Knowledge Based Situation Discovery for Avionics Maintenance

Luis Palacios Medinacelli[1], Yue Ma[2], Chantal Reynaud[2], Gaëlle Lortal[3]

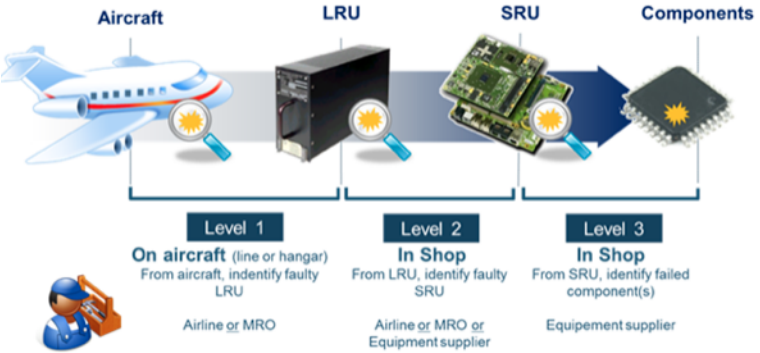
[1] CEA-LIST, Palaiseau, France
[2] LRI, Univ. Paris-Sud, CNRS, Université Paris-Saclay, France
[3] LRASC, Thales Research & Technology, Palaiseau, France

Published by KCAP 2019

Table of Contents

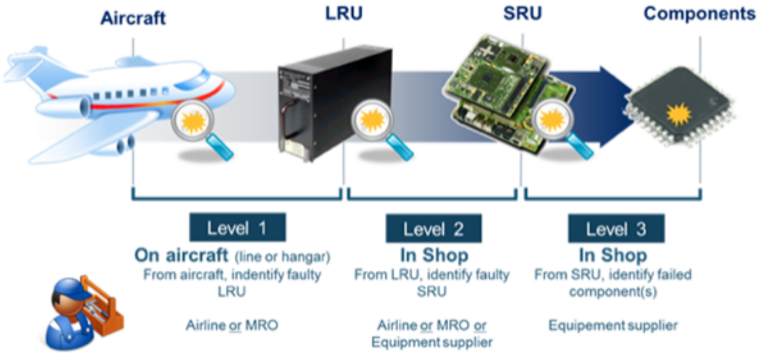
Context

LRU: Line Replacement Unit
SRU: Shop Replacement Unit

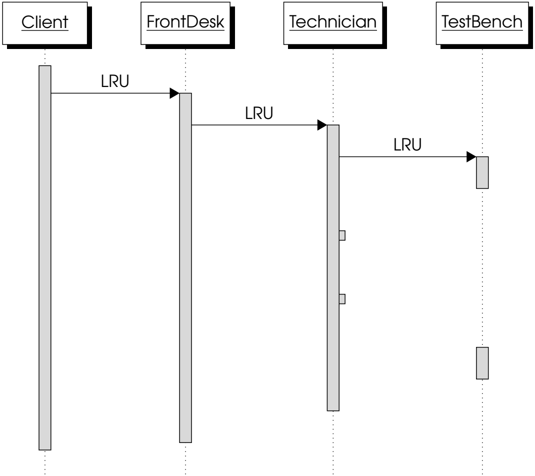


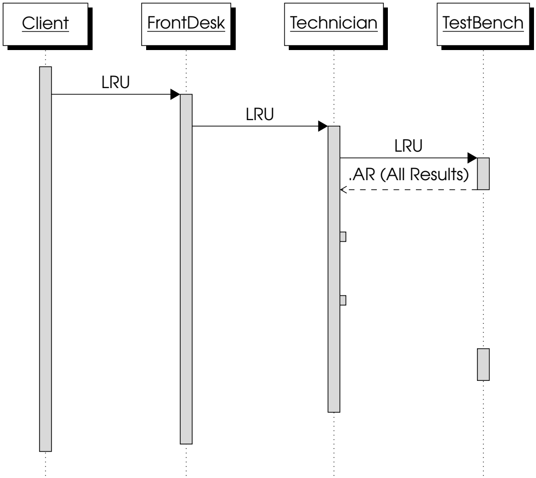
Context

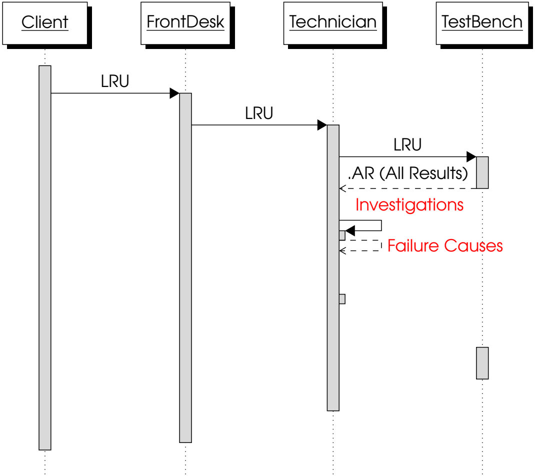
LRU: Line Replacement Unit
SRU: Shop Replacement Unit

