How to bring industry standards to your research software development

An incomplete, biased, and opinionated story

Jean-Claude Passy

ZWE Software Workshop

Future Opportunities for Software in Research 2022

MAX PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

Outline

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Outline

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Internal Software Engineers

- Each department/group has their own software engineers with **domain knowledge**: computer graphics, computer vision, deep learning, micro-controllers...
- Often not sufficient to build proper software as other criteria need to be met: standardization, reproducibility (results **and** code), modular, extensible, robust, verified, validated...
- Need someone with a scientific background and domain knowledge in software engineering, computer science, algorithmics, parallelization. . .

$\Longrightarrow$ Research **S**oftware **E**ngineer

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Software Workshop

The **Software Workshop** is an independent facility composed of RSEs, at the crossroads between **research**, **software engineering**, and **support**.
Typical profile:

- Scientific background (must have)
- General knowledge in sofware engineering (must have)
- Experience in the industry (nice to have)

$\Longrightarrow$ Hiring is difficult!

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

What we do

Three main tasks:

- Knowledge dissemination: trainings, workshops, code reviews, mentoring, wiki
- Maintenance (with IT) of our infrastructure for software development
- Projects: prototype to software, refactoring, optimization, collaboration, independent

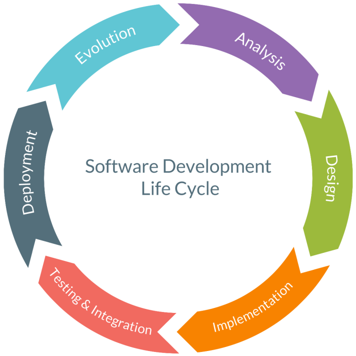
For any of these tasks, we try to follow and promote good practices and industry standards in terms of:

- workflow
- techniques
- infrastructure

Outline

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

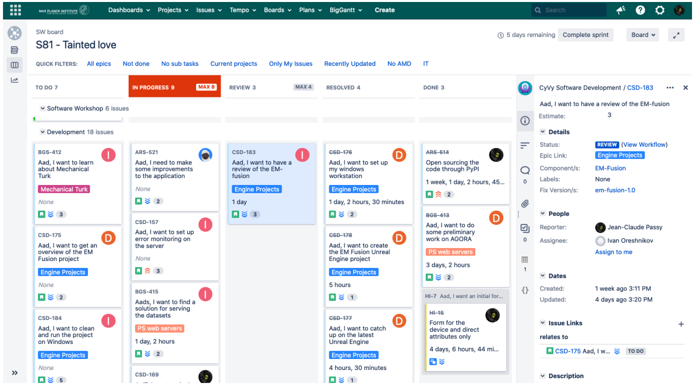
Software Development Life Cycle



Industrial process followed for the development of a software product.

Work management

It is essential to organize your work: productivity, reproducibility, who/what/when...
There are many project management tools out there, starting from your white board!
We recommend using Jira.



SW board
S81 - Tainted love
5 days remaining Complete sprint Board

QUICK FILTERS: All epics Not done No sub tasks Current projects Only My Issues Recently Updated No AMD IT

TO DO 7 | IN PROGRESS 9 MAX 8 | REVIEW 3 MAX 4 | RESOLVED 4 | DONE 3

Software Workshop 6 issues

Development 18 issues

BGS-412 Aad, I want to learn about Mechanical Turk — Mechanical Turk — None — 3
CSD-175 Aad, I want to get an overview of the EM Fusion project — Engine Projects — None — 2
CSD-184 Aad, I want to clean and run the project on Windows — Engine Projects — None — 5

ARS-521 Aad, I need to make some improvements to the application — None — 2
CSD-157 Aad, I want to set up error monitoring on the server — None — 3
BGS-415 Aads, I want to find a solution for serving the datasets — PS web servers — 1 day, 2 hours — 2
CSD-169

CSD-183 Aad, I want to have a review of the EM-fusion — Engine Projects — 1 day — 3

~~CSD-176~~ Aad, I want to set up my windows workstation — Engine Projects — 1 day, 2 hours, 30 minutes — 2
~~CSD-178~~ Aad, I want to create the EM Fusion Unreal Engine project — Engine Projects — 5 hours — 1
~~CSD-177~~ Aad, I want to catch up on the latest Unreal Engine — Engine Projects — 4 hours, 30 minutes — 1

~~ARS-514~~ Open sourcing the code through PyPI — 1 week, 1 day, 2 hours, 45... — 2
~~BGS-413~~ Aad, I want to do some preliminary work on AGORA — PS web servers — 3 days, 2 hours — 2
HI-7 Aad, I want an initial for...
~~HI-15~~ Form for the device and direct attributes only — 4 days, 6 hours, 44 mi...

