

User oriented language for powerful data mining with Ferda

Michal Kováč

December 6, 2007

Table of contents part 1/2

1 Introduction

- Table of contents
- What is Ferda Data Miner?
- Ferda as programming language
- What is missing? What should be done?

Table of contents part 1/2

1 Introduction

- Table of contents
- What is Ferda Data Miner?
- Ferda as programming language
- What is missing? What should be done?

2 New functionality in Ferda

- Network archive
- Boxes for math
- Lambda expression
- Other new boxes

Table of contents part 2/2

- 3 Example – executing four fold task recursively
 - Motivation
 - Linear interpolation
 - Connection of boxes
 - Results

Table of contents part 2/2

3 Example – executing four fold task recursively

- Motivation
- Linear interpolation
- Connection of boxes
- Results

4 Next steps

- Sequences and sets
- Reuse of code
- Better lambda
- Summary

Ferda

What is Ferda?

- User oriented application

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes
- Ferda Data Miner = Ferda + boxes for data mining

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes
- Ferda Data Miner = Ferda + boxes for data mining

History

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes
- Ferda Data Miner = Ferda + boxes for data mining

History

- LISp-Miner not so user friendly

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes
- Ferda Data Miner = Ferda + boxes for data mining

History

- LISp-Miner not so user friendly
- Software project at MFF

Ferda

What is Ferda?

- User oriented application
- For specification of tasks, execution and result browsing
- Works with boxes
- Ferda Data Miner = Ferda + boxes for data mining

History

- LISp-Miner not so user friendly
- Software project at MFF
- Master theses

Screenshot

The screenshot displays the Ferda Data Miner software interface. The main workspace shows a workflow diagram with the following components and connections:

- Input:** A 'Data' box (represented by a folder icon) feeds into a 'load2' box (represented by a pink box).
- Branching:** From 'load2', the flow splits into two parallel paths: one leading to a 'District' box and another to a 'Salary' box.
- Transformation:** Each 'District' and 'Salary' box feeds into a corresponding '1-1' and '1-2' box, respectively.
- Filtering:** The '1-1' and '1-2' boxes feed into 'District(S[1-1])' and 'Salary(S[1-2])' boxes, which include 'T' and 'F' filters.
- Output:** The filtered data from both paths, along with a 'Base' box (represented by a colorful box), feeds into a final '4b-Task' box.

The left sidebar contains a 'Data preparation' section with a tree view showing 'All', 'District', and 'Salary'. Below it are sections for 'Along connect', 'Against connect', 'Network', and 'Archive'. A 'New Boxes' section lists various tool categories like 'Data preparation', 'Guba Mining', 'Language', etc.

The right sidebar shows the 'PROPERTIES' panel for the selected 'District' box. It contains the following information:

- Name in res:** District
- Categories:** Cost of ca.77
- X-Category:** District
- Semantic:** Visible sockets
- Discrete:**
 - Frees: False
 - Name in res: False
- To:**
 - To: False
 - X-Category: False

Below the properties panel, there is a 'Context Help' section with the text: 'How to start in Ferda? District Each value one category Each value of the column becomes one category'.

Ferda as programming language

Box as function

Ferda as programming language

Box as function

- Box consists of functions

Ferda as programming language

Box as function

- Box consists of functions
- Sockets are parameters of these functions

Ferda as programming language

Box as function

- Box consists of functions
- Sockets are parameters of these functions
- Property is also socket

The diagram shows a box labeled "Founded Implication" with a socket on its left side. A green box with a yellow glow and the value "0,5" is connected to this socket. To the right, a configuration panel for the box is shown with the following settings:

Threshold	0,5
Units	Irrelevant
Visible sockets	
Missing Inf	False
Operation	False
Relation	False
Threshold	True
	True
	False

Threshold
Sets if the property is visible as a socket

What is missing?

What is missing?

- Moving work from one project to another

What is missing?

What is missing?

- Moving work from one project to another
- Basic math boxes

What is missing?

What is missing?

- Moving work from one project to another
- Basic math boxes
- Recursion

What is missing?

What is missing?

- Moving work from one project to another
- Basic math boxes
- Recursion
- Other language boxes

What is missing?

What is missing?

- Moving work from one project to another
- Basic math boxes
- Recursion
- Other language boxes
- Ferda specific language boxes

What is missing?

What is missing?

- Moving work from one project to another
- Basic math boxes
- Recursion
- Other language boxes
- Ferda specific language boxes
- Data mining specific boxes for user programming

Network archive

What is network archive?

- New place where user can store connection

Network archive

What is network archive?

- New place where user can store connection
- Independent on project

Network archive

What is network archive?

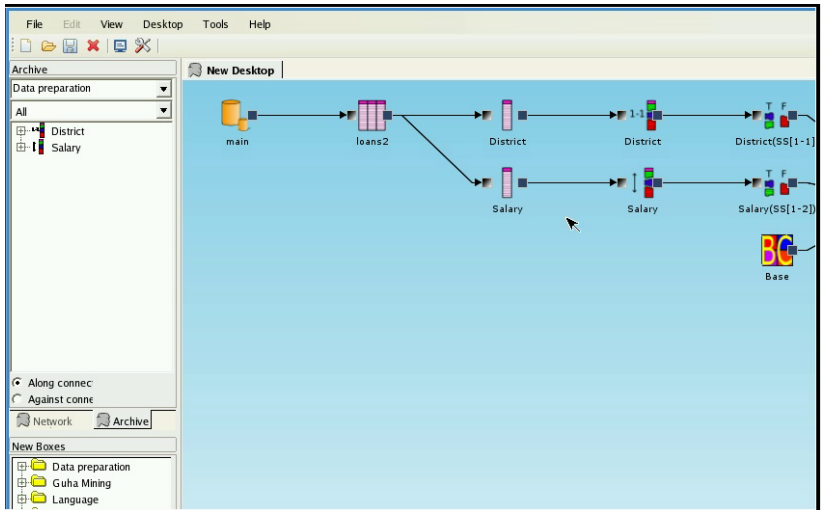
- New place where user can store connection
- Independent on project
- One network archive can be accessed from more computers

Network archive

What is network archive?

