### Life, Venus and Everything

Björn Grieger (Aurora Technology B. V. for the European Space Agency, ESAC, Spain)



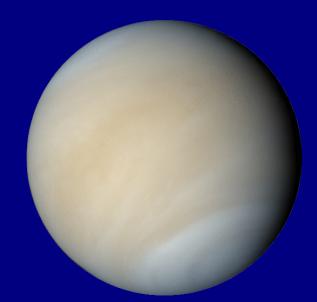
Life. Venus and Everything

Björn Grieger

Spreadsheets



### Venus in the visible



Life, Venus and Everything

Björn Grieger

#### Venera

Magella

EnVision

Functional programming

Overview

Spreadsheets

Spreadsheets

openox

The \*nix make utility

The arcs wrapper langua

for make A dedicated data flow

A dedicated data flow language

Example: Envisionary Embedding it in a Jupy Notebook

Snomonic projection Dump, occultation and travity

gravity The dataflow C++ emplate library

unctional progran ummary

P<sub>ro</sub>vision





### Venus in the visible





Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctiona

-unctional programming

Spreadsheets

The \*nix make utility

for make

A dedicated data flow language

Example: Envisionary Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation and

The dataflow C++

Functional programm summary

Provision





# Venera 13 (1981–1982)







The lander survived more than two hours at 460°C and 90 bar.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper language

for make
A dedicated data flow

A dedicated data flow language

Example: Envisionary Embedding it in a Jupyte Notebook

nomonic projection ump, occultation and avity

ravity he dataflow C+molate library

emplate library unctional program

vision

Toko k



# Venera 13 lander panorama



Life, Venus and Everything

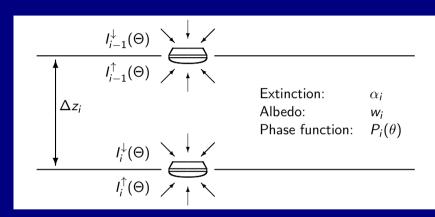
Björn Grieger

#### Venera





### Titan Inverse Radiation Model (TIRM)



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

rogramming

Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flow

A dedicated data flow anguage

anguage Example: Envisionary

mbedding it in a Jupy otebook

nomonic projection lump, occultation an ravity

e dataflow C++ nplate library

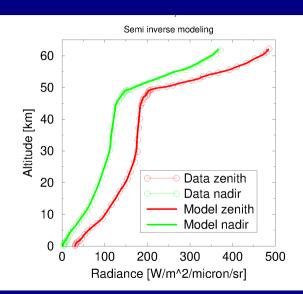
unctional progran ummary

P<sub>ro</sub>vision





### TIRM, Venus version



Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functiona

programming

Spreadsheets

OpenDX

The wais a

The arcs wrapper languag

r make A dedicated data flow

inguage

example: Envisionary

nbedding it in a Jupy otebook nomonic projection

Oump, occultation a

The dataflow (

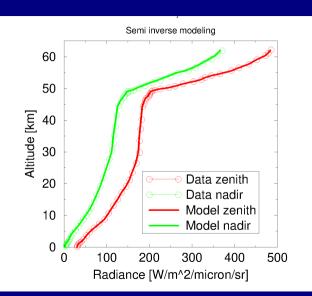
Functional progr summary

Provision





### TIRM, Venus version



Life. Venus and Everything Spreadsheets

Björn Grieger

Venera





62 km

50 km

20 km

10 km

5 km

2 km

1 km

0 km

# Surface from Venera 13 panorama + sky from TIRM



Life. Venus and Everything

Björn Grieger

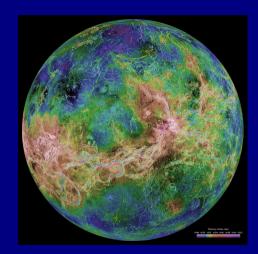
Venera





### The Magellan radar mission





Life, Venus and Everything

Björn Grieger

Vener

### Magellan

**EnVision** 

unctiona

<sup>-</sup>unctional programming

Overview

OpenDX

The arcs wrapper language

A dedicated data flow language

language
Example: Envisionary

nbedding it in a Jupy otebook

ump, occultation ar avity

nplate library nctional programm

Provision





The Death of Magellan

Life, Venus and Everything

Björn Grieger

Venera

Magellan

FnVision

LIIVISIOII

Functional programming

Overview

Spreadsheets

OpenDX The ∗nix make

The arcs wrapper languag

dedicated data flow

inguage xample: Envisionar

mbedding it in a Jupyl lotebook

ump, occultation an avity

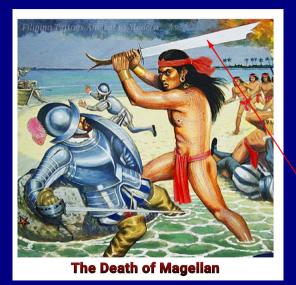
emplate library unctional program

Provision

rrovisioii

rake-nor





🔨 Kampilar

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVisior

- ..

programming

Overview

OpenDX

The ∗nix make u

The arcs wrapper languag

dedicated data flow nguage

anguage ixample: Envisionary

Embedding it in a Jupy Notebook

ump, occultation ar avity

The dataflow C emplate library

Functional programmary

rovision

Take-hoi





The Death of Magellan



Kampilar

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

EnVision

unctiona

rogramming

Spreadsheets

The \*nix make

The arcs wrapper language for make

A dedicated data flow language

Example: Envisionary
Embedding it in a Jup

Embedding it in a Jup Notebook Gnomonic projection

lump, occultation an ravity se dataflow C++ nplate library

unctional programn ummary

rovision

Take-I





The Death of Magellan



прпап

(ris

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVisio

Function:

programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language
for make

A dedicated data flow anguage

Example: Envisionary

Embedding it in a Jup Notebook

Dump, occultation and gravity
the dataflow C++

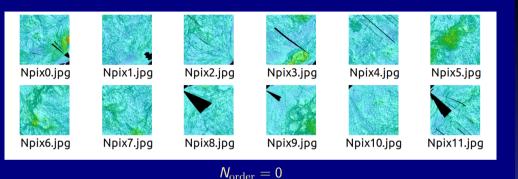
Functional program summary

Provision

Take-I



# **Hierarchical Progressive Survey (HiPS)**



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Envision

unctional

verview

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper language

for make

A dedicated data flow

A dedicated data flow language

Example: Envisionary

Embedding it in a Jupy Notebook

> nomonic projection ump, occultation an avity

he dataflow C+

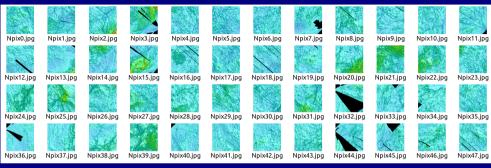
unctional program

P<sub>ro</sub>vision





# **Hierarchical Progressive Survey (HiPS)**



 $N_{\rm order} = 1$ 

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Eunstions

programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data flow language

language
Example: Envisionary

Embedding it in a Jupy Notebook

omonic projection

Imp, occultation and

ne dataflow C++
mplate library

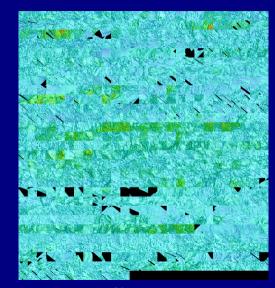
nmary

P<sub>ro</sub>vision





# Hierarchical Progressive Survey (HiPS)



Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

EIIVISIOII

**Functional** 

programmin

Overview

Spreadsheet

OpenDX

The \*nix make utility

he arcs wrapper langua or make

A dedicated data flow anguage

language Example: Envisionary

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation as gravity he dataflow C++

emplate library unctional programr

Provision

Fako homo







### **Options for displaying HiPS**

- 1. Locally with the application Aladin Desktop, or
- 2. embedded in a web page with Aladin Lite:

http://comsim.esac.esa.int/rossim/bgrieger/VENUS

Life, Venus and Everything

Björn Grieger

Venera

Magellan

FnVision

Envision

Functional programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flow

A dedicated data flow anguage

xample: Envisionary

mbedding it in a Jupyl lotebook

nomonic projection ump, occultation an avity

gravity
he dataflow C++

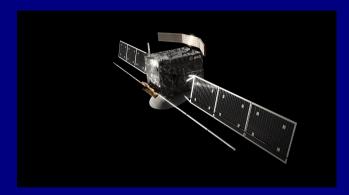
Functional programsummary

Provision





### **EnVision**



### Intruments

- ► Venus Synthetic Aperture Radar (VenSAR)
- ► Venus Subsurface Radar Sounder (SRS)
- ► Venus Spectroscopy Suite (VenSpec)
- ► Radio Science Experiment

### Orbit

- $ightharpoonup 1\frac{1}{2}$  hour period
- Near polar
- ► Inertially fixed orbit plane

Life, Venus and Everything

Björn Grieger

Venera

Magella

**EnVision** 

\_ ...

programmii

Overview Spreadsheets

OpenDX

The stole m

The arcs wrapper langua

The arcs wrapper languag for make

nguage

xample: Envisiona mbedding it in a J

Notebook

Dump, occultation a

The dataflow C template library

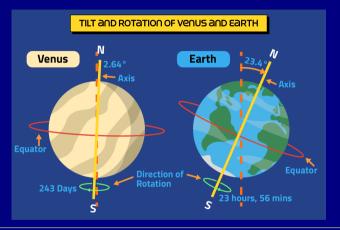
Functional programs summary

Provision





### Venus orbit and rotation



Venus year:

Venus siderial day (1 cycle):

Venus solar day:

EnVision nominal mission:

225 Earth days

243 Earth days

117 Earth days

6 cycles (4 Earth years)

Life. Venus and Everything

Björn Grieger

**EnVision** 

Spreadsheets

OpenDX





### Venus orbit and rotation

Note to self: show maps.pdf

Life, Venus and Everything

Björn Grieger

Venera

Magallar

EnVision

Functional

programmi

Spreadsheets

Spreadsheets

The \*nix make utility

The arcs wrapper langu

for make A dedicated data flow

language

Example: Envisionary Embedding it in a Jupy Notebook

Gnomonic projection Dump, occultation and gravity

gravity he dataflow C++ emplate library

Functional progr summary

Provision



### Mission Analysis and Payload Planning System



Map with field of view overlay (Mars Express)

Life, Venus and Everything

Björn Grieger

Venera

Magell

**EnVision** 

Envision

\_ ..

unctional

rogrammin

Considerate

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

or make

A dedicated data fl anguage

xample: Envision

imbedding it in a Jup Jotebook

nomonic projection

gravity The dataflow C++

template library Functional progr

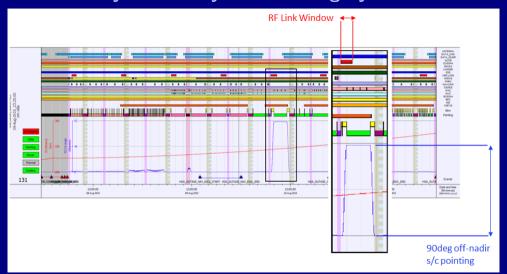
summary

rovision





### Mission Analysis and Payload Planning System



Timeline visualisation (Rosetta)

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

Functional

programming

Overview Spreadsheets

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper language

The arcs wrapper language for make

dedicated data flow

nguage xample: Envisionary

mbedding it in a Jupyt lotebook

Gnomonic projectio Dump, occultation gravity

The dataflow C template library

unctional progranummary

ovision





### Big and strong or small and nimble?

- MAPPS is used by several ESA planetary missions, some of these in operation.
- New developments and modifications have to be carefully planned, priority is given to missions in operation.
- ► MAPPS development cannot quickly react to EnVisions' needs (e.g., importing ROIs from files in weekly changing formats 😉).

