

Code development with MadGraph 5

Johan Alwall
Fermilab

ERC Miniworkshop
CERN, 27 Jan 2012

MadGraph 5

J.A., Herquet, Maltoni, Mattelaer, Stelzer, arXiv:1106.0522

- MadGraph 5 - a collaborative platform for matrix element-related development
- Modular code - easy to extend or plug'n'play
- Core development language: Python
- Full support for any Lagrangian-based models (renormalizable or effective) via FeynRules, UFO and ALOHA
- Presently supported output languages: Fortran, C++, Python

Development language: Python

- ➔ Exceptional flexibility (and speed where needed)
- ➔ Easy to read/learn/maintain
- ➔ Object oriented - Easy to create modular structure and “PnP”
- ➔ Rich standard libraries
- ➔ Automatic documentation
- ➔ Easy to incorporate test suite
- Fortran or C++ used for “heavy duty” code!

Code organization



Code organization

madgraph

Python source
code for
MadGraph 5

models

UFO interface
UFO and v4 models

aloha

Source code for
output of ALOHA
helicity amplitudes
from UFO models

Template

Fortran source
code and control
files for event
generation using
MadEvent

bin

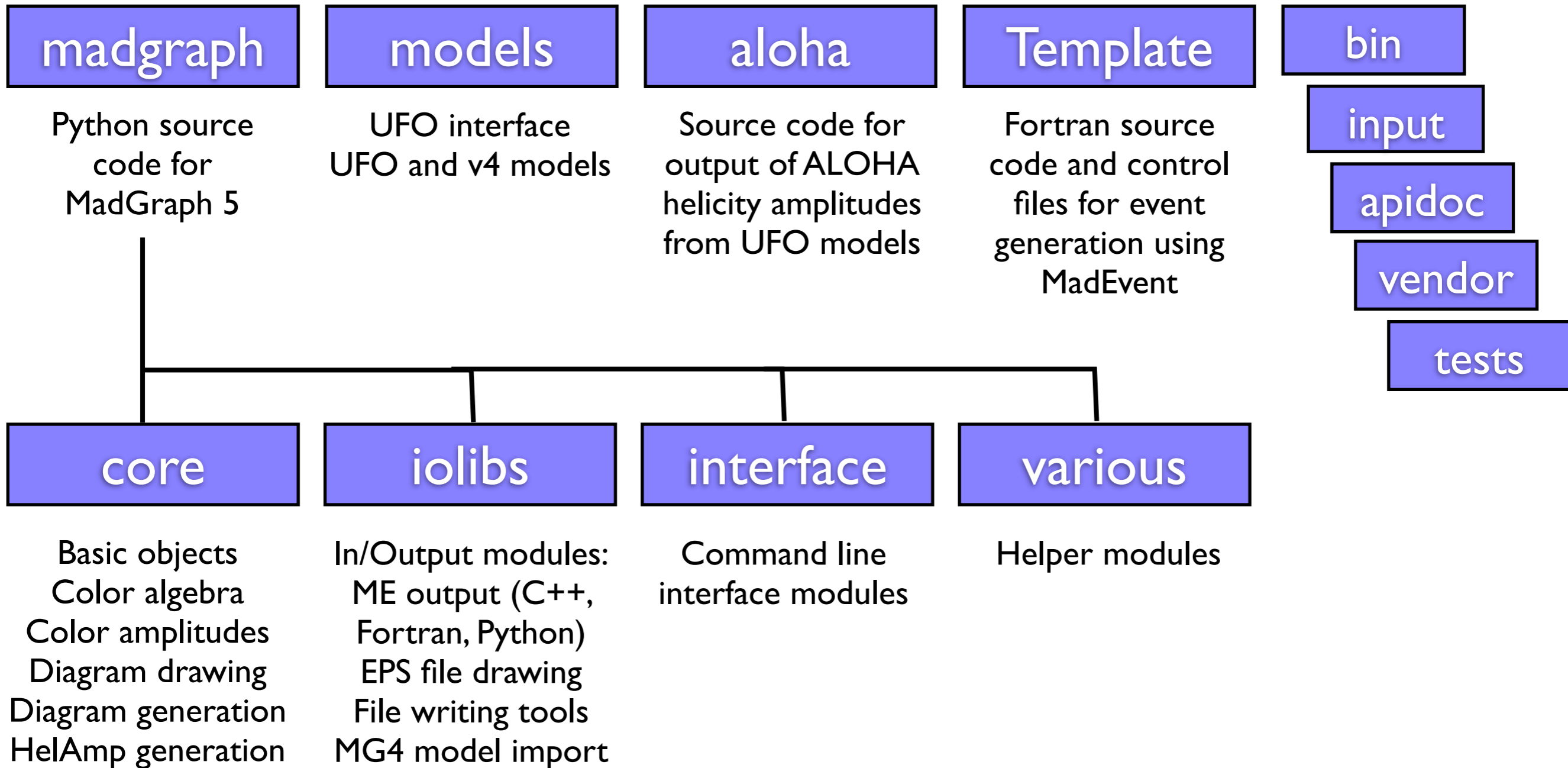
input

apidoc

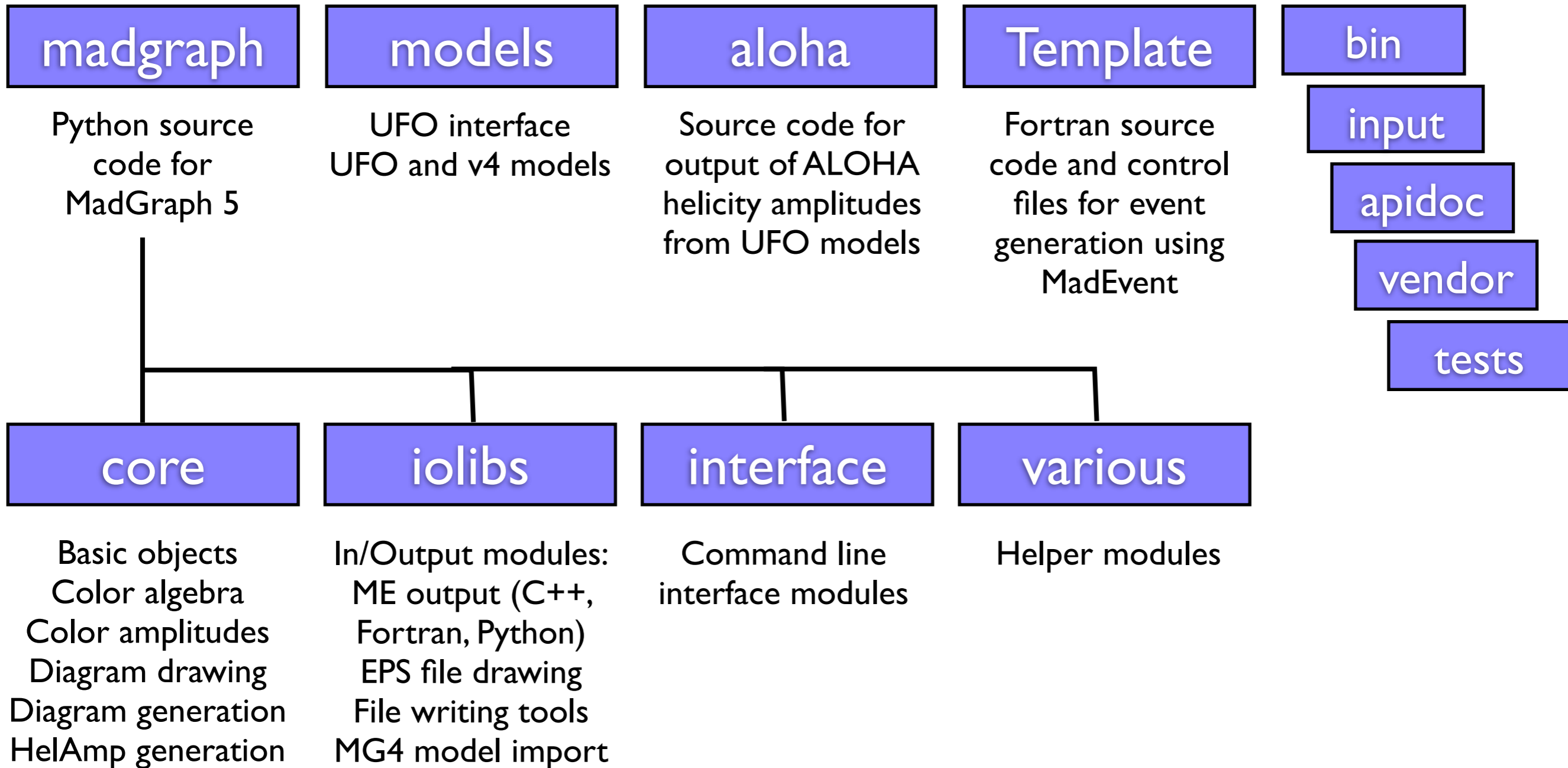
vendor

tests

Code organization

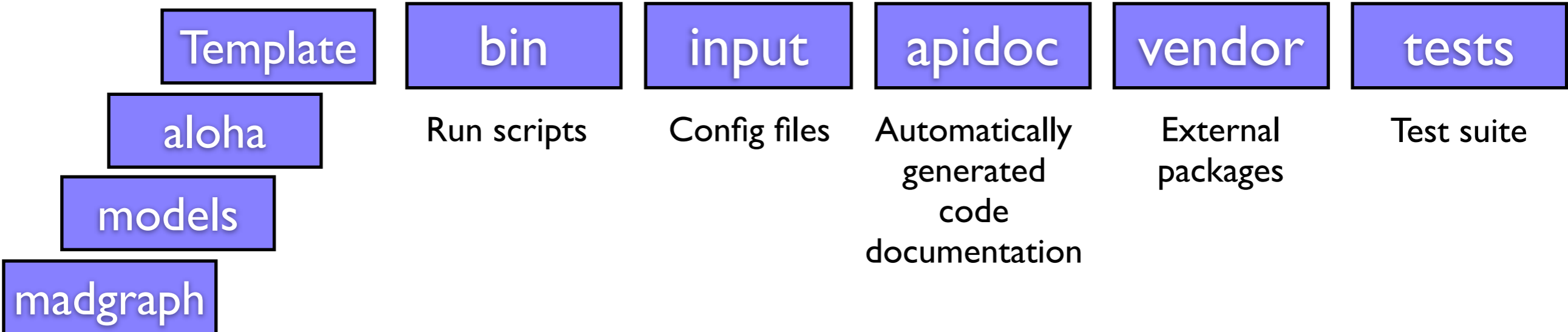


Code organization

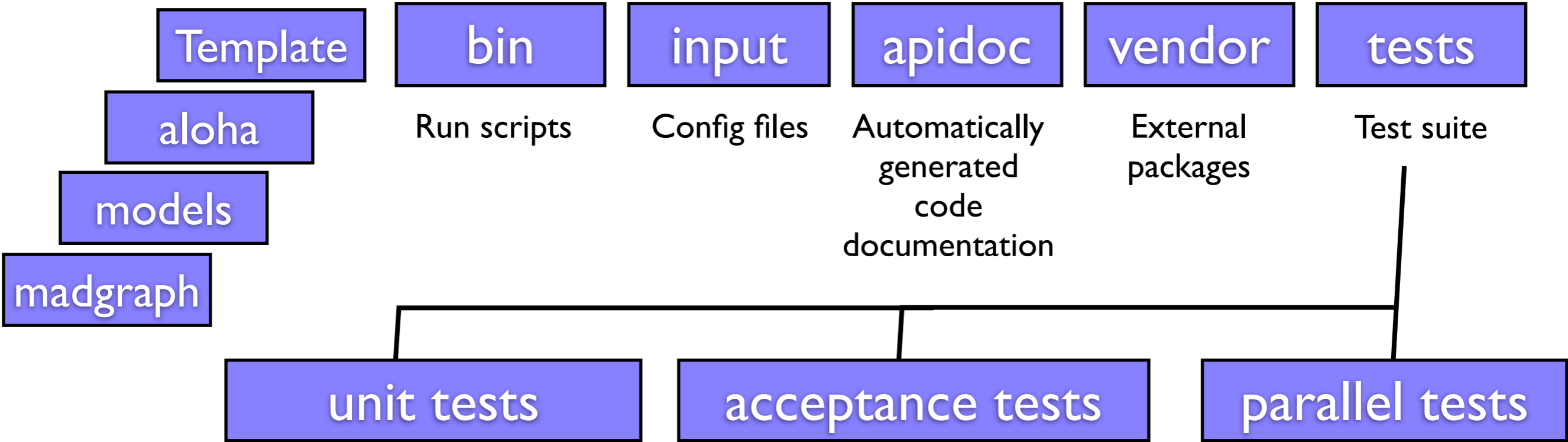


Pick and choose functionality/plug in new modules easy!

