



**HAL**  
open science

# CGAL: The Computational Geometry Algorithms Library

Monique Teillaud

► **To cite this version:**

Monique Teillaud. CGAL: The Computational Geometry Algorithms Library. GD23 - 31st International Symposium on Graph Drawing and Network Visualization, Sep 2023, Palerme, Italy. hal-04241529

**HAL Id: hal-04241529**

**<https://inria.hal.science/hal-04241529>**

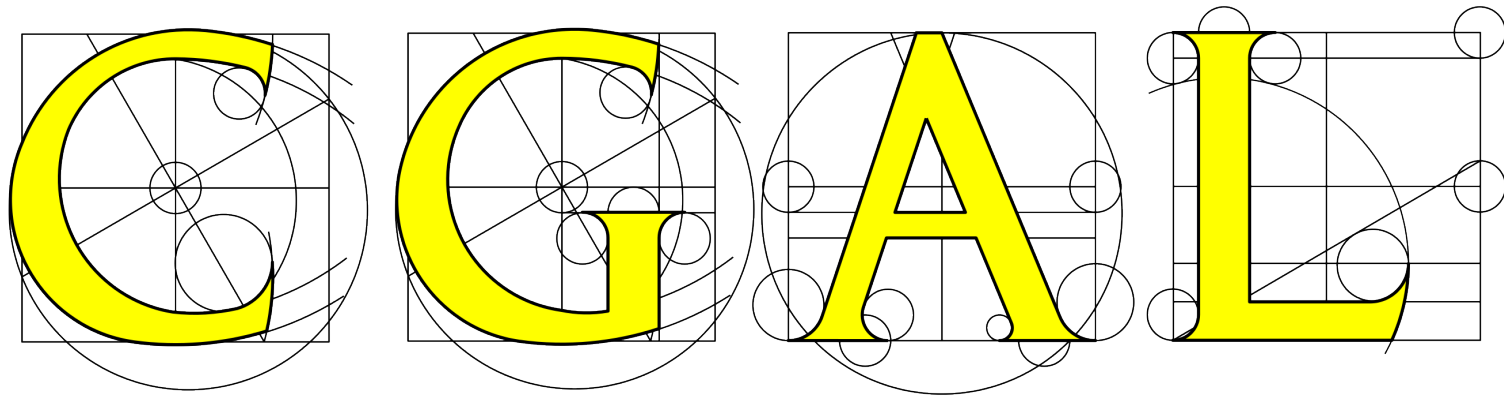
Submitted on 24 May 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



*The Computational Geometry Algorithms Library*



Monique Teillaud



# Travel by train

plane: Paris → Palermo 0.43 T CO<sub>2</sub>

train: Nancy → Palermo 0.08 T CO<sub>2</sub>

<https://www.goodplanet.org>



# This talk

# This talk

**NOT**

- a CGAL course
- an advertisement for CGAL

# This talk

## Ingredients to success

invitation email from PC chairs:

*“We hope that your invited talk would give us some insight in the famed CGAL project, its genesis, central deliberations scientifically and organizationally, and lessons learned over the years, with all the ups and downs regarding student code, quality considerations but also fruitful synergies.”*

Is CGAL successful ?

# Is CGAL successful ?

- several thousands of downloads for each release
- included in debian, macports, etc



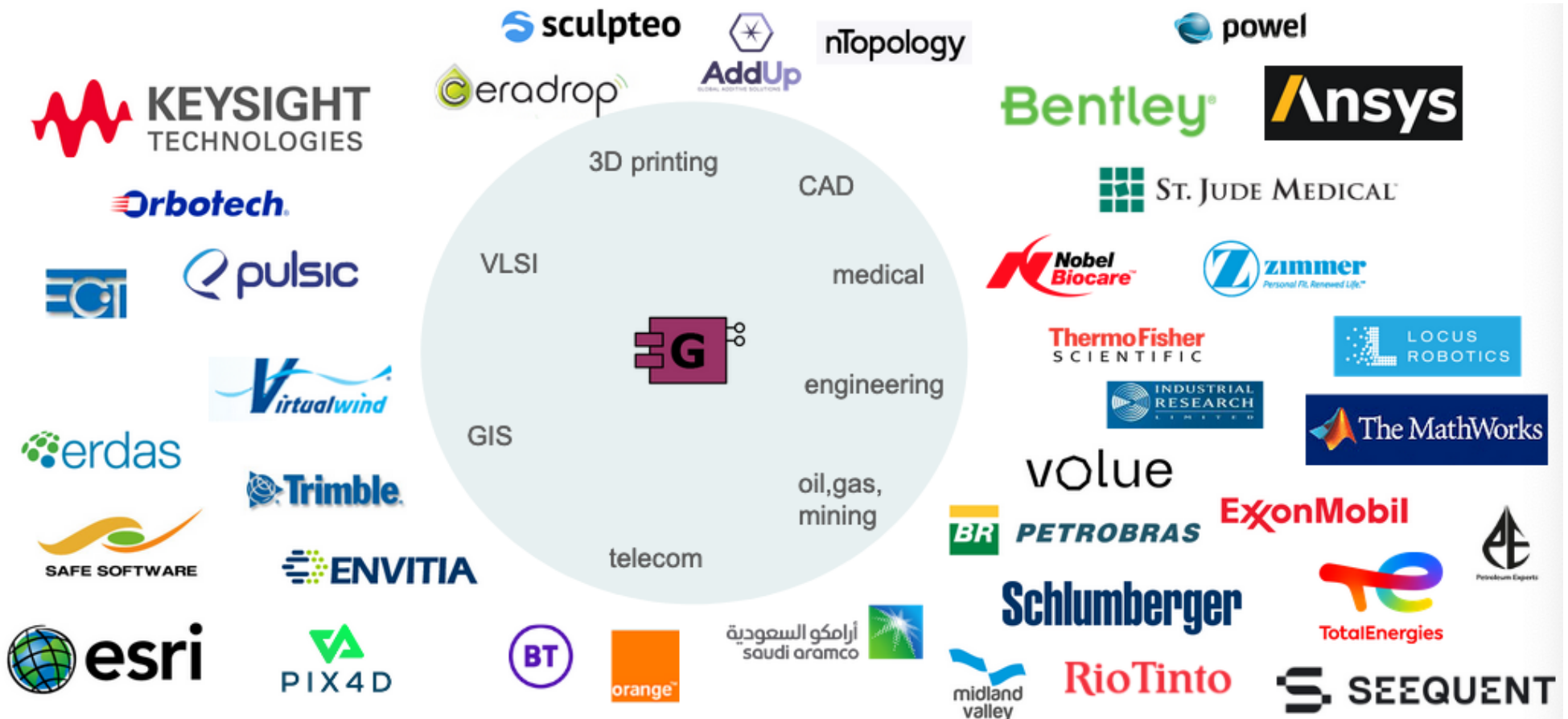
# Is successful ?