Life. Venus and Everything

Biörn Grieger

**EnVision** 

Spreadsheets

OpenDX





### Big and strong or small and nimble?

- MAPPS is used by several ESA planetary missions, some of these in operation.
- ► New developments and modifications have to be carefully planned, priority is given to missions in operation.
- ► MAPPS development cannot quickly react to EnVisions' needs (e.g., importing ROIs from files in weekly changing formats ⑤).
- ⇒ Envisionary, a lightweight tool to supplement (the heavyweight) MAPPS with very specialized functionalities.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

LIIVISIOII

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flow

A dedicated data flor anguage

cample: Envisionary

nbedding it in a Jupy

tebook omonic projection

ump, occultation a

e dataflow C++
nplate library

Functional program summary

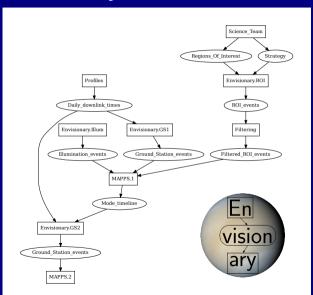
Provision







### **MAPPS** and **Envisionary**



Life, Venus and Everything

Björn Grieger

Venera

\_\_\_\_

**EnVision** 

Functional

programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper languag for make

dedicated data fl iguage

xample: Envisionary

mbedding it in a Jupy otebook

nomonic projectio ump, occultation avity

The dataflow C+ emplate library

Functional programs

rovision





# Functional(?) programming

▶ Developed in the 1950s, mostly used in academics, now big tech companies are picking it up.

- ► I was applying functional programming long before I first heard the phrase (and you probably, too).
- ► The name may not be quite elucidating to many:
  - ► All programming languages know *functions*, some e.g. R know nothing else
  - ► The definition of *pure* functions and the lambda calculus used to describe functional programming are quite abstract.
- ▶ Describing the same thing in terms of data flow is much more intuitive.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Programmii

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language or make

A dedicated data flov language

Example: Envisionary

mbedding it in a Jupy otebook

omonic projection mp, occultation a wity

avity e dataflow C+⊣ aplato library

unctional programi immary

Provision





# Tools using data flow descriptions

- ► (Spreadsheets)<sup>1</sup>
- ► OpenDX (aka IMB DataExplorer)
- ► The \*nix make utility
- ► The arcs² wrapper language for make
- ► The dataflow<sup>2</sup> C++ template library

Note to self: skip to The \*nix make utility

ロト 4 倒 ト 4 三 ト 4 三 ト 9 9 ()

Life, Venus and Everything

Björn Grieger

/enera

Magellan

EnVision

Functional programming

programming

Overview

Spreadsheets

OpenDX

The wale my

The \*nix make utility
The arcs wrapper languag

The arcs wrapper languag for make

for make

A dedicated data flow

language Example: Envisional

nbedding it in a Jup tebook

nomonic projection lump, occultation an ravity

The dataflow C++ emplate library unctional programmin

ımary

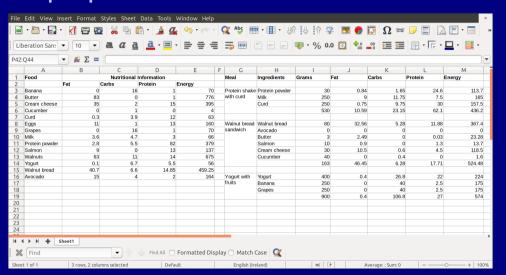
rovision



<sup>&</sup>lt;sup>1</sup>Not using data flow — only discussed to elucidate the not so obvious reason to call functional programming "functional".

<sup>&</sup>lt;sup>2</sup>Home grown.

### A simple spreadsheet



Life. Venus and Everything

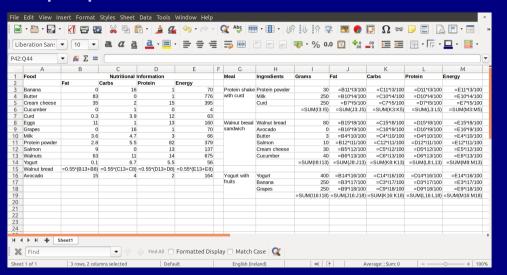
Björn Grieger

Spreadshoots





### A simple spreadsheet



Life. Venus and Everything

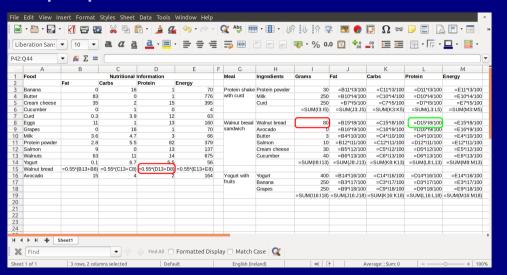
Biörn Grieger

Spreadshoots





### A simple spreadsheet



Life. Venus and Everything

Biörn Grieger

Spreadshoots

OpenDX





### Functions and data flow

- ► A spreadsheet cell can contain a function of values from other cells (which may also contain functions).
- Such a function establishes a data flow between cells.
- Note that you don't write a sequence of commands (imperative programming), but you define a data flow (functional programming). The program decides what to execute, and when.
- ▶ A data flow *can* be described by functions, but it does not *have to*.

Life, Venus and Everything

Björn Grieger

Venera

Magellar

**EnVisior** 

Functional programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data flow

A dedicated data flo anguage

xample: Envisionary mbedding it in a Jupy otebook

nomonic projection ump, occultation ar avity

e dataflow C++ nplate library

Functional programs

Provision





### **OpenDX**

▶ Powerful 3D visualization system introduced by IBM in 1991 as Data Explorer, also promoted by other Unix platform vendors, particularly SGI.

- Envisaged to superseed IDL in the world of scientific visualization, but never really succeded. In 2000 handed over to the open source community as OpenDX. Further development idled out about 2007.
- ► True data-flow implementation, all modules are pure functions (i. e., their outputs are fully defined by their inputs). Hence, processes are stateless with no side effects.
- ► Uses (pure) functions under the hood but in the visual programming interface you directly create a data flow.

Life, Venus and Everything

Björn Grieger

Venera

Magellar

**EnVision** 

Functional

programming

Spreadsheets

#### OpenDX

The \*nix make utility

The arcs wrapper language for make

A dedicated data flow anguage

anguage Example: Envisionary

mbedding it in a Jupy otebook

nomonic projection ump, occultation ar avity

ravity ie dataflow C+ polate library

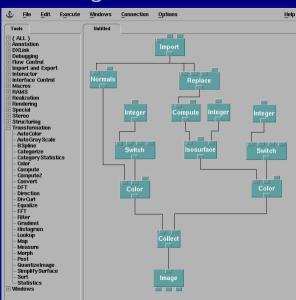
inctional programi immary

Provision





### **Visual Program Editor**



► A program is composed visually.

Life, Venus and Everything

Björn Grieger

Venera

Magella

FnVision

Functional

programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flow

A dedicated data language

xample: Envisionary mbedding it in a Jup

otebook nomonic projection

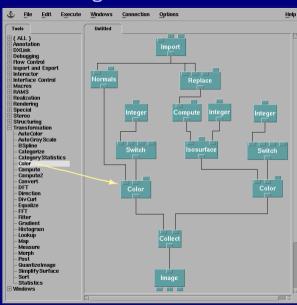
ump, occultation an avity e dataflow C++

implace library inctional prograi immary

rovision



### **Visual Program Editor**



- ► A program is composed visually.
- Modules from the tool bar are placed on the canvas.

Life. Venus and Everything

Biörn Grieger

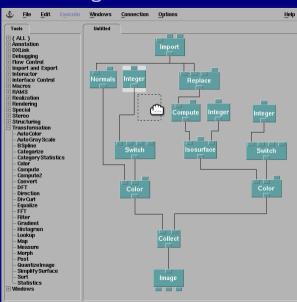
Spreadsheets

OpenDX





### Visual Program Editor



- ► A program is composed visually.
- Modules from the tool bar are placed on the canvas.
- Modules can be dragged around.

Life. Venus and Everything

Biörn Grieger

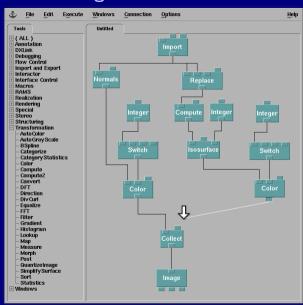
Spreadsheets

OpenDX





## Visual Program Editor



- ► A program is composed visually.
- Modules from the tool bar are placed on the canvas.
- Modules can be dragged around.
- Modules are connected by click and drag.

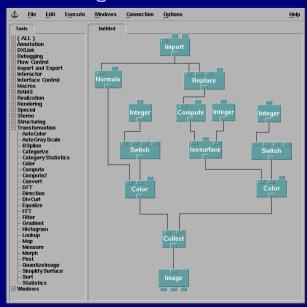
Life. Venus and Everything

Biörn Grieger

OpenDX



## **Visual Program Editor**



- ► A program is composed visually.
- Modules from the tool bar are placed on the canvas.
- Modules can be dragged around.
- ► Modules are connected by click and drag.
- ► The visual program defines a data flow.

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVisior

Functional programming

Overview Spreadsheets

OpenDX

OpenDX

The \*nix make utility
The arcs wrapper language

A dedicated data

language Example: Envis

example: Envisionary Embedding it in a Jup Notebook

nomonic projection ump, occultation an ravity

ne dataflow C+ mplate library

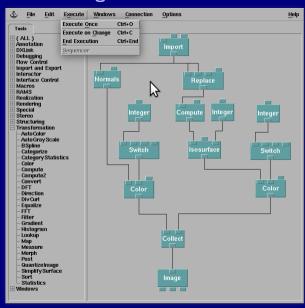
unctional progranummary

rovision





## **Visual Program Editor**



- ► A program is composed visually.
- Modules from the tool bar are placed on the canvas.
- Modules can be dragged around.
- Modules are connected by click and drag.
- ► The visual program defines a data flow.
- ► It can be executed on demand or automatically if something changes.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview Spreadsheets

OpenDX

OpenDX

The \*nix make utility
The arcs wrapper language

A dedicated data flo

language Example: Envisiona

Embedding it in a Ju Notebook

anomonic projection Dump, occultation a gravity

The dataflow Ctemplate library

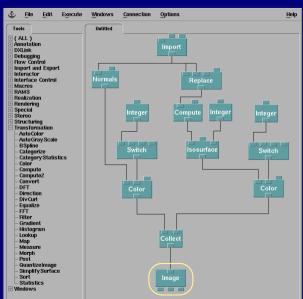
unctional programn ummary

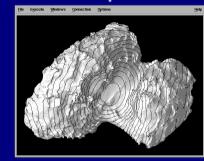
Provision

Take-hon









 When the program is executed, the Image module creates an image. Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview Spreadsheets

OpenDX
The \*nix make utility

The \*nix make utility

The arcs wrapper language

The arcs wrapper languag for make

A dedicated data fl language

Example: Envisionary
Embedding it in a Jupy

nemonic projection nomp, occultation and

e dataflow C++

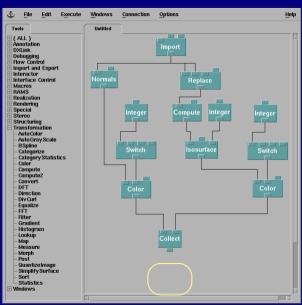
mplate library Inctional program

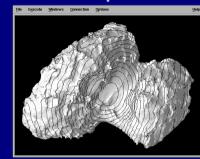
rovision

Take-hon









- ► When the program is executed, the **Image** module creates an image.
- If there was no Image module, the program would not execute at all!

