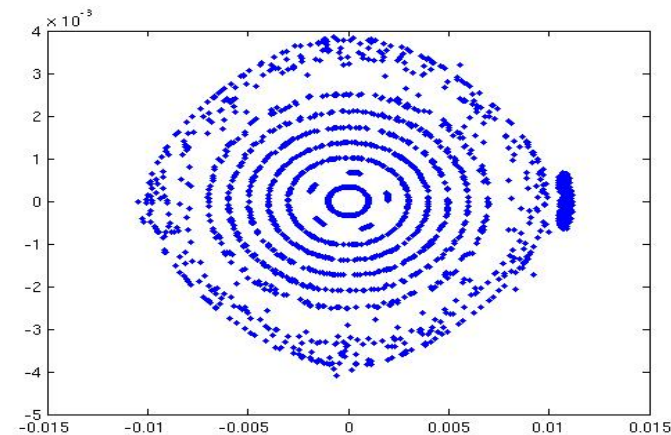
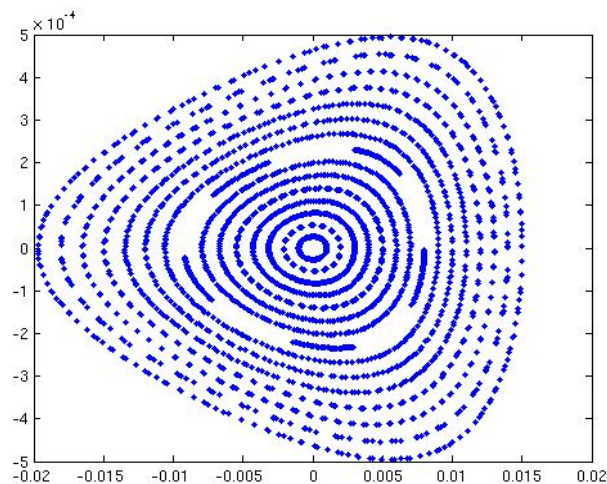




Accelerator Toolbox Collaboration Meeting

B. Nash



Outline

- Some AT history
 - AT 1.2, 1.3
 - AT at ESRF
 - atcollab
 - AT 1.4 and development principles
- A few ideas on documentation and code testing
- Roadmap for AT 2.0

AT overview/history

Accelerator Toolbox is one of the most widely used codes for linear optics, tracking, and lattice design, etc.

Created by Andre Terebilo at SLAC in late 1990's in collaboration with colleagues at LBL. It was needed as an online simulator for SPEAR3 commissioning. It was first tested at ALS.

4th order symplectic integrator originally based on code from Tracy ML

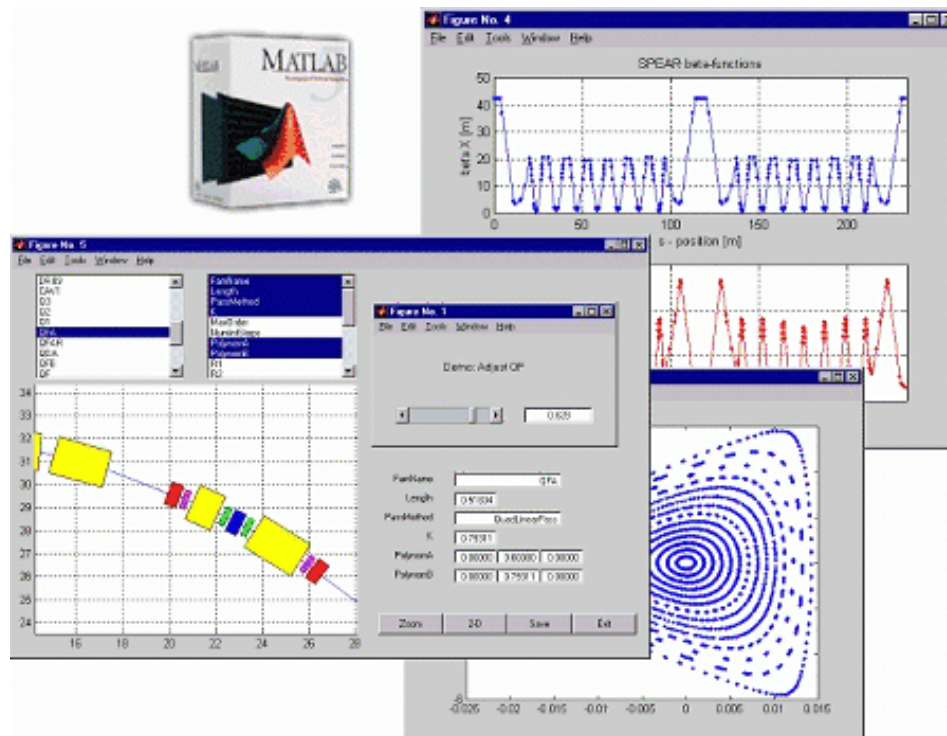
C code is compiled for use with Matlab interface, providing all of Matlab functionality

