The new fragment picker

Dominik Gront

RosettaCON 2010

Summary

1. Why fragmentpicker?
2. Fragmentpicking overview
3. Fragmentpicking quick examples
   - NNmake - style protocol
   - CS-Rosetta type protocol
4. Fragmentpicking details
   - All about quota
   - Fragments' scoring system
5. Fragmentpicker: under the hood
6. The end
   - There is no end...
   - Acknowledgements

When three is not enough...

There have been already at least 3 fragment pickers:

- NNmake

When three is not enough...

There have been already at least 3 fragment pickers:

- NNmake
- CS-Rosetta

When three is not enough...

There have been already at least 3 fragment pickers:

- NNmake
- CS-Rosetta
- core/fragment/picking/* by Andrew Ban
  (→ core/fragment/picking_old/)

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design
- extensibility

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design
- extensibility
- "One Ring to rule them all"

Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design
- extensibility
- "One Ring to rule them all"
- Protocols to recapture:

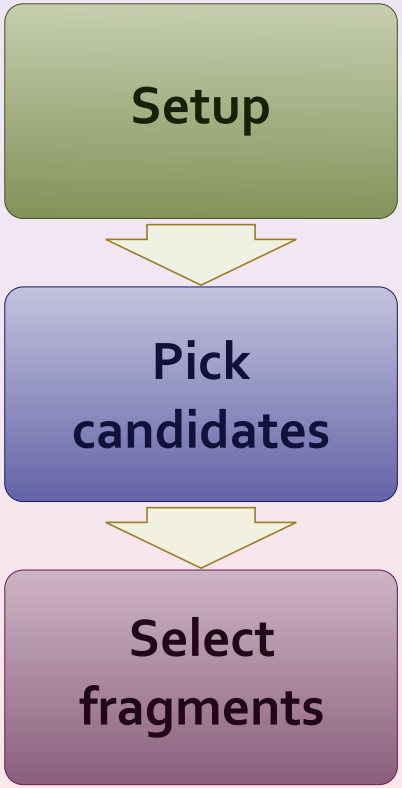
Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design
- extensibility
- "One Ring to rule them all"
- Protocols to recapture:
  - ab-initio protocol (NNmake)

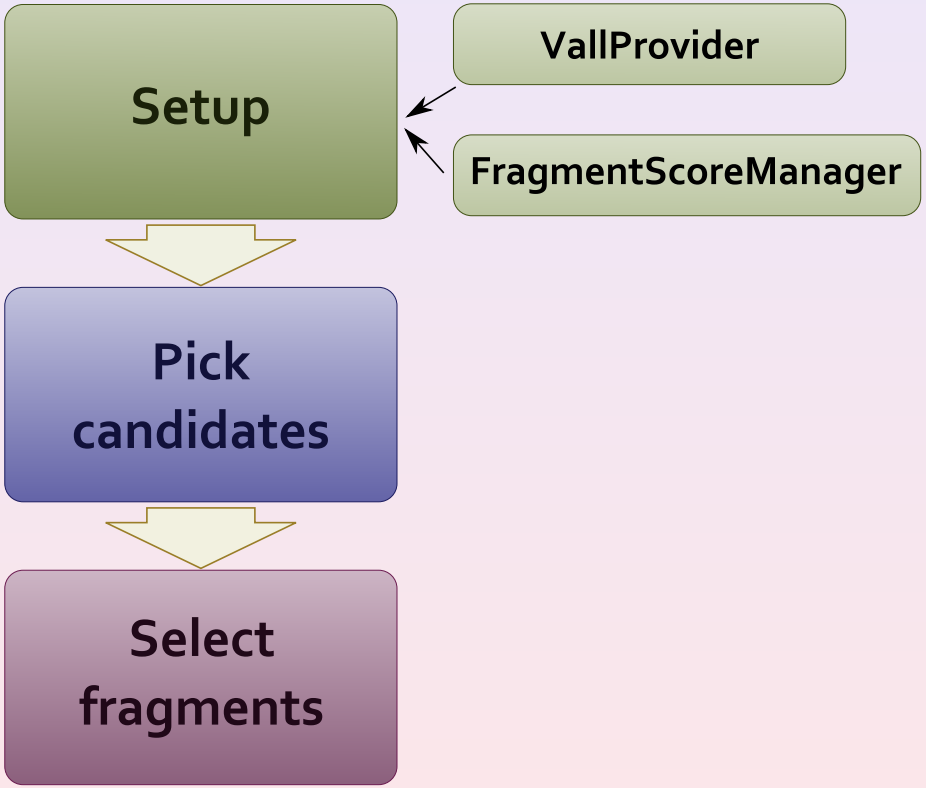
Why the fourth one?