CyVy Software Development / CSD-183
Aad, I want to have a review of the EM-fusion
Estimate: 3

Details
Status: REVIEW (View Workflow)
Epic Link: Engine Projects
Component/s: EM-Fusion
Labels: None
Fix Version/s: em-fusion-1.0

People
Reporter: Jean-Claude Passy
Assignee: Ivan Oreshnikov
Assign to me

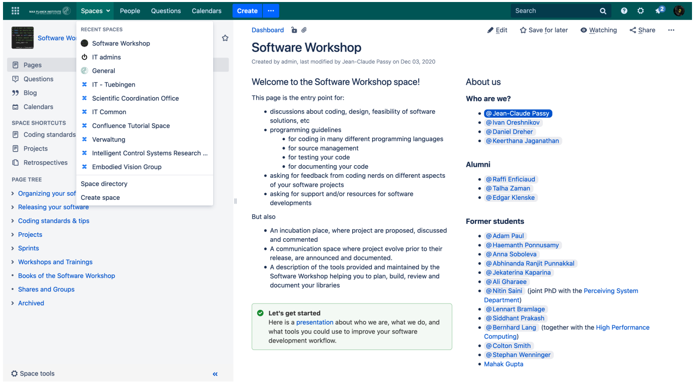
Dates
Created: 1 week ago 3:11 PM
Updated: 4 days ago 3:20 PM

Issue Links
relates to
CSD-175 Aad, I w... TO DO

Description

Sharing knowledge

The **know-how** gathered in a research institute **extremely** valuable asset.
It must be recorded and shared, and will constantly evolve.
For this, we set up our internal wiki with Confluence.



Software Workshop

Created by admin, last modified by Jean-Claude Passy on Dec 03, 2020

Welcome to the Software Workshop space!

This page is the entry point for:

- discussions about coding, design, feasibility of software solutions, etc
- programming guidelines
  - for coding in many different programming languages
  - for source management
  - for testing your code
  - for documenting your code
- asking for feedback from coding nerds on different aspects of your software projects
- asking for support and/or resources for software developments

But also

- An incubation place, where project are proposed, discussed and commented
- A communication space where project evolve prior to their release, are announced and documented.
- A description of the tools provided and maintained by the Software Workshop helping you to plan, build, review and document your libraries

Let's get started
Here is a presentation about who we are, what we do, and what tools you could use to improve your software development workflow.

About us

Who are we?

- @Jean-Claude Passy
- @Ivan Oreshnikov
- @Daniel Dreher
- @Keerthana Jaganathan

Alumni

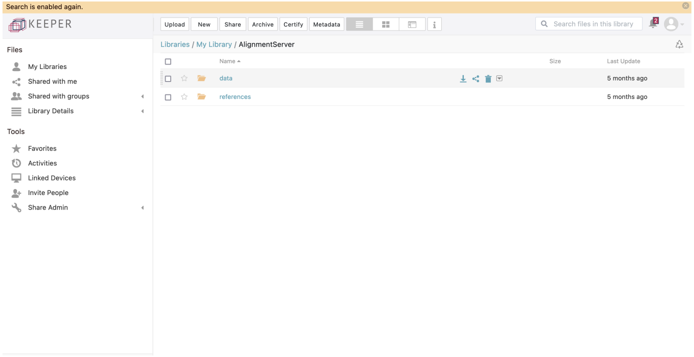
- @Raffi Enficiaud
- @Talha Zaman
- @Edgar Klenske

Former students

- @Adam Paul
- @Haemanth Ponnusamy
- @Anna Soboleva
- @Abhinanda Ranjit Punnakkal
- @Jekaterina Kaparina
- @Ali Gharaee
- @Nitin Saini (joint PhD with the Perceiving System Department)
- @Lennart Bramlage
- @Siddhant Prakash
- @Bernhard Lang (together with the High Performance Computing)
- @Colton Smith
- @Stephan Wenninger
- Mahak Gupta

Sharing data

For several years we have been using Nextcloud to share large data.
Recently, we moved to Keeper from the MPDL.


Search is enabled again.