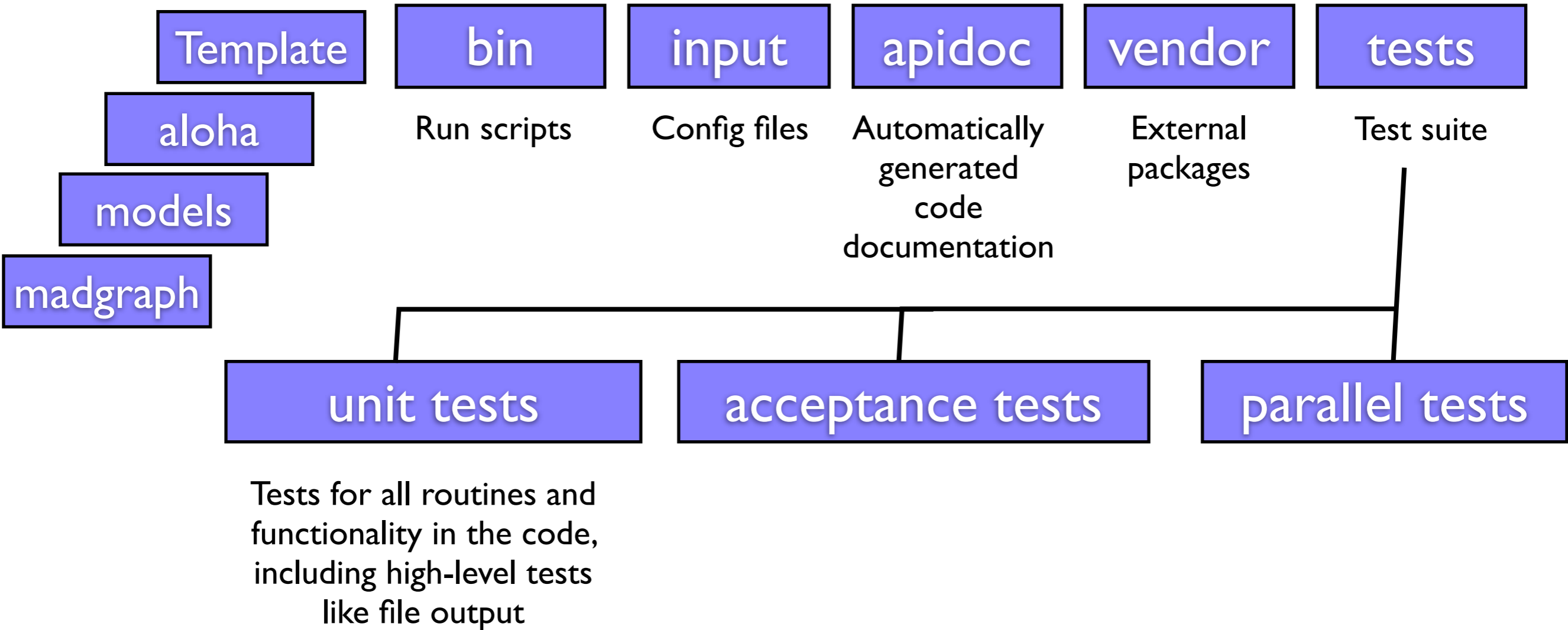
Code organization



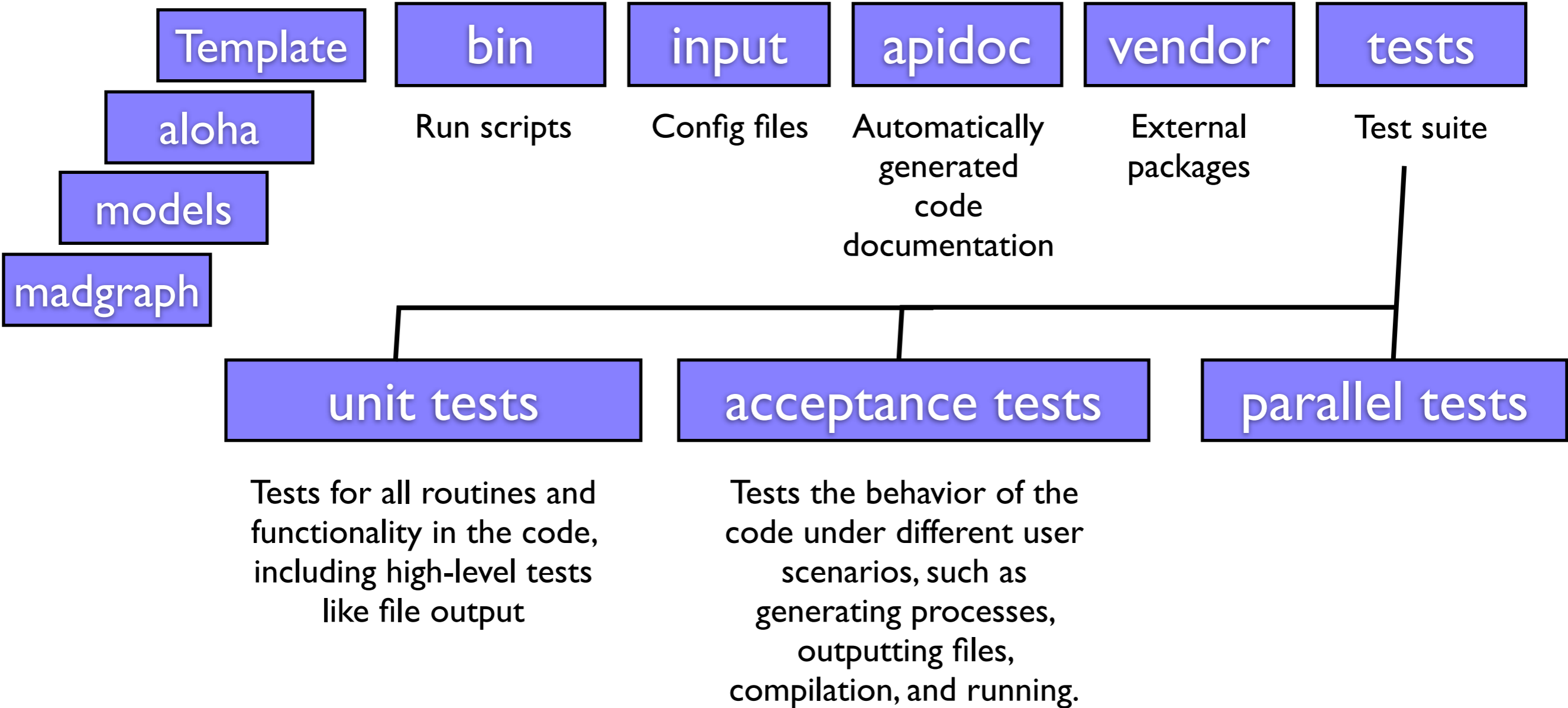