## Projects using CGAL

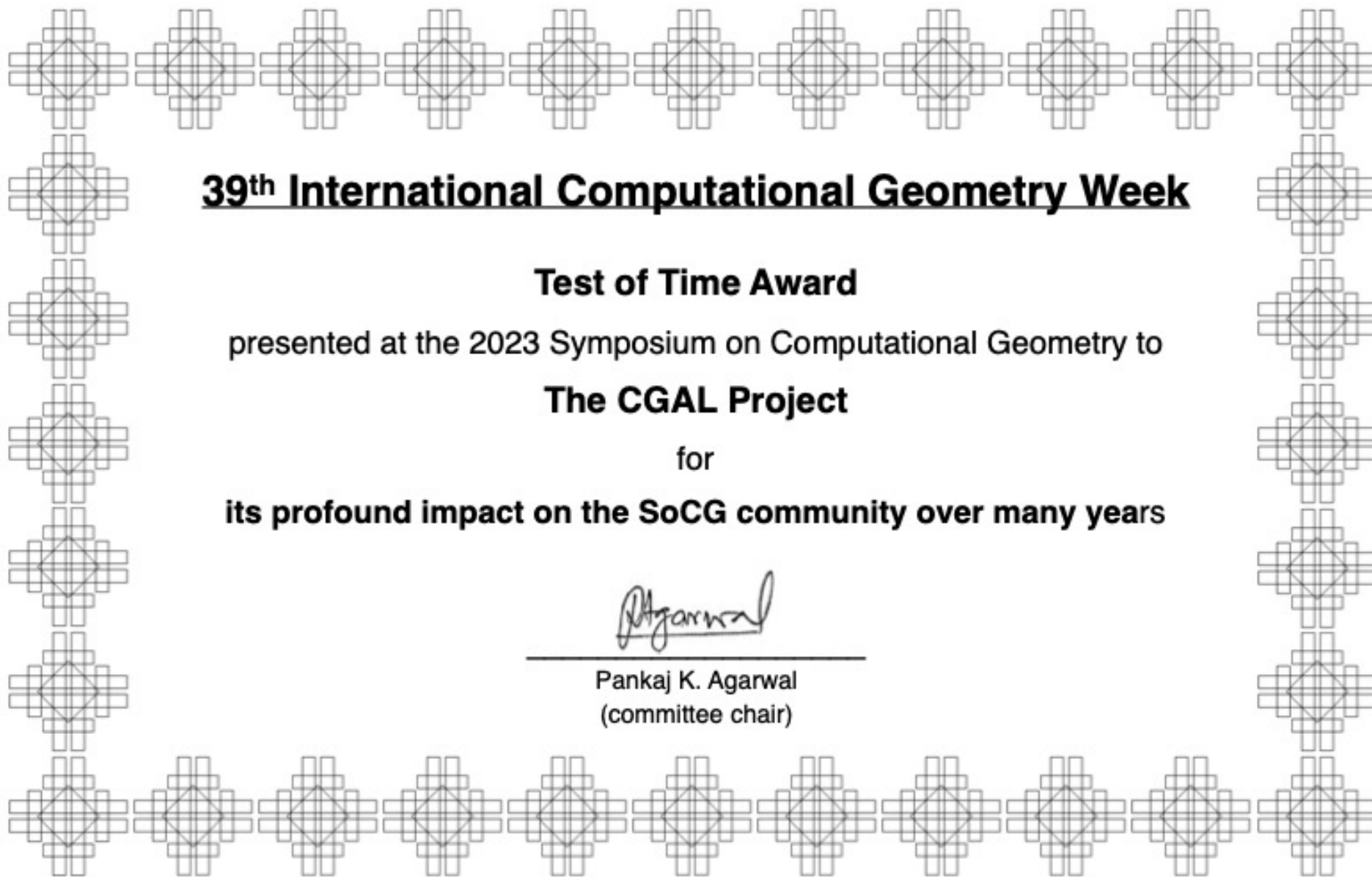
- Art
- Architecture, Buildings Modeling, Urban Modeling
- Astronomy
- Computational Geometry and Geometric Computing
- Computer Graphics
- Computational Topology and Shape Matching
- Computer Vision, Image Processing, Photogrammetry
- Games, Virtual Worlds
- Geographic Information Systems
- Geology and Geophysics
- Geometry Processing
- Mathematics
- Medical Modeling and Biophysics
- Mesh Generation and Surface Reconstruction
- 2D and 3D Modelers
- Molecular Modeling
- Motion Planning
- Optimization
- Particle Physics, Materials, Nanostructures, Microstructures, Fluid Dynamics
- Peer-to-Peer Virtual Environment
- Sensor Networks

# Is CGAL successful ?

A few commercial users



# Is successful ?



## **39<sup>th</sup> International Computational Geometry Week**

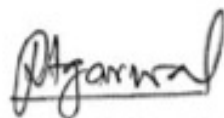
### **Test of Time Award**

presented at the 2023 Symposium on Computational Geometry to

### **The CGAL Project**

for

**its profound impact on the SoCG community over many years**



---

Pankaj K. Agarwal  
(committee chair)

# Is successful ?

*“ Dear CGAL Editorial Board,*

*The SoCG ToT Award Committee has chosen the **CGAL project** as one of the recipients of the*

**2023 SoCG Test of Time Award.**

*In this case the Committee deliberately and **exceptionally overstepped its mandate** of honoring individual papers since it strongly believes that CGAL is a **contribution** to the **SoCG Community** whose **impact** as a whole has been and still is at least as **strong** and **profound** as that of any individual paper.*

*Congratulations to all of you for this well deserved **recognition**. ”*

Pankaj Agarwal, Siu-Wing Cheng, Raimund Seidel, Emo Welzl  
(SoCG ToT Award Committee)

# Is successful ?

*“ Dear CGAL Editorial Board,*

*The SoCG ToT Award Committee has chosen the **CGAL project** as one of the recipients of the*

**2023 SoCG Test of Time Award.**

*In this case the Committee deliberately and **exceptionally overstepped its mandate** of honoring individual papers since it strongly believes that CGAL is a **contribution** to the **SoCG Community** whose **impact** as a whole has been and still is at least as **strong** and **profound** as that of any individual paper.*

*Congratulations to all of you for this well deserved **recognition**.* ”

Pankaj Agarwal, Siu-Wing Cheng, Raimund Seidel, Emo Welzl  
(SoCG ToT Award Committee)

# Historical Context

# An Assessment (90's)

Computational Geometry: mostly theoretical

Smart algorithms, often hard/impossible to implement

A few pieces of software here and there

Not stable

Code often lost when students leave

# 1996 CG Impact Task Force Report

Nina Amenta

Tetsuo Asano

Gill Barequet

Marshall Bern

Jean-Daniel Boissonnat

John Canny

**Bernard Chazelle**

Ken Clarkson

David Dobkin

Bruce Donald

Scot Drysdale

Herbert Edelsbrunner

David Eppstein

A. Robin Forrest

Steve Fortune

Ken Goldberg

Michael Goodrich

Leonidas J. Guibas

Pat Hanrahan

Chris M. Hoffmann

Dan Huttenlocher

Hiroshi Imai

David Kirkpatrick

D.T. Lee

Kurt Mehlhorn

Victor Milenkovic

Joe Mitchell

Mark Overmars

Richard Pollack

Raimund Seidel

Micha Sharir

Jack Snoeyink

Godfried Toussaint

Seth Teller

Herb Voelcker

Emo Welzl

Chee Yap



# 1996 CG Impact Task Force Report

Among the key recommendations:

- Production and distribution of **usable (and useful)** geometric codes
- **Reward structure** for implementations in academia

# Local software projects in Europe



Universiteit  
Utrecht

PlaGeo, SpaGeo

**ETH** zürich

XYZ GeoBench



max planck institut  
informatik

LEDA



C++GAL

# Local software projects in Europe



PlaGeo, SpaGeo

**ETH** zürich

XYZ GeoBench



LEDA



C++GAL

**cigale** (*cicada*)

Francis Avnaim



# A Strong Will to Gather Efforts

# Initial Funding (FP4 ESPRIT Program)

## Consortium



**ETH** zürich



**Mark Overmars**

Peter Widmayer, **Emo Welzl**

**Kurt Mehlhorn**

**Jean-Daniel Boissonnat**

Dan Halperin

Helmut Alt

Stefan Näher

Sabine Stifter

# Initial Funding (FP4 ESPRIT Program)

## Consortium



Universiteit  
Utrecht

**ETH** zürich



max planck institut  
informatik



TEL AVIV  
UNIVERSITY



Freie Universität Berlin



JOHANNES KEPLER  
UNIVERSITÄT LINZ



CGAL: Oct **1996** (21 months)

GALIA: Nov 1998 (18 months)

# Goals

## Promote the research in computational geometry

*“make the large body of geometric algorithms developed in the field of CG available for industrial applications”*

Mission Statement

CGAL EU Project Proposal, 1996

## Create a reward structure for implementations in academia

# Goals

## Promote the research in computational geometry

*“make the large body of geometric algorithms developed in the field of CG available for industrial applications”*

Mission Statement

CGAL EU Project Proposal, 1996

## Create a reward structure for implementations

in academia

High quality

- Robustness
- Tests
- Documentation
- Review
- etc



# Technical choices

# Pre-history

*At a meeting at Utrecht University in January 1995,*

*Olivier Devillers, Andreas Fabri, Wolfgang Freiseisen,  
Geert-Jan Giezeman, Mark Overmars, Stefan Schirra,  
Otfried Schwarzkopf, and Sven Schönherr*

*discussed the foundations of the **CGAL** kernel.*

*Many design and software engineering issues were addressed, e.g.*

- *naming conventions,*
- *coupling of classes (flat versus deep class hierarchy),*
- *memory allocation,*
- *programming conventions,*
- *mutability of atomic objects, points and vectors,*
- *storing additional information,*
- *orthogonality of operations on the kernel objects,*
- *viewing non-constant-size objects like polygons as dynamic data structures (and hence not as part of the (innermost) kernel).*

[ CGAL Manual ]

# Working Group

CGAL first generation *dinosaurs*

Andreas Fabri



Geert-Jan Giezeman



Lutz Kettner



Stefan Schirra



Sven Schönherr



# Working Group

Language

C++

# Working Group

Language

C++

Design

genericity/flexibility through **templates**

STL-like:

```
template <class T>
void swap(T &a, T &b)
{ T tmp = a; a = b; b = tmp; }
```

[ A. Fabri, G.-J. Giezeman, L. Kettner, S. Schirra, and S. Schönherr.  
On the design of CGAL a computational geometry algorithms library.  
Software: Practice and Experience 2000. ]

# Structure

## (Extendible) Kernel

Constant-size objects

Operations on them

[ S. Hert, M. Hoffmann, L. Kettner, S. Pion, and M. Seel.  
An adaptable and extensible geometry kernel.  
Workshop on Algorithm Engineering, LNCS 2001 ]

# Structure

## (Extendible) Kernel

Constant-size objects

Operations on them

[ S. Hert, M. Hoffmann, L. Kettner, S. Pion, and M. Seel.  
An adaptable and extensible geometry kernel.  
Workshop on Algorithm Engineering, LNCS 2001 ]