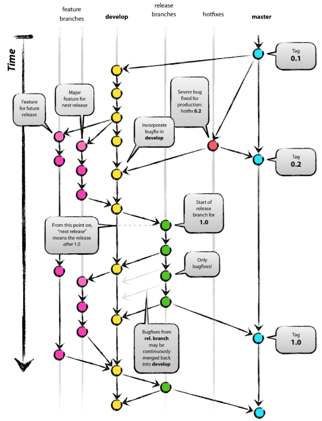
Libraries / My Library / AlignmentServer

Name	Size	Last Update
data		5 months ago
references		5 months ago

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Version control

Version control is the process of managing and organizing information changes. It is done using **V**ersion **C**ontrol **S**ystems (**VCS**). The most popular (and only option) is git.


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Severe bug fixed for production: hotfix 0.2
Major feature for next release
Feature for future release
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

Advantages:

- complete code base is stored on everyone's computer
- work collaboratively
- work simultaneously on several files
- work simultaneously on several tasks
- traceability
- rollback
- reproducibility

Tools for writing code

Some tools can help you to save time and improve the quality of your code.

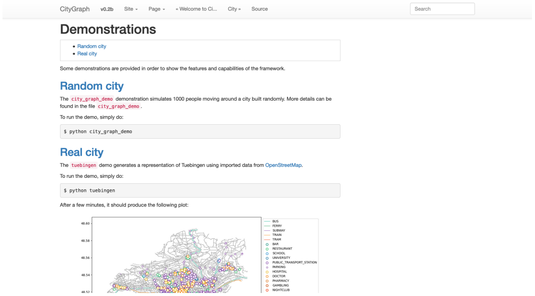
- **git clients**: SourceTree (macOS/Win), GitKraken
- **IDEs**: VS Code, QtCreator (C++), PyCharm (Python), Eclipse, XCode (macOS), Visual Studio (Win)
- **Style Guide**: PEP8 (Python), Google Style (C++)
- **Auto-formatters** (style): autopep8, black (Python), clang-format (C++)
- **Linters** (static code analysis): pylint, flake8 (Python)

MAX PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

Documentation

Writing documentation is **THE** most important stage in software development. Without it, your code will not be used and become obsolete. It is also a way to manage expectations and transfer knowledge.

- README: this is the **bare minimum**
- Proper documentation: Sphinx (Python), Doxygen (C++)



Demonstrations

- Random city
- Real city

Some demonstrations are provided in order to show the features and capabilities of the framework.

Random city

The `city_graph_demo` demonstration simulates 1000 people moving around a city built randomly. More details can be found in the file `city_graph_demo`.

To run the demo, simply do:

```
$ python city_graph_demo
```

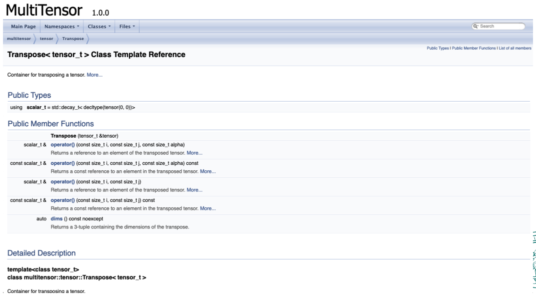
Real city

The `tuebingen` demo generates a representation of Tuebingen using imported data from OpenStreetMap.

To run the demo, simply do:

```
$ python tuebingen
```

After a few minutes, it should produce the following plot:



MultiTensor 1.0.0

Transpose< tensor_t > Class Template Reference

Container for transposing a tensor. More...

Public Types

using **scalar_t** = std::decay_t< decltype(tensor(0, 0))>

Public Member Functions

	Transpose (tensor_t &tensor)
scalar_t &	operator() (const size_t i, const size_t j, const size_t alpha) Returns a reference to an element of the transposed tensor. More...
const scalar_t &	operator() (const size_t i, const size_t j, const size_t alpha) const Returns a const reference to an element in the transposed tensor. More...
scalar_t &	operator() (const size_t i, const size_t j) Returns a reference to an element of the transposed tensor. More...
const scalar_t &	operator() (const size_t i, const size_t j) const Returns a const reference to an element in the transposed tensor. More...
auto	dims () const noexcept Returns a 3-tuple containing the dimensions of the transpose.

Detailed Description

template<class tensor_t>
class multitensor::tensor::Transpose< tensor_t >

Container for transposing a tensor.

Testing

Standard tools to use

- Python: unittest, nose, pytest
- C++: Boost Test, Google Test
- Web applications: Selenium

How do you know you need to write test?

- Code coverage (e.g. coverage.py)
- Find bugs/unexpected usage (increase test coverage)

Last piece of advice:

- Start writing tests very early on (almost TDD)
- Benchmarks are very useful for projects you are taking over

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Structure, packaging, and release

A good repository structure helps understanding the code and finding information more efficiently.