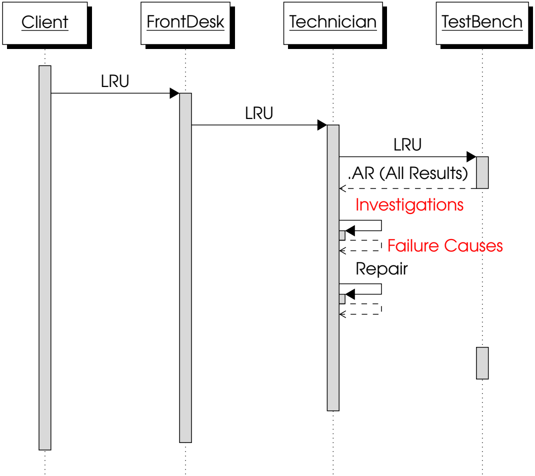


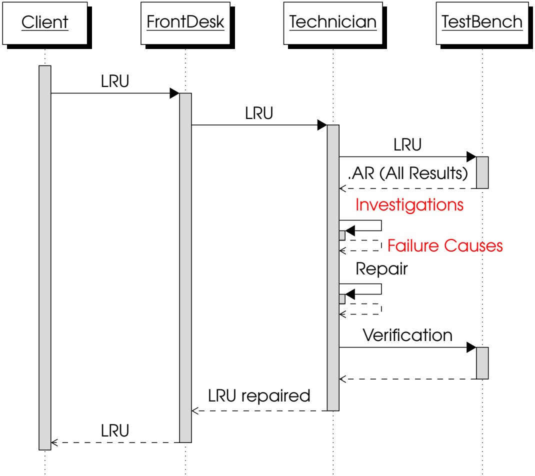
Our work concerns levels 2 and 3, i.e in shop maintenance.

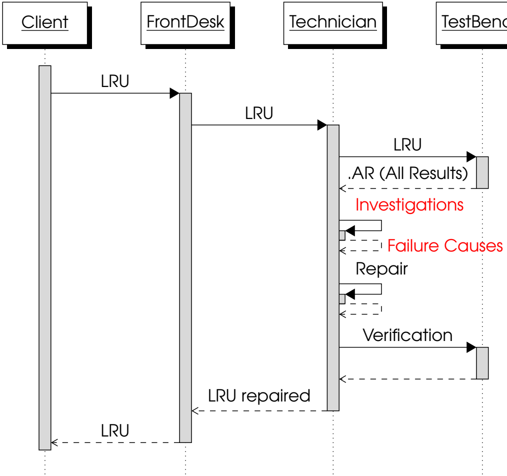

Client
FrontDesk
Technician
TestBench
LRU
LRU
LRU


Client
FrontDesk
Technician
TestBench
LRU
LRU
LRU
.AR (All Results)


Client
FrontDesk
Technician
TestBench
LRU
LRU
LRU
.AR (All Results)
Investigations
Failure Causes


Client
FrontDesk
Technician
TestBench
LRU
LRU
LRU
.AR (All Results)
Investigations
Failure Causes
Repair

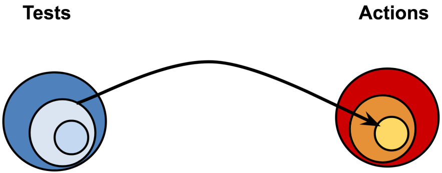

Client
FrontDesk
Technician
TestBench
LRU
LRU
LRU
.AR (All Results)
Investigations
Failure Causes
Repair
Verification
LRU repaired
LRU



The signature of a failure provides all the necessary information to understand, identify and ultimately repair a failure.

Objective : From an application point of view

Support avionics maintenance by discovering Failure Signatures and proposing corrective actions, in such a way that the suggestions are explainable.



Problematic

Our problem has the following constraints:

Problematic

Our problem has the following constraints:

- **Sparse data:** We can not rely on techniques that require large amounts of data to model a process (150 samples).

Problematic

Our problem has the following constraints:

- **Sparse data:** We can not rely on techniques that require large amounts of data to model a process (150 samples).
- **No explicit model knowledge:** for the diagnose process.

Problematic

Our problem has the following constraints:

- **Sparse data:** We can not rely on techniques that require large amounts of data to model a process (150 samples).
- **No explicit model knowledge:** for the diagnose process.
- **Number of suggested repair actions:** The suggested components to be replaced should be minimized.

Problematic

Our problem has the following constraints:

- **Sparse data:** We can not rely on techniques that require large amounts of data to model a process (150 samples).
- **No explicit model knowledge:** for the diagnose process.
- **Number of suggested repair actions:** The suggested components to be replaced should be minimized.
- **Provide an explanation for the given results:** Explainability of the system is important for technicians and security certification.

