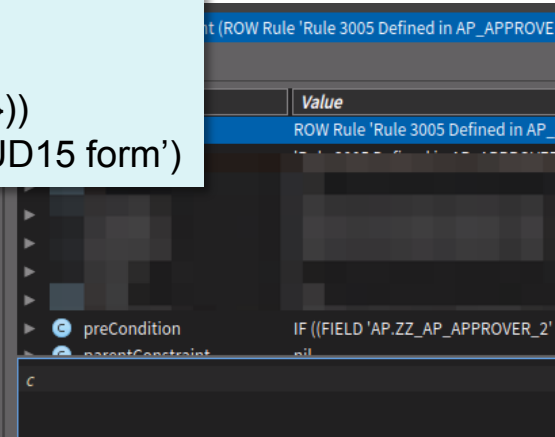
ROW Rule 'Rule 3005 Defined in AP\_APPROVER':  
**IF** ( (FIELD 'AP.ZZ\_AP\_APPROVER\_2' has value)  
**AND** (FIELD 'AP.EMAILID' is none of (case-insensitive) { ... 300 entries ... })))  
**THEN** ( **FAIL** with **MESSAGE**: 'Each AP Approver 2 must have approved AUD15 form')

Self-textual renderings ...

... of rule class instances

```
ROW Rule 'Rule 2004 Defined in AM_APPROVER' IF (ROW has ALL values in FIELDS: 'A
ROW Rule 'Rule 3002-3003 Defined in AP_APPROVER' IF (ROW has NO values in FIEL
ROW Rule 'Rule 3005 Defined in AP_APPROVER' IF ((FIELD 'AP.ZZ_AP_APPROVER_2'
ROW Rule 'Defined in GL_APPROVER' IF (ROW has NO values in FIELDS: 'GL.ZZ_GL_A
ROW Rule 'Rule 400x Defined in GL_APPROVER' IF (ROW has a value in FIELD: 'GL.ZZ
ROW Rule 'Rule 5010/5011/5012 Defined in REQUISITION_APPROVER' IF (ROW has N
ROW Rule 'Rule 5010a1 Defined in REQUISITION_APPROVER' IF ((FIELD 'RA.ZZ_REQ_
ROW Rule 'Rule 5010a2 Defined in REQUISITION_APPROVER' IF ((FIELD 'RA.ZZ_REQ_
ROW Rule '50XY Defined in REQUISITION_APPROVER' IF ((ROW has a value in FIELD:
```

ROW Rule '50XY Defined in REQUISITION\_APPROVER':  
**IF** ((ROW has a value in FIELD: 'RA.ZZ\_REQ\_AD\_HOC\_APPR')  
**AND** (ROW has at **LEAST ONE** value in FIELDS:  
 'RA.ZZ\_REQ\_APPROVER\_1', 'RA.ZZ\_REQ\_APPROVER\_2'))  
**THEN** ( **FAIL** with MESSAGE: 'If Requisitions Ad Hoc Approver is identified,  
 then other Requisitions approver levels must not be identified')





### The *Guerrilla Metaphor may*:

- Help you relate to real-life experiences when using Pharo in an Enterprise context and what that entails
- Help you consider what concrete Agile could be when *the tire hits the road*
- Stimulate devising other useful ways of using Pharo across all phases of the SDLC

### The **key thought** is that with Pharo:

- Low cost/high speed of building assistive tooling makes using Pharo to support the Guerrilla strategy (a) conceivable and (b) feasible
- Some types of that tooling (and associated simple practices introduced by it) can produce disproportionately good returns.



# Thank you

Comments, suggestions, feedback, etc.: please email [piotr@palacz.net](mailto:piotr@palacz.net)