- New place where user can store connection
- Independent on project
- One network archive can be accessed from more computers
- Way how to move connections from one project to another

Movie – network archive



Screenshot – add a connection to the network archive

The screenshot displays the Ferda software interface. On the left, there is a 'NETWORK ARCHIVE' panel with a 'New Boxes' list containing folders for 'Data preparation', 'Guha Mining', 'Language', 'Ontology Related Boxes', 'other', 'Sample Boxes', and 'Wizards'. The main workspace, titled 'NEW Desktop', shows a workflow diagram with boxes: 'main', 'loans2', 'Column', 'Salary', 'District', 'District(SS[1-1])', 'Salary(SS[1-2])', and 'Base'. A context menu is open over the 'Salary' box, listing various actions. The 'Add to network archive' option is highlighted, with the keyboard shortcut 'Ctrl+R'.

Action	Keyboard Shortcut
Modules for interaction	
Boxes asking for creation	
Layout	
Rename	F2
Copy	Ctrl+C
Clone	Ctrl+E
Validate	Ctrl+Q
Add to network archive	Ctrl+R
Delete from Desktop	Ctrl+D
Delete from Archive	Shift+Delete
Localize in Archive	Ctrl+L
Pack all Sockets	Ctrl+P

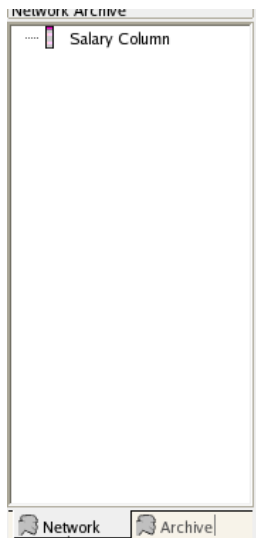
Screenshot – set a name of box in the network archive

The screenshot displays the Ferda Network Archive interface. On the left, a window titled 'NETWORK ARCHIVE' is empty. The main workspace, titled 'NEW DESKTOP', shows a workflow diagram with the following components and connections:

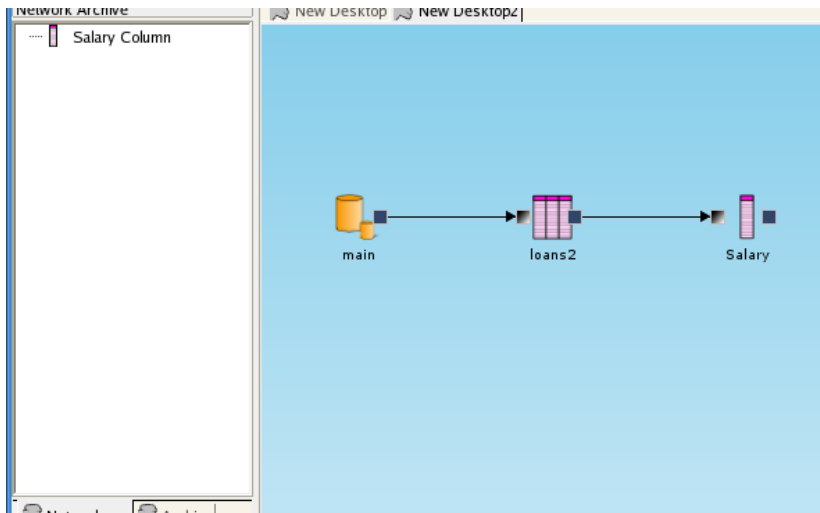
- main** (cylinder icon) connects to **loans2** (table icon).
- loans2** branches into two paths:
 - One path goes through a **Column** box (table icon) to a **Salary** box (table icon).
 - The other path goes through a **1-1** box (table icon) to a **District** box (table icon).
- The **Salary** box connects to a **Salary(SS[1-2])** box (table icon).
- The **District** box connects to a **District(SS[1-1])** box (table icon).
- Both **Salary(SS[1-2])** and **District(SS[1-1])** connect to a **Base** box (table icon).

A dialog box titled 'Label of the box in the netw' is open in the foreground, with the text 'Salary Column' entered in the input field. The dialog has 'OK' and 'Cancel' buttons.

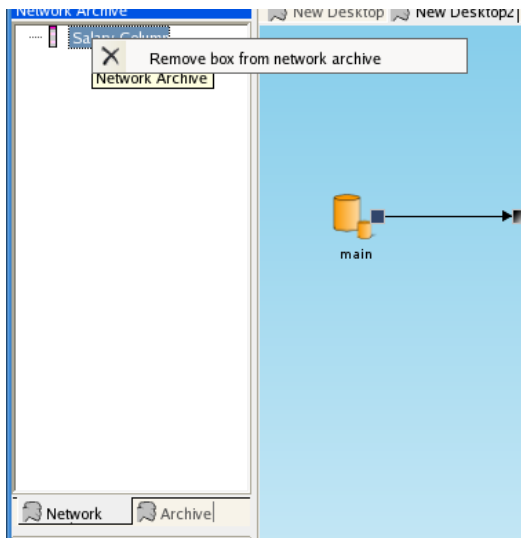
Screenshot – new box added to the network archive