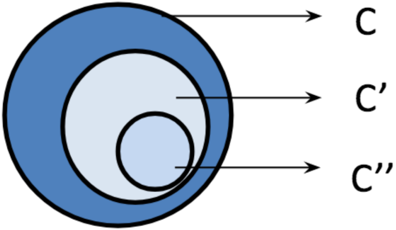
Objective : From a theoretical point of view

- Modeling the domain in a formal ($\mathcal{ELO}$) ontology
- Discovering concepts represent failure signatures that allow us to associate repair actions

Table of Contents

Intuition

Concept Refinement



Representative Concept

Definition (A representative concept)

$\mathcal{O}$: an ontology, Δ: the set of individuals in $\mathcal{O}$.
Let $\mathcal{X} \subseteq \Delta$. For a concept C, we say that $\mathcal{X}$ is represented by C if:

$$\mathcal{O} \models C(x) \text{ for all } x \in \mathcal{X}, \text{ and}$$

$$\mathcal{O} \not\models C(y) \text{ for any } y \in \Delta \setminus \mathcal{X}.$$

Representative Concept

Definition (A representative concept)

$\mathcal{O}$: an ontology, Δ: the set of individuals in $\mathcal{O}$.
Let $\mathcal{X} \subseteq \Delta$. For a concept C, we say that $\mathcal{X}$ is represented by C if:

$$\mathcal{O} \models C(x) \text{ for all } x \in \mathcal{X}, \text{ and}$$

$$\mathcal{O} \not\models C(y) \text{ for any } y \in \Delta \setminus \mathcal{X}.$$

Proposition

Representability$_C$*:*

Does C represent $\mathcal{X}$ w.r.t. $\mathcal{O}$?

can be solved in ***PTime****.*

Representative Concept

Definition (A representative concept)

$\mathcal{O}$: an ontology, Δ: the set of individuals in $\mathcal{O}$.
Let $\mathcal{X} \subseteq \Delta$. For a concept C, we say that $\mathcal{X}$ is represented by C if:

$$\mathcal{O} \models C(x) \text{ for all } x \in \mathcal{X}, \text{ and}$$

$$\mathcal{O} \not\models C(y) \text{ for any } y \in \Delta \setminus \mathcal{X}.$$

Proposition

Representability$_C$*:*

Does C represent $\mathcal{X}$ w.r.t. $\mathcal{O}$?

*can be solved in **PTime**.*
Representability$_n$*:*

Is there a concept C with $|C| < n$ that represents $\mathcal{X}$ w.r.t. $\mathcal{O}$?

*is **ExpTime**. If n is bounded by a constant, then Representability$_n$ is in **PTime**.*

Situation Discovery

The concepts representing a set $\mathcal{X}$ are equivalent in the sense of their instances.

We call each of these equivalent classes a situation in $\mathcal{O}$.

Definition (Situation in $\mathcal{O}$)

$\mathcal{O}$: an ontology; Δ: the set of individuals in $\mathcal{O}$; $\mathcal{X} \subseteq \Delta$.
A situation for $\mathcal{X}$ in $\mathcal{O}$ is the set:

$$||\mathcal{X}||_{\Delta}^{\mathcal{O}} = \{C \mid C \text{ represents } \mathcal{X} \text{ w.r.t. } \mathcal{O} \text{ and } \Delta\}.$$

Situation Discovery - All Situations

Definition (SD_n)

$\mathcal{O}$: an ontology; Δ: the set of individuals in $\mathcal{O}$; $\mathcal{X} \subseteq \Delta$; $n > 0$;

SD_n: Does there exist a situation C for some $\mathcal{X}' \subseteq \mathcal{X}$ in $\mathcal{O}$ with $|C| \leq n$?

Situation Discovery - All Situations

Definition (SD_n)

$\mathcal{O}$: an ontology; Δ: the set of individuals in $\mathcal{O}$; $\mathcal{X} \subseteq \Delta$; $n > 0$;

SD_n: Does there exist a situation C for some $\mathcal{X}' \subseteq \mathcal{X}$ in $\mathcal{O}$ with $|C| \leq n$?

Proposition

*SD_n is in **ExpTime**. If $|\mathcal{X}|$ and n are bounded by a constant, SD_n is in **PTime**.*

Situation Discovery - All Situations

Definition (SD_n)

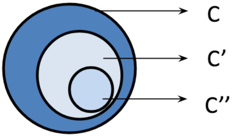
$\mathcal{O}$: an ontology; Δ: the set of individuals in $\mathcal{O}$; $\mathcal{X} \subseteq \Delta$; $n > 0$;

SD_n: Does there exist a situation C for some $\mathcal{X}' \subseteq \mathcal{X}$ in $\mathcal{O}$ with $|C| \leq n$?

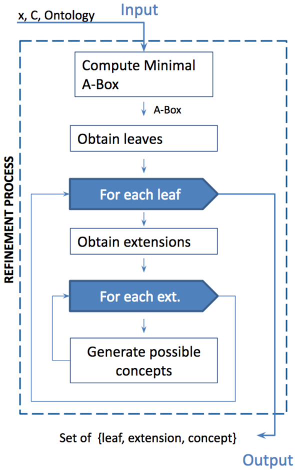
Proposition

*SD_n is in **ExpTime**. If $|\mathcal{X}|$ and n are bounded by a constant, SD_n is in **PTime**.*

⇝ An algorithm for finding situations for a set of individuals



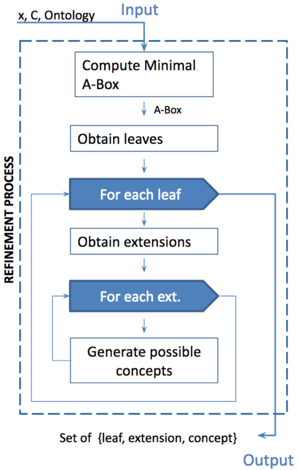
Refinement Algorithm Overview - Steps to refine a concept C