## Basic Library

Data Structures & Algorithms

Triangulations, Arrangements, etc

# Structure

## (Extendible) Kernel

Constant-size objects

Operations on them

[ S. Hert, M. Hoffmann, L. Kettner, S. Pion, and M. Seel.  
An adaptable and extensible geometry kernel.  
Workshop on Algorithm Engineering, LNCS 2001 ]

## Basic Library

Data Structures & Algorithms

Triangulations, Arrangements, etc

## Support Library

I/O, Visualization

~> rely on external tools



# Structure

GeomTraits **concept**

```
template < class Geom_traits, class Tds = Default >
class Delaunay_triangulation_3;
{
    typedef typename Geom_traits::Point_3 Point;

    Vertex_handle insert(const Point& p) { ... }
}
```

# Structure

GeomTraits **concept**

```
template < class Geom_traits, class Tds = Default >
class Delaunay_triangulation_3;
{
    typedef typename Geom_traits::Point_3 Point;
    Vertex_handle insert(const Point& p) { ... }
}
```

is a **model**

```
typedef CGAL::Exact_predicates_inexact_constructions_kernel K;
```

# Structure

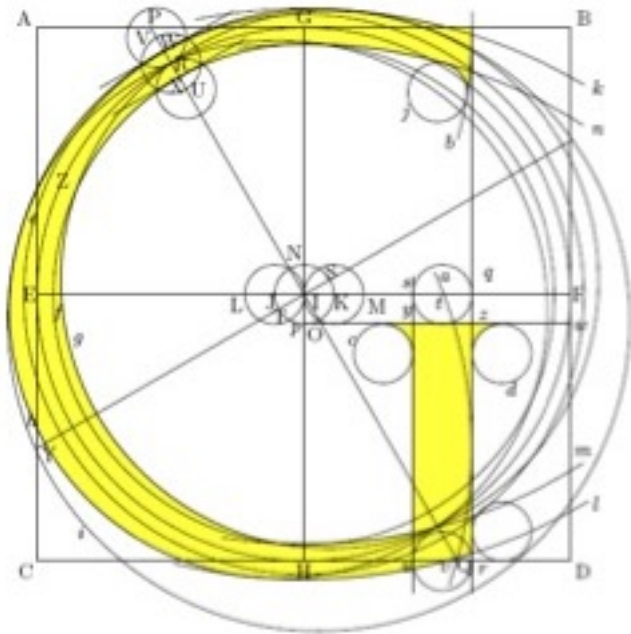
GeomTraits **concept**

```
template < class Geom_traits, class Tds = Default >
class Delaunay_triangulation_3;
{
    typedef typename Geom_traits::Point_3 Point;
    Vertex_handle insert(const Point& p) { ... }
}
```

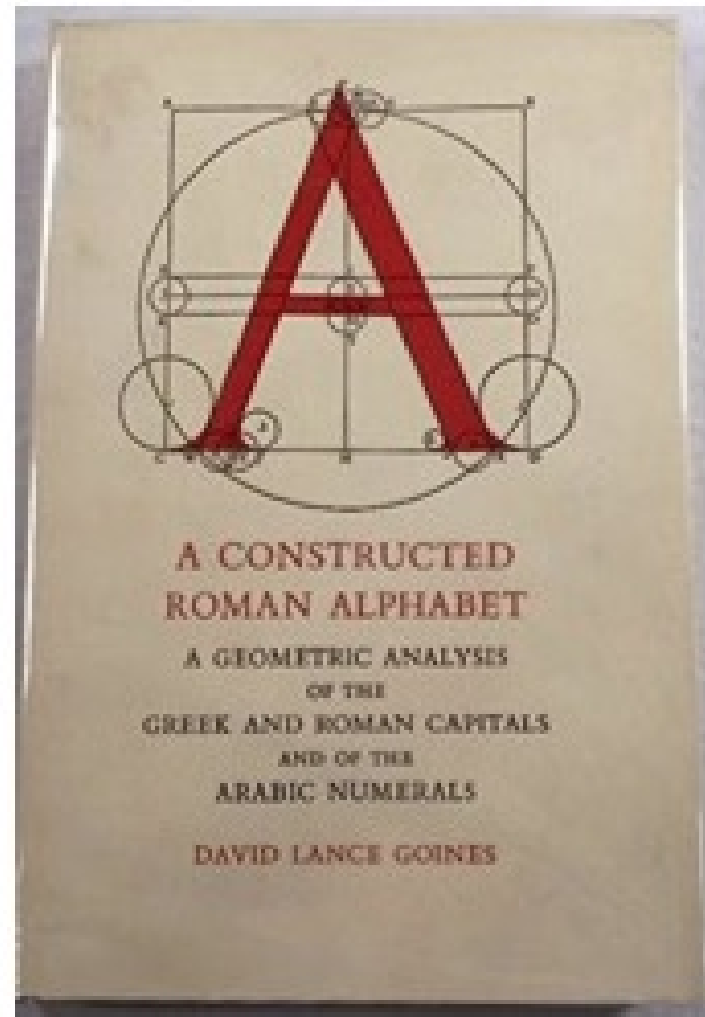
is a **model**

```
typedef CGAL::Exact_predicates_inexact_constructions_kernel K;
typedef CGAL::Delaunay_triangulation_3<K > Delaunay;
int main()
{
    Delaunay T;
    p = Point(0,0,0);
    T.insert(p);
}
```

# Visual identity



Remco Veltkamp



# Editorial Board

(2001)

# Reviews

new package  $\simeq$  journal article

Quality

*“Reward structure for implementations in academia”*

# Project Rules

- Responsibilities
  - Developer vs. Editor
  - Release / Review managers
- Voting rules
- Contribution process

# Project Rules

- Responsibilities
  - Developer vs. Editor
  - Release / Review managers
- Voting rules
- Contribution process