- new, pose dependent scoring terms (disulfide bonds, constraints, etc.)
- new experimental data: J-couplings, VEANS, TEDOR
- modular design
- extensibility
- "One Ring to rule them all"
- Protocols to recapture:
  - ab-initio protocol (NNmake)
  - CS-Rosetta (exp. data)

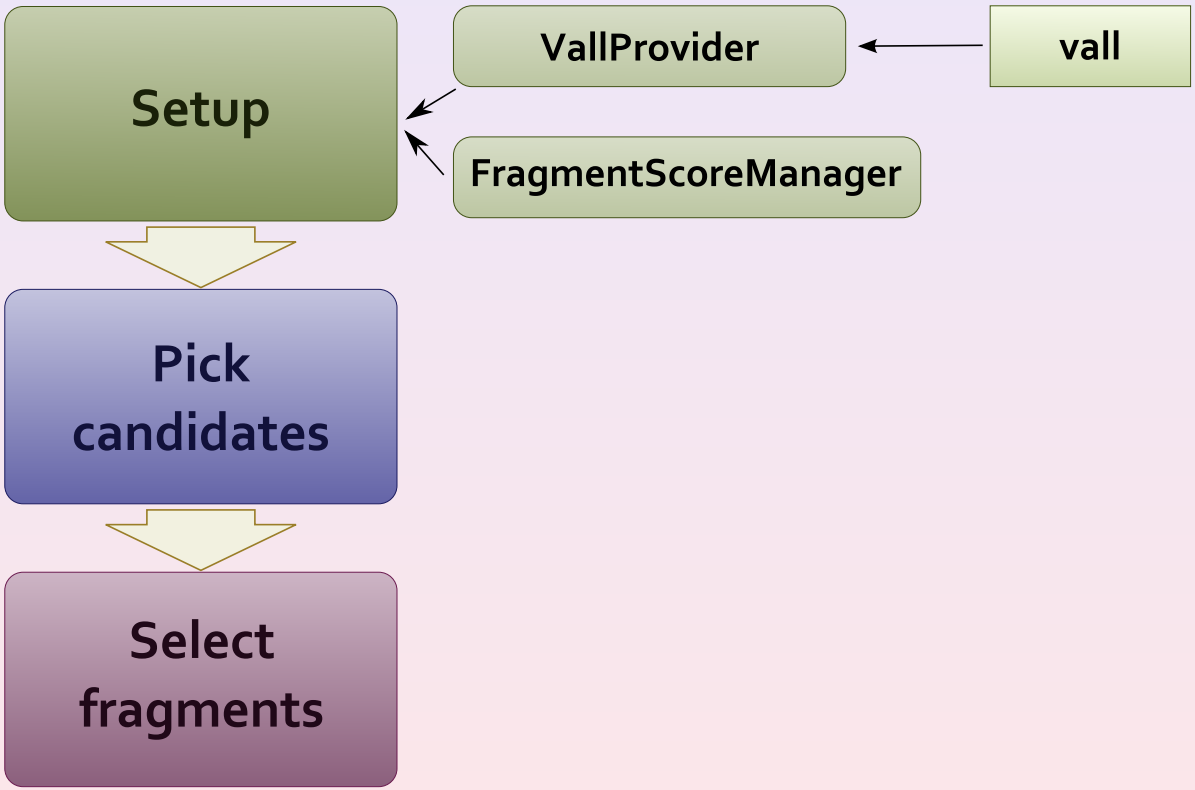
Overview & input files



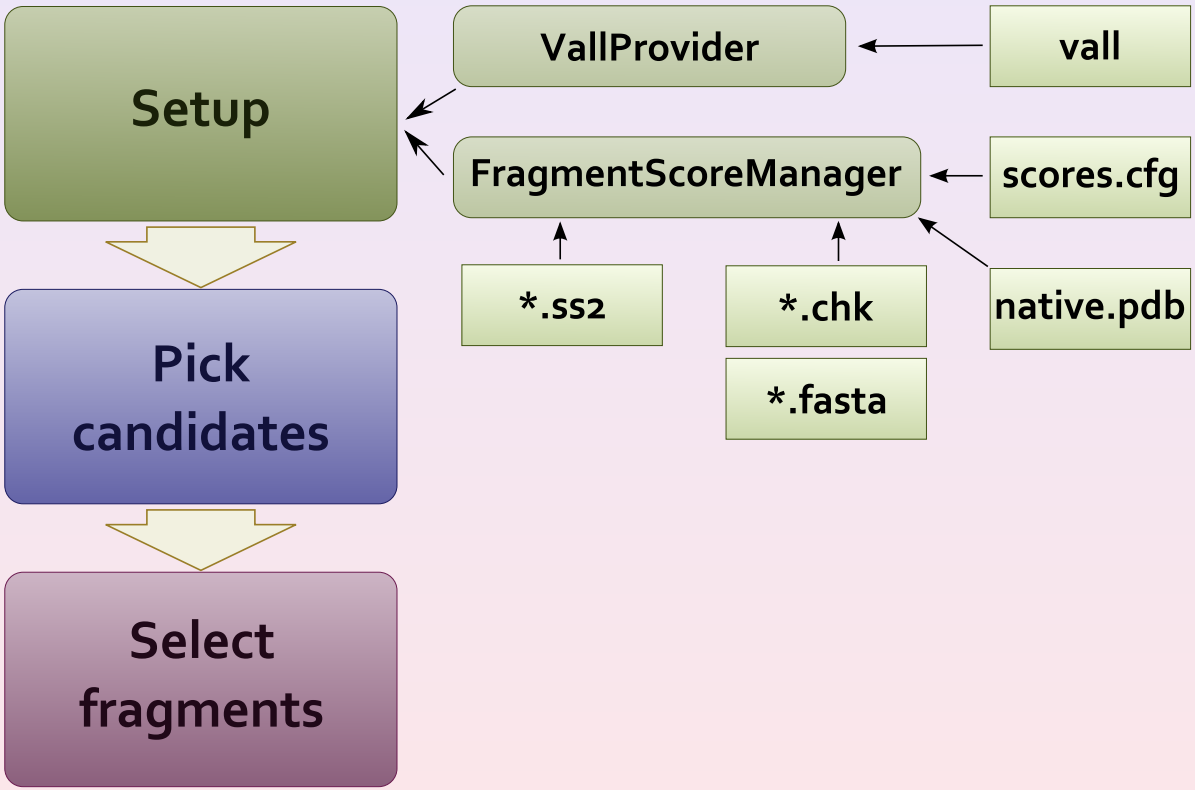
Overview & input files



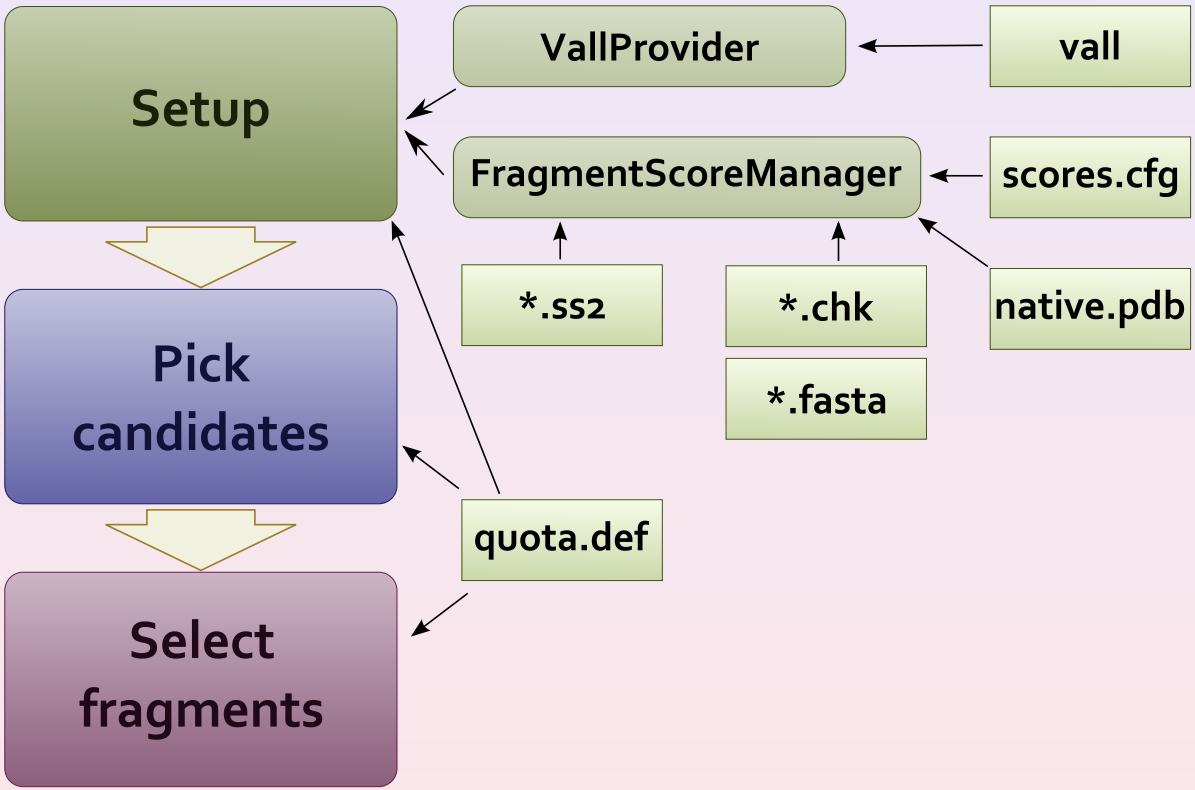
Overview & input files



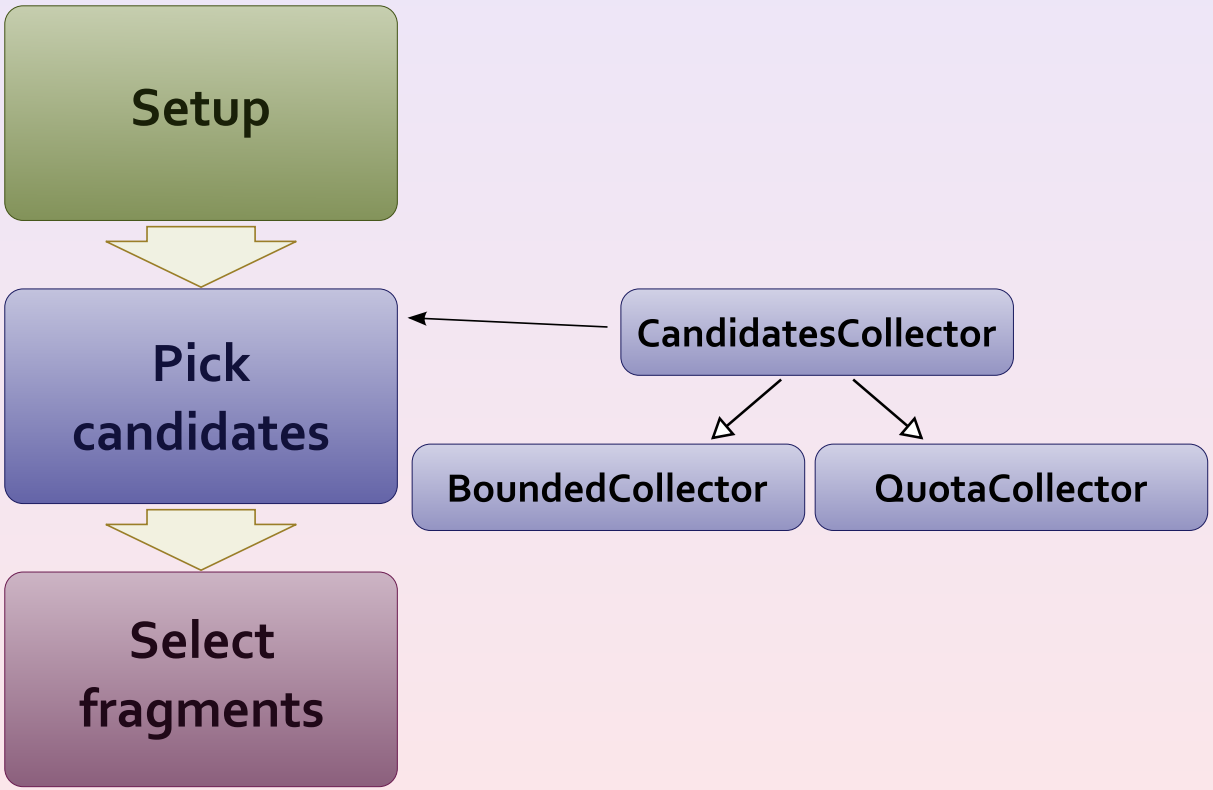
Overview & input files



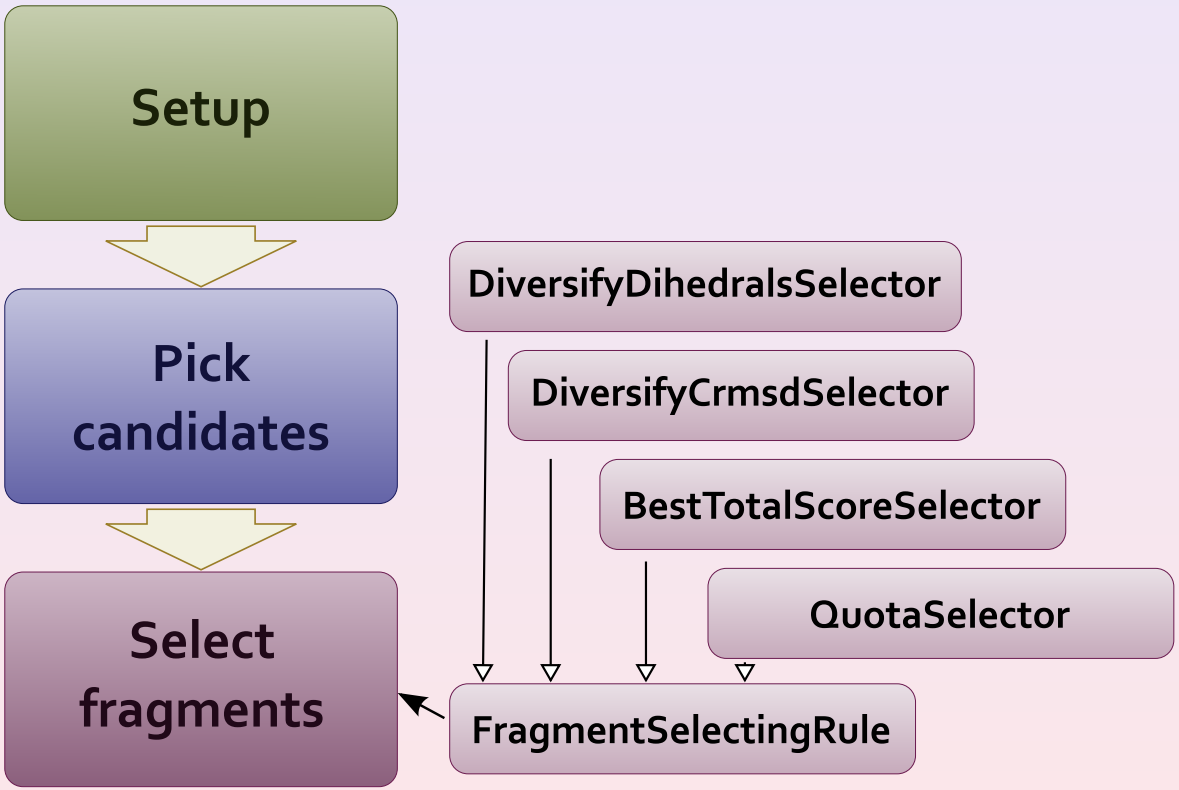
Overview & input files



Overview & input files



Overview & input files



NNmake - style protocol

See `Psi-Jufo-Sam-L1-Quota` demo