AT - Accelerator Toolbox

Andrei Terebilo

MATLAB[®] Toolbox for Particle Accelerator Modeling

Accelerator Toolbox is a collection of tools to model particle accelerators and beam transport lines in MATLAB environment. It is being developed by [Accelerator Physics Group](#) at [Stanford Synchrotron Radiation Laboratory](#) for the ongoing design and future operation needs of [SPEAR3](#) Synchrotron Light Source.



[What is Accelerator Toolbox](#)
[New in AT version 1.2](#)

[Download and Installation](#)
[Get Started](#)

[Collaboration](#)
[Publications](#)

[e-mail AT](#)

[Links](#)

Unofficial release 1.3

www-ssrl.slac.stanford.edu/at/welcome.html

<http://www.slac.stanford.edu/~terebilo/at/>

SLAC website for AT 1.3

New Accelerator Toolbox Homepage (Beta)

[Download atzip.zip](#) (all platforms)

Installation and test is done from MATLAB. Change working directory to the location of atzip.zip

```
ATROOT = 'c:\at'  
unzip('atzip.zip',ATROOT)  
cd(ATROOT)  
atpath  
atdiag
```

This version

Version of AT found on this cite is always the most recent one (latest and greatest)

It is a dynamic copy of AT used by the author and users at SSRL

It is updated 'continuously' - no version numbers

Log of all changes and code snapshots from past dates are available

Compatible with the most recent MATLAB version (7.*).

Most likely NOT compatible with older versions (MATLAB 6.*)

One distribution with binaries for Windows, Linux and Sun Solaris

Documentation

Most of AT 1.2 manual and notes still apply. There are [important differences](#).

[AT 1.2 \(old\) Home](#)

<http://www.slac.stanford.edu/grp/ssrl/spear/at/>

Changes from 1.2

Changes from AT1.2

Pass-Methods

1. PASSMETHODS command prints a list of all available pass methods in .../simulator/element subdirectory.
2. Calling a pass method function with no arguments returns a list of required and optional fields in element data structure:

```
>> [req,opt] = BendLinearPass
```

Optional fields are used only if present.

3. It is now possible to write pass method functions in MATLAB for faster prototyping or for problems where tracking speed is less of a concern. See for example mDriftPass.m and mIdentityPass.m
4. The following pass methods require 'Energy' (value in eV) field in the element structure:

```
StrMPoleSymplectic4RadPass,  
BndMPoleSymplectic4RadPass  
CavityPass
```

Note: Value in global variable GLOBVAL (soon to become obsolete) GLOBVAL.E0 is not used by pass methods but is required by another AT function OHMIENVELOPE.
To ensure consistency, set Energy field in all elements and GLOBVAL.E0 to the same value:

It is recommended that lattice file returns a lattice variable that is assigned a name in workspace. Pass this variable to other AT functions instead of THERING.

```
>> spearlattice = spear3;  
>> findm44(spearlattice,0);
```

2. There is no specific format for lattice files. The only requirement is that proper AT lattice variable is produced. AT lattice variable is a cell array of structures with fields, matching their pass methods. Compare for example spear2.m (AT1.2 style lattice file) and spear3.m. New function ATELEM may be useful in lattice files.