A profitable investment

prevent the same discussions from going on and on



# Misc.

**Discusses** arising questions

Maintains the **website**

**Represents** the CGAL project

etc

# Current Members

Pierre Alliez



Guillaume Damiand



*review manager*

Andreas Fabri



Efi Fogel



Michael Hoffmann



Sébastien Loriot



Marc Pouget



Laurent Rineau



*release manager*

Monique Teillaud



# Limited ego

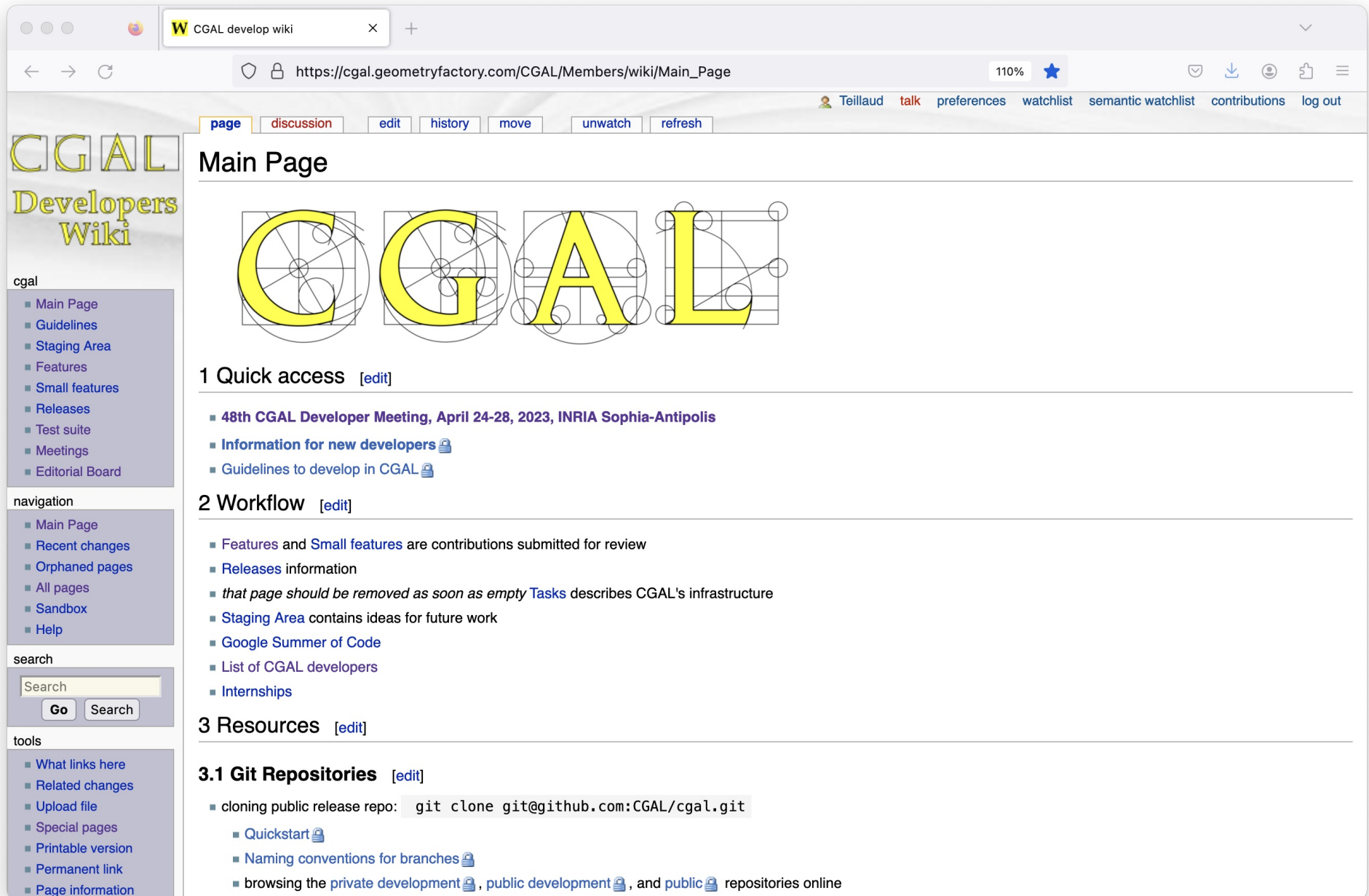
## No Chairperson

### Board Manager

- period of 6 months
- round-robin process

# Infrastructure

# Wiki



The screenshot shows a web browser window displaying the CGAL Developers Wiki Main Page. The browser's address bar shows the URL `https://cgal.geometryfactory.com/CGAL/Members/wiki/Main_Page`. The page features a navigation menu on the left with sections for 'cgal', 'navigation', 'search', and 'tools'. The main content area is titled 'Main Page' and includes a large 'CGAL' logo with geometric diagrams. Below the logo, there are three main sections: '1 Quick access', '2 Workflow', and '3 Resources'. The '1 Quick access' section lists the 48th CGAL Developer Meeting, information for new developers, and guidelines for development. The '2 Workflow' section details the submission process for features and small features, release information, and the staging area. The '3 Resources' section includes a subsection for Git repositories, providing a command to clone the public release repository and links to quickstart, naming conventions, and browsing repositories.

W CGAL develop wiki x +

← → ↻ `https://cgal.geometryfactory.com/CGAL/Members/wiki/Main_Page` 110% ★

Teillaud talk preferences watchlist semantic watchlist contributions log out

page discussion edit history move unwatch refresh

## CGAL Developers Wiki

cgal

- Main Page
- Guidelines
- Staging Area
- Features
- Small features
- Releases
- Test suite
- Meetings
- Editorial Board

navigation

- Main Page
- Recent changes
- Orphaned pages
- All pages
- Sandbox
- Help

search

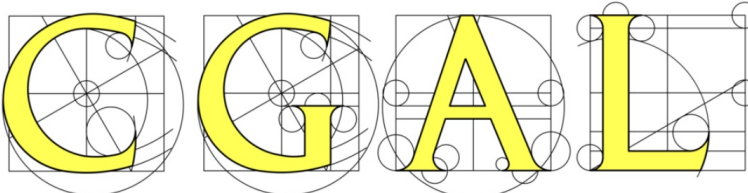
Search

Go Search

tools

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link
- Page information

## Main Page



### 1 Quick access [\[edit\]](#)

- [48th CGAL Developer Meeting, April 24-28, 2023, INRIA Sophia-Antipolis](#)
- [Information for new developers](#)
- [Guidelines to develop in CGAL](#)

### 2 Workflow [\[edit\]](#)

- [Features](#) and [Small features](#) are contributions submitted for review
- [Releases](#) information
- *that page should be removed as soon as empty [Tasks](#) describes CGAL's infrastructure*
- [Staging Area](#) contains ideas for future work
- [Google Summer of Code](#)
- [List of CGAL developers](#)
- [Internships](#)

### 3 Resources [\[edit\]](#)

#### 3.1 Git Repositories [\[edit\]](#)

- cloning public release repo: `git clone git@github.com:CGAL/cgal.git`
  - [Quickstart](#)
  - [Naming conventions for branches](#)
  - browsing the [private development](#), [public development](#), and [public](#) repositories online

## Submission / Review

The screenshot shows a Wiki page for 'Features/Vanilla/1st round'. The page has a navigation bar at the top with links for 'page', 'discussion', 'edit', 'history', 'move', 'watch', and 'refresh'. The user 'Teillaud' is logged in, with links for 'talk', 'preferences', 'watchlist', 'semantic watchlist', 'contributions', and 'log out'. The page content is organized into sections: '1 Submission', '2 Reviews', '3 Discussion', and '4 Conclusion'. A search bar at the bottom shows a search for 'rev' with 1 of 1 match. The left sidebar contains navigation and search tools.

**CGAL Developers Wiki**

cgal

- Main Page
- Guidelines
- Staging Area
- Features
- Small features
- Releases
- Test suite
- Meetings
- Editorial Board

navigation

- Main Page
- Recent changes
- Orphaned pages
- All pages
- Sandbox
- Help

search

Search

Go Search

tools

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

page discussion edit history move watch refresh

Teillaud talk preferences watchlist semantic watchlist contributions log out

### Features/Vanilla/1st round

< Features | Vanilla

*This subpage should contain more detailed comments by the authors that are specific for this submission, for instance, which pages contain new documentation, where one should read first, etc.*

Contents [show]

#### 1 Submission [edit]

*Use macro `\cgalModifBegin` and `\cgalModifEnd` to indicate the modifications. When you add a new function, use these macro around its documentation.*

You can also use a macro like `\tred{modified text}` if your modification concerns only a small part of text.

```
ALIASES += tred{1}="\htmlonly < b > < font color='red' >\endhtmlonly \1 \htmlonly < /font > < /b > \endhtmlonly "
```

**THE PAGE MUST CONTAIN THE FOLLOWING LINKS**

- [link to pdf file of 1st submission] - 2012-12-21 Monique Teillaud

#### 2 Reviews [edit]

- /Review Andreas Fabri - 2013-01-01

*Nobody is allowed to edit the comments of a reviewer. Answers to the review and the subsequent discussion takes place on a separate discussion page.*

#### 3 Discussion [edit]

- /Answer to review and discussion -- 2013-01-03 -- Irene

#### 4 Conclusion [edit]

- There is a major unsolved disagreement, see [Section 'major issues' of the discussion](#) -- 2013-01-10 Andreas Fabri
- However, we decided to close the round for now and await the next submission. -- 2013-01-11 Laurent Rineau (primary reviewer)

× rev ^ v  Highlight All  Match Case  Match Diacritics  Whole Words 1 of 1 match

# Documentation

Previously:

Home-made LaTeX tools

Nowadays:

Doxygen

+ Home-made scripts

# Tests

Optimal_transportation_reconstruction_2	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	15	CXX20-Release_master
Optimal_transportation_reconstruction_2_Demo	y	y	y	t	t	t	r	y	y	y	y	y	r	r	r	y	r	y	y	y	y	y	y	y	y	r	r	r	r	r	y	y	y	y	y	16	Debian-Stable_?	
Optimal_transportation_reconstruction_2_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	17	Debian-Testing_?
Orthtree	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	18	Debian-stable-Release_?
Orthtree_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	19	Debian-testing-clang-main_?
Partition_2	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	20	Fedora-32-Release_?
Partition_2_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	21	Fedora-32_?
Periodic_2_triangulation_2	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	t	y	y	y	y	y	y	y	y	y	y	y	y	y	y	22	Fedora-Release_?
Platform Description	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32			23	Fedora-rawhide-Release_?		
Periodic_2_triangulation_2_Demo	y	y	y	t	t	r	y	y	y	y	y	r	r	r	y	r	y	y	y	y	y	y	y	y	r	r	r	r	r	y	y	y	y	y	y	24	Fedora-rawhide_?	
Periodic_2_triangulation_2_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	25	Fedora-with-LEDA_?
Periodic_3_mesh_3	y	y	y	y	y	y	y	y	y	y	y	y	n	t	y	y	y	n	y	w	y	y	y	n	n	y	y	y	y	y	y	y	y	y	y	26	Fedora_?	
Periodic_3_mesh_3_Examples	y	y	y	y	y	y	y	y	y	y	y	y	t	y	y	y	y	y	w	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	27	Ubuntu-Latest-CXX11_?
Periodic_3_triangulation_3	y	y	y	y	y	w	t	y	y	y	y	t	y	y	y	w	y	t	t	y	t	y	y	y	w	y	y	y	y	y	y	y	y	y	y	28	Ubuntu-Latest_?	
Periodic_3_triangulation_3_Demo	y	y	y	t	t	r	y	y	y	y	y	r	r	r	y	r	y	y	y	y	y	y	y	y	r	r	r	r	r	y	y	y	y	y	y	y	29	Ubuntu-latest-GCC6-CXX1z_?
Periodic_3_triangulation_3_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	30	Ubuntu-latest-GCC6-Release_?
Periodic_4_hyperbolic_triangulation_2	y	y	y	y	y	w	y	y	y	y	y	y	y	y	y	w	y	y	y	y	y	y	y	y	y	y	y	y	y	w	y	y	y	y	y	31	Ubuntu-latest-GCC6_?	
Periodic_4_hyperbolic_triangulation_2_Demo	y	y	y	t	t	r	w	y	y	y	y	r	r	r	y	r	y	w	y	y	y	y	y	r	r	r	r	r	y	y	y	y	y	y	y	y	32	clang1200-release-64bits_bigsur
Periodic_4_hyperbolic_triangulation_2_Examples	y	y	y	y	y	w	y	y	y	y	y	y	y	y	y	w	y	y	y	y	y	y	y	y	y	y	y	y	y	w	y	y	y	y	y	y		
Periodic_Lloyd_3_Demo	y	y	y	t	t	r	y	y	y	y	y	r	r	r	y	r	y	y	y	y	y	y	y	y	r	r	r	r	r	y	y	y	y	y	y	y	1	MSVC-2019-Community-Release_?
Point_set_2	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	2	MSVC-2022-Community-Release_?
Point_set_2_Examples	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	3	MSVC-2022-Preview-Release_?
Point_set_3	y	y	y	y	y	r	y	y	y	y	y	r	r	r	r	r	r	r	r	y	r	r	y	r	r	r	r	r	r	y	y	y	y	y	y	y	4	MSVC2015-Release-64bits_?
Point_set_3_Examples	y	y	y	y	y	t	t	t	t	t	y	y	t	y	y	y	t	y	y	t	y	t	y	t	y	y	y	y	t	y	y	y	y	y	y	y	5	MSVC2017-Debug-64bits_?
Platform Description	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32			6	MSVC2015-Release-64bits_?		
Point_set_processing_3	y	y	y	y	y	r	t	t	t	t	t	y	r	t	r	r	r	t	t	r	w	t	t	t	y	r	r	r	t	r	y	y	y	y	y	y	7	Apple-clang-default_Darwin-18.5_Release
Point_set_processing_3_Examples	y	y	y	y	y	r	t	t	t	t	r	r	t	r	r	r	t	r	w	t	r	t	r	r	r	r	r	r	t	r	t	y	y	y	y	y		
Poisson_surface_reconstruction_3	y	y	y	y	y	r	t	y	t	t	t	y	y	t	y	y	y	t	t	y	r	t	t	t	y	y	y	y	t	y	y	y	y	y	y	y		
Poisson_surface_reconstruction_3_Examples	y	y	y	y	y	y	y	y	t	t	t	y	y	t	y	y	y	t	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	
Polygon	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y	y		