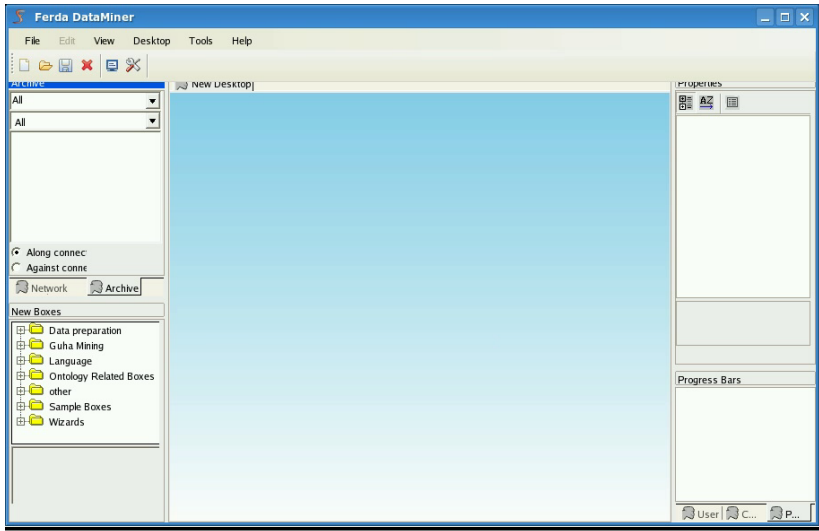
Screenshot – drop box to a desktop from the n. archive



Screenshot – remove box from the network archive



Movie – Binary operation



Binary operation

The screenshot shows the Ferda DataMiner software interface. The main workspace displays a workflow diagram with three boxes: two input boxes labeled '3' and '5', a central 'Binary operation' box with a red plus and asterisk icon, and an output box labeled 'result'. Arrows connect the input boxes to the operation box, and the operation box to the result box.

The left sidebar contains a project tree with the following structure:

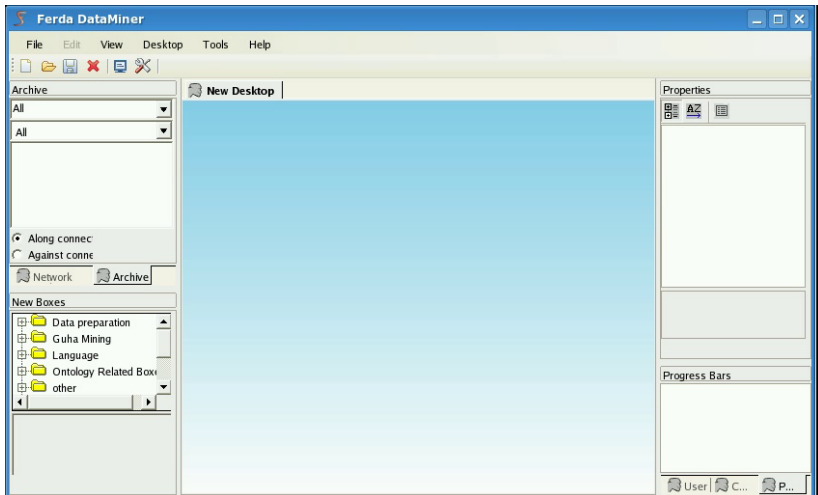
- All
 - NA 3
 - NA 5
 - Binary operation
 - result
- Along connec
- Against connec
- Network
- Archive
- New Boxes
 - Lambda
 - Variable
 - Math
 - Binary operatk
 - Compare
 - if-then-else
 - Ontology Related Boxes
- Binary operation

The right sidebar shows the Properties panel for the selected 'Binary operation' box:

value	15
Visible sockets	
value	True

Below the Properties panel is a section for Progress Bars, which is currently empty.

Movie – Comparison



Comparison

The screenshot displays the Ferda DataMiner application window. The main workspace shows a workflow diagram with three boxes: two input boxes labeled '1' and '5', a central comparison box labeled '<=' (less than or equal to), and an output box labeled 'result'. Arrows indicate the flow from the input boxes to the comparison box, and then to the result box.

On the left side, there is a 'New Boxes' panel with a tree view containing folders like 'Math' and 'Ontology Related Boxes', and a 'BoolT' box. Below it are options for 'Along connec' and 'Against connec', and buttons for 'Network' and 'Archive'.

On the right side, a 'Properties' panel is visible. It has a dropdown menu set to 'value' with 'True' selected. Below it, a 'Visible sockets' section shows 'value' set to 'True'. There are also 'Progress Bars' and a status bar at the bottom right showing 'User', 'C...', and 'P...'.

Movie – If expression

The screenshot displays the Ferda DataMiner application window. The title bar reads "Ferda DataMiner". The menu bar includes "File", "Edit", "View", "Desktop", "Tools", and "Help". Below the menu bar is a toolbar with icons for file operations. The main workspace is titled "New Desktop" and contains a workflow diagram. On the left, there are two green square boxes labeled "1" and "5". Arrows from both boxes point to a central yellow and black box representing an "if-then-else" expression. An arrow from this central box points to a final green square box labeled "result".

On the left side of the interface, there is a "New Boxes" panel. It shows a tree structure with "Variable" and "Math" folders. Under "Math", there are sub-items for "Binary operati...", "Compare", and "if-then-else". Below this panel, the text "(if ? (then) : (else))" is visible. To the right of the main workspace is a "Properties" panel, which is currently empty. At the bottom right, there is a "Progress Bars" section, also empty. The bottom status bar shows "User | C... | P..." and navigation icons.

If expressions

The screenshot shows the Ferda DataMiner application window. The main workspace displays a workflow diagram with the following components:

- Two input boxes labeled "1" and "5".
- A central comparison box labeled "<=".
- An "if-then-else" control box.
- A final output box labeled "result".

The workflow logic is as follows: The inputs "1" and "5" are fed into the "<=" comparison box. The output of this comparison is connected to the "if-then-else" box. The "if-then-else" box has two paths: one leading to the "result" box and another that loops back to the "<=" comparison box, indicating a recursive or iterative process.

On the left side, the "New Boxes" panel is visible, showing a tree structure with categories like "Variable", "Math", "Binary operatix", "Compare", and "if-then-else".

On the right side, the "Properties" panel is open, showing the following configuration:

value	5
Visible sockets	
value	True

Below the Properties panel, there is a "Progress Bars" section and a status bar at the bottom showing "User | C... | P..."