3. For many types of calculations it is no longer a requirement that lattice structure is stored in special global variable 'THERING'.

It is recommended that lattice file returns a lattice variable that is assigned a name in workspace. Pass this variable to other AT functions instead of THERING.

```
>> spearlattice = spear3;  
>> findm44(spearlattice,0);
```

Note: Some GUI functions (INTLAT, INTELEM) still require global THERING to simplify GUI callback code. Also fittune2 and fitchrom2 work with global THERING.

4. Global variable FAMLIST, a byproduct of the AT1.2 style lattice format, is now obsolete.

Widely used Codes for electron rings

--Elegant--
APS
M. Borland et. al

MAD-X/PTC
CERN- module structure
with organizers

???

AT

???

BMAD
Cornell
D. Sagan

OPA
PSI
A. Streun

Tracy2/3
J. Bengtsson

How to continue AT development?

Andrei stopped coordination of AT,
left Acc. Phys. in 2009

Many labs have AT 1.3, and have done some of their
own development

Without coordination, these distributions become
further apart and collaboration becomes more difficult.

AT at ESRF

Substantial AT development has continued at ESRF

- | | |
|-------------------|--|
| Laurent Farvacque | (maintenance and bug fixing, atplot, atfastring and much more) |
| Simone Liuzzo | (developed atmatch, much work on lattice optimization including errors, paralellization of AT) |
| Nicola Carmignani | (lattice optimization, Touschek lifetime, cavity pass method development, collective effects studies, paralellization of AT) |
| Simon White | Developed impedance tracking module in AT |
| Boaz Nash | work on AT infrastructure, collaboration, documentation and physics algorithm development |

Additional use and expertise of AT by Reine Versteegen (loss patterns and optics) and Thomas Perron (tune algorithms, dynamic aperture, optics)

atcollab



How to sustain an open source software collaboration spread across the many labs of AT users and developers?

Since 2010, AT 1.3 code, has been hosted on SourceForge using subversion (svn) for version control.

A mailing list, atcollab-general@lists.sourceforge.net has been created with 40 members, from 15 institutes

Most of the commits to the repository are from ESRF, but also contributions from other labs.

Many labs use AT via MML, but ESRF doesn't have expertise.

A collaboration with SOLEIL and LAL has been developed to confirm compatibility with MML. Several meetings were held and a roadmap developed.

The idea for this meeting today came from this collaboration.

AT community



mailing list: atcollab-general@lists.sourceforge.net

40 members from 15 institutes

ESRF
NSRC, Taiwan
LBL
IHEP, China
ALBA
SLAC
Australian Synchrotron
Mayo Clinic
INFN
Diamond

KIT, ANKA
SOLEIL
ELETTRA
Canadian Light Source
LAL
SESAME
Iranian Light Source
MAX IV
RRCAT/INDUS

Archives and subscription info at

<https://lists.sourceforge.net/lists/listinfo/atcollab-general>

on Twitter:
[@atcollab_tweets](https://twitter.com/atcollab_tweets)