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Functional

programming Overview

OpenDX

The \*nix make utility

The \*nix make utility

The arcs wrapper language

or make
A dedicated data flow

anguage Example: Envisionary

imbedding it in a Jupyt Jotebook

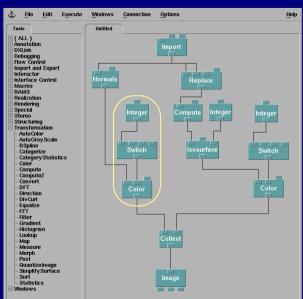
nomonic projection lump, occultation an ravity

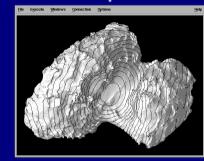
he dataflow C++ emplate library unctional programmir ummary

Provision









► This colors the shape (very quick).

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

Overview
Spreadsheets

OpenDX
The \*nix make utility

The \*nix make utility

The arcs wrapper language

or make
A dedicated data flow

anguage
Example: Envisionary

nbedding it in a Jupyto otebook

Snomonic projection Dump, occultation and travity

e dataflow C++ nplate library nctional progran

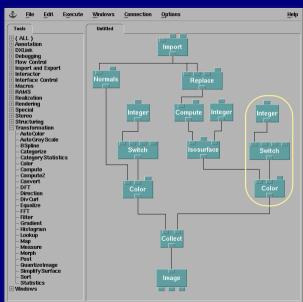
summary

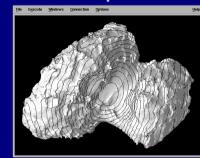
rovision

Take-hon









- ► This colors the shape (very quick).
- ► This colors the isolines (very quick).

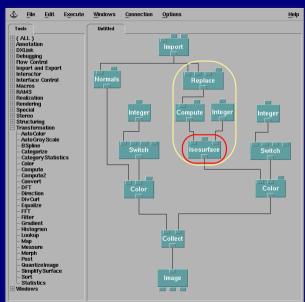
Life. Venus and Everything

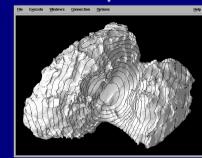
Björn Grieger

Spreadsheets OpenDX









- ► This colors the shape (very quick).
- This colors the isolines (very quick).
- ► This creates the isolines (a bit time consuming).

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Envision

programming

Spreadsheets

The \*nix make utility

The \*nix make utility

The arcs wrapper language

or make A dedicated data flow

anguage ixample: Envisionary imbedding it in a Jupyt

nbedding it in a Jupy otebook

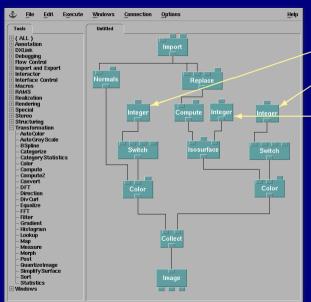
Dump, occultation gravity

emplate library unctional progran ummary

rovision









So called "Interactors" allow to change input values interactively. Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language for make

A dedicated data f anguage

ample: Envisionary

mbedding it in a Juj otebook nomonic projection

ump, occultation an

e dataflow C++
nplate library

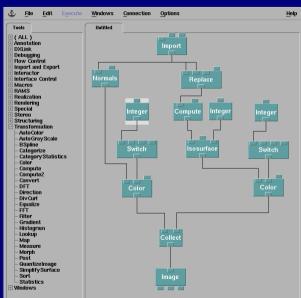
unctional programi immary

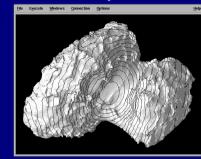
rovision

аке-пот









► If an input value, e.g., the surface color, is changed...

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming Overview

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language for make

dedicated data flow inguage

Example: Envisionary Embedding it in a Jupyt Votebook

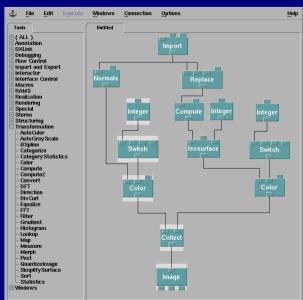
Gnomonic projection Dump, occultation and gravity

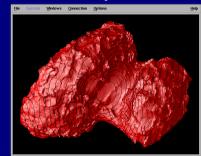
mplate library inctional programi immary

rovision

Take-hon







- ► If an input value, e.g., the surface color, is changed...
- ...only downstream modules are re-executed this is quick.

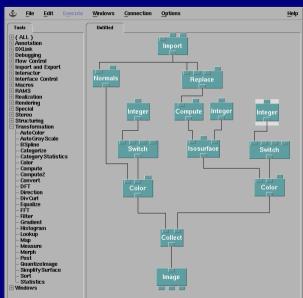
Life. Venus and Everything

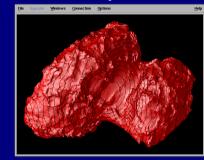
Björn Grieger

OpenDX









► Similarly, if the isoline color, is changed...

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

Overview

Spreadsheets
OpenDX

The \*nix make utility

The arcs wrapper language for make

dedicated data flow

xample: Envisionary mbedding it in a Jupyt

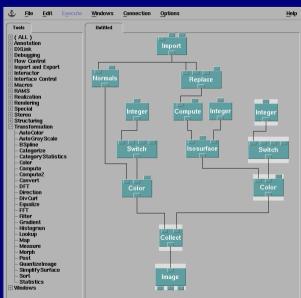
Gnomonic projection Dump, occultation and gravity

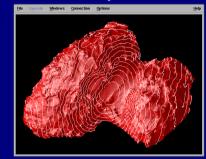
ne dataflow C++ mplate library inctional program

rovision









- Similarly, if the isoline color, is changed...
- ► ... it's quick.

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language for make

A dedicated data flow anguage

xample: Envisionary mbedding it in a Jupyt lotebook

Gnomonic projection Dump, occultation and gravity

ne dataflow C++ mplate library

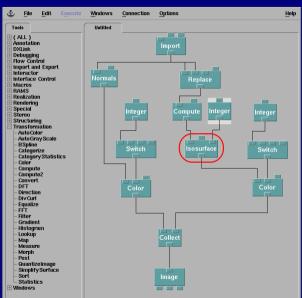
nctional programm nmary

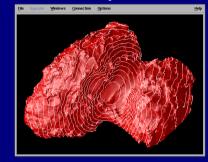
rovision

Take-hon









➤ Only if the isoline number is changed, the expensive Isosurface module has to be re-run...

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

The \*nix make utility

The \*nix make utility

The arcs wrapper language

for make A dedicated data flow

A dedicated data flow anguage

Example: Envisionary

mbedding it in a Jupy otebook

inomonic projection Dump, occultation and ravity

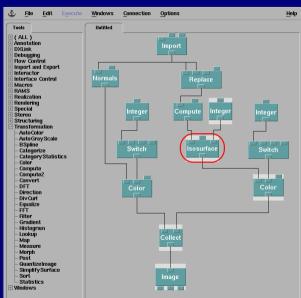
he dataflow C++ mplate library

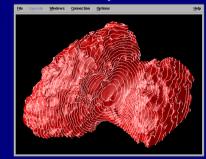
nctional programm mmary

rovision









- ➤ Only if the isoline number is changed, the expensive Isosurface module has to be re-run...
- ... and takes a bit of computation time.

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Envision

programming

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language for make

A dedicated data flow anguage

cample: Envisionary

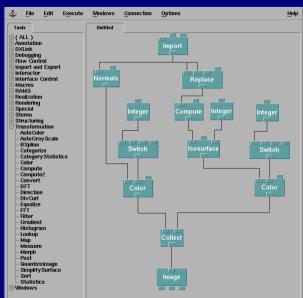
Gnomonic projection Dump, occultation a

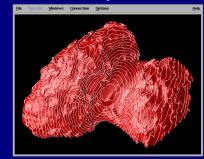
he dataflow C++ emplate library unctional programmin

rovision









- ► The data flow concept saves computation time...
- ... and even more important — development time!

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow

A dedicated data flow anguage

ample: Envisionary nbedding it in a Jupyto stebook

Gnomonic projection Dump, occultation and gravity

ravity ne dataflow C++ mplate library

npiate library nctional programn nmary

P<sub>ro</sub>vision





## The \*nix make utility

make - GNU make utility to maintain groups of programs

- ► Traditionally used to build executables.
- ▶ But can be used for all kinds of computations.
- ▶ Basic building blocks are **rules** (though there is much more).

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flow

A dedicated data flov language

example: Envisionary Embedding it in a Jupyt

nomonic projection ump, occultation an

Dump, occultation ar gravity 'he dataflow C++

Functional programsummary

Provision

Fake-home





Content

Name

**Examples** 

What to make?

What's needed?

How to make it?

Life. Venus and

Everything Björn Grieger

Spreadsheets

The \*nix make utility





Name What to make?

What's needed?

How to make it?

**Targets** 

List of files to make

Content

**Examples** Executables, object code files

Life. Venus and

Everything Björn Grieger

Spreadsheets

The \*nix make utility



What to make?	<b>Name</b> Targets	<b>Content</b> List of files to make	Examples Executables, object
What's needed?	Dependencies	List of files needed as input	code files Object code files, source code files
How to make it?		us input	source code mes

Life, Venus and Everything

Björn Grieger

/enera

/lagellan

EnVision

unctional rogramming

Overview

OpenDX

The \*nix make utility

'he arcs wrapper langua or make

dedicated data flow guage

nguage vample: Envision

ampie: Envisionary nbedding it in a Jup ntebook

omonic projection mp, occultation and wity

ravity

unctional progra

rovision





	Name	Content	Examples
What to make?	Targets	List of files to make	Executables, object code files
What's needed?	Dependencies	List of files needed as input	Object code files, source code files
How to make it?	Recipe	Sequence of commands to execute	Compile and link commands

Note to self: skip to Orchestrating computations

Life, Venus and Everything

Björn Grieger

/enera

/lagellan

EnVision

unctional

Overview

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper languag for make

A dedicated data anguage

inguage xample: Envisi

nbedding it in a Ju otebook

nomonic projection ump, occultation ar avity

dataflow C+-

ınctional prograi ımmary

rovision





```
Link object files to create executable
 Target.
     main : \
Dependencies
     main.o math sub.o plot sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

FnVision

Functional

programmir

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper lang

for make

A dedicated data flow

language

Example: Envisionary
Embedding it in a Ju

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation and gravity

The dataflow C template library Functional prog

summary

Talan Isaaa





```
Link object files to create executable
     main:
Dependencies
     main.o math sub.o plot sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```

maiı

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programmir

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow language

language
Example: Envisionary

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation an gravity

gravity
The dataflow C++

Functional prog

Provision





```
Link object files to create executable
 Target.
     main : \
Dependencies
     main.o math sub.o plot sub.o;
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```

```
main.o
```

```
math_sub.o
```

```
plot_sub.o
```

Life. Venus and Everything

Björn Grieger

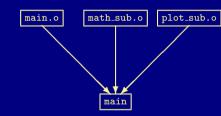
Spreadsheets OpenDX

The \*nix make utility





```
Link object files to create executable
 Target.
     main : \
Dependencies
     main.o math sub.o plot sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot sub.f ; \
Recipe
     gfortran -c plot sub.f
```



Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programmin

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data flor

language

Embedding it in a Jupy Notebook

Gnomonic projection Dump, occultation ar gravity

The dataflow C-template library

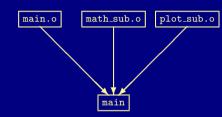
Functional programmary

P<sub>ro</sub>vision





```
Link object files to create executable
Target.
     main : \
Dependencies
     main.o math sub.o plot sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.ol:
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```



Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programming

Overview

Spreadsheets

The \*nix make utility

The arcs wrapper language

A dedicated data language

Example: Envisionary

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation a gravity

The dataflow C template library Functional prog

immary

Provision







```
Link object files to create executable
Target.
     main : \
Dependencies
     main.o math sub.o plot sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f;
Recipe
     gfortran -c plot sub.f
```

```
plot_sub.f
main.f
           math_sub.f
           math_sub.o
                         plot_sub.o
main.o
              main
```

Life. Venus and Everything

Björn Grieger

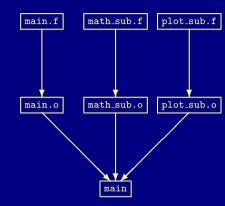
OpenDX

The \*nix make utility





```
Link object files to create executable
Target.
     main : \
Dependencies
     main.o math_sub.o plot_sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target
     main.o: \
Dependency
     main.f ; \
Recipe
     gfortran -c main.:
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```



Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programmir

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper langua

A dedicated data flo

language
Example: Envisionar

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation a gravity

The dataflow Ctemplate library

Functional programs summary

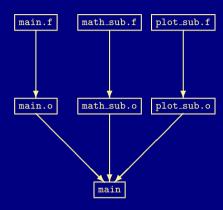
P<sub>ro</sub>vision







```
Link object files to create executable
Target.
     main : \
Dependencies
     main.o math_sub.o plot_sub.o ; \
Recipe
     gfortran -o main main.o math_sub.o plot_sub.o
Compile main program
Target.
     main.o: \
Dependency
     main.f ; \
Recipe
     ofortran -c main.f
Compile math subroutines
Target
     math sub.o : \
Dependency
     math sub.f : \
Recipe
     gfortran -c math sub.f
Compile plot subroutines
Target
     plot sub.o : \
Dependency
     plot_sub.f ; \
Recipe
     gfortran -c plot sub.f
```



The make syntax defines a data flow. It represents functional programming without functions.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper langua

A dedicated data flo

language

Embedding it in a Jupy Notebook

Gnomonic projection

Dump, occultation an

gravity

The dataflow C+ template library

Functional program summary

P<sub>ro</sub>vision





	Name	Content	Examples
What to make?	Targets	List of files to make	Executables, object code files
What's needed?	Dependencies	List of files needed as input	Object code files, source code files
How to make it?	Recipe	Sequence of commands to execute	Compile and link commands

Life, Venus and Everything

Björn Grieger

Venera

/lagellan

EnVision

unctional

programming Overview

Spreadsheets

The \*nix make utility

The arcs wrapper language for make

dedicated data flo

ample: Envisionary

mbedding it in a Jup otebook

omonic projection mp, occultation and vity

mplate library Inctional progra

2 vicion

Taka hawa





What to make?	Name Targets	<b>Content</b> List of files to make	Examples Output data files
What's needed?	Dependencies	List of files needed as input	Object code files, source code files
How to make it?	Recipe	Sequence of commands to execute	Compile and link commands

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional rogramming

Overview

OpenDX

The \*nix make utility

The arcs wrapper languag for make

A dedicated data anguage

anguage Example: Envisio

nbedding it in a Jupy otebook

nomonic projection ump, occultation ar avity

ravity ne dataflow C+

inctional prograi

rovision





What to make?	<b>Name</b> Targets	<b>Content</b> List of files to make	Examples Output data files
What's needed?	Dependencies	List of files needed as input	Input data files, program file
How to make it?	Recipe	Sequence of commands to execute	Compile and link commands

For conciseness, we assume here that the programs are interpreted and no compilation is needed.

Life, Venus and Everything

Björn Grieger

/enera

√lagellan

EnVision

nctional ogramming

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data

anguage

ampie: Envisionary nbedding it in a Jupy stebook

nomonic projection ump, occultation an ravity

ne dataflow C+ mplate library

unctional progran ummary

rovision





What to make?	Name Targets	<b>Content</b> List of files to make	Examples Output data files
What's needed?	Dependencies	List of files needed as input	Input data files, program file
How to make it?	Recipe	Sequence of commands to execute	Program call

Life, Venus and Everything

Björn Grieger

Venera

∕lagellan

EnVision

nctional ogramming

Overview Spreadsheets

The \*nix make utility

The arcs wrapper langu

A dedicated data

anguage

ample: Envisionary nbedding it in a Ju ptebook

nomonic projection ump, occultation ar

avity

e dataflow C++

unctional programmary

rovision





What to make?	Name Targets	<b>Content</b> List of files to make	Examples Output data files
What's needed?	Dependencies	List of files needed as input	Input data files, program file
How to make it?	Recipe	Sequence of commands to execute	Program call

... to orchestrating computations.

Life, Venus and Everything

Björn Grieger

'enera

/lagellan

EnVision

inctional ogramming

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

r make A dedicated data flow

anguage

example: Envisionary Embedding it in a Jup Hotebook

nomonic projection ump, occultation a

avity e dataflow C+4

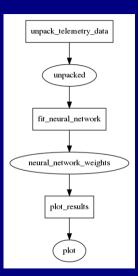
unctional program Immary

rovision





### Makefile for computation orchestration



```
Unpack telemetry data
Target.
     unpacked.dat : \
Dependencies
     unpack telemetry data.pv : \
Recipe
     ./unpack telemetry data.pv
Fit neural network (very time consuming)
Target.
     neural network weights.dat : \
Dependency
     unpacked.dat fit neural network.pv ; \
Recipe
     ./fit neural network.pv
Plot results
Target
     plot.png : \
Dependency
     neural_network_weights.dat plot.py ; \
Recipe
     ./plot results.pv
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

orogramming

Spreadshee

The \*nix make utility

A dedicated data flow

language Example: Envisionary

Embedding it in a Jupyte Notebook

Snomonic projection Dump, occultation and ravity

he dataflow C++ emplate library unctional programi

Provision





### Makefile for computation orchestration

```
Unpack telemetry data
unpack telemetry data
                             Target.
                                   unpacked.dat : \
                             Dependencies
                                   unpack telemetry data.pv : \
     unpacked
                             Recipe
                                   ./unpack_telemetry_data.py
                             Fit neural network (very time consuming)
  fit neural network
                             Target.
                                   neural network weights.dat : \
                             Dependency
                                   unpacked.dat fit neural network.pv ; \
neural network weights
                             Recipe
                                   ./fit neural network.pv
                             Plot results
    plot results
                             Target
                                   plot.png : \
                             Dependency
                                   neural_network_weights.dat plot.py ; \
       plot
                             Recipe
                                   ./plot results.pv
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programmin

Spreadshee OpenDX

The \*nix make utility

A dedicated data flow

language
Example: Envisionary

Embedding it in a Jupyt Notebook

Gnomonic projection

Dump, occultation ar

gravity

The dataflow C++

The dataflow C++ emplate library Functional progran

ovision





```
Unpack telemetry data
unpack_telemetry_data
                             Target.
                                   unpacked.dat : \
                             Dependencies
                                   unpack telemetry data.pv : \
     unpacked
                             Recipe
                                   ./unpack telemetry data.pv
                             Fit neural network (very time consuming)
  fit neural network
                             Target.
                                   neural network weights.dat : \
                             Dependency
                                   unpacked.dat fit neural network.pv ;
neural network weights
                             Recipe
                                   ./fit_neural_network.py
                             Plot results
    plot results
                             Target
                                   plot.png : \
                             Dependency
                                   neural_network_weights.dat plot.py ; \
       plot
                             Recipe
                                   ./plot results.pv
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programmin

Spreadshee OpenDX

The \*nix make utility

A dedicated data flow

language Example: Envisionary

Embedding it in a Jupy Notebook

Snomonic projection

Dump, occultation and prayity

he dataflow C++
emplate library
unctional programr

Provision





```
Unpack telemetry data
unpack telemetry data
                             Target.
                                   unpacked.dat : \
                             Dependencies
                                   unpack telemetry data.pv : \
     unpacked
                             Recipe
                                   ./unpack telemetry data.pv
                             Fit neural network (very time consuming)
  fit neural network
                             Target.
                                   neural network weights.dat : \
                             Dependency
                                   unpacked.dat fit neural network.pv ; \
neural network weights
                             Recipe
                                   ./fit neural network.pv
                             Plot results
    plot results
                             Target.
                                   plot.png : \
                             Dependency
                                   neural network weights.dat plot.pv ;
       plot
                             Recipe
                                   ./plot results.pv
```

Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

Functional

programmin Overview

Spreadshee OpenDX

The \*nix make utility

A dedicated data flow

language
Example: Envisionary

example: Envisionary Embedding it in a Jupyte Notebook

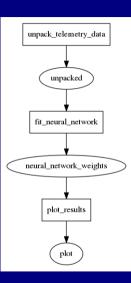
Snomonic projection Dump, occultation and ravity

e dataflow C++
nplate library
nctional program

rovision







The Makefile establishes a data flow between computations.

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional

orogramming

Spreadsheets

OpenDX

The \*nix make utility

or make

A dedicated data flow language

Example: Envisionary

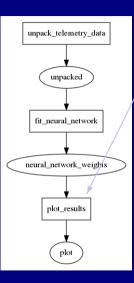
mbedding it in a Jupyt lotebook

ump, occultation ar avity

nplate library nctional programn nmary

Provision





The Makefile establishes a data flow between computations.

, If we change a color in the plot. . .

Life, Venus and Everything

Björn Grieger

Venera

N 4 - --- II -

EnVision

\_\_\_\_\_\_

programmin

Overview

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper langua or make

A dedicated data flow anguage

language Example: Envisionary

mbedding it in a Jupyt lotebook

nomonic projection ump, occultation ar avity

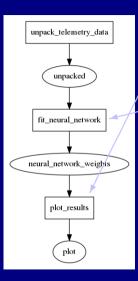
e datanow C++ nplate library nctional programn

mmary

rovision







The Makefile establishes a data flow between computations.

If we change a color in the plot...

... we don't have to rerun the (expensive) fitting (and make won't).

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

unctional

programming

Spreadshee

OpenDX

The \*nix make utility

or make

A dedicated data flow

A dedicated data flow anguage

Example: Envisionary Embedding it in a Jupy

otebook nomonic projection

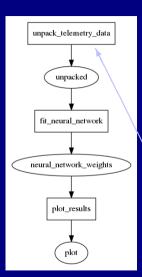
mp, occultation and

mplate library nctional program

.....







The Makefile establishes a data flow between computations.

If we change a color in the plot...

... we don't have to rerun the (expensive) fitting (and make won't).

<sup>∖</sup> If we correct a bug in the unpacking. . .

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

unctional

rogramminį

Spreadsheet

The \*nix make utility

The arcs wrapper language

A dedicated data flow

language
Example: Envisionary

Embedding it in a Jupy Notebook

> np, occultation an vity

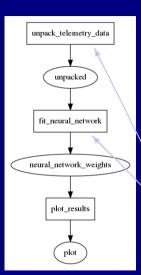
nplate library

unctional program Immary

rovision







The Makefile establishes a data flow between computations.

If we change a color in the plot...

... we don't have to rerun the (expensive) fitting (and make won't).

 $^{ackprime}$  If we correct a bug in the unpacking. . .

... we do have to rerun the fitting (and the plotting), and make will.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

Overview

OpenDX

The \*nix make utility

The arcs wrapper language for make

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupy

lotebook inomonic projection

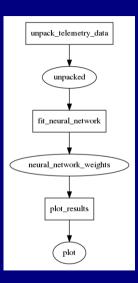
Dump, occultation ravity

plate library ctional programmi

mmary







The Makefile establishes a data flow between computations.