Consider:

$$\begin{aligned} C &\equiv \exists r_1.\top \\ ABox &= \{r_1(x,y), A(y), r_2(x,z), B(z)\} \end{aligned}$$

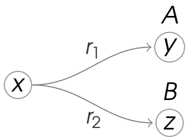
Refinement Algorithm Overview - Steps to refine a concept C



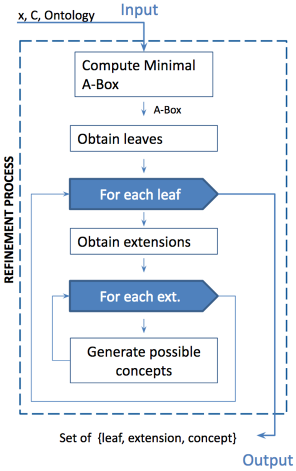
Consider:

$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$

The graph representation of x:



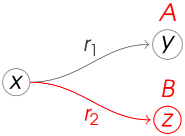
Refinement Algorithm Overview - Steps to refine a concept C



Consider:

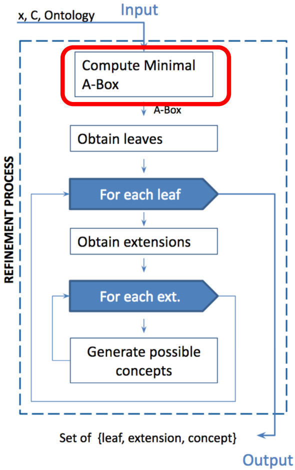
$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$

The graph representation of x:



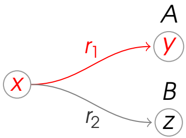
Some assertions are unnecessary

Refinement Algorithm Overview - Steps to refine a concept C



Consider:

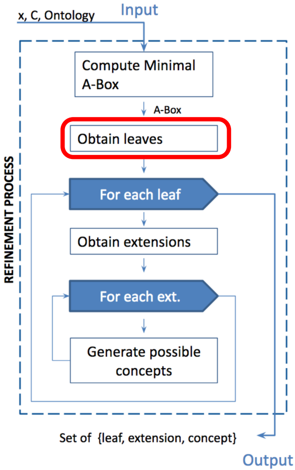
$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$



We can extract those necessary assertions

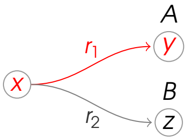
$$\text{Minimal } ABox = \{r_1(x,y)\}$$

Refinement Algorithm Overview - Steps to refine a concept C



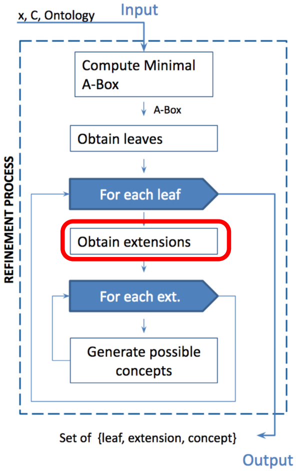
Consider:

$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$



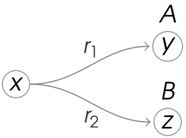
$$\text{Minimal } ABox = \{r_1(x,y)\}$$
$$Leaves_{x,C} = \{x, y\}$$

Refinement Algorithm Overview - Steps to refine a concept C



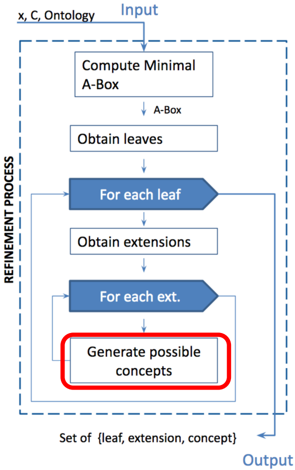
Consider:

$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$



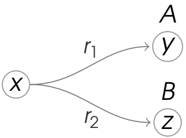
$$\text{Minimal } ABox = \{r_1(x,y)\}$$
$$Leaves_{x,C} = \{x, y\}$$
$$Ext_{x,C} = \{{\color{red}r_2(x,z), A(y)}\}$$

Refinement Algorithm Overview - Steps to refine a concept C



Consider:

$$C \equiv \exists r_1.\top$$
$$ABox = \{r_1(x,y), A(y), r_2(x,z), B(z)\}$$



$$\text{Minimal } ABox = \{r_1(x,y)\}$$
$$Leaves_{x,C} = \{x, y\}$$
$$Ext_{x,C} = \{r_2(x,z), A(y)\}$$

$$C'_1 \equiv \exists r_1.\top \sqcap \exists r_2.\top$$
$$C'_2 \equiv \exists r_1.(\top \sqcap A)$$

The Most Specific Representative (MSR)

Definition (Most Specific Representative MSR)

Given a set of individuals $\mathcal{X} = \{x_1, \ldots, x_n\}$ and the set of its representative concepts $||\mathcal{X}|| = \{\mathscr{S} \mid \mathscr{S} \text{ represents } \mathcal{X}\}$, the Most Specific Representative of the set $\mathcal{X}$, written $\text{MSR}_{\mathcal{X}}$, is the concept $\mathscr{S}_i \in ||\mathcal{X}||$ such that:

$$\forall \mathscr{S}_j \in ||\mathcal{X}||, \text{ we find } \mathscr{S}_i \sqsubseteq \mathscr{S}_j.$$

Summary

We have provided definitions, specifications an algorithms to:

Summary

We have provided definitions, specifications an algorithms to:

- Obtain the refinements of a concept C.