AT 1.4

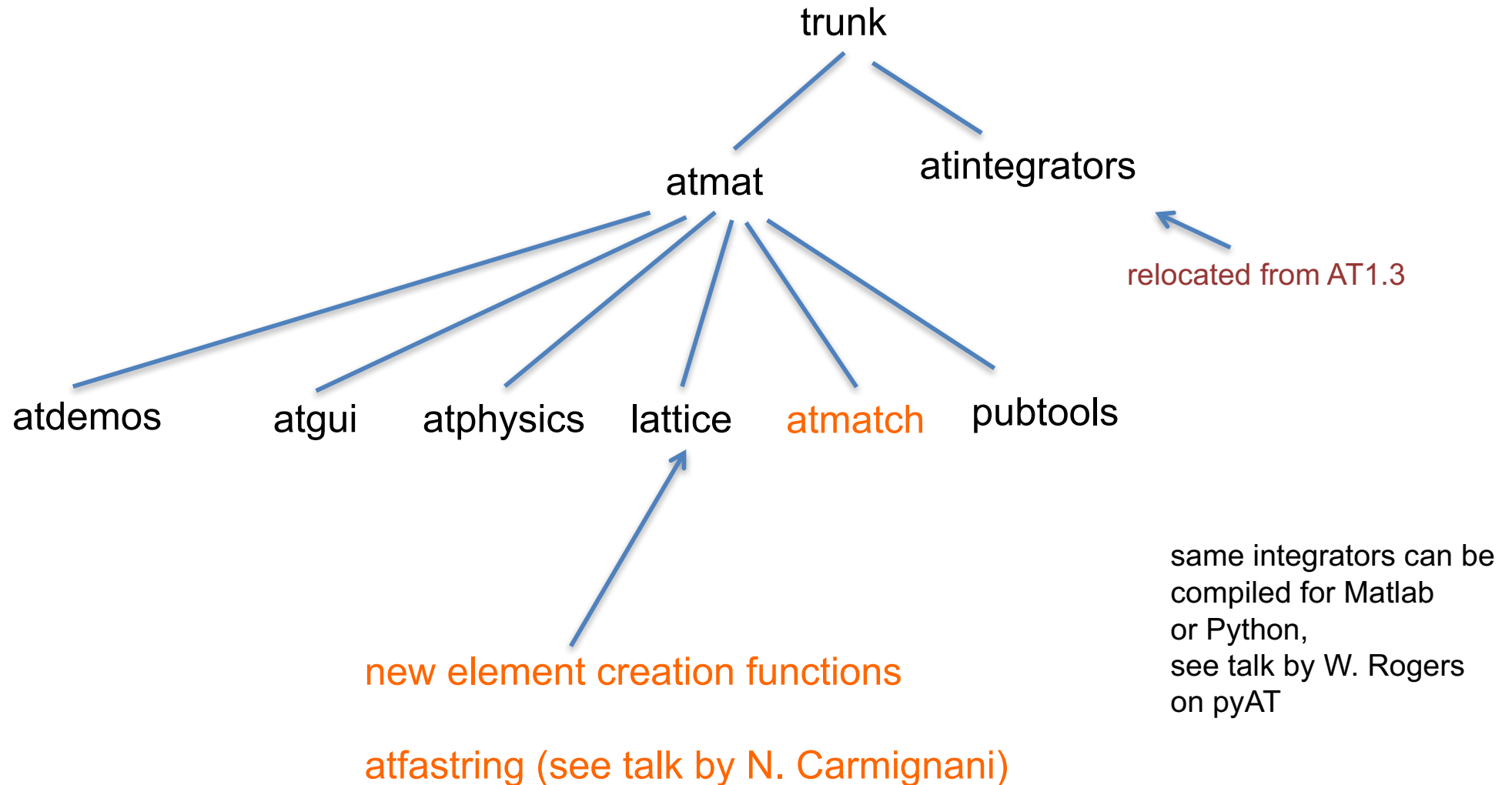
On 24/02/2017, we created a tag of our distribution, labeled AT1.4

Download a zip file here:

<https://sourceforge.net/projects/atcollab/files/latest/download>

AT 1.4 has many new features and is backwards compatible with AT 1.3 (more details in talk by N. Carmignani)

AT 1.4 repository structure



Functions in pubtools

created to distribute tools of varying maturity to be transferred to main directories after sufficient testing.