To have, e.g., neural\_network\_weights passed over from fit\_neural\_network to plot\_results requires three steps:

- 1. Insert a rule in the Makefile.
- Insert code to write the file neural\_network\_weights.dat in fit\_neural\_network.py.
- 3. Insert code to read the file in plot\_results.py.

This is tedious and error prone.

Can we do better?

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming

Spreadsheets

The \*nix make utility

The arcs wrapper langua

A dedicated data f anguage

Example: Envisionary Embedding it in a Jupy Notebook

nomonic projection ump, occultation a avity

e dataflow C++ nplate library

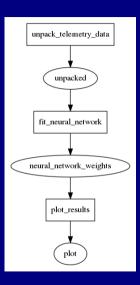
unctional progran ummary

rovision









#### Idea:

- ▶ Describe the data flow in a file (with an appropriate language).
- ► Have a compiler reading that file and
  - 1. writing a Makefile,
  - 2. inserting code to write respective files into the source code of all programs which output data,
  - **3.** inserting code to read respective files into the source code of all programs which input data.

Contrary to make, there is now a single point of maintenance for the "pathways of data between programs" which we simply call arcs — and likewise the language and the compiler.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional rogramming

Spreadsheets

OpenDX The wais make u

The \*nix make utility

or make

A dedicated data flow

#### A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyte
Notebook

nomonic projection ump, occultation ar

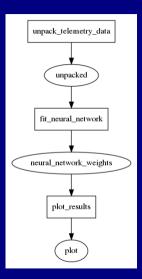
he dataflow Cemplate library

ınctional prograi

rovision









- ► In 2008, I coded continuously over one weekend, almost without eat, drink, sleep. After that, I had a basically working compiler.
- ➤ Since then, it was constantly developed "on the job". Now it has about 1000 lines of code (2000 with comments and white space).
- ► The invested time was minimal, so it was written quick and dirty, which has resulted in horrendous spaghetti code.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functiona

orogramming

Spreadsheets

The \*nix make utility

for make

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyto

lotebook inomonic projection

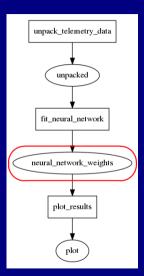
ump, occultation avity

plate library ctional programi

rovision









\neural\_network\_weights
 real wts(nl,maxna,0:maxnam)
 < fit neural network</pre>

> plot\_results

This is the description of an arc, a bundle of data that is passed from one module to another.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

programming

Spreadsheets

OpenDX
The \*nix make utility

The arcs wrapper language

A dedicated data flow language

Example: Envisionary Embedding it in a Jupy Notebook

nomonic projection

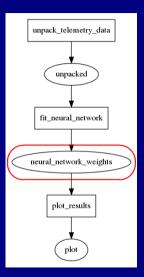
avity ≥ dataflow C++ polate library

unctional programmary

rovision









neural\_network\_weights
real wts(nl,maxna,0:maxnam)

- < fit\_neural\_network</pre>
- > plot\_results
- ► Name of the arc

Life, Venus and Everything

Björn Grieger

Venera

Manalla

FnVision

Functional

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow language

xample: Envisionary mbedding it in a Jupyt lotebook

omonic projection
np. occultation an

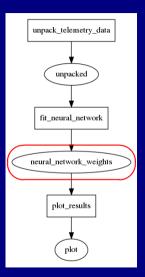
e dataflow C+ nplate library

ictional program

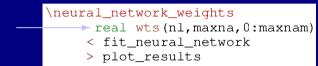
rovision











- Name of the arc
- Declaration of data content

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programming

Overview

Spreadsheets

OpenDX

The arcs wrapper languag

A dedicated data flow language

Example: Envisionary Embedding it in a Jupy

imbedding it in a Jup Jotebook Gnomonic projection

np, occultation an

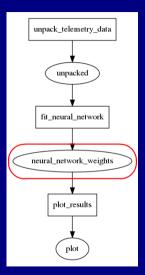
ne dataflow C+

ınctional prograi ımmary

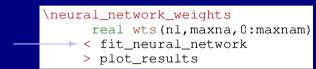
rovision











- ► Name of the arc
- Declaration of data content
- ► Module that outputs the arc

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

The waiv ma

The arcs wrapper languag

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyt
Notebook

omonic projection np, occultation an vity

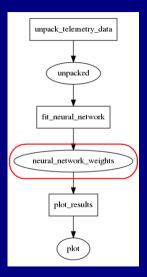
e dataflow C+ nplate library

nctional progran nmary

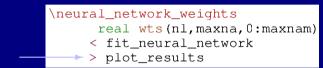
rovision











- ▶ Name of the arc
- Declaration of data content
- Module that outputs the arc
- ► Module that inputs the arc

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

The \*nix mak

The arcs wrapper languag

A dedicated data flow language

Example: Envisionary Embedding it in a Jupyt Notebook

nomonic projection ump, occultation ar avity

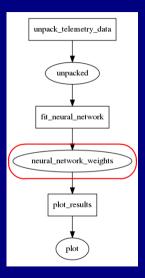
dataflow C+plate library

nctional progran nmary

rovision









### \neural\_network\_weights

real wts(n1, maxna, 0:maxnam)

- < fit\_neural\_network
- > plot\_results
- ► Name of the arc
- Declaration of data content
- ► Module that outputs the arc
- Module that inputs the arc

This is the *only* construct of the arcs language.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

orogramming Overview

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper langua

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyt
Notebook

nomonic projection ump, occultation an avity

e dataflow C++

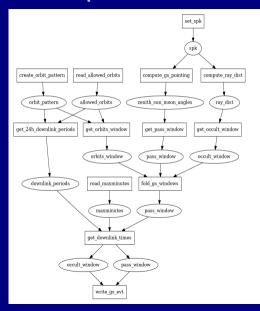
nctional program mmary

rovision





## Science operations reference scenario for EnVision





- Envisionary employs functional programming. The main control script is actually a data flow description using the arcs language.
- As such, it makes use of *lazy* evaluation, i. e., only program modules which need to are (re-)executed.
- Modules exchange data via the hard disk, so intermediate results are preserved between program runs.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional
programming
Overview

The \*nix make utility

The arcs wrapper langua for make

dedicated data flow anguage

Example: Envisionary
Embedding it in a Jupy

nomonic projection lump, occultation an ravity

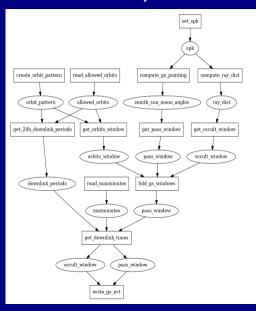
he dataflow C++ emplate library unctional programming emmary

P<sub>ro</sub>vision





### **Data flow description**



- ► Such diagrams are automatically created from source code.
- This one displays only a subset (13 out of ≈ 220) of modules to illustrate the principal data flow for the computation of ground station events
  - Boxes are program modules, (ovals) are data structures.

Life, Venus and Everything

Björn Grieger

Venera

N 4 - - - II

EnVision

Functional

Overview

Spreadsheets OpenDX

The \*nix make

The arcs wrapper language

A dedicated data flow anguage

language

Example: Envisionary

Embedding it in a Jupyte Notebook

> omonic projection mp, occultation and wity

ravity ne dataflow C+

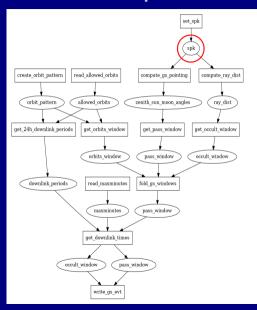
inctional progra mmary

rovision





### Data flow description



- ► Such diagrams are automatically created from source code.
- ► This one displays only a subset (13) out of  $\approx$  220) of modules to illustrate the principal data flow for the computation of ground station events
- Boxes are program modules, (ovals) are data structures.
- spk is the SPICE Spacecraft Position Kernel, which contains the spacecraft trajectory.

Life. Venus and Everything

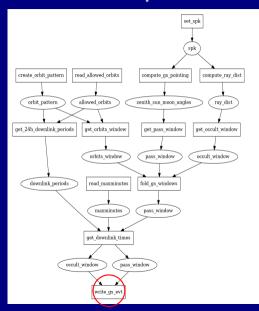
Biörn Grieger

OpenDX





### **Data flow description**



- ► Such diagrams are automatically created from source code.
- This one displays only a subset (13 out of  $\approx$  220) of modules to illustrate the principal data flow for the computation of ground station events
- Boxes are program modules, (ovals) are data structures.
- spk is the SPICE Spacecraft Position Kernel, which contains the spacecraft trajectory.
- write\_gs\_events is the module which writes the event file for MAPPS.

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programming

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data flow

language

Frample: Envisionary

Embedding it in a J Notebook

Snomonic projection Dump, occultation gravity

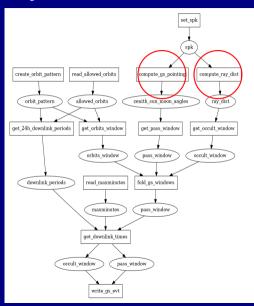
he dataflow C++
emplate library
functional programn

rovision





## Lazy evaluation



- ► The two compute\_\* modules are quite time consuming, O(h), everything else takes the wink of an eye.
- We have to re-run these modules (only) if the SPK changes (the spacecraft trajectory).

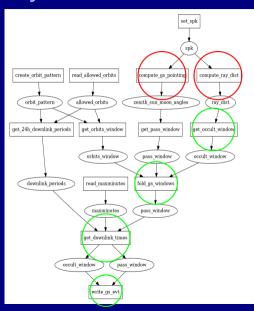
Life. Venus and Everything

Björn Grieger

OpenDX



### Lazy evaluation



- ► The two compute\_\* modules are quite time consuming, O(h), everything else takes the wink of an eye.
- We have to re-run these modules (only) if the SPK changes (the spacecraft trajectory).
- ► If we change the occulting thickness of the atmosphere, only downstream modules are re-executed.

Life, Venus and Everything

Björn Grieger

Vener

. . ..

**EnVision** 

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

A dedicated data flow

language

Example: Envisionary

Example: Envisionary Embedding it in a Jupyte Notebook

nomonic projection ump, occultation an avity

ne dataflow C+ mplate library

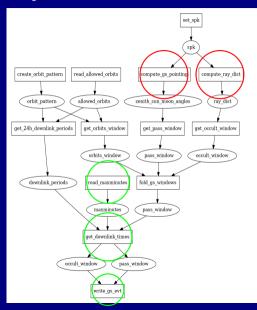
ınctional progra

rovision





## Lazy evaluation



► The two compute\_\* modules are quite time consuming, O(h), everything else takes the wink of an eye.

- We have to re-run these modules (only) if the SPK changes (the spacecraft trajectory).
- ► If we change the occulting thickness of the atmosphere, only downstream modules are re-executed.
- ► If the table with daily target downlink times is changed, only downstream modules from *there* are re-executed.

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programming

Overview Spreadsheets

OpenDX

The arcs wrapper languag

A dedicated data flow

language

Example: Envisionary

exampie: Envisionary Embedding it in a Jupyte Notebook

Snomonic projection Dump, occultation a gravity

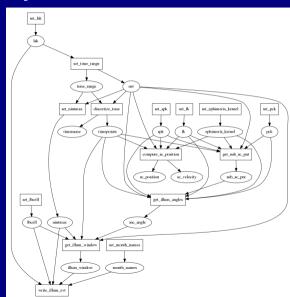
he dataflow C++ emplate library unctional programmir

Provision





## Full Envisionary as of 2019-10-18



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming

Spreadsheets

The \*nix make

The arcs wrapper languag

dedicated data flow

language

Example: Envisionary

Embedding it in a Jupyte Notebook

Gnomonic projection

Dump, occultation and

cravity

The dataflow C+template library

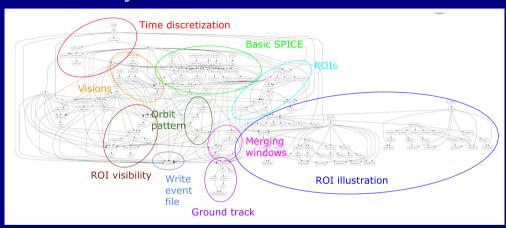
Functional progr summary

P<sub>ro</sub>vision





### Full Envisionary as of 2020-02-05



By now, Envisionary has grown much too big to be completely displayed in one graph. Selected subsets of modules can be displayed.