Code organization



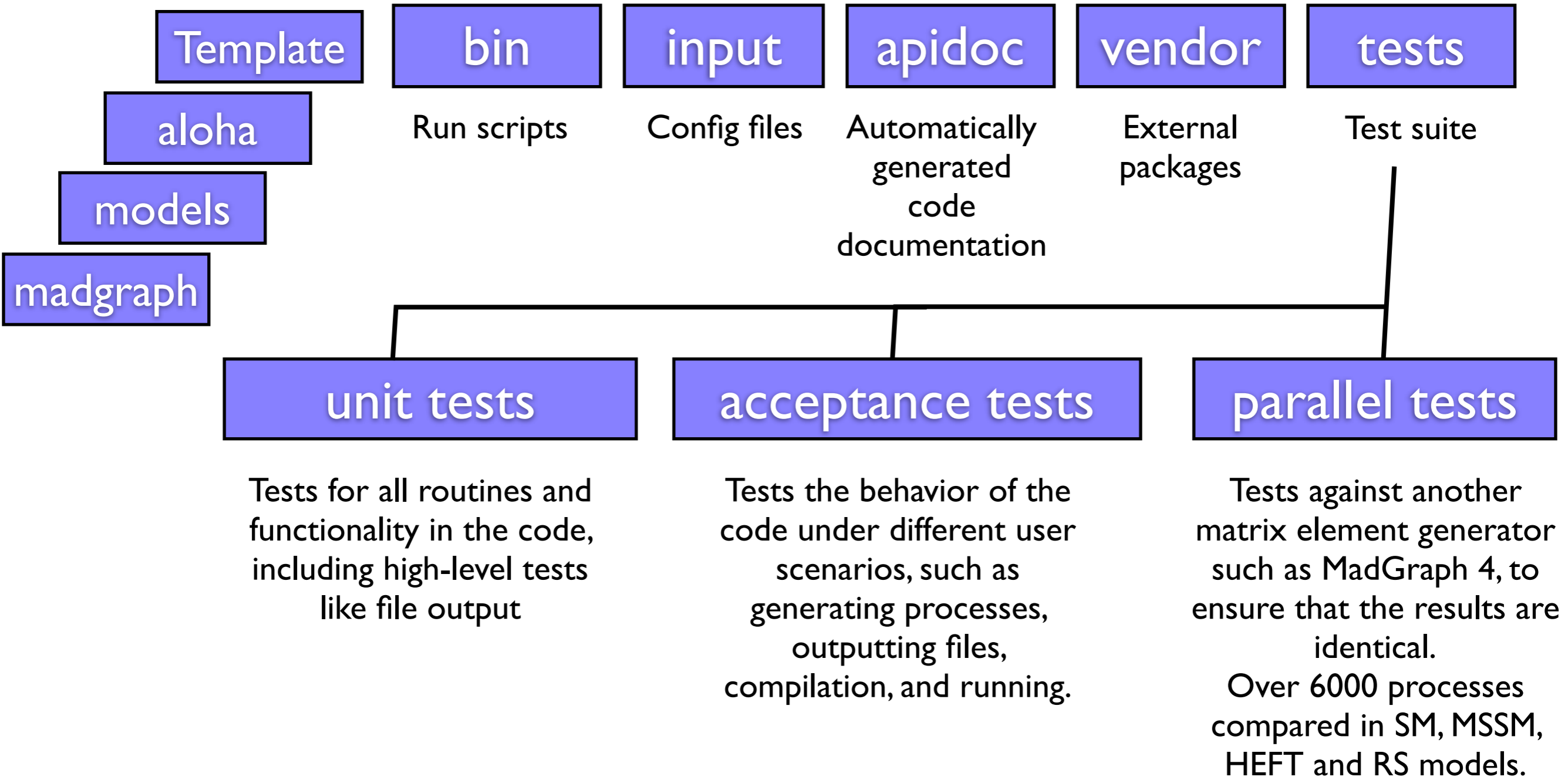
Code organization



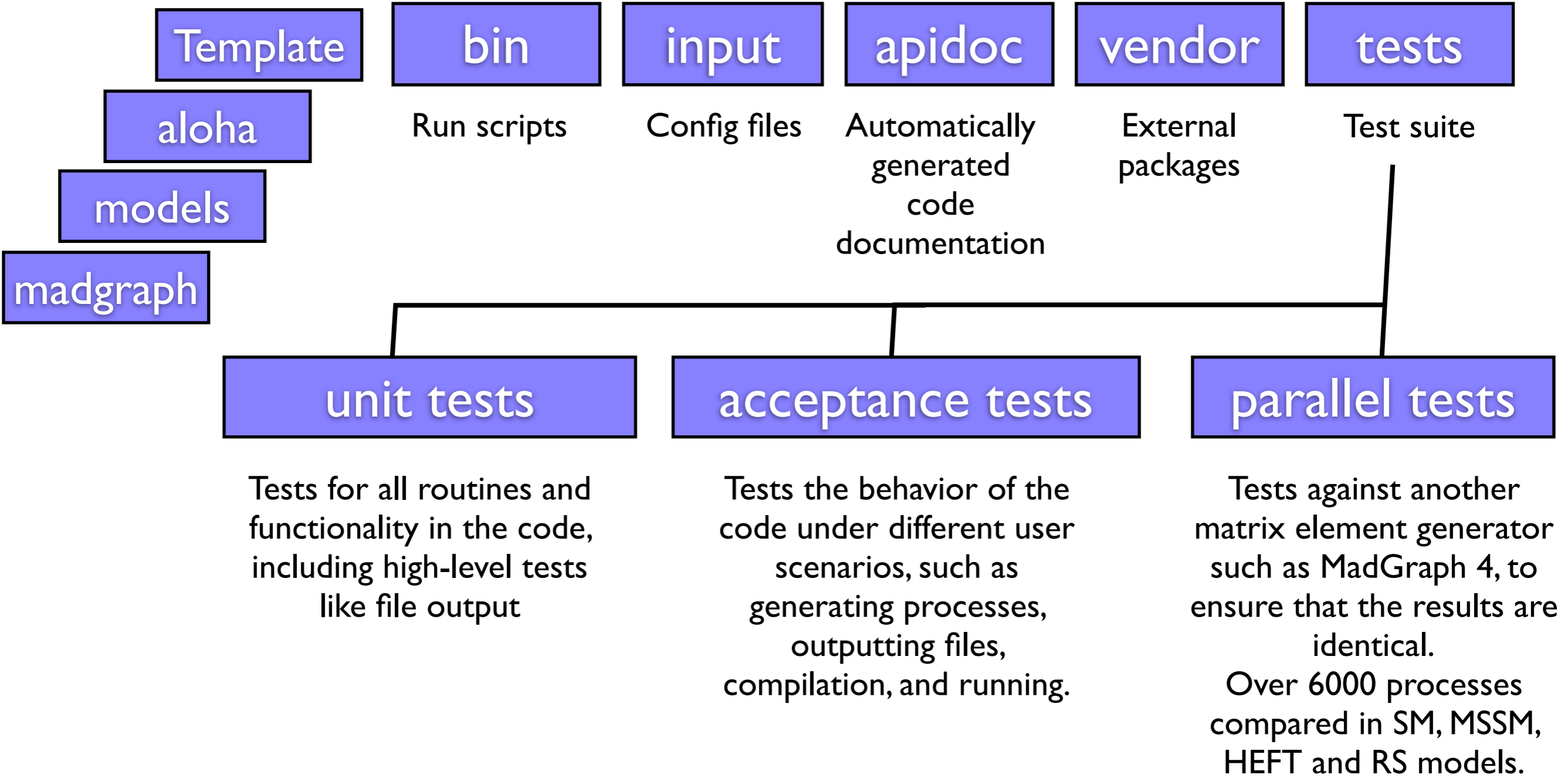
Code organization



Code organization



Code organization



Test suite fundamental for stable development!