- collector: QuotaCollector

NNmake - style protocol

See `Psi-Jufo-Sam-L1-Quota` demo

- collector: QuotaCollector
- selector: QuotaSelector

NNmake - style protocol

See `Psi-Jufo-Sam-L1-Quota` demo

- collector: QuotaCollector
- selector: QuotaSelector
- scoring:(file: `psi_sam_jufo_L1-Q.cfg`)

```
# score name        priority wght cutoff extras
SecondarySimilarity   350 1.0 - psipred
SecondarySimilarity   300 1.0 - sam
SecondarySimilarity   250 1.0 - jufo
RamaScore             150 50.0 -
ProfileScoreL1        200 1.0   -
PhiPsiSquareWell      100 0.0 -
FragmentCrmsd         30  0.0  -
```

NNmake - style protocol

See `Psi-Jufo-Sam-L1-Quota` demo

- collector: QuotaCollector
- selector: QuotaSelector
- scoring:(file: `psi_sam_jufo_L1-Q.cfg`)

```
# score name        priority wght cutoff extras
SecondarySimilarity   350 1.0 - psipred
SecondarySimilarity   300 1.0 - sam
SecondarySimilarity   250 1.0 - jufo
RamaScore             150 50.0 -
ProfileScoreL1        200 1.0   -
PhiPsiSquareWell      100 0.0 -
FragmentCrmsd         30  0.0  -
```

- quota setup (file: `Quota.def`):

```
#pool_id pool_name fraction
1  psipred  0.6
2  jufo     0.2
3  sam     0.2
```

CS-Rosetta protocol

See `CS-Talos-L1-Rama` demo

- collector: BoundedCollector

CS-Rosetta protocol

See `CS-Talos-L1-Rama` demo

- collector: BoundedCollector
- selector: BestTotalScoreSelector

CS-Rosetta protocol

See `CS-Talos-L1-Rama` demo

- collector: BoundedCollector
- selector: BestTotalScoreSelector
- scoring:(file: `CS-Talos-L1-Rama.cfg`)

```
# score name  priority  wght cutoff  extras
CSScore              375   3.0    -
RamaScore            400   50.0   -    talos
SecondarySimilarity  350   3.0    -    talos
ProfileScoreL1       200   1.0    -
PhiPsiSquareWell     100   0.0    -
FragmentCrmsd        30    0.0    -
GunnCostScore        20    0.0    -
```

Picking results

- fragment files

Picking results

- fragment files
- score files

How quota works

Vall fragment

12asA	F	E	167	163	...
12asA	V	E	168	162	...
12asA	H	E	169	161	...
12asA	S	H	170	160	...
12asA	Q	H	171	159	...
12asA	E	H	172	158	...
12asA	L	H	173	157	...
12asA	L	H	174	156	...
12asA	S	H	175	155	...