Lambda expression

Basic facts

- From lambda calculus $(\lambda x.(1 + x))(9)$

Lambda expression

Basic facts

- From lambda calculus $(\lambda x.(1 + x))(9)$
- Basics of functional programming

Lambda expression

Basic facts

- From lambda calculus $(\lambda x.(1 + x))(9)$
- Basics of functional programming

Lambda in C# 3

```
public delegate int function(int x);

public static void Main(string[] args)
{
    function plusOne = x => 1 + x;
    var a = plusOne(9);
    System.Console.WriteLine(a);
}
```

Lambda expression

Other languages

Lambda in F#

```
let onePlus x = 1 + x
do printf "%s" (onePlus(9))
```

Lambda expression

Other languages

Lambda in F#

```
let onePlus x = 1 + x  
do printf "%s" (onePlus(9))
```

Lambda in Python

```
plusOne = lambda x: 1 + x  
print plusOne(9)
```

Movie – Lambda basics in Ferda

The screenshot displays the Ferda DataMiner application window. The main workspace shows a workflow diagram with three boxes: '1', '5', and 'result'. Box '1' and box '5' are connected to a central box labeled with a red plus sign and a red asterisk, indicating a mathematical operation. Box 'result' is positioned to the right. The left sidebar contains a 'New Boxes' list with the following items: Command Output, Execute action, Get parameter, Lambda, and Variable. The 'Lambda' box is highlighted. The top menu bar includes File, Edit, View, Desktop, Tools, and Help. The bottom status bar shows 'User', 'C...', and 'P...'.

Basic use in Ferda

Lambda without parameters

The screenshot shows the Ferda DataMiner application window. The main workspace displays a workflow diagram with the following components:

- Two input boxes labeled '1' and '5'.
- A central mathematical operation box (represented by a plus and multiplication sign).
- A 'Lambda' box.
- A final output box labeled 'result'.

The 'Properties' panel on the right side of the window shows the following details:

value	
value	6
Visible sockets	
value	True

Below the Properties panel, there is a section for 'Progress Bars' which is currently empty.

Basic use in Ferda

One constant parameter specified

The screenshot shows the Ferda DataMiner software interface. The main workspace displays a workflow diagram with the following components:

- Input Boxes:** Four boxes labeled '1', '5', '9', and 'Lambda' are connected to a central operation box.
- Operation Box:** A box containing a red plus sign and a red asterisk, representing a mathematical operation.
- Output Box:** A box labeled 'result' is connected to the operation box.

The interface includes the following panels and elements:

- Menu Bar:** File, Edit, View, Desktop, Tools, Help.
- Toolbars:** Standard file and editing icons.
- Archive Panel:** Shows a tree structure with folders like '+', 'NA 1', 'NA 5', 'NA 9', and 'Lambda'.
- New Boxes Panel:** Lists various data types: DateT, DateTimeT, DoubleT, FloatT, and Group.
- Properties Panel:** Shows settings for the selected box, including 'value' (10) and 'Visible sockets' (True).
- Status Bar:** Displays 'User', 'C...', and 'P...'.

Implementation of lambda

How it really works

Algorithm

- Values of variables are cloned (whole subtree)

Implementation of lambda

How it really works

Algorithm

- Values of variables are cloned (whole subtree)
- Main function is cloned with substitution and returned

Factorial in C#

Structural version

First version of factorial

```
public static int Factorial(int x)
{
    if (x == 0)
    {
        return 1;
    }
    else
    {
        return x * Factorial(x - 1);
    }
}
```

Factorial in C#

Expression version

Second version of factorial

```
public static int Factorial2(int x)
{
    return (x == 0) ? 1 : x * Factorial2(x - 1);
}
```

Factorial in other languages

Python

```
fac = lambda x: x == 0 and 1 or x * fac(x - 1)
```

Factorial in other languages

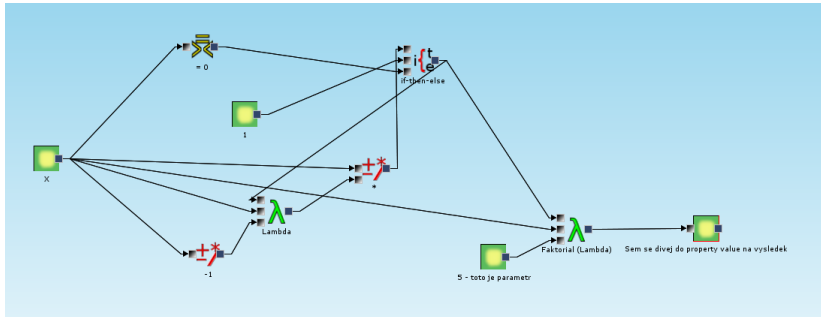
Python

```
fac = lambda x: x == 0 and 1 or x * fac(x - 1)
```

F#

```
let rec factorial n =  
    if n=0 then 1 else n * factorial(n - 1)
```

Factorial in Ferda



Movie – Get parameter

The screenshot displays the Ferda DataMiner application window. The main workspace, titled "new Desktop", contains a workflow diagram with the following components:

- Two input boxes labeled "1" and "5".
- A central box labeled "if-then-else" with a red border, which is the focus of the "Movie – Get parameter" task.
- A final output box labeled "result".
- Arrows indicate data flow from the input boxes to the "if-then-else" box, and from the "if-then-else" box to the "result" box.

The interface includes a menu bar (File, Edit, View, Desktop, Tools, Help), a toolbar, and several panels:

- Archive:** Shows a tree view of saved workflows, including "All", "NA 1", "NA 5", "if-then-else", and "result".
- New Boxes:** Lists available components such as "Variable", "Math", "Binary operatix", "Compare", and "if-then-else".
- Properties:** A panel on the right for configuring the selected box.
- Progress Bars:** A section at the bottom right for monitoring task progress.

At the bottom of the window, there are status indicators for "User", "C...", and "P...", along with navigation icons.

Get parameter

The screenshot displays the Ferda DataMiner application window. The main workspace shows a workflow diagram with the following components:

- Two input boxes labeled '1' and '5' (represented by yellow circles).
- A central 'if-then-else' box.
- A 'Get parameter' box (represented by a green square).
- A final 'result' box (represented by a yellow circle).

