ENABLING SHARED SOLUTIONS FOR EFFECTIVE AND EFFICIENT DIGITAL PRESERVATION

OPF SOFTWARE MATURITY MODEL

- Open licensing
- Quality
- Visibility
- Community integration
- Training

DEVELOPMENT LIFECYCLE

Funding
- OPF membership fees
- Project funding

Planning (Product Board)
- Prioritises features for release
- Defines the release
- Manages the roadmap

Requirements & community feedback
- Bug reports
- New feature requests
- Hack day activities
- Code contributions (pull requests)

Development & testing
- GitHub for open source development
- Jenkins and Travis for public continuous integration
- Codacy for static code quality assurance
- Priority testing with members

Build and test iteration
- Create test data to develop features and expose bugs
- Iterative development against test data

Development release (odd numbers)
- Build a set of test data (test corpus)
- Jenkins and Travis for public continuous integration
- Codacy for static code quality assurance
- Priority testing with members

Final test & release
- Production release (even numbers)
- Freely available to community
- Patches (essential fixes)

HOSTING
- fido

MAINTENANCE
- jpylyzer

SUSTAINABILITY

ADOPITION TIMELINE

- 2010
- 2014
- 2015
- 2017

HOW CAN YOU HELP?
- Join the Open Preservation Foundation
- Contribute to improving the software or documentation
- Make a donation

www.openpreservation.org