Examples

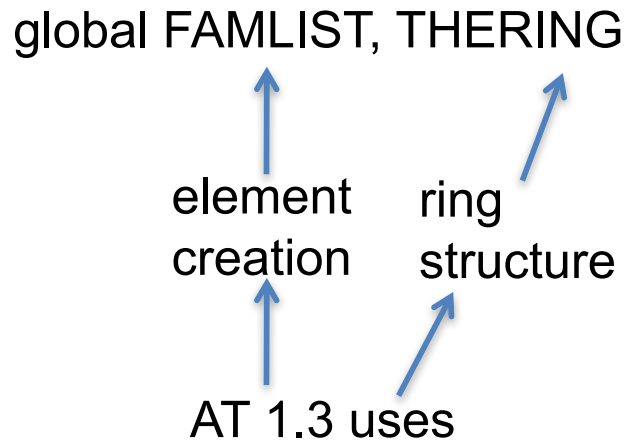
Lattice Conversion tools

AT2G4BL
AT2OPA
MADX2AT
AT2MAD8
ELEGANT2AT
MADX2G4BLAT2MADX
MAD82MADX

Lattice Tuning Functions with errors and corrections, see poster: WEPIK061

frequency map analysis tools including
nafflib naff tune algorithm
local lattice parameters: general coupling analysis
with A matrix, and etaoids/betaoids of E. Forest

Element and Ring Creation without global variables



Issue: We want to be able to create many lattices in one Matlab session

Most functions now require a ring variable as an argument.

example, `linopt` uses global THERING,

```
LinData = atlinopt(ring,dp,refpts)
```

non-global ring structure

Lattice Creation

```

% create a simple fodo lattice
QF=atquadrupole('QF',0.5,1.2,'PassMethod','QuadMPoleFringePass','Energy',3e9);
QD=atquadrupole('QD',0.5,-1.2,'PassMethod','QuadMPoleFringePass','Energy',3e9);
Bend=atsbend('Bend',1,2*pi/40,'PassMethod','BndMPoleSymplectic4Pass','Energy',3e9);

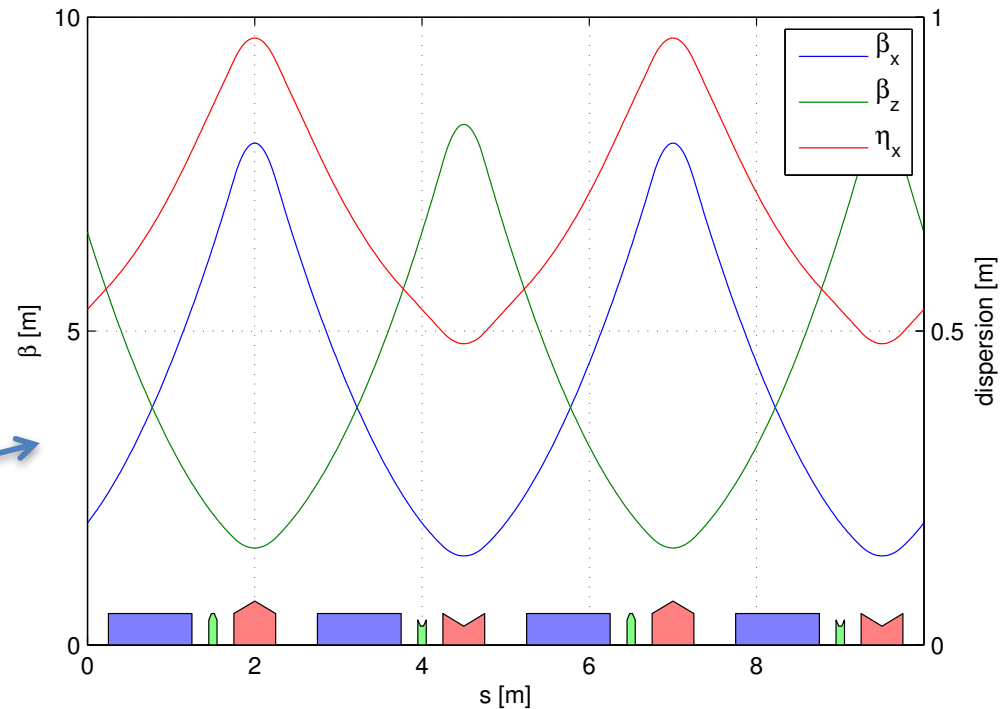
RFC=atrfcavity('RFCav');

Dr=atdrift('Dr',0.5); HalfDr=atdrift('Dr',0.25); p1Dr=atdrift('Dr',0.1);
p2Dr=atdrift('Dr',0.2);p3Dr=atdrift('Dr',0.3);

SF=atsextupole('SF',0.1,1);
SD=atsextupole('SD',0.1,-1);

cell=[{HalfDr};{Bend};{p2Dr};{SF};{p2Dr};{QF};...
{Dr};{Bend};{p2Dr};{SD};{p2Dr};{QD};{HalfDr}];
ring=repmat(cell,20,1);
ring=[{RFC};ring];
    
```