also for Documentation

in early times, distributed over the CGAL sites



# Files

Previously:

`cvs` then `svn` at INRIA

Nowadays:

`github`

# Installation

Evolution = Simplification

initially: rough process

then cmake

# Installation

Evolution = Simplification

initially: rough process

then cmake

nowadays **easier than ever**

- “ Since CGAL version 5.0, CGAL is **header-only** by default, which means that there is **no need to build CGAL** before it can be used. ”

# Installation

*Running demos (example MacOS)*

*install CGAL dependencies + Qt*

```
brew install cgal
```

```
brew install qt5
```

*download CGAL sources with examples and demos*

```
https://github.com/CGAL/cgal/releases/CGAL-5.6.zip
```

*set up environment variables*

```
CGAL_DIR=/Users/teillaud/CGAL-5.6
```

```
Qt5_DIR=/opt/homebrew/Cellar/qt@5/5.15.10_1/lib/cmake
```

*compile demos*

```
cd CGAL_DIR/demo/Your_favorite_package
```

```
mkdir build; cd build
```

```
cmake ..
```

```
make
```

*run the demo(s)*

# Open-Source Project

# Licence

November, 2003 (Release 3.0)

**Contributions welcome**

## Double licence



suits the needs of most academic users

Commercial for companies that sell their software



January, 2003, INRIA startup

**Andreas Fabri**





January, 2003, INRIA startup

**Andreas Fabri**



sells

commercial licenses

support

Ownership stays to the institutes/universities of authors





Responsible for **long-term maintenance**

C++ / compilers evolutions  
portability  
etc

Maintenance of **infrastructure**

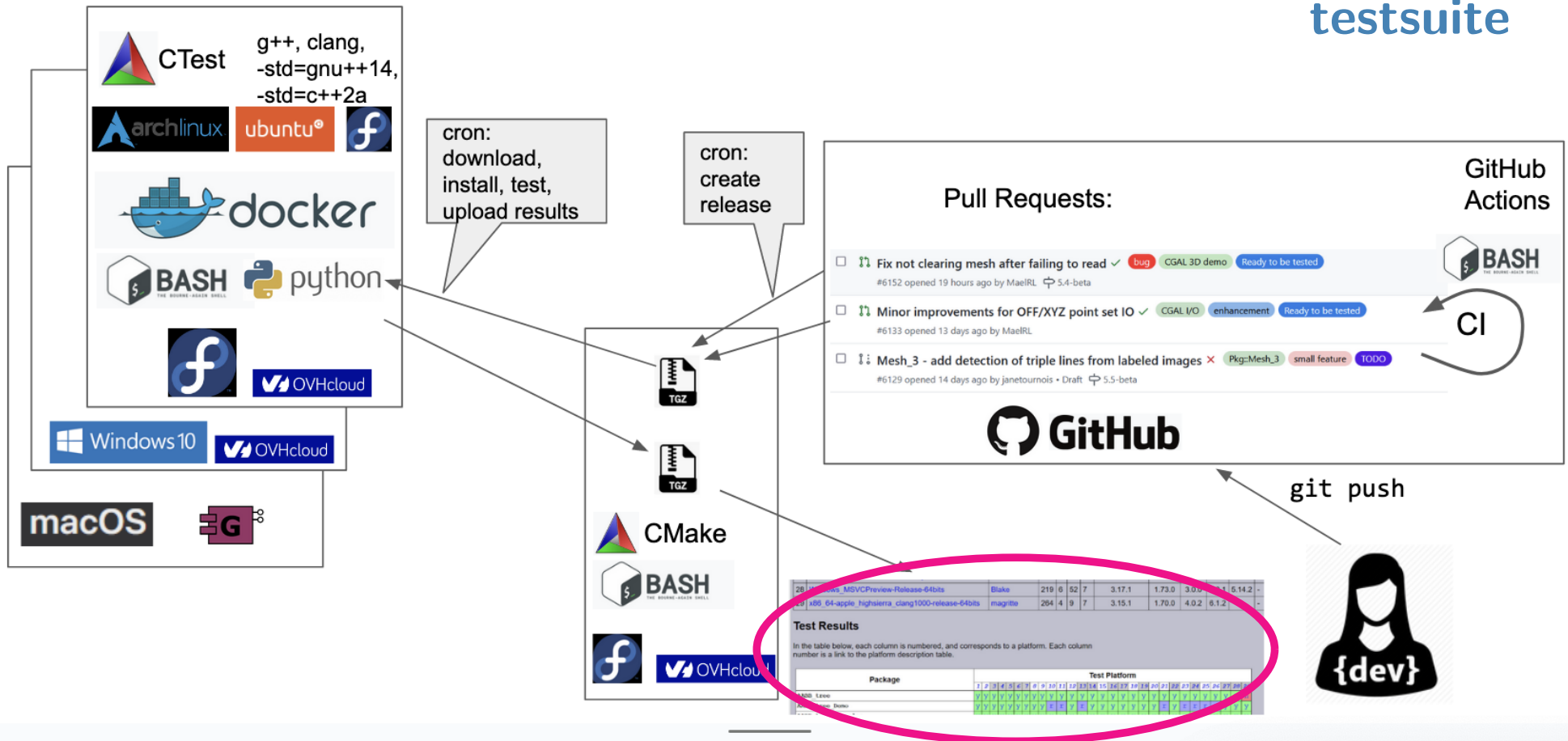
github  
wiki  
provider for the website  
testsuite  
documentation tools