Summary

We have provided definitions, specifications an algorithms to:

- Obtain the refinements of a concept C.
- Obtain the MSR for an individual x.

Summary

We have provided definitions, specifications an algorithms to:

- Obtain the refinements of a concept C.
- Obtain the MSR for an individual x.
- Obtain the MSR for a set $\mathcal{X}$ of individuals.

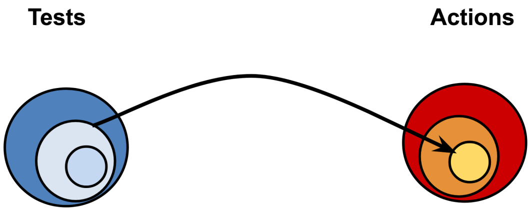
Summary

We have provided definitions, specifications an algorithms to:

- Obtain the refinements of a concept C.
- Obtain the MSR for an individual x.
- Obtain the MSR for a set $\mathcal{X}$ of individuals.
- Discover and characterize all situations present in an ontology $\mathcal{O}$.

Table of Contents

Intuition



The Data Sources - .AR files

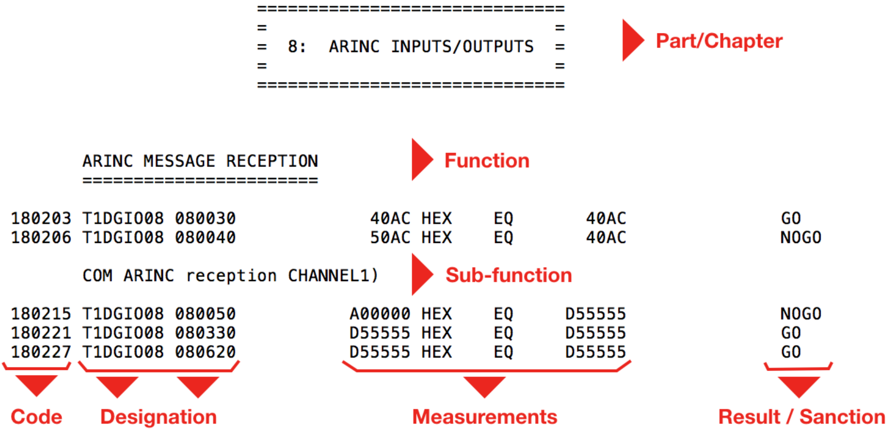


Figure: An extraction of the structure of an .AR file.

The Data Sources - Corrective Actions

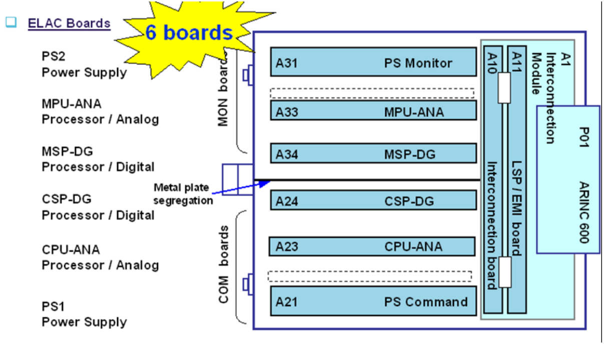
Corrective actions for **Elevator and Aileron Computer (ELAC)** .
We distinguish between composed and individual actions.

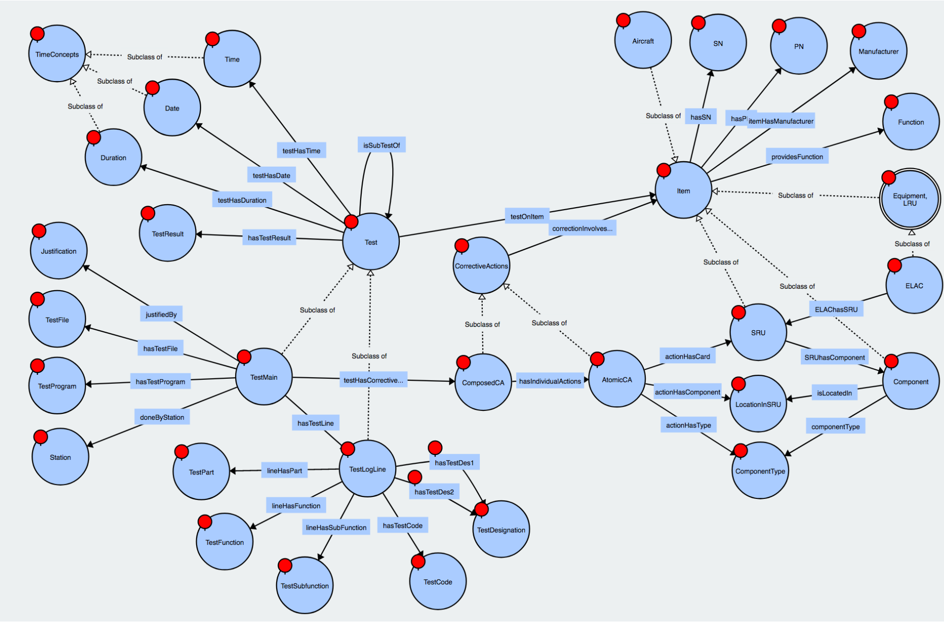
Ar File	SRU1 Type	SRU1 Component	SRU1 repere topo
20094-777-777.AR	MPU ANA	AMPLI	U30 U44
20182-777-777.AR	MSP DG	EPLD	U28
20030-777-777.AR	MSP DG	EPLD|RAM	U25|U35 U36