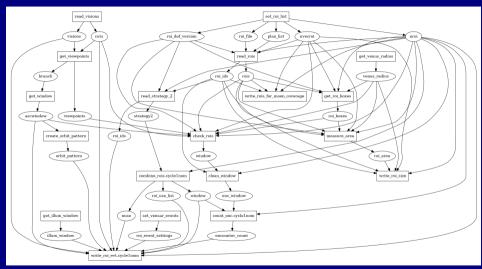
Life. Venus and Everything

Björn Grieger

OpenDX



# Selected modules for ROI coverage estimation



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

programming

Spreadsheets

he \*nix make utility

nake dedicated data flow

guage

Example: Envisionary
Embedding it in a Jupyte

otebook nomonic projection

ump, occultation ar avity e dataflow C++

nplate library nctional programn

Provision



## Real "agility"

- No high level plan is needed to get started. Just do first things first.
- The data flow approach is to some extent self documenting.
- The system is runnable at all times.
- Because of lazy evaluation, testing is conveniently done "on the job".

Life. Venus and Everything

Björn Grieger

Spreadsheets

OpenDX





# Quo vadis, Envisionary?





- ► I shall be retired long before EnVision flies.
- ▶ The arcs language is fairly simple and to some extend self-documenting, so anybody could take over the maintenance of Envisionary, but...
- the arcs compiler itself is horrendous spaghetti code, so any needed bug correction or functionality extension would cause the uninitiated bad headache.
- ▶ It would be great (but most probably impossible) if the arcs compiler could be properly rewritten to support arbitrary programming languages via configuration files (while it is currently hard wired to support only Fortran. Perl, and OpenDX).

Life. Venus and Everything

Biörn Grieger

OpenDX







# **Quo vadis, Envisionary?**





- Envisionary employing arcs has proven to be very efficient for the rapid development of a Science Operations Reference scenario.
- ▶ At some point, we have to switch (or gradually move) to something else.



Life. Venus and Everything

Björn Grieger

Spreadsheets OpenDX





# **Embedding it in a Jupyter Notebook (1)**

### Cross validation of MAPPS coverage

Besides documenting the work on this issue, this also serves as example of the integration of Envisionary and Moon Coverage in a Jupyter notebook

#### Jira issue

**ENVISOCMNGT-3** 

#### Set up Moon Coverage

#### Import packages

```
In [4]: import matplotlib.pvplot as plt
        from moon coverage import TourConfig, MetaKernel, VENUS
        from moon coverage.esa import EsaCremasCollection
        from moon coverage.ticks import date ticks, km ticks
        from moon coverage import ROI
        from IPvthon.display import Image
        import numpy as no
        import pandas as pd
```

Life. Venus and Everything

Björn Grieger

Spreadsheets OpenDX

Embedding it in a Jupyter Notebook





# **Embedding it in a Jupyter Notebook (2)**

#### **Envisionary**

Data flow for selected modules

In [5]: Image('../dat/selection xval cov.png') Out[5]: read\_orbit\_pattern\_2 orbit pattern set time range xval cov net time\_range discretize time.xval cov ncvcle timepoints

Life. Venus and Everything

Björn Grieger

Spreadsheets

Embedding it in a Jupyter Notebook





# **Embedding it in a Jupyter Notebook (3)**

#### Configuration file

```
In [11]: %writefile xval cov.acfg
      arcs -d -m compute coverage, write grid arcs.f
      set time range xval cov.f
         det = 300
      set npntqc.f
         npntqc = 36
      Overwriting xval cov.acfg
```

Life. Venus and Everything

Björn Grieger

Spreadsheets

Embedding it in a Jupyter Notebook





## **Embedding it in a Jupyter Notebook (4)**

#### Run Envisionary

```
In [12]: !./envy xval cov.acfg
         Configuration file: ../exe/xval cov.acfg
         arcs arguments: -d -m compute coverage, write grid arcs.f
         Configuring ../src/set time range xval cov.f ...
            Found det = 300
            OΚ
         Configuring ../src/set nontac.f ...
            Found npntgc = 72
            Setting npntgc = 36
         ... configuration done.
         Starting computation.
         make[1]: Entering directory '/lhome/bgrieger/VENUS/ENVISION/SPICE/ENVISIONARY/tmp'
         gfortran -02 -o set npntgc.x set npntgc.f
         ./set npntac.x
         echo done > set npntgc
         ./compute coverage.x
         echo done > compute coverage
         ./write grid.x
         echo done > write grid
         make[1]: Leaving directory '/lhome/bgrieger/VENUS/ENVISION/SPICE/ENVISIONARY/tmp'
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programmin

Overview

Spreadsheets

)penDX

he ∗nix make ut

The arcs wrapper language for make

A dedicated data flow language

xample: Envisionary

Embedding it in a Jupyter Notebook

Gnomonic projection

Dump, occultation and

The dataflow Ctemplate library

unctional program

Provision





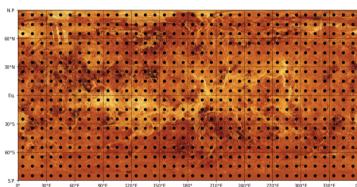
## **Embedding it in a Jupyter Notebook (5)**

#### Visualize coverage grid

We read the grid coordinates written by Envisionary and plot them on top of a Venus map:

```
In [13]: grid = pd.read_csv( '../tmp/cov_grid.txt' )
fig = plt.figure(figsize=[14, 9))
ax = fig.add supplot(projection=VENUS)
ax.plot( grid.Longitude, grid.Latitude, 'o', color='black')
```

Out[13]: [<matplotlib.lines.Line2D at 0x7f70a97bfb00>]



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

programmi

Spreadsheet

The \*nix make utility

A dedicated data flow

nguage

Embedding it in a Jupyter

Gnomonic projection

Dump, occultation and

gravity
The dataflow C+
template library

Functional programsummary

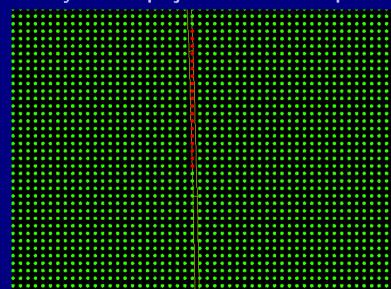
Provision







# **Equidistant cylindrical projection near the equator**



Life, Venus and Everything

Björn Grieger

Venera

Angolla:

EnVision

unctional

Overview

Spreadsheets

OpenDX

The \*nix make utility The arcs wrapper languag

he arcs wrapper languag ir make A dodicated data flow

nguage cample: Envisionary

mbedding it in a Jupy otebook

### Gnomonic projection

Oump, occultation and ravity

unctional program ummary

Provision





# Straight lines are not straight lines!



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

programming

Spreadsheets

The \*nix mak

The arcs wrapper language for make

A dedicated data flow anguage

anguage Example: Envisionary

Embedding it in a Jupy Notebook Gnomonic projection

Onomonic projection

Dump, occultation and

ravity

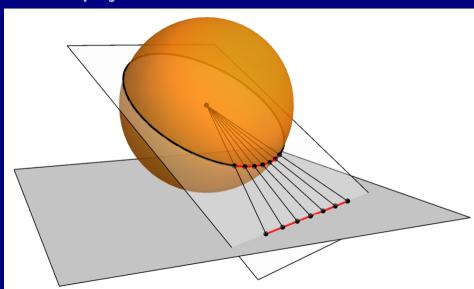
emplate library unctional program immary

rovision





# **Gnomonic projection**



Life, Venus and Everything

Björn Grieger

Vener

Magellan

EnVision

unctional

programming

Overview

OpenDX

The \*nix make utility

The arcs wrapper languag for make

dedicated data flow nguage

iguage ample: Envisionary

bedding it in a Jupy ebook

Gnomonic projection

Dump, occultation and

gravity
The dataflow C++

The dataflow C template library Functional prog

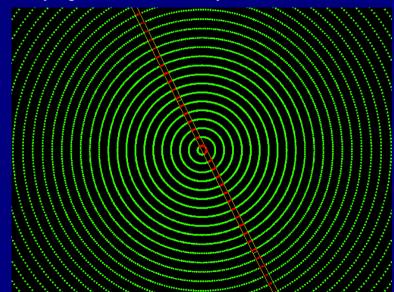
Functional programmary

rovision





# **Gnonomic projection near the poles**



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

unctional rogramming <sub>verview</sub>

OpenDX The ∗nix make utility

Fhe ∗nix make utility Fhe arcs wrapper language 'or make

A dedicated data flow anguage Example: Envisionary Embedding it in a Jupyto

Notebook

Gnomonic projection

Dump, occultation and gravity

template library Functional program summary

Provision

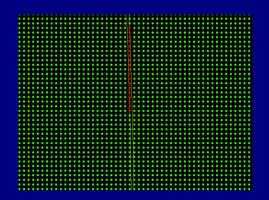
Take-hor

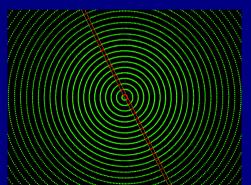




### Three cases

- ▶ If a swath crosses the equator, use equidistant cylindrical.
- ► If a swath lies completely in one hemisphere, use Gomonic for that hemisphere.





Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional
programming
Overview
Spreadsheets

OpenDX
The \*nix make utility
The area was proper learning

dedicated data flow

nbedding it in a otebook

Gnomonic projection

Dump, occultation and

The dataflow C+ template library Functional progra

P<sub>ro</sub>vision





# Radio gaga

- 1. The radio link can either be run in 1-way or 2-way mode.
- 2. The switching from 2-way to 1-way takes ten minutes. During switching, no radio link is available. The reverse switching from 1-way to 2-way can be considered instantaneous.
- 3. Dump can be done in either mode, 1-way or 2-way.
- 4. Gravity can only be done in 2-way mode.
- 5. Occultation can only be done in 1-way mode.
- **6.** Dump and gravity can only be done above 100 km height (of ground station to spacecraft line of sight).
- **7.** Occultation is done between 0 and 200 km height, with 100–200 km used for calibration.
- **8.** Occultation can only be done at egress if it was done at the preceeding ingress.
- **9.** Target is two occultation ingress/egress pairs per day.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

unctional

rogrammi Overview

Spreadsheet

OpenDX

The \*nix make utility
The arcs wrapper language

dedicated data floo

nguage kample: Envisiona

mbedding it in a Jupy otebook

Dump, occultation and gravity

The dataflow C emplate library

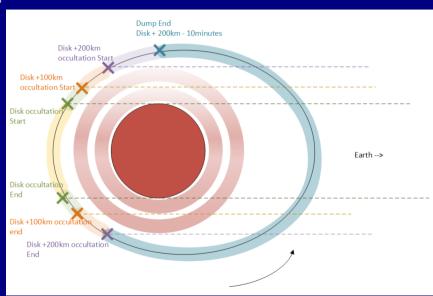
unctional programn

rovision





# Required events



Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programmin

programm Overview

Spreadsheets

The \*nix make utilit

The arcs wrapper language for make

A dedicated data flow anguage

language
Example: Envisionary

mbedding it in a Jup lotebook

Dump, occultation and gravity

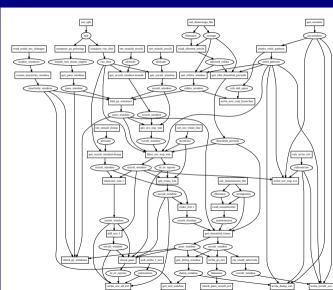
template library

P<sub>ro</sub>vision

Tako homo



# Mayhem orchestration



Life, Venus and Everything

Björn Grieger

Venera

Magellar

EnVision

Functional

programming

Overview Spreadsheets

OpenDX
The \*nix make ut

The \*nix make utility

The arcs wrapper languag

A dedicated data flo

language Example: Envisio

mbedding it in a Jup lotebook

Gnomonic projection

Dump, occultation and

gravity

The dataflow C++

template library
Functional programm summary

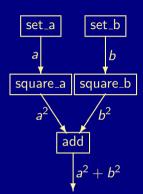
P<sub>ro</sub>vision

Take\_home





# The dataflow C++ template library



- ► With the arcs language discussed in the last section, modules are individual programs which exchange data over the hard disk.
- ► For usual programming with all happening in the RAM, e.g., the language Haskell offers functional programming, using functions to describe a data flow
- ► In order to describe directly the dataflow, I wrote a C++ template library. It is object oriented, each module is an object.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

EIIVISIOII

programming

Spreadsheets

OpenDX

The arcs wrapper language

A dedicated data flow

anguage Example: Envisionary

imbedding it in a Ju Jotebook

omonic projection imp, occultation and svity

The dataflow C++ template library
Functional programming

Functional programmin summary

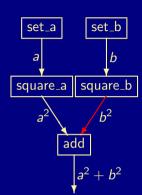
Provision







## **Data flow syntax**



To connect output 0 of module square\_b with input 1 of module add, one simply writes in the code:

CONNECT( square\_b, 0, add, 1 );

(which is expanded by the C preprocessor to something a little more cryptic, but that you never see)

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

-----

programming

Overview

OpenDX

The \*nix make

The arcs wrapper languag

dedicated data flow

nguage

mbedding it in a Jupy otebook

omonic projection imp, occultation and avity

The dataflow C++ template library

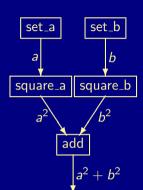
Functional programm summary

Provision









The module add is an object with a public function add.out0() that returns its output 0.

Life. Venus and Everything

Björn Grieger

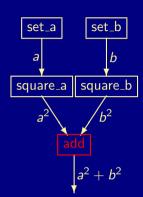
Spreadsheets

OpenDX

The dataflow C++ template library







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(),

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

make dedicated data flow

dedicated data flow nguage

cample: Envisionary

mbedding it in a Jupy

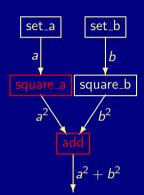
omonic projection mp, occultation and wity

The dataflow C++ template library Functional programmin

ovision







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(), and then:

update() calls the public function square\_a.version(). Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnV**isior

Functional

Overview

OpenDX

OpenDX

The \*nix make utility

The arcs wrapper language

A dedicated data flow

nguage

cample: Envisionary nbedding it in a Jupy otebook

omonic projection imp, occultation and wity

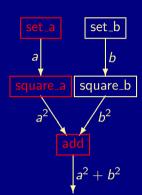
The dataflow C++ template library Functional programming

vicion

. . .







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(), and then:

- update() calls the public function square\_a.version().
  - square\_a.version() calls its own protected function update()

Life. Venus and Everything

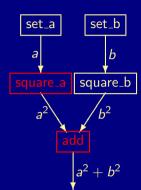
Björn Grieger

OpenDX

The dataflow C++ template library







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(), and then:

- update() calls the public function square\_a.version().
  - square\_a.version() calls its own protected function update()

    - square\_a.version() returns the version of its output.

Life. Venus and Everything

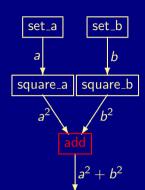
Biörn Grieger

OpenDX

The dataflow C++ template library







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(), and then:

- update() calls the public function square\_a.version().
  - square\_a.version() calls its own protected function update()
    - ▶ ...
  - square\_a.version() returns the version of its output.
- ► If the returned version is newer than the one from the latest computation, update() calls its own protected function compute() and increments the output version.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVisior

unctional

rogramming

Spreadsheets

The \*nix make utili

The arcs wrapper language

A dedicated data flow anguage

anguage Example: Envisionary

imbedding it in a Ju Jotebook

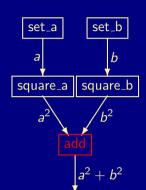
nomonic projection lump, occultation an ravity

The dataflow C++ template library

ovision







The module add is an object with a public function add.out0() that returns its output 0. If this function is called, it calls its own object's protected function update(), and then:

- update() calls the public function square\_a.version().
  - square\_a.version() calls its own protected function update()
    - **▶** ...
  - square\_a.version() returns the version of its output.
- ► If the returned version is newer than the one from the latest computation, update() calls its own protected function compute() and increments the output version.
  - compute() computes the output(s) from the input(s). This is the only user provided function.

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

rogramming

Overview Spreadsheets

Spreadsheets OpenDX

The \*nix make utility

or make A dedicated data flow

language

mbedding it in a J otebook

nomonic projecti lump, occultation ravity

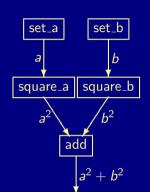
The dataflow C++ template library Functional programmin

vision





# The module object



### Each module object has:

- Variables (all protected)
  - ► Pointers to its input modules
  - Pointers to the outputs of its input modules
  - An array of the versions of its inputs
    - Its outputs
  - Its outputs' version
- Protected functions
  - update(), checks versions and calls compute() only if needed.
  - compute(), computes outputs, provided by the user
- Public functions
  - version(), calls update() and returns version
    - instantiation
    - establishing connections (by setting pointers)
    - requesting outputs
    - set() outputs (only inputless modules)

Life. Venus and Everything

Biörn Grieger

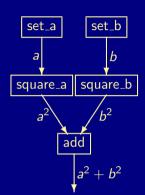
OpenDX

The dataflow C++ template library





### Data flow work flow



All this looks fairly complicated, but it's mostly transparent. To define a data flow, the user only has to

- define module classes by adding compute() functions to templates,
- ► instantiate module objects (like any object),
- connect module inputs and outputs (with the simple command shown before).

Then the user can

- ► set inputless modules,
- request module outputs.

On an output request, all necessary computations (and only those) will be performed.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

programming

Spreadsheets

OpenDX

The \*nix make utility

r make A dedicated data flow

nguage

Example: Envisionary Embedding it in a Jupy Notebook

nomonic projection ump, occultation an ravity

The dataflow C++ template library

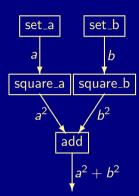
......

0.0000





### Demo



```
CONNECT( set_a, 0, square_a, 0 );
CONNECT( set_b, 0, square_b, 0 );
CONNECT( square_a, 0, add, 0 );
CONNECT( square_b, 0, add, 1 );
```

Life, Venus and Everything

Björn Grieger

Venera

Magella

**EnVision** 

Functional

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow anguage

xample: Envisionary

mbedding it in a Jupy otebook

ump, occultation an

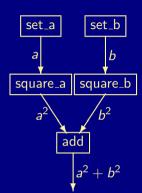
The dataflow C++ template library

ovision

10----







```
CONNECT( set_a, 0, square_a, 0 );
CONNECT( set_b, 0, square_b, 0 );
CONNECT( square_a, 0, add, 0 );
CONNECT( square_b, 0, add, 1 );
```

```
Note to self:
```

```
cd ~/VULCAN/PROGRAMMING/CPP/CPP_TUTORIAL/src
M df5
```

Life. Venus and Everything

Björn Grieger

The dataflow C++ template library





# The dataflow C++ template library

```
template< typename TinO, typename ToutO >
class Module 1 1: public Module (
 Module *m src0:
  const Tin0 *m in0:
  int v in[1]:
  Tout 0 m out 0;
  void update() {
    int v[1]:
   v[0] - m src0->version();
    for ( int i = 0; i < 1; ++i ) (
      if ( v[i] > v in[i] ) (
        old = true;
    if ( old ) {
      for ( int i = 0; i < 1; ++i ) {
        v in[i] = v[i];
      compute();
      ++m version:
  virtual void compute() = 0:
  Module 1 1() (
    for \ell int i = 0, i < 1, ++i) \ell
      v in[i] = 0:
   m version = 0:
  bool connect() ( Module& src, const Tin() *in ) (
   m_src0 = &src:
   m in0 - in;
  Tout( out() {
    undate():
    return m_out0;
  const Tout() *p out()() {
    return &m out 0:
```

- ► I implemented this as finger exercise while attending a C++ course.
- ► It has not yet been applied in the real world, thus I don't know if it is of any real use at all,
- ... but it was a nice exercise!

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

Overview

Spreadsheets

The ∗nix mal

The arcs wrapper langua

dedicated data flow

nguage xample: Envisionary

imbedding it in a Jupyt lotebook

> np, occultation an vity

The dataflow C++
template library

unctional programmir ummary

P<sub>ro</sub>vision



You may have applied functional programming without knowing it:

- Spreadsheet programs (most probably)
- ► The \*nix make utility (maybe)
- OpenDX (maybe not)

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

or make A dedicated data flov

A dedicated data flo anguage

xample: Envisionary imbedding it in a Jupyt

nomonic projection
ump, occultation an

Dump, occultation a gravity The dataflow C++

Functional programming

Provision





You may have applied functional programming without knowing it:

- Spreadsheet programs (most probably)
- ► The \*nix make utility (maybe)
- OpenDX (maybe not)

Functional programming can be realized without functions by describing directly the data flow.

Life. Venus and Everything

Biörn Grieger

Spreadsheets

OpenDX

Functional programming summary





You may have applied functional programming without knowing it:

- ► Spreadsheet programs (most probably)
- ► The \*nix make utility (maybe)
- ► OpenDX (maybe not)

Functional programming can be realized without functions by describing directly the data flow. Home grown examples:

arcs wrapper language for make: orchestrates data exchange between programs over the hard disk, intermediate results are preserved between program runs.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

Overview

Spreadsheets

OpenDX

The \*nix make utility

r make A dedicated data flow

l dedicated data flo anguage

cample: Envisionary

nbedding it in a Jupy

otebook nomonic projection ump, occultation ar

mp, occultation an vity dataflow C++

Functional programming

rovision





You may have applied functional programming without knowing it:

- Spreadsheet programs (most probably)
- ► The \*nix make utility (maybe)
- ► OpenDX (maybe not)

Functional programming can be realized without functions by describing directly the data flow. Home grown examples:

arcs wrapper language for make: orchestrates data exchange between programs over the hard disk, intermediate results are preserved between program runs.

**dataflow C++ template library:** implements module objects with inputs and outputs that can be connected to define a data flow.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional

Overview

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper language

dedicated data flow

A dedicated data f anguage

cample: Envisionary mbedding it in a Jupy otebook

nomonic projection ump, occultation an

ump, occultatio avity

template library

summary

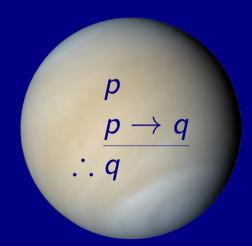
Provision







# Provision: Employing Prolog in rule based science operations planning



Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional

programming

Spreadsheets

Spreadsheets

penDX

The \*nix make utility

The arcs wrapper languag for make

dedicated data flow inguage

Example: Envisionary
Embedding it in a Jupy

Embedding it in a Jupy Notebook Gnomonic projection

onomic projection

Oump, occultation and ravity

The dataflow C emplate library

Functional program summary

Provision