**releases**

# GeometryFactory

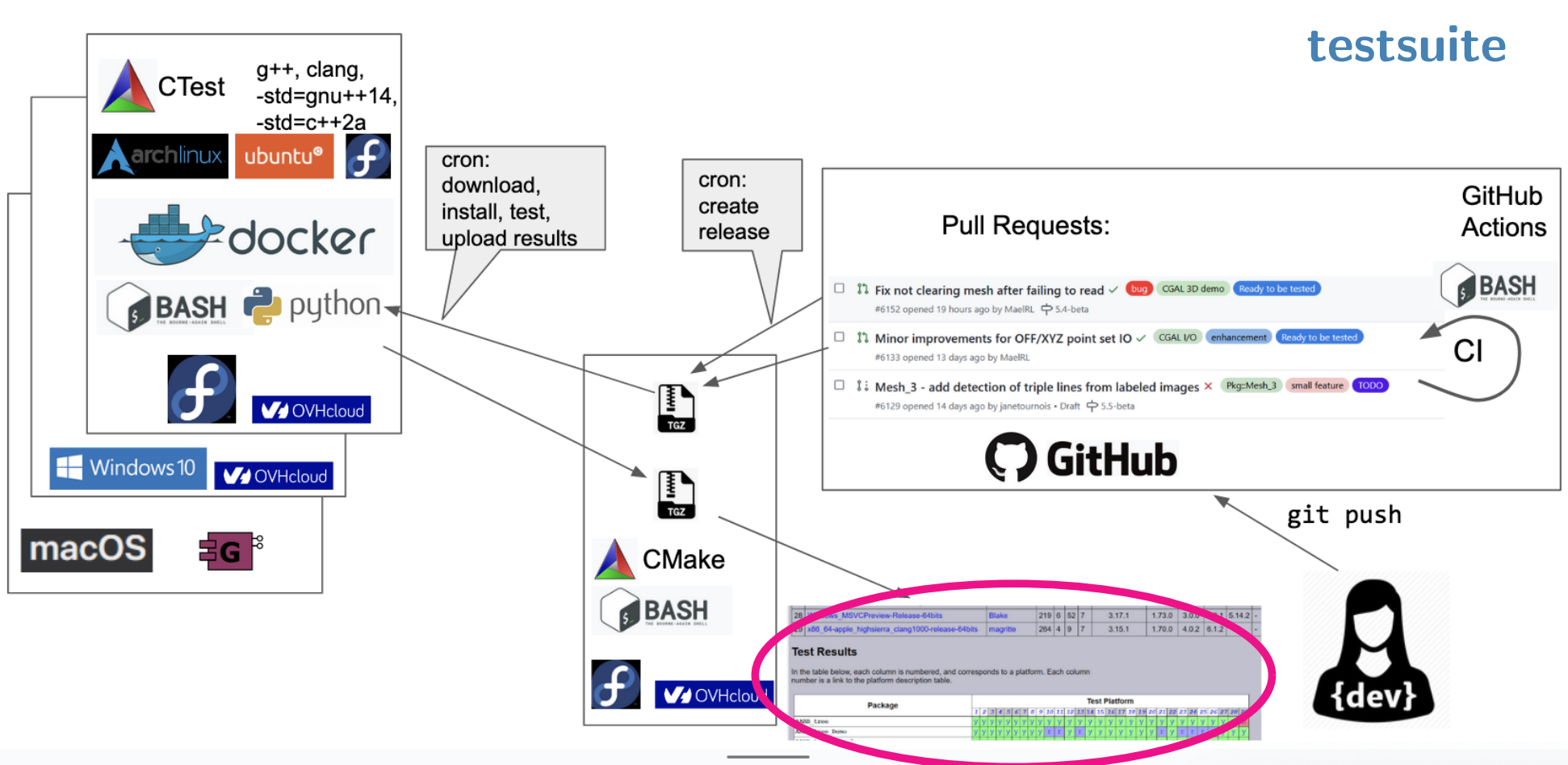


testsuite





Would the CGAL project have survived without GF...?





Andreas Fabri

Laurent Rineau

Sébastien Lorient

Jane Tournois

Mael Rouxel-Labbé

Sven Oesau

*Rao Fu (PhD student in collaboration with INRIA)*



## April fool jokes

### 2008: **The Large Hadron Collider Together With the CGAL Periodic Triangulations Prove the Existence of the Higgs Boson**

The downside of this breakthrough result is that the PhD student who worked on the periodic triangulations stopped working, telling everybody, that whatever he still finds out during his career is irrelevant, as he expects to get the Fields Medal in about 9, and the Nobel Prize for Physics in about 32 years.

*caught several (big) fishes!*



## April fool jokes

### 2008: **The Large Hadron Collider Together With the CGAL Periodic Triangulations Prove the Existence of the Higgs Boson**

The downside of this breakthrough result is that the PhD student who worked on the periodic triangulations stopped working, telling everybody, that whatever he still finds out during his career is irrelevant, as he expects to get the Fields Medal in about 9, and the Nobel Prize for Physics in about 32 years.

*caught several (big) fishes!*

### 2014: Google Summer of Code project **Google Grass**

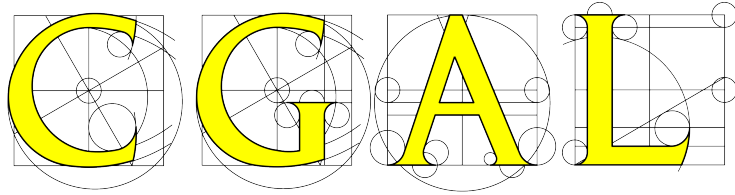
The goal is to find out if a moderate consumption of cannabis makes software developers more creative and more productive.

# Research

# Interplay



# Interplay



Result of research

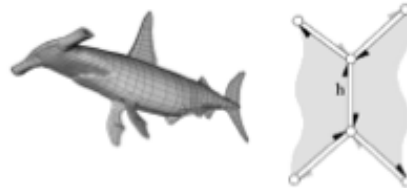
Teams integrate their **favorite** algorithms

# Interplay

## Examples



Bounding Volumes



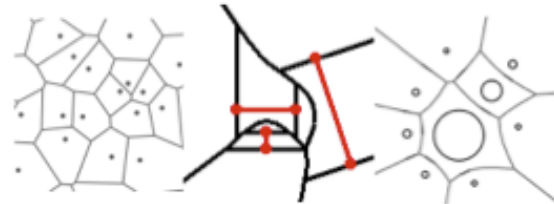
Polyhedral Surface



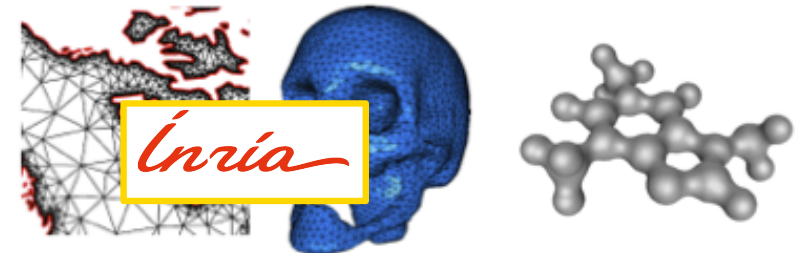
BooleanOperations



Triangulations



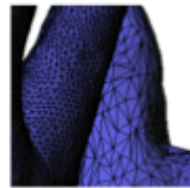
Voronoi Diagrams



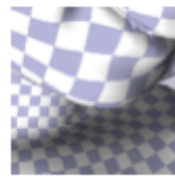
Mesh Generation



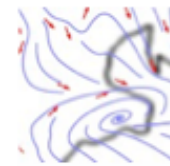
Subdivision



Simplification



Parameterization



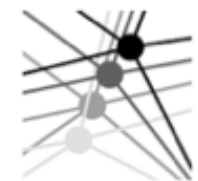
Streamlines



Ridge Detection



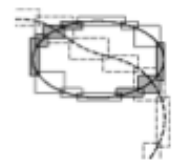
Neighbour Search



Kinetic Data structures



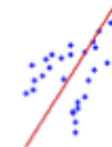
Lower Envelope Arrangement



Intersection Detection



Minkowski Sum



PCA

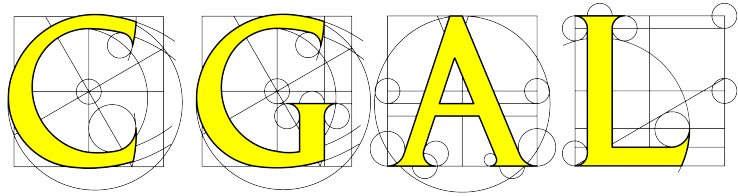


Polytope distance



QP Solver

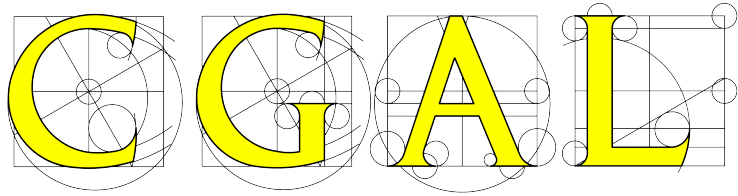
# Interplay



Result of research

Teams integrate their **favorite** algorithms

# Interplay

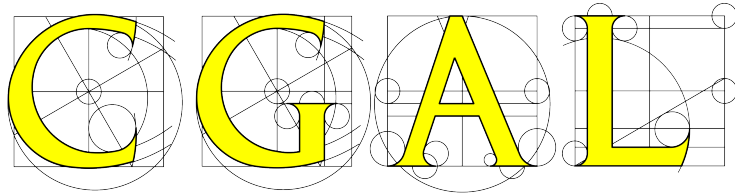


Result of research

Teams integrate their **favorite** algorithms

Helps researchers (in many fields)

# Interplay



Result of research

Teams integrate their **favorite** algorithms

Helps researchers (in many fields)

Raises new research questions

# Software Design

[ A. Fabri, G.-J. Giezeman, L. Kettner, S. Schirra, and S. Schönherr.  
On the design of CGAL a computational geometry algorithms library.  
Software: Practice and Experience 2000 ]

[ S. Hert, M. Hoffmann, L. Kettner, S. Pion, and M. Seel.  
An adaptable and extensible geometry kernel.  
Workshop on Algorithm Engineering, LNCS 2001 ]

# Software Design

[ A. Fabri, G.-J. Giezeman, L. Kettner, S. Schirra, and S. Schönherr.  
On the design of CGAL a computational geometry algorithms library.  
Software: Practice and Experience 2000 ]

[ S. Hert, M. Hoffmann, L. Kettner, S. Pion, and M. Seel.  
An adaptable and extensible geometry kernel.  
Workshop on Algorithm Engineering, LNCS 2001 ]

[ I. Emiris, A. Kakargias, S. Pion, M. Teillaud, and E. Tsigaridas.  
Towards an Open Curved Kernel.  
SoCG 2003 ]

[ P. de Castro, F. Cazals, S. Lorient, and M. Teillaud.  
Design of the CGAL 3D spherical kernel and application to  
arrangements of circles on a sphere.  
CGTA 2009 ]

etc

# Implementation “Details”

*“In theory, there is no difference between theory and practice.  
In practice, there is.”*



# Arithmetic Issues

The real RAM model **does not** exist!