Arrows indicate the flow of data from the input boxes through the 'if-then-else' box to the 'Get parameter' box, and finally to the 'result' box.

On the right side, the 'Properties' panel is visible, showing the configuration for the selected 'Get parameter' box:

- Main**: Name of parameter: then
- Visible sockets**: Name of parameter: False
- Name of parameter**: Name of parameter, which we want to get
- Progress Bars**: (Empty)

The left sidebar contains a 'New Boxes' list with categories like Execute action, Get parameter, Lambda, Variable, and Math. Below this list, a text box reads: 'Gets value of other box's parameter'.

Movie – Execute action

The screenshot shows the Ferda DataMiner application window titled "rekurzivniVypocetVysledek - Ferda DataMiner". The interface includes a menu bar (File, Edit, View, Desktop, Tools, Help) and a toolbar with icons for file operations and workflow management. On the left, there is a "New Boxes" panel with a list of data types: BoolT, DateT, DateTimeT, and DoubleT. The main workspace contains a workflow diagram with two boxes: "4ft-Task" and "result". The "result" box is highlighted with a green glow. On the right, the "Properties" panel is open, showing a "value" field with the number "0" and a "Visible sockets" section with a "value" field set to "True". At the bottom of the window, there are status indicators for "User", "C...", and "P...".

Execute action

The screenshot shows the Ferda DataMiner interface with a workflow diagram and a properties panel.

Workflow Diagram:

- The workflow starts with a **4ft-Task** box.
- It branches into two parallel paths:
 - Path 1: **4ft-Task** → **Execute action** box.
 - Path 2: **4ft-Task** → **Get parameter** box.
- The outputs of both paths converge into a single **Execute action** box.
- The final output is a **result** box.

Properties Panel:

The Properties panel on the right shows the configuration for the selected **Execute action** box:

- value:** 62
- Visible sockets:**

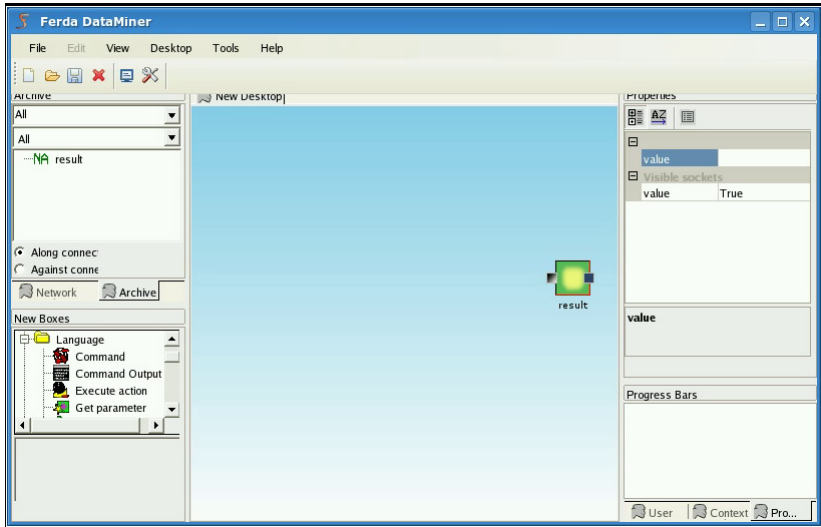
value	True
-------	------

Left Panel (New Boxes):

- Command
- Command Output
- Execute action
- Get parameter

Description: Executes action of first box and returns value of second box

Movie – Command and command output



Command and command output

Example in ferda

The screenshot shows the Ferda DataMiner application window. The main workspace displays a workflow diagram with four boxes connected in a sequence: "echo 'test'", "sed 's/t/m/g'", "Command Output", and "result".

The left sidebar contains a tree view of the workspace. Under "NEW DESKTOP", there is a folder "Archive" containing a "Command Output" box, an "echo 'test'" box, a "result" box, and a "sed 's/t/m/g'" box. Below this, there are options for "Along connec" and "Against connec", and buttons for "Network" and "Archive".

The bottom-left sidebar shows "New Boxes" with a tree view containing "Language", "Command", "Command Output", "Execute action", and "Get parameter". Below this is a "Command" input field.

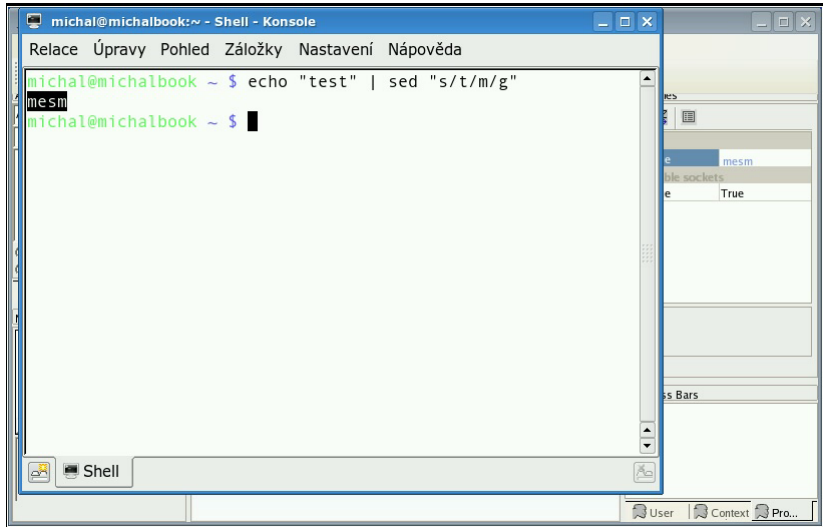
The right sidebar shows the "Properties" panel for the selected "Command Output" box. It contains a table with the following data:

value	mesm
visible sockets	
value	True

Below the table, there is a "value" label and a "Progress Bars" section.

The bottom status bar shows "User", "Context", and "Pro...".