```
human(socrates).
mortal(X):- human(X).
```

Life, Venus and Everything

Björn Grieger

Venera

EnVision.

Functional

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyt

mbedding it in a Jupyt lotebook

Dump, occultation a gravity

emplate library Functional programm

Provision





```
\begin{array}{ll} \operatorname{human}(\operatorname{socrates}). & \longleftarrow \operatorname{\mathsf{Fact}} \\ \operatorname{\mathsf{mortal}}(X): - \operatorname{\mathsf{human}}(X). \end{array}
```

Life, Venus and Everything

Björn Grieger

Venera

EnVision

Functional

programming

Overview Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper langua for make

A dedicated data flow language

Example: Envisionary

imbedding it in a Jupyt Jotebook

Gnomonic projection Dump, occultation and gravity

emplate library

Provision





```
\begin{array}{lll} & \text{human}(\text{socrates}). & \longleftarrow & \text{Fact} \\ & \text{mortal}(\texttt{X}):-& \text{human}(\texttt{X}). & \longleftarrow & \text{Rule} \end{array}
```

Life, Venus and Everything

Björn Grieger

Venera

EnVision.

Functional .

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow

A dedicated data flow language

Example: Envisionary Embedding it in a Jupyte

Embedding it in a Jupy Notebook

Dump, occultation and ravity

Functional programs

Provision





A collection of facts and rules is called a knowledge base.

Life, Venus and Everything

Björn Grieger

Venera

. . ..

EnVision

- .. .

Functional programming

Spreadsheets

OpenDX

The \*nix make ut

The arcs wrapper langua

A dedicated data flow

anguage

mbedding it in a Jupyt lotebook

nomonic projection ump, occultation an avity

template library

Functional programm

mmary

### Provision





```
human (socrates).
                  ← Fact
                      ← Rule
mortal(X):- human(X).
```

A collection of facts and rules is called a *knowledge base*.

When one or more knowledge bases have been loaded, queries can be posed:

```
?- mortal(socrates).
true.
?- mortal(plato).
false.
```

Life. Venus and Everything

Björn Grieger

OpenDX

Provision





# **Inverse query**

Extended knowledge base:

```
human(socrates).
human(plato).
mortal(X):- human(X).
```

Life, Venus and Everything

Björn Grieger

Venera

. . ..

EnVision

unctional programming

Overview

Spreadsheets

The strain make utility

The arcs wrapper langu for make

A dedicated data flow

anguage

mbedding it in a Jupy otebook

Snomonic projection Dump, occultation a gravity

gravity
The dataflow C++

unctional progra

Provision





# **Inverse query**

### Extended knowledge base:

```
human (socrates).
human(plato).
mortal(X):- human(X).
```

### Direct queries:

true.

```
?- mortal(socrates)
true.
?- mortal(plato).
```

Life. Venus and Everything

Björn Grieger

Spreadsheets

Provision





# **Inverse query**

Extended knowledge base:

```
human(socrates).
human(plato).
mortal(X):- human(X).
```

### Inverse query:

```
?- mortal(X).
X = socrates;
X = plato.
```

Life, Venus and Everything

Björn Grieger

/enera

. ..

En\/ision

unctional rogramming

Overview Spreadsheets

nenDX

he ∗nix mak

The arcs wrapper langua

dedicated data flow

nguage

nbedding it in a Jupy otebook

omonic projection

imp, occultation ivity

ne dataflow C+ mplate library

unctional program

Provision

Tako homo





# **Provision**

► We apply scripts to convert event files written by Envisionary for MAPPS to Prolog knowledge bases.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

Functional programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flov language

ranguage Example: Envisionary

mbedding it in a Jupy otebook

imp, occultation and avity

The dataflow C++ emplate library functional programn

ummary

Provision





# **Knowledge bases**

### Head of 'roi\_knowledge\_2.pl':

```
roi_enc_orb_asc(001,3,4275,0).
roi_enc_orb_asc(001,3,4276,0).
roi_enc_orb_asc(001,3,4277,0).
roi_enc_orb_asc(001,3,4278,0).
roi_enc_orb_asc(001,3,4279,0).
```

### Head of 'gs\_knowledge.pl':

```
gs_pat_orb_asc_cyc(1,10002,0,3).
gs_pat_orb_asc_cyc(1,10017,0,3).
gs_pat_orb_asc_cyc(1,10032,0,3).
gs_pat_orb_asc_cyc(1,10048,0,3).
gs_pat_orb_asc_cyc(1,10063,0,3).
```

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

Functional

Overview

OpenDX

The \*nix make utility

make dedicated data flow

nguage

mbedding it in a Jupy otebook

otebook nomonic projection

lump, occultation a ravity se dataflow C++

emplate library unctional program ummary

Provision

### Take\_home





# **Example query**

"Tell me, for cycle 1, the orbit branches with orbit pattern number, orbit number, and branch direction, where downlink is geometrically possible and where no ROI observation is planned."

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

....

rogramming

)verview

Spreadsheets

The \*nix make utility

The arcs wrapper language for make

> dedicated data flow inguage

ample: Envisionary

mbedding it in a Jupy otebook

nomonic projection ump, occultation an avity

nplate library nctional programmin

Provision



#### Query answer

Björn Grieger Provision

Life. Venus and

Everything

What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )?

Life, Venus and Everything

Björn Grieger

/enera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

The \*nix make utility

The \*nix make utility

The arcs wrapper language

r make A dedicated data flow

language

Example: Envisionary
Embedding it in a Jupy
Notebook

nomonic projection ump, occultation and

Dump, occultation ar gravity 'he dataflow C++

emplate library iunctional programn ummary

Provision



What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )?

If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper langua

A dedicated data flow

anguage Example: Envisionary

mbedding it in a Jupyt lotebook

nomonic projection ump, occultation an avity

template library Functional programm summary

Provision

Tako homo





What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )?

If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

$$X = n \cdot D$$
$$Y = m \cdot D$$
$$X + Y = (n + m) \cdot D$$

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional programming

Overview

Spreadsheets OpenDX

OpenDX

The \*nix make utility

make dedicated data flow

dedicated data flow guage

xample: Envisionary mbedding it in a Jupy otebook

omonic projection mp, occultation and wity

emplate library

Provision



What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )?

If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

$$X = n \cdot D$$

$$Y = m \cdot D$$

$$X + Y = (n + m) \cdot D$$

$$X + Y = p \cdot D'$$

$$X = q \cdot D'$$

$$Y = (p - q) \cdot D'$$

Life, Venus and Everything

Björn Grieger

Venera

Magella

EnVision

Functional

Overview

Spreadsheets OpenDX

he \*nix make utility

The arcs wrapper languag

A dedicated data flow

anguage Example: Envisionary

Embedding it in a Jup Notebook

omonic projection mp, occultation and wity

emplate library

Provision



What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )?

If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

EnVision

Functional programming

Overview

Spreadsheets OpenDX

The \*nix make utility

The arcs wrapper langua

A dedicated data flow

anguage Example: Envisionary

mbedding it in a Jupyt lotebook

nomonic projection ump, occultation an avity

template library Functional programm summary

Provision

Tako homo





What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )? If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

```
gcd(X,X,X).
gcd(X,Y,D) :- X<Y, Y1 is Y-X, gcd(X,Y1,D).
gcd(X,Y,D) :- Y<X, gcd(Y,X,D).</pre>
```

Life, Venus and Everything

Björn Grieger

Venera

. . ..

EnVision

unctional programming

Overview Spreadsheets

OpenDX

The \*nix make utility

dedicated data flow

nguage xample: Envisionary

bedding it in a Jupyt tebook

omonic projection mp, occultation and vity

emplate library unctional programn

Provision



What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )? If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

```
gcd(X,X,X).
gcd(X,Y,D) :- X<Y, Y1 is Y-X, gcd(X,Y1,D).
gcd(X,Y,D) :- Y<X, gcd(Y,X,D).</pre>
```

```
?- \gcd(6,4,D).
```

Life, Venus and Everything

Björn Grieger

Venera

N 4 - - - II -

EnVision

Functional programming

Spreadsheets

OpenDX
The \*nix make utility

The arcs wrapper languag for make

dedicated data flow

xample: Envisionary mbedding it in a Jupyt Intebook

nomonic projection ump, occultation an

ravity
ne dataflow C++

inctional programi immary

Provision





What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )? If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

```
gcd(X,X,X).
gcd(X,Y,D) :- X<Y, Y1 is Y-X, gcd(X,Y1,D).
gcd(X,Y,D) :- Y<X, gcd(Y,X,D).</pre>
```

```
?- gcd(6,4,D).
D = 2
```

Life, Venus and Everything

Björn Grieger

Venera

Magalla

EnVision

Functional programming

Spreadsheets

The \*nix make utility

make dedicated data flow

dedicated data flow inguage

xample: Envisionary mbedding it in a Jupy otebook

omonic projection mp, occultation an vity

he dataflow C+emplate library unctional program

unctional progran ummary

#### Provision



What is the greatest common divisor (gcd) D of X and Y (with  $X, Y, D \in \mathbb{N}$ )? If D is the gcd of X and Y, then D is also the gcd of X and X + Y.

```
gcd(X,X,X).
gcd(X,Y,D) :- X<Y, Y1 is Y-X, gcd(X,Y1,D).
gcd(X,Y,D) :- Y<X, gcd(Y,X,D).</pre>
```

```
?- gcd(10000002,20000001,D).
D = 3
```

Life, Venus and Everything

Björn Grieger

Venera

Magalla

EnVision

unctional programming

Overview Spreadsheets

Spreadsheets

The \*nix make utility

The arcs wrapper language

dedicated data flow

nguage

mbedding it in a Jupy otebook

omonic projection mp, occultation and wity

emplate library unctional programi

ummary

#### Provision





Life, Venus and Everything

Björn Grieger

/enera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

for make

A dedicated data flow language

Example: Envisionary
Embedding it in a Jupyt

Embedding it in a Jupy Notebook

Dump, occultation and gravity

emplate library unctional programn

Provision







Venus is more beautiful than Mars.

Life. Venus and Everything

Björn Grieger

Spreadsheets







Venus is more beautiful than Mars.

Life, Venus and Everything

Björn Grieger

/enera

Magellan

EnVision

Functional programming

Overview

Spreadsheets

)penDX

The \*nix make utility

or make A dedicated data flow

A dedicated data flow anguage

Example: Envisionary Embedding it in a Jupy

mbedding it in a Jupy otebook nomonic projection

Dump, occultation and ravity

mplate library Inctional programm

mmary









Venus is more beautiful than Mars.

Life. Venus and Everything

Björn Grieger

Spreadsheets







Venus is more beautiful than Mars.



EnVision planning is quite an entertaining puzzle.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

unctional programming

Overview

Spreadsheets

OpenDX

The \*nix make utility

The arcs wrapper languag for make

A dedicated data flow language

ample: Envisionary

omonic projection
mp, occultation an

avity ≥ dataflow C++ polate library

nctional program

Provision







Venus is more beautiful than Mars.



EnVision planning is quite an entertaining puzzle.



There is programming beyond Python.

Life, Venus and Everything

Björn Grieger

Venera

Magellan

**EnVision** 

Functional programming

Overview

Spreadsheets

OpenDX

The wais make u

The \*nix make utility

The arcs wrapper language

A dedicated data flow

A dedicated data fit language

> cample: Envisionary nbedding it in a Jupy otebook

omonic projection

e dataflow C

plate library ctional prograi

nmary

P<sub>ro</sub>vision