Packaging your code eases sharing, installation, and deployment:

- Python: pip
- C++: CMake, Makefile

Release your code often: 1.1, 1.1.1, 1.2b

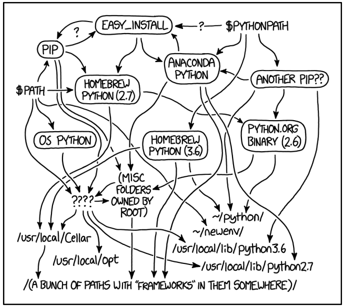
```
project
├── doc
├── package
│   ├── __init__.py
│   ├── version.py
│   └── ...
├── LICENSE.md
├── README.md
├── requirements.txt
├── setup.py
└── tests
```

MAX PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

Environment isolation

For reproducibility, easier maintenance, and large scale delivery, it is often a good idea to isolate your development and production environments.

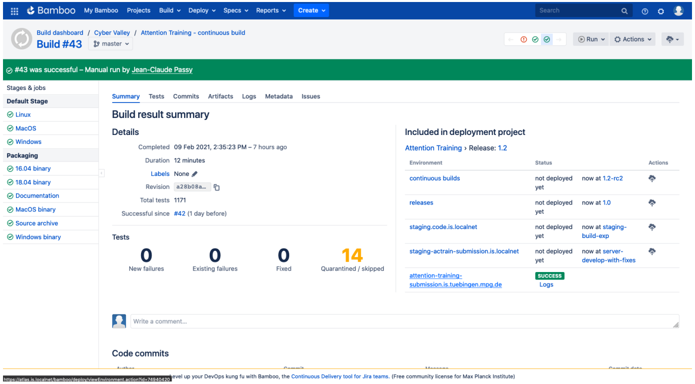
- Python: Virtual Environments
- Containers: Docker, Singularity
- Virtual Machines: VMWare, VirtualBox



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

Continuous Integration

CI is the process of integrating and validating changes to source code frequently and during code development. An important component is **continuous testing**. There are many options (GitLab CI, Jenkins, CircleCI, Pipeline,...) and we chose Bamboo.



Build dashboard / Cyber Valley / Attention Training - continuous build
Build #43
master

#43 was successful – Manual run by Jean-Claude Passy

Stages & jobs
Default Stage
- Linux
- MacOS
- Windows

Packaging
- 16.04 binary
- 18.04 binary
- Documentation
- MacOS binary
- Source archive
- Windows binary

Summary | Tests | Commits | Artifacts | Logs | Metadata | Issues

Build result summary

Details

Completed	09 Feb 2021, 2:35:23 PM – 7 hours ago
Duration	12 minutes
Labels	None
Revision	a28b08a...
Total tests	1171
Successful since	#42 (1 day before)

Tests

New failures	Existing failures	Fixed	Quarantined / skipped
0	0	0	14

Included in deployment project

Attention Training › Release: 1.2

Environment	Status		Actions
continuous builds	not deployed yet	now at 1.2-rc2	
releases	not deployed yet	now at 1.0	
staging.code.is.localnet	not deployed yet	now at staging-build-exp	
staging-actrain-submission.is.localnet	not deployed yet	now at server-develop-with-fixes	
attention-training-submission.is.tuebingen.mpg.de	SUCCESS Logs		

Write a comment...

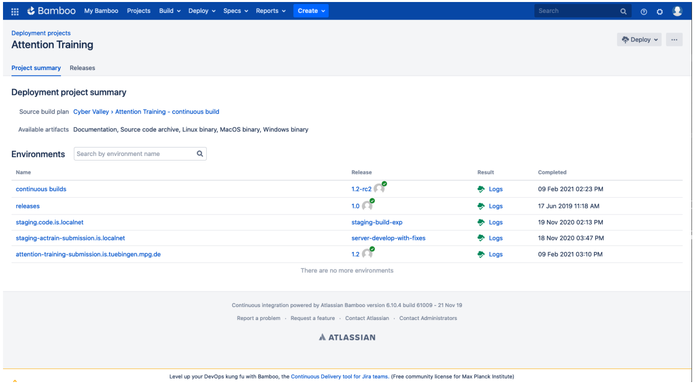
Code commits

Author | Commit | Message | Commit date

https://atlas.is.localnet/bamboo/deploy/viewEnvironment.action?id=74940420 evel up your DevOps kung fu with Bamboo, the Continuous Delivery tool for Jira teams. (Free community license for Max Planck Institute)

Continuous Delivery/Deployment

CD is the process of delivering and/or deploying applications to production environment frequently. This step is strongly linked with CI, and is also done with Bamboo and Ansible, a tool for configuration management and automation.