$v_x = 0.522$ $\delta p/p = 0.000$
 $v_z = 0.492$ 1 period, $C = 10.000$



plot with atplot

Example Lattice files

We include a set of lattice files in machine_data directory

Australian Synchrotron
Canadian Light Source
ESRF
Soleil
Thom-X
DBA
FODO

note that the Soleil and Thomx lattices test the different fringe field models that are now available.

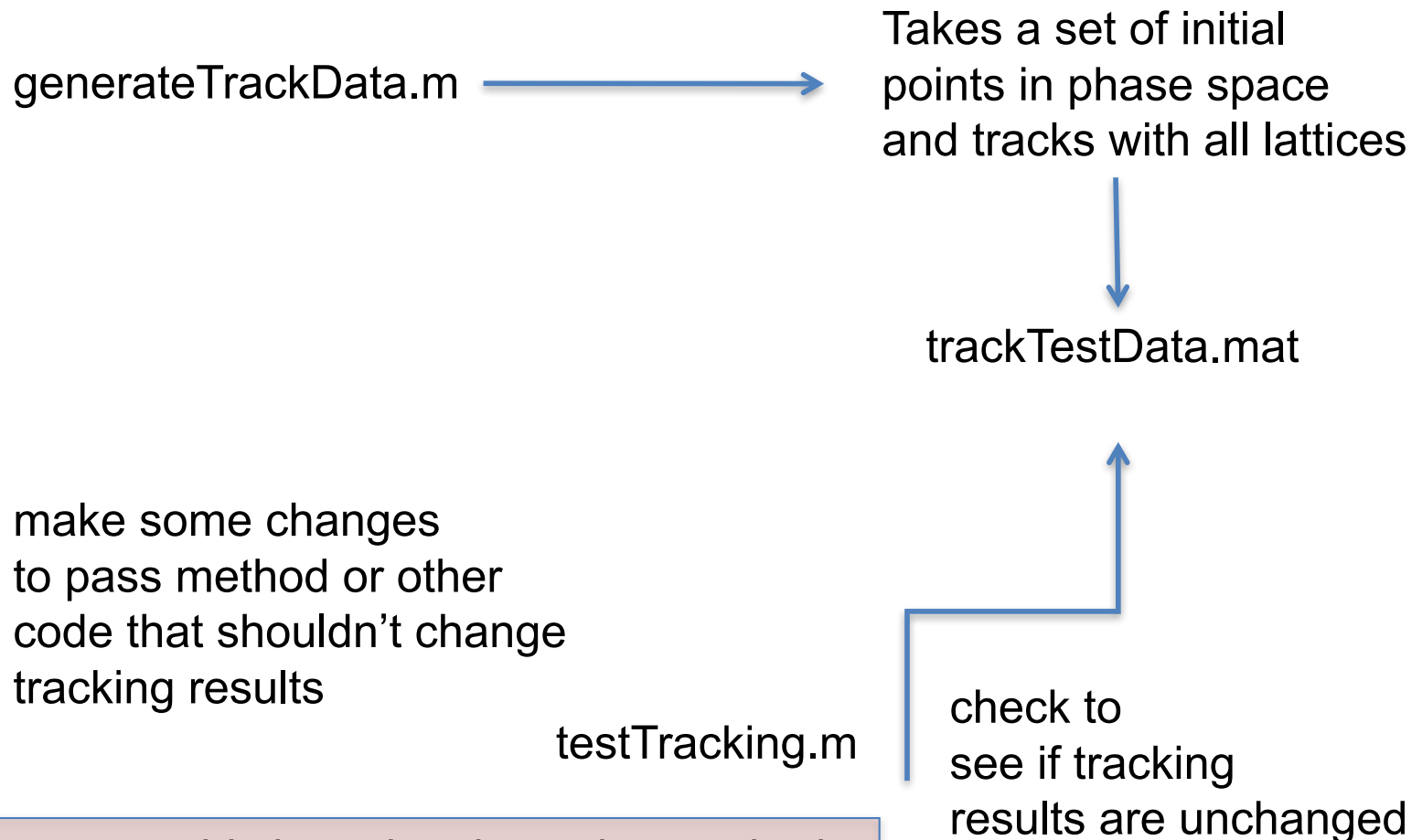
Dipole and quadrupole fringe fields are also available

