1. Introduction
   1. Propose to continue development of pychron
   2. Start a community based effort to create robust and sustainable software tools
   3. Pychron fully featured data acquisition and processing package analogous to Mass Spec
      1. Operated at NMGRL since 2012.
         1. Collected ca. 40,000 analyses in that time
      2. Distributed control paradigm
      3. Data reduction implemented starting end of 2013
         1. Validated during spring of 2014
         2. Distributed internally during the summer of 2014
      4. Interfaces with various hardware components
         1. Photon Machines Fusions CO2/Diode
         2. Custom Photon Machines UV system
         3. New generation Thermo Scientific mass spectrometers using Qtegra via RemoteControlServer.cs
            1. Developed at NMT
            2. Distributed around the world
            3. Packaged with Mass Spec for all new Thermo installs
      5. Controllable by third party software via a TCP/UDP interface
2. Project Summary
   1. Software integral to Ar/Ar and analytical labs in general
   2. Current model of isolated development makes collaboration and sharing of data difficult, inefficient and error prone
   3. No validation of the computations both internally among different versions and external among different applications
3. Software integral to Ar/Ar
   1. Data acquisition
      1. Automation
      2. Hardware interface
   2. Data processing
      1. DB
         1. Management of large datasets
      2. Calculations
         1. Efficient
         2. Consistent
         3. Fast
      3. Visualization
         1. Publication ready tables and figures
4. Current Problem
   1. Current products becoming out dated
      1. Bloated and difficult to modify
         1. Difficult to add functionality
      2. Not platform for future expansion
   2. Minor community collaboration
   3. Difficult to share analytical data
      1. Cannot efficiently share data among different programs
      2. No networking mechanism to seamlessly transfer data
   4. No systematic software validation
   5. Laboratories beholden to few developers
   6. Computations a black box
   7. Inefficiencies in handling legacy data?
   8. Lack of documentation/training materials
      1. Documentation that does exist is static and outdated
5. Solution
   1. Provide a platform for future development
      1. Sustainable
      2. Distributed but centralized
   2. Start a community based development process
      1. Centralized control by NMT
         1. Jake ross benevolent dictator
            1. Helps define vision and future development
      2. Individual labs free and encouraged to develop on both main and local branches/forks
6. Mass Spec
   1. advanced Ar-Ar technique
      1. Used around the world
      2. Streamlined data acquisition and processing
      3. Ability to control large number of hardware components.
   2. Not platform for future
      1. Single developer
      2. “Closed” source
      3. Monolithic
      4. Proprietary platform
      5. Difficult/inefficient to update
         1. No collaboration tools
            1. Changes need to be manually merged/incorporated
            2. No branching mechanism
            3. Version control handled in namespace

Inefficient

Error prone

* + - 1. No testing
         1. Unit
         2. Regression (do updates break previously working functionalities)
    1. No validation
       1. No systematic comparison between other software
    2. No data interoperability
       1. Difficult to import/export data to other formats
    3. No extensive documentation
       1. Legacy PDF user guide
       2. No API or developer guide
    4. No efficient distribution mechanism
    5. Poor handling of legacy data

1. Pychron
   1. Platform for future development
      1. Open source
         1. Transparent isotopic calculations
      2. Written in popular language- python
         1. #1 teaching language at top universities
         2. Widely used in scientific community.
         3. Gaining popularity
         4. Current version 2.7 supported until 2020
            1. New version 3.0 gaining acceptance and popularity
         5. Highly functional
            1. Web programming

Numerous popular frameworks

* + - * 1. Desktop apps

High quality GUI toolkits

* + - * 1. Scripting
        2. “Batteries included”
        3. Thousands of third party packages

Numpy

Scipy

Uncertainties

Sqlalchemy

* + 1. Effective/efficient collaboration and version control via GIT
       1. Hosted at github
       2. Issue tracking
       3. Repository metrics
          1. Traffic
          2. Contributions
       4. Website hosting
       5. Convenient location for discussion
    2. Online documentation
       1. Readthedocs.org
          1. Continuously build

Always up to date

* + - * 1. Exportable as pdf/epub
    1. Unit testing
       1. Validates calculations
       2. Regression testing
       3. Systematic comparison with other software (Mass Spec)

1. Goals
   1. Finalize pychron for use at NMT
      1. Reduce bugs
         1. Usability
         2. Functional
      2. Increase performance
         1. Speed up computations
         2. Streamline user interface/interaction
   2. Setup support infrastructure
      1. Website
         1. Tutorials
         2. Wiki
      2. Documentation (living docs)
         1. User guide
         2. Developer guide
         3. API
   3. Ease software interoperability and sharing of analytical data
   4. Systematically validate with other software
      1. Mass Spec, ArArCalc, TopSoil/Isoplot
      2. Move towards standardized data format for sharing
         1. Import/export xml(?) file
   5. Proof of concept distribution to another laboratory
      1. Setup model for collaboration
         1. Where possible laboratory modifies/extends pychron for the “last mile” of integration
            1. Develop hardware drivers
            2. Writes their own extraction/measurement scripts
      2. Streamline installation
2. Broader impacts
   1. Mentor student
      1. Provide opportunities to CS/Earth science students interested in programming
   2. Talk with primary/secondary/high school students about programming
      1. Benefits as a skill
      2. Joys of problem solving and creating things
         1. Technical but creative
   3. Techniques and principles applicable to other fields
      1. Pychron can be adapted to other isotopic systems