“Extreme programming” philosophy

- Modular and highly documented code allows fast turnaround for bugs/new features
- Extensive test suite (from subroutine level to overall functionality) increases flexibility (optimize where needed) and ensures that new functionality doesn’t “break” existing functionality (~ 50% of total code is in the test suite!)
- Preferably use “pair programming” (at least in initial stage) - four eyes see better than two!
- Code public at every stage, using Bazaar and Launchpad
- Code review of independent developer(s) before merging new branches

Result: Fast development and high code stability



The MadGraph Matrix Element Generator version 5

Johan Alwall (johan-alwall) • [Log Out](#)

[Overview](#) **Code** [Bugs](#) [Blueprints](#) [Translations](#) [Answers](#)

Bazaar branches of MadGraph5

MadGraph5 » Code

You can [browse the source code](#) for the development focus branch or get a copy of the branch using the command:

```
bzr branch lp:madgraph5
```

📄 There are [download files](#) available for MadGraph5.

MadGraph5 has 5 [active reviews](#).

MadGraph5 has 13 active branches owned by 2 teams. There were 52 commits by 4 people in the last month.

Branches with status:

Name	Status	Last Modified	Last Commit
📁 lp:madgraph5 Series: trunk	Mature	2012-01-25	199. Slight update to UpdateNotes.txt
📁 lp:madgraph5/2.0 Series: 2.0	Experimental	2011-10-16	177. Turn off heavyrad when there are BWs ...
📁 lp:~maddevelopers/madgraph5/complex_scheme	Development	11 hours ago	167. merge with last 1.4.x version
📁 lp:~maddevelopers/madgraph5/fix_fermion_order_interacti on	🔒 Development	20 hours ago	201. Removed some commented lines and stup...
📁 lp:~maddevelopers/madgraph5/new_color_ordering	Experimental	2012-01-24	207. Use itertools.permutations and the co...
📁 lp:~maddevelopers/madgraph5/upgrade_pythia_ compatibility	🔒 Development	2012-01-21	344. improve auto-completion with correct ...
📁 lp:~maddevelopers/madgraph5/python_standalone	Development	2012-01-20	192. upgraded version
📁 lp:~maddevelopers/madgraph5/spin_three_half	Development	2012-01-20	336. merge with the last 1.4 version
📁 lp:~maddevelopers/madgraph5/NLO	🔒 Development	2012-01-05	213. 1. Fixed a bug in the loop model rest...
📁 lp:~maddevelopers/madgraph5/decay_calculator	Experimental	2011-12-13	192. Delete intermediate interactions thor...

New branches you create for MadGraph5 are **public** initially.

[Register a branch](#) ➔

➕ [Import a branch](#)

🔧 [Configure code hosting](#)

Ongoing developments

- Color-ordered recursion relations (J.A., Maltoni, Takaesu)
 - MadLoop5 (Hirshi, J.A.)
 - MadFKS5 (Zaro, Frederix)
 - MadGolem5 (Gonzalez-Netto et al.)
 - MadDM (Backovic, Kong, McCaskey)
 - MadDecay (Shen, J.A., Mattelaer)
- ➔ No difficulty to have multiple parallel developments thanks to high level of modularity + OO structure!

Thanks for listening!

- Development in MadGraph 5 is easy and fun!
- A big welcome to the newcomers in the game!
You will have a great time!

Backup slides

Diagram generation speed benchmarks

Full MadEvent subprocess directory output

Computer: Sony Vaio TZ laptop

Process	MADGRAPH 4	MADGRAPH 5	Subprocesses	Diagrams
$pp \rightarrow jjj$	29.0 s	25.8 s	34	307
$pp \rightarrow jjl^+l^-$	341 s	103 s	108	1216
$pp \rightarrow jjje^+e^-$	1150 s	134 s	141	9012
$u\bar{u} \rightarrow e^+e^-e^+e^-e^+e^-$	772 s	242 s	1	3474
$gg \rightarrow gggggg$	2788 s	1050 s	1	7245
$pp \rightarrow jj(W^+ \rightarrow l^+\nu_l)$	146 s	25.7 s	82	304
$pp \rightarrow t\bar{t} + \text{full decays}$	5640 s	15.7 s	27	45
$pp \rightarrow \tilde{q}/\tilde{g} \tilde{q}/\tilde{g}$	222 s	107 s	313	475
7 particle decay chain	383 s	13.9 s	1	6
$gg \rightarrow (\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)(\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)$	70 s	13.9 s	1	48
$pp \rightarrow (\tilde{g} \rightarrow jj\tilde{\chi}_1^0)(\tilde{g} \rightarrow jj\tilde{\chi}_1^0)$	>> 10^7 years	251 s	144	11008
$gg \rightarrow (\tilde{g} \rightarrow u(\tilde{u}_l \rightarrow \bar{u}(\tilde{\chi}_2^0 \rightarrow Z\tilde{\chi}_1^0)))(\tilde{g} \rightarrow u\bar{d}\tilde{\chi}_1^-)$				

Diagram generation speed benchmarks

Full MadEvent subprocess directory output

Computer: Sony Vaio TZ laptop

Process	MADGRAPH 4	MADGRAPH 5	Subprocesses	Diagrams
$pp \rightarrow jjj$	29.0 s	25.8 s	34	307
$pp \rightarrow jjl^+l^-$	341 s	103 s	108	1216
$pp \rightarrow jjje^+e^-$	1150 s	134 s	141	9012
$u\bar{u} \rightarrow e^+e^-e^+e^-e^+e^-$	772 s	242 s	1	3474
$gg \rightarrow gggggg$	2788 s	1050 s	1	7245
$pp \rightarrow jj(W^+ \rightarrow l^+\nu_l)$	146 s	25.7 s	82	304
$pp \rightarrow t\bar{t} + \text{full decays}$	5640 s	15.7 s	27	45
$pp \rightarrow \tilde{q}/\tilde{g} \tilde{q}/\tilde{g}$	222 s	107 s	313	475
7 particle decay chain	383 s	13.9 s	1	6
$gg \rightarrow (\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)(\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)$	70 s	13.9 s	1	48
$pp \rightarrow (\tilde{g} \rightarrow jj\tilde{\chi}_1^0)(\tilde{g} \rightarrow jj\tilde{\chi}_1^0)$	>> 10^7 years	251 s	144	11008
$gg \rightarrow (\tilde{g} \rightarrow u(\tilde{u}_l \rightarrow \bar{u}(\tilde{\chi}_2^0 \rightarrow Z\tilde{\chi}_1^0)))(\tilde{g} \rightarrow u\tilde{d}\tilde{\chi}_1^-)$				