Examples in different packages should be written to use these lattices

repository for lattice files has been started:

<https://sourceforge.net/p/acceleratorlattices/wiki/Home/>

Tracking tests



Other more sophisticated code testing methods are welcome!

Road Map to AT2.0

We made a list of priority 1,2,3 items to do

After completing priority 1 and 2 items, we will release AT2.0

This list is up for discussion and feedback is welcome!

We will come back to this topic in more detail in the group discussion.

SourceForge/Github svn/git

Sourceforge and SVN were chosen in 2010.

This worked well with a small group mostly centered at ESRF, with a few contributors elsewhere.

For improved collaboration between more labs, we are considering moving to git and GitHub.

A few issues:

mailing list hosted by sourceforge.

We can either keep this, or find another solution.

We discuss this issue more later, during the group discussion.

Conclusions

- AT is a well tested particle tracking code with a large user base
- We are continuing the development, with AT1.4 containing many recent additions
- With some additional features and improved documentation, we head towards AT2.0

Many thanks to colleagues:

Simone Liuzzo, Laurent Farvacque, Simon White

Nicola Carmignani, Laurent Nadolski

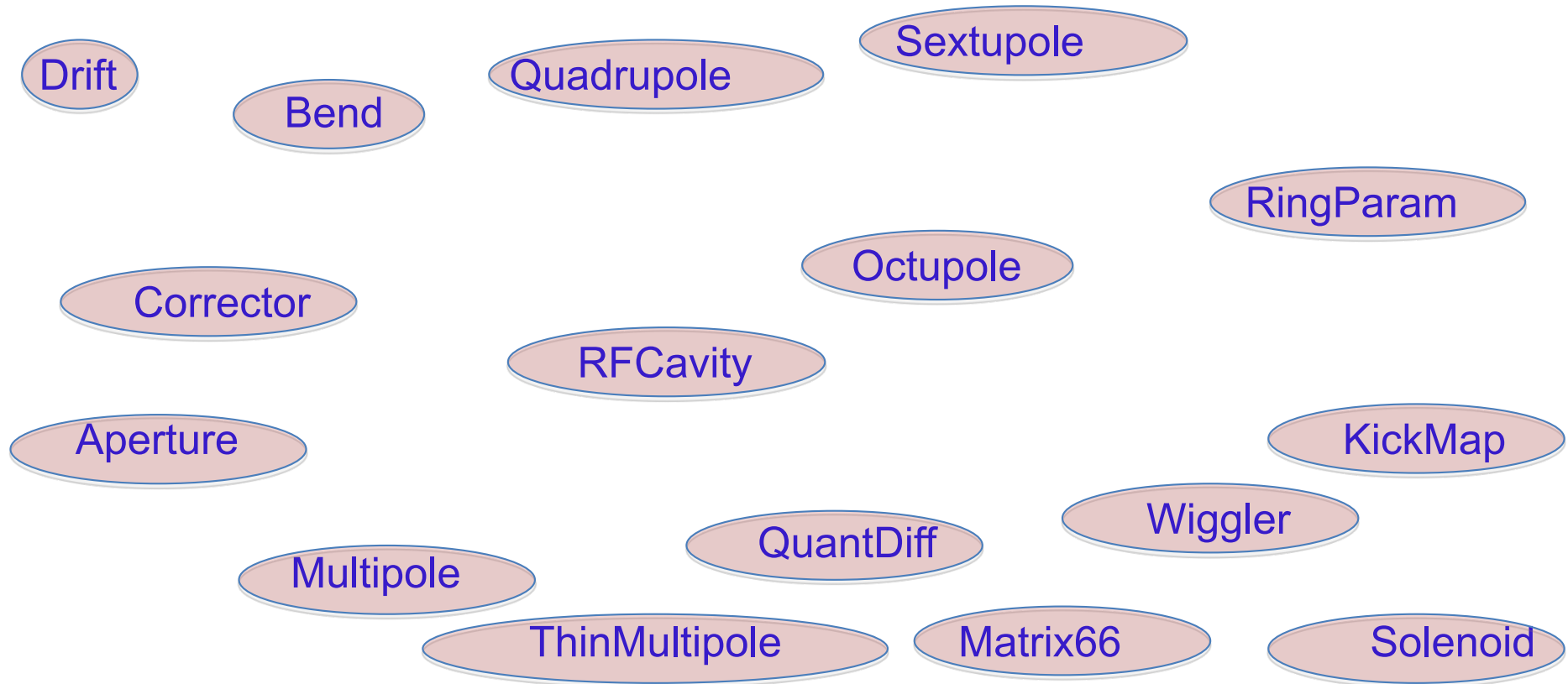
And thanks for your attention!!

Extra Slides

Classes

Issue: how to write functions that identify particular elements that will work for lattices at different labs?

Solution: add a “Class” field to all elements



Example: `indsext=findcells('ring','Class','Sextupole')`
gives indices of all sextupoles

RingParam

Issue: if we want to define many different ring structures at the same time, we need a way to define global parameters for each one.

Idea: create an element in each ring to store global parameters for that ring.

`ATRINGPARAM(FAMNAME,E0,NBPERIODS)`

reference energy

periodicity of lattice

Not many functions yet developed to take advantage of this feature.

(`IdentityPass` needed as `PassMethod` for `RingPass`)

Integrators

- AT has the following integrators in the atintegrators directory:

AperturePass.c

BendLinearPass.c

BndMPoleSymplectic4E2Pass.c

BndMPoleSymplectic4E2RadPass.c

BndMPoleSymplectic4Pass.c

BndMPoleSymplectic4RadPass.c

CavityPass.c

CorrectorPass.c

DriftPass.c

EAperturePass.c

IdTablePass.c

IdentityPass.c

Matrix66Pass.c

QuadLinearPass.c

SolenoidLinearPass.c

StrMPoleSymplectic4Pass.c

WiggLinearPass.c

GWigSymplecticPass.c

Others have developed different integrators.
We should find the most accurate and fastest integrators.