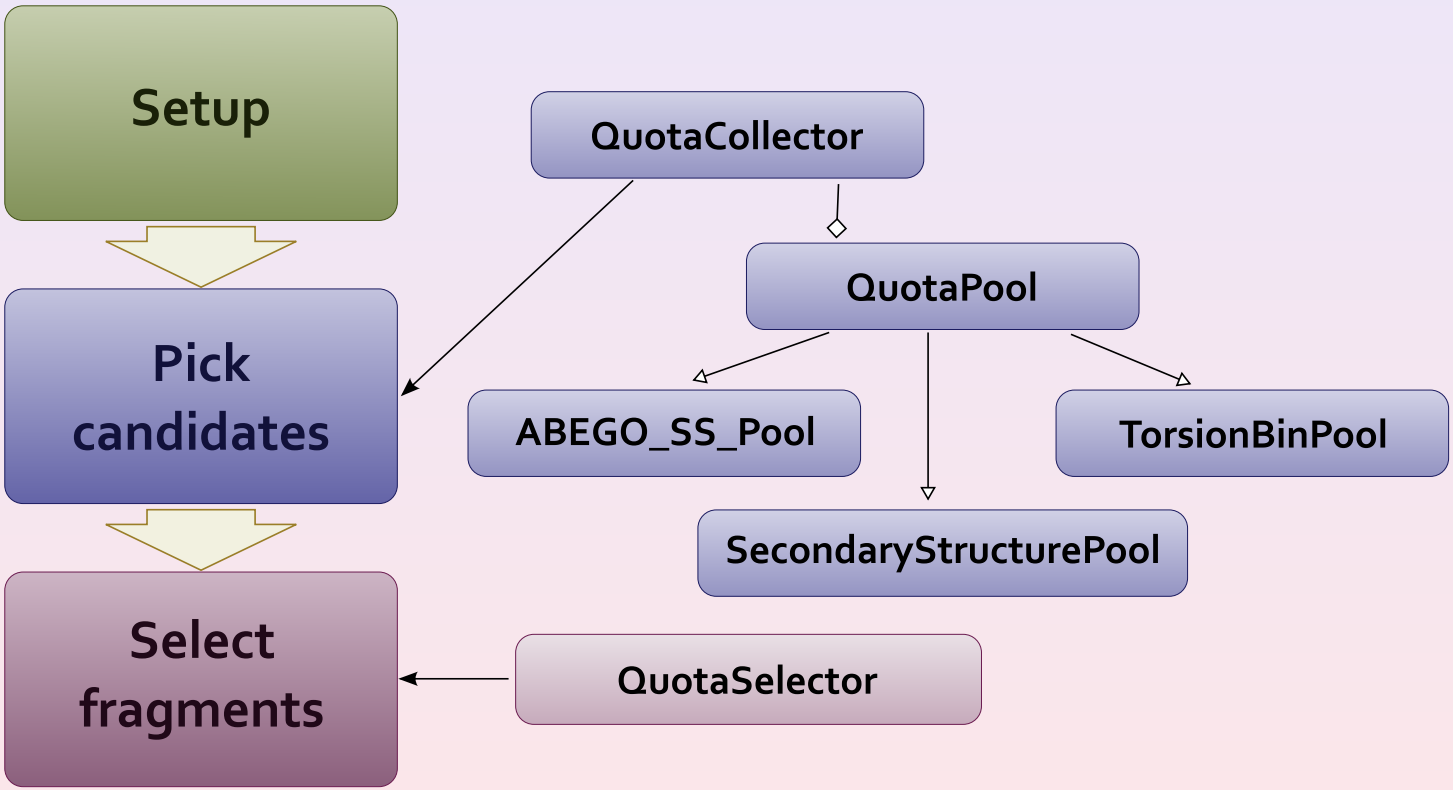
```
#pool_id pool_name fraction
1  psipred  0.6
2  jufo     0.2
3  sam      0.2
```