Command and command output

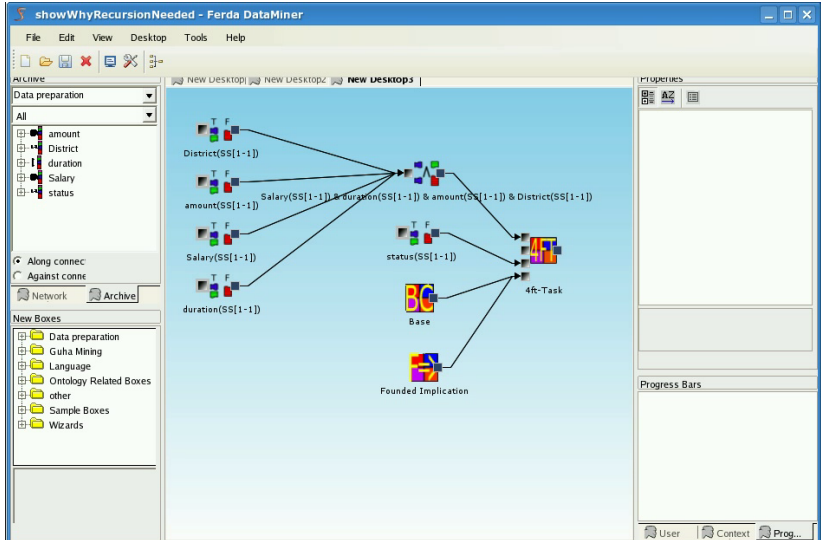


The screenshot shows a terminal window titled "michal@michalbook:~ - Shell - Konsole". The window has a menu bar with "Relace", "Úpravy", "Pohled", "Záložky", "Nastavení", and "Nápověda". The terminal content shows the following sequence of events:

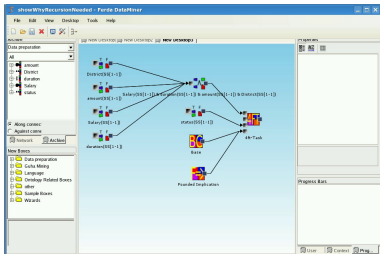
```
michal@michalbook ~ $ echo "test" | sed "s/t/m/g"
mesm
michal@michalbook ~ $
```

The output "mesm" is highlighted in blue. The terminal window is part of a larger desktop environment, with a taskbar at the bottom showing "Shell" and "User", "Context", "Pro..." icons. The system tray at the bottom right contains navigation icons.

Movie – When lambda can be useful – 4FT task

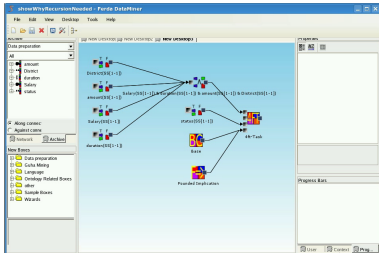


When lambda can be useful – 4FT task



Problem

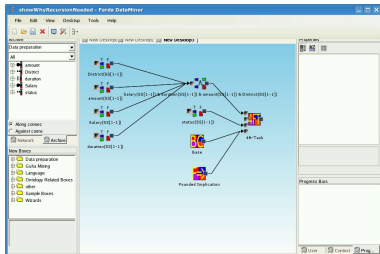
When lambda can be useful – 4FT task



Problem

- User tries some setting of quantifiers

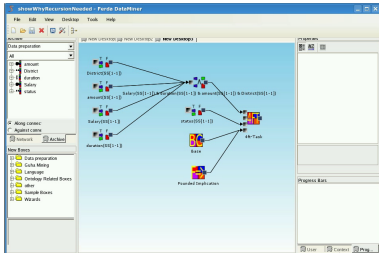
When lambda can be useful – 4FT task



Problem

- User tries some setting of quantifiers
- If he fails, he tries again with other settings?

When lambda can be useful – 4FT task



Problem

- User tries some setting of quantifiers
- If he fails, he tries again with other settings?
- It's manual and confusing

User specified automation of settings

User wants

- find the best settings for task

User specified automation of settings

User wants

- find the best settings for task
- suggest way how to find best settings

User specified automation of settings

User wants

- find the best settings for task
- suggest way how to find best settings
- have for different tasks different methods

User specified automation of settings

User wants

- find the best settings for task
- suggest way how to find best settings
- have for different tasks different methods

Programming finding best settings by user

User specified automation of settings

User wants

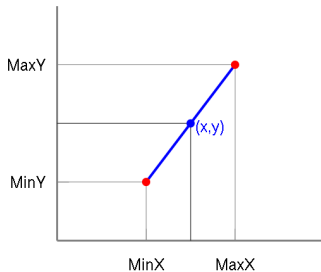
- find the best settings for task
- suggest way how to find best settings
- have for different tasks different methods