Diagram generation speed benchmarks

Full MadEvent subprocess directory output

Computer: Sony Vaio TZ laptop

Process	MADGRAPH 4	MADGRAPH 5	Subprocesses	Diagrams
$pp \rightarrow jjj$	29.0 s	25.8 s	34	307
$pp \rightarrow jjl^+l^-$	341 s	103 s	108	1216
$pp \rightarrow jjje^+e^-$	1150 s	134 s	141	9012
$u\bar{u} \rightarrow e^+e^-e^+e^-e^+e^-$	772 s	242 s	1	3474
$gg \rightarrow gggggg$	2788 s	1050 s	1	7245
$pp \rightarrow jj(W^+ \rightarrow l^+\nu_l)$	146 s	25.7 s	82	304
$pp \rightarrow t\bar{t} + \text{full decays}$	5640 s	15.7 s	27	45
$pp \rightarrow \tilde{q}/\tilde{g} \tilde{q}/\tilde{g}$	222 s	107 s	313	475
7 particle decay chain	383 s	13.9 s	1	6
$gg \rightarrow (\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)(\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)$	70 s	13.9 s	1	48
$pp \rightarrow (\tilde{g} \rightarrow jj\tilde{\chi}_1^0)(\tilde{g} \rightarrow jj\tilde{\chi}_1^0)$	>> 10 ⁷ years	251 s	144	11008
$gg \rightarrow (\tilde{g} \rightarrow u(\tilde{u}_l \rightarrow \bar{u}(\tilde{\chi}_2^0 \rightarrow Z\tilde{\chi}_1^0)))(\tilde{g} \rightarrow u\tilde{d}\tilde{\chi}_1^-)$				

Diagram generation speed benchmarks

Full MadEvent subprocess directory output

Computer: Sony Vaio TZ laptop

Process	MADGRAPH 4	MADGRAPH 5	Subprocesses	Diagrams
$pp \rightarrow jjj$	29.0 s	25.8 s	34	307
$pp \rightarrow jjl^+l^-$	341 s	103 s	108	1216
$pp \rightarrow jjje^+e^-$	1150 s	134 s	141	9012
$u\bar{u} \rightarrow e^+e^-e^+e^-e^+e^-$	772 s	242 s	1	3474
$gg \rightarrow gggggg$	2788 s	1050 s	1	7245
$pp \rightarrow jj(W^+ \rightarrow l^+\nu_l)$	146 s	25.7 s	82	304
$pp \rightarrow t\bar{t} + \text{full decays}$	5640 s	15.7 s	27	45
$pp \rightarrow \tilde{q}/\tilde{g} \tilde{q}/\tilde{g}$	222 s	107 s	313	475
7 particle decay chain	383 s	13.9 s	1	6
$gg \rightarrow (\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)(\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)$	70 s	13.9 s	1	48
$pp \rightarrow (\tilde{g} \rightarrow jj\tilde{\chi}_1^0)(\tilde{g} \rightarrow jj\tilde{\chi}_1^0)$	$\gg 10^7$ years	251 s	144	11008
$gg \rightarrow (\tilde{g} \rightarrow u(\tilde{u}_l \rightarrow \bar{u}(\tilde{\chi}_2^0 \rightarrow Z\tilde{\chi}_1^0)))(\tilde{g} \rightarrow u\tilde{d}\tilde{\chi}_1^-)$				

Diagram generation speed benchmarks

Full MadEvent subprocess directory output

Computer: Sony Vaio TZ laptop

Process	MADGRAPH 4	MADGRAPH 5	Subprocesses	Diagrams
$pp \rightarrow jjj$	29.0 s	25.8 s	34	307
$pp \rightarrow jjl^+l^-$	341 s	103 s	108	1216
$pp \rightarrow jjje^+e^-$	1150 s	134 s	141	9012
$u\bar{u} \rightarrow e^+e^-e^+e^-e^+e^-$	772 s	242 s	1	3474
$gg \rightarrow ggggg$	2788 s	1050 s	1	7245
$pp \rightarrow jj(W^+ \rightarrow l^+\nu_l)$	146 s	25.7 s	82	304
$pp \rightarrow t\bar{t} + \text{full decays}$	5640 s	15.7 s	27	45
$pp \rightarrow \tilde{q}/\tilde{g} \tilde{q}/\tilde{g}$	222 s	107 s	313	475
7 particle decay chain	383 s	13.9 s	1	6
$gg \rightarrow (\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)(\tilde{g} \rightarrow u\bar{u}\tilde{\chi}_1^0)$	70 s	13.9 s	1	48
$pp \rightarrow (\tilde{g} \rightarrow jj\tilde{\chi}_1^0)(\tilde{g} \rightarrow jj\tilde{\chi}_1^0)$	$\gg 10^7$ years	251 s	144	11008
$gg \rightarrow (\tilde{g} \rightarrow u(\tilde{u}_l \rightarrow \bar{u}(\tilde{\chi}_2^0 \rightarrow Z\tilde{\chi}_1^0)))(\tilde{g} \rightarrow u\bar{d}\tilde{\chi}_1^-)$				

Super-fast decay chains allow completely new types of processes!