The Data Sources - Corrective Actions

Corrective actions for **Elevator and Aileron Computer (ELAC)** .
We distinguish between composed and individual actions.

Ar File	SRU1 Type	SRU1 Component	SRU1 repere topo
20094-777-777.AR	MPU ANA	AMPLI	U30 U44
20182-777-777.AR	MSP DG	EPLD	U28
20030-777-777.AR	MSP DG	EPLD|RAM	U25|U35 U36




TimeConcepts
Time
Date
Duration
Subclass of
testHasTime
testHasDate
testHasDuration
isSubTestOf
TestResult
hasTestResult
Test
Justification
justifiedBy
TestFile
hasTestFile
TestProgram
hasTestProgram
Station
doneByStation
TestMain
hasTestLine
testHasCorrective...
TestPart
lineHasPart
TestLogLine
hasTestDes1
hasTestDes2
TestDesignation
lineHasFunction
TestFunction
lineHasSubFunction
TestSubfunction
hasTestCode
TestCode
testOnItem
correctionInvolves...
CorrectiveActions
ComposedCA
hasIndividualActions
AtomicCA
actionHasCard
actionHasComponent
actionHasType
SRU
LocationInSRU
ComponentType
SRUhasComponent
isLocatedIn
componentType
Component
ELAChasSRU
ELAC
Equipment, LRU
Item
Aircraft
SN
PN
Manufacturer
Function
hasSN
hasPI
itemHasManufacturer
providesFunction

Situations in Avionics Maintenance

The .AR files become the individuals we want to distinguish.

Definition (Signature in $\mathcal{O}$)

Given a set of .AR files $\{f_1, \ldots, f_n\}$ and an ontology $\mathcal{O}$, the signature $\mathscr{S}$ of $\{f_1, \ldots, f_n\}$ is their MSR.

Situations in Avionics Maintenance

The .AR files become the individuals we want to distinguish.

Definition (Signature in $\mathcal{O}$)

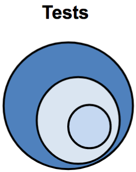
Given a set of .AR files $\{f_1, \ldots, f_n\}$ and an ontology $\mathcal{O}$, the signature $\mathscr{S}$ of $\{f_1, \ldots, f_n\}$ is their MSR.

Definition (Most Specific Signature)

Given an .AR file f_x and an ontology $\mathcal{O}$ that contains learned signatures. Let $f_x \in \mathscr{S}_1, \ldots \mathscr{S}_n$ be the signatures for f_x. A most specific signature for f_x is defined by:

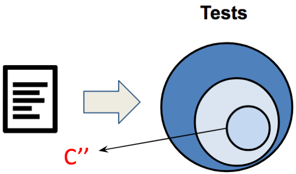
$$\mathscr{S}_{f_x} = \{\mathscr{S}_i \mid \nexists \mathscr{S}_j \text{ with } \mathscr{S}_j \sqsubset \mathscr{S}_i\}$$

Consult - Overview



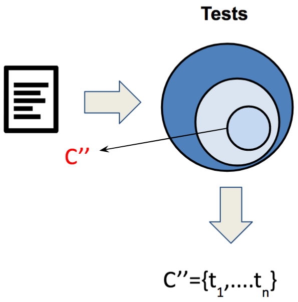
The KB is trained with historical data to learn the signatures.

Consult - Overview



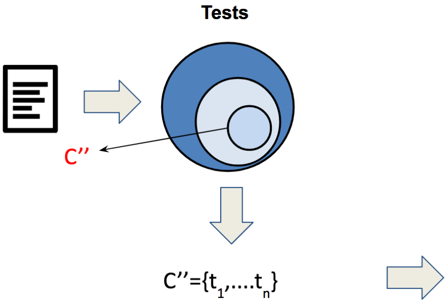
With the signatures available, we can classify new, unseen .AR files.

Consult - Overview



For each signature a set of files is associated.

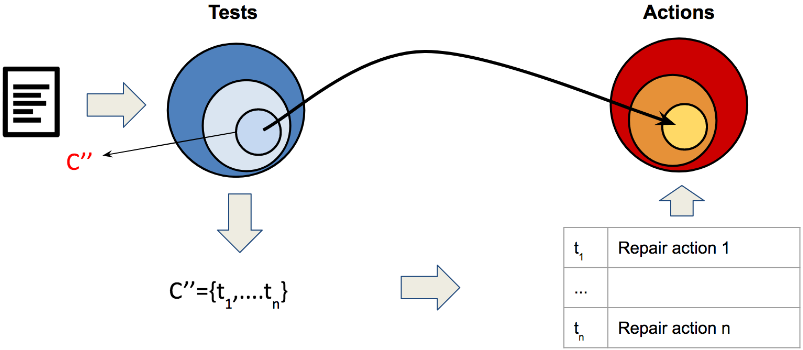
Consult - Overview



t_1	Repair action 1
...	
t_n	Repair action n

The historical data tells us the corresponding corrective actions.