PsiPred prediction

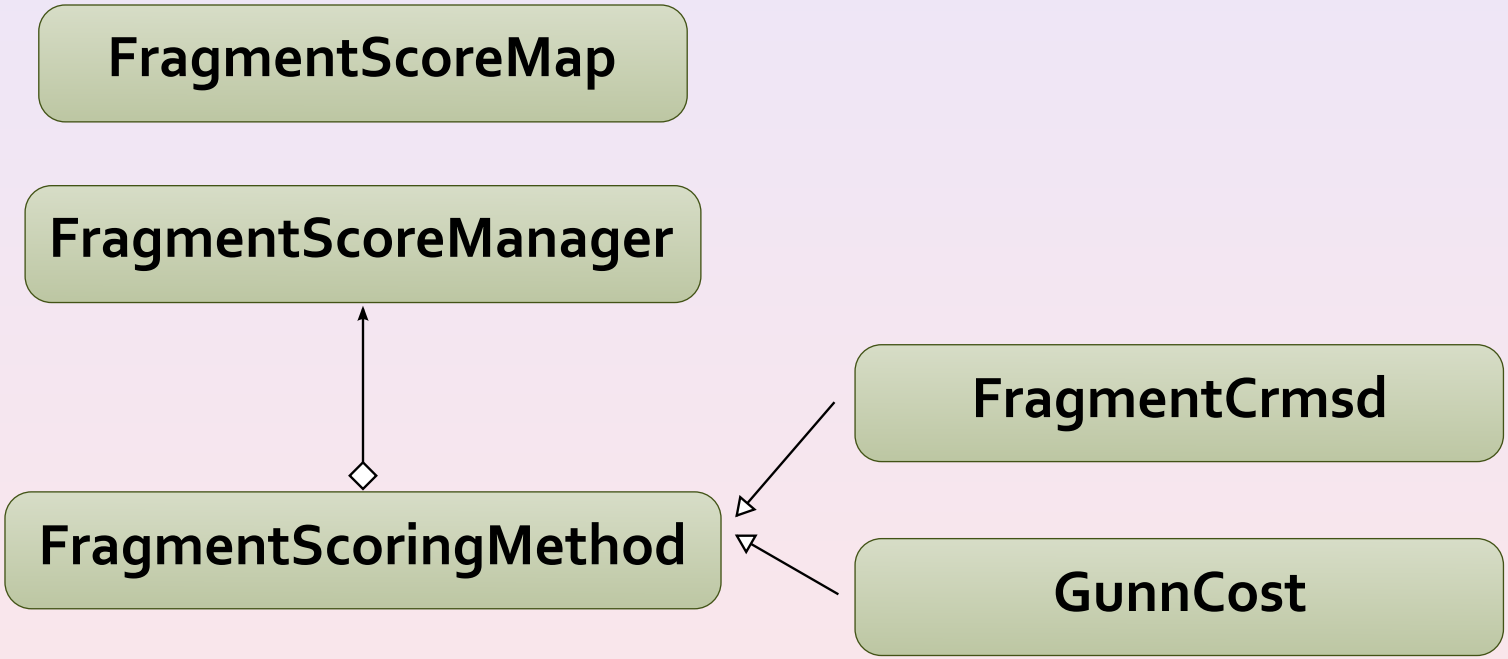
16	T	E	0.030	0.016	0.924
17	T	E	0.065	0.207	0.636
18	T	E	0.101	0.258	0.569
19	E	H	0.114	0.586	0.348
20	A	H	0.083	0.335	0.582
21	V	H	0.125	0.521	0.354
22	D	H	0.545	0.177	0.215
23	A	H	0.079	0.916	0.011
24	A	H	0.052	0.933	0.003

psipred-C : $0.125 \times 200 \times 0.6 = 15$
psipred-H : $0.521 \times 200 \times 0.6 = 63$
psipred-E : $0.354 \times 200 \times 0.6 = 42$

