

An Overview of GAP

Max Neunhöffer



University of St Andrews

St Andrews, 17.6.2009

What is GAP?

- “Groups, Algorithms and Programming”

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.
- Method selection based on current type of all arguments.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.
- Method selection based on current type of all arguments.
- GAP is a platform to implement mathematical (mostly discrete) algorithms (→ currently ~ 60 packages)

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.
- Method selection based on current type of all arguments.
- GAP is a platform to implement mathematical (mostly discrete) algorithms (→ currently ~ 60 packages)
- GAP has a sizable user and developer community.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.
- Method selection based on current type of all arguments.
- GAP is a platform to implement mathematical (mostly discrete) algorithms (→ currently ~ 60 packages)
- GAP has a sizable user and developer community.
- GAP contains large collections of math. knowledge.

What is GAP?

- “Groups, Algorithms and Programming”
- A CAS directed at computational group and representation theory.
- Consists of a small(ish) kernel written in C ...
- ... an interpreted computer language called GAP ...
- ... and a library of algorithms written in GAP.
- The language has rather unique object oriented features invented to model mathematical objects:
 - objects learn during their lifetime and
 - change their type in the process.
- Method selection based on current type of all arguments.
- GAP is a platform to implement mathematical (mostly discrete) algorithms (→ currently ~ 60 packages)
- GAP has a sizable user and developer community.
- GAP contains large collections of math. knowledge.
- GAP is open source (GPL).

What is GAP not?

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

- A general purpose computer algebra system (like Maple, Mathematica or MuPAD)

What is GAP not?

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

- A general purpose computer algebra system (like Maple, Mathematica or MuPAD)
- A numerical algorithms toolbox/system (like Matlab)

What is GAP not?

- A general purpose computer algebra system (like Maple, Mathematica or MuPAD)
- A numerical algorithms toolbox/system (like Matlab)
- A push-button solution for arbitrary problems involving groups.

Algorithms and data available in GAP

- permutation groups,

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))

Algorithms and data available in GAP

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))
- interfaces to other systems (→ [SCIENCE](#) project),

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))
- interfaces to other systems (→ SCIENCE project),
- small groups (→ [SmallGroups](#) package)

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))
- interfaces to other systems (→ [SCIENCE](#) project),
- small groups (→ [SmallGroups](#) package)
- crystallographic groups (→ [CRYST](#) and [CRYSTCAT](#))

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))
- interfaces to other systems (→ [SCIENCE](#) project),
- small groups (→ [SmallGroups](#) package)
- crystallographic groups (→ [CRYST](#) and [CRYSTCAT](#))
- character tables (→ [ctbllib](#) package)

Algorithms and data available in GAP

- permutation groups,
- other representations for groups (matrix, polycyclic, presentations),
- semigroups/monoids
- linear algebra (in particular over finite fields),
- combinatorics (graphs → [GRAPE](#), codes → [GUAVA](#), designs → [DESIGN](#), cohomology → [cohomolo](#), [HAP](#))
- interfaces to other systems (→ SCIENCE project),
- small groups (→ [SmallGroups](#) package)
- crystallographic groups (→ [CRYST](#) and [CRYSTCAT](#))
- character tables (→ [ctbllib](#) package)
- access to the WWW Atlas of group representations (→ [AtlasRep](#) package).

What one can do with GAP

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

- **Rapidly** implement new algorithms for discrete problems involving (semi-) **groups**.

What one can do with GAP

What is GAP?

What is it not?

Things available in
GAP

What we can
provide

- **Rapidly** implement new algorithms for discrete problems involving (semi-) **groups**.
- Provide a **platform** for this with **access** to a lot of previously assembled **algorithms and data**.