Event generation speed benchmarks

Generation of 10,000 unweighted events

Computer: Sony Vaio TZ laptop / *128-core cluster

Process	Subproc. dirs.		Channels		Directory size		Event gen. time	
	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5
$pp \rightarrow W^+ j$	6	2	12	4	79 MB	35 MB	3:15 min	1:55 min
$pp \rightarrow W^+ jj$	41	4	138	24	438 MB	64 MB	9:15 min	4:19 min
$pp \rightarrow W^+ jjj$	73	5	1164	120	842 MB	110 MB	21:41 min*	8:14 min*
$pp \rightarrow W^+ jjjj$	296	7	15029	609	3.8 GB	352 MB	2:54 h*	46:50 min*
$pp \rightarrow W^+ jjjjj$	-	8	-	2976	-	1.5 GB	-	11:39 h*
$pp \rightarrow l^+ l^- j$	12	2	48	8	149 MB	44 MB	21:46 min	3:00 min
$pp \rightarrow l^+ l^- jj$	54	4	586	48	612 MB	83 MB	2:40 h	11:52 min
$pp \rightarrow l^+ l^- jjj$	86	5	5408	240	1.2 GB	151 MB	49:18 min*	16:38 min*
$pp \rightarrow l^+ l^- jjjj$	235	7	65472	1218	5.3 GB	662 MB	7:16 h*	2:45 h*
$pp \rightarrow t\bar{t}$	3	2	5	3	49 MB	39 MB	2:39 min	1:55 min
$pp \rightarrow t\bar{t} j$	7	3	45	17	97 MB	56 MB	10:24 min	3:52 min
$pp \rightarrow t\bar{t} jj$	22	5	417	103	274 MB	98 MB	1:50 h	32:37 min
$pp \rightarrow t\bar{t} jjj$	34	6	3816	545	620 MB	209 MB	2:45 h*	23:15 min*

Event generation speed benchmarks

Generation of 10,000 unweighted events

Computer: Sony Vaio TZ laptop / *128-core cluster

Process	Subproc. dirs.		Channels		Directory size		Event gen. time	
	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5
$pp \rightarrow W^+ j$	6	2	12	4	79 MB	35 MB	3:15 min	1:55 min
$pp \rightarrow W^+ jj$	41	4	138	24	438 MB	64 MB	9:15 min	4:19 min
$pp \rightarrow W^+ jjj$	73	5	1164	120	842 MB	110 MB	21:41 min*	8:14 min*
$pp \rightarrow W^+ jjjj$	296	7	15029	609	3.8 GB	352 MB	2:54 h*	46:50 min*
$pp \rightarrow W^+ jjjjj$	-	8	-	2976	-	1.5 GB	-	11:39 h*
$pp \rightarrow l^+ l^- j$	12	2	48	8	149 MB	44 MB	21:46 min	3:00 min
$pp \rightarrow l^+ l^- jj$	54	4	586	48	612 MB	83 MB	2:40 h	11:52 min
$pp \rightarrow l^+ l^- jjj$	86	5	5408	240	1.2 GB	151 MB	49:18 min*	16:38 min*
$pp \rightarrow l^+ l^- jjjj$	235	7	65472	1218	5.3 GB	662 MB	7:16 h*	2:45 h*
$pp \rightarrow t\bar{t}$	3	2	5	3	49 MB	39 MB	2:39 min	1:55 min
$pp \rightarrow t\bar{t} j$	7	3	45	17	97 MB	56 MB	10:24 min	3:52 min
$pp \rightarrow t\bar{t} jj$	22	5	417	103	274 MB	98 MB	1:50 h	32:37 min
$pp \rightarrow t\bar{t} jjj$	34	6	3816	545	620 MB	209 MB	2:45 h*	23:15 min*

Event generation speed benchmarks

Generation of 10,000 unweighted events

Computer: Sony Vaio TZ laptop / *128-core cluster

Process	Subproc. dirs.		Channels		Directory size		Event gen. time	
	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5	MG 4	MG 5
$pp \rightarrow W^+ j$	6	2	12	4	79 MB	35 MB	3:15 min	1:55 min
$pp \rightarrow W^+ jj$	41	4	138	24	438 MB	64 MB	9:15 min	4:19 min
$pp \rightarrow W^+ jjj$	73	5	1164	120	842 MB	110 MB	21:41 min*	8:14 min*
$pp \rightarrow W^+ jjjj$	296	7	15029	609	3.8 GB	352 MB	2:54 h*	46:50 min*
$pp \rightarrow W^+ jjjjj$	-	8	-	2976	-	1.5 GB	-	11:39 h*
$pp \rightarrow l^+ l^- j$	12	2	48	8	149 MB	44 MB	21:46 min	3:00 min
$pp \rightarrow l^+ l^- jj$	54	4	586	48	612 MB	83 MB	2:40 h	11:52 min
$pp \rightarrow l^+ l^- jjj$	86	5	5408	240	1.2 GB	151 MB	49:18 min*	16:38 min*
$pp \rightarrow l^+ l^- jjjj$	235	7	65472	1218	5.3 GB	662 MB	7:16 h*	2:45 h*
$pp \rightarrow t\bar{t}$	3	2	5	3	49 MB	39 MB	2:39 min	1:55 min
$pp \rightarrow t\bar{t} j$	7	3	45	17	97 MB	56 MB	10:24 min	3:52 min
$pp \rightarrow t\bar{t} jj$	22	5	417	103	274 MB	98 MB	1:50 h	32:37 min
$pp \rightarrow t\bar{t} jjj$	34	6	3816	545	620 MB	209 MB	2:45 h*	23:15 min*

No problem running processes like $t\bar{t} + 0,1,2j$ on a laptop!