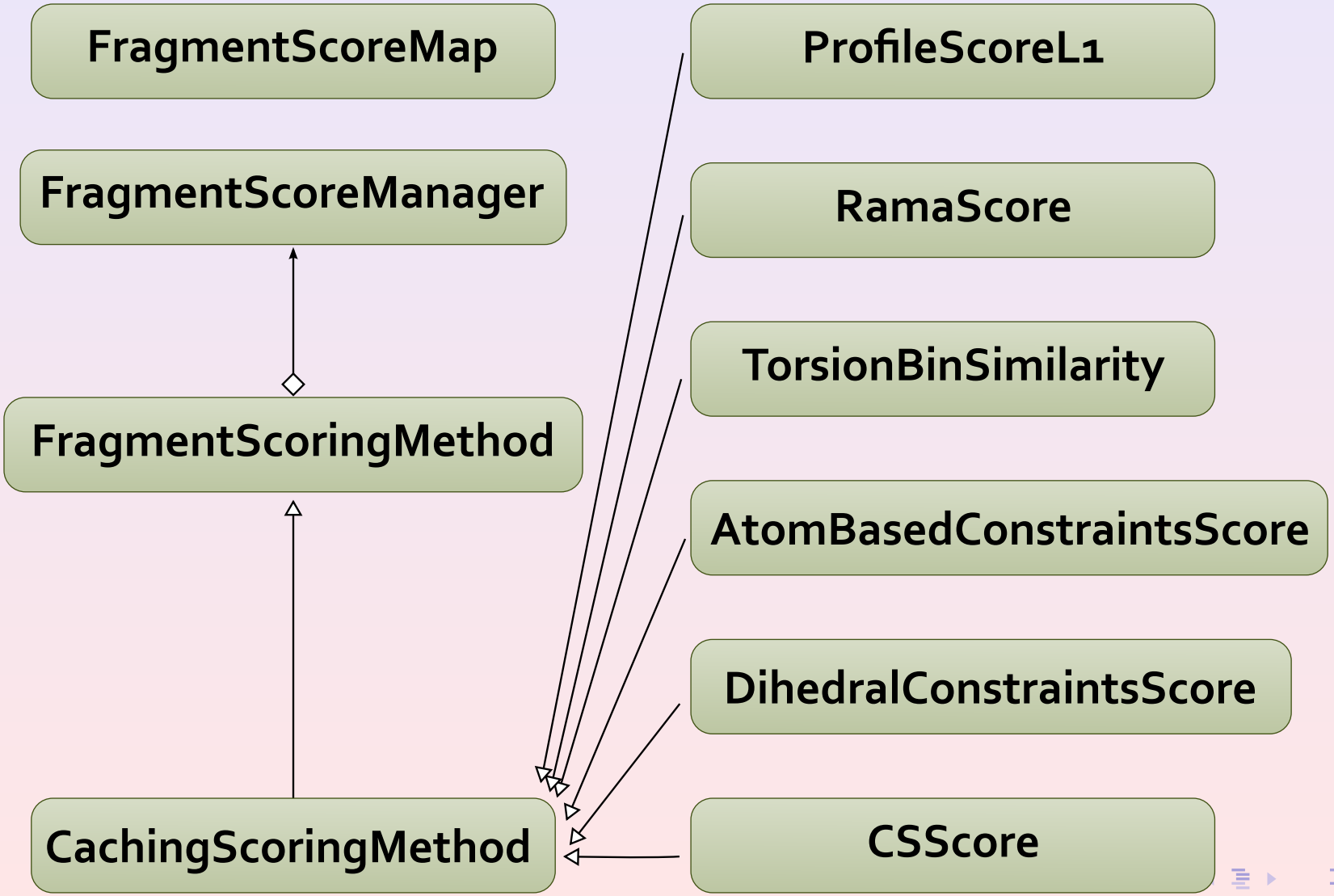
Quota-related parts



Scoring-related classes



Scoring-related classes (2)



fragmentpicker_integration_demo.cc

see
mini/src/apps/pilot/dgront/fragmentpicker/fragmentpicker_integration_demo.c

Other random things

- the application is called picker.* (e.g. picker.linuxgccrelease)

Other random things

- the application is called picker.* (e.g. picker.linuxgccrelease)
- Chunk-wise vall processing paradigm (chunk is a complete part of a protein chain (no missing residues))

Other random things

- the application is called picker.* (e.g. picker.linuxgccrelease)
- Chunk-wise vall processing paradigm (chunk is a complete part of a protein chain (no missing residues))
- pick any fragment size you want: `-frags::frag_sizes 3 6 9 17`

Other random things

- the application is called picker.* (e.g. picker.linuxgccrelease)
- Chunk-wise vall processing paradigm (chunk is a complete part of a protein chain (no missing residues))
- pick any fragment size you want: `-frags::frag_sizes 3 6 9 17`
- restrict sequence range: `-frags::picking::query_pos 110 111`

Other random things

- the application is called picker.* (e.g. picker.linuxgccrelease)
- Chunk-wise vall processing paradigm (chunk is a complete part of a protein chain (no missing residues))
- pick any fragment size you want: `-frags::frag_sizes 3 6 9 17`
- restrict sequence range: `-frags::picking::query_pos 110 111`
- drink Porter, no more Jufo

Future plans

- more scores

Future plans

- more scores
- more experimental data

Future plans

- more scores
- more experimental data
- SQL database instead of vall

Acknowledgements

- David Baker

Acknowledgements

- David Baker
- Andrew LF & Andrew B

Acknowledgements

- David Baker
- Andrew LF & Andrew B
- David K

Acknowledgements

- David Baker
- Andrew LF & Andrew B
- David K
- James

Acknowledgements

- David Baker
- Andrew LF & Andrew B
- David K
- James
- Rob