Deployment projects
Attention Training

Deployment project summary

Source build plan Cyber Valley › Attention Training - continuous build

Available artifacts Documentation, Source code archive, Linux binary, MacOS binary, Windows binary

Environments

Name	Release	Result	Completed
continuous builds	1.2-rc2	Logs	09 Feb 2021 02:23 PM
releases	1.0	Logs	17 Jun 2019 11:18 AM
staging.code.is.localnet	staging-build-exp	Logs	19 Nov 2020 02:13 PM
staging-actrain-submission.is.localnet	server-develop-with-fixes	Logs	18 Nov 2020 03:47 PM
attention-training-submission.is.tuebingen.mpg.de	1.2	Logs	09 Feb 2021 03:10 PM

There are no more environments

Continuous integration powered by Atlassian Bamboo version 6.10.4 build 61009 - 21 Nov 19

Report a problem · Request a feature · Contact Atlassian · Contact Administrators

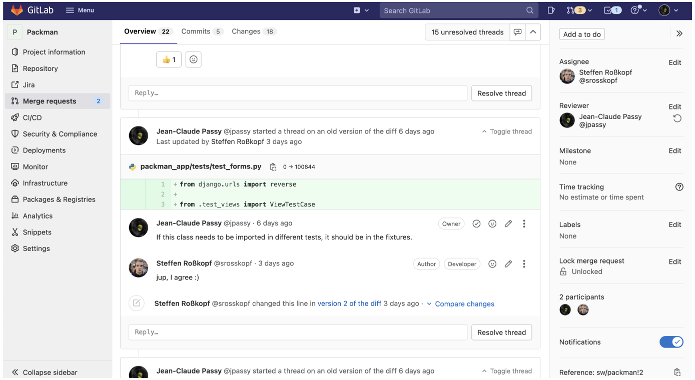
ATLASSIAN

Level up your DevOps kung fu with Bamboo, the Continuous Delivery tool for Jira teams. (Free community license for Max Planck Institute)

PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

Reviews

Code designs and implementations should almost always be reviewed by people other than the developers: catching bugs, improved quality, learning... There are many options integrated to hosting services and PRs, we use GitLab MR.



Jean-Claude Passy @jpassy started a thread on an old version of the diff 6 days ago
Last updated by Steffen Roßkopf 3 days ago

packman_app/tests/test_forms.py 0 → 100644

```
+ from django.urls import reverse
+
+ from .test_views import ViewTestCase
```

Jean-Claude Passy @jpassy · 6 days ago
If this class needs to be imported in different tests, it should be in the fixtures.

Steffen Roßkopf @srosskopf · 3 days ago
jup, I agree :)

Steffen Roßkopf @srosskopf changed this line in version 2 of the diff 3 days ago ·

Jean-Claude Passy @jpassy started a thread on an old version of the diff 6 days ago

Assignee
Steffen Roßkopf
@srosskopf

Reviewer
Jean-Claude Passy
@jpassy

Milestone
None

Time tracking
No estimate or time spent

Labels
None

Lock merge request
Unlocked

2 participants

Notifications

Reference: sw/packman!2

PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

Outline

MAX PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

A message of hope

- We know how important software development is for research
- We wish everyone would agree with us
- We wish we would be taken more seriously

Situation has already largely improved over the last 10/15 years

- We have a name
- We are allowed to meet
- We are allowed to publish (JOSS)
- Mentalities are changing

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

A message of hope

- We know how important software development is for research
- We wish everyone would agree with us
- We wish we would be taken more seriously

Situation has already largely improved over the last 10/15 years

- We have a name
- We are allowed to meet
- We are allowed to publish (JOSS)
- Mentalities are changing

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

A message of hope

- We know how important software development is for research
- We wish everyone would agree with us
- We wish we would be taken more seriously

Situation has already largely improved over the last 10/15 years

- We have a name
- We are allowed to meet
- We are allowed to publish (JOSS)
- Mentalities are changing

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

A message of hope

- We know how important software development is for research
- We wish everyone would agree with us
- We wish we would be taken more seriously

Situation has already largely improved over the last 10/15 years

- We have a name
- We are allowed to meet
- We are allowed to publish (JOSS)
- Mentalities are changing

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

A message of hope

- We know how important software development is for research
- We wish everyone would agree with us
- We wish we would be taken more seriously

Situation has already largely improved over the last 10/15 years

- We have a name
- We are allowed to meet
- We are allowed to publish (JOSS)
- Mentalities are changing