# Arithmetic Issues

The real RAM model **does not** exist!

Exact Geometric Computation paradigm [Yap]

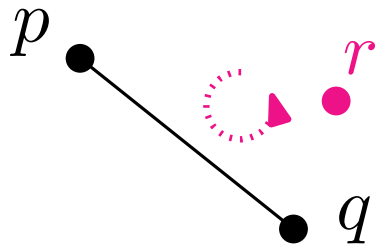
decisions from **exact predicates**

$\neq$  exact arithmetics

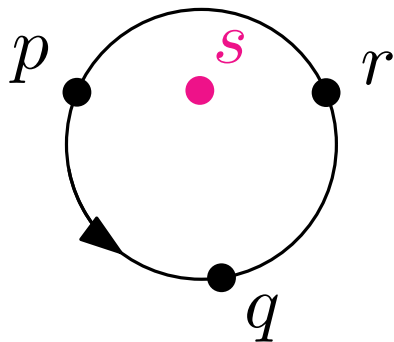
**Filtering**

# Arithmetic Issues

ex: **predicates** for Delaunay triangulations



$$\text{orient}(p, q, r) = \text{sign} \begin{vmatrix} 1 & 1 & 1 \\ p_x & q_x & r_x \\ p_y & q_y & r_y \end{vmatrix}$$



$$\text{in\_disk}(p, q, r, s) = \text{sign} \begin{vmatrix} 1 & x_p & y_p & x_p^2 + y_p^2 \\ 1 & x_q & y_q & x_q^2 + y_q^2 \\ 1 & x_r & y_r & x_r^2 + y_r^2 \\ 1 & x_s & y_s & x_s^2 + y_s^2 \end{vmatrix}$$

*center and radius are **not** computed*

sign of polynomial expression

rational input points

# Arithmetic Issues

Approximate evaluation  $P^a(x)$   
+ Error  $\epsilon$

$$P^a(x) > \epsilon$$



Yes

No

$$\text{Sign}(P(x)) = \text{Sign}(P^a(x))$$

Exact computation

# Arithmetic Issues

Approximate evaluation  $P^a(x)$   
+ Error  $\epsilon$

$$P^a(x) > \epsilon$$



Yes

No

$$\text{Sign}(P(x)) = \text{Sign}(P^a(x))$$

Exact computation

easy cases are more frequent

$\implies$  cost  $\simeq$  cost of approximate (double) computation

# Arithmetic Issues

[ L. Kettner, K. Mehlhorn, S. Pion, S. Schirra, C-K. Yap  
Classroom examples of robustness problems in geometric computations.  
ESA 2004 - CGTA 2008 ]

[ C. Li, S. Pion, C-K. Yap  
Recent progress in exact geometric computation.  
Journal of Logic and Algebraic Programming 2005 ]

etc

# Degenerate cases

*“ We will assume that no subset of  $d + 2$  sites are cospherical or lie on a same hyperplane. ”*

Degenerate cases **do** exist!

# Degenerate cases

*“ We will assume that no subset of  $d + 2$  sites are cospherical or lie on a same hyperplane. ”*

Degenerate cases **do** exist!

Removing a vertex in a 3D Delaunay triangulation problem with co-circular points (in 3D)

symbolic perturbation

only perturbs the `in_sphere` predicate

[ O. Devillers and M. Teillaud.  
Perturbations for Delaunay and weighted Delaunay 3D Triangulations.  
SODA 2003 - CGTA 2011 ]



# Efficiency

## Algorithmic tricks

Olivier Devillers' EuroCG'12 invited talk

<https://members.loria.fr/Olivier.Devillers/EuroCG2012/>

# Users' questions

ex: 3D periodic triangulations, requested by astrophysicists



ESA' 09

generalization: SoCG'11 + DCG'16

CGAL'09 package

+ CGAL'18 3D periodic mesh generation

PhD thesis [Manuel Caroli, 2011]

+ inspiration for further work...

# An issue (not specific to CGAL)

Evaluation of experimental work  
in the CG community

# An issue (not specific to CGAL)

## Evaluation of experimental work in the CG community

*The article is well written. However, it only includes one theoretical result, albeit useful for applications. As a theoretical scientist I cannot accurately assess the importance and impact of the implemented algorithms on the field.*

Anonymous SoCG reviewer

# An issue (not specific to CGAL)

## Evaluation of experimental work in the CG community

Confidence

1. No familiarity (I do not understand the results).

Paper summary and comments for authors

*“This reviewer did not understand the actual result.  
The paper studies flips of triangulations of surfaces.  
[...]*

*It is unclear [...] why it should be interesting to a  
general algorithms audience.*

*I think it’s a clear rejection.”*

Anonymous SODA reviewer

People

# Diversity

gather **many** skills:

mathematics

algorithms

C++

development tools

organization / project management

contacts with application fields

etc

# Visibility

*“Reward structure for implementations in academia”*

## Past and Current Contributors

Web page

Dmitry Anisimov  
Ken Arroyo Ohori  
Dror Atariah  
Alon Baram  
Matthias Bäsken  
Eric Berberich  
Mikhail Bogdanov  
David Bommes  
Mario Botsch  
Sylvain Brandel  
Yves Brise  
Hervé Brönnimann  
Francisc Bungiu  
Fernando Cacciola  
Manuel Caroli  
Pedro de Castro  
Frédéric Cazals  
Raphaëlle Chainé  
Panagiotis Cheilaris  
David Cohen-Steiner  
Éric Colin de Verdière  
Keenan Crane  
Tran Kai Frank Da  
Christophe Delage  
Olivier Devillers  
Katrín Dobrindt  
Pavel Emeliyanenko  
Ester Ezra  
Kaspar Fischer  
Eyal Flato

Julia Flötotto  
Wolfgang Freiseisen  
Ben Galehouse  
Xiang Gao  
Bernd Gärtner (former editor)  
Geert-Jan Giezeman  
Maxime Gimeno  
Simon Giraudot  
Marc Glisse  
Nir Goren  
Gaël Guennebaud  
Philippe Guigue  
Ankit Gupta  
Peter Hachenberger  
Iddo Hanneil  
Idit Haran  
Sariel Har-Peled  
Michael Hemmer (former editor)  
Ross Hemsley  
Thomas Herrmann  
Susan Hert (former editor)  
Shai Hirsch  
Thien Hoang  
Kai Hormann  
Samuel Hornus  
Kan Huang  
Iordan Iordanov  
Clément Jamin  
Menelaos Karavelas (former editor)  
Konstantinos Katrioplas

Waqar Khan  
Michael Kerber  
Lutz Kettner (former editor)  
Stephen Kiazky  
Michal Kleinbort  
Alexander Kobel  
Nico Kruithof  
Sandeep Kumar Dey  
Florent Lafarge  
Sylvain Lazard  
Eran Leiserowitz  
Bruno Lévy  
Sebastian Limbach  
Eugene Lipovetsky  
Naama Mayer  
Abdelkrim Mebarki  
Quentin Mérigot  
Naceur Meskini  
Andreas Meyer  
Jocelyn Meyron  
Michal Meyerovitch  
Philipp Moeller  
Sebastian Morr  
Liangliang Nan  
Oren Nechushtan  
Gabriele Neyer  
Sven Oesau  
Steve Odot  
Markus Overthel  
Eli Packer

Evanthia Papadopoulou  
Frédéric Paradis  
Dmitrii Pasechnik  
Aymeric Pellé  
Luis Peñaranda  
Sylvain Pion (former editor)  
Roza Pogalnikova  
Asaf Porat  
Sigal Raab  
François Rebufat  
Joachim Reichel  
Mael Rouxel-Labbé  
Daniel Russel  
Niv Sabath  
Laurent Saboret  
Nader Salman  
Oren Salzman  
Stefan Schirra  
Sven Schönherr  
Michael Seel  
Ophir Setter  
Shahar Shamai  
Le-Jeng Shiue  
Weisheng Si  
Daniel Sieger  
Johannes Singler  
Olga Sorkine-Hornung  
Andrea Tagliasacchi  
Hans Tangelder  
Stéphane Tayeb

Remy Thomasse  
Alexandru Tifrea  
Jane Tournois  
Quincy Tse  
Alex Tsui  
George Tzoumas  
Radu Ursu  
Carl Van Geem  
Thijs Van Lankveld  
Amir Vaxman  
Christina Vaz  
Remco Veltkamp (former editor)  
Yannick Verdie  
Aurélien Vialon  
Ivo Vigan  
Ron Wein (former editor)  
Wieger Wesselink  
Frans Wessendorp  
Patrick Wenzlaff  
Camille Wormser  
Shihao Wu  
Ning Xu  
Yin Xu  
Ilker O. Yaz  
Marianne Yvinec (former editor)  
Lingjie Zhu  
Afra Zomorodian  
Guy Zucker  
Baruch Zukerman  
Tali Zvi



# Visibility

*“Reward structure for implementations in academia”*

Package Authors

Manual

## CGAL 5.0.1 - 3D Periodic Triangulations

CGAL 5.0.1 - 3D Periodic Triangulations

- ▶ User Manual
- ▶ Reference Manual
- Refinement Relationships
- Is Model Relationships
- Has Model Relationships
- Bibliography
- ▶ Class and Concept List
- ▶ Examples

### User Manual

**Authors**  
Manuel Caroli, Aymeric Pellé, Mael Rouxel-Labbé and Monique Teillaud

The image shows a 3D visualization of a periodic triangulation. It consists of a grid of points connected by lines, forming a complex, interconnected structure. A yellow sphere is visible at the center of the grid, and the overall structure is rendered in a light gray color.