Programming finding best settings by user

- biggest variability

Linear interpolation

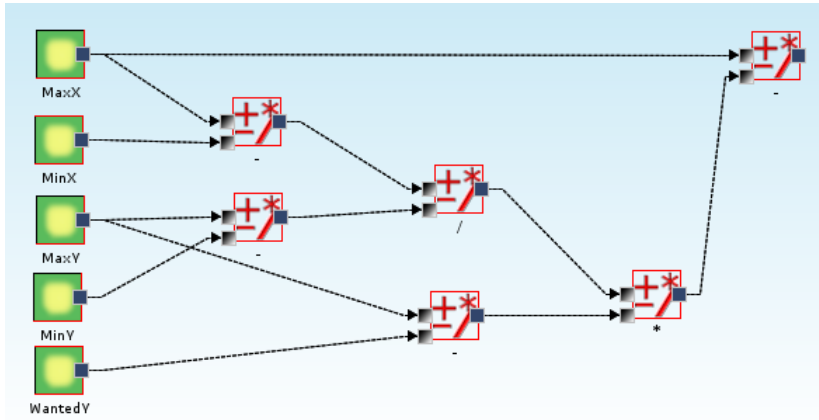
Basics



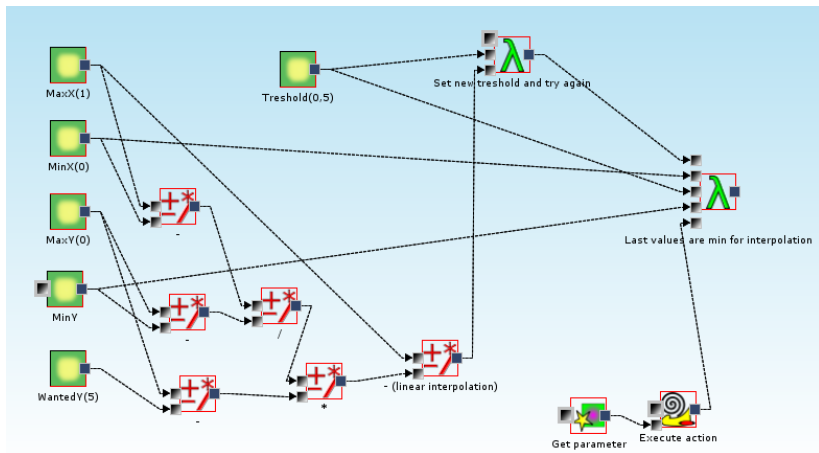
Formula

$$x = \text{MaxX} - (\text{MaxY} - \text{WantedY}) \frac{\text{MaxX} - \text{MinX}}{\text{MaxY} - \text{MinY}}$$