Consult - Overview



These actions become the suggestions for the technician.

Consult the KB - HMI

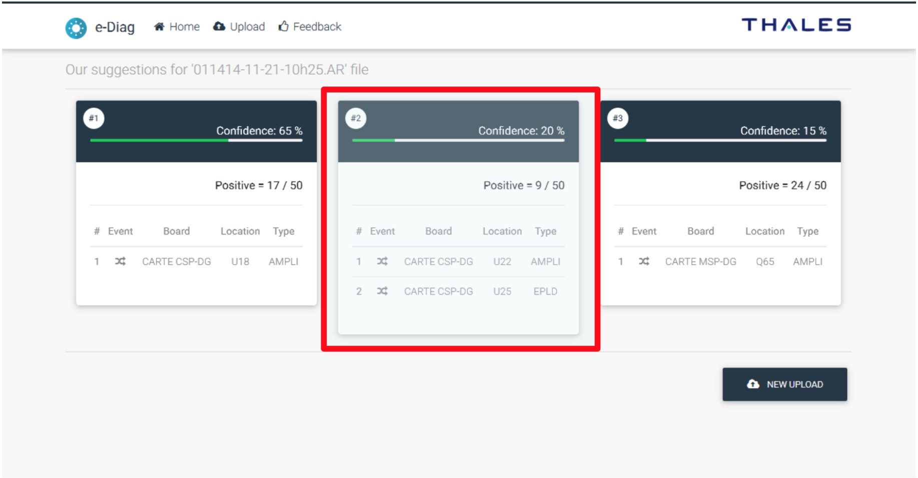
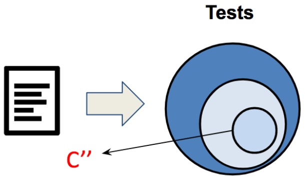


Figure: The suggested corrective actions for the consulted file.

Feedback - Overview

Two main tasks on the feedback process:

- **Integrate new corrective actions:** added to the historical data.



Feedback - Overview

Two main tasks on the feedback process:

- **Integrate new corrective actions:** added to the historical data.
- **Discover new signatures:** incrementally enrich the knowledge base (ontology).

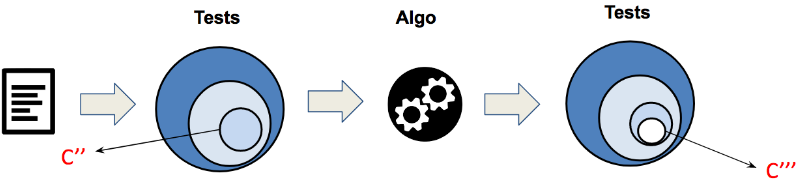


Table of Contents

Similar Approaches - DL-learner

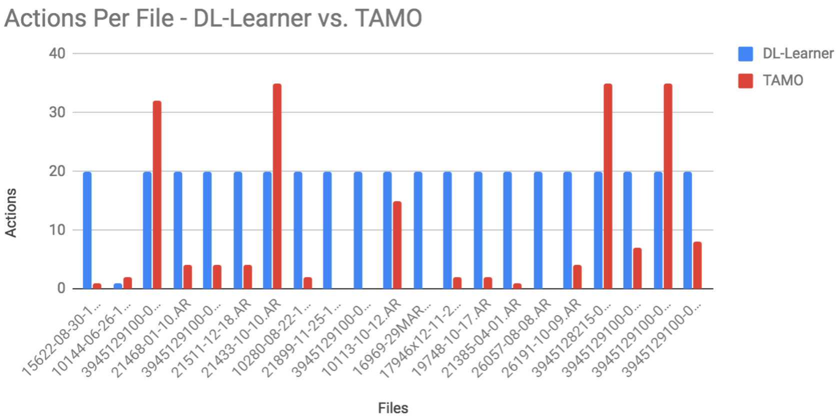


Figure: The less number of actions the better.

Relevance and Specificity of the Suggestions by TAMO

X-axis: each consulted file
Y-axis: the number of relevant individual actions

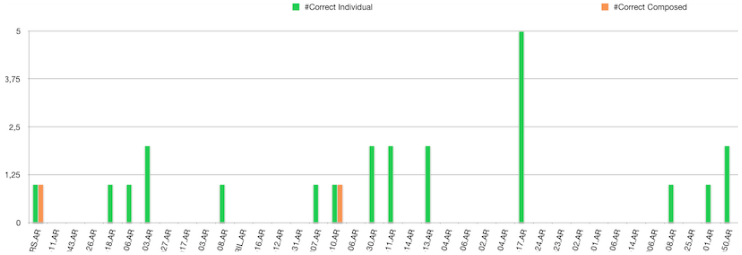
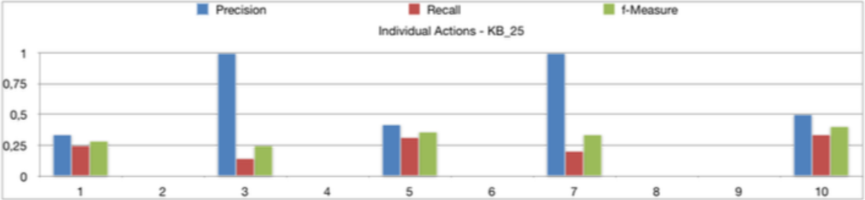


Figure: The suggestions for the test files, when consulting the knowledge base learned from 100 files. In the figure are shown the correct atomic actions (green) and the correct composed actions (orange).