# Visibility

*“Reward structure for implementations in academia”*

CGAL developers appreciated by

companies (Google, Intel, Apple, ...)



research institutes (INRIA, ETHZ, ...?)

# Visibility

*“Reward structure for implementations in academia”*

Personal experience

Visibility **outside** computational geometry

*“ Are you the Monique from CGAL ? ”*

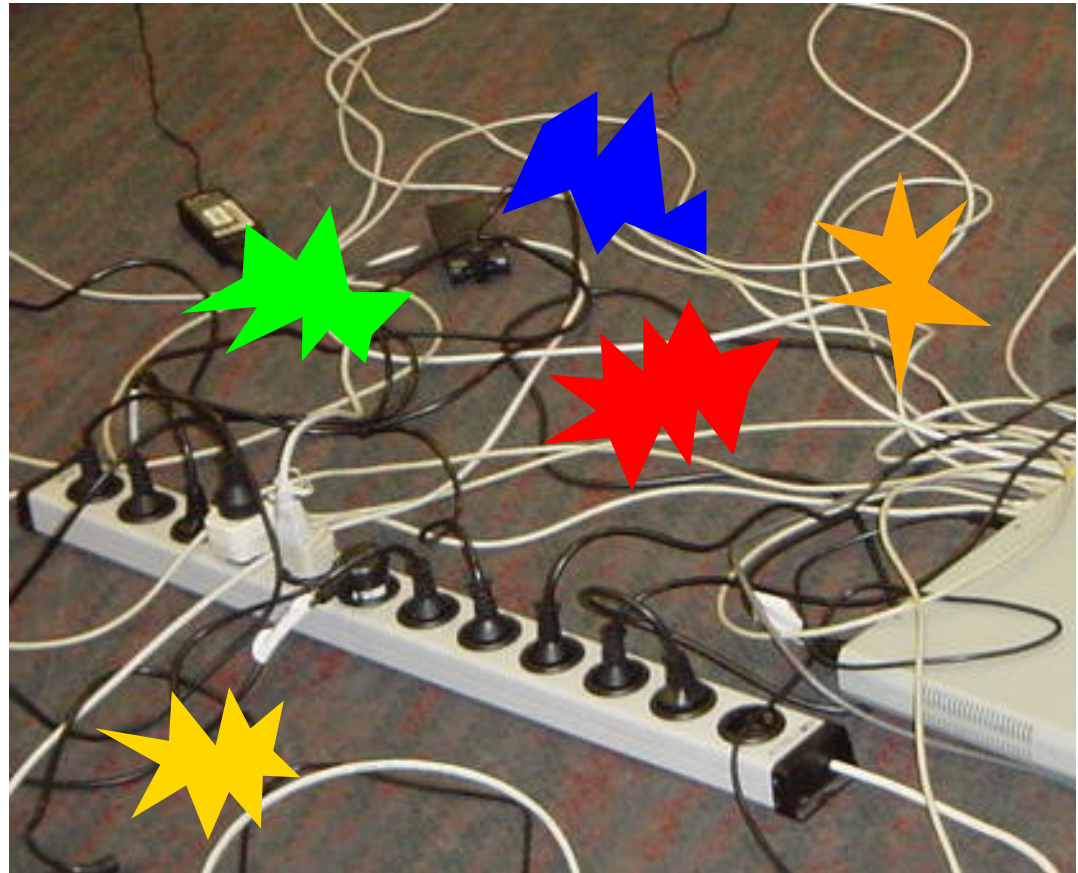
My most cited paper: “Triangulations in CGAL”

# Developer Meetings

1 or 2 / year

~ 10-20 participants = developers + guests

Discussions  
/ decisions



Dagstuhl, Sophia Antipolis, Zurich, Tel Aviv,  
Nancy, Heraklion, Lyon, Paris, Utrecht

# Developer Meetings

1 or 2 / year

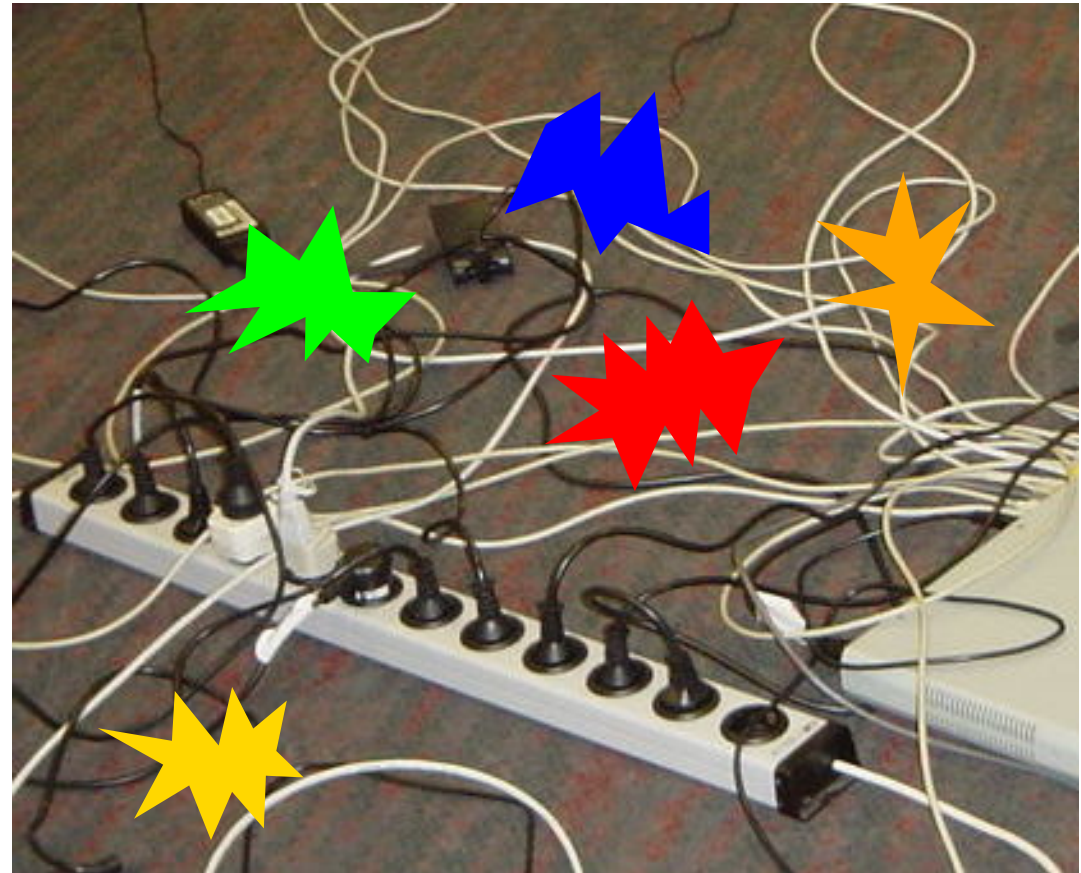
~ 10-20 participants = developers + guests

Discussions  
/ decisions

Integration

technical help

sense of belonging



Dagstuhl, Sophia Antipolis, Zurich, Tel Aviv,  
Nancy, Heraklion, Lyon, Paris, Utrecht

# Developer Meetings

pictures deleted  
for HAL version

despair?

a bug?

a negative review?

*playing Werewolves*

*in the wine cellar*

*in Dagstuhl*



pictures deleted  
for HAL version

*48<sup>th</sup> developer meeting: 2023 April 24-28, Sophia Antipolis*

picture deleted  
for HAL version



# Ingredients - Summary

Will + Goals

Technical choices

Project Management / Organization / Rules

Infrastructure

GeometryFactory

Research

People

# Ingredients - Summary

Will + Goals

Technical choices

Project Management / Organization / Rules

Infrastructure

GeometryFactory

Research

People

**THANK YOU**