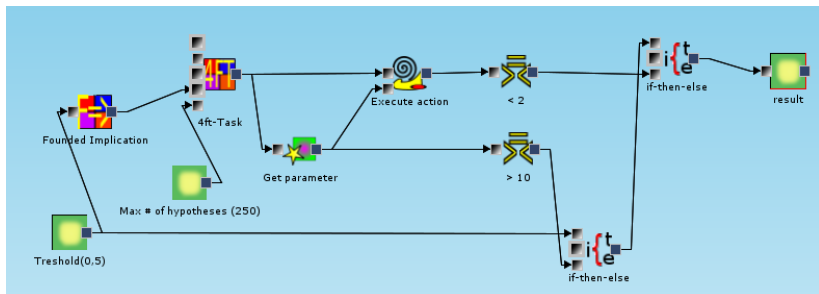
Interpolation as connection of boxes

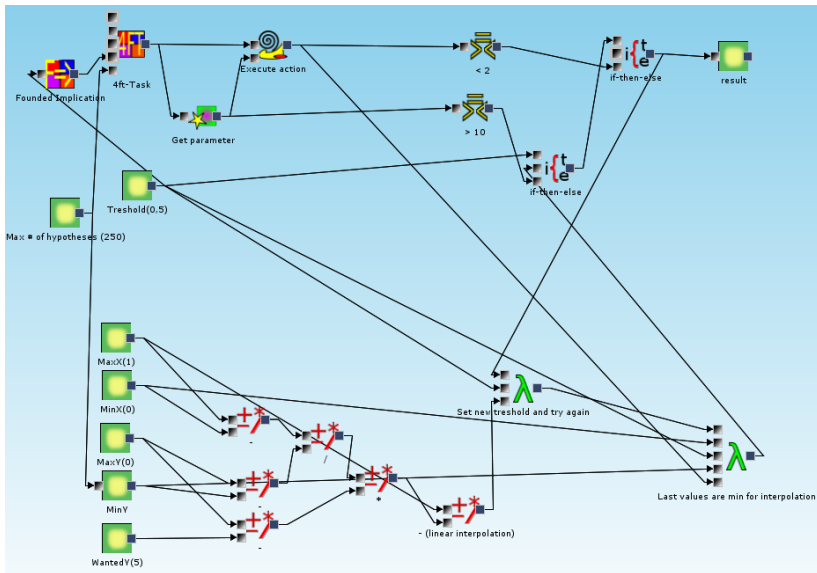


Result should be between

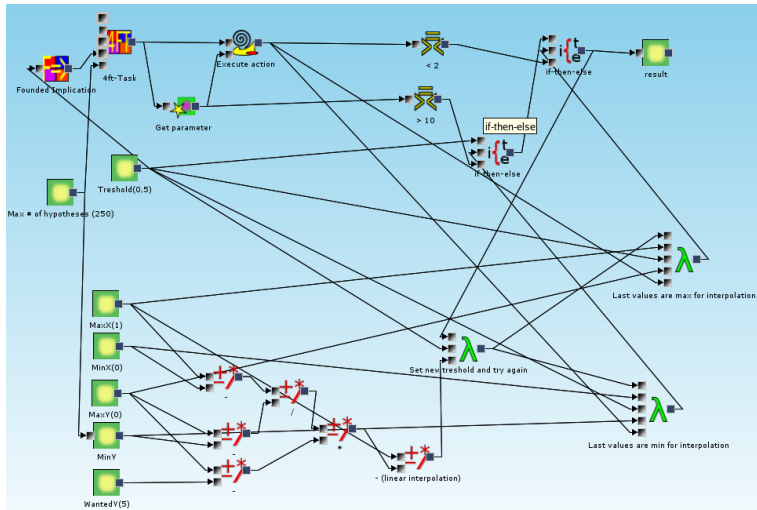


If not interpolate and set new max/min

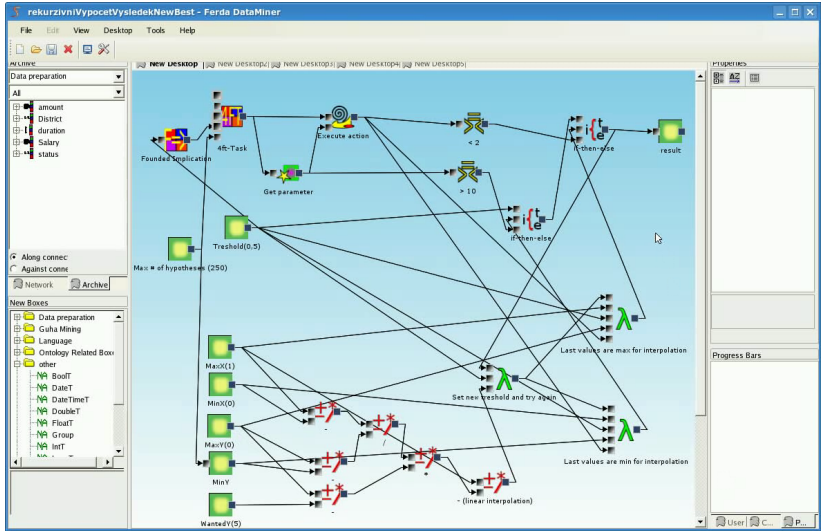




Connection



Movie – Result



How to do it better

Is the recursive computing of 4FT needed

How to do it better

Is the recursive computing of 4FT needed

No – BestN algorithm

Is user programming needed

How to do it better

Is the recursive computing of 4FT needed

No – BestN algorithm

Is user programming needed

How to do it better

Is the recursive computing of 4FT needed

No – BestN algorithm

Is user programming needed

Yes – even with BestN for different projects different value function

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

- Make a sequence of functions

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

- Make a sequence of functions
- Do something for each item in the sequence

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

- Make a sequence of functions
- Do something for each item in the sequence
- Add an item

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

- Make a sequence of functions
- Do something for each item in the sequence
- Add an item
- Subsequence

Sequences and sets

Do we need sequences?

- Every mature programming language has something like sequences
- Sequences can be simulated, but the price

What we would like to do with sequences

- Make a sequence of functions
- Do something for each item in the sequence
- Add an item
- Subsequence
- Concatenation

What should be done for support of sequences?

New boxes

- Sequence as head/tail

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate
- New group box

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate
- New group box
- Box for conversion from sequence to group box

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate
- New group box
- Box for conversion from sequence to group box

Changes to Ferda core

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate
- New group box
- Box for conversion from sequence to group box

Changes to Ferda core

What should be done for support of sequences?

New boxes

- Sequence as head/tail
- Sequence as array of items
- ForEach
- AddItem, Subsequence, Concatenate
- New group box
- Box for conversion from sequence to group box

Changes to Ferda core

Socket should accept new group box the same way it accepts the old one

Sequence example on DM

Problem

You have a table and don't know anything about. You would like to know something about it.

Resolution

Sequence example on DM

Problem

You have a table and don't know anything about. You would like to know something about it.

Resolution

- Connect all columns you want to analyze to the sequence

Sequence example on DM

Problem

You have a table and don't know anything about. You would like to know something about it.

Resolution

- Connect all columns you want to analyze to the sequence
- Regarding types of columns and their data, create attributes for each column in the sequence

Sequence example on DM

Problem

You have a table and don't know anything about. You would like to know something about it.

Resolution

- Connect all columns you want to analyze to the sequence
- Regarding types of columns and their data, create attributes for each column in the sequence
- Create basic tasks

Sequence example on DM

Problem

You have a table and don't know anything about. You would like to know something about it.

Resolution

- Connect all columns you want to analyze to the sequence
- Regarding types of columns and their data, create attributes for each column in the sequence
- Create basic tasks
- See result of these tasks

Reuse of code

Problem description

Problem

The user wants to reuse boxes which he created.

We have

Reuse of code

Problem description

Problem

The user wants to reuse boxes which he created.

We have

- Project files

Reuse of code

Problem description

Problem

The user wants to reuse boxes which he created.

We have

- Project files
- Network archive

Reuse of code

Better network archive

What can be done

- Labels

Reuse of code

Better network archive

What can be done

- Labels
- More network archives

Reuse of code

Better network archive

What can be done

- Labels
- More network archives
- User access rights

Reuse of code

Project files

User could load boxes from other project

- Easy to implement

Reuse of code

Project files

User could load boxes from other project

- Easy to implement
- There is workaround with network archive

Reuse of code

Project files

User could load boxes from other project

- Easy to implement
- There is workaround with network archive

Including other project

Reuse of code

Project files

User could load boxes from other project

- Easy to implement
- There is workaround with network archive

Including other project

- Like other programming languages

Reuse of code

Project files

User could load boxes from other project

- Easy to implement
- There is workaround with network archive

Including other project

- Like other programming languages
- Strong

Reuse of code

Project files

User could load boxes from other project

- Easy to implement
- There is workaround with network archive

Including other project

- Like other programming languages
- Strong
- Harder to implement

Better lambda

- Parameter could be function with parameter, not only constant

Better lambda

- Parameter could be function with parameter, not only constant
- Lambda is slow

Better lambda

- Parameter could be function with parameter, not only constant
- Lambda is slow

How to make lambda quicker

Better lambda

- Parameter could be function with parameter, not only constant
- Lambda is slow

How to make lambda quicker

- Quicker creating of boxes

Better lambda

- Parameter could be function with parameter, not only constant
- Lambda is slow

How to make lambda quicker

- Quicker creating of boxes
- Functions could be called only once

Better lambda

- Parameter could be function with parameter, not only constant
- Lambda is slow

How to make lambda quicker

- Quicker creating of boxes
- Functions could be called only once
- Boxes could be cloned only in time it is really needed

Other things to do

- Most of modules for interactions should work on top of functions not boxes

Other things to do

- Most of modules for interacioints should work on top of functions not boxes
- Boxes for GUHA – parts

Other things to do

- Most of modules for interactions should work on top of functions not boxes
- Boxes for GUHA – parts
- User could choose on which computer a box is running

Other things to do

- Most of modules for interactions should work on top of functions not boxes
- Boxes for GUHA – parts
- User could choose on which computer a box is running
- Box which can be programmed by user at runtime in some scripting language

Summary

Programming language in Ferda should

- offer more power for data mining

Summary

Programming language in Ferda should

- offer more power for data mining
- allow to do things which were not designed before

Summary

Programming language in Ferda should

- offer more power for data mining
- allow to do things which were not designed before

Next steps

Summary

Programming language in Ferda should

- offer more power for data mining
- allow to do things which were not designed before

Next steps

- there are many things to do