Note that: only 30% of the files have the possibility to be correctly classified due to data sparsity.

Evolution of the KB - Number and Specificity of Signatures

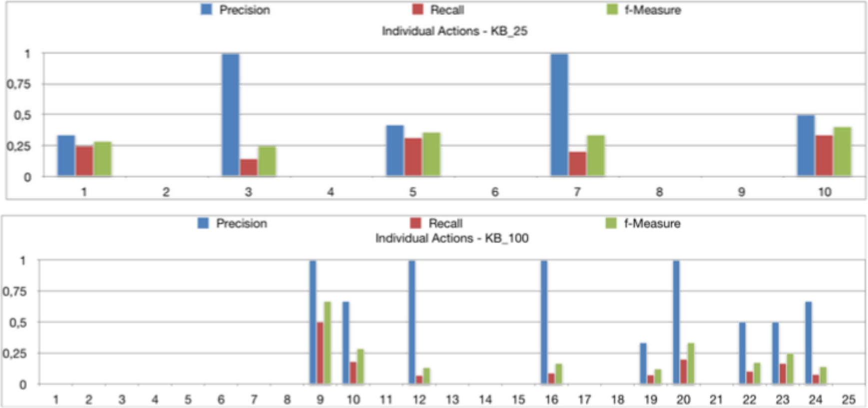
X-axis: each signature
Y-axis: the valued of Precision/Recall/F-score



- 50 files consulted against a knowledge base trained with 25 files.

Evolution of the KB - Number and Specificity of Signatures

X-axis: each signature
Y-axis: the valued of Precision/Recall/F-score



- 50 files consulted against a knowledge base trained with 25 files.
- 50 files consulted against a knowledge base trained with 100 files.

Table of Contents

Conclusions

Conclusions

We have formally defined the situation discovery problem and provided upper bounds and algorithms to compute them.

The discovered concepts provide a meaningful description of the main features shared by the individuals in the situation.

Conclusions

We have formally defined the situation discovery problem and provided upper bounds and algorithms to compute them.

The discovered concepts provide a meaningful description of the main features shared by the individuals in the situation.

Application to an Avionics Maintenance task by characterizing failure signatures as that of discovering situations.

Conclusions

We have formally defined the situation discovery problem and provided upper bounds and algorithms to compute them.

The discovered concepts provide a meaningful description of the main features shared by the individuals in the situation.

Application to an Avionics Maintenance task by characterizing failure signatures as that of discovering situations.

More expressive DLs The current approach in $\mathcal{ELO}$. Natural extensions to be considered are disjunction ($\sqcup$) and negation ($\neg$).

Conclusions

We have formally defined the situation discovery problem and provided upper bounds and algorithms to compute them.

The discovered concepts provide a meaningful description of the main features shared by the individuals in the situation.

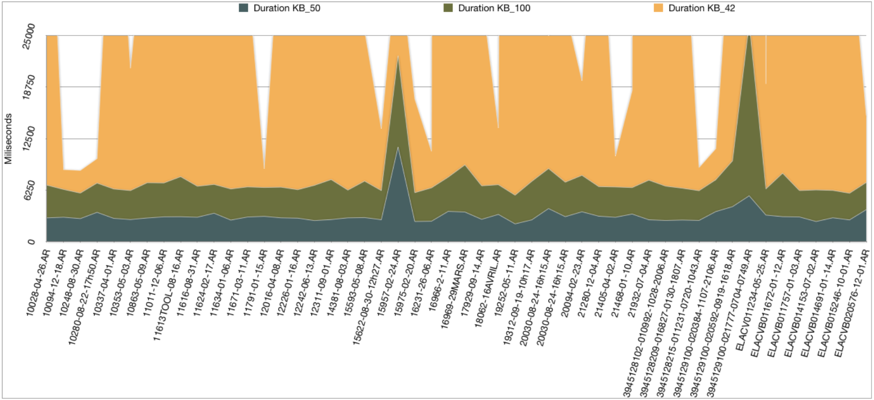
Application to an Avionics Maintenance task by characterizing failure signatures as that of discovering situations.

More expressive DLs The current approach in $\mathcal{ELO}$. Natural extensions to be considered are disjunction ($\sqcup$) and negation ($\neg$).
Parallel Processing The modifications made split the ABox in several consistent partitions. Each ABox can be consulted by a separate process (i.e. parallel processing) and then aggregate the results.

Thanks for your attention

Response Times

50 files consulted against V1 and V2 of the prototype.



- On the background the consult time using V1 (KB trained with 50 files).
- On the front most the consult time with V2 (KB trained with 50 files).
- On the middle (green) consult time V2 (KB trained with 100 files).